



May
June
1994

Volume 5
Number 5

The First Apple IIGS[®] Magazine + Disk Publication!

It's Movie Time!

Use FLI Convert to View IBM PC .FLI Animations With Your IIGS!



While Enjoying Our Feature Attraction Why Don't You . . .

BUY! . . . Stuff From Our Advertisers!

LEARN! . . . More About Programming the IIGS!

FIND OUT! . . . How to Safely Shop Through the Mail!

FALL IN LOVE! . . . With Joe as he Writes About the List Manager!

READ SOME REVIEWS!

DiscQuest, Spectrum, Salvation: Bakkup v2.0
MS-DOS File Utilities

Well, the taxes are finally done, and I'm happy to say that I didn't have to pay that much. (Less than \$100.) Tax time is always a time when I have to take a long look at the business of supporting the IIGS and ask myself, "is it worth it?" Well, last year *was* worth it, and this year looks like it will be even more worthwhile. In other words, business is good. So, to those of you that have put your money where your mouth is by supporting us, I just want to say "Thank You!"

Of course, when I say that "business is good," there is a hidden message that "it could always be better!" After all, the more subscriptions we have, the longer we can stay in the business of supporting the IIGS. So, if you have a friend with a IIGS, why not sit them down for a heart to heart talk and try to get them to subscribe? And, if they already subscribe, give them a big wet kiss for me.

Newton Update

And, in case you were wondering how our Newton development efforts were going . . . they aren't really going anywhere at this point. We did a couple of shareware products, released them, and then waited for some feedback. The trouble is that, from a shareware point of view at least, the Newton market seems to be even deader than the Apple II market! We've gotten only about seven shareware payments over the past six months, and we've gotten even fewer letters and e-mail about the products.

Of course, it *is* possible that our stuff stinks, but the feedback we *have* gotten on it doesn't seem to support that theory. I guess the most likely explanation is that the Newton market just hasn't reached "critical mass" yet. At least, I hope that's what the problem is!

Speaking of that "critical mass," I finally had a chance to see the new Newton model, the MessagePad 110. Once again, Apple's second-generation unit is what the first generation *should* have been. If you weren't impressed with the first Newton model, you really should check out the latest one. It has improved handwriting recognition (really!), more memory (thank goodness!), a protective cover (just like the early models that Apple showed before they released the first Newton), longer battery life (hurray!), and it costs less (those scum)! If *this* model doesn't sell like hot-cakes, something is truly screwy in the universe. Best of all, you can

upgrade your original Newton to have most of these new features (everything except the new case) for \$99 (plus tax and shipping). For more information on upgrading your Newton, call 1-800-SOS-APPL. (True story: I just got off the phone with a Newton owner that wanted to know what the phone number was for the S.O.S. Apple program.)

Do You Write IIGS Shareware?

If you write shareware for the IIGS, my story about the Newton probably sounds familiar. So, what can you do? Well, for one thing, you could try selling your programs to us. Of course, you won't get rich, but I can almost guarantee that we will pay you more than you will *ever* receive in shareware payments. So, think twice before you release that program as shareware, and give us a call.

Burned Again?

So, let's see, Apple's burned me on the original IIGS, the Mac LC, and now, the Newton. Of course, I deserved it. When I bought each of those products, I did so knowing exactly how Apple operates. So, the question now is, am I going to buy a Power Macintosh? Well, yeah—but, not yet. I'm going to wait for the second or third generation, just like I should have done with the Newton. However, I must say that I've had a *tough* time resisting these new machines! They really are amazingly fast, and the prices are incredible, especially for Apple products.

Of course, if Apple had the sense to put out an Apple IIGS emulator for the Power Macintosh, I'd buy one in a heartbeat. But, over the last few months, sources at Apple have been quoted as saying that they have *no* intention of creating such an emulator. Come on Apple, wake up! Just think of all the schools out there that bought IIe emulators for the Mac LC! Don't you realize that those same institutions would probably kill for the ability to run *all* their Apple II, IIGS, Mac and PC software on one box? If you don't want to do it, give *us* the contract! Giving the Power Macintosh the ability to run Apple IIe and IIGS software would *finally* get Apple Computer, Inc. "off the hook" with Apple II owners that feel abandoned, and it would give those of us that have been supporting the Apple II a new market to start casting our nets into.

Come on Apple, it's *not* to late to win back the loyalty of the millions of Apple II owners out there. All you have to do is make the effort!

And Finally

Just so you don't think that Professor Gumby is making this stuff up, Applied Engineering *is* out of business. At this point, I have absolutely no idea how or why it happened (the phones are all disconnected and former employees don't seem to want to talk about it), but I am trying to find out. In the meantime, the question for Apple II owners is "what about all the stuff I bought from them?" Well, I'm afraid that I don't have a concrete answer for that one either. I know that several companies are trying to buy all of their old Apple II stock, but, thus far, no one has come forward and offered to service or support that equipment. (A few former A.E. employees have promised to provide what help they can via the online services, but none that I've seen have offered to repair stuff.)

So, what *did* happen to A.E.? Well, as I said, I really don't know. However, based on the feedback I've gotten from my Macintosh-owning friends, the Mac products that A.E. made were underpowered and overpriced. Not a good combination in a market that is as cut-throat as the Mac market. On top of that, A.E. did an *excellent* job of alienating their once-loyal Apple II customer base and their Amiga products seem to have been a bust. So, in hindsight, I guess it was a miracle that they hung on as long as they did.

Over the years, I have really learned a lot about how *not* to run a business from Applied Engineering. I suppose that the final lesson to learn here is: If you are an Apple II vendor and you *must* branch out into other markets to stay in business, don't forget about your current customers! Unless, like Apple, you have about a billion dollars in the bank.

New Hours

If you read our last issue very closely, you may have noticed that we have new hours of operation. We now open at 9 a.m. and close at 5 p.m. (all times Eastern), Monday through Friday. The reason for this is that everyone else goes home at 5, and I was tired of sitting here all by myself for an hour. (Besides, my doctor told me that I had to do *something* to reduce my stress levels, and killing everyone here at the office would probably have been counter-productive.)

CONTENTS

FEATURES

Programming the IIGS - Part II: Programming the <i>IIGS</i>	6
Mr. Priceguide Looks at Hard Disks.....	10

PROGRAMS

FLI Convert.....	16
More Sound.....	40
Working With the Toolbox Part 11: The List Manager.....	41

DEPARTMENTS

Writer's Block.....	inside front cover
Letters.....	3
Moving.....	21
Rumors, Wishes & Blatant Lies.....	23
How to Use Your <i>GS+</i> Disk.....	24
How to Get System 6.0.1.....	26
Problems.....	32
What's New.....	39
Warranty Disclaimer and Copyrights.....	46
Buying Ad Space.....	47
Glossary.....	48
<i>GS+</i> Ordering Information.....	inside back cover
<i>GS+</i> Back Issue Information.....	on disk

REVIEWS

• DiscQuest.....	28
• MS-DOS File Utilities	31
• Salvation: Bakkup v2.0.....	33
• Spectrum.....	35

Products marked with a bullet (•) were review copies provided by the publisher.

ADVERTISERS

Alltech Electronics.....	22
Digisoft Innovations.....	35, 37
Other World Computing.....	12, 13
Parsons Engineering.....	inside back cover
Seven Hills Software.....	20
Sequential Systems.....	30, 47
Shreve Systems.....	back cover
Terry's Tech Service.....	15
Westcode Software.....	27

GS+

Magazine

May-June 1994
Volume 5, Number 5

Director

STEVEN W. "FATTY FATTY 2x4" DISBROW

Grip, Best Boy, Caterer, Music, Gaffer
JOSEF W. "GONZO" WANKERL

Idiot Wrangler

MICHELLE B. "EESHUL" RIBARIC

Stunts, Camera, Sound Effects
ROBERT A. "BOB" RIBARIC

On The Cover

The entire *GS+* film crew screening their latest release.

GS+ Magazine and its companion program diskette are copyright © 1994 by EGO Systems. No part of the magazine or its companion program diskette may be reproduced without the written permission of EGO Systems. The programs on the companion program diskette are *not* public domain or shareware!

GS+ is a trademark of EGO Systems.

All references to either Apple or third party products are trademarked and should be so noted.

GS+ Magazine is an independent publication, not affiliated in any way with Apple Computer, Inc.

Opinions expressed in this publication are those of the individual authors and do not necessarily represent those of *GS+* Magazine or EGO Systems.

Subscription rates - Magazine only:

1/2 year (3 issues) - \$8

1 year (6 issues) - \$15

Subscription rates - Magazine w/Disk:

1/2 year (3 issues) - \$20

1 year (6 issues) - \$36

Tennessee residents add 7.75% sales tax.

Add \$1.50 per issue if you want First-Class delivery.

Canadian and Mexican orders add \$1.50 per issue.

Other foreign orders add \$1.50 per issue for surface delivery or \$5 per issue for Air Mail.

Send orders, ads, inquiries, and address changes to:

GS+ Magazine

P.O. Box 15366

Chattanooga, TN 37415-0366

Or phone Monday-Friday 9 am-5 pm Eastern Time:

(800) 662-3634 (orders only)

(615) 843-3988 (tech support)

(615) 843-3986 (fax)

If you have a submission for *GS+* Magazine, send it to:

GS+ Submissions

P.O. Box 15366

Chattanooga, TN 37415-0366

GS+ Magazine can also be contacted on these online services:

America Online: send mail to GSPlusDiz

or visit our online area by using the keyword GSMAG.

Delphi: GSPlusDiz

GEnie: JWankerl

InterNet: gsplusdiz@aol.com

GS+ Magazine is published bimonthly by:

EGO Systems

7918 Cove Ridge Road

Hixson, TN 37343-1808

(DO NOT SEND MAIL TO THIS ADDRESS—USE FOR UPS AND OVERNIGHT DELIVERIES ONLY!)

GS+ Magazine is produced on the Apple IIGS using the following products:

EGOed - Text Editing

GraphicWriter III - Interior Page Layouts

AppleWorks GS - Cover Layouts

Dream Grafix & Platinum Paint - Screen Shots

Proof pages are printed on an Apple LaserWriter IINT.

We use a Macintosh LC III as a file server because we have to.

Letters

[Editor's Note: Starting this issue, we are going to print the InterNet e-mail address (if available) for each of our letter writers. Why are we doing this? The main reason is that if we can't help you with a problem here on the letters page, maybe one of our readers can, and an e-mailed response will get to you much faster than the next issue of this magazine will. Also, as Apple IIGS owners become fewer in number, this will give everyone out there an easy (and non-threatening) way to make contact with other Apple IIGS owners. But, if you don't want your e-mail address printed, please be sure to say so in your letters! If you *do* want it printed, be sure to include it!]

Dear Joe,

I have been playing Kabloolie since I received my November-December [1993 - V5.N2] edition of *GS+* Magazine and have achieved some reasonably respectable scores. (Beginner - 19, Intermediate - 128, and Expert - 353.)

I can't understand why you think Kabloolie is so addictive. I haven't had any problem leaving it alone. I can quit any time after 200 or 300 games. (My longest play session was 342 consecutive games, but that included a lot of "bust" conditions on the first try.)

Kabloolie is a remarkable game and very similar to Minesweeper on Windows. I had been praying that someone would port it to the IIGS or write a similar game before I succumbed and bought a machine that runs Windows. Whew! It was close. Thank you.

Seriously, however, not only is Kabloolie one of the better IIGS games, it is an extremely good tool for honing the analytic skills involved in problem solving. Teachers, take note!

John R. Engberg
Bowie, MD

Thanks for the letter John. Glad to hear that you liked Kabloolie! We've gotten lots of very positive feedback on this game (which, by the way, was written by Curt Clifton—Joe just "tested" it for about 400 hours), and we are trying to come up with more game ideas that will be as popular. If anyone has any ideas for games, let us hear them!

By the way, we have gotten two problem reports on Kabloolie! The first came from a gentleman that was having trouble

*with Kabloolie! automatically uncovering squares when he would single-click on them. Apparently, his mouse button is "bouncing" and reporting two clicks when it should only be reporting one. The only solution here is to either clean your mouse (see "IIGS Maintenance—Part 1: The Mouse and Keyboard" in *GS+ V5.N2*) or buy a new one.*

The second problem was reported by an extremely patient reader that simply couldn't get Kabloolie! to run. After several phone calls, we finally determined that Kabloolie! would not run because of the 5.25-inch disks connected to her system. Thus far, no one else has reported a similar problem, but, if anyone else out there is having trouble running Kabloolie!, try disconnecting your 5.25-inch drives and see if it fixes things.

Diz

Dear *GS+*:

... I would like to run a PaintWorks animation from my own [AppleSoft] BASIC program. The FTA provided such a program, however the program does not return [properly] to BASIC.

As I look back over issues of such magazines as *A+InCider*, *Nibble*, et., I have found that there are many other people that would like to have this kind of program. I do not have a modem. I have written to such people as Big Red [Computer Club] for help, with no luck.

If you know of someone who can help me with this problem, please let me know.

Peggy Hammer
Glenwood Springs, CO

Sorry Peggy, but I don't know of anything like that. Perhaps one of our readers can help. How about it folks?

Diz

G'day Peoples:

Just a quick letter to let you all know what a terrific magazine and disk that you're producing!

With the release of Balloon [*GS+ V5.N2*] and recently EGOed version 2.0 [*GS+ V5.N3*], you've justified the cost of the mag/disk subscription for me! Add to that Cool Cursor [*GS+ V5.N1*], CD-ROaM [*GS+ V5.N2*] and other really neat (and useful) utilities then this subscription

is really worth it! I've been waiting ages for a way to exchange files (with formatting) between my IIGS and the DOS boxes and Macs at work, now with the Rich Text Format (RTF) included [in EGOed v2.0] it's heaven... Please keep it up!

A small suggestion on how to improve Balloon. Include some error indication, specifically if trying to unshrink a file to a full volume, an error dialog stating the error would be helpful (with the option to abort the extraction)...

Have you ever thought of including all the text of the magazine as a text file on the disk, or making available all the text from all the magazine on a series of disks, similar to Resource Central On Disk?...

Anyhow, keep up the good work as we appreciate the hard work one and all there are putting in. Cheers!

Stephen C. Davidson
Victoria, Australia
scd@ausom.oz.au

Thanks for the comments and suggestions Stephen! We'll look into that Balloon suggestion and see about getting it fixed in the next version. As for putting the text of our sold-out magazines on disk, that's something that I've been considering for a long time now. There have been two main obstacles to doing it though: Lack of time, and fear of piracy. Sadly, I always seem to be short on time, and the piracy situation in the Apple IIGS market does not seem to have improved any. So, releasing our back issues in this way could be a big mistake for us.

Diz

Dear *GS+*:

... I would like to congratulate you on EGOed v2.0. The ability to load and save Rich Text Format files is a great improvement. It makes EGOed the best IIGS editor available and it has made the Apple IIGS even more versatile. Things I would like to see new in EGOed are:

- 1) EGOed as a stand-alone application.
- 2) [Page numbers.]
- 3) More than one ruler per document.
- 4) [The ability to load Medley files and keep the font information.]

I suppose not many people own Medley and this would not be much requested but it would be a great help to me. I use

Medley as my main word processor but it has two major faults: Maximum font size is limited to 48 points and it does not use the system clipboard. The only way I can import the files into other word processors is to save as a text file and I thus have to reformat the document. I have never studied the workings of IIGS word processors and so find it strange that the maximum font size is limited to 48 points. But, [with] the Medley paint tool, you can have sizes over that—why is it different for the word processor? . . .

One item I would like to see more of in *GS+* Magazine is HyperCard/HyperTalk programming. The stack you did to print out names and addresses in *GS+* V3.N1 was one of the best things I have seen for HyperCard IIGS. I have changed and adapted it for my own use and have found it invaluable. You have done nothing for over two years on HyperCard IIGS programming and as *GS+* Magazine deals with programming and HyperTalk is a programming language I thought you might have done more on that . . .

Christopher Beckett
Shropshire, England

*Hmmm, well, you certainly do have a point there. I kind of dropped HyperCard IIGS like a hot potato after I moved my customer database to the Macintosh. (Please don't hate me! I had to do it! HyperCard was destroying two or three customer records a day!) But, another reason we've not done any other HyperCard stuff is that, well, no one else has asked for it. However, you are right that HyperTalk is a IIGS programming language, and that is one of the things we are supposed to cover. I've got some half-finished HyperTalk projects lying around here somewhere . . . What about everyone else? Would you like to see some HyperCard or HyperStudio projects in *GS+* Magazine?*

Getting back to your EGOed questions . . . 1) I have been considering writing a stand-alone word processor, but it would be a completely separate project from EGOed. 2) Page numbers, eh? Maybe. 3) Multiple rulers really aren't possible in EGOed or any other editor that is based on the TextEdit tool set. However, if item number 1 comes to pass, it will have multiple rulers, page numbers and all those other neat things. 4) The Medley format is documented, so it should be possible to do a translator. But, you are right, no one else has asked for such a thing. I'll look into it though.

As for why Medley has a 48 point font size limitation, this is just a guess, but I

think its because back when Medley (and AppleWorks GS) was first written, the IIGS Font Manager was a little flaky when it came to working with fonts larger than 48 points. So, the authors of the software decided to be safe rather than sorry and put in the limitation. As for why the paint module in Medley doesn't have this limitation (again this is a guess), the paint module doesn't actually have to manipulate the characters you type after you type them. All the paint module has to do is ask the Font Manager to generate the characters at the requested size, and then spit them out onto the screen. After that, they are treated as bit-mapped graphic images, and not as text. So, the Font Manager doesn't have to mess with them any more, so it can't mess them up.

Diz

Dear Diz:

. . . I have run into one or two problems, at least they are problems to me:

1. Does EGOed v2.0 replace EGOed lite, or is EGOed lite a distinct program with either separate or complimentary applications?

2. I understand how to change right and left margins on an EGOed document, but how do I adjust the top and bottom margins? There is no "View Header" or "View Footer" in the EGOed Document menu as there is in AppleWorks GS.

3. I think that the addition of color to an EGOed document is great, but I have not been able to print in color. I have an ImageWriter II printer with a color ribbon, and it works fine with Print Shop, but I have not been able to print the color that shows up on the screen on an EGOed document.

Thank you for *GS+* Magazine. Please count on me as a regular subscriber from now on.

Gordon M. Sauls
Livonia, MI

Boy, I didn't realize we had so many EGOed questions out there! (Don't forget to read that documentation file boys and girls!) Anyway, here are some answers for you Gordon:

1. EGOed and EGOed lite are separate programs. EGOed lite was created for those folks that wanted the convenience of an NDA text editor, without taking up all disk space that the regular EGOed does. EGOed lite has a much smaller feature set than EGOed, but a lot of people use it

due to limited memory availability and/or disk space.

2. Well, the truth is that you can't change the top and bottom margins in EGOed. Sorry about that. But, it is definitely on my "wish list" of features to add.

3. When I read your letter, I realized that I had never even tried printing in color with EGOed, so I gave it a try and . . . it worked great! All I can guess is that you have not told EGOed that you want to print in color. To do this, pick Print from the EGOed File menu, and then, when the print dialog appears, pick "Color" from the Chroma menu (it's near the bottom of the dialog). This should let you print in color. (Note that you must have a color ribbon installed before you pick Print from the File menu, or you will not be able to select color printing.) This trick should work from any desktop program that is supposed to print in color.

Diz

Dear Diz and Co.:

. . . While looking through your back issues, I found a review of "Sinbad and the Throne of the Falcon" from CinemaWare. Where did you get this? I have searched high and low for this game (and other CinemaWare stuff) and have not been able to find them anywhere! . . .

Colin Williamson
Elizabethtown, PA
Colin.Williamson@f213.n270.z1

Well, actually, I got Sinbad and the Throne of the Falcon as a birthday gift from a friend that worked for CinemaWare at that time. They had just finished it and were about to start shipping (which is why we ran the review), when the company went under.

*After that, Quality Computers (800-777-3642) supposedly bought just about all of CinemaWare's IIGS stock. However, I have seen tons of their old IIGS stuff at various software liquidators in Atlanta Georgia. But, I've never seen (or heard of) a copy of the IIGS version of Sinbad and the Throne of the Falcon other than the one I have here. I guess that it's possible that I have the only one in existence! Does anyone else out there have a copy? If you do, you could probably sell it in a flash in the *GS+* Classifieds or online! (Sorry, but my copy isn't for sale.)*

Diz

Greetings from Wilmington, NC.

You know, I get your magazine every other month and usually what I do is check out the programs on the disk; see if there is anything useful to me (which, by the way, there usually is), then I'll read the articles that currently fascinate me and then throw it in a special pile, consisting of only *GS+* mags, to sit until I need to refer to it again for some particular reason.

Well, just recently I became fascinated with Object Oriented Programming (OOP). I have had a couple of weeks of C++ in a class here at University of North Carolina at Wilmington, so I have done a small, very small, amount of OOP. Last night a thought struck me, (don't worry it didn't hurt) "Didn't *GS+* do an article or two on OOP?". Then I began searching through my special pile, you know the one! And finally came across the September/October 1993 edition with the OOP article. WOW! So, me being who I am, read the article this morning and was greatly enlightened. Thanks! Like I said, I had done some C++ OOP programming but now that I look back at it, I didn't understand what the heck I was doing. Objects just went in one eye and out the other. Your article, "An Introduction to Object Oriented Programming" truly explained it all. (Well, not all.) I went back this afternoon and re-examined my old C++ programs and, although they are quite simple, they finally became clear. I understood them and even added some enchantments. WOW! (Again.) This OOP stuff is great!

Did I have a point? Why yes! Here it comes now: Would you, could you, might you possibly do some more articles on OOP? I would greatly appreciate it.

Point made.

I hope all of your Tennessee days are filled with sunshine.

Craig N. Caroon
Wilmington, NC
caroon@sol.cms.uncwil.edu

Actually Craig, the weather around here has been really great the last week or so. Thanks for the help!

What? Oh, yeah, the OOP stuff. Well, I am working on a new OOP application for the magazine, which should be in the next issue. Along with it will be a pretty much complete rewrite of the OOP modules I provided a couple of issues back and an article on how these new modules can be used to quickly (very quickly) put together a IIGS application.

Beyond that, I'm open for suggestions as to ideas for OOP topics.

Diz

Diz:

I just bought a Mac Quadra 610 and was wondering if it is possible to share my SCSI hard drive and CD-ROM with both my IIGS and Mac so that both computers could access them at the same time.

Michael Ash
Tomah, WI
MichaelA50@aol.com

*Yep. All you have to do is hook the hard disk and CD-ROM up to the Mac, and share the files on them using the Mac's Personal AppleShare. (Details on this can be found in your Macintosh user's manual [look under "file sharing" in the index] and in Joe's "Apple (Jive) Talkin" article from *GS+ V5.N1.*) Of course, you won't be able to boot your IIGS from any of the devices that are shared this way, but you will be able to access the files on them just as if they were on a drive hooked directly to your IIGS.*

Diz

Hi!

I just wanted to let you know that I am very pleased with *GS+* Magazine, especially with this (V5.N4) issue! You realized my suggestion to enhance LASERbeam, came up with a very useful program I never thought of before (What Is This?), did some fine reviews and dropped the ugly underlining on the cover page. And the rest of the content isn't worse either. Thank you!

Dirk Froehling
Dortmund, Germany
dirk@gaga.maschinenbau.uni-dortmund.de

Diz:

In *GS+ V4.N6* in the article describing System 6.0.1 ["System 6.0.1—For Users"], there was mention of a way to generate a PostScript file by having the Direct Connect printer control panel set to LaserWriter. How is this done? What exactly needs to be installed to get this to work and where does the file get saved? I have tried to install just the LaserWriter driver file since I do not have AppleTalk installed but this does not seem to work. More details on this "nifty feature" would be greatly appreciated.

Phil Bivens
Naperville, IL
PJBivens@aol.com

No problem Phil. It sounds like you have done just about everything necessary, but there is one additional step. But, for the sake of those coming in in the middle of this, I'll go over all of the steps again.

1) Run the System Software installer, choose the customize option, and install the LaserWriter update on your boot disk.

2) Reboot your IIGS.

3) Open the Direct Connect control panel, choose a port from the list of ports, and pick the LaserWriter item from the list of printer types.

4) Close the Direct Connect control panel, and run your text editor.

5) Open the file you want to save in PostScript format, and check the page setup to make sure it is the way you want it.

6) Select print from the File menu.

7) When the Print dialog appears, press and hold down the Command and "F" keys at the same time.

8) While still holding down Command-F, click the mouse (don't press the return key) on the "OK" button. You should then see a dialog that says "Creating PostScript file."

9) The PostScript file will be saved in the *:System:Drivers folder. It will have a name like "PostScript.GSxx" where xx is a number between 0 and 99. You can download this file to a PostScript printer using a utility like SendPS on the Mac or our own LASERbeam (from *GS+ V5.N4*) on the IIGS.

Diz

If you have a question, comment, or criticism about *GS+* Magazine, we want to hear it! Due to space limitations, letters may have to be edited and we can not answer every letter here in *GS+* Magazine.

If you want a personal reply, please include an e-mail address, a daytime phone number, or enclose a self-addressed, stamped envelope with your letter.

Please address all letters to:

GS+ Letters
P. O. Box 15366
Chattanooga, TN 37415-0366 *GS+*

Programming the IIGS - Part II: Programming the IIGS

By Steven W. Disbrow

The first article in this series looked at what you need to do and have in order to program the Apple IIGS. This article will look at the actual types of programs that you can create for your IIGS, and the things that set them apart from one another. In future articles, I'll look at each of these types of programs in detail. So without further delay, let's get to it!

Applications

Applications are what make or break a computer. If a computer doesn't have a good selection of useful applications available, no one will buy the computer, and it will die. On the IIGS, we've had our share of excellent applications (AppleWorks GS) and our share of turkeys (AppleWorks GS). But, over the years, the IIGS community has been fortunate to have a steady trickle (not quite a *flow*) of decent, useful applications. However, a lot of those applications weren't actually designed for the IIGS, they were just old 8-bit Apple II applications with a different name and a slapped-together desktop interface. Fortunately, that is *finally* changing, and we are beginning to see some excellent applications written specifically with the IIGS in mind. So, with this in mind, let's look at what you need to consider when writing a IIGS application.

The flow of a IIGS application is actually very simple. In the simplest terms possible, here's what you have to do:

- 1) Set up your environment
- 2) Wait for the user to make a request and respond to it
- 3) Clean up after yourself and return to the previous program

Now if you are a computer science student, you are probably thinking "That's just like every other program I've ever written!" For the most part, this is true. However, the details of each step are a bit more complicated than they appear on the surface. So, let's look at each one in detail.

Step 1 - Setting Up

For any program you write, on any computer, you will probably have to take some time at the beginning of the program to set up the environment you want your program to run in. This can be something as simple as setting the default values for menu choices, or something as

complex as building a table of square roots to speed up math calculations that will be done elsewhere in your program. (Note that these are just examples, you don't have to do either of these things to write a IIGS application!) When you are writing a IIGS desktop program however, you have a few extra steps that you have to go through to actually create the desktop environment that users are familiar with.

First of all, you have to start up any of the IIGS *tool sets* that you will need to use. These tool sets are just what the name implies—collections of routines that you can use to make a task easier. Collectively, these tool sets are known as the IIGS *Toolbox*. So, to stress the point (and throw in an analogy), when you need to drive a nail into a piece of wood, you borrow a hammer from your neighbor (or you go buy your own pre-fabricated hammer). When you want to work with a window on the IIGS screen, you use a routine from the Window Manager tool set. Of course, if you don't want to bother your neighbor (or drive to the hardware store), you can always tie a rock and a stick together and make your own hammer. And, if you don't want to use the Window Manager, you can always write your own window management routines. However, in both cases, your hand-made solution will probably be less effective than the factory-made one, and it will probably break too.

After you have your tools started (don't worry, we'll discuss how to do this in detail in a later article), you can use those tools to set up the user interface of your program. For example, you use the tools in the Menu Manager to set up and draw your application's menu bar, and you could use the tools in QuickDraw II to draw any welcoming screen that you want to display. All of this has to occur *before* the user is actually allowed to interact with the program. Once you have all of this set up work completed, you can move on to the next step.

Step 2 - Waiting Around

After everything is set up, a IIGS application goes immediately into what is called an *Event Loop*. Looking at that name for the first time, you might think that the Event Loop is something that *generates* events. However, just the opposite is true—an Event Loop's main

job is to simply sit there, and *wait* for something to happen. When something *does* happen, the Event Loop looks at the thing that happened (which is called an "event"), and dispatches control to a part of the application that is set up to handle that particular type of event. Once the event has been handled, control returns to the Event Loop, and it begins to patiently wait for the next event to occur.

While the concept of the Event Loop is simple, you must realize that this is where the "meat" of a IIGS application is. Just about every single task that an application can do is called from the application's main Event Loop. (Yes, an application can have more than one Event Loop. For example, a dialog window might have it's own Event Loop. For right now though, you only need to worry about the *concept* of an Event Loop, and not how many of them there may or may not be.)

Step 3 - Cleaning up

One of the events that an Event Loop must handle is responding to the selection of menu items. And, one of the menu items that every IIGS application must have is a "Quit" item. When the user selects this item, you know it's time to quit your Event Loop, clean up after yourself, and return to the previous program.

The only tricky part here is the "cleaning up." If your program grabbed any memory during its execution (i.e. in the Event Loop or one of the routines that it called), you must make sure that you dispose of that memory now, so that the rest of the system can access it. (We'll discuss the concepts of memory management in a future article.) Also, you must shut down any tools that you started in Step 1.

The final act of returning to the previous program can either be easy, or almost easy. If you are writing your application using a high-level language like C or Pascal, the compiler will take care of all of the details for you. If you are working in Assembly Language however, you will have to make a GS/OS Quit call to tell GS/OS that your program has finished executing and that control should return to the previous application.

So, as you can see, writing a IIGS application is fairly straight-forward. In

fact, it's quite a bit like the non-desktop programs you may have written before. The main difference is that instead of presenting one menu at a time (as in say, AppleWorks Classic), you present the user with all menus at once, and you have to be able to respond to any of the available actions at any given time.

Executable Files

Executable files ("EXE" files) *can be* just like applications. The only *required* physical difference is that the file types have to be different (applications have a file type of \$B3, and EXE files have a file type of \$B5). The main difference is *where* these types of files are intended to be used. An application is something that you can expect people to use from the Finder or some other user-oriented program launcher. An EXE file however, is intended to be a utility program that will be used from a development environment. As such, many EXE files are written under the assumption that they will *always* be run under certain conditions, which may or not be present in the Finder. This means, that while you *can* change the file type of a normal application file to that of an EXE file, and that you *can* run that file from a development environment, the reverse is not necessarily true. In other words, you *can't* change the file type of an EXE file to that of a normal application, run it from the Finder, and expect it to run correctly. Nine times out of ten, EXE files that have been turned into applications will crash when you launch them from the Finder.

Having said that though, you should also know that *internally*, applications and EXE files can be exactly the same. The only difference is the file type, and that the EXE file might *not* be able to run from the Finder. Also, since they are intended to be run from a development environment, most EXE programs have a text-only interface (like old Apple II programs) and do *not* use the desktop interface. However, EXE programs can use the same tool sets that desktop-based applications can.

New Desk Accessory

New Desk Accessories (NDAs - file type \$B8) can be thought of as "mini-applications" that run inside other desktop programs. This allows the user to access and use an NDA without having to leave the program that they are currently using. This capability has made NDAs *very* popular with IIGS owners, because they give the IIGS a cool multi-tasking "feel." However, it's not true multi-tasking at all (at least not in the strict definition of the term). What's happening is that, during

the Event Loop of the host application, the application is giving any active NDAs a chance to "do their thing." When all of the active NDAs are finished, control is then returned to the host application. (How does all of this happen? Basically, the System Software takes care of it for you!)

NDAs are only a little more difficult to write than applications. The two main differences are that an NDA must have a special "header" at the start of it, and that there are four special procedures that *must* be present in the NDA for it to function. Within those restraints, NDAs are similar to applications in that NDAs get a chance to customize their environment, they wait for events, and they have to clean up after themselves. First, let's talk about the differences

Four Special Routines

While an NDA can contain as many routines as you want it to, there are four very special routines that *must* be present in every NDA for it to work properly. These four routines are called by the System Software at various times so that the NDA can take the necessary steps to set itself up, shut itself down, open and close its window (or windows), and respond to any events that are sent to it. These four routines are:

DAInit - This routine is called by the system during the IIGS boot process, and when the Desk Manager tool set starts up and shuts down (this happens when a desktop application starts up and when it shuts down). As you might guess from its name, the job of the DAINit function is to do whatever is necessary to get the NDA ready for work. What you might not guess from the name is that it is also responsible for shutting down the NDA when the Desk Manager shuts down. The DAINit function knows which task it is supposed to perform by checking the value of a flag that the Desk Manager passes to it. If the flag is set to zero, it means that the Desk Manager is shutting down, and the NDA should do the same. If the flag is non-zero, it means that the Desk Manager is starting up and the NDA should prepare itself for use.

DAOpen - The DAOpen routine is called the first time the user selects the NDA from the Apple Menu. Again, the name of the routine gives us a pretty good clue as to what it's supposed to do. The job of this routine is to open the NDAs window, so that the user can interact with it. This routine must also make sure that all of the tool sets the NDA needs are started up. If the host application has not started a needed tool set, the NDA must

start it up itself. (The NDA should also shut down the tool set when it has finished with it.) When the DAOpen routine finishes, it returns to the Desk Manager a pointer to the window it just opened. (Don't know what a "pointer" is? Go back and read part one of this series.) The Desk Manager uses this pointer to keep track of the window so that it can tell the NDA when there is an event that it needs to take care of.

DAClose - If you have already guessed that this routine is supposed to close the window that was opened by DAOpen, you guessed right! Note that if you start up any tools in the DAOpen routine, this is *not* the best place to shut them down. A better time and place to do this is in the DAINit procedure when it receives the shut down message. The reason for this is that, if your NDA starts up a tool, another NDA that needs that same tool might come along and, since you've already started it, will go on without starting the tool itself. If you then shut down the tool while that second NDA is still open, it might try to use the tool and crash the system!

DAAction - Like the Event Loop of an application, this is where the "meat" of an NDA is. Unlike an application however, an NDA doesn't need to have its own Event Loop. Instead, the NDA receives events that are passed to it (via the Desk Manager) from the Event Loop of the host application. After receiving the event, the DAAction routine reacts to it just as a normal Event Loop would, and calls another routine to handle the event. When it finishes handling the event, it returns a boolean (true or false) value to the Desk Manager, telling if it actually handled the event or not.

The NDA Header

In case you've never heard the term before, a "header" is simply a block of data, in a very specific format, that comes at the *start* of a file or record. The information contained in the header is usually used by another program to help it work with the data that comes after the header. A IIGS NDA header is used by the Desk Manager to find out exactly *where* in memory the four special routines for an NDA are. Without this information, the Desk Manager would not know how to access these routines and the NDA would not work properly.

The NDA header also contains information that tells the Desk Manager how often the NDA should be called, what kinds of events it can handle, and the string that it should use when it puts the NDAs name in the Apple menu.

If all this sounds a bit complex, don't worry. If you are writing your NDA in a high level language, the compiler will take care of all the tiny details of writing an NDA (like generating the NDA header) for you. All you really have to worry about is getting your NDA to do what you want it to do. If you are working in Assembly Language however, you have to do all of this dirty work yourself!

For more information on writing NDAs, check out the Desk Manager chapter of the *Apple IIGS Toolbox Reference: Volume 1* (chapter 5), chapter 4 of the ORCA/C, ORCA/Modula-2 or ORCA/Pascal manuals and the Apple II File Type Note for file type \$B8. The Toolbox Reference will give you the down-and-dirty Assembly Language details you need to know, the ORCA manuals will tell you how to more easily write an NDA using the compiler that you have, and the file type notes will refer you to any other technical notes that you should read on this subject.

Classic Desk Accessories

Like New Desk Accessories, Classic Desk Accessories (CDAs - file type \$B9) can be thought of as mini-applications that you can access from inside other programs. However, unlike NDAs, CDAs are available from *any* desktop or ProDOS 8 program! However, CDAs do not run inside a host program as NDAs do. To access a CDA, you have to temporarily suspend execution of the program you are using and go to the CDA menu (by pressing Command-control-escape at the same time) to select the CDA you want to work with. When you finish working with a CDA, you return to the CDA menu and then to the application you were originally in.

Like an NDA, a CDA has a special header. It also has two special routines that must be present to function properly. As for an NDA, the header contains the name of the CDA, and points to the two required routines so that the Desk Manager can easily find them. So, let's discuss the two special routines briefly:

StartOfDAGCode - In a "normal" program, this would be the "main" program code that's called when a program is first executed. In other words, when the user selects this CDA from the CDA menu, this is the routine that gets called by the Desk Manager. This routine then presents the user with the options that are available in the CDA, and handles all of the choices that are made. Since CDAs are available from *anywhere*, they're required to use a text-based interface (like old Apple II programs) to interact with the user.

ShutDownRoutine - This routine is called whenever the Desk Manager is shut down (i.e. when you quit a normal desktop application). This gives the CDA a chance to clean up anything that it may have done in the StartOfDAGCode routine.

The presence of a header and these two routines may make you think that a CDA is a lot like an NDA, but it isn't. The main difference is that a CDA can't assume *anything* about the environment it was called from. For example, if you want your CDA to be useable from ProDOS 8 programs, you can't assume that GS/OS or the Toolbox is going to be active. This makes CDAs very tricky to write and has led to a lot of CDAs that work in one environment or another (ProDOS 8 or GS/OS), but not both.

As with NDAs, to get more information on writing CDAs, check out the Desk Manager chapter of the *Apple IIGS Toolbox Reference: Volume 1* (chapter 5), chapter 4 of the ORCA/C, ORCA/Modula-2 or ORCA/Pascal manuals, and the Apple II File Type Note for file type \$B9.

Temporary Initialization File

A Temporary Initialization File (TIF - file type \$B7) is a file that is loaded and run as the IIGS boots. After it finishes, it returns control to the operating system and is removed from memory. As you might expect, because they don't hang around long, TIF files aren't the most useful things in the world. On the IIGS, TIFs have been used for such things as playing startup sounds, displaying startup pictures and other "fun" tasks that make the time spent waiting for the IIGS to boot more enjoyable. When a TIF file gets control, the IIGS is in a very fragile state (i.e. you can't assume *anything* about the state of the operating system or the Toolbox) so they must exercise extreme caution in performing their tasks. And, unlike an application, a TIF file must end with an Assembly Language RTL (ReTurn Long) instruction and not a GS/OS Quit call. Of course in Assembly Language, this is no problem, and as it turns out, it isn't much of a problem in any of the ORCA languages either. This is because all of the ORCA compilers have an "RTL" compiler directive that instructs the compiler to end your program with an RTL instruction instead of a Quit call.

For more information on writing TIF files, see the Apple II File Type Note for file type \$B7.

Permanent Initialization File

A Permanent Initialization File (PIF - file

type \$B6) is similar to a TIF file in that it is loaded and executed while the IIGS is starting up, it has to be careful when it first gets control of the machine, and it must end with an RTL instruction. Unlike a TIF however, a PIF file is *not* yanked out of memory after it finishes executing. This means that the PIF can continue to perform its task in the background after the boot process is finished.

So, in most cases, a PIF file uses the first time it gets control (during the boot process) to set things up so that it can take control again at some later time. After the boot process is complete, the PIF sits quietly in the background waiting to get control again so that it can do its duty. For example, during the boot process a screen saver program might install a timed task that would periodically take control of the IIGS to see if the mouse had moved or a key had been pressed since the last time it had control.

For more information on writing PIF files, see the Apple II File Type Note for file type \$B6.

Drivers

Drivers (file type \$B8) are programs, which are loaded at boot time, that allow GS/OS communicate with hardware devices that are attached to the computer. There are several different "flavors" of drivers: printer drivers, port drivers, Media Control tool set drivers, etc. However, they all have the same goal of allowing the IIGS to effectively communicate with a particular piece of hardware. For example, in System 6, there are individual drivers for 3.5-inch drives, SCSI hard disks, CD-ROM drives, the modem port, the printer port and several other devices.

The specifics of how to actually write a driver are a bit beyond the scope of this series of articles, but if you want to write one on your own, you should check out the *GS/OS Device Driver Reference*.

User Tool Set

Earlier, I discussed the advantages of using a pre-made tool over using one you build yourself. While this is a good rule of thumb, there are times when the tools you have simply don't do what you need them to do. So you might have to build your own tools. And, if they are useful enough, you would probably want to be able to use them again. For that reason, the IIGS supports the use of user tool sets. These are collections of routines that you write yourself and then access via the same means that you access the built-in IIGS tool sets.

Here again, writing a user tool set is a bit beyond what I want to discuss in this series. But, if you have a burning need to write your own tool set, see the Tool Locator chapter and Appendix A of the *Apple IIGS Toolbox Reference: Volume 2* for complete information on how its done. (You may also want to check out Apple IIGS Technical Note #73, and the Apple II File Type Note for file type \$BA for even more information.)

Control Panels

Control panels (also known as "CDevs" - file type \$C7) are small programs that can be accessed through the Control Panels NDA in the Apple menu. The purpose of these programs is to give the user control over some part of the IIGS that they would not ordinarily have control over. For example, control panels exist that allow you to set up a RAM disk, assign sounds to be played when certain events happen and to customize the pattern that is displayed on the desktop.

Writing a control panel involves several concepts that we have not yet discussed, so, I'll save the specifics for a future article so that we can discuss them in greater detail. In the meantime, I'll describe the basic structure and operation of a control panel and tell you that if you want to know more about how to write control panels, the definitive resource is the Apple II File Type Note for file type \$C7.

The Basics

A control panel is made up of three individual resources that are kept in the resource fork of the control panel file. (If you don't know what a "resource" or a "resource fork" is, don't worry, we'll be discussing that in the next part of this series.) These three resources are:

Icon Resource - This is the icon that the Control Panels NDA uses to represent this control panel in the list of control panels.

CDev Flags Resource - This resource contains configuration information about the control panel that the Control Panels NDA uses to determine how it should interact with the control panel. Among the information in this resource is the name of the control panel, the name of the author and a series of flag bits that tell the Control Panels NDA which types of events the control panel can respond to.

CDev Code Resource - This resource contains the actual program code that does the work of the control panel. Like a TIF or a PIF, this code must return to its caller by using a RTL instruction instead of a Quit call. All of the ORCA

compilers have special directives that create the appropriate code for this. For more information, see chapter 4 of your ORCA language manual.

So, the way a control panel works is this: When the system is booting up, the Control Panels NDA is loaded and in turn, it loads all of the CDev Flags and icons from all of the active control panels. During this process, it checks the CDev Flags of each control panel to see if any of them need to do anything at boot time. If they do, control is passed from the Control Panels NDA to the CDev Code so that it can perform its boot time activities. After the boot process is finished, the user can access the control panel by opening the Control Panels NDA and selecting it from the list of available control panels. When a control panel is selected, the Control Panels NDA loads the CDev Code from disk and begins passing it information ("events") about what the user is doing. The control panel simply reacts to these events just like an application or NDA would, until the user closes the control panel's window.

As you can see, control panels are a bit of a cross between PIFs and NDAs. Since they have to opportunity to act at boot time, they can arrange to "hang around" and perform some task in the background just like a PIF file. And, since they are "fed" events by the Control Panels NDA, they can be set up internally to work like an NDA's DAAction procedure. This hybrid nature makes control panels very flexible and easy to write.

Finder Extras

With the introduction of System 6, Apple gave the IIGS Finder a powerful new feature—the ability to use add on modules called "Finder extras." While you are in the Finder, these extras are available from the Extras menu at the far right of the menu bar. Writing a Finder extra is fairly straight forward, about the only thing you have to remember is that a Finder extra must end with an RTL instruction (just like a TIF or PIF).

You may have also heard Finder extras being referred to as "Finder extensions." While these terms are basically interchangeable, in the truest sense of the terms, a Finder extra is something that is active only when the Finder is active, and a Finder extension is *any* program that communicates and interacts with the Finder. For example, our EGOed program is a new desk accessory that you can use from any desktop program to read text files. However, if you are in the Finder, EGOed realizes this, and communicates with the Finder to allow

you to automatically load and read files simply by double-clicking on them.

EGOed and other Finder extras communicate with the Finder through a facility known as Inter-Process Communication (IPC). By using IPC, a Finder extra and the Finder can communicate to perform tasks that the Finder was not otherwise intended to be able to do.

We will discuss IPC, and how to write a Finder extra in greater detail in an upcoming article. For now though, you can refer to the *Programmer's Guide to System 6* and to the article "IPC (Igor's Playful Code)" in *GS+ V5.N3* for more information about how to write your own Finder extras.

And All the Rest

Well, that about covers all of the different types of programs you can write for the IIGS. There *are* other types, but they are add-on modules for third party programs (e.g. a Nifty List command module or a HyperStudio New Button Action). The types I've gone over here are what you might call the "official" list of types that were created by Apple for use with the IIGS. If you want to write an add-on module for a third party program, your best bet for information is to contact the publisher of the program, and to go through the Apple II File Type Notes to see if any information has been published there.

What's Next?

In the next part of this series, we'll discuss some of underlying concepts of the IIGS Operating System (GS/OS), the IIGS Toolbox and all of the miscellanea that you need to know to actually write a IIGS program. Until then, keep hacking away at those keyboards! **GS+**

Mr. Priceguide Looks at Hard Disks

By Erik "Lurch" Kloeppel

How many of you computer people out there have a hard drive? If you have an Apple IIGS (as seems likely, since you are reading this magazine), the odds are pretty darn good that you have at least a small one.

Note the operative word there: "small." It doesn't seem to matter a bit what size your hard drive is, it's just *too* small! So, what can you do about it? You have several options: Ignore it and hope it goes away, shell out a million bucks for a drive with case and power supply, or simply upgrade your existing drive from microscopic to something visible with the naked eye, by replacing the actual hard drive mechanism.

It is not within the scope of this particular article to go through the few steps necessary to putting a new mechanism in your hard drive case; that's actually the

easy part. The hard part is finding the money with which to do it, which, of course, always leads to the painful cry from the dark corners of your computer room: "How much is this going to cost me?"

Some time ago, I was asking myself that same question, and having a few quarters to rub together, sat down one evening to see what I could afford. That evening turned into a full weekend as I dug through all the computer magazines I could find, looking at the ads. When I was done, I had my answer: "Not nearly as much as I thought it would."

Think about this: When I bought my first hard drive in 1988, it was a 62MB Seagate with access times measured against geologic movement (I think some CD-ROM drives are faster). The cost? Nine hundred dollars, which was *cheap*

back then! The same drive was available from other sources for over \$1,200.

Today, you can't even find a 60MB drive on the national market. Generally speaking, the smallest drive available today is in the 120MB range, and the most common are in the 170-200MB range, and it's easy to find drives in the *gigabyte* size range for under \$800.

Now, if you have read this far, I think I can presume you are interested in a hard drive with greater capacity. If you continue reading, you should discover that it really is less expensive to assemble your own drive, as opposed to buying one fully assembled from your favorite Apple II vendor. So, with that in mind I've assembled the list shown in "Mr. Priceguide's Guide to Hard Disk Mechanisms." In this list, I've included a number of drives from the May 1994

Mr. Priceguide's Guide to Hard Disk Mechanisms

<u>Mechanism</u>	<u>Capacity</u>	<u>Access Time</u>	<u>Size (inches)</u>	<u>Price</u>	<u>Cost/MB</u>	<u>Interface</u>	<u>Vendor #</u>
Seagate 6080S	80MB	17ms	2.5	\$89	\$1.11	SCSI	1
Quantum ELS127S	127MB	17ms	3.5 x 1	\$185	\$1.45	SCSI	3
Quantum GLS127S	127MB	16ms	2.5	\$269	\$2.12	SCSI-2	5
Quantum ELS170S	170MB	17ms	3.5 x 1	\$179	\$1.05	SCSI	2
Quantum LPS170S	170MB	14ms	3.5 x 1	\$185	\$1.08	SCSI	4
Quantum GLS170S	170MB	16ms	2.5	\$289	\$1.70	SCSI-2	5
Seagate ST2209N	183MB	18ms	5.25 HH	\$169	\$0.92	SCSI	1
Toshiba MK2224	200MB	17ms	2.5	\$439	\$2.19	SCSI	11
Quantum LPS240S	240MB	16ms	3.5	\$249	\$1.04	SCSI	5
Maxtor XT7245S	245MB	15ms	3.5 x 1	\$239	\$0.96	SCSI	5
Quantum GLS256S	256MB	16ms	2.5	\$379	\$1.48	SCSI-2	5
Quantum LPS270S	270MB	16ms	3.5 x 1	\$240	\$0.89	SCSI	4
Quantum LPS340S	340MB	12ms	3.5 x 1	\$282	\$0.83	SCSI	2
Seagate ST3390N	341MB	12ms	3.5 x 1	\$324	\$0.95	SCSI	8
Conner CP30340	343MB	13ms	3.5	\$305	\$0.89	SCSI	7
Maxtor XT7345S	345MB	14ms	3.5 x 1	\$334.75	\$0.97	SCSI	9
Seagate ST2502N	435MB	16ms	5.25 HH	\$299	\$0.69	SCSI	6
Seagate ST3550N	456MB	12ms	3.5 x 1	\$359	\$0.79	SCSI	10
Fujitsu M2624FA	520MB	12ms	3.5	\$494.85	\$0.95	SCSI	9
Quantum LPS525S	525MB	10ms	3.5 x 1	\$499	\$0.96	SCSI	3
Seagate ST3600N	525MB	12ms	3.5 x 1	\$425	\$0.81	SCSI-2	1
Seagate ST3655N	540MB	12ms	3.5 x 1	\$420	\$0.78	SCSI	10
Quantum LPS540S	540MB	10ms	3.5 x 1	\$439	\$0.81	SCSI	8
Conner CP30540	540MB	10ms	3.5 x 1	\$494	\$0.91	SCSI	5
Seagate ST4702N	601MB	16.5ms	5.25 FH	\$399	\$0.66	SCSI	6
Seagate ST4767N	665MB	12ms	5.25 FH	\$499	\$0.75	SCSI-2	6
Fujitsu M2263HA	672MB	16ms	5.25 FH	\$379	\$0.56	SCSI	1
Seagate ST41200N	1037MB	15ms	5.25 FH	\$699	\$0.67	SCSI	6

Key: Drives are sorted by capacity and price from lowest to highest. Access times are in milliseconds and lower numbers indicate faster times. Drive sizes are in inches and represent the physical dimensions of the drive mechanism. The "FH" and "HH" designations stand for "Full Height" and "Half Height" mechanisms respectively. Interface type (i.e. "SCSI" or "SCSI-2") should be verified before ordering. Vendor addresses and phone numbers can be found in the "Vendor Contact Information" table at the end of this article.

Table 1: Prices of Pre-Assembled Drives Versus Bare Drives

<u>Capacity</u>	<u>Apple Vendor 1</u>	<u>Apple Vendor 2</u>	<u>Bare Drive (Best)</u>	<u>Do It Yourself Savings</u>
170MB	\$349	\$239	\$179	up to \$170
240MB	\$399		\$239	\$160
340MB		\$349	\$282	\$67
540MB		\$579	\$420	\$159

Note that bare drive prices do not include additional hardware (case, power supply, cables, etc.) needed to actually use the drive. Pre-assembled drives include everything you need to use the drive immediately.

issues of various computer magazines—virtually every drive that I could find that was listed at under \$500 (with one exception) is included.

Before you get all excited by these prices however, I must point out again that these are *bare* drives—no enclosures or power supplies are included! If you don't already have a hard drive that you can steal the enclosure from, or don't want to give up your current drive in it's enclosure, you can get a suitable enclosure from several sources for less than \$100. If you don't feel up to a little screwdriver work, you *can* get assembled hard drives (drives in enclosures, ready to go) from one of the very few vendors who still support the Apple II—several of whom advertise in this very magazine. (Three of these vendors are shown in the "Vendor Contact Information" sidebar.)

(As I said before, discussing the process of actually installing one of these drive assemblies is beyond the scope of this article. In other words, if you are going to buy one of these drives and try installing it yourself, you *must* know what your are doing—or know someone that can do it for you! However, if you are interested in learning how to do the installation yourself, the publisher has informed me that he would be willing to publish a series of articles on the subject, so just let me know.)

Also, these are all SCSI devices. Some early Apple drives (such as the ProFILE) used other types of drives, which are difficult to upgrade, and probably not worth the effort. Since SCSI is the de-facto standard, that's what I've concentrated on here. So, you needn't worry here about whether a particular drive is SCSI, IDE, ESDA or whatever—they are *all* SCSI.

However, there are actually *two* flavors of SCSI: Original SCSI and the newer SCSI-2 standard. The problem for Apple IIGS owners is that SCSI-2 drives won't

work reliably with anything but the RamFAST SCSI card. So, if you have a RamFAST you have nothing to worry about. But, if you have an Apple SCSI card (either the old revision C card or the newer High-Speed card), you need to make sure you get a drive that conforms to the original SCSI standard. I've tried to ascertain what the interface is for each drive, but few ads contain this information. So I would suggest that you confirm that the drive you are interested in is of the type you can use before you actually commit to the order. (Note that some SCSI-2 drives are backward compatible with the original SCSI standard, so be sure to ask the vendor when you order!)

There are some claims that certain brands of hard drive are better, that is, more reliable, than others. I cannot say this from my own experience, and since the Mean Time Between Failures (MTBF - or how many hours the drive should be able to operate before it fails) is steadily on the rise for all manufacturers, the criteria for judging are getting rather slippery. In general, I cannot detect a difference between any of the major name manufacturers (Conner, Seagate, Micropolis, Quantum, Maxtor), and doubt you can go far wrong with any one of them. And since everything comes with a warranty these days, even if the drive arrives at your front door in a casket or takes a powder in the first year or so, you should be protected.

The drives shown in "Mr. Priceguide's Guide to Hard Disk Mechanisms" are sorted by their capacity and how big a dent they would make in your wallet. The manufacturer, model, speed, and physical size are followed by the best price I could find (as of this writing) and the vendor that is offering that price. If you look carefully, you'll notice that the capacity of the drive has only a passing relationship to the cost, which is based largely on the data I've included with the make/model. However, the really

important information is the cost per megabyte figure for each drive, because the most important aspect of your search for a new drive is the ratio between capacity and how much you spend. These figures ranges from a crummy \$2.19 per megabyte for a 200MB 2.5-inch drive to \$0.67 per megabyte for a 1 gigabyte 5.25-inch drive. In general, anything in the vicinity of \$1 per megabyte is more than acceptable for a "small" (less than 300MB) drive, and as capacity increases, cost per megabyte goes down.

So, you may be asking yourself, "Is it worth the trouble of buying one of these drives and installing it myself?" In other words, how do these prices compare with buying a pre-assembled drive from your chosen Apple II vendor? A quick perusal of ads in the most recent issue of our favorite Apple II magazines shows prices like the ones in Table 1.

If you add about \$60 to the prices shown for a bare drive mechanism to purchase an enclosure, cables, etc. (all of which are needed to actually *use* one of these mechanisms), you will see that in some cases the pre-assembled drives come close to the same cost; but more frequently, pre-assembled drives are substantially more expensive. Not to mention that the selection of drive capacities is rather more limited.

I wish to point out that all of the prices shown in these tables are mail order prices, and gleaned from various computer magazines from all segments of the computer industry. So, if you want the price indicated, you should call the vendor listed by the drive of your choice.

Mail order also means shipping and handling—costs not included in this list. Prices obtained from magazine ads are also at least months old, in some cases as many as five months, so they are hopelessly out of date. This means that prices at your local computer store may well be significantly lower.

APPLE II EXPANSION HEADQUARTERS!

WE HAVE THE PRODUCTS YOU NEED WITH THE PRICES YOU WANT!

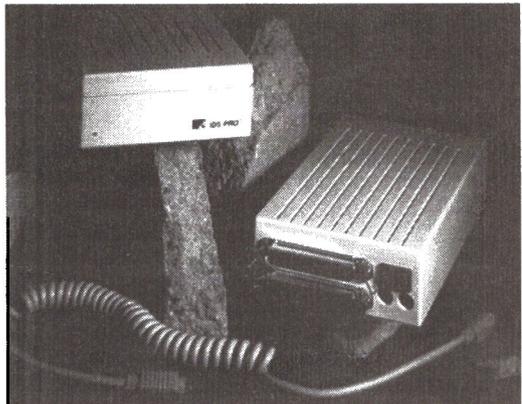
HARD DRIVES

Our NCS Pro SCSI Hard Disk drives have all of these great features:

- Two Year Manufacturer's Warranty on drives from 170mb. to 340mb.
- Five Year Manufacturer's Warranty on drives from 540mb. to 1800mb.
- Thirty Day Satisfaction Guarantee
- High impact ABS case (not flimsy plastic material)
 - Access indicator light
- Internal termination accessible via access door on drive
 - External SCSI ID selector

Every NCS ProDrive comes complete with all necessary cables for the Apple IIs. All NCS Pro Hard Drives for the IIs come with System 6.0.1 and 25 megabytes of shareware, freeware and public domain software at your disposal. NCS Pro Hard Drives can also be ordered for the Apple IIe with the latest ProDOS operating system installed.

And if you're not satisfied, just return the drive by taking advantage of our no-hassle thirty day satisfaction guarantee for a refund.



NCS Pro Drives

170mb. Quantum LPS	*NEW*	\$269
270mb. Quantum LPS		\$299
340mb. Quantum LPS		\$369
540mb. Quantum LPS		\$499
1080mb. Quantum Empire		\$859
1800mb. Quantum LPS		\$1099

30 DAY MONEY-BACK GUARANTEE

Orders over \$300 automatically upgraded to Airborne Express!

NCS Pro Removable Drives

***Special* NCS Pro R44mb. \$199 (limited supply)**

44mb. Syquest Removable Drive

NCS Pro R105mb. \$429

105mb. 3.5" SyQuest Read and Write

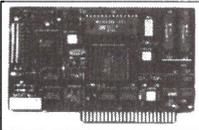
All removable drives include one cartridge. Extra 44 / 105 carts are \$68 / \$89

HARD DRIVE INTERFACES

SEQUENTIAL SYSTEMS

RamFast SCSI Card

The RamFast is the most powerful SCSI interface available for the Apple IIs Computer. It features expandable Cache, and upgradable ROM support.



RamFast GS v3.01

w/256k Rev.D **\$139 / \$149**

w/1mb.Rev.D **\$189 / \$199**

(with hard drive / without hard drive)

RamFast v3.01 Upgrade (for current RF users) Rev. C or D \$15

TRADE-IN YOUR APPLE HIGH SPEED SCSI CARD FOR A RAMFAST SCSI CARD, CALL FOR DETAILS!

APPLE High Speed SCSI Card \$99

\$109 without hard drive purchase

SEQUENTIAL SYSTEMS

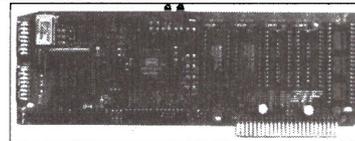
RAM GS

(4megs/GS) **\$119***

*\$129 without any other purchase

- 4MB. ON BOARD FOR IIGS
- 100% DMA COMPATIBILITY
- INCLUDES DIAGNOSTIC UTILITIES DISK

ZIP ACCELERATORS



Bring your IIs back to life with ZIP Accelerators. Over **300% faster** with a Zip 9/32 installed. Makes your GS Finder like new!

Zip 7mhz/8k Cache	\$149
Zip 8mhz/16k Cache	\$189
Zip 9mhz/32k Cache	\$239
Zip 10mhz/64k Cache	\$339 Special

SEIKO DATA WATCH SPECIAL!

Put the scheduling power of your Apple II on your wrist. Allows you to keep daily details, appointments all on your wrist. Uses the easy to use Time-Trax 23-month scheduling software. For any Apple II computer! Includes interface cable, transmitter, Silver slim-line LCD watch, software and manual.

\$24.95
DELIVERED OVERNIGHT! (MOST AREAS)
\$19 WITH ANY OTHER PURCHASE

Formerly I.R.O. Computer Sales, Inc.



School and Government P.Os Welcome!

Serving the Apple II Community for over 5 Years!

Other
World
Computing

THE EVEN SMALLER ITALIC PRINT!

No Credit Card Salesman! Standard shipping charges for UPS Ground or Priority Mail. Cash orders \$6. No COD sales. Orders over \$500 require a \$50.00 cash deposit. Charge of \$6. for UPS GND and \$10. for Airborne Express. Call for international charges for 44 and 105mb. drives. Prices, products and tax subject to change. Sales tax not included in state purchases. Returns of hardware for up to 30 days with original receipt. Prices call before returning merchandise. No returns accepted without RAM. All returns for credit are charged a handling fee and shipping is not refundable. Returns for all IIs and complete product returns are at our discretion. All returns except for hard drives are subject to a 10% restocking and handling fee. Shipping charges vary. Prices and availability subject to change. This advertisement is a 1984 Other World Computing Corp. a division of New Concepts Development Corporation. All systems are to be compatible by law and feature of a system in use. Vendor's name is not to be included. Prices, availability, specifications, small print, Product Uniform Price Code and the weather are subject to change without notice. (Continental USA only)

VITESSE

Quickie / Inwords Bundle.....\$199

Includes Vitesse's Quickie Handscanner for the Apple II and WestCode Software InWords v1.1 OCR Software.
Quickie (separately).....\$149

HEAVY DUTY POWER SUPPLIES

Apple IIgs H-D Power Supply \$79

Apple IIe H-D Power Supply \$69

All Heavy Duty Power Supplies have a 1-year Warranty!

NEW CONCEPTS DEVELOPMENT

IIgs 3200 Color Graphics Multi-Conversion Utility \$34.95

MISC.

AMR 5.25" for LC or A2.....\$129
AMR drives are daisy chainable

AE SuperDrive.....\$229
1.44/800K 3.5" Floppy Drive for A2 & Mac

Apple SuperDrive I/F.....\$138
3.5" interface for A2 and IIgs / Req'd for 1.44mb.

NCDC OCR Scanner.....\$79
Allows any computer with a serial interface to take in text from any typewritten page.

Memory Upgrades

1MB SIMMs.....\$46

1MB RamFast Upgrade.....\$59

1MB Expansion Set (8 Chips).....\$49

1MB GS-Ram III Upgrade.....\$59

256k GS-RAM Ultra Upgrade.....\$15

256k Apple Exp. Kit.....\$18

SEQUENTIAL SYSTEMS DiscQUEST™

SEQUENTIAL SYSTEMS DiscQUEST™ Software bundled with a CD-ROM drive allows YOU the Apple IIgs user to explore the hundreds of CD-ROM disks previously inaccessible.

SEQUENTIAL SYSTEMS DiscQUEST™ has powerful features you would not expect, for instance:

- Search CD-ROMs by Word, Title, Author
- Display Graphics from Mac or IBM CD-ROM disks
- Play audio from CD-ROM disks
- Browse through Mac and IBM CD-ROM disks

SEQUENTIAL SYSTEMS DiscQUEST™ CD-ROM Bundle **\$569**
-includes Software, RamFast SCSI, CD-ROM Drive and 4 CD-ROM titles!

SEQUENTIAL SYSTEMS DiscQUEST™ Software alone **\$79**
-includes Family Doctor CD-ROM



SOFTWARE

SEQUENTIAL SYSTEMS

Switch-It!.....\$48.00
Switch-It! allows you to run several applications at once, and switch between them in an instant. Compatible with practically all GS/OS programs. Requires GS/OS 6.0, and 4Mb for optimum performance (2Mb minimum)

DREAMWORLD

DreamGrafix.....\$59.00
3200 COLOR PAINT PROGRAM FOR II GS

ECON TECHNOLOGIES

Universe Master.....\$64.95
COMPLETE II GS HARD DRIVE MANAGEMENT UTILITY

AutoArc.....\$24.00
NEARLY DOUBLES THE SIZE OF YOUR II GS HARD DRIVE

Roger Wagner Publishing

HyperStudio.....\$97.00
FUN AND EASY TO LEARN HYPERMEDIA FOR THE II GS

WestCode SOFTWARE

Pointless.....\$46.00
TRUE TYPE FONT INTERPRETER FOR II GS

InWords.....\$78.00
ALLOWS THE QUICKIE TO SCAN IN TEXT FOR EDITING

Vitesse

Harmonie.....\$29.00
II GS PRINTER DRIVERS FOR HP PRINTERS

InTrec

ProTerm 3.1 NEW.....\$78.00
HIGHLY RATED APPLE II COMMUNICATIONS PROGRAM

TimeWorks

Publish It! v4.0 **SPECIAL.....\$29.00**
APPLE II Desktop Publishing Program

Seven Hills Software

(ALL COMPATIBLE WITH THE MANAGER)

Shoobox.....\$34.00
HOUSEHOLD RECORD KEEPING AND MESSAGING

Express.....\$24.00
II GS PRINT SPOOLER

Kangaroo.....\$28.00
TIMESAVING FOLDER UTILITIES FOR II GS

Manager.....\$48.00
MULTITASKING ON THE II GS

SPECTRUM (NEW in stock).....\$84.95
SYSTEM 6.0 BASED COMMUNICATIONS PROGRAM, SUPPORTS ZMODEM AND COMPUSERVE B+ PROTOCOLS!

Apple Computer, Inc.

GS/OS System 6.0.....\$39.00

HyperCard GS.....\$69.00

// Productive

BullsEye TrueType Font Collection for the Apple IIgs. Includes over 300 TrueType Fonts, Font Viewing Utility, America Online Software Kit and MUCH MORE for only.....\$45.99

MODEMS

SUPRA v.32bis, 14.4bps Send / Receive Data/Fax Modem (Universal).....\$209

Zoom 2400 with Cable for IIgs.....\$79

AE ReadyLink Communication Software only \$39 with Modem Purchase

Call us toll-free to order:

(800) 275-4576

Hours : 9am - 8pm CST M-F and 11am - 4pm CST Saturdays

FAX ORDERS : (815) 338-4332
TECH SUPPORT (815) 338-8746
INTERNATIONAL (815) 338-8658
ILLINOIS RESIDENTS (815) 338-8685
224 WEST JUDD STREET
WOODSTOCK IL 60098

These prices (and the prices in my complete guides, which can be found in the Apple II Special Interest Group on DELPHI) are intended as a *guide*. In other words, you should never have to pay *more* than what's posted in my lists, and you shouldn't be surprised at lower prices.

If you found this article of some use, but are more interested in price comparisons

for laser printers, floptical drives, modems, or some other equipment, please let me know, and I will make every effort to provide the information you seek in an upcoming issue. (You can contact me in care of this magazine or by sending e-mail to lurch@delphi.com if you have access to the InterNet.) In fact, in our next installment, we'll be looking at low cost (under \$700) laser printers that you can

use with your IIGS. Until then, remember Mr. Priceguide's . . .

Standard Disclaimer: Neither Mr. Priceguide nor *GS+* Magazine endorse any company on or off this list, and can make no claims regarding the business practices, quality of service, or lifestyles of any of the companies included. However, their prices are pretty good. **GS+**

How to Buy Through the Mail . . . and Live to Tell About it!

Most people I know are very nervous about spending any kind of money on mail-order. There is *some* justification for this, as in any business there are unsavory types who just want your money and could care less about providing what you asked for. However, the most common problems that you might have with mail-order purchasing will be the result of misunderstandings between yourself and the vendor.

Keep that in mind when you place an order. It's a good idea to have all the details written down in advance; manufacturer, model number, size, extras, whether you will accept an equivalent substitute—even the shipping and taxes should be determined in advance, so you know *exactly* what you are expected to pay, and when.

You should also be aware of what are commonly termed "Seller Obligations." Basically, your order must be shipped within 30 days of placement, unless another time frame is clearly defined before you make your purchase. If, for some reason, the vendor cannot meet the deadline, he may advise you *in writing* in advance of the due date, and advise you of a new shipping date, if he knows it. If you don't respond he may assume you agree to the delay, but still he must ship your order within the original 30 days. Most importantly, you *always* have the right to cancel your order, even if you agree to a delay.

There are some steps you can take to protect yourself from fraudulent or incorrect billings, and together they add up to a pretty potent set of tools should you feel that you have indeed been ripped off.

1. Place your order by fax. This *requires* that you have a physical piece of paper at

your end detailing all aspects of your request. Make sure the information on the fax is clear and legible. Doing this eliminates the possibility of error by an employee taking your order over the phone and writing down the wrong item. (They still might send you the wrong part, but it couldn't be construed as *your* fault.) Also, since most fax machines date and time stamp delivery of a document, there would be a record of exactly when the order arrived at the company.

2. Keep accurate records of who you talked to—and *when*—by telephone. If you are unsure of the spelling of an employee's name, *ask!* Make a note of order numbers as well. Maintain this record until the warranty expires, or better yet, for as long as you own the equipment.

3. Most important, particularly if you are unfamiliar with the company, *use a credit card!* This is one instance where your credit card (or your mom's) is invaluable. There are several reasons for this, not the least of which is the convenience.

Most mail-order companies readily take common credit cards where they hesitate to take money orders, and flatly refuse to take personal checks. This is mostly because with credit cards, they *know* they will get paid, and they don't have to worry about extensive billing. The credit card company will pay the bill, and charge you.

On the flip side, if the company fails to deal with you in good faith (charge your card without sending the product, charge you twice, etc.) you have simply to write a letter to your credit card company presenting the facts at hand, and they will either A) remove the charge(s) from your bill, or B) target the company to recover

the funds they paid, once they agree with you. (Now do you see why items 1 and 2 are so important?) Of course, if they disagree with you, the charges pop right back onto your bill, so be sure that you are right.

So, what happens if you get your order and there is something unacceptably wrong with it? The instant you find the problem, get on the phone to the vendor who sold it to you and let them know. Do *not* send the order back until the vendor specifically tells you to, and then make and keep copies of all packing slips and receipts. (A fax machine can come in handy for making quick copies.) Ship the item(s) back in the original packing/boxes if at all possible.

Normally, vendors are more than happy to resolve disputes, but what do you do if absolutely *nothing* works? According to *Computer Shopper* magazine, you have a couple of options: Write to the consumer protection agency in the vendor's state, giving them the full details. Or, you can write to:

Mail-Order Action Line
1101 17th Street NW Suite 705
Washington, DC 20036

But, don't let all of this information cause you too much concern. I've been dealing extensively with mail-order for over 12 years, and have *never* had a problem that wasn't resolved with one phone call. Just be sure to follow these simple guidelines for the sake of safety, but don't expect to need them. **GS+**

Terry's Tech Service

RR#8 Parkhill, Ontario, Canada
 NOM 2K0 (519) 294-0648



All Prices in Canadian Funds, no duty, no broker fees, no 40% US exchange. Just add any applicable taxes. 2 Year Canadian Warranty on all hard drives. All types of used equipment is available, call for more info. Dealer for Seven Hills, Sequential Systems, InTrec, and WestCode. Some prices vary due to US dollar exchange rate. Fax # (519) 294-0589. Office Hours: Mon-Fri 9:30am-6:00pm Eastern.

Seven Hills Software:	Sequential Systems:	Misc. Items:	Component level repairs to most Apple products and peripherals, call for rough estimate on your specific problem. All repairs guaranteed 90 days. Good selection of replacement parts at competitive prices.
Kangaroo.....\$34.00	RamFast 256k.....\$220.00	Pair of 1Megx4 Chips.....\$50.00	External SCSI hard drives are mounted in a CSA Approved case 6.2"W x 2.6"H x 9"L. They include SCSI ID switch, Dual SCSI connectors, Fan, 25Watt Power Supply, and External Terminator.
The Manager.....\$60.00	CVRAM 4Meg.....\$300.00	GS Power Supply (Exchange) \$40.00	
Formulate.....\$44.00	Ram GS 4Meg.....\$200.00	Ile Power Supply (Exchange) \$35.00	
Space Fox.....\$27.00	InTrec:	Floptical(External with 2 Disks) \$519.00	
Gate.....\$27.00	ProTerm 3.1.....\$111.00	High Speed Modem Cbl...\$10.00	
SuperConvert.....\$34.00	WestCode:	GS to ImageWriter II Cbl....\$8.00	
Express.....\$34.00	Pointless.....\$60.00	SCSI 50-50 or 25-50 Cbl. \$12.50	
Independence.....\$34.00	TypeWest.....\$43.00	6' ADB or 6' ADB Ext. Cbl.. \$8.00	
ShoeBox.....\$46.00	External Hard Drives:	Phone-Talk kit (For AppleTalk) \$17.50	
TransProg III.....\$34.00	Quantum LPS170 \$480.00	GS RGB Monitor (New) \$200.00	
Super Menu Pack.....\$27.00	Quantum LPS270 \$520.00	GVC 14.4 Fax/Data Modem \$240.00	

No credit card orders, personal cheques ok, first order ships COD. All orders ship by Canada Post.

Vendor Contact Information

- | | | | | | |
|----|---|----|--|-----|--|
| #1 | Hi-Tech Component Distributors
59 S. La Patera Ln.
Goleta, CA 93117
805-681-9961 | #5 | Hard Drive Super Source
6225 Jarvis Ave.
Newark, CA 94560
800-252-9777 | #9 | Vektron International
2100 Highway 360, Ste 1904
Grand Prairie, TX 75050
800-725-0081 |
| #2 | D C Drives
1110 NASA Road One,
Ste. 304
Houston, TX 77058
800-473-0960 | #6 | Jem Computers
443 Western Ave.
Boston, MA 02135
617-497-2500 | #10 | V.I.P. Data Systems
5025C 28th SE
Grand Rapids, MI 49512
800-352-1150 |
| #3 | Global Horizons
1051 E. High St, Box 644
Mundelein, IL 60060
800-860-4562 | #7 | Computability Consumer Electronics
Box 17882
Milwaukee, WI 53217
800-554-9950 | #11 | Worldwide Technologies
437 Chestnut St.
Philadelphia, PA 19106
800-922-4640 |
| #4 | Tribeca Peripherals
10 Reuten Dr.
Closter, NJ 07624
800-445-6222 | #8 | SAG Electronics
867 Turnpike
North Andover, MA 01845
800-989-3475 | | |

Drive Enclosure Vendors

AllTech Electronics
602 Garrison St.
Oceanside, CA 92054
800- 995-7773

Tulin Technology
2156H O'Toole Ave.
San Jose, CA 95131
408 -432-9057

JDR Microdevices
2233 Samaritan Dr.
San Jose, CA 95124
800- 538-5000

FLI Convert

By James C. Smith

In the past couple of years, computer animations have become more and more popular. You find them featured in everything from the Lawnmower Man, to mouth-wash commercials, to Jurassic Park. FLI Convert will help you bring some spectacular computer animations to your IIGS. You won't be able to use your IIGS recreate anything as spectacular as a Tyrannosaurus-Rex, but you will be able to see some rather impressive animations.

The tools available for creating animations on your IIGS are currently limited to paint programs that will let you draw a series of pictures and then bring them together into one animation file. Platinum Paint, DreamGrafix, PaintWorks Plus, PaintWorks Gold, and just about every other paint program ever written for the IIGS have had an option to create and play animations. Creating an animation requires that you draw each of the individual frames, save them each in separate files, and then compress all the individual picture files into one animation file. This compressed animation file is usually referred to as a PaintWorks animation file (file type \$C2) because Activation's PaintWorks was the first program to use the file format. Once you have a PaintWorks animation file, you can play the animation with any one of several different animation viewer programs including the Movie Theater screen saver module in Twilight II.

On the MS-DOS computer platform there are several much more sophisticated animation creation tools. AutoDesk Animator allows you to edit animation files directly rather than working with several separate picture files. These files that AutoDesk Animator creates all have a file name ending with ".FLI" and are generally referred to as "FLIs." Some other MS-DOS application make creating animations even easier. AutoDesk 3D Studio lets you define three dimensional objects and the paths of movement that they will follow. With the click of a mouse button the computer renders each frame to create some very realistic looking animations. Recently, "morphing" programs have become very popular on the PC. You give the program two pictures and it creates an animation with the object(s) in the first picture slowly transforming into the object(s) in the second picture.

Unfortunately, there aren't any programs available on the IIGS to create 3D object-based animations or morphs. However,

with all of these programs available on the MS-DOS platforms, you can find literally thousands of animations on on-line services and BBS systems that were created with AutoDesk Animator, AutoDesk 3D studio, Morph, or some other animation program. Most of these files are stored in the standard FLI animation format. Wouldn't it be great if there were some way to use these files on your IIGS? FLI Convert is a IIGS application that allows you to do just that. FLI Convert will open FLI files created on MS-DOS computers and convert them into PaintWorks animation files that can be used on your IIGS.

Basic Program Functionality

Because of the limitations of the Apple IIGS graphics hardware, converting 256 color FLI animations to the IIGS can get a little tricky. FLI Convert is designed to make this process as simple as possible. We will begin by converting a FLI animation using the default settings to get quick and easy results. Later we will examine exactly what some of the extra options mean and how you can use them to fine tune the conversion process.

1. Getting A FLI File

The first thing you need is a FLI animation to convert. You can find hundreds of these animations on most on-line services and BBS systems in the MS-DOS/Windows areas. For example, America Online has several hundred FLIs in the IBM Graphics and Animation area. When you download a FLI it will most likely be compressed with some type of compression utility. If the file name ends in ".FLI" it is not compressed and you can directly open it with FLI Convert. If it ends in ".ARC" or ".ZOO" you can uncompress it with GS-ShrinkIt. If the file name ends in ".ZIP" you will need to unzip the file with PMPUnZip on the IIGS or any other unzipping utility that you have. If the file name ends in ".EXE" then the file is most likely a self-extracting archive and cannot be uncompressed without executing the file on an IBM computer.

You may also find FLIs on CD-ROMs and floppy disks made for the IBM. If you have the hardware to read these disks (a CD-ROM drive or a Super Drive), you can copy the FLIs off these disks and use them with FLI Convert. I found one CD-ROM, made for the IBM, which comes with a book called *Walk Throughs and Flybys*. (Written by Phil Shatz. ISBN 1-878739-40-9, \$29.95.) This CD contains

over 200 FLIs totaling over 100MB in size, and they all work great with FLI Convert. (It also contains a lot of other programs and multimedia files that aren't as useful on the IIGS.)

Your GS+ Disk contains a few sample FLIs that you can use for now to get familiar with the program.

2. Execute FLI Convert

FLI Convert is a regular IIGS desktop application that you launch from the Finder or from your favorite program launcher. It requires System Software v6.0 or later. One thing you may want to do (if you haven't done so already) before launching FLI Convert is to visit the Sound control panel and assign some sounds to a few events. FLI Convert uses the Task Completed and Bad Key Press sound events and a new event defined in System 6.0.1 called Begin Long Operation. If you have sounds assigned to these events you will get a little more feedback from FLI Convert.

3. Open A FLI File

After launching FLI Convert, select Open from the File menu. You will see a standard open file dialog with a list of all the FLI files in the current folder.

Since FLIs do not have their own file type assigned to them on the IIGS, FLI Convert looks for files with a file type of unknown (\$00), binary (\$06), or text (\$04). Files with any other file type are ignored. FLI Convert verifies that a file actually is a FLI before it adds the file to the list of selectable files. If you have a file that you think is a FLI, but it is not showing up in the list, then check the file type with NoDOS (from GS+ V4.N4) or any other file utility program. If the file type is set to unknown, binary, or text, and the file still does not show up in the open dialog, then the file is not a FLI.

There are a few other formats on the IBM similar to FLI such as FLC. You may actually have an FLC file that someone named with a FLI extension. FLCs and other non-FLI animations are not recognized by version 1.0 of FLI Convert.

You should be able to open any size FLI no matter how much memory you have. For example, FLI Convert will let you open, preview, and convert a 4MB FLI on a IIGS with only 2MB or RAM. This is because FLI Convert never loads the whole FLI into memory all at once. It just works with one frame at a time.

After selecting your file from the open dialog, you will see a window on the screen representing the FLI. (See screen shot.) This window contains some information about the FLI such as the file name, the number of frames in the animation, the number of colors used in the animation, and the speed at which the animation will play. Some of these fields will not be filled in until you begin the conversion process. This window also has a number of controls that let you fine tune the conversion process. We will discuss these controls a bit later.

4. Preview The Animation

Click the Preview button or press the "P" key to get an idea of what the converted animation is going to look like. The first thing you will see is a status window showing the progress of the conversion process. You will see messages such as "Counting Colors," "Building Color Palette," and "Matching Colors." Converting a FLI to a format that is viewable on a IIGS is *not* a simple task, so the process may take some time. FLIs use 8 bit color which means they can use up to 256 colors on screen at once. The IIGS only uses 4 bit color and can only display 16 colors at once. To display a FLI, FLI Convert needs to tabulate all the colors that are used in the FLI, determine which 16 colors are used most often, make a IIGS color palate containing those 16 colors, and then decide which of the 16 colors will be substituted for each of the 256 colors used in the FLI. When all of this is done, FLI Convert can begin to show you what the animation looks like—but it still has some work to do.

As FLI Convert displays each pixel from the FLI, it needs to look up every 8 bit color value in an internal lookup table to determine which 4 bit color will be substituted. It has to go through this lookup process for each and every pixel that is plotted on the screen. What all of this means is the animation is going to be a little slow. But remember, this is just a *preview*. When you use the convert option you can turn this FLI into a PaintWorks animation that can be displayed on the IIGS at full speed.

While the FLI is being previewed, you can press any key or the mouse button to stop the animation and return to the options screen. If you don't do anything, the animation will play to the end and you will automatically return to the options screen. Now, take a look at the color statistics. They were filled in during the color conversion process. You will later be able to use that information and the quality of the animation preview to help you determine how to adjust the

conversion options to get the best quality animation possible.

5. Converting To PaintWorks Animation Format

If you decide you like what you saw in the preview and want to convert the FLI to a PaintWorks animation file so it can be viewed at full speed and used in other IIGS applications, click the Convert button or press return. The first thing you will see is a standard save file dialog box prompting you for the name and location of the PaintWorks animation file that will be created. Change the name and/or the location if you like and then click on the Save button.

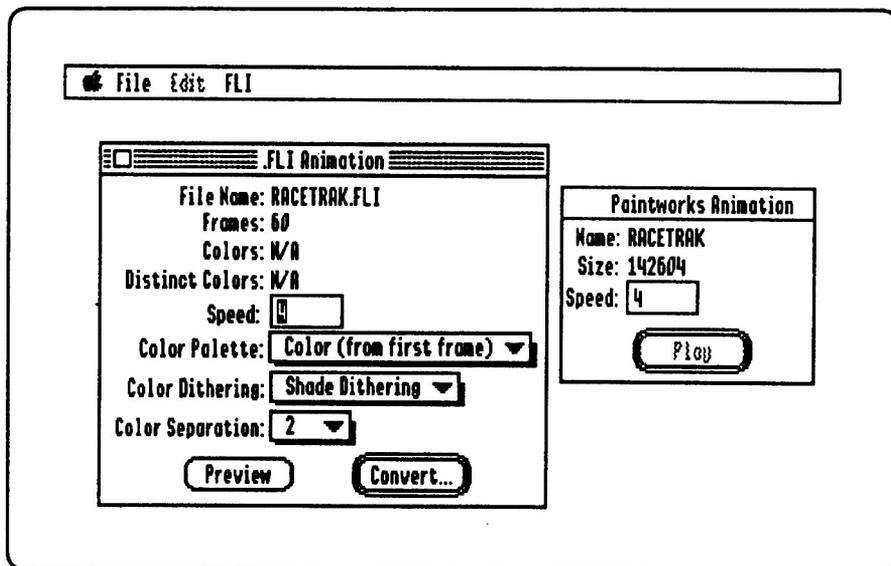
Once you specify an output file name the conversion process begins. You will once again see the animation on your screen but this time slightly slower than in the preview. This is because FLI Convert is going through the same process it did to preview the animation, but it is now also saving each frame to disk in the PaintWorks animation format.

When the conversion is complete you will hear the Task Complete sound played (if you have a sound assigned to it). FLI Convert then opens the new PaintWorks animation file and plays back the animation at full speed. The animation will continue to play over and over again in a loop until you press a key or the mouse button to stop it and return you to the options screen. You can also press the left and right arrow keys to slow down and speed up the animation. When you return to the options screen there will be a new window open. This window represents the PaintWorks animation file that was created. The window contains the file's name, size, and speed setting as well as a button you can press to play the animation again.

When FLI Convert plays a PaintWorks animation, it loads the entire file into memory all at once to get the best possible playback performance. If the PaintWorks animation file is bigger than the largest block of available RAM in your IIGS, FLI Convert will not be able to play the animation. You can try to free up some memory by closing the FLI animation window but this doesn't really free up that much memory. You may want to disable some of your desk accessories and inits that are taking up RAM or decreasing the size of your RAM disk (if you have one). You can also try playing the animation with a smaller program such as AniShow. This program is smaller than FLI Convert which leaves more available RAM for the animations to play. All of these things will only help if you have enough RAM to hold the whole animation. If you want to try to play an 8MB PaintWorks animation on a IIGS with only 4MB of RAM you can disable everything in your system folder and still not have enough RAM. The only way to view huge PaintWorks Animations that won't fit in RAM is to use a program that does not load the whole animation into RAM. The only program currently available that can do this is DreamGrafix. DreamGrafix will allow you to play a 10MB PaintWorks Animation on a IIGS that only has 2MB of memory. But, since the animation will be loaded from disk as it is played, the animation will probably be slow and jerky.

6. PaintWorks Animation Files

When you convert a FLI to a PaintWorks Animation file, FLI Convert automatically opens and displays the resulting PaintWorks Animation file. You can also open any PaintWorks Animation file you have on disk. When you use the Open option in File menu, you can select FLI or PaintWorks



animation files. After selecting a PaintWorks animation file from the open dialog, FLI Convert displays a small window representing that PaintWorks animation. You can press the Play button to view the animation. You can type a new value in the Speed field to specify the speed the PaintWorks animation should be played at. After changing the speed of a PaintWorks animation you can save the file with the new speed. There is no limit to the number of PaintWorks Animations you can have open at once. (However, you can only have *one* FLI file open at a time.) Each time you open a PaintWorks animation a new window is displayed on the screen. The actual animation is not loaded into memory until you press the Play button. This allows you to have several animations all open at once even if you don't have enough RAM to have them all loaded into memory simultaneously. Animations will be purged from memory and reloaded when needed.

Why FLI Conversion Can Be Complicated

The hardest part about converting FLI animations to a IIGS format is compensating for the hardware differences. The first major difference is color depth. As mentioned before, FLIs use 8 bit color which allows them to display up to 256 colors on screen at once. The IIGS is only capable of displaying 4 bit color limiting it to displaying 16 colors on screen at once. This means that FLIs can have *16 times* more colors than the resulting PaintWorks animation will have. You may have heard of special graphics modes on the IIGS that allow 256 color on the screen at once or even 3,200 colors on screen at once. It is possible to get the IIGS to display 3,200 color at once but you still are only using 4 bit color and are still limited to 16 colors *per line*. Also, the techniques used to get 256 and 3,200 color on a IIGS work well for still images but are not practical for use in animations. For all practical purposes, a IIGS animation can only use 16 colors at a time. This means we have to decide which 240 colors in the FLI are not that important, and which 16 we should use in the IIGS color palette. FLI Convert does this by counting how many times each color was used in the FLI. It then finds colors that are very, very similar to each other and eliminates the one that is used the least. (After all we wouldn't want to waste one of our valuable 16 color palette entries on a shade of light blue that is almost exactly the same as another shade of blue that is already in the color palette.) FLI Convert then uses the 16 remaining colors that are used the most. Any color that is not in the top 16 is thrown out.

Any pixel that uses a color that was thrown out will end up using one of the 16 colors that was kept.

The second hardware difference is the number of shades of each color that is available. FLIs can use up to 64 shades of any given color. The IIGS can only display 16 shades of a color. This means that a FLI containing smooth color gradients is very difficult to reproduce on the IIGS. This can all be summarized by saying that FLIs can use 256 colors at once out of a palette of 262,144 colors, but the IIGS can only display 16 colors at once out of a palette of 4,096 colors.

Color Modes

When you convert a FLI, you can select one of three ways to build the IIGS color palette. The simplest is to just use a gray-scale palette. This requires no color counting, or color matching, or most of the other time-consuming processes associated with mapping a 256 color FLI to a 16 color IIGS screen. Gray-scale can be used to get a quick preview of what the FLI looks like without waiting for FLI Convert to build a color palette. Gray-scale can also be used when the FLI simply has too many colors and cannot be accurately converted to 16 colors. In cases like this, a gray-scale image often ends up looking better than a color image that doesn't have enough colors.

Usually it is possible to do a good color conversion. It is just going to take a little longer than a gray-scale conversion. By default, the Color Palette conversion pop-up menu is set to "Color (from first frame)". With this setting, FLI Convert will examine all the colors used in the first frame of the FLI to determine what colors to use in the IIGS color palette. This is usually a fairly quick and accurate way of building the IIGS color palette.

Sometimes a FLI's first frame will not contain some of the colors used in frames further into the animation. Some FLIs even start with a totally black first frame. If FLI Convert tries to build the IIGS color palette from the first frame of such a FLI it is going to end up creating a PaintWorks animation that looks very ugly or may even be totally black. This is usually not the result you are looking for. In cases where the colors used in the first frame of the FLI aren't very representative of the colors used throughout the rest of the animation you will have to change the Color Palette setting to "Color (from all frames)". This will force FLI Convert to examine each and every frame in the FLI to build the IIGS color palette. This will usually result in a much better looking animation

but will also require a lot more time to build the color palette.

Color Dithering

One way we can try to compensate for the shortcomings of the IIGS hardware is to use *color dithering*. Color dithering is the process of mixing two colors to make a third. For example, we can dither, or mix together, red and blue pixels to simulate purple pixels when we don't have purple in the color palette. Dithering can be used to simulate more than 16 colors on screen at once, but dithered colors usually don't look as good as a true color. You will often notice a somewhat annoying checkerboard affect when dithering is used. If you notice this in an animation that FLI Convert creates and you don't like it you should turn off the dithering option.

FLI Convert can use two slightly different types of dithering. The first type is called Color Shade Dithering. This is used to mix two shades of one color together to make a third shade of the same color. This is very useful when a FLI contains several shades of the same color. This type of dithering works equally well for color and gray-scale animations. In gray-scale animations, Color Shade Dithering simulates 31 shades of gray rather than the standard 16. For color images, Color Shade Dithering allows you to get many more shades of each color used in the IIGS Color palette. This type of dithering will only be used to make slight color shade adjustments. FLI Convert will never use Color Shade Dithering to try to dither black and white together to make gray. It is always used for much more subtle effects. Such as mixing two shades of orange to make a third shade of orange.

The second type of dithering that FLI Convert supports is called Color Hue Dithering. When Color Hue Dithering is turned on, FLI Convert will try to mix two different colors to make a third color. Unlike Color Shade Dithering, Color Hue Dithering may dither two very different colors together such as red and blue to make purple. Color Hue Dithering is usually not as subtle as Color Shade Dithering. Also, Color Hue Dithering is totally useless for gray-scale animations since gray-scale means there are no hues. Color Shade Dithering is still performed when Color Hue Dithering is turned on.

Color Separation

Color separation is the name of the process that FLI Convert uses to throw out very similar colors to reduce the number of colors used in an animation. This is what is used to determine the number of "distinct" colors. After FLI

Convert counts the number of colors used in an animation, it updates the "Colors:" field to show exactly how many different colors were used. FLI Convert then performs the color separation to throw out similar colors. The number of colors that are left after the color separation is displayed in the "Distinct Colors:" field. The Color Separation setting in the FLI window lets you adjust how similar colors need to be before they are thrown out. If you set this value to Off, then only colors that are exactly the same will be thrown out. A setting of 1 causes colors that are very, very similar to be thrown out but keeps colors that are only somewhat similar. A setting of 15 causes color that are even remotely similar to be thrown out.

If you have an animation that uses many, many shades of the same color (such as a big red ball with several shades of red) and there are not many other object in the animation, you may want to turn color separation off. This will tell FLI Convert to keep as many shades of red as possible to make the ball look as smooth as possible. However, if this animation also had some other objects in the background that were all different colors, you may want to set the color separation value higher so that the whole color palette isn't taken up by shades of red. A higher color separation value will cause some similar shades of red to be thrown out to make room for some of the colors used by the other objects in the animation.

If the Distinct Colors field contains a value greater than 150, this could be a sign that the color separation is set too low and should be raised to a higher number. If there are still 150 colors left after removing similar colors, FLI Convert is going to have a hard time deciding which of those 150 colors to keep. Raising the color separation will lower the number of distinct colors and help FLI Convert decide which 16 colors to keep.

Conversely, if the number of colors is very high but the number of distinct colors is less than 20 or 25, you may have the color separation set too high. Lowering the color separation will raise the number of distinct colors and give FLI Convert a few more shades of color to work with.

After making a change to the Color Separation setting, you can select "Count Colors" from the FLI menu to have FLI Convert update the Distinct Colors field so that you can immediately see exactly how many distinct colors you have based on the new color separation setting. You

can also use the "View Color Matching" option to see exactly what colors will be used in the IIGS color palette based on your current setting.

Viewing Color Palettes

The "View FLI Color Palette" option in the FLI menu will let you see all 256 colors used in the FLI. Simply select the menu item, and the screen will be filled with a 16 x 16 grid of all 256 colors. Because the IIGS can only display 16 shades of a color, the display of the FLI colors can not be 100% accurate. A FLI may contain 64 different shades of blue but FLI Convert will display them as 16 shades of blue because that is the best the IIGS video hardware is capable of at one time. As a result, you may see several colors in the FLI color palette that look identical. They are most likely slightly different shades that cannot be distinguished on the IIGS.

The "View Color Matching" option in the FLI menu will let you see which 16 colors of the 256 were selected for the IIGS color palette. You will also be able to see which IIGS color will be used for each FLI color. When you select the "View Color Matching" option, the screen is filled with 16 rows of color. The bottom most row contains the 16 colors selected for the IIGS color palette. The top 15 rows show the colors in the FLI color palette and the IIGS colors that will be substituted for them. The top half of each row contains 16 colors from the FLI color palette. The bottom half of each line contains the 16 IIGS colors that will be substituted for each FLI color. If all goes well, the top and bottom half of each row will be almost exactly the same. This would indicate the FLI Convert was able to match all the FLI Colors to suitable IIGS colors. If the top and bottom half of the rows don't really match then FLI Convert was not able to do a very good job of selecting a color table that accurately represents all the colors in the FLI color palette.

When the FLI color palette is displayed at the same time as the IIGS color palette, as it is when viewing color matching, only 240 of the FLI colors can be displayed at once. To see the last 16 entries in the FLI color palette you need to press the down arrow to scroll down to the bottom of the FLI color palette. You can use the up arrow to scroll back up to the first 240 color palette entries.

Seeing exactly which colors FLI Convert selected to be substituted for each FLI color palette entry should help you decide how to adjust the color conversion options. If the FLI contains several

shades of red but they are all being matched to the same shade or red in the IIGS color palette, then you may want to decrease the color separation value so that FLI Convert will select more than one shade of red for the IIGS color palette. On the other hand, if the IIGS color palette is filled with several shades of red and there are other colors in the FLI color palette without suitable IIGS colors to match to, you may want to increase the color separation to get a better variety of color in the IIGS color palette.

Hopefully this information will help you get the best results possible. Unfortunately, fine tuning the color conversion settings can be more of an art than a science. Some times you just have to fiddle with the settings a little and use the Preview option a lot.

Twilight II's Movie Theater

One of the things I like to do most with all the PaintWorks animations I have is to use them as screen savers. Twilight II has a very useful screen saver module called Movie Theater. This module allows you to use any PaintWorks animation, including the ones created by FLI Convert, as a screen saver. Unfortunately, there is one bug in Movie Theater you should know about. (Don't worry, it's very easy to avoid). Movie Theater does not properly handle PaintWorks animations that have a speed setting of zero (0). Movie Theater accidentally handles a speed of zero as if it were a speed of 65,537. This means that there will be more than an 18 minute delay between each frame in the animation. All you need to do to avoid this problem is to make sure you don't create any PaintWorks Animations with a speed of zero. Just change the speed to one (1) and it will work great. I have found that a lot of FLIs use a speed of zero, so make sure that you watch out for this. I have also found some PaintWorks Animations created by other programs that have a speed of zero. To get these animation to work with Twilight II's Movie Theater, you can open them with FLI Convert, change the speed to 1, and then pick Save from the File menu.

So, now that we've seen what FLI Convert can do, let's take a detailed look at how it actually does it. To follow along, you'll want to break out the FLI Convert source code that is on your GS+ Disk.

Program Structure

The FLIConvert.C file contains all the functions necessary to create the user interface to FLI Convert. It handles responding to menu item selections,

creating windows, responding to button presses and other events. It coordinates all the functions used to read and display FLI files and write and display PaintWorks animation files. Most of this code is very common event handling code and should be rather straight-forward.

FLI.C contains all the functions used to read FLI files and manipulate the FLI data structures.

C2.C contains all the functions used to write animations in the PaintWorks Animation file format.

PlayC2.ASM contains the function used to play PaintWorks animations.

FLIConAux.C contains a few simple functions used to display the FLI color palette and the color matching.

Deciphering the FLI File Format

The FLI file format is a little complicated. It is made up of a hierarchy of structures including a file header, frames, chunks, and packets. The **FLI.Format.Specs** file on your **GS+** Disk contains detailed descriptions of these data structures and how they relate to each other. You can find C type definitions for these structures in the **FLI.H** file.

The **ProcessFrames** function in **FLI.C** does most of the work involved in reading FLI files. After reading the FLI header, a for loop is entered and repeated once for every frame in the file. Inside this loop, the frame header is read and another loop is entered to handle each chunk in the frame. A switch statement is used to handle each chunk type properly. The chunks are processed differently depending on the current value of the

FLIProcessingMode variable. Sometimes **ProcessFrames** is called to simply read the FLI's color palette, or to count the number of occurrences of each color, or it is called to actually display the pixels in each frame. **CountColorsInFirstFrame** and **CountAllColorsInChunk** are used to count the occurrences of colors. **PlotPixelsInChunk** is used to actually display the animation. If **ProcessFrames** is being used to create a PaintWorks animation, **SaveFrameC2** is called after displaying all the chunks in a frame.

Building Color Palettes

Because FLIs use up to 256 colors on screen at once and the IIGS can only display up to 16 colors on screen at once, a fair amount of color palette manipulation is required to display FLIs on a IIGS. This process involves counting all the colors used in the FLI, selecting the 16 colors used most often to make a color palette for the IIGS, and deciding which of the 16 IIGS colors will be used in place of each of the 256 colors in the FLI.

To count the colors, **FLIProcessingMode** is set to **DM_COUNT_FIRST** or **DM_COUNT_ALL** depending on the setting of the **Color** pop-up setting. **ProcessFrames** is then called to walk the FLI file structure and count all the colors it encounters.

A process called "color separation" is then used to eliminate colors that are very similar to each other so that we don't end up selecting 16 very similar colors for the IIGS Color Palette when we would be better off with a variety of colors. The **MakeColorTable2** function compares

each of the 256 colors in the FLI Color Palette to every other color in the FLI Palette and calculates the difference between the two colors. If the difference is less than the minimum color difference, then the count of occurrences of one of the two colors is transferred to the other. This essentially combines the two colors into one.

The **qsort** function is used to find the 16 colors that are used the most. The color palette is sorted in descending order by color count, and the top 16 colors are used to create a IIGS color palette. The IIGS color palette is then sorted in ascending order by color intensity to make black the first color and white the last color.

The last step in this process is called "color matching." The **MatchColors** function loops through all 256 colors in the FLI and compares the FLI color to each of the 16 colors in the IIGS color palette. The IIGS color that matches the FLI color most closely will be used every time the FLI bit map specifies the FLI color.

PaintWorks Animation File Format

To understand the PaintWorks Animation format you need to have a good understanding of how IIGS Super High Resolution (SHR) graphics work, and how they are stored in memory. Refer to chapter 4, *The Video Display*, in the *Apple IIGS Hardware Reference* if you are not familiar with SHR memory.

The PaintWorks animation file format is very simple, especially compared to the FLI file format. It basically consists of three blocks of data. The first is a 32K block of data that is an exact copy of the

Incredible 50% Off Deal! through 7/31/94

Spectrum™ ————— **\$64.98**
Telecom (requires System 6). \$129.95

Express™ ————— **\$19.98**
Software printer spooler. \$39.95

Kangaroo™ ————— **\$24.98**
Standard File utility. \$49.95

TransProg III™ ————— **\$19.98**
Integrated program launcher. \$39.95

Super Menu Pack™ — **\$14.98**
Key caps/font menu utility. \$29.95

GraphicWriter III™ — **\$74.98**
Desktop publishing. \$149.95

All products require an Apple IIGS
with 1.25mb memory.
Hard drive recommended.

Seven Hills
SOFTWARE

2310 Oxford Road
Tallahassee, FL 32304-3930

Checks-MOs-Visa-Mastercard-Discover

America Online, GENie: *SevenHills*
CompuServe: 75300,1743
Internet: *sevenhills@aol.com*

(904) 575-0566, M-F, 9am-5pm ET
FAX (904) 575-2015 anytime

SuperConvert™ ————— **\$19.98**
Graphics converter. \$39.95

ShoeBox™ ————— **\$29.98**
Home information organizer. \$59.95

The Manager™ ————— **\$34.98**
Like the Mac's MultiFinder. \$69.95

Font Factory GS™ — **\$19.98**
Bitmap font editor. \$39.95

Independence™ ————— **\$19.98**
HP printer drivers. \$39.95

Drive Cleaner GS™ — **\$17.48**
3.5" disk drive cleaner. \$34.95

first frame of the animation. This frame is stored in the file in the exact same format as it is stored in SHR memory when displayed on the screen. That is, 32,000 bytes of pixel data, 200 scan-line control bytes, 56 unused bytes, and 16 color tables each 32 bytes in size.

The next block is a 12 byte structure with a little information about the rest of the file. The structure looks like this:

```
typedef struct {
    unsigned long FileSize;
    unsigned Short Speed;
    unsigned Short Reserved;
    unsigned long Offset;
} PAINTWORKS_HEADER
```

FileSize specifies how many bytes of data are in the last block. Speed is the number of ticks (60ths of a second) there should be between each frame during playback. Offset specifies how many bytes there are before the beginning of the last block including the offset field. This value should always be four (4).

The last block contains the data for all the frames. Rather than storing the entire bit map for each frame, the PaintWorks animation format saves only the pixels that are different than the last frame. The last block of the file contains a series of word pairs that specify which pixels changed and what their new values are. The first word in the pair is an offset into SHR memory of a word (4 pixels) of data that needs to be changed. The second word is the new data that will be stored at the location specified by the first word. These word pairs are repeated until every pixel on the screen that needs changing in the frame has been specified. The end of a frame is marked by an offset word value of zero (0). The next frame follows immediately after the last word pair of the previous frame. The format continues for every frame in the animation.

Saving PaintWorks Animations

The source code in the C2.C file handles

the creation of PaintWorks animation (file type \$C2) files. To create a PaintWorks animation file, the InitC2 function is first called to initialize the PaintWorks animation file creation routines. The PAINTWORKS HEADER structure (shown above) is initialized with information about the animation and three buffers are allocated that will be used to store the frames for the FLI and PaintWorks animations files. Next, the FLI file is displayed the exact same way it played for preview. After displaying each frame of the FLI, the SaveFrameC2 function is called with a pointer to the SHR memory that contains the frame to be saved. The first time SaveFrameC2 is called, the whole frame and the PAINTWORKS HEADER structure are written to the PaintWorks animation file. A copy of the first frame is also saved in the FirstFrameC2H buffer to be used later to write the necessary information for looping from the last frame the first frame. For every other time SaveFrameC2 is called the OutputSHRDifferences function is called with a pointer to the current frame and the previous frame.

OutputSHRDifferences compares every word in the buffer pointed to by OldFrameC2 with the words in the buffer pointed to by NewFrameC2. Every time a changed word is found, the offset of the word and the new value of the word are written to the DeltaBuffer. When all the changes have been found, the DeltaBuffer is written to disk and the FileSize value of the PAINTWORKS_HEADER structure is incremented by the size of the data in the DeltaBuffer. After the last frame is displayed, the DoneC2 function is called. This function first calls OutputSHRDifferences with pointers to the buffers containing the last and first frames. This puts one last frame in the PaintWorks animation file that creates a loop back to the first frame. Finally, the DoneC2 function goes back to the second block of the file and rewrites

the PAINTWORKS HEADER structure with the updated FileSize value.

Playing PaintWorks Animations

For maximum performance, the function that plays PaintWorks animations is written entirely in assembly language. The function is called PlayC2Anim and is located in the PlayC2.ASM file. This function is really very simple. To display the first frame, the first 32K bytes of the file is simply copied into SHR memory with the BlockMove tool call. The wait function is then called to create the proper delay between the first and second frame. Next, the main loop is entered to display each frame of the animation. First, the offset word is loaded. If it doesn't equal 0 it is placed in the x index register and the next word is loaded containing the new pixel data. This word is then stored in SHR memory using the offset in the x register. Next the BRA instruction is used to loop back to the Loop label and load the next offset word. If the offset is equal to 0 then wait is called to create the proper delay between frames. Next, the current offset into the animation data is compared to the total size of the animation data. If the current offset is greater than the total size, then the end of the animation has been reached and the offset into the animation data is reset to the beginning of the animation to loop back to the first frame. The animation will continue to repeat in the loop over and over until any call to the wait function returns a non-zero value indicating that the user has pressed a key to stop the playback.

So, there you have it. Everything you could want to know about FLIs, PaintWorks animations, and how FLI Convert works with them. If you have any questions, be sure to write to me in care of GS+ Magazine, and I'll see if I can get you an answer. Until then, I hope you enjoy FLI Convert! GS+

Moving?

Well, don't forget to tell us! The Post Office does not normally forward Third-Class mail (they simply destroy it!), and we can't afford to replace magazines that were lost because a subscriber forgot to send us a change of address! If you miss an issue, we will extend your subscription, but you will have to buy the missed issue as a back issue! So, to avoid this hassle, send us a change of address as soon as you know your new address!

Simply remove your mailing label from a previous issue of GS+ Magazine, affix it to a change of address form (available at your local Post Office), fill in your new address, and send it to us at:

GS+ Subscription Services
P. O. Box 15366
Chattanooga, TN 37415-0366

APPLE II PERIPHERALS - UPGRADES - SERVICE PARTS - HARD DISKS SCHOOL & UNIVERSITY P.O.s ACCEPTED! - SCHOOL QTY. DISCOUNTS AVAILABLE*

RAM CHIPS

RAM Chips for your...
Expanding your... Apple IIGs Memory Card, AE GS-RAM, AE RAMWorks, Super Expander & Compatibles
768K Exp. Set (24 41256).....\$24.00
41256.....\$1.25
Expanding Your GS Juice+, S&S AEGS RAM + / II, CV Tech or Chinook 4000
41024 (1 Meg)\$3.00
Bank of 8.....\$24.00
Expanding Your AE GS-RAM Ultra or RAMPak4 GS
44256 (256x4)\$3.00
Bank of 8.....\$24.00
Expanding Your OctoRAM, GS Sauce, Q-RAM GS, Pacemark 4 Meg RAM Card
1 Meg SIMM\$24.00
AE GS RAM III 1 x 4 Zip.....\$25.00

PC Transporter

101 Keyboard for PCT\$29.00
PCT IBM Keyboard Cable.....\$7.00
MS DOS 5.0 Package.....\$49.00
3.5" 720K Ad-on Drive.....\$49.00
4464 Zip /PCT Mem. Exp.\$2. ea
TransDrive 5.25/3.5" Combo\$89.00

Motherboards/Service Parts

IIGs Power Supply Exch.....\$59.00
IIGs Power Supply New.....\$79.00
IIGs Keyboard Exchange.....\$45.00
IIGs Keyboard Refurb.....\$55.00
IIGs Keyboard NEW.....\$69.00
Apple //e or //c Case.....\$10.00
IIGs Case.....\$25.00
ROM 01 Motherboard Exch.....\$149.00
ROM 03 Motherboard Exch.....\$199.00
Upgrade: ROM 01 to ROM 03
W/Working ROM 01.....\$125.00
W/NonWorking ROM 01.....\$175.00

Refurbished Printers

ImageWriter I.....\$129.00
ImageWriter I (Wide).....\$139.00
ImageWriter II.....\$199.00
ImageWriter II Sheet Feeder..\$79.00
9 Pin Parallel Printer.....\$75.00
IW-I/II Ribbons.....6 for ..\$12.00
Print Heads
IW-I (Exch Only).....\$35.00
IW-II (Exch. Only).....\$69.00
Epson 9 Pin Heads (Most).....\$59.00
We rebuilt print heads

Accessories

IIGs Internal Fan.....\$14.95
9 Pin Joystick, IIGs.....\$12.95
Mouse Pad (Rubber/Staticfree).....\$2.95

Cables & Switch Boxes

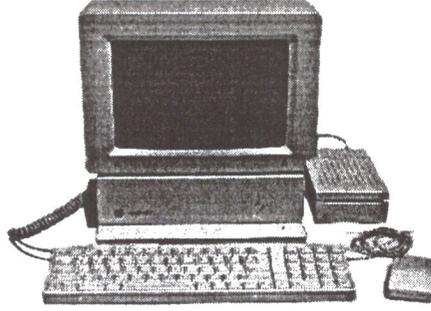
METAL, GBBS Sysops. HST/v.32bis hardware hndshk cables! (Specify).....\$14.95
19 Pin Drive Converter.....\$14.00
IIGs, e.c to ImageWriter I/II.....\$7.95
Fullnet Connector.....\$14.95
Switch Boxes.....\$19.00 - \$29.00
ADB Replacement cable.....\$5.95
SCSI Cable, 25-50 or 50-50.....\$9.95
Power Cord (US or Intl).....\$3.00

PERIPHERAL CARDS

Zip GS 7Mhz /32K Cache ..\$159.00
Zip Chip 8 Mhz (//e, //c).....\$139.00
AEC SCSI Card.....\$59.00

Manuals & Software

Misc. Software. Games, Utils, More. \$5.95 to \$19.95. Ask For List.
Apple Manuals Avail. Ask For List



System Includes:
All New
AppleColor RGB Monitor
External Apple 3.5" Drive • Keyboard
Mouse • 1024K System 6.0.1 ROM 03
\$599.00

More Apple IIGs Accessories

Cordless ADB Mouse 6 foot range. Works Great!.....\$59.00
Apple ADB Mouse II (New) \$69.00..Refrb ADB Mouse \$49.00
Apple IIGs System 6.0.1 Disk/Manual Set.....\$35.00
Apple IIGs System Transport Case (Full System).....\$19.95

Modems

USR 1200 Modem (NEW).....\$19.00
Hayes SM1200 (Refurb).....\$29.00
2400 External (New).....\$49.00

Cable/Modem Combo.. add\$5.00
Modems include ProTerm 3 Demo.

Bulk Disks

3.5" DSDD Disks 50 for\$14.50
3.5" DSHD Disks 10 for\$3.90



SoundMeister
.....
Stereo/Digitizer for
Your Apple IIGs!
\$59.00

Alltech Electronics Co.

602 Garrison St. Oceanside, CA 92054
9-6 PM Mon.-Fri. - 9-5 PM Sat.

Orders Only:

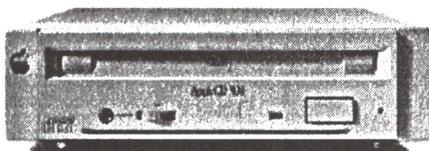


All other Inquiries: & Tech Support: 619/721-7733

800/995-7773

Fax (School PO's, Quotes, Info.): 619/721-2823

BBS (FutureNet #30) 300-14.4 v.32: 619/721-0705



DiscQUEST..... \$79.00

DiscQuest: CD ROM for your IIGs!
Sequential Systems CD ROM Software
DiscQuest (Incl. Family Doctor).....\$79.00
DiscQuest & CD 150.....\$265.00
DiscQuest, CD 150 & RAMFast.....\$395.00
Apple CD-ROM 150 External.....\$189.00

Monitors

AppleColor RGB, IIGs (Refurb).....\$169.00
AppleColor RGB IIGs (NEW).....\$225.00
Apple Color Comp or Color 100 (exch) ..\$109.00
Composite Green (Refurb).....\$35.00
Composite Green "Box" Monitor 9".....\$39.00
Apple Monitor II or III (Refurb).....\$49.00

SEQUENTIAL SYSTEMS

Sequential Systems RAM-GS 4096K.....\$135.00
Meg 80 Z (1 Meg, Apple //e).....\$59.00
Q:Talk LTO (AppleTalk & 32K for IW-II).....\$69.00
RAM //c with 1024K (//c or //c+).....\$119.00
bit Mouse card \$49.00.....with mouse.....\$65.00
SuperCOMM \$54.00.....SuperCOMM //c.....\$45.00

RAM Cards

Apple 1 Meg Memory Exp. with 1 Meg.....\$39.00
GS SuperRAM 4 Meg with 1 Meg.....\$74.00
GS Super RAM w/2 Meg \$99.00. 4 Meg \$129.00

SCSI Hard Disks

40 Meg External w/AEC Card (Refurb).....\$159.
40 Meg External Drive (Quantum).....\$139.
80 Meg Ext.....\$189.00 w/Apl SCSI.....\$295.
SyQuest 44.....\$259.00 SyQuest 88.....\$359.
External SCSI Drive Cases
Case & P/S for 3.5" or 5.25" HH.....\$45.
Case & P/S for Full Height 5.25" or 2 HH.....\$65.
3.5" Drive footprint External Case for QELS Type Drive. 2 50 Pin Connectors and ID Selector. Gets Power from SmartPort.....\$19.



With Allison™
Digitizing software
Instantly grab's
from any video
source. Upgrades
available. Call.

VisionPlusEnhanced IIGs \$179.00

Apple FDHD \$199.00

Disk Drives

5.25" Generic Full Height w/19 Pin.....\$69.00
5.25" Apple Disk II (Refurb) add \$10.00 for above.
Apple DuoDisk 5.25" Exch.....\$129.00
Apple DuoDisk 5.25" (Refurb).....\$159.00
AppleDisk 3.5" (Platinum).....\$165.00
AppleDisk 5.25" (Refurb).....\$165.00
Apple 3.5" 800K Drive Service Exchange \$85.00

CA Residents please add 7.75% Sales Tax • COD Orders, Cash/Certified Funds Only • Minimum Shipping & Handling Charge \$5.00 (This covers most orders) • Some products are refurbished products and carry a 120 day warranty. • New products are covered by a one year warranty. All repairs are warranted for 120 days. • Some RAM Chips may be new but pulled, all RAM has a one year warranty. • NO Returns without an RMA number, please call before shipping anything back to us. • A 20% Restocking Fee is applied to all returns due to incompatibility or unneeded product or Products purchased in error. Please be sure what computer you have before placing an order. • Educational P.O.s Accepted, Others on AOC only. • Prices and availability subject to change at any time without any notification whatsoever. • Discounts on certain merchandise only and min. Qty Applies.

Rumors Wishes & Blatant Lies

By Prof. G.S. Gumby

Taps

Unless you live in a tiny metal drum (which would be like a palace compared to my office), you know by now that former U.S. President Richard M. Nixon passed away on April 22. But, you might not know about the Apple II companies that also passed away in the last few weeks.

Applied Engineering was the first "Apple II" company to go this year. But, the boss has told me not to say anything about it. So, for more information, I've been instructed to tell you to read this issue's "Writer's Block" column.

The Apple Catalog was this year's second Apple II related casualty. It's not clear why Apple pulled the plug on this (apparently) popular venture, but some speculate it's because it really *wasn't* all that popular, especially with Apple's resellers! It seems that they simply didn't like competing with Apple at the "point of sale" level, and threatened to stop carrying Apple products. Whatever the reason, when you call the Apple Catalog number now, you get a recording telling you that they "are not offering Apple Catalog services at this time." This is a real loss for the Apple II community, because the Apple Catalog was one of the last places to order manuals for Apple's Apple II products.

Finally, it would appear that TMS Peripherals has also gone out of business. I say that it "appears" they have gone out of business, because I simply can't confirm it. (However, now that their phone have been disconnected, it seems to be a fairly inescapable conclusion.) However, I have been able to piece together some of the events leading up to their demise. (Note however, that some of these details come from the second-hand stories of former TMS employees, so they may not be 100% accurate.)

The problems at TMS started a couple of months back, when their offices were burglarized and most of their equipment and stock was taken. (It was after this robbery that they decided not to advertise on the back cover of our last issue.) Soon after that, two of the three owners decided to leave TMS in what has been described as a "friendly parting." About this time, rumors began to fly that TMS had changed its name to "PowerCity" to capitalize on the upcoming release of the PowerPC Macintoshes. However, I contacted PowerCity, and found out that they have *no* relation to TMS Peripherals,

except for the fact that one of their employees used to work for TMS.

So, the bottom line is, TMS Peripherals is gone and, as of this writing, no one has come forward to service or support the customers they left behind.

Now, Good News!

Across the pond, over in merr-ie old-e England-e, a group of IIGS enthusiasts has gotten together to form The Phoenix Project. As you might guess from the name, their mission is to "resurrect" the IIGS by whatever means they can. And, unlike some previous efforts, they seem to have a plan!

First, they're going to attempt to complete the Foundation resource editor. Towards that end, they've acquired the source code for Foundation from the folks at Lunar Productions and are working on it even as I type this. (Of course, the Foundation source code has been uploaded to various online services, so you could finish it yourself if you really wanted to.)

And after that, who knows? The Phoenix Project has put forth several ideas for projects (IIGS file server software, an update to Genesys, etc.), but they need time and volunteers to make them happen. And, since The Phoenix Project is a volunteer organization, that means they need people like *you* to lend a hand. If you want to volunteer your services, you can send an e-mail message to Richard King, the coordinator of the Phoenix project at "tombo@mail.on-line.co.uk".

SoundMeister Redux

A couple of months ago, when ECON Technologies announced that they were going to be discontinuing their line of hard drives, several readers called up asking what was going to happen to ECON's popular SoundMeister digitizer board. Well, it can now be told that ECON has signed a deal with AllTech Electronics to take over manufacturing and sales of the SoundMeister board and the Digital Session software. And, according to Tony Diaz at AllTech, the price from AllTech will be even *lower* than the low price ECON was selling the board for! So, if you need a sound digitizer card for your IIGS, give AllTech a call at (619) 721-7733.

Nexus Fer Free!

Nexus, one of the first hypermedia systems ever available for the IIGS, has just become freeware! With Nexus, you

can link text and graphics files together, without changing the files at all! (It's important to note that the original publisher still retains all rights to Nexus, and that if you want to distribute it with a commercial product, you must receive written permission first.) To find a copy, hop on your favorite online service. To get more information on distributing Nexus with your product, contact:

LinksWare Corporation
812 19th St.
Pacific Grove, CA 93950

New Spreadsheet?

By the time you read this, a new *IIGS-specific* spreadsheet should be available from the Byte Works! Among the planned features are 3D spreadsheets, support for all spreadsheet functions in AppleWorks Classic and AppleWorks GS, charting, different fonts for each cell, and a little thing called "publish and subscribe." For more information, contact the Byte Works at (505) 898-8183.

So Blue

SHH Systeme of Germany (makers of the Turbo IDE hard disk controller) are reported to be working on a new disk controller card, called BlueDisk, that will allow you to hook inexpensive IBM PC 5.25-inch and 3.5-inch drives to your IIGS. This card, along with Peter Watson's shareware MS-DOS utilities, will allow you to read and *write* data on an MS-DOS disk without having to buy a PC Transporter, or an Apple II SuperDrive controller. The card is still in development at this time, but you can get more information on it by sending e-mail to SHH Systeme at "J.LANGE7@genie.geis.com" or write to:

SHH Systeme
Dipl. Ing. Joachim Lange
Bergstrasse 95
82131 Stockdorf
Germany

New Name Game

You might not notice it otherwise, but LRO Computer Sales has changed their name to Other World Computing. So, be sure to make a note of it.

Got a IIGS rumor, wish or blatant lie? Here's my address, the rest is up to you!

GS+ Rumors
P. O. Box 15366
Chattanooga, TN 37415-0366

GS+

How to Use Your GS+ Disk

The first thing you need to do is make a **backup copy of your GS+ Disk with the Finder!!!** Do *not* make your backup on your hard disk! Instead, copy the GS+ Disk to another 3.5-inch disk (this is *very* important). Next, put the original in a safe place. If you are having a problem making a backup copy, give us a call at (615) 843-3988. If your disk is damaged, let us know, and we'll get a new one to you as soon as possible.

Installing The Software

To install the software on this issue's GS+ Disk, start up your computer using System Software v6.0.1 or later. (Note that all of the programs on this issue's disk [except EGOed lite] *require* System 6.0.1!) Next, place your *backup* copy of the GS+ Disk in a drive. (You *did* make a backup didn't you?) Now run the Installer program that is on your backup GS+ Disk. (From the Finder, just double-click on the Installer icon.) *It is extremely important that you use the Installer that is on your backup GS+ Disk! Do not use any other copy of the Installer!*

When the Installer window appears, select the item you want to install from the list on the left-hand side of the window, and the disk you want to install it on from the list on the right-hand side of the window. Then click on the Install button. For more information on using the Installer, refer to your IIGS owner's manual.

Before you attempt to use your backup GS+ Disk, please take a few minutes to read the **a.Read.Me** file for any last minute corrections or information. If you do not already have our EGOed lite text editor installed in your system, you can use the Teach application supplied with System Software v6.0 to read this file.

Installing EGOed lite

The following is a detailed example of how to install EGOed lite. The other programs are installed in a similar manner.

- Start up your IIGS with System Software v6.0 or later—the version of EGOed lite that is on this GS+ Disk *requires* System 6! (Your GS+ Disk is *not* a startup disk, so don't try starting your computer with it.)
- Insert your backup copy of the GS+ Disk into a drive and run the Installer program that is on your backup GS+ Disk. It is *very, very* important that you run the Installer that is on your backup

GS+ Disk and *not* some other copy of the Installer.

- When the Installer finishes loading, click on the Disk button on the right-hand side of the Installer window until your startup disk appears. (If you only have one 3.5-inch disk drive, you will have to remove the backup GS+ Disk from the drive and replace it with your startup disk. You should also refer to the "Making Room" section below for hints on how to free up room on your boot disk.)

Please Remember . . .

The contents of the GS+ Disk are not public domain or shareware! We depend on your honesty to stay in business. Please do not give away copies of the GS+ Disk or any of the programs on it. If you do, we will not be able to stay in business. It really is that simple!

- On the left-hand side of the Installer window, you will see a list of the items on the backup GS+ Disk. One of the items in this list should be "EGOed lite." (If EGOed lite is *not* in this list, quit the Installer and begin again. Be sure that you are running the copy of the Installer that is on your backup GS+ Disk!) Once you see the EGOed lite item, click the mouse on it so that it becomes highlighted.
- Click the mouse on the Install button in the middle of the Installer window. The Installer will then install EGOed lite on your startup disk. If you only have one 3.5-inch disk drive, you may have to switch disks several times. Just insert each disk as the Installer asks for it.
- When the Installer has finished, click on the Quit button in the middle of the Installer window. This should cause your IIGS to restart.
- When your IIGS finishes restarting, pull down the Apple menu and select EGOed lite (note that you have to be in a

desktop program like the Finder to have access to the Apple menu).

- When it finishes loading, notice that EGOed lite has its own menu bar. Select Open from the *EGOed lite* File menu and then put your GS+ Disk in a drive. You should see a list of the files and folders on the GS+ Disk.
- Open the **Documentation** folder on your backup GS+ Disk and then open the file **EGOed.lite.Docs**. This file contains complete documentation on how to use EGOed lite. *Please take a few minutes to read this documentation.*

Making Room

If you do not have a hard drive, you will probably have to remove some files from your startup disk to make room for the New Desk Accessories, control panels, and other system files on your GS+ Disk.

Towards that end, we have prepared the following list of "expendable" files that you can "safely" remove from your System Software v6.0.1 startup disk to free up some space. (We've put quotes around "expendable" and "safely" because almost *all* of the files in the IIGS System Software have some sort of use! The files listed here are the ones that are the "least" useful for a specified hardware setup.)

Be sure that you *never* delete *any* files from your original System Software boot disk! Always work on a backup copy!

System Software v6.0.1

If you use the System 6.0.1 :Install disk to create a minimal, 800K, System 6.0.1 boot disk, that disk will have 26K free when the installation is finished.

It must be noted that *all* of the files on this disk are *very* important and the files that you can *safely* remove depend, for the most part, on your hardware setup. So, please read these instructions carefully before removing *any* files.

The first two files you can delete depend on what you will be doing with your IIGS. If you will not be running AppleSoft BASIC programs, you can remove the file **BASIC.System** (11K) from the root directory of the disk. If you will not be running ProDOS 8 software, you can remove ***:System:P8** (18K).

If you do not care what time it is, you can delete the following file:

***:System:CDevs:Time** (10K)

After that, the files that you can safely remove depend on your *hardware setup*.

If you have a ROM 01 IIGS, you may delete the file:

*:System:System.Setup:TS3 (42K)

If you have a ROM 03 IIGS, you may delete the following file:

*:System:System.Setup:TS2 (37K)

If you do *not* have a 5.25-inch drive, you may delete the following 8K file:

*:System:Drivers:AppleDisk5.25

If you do *not* have a printer, you may delete the following file:

*:System:CDevs:Printer (5K)

Finally, if you have deleted all control panels, and you won't be installing any control panels from the GS+ Disk, you can also delete the 18K file:

*:System:Desk.Accs:ControlPanel

Removing some or all of these files will give you ample room (up to 138K on a ROM 01 IIGS and up to 133K on a ROM 03 IIGS) on your startup disk to install EGOed lite or any of the other system utilities from your backup GS+ Disk.

What Is EGOed lite?

EGOed lite is a New Desk Accessory (NDA) text editor that we provide in each issue of GS+ Magazine.

When you install EGOed lite on your startup disk, you can use it to edit and print ASCII text, Teach, AppleWorks Classic and AppleWorks GS word processor files from inside any desktop program that properly supports NDAs.

To use EGOed lite, you must install it on a IIGS System Software v6.0 (or later) startup disk with at least 40K of free space.

Note: You will *not* be able to print from EGOed lite or any other desktop program when using an 800K, System 6.0 boot disk. (There isn't enough room for all of the required drivers and control panels.)

If you want to save even *more* space, you might want to consider using Autopilot (from GS+ V4.N1) as a replacement program launcher. With Autopilot installed on the minimal System 6 boot disk, initial free space goes up from 26K to 163K! You can then use Autopilot to autolaunch the Finder from a second 3.5-inch disk drive and still have plenty of room on your boot disk for lots of system extensions. For more information on Autopilot, refer to the "Autopilot v2.0" article in GS+ V4.N1 or give us a call.

Self-Extracting Archive

We use GS-ShrinkIt v1.1 to compress the *source code* and related files on the GS+ Disk into a *self-extracting archive*. To extract the files from the archive, simply double-click on the GSP.V5.N5.SEA program on your backup GS+ Disk. *You do not need to have a copy GS-ShrinkIt in order to use any of the programs or other materials on this GS+ Disk!* However, you will gain better control over

IMPORTANT!

Use scissors or a knife to open disk bag!
Do not attempt to pull bag away from magazine!

the files you wish to extract if you have GS-ShrinkIt v1.1. If you do not have GS-ShrinkIt v1.1 and you would like a copy, check with your local user group or give us a call here at *GS+* Magazine and we will try and help you locate a copy.

What's On The Disk

There are nine items in the root directory of this disk:

a. Read.Me

A lot can happen from the time we send this magazine to the printer and the time we get ready to mail them out. If anything does happen, we will put everything we can find in this file. Please read this file before using the *GS+* Disk.

Documentation

This folder contains the EGOed lite documentation, the *GS+* Glossary, and *GS+* back issue information. The EGOed lite documentation is a Teach file which can be read using EGOed lite. The *GS+* Glossary file is a text file containing all of the terms defined in past installments of the "*GS+* Glossary". The back issue information is also in a text file.

FLIs

This folder contains some sample .FLI animation files. You can view and convert the .FLI files to PaintWorks animation format using the FLI Convert program.

GSP.V5.N5.SEA

This is a self-extracting archive (SEA) containing the source code and related files for all the programs contained on this *GS+* Disk. The archive also contains the Miscellaneous Library. Technical information, such as the Miscellaneous Library documentation is supplied in the archive as well. The complete List Manager demo program, its source code, and some technical notes relating to the List Manager are enclosed in the archive, too. To extract the files from the archive, simply double-click on this file from the Finder. You will then be presented with a dialog asking you where you want the files extracted to. Note that if you try to extract *all* of the files from this archive at one time, they will *not* fit on an 800K disk!

Icons

This folder contains Finder icons used by the various programs on the *GS+* Disk.

Installer

This is the Apple IIGS Installer. The installer requires System Software v6.0 or later. Run it to install the other programs on this issue's disk. For more information on using the Installer, be sure to read the example on the previous pages, and refer to your owner's manual.

Programs

This folder contains the EGOed lite, FLI

Convert, and More Sound programs. Use the Installer provided on your backup *GS+* Disk to automate the installation of these files. EGOed lite requires System 6 to operate. All the other programs on this disk require System 6.0.1 to operate. Note that the List Manager demo program is contained in the self-extracting archive and not in the Programs folder.

Scripts

This folder contains all of the scripts that are used by the Installer to install the files from this *GS+* Disk.

Talk.To.GSPlus

This folder contains our feedback form, a troubleshooting guide, a problem form, and our writer's guide. The feedback form is a plain ASCII text file. Fill it out, and let us know what you thought of this issue. The troubleshooting guide contains tips on how to resolve some of the more common problems you may experience while trying to use the programs on your *GS+* Disk. If you are having a problem, *please* read this file before you go to all the trouble of filling out a problem form! But, if the troubleshooting tips don't help, *please* fill out the problem form and send it to us! This is a Teach file, you may use EGOed lite to view it. The writer's guide is a Teach file that explains what you need to know to write for *GS+* Magazine—you may use EGOed lite or the Teach application to view it. *GS+*

How to Get System 6.0.1

Everyone should have a copy of System 6.0.1. Fortunately, we have a license to distribute it to our magazine-and-disk subscribers as a part of their subscription. Unfortunately, we can't afford to mail all five of the disks that System 6.0.1 takes up to every magazine-and-disk subscriber. However, we still want to make it easy for you to get System 6.0.1. So, if you are a subscriber to *GS+* Magazine with the companion *GS+* Disk (sorry, but we can *not* distribute System 6.0.1 to our magazine-only subscribers), send us the following items and we will send you System 6.0.1:

1) Five (5) *blank and formatted*, 3.5-inch diskettes to our P. O. Box address (which is shown on the back of your magazine). We are asking for "blank and formatted" disks because formatting takes time that we don't have, and it's a great way to tell if a disk is good before you send it to us. *If you send us a bad disk, we aren't going to replace it.*

2) A *self-addressed* return disk mailer with enough postage on it to mail the

five disks back to you. (Foreign subscribers without access to United States postage may include International Postal Coupons instead. See your local post office to obtain these.) *If you don't provide a postage-paid, self-addressed return mailer, your disks will be considered "gifts" and will be used for backups.*

3) That's all. Don't send any money. We don't want any money for this.

How Else Can You Get It?

If you are a magazine-only subscriber, here are some other ways to get System 6.0.1.

Your Apple dealer. Bug them until they get it in for you. The retail price is \$39, but that includes manuals. The part number is #A0077LL/A. For the name of your local Apple dealer, call (800) 538-9696.

Your user group. Take your own disks and they should only charge you a small copying fee. Some user groups may have it already copied for you and available for

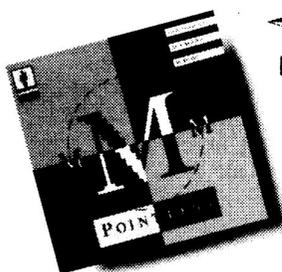
a nominal charge. (Note that some user groups make these services available only to their members. Of course, you do plan on joining, don't you?) If you need to know where your local user group is, call the Apple User Group Connection at (800) 538-9696 extension 500.

Resource Central. You won't have to bug them, they have it in stock, and in no less than two different "flavors." For just the disks (item number DA-006), the price is \$24. For the complete end-user package, including manuals, the price is \$39 (item number DA-0013). Take your pick, and then give Resource Central a call at (913) 469-6502.

And, of course, if you have a modem, you can download it from your favorite online service. The total download time is about 5 hours. *GS+*

The Best of the West

Since 1990, WestCode Software has created the best software you can get for your Apple II computer. And, now for a limited time, we're offering you the best prices ever on our highly-acclaimed products.



◀ Fantastic looking text on your printer.

Pointless 2.0 All your documents will look better than ever before, without unsightly hard-to-read jagged text. Pointless brings TrueType font technology to your Apple IIGS, providing impressive laser-sharp text on your ImageWriter, StyleWriter, DeskJet, LaserJet, and other printers. Plus, text on the screen will appear smooth at even the largest sizes.

For the Apple IIGS, System 5 or later. Reg. \$69.95

Sale \$39.95

**The quick brown fox
The quick brown fox**

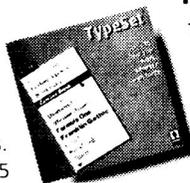
ImageWriter II printing before and after Pointless



TypeWest An outstanding collection of 40 TrueType fonts for use with Pointless. These professionally-designed typefaces are perfect for home, school, or work. Don't waste your time and money on lesser quality fonts from other sources.

For the Apple IIGS with Pointless. Reg. \$49.95

Sale \$29.95



TypeSet The perfect companion for Pointless that helps you manage your TrueType fonts. It provides a What-You-See-Is-What-You-Get Font menu that displays your fonts in their own typefaces, font sets to work with custom groups of fonts, and extensive font reports.

For the Apple IIGS with Pointless. Reg. \$49.95

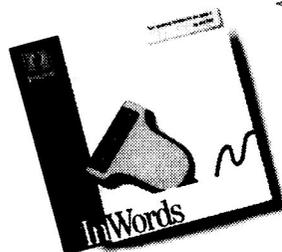
Sale \$29.95

Stretch your disk space. ➤

HardPressed 1.0 If you're always searching for more disk space, then you need HardPressed — the best disk compression software for your Apple IIGS. It virtually doubles the space on any type of disk. Your 40 MB hard drive will be able to store 80 MB, or a 2 MB RAM disk can store 4 MB! HardPressed works behind the scenes, transparently compressing your files as you save them. When you open a compressed file or launch a compressed application, HardPressed automatically expands it for you.

For the Apple IIGS, System 6 or later. Reg. \$69.95. AutoArk trade-up \$24.95

Sale \$39.95



◀ It's like typing 500 words per minute.

InWords 1.1 Optical Character Recognition (OCR) software that let's you use a Quickie or LightningScan hand-held scanner to scan pages from books, magazines, letters, etc. into text that you can load and edit in your favorite applications: AppleWorks, PublishIt, BeagleWrite, GraphicWriter, HyperStudio, and others. NEW version 1.1 adds improved recognition, speed, line handling and more! Why type? Get InWords instead!

For the Apple IIe or Apple IIGS. Reg. \$129.00

Sale \$49.95

Call 1-800-448-4250



Order direct from WestCode Software and save! 30-day money-back satisfaction guarantee.

WestCode Software, Inc., 15050 Avenue of Science, Suite 112, San Diego, CA 92128 • (619) 487-9200 • Fax (619) 487-9255

Limited time offer, prices subject to change. All brand and product names are trademarks of their respective companies, and you know who they are. © 1993 WestCode Software, Inc.

Reviews

DiscQuest

By Jawaid Bazyar

Retail pricing:

Software only - \$99.95 Includes the Family Doctor CD-ROM.

Bundle #1 - \$595 This price includes the DiscQuest software, a RamFAST SCSI card, a single-speed CD-ROM drive, speakers, and four CD-ROM titles.

Bundle #2 - \$895 This price includes the DiscQuest software, a RamFAST SCSI card, a double-speed CD-ROM drive, speakers, and four CD-ROM titles.

Educational discounts are available on all of the above. Contact Sequential Systems for details.

Typical mail-order pricing:

Software only - \$79

Bundle #1 - \$569

RamFAST ROM updates to version 3.01E:

For Rev C RamFAST - \$15

For Rev D RamFAST - \$9.50

Not copy protected

Requires 2MB RAM, one 3.5-inch drive, SCSI interface, CD-ROM drive, System 6 or later. More memory and a hard disk are *strongly* recommended. Headphones and/or powered speakers are also recommended for listening to audio from CD-ROMs. RamFAST users must have ROM version 3.01E.

Sequential Systems
1200 Diamond Circle
Lafayette, CO 80026
(303) 666-4549
(800) 759-4549

Reviewed by Steven W. Disbrow

Over the last few issues, I've commented on all of the phone calls that we get here at *GS+* Magazine about various topics. For the past few years, a favorite subject of callers has been, "Why can't I use Mac and PC CD-ROMs on my IIGS?" The answer, of course, is that the data on these CD-ROMs is stored in formats that the IIGS can not easily read. The solution, as I explained to dozens of callers, would be for someone to write a program that could read those file formats and convert them into information that your IIGS could make sense of. Well, guess what boys and girls! Someone has done just that!

DiscQuest is the first IIGS application intended specifically to read the data stored on certain types of CD-ROMs intended for the Macintosh and IBM PC. To use it, you simply put the CD-ROM in your CD-ROM drive, and run the *DiscQuest* software. You can then browse through the contents of the CD-ROM using a browse window that looks sort of like a Finder directory window that is displaying files in a list view (see first screen shot).

Since the *DiscQuest* interface is vaguely similar to the Finder's, it's very easy to learn to use: To open a folder on a disk, you simply double-click on it. To view a text file on a disk, you simply double-click on it. To see a picture or listen to a sound on the disk (you guessed it) you simply double-click on it. The only confusing thing that you might find is that sounds that you can play are labeled with a "P" icon, which is exactly the same

Currently Supported DiscQuest Titles

(All prices shown are for direct orders from Sequential Systems)

The Classic Collection (4 CD's): \$79.95
Shakespeare
Sherlock Holmes on disc!
Multimedia Audubon's Birds
Multimedia Audubon's Animals
American Family Physician: \$49.95
1991 Medical Yearbooks: \$49.95
The Family Doctor: \$59.95
Great Literature: \$39.95
Parenting — Prenatal to Preschool: \$49.95
Monarch Notes® on CD-ROM: \$49.95
Total Baseball: \$49.95
US History on CD-ROM: \$99.95
History of the World on CD-ROM: \$69.95
Countries of the World on CD-ROM: \$239.95
Multimedia World Fact Book: \$59.95
Darwin Multimedia: \$79.95

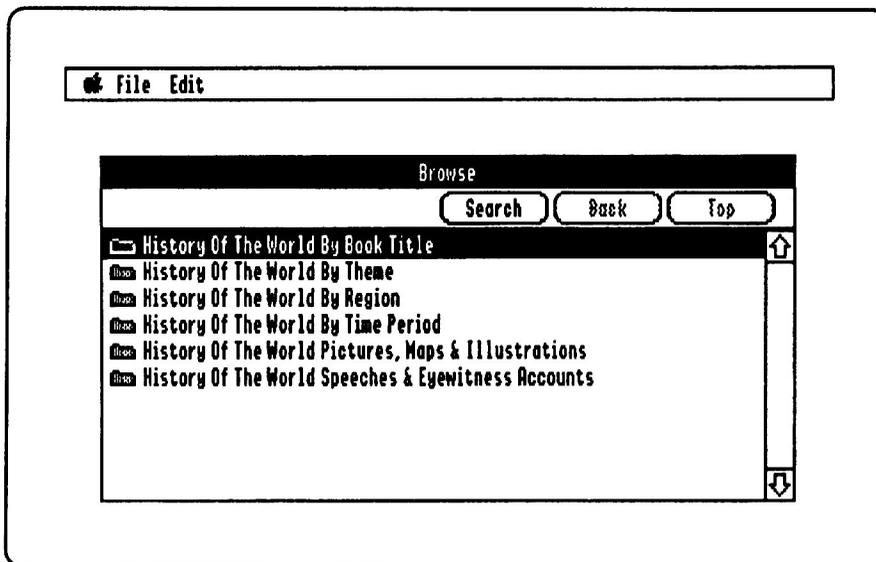
way the pictures you can view are labeled. (The *DiscQuest* documentation says that sounds should be marked with a loudspeaker icon, but for some reason, they aren't. Text files are labeled with a "T" icon. Folders, of course, are represented by the same folder icon that you see in the Finder. See the second screen shot for an example.)

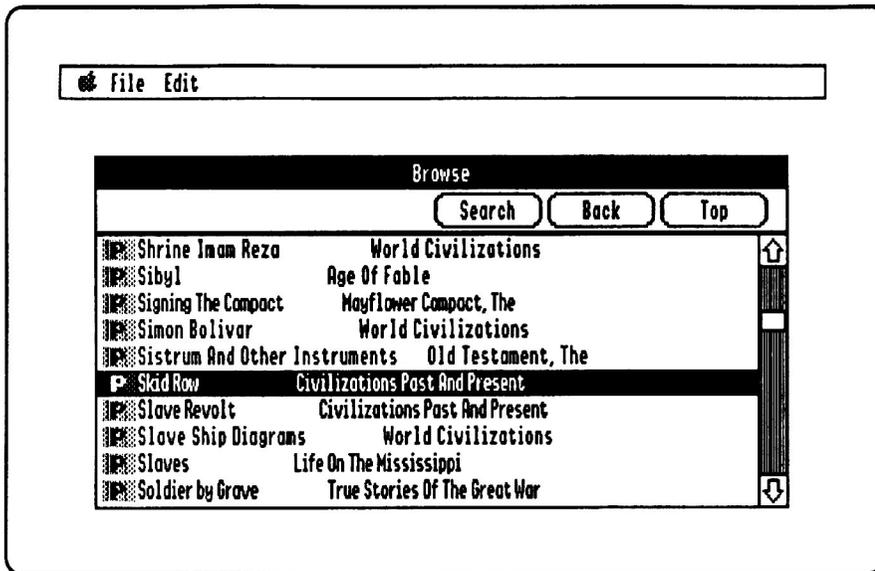
Since CD-ROMs are famous for holding *tons* of information, you might worry that it could take a long time to browse through a disk to find everything you were interested in. So, if you want, you can use the Search facility built into *DiscQuest* to quickly search the contents of a disk by four different indexes: Words, Subjects, Titles or Authors. (See third screen shot for an example.) After *DiscQuest* finds all of the items matching your search criteria, it places them in a single list. You can then simply double-click on an item in the list to display it or listen to it.

Basically, that's all there is to *DiscQuest*. It works, and it works very well. Of course, there are some details that you should know about

Text Display

The best thing about *DiscQuest* is that it actually works. It takes the text, pictures





and sounds on a CD-ROM and presents them in a way that is easy to use. When you read a text file, it is displayed in a nice monospaced font (very similar to the system font, Shaston) that is easy on the eyes and scrolls quickly. And when you have an article on the screen, DiscQuest also creates a menu containing any picture or sound items related to the text you are reading. To access these related items, you simply pick them from the menu and DiscQuest presents them to you. (Note that you can also print out any text file or picture that you might want to. This is a very nice touch.)

A Picture Is Worth . . .

Speaking of pictures, DiscQuest does an excellent job of converting the pictures on these CD-ROMs into a form that the IIGS can display. However, since the pictures were intended for systems with higher resolution monitors and/or more available colors, some of the results can be a bit disappointing. (Maps and other pictures with fine details seem to suffer the most.) Picture displays also tend to be a bit slow, even on an accelerated IIGS. To help work around these problems, DiscQuest has a Preferences window (see fourth screen shot) that allows you to change several options for picture conversion so that you can get the best, or fastest, results possible. (One nit to pick here: The Preferences window has a close box, but clicking it doesn't do anything. To get rid of the preferences window, you have to click on the Done button.)

The Sound of . . .

Another thing that DiscQuest does well is play sounds. The main reason I was so pleased with this part of DiscQuest is that the creator of the RamFAST SCSI card once told me, using very small words so that I would understand the finality of the statement, that there was no way in heck

that the RamFAST card would ever support any kind of audio on a CD-ROM disc! Fortunately, when Sequential Systems took over the RamFAST, they decided to do something about that problem, and DiscQuest is the first example of that commitment. However, to use this feature, you *must* have the latest ROM v3.01E in your RamFAST card. (If you don't have the correct ROM version, your system will probably crash whenever you try to access a new CD-ROM with DiscQuest—mine did.)

Best of all, listening to sounds with DiscQuest is just as easy as looking at a text file. You simply double-click on the sound, and DiscQuest plays it through the headphone jack of your CD-ROM drive. That's an important point: The sound does *not* come out of the IIGS speaker! You must have a set of headphones or some powered speakers that you can plug into your CD-ROM drive to hear the sounds that DiscQuest plays. Another neat thing about DiscQuest playing

sounds is that it plays them in the background. For instance, I'm typing this in EGOed, while George Kennedy (from the *Naked Gun* movies) is reading the Emancipation Proclamation in the background.

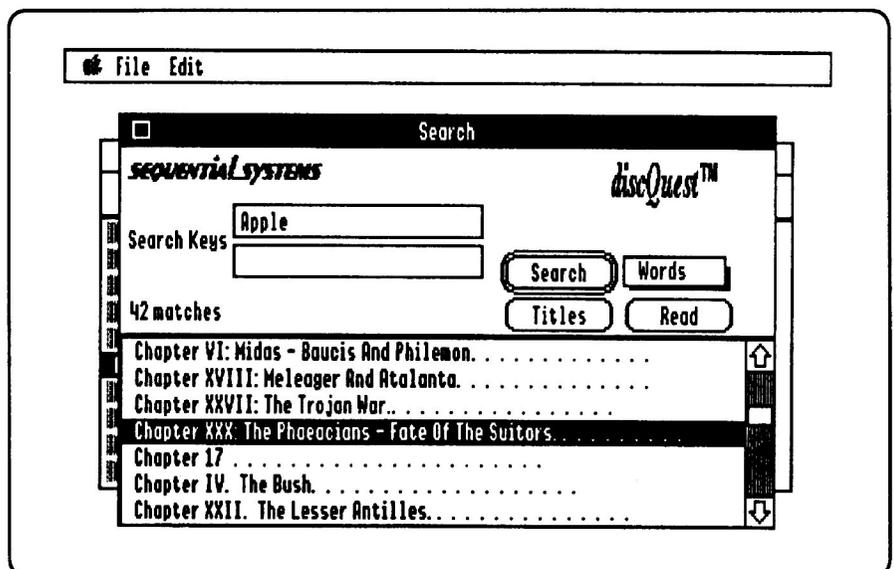
Hardware Considerations

Up until now, I've referred to the use of DiscQuest with the RamFAST SCSI card. The main reason for that is that's the card I have in my system. However, you can also use DiscQuest on a system equipped with an Apple High-Speed SCSI card. With this setup though, your choice of supported CD-ROM drives goes down drastically, and some of those drives that are supported will not allow you to play any audio that is on your disks. If you use a RamFAST card however, you can use just about any CD-ROM drive ever created, with full support for all of DiscQuests features. (If you want to be sure that your setup will work, you should call Sequential Systems for the latest compatibility information.) Of course, if you don't have a SCSI card or CD-ROM drive yet, you could just buy the DiscQuest hardware and software bundle and you would be sure to get 100% compatible equipment.

The Only *Real* Problem

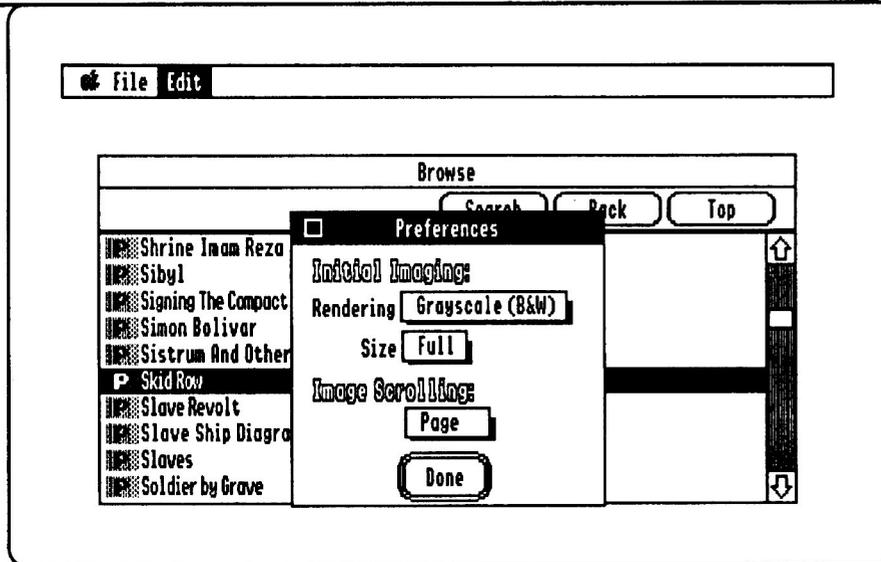
At the start of this review I said that DiscQuest is intended to "read the data stored on *certain types* of CD-ROMs" and you may have wondered what I meant by that. Well, I'll tell you.

DiscQuest is a program that allows you to read CD-ROMs that are intended for use with a program called "DiscPassage" on the Macintosh and PC. DiscPassage is a "front end" program that allows Mac and PC owners to browse the contents of CD-ROMs in the same way that DiscQuest now allows IIGS owners to browse them.



This basically means that DiscQuest is a "clone" of the DiscPassage software, and that it can read and display only the same types of data files that the DiscPassage software can. On the positive side, this means that theoretically, DiscQuest will allow your IIGS to use *any* Mac or PC CD-ROM (past, present or future) that uses the DiscPassage software as its user interface! On the negative side however, this means that DiscQuest will *only* work with DiscPassage disks! If you use it to try and browse another type of CD-ROM, you will get the message "Couldn't find a DiscPassage CD." (Another nit to pick: Sometimes when you try to search a CD-ROM you will get a "Big Dog Error". Funny, but not very informative.)

Of course, there are hundreds, if not thousands, of Macintosh and PC CD-ROMs out there, so at least a hundred of them must be DiscPassage CDs, right? Wrong. The list of compatible titles that Sequential Systems publishes (see sidebar) has only 16 titles on it. Even worse, based on my experiences in trying to find additional DiscPassage CDs for use with DiscQuest, I am afraid that there might never be more than that. Apparently, DiscPassage was an attempt at a standard CD-ROM interface that never quite caught on. Most of the latest CD-ROMs that are made for the Mac and PC have their own unique interfaces that have nothing to do with DiscPassage. This means that you simply can't use these



CD-ROMs with DiscQuest. So, unless DiscPassage catches on in the Mac and PC markets, or DiscQuest expands the types of CD-ROMs that it can read, you may be able to forever count the number of DiscQuest compatible titles on your hands and feet.

Conclusion

Even with the lack of titles, DiscPassage is a very useful and important piece of IIGS software. In fact, with the release of DiscQuest, Sequential Systems has single-handedly almost tripled the number of CD-ROMs that the IIGS can use! And even though there are only 16 compatible

titles, those 16 titles translate into roughly 9 gigabytes of information that is now available for you to quickly and easily peruse on your IIGS.

So if you are one of those people that called me up in the last few years wanting to know if you should buy a CD-ROM drive for your IIGS. My answer is now an enthusiastic "yes!" DiscQuest is exactly what you were waiting for! And, if Sequential Systems supports it the way they have supported their other products, it will only get better! GS+

The World Just Got Larger.



Featuring:

Compton's Interactive Encyclopedia™

This June, EXPAND Your World of Information with Compton's Interactive Encyclopedia™ and discQUEST Encyclopedia™ software for your Apple IIGS.

Now, for a limited time, buy the original discQUEST™ software and a copy of Compton's Encyclopedia™ on CD from Sequential Systems or your favorite dealer, and you'll receive the discQUEST encyclopedia™ software FREE!

Want to know more? Call Sequential Systems or your Sequential Systems reseller for details.

1-800-759-4549
SEQUENTIAL

Having Problems?

If you are having a problem with your **GS+** Magazine subscription, or one of the programs on the **GS+** Disk, we want to help! But we can't help if we don't know about it! For subscription problems, or if your **GS+** Disk is *defective*, you can call us at (615) 843-3988, Monday through Friday between 9 a.m. and 6 p.m. Eastern Time. For problems with one of our programs, *please* fill out the problem form that is on your **GS+** Disk and send it to us.

For any of the above, our address is:

GS+ Magazine
P. O. Box 15366
Chattanooga, TN 37415-0366

files (i.e. copying a Word file to your boot disk won't change it into an AppleWorks GS file). To actually use the files you copy, you will need an appropriate application.

- These utilities can't be run from the desktop. You *must* use some sort of shell.

- The utilities will not let you make or delete directories on MS-DOS disks. However, when you pay the shareware fee, you will receive additional utilities that *will* let you make and delete directories, along with some other neat utilities.

What's Wrong With Them?

Actually, not a lot. The only operational problems I had with the MS-DOS file utilities were:

- The utilities would not recognize some of my blank, pre-formatted MS-DOS disks as MS-DOS disks. (The office

Macintosh, with Access PC installed, did recognize these disks with no problem.) However, all I had to do was reformat the disks with the utilities and they worked fine from then on.

- All MS-DOS disks must be accessed by using a GS/OS device number. While a utility is provided to determine the number of each device, it would probably be easier for everyone if you could assign and use an MS-DOS-style drive designator, such as "B:". This situation is made even more confusing by the fact that the utilities themselves often report their progress using the MS-DOS type of drive designator.

- If you are familiar with MS-DOS, these commands, and their parameters, can be confusing. However, this isn't *really* a problem, its just a matter of preference and experience. If you have *no* MS-DOS experience, you won't know the difference.

Other than these problems, the only other complaint I have is that there isn't a desktop version of these utilities. While the author has said that he would like to do a desktop application to use the utilities from, I personally would like to see a New Desk Accessory version. That would make the utilities available from all desktop programs, and make them much more convenient to use.

Are They Worth It?

If you just need to *read* data from MS-DOS disks, you can probably get by with the MS-DOS FST that comes with System 6.0.1. However, if you need to write data *back* to those MS-DOS disks, and you don't have (or don't want) a PC Transporter, the MS-DOS file utilities are the only game in town. And, with a shareware price of only \$15 (\$20 for Australians), they are a great bargain too. If you have the need to work with data from MS-DOS disks, I highly recommend these utilities. **GS+**

```
Inbuilt commands
-----
PREFIX [path] - Change (or display) the default pathname prefix for commands
CAT [path] - Catalog the contents of the current (or another) directory
TYPE path - Display the contents of a text file
CREATE path - Create a new subdirectory
DELETE path - Delete a file (alias DEL)
RENAME old new - Rename a file (enclose name in quotes if it contains spaces)
FTYPE path [type] [aux] - Display or change a file's (hex) filetype and auxtype
VER - Display this program's version number and copyright details
HELP [command] - Display help for external commands (from the 'HELP' directory)
                (Enter 'HELP MSDOS' to see a summary of the MSDOS commands)
QUIT - Return to the GS/OS program launcher (alias EXIT, BYE)

External commands and Applications (including Applesoft BASIC programs)
-----
Any command other than those listed above will be executed as a program from
the default directory (or from another disk or directory if specified)
Eg 'MDIR 3' will be executed from the default directory (see PREFIX command)
'MSDOS TOOLS MDIR 03' will be executed from the 'MSDOS TOOLS' disk
'STARTUP' will execute an Applesoft BASIC program called STARTUP
```

Salvation: Bakkup v2.0

By Christopher Warner

Version 2.0 by Steven R. McQueen

Typical mail-order price: \$69.95
(\$39.95 to upgrade from previous versions)

Requires System Software v6.0 or later, 1.5MB of RAM. At least 150K of free space is required for installation on a hard disk.

Vitesse, Inc.
P. O. Box 929
La Puente, CA 91747-0929
Orders: (800) 777-7344
Tech Support: (818) 813-1274

Reviewed by Bill Moore

Vitesse is in the habit of making software that can save the day. Appropriately enough, this series of disk utilities is called the Salvation series. If your disk is having a bad hair day, grab a copy of Deliverance. If your IIGS starts doing the cybernetic equivalent of spinning it's head around and vomiting pea soup, heal it with Salvation: Exorciser (a virus-killer). If your hard drive's feeling a little fragmented, put the pieces back together again with Renaissance. But the best medicine for many computer woes is prevention—namely to do routine backups of all (at least, all your valuable) data and programs. For this purpose, Vitesse created Salvation: Bakkup.

Bakkup has recently seen a facelift, to make life a little easier for System 6 users. The first program in the Salvation series, Bakkup has seen more changes to it's last name than Elizabeth Taylor. (But probably not as many facelifts.) Originally called just "Salvation" (and reviewed way back in *GS+* V1.N4), it later became known as "Salvation: Guardian," then recently it became "Salvation: Bakkup." For this review though, I'll just call it "Bakkup."

Bakkup's new and improved manual is up to Vitesse's usual standard of excellence. Thirty-five pages long, it covers the abilities of the program pretty exhaustively, and gives you much more than "the basics." For first-time users of backup software, Vitesse gives you easy, step-by-step instructions to follow in making backups of your hard drive. This, plus the program's simple, intuitive style, means even MENSA rejects can get their valuable programs and data safely backed up. Bakkup provides support for backup to floppies (low or, with the appropriate hardware, high density), SCSI tape mechanisms, and backup to a file. This

last option allows you to backup to another hard drive partition or other SCSI device, like a floptical or SyQuest drive.

Back when the first incarnation of Bakkup burst upon the scene, it was hailed as a visionary program, mainly because it's only competition was ProSel-16 (reviewed in *GS+* V4.N3). In the time since, however, the market for backup programs has gotten a bit crowded. In addition to ProSel, programs like Universe Master (reviewed in *GS+* V4.N2), Archiver, UtilityWorks GS (reviewed in *GS+* V3.N6), and the RamFAST.System program (which is built into the RamFAST SCSI card), to name just four, have appeared and will allow you some type of backup option.

The program's look and feel is basically the same as in previous versions (see screen shot). While not as straightforward as Apple's Archiver program, Bakkup gives you a little more flexibility in backing up your drives. A simple menu bar gives you access to most of Bakkup's power thru the file menu. From here, you can choose the main options you'll use Bakkup for—either to make a new backup, continue a backup, or restore one. One *big* difference between Bakkup and Archiver is the "Backup Macros" menu in Bakkup—which I'll discuss shortly.

A nice touch is the reminder window that appears when you start the program. It tells you when the last backup was done with Bakkup, along with what got backed up. This can serve as a handy reminder that it's "time to do the backups."

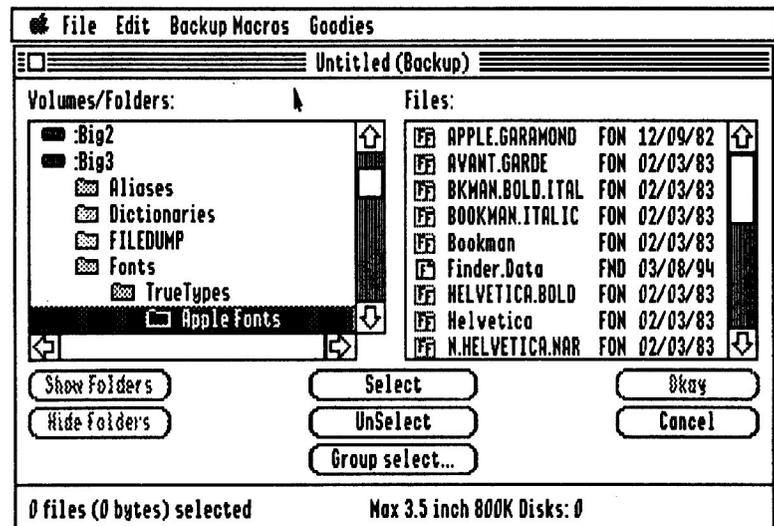
I've used Bakkup along with Apple's Archiver to try backing up several

volumes, including ones on my machine at the office and my IIGS at home. This was done using the Volume backup option in both programs, with compression of the archived data. When I used each program to back up a partition to floppies, Bakkup used one more floppy than Archiver did. This was interesting to me, because when I used both Bakkup and Archiver to back up my boot partition at home (using the "back up to a file" option with compression), the Bakkup file(s) were actually two and a half meg *smaller* than the counterpart Archiver file.

While at home, I used Bakkup to back up the contents of my boot partition to a file on another partition. After a half-hour of work, the program stopped accessing my drive. After scratching my head in puzzlement, I canceled the operation and took a look at the destination folder. The target disk had filled up, and Bakkup didn't tell me a thing—it just quit, leaving a bunch of temporary files scattered through the destination folder. This is a major annoyance that needs to be looked at by Vitesse.

Another thing that is a minor cosmetic annoyance to me is the fact that backing up to a file actually puts *two* files in the target directory. One is type Unknown, and the other is an EDASM 816 relocatable file. It would've taken almost no effort to get Bakkup specific filetype/auxtype assignments from Apple, and would help in keeping things more organized.

Bakkup 2.0 does have some nifty things going for it, however. One thing that Bakkup can do that is a real timesaver is that it allows you to select more than one



volume for a backup process (unlike Archiver, which only lets you work with one volume/partition at a time.) This, along with the ability it has to define macros (see below), can make doing backups of even large drives (like my 240MB NCS Pro) less tedious and much, much easier.

As I mentioned above, you can define up to ten backup macros to ease the process of doing regular backups. Notice the screen shot in Figure 1. I have macros in my copy of Bakkup set to do a complete volume backup for several volumes (which I try to do once a month), plus incremental backups of changed files (which should really be done weekly).

Speaking of incremental backups, Bakkup allows this to be done but, in my humble opinion, it's not as easy as it should be. In order to do a backup this way (or to define a macro to do it), you have to start a New Backup (Command-B from the File menu), then select the Volume(s) you're backing up, then click on the "Group Select" button, which brings up a dialog box allowing you to set these options. This box gives you a lot of control over what files you want to select (see Figure 2). For example, when I defined my "Picard backup Modified Files" macro for weekly backups (see Figure 1; it's Command-1 under "Backup Macros"), I told Bakkup to do all files modified since the last backup (top radio button on left side in Figure 2) and checked the "Reset 'Backup Needed' bit" box. This way, only changed files get backed up and are marked as no longer needing backup. Being able to define a macro makes this process a little less painful since you should only have to do it once, but you have to know what to do in order to make the macro to begin with.

Figure 2 - Group Selection Options

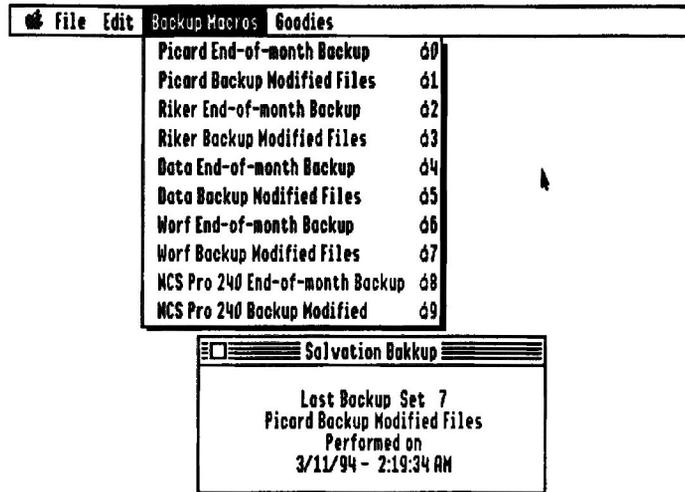
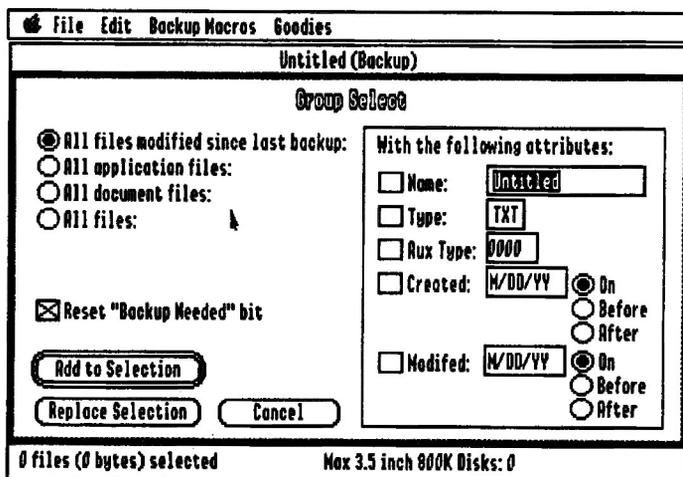


Figure 1 - The Backup Macros Menu

Bakkup can handle another tedious chore of the process of backing up data for you by printing out labels for your backup disks. This part of the program, however, seems to have been written with only ImageWriter II's and single, continuous form 3.5-inch disk labels in mind. Although that will admittedly cover the majority of IIGS owners (including me), the lucky folks that actually have laser printers are left to their own devices to print out disk labels.

Bakkup will also allow you to print out a report showing exactly what's been backed up. It prints out while the backup is happening, so expect a wait, and a long report, if you're backing up a big volume.

According to Vitesse, they're not thru with this program. In the middle of writing this review, I got a new version (2.0.3) from Vitesse which, according to

them, kills some bugs that made life miserable for SyQuest owners, among other things. Future improvements call for improved floptical support (where as I understand it, Bakkup will treat a floptical disk as a huge 3.5-inch drive, like it should). I wish Vitesse would implement my other suggestions, along with the ability to highlight more than one volume in a backup selection window. That would make doing a multiple volume backup less tedious. With IIGS owners getting bigger and bigger hard drives, steps need to be taken to make backups of these large drives easier, and allow disk backups to larger devices like SyQuests and flopticals—unless you'd rather feed your 3.5-inch drive 250 disks!

Conclusion

If you already have a copy of Guardian (or Salvation, or Bakkup, depending on what day of the week your copy shipped), this would be an excellent upgrade. If you're looking at Bakkup as a first-time buyer, I'd take a long hard look at it and see what it's got that you've simply got to have (and what Archiver, which is free since it comes with System 6, doesn't have compared to Bakkup). When Diz handed me a copy of Bakkup 2.0 to review, I had essentially the same gripe with Bakkup that he had with Wings as compared to the Finder. (See Wings review in GS+ V2.N1.) Yes, it's nice, and does some nifty stuff, but what will it do that my free program won't that justifies the cost? The answer is that it does several extra cool things like macros, multiple volume selection, and a reminder window telling me how long its been since my last backup! GS+

Spectrum

By Ewen Wannop

Retail price: \$129.95

Typical mail-order price: \$84.95

Not copy protected

Requires 1MB RAM and one 3.5-inch drive (more memory and a hard disk are strongly recommended), System 6 or later, and a modem. Installation on a hard disk requires approximately 600K of free space.

Seven Hills Software, Inc.
2310 Oxford Rd.
Tallahassee, FL 32304-3930
(904) 575-0566

Reviewed by Steven W. Disbrow

Spectrum is the long-awaited IIGS-specific telecommunications program from Seven Hills Software. It allows you to connect to electronic information services using your IIGS, *without* having to give up access to your New Desk Accessories, control panels, or any of the other elements of the IIGS desktop. This makes Spectrum the *only* full-featured, desktop-based telecommunications package available for the IIGS. But, by now, most every IIGS owner out there with a modem already has a telecommunications program (probably either ProTerm or ANSITerm [reviewed in

GS+ V3.N4]) that they are fiercely loyal to. So, with Spectrum's late entry into the game, the question is, "Was Spectrum worth the wait?" Well, before we can answer that question, we need to go over what you get with Spectrum and what it can do for you.

What You Get

The Spectrum package includes two manuals, a reference card and two disks.

With the Spectrum documentation, Seven Hills has once again set the standard for quality documentation. The first manual, "Getting Started and Reference," is exactly what the name implies—a tutorial and reference for using Spectrum. But, it's a very complete reference that goes beyond the "this menu item does this" standard that most documentation follow. In addition to very detailed instructions on how to install Spectrum on both floppy and hard drive based systems, the manual also contains these other points of interest:

- A very thorough description of each and every item in the Spectrum menu bar. This includes detailed descriptions of each and every item in all of the dialog boxes that you will run into as well.

- A discussion of potential conflicts with system extensions (control panels, NDAs, etc.) and how to resolve them.

- A look at the process of transferring files. This discussion includes complete information on all of the file transfer protocols that Spectrum supports (CIS B+, XModem, YModem, ZModem and plain text), and which options in Spectrum affect these protocols.

- Information on how to use Spectrum successfully with The Manager, Twilight II, and a high-speed modem.

- A question and answer section that goes over some of the troubles you are most likely to have.

- A wonderful wiring diagram that shows exactly how to make a hardware handshaking cable for use with high-speed modems. (Just in case you aren't handy with a soldering iron, contact information is also included for a company that can sell you a pre-made cable.)

- A table of the basic commands that are used with Hayes-compatible modems. (This will be invaluable for you if, like myself, you have absolutely no idea where your original modem manual could actually be.)

- A complete index!

The only things that I found missing from this manual were discussions of all of the terminal emulations that Spectrum does

WANTED APPLE II SPECIFIC CD

So, you finally bought a CD-ROM drive and are happily using discQuest to open a world of information for your Apple IIGS...great! But you've probably noticed that one thing is missing: an up-to-date Apple II specific compilation CD—but not for much longer!

Coming this summer: the answer to your desires. DigiSoft's groundbreaking Apple II specific compilation CD (currently unnamed) will be here, chock full of *over 600 megabytes* of data and software for your Apple II, at around \$55. Call or write for more information!

DigiSoft
Innovations

P.O. Box 380
Trumbull, CT 06611
Phone 203.375.0837
email: digisoft@aol.com

Infinitum!

© 1994 DigiSoft Innovations. All trademarks belong to their respective holders.

(however, these are discussed in documentation files on the disk), and a discussion of all of the commands available in the Spectrum scripting language. Which brings us to the second manual that comes with Spectrum, "Scripting."

The Scripting manual is a complete (and I do mean *complete*) reference to the Spectrum scripting language. Just by flipping through it, you will quickly come to realize that Spectrum has a *very* powerful scripting language! As you might expect, there are lots of commands that you will find in other programs (commands to send a break, hang up the modem, etc.), but there are also commands that are specific to the IIGS. For example, you can write a script that draws an icon, picture or window on the screen, plays a sound or brings up a Standard File dialog that will let you specify a file to transfer. This manual tells you everything you need to know to do all these things, and more.

As if that weren't enough documentation, Spectrum also includes a reference card that can quickly point you to the manual page that describes any part of the program you might be interested in. On the front of the card are pictures of almost all of the Spectrum menus (three hierarchic sub-menus are omitted), and room for lots of notes. Listed next to each individual menu item is the page number in the reference manual that discusses the item. On the back of the card is a list of all the Spectrum scripting commands along with the page numbers they appear on in the scripting manual.

On the two disks that you get, you will find the Spectrum application (of course), lots of extra documentation (for the various terminal emulations), a *ton* of

custom sounds, and a copy of GS-ShrinkIt, complete with its documentation. This last item is a very pleasant surprise, because I know from personal experience that there are a *lot* of IIGS owners out there that don't have GS-ShrinkIt, and GS-ShrinkIt is indispensable if you want to be able to download and use software for your IIGS.

So, now that we know what comes with Spectrum, let's take a look at what you can do with it all.

What It Does

First and foremost, Spectrum is a program that allows you to use your IIGS to access electronic information services through a modem connected to your computer. Depending on the service, this will allow you to send electronic mail, attend real-time conferences and upload or download files. (If you don't know what all of this telecommunications terminology means, be sure to read "The World at Your Fingertips" in GS+ V4.N3 for a complete introduction to telecommunications.) Without ever writing a single line of script code you can use Spectrum to do all of these things and more. In fact, there are so many things Spectrum can do, I haven't been able to come up with a really creative way to list all of them other than to simply list them in no particular order. (Not creative, but it gets the job done.) So, here goes:

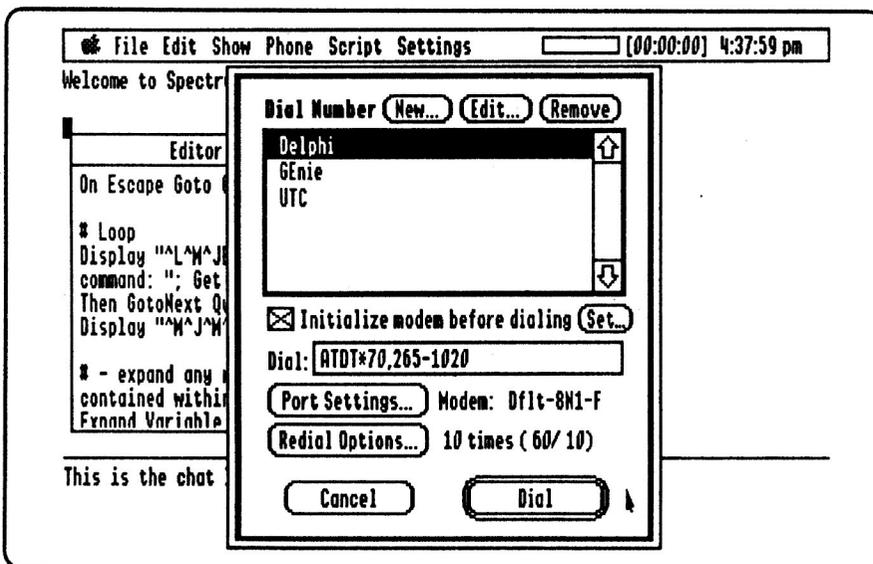
- Spectrum allows you to connect to services at speeds of up to 57,600 baud. (Of course, the connect speed depends on the maximum speed of your modem and the maximum speed of the service you are connecting to.) It also gives you complete control over the port your modem is connected to. (Spectrum also works with most Apple II internal modems.)

- Spectrum has extensive online help, which covers most of the same ground as the discussion of menus in the reference manual. It also includes information on last minute changes to the program that didn't make it into the manual.

- Spectrum has separate scrollbar and capture buffers, both of which can be independently configured to hold up to 3MB of incoming data (if you have enough memory that is). Spectrum can also capture data directly to a new file, or it can append incoming data to an existing file. (Spectrum even lets you set up a default capture file on any volume you have online. This is very handy if your hard disk is about full, or you want to always capture to your RAM disk.) Of course, you can copy information out of either buffer simply by using standard editing commands.

- Spectrum is very fast for a desktop communications program. Speed (actually, the lack of it) has always been one of the reasons that IIGS developers have cited as to why there haven't been any desktop communications programs. Of course, Spectrum can't completely solve the IIGS speed problem, but it does offer several ways around it. If you are a desktop purist, you can run Spectrum completely in desktop mode and have constant, uninterrupted access to the IIGS desktop. However, if you don't mind a bit of switching back and forth, and you need to telecommunicate at the fastest speeds possible, you can use one of Spectrum's text-based terminal emulations to get the maximum amount of speed possible. From these text-based screens, you can easily switch back to the desktop screen simply by pressing Command-W. While this might sound like a bit of a pain, and I personally would prefer it if Spectrum presented all of its displays in actual *windows*, it is a very good compromise and you get used to it after working with the program a short while.

- Speaking of terminal emulations, Spectrum has several of them. (Spectrum calls these emulations "Online Displays.") Included with Spectrum are the following emulations: ANSI, ProTerm Special, ViewData, and VT100. It is important to note that these are *add on* emulations that are included in addition to the three online displays that are built into Spectrum. This means that, if there is a new terminal emulator developed for Spectrum, all you have to do is copy it into the Spectrum Add.Ons folder and it will be available immediately! This also means that if there is ever a problem with one of the emulators (as there was with the first version of the Spectrum's VT100



emulator) all you have to do is download an update (when available) from one of the online services and use it to replace the old version!

- As I mentioned earlier, Spectrum supports just about every file transfer protocol that there is. I've personally used it to perform every type available except for a CIS B+ transfer, and that's simply because I don't have access to any systems that use that protocol. From my experience, Spectrum does an excellent job of transferring files. However, on a couple of occasions I did have trouble with ZModem transfers, but, it turned out to be a conflict with our very own Cool Cursor control panel. But, all I had to do was turn on Cool Cursor's "Animate Only When Safe" option and the problem disappeared. (I should note that it was actually the Spectrum documentation that tipped me off to this potential problem.) And, while we are on the subject of downloads, this is a good place to note that Spectrum also allows you to specify the default folder for all of your downloads to go into.

- Spectrum has several text formatting options that you can use to change incoming or outgoing text. These filters allow you to do such things as strip Line

Feeds (used by most online services, but worthless in the Apple II world), break paragraphs into individual lines (great for sending messages to services that require a carriage return at the end of each line), and eleven other formatting operations that can make communicating with the IBM/Mac-dominated online universe easier. Along the same lines, Spectrum includes a very cool feature, Paste as Reply, which allows you to automatically paste text from the clipboard into a message complete with "message quoting" added. So, for example, if you wanted to reply to a message that reads, "How are you today?", you would copy that text out of the scrollback buffer, and then use Paste as Reply to insert it into your reply like this, ">> How are you today?"

- Spectrum also includes a cool character filter that you can use to *automatically* weed out or change certain characters from your incoming messages. For example if you wish, you could set up a filter that would automatically change all incoming line feeds into "\$". Spectrum allows you to have a separate filter for incoming and outgoing characters. Best of all, you can make as many of these filters as you wish (perhaps one for each service you call), and switch between them using a pop-up menu.

- In addition to all of the above, Spectrum includes several other features that you would expect from a good telecommunications program: a session timer (complete with cost computation, if you are willing to write a script), a chat line (similar to the "edit line" in the AppleWorks GS telecommunications module), and a dial directory that can hold up to 32 entries.

- Spectrum can also launch other IIGS applications, with the option to *not* hang up the phone, and return you to Spectrum after you are done with the other application. This can be very handy if you realize that you need to use another application (GS-ShrinkIt perhaps) while you are online, but you don't want to log off and log back on. (It can also get expensive if you *forget* that you are still online!)

- You may have noticed earlier that I mentioned that Spectrum has three hierarchic sub-menus. (These are used to specify file transfer protocols and to select a capture file.) This means that, like a lot of Seven Hills programs, Spectrum has to use the Hierarchic init. You might think that this means that if you shift-boot so that no DAs or inits are installed, Spectrum won't function. However,

Twilight IITM

Over 40 effects now,
with more coming soon!

Requirements: Apple IIGS with two 3.5" drives or one and a hard drive with System 6. 100k free RAM min. is required. (1.5mb total suggested, but 1.125mb will work on a stripped down system.)

Twilight II is a *complete* rewrite of Twilight I! List price: \$39.95. Also available from Big Red Computer Club & Resource Central. School purchase orders are accepted, low cost site licenses also available. Dealer and user group inquiries are welcome. Overseas orders add \$5 for air mail shipping.

Competitive Upgrades: (only available direct)

- From Twilight I, with copy of Twilight I shareware payment check: \$22 (\$20 + \$2 S&H).
- From Q Labs' Signature GS or Simplexity's Desktop Enhancer, with original manual cover: \$26 (\$24 + \$2 S&H).

DigiSoft Innovations

P.O. Box 380
Trumbull, CT 06611
Phone 203.375.0837
email: digisoft@aol.com

We're a small group of die hard Apple II enthusiasts who love expanding the "limits" of the GS. We are committed to create and update high quality products at modest prices. Twilight II v1.1 is our latest example of this, and it certainly won't be our last! Many more treats are in store (such as even more effects!) for the next version of Twilight III! We're not content to sit back and watch our product sell, issuing only bug fix updates. We listen to user's suggestions, feedback, and requests for features, and incorporate a great deal of these into future versions. You're buying a product with a future—a future that you can have some say in.

© 1994 DigiSoft Innovations. All trademarks belong to their respective holders.

II Infinitum!

Spectrum was designed specifically to work around this problem. If Spectrum finds that the Hierarchic init has not been loaded, it presents a dialog telling you this, and gives you the option to load it and continue or quit back to the previous application. This is a *very* nice touch, and another indication of the amount of thought that has gone into this program.

Scripting

Like I said before, Spectrum has a powerful scripting language built into it. With this language, you can easily automate simple tasks that Spectrum can perform. For example, you can have Spectrum watch what you do while logging onto a system and automatically generate a script that will "play back" those actions whenever you wish.

Now if all you want to use scripts for is to log onto your favorite services, the automatic script generation features of Spectrum will probably be all you ever need to use. However, if you want to do something a bit more complex, like automatically capturing all of your e-mail, you'll need to break out the Spectrum Scripting manual and write a custom script of your own.

As I said earlier, the Spectrum scripting language is very powerful. It contains hundreds of commands that allow you to do things that are as simple as setting the baud rate as well as things that are as complex as a For-Next loop. In fact, there are so many of these commands, I believe that, given some time, someone out there is going to write a BBS script that will allow other Spectrum users to call up and navigate the BBS as if they were using a regular IIGS application! (Well, maybe not, there don't seem to be any scripting commands to read or otherwise access the mouse easily.) In the short time I've had Spectrum, I've personally been able to write scripts that log onto my GENie and Delphi accounts, capture all of the mail and messages I might be interested in and log off. This means I can start up Spectrum first thing in the morning, have it begin checking my mail, and then go do something else that needs to be done before everyone else shows up at the office. When Spectrum is finished, I can open the capture file, review its contents, and then respond to anything that needs responding to. If that sounds cool, it is. And I've just begun to scratch the surface of what Spectrum scripts can do.

After you have a few scripts written (either by Spectrum or by hand), you'll probably want to be able to run them quickly. That's why Spectrum allows you to add your most commonly used scripts to the

Scripts menu, complete with option-key equivalents! So, when I want to call Delphi, I just hit option-D and Spectrum runs my script to log onto Delphi! Another option-key combination logs me onto GENie, and yet another appends my standard signature to my outgoing mail. This is very, very nice.

Finally, if you are a Talk-Is-Cheap user, you'll be happy to know that Spectrum can run your Talk-Is-Cheap scripts with little or no modification. (Note that I don't have Talk-Is-Cheap, so I wasn't able to verify this capability. But, since Spectrum has been able to do everything else that Seven Hills has claimed, I see no reason to doubt it!)

Bad Points

Of course, even with all of these great things going for it, I was able to discover just a few *minor* things wrong with Spectrum.

- If you have an NDA open in Spectrum, and you switch to a text-based terminal emulation, the NDA is closed. As a programmer, I know *why* this has to happen, but it still made me lose a reply that I had typed into EGOed. Here again, I find myself wishing that Spectrum presented its emulations in real windows. It would be slower, but the interface would be a *lot* more consistent.

- The Spectrum editor only lets you have one document open at a given time. And while this may sound like a reasonable limitation, I often find myself needing to be able to reference several files at one time when I am trying to reply to something online. The one file limitation can also be a pain when you are trying to build a new script by cannibalizing pieces of other scripts.

- Another problem with the editor is that you can't just use the New menu item to create a new editor document. You either have to use the Open Editor Document item or the Show Editor item to bring the editor up. All the New menu item does is clear out the contents of the Editor window if it is *already* open. This behavior is not very intuitive, and different from the way every other IIGS editor works.

- As I mentioned at the beginning of this review, Spectrum comes with a lot of sounds. It uses these sounds to let you know when certain events happen. For example, Spectrum includes sounds to let you know when a file transfer succeeds or fails, when you have connected to a system successfully, and lots of other telecommunicating oriented events. It

also includes welcome and goodbye sounds that are played when you run and quit Spectrum. Personally, I don't mind the informative sounds that are played while actually *using* Spectrum, but the welcome and goodbye sounds quickly become annoying. Unfortunately, if you turn those sounds off, you turn *all* of the sounds off. It would be much better if Spectrum let you turn those sounds off separately.

- While Spectrum does include GS-ShrinkIt, it is still a separate application, and you have to quit Spectrum to run it. It would be *much* nicer if Spectrum could unpack ShrinkIt files all by itself. (In case you were wondering, Spectrum does automatically handle Binary II headers.)

- Finally, if you change the time on the IIGS internal clock while you are using Spectrum, the Spectrum clock display does not display the new time until you quit and re-run Spectrum.

So, Was it Worth the Wait?

As you can see, there's a whole heck of a lot more that's right with Spectrum than there is wrong with it. In fact, I think the only real bug that I found with it was the problem with the clock. For a program that is this complex and powerful, that is an amazing accomplishment. Seven Hills and Ewen Wannop should be extremely proud of Spectrum. It *was* worth the wait.

The bottom line is that Spectrum is a great product. If you are a IIGS owner with a modem, there are only two things that should keep you from buying Spectrum: 1) You are hopelessly devoted to your current telecommunications program. 2) You don't have, and can't afford, the hardware and software configuration to run Spectrum. If neither of those excuses apply to you, and you don't buy Spectrum, you are doing a disservice to yourself and the IIGS community as a whole. This is exactly the kind of high-powered, IIGS-specific software that IIGS owners have been waiting years for. **GS+**

Get Hyper! Again!

It's summer time again, and that means that it's time for the second annual HyperStudio Festival! If you are a user of Roger Wagner Publishing's HyperStudio for either the IIGS or the Mac, you'll want to be sure and attend this information packed conference in San Diego California. The Festivities begin on July 13th and run through the 16th. Among the scheduled events are two all day workshops on how to get more out of HyperStudio (one for the Mac version and one for the IIGS version), and a Friday night beach party. During the workshops, you'll learn how to use video digitizers, sound digitizers and other hardware additions with HyperStudio. During the beach party, you'll get to see Roger in nothing but a Speedo and a tie. (Ladies! No hyperventilating please!)

But, attendance is limited to just 300 people, so you better hurry and make your reservations now! The cost is \$150 per person for the festival and conference, and the individual advanced HyperStudio workshops are \$95 per person. Also, Roger Wagner Publishing has put together special travel and hotel packages for conference attendees. So, for complete information, contact:

Roger Wagner Publishing, Inc.
1050 Pioneer Way, Suite "P"
El Cajon, CA 92020
(800) 421-6526

HyperStudio Network Catalog

Speaking of HyperStudio, the HyperStudio Network has just come out with their latest catalog of HyperStudio products. Included in the catalog is information about the latest version of HyperStudio for the Mac, information on dozens of disks of HyperStudio stacks (in both IIGS and Mac versions), and a description of a new book, *HyperStudio Quick Course*, that is to be published this year.

If you are a HyperStudio user, you shouldn't be without this catalog. So, get yourself up and contact:

HyperStudio Network
Box 103
Blawenburg, NJ 08504
(609) 466-3196

SoftDisk G-S, New and Improved!

SoftDisk publishing has announced that, beginning with issue 54 of SoftDisk G-S, the SoftDisk G-S shell takes full

advantage of System Software 6.0, and is fully compatible with The Manager from Seven Hills Software. The new SoftDisk G-S shell was written their new Technical Editor, Greg Templeman and takes up only 31K on disk (compared to the old shell which was more than 50K).

In other words, SoftDisk G-S is better than ever, and if you don't subscribe, you should! (And no, they didn't pay me to say that.) For more information, contact:

SoftDisk G-S
606 Common St.
Shreveport, LA 71101
(800) 831-2694

Shareware Solutions Too!

Speaking of things you should subscribe to, Joe Kohn's *Shareware Solutions II* is adding a new feature, and instituting new subscription rates effective May 1st.

The new feature is a "What's New" type column written by none other than A+inCider veteran Cynthia Field! In this column, Dr. Field will summarize new products and services for all Apple II computers.

The new rates for a Shareware Solutions II subscription reflect the fact that it is an even bigger publication than Joe originally planned (20 pages an issue rather than the 12 pages per issue that were originally advertised). But, they are still reasonable, especially for the amount of information you get! Those new rates are as follows:

A one year subscription (6 issues), which starts with the current issue, is \$20 for U.S. and Canadian subscribers and \$35 for everyone else.

A two year subscription (12 issues), which will be back-dated to start with the very first issue, is only \$35 for U.S. and Canadian subscribers and \$50 for everyone else.

All of these prices include first class or air mail delivery. So, to get your subscription started, send you check or money order (in U.S. funds) to:

Joe Kohn
166 Alpine Street
San Rafael, CA 94901-1008
USA

Online Enhancements

Do you hate that rancid old America

Online (AOL) software that IIGS users have to use? Well, you aren't alone. Fortunately, a couple of intrepid IIGS America Online users have finally done something about it. The new Online Enhancer (OE) is a IIGS desktop-based application that fixes a lot of bugs and adds new functionality to that old America Online software. What sorts of stuff does OE do for AOL? Glad you asked:

- OE improves interrupt handling by the AOL software, making it compatible with The Manager.
- OE fixes many of the problems that AOL had with NDAs. In fact, you can get to your NDAs by switching to the OE shell while online.
- OE adds support for AOL sound events (downloading files, chat room sounds, etc), using the Sound control panel or raw sound files.
- OE supports extended keyboards for quick navigation around AOL.
- OE gives you access to the LaPub libraries.
- OE lets you bring up the AOL Goto and E-Mail windows at any time.

And those are just the features that I have room for! Online Enhancer does a lot more stuff, and the price is only \$30.

For more information, contact "AFA AndyW" on America Online or send your check or money order and AOL screen name to:

Andy Wells
307 Klotter Ave.
Cincinnati, OH 45219

Your Product In Lights!

Well, not in *lights*, but we will let folks know about it... *if you send us a press release!* Remember, if you don't send us a press release, we can't let folks know about your product—after all, we can't read your mind! So, if you have a new product or a service that IIGS owners would be interested in, send us a press release via e-mail (send it to "GSPlusDiz@aol.com") or mail it to us at:

GS+ What's New
P. O. Box 15366
Chattanooga, TN 37415-0366 GS+

For System 6.0.1, Apple defined nine new assignable sound types for the Sound control panel, and in recent Apple style, they, for some long-lost and forgotten reason, neglected to update the Sound control panel to include the new sound task types. I've seen small updater programs that add new predefined sound types to the Sound control panel, but I have never seen one that lets you add *any* sound type. I decided it was time for one such utility program to emerge—its name: More Sound.

The New Sound Types

Figure 1 presents the list of new "official" sound types for System 6.0.1. Most of the sounds are related to communications software. There are a few general-purpose sound types, though. It's interesting to note that *all* of the new sound types can almost certainly be directly attributed to Spectrum. And, if you purchase Spectrum, Seven Hills ships an installer program that will automatically add the new sound types to your Sound control panel. As more programs enter the 6.0.1 world, they'll take advantage of the new sound types, too, and if you don't have Spectrum to update your Sound control panel, you'll be "missing out." (For instance, our FLI Convert program uses the "Beginning long operation" sound when you start to convert an animation.)

Adding Sounds

So, let me walk you through an example of how to add sounds to your Sound control panel. First, launch More Sound. Next, choose the Open menu item and locate your Sound control panel. A window will then appear which contains a list of all the sound types that your

Figure 1 — Additional 6.0.1 System Beeps

Code	Task
\$0070	Beginning long operation
\$0F80	File transfer complete
\$0F81	Real-time message
\$1000	Connected to service
\$1001	Disconnected from service
\$1002	Entered real-time chat
\$1003	Left real-time chat
\$1010	Feature enabled
\$1011	Feature disabled

system knows about. To add a new sound, all you have to do is fill in the blanks at the bottom of the screen and click on the Add button. Let's take an in-depth look at how you actually do this. First, locate the sound type you want to add in Figure 1. Now type the sound's code in the code LineEdit control, leaving off the "\$." Next type the sound's task name in the task LineEdit control. Now click on the Add button to add the sound to the list of sounds. It's that easy.

Removal

More Sound can also remove sounds from the list. If you never use a particular sound type, you can customize your Sound control panel to only have the sounds you frequently use. To remove a sound type, select it in the list of sounds and then click on the Remove button. (Note that if you have a sound associated with the type you are removing, it's probably best if you use the Sound control panel to remove the assignment before you remove the sound type. Otherwise, you'll be left with a sound

attached to the task you removed and the Sound control panel can never remove the assignment because the type has been removed from the Sound control panel. The assignment is still valid, though, since assignments are independent of the known sounds. The only way to remove the assignment would be to add the sound task back and then remove it.)

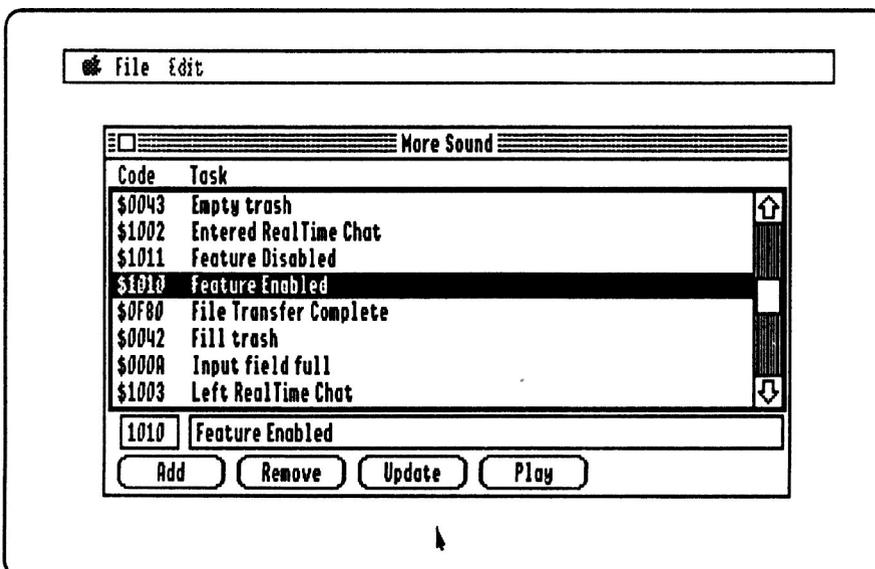
Updating

More Sound will let you update the information on existing sound types. The primary function of updating is to let you change a new sound type that you may have entered incorrectly. For example, if you typed the code for "Feature enabled" as 1001 instead of 1010, you could select the Feature enabled task, change the code to 1010, and click on the Update button to fix your initial mistake. You can also update some of the other pre-defined sound types. Most likely, you'll change the name of the task. For example, you could change the "Empty trash" sound task name to "Empty recycle bin".

Playing

The final button that I haven't talked about is the Play button. When you select a task, you can listen to the sound which has been assigned to it by clicking on the Play button. You can't change the assignment with More Sound (that's what the Sound control panel is for), but you *can* listen to what has been assigned already to see if you like it.

That's pretty much all there is to using More Sound. For those wishing to delve into the depths of More Sound's inner workings, the source code is on your GS+ Disk. If you have any problems be sure to contact me. A good way to let me know what's really going on with some of those problems is for you to fill out that nifty problem form that's provided on your GS+ Disk. GS+



Working With the Toolbox

Part 11: The List Manager

By Josef W. Wanklerl

I wrote my last installment of Working With the Toolbox in GS+ V4.N4—over a year ago. I had hoped that people would write in and tell me what other tool sets they wanted to see covered, but nobody did. So, I figured it was up to me to blaze the trail. I wrote myself a letter that I'll share with you now:

"Dear Joe,

I absolutely *love* your "Working With the Toolbox" series of articles. They are both informative and insightful. I've always wondered, though, how to use the List Manager from my own programs. I've read all the material there is pertaining to the List Manager (volumes 1 and 3 of the *Apple IIGS Toolbox Reference*, as well as the *Programmer's Reference for System 6.0 and 6.0.1*—I've even read all the technical notes) and there are still some fuzzy areas, namely on how to manage memRec structures. Can you give me some good examples of how to use a list control?

Love,
Joe"

In response to this letter, I decided that now would be a good time to cover the List Manager. I have had a lot of experience dealing with the List Manager. In fact an entire section of my Miscellaneous Library (MiscLib) code deals with the List Manager—specifically in managing memRec structures, and I recommend that if you're going to do any serious work with the List Manager that

you check out the MiscLib routines. I can personally vouch that the MiscLib will make dealing with the List Manager absolutely a breeze!

Only The New Stuff

There are a lot of "older" List Manager calls which are described in the *Apple IIGS Toolbox Reference: Volume 1*. For the most part, you can ignore all of them. I'm not saying that you shouldn't read the volume 1 chapter, though! There's a wealth of information presented in there. However, the calls that are in volume 3 are much easier to make and they also know all about resources.

LISTen to Me!

OK, so what exactly does the List Manager do? Well, the List Manager is a tool set which adds one control type to the Control Manager: a list control. While this sounds rather lame, in essence, that's all the List Manager really does. However, a list control is a complex being, and it requires a number of support routines to operate. The List Manager tool set provides most of these support routines. (There are, however, a few routines you have to write yourself!)

I'm sure you've all seen a list control before, so I won't go into the details of what a list control lets you do. Chances are, you know you need a list control for a particular program, but you just aren't too sure how to make it work. After all, there is a bunch of information about list controls out there, and digesting it all in one sitting is no easy task.

What List?

First off, let's discuss what's necessary to create and maintain a list control. First off, you have to create the list control. This is pretty easy. You create it just like you would any other control in the world. There are, however, a few special considerations you need to make when setting up the control template, which I'll discuss in a minute. You also have to create all the items that will go inside the list. This is where the fun part comes in. There are a number of ways to maintain the list items. The easiest way, for me at least, is to use the MiscLib routines. However, if you like doing things the hard way, you can write your own custom routines. (Chances are the code may be smaller if you write your own routines, but it will also probably be more special-case code which you'll have to fine tune every time you want to create a new list—the MiscLib routines are very general in nature and can be used for just about every list you'll ever want to make.)

First off, let's look at a sample list control template resource presented in Figure 1. It looks pretty much like any other control template, doesn't it? That's the beauty of controls—they're so easy to make! Let's look at some of the list control specific fields, though. (Note that this information is compiled from *all* sources of List Manager related material, so you can use this as a guide to know what exactly is contained in each field. If you read each reference one by one you'd eventually know, but the information is spread out over more than four different

Figure 1 — A List Control Template Resource

```
resource rControlTemplate ($00000001, $0000) {
    $00000001, // ID
    { 23, 4,125,380}, // rect
    listControl {
        $0000, // flag
        $3401, // moreFlags
        $00000000, // refCon
        $0000, // listSize
        $000A, // listView
        $0000, // listType
        $0000, // listStart
        $000A, // listMemHeight
        $000C, // listMemSize
        $00000000 // listRef
    }
};
```

Figure 2 — The flag Field

bit 0	fListString from template's listType field (0 for Pascal strings, 1 for C strings)
bit 1	fListSelect from template's listType field (0 for multiple selections, 1 for single selection)
bits 2-5	reserved (must be zero)
bit 6	testMemNever bit (0 if memNever is ignored, 1 if memNever is tested)
bit 7	ctlInvis bit (0 if control is visible, 1 if control is invisible)

texts and it can be a bit of a mess to quickly find exactly what you want to know.)

The first list specific field is the flag field. (OK, so it's not specific to list controls, but the flag field *does* contain different information depending on what kind of control it's for.) The meaning of the information contained in the flag field will be discussed later, but the content is presented in Figure 2.

The next field is the moreFlags field. (As with the flag field, the moreFlags field isn't specific to list controls but it does contain different information depending on what kind of control it's for.) Starting with System 6, you can set the fCtlTarget, fCtlCanBeTarget, and fCtlWantEvents bits. With the fCtlWantEvents bit set, the list control will automatically know what to do with keyDown and autoKey events. It will select the next or previous item in a list when you use the arrow keys, and you can also type the first few letters of an item name to select that item. With the fCtlCanBeTarget bit set, the list control will receive a focus frame when it becomes the target control. (If the list control is the only targetable control in a window, it's common practice to leave the fCtlCanBeTarget bit clear, otherwise it should be set.) If you would rather handle keyDown and autoKey events instead of having the List Manager do its job, you can leave the fCtlWantEvents bit clear. This is useful if the items you are putting in your list are not text strings or if the items aren't sorted. The other bits relating specifically to the list control tell how to reference the original list members when the control is created. The original reference can be by pointer, handle, or resource ID. (Usually you'll ignore this as your original list will probably be empty and you'll add in your items after the list is created. However, if you want all the list items to be added when the control is created, you have that option, too.)

The next list control specific field is the listSize field. This field simply contains the number of items initially in the list control. Unless you know exactly

how many members will be in the list control in advance, you can't set this field. Usually you'll just set this field to zero, create the list control, and then add in the members.

The next list control specific field is the listView field. This field tells the list control how many members are visible at a given time without scrolling the list. This field is closely related to the rect and listMemHeight fields. For lists which contain text strings, you can calculate the listView field with the formula: $((v2 - v1) - 2) / listMemHeight$. (Basically, each member in the list is listMemHeight pixels high and there are two additional pixels for the control frame. You just multiply the number of items you want to see in the list by the listMemHeight value, add two for the frame, and then use that as the height of the control. For simple text strings as members, the listMemHeight value should be ten.)

The next field is the listType field. This field controls whether only one selection is available at a time or if multiple members can be selected. The listType field also controls whether the list members are Pascal strings or C strings. (Some references also *incorrectly* have the listType field controlling whether or not the list scroll bar is drawn inside or outside the defining rectangle.) Of course if your list has a custom drawing procedure (discussed in a while), the Pascal or C strings bit doesn't mean anything. Be sure that you set the appropriate flag field bits to also match the bits in the listType field.

The next field is the listStart field. This field contains the item number of the member that will be shown at the top of the list control after it's created. Again, you'll usually ignore this field as your list will probably initially be empty.

The next field is the listDraw field. (This field is absent from the resource template in Figure 1 because the Rez template definition automatically fills this field in with a NIL value, meaning there is no custom draw routine.) This field contains a pointer to a custom member drawing procedure. If all you're going to have in your list control is strings of text,

you won't have to worry about a custom drawing procedure—you can just let the List Manager do all the work for you. However, if you want to draw something other than one string of text for each member, you'll have to write your own member draw procedure. The list control template I've given above is in a resource, so you may wonder how to set this field correctly if your program may load into memory at a different location every time. The answer is for you to leave this field alone in the resource template and change it with the NewList2 call when you set up your initial list members. This will, of course, be discussed in a bit more detail later.

The next field is the listMemHeight field. This field contains the height of an individual list member. For simple text strings, the height should be ten. For custom drawn list items, you must figure your own value to place here.

The next, and perhaps the most important, field is the listMemSize field. This field contains the number of bytes a list memRec will take up. It's difficult to explain the importance of this since I haven't discussed what a memRec is yet, but believe me, if you screw up this value, your list will be drawn all wrong, except for the very first member in the list. So, if you ever run into the case where the first member of your list is drawn correctly but everything else is trash, suspect that your listMemSize field is set wrong.

The final field is the listRef field. This contains a reference to the initial members to be placed in the list control. As before, you'll probably ignore this and add the members after the list has been created.

List Members

The most important data structure you need to worry about when working with lists is the list of member records. This data structure tells the list control what items are in the list. The list of member records structure is defined as a simple array of memRecs. A memRec (short for "member record") is a fixed length structure which defines a single list item. (The length is defined by the listMemSize field in the control

Figure 3 — The memRec Structure

<u>Offset</u>	<u>Field</u>
\$0-\$3	memPtr
\$4	memFlag
\$5-n	additional application-specific data

template.) The definition of a memRec is described in Figure 3. For "standard" list memRecs, the memPtr field is a reference to the string to be drawn for the member. If you're using a custom list drawing routine, the memPtr field can be anything you'd like. The memFlag field controls important information about the list member. The field is described fully in Figure 4. Let's now talk about each of the bits in the memFlag field in more detail. First off, bits 0 and 1 tell the List Manager what to do with the value in the memPtr field. The reference to the string can be by pointer, handle, or resource ID. While the references to the string may differ between individual memRecs, the type of string will be the same. The type of string (Pascal or C string) is defined by the listType field in the control template. If you have a custom member drawing routine, you can pretty much ignore the list reference type bits. The reason for this is because you'll probably only use one reference type and you'll know in advance what it will be. However, if you're going to write a replacement for the standard drawing routine, you'll have to honor these bits.

The final three bits in the memFlag field are related to the item's selection status. The memSelected bit tells the list control whether or not the item is actually selected. The memDisabled bit tells the list control if the item is grayed out or not. Note that if the memDisabled bit is set, the memSelected bit should not normally be set, but there is no formal restriction there. The reasoning behind this was so that you could select disabled members for some kind of help system. Finally, the memNever bit controls whether or not the item can ever be selected, regardless of the memDisabled bit. However, for the memNever bit to be honored, the memTestNever bit should be set in the flag field of the

control template. If you're writing a custom member drawing routine, you'll have to honor only the bits that you know your program will use.

Setting Up the Control

OK, so now that you know everything that makes up a list control, let's see how you set one up. First you'll create the control using NewControl2 (or NewWindow2 if the control is in a control list tied to a window). Next you'll create the list of member records. (This is the tricky part and will be discussed in the next section.) Then you'll add the members to the list using the NewList2 call. With the NewList2 call, you have the chance to override some of the values you initially provided with the initial control template. The values you can override are the listDraw, listStart, listSize, and listRef fields, and in overriding the listRef field, you can also change whether the list of member records is referenced by pointer, handle, or resource ID. You can pass a -1 value for any of those values to tell the list control to *not* override that field. After you've done that, your list will be set up and ready for user interaction.

Fun With Lists

Once you've got a list control set up, it's nice to be able to manipulate the list. The List Manager provides you with the following list control manipulation routines:

- DrawMember2 — This forces the list to redraw one member (or all members) of the list. You'll use this if you change something in a one or all of the list memRecs. (If the number of memRecs changes, you'll have to use NewList2 to tell the list control about the change. In doing so, you don't have to call DrawMember2.)

- NextMember2 — This asks the List Manager to search the memRecs, starting at any memRec, and tell you what the next selected member is.

- ResetMember2 — This call is similar to NextMember2 in that it searches the memRecs to find the next selected member. The difference is that the search *always* starts at the first member in the list, so you're always looking for the first selected member. Also, when the first selected member is found, its memSelected bit is cleared.

- SelectMember2 — This call selects a member in a list, deselecting all other selected members first.

- SortList2 — This call sorts all the list memRecs. Sorting will be discussed in a section below.

- ListKey — This is the routine which TaskMaster calls when a keyDown or autoKey event is detected and the fCtlWantEvents bit is set. If you have the fCtlWantEvents bit clear for your list control, you can manually call ListKey for keyDown events to get the same effect.

That's All!

Yep, that's right, that's almost all there is to the List Manager. A question you've probably had in your mind since the beginning of the article is: "But how do I add and subtract members from a list?" This is a very good question. The answer is that you have to maintain the list of member records all by yourself. The way you do that is, well, up to you. Maintaining the list of member records is not a very easy chore. It's not that maintaining the list is complex, it's just not very easy. I recognized this a long time ago, and hence wrote some MiscLib routines to do all the dirty work for me.

MiscLib

There was a time when I absolutely *hated* using the List Manager. In fact, when I

Figure 4 — The memFlag Field memRec

bits 0-1	00 = memPtr is a pointer to a string 01 = memPtr is a handle to a string 10 = memPtr is a resource ID of a string 11 = invalid
bit 2-4	reserved (must be zero)
bit 5	memNever bit (0 if member is selectable, 1 if member is not selectable)
bit 6	memDisabled bit (0 if member is enabled, 1 if member is disabled)
bit 7	memSelected bit (0 if member is not selected, 1 if member is selected)

was writing Replicator, our disk copying program from GS+ V3.N3, I farmed out most of the dirty list related stuff to Diz. (He wrote it in C, which is much more tolerant of variable length structures than Pascal is.) I believe it was after that experience that I decided to write some list related routines. Now, with the help of my MiscLib routines, using lists is one of my favorite things to do, believe it or not!

If you're going to use the MiscLib to help manage your lists, you should read that chapter of the MiscLib documentation. The documentation gives in-depth explanations of all the calls you can make. But, in the interest of keeping things together, I'll give some brief examples here so you can get a general feel of what MiscLib will do for you.

The MiscLib routines all depend on the list of member records being maintained in a single handle. Adding and subtracting members will grow and shrink the handle. So, to start a list, you simple create an empty handle (one with a size of zero). During the lifetime of the list, you make MiscLib calls to add and subtract members from the list. Then, when you're through with the list, all you have to do is dispose of the handle you originally created. (Of course, if you have memRecs referencing memory, you'll want to dispose of that, too.)

Here are some of the more important MiscLib calls you can make in regards to maintaining lists:

- AddMemRec — This adds a memRec to a list of member records.
- DeleteMemRec — This deletes a memRec from a list of member records.
- GetMemRecSize — This tells you how many memRecs are present in a list

of member records.

- FindMemRecPtr — This returns a pointer to a specific memRec in a list of member records.
- GetMemRec — This returns a copy of a specific memRec in a list of member records.
- NextMemRec — This is similar to the NextMember2 call.
- ResetMemRec — This is *almost* similar to the ResetMember2 call.
- SelectMemRec — This is similar to the SelectMember2 call.
- AddSelectMemRec — This is similar to SelectMember2 except nothing is deselected first.
- SortMemRec — This is similar to the SortList2 call.

You may be wondering why there are calls that are similar to routines that are already present in the List Manager. The reason for this is that the List Manager calls will only operate on list *controls*. The MiscLib calls operate on the list of members handle. This means that you can work with the memRecs without having them tied to a control.

Sorting

One of the things that the List Manager makes easy is the sorting of list memRecs. You can have the List Manager perform a case sensitive, case insensitive, or custom sort. When you call SortList2 (or the SortMemRec MiscLib routine) you give a pointer to a custom sorting routine. If the pointer is NIL, a case sensitive sort will be done. If the pointer is \$00000001, a case insensitive sort will be done. If the pointer is any other value, it is assumed to be a pointer to a custom sorting routine. A custom sort routine gets passed a pointer to two memRecs. It returns a flag value telling whether the first memRec is smaller than the first one

or not. Before System 6, you had to use at least some kind of assembly language glue code to write a custom compare procedure because the result was returned in the carry bit of the processor status register, and high level languages simply don't have access to the processor status register. With System 6, you can set bit 31 of the custom sort routine pointer and the result will be returned on the stack instead of in the carry bit.

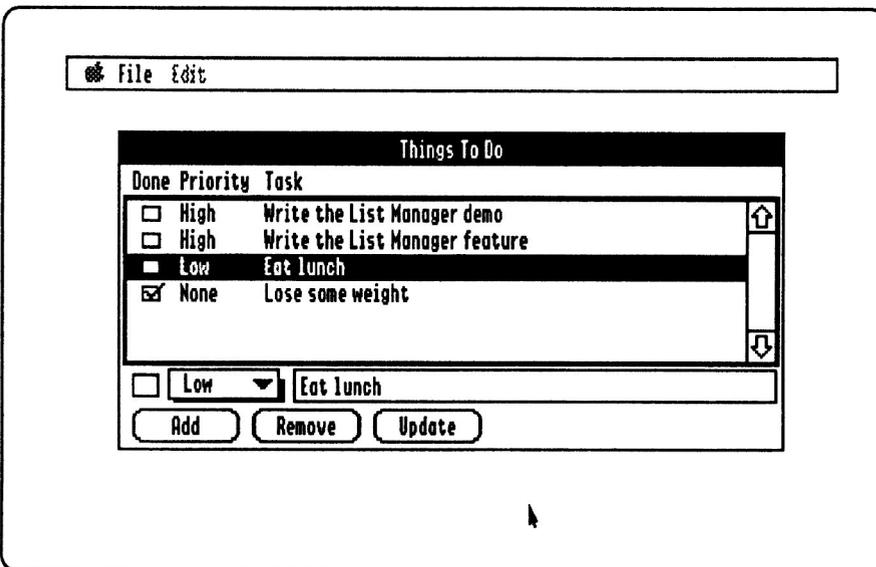
When you ask the List Manager for a case insensitive sort, it uses a routine, CompareStrings, added in System 6 to compare Pascal strings. This routine can be used by your program to compare strings at any time, provided the List Manager is started. The compare will always be case insensitive. Starting with System 6.0.1, the ability to compare GS/OS class 1 input strings was also added.

Custom Member Drawing

A custom member drawing procedure is passed a pointer to the memRec to draw, a pointer to the rectangle in which the member should be drawn, and the handle to the list control which the member is being drawn in. The first thing your drawing procedure should do is erase the rectangle with an EraseRect call to clear out whatever may have been there previously. Next you should do the drawing specific to your member. Be sure to check the memDisabled and memSelected bits if necessary and draw the list member appropriately. An example of a custom member drawing routine is presented in assembly language in IIGS Technical Note #74. (There are two other technical notes relating to the List Manager on the GS+ Disk so you can read up on as much list information as possibly can.) You can ignore the test of the clipping region since System 6 does all of that work for you. What's interesting in that source code example is the way that the item is drawn when the memDisabled bit is set. Basically a custom pen mask is set up after the complete member has been drawn, and then the item rectangle is erased using the custom mask. The previous pen mask is then restored.

List Manager Demo

By now you should be getting used to the demo programs and the common setup and event loop code that they share. For the List Manager demo, I've made a couple of modifications to the CheckFrontWindow procedure that you may have seen before. It's now more modular and much easier to maintain. If you're up to speed with the rest of the articles, you should be able to concentrate



only on the List Manager related material and not worry so much about all the events that are involved in order for control to get to the list related code. Before I go into detail describing the code, let me tell you what the program actually does.

When thinking about what kind of demo program would best showcase the List Manager, the only real idea that sprang to my mind was something that would manage a list of to-do items, much like what I have on my Newton MessagePad. (That's my favorite part of the MessagePad.) Of course the average everyday stock IIGS system doesn't come with handwriting recognition, so entering items into the list takes place with standard IIGS controls: a check box, a pop-up menu, and a LineEdit control (see the screen shot). To add a new item to the list, you set the three input controls the way you want them and then click on the Add button. To remove an item from the list, simply select the item in the list you want to remove and then click on the Remove button. To change the information contained in a list item, first select the item in the list you want to change. You'll see that the input area will change to reflect what the selected item in the list contains. You can then change the input area and click on the Update button. If all you want to do is change the done status for a list member, just double-click on the list member. That's pretty much all the List Manager demo program does. It doesn't save the list, so every time you run the List Manager demo program you'll be presented with an empty list. Of course, you can remedy this on your own. (After all, it's a List Manager demo program, not a demo program to show you how to save stuff.)

Setup in Depth

OK, let's look at the setup for the demo program in a little more detail. The first thing you should want to know is the structure of the memRecs. Figure 5 explains the memRec definition for the

List Manager demo program. Now, since the memPtr field is a handle to a Pascal string, bits 0 and 1 of the memFlag field must be set to accommodate an handle. However, I'm using a custom list member drawing procedure, so I can ignore this. (I set the bits appropriately anyway, though. It's just a nice thing to do. Plus it makes initial testing easier because you can use the standard list drawing procedure.) An important thing to note throughout all the code is that all the memory relating to the list control has a modified memory ID to make cleanup easy. All that needs to be done is to call DisposeAll on the memory ID and all the memory is disposed of automatically. Sometimes it's not always possible to have a unique memory ID for a list (pretend you have over 16 lists—that's more lists than there are auxiliary IDs) so what you'd have to do in that case is to run through the entire list and dispose of any member related memory one member at a time.

Adding, Removing, and Updating

There aren't any hidden tricks in these routines. They simply do exactly what you'd expect them to. For adding, the information for the new member is obtained from the input area controls and placed in a new memRec structure. The memRec is then added to the list of member records and the list control is notified of the addition by using the NewList2 call. With the new member in the list, the list is finally sorted. For removing, the selected memRec is found, the memory associated with it is disposed of, and then the memRec is deleted from the list of member records. The list control is notified of the deletion by using the NewList2 call, and there is no need to sort the list since deleting a member does not change the sorted state of a list. For updating, the selected memRec is found, and its contents are replaced by what's in the input area controls. Since there was no addition or deletion, the NewList2 call doesn't need to be made, however the list *does* need to be sorted. (Also, the code for when you double-click

on a list entry does almost exactly the same thing as an update except that only the done state of the selected list member is changed.)

Other Considerations

Whenever a hit is detected on the list control, the task record is checked to see if the hit was a double-click or not. If the hit was a double-click, the appropriate code is called. If the hit was anything else, the current list selection is determined, and the input area is changed to reflect the selected member.

Another thing to keep track of is the Add, Remove, and Update button states. They must be correctly set depending on whether or not the list control is empty or not and if the LineEdit control is empty or not. This is handled by the CheckTask procedure.

Drawing

I think the most interesting code in the entire demo is contained in the custom member draw procedure. The "common" way to highlight the selected member is to simply call InvertRect on the item's rectangle after everything has been drawn. However, there are quite a few programs out there that highlight selected members in green. (Some notable programs that do this are GS-ShrinkIt and SuperConvert.) The trick to doing this is to highlight the member rectangle in green *first*, and then draw the member content over the green background. For drawing the member content, the List Manager demo program needs to draw three things: the done status, the priority, and the task name. The done status is represented by an icon and drawn by using the DrawIcon Toolbox call. The priority is then drawn using the DrawString Toolbox call, and finally, since the length of the task string is variable, it's drawn using the DrawStringWidth Toolbox call. The hardest part of coding a custom member drawing procedure is figuring out how far over to draw each item so they line up under the correct heading.

Figure 5 — The memRec Definition For the Demo Program

```
ListRef = Record
  memPtr : Handle;           { Contains a handle to the task Pascal string }
  memFlag : Byte;           { Will be either $01 for unselected or $81 for selected members }
  memDone : Integer;        { Done check box flag value }
  memPriority : Integer      { Priority value (see notes below) }
End;

{ Priorities are values between 1 and 4. Value 1 means the item is not prioritized. }
{ Value 2 is high priority, 3 is medium priority, and 4 is low priority. }
```

Sorting

The second most interesting code in the List Manager demo is the custom compare procedure that is used for sorting the list. Of all the routines, I had the most difficulty coding this one. (Not because it was difficult to code, it was just because my brain couldn't figure out how to sort based on multiple keys. I had everything exactly backwards at one time so the list was sorting in reverse order. I also had it so that it was sorting completely randomly at one time, too!) The compare routine sorts based on three keys. The first is the done status. An undone member should definitely come before a done member. The next key is the priority. High priority items come before

Medium priority items, which come before Low priority items. Unprioritized items always come last. The final key is the task string. I use the CompareStrings procedure to get a case insensitive comparison between task strings.

That's All, Folks!

Using the List Manager tool set is a piece of cake. The source code for the List Manager Demo program covers just about every List Manager related topic you should ever need. If you find you need to do more than the List Manager Demo program does, you should read up on the Toolbox and System 6 references, and most likely the MiscLib List section

documentation. If you had trouble following this article, or with the Toolbox references, let me know and I'll attempt to clarify.

I am still writing "Working With the Toolbox" installments "on request." What that means is that you write in and tell me that you'd like to see a program that uses the <insert tool set name here> tool set and I'll work on it. "Common" tool sets will take precedence over obscure ones (i.e. TextEdit, Font Manager, and Standard File will take precedence over the Apple Desktop Bus tool set). I look forward to seeing your requests, and maybe I won't have to write myself a letter next time!
GS+

Warranty Disclaimer And Copyrights

EGO SYSTEMS' LICENSOR(S) MAKES NO WARRANTIES, EXPRESS OR IMPLIED, INCLUDING WITHOUT LIMITATION THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE, REGARDING THE SOFTWARE. EGO SYSTEMS' LICENSOR(S) DOES NOT WARRANT, GUARANTEE OR MAKE ANY REPRESENTATIONS REGARDING THE USE OR THE RESULTS OF THE USE OF THE SOFTWARE IN TERMS OF ITS CORRECTNESS, ACCURACY, RELIABILITY, CURRENTNESS OR OTHERWISE. THE ENTIRE RISK AS TO THE RESULTS AND PERFORMANCE OF THE SOFTWARE IS ASSUMED BY YOU. THE EXCLUSION OF IMPLIED WARRANTIES IS NOT PERMITTED BY SOME JURISDICTIONS. THE ABOVE EXCLUSION MAY NOT APPLY TO YOU.

IN NO EVENT WILL EGO SYSTEMS' LICENSOR(S), AND THEIR DIRECTORS, OFFICERS, EMPLOYEES OR AGENTS (COLLECTIVELY EGO SYSTEMS' LICENSOR) BE LIABLE TO YOU FOR ANY CONSEQUENTIAL, INCIDENTAL OR INDIRECT DAMAGES (INCLUDING DAMAGES FOR LOSS OF BUSINESS PROFITS, BUSINESS INTERRUPTION, LOSS OF BUSINESS INFORMATION, AND THE LIKE) ARISING OUT OF THE USE OR INABILITY TO USE THE SOFTWARE EVEN IF EGO SYSTEMS' LICENSOR HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. BECAUSE SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF LIABILITY FOR CONSEQUENTIAL OR INCIDENTAL DAMAGES, THE ABOVE LIMITATIONS MAY NOT APPLY TO YOU. EGO Systems' Licensor's liability to you for actual damages from any cause whatsoever, and regardless of the form of the action (whether in contract, tort (including negligence), product liability or otherwise), will be limited to \$50.

And furthermore,

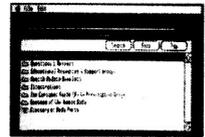
Some of the programs on this GS+ Disk contain materials from the ORCA/C, ORCA/Pascal, and ORCA/Modula-2 Run-Time Libraries, copyright 1987-1993 by Byte Works, Inc. Used with permission. All other material is copyright 1994 by EGO Systems, unless otherwise noted. Purchasers of the GS+ Disk may use the source code and/or libraries in their own programs, provided that the following notice appears somewhere in the program or in its documentation:

"Portions of this program were created using source code and/or libraries from GS+ Magazine, and are copyright 1989-1994 by EGO Systems. Used with permission. For more information on GS+ Magazine, write to:

GS+ Magazine
P. O. Box 15366
Chattanooga, TN 37415-0366"

A World of Information on your Apple IIGS...

Sequential Systems introduces you to DiscQUEST™, and the world of information now available on CD-ROM. With DiscQUEST™ software and your IIGS, you now can access literally THOUSANDS of MEGABYTES of Text, Graphics and Audio data currently available on CD-ROM. Search and explore huge CD-ROM libraries of valuable, interesting data from popular titles like "The Family Doctor", "Total Baseball", "Parenting" or the "Great Literature" series. And since DiscQUEST™ uses Standard, Off-the-shelf CD-ROMS for MAC & MS-DOS, your investment in Multi-Media will be intact for virtually any computer you'll be using down the road! (Call for a list of supported titles.)



Call your dealer or Sequential Systems now for more information!



**NEW Features!
NEW Color Graphics!
NEW Titles!**

1-800-759-4549
SEQUENTIAL SYSTEMS

Buying Ad Space in GS+ Magazine

Full Page Ads

Each page should be 7 & 1/8 inches wide and 9 & 1/2 inches tall

1 page - \$150
2 pages - \$250

Partial Page Ads

Partial page ads can be any shape or orientation, but must be no more than 7 & 1/8 inches wide and 9 & 1/2 inches tall

1/3 page (no more than 22.5 square inches) - \$60
1/2 page (no more than 33 square inches) - \$90
2/3 page (no more than 45 square inches) - \$120

If you want your ad to appear on the back cover or inside back cover, add \$100. Call for cover reservations.

Spot color is available for an additional \$15 per page. Call for details.

Save an additional 20% by placing the same size ad in more than one issue!

Prices good through August 22, 1994

Request your ad space today to guarantee these low prices!

DEADLINES:

July-August 1994 (Volume 5, Number 6) July 1, 1994
September-October 1994 (Volume 6, Number 1) August 22, 1994

All ads must be camera-ready copy. Payment must accompany ad. Make checks payable to EGO Systems, or call us to charge it on your credit card. For more information contact:

GS+ Magazine
P.O. Box 15366
Chattanooga, TN 37415-0366
Voice phone: (615) 843-3988

If you need to ship your ad to us using a service other than the U.S. Postal Service, please call to make arrangements.

If you wish to place an ad for a product we have not reviewed, we request that you include a review copy with your ad.

Glossary

In each issue of *GS+* Magazine, we present a glossary of some of the more common terms in the IIGS world and some of the more uncommon terms that we use in each issue. If you have a term or bit of jargon that you would like to see explained, let us know and we'll try to get it in a future "Glossary" installment. Also, don't forget about the glossary that's in your IIGS owner's manual! At this point, it contains many more terms than the *GS+* Glossary!

Past installments of the *GS+* Glossary can be found on your *GS+* Disk in the plain ASCII text file, *Glossary* (see "How to Use Your *GS+* Disk" for more information). Entries marked with an "*" have appeared in previous installments of the *GS+* Glossary and are repeated here for our beginning readers or because they have relevance to topics discussed in this issue.

CD-ROM *

"CD-ROM" stands for "Compact Disk-Read Only Memory." Basically, this is simply a compact disk (physically similar to the one's you buy at a music store), that can contain text, pictures and sound instead of just music. Generally speaking, CD-ROMs are not usable in an audio (music) CD player.

A CD-ROM can hold well over 600 megabytes of information, making it ideal for distributing large amounts of information (like an encyclopedia).

LineEdit

The term "LineEdit" can refer to one of two things: The LineEdit tools that are built into the IIGS, or a LineEdit control that has been created by the LineEdit tools. The LineEdit tools allow IIGS programmers to create LineEdit controls which a user can type a single line of information into. These controls are usually used for passwords and other types of data entry that only require a single line of text. (Multi-line text entry must be done using the TextEdit tools. [See below.]

List Manager

The List Manager is a tool set that allows IIGS programmers to create and maintain one-dimensional lists of items. Users can interact with these lists by clicking on an item (or items) in the list with the mouse.

MS-DOS FST *

The MS-DOS FST (File System Translator) is a read-only FST that comes with Apple IIGS System Software v6.0.1.

If you have an Apple SuperDrive (or an equivalent disk drive) attached to your IIGS with an Apple II SuperDrive controller card, or a floptical disk drive attached to a SCSI interface card, this FST allows your IIGS to read disks formatted for use with MS-DOS. At this point in time, this FST will not allow you to write data to an MS-DOS disk.

MTBF

"MTBF" stands for "Mean Time Between Failures." This is a term used to specify the amount of time (in hours) that a hard disk should function before an error occurs when trying to use the disk. The larger MTBF value a particular drive has, the longer it should last during normal use. Of course, the key phrase here is "normal use." Most MTBF figures are calculated under ideal operating conditions, and do not usually reflect the average computer user's work area or habits.

rSoundSample *

"rSoundSample" is the name given by Apple to a sound sample that is stored in the resource fork of a file. These are the types of sounds that the Sound control panel plays. However, the term rSoundSample is also commonly (but incorrectly) used to refer to a type of file (file type \$D8 and auxiliary type \$0003) that has the express purpose of holding one or more rSoundSamples. The proper name for this type of file is actually a "sound resource file."

SCSI *

"SCSI" stands for "Small Computer Systems Interface" and is pronounced "Scuzzy." Put simply, SCSI is a standardized way for computers to communicate with peripherals (hard disks, scanners, etc.) Because it is a standard, a single SCSI device can be used on many different types of computers. For example, most SCSI hard disks can work "out of the box" with both the Macintosh and NeXT computers. With the appropriate SCSI adapter card, the same drive could be used on a IIGS, IIE, Commodore Amiga, or IBM PC clone. The SCSI standard also allows devices to be daisy-chained together so that you can have more than one SCSI device online at a time.

SCSI-2

SCSI-2 is a newer version of the original SCSI standard. Almost all new SCSI devices are being made to conform to this newer standard. This means that these devices will sometimes not work (or not

work reliably) with older SCSI devices and controller cards. However, most SCSI-2 devices are backwards compatible with older SCSI devices. For IIGS owners, the question is, "Will my SCSI controller work with SCSI-2 devices?" The answer is that the RamFAST SCSI card is compatible with most SCSI-2 devices and that the SCSI cards made by Apple should work with SCSI-2 devices that are backwards compatible with the original SCSI standard.

SCSI Device *

A SCSI device is any device that uses the Small Computer Systems Interface (SCSI) to talk to other devices.

SCSI Terminator *

Like all other computer equipment, SCSI devices communicate with each other using electrical signals. When these signals come to the beginning or end of the chain of SCSI devices, they need to be stopped, or they can "bounce back" to the other end of the chain and confuse the other SCSI devices. The job of a SCSI Terminator is to stop those signals before they bounce back. Therefore, both the first and last device in a SCSI chain should have terminators. Devices in the middle should not have a terminator.

Some devices are internally terminated (that is, the terminator is inside the device case), while others require an external terminator to be attached to one of the devices external SCSI ports. The best way to tell if a device is internally or externally terminated is to check the manual that came with it. (Also see "SCSI" above.)

TextEdit

The term "TextEdit" can refer to one of two things: The TextEdit tools that are built into the IIGS System Software, or a TextEdit control that has been created by the TextEdit tools. The TextEdit tools allow IIGS programmers to create TextEdit controls which a user can type multiple lines of text into. A TextEdit control can hold as much information as memory will allow, and that information can be formatted with any combination of fonts, sizes and styles. Most IIGS editors (with the exception of AppleWorks GS, and BeagleWrite GS) use the TextEdit tools to implement their editors. However, for all of TextEdit's power, it does have a few major drawbacks: It only supports one ruler per document, and it does not support the superscript and subscript styles. GS+

GS+ Ordering Information

GS+ Magazine is published bimonthly and sold for \$4.00 an issue for the magazine only, and \$8.00 an issue for the magazine + disk. But, if you sign up for a 1-year subscription (six issues) or a 1/2-year subscription (three issues), you can save 11-25%! To sign up, send this completed form (or a photocopy) along with a check or money order (payable to "EGO Systems"), or your credit card number, to:

GS+ Subscription Services
P.O. Box 15366
Chattanooga, TN 37415-0366

Name:..... Phone: (.....).....

Address:.....

City:..... State:..... Zip:.....

Please ✓ item you wish to order:

- 1-yr subscription (mag + disk) - \$36
- 1/2-yr subscription (mag + disk) - \$20
- 1-yr subscription (mag only) - \$18
- 1/2-yr subscription (mag only) - \$10
- Sample issue (mag + disk) - \$8
- Sample issue (mag only) - \$4

Please ✓ delivery method:

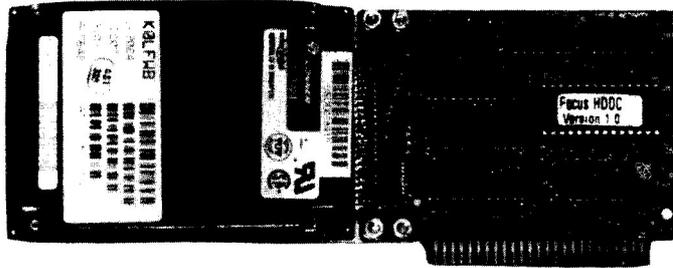
- Third Class to U.S. - free
- First Class to U.S., or Canada/Mexico, or Foreign surface - \$2.00 per issue
- Foreign Air Mail - \$5 per issue

Please ✓ payment method:

- Check or Money Order enclosed
- Bill my MasterCard
- Bill my VISA
- Card #:.....
- Expiration Date:/.....
- Signature:.....

If you prefer to use your credit card to order by phone, give us a call at 1-800-662-3634, Monday through Thursday, between 9 a.m. and 6 p.m. Eastern Time. (9 a.m. till 5 p.m. on Fridays.) All subscriptions will start with the next issue published. GS+ Magazine is published bi-monthly, so please allow up to 12 weeks for delivery of the first issue of a subscription. Tennessee residents add 7.75% sales tax. System 6, 4MB of RAM, and a hard disk are recommended for magazine + disk subscriptions. Please remember to indicate delivery method and include additional charge if indicated.

FOCUS HARD CARD FOR THE APPLE II



FEATURES:

- * Small quiet Internal Hard Card for the Apple IIs and Iie systems.
- * Complete 2.5" Hard drive with Controller.
- * Hard disk sizes from 20 to 250 Megabytes.
- * Easy to install, just plug in and power up the system, No cabling required.
- * Uses existing internal power supply.
- * Full size Apple II card footprint
- * Can occupy any slot except slot #3.
- * Up to 2 FOCUS drives per system using the standard Apple II power supply.

- * Drives can interchange between IIs and Iie without loss of data or configuration changes.
- * Price includes hard card, GS/OS and ProDOS system files, and utility software. Nothing else required.

PRICING:*

- | | |
|---|---|
| * FOCUS 20, 20 Megabyte Hard Card....\$249.00 | * FOCUS 80, 80 Megabyte Hard Card....\$399.00 |
| * FOCUS 40, 40 Megabyte Hard Card....\$299.00 | * FOCUS 120, 120 Megabyte Hard Card....\$529.00 |
| * FOCUS 60, 60 Megabyte Hard Card....\$349.00 | * FOCUS 250, 250 Megabyte Hard Card....\$699.00 |

TERMS:

- | | |
|---|--|
| * California residents add 8.25% sales tax. | * American Express credit card accepted. |
| * Add \$15 for shipping & handling. | * User group, school and dealer discounts available. |
| * Add \$7 for COD orders. | * 1 Year warranty on controller card and disk drive. |

Parsons Engineering

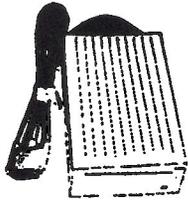
5010 Rimhurst Ave.
Covina, California 91724

***For an extra 10% discount mention GS+ Magazine ***

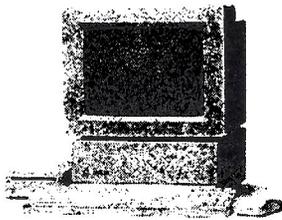
Sales/Service: (818) 966-5538
Fax: (818) 966-5701

Apple II / Macintosh

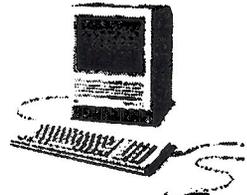
Your #1 Source for Apple® Parts & Systems!!



Genuine Apple 3.5 drive
Only.....\$169.



•• Worlds Least Expensive ••
•• Color Macintosh!! ••
Includes Mac LC, 2 meg RAM, 40 meg hard drive,
color monitor, 650k disk, keyboard, and mouse.
Mac LC 2/40.....\$699.



Apple II Parts

II+ keyboard	\$29.
All+/Ile power supply.....	79.
Allc keyboard	89.
Allc platinum keyboard	99.
Allc logic board	129.
Allc case	19.
Allc logic board	149.
Allc keyboard	69.
Allc case	19.
Allgs case	29.
Allgs logic board	229.
Allgs power supply	119.
Allgs keyboard	119.

Apple II Specials

Apple II mono monitor	\$99.
Disk II floppy drive	129.
Iigs RGB monitor (genuine Apple)	229.
Iigs 3.5 floppy	169.
Iigs 5.25 unidisk	169.
Apple personal modem 1200	49.
3.5 Unidisk	159.
Ilc monochrome monitor	119.
Imagewriter I (w/cable).....	179.
Iigs 256k memory expansion board	49.
Ile 64k/80 col board.....	45.
Iigs cooling fan	19.
Imagewriter II localtalk option.....	139.

Apple Iigs system

Includes: 1.25-meg of RAM, RGB monitor, Apple 3.5 drive, keyboard and mouse.

Only.....\$699.

Exchange Specials

Iigs logic.....	\$199.
Ile logic.....	99.
Ilc logic.....	99.
Mac Plus logic.....	99.
Mac SE logic.....	99.
Mac Plus power supply	99.
Mac SE analog bd.....	99.
Mac SE power supply	99.
Mac LC logic bd	99.

Macintosh SE

Includes: 1-meg of RAM, 20-meg internal hard drive, keyboard and mouse.

Only.....\$349.

Mac CPU's

Mac Plus	\$299.
Mac Classic 4/40	499.
Mac LC 2/40	399.
Mac II 1/0	399.
Mac IICx 1/0	699.
Mac IICi 0/0	899.
Mac IIfx 0/0	1299.

Apple II CPU's

All plus	\$99.
Apple Iie w/128k.....	199.
Apple IIC	249.
Apple IIC plus.....	299.
Apple Iigs	349.

School and Government P.O.'s accepted.

**Shreve
Systems**

1200 Marshall St Shreveport LA 71101
FAX 318-424-9771 Tech. 318-424-7987

We buy Mac and Apple II equipment, large or small quantities.
Please call our purchasing department. 318-424-9791

All equipment is used or demo unless otherwise stated.
Equipment carries a 120-day warranty.
Returns subject to a 15% restocking fee.

800-227-3971

GS+ Magazine
P O Box 15366
Chattanooga TN 37415-0366

Forwarding & Return Postage Guaranteed

Bulk Rate
U.S. Postage
PAID
Chattanooga, TN
Permit NO. 616

Downloaded from www.Apple2Online.com