GS+

The *First* Apple IIGS Magazine + Disk Publication!

$3.00
(magazine only)

$8.00
(magazine + disk)

July
August
1990

Volume 1
Number 6

IN THIS ISSUE:

So Much Great Stuff, We Don't Have Room For A Cover Photo!!

WRITER'S BLOCK

In most every issue, I try to come up with something brilliant to say in this column. Usually, I have to settle for simply not making a fool of myself. This time however, I do in fact have something very important to say. Ahem...

WEAR YOUR SEATBELT!

Why did I say that? Well, just the other day, Noreen and I were driving Joe to the sprawling Chattanooga Airport to put him on his plane for KansasFest. As we were traveling down Bailey Avenue, just before you get to National Street (of course, this means nothing to 99% of you, but I've never been afraid of pandering to 1% of anything) we were rear-ended... **hard.** Fortunately, we were all wearing our seatbelts and miraculously, no one was hurt (even more amazing when you consider that there were 5 cars involved in this accident). Noreen and I have stiff necks which should go away, and Joe, well, Joe is at KansasFest right now and he's too hyped up to feel any pain. In fact, no one in any of the 5 cars involved was hurt. We were **all** wearing our seatbelts.

So, if you are one of those folks that can't be bothered to buckle up, please take some time to think about what sort of effect a busted face could have on you and your loved-ones.

Diz

LETTERS

Dear Sir,

I received my subscription to *GS+*. In my subscription letter, I asked to make it retroactive to your first issue, but I received issue number 5 in the mail. Will you be able to make it retroactive?

Thomas C. Bailey
Houston, TX

The short answer is, no. For about the first 6 months we were in business, we did indeed allow folks to start their subscriptions retroactively. However, now that we have more than a handfull of subscribers, and, (more importantly) since we have had an issue sell out, it has become very difficult for us to keep correct track of retroactive subscriptions. So, we just don't do that anymore. If you want to obtain a back issue of GS+, see the back issue information on page 19.

Diz

Dear Steve,

I was interested in your recent review of Graphic Writer III. I share your enthusiasm for the program despite the fact it can drive one up the wall with its foibles. One problem that caused me a bit of grief is one you mentioned — its inability to handle large sized fonts. I went around and around trying to print out a title page of a newsletter that used fonts in sizes 36 and 48. Finally, after a bit of experimenting, I decided that the difficulty is with the GW.ImageWriter driver itself. Using New York Headline as the font, I printed out text in sizes 24, 36, 48, and 72 with both AppleWorks GS and GraphicWriter III using three ImageWriter drivers: the plain vanilla one included with System Disk v5.0.2, the Claris driver on AppleWorks GS and the GW driver with GraphicWriter III. My results were:

AppleWorks GS: Simply won't accept any font larger than 48 points. The plain and Claris drivers will print whatever AWGS will accept. The GW driver won't print AWGS text at all.

GraphicWriter III: Accepts all of the size fonts I tried. (Haven't tried anything larger than 72 points, however.) The plain and Claris drivers print them out OK. The GW driver will print out all sizes in "Faster" mode but will only print up to and including 24 points in "Best" mode.

I frankly can't tell the difference in quality between the three so I usually just use the Claris driver all of the time even with GraphicWriter III.

One suggestion: you say you put out *GS+* with a LaserWriter IINT. Some of us would love to use one but are just plain intimidated by the thought of getting a LaserWriter to run with our systems. How about an article or articles detailing how you manage it, how it's connected, what fonts are available, how you handle graphics (printed or pasted in), etc.?

One other thing about your publication. Pleez, the possessive of IT is ITS, not IT'S!! Sorry, but that ubiquitous mistake drives me crazy, and I expect I'm not the only one. Criticism aside, I hope that some of this drivel will be useful. Thanks again for a good magazine.

Constance L. Graves
San Jose, CA

Well, if that's drivel, I'm a Macintosh™©® salesman! ("Only $5000 down and you can have it paid for in 48 months! What? You want color? OH! Do you have a house?") That's some great information, and a great suggestion. Look for a LaserWriter Prep (an inside joke) article in a future issue of GS+.

As for the "ITS" problem—it drives Noreen crazy too! ("If you guys do

(continued on page 2)

CONTENTS

FEATURES

PROGRAMS

REVIEWS

DEPARTMENTS

STEVEN W. DISBROW
Publisher, Editor

NOREEN RIBARIC
Associate Editor, Layout

JOSEF WANKERL
Contributing Editor

FingerPrint GSi is used to freeze the screen so the screen photographs can be taken.

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

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

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

GS+ is published bimonthly by:
EGO Systems
4126 Mountain Creek Road #20
Chattanooga, TN 37415-3428
(615) 870-4960

If you have a submission for *GS+*,
send it to:
GS+ Submissions
c/o EGO Systems
P.O. Box 15366
Chattanooga, TN 37415-0366

Subscription rates for one year
(six issues) are:
Magazine only - $15
Magazine + Disk - $36
Canadian and Mexican orders add $1.
Other foreign orders add $5 for surface delivery or $25 for airmail.
Send subscription orders, ads, inquiries, and address changes to:

GS+ Subscription Services
c/o EGO Systems
P.O. Box 15366
Chattanooga, TN 37415-0366
or call (615) 870-4960

pro-gsplus BBS (615) 875-4607
2400 Baud/8 data bits/no parity/1 stop bit

GS+ is produced on an Apple IIGS using GraphicWriter III, AppleWorks GS v1.1, and an Apple LaserWriter IINT.

this again, I'm gonna stuff a toner cartridge in yer face!" Such a delicate flower.)

Joe and I both promise that we'll do better from now on. Honest. No, really! Look at this paragraph for example. Its not got a single mistake in it's body!

Diz

Dear Steve,

I was sorry to hear ***GS+*** V1.N4 was unavailable. May I purchase a copy of the "Beginner's Guide To the Finder Part 3—All About Icons"?

Alan B. Latz
McHenry, IL

Well, we have been toying with the idea of reprinting the entire "Beginner's Guide To The Finder" in a separate edition, but we are not too sure how many folks would go for it. It would probably cost about $3 and contain all of the "Beginner's" articles that we have published. What do you think?

Diz

Dear ***GS+***,

I'm sure that you've heard this a thousand times, but I think you should hear it again: ***GS+*** is great! I haven't read a magazine that even comes close to comparing with yours. When I read something in ***GS+***, I know that it's the unbiased truth. Your "Rumors, Wishes & Blatant Lies" articles are the funniest I've ever read. The thing I like about your programs is that all the version 1.0's work as promised, unlike a lot of new programs (AppleWorks GS v1.0, HyperStudio v1.0, etc.). Although, I do think that it's kind of sad that ***GS+*** is so great when compared to other Apple II magazines. Every magazine should be as good as ***GS+*** is.

John Larkowski
Chippewa Falls, WI

Dear ***GS+***,

Just received your May/June issue in the mail today, and wanted to thank you for the nice review of our SD45RM/A2S. I would also like to respond to a few issues you raised in the review.

CMS has never really left the Apple II hard drive business, as you mentioned in the Unfortunately... section. In fact, we have been selling Apple II hard disk subsystems through our dealer network for over three years. We simply saw no need to do any major upgrades until we ran into compatibility problems with System Software v5.0. Further, all of our MacStack Series hard drives, in capacities from 20 to 140 MB, have always been compatible with Apple's Revision C SCSI card.

On another note, there is a correction which needs to be noted: The driver we are writing will be for any SCSI hard drive on the CMS card. Like Apple's driver, it will be a universal driver for the card and not the drive. Otherwise, I thought that your review was fair, honest, and accurate....

Bill Stoker
CMS Apple Technical Support

For more stuff that I screwed up in that review, check out Errata on page 14.

Diz

Dear Editor,

I recently read your Rumors, Wishes and Blatant Lies column in ***GS+*** V1.N5. Contrary to what you describe, the Applied Engineering Conserver does not require any additional cable or connectors to use with two Apple 3.5 or AE's own 3.5-inch drives. The only time you need to use the daisychain cable is if you want to connect a third drive, an Apple 5.25-inch drive, Disk II or compatible.

If you have only two 3.5-inch drives, put the first drive (Drive 1) in the left hand bay, and connect the cable from the drive in the right hand bay (Drive 2) to the daisychain port on the left hand drive. The cable from Drive 2 must be routed around the post behind the drive. Excess cable can be stuffed into the space behind Drive 1. You can now easily use two 3.5-inch drives.

In addition, the Conserver has 6 outlets vs. [4] for the System Saver, and three front panel switches, allowing a greater number of switching arrangements that is possible with the Kensington product. It does not however, have the same aesthetic appeal as the System Saver, but does make for a neater desktop.

By the way, I have no affiliation whatsoever with AE. [I am] just a satisfied customer.

Michael D. Sobel
Massapequa, NY

COMING IN A FUTURE ISSUE OF GS+

Features:
Using A LaserWriter With Your IIGS
Beginner's Guide To Telecommunications

Programs:
Gee, Sound!
Fractals-GS v2.0

Reviews:
The Apple II High-Speed SCSI Card
InnerExpress
Renaissance
Orange Cherry Talking Schoolhouse
Quickie Hand Scanner
Design Master vs Genesys
Qix
Cribbage King • Gin King

PROGRAMMER'S QUEUE & A

Queue:
I have enjoyed your first few issues, particularly your advice and articles on learning to program.

After reading the books you recommended, and others, I am still struggling to make use of all the power in the Toolbox. Part of the reason is that most of the books about programming the GS were written before the newer tool calls were available (TextEdit, SuperControls, etc.) Also, books which try to cover the whole field can only show the use of a few tool calls from each tool set.

This is where *GS+* comes in: you could take us through a different tool set each issue. How about an article on using the List Manager, for starters?

Gareth Tucker
Deep River, CT

A:
Well, one of the best ways to learn to program the newer tools that you mention, is to take apart the source code on your *GS+* disk. However, your idea for a series of articles, each spotlighting a particular tool set, is very good. Look for the first installment in just such a series in the next issue or two of *GS+*.

Diz

Queue:
[I have some suggestions for] improvements to your programs:

EGOed (*GS+* V1.N2)
1) A striped title bar.
2) An I-beam cursor.
3) A "Color" menu (like the Finder's) for colored text.

DeskColor (*GS+* V1.N4)
1) The ability to edit patterns with a fat-bits dialog box.
2) I was using DeskColor with an aqua desktop. I needed to make some more space, and since CDevs are expendable, I deleted it. I was expecting the desktop to return back to the standard (periwinkle). The next time that I started up, I had a white desktop! The thing is, I only had a white desktop on the startup application (Finder), when I ran other applications, the color changed to periwinkle. But, when I came back to the Finder, it was white again! Why? And please fix it.

Rotator (*GS+* V1.N3)
1) Make it faster!!! That's all.

NoDOS (*GS+* V1.N1)
1) I'd like to see those improvements that you have scribbled on a napkin. I think that NoDOS is great! But, as with any program, it could be even better. I've been waiting for the update ever since I got the first issue.

John Larkowski
Chippewa Falls, WI

A:
Thanks for the suggestions and questions! Taking them one at a time:

EGOed
I'd love to put in an I-beam cursor, but TML Pascal II seems dead set against it. This should be resolved when I rewrite EGOed in ORCA/C. Hmm, lots of folks seem to be asking for a striped title bar, maybe, but I think a solid black title bar looks really cool. If you have access to a resource editor like Genesys, you can change that yourself. As for colored text... does anybody really use that? (For anything other than ransom notes that is.) If enough people ask for it though, I'll put it in there.

Diz

DeskColor
I plan to put a pattern editor in DeskColor just as soon as possible. Maybe next issue...

Diz

(And Joe Wankerl adds)
I, too, was perplexed by the persistence of the background pattern when I first ran across this problem, but then I remembered that the Finder stores the background pattern in the Finder.Root file. The Finder updates this file whenever you have a change to the desktop and your Preferences are set to save changes. So what happened is the last time the Finder.Root file was saved, you had a white desktop (I assume). With DeskColor installed, it automatically overrides the Finder's default saved background pattern. When you deleted DeskColor, there was no longer a message to override this pattern, so it shows up. I really don't understand the reasoning Apple chose to save the background pattern in this file. The easiest way to fix this is to delete the Finder.Root file, as the Finder will create it again if it can't find it. By deleting this file, however, you will lose all of your window and icon positions that may be open on the desktop, but you can easily re-create them, of course.

Joe

Rotator
Well, Rotator was just supposed to be an example for folks to use to construct their own desktop programs. The hexagon was thrown in to keep it from being too boring. Jeff Walker and I kicked around the idea of expanding it into a full-fledged 3D shape animator, but Jeff has gone to work in the Macintosh field and I would rather work on more useful stuff. However, I do plan on writing an article focusing on mixing languages under APW, and redoing parts of Rotator in Assembly might make for a good article.

Diz

NoDOS
Oh, OK. I guess I can do a NoDOS update. But Joe will have to do more reviews!

Diz

HOW I SPENT MY SUMMER VACATION

The KansasFest Report
By Josef Wankerl

I've got about another hour of so before I have to leave and find my way to the Kansas City Airport for my flight back to Chattanooga. Everyone else is leaving as well, so now I have a bit of free time to (*finally*!) sit down and think about exactly what all went on here at this year's A2-Central Summer Conference.

I arrived on Wednesday and spent a few hours out on the porch chatting with Tim Swihart (who is in charge of Apple II development tools) and a group of other late-nighters. I didn't really get to sleep that night... it just drifted into the next day, the day of the Apple IIGS college. It was quite a treat! The Apple II Developer Technical Suppport (DTS) people hosted seminars on various parts of programming the Apple IIGS. You could basically ask any question and get answers. Also, you could make suggestions on what you would like changed and enhanced for future versions of the system software. I attended as many sessions as I could. There were really great because each class was small and we had lots of extra time to ask questions. I would have really liked to have a few debugging sessions in the college as well as in the main conference.

I finally got some sleep that night and woke up late enough to catch the end of the system software update. Nothing really exciting happened. Then came the HyperCard IIGS seminars. I really can't say much about these seminars because I signed a non-disclosure agreement, but I will say this: WOW!

The most interesting seminars I attended were the ones on sound, animation, and AppleTalk. Apple's new sound tools, MIDISynth, were introduced by their designer, Mark Cecys. They are called the MIDISynth tools because they integrate your computer's built-in synthesizer with the connectivity power of MIDI. You can use a MIDI (Musical Instrument Digital Interface) keyboard to play your GS! The tool is also a sequencer. You can play a tune on your keyboard and have the sound tools record all the MIDI data to play back later. In order to use these new and powerful tools, a program was also introduced called synthLAB. It allows for the creation of different instruments and sequencing. The demo was breathtaking!

Next came the new animation tools. From what I saw, they were really nice. Included in the package will be the ability to compile shapes. Compiling shapes is the process where a bit map image is transformed into a bunch of store operations so the CPU doesn't have to interpret the map. Compiled shapes are the fastest way to draw. Almost all action games use them. Also, the shape compiler strategically inserts NOPs in the code to compensate for the hardware sync delay from switching from fast memory to slow memory. You don't have to use all the features of the animation tool kit (of which there are a lot!), only the ones you need. If you don't need the shape compiler, don't use it. If you don't need the screen fades, don't use them. The tool is dynamically loaded, so each part you don't use won't be loaded into memory. Also, to speed up the calling process, you can return the addresses of the internal animation routines so you can jump directly to them from your program, thus avoiding all overhead of the tool locator.

I benefited the most from the AppleTalk seminar. I had prior knowledge of sound and animation, so those conferences were just updates in what I knew. But I knew absolutely nothing about AppleTalk. The AppleTalk conference packed quite a wallop on describing the various protocols in English. The books on AppleTalk that I had seen before were cryptic and didn't make much sense. Having a human get up and explain, step-by-step, what was going on made all the difference in the world. In fact, I plan to start on a major AppleTalk project soon, so you might want to look for it in a later issue of *GS+*. (Much later... I've got quite a few projects lined up to do before I even start playing with AppleTalk.)

Many helpful hints and techniques were given out on animation. "Burger" Bill Heineman, Lane Roath, and Chris McKinsey were on hand to answer most of the technical points on animation. It was a real treat to be able to learn how games like Crystal Quest and Battle Chess worked.

Perhaps the most unique experience I had was rapping with Greg Branche, a key person on the IIGS's operating system team, about the future of the IIGSs operating system. We exchanged our views on what should happen and what shouldn't, and I think we both benefited by seeing what the other wanted. Also, I got help from Jim Luther, author of the CD Remote Control and Video Keyboard NDAs. He advised me on a few modifications that I might make to Transfusion in the future.

The human debugger, Dave Lyons, was on hand to figure out why Transfusion crashed when run from the Advanced Disk Utility program. He traced the code into the toolbox routines and suddenly proclaimed that the grafPort wasn't set right. I, along with quite a few other people, was astounded as to how he came to this conclusion. He made a modification from his famed utility that no programmer should be without, Nifty List, and Transfusion worked with out a hitch.

Matt Gullick, Apple's Small Computer Standard Interface (SCSI) engineer (also deemed the "all-around SCSI guy"), was demonstrating the power of SCSI devices. He showed the famed *Star Wars* demo (which is actually part of the asteroid scene from *The Empire Strikes Back*). A sequence of digitized frames were loaded directly off a SCSI hard drive and into screen memory through an Apple DMA SCSI card and shown in real time! Actually, the frames appeared slightly *faster* than they would in the movie!

What surprised me most was the number of developers that had CD-ROM drives.

About half of them did, and said they couldn't live without it. I guess I'll have to look into getting one now, as it seems that Apple is pushing their CD-ROM drive very hard to developers. They are distributing almost everything on CDs, and when I mean everything, I mean that you can get all the development tools on one CD! What a bargain! Another point I noticed was that quite a few people resided on GEnie. There were a few America Onliners, but the GEnie folks had them all outnumbered.

The meat of the conference wasn't just attending various seminars. There were a couple of rooms set up so that companies could show off their new products. I ran into Jim Carson from Vitesse and saw the new Wings program launcher. It looked pretty good. I also managed to pick up a Quickie hand scanner (and a T-shirt that I *love*!) so look for a review of it soon. Mike Westerfield was on hand giving out copies of ORCA/C version 1.1 to registered owners, so I picked mine up. You should expect yours within a few weeks. (See Product Updates on page 45 for more information.)

There was so much more that happened in Kansas City that I would love to tell you about, like the frisbee wars or the "love seat" in the van, but we just don't have room. But before I go, I'd like to give special thanks to some people without whom, Kansas would have been rather normal: John Gibson and Wayne Rudd for helping me write GSBust, Matt Deatherage for pestering me and others to no end (What? I'm thanking this guy for pestering me??), and all the guys who stayed up with me late at night playing with the computers.

Well now that I've told you what did happen in Kansas City, let me tell you what didn't happen. The elusive ROM 04 didn't happen. In fact, I didn't hear a single word about it the entire time I was there. I could have been in the wrong place, though... there was so much going on and I couldn't be everywhere at once. Also, Central Point Software announced that they won't be making a IIGS clone. That was disappointing, but the real bummer was their announcement that they will no longer make any updates to Copy II+, perhaps the best Apple II utility program ever written.

Everybody was really nice during the conference! I don't think I've ever been with a group so large that I could feel totally at ease with. Everyone was (for the most part) cheery and had a positive outlook on the future of the Apple IIGS. I, for one, was glad to see this. Even the representatives from Apple made statements to the effect that the Apple II has not been forgotten, and is about to make a comeback.

I will definitely find a way to attend next year's A2-Central Summer Conference. I wouldn't miss it for anything. It's the best thing I've seen since System Software v5.0.2!

OTHER KANSASFEST COMMENTS

Compiled By Steven Disbrow

As Joe said in his article, he could not be everywhere at once. So, I've been listening especially hard to comments made by other folks that were at the show.

The aforementioned John Gibson tells me that he was keeping count, and that Apple sent 31 people to this show! This was not just a bunch of folks from Apple Developer Technical Support, there were also quite a few people from Apple's management. John says that in all of his conversations with these people, they wanted to cut through the BS and find out exactly what was needed to make the Apple II great again. If that is not a positive sign, I don't know what is. Also, when you consider the salaries Apple pays, sending 31 people is a fairly major commitment for them to make. Especially when you consider that this show had, (according to Joe) less than 200 attendees!

"Burger" Bill Heineman said that he had the same feeling: Apple was there to support the Apple II and let developers know that there was no way they were going to let it die. I've spoken with Bill on several occasions since starting *GS+*, but I've never heard him as excited about anything as he was when he told me about Apple's presentations at KansasFest.

Over and over, I have heard that Apple just does not know how to position the Apple II in general and the IIGS in particular. As near as I can tell, they don't want to ignore the Apple II market, but they don't want the Apple II to cut into Macintosh sales. This is, as they say, a very sticky wicket.

In the next few weeks, Apple will be working on a new marketing strategy for the Apple II. When they come up with one that they feel meets the needs of Apple II users and the goals of Apple Computer, Inc. (a neat trick indeed), we should start seeing a real turn around in the Apple II market. Apple plans on having this new strategy in place within the next two months. If they can do it, it will be just in time for the big Christmas shopping season. And, if they can convince developers and consumers that they are sincere, it may be just in time to save the Apple II.

BEGINNER'S GUIDE TO SYSTEM DISKS

Part II: Start Me Up!
By Josef Wankerl

This installment of the Beginner's Guide To System Disks will discuss how the System Disk starts up and loads everything into place as well as clearing up what most of the files on the disk are used for. Where the first part of this series was basically an overview of the System Disk, this installment is mainly GS/OS information, and not generic System Disk information. While the information presented in this article is not necessary for you to be able to use a System Disk, it is information that will generally be useful to know, even if all you use your computer for is running your word processor or playing games.

Just how the heck does a System Disk start everything? Well I'll tell ya... On the first part of every disk there is something called the *boot block*. When you turn your computer on, it starts searching all your devices to see if it can read in a boot block. If it can, it will read it in and then turn control over to whatever was read. In nearly every case, the boot block contains a short piece of code which loads in a *bootstrap* program. In the case of a IIGS System Disk, this bootstrap program is called *ProDOS*. It is located in the root (i.e. the "topmost") directory of your System Disk. The ProDOS file does a whole bunch of interesting things such as determining what operating system to load (ProDOS 8 or GS/OS). The boring case is ProDOS 8—it loads in the file called P8 from the ***:System:** directory and then executes it. The interesting case is when the ProDOS file decides that GS/OS needs to be loaded. First, a file called Start.GS.OS is loaded and executed. This file contains all the code necessary to load in the rest of GS/OS.

The Start.GS.OS file does an enormous amount of work. It loads in the file GS.OS which contains the Loader and most of GS/OS. It also loads in the ERROR.MSG file. This file contains plain English translations of most GS/OS error codes. Next, all of the File System Translators (FSTs) are loaded, then Device Drivers are either loaded or generated. After that, Initialization (Init) files are executed, then New Desk Accessories (NDAs) and Classic Desk Accessories (CDAs) are loaded. At this point, everything required for GS/OS to operate should be in memory.

The final part of the boot procedure is searching for the startup application. The first thing that is looked for is the ***:System:Start** file. If it is found, it is loaded in and executed. This file is normally the Finder. If Start isn't found, the system then searches for any GS/OS load file (File Type $B3) with a suffix of .SYS16, or a ProDOS 8 system file (File Type $FF) with a suffix of .SYSTEM. If such a file is found, then it is executed. However, if the file P8 is not present, a ProDOS 8 system program will not be executed.

TERMINOLOGY

I have seen many people refer to GS/OS v5.0.2 and System Disk v3.0. This is incorrect. The current version of GS/OS is, as I write this, version 3.0. The current version of the System Disk is 5.0.2. While this may seem like a trivial point to nitpick, you'll really sound like an idiot if you spout these terms to someone who knows better. For instance, you wouldn't call a 3.5-inch drive a 3.25-inch drive, and you wouldn't call a television a radio. A valid statement would be that GS/OS v3.0 is distributed on System Disk v5.0.2.

BUT WHICH ONE DO I HAVE?

So far, I've been talking strictly about System Disk v5.0.2. The way to determine which version System Disk you have is to press a key right after your computer starts to boot. You have to be quick, though! If you manage to hit a key in time, then instead of seeing the familiar red thermometer, you'll get a text screen showing versions of various parts of GS/OS as they are loaded in. If the GS/OS version is 02.00 then you are using System Disk v4.0. If the GS/OS version is 03.00 and everything else is 03.00, then you are using System Disk v5.0. If the GS/OS version is 03.00 and the ProDOS FST is version 03.01, then you are using System Disk v5.0.2. This is the quickest and easiest way to determine which System Disk you have. There are other ways to find out which System Disk you are using, such as examining tool set versions, but the text boot screen seems to be the best way. I agree, though, that Apple should have labeled the version on startup. It would save lots of headaches.

HUH?

I have been throwing out terms like Device Driver, FST, and Init file like you should know what I'm talking about. So now I will define what these things actually are.

A *Device Driver* is the interface between a particular piece of hardware and the operating system. For example, if the operating system wants to write some data to a disk drive, it calls the Device Driver to do the work. Using Device Drivers makes it easier to add new equipment to your system. All you have to do is install the hardware, then install a Device Driver in the ***:System:Drivers:** directory to let the operating system use it. This is a much simpler solution than having the operating system know how to do everything with every product ever made. But, it can also be a pain because you might have the hardware (for instance an internal modem) but no Device Driver to take advantage of it. The operating system will not know how to fully communicate with the hardware without the driver.

A *File System Translator* interprets data formatted for different file systems. Without an FST, the data would just be a meaningless stream of bytes. With the appropriate FST installed in the ***:System:FSTS:** directory, the data is interpreted to be files and subdirectories. There are many file systems available. Normally, there is only one file system for a particular computer. In the past, the file system for the Apple II was DOS 3.3, then it progressed to ProDOS. The file system for an IBM PC is MS-DOS. The file system for a Macintosh is HFS (Hierarchical File System). While most of these file systems are similar in concept, (i.e. they use folders and such to subdivide and organize files), they are incompatible in the way they actually organize the data that they store. GS/OS was designed to allow these different file systems to coexist peacefully. All that is needed to use a different file system is to add a FST for it. It's not simple to create a FST, however. Although Apple wants FSTs to be as modular as Device Drivers, they are not. In order to make a new FST it is necessary to change GS/OS itself! For this reason, Apple has not published the format for FSTs. Another reason is that if there were multiple companies writing FSTs for an file system, there would be no guarantee that they would be compatible. Apple wants to make sure that file systems are interpreted only one way, so they have retained the right to be the sole makers of FSTs.

An interesting thing to note is that when using GS/OS with most devices, you are actually just using the ProDOS 8 file system. There is no unique file system for GS/OS. If Apple comes out with a HFS FST, you could use it just as well as the ProDOS FST. All the programs you used could instantly take advantage of all the HFS features. In fact, if Apple does release a HFS FST, I can reasonably predict that it will be used much more often than the ProDOS FST because it would allow for devices which hold more than 32 megabytes of information, as well as more versatile file names.

An *Init* file is a file that GS/OS loads during the boot process that performs some sort of initialization process. Init files reside in the directory, ***:System:System.Setup:**. There are two types of Init files: *temporary* and *permanent*. A temporary Init is one that GS/OS loads, executes, and then disposes of. A good example of a temporary Init file is Start Screen by Guy T. Rice. This Init file displays a picture while your IIGS starts up. When the boot process is complete, the picture goes away. A permanent Init file is one that GS/OS loads, executes, and leaves in memory. A good example of a permanent Init is Start Sound, also by Guy T. Rice. When it is first executed, it replaces the boring IIGS error sound with a sound of your choosing. Since it is left in memory, it can play your new error sound whenever you would ordinarily get that boring old error sound.

HERE COMES THE TRAIN

One of the greatest features of System Disk 5.0 has been ExpressLoad. This is an addition to the system Loader which speeds up the process of loading files. The Express Loader is not a requirement, however. Your programs will load just as well without it. Only special files that have been preprocessed to have an ExpressLoad header can gain the speed advantage of the Express Loader. A fun way to see if you have the ExpressLoad extension is to hold down the Open-Apple and option keys when the thermometer starts moving on the GS/OS boot screen.

WHAT DOES IT ALL MEAN?

Whew! That certainly is a lot to get a handle on! Taken all at once, it can be quite difficult to see what is happening when you start up your IIGS. However, when you break it down into its individual steps, it can become quite simple. If you had any trouble digesting this article, be sure to let me know so that we can clarify the parts you didn't quite grasp in the next installment of this series.

MOVING?

Well, don't forget to tell us! Simply remove your mailing label from a previous issue of **GS+**, 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

TRANSFUSION: USER'S GUIDE

By Josef Wankerl

NOTE:

This article assumes that you have an *external* modem attached, although this is not necessary. You may, in fact, just be connected straight to another computer or serial device. Transfusion doesn't care. It sees the internal port as a modem device and can communicate to any such modem device in your system. **At this point, Transfusion does *not* work with internal modems.**

WARNING!

These articles contain some information that beginners might choke on. Never fear! Everything is eventually explained. If you are a beginner, just weed your way through all the foreign terms and check out the sidebars. If you still don't know what's going on, then I might be out of a job!

INTRODUCTION

Transfusion is a New Desk Accessory (NDA) which takes advantage of the Apple IIGS's internal modem port to allow simple terminal functions along with XModem file transfers. Transfusion was written with ease of use in mind. If you can use EGOed, you can use Transfusion. In fact, even if you have trouble using EGOed, chances are that you can still use Transfusion to its full potential since there aren't *half* the editing capabilities.

The best way to learn Transfusion is to fire it up and mess with some of the options. The Transfusion program is located on your ***GS+*** disk in the folder **Transfusion1.0**. To install it, use the Finder (or any other copy program since Transfusion **does not** have a resource fork, nor does it directly use resources of any kind) to copy the file Transfusion to the ***:System:Desk.Accs:** folder and then restart your computer.

Transfusion is available from any program that supports New Desk Accessories. Just select the Transfusion menu item from under the Apple Menu. A window will appear while Transfusion's tools load, and then you will be presented with a window containing a menu bar and a standard TextEdit control.

WHAT DO I DO NEXT?

Anything you type, when Transfusion is the front window, is sent to your GS's internal modem port (and hence to any modem that may be attached). If you want to dial up your favorite BBS, such as our own pro-gsplus—(615) 875-4607, just enter in the command to make your modem dial (such as the Hayes standard ATDT1-615-875-4607). Confused yet? If so, just ignore this for now and when you're done reading this article, check out the "Transfusion: Serial Port and Modem Commands" article on page 10.

Transfusion doesn't have a "Dial" menu option (or hang up, for that matter). Why not? Well, because this really isn't supposed to take the place of your favorite communications program. Transfusion assumes you know what you are doing with your modem. Actually, there doesn't have to be a modem connected at all, so dialing wouldn't be a relevant command, although in most cases, a modem will be present. Transfusion allows you the freedom to directly send your modem commands. Although having to remember phone numbers and commands is a royal pain, it greatly simplified the programming and the user interface. If a dial list was present, it would have to be a separate file that the system had to carry around, or it'd have to be part of Transfusion's resource fork. Since Transfusion doesn't use resources (at all!), putting in a resource dial list would have been a nightmare (especially in meeting my deadline for this issue). Also, it would add to the complexity of the user interface, which, right now, is extremely simple and versatile.

Transfusion is meant for *simple* terminal facilities and background file transfers. Because of this fact, Transfusion will be slower than other communication programs you may be familiar with. This is because Transfusion has to share events with its host application (and because it uses GS/OS calls, not direct calls to the firmware, and it uses TextEdit which really wasn't meant for large volumes of information to pour through it at once). Also, because your application must share events with Transfusion, you will notice a marked slowdown in response time from it as well. But this is a fair price to pay for a utility of this magnitude.

OH! WAITER, CAN I SEE THE MENU, PLEASE?

Transfusion has a menu bar with four menus: File, Edit, Send, and Receive. The File menu resembles most applications file menus, although some of the items may not do exactly what they do for other programs. The New menu item doesn't create a new window, it takes the text currently in the Transfusion window and throws it away, leaving you with a new, fresh slate. The Close menu item, as always, closes Transfusion. The Save and Save As options allow you to save the text that is currently in the Transfusion window. The Info menu item displays information about the text in the Transfusion window, the modem device, and the current save file's name. The About menu item displays the usual information for an about item.

The Edit menu contains the standard Copy and Paste items, but not the Undo, Cut, or Clear options. This is because Copy and Paste make sense for a modem device, but the others do not. You can select text in the Transfusion window and Copy it to the clipboard for use later, or you can Paste text already in the clipboard and it will be sent out through the internal modem port. The Select All menu item selects all the text within the Transfusion window. This feature is very handy for changing the font to something more readable or copying everything to the clipboard. If the text you are about to copy to the clipboard is large, it is better to save the text and then load it from another application than to use the clipboard, as the clipboard wasn't meant to hold very large amounts of information. The Choose Font menu item lets you select the font for a particular selection of text or the entire TextEdit control. The default font is plain Courier 10 point. I highly recommend that you use either this font or Monaco 9 since they are of a reasonable size and they are mono spaced. If you want to go for speed, select Shaston 8 since it is optimized for speed with System Software v5.0.2, however, you will give up the consistency of a mono-spaced font.

I THOUGHT YOU SAID THIS THING COULD SEND AND RECEIVE FILES!

So far I've just dwelled on the terminal part of Transfusion. But yes, Transfusion can transfer files in the background! You can be in the Finder doing general file maintenance, or you can be editing icons with your favorite desktop icon editor, and at the same time be downloading the newest game off of a local BBS. This was the main reason Transfusion was written.

The send menu allows you to send either straight text files with no error checking, or any type of file with the ever-popular XModem protocol. A text transfer is rather straightforward. Transfusion just reads the file and send it across to your internal modem port. An XModem transfer is a bit more involved. Transfusion reads the file and sends it across to your internal modem port with some additional information so errors in transmission (due to line noise or whatever) can be detected and corrected. When you select a send option, a standard file dialog appears and you can select the file you wish to send. A send window then pops up and the transfer continues until the entire file has been sent, the transfer is cancelled, or the transfer is suspended.

The receive menu allows you to receive files with XModem-checksum or XModem-CRC. The checksum and CRC are different ways of detecting errors. Why can't you select whether to send in checksum or CRC mode? Well, if you want to get technical, see the "Transfusion: XModem Briefly" article. But in a few words, the receiver tells the sender what kind of error checking to use. The sender only follows what the receiver tells it. For most transfers, XModem-CRC should be selected because it is the most reliable protocol. Also, if the sending program does not support the CRC option, Transfusion will automatically sense this and step down to a checksum receive. When a receive option is picked, a standard file dialog box appears and you must type in the name of the file to receive. Unfortunately, the XModem protocol does not support sending file names with the data. Once you give the file a name, a receive window pops up and the transfer continues until the entire file has been received, the transfer is cancelled, or suspended.

A transfer is cancelled by one of three methods. The first is when the sender decides that the transfer should abort for some reason (like the operator at that end doesn't want the rest of the file to be sent, so he manually cancels it). The second is when the receiver decides that the transfer should abort for some reason. The third, and most nasty reason, is when ten consecutive errors occur during transmission. The transfer window displays the error count in two ways. The first is the total number of errors that have occurred, and the second, displayed in parenthesis, is the number of consecutive errors that have occurred.

WHAT'S ALL THIS SUSPENDED GARBAGE, EH?

A great feature of Transfusion is its ability to "suspend" a file transfer. When you are transferring a file, you can click on the close box of the transfer window and the file transfer will enter a suspended state. Take note that the other computer you are connected to will not know about this and will eventually abort its transfer if enough time elapses without Transfusion restarting the transfer (this will not happen with text sends, however). To restart a transfer, just open Transfusion again by selecting its menu item from under the Apple menu. A dialog box will appear asking you if you want to attempt to resume the transfer. You now have the option of resuming it, or cancelling it. If you select resume, Transfusion will attempt to pick up where it left off. If you select cancel, Transfusion will enter terminal mode.

YOU'RE SUSPENDED!

Suspending a transfer does not really stop the transfer entirely. Your computer has stopped for the moment, but the remote computer is still trying to transfer. When it doesn't get any response from your computer, it generates an error, and after ten consecutive errors, it will time out and quit. You should not suspend a transfer unless it's really necessary because you might not be able to recover from it if you wait too long. If you switch from an application to the Finder and then from the Finder to some other application, you should take a moment in the Finder to reopen Transfusion and let another block get transferred. This will reset the error count on the remote computer so you will have ten more timeouts to work with. Again, and I can't stress it enough, please be careful with suspending transfers.

If the remote computer is still trying to transfer a file when you select cancel, it will not know that Transfusion does not want to continue. The remote computer will continue to try transferring the file until it finally aborts due to timeout errors (the remote computer waits a specified amount of time for you to send something across and if it doesn't get what it wants, it increments its error count). This can take a fairly long time. A nice trick to try is to type <control-X> a few times, as this is the transfer cancel character. It takes two consecutive cancel characters, within one second, to abort the transfer. Sometimes a remote computer does not recognize the <control-X> sequence (the cancel sequence is not really built into the

XModem protocol, it's just a standard enhancement that Transfusion supports). A quick way out of this, if you are receiving, is to type <control-U>, as this is the command that tells the sender that the packet it sent somehow got an error. The sender will then resend the packet, where you can then type <control-U> again until the sender gets fed up with the number of errors and aborts. A quick way out of a situation where the remote computer doesn't recognize the <control-X> sequence and you are sending is to type <control-D> a few times. This is the command that tells the remote computer that you have finished sending the file.

Not only can you suspend transfers within a single application, but if you quit the current application while Transfusion is transferring a file, launch another application that supports New Desk Accessories, and then open Transfusion again, Transfusion will know that it was previously transferring a file and present you with the restart dialog box. This also means that you can switch between two applications and remain on-line whether you are transferring a file, or just in plain terminal mode.

IS THAT ALL?

Basically, that's all there is to Transfusion. To learn more about how to use your modem and the internal modem port, see the "Transfusion: Serial Port and Modem Commands" article below. To learn more about XModem, see the "Transfusion: XModem Briefly" article on page 15. To learn more about the innards of Transfusion, see the "Transfusion: Programmer's Guide" article on page 13.

WHAT'S YOUR PROBLEM?

If you find a problem with Transfusion, I want to hear about it. In order to make things easier on everyone, I have devised a bug report form, which is in Teach Text form, on your *GS+* disk in the file, Problem.Form (use EGOed or another program that can read Teach Text to view it). Print it out, fill it in, and either mail it to us or have it handy if you decide to contact us via voice telephone, or just fill in the blanks and E-Mail it to us here at pro-gsplus or America Online.

There is one "feature" that I already know about but couldn't figure out how to fix in time for this version. It is not serious, just annoying at times. The problem that I couldn't fix is with the TextEdit control. When the Transfusion window gets about 4000 to 6000 characters in it, it becomes aggravatingly slow. The current work around for this is to use the New menu item. It clears out the current text and starts with a fresh, fast control. [Technical conjecture follows. Non-programmers skip to the last sentence. - Ed.] I believe this problem is due to the fact that I'm placing data in the TextEdit control with the TEInsert and TEReplace tool calls instead of letting TaskMaster do it for me, but still, this should not cause a problem. I do this because if I let TaskMaster handle all events then I will never know when to send a character to the modem port because TaskMaster would have already processed it, so I intercept keyboard events myself. If you are a programmer and have suggestions on how this can be fixed, **let me know!** Also, if there are some additional options you would like Transfusion to support, let me know them, as well. I want to make Transfusion a product to be proud of.

TRANSFUSION: SERIAL PORT & MODEM COMMANDS

By Josef Wankerl

To fully control the IIGS internal modem port and a modem, you must understand some basic commands that can be sent to them. Transfusion automatically configures the internal modem port to the settings that it can work with correctly, so, for the most part, you can ignore all the serial port commands. But, if you do run into an occasion where you need maximum control over the serial modem port, the basic commands are presented here for you to make some sense of.

The most popular type of modem available today is the Hayes compatible, so this is the type of modem I will discuss. A Hayes compatible modem has two states. The first state is the *command state* where the modem will recognize commands. The second state is the *online state* where the modem transfers data to and from your computer. When the modem is first powered up it should be in the command state. You can then issue commands to make the modem do a multitude of functions, the least of which are dial and hang up the phone.

The commands that I am about to describe are only for Hayes compatible modems. If your modem is not Hayes compatible, it may not support any or all of the commands below. I highly recommend that you read the user manual that comes with your modem in order to get familiar with all the commands you can use, whether you have a Hayes compatible modem or not. This article goes over just a few of the basic commands, but it does not enter into the more elaborate functions that your modem can perform.

HAYES COMPATIBLE MODEM COMMANDS.

The most basic command for a Hayes compatible modem is the AT command. AT stands for attention. If you type in AT and press return, the modem should respond with "OK". This means that the modem is in the command state and ready for your commands. All other commands, except two, begin with the AT command. One of the first commands you might give your modem is the ATZ command, which resets the modem's internal registers to their default state. Notice that the command began with AT and then was followed by Z, which the modem interprets as a function (AT to get the modem's attention and

n	Baud rate
0	Default*
1	50
2	75
3	110
4	134.5
5	150
6	300
7	600
8	1200
9	1800
10	2400
11	3600
12	4800
13	7200
14	9600
15	19,200

* Set with the Control Panel.

Figure 1
Baud-Rate Selections

then Z to tell the modem what to do now that it is listening).

Another command that will be done fairly frequently is the ATD command. This makes the modem dial a number. For example, ATD1-615-875-4607 will cause your modem to dial the number (615) 875-4607 (the number of pro-gsplus, our bulletin board). A frequent option that accompanies the dial command is T or P, which tells the modem to dial using touch tone codes or pulse (rotary) codes. ATDT1-615-875-4607 will dial pro-gsplus using touch tone codes while ATDP1-615-875-4607 will dial using pulse codes. The hyphens are not necessary and are ignored by the modem. They are supported so that humans can more readily make sense of the phone numbers that we tell our modems to dial.

There are two commands that the modem recognizes without the AT prefix. The first is the A/ command. This tells the modem to repeat the last command that it issued. You do not have to press the return key after you type A/. This feature is handy if you dialed a number and it was busy. If you wish to retry dialing it again, just type in A/ and the modem will automatically dial the number again.

The second command that the modem recognizes without the AT prefix is the +++ command. This is the only command that the modem recognizes while in the online state. It switches you from the online state to the command state. You must wait about a second in-between typing each plus for the modem to recognize this command. The +++ command is almost exclusively used to return to command state so that the phone can be hung up with the ATH command. If you have made a mistake and wish to return to the online state from the command state, the ATO command will do this (Note that this is the letter O, not a zero).

APPLE IIGS INTERNAL SERIAL PORT COMMANDS.

The following commands I'm about to describe are only for the IIGS's internal serial modem port. Although most of them are commands for *any* Apple Super Serial Card or internal modem card, there are a few, very major differences. I recommend that you read the *Apple IIGS Firmware Reference* published by Addison–Wesley to fully understand the meaning of these commands.

Take note that any commands you issue will be lost if Transfusion is closed. So, if you switched to 1200 baud from 2400 baud to send a file and then suspend the transfer for a few seconds, then reopened Transfusion in order to continue the transfer, the baud rate would be back at 2400 and the transfer would eventually abort due to errors. Use caution when issuing **any** of these commands.

All serial port commands start with a <control-A> (hold down the <control> key and press 'A'). This tells the serial card that a command is about to follow (from now on, I will refer to the internal serial modem port as a **serial card,** as that is what it is emulating). Following the <control-A> comes data the serial card interprets as a command.

Transfusion will set the serial card to a state in which the serial card will most effectively communicate with Transfusion and the remote computer. You do not have to issue any of the following commands to make Transfusion work because Transfusion issues the commands automatically. Changing some of the serial card parameters may adversely affect Transfusion, as well as any other communication program.

Perhaps the safest commands to use are those that change the baud rate, data bits, and parity of the serial card. To issue the baud rate change command, prefix the command with the standard <control-A>, then type a number from Figure 1 which corresponds to the baud rate you wish to set, then type the letter B. I will denote this command as <control-A> nB, where <control-A> is the serial card command character, n is a number from Figure 1, and 'B' is the command character. Similarly, to change the data bits, the command to use is <control-A> nD, where n is a number from Figure 2. Lastly, to change the parity, the command to use is <Control-A> nP, where n is a number from Figure 3 on page 12.

The next set of commands are the ones that take an Enable or Disable option. To set echo on, the command would be <control-A> EE, for Echo Enable. This command means that everything you send

n	Data bits	Stop bits
0	8	1
1	7	1
2	6	1
3	5	1
4	8	2
5	7	2
6	6	2
7	5	2

Figure 2: Data-Format Selection

to the modem will be echoed back. Normally a BBS will automatically echo characters back, so this option should normally be left Disabled. To make sure it is disabled, you can use the command <control-A> ED, for Echo Disable. If you are connected to another computer which does not automatically echo characters back to you, you may enable echoing.

The next Enable/Disable command is XON/XOFF. The default state for this is disabled. What XON/XOFF does is, normally, when a buffer is full on one computer, it will send an XOFF to the sending computer telling it to temporarily halt sending characters until the buffer can once again accept characters reliably (by sending XON). This command is <control-A> XE, for XON/XOFF Enable, or <control-A> XD, to Disable. Normally you won't have to fool with this command, but higher speed modems sometimes require that this option be enabled. Also note that this option must be enabled on both computers for it to have any effect.

From here on out, the commands that I am about to describe can very adversely affect the performance of Transfusion, ranging from lines that aren't very pretty, to files that won't transfer correctly. Use these commands with extreme caution, if at all.

If you wanted to automatically add a line feed character after every return character you send out, you can enable this feature with the <control-A> LE command. Normally this command should be left off, unless, of course, you are transacting with a computer that needs this feature on. To make sure this is off, you may issue the <control-A> LD command.

Conversely, if a computer is sending you line feeds after every return character, you can tell the firmware to ignore it with the <control-A> ME command. This should normally **not** be enabled, unless, of course, you are transacting with a computer that is sending line feeds after every return. To make sure this is off, you may issue the <control-A> MD command.

A command specific to the IIGS internal modem port is the Buffer command. The GS maintains a default 4K buffer for text. For Transfusion to work properly, this buffer **must** be on or else Transfusion would lose characters. To make sure the buffer is enabled, issue the <control-A> BE command. If, for some insane reason, you want to disable the buffer, you may issue the <control-A> BD command.

The following commands should **not** be issued to the serial card **at all** when Transfusion is in control. If they were, results would be disastrous. If any of these commands (or any of the previous ones, for that matter) are issued and you wish to return the serial card to a default state, close Transfusion and then reopen it.

To reset the serial card to default values, you may issue the <control-A> R command. This will make the card revert to its built-in set of defaults and may undo some of the commands that have been issued previously.

If you got tired of using <control-A> as a command character for the serial card, there is a mechanism to change it to another, more convenient, control character. You do this by typing the current control character (<control-A>, for example) followed by the new control character that will be the new command (<control-B>, for example). Transfusion assumes that <control-A> is always the command character, so changing it will not allow Transfusion to issue any commands to the serial card. **This is not good.**

The final command that I will describe here is a very dangerous one. It is the <control-A> Z command. The Z command tells the serial card to ignore the command character. This means that neither you, nor Transfusion, can issue any more commands to the serial card. The only way to recover from this is to initialize the serial card, which is done by Transfusion when it opens.

So there you have it. A few basic modem commands, and a generous helping of internal serial modem port commands. Take note that whatever you do, it is not irrevocable. So go ahead and play with them. The most harm you can do is put Transfusion in a state where it needs to be closed and then opened again (or maybe annoy your neighbor with prank phone calls).

IMPORTANT NOTE:
These tables were plagiarized from the *Apple IIGS Firmware Reference*, published by Addison-Wesley, pages 88 and 89.

n	Parity value
0	None (Default)
1	Odd
2	None
3	Even

Figure 3: Parity Selections

TRANSFUSION: PROGRAMMER'S GUIDE

By Josef Wankerl

After many months of pain and elation, I am proud to present the first version of Transfusion, my New Desk Accessory (NDA) terminal and file transfer program. Transfusion was written entirely in ORCA/C using the compiler version 1.1. The source code is commented with blocks before each function to describe what is happening. I find comments inside the code very distracting. There are quite a few points to make about this program, and to tell you everything I have learned from this project would take an awfully long article, so I'll just try to hit the highlights.

While Transfusion is by no means a simple program, most of the routines were really quite easy to program. The hard part was combining them all in a logical fashion and deciding which actions to make a separate function. Perhaps the hardest part (well not the hardest, but definitely the source of a good migraine) of programming is deciding on a suitable name for the project. This one started out as X-fer, then changed to X-fer terminal since it wasn't going to support file transfers in the first version, then back to X-fer when Diz said that it had to. Then it went through a rash of suggestions when it was learned that my ingenious name was already in use. Suggested names ranged from EGOterm to HyperCom to *GS+* Term. Finally, I ended up with Transfusion because I figured that the blood of a computer is information... and this program sends computer blood from computer to computer. But hey, enough about the trivial parts of programming, let's get into the fun part!

Transfusion is broken up into five files: **XFusion.H**, **XWindow.CC**, **XCom.CC**, **XTrans.CC**, and **XFusion.CC**.

These files are not separately compiled. The reason I did it this way was because the longest part of the compile is processing all the #include files, not compiling the actual code. The difference between compiling everything and compiling just one module would be negligible, plus I can make changes to any of the files and the next compile would include those changes—no forgetting what file you changed and having your link and run fail for no apparent reason.

The **XFusion.H** file contains all the global variable definitions as well as window, control, and menu templates. Anything that is not real code is in this file. There are a couple of macros defined here, too. I used enumerated types instead of a multitude of #defines for a few of my variables as it makes debugging a bit easier.

The **XWindow.CC** file contains everything related to drawing the various windows Transfusion presents. The content draw procedures for the main window as well as the thermometer routines for the transfer window are included here. Almost all of the routines in this file are straightforward. Although, there were two nice math problems here.

The first problem was computing the number of blocks a send would take. This was done by a straight integer divide, then a check to see if the number of blocks multiplied by the number of bytes per block was less than the number of bytes to send. This could (and most often does) happen because an integer divide truncates the result.

The second, and more complicated, problem was computing how far the thermometer had to be drawn. I originally had an algorithm I "borrowed" from a graphics book which was supposed to draw a straight line between two points using only integer arithmetic. This routine had to be modified and I didn't do it right. So I just copped out and put in a floating point divide, which worked superbly - smooth drawing the entire way. But for some odd reason, this didn't work on Diz's machine. So we stayed up all night and Diz hacked out an algorithm which seemed to almost work. It was really pitiful, though. [Hey! I was tired... - Ed.] Finally I came across the idea to use fixed point instead of floating point... and thus my routine was born. It's only one line of code, too. I first multiply the numerator of my division by 100 (trial and error number, 10 was too jumpy, 1000 was real smooth, 100 was a good compromise) to move the decimal over, then I perform my division and multiplication, then I divide by 100 to move the decimal back over to where it should be, thus giving the illusion of a fixed point divide with straight integers. I should have thought of this sooner, but sometimes it takes a while for the best method to emerge.

The **XCom.CC** file contains all the routines to access the internal serial port. There are three tricks I used in this file. The first was with the `ReadTimedChar ()` routine. In order to determine how much time had elapsed, I used the `GetTick ()` function, which gets incremented once every 1/60 seconds. By multiplying the number of seconds by 60 and adding this to an initial tick count, I found the time limit for my routine.

The second trick, and one that almost caused me to be put in a looney bin trying to track down what was going on, is in the `WriteChar ()` routine. The task was to output a command character (<control-A>) without having the serial card interpret the next data as a command. Without this code in place, my file transfers died miserably. I checked my send code over and over, but could find nothing wrong. The only way to output a command character is to first change it to a different character and then send the initial command character through. Then I change the command character back. The serial port interprets a <control-A>

with and without the high bit set, so I have to mask it off before I check, and then I have to send out the correct character (with or without high bit).

The third problem I had, and one that also caused a few days of minor head trauma, was deciding how to handle backspaces within the TextEdit control within the `PollModem ()` function. The way I do this is to have a starting and a current cursor position. If a backspace character is encountered and the current cursor position is not zero then the cursor position is decremented. If the cursor position is zero, then the start is incremented. When all the characters have been read in, the start position is checked to see if it is greater than zero. If so, text is **replaced** in the TextEdit control, working the same as a backspace. If not, then the text is inserted.

The **XTrans.CC** file contains all the code necessary to perform file transfers. There are three file transfer routines which are called every run action (actually every seven run actions, see the `NDAAction ()` procedure for more details) and they will perform one packet worth of the transfer. The other routines in this file are support routines for setting up and cancelling a transfer.

One of the last bugs to be extracted from Transfusion was the ability to restart suspended transfers after an application switch. What was happening was the memory that the filename and pathname of the file to be transferred belonged to was getting trashed because its ID belonged to the host application, not the NDA. To prevent this, I added the `MakeNamesPermanent ()` function. This function allocates new memory for these names using a new ID which will not be disposed of when the application's memory is discarded.

I didn't want to kill the TextEdit control when I switched to a file transfer because it would trash everything that was previously in the control. In the `SetupTransfer ()` routine, I moved the TextEdit control out of the transfer window's visible region. This kept the contents of the control while keeping it hidden from the user. When the file transfer is over, the control is moved back to its original position.

The **XFusion.CC** file contains the guts of the NDA. Most everything is straight forward here except for the main control routine, `NDAAction ()`. The first thing `NDAAction ()` does is check the status of the menu bar and updates it if any changes need to be made. Next, the type of action is determined and acted upon. The interesting cases are runAction and cursorAction.

When a runAction occurs, the TextEdit control is updated by calling `PollModem ()`, unless a file transfer is taking place. If a file transfer is in progress, a counter is checked to see if it is okay to send or receive another packet. If it is okay then the appropriate routine is called. The reason for this counter is to allow the host application of the NDA to get some work done. Without this counter, Transfusion would steal most of the work time.

When a cursorAction occurs, the window's content region boundary is determined by subtracting off the menu bar height from the top and the scroll bar width from the right. Then the position of the mouse is obtained. There are now four possible actions to take. The first is if the mouse is outside the content region and the cursor is the I-beam—the action is to turn the cursor back to the system cursor. The second is if the mouse is inside the content region and the cursor is the system cursor—the action is to turn the cursor into the I-beam. The other two cases, the mouse is outside the content area while the cursor is the system cursor or the mouse is inside the content area while the cursor is the I–beam, require no action.

There is a lot more stuff going on behind the scenes than I described here, but it would be pointless to go into such detail. I've outlined most of the tough parts, excluding the actual file transfer code, which you should be able to figure out with the XModem guide on the *GS+* disk. I hope you enjoy Transfusion. I had a blast programming it! As always, send all problems you have or enhancements you'd like to me here at *GS+*.

ERRATA

CORRECTIONS TO OUR REVIEW OF THE CMS SDRM 45 REMOVABLE HARD DRIVE

The day we went to press with *GS+* V1.N5, we finally got an Apple II High Speed SCSI card, and hooked it up to the CMS SDRM 45 MB Removable Drive that was reviewed in that issue. As we were told by CMS tech support, it worked beautifully! The only problem we had was that we had to reformat and re–partition our cartridge before it would recognize all the space on it. A couple of quick backups and restores (using Salvation: Guardian) later, and we were in business.

The retail prices printed in our review of the CMS SDRM 45 Hard Drive are very much in error! The suggested retail prices should be:

CMS SDRM 45 Removable Hard Drive:	$1,400.
Additional 45 Megabyte cartridges:	$139.
CMS SCSI Card:	$129.

(Note: Since CMS drives work with the Apple II High Speed SCSI Card, CMS is no longer bundling a SCSI card with their drives.)

TRANSFUSION: XMODEM BRIEFLY

By Josef Wankerl

XModem is the most widely implemented file transfer protocol in existence. This is because it was one of the first to be developed, and it's very simple. This article will not attempt to explain every detail of XModem, it is simply an introduction, and a guide to how it works. For complete XModem specifications, see the file XMODEM.TXT, in the folder **Transfusion1.0** on the ***GS+*** disk.

XModem exists because there are a hoard of evil, menacing creatures out there that cause data bits to go haywire. XModem makes an attempt to thwart these evildoers. XModem does this by breaking up the transmission into 128 byte packets with some additional control information, as well as some data, to detect errors and to tell which packet is being sent. If a receiver notices that an error has occurred with the transfer of a packet, it asks the sender to send it again.

I will now describe how a standard XModem checksum transfer takes place. All ASCII codes and numbers in this article are in decimal. The first thing a receiving program does when it wants to receive a file is to send out some control information telling the sending program to start sending packets. The receiving program does this by sending a NAK. NAK, for Negative AcKnowledgement, is the equivalent to typing <control-U> on the keyboard (ASCII 21). When the sending program sees a NAK, it starts sending data packets.

The first part of a data packet is a SOH. SOH, for Start Of Header, is the equivalent to typing <control-A> on the keyboard (ASCII 1). Following the SOH comes a packet number and its complement. Then comes the 128 data bytes. Finally, a checksum byte is sent. Packet numbers start at 1 and increment by 1 for every packet. Packet numbers wrap from 255 to 0. A packet number complement is computed by exclusive ORing the packet number with 255, or by subtracting the packet number from 255. The checksum is the sum of all 128 data bytes modulo 256.

The receiver receives all this data and computes its own checksum to see if everything came across correctly. If it didn't, the receiver sends a NAK back to the sender and the sender will then re-transmit the packet. If the receiver got the packet correctly, it will send an ACK to the sender and the sender will then transmit the next packet. ACK, for ACKnowledge, is the equivalent to typing <control-F> on the keyboard (ASCII 6).

When the final packet has been sent by the sender and ACKed by the receiver, the sender sends an EOT. EOT, for End Of Transmission, is the equivalent to typing <control-D> on the keyboard (ASCII 4). When the receiver receives an EOT it knows that it has received all the data and responds by sending an ACK back to the sender. Now the file transfer is complete.

Note that all file sizes are not even multiples of 128. This means that the final packet to be transmitted might not contain 128 valid data bytes. The XModem protocol requires that 128 bytes be transmitted on all packets. So, to compensate, the remaining characters used to pad the final packet should be, by convention, NUL (ASCII 0) characters.

The Cyclic Redundancy Check (CRC) option to XModem is not very different from the checksum protocol. Instead of sending one checksum byte every packet, two bytes representing a 16 bit CRC on the 128 data bytes are sent. The high byte is sent first followed by the low byte. The initial handshaking for CRC is slightly modified as well. Instead of sending an initial NAK, the receiver sends a 'C' (ASCII 67). If the sender does not recognize the CRC option, it will not respond. If it does recognize the C, it will begin the transfer using a CRC instead of a checksum.

So how does the Cyclic Redundancy Check work? It works on the principle that the same formula is applied to the data on both the sending and receiving side of the transfer and if the same result is not obtained, then the transfer must have come across bad. At first, I did not see how a CRC was much different from a checksum. Both methods used a formula, but the CRC was a bit more complicated than the checksum. Take it from me that there are some very involved mathematical proofs using binary polynomial algebra which indeed show that CRC-16 can catch most errors. The Transfusion source code as well as the XModem specifications on the ***GS+*** disk have code which demonstrates how a CRC is generated. It isn't plainly obvious that binary polynomial algebra is taking place, but it is! If you have a bit stream of 10110011, you could interpret it as being the $X^7+X^5+X^4+X+1$ polynomial. Since the only possible states for a variable would be on or off, the theory behind adding and subtracting becomes the same as an exclusive OR. The way to generate a CRC-16 on a sixteen bit number is to first multiply the data by X^{16} (sixteen left shifts) and then divide by a generator polynomial (which should have special properties in order to detect certain errors) and the remainder is the CRC. Since all XModem data is composed of eight bits, a slightly modified algorithm is used, but the concept is still the same. By using a generator polynomial of $X^{16}+X^{15}+X^2+1$, which also equals $(X+1)(X^{15}+X+1)$, it can be proven that CRC-16 can catch a huge amount (over 99%) of all the errors that could occur. Now remember, folks, this is binary algebra we're doing here. A 2X term doesn't make any sense in binary. How would you represent it in a bit stream? Perhaps you could use a 10? You're only allowed one bit per term, so 10 just won't cut it. If you recall, additions are now treated as exclusive ORs, so a 2X would be X+X, which, when exclusive ORed, becomes a 0X. But you won't catch me proving anything any further! All this stuff gives me a headache. I am by no means an expert on this subject, but by programming Transfusion, I feel fairly comfortable with the subject. If you have any questions about the XModem protocol, I'll be more than happy to try to answer them.

RANDOM IIGS PROGRAMMING NOTES

By Steven W. Disbrow

This issue, we have updates to PreFixer (from *GS+* V1.N5) and EGOed (from *GS+* V1.N2). The updated programs (complete with source code) are on your *GS+* disk. (For more information on how to install these programs, see "How To Use The *GS+* Disk" on page 24.) This article describes the changes to these programs and any programming tricks that were used in creating the changes.

PREFIXER UPDATE

PreFixer is, as you may recall, a Control Panel Device (CDev), that allows you to view, edit, set, save and load any or all of the 32 prefixes that GS/OS uses to find files on your disks. For more information on prefixes and how to use them, refer to the original PreFixer article on page 11 of *GS+* V1.N5 (May/June 1990). The new version of PreFixer, v1.01, contains several enhancements to make it more useful and much easier to use.

The biggest change in this version is the addition of a View All selection to the View menu. When you select this item, PreFixer will show you the values of the Boot Prefix and the 32 other GS/OS prefixes. You can then use the standard Text Edit Tool Set cursor control keys to scroll through the list of prefixes. Why isn't there a scroll bar? Well, I tried to put one in there, but the Control Panel NDA seems to offset the thing about 200 pixels to the right, so, in order to get the scroll bar to show up, you have to make the Text Edit box about 50 pixels wide! Not a good solution. It should also be noted that when you select the View All item the Set and Load menus will be disabled. When you select a *single* prefix from the View menu these items will be enabled again.

There have also been a couple of small changes in the way PreFixer works. Both of these pertain to the selection of one of the prefixes 0: through 31: from the View menu. The first change is that when you select one of these prefixes, the value of the Set menu changes to the same thing. The second change is that when you select one of these prefixes, not only is the value of the prefix shown in the edit box, the value is also selected. So, for example, if you pick prefix 8: from the View menu, the value of prefix 8: will be shown and selected in the edit box and the selection shown in the Set menu will be changed to '8:' also. This makes it much more convienient to edit and set the value of the prefix you are viewing.

The last change in this version of PreFixer is in the online help. When you have PreFixer activated and you click on the Help button at the lower left of the Control Panel NDA window, you will be presented with the standard CDev help dialog. The difference is that instead of displaying a tiny amount of help text in this dialog, PreFixer uses a TextEdit control to display much more detailed help information. The only trick here is that the text for the Text Edit control must be stored in a rText resource, not in a rTextForLETextBox2 resource! Why? We aren't quite sure, but rText is the only way that seems to work.

EGOED UPDATE

This EGOed update, v1.3, is a biggie! In fact, it's such a biggie that EGOed once again tips the scales at a whopping 25K+! The next EGOed update will address this problem, but, for now, let's look at what you get in that extra 5K.

The number one change that folks have requested in EGOed is the ability to save font information. My position was that I would not include such a capability until a standard file format became avialable. Well, a few months ago, Apple introduced just such a file format. It's called the Teach format and EGOed v1.3 supports it fully. The Teach format is so called because it is based on a similar format used on the Macintosh, called Teach Text. For simplicity sake, and because 'Teach Format' sounds kinda stupid, I will refer to the format as "Teach Text." Files that are saved as Teach Text contain not only font and style information, they also contain information that tells the size and shape of the edit window. When you open a Teach Text file, the edit window will be changed to the same size, shape and position that it was in when the file was last saved. This may sound useless at first (it did to me), but after you play with it for a while, you should find quite a few uses for it.

The actual Teach Text format is discussed in detail in Apple's IIGS File Type Note FTN.50.5445. Briefly, here is how it is set up: A Teach Text file is a GS/OS extended file with a file type of $50 and an auxillary file type of $5445 (the ASCII codes for 'TE'). In other words, it has both a data fork and a resource fork. The data fork contains plain ASCII text. The resource fork contains two resources. The first is an rStyleBlock resource that contains all of the font and style information that should be applied to the text contained in the data fork. This rStyleBlock information can be gotten directly from the `TEGetText()` Toolbox call and can be passed directly to the `TESetText()` or `TEInsert()` Toolbox calls. The second resource is a resource that contains window position information. Its format is shown in Figure 1 on page 17. Since Apple did not bother to give it a name in its Tech Note, I have called it rWindowPos. Both of these resources must have a resource ID of 1. From a programming point of view, there was very little to do to add Teach Text support to EGOed. The only hard part was making use of the window position information. To see how simple it is, look at the EGOed source code on your *GS+* disk.

At this point, EGOed is the only editor that we know of that fully supports the Teach Text format. *WordWorks* (from SoftDisk G-S) and *WriteIt!* (by C. K. Haun), both save font and style information correctly, but neither of these programs saves the window position information. EGOed will read files created by these programs, and they can read files created with EGOed. However, these programs will ignore any window position information that is in a file created with EGOed.

CHANGES IN THE MENU BAR

So, that's the big change. To support this big change, a bunch of smaller changes were needed, and that's what we will look at now. The easiest way to go over these changes is to simply start at the File menu and go over each new or changed item. So, let's do just that.

The only new item in the File menu is the *Save As Teach...* item. When you select this item, a standard Save File dialog appears and asks you where you want to save the file. When you save a file with this selection, the font, style and window position information is saved in the file's resource fork.

The only other changes in the File menu are that the Command Key for the Info... item has been changed to Open-Apple-? and there is no longer a Command Key for the About EGOed... item.

Moving to the Edit menu, you will find three new selections. The first is *Select All*. When you select this item or press Open-Apple-A, all text in the EGOed window will be selected. Very handy.

The last two items, *Clear Hi-Bits* and *Set Hi-Bits*, are intended for those of us that have to deal with silly programs like Merlin and DBMaster that set the Hi-Bit of characters in the text files that they create. If you don't know what this Hi-Bit nonsense means, you can usually tell a file that has its Hi-Bits set by the fact that it appears to be chock full of garbage. If you have such a file, you can select the offending characters with the mouse (or you can use Select All to select the whole file) and clear the Hi-Bits with the Clear Hi-Bits item. If you want, you can even set the Hi-Bits with the Set Hi-Bits item. Be careful though! While these Hi-Bits are, for the most part, a nuisance, they are also used to generate some fairly neat characters like: ©†®¥™. In other words, some of that "garbage" may be intentional!

The last change in the Edit menu is that it is now very smart. If there is no text, you can't use Select All. If there is no text in the clipboard, you can not select Paste. If there is no text selected, you can't use Cut, Copy, Clear, Clear Hi-Bits or Set Hi–Bits.

The next menu is a new one: the *Font* menu. This is where the Choose Font... item lives now. Underneath the Choose Font... item is a list of the different fonts that you have in your system. When you pull down the Font menu, there will be a check mark beside the font that is applied to the currently selected text.

The *Style* menu is another new menu. It shows you, and allows you to select, the different styles and sizes that can be applied to the currently selected block of text. When you pull the Style menu down, there will be check marks next to the styles and size that are applied to the currently selected text. Below the list of font styles, several of the more common font sizes are shown. However, you may not have all of these sizes in your system. The sizes that you have installed will be displayed in boldface.

The last new menu in EGOed v1.3 is the *Preferences* menu. The settings you choose in this menu are saved in the EGOed resource fork, so you do not have to reset them each time you use EGOed.

The first selection in this menu is *Default Font...*. When you select this item, the standard font chooser dialog is displayed. From this, you can choose the font, size and styles that you want applied to every plain text or AppleWorks file that you open. The next time you open a plain text or AppleWorks file, it will be displayed in that font, style and size. Teach Text files are not affected by this option.

The second selection is *AWP Warnings*. You use this selection to tell EGOed whether or not you want that annoying warning displayed whenever you save a file in AppleWorks format. The default is to display the warning.

OPERATIONAL CHANGES

The last two changes in this version of EGOed are operational. The first change that you will notice is that EGOed now tells you what type of file you are editing (Ascii, Teach Text, AppleWorks or APW Source) in the title bar of the EGOed window.

The last change is in the way the Open... item works. Previously, if you selected Open... and the file you were working on had not been changed, it was removed from the EGOed window immediately. Even if you cancelled the open operation, the file you were editing previously was gone. Now, when you select Open..., you are asked to select a new file to open **before** the old file is cleared. If you cancel the open operation, you are returned to the file you were editing. If you pick a new file to open and the old file was changed, you are asked if you want to save the changes. If you click cancel, you are again returned to the file you were editing.

SO WHAT'S NEXT?

In coming issues of ***GS+***, you can look for a complete rewrite of EGOed (using ORCA/C), support for the @: prefix in PreFixer, and the addition of a couple of new file transfer protocols to Transfusion. As always, be sure to let us know what you think of what we've done so far, and what you want to see in the future!

Figure 1
REZ Definition For A Window Position Resource

```
#define rWindowPos 0x7001
/* Default EGOed Window Position: 640 mode. */
data rWindowPos (0x2) {
 $"8A00" /* Word - Window height in pixels.         */
 $"7402" /* Word - Window width in pixels.          */
 $"2600" /* Word - Y coordinate of window top.      */
 $"0600" /* Word - X coordinate of windows left edge.*/
 $"0000 0000" /* Long - Version Number. Must be 0.  */
};
```

THE MOLEHILL

By Joe Wankerl

During the course of writing the Beginner's Guide to System Disks, I explored previous versions of the ever–popular System Disk to see exactly how much better off we are now than years ago. And, to my surprise, there has been an unbelievable amount of improvement over the first System Disk that I have. The first System Disk started booting with that insidious ProDOS 16 and took forever, only to end up in the Launcher program, which was just a standard file dialog.

After much playing, I ran across a program I hadn't fiddled with in years, called The Apple II DeskTop. [It should be noted that the 8-bit DeskTop is no longer supplied on IIGS System Disks and we are not sure if Apple ever released it as public domain or shareware. In other words, if you don't have one of the older System Disks, you will need to find another 8-bit program selector. - Ed.] Although similar to the Finder in most respects, this program runs under ProDOS 8 and uses double high resolution graphics. Because of this, it cannot support real NDAs, but it has a custom NDA program format just for the DeskTop. I can recall sitting for hours playing the puzzle game... that was before I got any real application for my GS. Those were the days, eh?

So now I bet you're wondering why I'm telling you all this, right? Well there is a reason, I assure you. I have two Apple 5.25 drives connected to my system. I almost never use them, so I have the GS/OS driver for them inactive. When I do want to use them, it is almost exclusively for ProDOS 8 applications. If I want to do DOS 3.3, I'll just boot from the 5.25-inch drive and not worry about interacting with my hard drive at all. I hate enabling the GS/OS driver and rebooting just to use my 5.25-inch drives with a consistent Finder-like user interface... so now, as you might have guessed, I use the DeskTop. (I'll refer to The Apple II DeskTop program selector as the DeskTop from now on).

The DeskTop is a ProDOS 8 program launcher. I was originally going to shell out lots of money for a shareware or commercial program launcher for ProDOS 8 until I found this beauty that Apple provided to me along with my GS. Most ProDOS 8 program launchers are text-based since they have to work with the II+, IIe and the rest of the II line. The DeskTop runs only on machines that support double high resolution graphics, and can therefore take advantage of some features that less powerful IIs lack.

The DeskTop is fast! Not quite as fast as the Finder (although it is faster than the first versions of the Finder... play with old System Disks and see for yourself!), but fast enough to not be a nuisance. The user interface is not exactly what it is in most GS applications, but this is because it is written without using QuickDraw II or the Window Manager—it's all custom routines.

Another bonus is the Selector, a part of the DeskTop system. The Selector is run before the DeskTop takes control and allows you to launch applications from a menu instead of opening seven or eight windows to get to the application you want to use. Since the DeskTop does not support dragging icons onto the desktop like the Finder, this was a reasonable alternative. I have ShrinkIt, BASIC.System, BinSCII, KERMIT, and other ProDOS 8 programs in my Selector menu since I like to quickly switch between them without returning to GS/OS (which requires an operating system swap - faster than ever before, but yet still slow).

If you have a need for a ProDOS 8 program launcher, or you just want the convenience of having your 5.25 drives accessed with a desktop program without enabling a driver and rebooting, I highly recommend digging up The Apple II DeskTop.

ADVERTISERS' INDEX

Advertiser	Page	Phone
Price Busters	23	(619) 589-0081
Silicon & Software	20	(714) 250-8101

We here at *GS+* have made every attempt to make sure that you don't get ripped off. The mail-order firms advertising in this issue of *GS+* have been checked out through the Better Business Bureau, and the S&S RamCard advertised by Silicon & Software received a favorable review in *GS+* V1.N5.

However, if you have ***ANY*** problems with any of these advertisers; ***LET US KNOW***. We need your help in making sure that we have the most dependable advertisers around. And remember, when you place an order with one of these advertisers, ***BE SURE*** to tell them that you saw it in *GS+*!

GS+ BACK ISSUE INFORMATION

September/October 1989 (Vol. 1, Number 1)
$4 magazine only
$5 disk only
$6 magazine + disk

- System Software 5.0 Compatibility Chart
- NoDOS - A file utility New Desk Accessory complete with ORCA/C source code on disk
- Graphics Galore - Drawing "how-to" with 3 pictures on disk
- Reviews of Arkanoid II (new custom levels on disk), Crystal Quest, ORCA/C, Rocket Ranger, Silpheed, Test Drive II, TransWarp GS, Turbo Mouse ADB

PLUS: Icons, Rumors, and more!

November/December 1989 (Vol. 1, Number 2)
$4 magazine only
$6 disk only
$9 magazine + disk

- Beginner's Guide to the Finder - Part 1: The Basics
- EGOed - An NDA text editor (TML Pascal II source code on disk)
- Brush with Greatness - Tips on drawing faces.
- Reviews of TML Pascal II, Source Code Library II, Cutting Edge Keyboard, Battle Chess, Dark Castle, Dungeon Master, Neuromancer, Laser Force

Plus lots more!

January/Febrary 1990 (Vol. 1, Number 3)
$4 magazine only
$6 disk only
$9 magazine + disk

- Beginner's Guide to the Finder - Part 2: Mousing Around
- Rotator - A beginner's desktop programming tutorial and program w/source code written in ORCA/C
- Winning Arkanoid II Levels
- Brush with Greatness - Space graphics
- Reviews of HyperStudio v2.0, Graphic Disk Labeler, Programmer's Online Companion, Xenocide, Keef the Thief, Life & Death, The Three Stooges

Plus much more!

March/April 1990 (Vol. 1, Number 4)
$4 magazine only (SOLD OUT!)
$6 disk only
$9 magazine + disk (SOLD OUT!)

- Beginner's Guide to the Finder - Part 3: All About Icons
- All About Control Panel Devices - with Desk Color CDev and ORCA/C source code on disk
- Brush with Greatness - Architecture on your IIGS with pictures of the CitiCorp building and Frank Lloyd Wright's house on disk
- Reviews of InnerDrive, Vulcan, Salvation - Guardian, ORCA/Dissembler, Computer Eyes (pictures on disk), Jam Session (songs on disk), Tunnels of Armageddon, Where in the World is Carmen Sandiego

May/June 1990 (Vol. 1, Number 5)
$4 magazine only — $6 disk only — $9 magazine + disk

- AppleFest Report
- Beginner's Guide to System Disks - Part 1
- All about GS/OS prefixes - with PreFixer CDev and ORCA/Pascal source code on disk.
- Brush with Greatness - How your IIGS makes colors
- Reviews of CMS 45 Megabyte Removable Drive, S&S-Ram Card, DataLink Express, Visionary GS Digitizer, GraphicWriter III, ZapLink, McGee, MathBlaster Plus IIGS

TIPS ON USING GRAPHICWRITER III

By Noreen Ribaric

If you have a Apple II Video Overlay Card installed in your IIGS, the VideoMix NDA *must* be active in order to do anything in GraphicWriter III. If it isn't, GWIII will crash horribly or lock-up (at first we thought we had a corrupted file and were going to shoot ourselves)! We have not experienced this problem with any other program that we have used.

GraphicWriter III does *not* use the standard system clipboard. So, if you have an NDA open that supports the standard system clipboard (EGOed, for example) and copy something into the clipboard using the NDA's menu, you cannot paste it into a GWIII file, because as far as GWIII is concerned, there is nothing in *its* "clipboard." You *can* do this with AppleWorks GS, however, because it *does* support the standard system clipboard.

It is possible to import Teach Text files into GWIII if you don't care to lose all the formatting (font and style information). If you select the Universal translator and deselect the "Choose from TEXT files only" import option, you can select a Teach Text file to import. GWIII will just ignore the Teach Text file's resource fork (see "Random IIGS Programming Notes" on page 16 for more information on Teach Text file formats), and import it just like a plain ASCII file.

RUMORS, WISHES & BLATANT LIES

(Guaranteed To Give Our Lawyers Ulcers)

OUR LAWYERS MADE US SAY THIS...

Just about every single word in this magazine is somebody's trademark. You can't swing a dead cat these days without getting sued (or yelled at by animal rights activists), so we just wanted to let you know that we don't make any claim to these names, we are just using them to get your attention. Please don't sue us. However, we *do* own the rights to the names "EGOed" and "Transfusion", and we will sue the pants off of any big-headed editor or hospital that we catch using them. Special site-licensing is available for hospitals and anybody that publishes anything that the Bush administration might object to.

HYPERCARD GS!

Well, since we he had to sign a Non-Disclosure agreement to get to look at HyperCard GS, we can't really tell you very much about it. What we can tell you though is that most of the stuff we told you in the last few issues is true. It is a real product, and it will be compatible with HyperCard for the Macintosh. This one fact (and the fact that Apple sent 31 people to KansasFest) implies a whole host of wonderful things for the IIGS! (None of which we can tell you about.) However, it will not be compatible with HyperStudio. (Awww! Bummer!) Also, you probably won't have to pay a licensing fee to distribute HyperCard GS stacks. Why not? Well, since Apple will probably give it away with every new IIGS sold and will probably sell it to those of us that already have IIGSs (HyperCard GS will *not* be Public Domain you know!) for a measly $49, most everyone will have a copy. So, there won't be any need for a "Run-Time Module" like there is with HyperStudio. Of course, the final version of HyperCard GS may be a complete dog that no one will want—but we don't think so!

THE NEXT MACINTOSH?

For quite some time now, there have been rumors of a new low-end color Macintosh capable of using Apple II programs. There will be such a machine, but, surprise! It's not a Macintosh at all! Apple, due to legal problems with the use of certain key chips in the Macintosh, has decided to put its eggs in two baskets and will be introducing an upgraded Apple IIGS that will be fully Macintosh compatible. This new Apple II will operate at approximately 13MHz and is intended to compete in the low end market with IBM PC Clones and the new PS/1.

LOW COST, LOW SPEED

For several months now, there have been rumors that "a certain company" was coming out with a IIGS printer driver for Hewlett-Packard's 300 dots-per-inch DeskJet Printer. Well, it's true! When this driver is released, you will be able to get LaserWriter-quality graphics output for about $600 total cost ($500 for the printer + $100 for the driver). What we've seen so far looks great. The biggest problem is that the only way you can use the driver is with the special parallel printer card that will come with it. This drives the cost of the product up and is one of the more heinous forms of copy-protection. However, without the parallel card, the DeskJet is so slow, you probably would not want to use it anyway.

HE'S GOT THE LOOK

Last issue, we inadvertently forgot to give you the name of person whose backside graced our cover. It was a gross oversight, and we promise that it won't happen again. Um, unfortunately, that same person wrote several reviews in this issue under a number of false names and he insists that we not use his real name. So, er, forget we mentioned it. [Are you happy now, Dave? - Ed.]

THE COMPLETE STORY

Rumor has it that TML Systems has pretty much gone under. Not to worry though, the same rumor says that Vince Cooper, TML Systems former head of IIGS products, is taking TML Pascal II and TML Basic and renaming them, "Complete Pascal" and "Complete Basic." After all the dust settles, Mr. Cooper plans to "Completely" rewrite both packages to include, among other things: OMF 2 output, an "open architecture" desktop environment similar to the ORCA/Desktop, a new editor (hooray!) and bunches of other stuff. Sounds good to us, as we are "Completely" fed up with TML Systems' lack of support for their IIGS products. If you want more information, this rumor comes complete with a company name and a phone number: Complete Technology, (904) 731-7181.

NO, REALLY!

Did you know that the special graphic effects on Star Trek®: The Next Generation™ are done on a IIGS? Seriously! They use a pre-release version of the fabled Turbo-Rez graphics board, and StickyBear-Render software. It's true. Honest! Wait! Where are you going?

YOU CAN DO BETTER?

Got a rumor? Got a wish? Got a blatant lie? Got a nasty rash? See a doctor for the rash, and send the others to:

GS+ Libel Department
P. O. Box 15366
Chattanooga, TN 37514-0366

THE *GS+* USERS' GROUP CONNECTION

We want to compile a list of IIGS Users' Groups and/or IIGS Special Interest Groups (SIGs) that are a part of regular Apple II Users' Groups. If you are a member of such a group, have your president contact us. All we need for the list is the name and address of the group. However, if you give us a free subscription to your groups newsletter, we'll give your group a free magazine-only subscription to *GS+*! Send that information and/or newsletter subscription to:

GS+ Users' Group Connection
P. O. Box 15366
Chattanooga, TN 37415-0366

GS+ CLASSIFIEDS

For Sale

Destroyer-GS (Epyx) $14.95=, Apple 5.25 Drive (buff, daisychainable) $149.95=, CP/M+ System IIGS (Cirtech) $89.95*, Defender of the Crown (Cinemaware) $24.95, Deluxe Art Parts 1/2/Seasons & Holidays (Electronic Arts) Each $14.95, Monte Carlo (PBI) $19.95, Gauntlet (Mindscape) $19.95, GraphicWriter III (Seven Hills) $79.95, Geometry (Broderbund) $59.95, MacroMate (Wagner) $29.95, Merlin 8/16 (Wagner) $59.95, SoftSwitch (Wagner) $29.95=, TML Pascal $79.95, VIP Professional $99.95*, Test Drive II: The Duel w/all 3 library disks $59.95*.

All items are brand new, except those marked with *, and items marked with = denote the fact that we have small quantities of these available. We can accept VISA for all US-$ purchases, or take MasterCard with total converted to £-sterling. No checks accepted. We need $9.95 added for airmail postage on any order I'm afraid.

Ask for full list, (we've got thousands of non-GS Apple II products to clear, and we've got lots of Apple II products simply not available in the USA...).

Contact: MGA SoftCat
41 Cinque Ports Street
Rye, E. Sussex TN31 7AD
England
TEL: 44-797 226601
FAX: 44-797 226721

For Sale

File Manager is a shareware NDA that lets you delete, move, and rename files/folders, copy any GS/OS files, change file/auxiliary type and access attributes of files/folders, search a disk/folder for a (partial) file name, and view files in a movable window. All utilizing an intuitive-icon-based interface. Available on the major information services or from the author for $15 ($20 foreign). Includes disk and manual.

Contact: Jeff Hartkopf
533 Wildrose Court
Louisville, CO 80027

1/2 Price Software

Ancient Land of Y's - $22.50
The Hunt For Red October (novel not included) - $20
Silent Service - $20
Sub Battle - $5
The Three Stooges - $25
Where in the U.S.A. is Carmen Sandiego - $22.50

Contact: *GS+* Magazine
c/o EGO Systems
P.O. Box 15366
Chattanooga, TN 37415-0366
(615) 870-4960

Readers can place an ad in the *GS+* Classifieds for only $5. This cost buys 25 words in one issue of *GS+*. Additional words are just 10 cents each. The *GS+* Classifieds are the perfect way to contact all of the other IIGS owners out there. The deadline for inclusion of a classified ad in the next issue (Volume 1, Number 6) of *GS+* is June 25, 1990. Simply fill out a photocopy of the coupon below; or send your ad along with your name, address, phone number, number of issues to run, and payment (**made payable to EGO Systems**) to us here at *GS+*; or call us at (615) 870–4960 to place an ad with your MasterCard or VISA.

GS+ CLASSIFIED AD ORDER FORM

Ad copy: ____
____
____
____
Number of issues to run: ____ Number of words: ____ Total enclosed: $____
Name: ____________________________ Phone: ____________
Address: ____
City: ________________________ State: ________ Zip: __________

MOUSE POSITION

By Jeff Hartkopf

Mouse Position is a new desk accessory for the Apple IIGS written in ORCA/C that gives the position of the mouse in both global and local coordinates. It's useful in programming, graphics, and lots of other applications.

INSTALLATION

To install Mouse Position, copy the file **Mouse.Position** from the folder, **MousePosition** on your *GS+* disk, into the folder ***:System:Desk.Accs:**, where "***:**" is the name of your startup disk. After restarting the computer, select Mouse Position from the Apple menu of any desktop application.

USING MOUSE POSITION

After you select Mouse Position from the Apple menu of a desktop application, a window will appear on the screen. It contains the local and global coordinates of the mouse in pixels, and is updated as often as possible. Horizontal and vertical components of the position are designated by horizontal and vertical arrows. Global coordinates give the position of the mouse relative to the screen, i.e., (0, 0) is the upper-left corner of the screen, while (639, 199) is the lower-left corner in 640-mode. Local coordinates give the position of the mouse relative to the top window on the desktop, i.e. (0, 0) is the upper-left corner of the content region of the top window. To bring a window to the top, click on it.

When you are finished using Mouse Position, click the close box in the title bar or select Close from the File menu of the application.

THE LOWEST PRICES ON APPLE IIGS UPGRADES!

SAVE MONEY ON EVERY PURCHASE!

UNIVERSITY & SCHOOL P.O.'s WELCOME!

IIGS RAM BOARDS 5 YR. WTY.

4 MEG GS RAM BOARD. 100% Replacement for Apple's GS RAM board. Fully DMA compatible. Fully compatible with WOZ Edition, ROM 01, and ROM 3 version IIgs computers. User upgradable to 4 Full Megs (uses standard 1MEGx1 RAM), also available from PBC.

w/0MEG	Order# AGS-AI0MG-0	$65.	w/1MEG	Order# AGS-AI1MG-1	$129.
w/2MEG	Order# AGS-AI2MG-2	$194.	w/4MEG	Order# AGS-AI4MG-4	$349.

6 MEG GS RAM BOARD. GS RAM PLUS by Applied Engineering. 100% replacement for Apple's GS RAM board, fully DMA compatible, and User upgradable (uses standard 1 MEGx1 RAM) to 6 Full Megabytes.

w/1MEG	Order# AAE-GRP1M-1	$219.	w/2MEG	Order# AAE-GRP2M-2	$294.
w/4MEG	Order# AAE-GRP4M-4	$444.	w/6MEG	Order# AAE-GRP6M-6	$594.

IIGS RAM UPGRADES

256K RAM SETS (8PC). If you already own the Apple's 1 Meg GS RAM board, Super Expander GS 1 Meg, A.E. GS RAM 1.5 Meg, or A.I. GS Juice 1 Meg RAM board, now you can add 256K sets at PBC's factory direct low price.

Our 256K RAM sets are NEW, 120ns, CAS/RAS, and come with a Five Year Replacement Warranty!

Order# APX-256KR-1 $29.

1 MEG RAM SETS (8PC) Used for GS RAM expansion on above 4 MEG RAM board, above A.E. GS RAM PLUS, Apricorn RAMPRO, and A.I. GS Juice - RAM boards.

Our 1 MEG RAM sets are NEW, CAS/RAS, and come with a Five Year Replacement Warranty!

Order# APX-1MGKR-2 $75.

256Kx4 RAM SETS (8PC) Used for GS RAM expansion on A.E. GS RAM Ultra and Orange Micro GS RAM boards. Our 8PC sets add a full 1 Meg to your board.

The 256Kx4 RAM sets are NEW, CAS/RAS, and come with a Five Year Replacement Warranty!

Order# APX-256X4-3 $99.

IIGS ADD ON DRIVES

5.25" DRIVE Used as the last link in your daisy chain line.

The built in direct drive mechanism offers quiet operation, with reliable data transfer your can trust. How reliable are our drives? So reliable, we offer a full TWO YEAR WARRANTY, including parts and labor!

The drive comes complete with cable, to plug directly to the rear of your Apple IIgs or Apple 3.5" drive.

5.25" Unidisk Compatible Drive	Order# APX-DDHHC-1	$119.

5.25" DAISY CHAINABLE DRIVE Fully daisychainable, with the same high performance on our last link drive. Comes complete with cable to plug into the rear of your IIgs or existing daisychain drive.

The rear of the drive has it's own connector, which matches Apple daisychain specifications 100%.

5.25" Daisy Chain Drive	Order# APX-MDXX-2	$139.

3.5" AMR DISK DRIVE Add a second 3.5" disk drive to your IIgs as about half the price of Apple's. The AMR 3.5" drive is fully daisychainable, completely IIgs compatible, comes complete with cable, and has it's own rear port. AMR offers a full 1Year warranty on this unit.

Order# APX-DD35C-3 $194.

3.5" AE DISK DRIVE Fully daisychainable and 100% replacement for Apple's 3.5" drive.

Order# AAE-DD35C-4 $199.

GS COOLING FAN

GS SUPER COOLER A direct replacement for Apple's GS cooling fan, at less than half the cost, plus 8 times the warranty.

This internal Super Cooler draws less than .18 amps for operation, causes NO audio line interference, and operates so quietly, you forget it's there!

Order# AGS-GSFAN-1 $24.

GS CLOCK BATTERY

GS CLOCK BATTERY We all thought that old clock battery would last for five years or so. More & more have been failing in two or three years. Your local dealer will charge for battery, plus installation. Some IIgs owners have been charged as much as $200, just for a battery change.

PBC offers a quick & easy replacement Gs clock battery that anyone can install, no soldering required. Complete installation instructions are included, and the battery has a two year warranty!

Order# AGS-CKBAT-2 $15.

IIGS ACCELERATORS

TRANSWARP GS by Applied Engineering. An experience you should not miss, this GS accelerator will increase that 2.6 Mhz speed to almost 7 mhz. That's over twice as fast as the normal Gs speed! Built in ROM allows speed changes of normal, fast, & Transwarp.

Order# AAE-TWPGS-3 $285.

GS ZIP BOARD by ZIP Technology. Avail 11/90. The GS Zip Board is the latest accelerator form the makers of the ZIPCHIP. Latest reports track the speed at between 6 & 8 Mhz.

Order# AZP-GSZIP-3 $248.

IIGS HARD DRIVES 1 YR. WTY.

VULCAN By AE. A superfast internal hard drive and Power Supply w/Fan, that replaces the IIGS P/S. Already formatted & setup with GS/OS. The vulcan controller bd can be used in any rear GS expansion slot, and uses a 16 Bit controller. Complete software & operation manual included.

20 MEG VULCAN	Order# AAE-HD20G-3	$495.
40 MEG VULCAN	Order# AAE-HD40G-5	$645.

OTHER IIGS ENHANCEMENTS 2 YR. WTY.

JOYSTICK III The ultimate in quality joysticks, designed for long lasting performance with your IIGS. Similar to the Mach III (by C&H), with a large fire button on the stick, the JOYSTICK III plugs to the rear of your IIGS.

Order# APX-JYSK3-2 $24.

GAMEPORT SWITCHBOX w/Master Cable Allows easy switching between any two gameport peripherals, plus saves wear and tear on your peripheral connectors.

Order# APX-ABDB9-2 $29.

SERIAL/PARRALLEL CONVERTER w/5 YR WTY

APRICORD 8 by Apricorn. Allows interface of your Apple IIGS computer to any standard centronics parallel printer. Includes all cables for connection.

Two models of the APRICORD 8 are available. One model is for stock Epson model printers, and the second model is for non-Epson printers.

For Epson Printers	Order# APC-SPCVE-3	$59.
For Non-Epson Printers	Order # APC-SPCVN-4	$59.

COPY II PLUS 9.0 The Ultimate Package for your Apple IIGS. Full Support provided for 5.25" and 3.5" diskettes. It evens copies files to Hard Drives.

Copy complete diskettes, files, and copy protected software. Undelete files, verifys diskettes, files, and disk speed for errors. Built in sector editor! Complete operators manual included.

With Copy II Plus 9.0, you won't need any other copy utility!

Order# APS-C2P53-5 $25.

IIGS IMAGEWRITER CABLES

IIgs to Imagewriter I Cable	Order# APC-GSIM1-1	$11.
IIGS to Imagewriter II Cable	Order# APC-IMT03-3	$11.

IIGS INTERFACE BOARDS 5 YR. WTY.

SUPER SERIAL BOARD 100% replacement for Apple Super Serial Card, for those that do not want to use the internal GS serial ports.

	Order# APX-SSEXP-2	$48.
10' Cable SSB to Imagewriter I	Order# APC-SECMM-1	$12.
10' Cable SSB to Imagewriter II	Order# APC-IMT02-2	$12.
10' Cable SSB to External Modem	Order# APC-SECMM-1	$12.

GRAPHIC PARALLEL BOARD w/6ft Cable Compatible with Epson, NEC, Okidata, Panasonic, and all other centronics type graphic printers. On board ROM control allows versatility of graphic commands which include normal & inverse print, enhanced print, double size print, & 90 degree rotated print. In the text mode, this board allows full use of your printer modes, plus text screen print from any line.

Order# APX-GPEXP-2 $44

VISA & MASTERCARD ACCEPTED (NO EXTRA FEE) COD OKAY

PRICE BUSTERS (619) 589-0081 10am-6pm/Mon-Fri

4233 Spring St., Suite 402, La Mesa, CA 92041
PACIFIC TIME

HOW TO USE THE GS+ DISK

The first thing you need to do is **MAKE A BACKUP COPY OF YOUR *GS+* DISK WITH THE FINDER!!!** Next, put the original in a safe place. If your have a problem making a backup copy, give us a call at (615) 870-4960. If your disk is damaged, we'll get a new one to you as soon as possible. There are ten items in the root directory of this issue's disk. In alphabetical order they are:

a.Read.Me
Lots of stuff 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 out about it in this file. This is a Teach Text file. Use EGOed v1.3 to read it.

Compat.Guides
This folder contains the complete text of the previous installments of our Compatibility Guide department. These three files, **V1.N1.Compat**, **V1.N2.Compat** and **V1.N5.Compat** are plain text files that you can read with EGOed or any other program that can read text files. We were expecting Apple to release a new version of the System Software at KansasFest but, they didn't, so we don't have any new compatability information for this issue.

EGOed.1.3
The EGOed update is in the folder named EGOed.1.3 on your *GS+* disk. There are seven files in this folder:
EGOed - This is the EGOed NDA.
EGOed.1.3.p - This is the TML Pascal II source code for EGOed v1.3.
EGOed.r - This is the TML Pascal II resource file for EGOed v1.3.
EGOed.rez - This is the REZ language file that is used to create the file EGOed.r.
HiBit.Example - This file contains an example of what you might see when you open a file the has Hi-Bit characters in it.
Smaller - This is an APW EXEC file that is used to compact EGOed after it has been compiled with TML Pascal II.
Teach.Example - This is an example Teach Text file.

EGOed works only with System Software v5.0 and later. This means that your system must have at least 512K of memory (but more is better). It will NOT work with System Software v4.0! This is because several of the tools required by EGOed (the Text Edit Tool Set and Resource Manager in particular) do not exist in System 4.0. To install EGOed, use the Finder to copy the file EGOed from the **EGOed.1.3** folder on your GS+ disk to the ***:System:Desk.Accs:** folder on your startup disk. Do not rename the EGOed file! Once you have the file copied, you must restart your IIGS to make EGOed available from the Apple menu.

Icons
This folder contains the Finder icons discussed in Icons article on page 25. To use them, copy them into the **Icons** folder of the disks that you run each of these programs from. The next time you are in the Finder, you should see the new icon for the program.

MDBasic.Rev
This folder contains the two examples referred to in the MD-BASIC review on page 30. These are both plain text files.
MDBasic.Source - This is the MD-BASIC source code that creates the example AppleSoft Basic program in the file ASoft.Result.
ASoft.Result - This is the AppleSoft Basic equivalent generated by the source code in the file MDBasic.Source. This is not an AppleSoft Basic file.

MousePosition
This folder contains two items:
Mouse.Pos.CC - This is the ORCA/C source code for the Mouse Position NDA.
Mouse.Position - This is the Mouse Position NDA. For more information on how to install and use Mouse Position, see page 22.

PreFixer.1.01 .
This folder contains seven files:
PreFixer.CDEV - This is the PreFixer v1.01 Control Panel Device. Using the Finder, copy it into the ***:System:CDevs:** folder of your startup disk. It will then be available from the Control Panel NDA.
PreFixer.Pas - This is the ORCA/Pascal source code for PreFixer.
PreFixer.REZ - This is the REZ language code for the PreFixer CDev.
Make, Make2, Make3 and Make4 - These are APW EXEC language files that are used to compile and link PreFixer.

Problem.Form
This is the *GS+* problem report form. If you have a problem with, or find a bug, in one of our programs, fill out this form and send it to us via one of the means listed on the form. This is a Teach Text file. Use EGOed v1.3 to read it.

Transfusion1.0
This folder contains nine items:
Transfusion - This is the Transfusion NDA. For a discussion of its installation and use, see page 8.
XFusion.H, XWindow.CC, XCom.CC, XTrans.CC, XFusion.CC - These are the ORCA/C source code files for the Transfusion NDA. For a more detailed discussion of these files, see page 13.
XFusionHistory - This is a text file detailing the changes that Transfusion has gone through.
Make - This is an APW EXEC file that is used to compile Transfusion.
XMODEM.TXT - This is a plain text file that details the XModem file transfer protocol. This file is public domain.

Writers.Guide
There is only one file in this folder: **Writers.Guide**. This is a Teach Text file that tells you what you need to do to write reviews, articles, programs, etc. for *GS+*. Use EGOed v1.3 to read or print this file.

Remember, the contents of the *GS+* disk is not public domain or shareware! Please do not give away copies of it or any of the programs on it. If you do, we will not be able to stay in business. It really is that simple!

ICONS

By Steven W. Disbrow

If you love Finder icons, you are going to love what you find in the **Icons** folder of your *GS+* disk! Let's cut the small talk and get right to the descriptions, shall we?

The following icons were done by Jami Lowery of Chattanooga, TN:

Blockout.Icon
This is an icon for the great new game, BLOCKOUT (see review on page 37).

TForce1.Icon,TForce2.Icon, TForce3.Icon
When I asked Jami to do an icon for Task Force (see review on page 35), she came back with three. They were all so good, I could not decide which one to publish. So, I decided to let you decide!

As part of his OMEGA review (see page 38), Bryan Walker, stationed somewhere in Germany also sent along an icon. It's in the file **Omega.Icon**.

In our last issue, we published several icons by Dino Bagdadi of N. Miami Beach, FL. Since then, several people have either written or called to say how much they liked Dino's new folder icon. So, we are publishing it again in this issue! It is in the file **NewDir.Icon**.

These last few icons were done by me:

Crylla.Icons
A set of icons for the game 2088: The Cryllan Mission (see review on page 40).

DBMaster.Icon
An icon for the relational database manager from Stone Edge Technologies, Inc.

Rev.76.Icon
An icon for the game, Revolution '76 (see review on page 44).

TeachText.Icon
An icon to help you easily identify any Teach Text files you might create with EGOed v1.3.

XFusion.Icon
An icon for the Transfusion NDA.

To use any of these icons, simply copy them from the **Icons** folder of your *GS+* disk into the **Icons** folder of the disk that you run the program from. You will probably want to copy the **NewDir.Icon, TeachText.Icon,** and **XFusion.Icon** into the **Icons** folder of the disk you start your computer with.

As always, we want to see your custom icons! If you have any that you have done, send them to us here at *GS+* so that we can share them with the rest of our readers.

AMR AS800K 3.5-INCH DRIVE

Retail price - $259
Typical mail order price - $189

American Micro Research
13505 Yorba Avenue, Unit A
Chino, CA 91710
(714) 590-3900
FAX: (714) 590-3643

Reviewed By Steven W. Disbrow

LIFE IS HARD...

Even if you already have a hard drive, having two 3.5-inch drives can make your life much easier. And, if you don't have a hard drive, a second 3.5-inch drive can make your life infinitely easier! However, like most everything else made by Apple Computer, Apples 3.5-inch drives are priced to be bought by Government agencies and not your everyday consumer. So, for most IIGS owners, about the only time the purchase of a 3.5-inch drive is even considered is when they first buy their IIGS. Fortunately, AMR has changed all of that.

COMPATIBLE?

The AMR 3.5-inch drive is an 800K 3.5-inch drive that is (as far as I can tell) 100% compatible with the Apple 3.5 drive. Over the last two months, I have put the AMR 3.5-inch drive to just about every use that a 3.5-inch drive can be put to: copying files, copying disks, formatting disks, ejecting disks, inserting disks, even sticking a paper-clip into the little hole to eject a disk when the power went out! In all of these tasks, the AMR drive performed flawlessly. And, from the software side of things, I have yet to find a program that does not think it is an Apple 3.5 drive.

The AMR 3.5-inch drive also has a daisy chain port on the back. So, just like an Apple drive, you can connect other drives to it. I have had it as the first and second drive on my system and in both positions, it worked flawlessly. I also tested it on a ROM 03 IIGS; again, the AMR 3.5-inch drive performed perfectly.

DEPENDABLE? YOU TELL ME...

I used the AMR 3.5-inch drive to duplicate all 500 of last issue's *GS+* disks and I am using it again to duplicate all 500 of this issue's *GS+* Disks. I've used it to format over 1000 disks in a 3 month period, and the only disk that has been returned so far was one that I forgot to format. To me, that's *very* dependable.

DIFFERENCES?

Even with its perfect compatibility and dependability, there are a few differences between the AMR 3.5-inch drive and the Apple drive. The first is that the AMR 3.5-inch drive looks different. It's just a bit smaller than the Apple drive and it has a much larger disk eject button. The color is the same: dirty grey (Apple calls this color "Platinum.") The only other difference is that the AMR 3.5-inch drive costs about $150 less.

PROBLEMS?

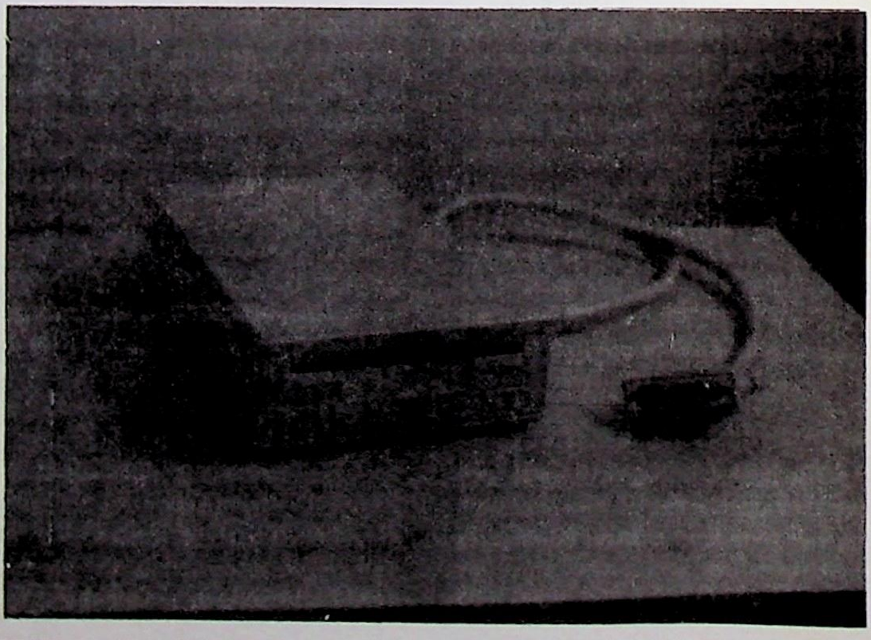
Oh, there are a few problems with the AMR 3.5-inch drive. But, for the price, I can live with them.

The first problem is that the AMR 3.5–inch drive is noisy. Not like an airplane taking off or anything, but it's loud enough that you don't need the little red light to tell you that it's working. In fact, if you rest your wrists on your desk when you type, you can *feel* it vibrating when it's working. And when the AMR 3.5-inch drive ejects a disk, it sounds like it's been drinking all night.

The only other problem with the AMR 3.5-inch drive is that it does not "grab" disks when you insert them. You have to push the disk all the way in yourself. This is no big deal, but if you are used to the way Apple drives "swallow" disks, it can take some getting used to.

SO, SHOULD YOU BUY IT?

The AMR 3.5-inch drive is a great product. That's why this review is so short. If you want a first or second 3.5-inch disk drive (for any reason), the AMR 3.5-inch drive is the drive to get. You don't get the little Apple on the front, but, with the AMR 3.5-inch drive's excellent performance, reliability, and lower price, that really does not matter does it?



SALVATION: THE EXORCISER

By Joe Jaworski
Retail price - $39.95
Typical mail order price - $27
Not copy protected
Requires 768K

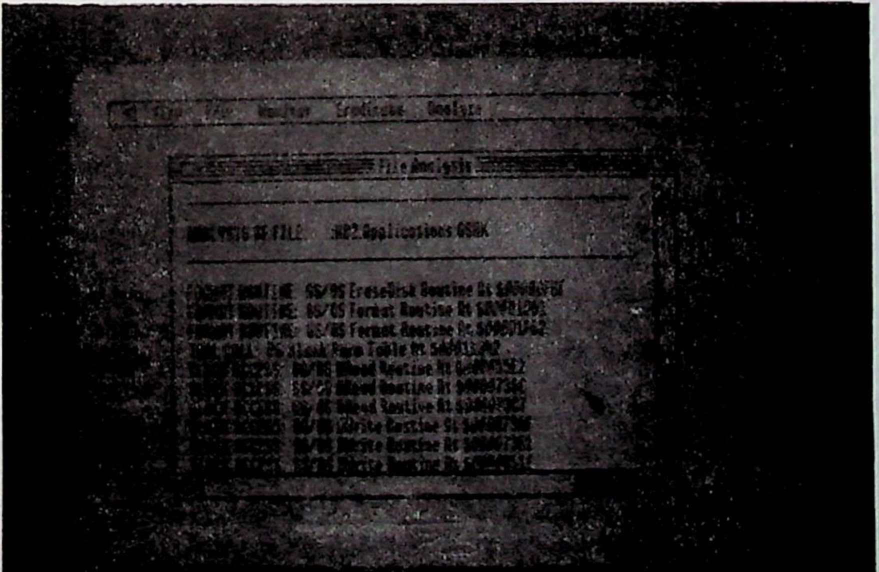
Vitesse, Inc.
13909 Amar Road, Suite 2A
La Puente, CA 91746
Technical Support (818) 813-1274
Orders: (800) 777-7344

Reviewed by Dino Bagdadi

The past several years have seen great technological advancements in the field of computers. Along with these advancements have come drawbacks, one of them being computer viruses. Computer viruses are not, for the most part, friendly. Each virus works in a different way, but most of them serve the same purpose: to cause a great upheaval in your computer system. In general, a computer virus is a self-contained program which attaches itself, or is buried within, another program. After a determined sequence of events, ranging from how many times you execute the program to what date and time it is executed on, the virus is triggered and—BOOM—your nightmare has begun. There are very few viruses in the Apple II world, but they do exist. Fortunately, we now have The Exorciser, a virus detector and cure.

BASIC FEATURES...

The Exorciser serves three distinct functions: checking for specific viruses and curing them; monitoring a file for changes, and analyzing a file for new viruses. The three specific viruses The Exorciser currently checks for are: CYBERAIDS, FESTERING HATE, and LODE RUNNER. As new viruses appear (and let's keep our fingers crossed that they don't), The Exorciser will have the ability to check for those too. "How?", you may ask. To combat new viruses, the Exorciser is expandable through ".VCheck" modules. These ".VCheck" modules can be added to the program much like Desk Accessories can be added to your system disk, or like XCMDs can be added to HyperStudio. The basic functions of The Exorciser will not change with the addition of these modules, but they will expand on the viruses that the program can recognize and destroy. These modules, each with their own addendum to the manual, will be available to all Exorciser users for a fee (not determined at the time this review was being made), and included in new releases as they become available (and, unfortunately, necessary). Speaking of .VChecks, The Exorciser has its own .VCheck module to protect itself from being infected by any viruses! The program will not execute if this .VCheck is not present on the same directory as the actual program.



IN DETAIL...

The feature that you would probably use the most in The Exorciser, is Eradicate. This option checks for and destroys, the viruses for which you have .VCheck files installed. Easily enough, you just select the volume (any ProDOS volume on line, be it 3.5-inch or 5.25-inch disks or a hard drive partition) which you would like to check from a standard GS/OS dialog box and the program takes over from there. In this mode, you can check for all viruses at the same time, or one at a time. The program, whether or not it finds the virus it is checking for, will alert you of its results, and allow you to take the necessary measures to destroy the virus; this way you are not left wondering if you were infected by a virus or not. As new .VCheck modules appear, the list of viruses that the program can automatically check for, will increase. The other two features, Monitor and Analyze files, are the tools you will need in checking for new "viral strains".

The Monitor feature allows you to make a Monitor file for any file on a volume. This Monitor file contains information that allows The Exorciser to check a file for changes in the following things: Data Fork, Resource Fork, File Type, Auxiliary Type, Storage Type, Access code, Creation Date, and Modification Date.

Alterations in the Data or Resource Fork, can be signs of a virus setting up shop. The other parameters can change due to normal circumstances, but that does not mean that a virus cannot affect them. By normal circumstances, I mean making a file such as a Desk Accessory, INIT, or Driver, active or inactive, locking or unlocking any file, or modifying the date of a file by working on it and then saving it (like a text file or picture, for example).

To generate a Monitor file, you simply select the file you wish to monitor and then save it through the standard GS/OS Save File dialog box. The program will automatically assign a name closely resembling the original file and append the suffix ".Mon" to it. You can, if you wish, assign it your own name, of course. The way the program does it though, is very effective and less confusing than assigning the file a random name. One thing you should keep in mind when generating Monitor files is to NOT move the original file from the current directory. The reason being that Monitor files record the pathname of the file they monitor; thus, moving or renaming the original file might cause the Monitor file to not find the original file when you want to check on it. This is, of course, all stated in the very clear and detailed manual which is provided with The Exorciser. Once you generate and save a Monitor file, you can always use the program to check on the original file using the Monitor file. If any changes in the parameters on which it checks are noted, The Exorciser will inform you of them, one by one. If no changes are noted, The Exorciser will tell you so.

The third feature of this program is the ability to analyze any file for new viruses. "How can you analyze a file to find a virus?" you may ask. The Analyze function is a kind of mini-disassembler which scans for "destructive" ProDOS 8, GS/OS, and Tool calls. "Destructive" calls means those calls that perform format or erase operations, block-level reads and writes, and other calls that have the potential to modify or destroy data on any volume. The type of calls the Analyze function reports are as follows:

1. Time-Related Checks: tool calls like ReadTimeHex, WriteTimeHex, and GetTick, as well as the ProDOS 8 functions: Get_Time, $BF06 calls, or $BF9x location reads.
2. Block-Related Checks: ProDOS 8 and ProDOS 16 Read_Block and Write_Block calls, and GS/OS (inline or stack-based) calls to DRead and DWrite.
3. Format-Related Checks: SmartPort signature and Slot 5 access; $C08x +I/O Slot access, and ProDOS 8, ProDOS 16; and GS/OS Format and EraseDisk calls.
4. Tool Call Checks: GS/OS Toolbox calls to the Disk Utility tool set; ProDOS 8, ProDOS 16, or GS/OS calls where the parameters are empty; any call to the BootInit or Reset functions of any tool; and calls to the Resource Manager's AddResource function.

To analyze a file, you would simply select the file you want to analyze from a standard GS/OS Open File dialog box, which would in turn load the file into memory and check for the specific calls mentioned above. After this was done, The Exorciser would report to you if any destructive calls were found in a scrollable window (see photo on page 27). From this point you have two choices: save the analysis information as a text file or print the information. The Analyze feature is something most people will not have to learn how to use to find The Exorciser useful, but it is provided for the experienced machine language programmer. The Analyze function does not discriminate on which file it analyzes. By this, I mean that Analyzer will disassemble any byte pattern that is the same as any destructive call. For example, a byte sequence in the middle of a graphic file could be the same as that of a program which formats a disk. There is no way, of course, that the graphic file's byte pattern can perform this formatting operation without any additional instructions. The bottom line is: if you can not tell if the byte sequence is a destructive call or just a coincidence, then you will not gain anything from the Analyze function. However, in my opinion, the Analyze function is a nice extra feature to this already useful program.

THE BUILDING BLOCKS...

The Exorciser, besides doing a great job at what it is supposed to do, does it with style. The program follows Apple Human Interface Guidelines very strictly, making the user feel at home. Not only does it do it with style, it does it in a very colorful way—all Open File and Save File dialogue boxes have a red frame around them! This program was so intuitively easy to use, that the manual was really not needed. Nevertheless, it was very clear in its explanations and informative to read. The manual actually explains on the working mechanism of the viruses it checks on, along with sample screens of what each virus displays when it takes its effect. The manual comes in an attractive loose leaf binder, which I find to be a great idea, to allow for the addition of the .VCheck manuals as they come available. The Exorciser, because it is not copy-protected, includes a copy of Apple's Installer to install the program and its icons to other volumes.

THE STUMBLING BLOCKS...

This program has really nothing to pick at with the exception of one thing I discovered through its usage. If you tried generating a Monitor file of or Analyzing any file with 0 bytes in its data fork, it would lock up and crash. "Unreal!" I thought. I quickly contacted one of Vitesse's representatives and explained to him my problem. Needless to say, he relayed the problem to Joe Jaworski who had a new version with the fix for it three days after! Is this quick customer support, or what? This new version, 1.01, can be acquired from Vitesse by mailing your original disk in a reusable disk mailer with $1 to cover the shipping costs. That's it! Vitesse's customer support and dedication to their business is very hard to beat... an example that other companies should strive to emulate!

AREAS OF IMPROVEMENT...

There is only one other thing that I would like The Exorciser to let me do: save or print the report of Monitor files the same way that you can with the Analyze function. Besides that, the program is very complete in every aspect.

PUTTING IT ALL TOGETHER...

The Exorciser serves its function very well. It is user-friendly, easy to use, and most important, expandable to allow for new viruses. This program is intended for people who believe they are at risk of catching a virus. These usually include people that are involved in telecommunication systems, be it the so called "Pirate" boards, which typically have no access charges and have available deprotected copies of commercial software, or nationally-known BBS's (Bulletin Board Systems), like GEnie, America Online, and CompuServe. These nationally-known boards however, are usually (and in my experience, **always**) very safe, because their operators, beings of high moral fiber, are constantly on the lookout for detrimental files in the system, and will reject any suspicious uploads. This does not mean, however, that you cannot catch a virus from a commercial on-line system, nor does it imply that all "free" BBS's have infected programs. At any rate, if you are such a person, The Exorciser is for you!

DISK ACCESS

By Steve Stephenson
Retail price - $49.95
Not copy-protected
Requires System Software v4.0 or later
and 128K free RAM

Seven Hills Software
2310 Oxford Road
Tallahassee, FL 32304-3930
Orders: (800) 627-3836
Technical Support: (904) 576-9415
FAX: (904) 575-2015

Reviewed by Steven W. Disbrow

VERY IMPRESSIVE!

Disk Access is a New Desk Accessory (NDA) that performs an incredible number of disk maintenance functions and is always available from any application that correctly supports NDAs. You can use Disk Access to copy files, delete files, rename files, move files, format disks, eject disks, print files... the list goes on and on. Disk Access works with the newer, GS/OS extended files (i.e. files with Resource Forks) as well as with older files. Disk Access does exactly what it promises, which is rare for any program on any computer.

Another rarity that you get with Disk Access is great documentation. Disk Access comes with a 40-page user's manual that contains very complete descriptions of all of Disk Access' functions. There are four appendices, a very complete index, even a pull-out quick reference card!

Another bonus that you get with Disk Access is Out To Launch. It's a very simple program that does the only thing that the Finder can do that Disk Access cannot—launch other programs. To be honest, I love the Finder and have too much time invested in custom icons to even consider using Out To Launch. But, if you loathe the Finder, and hate them darn icons, Out To Launch and Disk Access make for a very usable substitute.

IT'S NOT FOR EVERYBODY.

For all its power, however, Disk Access is not for everyone. Its size, 52K, combined with the disk space requirements of GS/OS, pretty much rules out using Disk Access on anything but a hard drive system. An 800K floppy just can't hold everything, unless you replace the Finder with Out To Launch.

Due to the large number of options that it has, Disk Access is fairly complex. When you first open Disk Access, you are presented with a list of all of the volumes that Disk Access recognizes and 15 different buttons for you to click on. Double-click the mouse on one of the volumes in the list and you will be given a list of the files and folders on that volume. You can open folders by double-clicking on them and you can select files and/or folders by clicking, shift-clicking or Open-Apple-clicking on them. Once you select the files/folders you want to work on, you click one of the buttons to perform a particular operation on the items you have selected (delete, for example.) There is a command key equivalent for almost every button, so, if you want, you can use those instead of the mouse. The only catch here is that the pathnames of the files and folders that Disk Access can work with cannot be more than 64 characters long. If they are, Disk Access will return a "Pathname too long" error and refuse to complete the operation. The fact that Disk Access is best used from a hard drive (where pathnames longer than 64 characters are common) makes this limitation a little ironic and more than a bit frustrating! Hopefully, Seven Hills will update Disk Access to remove this limitation.

A NEAT TRICK.

Another nice feature of Disk Access is that it works well in both 320 and 640 graphics modes. This is an especially neat trick when you consider how much information Disk Access has to display at any given time. How does Disk Access do it? Well, if you activate Disk Access, either by clicking in its window or selecting it from the Apple menu, while you are in a 320 mode program, Disk Access changes the resolution to 640 mode! When you deactivate Disk Access, it switches the resolution back to 320 mode. While this scared the crud out of me the first time I saw it happen (the screen goes black for a split second), I am very glad that Mr. Stephenson went to the trouble to accommodate those of us that do sometimes work with 320 mode programs. Hopefully, this is something that other desk accessory authors will do as well.

IS THERE NOTHING WRONG WITH THIS PROGRAM?

The only real problem I have with Disk Access is that, if you close the Disk Access window and open it again, it does not remember the folder you were working in when you closed it. It is a royal pain having to reopen a bunch of folders just to get where you want to go. Of course, you could always leave the Disk Access window open, but it's just so big! It would be nice if Disk Access could use one of the 32 GS/OS prefixes to keep track of where you were the last time you used Disk Access.

IF YOU NEED IT, BUY IT.

Disk Access is a great product—it is powerful, well-written, well-documented, fairly-priced and goes a long way towards increasing a IIGS user's productivity. However, if all you use your IIGS for is games or AppleWorks, Disk Access is not for you. But, if you live, as I do, in the world of the Finder and other desktop programs, and if you have the necessary hardware (a hard drive) and enough memory, you should buy Disk Access.

MD-BASIC
By Morgan Davis
Retail price - $49.95
Not copy-protected
Requires 1.25 MB stand-alone or 250K above your shell's requirements

The Morgan Davis Group
10079 Nuerto Lane
Rancho San Diego, CA 92078-1736
Voice: (619) 670-0563
BBS: (619) 670-5379

Reviewed By Daniel Davidson

Does the thought of writing programs in AppleSoft BASIC make you want to gag? Do you have a need to write programs in AppleSoft, but hate the AppleSoft environment? Anyone who has to write programs in AppleSoft knows what a pain it can be. Its limitations make program maintenance a nightmare. Limits such as two character variable names, terrible editing features, and poor formatting of the program listings makes it difficult to write, and later improve, the sort of useful programs that can be easily done in other languages such as C.

Unfortunately, AppleSoft will not die. Even with its limitations, it is built into every Apple II made since the II+ came out, so it is a common denominator in the Apple II world. As a result, a lot of programs are written in AppleSoft so they can work on any Apple II.

AppleSoft itself may never get any better, but writing programs in AppleSoft just did. MD-BASIC may well change the way people use and think about programming in AppleSoft.

DESCRIPTION

MD-BASIC is best described as a quasi-compiler that runs on the Apple IIGS, under most program launchers, including ORCA/M, ECP 16, ProSEL 16, and even the Finder. It takes structured, free-form BASIC programs and converts them into everyday AppleSoft. Why, you may ask, would anyone want to do that instead of just typing up the program from the ']' prompt? Once you see what you can do with MD-BASIC, you will understand.

The input for the MD-BASIC compiler is a standard text file containing a structured BASIC program (see the file EXAMPLE1 in the folder **MDBasic.Rev** on the *GS+* disk) which can contain long, meaningful variable names and labels. The output file will be normal AppleSoft with one- and two-letter variables, and line numbers just like any other AppleSoft program (see the file EXAMPLE2 in the folder MDBasic.Rev on the *GS+* disk).

The more organized listings, and the use of real variable and labels names would be enough for many people, but MD-BASIC offers much more than just that. MD-BASIC adds If-Then-Else, While-Wend, Do-Loop and Repeat-Until constructs, so programming is much easier, and the readability of the source code is greatly increased over standard AppleSoft. To make your source even more readable, you can add as many comments as you want, without worrying about them taking up space in the final output file.

The MD-BASIC Compiler has many compiler directives to make your life easier. These directives permit inclusion of definition files and other source modules, conditional compilation, program optimization, variable cross reference generation, and many other useful operations.

INSTALLATION

The installation of MD-BASIC is accomplished in one of two ways. If you do not use the ORCA/M or APW shell, there is an installation program that uses an AppleWorks-like file card menu system. The installer will let you set up a stand-alone version of MD-BASIC, or it will install it into the external program directory of shell programs like ProSEL, and ECP 16. The program will walk you through the installation, and give you new information about the program that has not yet found its way into the manual. If you are using the ORCA/M or APW shell, there is a installation script that runs under the shell to move the needed file to your system disk.

USING MD-BASIC

Since MD-basic can be used either as a stand-alone program or from a shell, and since I do most of my programming under a shell (UNIX at school and ORCA/M at home), I started with using it from a shell.

When used from a shell, MD-BASIC works much like any normal compiler. You use your favorite editor to type up the program, and then "compile" it with a command from the shell. To test it, you have to start up BASIC.SYSTEM and run the program. If there is a problem, you type "BYE" from the AppleSoft prompt, and it returns you to your shell, where you can fix the problem and then try it again.

For some small problems, it is possible to edit the AppleSoft code that is generated by the compiler. Be warned, though, that MD-BASIC changes all the variable names, and does some optimization, so the AppleSoft program may look quite strange. Also, since MD-BASIC doesn't use the line input routines in creating the lines of BASIC code, some lines will be to long to edit from the AppleSoft ']' prompt. This is not a big drawback, since it is easier to edit the MD-BASIC source code than it is to edit the AppleSoft output.

The stand-alone version uses a very simple text-based menu system. It only allows you to set the prefix, edit files, compile files, decompile files, enter BASIC.SYSTEM, and quit. Not very many options, but enough to let you take advantage of MD-BASIC without having a real shell. Fortunately, the user interface is simple, and even the most novice user can figure it out. The most glaring thing lacking from the stand-alone setup is the ability to get a catalog from the disk without dropping into BASIC.SYSTEM first. Admittedly, this is a nasty drawback, but luckily several free and inexpensive shells are available, so you don't have to use the stand-alone version.

MicroEmacs is included with MD-BASIC, and is the editor that is installed when you choose to use MD-BASIC in the stand-alone configuration. Emacs is a text editor written by the Free Software Foundation (the people writing the UNIX clone called GNU) that was originally written to run on UNIX systems. MicroEmacs is a smaller version of Emacs that will run on personal computers. It is a very powerful editor that lets you have multiple files on screen at once, but the commands are a bit difficult to learn. Included is a help file about MicroEmacs, but it is not as helpful as it could be. Most people who don't already know how to use Emacs or one of the many Emacs clones like JOVE or Amacs, should strongly consider using a different, more user-friendly, text editor.

Since I like command line interfaces, I quickly tired of the stand-alone configuration, and was ready to get back into the ORCA/M shell. People who are used to using shells similar to ORCA/M will want to make use of a shell instead of the stand-alone version. For those who cannot afford the price of ORCA/M, one option to consider is ECP 16 by Don Elton. ECP 16 is freeware, and will make using MD-BASIC much more convenient than is possible with the stand-alone version.

MORE GOODIES & SUPPORT TOO

Another option for people who don't like command line shells, or the stand-alone mini-shell that comes with MD-BASIC, yet another option is the Finder. To launch MD-BASIC from the Finder all you need to do is change the file type from EXEC to SYS16, and use it like any other program. With a NDA text editor such as EGOed, this is a very convenient way to use MD-BASIC. Simply edit and save the files with EGOed, and then click on the MD-BASIC icon. It will give you a line to type the compiler options on, and then away you go. Fair warning, though; this is not the normal pretty GS/OS program that uses windows and the mouse. People who must have windows should look somewhere else. This is a text-based programing tool, not an overly user-friendly desktop program.

When I first tried to launch the MD-BASIC compiler from the Finder it gave me the famous "Unable to find Application for this Document" error. So, I whipped out my handy-dandy file type changer, and changed the file type to SYS16. I then tried to launch it again, and it worked just fine until it went to quit. Instead of quitting back to the Finder it crashed into the monitor. Not a good sign, I thought. I looked in the manual, and called the Customer Support number. Since it was after 9:00 P.M., I really didn't expect an answer, but-low and behold-there was, and it was a real person too, not just an answering machine. I told the person who answered the phone what the problem was, and she had me wait just a second while she went and got the author. Morgan walked me through what I needed to do to run MD-BASIC from the Finder, and it still did not work. It compiled things fine, but still crashed when it went to quit. After a minute of walking through the possible problems, he thought It might be that I have an older version of MD-BASIC, and arranged to send me a new one via E-mail so I could have it A.S.A.P. Shortly thereafter, the newest version arrived, and I tried it again. It worked exactly as Morgan said it would.

TESTING MD-BASIC

To test MD-BASIC, I took a Sysop utility I had written for my BBS, and tried converting it to MD-BASIC. First I used the "Decompiler" that comes with MD-BASIC. It takes a normal AppleSoft program, and converts it into the MD-BASIC. The conversion is not perfect, but it is very good. It will convert more than ninety percent of your program accurately, which is no small feat. All that was left to do was edit some of the problem areas, and re-compile the program. In all, I spent less then two hours editing, commenting, and improving my program. When I was finished, I had a very nice looking source file with lots of comments, and a working BASIC program that was shorter, faster, and more robust than the original program. Now, if I ever decide to improve on this program it is very easy to do since I have it commented, and easily readable.

RANDOM THOUGHTS

Let me clarify a few things about MD-BASIC. It requires a IIGS to run, but the programs it creates do not since they are just plain old AppleSoft BASIC. This means that you cannot write the next great GS/OS application with it, but what you do write can be run on just about every Apple II computer ever made. You can however use ampersand add-ons with it that you can use with AppleSoft. So, If you want to write a BBS, you can use ModemWorks. If you want to write a program that uses the IIGS sound and graphics, you can use something like Spectra Graphics. You can also run the output of MD-BASIC through any of a number of AppleSoft compilers like the Beagle Compiler. In short, since MD-BASIC outputs normal AppleSoft programs, anything you can do from AppleSoft can be done from MD-BASIC.

One very helpful feature of MD-BASIC are the included libraries. With them you can very easily do AppleWorks style file card menus, MouseText windows, ROM and ProDOS MLI calls, and many other things that are time-consuming to do from normal AppleSoft. If you find that the libraries are not quite what you want, you can always make your own library files to improve on the ones that are included.

CONCLUSIONS

I can't say enough good things about the program to do it justice. It is a wonderful tool for creating AppleSoft programs, and with every release, it just seems to get better. The only thing I can think of that would make it better, is if there were a version that created GS/OS programs. The support from the Morgan Davis Group is superb. When I first started to write about MD-BASIC it only ran from the ORCA/M (APW) shell. Several people complained to the author about it, and a new version was released that would run from any program launcher. For each question I have had, I was able to get an E-mail answer later the same day, or the next day. When I used the phone for support, Morgan was helpful, and very quick to find the answer. If you have a need to write AppleSoft programs, you should take a very close look at MD-BASIC. If you want to see what MD-BASIC can do, you might consider getting a copy of the demonstration version that is available on their BBS.

KATIE'S FARM

By Frank Andrews and James McCarthy
Retail price - $39.95
Typical mail order price - $28
Not copy-protected
Requires 1 MB

Lawrence Productions, Inc.
1800 South 35th Street
Galesburg, MI 49053-9687
(800) 421-4157

Reviewed by Greg Zimmerman

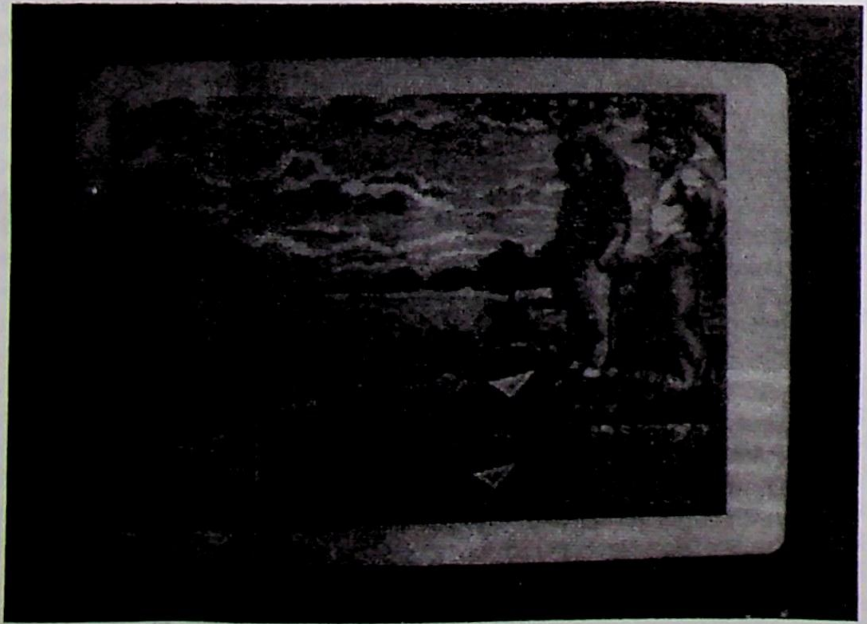
Katie's Farm is the second preschool children's software offering from Lawrence Productions. It is a sequel, of sorts, to the outstanding children's program McGee (see review in *GS+* V1.N5) from the same publisher.

The program is all about a visit to Katie's Farm by her friend McGee. At the direction of your child, Katie and McGee explore all the different things for kids to do at the farm. The program contains no text whatsoever, and there is absolutely no use for the GS keyboard. All of the activity selections the child makes for Katie and McGee are made with the click of the mouse.

The program begins with McGee being dropped off at Katie's Farm. From that moment on, your child decides what activities Katie and McGee will do. On the opening screen, as well as all others, there are four activity choices for the child to choose from. These choices are small pictures at the bottom of the screen which are smaller versions of parts of the main screen graphic. The child clicks the mouse on one of the small pictures, and the screen changes to another scene, or an activity commences. With Katie's Farm, children as young as 2 years old can use the mouse with ease, because the cursor is locked into the lower screen area which contains the four pictorial selections. No matter where the cursor is at the time that the child clicks the mouse, an event will occur. And with each change of scenery, as the children visit different parts of the farm, new selections appear.

So you big city folks ask, "What do people do on a farm, anyway?" Well...

Your child can take Katie and McGee into the garden, and once there, they can play with the scarecrow, pick cherries, or go down to the pond. At the pond, they can go fishing, play with a toy sailboat, or play with a turtle that's resting in the grass. (At the pond, all the kids' activities are conducted under the watchful eyes of Katie's parents.) Your child can take Katie and McGee to the stable, and have Katie's father put McGee up on a horse. Or Katie can feed the horse a carrot. At the chicken coop, they can feed the chickens, gather eggs, or play with the baby chicks. (You mean some of you eat these cute things?) In the barn, Katie, McGee, and your child can watch a cow being milked, they can give some of that milk to the barn cats to drink, or they can jump into the a big pile of hay. Or maybe they just want to relax, and watch some baby birds chirping in their nest up in the old tree out front. Or they can watch the squirrel climb up into the big knothole in the tree.



The graphics are of high quality, and all the activities are animated. Appropriate sounds accompany the activities, such as the cat's meow, the horse's neigh, and McGee's laugh, and the sound quality is as good as the little GS speaker can deliver. Whatever your child chooses to do at Katie's, there are lots of possibilities, and I didn't notice a television blaring cartoons anywhere on the farm.

NUTS AND BOLTS

Katie's Farm does not follow the Apple Human Interface Guidelines (of course Roget's Thesaurus does not give "two-year-old" as a synonym for "human"). No pull-down menus, no sky blue, and no desktop metaphor. The "menu" is the row of pictures at the bottom of each screen from which your child makes selections. This user interface is what allows extremely young children to operate the program, interact with the computer, steer the story in any direction, and set any comfortable pace the child desires.

Because Katie's Farm is ALL graphics and sounds, the program comes on two disks. It is hard-drive-installable, with easy-to-follow installation instructions included in the manual (this is the only reason that most parents will ever need to look at the manual). Katie's Farm is also System Software v5.0.2 compatible, and will run on either a ROM 01 or ROM 03 GS.

Loading time naturally varies depending on your equipment. From a 40 MB InnerDrive, with a 7 MHz TransWarp, on a ROM 01 machine, it was 11 seconds to the "Hi, I'm McGee" screen, and 47 seconds until the child could begin making selections. Booting from the 3.5-inch floppies with two 3.5-inch drives and a TransWarp installed lengthened these times to 46 seconds to "Hi", and to a full 1:34 until activities could be selected. No TransWarp and only one drive? 1:57 is the wait time until activities can be selected, with one disk swap along the way. To the publisher's credit, there are two animated/sound scenes that appear on the screen during part of this wait, which helps pass the time a little easier.

The program installs a Classic Desk Accessory (CDA) upon boot-up, which contains the names of the people who developed Katie's Farm. Also, when the child is finished using the program, hard drive users can hit Open-Apple-Q to get back to the program launcher.

Lawrence Productions has a toll-free number you can call for customer service. Though you are unlikely to have any questions about the program, if you do have a problem, you'll be happy to know I have always gotten through to them on the first try, and received informative, helpful answers to all of my questions. When I called them, I did ask the most important question, "Is more GS software on the way?" The answer is yes, at least two more GS specific programs are in development. Judging by the quality of their first two offerings, this is good news for GS owners.

DO I LIKE IT?

This program is so smooth, so intuitive, and so good for its intended audience of two- to four-year-old children, that I rate it as an excellent piece of software. Here's why:

Katie's alarmingly simple user interface is an outstanding feature for young children's programming. My two-year-old has no trouble working his way through the different activities at the farm with Katie and McGee. Locking the cursor into the lower screen area makes it easy for the youngest of users to begin working with the mouse, and to build confidence in their own ability to use the computer in a non-frustrating and comfortable way.

Another good feature is that when the child selects the same activity a second time, the exact same thing doesn't always happen. Virtually all of the activities have an alternate event that occurs. So one time when the kids go fishing, they come up with an empty hook. But the next time, they may catch something.

The sounds are good, the graphics are great, and the animation is on par with other top-quality GS programs.

Katie isn't just a way to keep your child busy. There is some real educational value here, particularly for very young kids. ("Mommy, why was the egg under the chicken?")

The thing that most keeps the youngest of children glued to the computer with Katie and McGee, is the control that the child has over the program. The child decides the direction the program will go, and the child decides what activities to select. The ease with which selections can be made, and the interesting nature of the program material, along with the high quality sounds and graphics, can keep even a two-year-old sitting in his (mine's a "his") chair for up to an hour.

The fact that the software is not copy-protected is a real plus. Little children tend to have big accidents, but with Katie's Farm, you can protect your investment by easily making backup copies.

IS EVERYTHING GREAT?

Well, the loading time leaves a lot to be desired. Even though the publisher has tried to ease the pain with some entertainment while your child waits, the child is still waiting nonetheless. I don't think patience is hereditary, so most two-year-olds haven't learned to sit around doing nothing for even short periods of time (teenagers are best at that). However, the anticipation of the child, and the on-screen entertainment made it more bearable for the two-year-old than it was for me.

Users that launch Katie's Farm from the 3.5-inch disks will find that using Open-Apple-Q to leave the program re-launches Katie's Farm if disk 1 is still in the boot drive. So get it out of there before quitting, or you will have to wait through the startup routine for disk access to end before you can safely eject the disk.

Getting really picky now, the screen border is always black throughout the program. I assume that this is intended to better highlight the graphics, but a little lighter, brighter, more cheerful color would be my preference. Again, the two-year-old had no complaint.

SHOULD I BUY THIS PROGRAM?

If you have a child between the ages of two and four years old, I would highly recommend Katie's Farm. This program is one of the best for that age group.

Lawrence Productions has done an excellent job of developing a child's software package that is both fun and easy to use. My two-year-old has sat down with Katie's Farm at least 20 times in the six weeks we have had the program, regularly choosing it over some other very highly acclaimed software.

The program builds confidence in the child because the kid can really use the computer without help while running Katie's Farm. It also builds confidence in that the child decides what events are going to take place as he explores the farm with Katie and McGee. There is some educational value for very young children, and the ease of use, combined with the high quality of the sounds, animation, and graphics, will keep your child entertained for periods of time you previously thought were not possible. I recommend this program without reservation.

WHERE IN THE U.S.A. IS CARMEN SANDIEGO?
By Peter Adams of Sculptured Software
Retail price - $44.95
Typical mail order price - $29
No on-disk copy protection
Requires 1 MB

Broderbund Software, Inc.
17 Paul Drive
San Rafael, CA 94903-2101
(800) 527-6263 or (415) 492-3500

Reviewed by Noreen Ribaric

Where in the U.S.A. is Carmen Sandiego? (Carmen USA) is the second of the Carmen Sandiego line to have a GS-specific version released. The first one was Where in the World is Carmen Sandiego? (Carmen World), released a little over a year ago, which we reviewed in the March-April 1990 issue of ***GS+***. For those of you who are not familiar with the Carmen Sandiego line, these programs combine learning geography with adventure, mystery, and humor, resulting in a educational game that's fun to play. The excellent graphics and sounds in the IIGS version add to the enjoyment as well.

The first thing I noticed about the GS version of Carmen USA was that, unlike earlier Carmen Sandiego releases, it has *no* on-disk copy protection (even though the manual states it is key-disk protected). The form of copy protection in this version is that you are required to answer a question from the reference book that comes with the game each time you are promoted in the game. Two thumbs up to Broderbund for the move to no on-disk protection!

The second thing I noticed is that Carmen USA does NOT support desk accessories (even though the game displays the Apple menu and has an About Carmen item in it). I was very disappointed, especially since the GS version of Carmen World *did* support desk accessories. I was hoping to use EGOed to do this review while I played the game, as I did while writing the last Carmen review, but alas, it was not to be. Two thumbs down to Broderbund for this oversight.

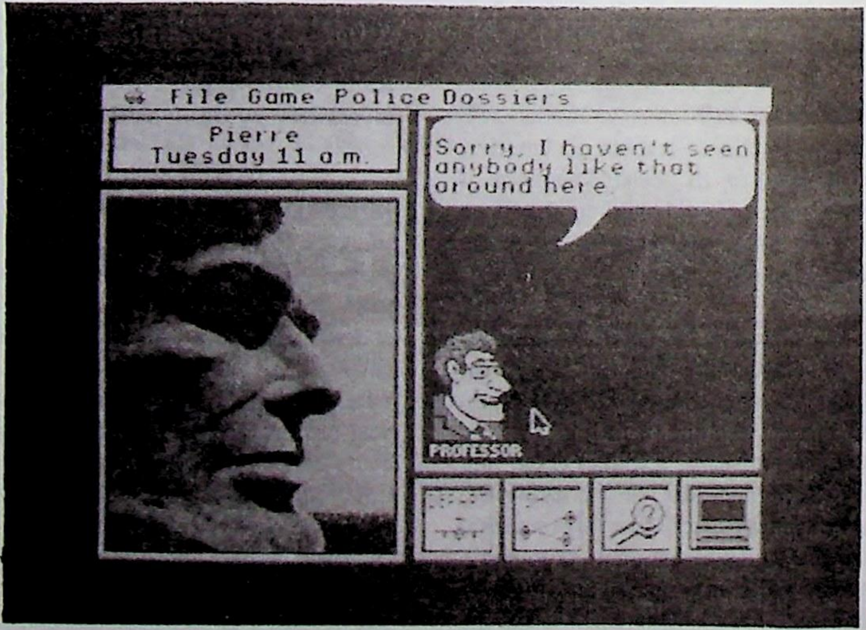
Carmen USA installs easily on a hard drive. There is even a nifty icon for the program included on the disk. The documentation for the game is very good. In addition to the manual describing how to play the game, they also include a SCRAPBOOK with information on all the V.I.L.E. criminals, a poster-map of The United States with all 50 states and major cities labeled, and the Broderbund Edition of *FODOR'S USA*, a travel guide to the United States.

The premise of Carmen USA is that Carmen Sandiego, the leader of the one of the worst criminal gangs in the world, has broken out of her cell in a European prison and headed to the United States to reestablish her Villain's International League of Evil (V.I.L.E.). It is up to you, an agent for the Acme Detective Agency, to track her and her gang down once more. It's not that easy, either. There are 16 villains, including Carmen, and they can be hiding in any of the 50 states or the District of Columbia!

Carmen USA is almost entirely mouse- and menu-driven. Except for entering your name at the start of each playing session, you don't have to use the keyboard at all to play this game. In addition to the Apple menu, there is a File menu which contains the items New Game and Quit. The Game menu displays the Acme Detective Roster and Hall of Fame. The last menu is the Police Dossiers menu which allows you to view files on all the V.I.L.E. villains. There is a small bug when you pull down this menu—the item selection bar extends across the entire screen instead of just the menu. That is, except for pulling down the menu while using the Crime Computer—in this case, you don't see the selection bar at all. A minor annoyance, but something that shouldn't have gotten past Beta testing. A call to Broderbund's technical support line informed me that they know about this bug, but there are no plans to fix it.

At the start of the game, you are presented with an animated sequence with excellent sound and graphics showing Carmen breaking out of prison. It will repeat until a key is pressed, or can be skipped entirely by pressing the escape key.

You start out as a Gumshoe. One of Carmen's villains has stolen a US artifact (treasure?). Your job is to track the villain down by doing some investigation at each city the villain flees to. As you arrive at each location, you are presented with a very nice graphic representing an important aspect of the city or state you are in. You are also given a description of the location, including interesting trivia, history, and other things unique to that area. Be sure to remember these facts for future cases!



Pierre
Tuesday 11 a m.

Sorry, I haven't seen anybody like that around here.

There are four possible actions at each location: INVESTIGATE, COMPUTE, DEPART, and SHOW. You can unearth clues about the villain and the villain's next destination (such as landmarks, places of interest, and local history) by clicking on an icon of a magnifying glass. If you need help determining what city or state the clues are referring to, you can look them up in *FODOR'S USA*. Clues you find about the villain can be entered in the Crime Computer by clicking on the icon of the computer. Enough clues must be entered to single out one villain so that a warrant for arrest can be issued (you can't solve the case without a warrant). After the villain's next destination is determined, clicking on the icon of the airplane will show a map indicating all connecting flights. Just click on a city to take off (and enjoy the pretty realistic sound of jet engines)! Another way to get a list of connecting flights is to click on the SHOW icon. This list can remain visible while performing any of the other three actions. You can also double-click on a city in this list instead of using the DEPART icon.

You only have six days to solve each case, so perform as few actions as possible. Each time you investigate, you use two to four hours. Computing takes three hours, although entering clues takes no time. Departing takes three to eight hours. Be careful about guessing, because a flight made to Hawaii in error will waste 16 hours! If you pick the right city to investigate, you will see an animated graphic of a V.I.L.E. henchman, or some other strange (and sometimes humorous) thing, to let you know you're on the right track. When you finally catch up to the villain, you will be warned with axes, guns, or flower pots and sacks of cement dropped on your head! The animated sequence when you arrest the villain is very good, but be sure you have a warrant—and one for the right villain—or you will get a different reaction!

After solving one case you will be promoted to Jr. Detective (if you have *FODOR'S USA*, of course). You progress through ten levels of experience, with the clues becoming harder and scarcer as you go, in addition to having to track the villain farther. Once you become a Super Sleuth and catch Carmen Sandiego (after solving over 40 cases—which is whole lot more than was necessary in Carmen World, by the way), your name will be placed in the Acme Hall of Fame. If you wish to solve more cases after that, you must begin again as a Gumshoe under an assumed name (so Carmen's hit men won't find you)!

Overall, this is an excellent game that is fun to play and makes learning enjoyable. I hope they continue to produce IIGS-specific programs. Although Broderbund has none that are due to be released in the next six months, they had planned to release a couple more (Shufflepuck Cafe and Where in Europe Is Carmen Sandiego?) in the near future. Let's show Broderbund that we want more IIGS-specific programs by purchasing the ones they have released, and calling them to let them know we want more! By the way, if you already have the IIe/IIc version of Where in the U.S.A. is Carmen Sandiego, you can get the IIGS version for just $7.50! Call Broderbund at 1-800-527-6263 for details.

TASK FORCE
By Visual Concepts, Ltd.
Retail price - $49
Typical mail order price - $24
Copy-protected
Requires 768K

FanFare/Britannica Software
P. O. Box 77186
San Francisco, CA 94107
(415) 546-1866
(800) 572-2272

Reviewed by Joe Wankerl

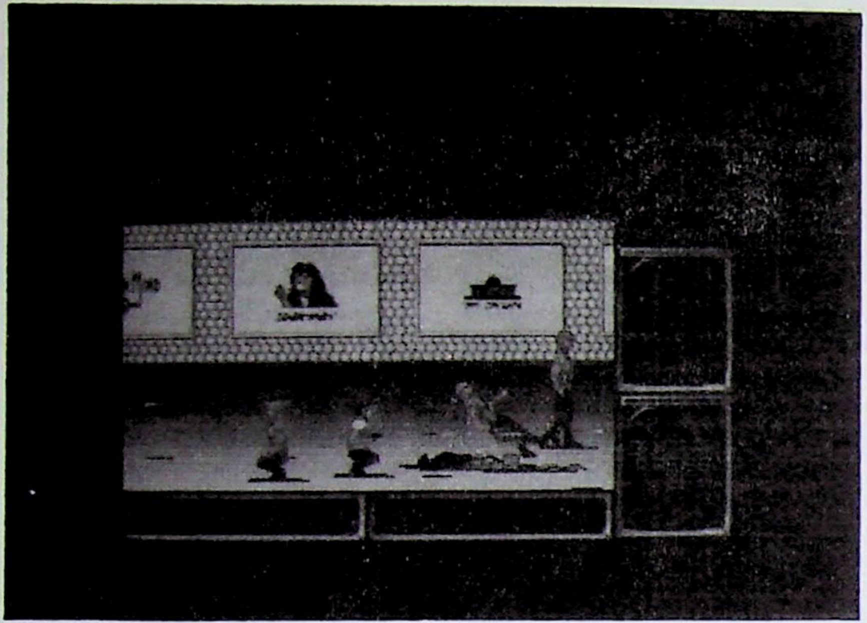
Have you ever had one of those days where you just wanted to get up and shoot things? Ever wanted to take a bazooka to hordes of foes? Well get ready for Task Force!

You are one of two F.B.I. agents selected to take on five cities in the U.S.A. that have been sieged by bad guys. Armed only with your trusty pistol, you (and another player if you want—yes, two players simultaneously!) select your first city, maybe Chicago, and start plugging your enemy with lead!

You watch them go down like flies. "Kill the copper!" they yell as they come at you with even more firepower. You retaliate—blam, blam, blam! HA! What luck! One of them dropped a machine gun! Pick it up and show them what for! Uh oh, you're running low on ammo. You'd better hope somebody drops some soon. "Ahhhh!" another one screams as he bites the dust. Here comes the helicopter—too bad you don't have a missile weapon to take it out of the sky. Watch out, it's dropped a grenade toward you—**BOOM!** You sail through the air and watch your health fall rapidly. What's this? A flame thrower? So early in the game? Let's torch some criminals! Listen to them scream! (And so the game continues....)

As you can see, I really got into this game. I lost many hours of prime study time trying to get through all five cities. It's meant to be a two-player game, one person on the keyboard, and another on a joystick (the joystick is optional, both players can play on the keyboard; however, the joystick play is much more responsive). Steve and I played the two-player variation quite a few times. I realized that you have to have a strategy before hand—know when to squat down and blast the suckers to bits and know when to stand up and run for the next wave. It gets quite frustrating when you want to get up and run and your partner wants to stand and fight.

Although you need just a little bit of strategy, this is not really a thinking type of game. It's a "blast 'em to bits" game. Since I find such games enthralling, I loved this one. If you find such games repulsive, I guess you'll hate this one for all it's worth. All you have



to know is how to press the fire button and how to move out of the way of incoming grenades and enemy bullets. Luckily, that's all your opponents have; if they had more advanced weapons like a tazer or rocket launcher, the game would pretty much be impossible.

There is a marked sequence of events to this game. You start out in F.B.I. headquarters where you will select a city to take on. At first you may not be able to visit some cities because intelligence (military intelligence is an oxymoron) hasn't gotten a report back from the hostile area. Hopefully, you'll start with an easy first city like Chicago or Los Angeles, and not with the killer city, New York. After you select your target city, you are flown by airplane to your destination. You'll see your airplane land at the airport and hear the stewardess announce your arrival. Then the fun begins. You appear on the screen and attempt to kill your quota of bad guys. Once you have caused enough destruction, a red arrow appears pointing you in the direction to the exit so you can get to the next wave of criminal activity. In between some waves, you will be able to contact F.B.I. headquarters to get further instructions and inform them of your progress. After about two or three waves (or more, in the case of New York!) you will have cleared the entire city of corruption and will return to headquarters so you can select your next city.

The animation and sound for this game are excellent! The scrolling background is smooth and flicker-free. Large objects, such as the helicopter and the airplane when it lands, are animated smoothly as well—an impressive achievement! The digitized sounds of the men crying out as they die and the noises your weapons make really enhance the game and provide hours of entertainment.

There were a few annoying features of the game, however, but none of them were bad enough to warrant real complaints to the authors. The airplane scene, in between where you choose what city you want to clean up and where you actually start blasting at things, is a bore. You'll watch it once and then you never want to see it again, although the animation is nice. But it takes so long to load in the sequence and then uncompress it; all I wanted to do was play the game. The same goes for the intermission between waves of criminals where your satellite decoder terminal comes up and you report your progress back to F.B.I. headquarters—you watch it once and then want to skip over it from then on.

Yes, sadly enough, this game is copy-protected. Luckily, it's only a key-disk copy protection so all you have to do is put in your original for a brief second for verification and then you can take it out and lock it back up in the fireproof vault. This also means that it will work on a hard drive, or so the manual says. I was unable to get it to work on my 45 megabyte internal Vulcan or Steve's 60 megabyte CMS external SCSI with an Apple Revision C SCSI controller card. But I was able to work it from a 40 megabyte CMS removable SCSI with the new Apple DMA SCSI controller card. The game uses ProDOS 8, too, so you don't get any GS/OS advantages such as a cache and being able to squirrel the game away in some directory which needs 6K of pathname information to reach.

Through Los Angeles, San Francisco, Chicago, Washington D.C., and finally New York, your nerves will become frazzled making your F.B.I. agent gun down Skinheads, gangsters (my favorite because you can just hold down the fire button and let your machine gun cut them down to ribbons, heh, heh!), punkers, street gangs, and corrupt politicians. Picking up different weapons along the way is a must. A machine gun, flame thrower, tazer, rocket launcher, and the ultimate bazooka all help you to suppress crime. I found myself screaming **"DIE! DIE! DIE!!!"** as I took the rocket launcher to hordes of corrupt politicians, scattering them through the air—and I'm a rather tame, docile person by day. There is also a grenade that you can acquire, although I have yet to run across one. I figure that you have to be in the last wave of the last level with a zillion points to find one. As I stated before, you must kill your quota of bad guys before you can move on to the next wave. Sometimes it's easy when the ammo is plentiful, and sometimes it's sheer heck! I have made it through all cities except for New York, the toughest since you have to make it through the city, then the subway, and finally up the Empire State building. So far I've gotten to the third story—I pray it's the last!

BLOCKOUT
By PZK Co. Development Group
Retail price - $49
Typical mail order price - $25
No on-disk copy protection

California Dreams
780 Montaque Expwy., Suite 403
San Jose, CA 95131
(408) 378-0340

Reviewed By Steven W. Disbrow

ARGHH!

Lord have mercy! It took me almost two years to break my addiction to Tetris, and along comes BLOCKOUT. [Anyone that's ever visited us here at *GS+* can tell you that Tetris, Arkanoid, and Crystal Quest are the real reasons that the magazine keeps coming out later and later - Ed.]

For those of you that just bought your IIGS or have had your head buried in a compiler for the last couple of years, Tetris is a game by Spectrum HoloByte in which you arrange falling shapes in such a way that they do not fill up a two dimensional 'pit'. When it first came out, Tetris was something of a curiosity because it was invented by, "one of them commies" in the Soviet Union. However, Tetris was, and is, so damned addictive, that it was quickly forgotten where the game originated (except for some of us right-thinking Americans that thought it was a plot to reduce American Productivity and Resolve). Well, time went by, and the Soviet Union went from being an "Evil Empire" to just another few digits on a McDonalds sign, and the Tetris craze kind of died out.

OH NO! NOT AGAIN!

So now, the mind control experts at California Dreams have conspired to delay your *GS+* magazines even more by introducing BLOCKOUT. BLOCKOUT uses exactly the same game concept as Tetris, except it's in 3 dimensions. Shapes made of one or more individual blocks fall down a rectangular pit and you try to twist and turn them so that they completely fill a vertical "slice" of the pit. When you completely fill a slice, that slice disappears and the stack of shapes drop a level. If the shapes stack up to the top of the pit, the game is over.

OK. I know, it sounds simple and it sounds stupid. But so did Tetris. Once you try this game though, you just can't stop playing it. Its simplicity is what makes it so addictive. It's so bloody simple that you can't believe you did that poorly and you have to try again. And again. And again. It's amazingly infuriating and fun.

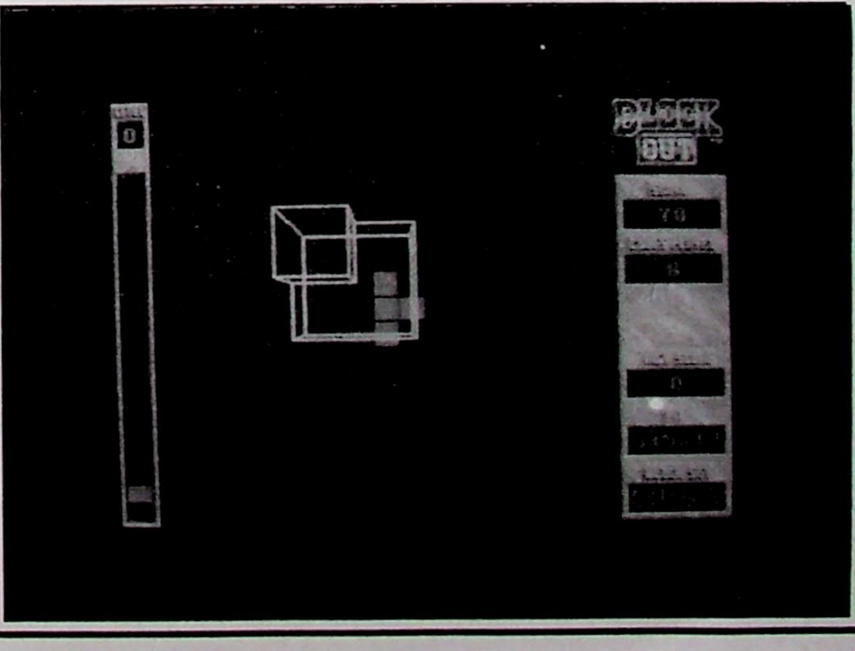
WHAT'S NEW?

So now that I've established that this is a game that you should not be without, let's talk about some of the features that make it more than just a successor to Tetris. First of all, you can configure BLOCKOUT to be as easy or as hard as you want. You can change the physical dimensions of the pit, change the complexity of the shapes that drop down into the pit (there are 41 different shapes made up from one to five individual blocks) and even turn the sound on or off. There are three predefined BLOCKOUT setups and once you hit on a design you really like, you can save it to disk so that the next time you play, you won't have to reconfigure the game. There is an animated help screen that shows you what all of the game controls are and exactly what effect they have on the shapes. Another nice touch is that BLOCKOUT keeps a separate high score screen for each possible pit configuration. All of these things combine to take a great game concept and turn it into an incredibly fun and playable game.

Of course, BLOCKOUT is a game, so it's copy-protected. However, there is no on-disk copy protection, so BLOCKOUT installs easily on a hard drive. California Dreams has chosen a very clever and fairly easy to live with form of protection though. On the inside cover of the manual (which is excellent and, for the most part, unnecessary) are colored drawings of the 41 shapes that BLOCKOUT will throw at you during the course of the game. In the drawings, each individual block in a shape is a different color. When you first start BLOCKOUT, it shows you one of the shapes with one of the individual blocks shaded grey. All you have to do is look at the manual and tell the game what color the shaded block should be. This is the only problem that I have with BLOCKOUT, but I can live with it.

KISS THAT DEADLINE BYE-BYE...

If you didn't like Tetris, you won't like BLOCKOUT. That's for sure. If you did like Tetris, you should get BLOCKOUT as soon as possible. It will bring back memories of those all night Tetris sessions that made you feel like your eyes would fall out. If you just got your IIGS or you want your eyes to fall out, you should get both Tetris and BLOCKOUT. Tetris is a classic that no IIGS owner should be without. BLOCKOUT is an excellent successor to Tetris that will soon be a classic in its own right.



OMEGA
IIGS conversion by MicroMagic
Retail price - $49.95
Typical mail order price - $34
Not copy-protected
Requires 768K

Origin Systems, Inc.
P. O. Box 161750
Austin, TX 78716-1750
(512) 328-0282
BBS: (512) 328-8402

Reviewed by Bryan Walker

Years ago, college students started playing a game, generally known as "Core Wars," on university mainframes across the country. Writing programs to outsmart and erase their adversary's, these pioneers gave us the possibility of true games with the same concept, and undoubtedly some nasty virus techniques as well.

In the early 1980's, Muse Software gave Apple II users the next step: "Robotwar." In this game, players programmed "robots" to engage in arena-type battles. This game became a rage—a couple of nationwide tournaments cropping up as a result.

That's history. The future lies with MicroMagic, Origin Software, and OMEGA!

In OMEGA, the player is given the task of designing the driving intelligence and rolling chassis of monstrous weapons known as Cybertanks. MicroMagic, which converted the game to GS format, has done an absolutely outstanding job of taking advantage of the GS's mouse-driven capabilities, greatly reducing the work load on the aspiring Cybertank designer. The menu windows are even "armored," adding to the atmosphere of the ironclad.

In booting up the game, you'll be presented with the ability of password-coding your disk, to prevent the competition from discovering your latest breakthrough in brilliance. The game also does a "Retina Scan," which, while graphically interesting, loses all novelty after the first viewing. After proving you are indeed you, the program takes you to the Bulletin Board area of the Organization of Strategic Intelligence, who is your game-self's employer. Every few seconds, various memos and news briefs are shown on the board. While interesting initially, they are too repetitive and provide no help or assistance in regards to designing your first Cybertank. Move out smartly to the window options at the top of your screen.

About this time, you'll have chosen a few of the menu commands, and realized that the titanic manual enclosed with the game wasn't included to keep the printers busy. OMEGA is sophisticated from top to bottom, so breaking with tradition and reading the manual is a necessity if you want to be half as good a Cybertank Engineer as you think you are. Fortunately, the manual is well-organized, and presents everything you'll need to know to design the death-dealer of your dreams. In fact, they take you step-by-step in designing your first tank, "Gamma," which is good enough to beat the first few Cybertanks that you will face. Origin cut no corners on the manuals, even assisting you in breaking down human thought processes and converting them into the Cybertank Control Language, or CCL.

In programming the Artificial Intelligence for your tank, you type in English-like phrases to command specific actions of your project. Here's an example:

```
Start
Hunt
    Scan for enemy tank
    If enemy tank is within weapon range
        then do Attack
    If enemy tank was not found
        then do Search
    Do ChargeIt
    Resume
```

This code simply tells the tank to first look for an enemy tank. If a tank was found, and it's within your weapon's range, then do the routine labeled Attack. If no enemy tank was found, then do the routine called Search. By default, if an enemy tank beyond weapon range is found, your tank proceeds to the routine called ChargeIt. The language and interface that OMEGA employs is easier to learn than falling off a log.

For advanced Cybertank Engineers, an abbreviated version of CCL is also available, which isn't quite so verbose.

There's also a selection of "Capsule" routines available to use. These are canned routines that you can easily add into your tank's program, saving time and providing solid code to build on as well. The text-based programs can also be edited easily, using the friendly "Cut and Paste" options that are familiar to nearly every GS owner. Ease of use is this game's party line!

Using CCL, you can program your tank to act on many variables including damage to your Internal components, Armor, Scanners, Treads, and Weapon. You can also gauge decisions based on the fuel level of your tank. Any number of movement and search patterns can be employed, as well as attack tactics. You can duke it out, or play stick and move. It's up to you and your trusty keyboard. Upon completing your program, you "Authorize" the tank. This just compiles the code, and insures that you have the necessary hardware that comprises a Cybertank. Any anomalies in your design are brought to your attention at this time.

In addition to designing your tank's intelligence, you can also customize your baby with different chassis, weapons, fuel tank capacities, scanners, engines, and other special items. The chassis range from heavy, poorly-armored models to light, strong, and amphibious types. Weapons range from low-power shells to lasers and nukes (not as powerful as it sounds). Scanners vary in their ranges and scanning width. Engines go all the way from wimpy little Light drive systems to Fusion and Ion propulsion.

The Special Items are a collection of non-required goodies that can really help your tank kick butt and take names. These range from Remote Scanners that your tank can launch and get a bird's eye view of the battlefield, to Repair Kits, Electronic Shields, Fuel Misers, and whatnot. A few of these on your tank, and you'll be amazed at the difference! My favorite is the Accelerator, which doubles the computing speed of your tank.

All these wonderful items are expensive, and you only start with 1000 clams at the beginning of the game. Until you build your Security Clearance up to the OMEGA level, you'll have a limited supply of money. Fret not. By fighting your tank against OSI's tanks in the Personnel Evaluations, you can increase your Security Clearance, and the amount you can spend on your tank. As I mentioned before, the tank in the manual, "Gamma," is capable of winning the first fight or two. This should give you both the experience and money to really get going. Don't rest on your early laurels, because the competition is going to get tough quickly.

In the Personnel Evaluation, you pit your tank against the vicious OSI juggernauts Dinky, Buster, and Ducky in the first few battles. The battlefield is one of three areas: Small, Houston (big but fairly open), and Austin (big and heavily wooded). The density of obstacles can make a tremendous difference in the required programming for your Cybertank. Some obstacles can be destroyed by either ramming or shooting, while others are indestructible. Your tank has the capability of telling the difference between obstacles around it, simplifying the task of negotiating them. In the Evaluation, you must win 7 of 10 battles, so it can take some time to finish, particularly on the Austin battlefield. By turning off the graphics and sound, the process is accelerated considerably, but in the upper-level scraps, plan on doing something else while the battle rages. I recommend watching your tank for the first several engagements, so you can ensure that your creation is behaving like you expected. I **strongly** recommend designing a manual-control Cybertank to face the high-level OSI tanks Mirage and Ogre.

The graphics and sound of the battles aren't spectacular by any stretch, but they do perform their function. I would have preferred some different sounds on some of the weapons, but that's a small quibble. This is by no means an arcade game. The battle you see is frosting on the cake, with the real conflict being between the invisible lines of code.

If your tank is acting up, and you can't figure out why, there are several courses available to you. You can print a hardcopy of your program, or trot over to the testing module and actually view the program line by line as your tank participates in a test battle. This can really help stamp out those bugs.

In single-tank combat, a page or two of program can do all the tasks you wish to perform. However, it's possible to also have team competition with up to seven tanks per team! With certain Special Items and programming techniques, tanks can transmit what they find to other members of the team. Teams need not be homogenous, either. Several different types of tanks can participate in the fracas. The team battle can be decided by either wiping out the other team entirely, or by destroying their HQ building. The tactical and strategic possibilities are endless with such a situation. Team combat is what separates the men from the MS-DOS weenies.

If you're so inclined, you can develop customized battlefields using the exceptionally powerful battlefield editor in OMEGA. Using the "Tile Graphics" system that Origin is so famous for, you can really let your creativity run wild. It should be noted that some field designs can easily frustrate your Cybertank's intelligence, so it pays to be aware of your tank's capabilities before going hog wild with obstacles.

My only complaints about this game are in regards to the long Evaluation battles that you have to engage in to raise your Security Clearance up to the OMEGA level. I think going 3 or 4 of 5 is much more appealing than going 7 of 10 battles.

Another gripe is that in printing your program, you're forced to waste two pages of paper on pseudo-official foolishness. Yeah, it looks businesslike, but there's no need to waste the paper, is there? I would also like to see the ability to allocate armor thicknesses on the different sections of the hull and turret.

Additionally, watch your disk usage carefully, as I occasionally encountered some "weirdness" when programming and saving the tanks to disk.

In summary, OMEGA isn't a game for everybody. If you want joystick-bending action, forget it. While it is possible to program a tank that you can command from your keyboard during battle, there are no arcade thrills in OMEGA. This is a thinker's game, requiring patience, logic, and the ability to change your tank's programming to suit the scenario. Anyone who harbors Arthurian fantasies about designing the Ultimate Cybertank will be disappointed, for in single combat, nearly every programming technique will have a counter (sort of like "Scissors, Paper, Stone"). Team combat offers the greatest challenge to the person with the fortitude to try it.

If you want one of the most intellectually challenging GS games out, then OMEGA will be one of the best investments you can make. The price is right, the challenge continuous. I've enjoyed this game like few others. Buy it, and demand more from MicroMagic and Origin!

Note: I should mention that Origin also sponsors tournaments, allowing players to upload their designs directly into Origin's BBS. OMEGA includes further details and instructions on how to join in. I'm counting on all ***GS+*** readers to trample the other computers underfoot in the electronic arena!

2088: THE CRYLLAN MISSION

By Vinay, Vivek, and Vijay Pai
Retail price - $69.95
Typical mail order price - $39
No on-disk copy protection
Requires 1.25 MB

Victory Software
P.O. Box 821381
Houston, Texas 77282-1381
Orders: (800) 232-3828
Tech Support: (713) 493-3232
America Online: Victorysft (E-mail)
Also under Direct Connect in the Games and Entertainment Forum

Reviewed by Sir Thomas More

THE GENERIC INTRODUCTION

2088: The Cryllan Mission is an adventure game designed for the IIGS that uses the desktop interface extensively, making the game exceedingly easy to play. All of the available options and commands in the game are accessed through the menu bar's pull-down items. Pointing and clicking are the only skills that you need to master to play this game. All forms of "Party maintenance" (i.e. restoring health, equipping armor and weapons, etc.) are accomplished by using the mouse. There are many keyboard commands that may also be used for faster game play. The only keyboard input required in the game is during character creation and documentation checks. Although there is no form of copy protection on the game disks themselves, you will not be able to win the game unless you can answer specific questions using the game manual.

The game is shipped on three 3.5-inch disks. After booting the game no disk swapping is required due to a unique data compression system that eliminates disk swapping during game play. The entire program shows that the developers went out of their way to simplify the actual operations of the game and allow you to concentrate on playing. The game may be installed on a hard drive. It operates under GS/OS 5.0 and requires 1.25 MB of RAM for a ROM 01 GS and will run on any standard (1 MB) ROM 03 GS. Now for...

THE GOOD STUFF!

2088 is a welcome addition to GS-specific adventure games. The game uses the desktop interface to simplify operations and offers several innovations to the adventure genre. All routine operations have been simplified to allow you to enjoy the game. This game takes a while to boot (about 5 minutes) if you are using a standard GS without a TransWarp or hard drive, although once the game is started you are not required to swap disks again. The manual is very well-written and extremely detailed as to game functions. It is an excellent addition to the game having been designed with the novice user in mind. The manual explains every aspect and function in the game in easy-to-understand terms.

CREATING CHARACTERS

The game starts out at the Kinnar Space Academy where character creation takes place. Character classes are: Soldier - your basic Grunt who specializes in weapons and combat, Science Officer - this person operates various scanning devices that are necessary to move around in the game, Nurse - this member can heal wounds (recover partial body status points) or restore health (full recovery of body status points), Doctor - this specialist can heal wounds, restore health and resurrect dead party members (a most useful skill). As is to be expected, the specialist positions (Science Officer, Nurse, and Doctor) require more experience points to advance in levels. Each character class has different attributes that determine how effective the character is at his job and in combat. Each character has Marksmanship (hitting your target), Intelligence (healing and science skills), Kinetics (dodging enemy fire), Dexterity (tossing plasma grenades), and Stamina (absorbing hits from enemy fire) attributes that affect his performance. Once you have created your character and assigned him a class, he undergoes training and his abilities scores are changed to reflect this. In other words, Soldiers gain more body status points, better marksmanship skill, and gain dexterity (which increases the accuracy of their grenade throwing skills). Other classes have their skills increased to reflect their specialty. A word of caution here: you may lose some combat attributes on your specialty classes to make up for their specialty bonuses. Most of these losses are very minor, though.

After assembling your team (of up to six members) you are sent to the peaceful planet of Crylla to investigate the disappearance of the crew of the USS Houston. One of the first things that you will discover is that the planet is not as peaceful as you were led to believe. Your adventure starts as you battle to remain alive on a foreign world.

Movement on Crylla is easy. Your team is represented by an icon which is guided around the viewing window with the mouse. Other icons represent groups of "monsters" (which desire to make your acquaintance and deprive you of the use of your life) and entrances to towns, caverns, and buildings. To get from point A to point B in the viewing window, you simply click th mouse on point B (you can also use the numeric keypad). A logical movement routine will automatically guide your team to the destination while avoiding obstacles. Transports may be purchased (after you have acquired the necessary papers) that allow you to move at a faster rate and over greater distances. They also add to your firepower in combat and protect the player that remains inside to operate the gun. Both land and water transports are available in varying degrees of power and protection. A great feature is the fact that transports can never be stolen in the game.

Combat uses an Ultima style tactical formation with significant improvements. Each combat brings up a tactical window in addition to the regular viewing and message windows. The viewing window contains icons of each of your players in a prearranged combat formation. Icons representing your enemies are located in their formation at the top of the viewing window. (I recommend that you disperse your players as soon as they are equipped with long range weapons.) The tactical display window outlines the same thing as the viewing window but each character is assigned an alphanumeric indicator

(P0–5 for the party and M0-whatever for the monsters). A nifty feature that allows you to scan your opponent's body status, weapons, and armor rating can be used in between combat rounds. Another point that should be noted is that you can heal players, re-equip armor or weapons, change weapons, and check your player status between combat rounds. In other words, as long as you have the proper medical supplies on hand, you can restore your characters' health in between combat rounds! If you closely monitor your characters they should never die. Of course, you will use a lot of medical supplies to keep them going. If a character is killed, then a Doctor may resurrect them and they will be returned to full health in a few days. They can still function, but they have reduced attributes until they are fully recovered. If the entire party buys the farm, they will be resurrected, but missing almost all of their weapons and armor.

Weapons can be long range, short range, or plasma grenades (area effect). You can designate which of your characters' weapons will be readied and, through the use of the combat preferences menu, actually designate various combat strategies for the computer to follow. You can manually control the combat or let the computer do it for you (unlike many other games the computer actually does a pretty good job directing combat). You can target one monster for the whole group to fire at or disperse your fire among different targets. You may designate rules for the computer to follow and determine which team member will lead the party. Party leaders get more experience after the combat. To designate a target you select a party member by clicking on him and then drag the cursor to the target that you want him to fire at. You can also throw grenades at groups of targets to inflict damage on several targets at once (one of my favorite techniques). It should be noted that the computer does not do a good job throwing plasma grenades, it only throws them on positions occupied by a monster. When you are using grenades you can often engage more targets by hitting the spaces in between the monsters rather than a specific monster. Each grenade bursts over a three square area. After combat, the combat spoils appear in the message window. Level advancement is automatic for characters.

One of the really nice features is that the random encounters get tougher as the game progresses. Unlike other games, as your characters get stronger, so do the wandering monsters. Remember in Bard's Tale how after you got through kicking Mangar's butt you felt foolish having to deal with a random encounter like three wolves or two skeletons? In 2088 the monsters improve as you do. They get stronger with better weapons and even transports to match your own! This is a great feature that keeps you on your toes throughout the game.

The Documentation checks are cleverly designed to enhance game play and are not a bother at all. I like the way Victory Software has incorporated them into the story. I won't elaborate on how it was done (to help protect the game) but feel that it is a unique and agreeable way to handle the situation. Victory Software has created a great game. Let's reciprocate their efforts by making only backup copies and deterring piracy.

Two of the features that I thought were unique and significant improvements are the talking and camping routines. The talking routine uses a pop-up window and buttons to really add detail to the game. You can receive some background information or an introduction on the person by clicking on the appropriate box. The person can continue to give you more detail on a subject or you can change to a new topic by clicking on either of those option buttons. Conversations are one-sided (the NPC, or Non-Player Character—one that you have no control over—speaks, you listen) but very informative and can be extremely detailed. The whole system tends to add an atmosphere of reality to the game. The camping routine takes into consideration one of the features that I've always felt were lacking in adventure games: security. Camping is a cheap and easy way to recover lost body status points. However, the first time you camp in each game, you must decide how many people are going to stand guard and during what shift. Failure to set adequate security can result in the party being ambushed or having goods and money pilfered. You can also vary the amount of time you can rest. There is also a nap option that does a similar feature, but only for a few hours. After the first time that you set this security setup feature, it will remain in place as the default each time that you camp. You can always change it if necessary. Both of these features add realism to the game play.

NOW FOR THE IMPROVEMENTS

There is a lot more to the game than I have mentioned here that makes this a great game. The desktop interface is one of the major strengths of this game. It really simplifies game play. It also allows you to rearrange the viewing screen and check or change the status of your characters quickly and easily. It sets this game light years ahead of the standard DOS 3.3, 5.25-inch disk games that are the staple of the Apple adventure game market. But every game has its faults. To Victory Software's credit, most of my complaints are just minor inconveniences that do not really detract from the game itself.

The dungeon graphics are not really that great. In fairness I must say that I use Bard's Tale IIGS as the standard that I measure them against. I spoke with Victory Software and they mentioned that although the graphics are in the 640 mode they used a dithering effect to have 16 colors available on the screen. I believe that they could be improved upon. Another feature that I would like to see added would be to include more keyboard equivalents for two of the most used game functions: Lifeform and Terrain scans. Others could also be added, but I feel that these two in particular are used so often that it should have been included. Being able to sell used transports should also be included, as well as being able to sell the special weapons that you acquire later in the game. One annoying little feature is the toggling of the active window in the weapons shop. After choosing the talk option (which presents the pop-up windows for the weapon shop), you get the standard weapons shop window, and the message window informs you of the type of weapons for sale. At this point the message window becomes the active window (for no particular reason) and you must click on the shop window before you can transact business. In speaking to Victory Software, I was told that the whole thing is an effort to remind the player of the ranges of the type of weapons that are available. It is a minor irritation that you must do this throughout the entire game.

Another feature that I would like to see is a drop command. Although there are no encumbrance limits, you cannot sell used armor. As a result, you can't get rid of it at all. Admittedly, this is a minor point, but it would clear up the Group statistics window. I would like to see a rough map of Crylla included somewhere (perhaps an auto-map feature similar to the one in the dungeon) to facilitate navigation on the outside. It is frustrating to spend 30 minutes to an hour looking for a specific town's location that escapes your memory. The lifeform scans only help so much. An auto-reequip feature for plasma grenades would take a lot of drudgery out of party maintenance. Another minor fault occurs any time that you restart the game. The leadership position is automatically assigned to your first player. I like to rotate the leadership bonus among my specialty characters to make up for their slower rate of advancement. Unfortunately, they are not my first character. I admit that this is a petty gripe, but it could easily be addressed. Once again, Victory Software has stated that these problems have already been corrected in their next game.

My biggest and final gripe is what I feel is a major letdown in the final confrontation to win the game. This game is primarily a hack-and-slash game with a good, but not too challenging, story line. It is designed around the combat features. The final battle lacks the fear and intensity that should have been developed. Once again, you enter the final fortress all pumped up and ready for universal Armageddon combats, only to discover... well, I'll let you find that out for yourself. I just think that it could have been handled better.

SO WHAT'S THE BOTTOM LINE?

I feel that 2088: The Cryllan Mission is an excellent game that breaks new ground and offers significant improvements for the genre. I really enjoyed the hell out of this game. Although the game has a few very minor faults, it is well worth playing. All in all, it is an exciting new addition to any GS library that I wholeheartedly recommend. Technical support is outstanding. I posted many questions in the Victory Software area on America Online and received prompt and informative replies. I did not tell them that I was reviewing it for a magazine. Their quick responses are far better than most other direct-connect areas online. Their telephone support is also outstanding (although the number listed in their manual has been changed—the correct number is listed for you at the beginning of this review). I can't wait for their next game product. This is a great game.

THE HUNT FOR RED OCTOBER

By Todd Daugherty & Alex Villagram
Retail price - $49.95
Typical mail order price - $22
Copy-protected
Requires 768K

The Software Toolworks
19808 Nordhoff Place
Chatsworth, CA 91311
(818) 885-9000

Reviewed by Admiral Lord Nelson

You've read the book. You've seen the movie. Now play the computer game. Well, wait—before you do that, read this review. The Hunt for Red October is a strategic simulation of modern submarine combat. Please notice that I don't consider this to be a game. Most games can be won. This is one of the most frustrating "games" that I have played in a long time. Trying to win the game is like trying to defend Charles Manson in a court of law.

Your mission is to take the newest and most technologically advanced Nuclear Attack Submarine, the Red October, on sea trials to test out its revolutionary (pardon the pun) propulsion device. This device, known as the caterpillar, allows your sub to travel with extreme stealth. (In submarine warfare: if you can be heard, you can be sunk—it pays to move quietly.) You, the Captain, have different plans. You intend to defect to the West and turn over the Red October in the process. However you must not tell your crew; otherwise, you might have a mutiny aboard. As if that were not enough the West is not sure if you are for real or if it's a trick. In other words, you will have the entire Soviet, NATO, and US Atlantic fleets looking for your sub. The Russians will be gunning for you while the West will be waiting with loaded guns just in case. It sounds like a perfect way to spend a month at sea—dodging three navies.

Red October doesn't wait long to get you into action. From the moment you start you are in contact with a Soviet trawler. You must traverse the Reykjanes Ridge (an underwater topographical nightmare) and avoid detection from the Soviet Navy. You are positioned off the East coast of Iceland and frankly, you'll be danged lucky just to get out of there alive. To be honest with you, I've only made it past this stage of the game once (I was immediately sunk before I could save my game). I've tried about 50 times. Therein lies the major fault of this game. Although it has *many* good things going for it, it's *extremely* difficult to win. All of the game's outstanding attributes are overshadowed by that fact. It's really a shame because Hunt for Red October has some really good features.

THE GOOD STUFF

First of all, the graphics in Hunt for Red October are very good. Throughout the game they are consistently excellent. The periscope views are better than Silent Service and leave Sub Battle in the dry dock. The maps that are used are very good but they should have a zoom feature for the Atlantic map. One of the innovations is a sonar contour map of the ocean floor. This allows you to literally "hug the bottom" when you are trying to avoid enemy contact. Since you are constantly trying to avoid contact, you will use this map extensively. The map shows elevation differences by using different shades of the color blue. The darker the shade, the deeper the ocean. You should avoid the light shades as much as possible. If you stay in shallow water then airplanes can drop torpedoes on you or sonar buoys that will give away your position. Ain't life grand?

The game also comes with a copy of the book Hunt for Red October. In case you haven't read it, or if it has been a while since you have, go ahead and read the book. A ship recognition chart is also provided for you (I assume it's for when you make contact with the Americans—I rarely used it).

All facets of the game are controlled by the mouse. The High Level Command Screen is broken into four parts. The largest section is the Main Display Window. This displays your Maps (Contour, Sonar, and Terrain), periscope views, and Ship Recognition Charts. On the port (left) side, the Main Control Panel gives you quick access to the ship's controls. Want to increase depth? Then just click on the depth gauge at the approximate height that you would like. All of the sub controls are as simple as that. You can adjust the Red October's speed, heading, depth—and you can even control the speed of the game—in this area. For the extremely detailed personalities, there is a section (the Order Execution Panel) that allows you to control the speed to tenths of a knot (the nautical equivalent of miles per hour is knots per hour) and depth to individual meters. The next section located on the keel (bottom) of the screen is the Message Window. Actually, the Message Window pulls double duty. It also serves as the Sonar Information and Fire Control System Window. When it is in the Sonar/Fire Control mode, it will display the sonar contact statistics such as Bearing (the direction between you and the enemy vessel), Range (in nautical miles), Depth of the enemy (only if it is a sub), Heading (the direction the enemy is going), and Speed (never figured out this one—just kidding). In its native mode, the Message Window confirms your orders and alerts you to sonar contacts, among other things. You should constantly click back and forth between these windows during combat.

The last portion is located on the starboard (right) side of the screen and consists of several icons. These icons allow you to control the different functions of the Sonar, Engines, Weapons, Periscope, and Maps. When you click on the appropriate icon, several choices appear. The Sonar icon lets you access the Contour/Sonar display, activate the Sonar signal (which lets you lock on a target more accurately, but also helps the enemy pinpoint your position), display a side or forward view of the ocean terrain (the Contour display shows you the elevation from directly above the feature, the Terrain Map allows you to see it in relation to your sub), and the Hydrophonics. Hydrophonics lets you compare the noise "signatures" of the enemy with a list of known signals, thereby letting you identify the type and class of the enemy vessel. The Engines icon allows you to choose between Nuclear (great speed, but you lose it in the game as part of your defection scheme), Diesel (noisy, but reliable, and necessary after losing the Nuke drive), Caterpillar (very quiet, but slow), and Propeller (fast, noisy, and guaranteed to attract bad guys) propulsion. The Weapons icon allows you to switch torpedo tubes (fore and aft) while displaying the number left, manually override the Fire Control settings for the torpedo, fire torpedoes (and make a lot of noise in the process), and lay flak (electronic decoys). You only have four charges of flak so use them sparingly. The Scope icon allows you to search, target, use night vision, and intercept radio messages with the Electronic Surveillance Monitor (E.S.M.). The last icon toggles between the Atlantic Ocean map and the Contour Map in the Main Display Window. All of these icons allow a simple and easy-to-use system of playing. Hunt for Red October does a great job in making a complicated, technological submarine easy to handle. It even has three difficulty options for you to choose from.

NOW FOR THE BAD PART

I really want to recommend this game. I like the improvements made to the submarine game market. I really want to like this game. But I can't. Even with its great new achievements, Hunt for Red October fails on one major point: it is simply too hard to win. Even on the lowest difficulty option, I failed miserably. The longest session that I played (that finally got me close to the Atlantic Ocean before I was sunk) lasted about 40 minutes. Quite simply, unless you like frustration, don't get this game. I don't claim to be a super strategist or game master. I've talked to people that have actually won this game and they agree that it is incredibly difficult. They suggested that I reread the book. I did. I still failed. If you really want a challenging game, try this one. If you don't want to be constantly frustrated, don't buy it.

Some of the lesser deficiencies of the game must also be noted. It is copy-protected. It cannot be run from a hard drive due to a data compression scheme used in the program. There is no sound whatsoever in the game. This is really a pity, because this game has a lot of potential to utilize sounds. Silent Service is a much more involving game due to the wide variety of sounds that it employs. When you hear the dive horn and the surf of the waves then you can't help but be impressed. I hated firing torpedoes and not hearing the audio confirmation that I was used to. Red October is graphically outstanding but sound poor. The game does allow you to save a game in progress, but you must do so on a separate disk. It also does not support a two-drive system.

The manual was written for every popular computer *but* the GS. It goes into explicit detail on each computer's boot method and for the IBM it even tells how to install it on a hard drive. The manual explains the different commands and how each computer executes them. It details the operation of the game quite well. At no point does it mention the GS. An addendum sheet briefly outlines the Apple-specific commands, but it is clear that this game was designed with other computers in mind.

The last nagging thing about the game is the quit routine. When a game ends (a newspaper appears with an article about the Soviets sinking one of their own subs) and you select either "Quit" or "Load new (or saved) Game", you are presented with a pop-up window that asks you if you want to abort the current game. This is incredibly stupid because there is no longer any game to abort (you lost). You can't even access any of the game functions when this appears. So why does it ask it in the first place?

In summary, although the game makes new contributions to the submarine game market, I can't recommend that you spend $49.95 to be extremely frustrated. If you want a strategic challenge that will take incredible patience to meet, then this is the game for you. I recommend that if you are looking for the best submarine game that takes advantage of the GS's capabilities, hunt for Silent Service instead.

REVOLUTION '76

By Edward Bever, Ph.D.
Retail Price - $49.95
No on-disk copy protection
Requires 1.25 MB

Britannica Software
345 Fourth Street
San Francisco, CA 94017
(800) 572-2272
(415) 546-1866

Reviewed by Joshua Chamberlain

Revolution '76 is a historical simulation of the Revolutionary War. It is also a game that challenges your mind and political skills in your attempt to create a unified nation out of 13 colonies that encompass widely varying resources and political thoughts. This is, without a doubt, one of the most challenging simulations/games on the market. To me, it is also one of the most addicting games in that my pride will not let me stop playing this game until I beat the British. It's a lot easier to say than do.

Revolution '76 comes on three 3.5-inch disks and can be installed on a hard drive. The disks are not copy-protected, but you will be prompted for a password before you can play the game. The game comes with a detailed Player's Reference Manual and a Historical Perspective Manual. The password will come from the Player's Reference or Historical Reference Manual so you will have to keep those handy when you want to begin. Both of the manuals are exceptionally well-written and very helpful. I highly recommend that you read the Historical Reference Manual before you play the game. It offers many valuable insights into how the game takes various historical factors into account during game play. It also contains mini-biographies of all of the major generals and politicians for both sides. The Player's Reference Manual complements the Historical Reference Manual quite nicely and has many helpful tips for playing and succeeding at the game.

What could be so hard about winning the Revolution? All you should have to do is put your best people in your best positions and then sally forth and destroy the Redcoats, right? WRONG! Revolution '76 accurately reflects the workings of the Continental Congress and the attitudes of the various regions of the Colonies. If you could put your best people in the right positions the game would be much easier. It would also be inaccurate. The game is divided into turns that represent one year. Each turn is broken down into various phases. The first order of business is to organize Congress and assign Congressional leaders. Each member of Congress has ratings on their experience, negotiation, and administration. They are also identified by their home region and their political faction. Each position of responsibility in Congress (Secretary of War, Navy, Finance, and Foreign Affairs) calls for strong administrative or negotiating skills. However, each region of the colonies (New England, the South, and the Mid-Atlantic states) must have adequate representation. If you alienate one of the regions by not giving them enough participation in the government, you will see Patriotism in that region decrease and Tory (English sympathizer) activity increase. As if that wasn't enough to balance out, you must also maintain a balance between Radicals and Moderates in the Congress. Unfortunately, you are forced to put some of your less-talented members into position of great responsibility in order to maintain these regional and factional balances. Later in the game you will assign members of Congress to become ambassadors to European nations. You have to find the right balance between keeping things at home going smoothly and gaining the all-important European support for the Revolution.

Once your government is established, you must administer to the needs of your revolution. You must decide how to treat the Tories in each region, how much to tax each region, how large their militias are to be, and the status of privateers in each region. (In the 18th century during times of war, countries with small navies would hire out sea Captains to plunder the enemies merchant shipping. In return for their services they were allowed to keep a part of the booty. Naturally, these actions would give the enemy navy fits. In the formal rules of 18th Century naval warfare, it is basically legal piracy.) Then you will supply your Army and build your Navy. Supply in the Army directly affects their combat performance and morale. However, you are also in charge of the economy, so sometimes you will have to cut down on Army supplies to keep expenses down.

Whenever Combat starts, you will assign Generals to units (maintaining the same regional balance as before) based upon their ratings. Combat is accomplished by moving your Regulars (Colonial Soldiers as opposed to state militias) to the appropriate threatened area. Combat is not shown. A screen stating that combat is taking place in a region will be followed by a screen giving you the results. (The few seconds between screens will be some pretty anxious moments for you while you wait for the results.) Poor battlefield performance results in lowered patriotism in the country and lower morale for the Army. Winning greatly improves both factors. Your small Navy can also boost morale significantly with victories. However, when you encourage privateers and large state militias you will see fewer regulars from that region. (Why join the Army when you can stay at home and be in the militia, or join up with some privateers and make some money?) Privateers drive up the British Merchant Fleet's insurance rates and contribute to opposition to the war in England. Somehow you must find a balance between all of these effective strategies.

Eventually, you will be forced to declare Independence and reorganize your government into a Confederation or an Executive-based nation. Each type of government must decide how to deal with issues concerning the type of Congressional representation (proportional by state or Bicameral), Taxation (Federal, or the states support the government), Western land claims (Federal ownership, or State Claims mediated by the Federal government), and the status of slaves for taxation and Congressional representation (not counted, fully counted, or partially counted). If you cannot get enough European financial support and inflation gets out of hand, you will have to reform the economy to stave off economic ruin. Throughout the game you will have to keep spending in mind and watch your inflation rates.

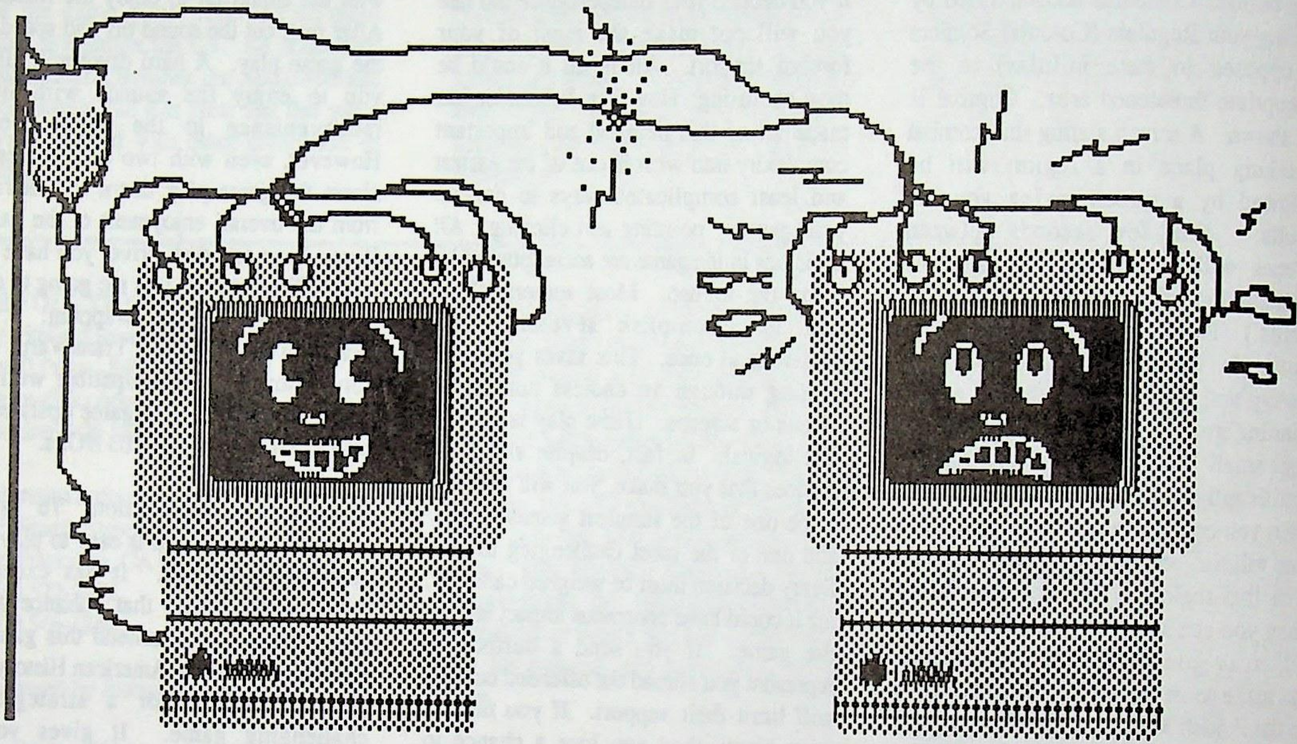
This is an incredibly detailed simulation! Each decision that you make has an impact on game play. If you alienate any section of the country you suffer a loss of Patriotism in that region. If you declare Independence too soon you will alienate the Moderates. If you create an Executive too soon you will alienate the Radicals. If you declare your Independence too late you will not make the most of your foreign support. All in all it could be most confusing. However, Britannica has made all of this detailed and important complexity into what is one of the easiest and least complicated ways to control your game: pointing and clicking. All decisions in the game are accomplished by using the mouse. Most screens allow you to accomplish several policy decisions at once. This saves you from cycling through an endless number of menus or screens. Game play is smooth and logical. In fact, despite all of the choices that you make, you will find this to be one of the simplest games to play and one of the most challenging to win. Every decision must be weighed carefully, for it could have enormous impact later in the game. If you send a buffoon to represent you abroad the offended country will limit their support. If you neglect your Navy, then you lose a chance to increase patriotism when the Army fares badly (and, therefore, lowers patriotism). In other words, there is little room for error. Fortunately, you can decide what Victory conditions you will accept/negotiate. You can request pardons (admitting defeat but allowing leaders to live), Autonomy (limited victory—you stay in the Empire), or Independence (total American victory). There is also total British victory, but you will see that enough when you first begin playing.

The game also takes full advantage of the IIGS's graphics and sounds capabilities. From the boot screen to the battlefield, you will be impressed by the excellence of both. I recommend that you play it once with the sound on to enjoy the features. After that, cut the sound off and speed up the game play. A hard drive will allow you to enjoy the sounds with little inconvenience to the game speed. However, even with two disk drives, it slows the game play down and detracts from the overall enjoyment of the game. If you have one disk drive, you have my sympathies, because you are going to do a heck of a lot of disk swapping. I was unable to test it with a TransWarp, so I don't know if it is compatible with the accelerated speed. The game operates on both ROM 01 and ROM 03 IIGSs.

In summary, Revolution '76 is an outstanding game that is easy to play, yet challenging to win. It has excellent graphics and sound that enhance game play. I highly recommend this game to anyone interested in American History and anyone looking for a strategically challenging game. It gives you a tremendous amount of respect for those men that led this nation away from British tyranny and gave the world our amazing Democratic experiment. It is an excellent way to introduce people to the American Revolution and stimulate interest in the formation of our country. Now if you'll excuse me I'm going to try to whip the British again. Now where did I place my tricorne?

PRODUCT UPDATE

After many moons, and an almost equal number of Beta versions, ORCA/C v1.1 is finally shipping! If you are a registered owner of ORCA/C v1.0 (or one of the many Beta versions) you should be getting the update in the mail soon. If you have not yet sent in your registration card, do it now! For more information, contact the Byte Works, Inc. at (505) 898-8183.



Inside This Issue Of GS+: *Transfusion* by Josef Wankerl

(Illustration by Michael J. Quinn)