

A2-CentralTM

formerly
Open-Apple

June 1989
Vol. 5, No. 5

ISSN 0885-4017

newstand price: \$2.50

photocopy charge per page: \$0.15

A journal and exchange of Apple II discoveries

Apple announces faster IIgs

The faster, more powerful IIgs that Apple users everywhere have been asking for was on display in Apple's exhibit at AppleFest Boston May 5-7. In addition to shipping the faster machines to new customers (starting sometime this summer) Apple has arranged to upgrade the machines of *all existing customers* for little or no cost.

Apple was able to provide the benefits of its faster IIgs to existing customers by tweaking the machine's *software*, rather than its *hardware*. The new software is called *Apple IIgs System Software 5.0*. It improves the performance of desktop-based IIgs applications as much or more than faster hardware could.

Hardware improvements have the potential to improve the speed of the IIgs by a factor of 2 or 3, as accelerators have done for the IIe and IIc. Software improvements, on the other hand, can increase the speed of some operations 5 to 15 times or more, as we saw the *Beagle Compiler* do to Applesoft string manipulations. Software improvements don't improve everything across-the-board, as a faster microprocessor would. But by concentrating on bottlenecks, software improvements can easily do more to increase the apparent speed and power of a computer than a faster microprocessor. Software improvements have the additional advantage of improving the performance of existing machines—not just new ones.

System Software 5.0 adds patches to much of the Apple IIgs Toolbox to provide more speed. In some cases applications will have to be slightly modified to take full advantage of the increases, but other improvements apply to existing applications. In the best case, the speed at which the IIgs draws on the super-high-res screen appears to be several times faster than before.

A feature called *ExpressLoad* offers dramatic improvements in program loading speed. Developers can convert most existing applications into *ExpressLoad* applications with a utility program. The utility adds some information to files that allows the system 5.0 loader to deal with them much more quickly than the system 4.0 loader could. (*ExpressLoad* files are standard OMF files with an additional static, private segment. Thus there is complete backwards compatibility: the system 4.0 loader can handle *ExpressLoad* files, albeit as slowly as before. Applications that use *ExpressLoad* should load their segments by name, however, not by number; the *ExpressLoad* segment is added at the front of the OMF file and changes the numbering of any other segments in the file.)

A new Apple SCSI Manager and a new Apple 3.5 driver provide quicker access to storage devices. The Apple SCSI manager is completely new, works only with Apple's SCSI card (it bypasses the ROM on the card and deals directly with the card's hardware where possible) and increases throughput from 18K per second to 80K per second. The new Apple 3.5 driver includes a feature called "scatter read". On most multiblock reads of five blocks or more, the driver reads entire disk tracks into memory in one revolution of the disk. This gives the disk an effective 1-1 interleave while reading.

According to our sources at Apple, *ExpressLoad* lowers the Apple 3.5 startup time of *AppleWorks GS* from about 4 minutes to about 1 minute. The new Apple 3.5 driver with scatter read further lowers it to about 35 seconds.

Other changes in system 5.0 allow much faster switching between ProDOS 8 and GS/OS applications. The time to get from a

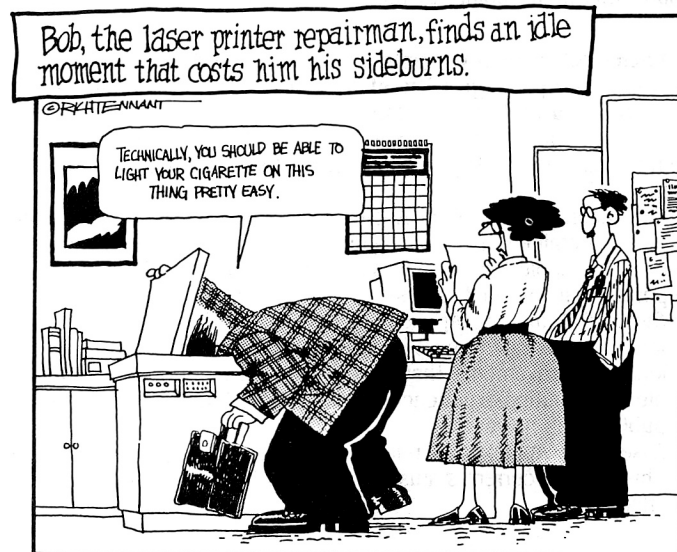
ProDOS 8 application back to the *Finder* has been reduced to about 3 seconds. This is accomplished by leaving much of GS/OS in non-purgeable memory while the ProDOS 8 application runs and vice-versa. Since most ProDOS 8 programs don't use IIgs memory, this shouldn't be noticeable, although it could impact users switching back and forth between *AppleWorks* and GS/OS applications (the *AppleWorks* desktop will be smaller than under system 4.0, but the additional switching speed is worth it).

System 5.0 includes version 1.8 of ProDOS 8, which comes with the delete bug fixed (see our January issue, page 4.89). It also includes a minor change that allows GS/OS to use lower case file-names on ProDOS volumes. Also on system 5.0 is *Basic.system 1.3*, which fixes the CHAIN bug (see "CHAIN bugs and overlays", December 1987, page 3.87), fixes BSAVE so that it chops off the end of long files when a shorter file is saved with the same name (unless the B parameter is used as part of the BSAVE—see our discussion on this in the July 1985 issue, page 1.51), and replaces the do-nothing MON command with a MTR command that enters the Monitor for you (good-bye CALL -151).

Other features of system 5.0 include a New Desk Accessory (graphics-based) control panel. The old control panel can still be used, of course, but the new one can do everything the old one does plus more. And features can be added to the new control panel simply by placing a special kind of file in a new system subdirectory called "CDEV". Among other things, the "Choose Printer" function normally found in the "File" menu on GS/OS applications and the Disk Cache NDA have been moved to the control panel.

System 5.0 supports scrolling and pop-up menus. Up to 255 items can be in a menu and any menu item can have a checkmark or key-equivalent. Pop-up menus are typically used in a dialog box that can have several settings, but only the current setting is normally displayed. The Classic Desk Accessory menu now scrolls, too, if your system has more than 512K memory.

A new parallel card driver supports a wider range of parallel cards, including Apple's Wozniak-designed original. New printer drivers (nothing new for the ImageWriter LQ, however) are faster when



enough memory is available. The memory manager is faster. A "key-board translation" feature has been added that makes it easier to add accent marks to characters and provides a standard method of doing so. The "console driver" (text screen) is faster and includes more "traps" or "hooks" for programmers. A new "notification queue" allows applications to be asked to be notified when changes such as disk switches occur, which means the *Finder* shouldn't click 3.5 drives as often as before. An "out of memory queue" has been added to allow programs to take action when the system is about to run out of memory and reduce the number of mysterious out-of-memory crashes.

The system 5.0 *Installer* now allows you to select several items to be installed at one time by holding down the shift key while you select the ones you want. It also requires a reduced number of swaps on one-drive systems. It's more important than ever that you use it, however. Just copying the files you think you need, particularly with a ProDOS 8 file utility, is asking for trouble. Some of the system 5.0 files include a "resource fork" that ProDOS 8 programs can't deal with.

The system 5.0 *Advanced Disk Utility* has been modified to allow 7 devices per interface card and 32 partitions per device. Previously it allowed just 7 partitions no matter how many devices were available.

Software developers get several new tools and features in system 5.0. Our sources say it is now easier to write many desktop programs

...it is now easier to write many desktop programs for the IIGs than for the Macintosh because the IIGs tools do more of the work.

for the IIGs than for the Macintosh because the IIGs tools do more of the work. System 5.0 also includes a new text editing tool, which essentially gives developers a built-in desktop-based full-screen editor they can use in their software, and a resource manager. The resource manager allows programmers to separate program code from program resources, such as the text and icons displayed by the program. This allows programmers to make changes in text or in the screen position of an icon without recompilation of the program, which saves a bunch of time. Other new features for programmers are calls that will start up and shut down all the tools an application needs at once and a rewritten

GS/OS-AppleShare-aware standard file selection tool that can display only files that match specified file types and aux types and that allows the user to select multiple files, not just one.

System Software 5.0 not only replaces version 4.0 for non-networked computers, it replaces system 3.2 and *AppleShare IIGs Workstation Software* for systems connected to AppleTalk networks. Under system 5.0, AppleShare support is provided through a new FST or file system translator. AppleShare file servers show up on the *Finder* desktop (and in GS/OS programming calls) just like any other storage device. In addition to being able to access the network by starting up from local disks, users may boot GS/OS over the network. This means computers can operate on the network without their own disk drives. System 5.0 includes an "@" prefix that points to the user's folder on AppleShare for use by configuration files (if you're not using AppleShare, this prefix points to the subdirectory the application resides in).

Uncharacteristically, Apple announced the Apple IIGs System Software 5.0 package without having it ready for release--without even a formal release date. This is unusual for Apple. I think it means that Apple wants to show it has heard user concerns about its commitment to the Apple II and wants to demonstrate its commitment.

The second paragraph of Apple's press release on the new system software contained the following quote from Jean-Louis Gasse, president of the Apple Products division, "The Apple II continues to be an important part of our business. Products such as *Apple IIGs System Software 5.0* and the Apple II Video Overlay Card, introduced last month, demonstrate Apple's continued commitment to providing evolutionary improvements that add functionality and richness to the Apple II computing experience." (Don't expect *MacWeek* or *MacWorld* to publish that, however.)

A separate Apple press release distributed at AppleFest reviewed the history and current status of the Apple II. It ended with a section entitled:

Apple II and Macintosh Co-existence

The Apple II product line represents an important part of Apple's business. Apple will continue to enhance the Apple II product line and offer new functionality through incremental improvements that extend the value of our Apple II customers' current investment. Apple intends to sell Apple II products as long as Apple customers want to buy them, and Apple will provide these customers with a high level of support for many years to come. The Apple II product line will continue to be a viable choice for many customers into the 1990's.

Apple's philosophy is to change the world by providing tools that enable individuals to be more creative and efficient--as well as do things never done before. The Apple II has been a powerful start in that direction. Years of experience with the Apple II have taught the company a great deal and paved the way for the Macintosh, the next generation of computers with extraordinary technological capabilities. Apple's customers are finding that both the Apple II and the Macintosh provide excellent value, and that in environments calling for multiple computers, both products work well together for complementary tasks. This co-existence is fostered by the AppleTalk Network System and many peripherals such as printers, file servers and disk drives, which are designed to be shared by both families of products.

"We recognize that our customers have different needs and requirements for the computers they choose, and we are fortunate to be able to offer them a growing range of choices," said Jean-Louis Gasse. "Just because the world now has automatic transmissions doesn't mean that we no longer have stick shifts."

While I don't believe Apple's management has made a complete turn-around in its attitude toward the Apple II, I do believe the ship is turning. I'd be more at ease if Gasse had compared the Macintosh to a luxury sedan and the Apple II to a sports car and if the press release had recognized that the Apple IIGs belongs to the same generation of computers as the Macintosh, but the seed of the idea is there and it's the Apple II community's job to nurture it--using positive response to good Apple developments and continued pressure.

Opinions as to what exactly Apple should do to keep the Apple II community (and its essential core of developers) enthused, happy, and loyal vary widely. But I'll clearly state my own opinion: the problem is *not* technology. While Apple's technological commitment to the Apple II isn't perfect (where's the ImageWriter LQ driver?), it's good. The problem is Apple's marketing commitment.

First, Apple must give the Apple II a distinct identity from the Macintosh. Apple's current positioning of the Apple II as a computer that does what Macintosh does, but not quite as well, is insane. This identity could be based on cost, color, the keyboard interface, the ability of the Apple II to run *both* traditional and graphics-based programs, or other factors. The important thing is that Apple must give the Apple II a unique market position and identity that's as distinct from the Macintosh as beer is from wine.

Second, Apple should attack potential markets with the Apple II rather than continue its retreat of the last ten years. The natural market for the Apple II is the consumer market, with strong links to both education and small business. There is no reason Apple couldn't sell 100 million Apple IIs if it would just think of it as a consumer product similar to a VCR, television, or microwave oven.

Third, Apple must sell its *own* employees on the Apple II. Apple's customers have far more respect and loyalty to the product line than most of its own employees do. Outside of a few pockets of Apple II enthusiasts in engineering, product marketing, and developer services, Apple employees know far more about Macintoshes than they know about Apple IIs. If Apple's employees had feelings for the II as positive as those they have for the Macintosh, they'd do a better job

of selling it to dealers, who, except for a few pockets of enthusiasm, reflect Apple's ill-deficient attitude perfectly.

The 1980s are the decade in which the Apple II allowed Apple Inc. to build the Apple III, the Lisa, and the Macintosh. The Apple II allowed Apple Inc. to make lots of mistakes in the process. If Apple Inc. would make just one year, 1990, the year of the Apple II—concentrating on the future of the Apple II, not its past; concentrating on giving the Apple II a unique, positive identity in the consumer, school, and small business markets; and concentrating on selling its own employees on the Apple II—the Apple II would reward it by continuing to produce a significant part of the company's revenues through the turn of the millennium.

On the other hand, if Apple continues down its present path—building the Apple II primarily for the cash it generates, without a clear sense of market position or identity; retreating from markets at the first sign of resistance; and training its employees that one of its product lines is a generation ahead of the other when, in fact, the other has caught up—other companies, be they Laser, Sony, Panasonic, or Applied Engineering, are going to step in and take the Apple II away from Apple. There just aren't many \$1 billion-a-year consumer markets as ripe for plundering as this one is.

Just touch a button and...

If I were going to give an award for Most Questionable Over-Exaltation of a new computer product in the last five years, it would have to go to Apple's aptly named *HyperCard*. Not that the product isn't important, mind you, but the sight of the entire Macintosh community falling all over itself in reaction to software that was, at heart, an advanced version of Beagle Bros' perennial worst-seller, *Frame-Up* (written by yours truly), seemed a bit overdone to me.

Now the Apple IIgs community has its own *HyperCard*-like product, *HyperStudio*, which was released by Roger Wagner Publishing at AppleFest. It is the essence of *HyperCard*, optimized for the color and sound of the Apple IIgs, without any of the hype, as befits the Apple II.

HyperStudio allows individuals to easily create and view presentations made up of text, computer graphics (still or animated), and sounds. With additional hardware, the presentations can also include television-quality video (still or movie). The ability to present a mix of these traditional media on your computer screen and speakers is one of the reasons *HyperStudio* and similar software have earned the classification, "hypermedia".

In addition, programs in the world of hypermedia must give their users the ability to access these media mixtures in multi-linear ways. Hypermedia offers many paths through a body of information, not just one. Paths are taken by pushing buttons (well, you use a mouse to guide a pointer to a sweet spot on the screen and then push the button on the mouse). Modern mythology adores the idea that there is nothing easier or better than pushing a button. That element of modern mythology comes alive in hypermedia.

At the end of an encyclopedia entry on Thomas Edison you are likely to find a note that says *see motion picture, phonograph, telegraph*. Inside an encyclopedic hypermedia presentation on Thomas Edison, on the other hand, you are likely to find *buttons* you can push to obtain additional information on these same things.

A hypermedia author uses buttons to provide users with multiple paths through his or her presentation. (Theoretically, users can also add buttons of their own to connect or associate ideas; many people writing about hypermedia consider this one of its primary characteristics. It's a powerful feature, but let's be real—it's not *magic*. The user has to find the associated information by hand and has to install the new button by hand. How often will installing the button be worth the trouble? Many enthusiastic hypermedia promoters seem to think hypermedia gives ordinary computers supernatural powers. In fact, only *UltraMacros* can do that.)

The most basic presentation element in *HyperStudio* is a

It's a powerful feature, but let's be real--it's not magic.

"card" (there's no place on a sound or video to attach a button). A card consists of a background graphic image and the author's choice of buttons, text objects, and graphic objects (still or animated). Each card can also play one sound and one video disk sequence when displayed. A set of related cards is called a "stack".

Buttons are active screen areas. You can have as many of them on a card as you can fit. You can leave each active button area blank (but why?) or you can cover it with a standard button image, a button image of your own design, or any other interesting graphic (how about a map?). *HyperStudio* buttons can bring up any other card in the stack (using one of a number of visual transition effects, such as top-to-bottom or side-to-side wipes, fades to black or white, and more). Buttons can also activate another stack or another program; they can play a sound or a video, activate an animation, or trigger an external assembly language routine you write yourself; and they can add to a score saved in a text file for testing purposes. Buttons can also be set to auto-activate after a delay you specify.

Text objects are windows with text in them. The text that appears in these windows can be typed at the keyboard (by either the author or the user of the stack) or it can be imported from ASCII text or AppleWorks word processing files (by the author). If there is more text than will fit in a window, *HyperStudio* will add a scroll bar, unless the stack author has prevented it (authors may limit some windows, such as a window for an address book).

The contents of text objects can be stored inside the stack ("embedded data"—allows faster movement between cards and allows the entire stack to be in one file) or left in separate files on a disk. If left on disk, a copy of the text can be stored either in the same sub-directory as the stack itself ("local data"—much larger stacks are possible but the whole subdirectory must be copied to transfer a stack to another disk) or the text object can just point to the original text on the disk, wherever it is ("extended data"—allows updating of information independently of *HyperStudio* and information sharing by several stacks, but at the price of speed and transferability).

Graphic objects can be created within *HyperStudio* using its built-in painting tools. Or you can use any other IIgs graphics program that uses 640-mode Super High Resolution. An animation is a set of related graphics in files with enumerated names such as BOUNCE01, BOUNCE02, BOUNCE03. *HyperStudio* can import any standard 640-mode SHR graphic. It allows you to make windows for your graphic objects any size you like and you can move any part of a graphic into a window. Like text objects, the actual graphic data can be embedded, local, or extended.

HyperStudio's painting tools can also be used to create card backgrounds. An "Add Clip-Art" function for pasting specific parts of 640-mode SHR graphics on to backgrounds is available, as is a text tool for putting text on backgrounds. Stacks are smaller if many cards share the same background—*HyperStudio* keeps just one copy of each background no matter how many cards use it.

HyperStudio also includes a "print" command for printing a card, a group of cards, or a stack; and a "find" command that will display cards that include specified text.

HyperStudio comes with a microphone, a small digitizing board that doesn't use a slot, and a separate program, called *Sound Shop*, for recording and manipulating sounds. Sounds can also be recorded from within *HyperStudio*, but you have more control over the quality of a recorded sound if you start in the *Sound Shop* and then load the resulting sound file into *HyperStudio*. *Sound Shop* and *HyperStudio* both use the sound chip in the IIgs and produce high quality sounds that could be used, for example, to teach the pronunciation of words.

Sound Shop gives you control over record and playback rates, volume, echo, and number of repeats. It also allows you to fade sounds in or out, filter out hiss, layer one sound over another, reverse a sound, and create stutter effects. It will add playback rate and volume data to sound files from other digitizing systems so that they can be more easily used by *HyperStudio*.

Stacks are made up of cards consisting of:

- backgrounds
- buttons
- text objects
- graphic objects
- sounds, and
- videos.

HyperStudio comes with four disks. The first is a GS/OS system disk that includes *Sound Studio*, *Browser* (a program for browsing sound files), and programs that make your IIGs display a graphic of your choice at boot-up, a sound of your choice at boot-up, and a sound of your choice when an error occurs. The second disk includes *HyperStudio* and a number of sample stacks. The third disk is full of sample graphics files. The fourth disk is full of sample sound files.

In working with *HyperStudio*, I quickly realized that the ability to add sound clips to cards and buttons added a whole new dimension to potential presentations. Dramatic or funny sounds can be used as part of the transition from card to card. Or sound can be used to teach music, foreign languages, or what a heart murmur sounds like.

Yet another dimension is possible by adding digitized photographs to cards. IIGs 640-mode SHR graphics aren't quite photographic quality, but they are close enough to be surprising. *HyperStudio* doesn't come with any equipment for digitizing video or scanning photographs or art work—that would easily double or triple the \$124.95 price tag of the product. But the market for IIGs digitizers and scanners will improve quickly because of *HyperStudio*.

HyperStudio has built-in commands for controlling a Pioneer 4200 videodisk.

Beyond digitized images are video images. These television-quality pictures can't be stored in a disk file, but with the addition of a video disk player and Apple's new Video Overlay Card you can get them to appear on your computer screen—either as stills or in motion. *HyperStudio* has built-in commands for controlling a Pioneer 4200 videodisk player, or compatible, through the serial port on the IIGs.

But what good is it? *HyperStudio* could be used for training, reference, entertainment, or other purposes. As a medium of communication, hypermedia excels in situations where text, graphics, or sound alone isn't enough and where it's helpful to give the user control over the order in which information is presented. Weaknesses—for the user—are that hypermedia is tied to some fairly expensive, immobile computer equipment.

Clearly, early users of *HyperStudio* will be stack authors. If you're not interested in creating stacks—only using them—there isn't much available to use at the moment. Authors who switch from other media to hypermedia will do so to be able to take advantage of its strengths, to get some leverage from its newness and sexiness, and because it offers a relatively low entry-cost that may lead to a lot of self-publication compared to other media.

On the other hand, creating useful stacks will turn out to be every bit as difficult as writing useful books—if not more so because of the necessity of dealing with several media. Stacks I've seen in the past for the Macintosh have generally been too shallow to be worth using more than once. For example, Apple has a *HyperCard* system set up in its company library that shows general information such as what part of the library magazines are in, but it doesn't include anything as useful as a list of the magazines actually in the library's collection, what issues are available, or what the call numbers are. A good stack is hard to find.

Since Apple includes *HyperCard* in the box with every Macintosh, every Macintosh is a potential *HyperCard* machine. A significant problem for *HyperStudio* stack authors is that their market is limited by the sale of *HyperStudio* itself. To get around this, Roger Wagner Publishing should develop a "run-time" or "display module" for *HyperStudio* that allows stacks to be displayed but not created or edited (no menu bar). This could be licensed to stack authors, who could include it with their products. This would open up the entire universe of Apple IIGs machines to "StudioWare".

One important note. The version of *HyperStudio* released at AppleFest is not the final version and it doesn't include a few of the features I've mentioned here (nor does it include a few features it's supposed to have that I haven't mentioned). It does come with the promise of two free updates between now and the end of the year, however, and it certainly has enough working features for those who want to get started creating stacks. Because of the programmable nature of *HyperStudio*, we've decided to carry it in our catalog at \$99 for those of you who are interested in getting a copy. And Uncle DOS is interested in any observations or questions about the product that those of you using *HyperStudio* come up with.

Miscellanea

Laser Computer announced a one-year warranty on all its products sold to consumers as of May 1. "This gives us a tremendous edge in the Apple II market and shows the industry that Laser Computer is serious about standing behind the products we sell," said Frank Pollard, Customer Support Manager of Laser Computer, Inc. Currently, all Apple products come with a only a 90-day warranty. Laser has also formed an Educational Advisory Board consisting of fourteen experts in various positions in education to identify high-quality educational software that runs on Laser computers and to help Laser define future products for educational institutions.

Claris announced its new site licensing program for AppleWorks and *AppleWorks GS* at AppleFest. Site licensing allows a school to legally duplicate an unlimited number of copies of Claris software for on-site use. The cost of the license is \$1,750 per site for either *AppleWorks* or *AppleWorks GS* or \$2,250 for both. Claris considers a "site" to be a single school, although multi-school quantity discounts are available to districts. Claris is also offering site-license discounts to schools that already own its software. An upgrade unit of 5 single copies, a 10-pack, or a network version of either *AppleWorks* or *AppleWorks GS* is worth a discount of \$500 off of the cost of the license. There are no annual fees for maintaining the license, however, there may be additional fees to upgrade to more advanced versions of a product. For more information or a contract, call Claris at 800-747-SITE or 408-987-7000. Site licenses are currently available for U.S. schools and Claris is considering making them available in Canada and Australia as well.

Further notes on the *AppleWorks GS 1.0v2* update sent to all registered users last month (see "News from Claris", page 5.26)—existing 1.0 files can be opened directly with the new version. However, data base and spreadsheet files created with 1.0v2 cannot be opened with the original *AppleWorks GS*. The update includes a 34 page manual, most of which supplements the original *AppleWorks GS* manuals.

Registered Claris users who live in the U.S. have also been receiving a 16-page newsletter on *AppleWorks* and *AppleWorks GS* every other month since the beginning of the year. It's called *Technical Solutions*. It's very nicely done and well worth a bi-monthly read. Claris has been doing a lot of things right since it got on its feet and discovered the importance of the Apple II market—*Technical Solutions* is among its best.

Another new Apple II publication is *II at Work: Apple II Productivity News*. *II at Work* is a controlled circulation newspaper. We've made arrangements for all of our current subscribers to receive at least one copy in the mail for free. Look for it and fill out the free subscription form on the back. Barney Stone, publisher of *DB Master*, is the source of the energy behind *II at Work*.

Meanwhile, the *AppleWorks Journal* from Sage Publications has quietly disappeared.

Beagle Bros announced three new products at AppleFest, *TimeOut TeleComm* (\$69.95), *TimeOut ReportWriter* (\$79.95), and *GS Font Editor* (\$49.95). In addition, new Beagle versions of two other products were officially released, *Program Writer* (\$49.95), formerly published by The Software Touch, and *Point-to-Point* (\$99.95), formerly published by Pinpoint Publishing.

(Beagle is also offering trade-ups to its products from a number of Pinpoint products at reduced prices: *KeyPlayer* to *UltraMacros*, \$30; *Spelling Checker*, *Document Checker* (or any other Apple II spelling checker) to *QuickSpell*, \$35; *Graphic Edge* to *TimeOut Graph*, \$45; *Pinpoint Desk Accessories* to *TimeOut DeskTools*, *DeskTools II*, or *PowerPack*, \$25; *Pinpoint Point-to-Point* to Beagle *Point-to-Point*, \$30; include \$3.50 shipping on any trade-up; Beagle Bros, 6215 Ferris Square, Suite 100, San Diego, CA 92121 619-452-5500).

TimeOut TeleComm and *Point-to-Point* are both telecommunications packages. *TimeOut TeleComm* works from within *AppleWorks*. *Point-to-Point* is an award-winning stand-alone program that uses an *AppleWorks*-like interface.

GS Font Editor is a ProDOS 8 program that can handle fonts up to 127 points in size. Normal, bold, italic, outline, shadow, and underline styles of the character you are working on are all displayed together during editing.

TimeOut ReportWriter gives you a relational report generator that can take information from up to 11 separate files. The files can be data base, spreadsheet, or word processor files. Only word processor files must be on the desktop when generating a report; data base and spreadsheet files can be either on the desktop or on disk. You can have as many as 250 fields in one report, which you lay-out using a word-processor-like editor. Reports can be up to 127 lines long and 240 characters wide. They automatically print to disk when a report is generated. From there you can print them to your screen or printer.

ReportWriter has a rich set of functions that can be used within fields. These include numeric range functions (Average, Count, Maximum, Minimum, and Sum), numeric non-range functions (Absolute Value, Integer, Negate, Maximum, Minimum, Page Number, Square Root), rounding functions (Round, Round Down, Round Up—you can select the number of decimal places with each of these), text functions (Capitalize First Letters, Convert to Upper Case, Convert to Lower Case, Left, Middle, Right, Concatenate, Find, Change a Number to Text), special functions (Choose, If, Match, Select Records), and date functions that can convert dates between AppleWorks format, text, and Julian numbers and that can give the current date or time. The date functions make date calculations for aging reports simple to do.

As a report is being generated, data can also be posted back into data base or spreadsheet files on the desktop. The posted data can overwrite existing data or be added to or subtracted from it. Thus, for example, you can generate a sales report that automatically posts the sales data to an inventory file, subtracting the units sold from inventory on hand.

I think **TimeOut ReportWriter** is the most significant ProDOS 8 product announced at AppleFest. It adds a number of powerful, relational data base features to AppleWorks that, particularly in combination with **TimeOut UltraMacros**, give users pretty incredible power from a point and shoot interface.

Even Newsweek has gotten on the AppleWorks band wagon. Social Studies or Economics teachers who order a semester's worth of magazines for a class (45 cents per student per week) get **NewsWorks**, a set of AppleWorks data base files filled with statistics that students can analyze and evaluate to arrive at their own conclusions. A 36-page Teacher's Guide, with lesson plans and student activity sheets that direct students in using the data base files, is included

(Newsweek Education Division, P.O. Box 414, Livingston, NJ 07039 203-646-0492).

Kyan Software is still distributing its line of Pascal products, however, the company doesn't actively market the product nor do they provide support other than a money-back guarantee. For more information write Kyan at 1850 Union Street, #183, San Francisco, CA 94123.

Iigs replacement batteries—identical to the original equipment except that they have special slide-on terminal connectors that don't require soldering—are available for \$9.35 each from Nite Owl Productions, 5734 Lamar, Mission, KS 66202 913-362-9898. Your original battery (you have to lift the power supply out of the Iigs to find where it's soldered to the motherboard) should be removed by cutting its wire leads, leaving about half an inch of each wire sticking up and still soldered to the motherboard. The new battery slides onto the old battery leads. Old batteries should be sent to Bob Shofstall at Nite Owl, who uses a special process to convert the environmentally hazardous lithium in the batteries into fish bait.

Apple gave a new CD-ROM disk to attendees of its early May developer conference. The disk is split up into three areas: ProDOS, Macintosh HFS, and music. The ProDOS section is full of just about everything Apple has provided Apple II developers in the past, including technical notes, sample source code, various versions of system software, and other stuff. The musical section includes 10 Apple songs. Apple reportedly also offered developers significant discounts on its CD-ROM players.

We've discovered the Apple International contact in Apple's Customer Service Department. She's Diana Hogue, Apple Computer, Inc., 20525 Mariani Ave, M/S 37-S, Cupertino, CA 95014.

Hometown, U.S.A. is a neat little software package that consists of graphic files of buildings you can decorate, print out, and then glue together to make them three-dimensional. The buildings are HO scale. The package includes a bakery, bank, church, drugstore, gas station, hotel, library, one- and two-story houses, several other buildings, and eight files with doors, windows, signs and store fronts you can cut and paste onto the buildings with your own graphics software. **Hometown, U.S.A.** is available for \$39.95 in either Iigs (Super High Res) or Ile/Ilc (standard resolution) versions from Publishing International, 353 West El Camino Real, Suite 222, Sunnyvale, CA 94087 408-738-4311.



Ask (or tell) Uncle DOS

Applesoft bootstrap

I have an Apple IIe and would like to learn Basic programming. Could you recommend any books or disks to start out with?

John Wheeler
Shelburn Falls, Mass.

If you can borrow a copy from a friend who has purchased a new Apple II within the last year or two, start with **A Touch of Applesoft BASIC**, which Apple now puts in the box with new computers. It's a good first-step introduction. (It's not perfect, however. Subscriber David Lyons has sent us list of 24 corrections he recommended Apple make to the book.

The most important are: page 67, line 110, GOTO 17 should be GOTO 170; page 90, the argument in the second INT example should be -273.16, not -273, 16; page 92, the description of ONERR GOTO should mention that any statements following the command on the same line are ignored, no matter what.)

If you want more after you've polished that off, get Apple's **Applesoft Tutorial** and its **Applesoft Programmer's Reference**. Tens of thousands of people have learned Applesoft on their own using these two books. The Applesoft Tutorial comes with a disk.

If you want to do much of anything with a disk drive you should also get Apple's **BASIC Programming with ProDOS** (includes a disk) or **ProDOS Inside and Out** (disk optional) or both. The latter was written by our own Dennis Doms and me. I recommend you stick to ProDOS, but if for some reason you need to learn about DOS 3.3, the obvious choice is **The DOSTalk Scrapbook** by me and Bert Kersey.

There are other, non Apple II specific, books available on programming in BASIC. Dennis recommends **BASIC and the Personal Computer** by Dwyer and Critchfield. This is a generic book, not fully Applesoft compatible, but with some practical program examples and some philosophy of programming that Apple's books don't get into. It's easy to write very hard to read programs in Applesoft. Learning how to write clearly from the beginning is a good habit

that pays off quickly when you find yourself revising a program you've written some time ago.

Now a word from our sponsor: we carry all of the above books (except for **A Touch of Applesoft BASIC**) in stock—because they are the ones we feel are useful. See our catalog for pricing.

Integer BASIC lives

I bought a disk recently at a local discount store for seven bucks that included four sports games, two of which were written in Integer BASIC. I was able to modify one program to accept single joystick (rather than dual paddle) input. But it does funny things if my computer happens to be in 80-column mode when I start the program and I can't figure out how to fix that. I tried the standard PRINT CHR\$(21) for turning off 80 columns, but that generates a ***STRING ERR message, so I assume it must be done with POKes. Do you have a tip?

Harry Barr, III
St Louis, Mo.

Integer BASIC lacks a CHR\$ command (although Bruce Tognazzini's ingenious **Infinite Number of Monkeys**, which you could get from Apple dealers a decade ago, demonstrates how to simulate one). Consequently, Integer BASIC thinks your CHR\$ is a reference to a string variable.

In Integer BASIC, strings are handled differently than in Applesoft. First, all Integer BASIC strings must be declared and DIMensioned before they are used the first time. But you can recover part of a string (like the MID\$ command in Applesoft) by just indicating the start and end points you're interested in. Thus:

```
10 DIM A$(12)      : REM A$ can be up to
20 A$="ABCDEFGHJKLM" : REM 12 bytes long
30 PRINT A$(9)      : REM prints "IJKL"
40 PRINT A$(4,6)    : REM prints "DEF"
50 END
```

In Integer BASIC, PRINT CHR\$(21) means to print the string CHR\$ beginning at the string's 21st element. But CHR\$ hasn't been DIMensioned (to say nothing of not having a 21st element) hence, *** STRING ERR.

Back in the days of Integer BASIC the primary solution to the lack of a CHR\$ function was to simply embed control characters within quotes. For example:

```
10 DIM BELL$(1) : BELL$="" : REM "control-G"
20 PRINT BELL$
30 END
```

You can't see it on the screen, but when entering line 10 you have to actually press control-G between the quotation marks. Unfortunately, this trick doesn't work with control characters that move the cursor (the Apple II firmware sees the character, moves the cursor, and doesn't pass the character on to Integer BASIC), such as control-U (forward arrow), which is also used as the character that turns off 80-columns. So here's a more brute force way to do it:

```
10 POKE 768,169
20 POKE 769,149 : REM LDA #$95 (control-U)
30 POKE 770, 76
40 POKE 771,237
50 POKE 772,253 : REM JMP $FDED (COUT)
60 CALL 768 : END
```

Universal parallel driver

Your latest issue mentions the Apple GS/OS "parallel driver" as does the previous one and indicates that various parallel cards can be used if the driver is modified to look for the correct ID byte. Maybe someone has mentioned this, but I sure could use some information on what the ID byte is for my ProGrappler card and how to modify the driver to work with it. Alternatively, is there a way to have the system poll the card and ask for the ID byte so that I can find out what it is? Thanks.

Richard Greeley
Merced, Calif.

I should have added some detail on this before now. And system disk 5.0, when it's released, will include a new parallel card driver that should solve this problem. Meanwhile, with 4.0, first install a copy of the GS/OS parallel card driver in the SYSTEM/DRIVERS folder of your System Disk. Then work your way to the Applesoft prompt and enter:

```
PREFIX /SYSTEM.DISK/SYSTEM/DRIVERS
REM use correct name for your system disk
SLOT = 1
REM use correct slot number for your card
POKE 768, PEEK 49164 + (SLOT*256)
BSAVE PARALLEL.CARD,T$BB,A768,L1,B$2FD
RENAME PARALLEL.CARD,MY.SLOT.1.CARD
REM use correct slot number for your card
```

After making the patch and renaming the file (we don't want a bunch of patched PARALLEL.CARD files roaming around posing as the real thing), reboot and run a GS/OS application. Pull down the file menu and select "Choose Printer". The "My slot 1 card" selection should now appear in the interface cards list. We'd be interested in hearing what type of card and what results you have when you try this.

AppleLink, Pers. vs Indust.

What is AppleLink Industrial Edition? How does it differ from AppleLink Personal Edition? To whom is it restricted?

Gary McLaughlin
Freeland, Mich.

In the beginning was **AppleLink**. It is Apple's internal electronic mail and messaging system. Apple has allowed its vendors, dealers, developers, national accounts, user groups, and significant others access to this system.

Next came **AppleLink Personal Edition**, Apple's online information service for consumers. After introducing **Personal Edition**, Apple started calling its internal system **AppleLink Apple Edition**. We think "Industrial Edition" is a much more meaningful name, however, and that's what we've decided to use to distinguish the two.

AppleLink PE mailbox

How about getting an email address on AppleLink Personal Edition? I found GEnie, the time I tried it, not easy to navigate, whereas I find AppleLink very easy to navigate. That way some of the rest of us can reach you in a timely manner.

Stephen Bach
Scottsville, Va.

There's no doubt that in the beginning **AppleLink Personal Edition** is easier to navigate than more traditional online services. But we've put a lot of effort into providing Apple II-specific help files on GEnie that have worked well at getting our subscribers going without much effort. After a few weeks on GEnie you can be using fast, automated retrieval that gets you lots of interesting information at the expense of relatively little time online. After a few weeks on AppleLink, however, you're still opening windows one-by-one. I think you'll eventually find it both tedious and unnecessarily online-time consuming.

While GEnie is where we're putting all of our online effort, we do maintain a low profile on a few other services. You can reach us on **AppleLink** by sending email to "DennisDoms" or on **CompuServe** by sending email to account 70120,202.

AppleLink and Lasers

I have a two-year-old Laser 128 with ROM v3.0. I have been very happy with this computer, which has in all respects performed up to or better than specifications. I use it with a Laser peripheral box, containing a 1-meg RamFactor and a Transend Modemcard. The modem is Hayes Micromodem compatible.

I had hoped to run **AppleLink Personal Edition** on my computer, but it is one of the very few Apple II programs that doesn't seem compatible with the Laser 128. Using AppleLink software I can get as far as dialing out to the local Tymnet number, but when Tymnet

answers, the program locks and nothing further happens. AppleLink customer service says that the program won't run on a Laser 128. Before I abandon this software, I would like to know whether you know of any way that I can run the program on my computer.

Stuart L. Astor
Bayville, N.Y.

Laser technical support says the current ROM for the Laser 128 is v5.0. We've had one unconfirmed report that **AppleLink Personal Edition** will work with this ROM, however, Laser itself wasn't sure. Laser was sure that the **AppleLink** software was incompatible with its previous v4.5 ROM.

ROM upgrades for the Laser 128 are available for \$25 (550 E Main St, Lake Zurich, IL 60047 312-540-8911).

Apple IIs and Fax

Can you communicate with a fax machine through your Apple II? If so, how?

Ken Buesing
Dubuque, Iowa

Apple makes a device it calls a "fax modem" that can transmit and receive pages of graphics, however, the software that comes with the device is only for the Macintosh family. The combination of the fax modem and a Macintosh makes for a **very** expensive fax machine (and the fax modem doesn't even double as a regular modem).

If what you need to do is **send** fax messages, you can write the message on an Apple II word processor and send it, using a standard modem, to a service that can forward the message by fax for you. CompuServe has announced such a service (prices are 75 cents for the first 1,000 characters and 25 cents for each additional 1,000 characters).

If what you need to do is **receive** fax messages, you're probably better off with a stand-alone fax machine. The cost of dedicating even an Apple II to full-time phone-readiness would be prohibitive for most fax applications.

Ilgs PAL follow up

I have taken Peter Sandy's advice (December 1988, page 4.88) and spoken to Frank Revell at Apple Australia. Apple has come up with a PAL video output conversion kit for the Ilgs in Australia. It involves a daughterboard that is placed in the socket of the video coder chip (MC1377). This is something of a problem, since up until about six months ago, this chip was soldered to the Ilgs motherboard on units shipped to Australia.

Consequently, when you get the conversion kit, the daughterboard comes attached to a new motherboard! (The old one is sent back to Singapore to be reworked, I'm told.) Cost of the conversion is \$A40 and I expect the dealer to charge another \$A65 to fit it. I'll send another report after I actually get one.

By the way, the real reason that Apple is offering this modification is that a lot of government/education contracts are specifying that computers must have PAL video output. Apple hasn't done this just out of the goodness of its heart.

John D. Smyth
Blackburn, Vic.

JSR DOS 3.3

How is DOS 3.3 disk access performed from assembly language?

Jonathan Famous
Gilbertsville, Pa.

There are three different ways to accomplish this. One is to print ASCII strings containing the normal DOS 3.3 commands you'd use with Applesoft. The second is to fill out a table called the File Manager Parameter list and call the DOS 3.3 file manager (much like using the ProDOS machine language interface). The third is to fill out a table called the input/output control block, or IOB, and call the Read/Write Track Sector routines inside DOS 3.3.

Each of the three ways includes some tricky steps too complicated to go into here. However, there's an excellent book available (if I do say so myself) that includes chapters with complete information—including all tricks—on each of these ways of accessing DOS 3.3 from assembly language (to say nothing of many other chapters of good stuff). It's called **The DOSTalk Scrapbook**; it's by me and Bert Kersey; and it's available through our catalog. Another possibility is **Beneath Apple DOS** by Worth and Lechner if you can find a friend who has a copy; it's out of print.

File recovery via TXT

I have a good easy hint for recovering some of those funny AppleWorks files that get messed up and won't load. Change the filetype of the file to TXT and "make a new file for the word processor...from an ASCII text file" with it. I used an obscure part of Beagle Bros' AppleWorks add-on FileMaster to change the identity of my lost file from an unreadable AWP (AppleWorks word processor) file to a TXT (generic ASCII text) file. AppleWorks read the TXT file without any problem. My lost text was recovered, only needing to have little bits of formatting garbage edited out. This is easy to do using the AppleWorks "replace" function.

John J. Smith
Arlington, Mass.

We missed **Timeout FileMaster's** ability to change file types when we gave our overview of the Timeout series in December. However, you don't need it. As mentioned here in August 1987 ("Another use for AppleWorks", page 3.55) AppleWorks doesn't actually check a file's type when you tell it "make a new file from a text (ASCII) file". It can make a new file out of **anything**, including an AWP file. Just type in the name of the file you want to load. (If you load text files using the point-and-shoot enhancement included with Late Night Patches (see December 1988, page 4.84), however, you **are** limited to TXT files only.)

AppleWorks file dates

How come when I save an AppleWorks file the date and time of creation as shown in the catalog get changed? I thought it would show the original date that the file was created and that subsequent changes would have the new date and time in the date modified column of the catalog.

Jim McMahon
Southampton, Mass.

Your understanding of how creation date and modification date are supposed to work is correct. The problem is that when AppleWorks saves a file, it first saves it with the name

APPLEWORKS.TEMP. Since there is no other APPLEWORKS.TEMP file on the disk, this is a new file and it gets a creation date of today. If APPLEWORKS.TEMP is successfully saved, AppleWorks next deletes your original file and changes the name of APPLEWORKS.TEMP to match the original.

The reason for all this thrashing about is that you still have a disk copy of the original file if something bad happens while you are saving the revised version (or if you decide to cancel the save by pressing ESCAPE). It has a couple of ramifications—one of them being the creation date confusion you've mentioned, the other being that you have to have enough spare room on your disk for the entire file, not just for your modifications, when you try to save a file.

Full-time power

Explain uninterruptable power supplies, please. Gotta be costly, no?

Keith Whaley
Los Angeles, Calif.

We've been using uninterruptable power supplies around here since 1985. Our January 1986 issue has a good overview ("Reviewer's Corner", page 1.98). These things consist of a metal box with a marine battery (about the size, shape, and weight of a car battery) inside. The box has a cord that plugs into the wall and an outlet that you plug your computer into. Inside the box are electronic gizmos that keep the battery charged when everything is normal and that supply your computer with 120 volt AC power (made out of the battery's 12 volt DC power) whenever the voltage coming in from the wall starts to drop or falter.

If you have a hard disk, you should also have an uninterruptable power supply on your wish list, because if the power fails while you're writing to the disk you're likely to lose everything on it. (But then, if you have to have the uninterruptable power supply anyhow, why not skip the hard disk and just use a RAMdisk? Up until GS/OS made hard disks pretty much a required piece of equipment, we leaned pretty heavily toward the UPS/RAMdisk combination in preference to a hard disk, and said so a number of times—see, for example, "RamFactor as hard disk", December 1986, page 2.86.)

According to the latest **Computer Shopper**, a 200 watt UPS, which is large enough for an Apple II, hard disk, and monochrome monitor, can be purchased from several sources for less than \$250.

GS/OS and disk space

I own a IIgs with one 3.5 drive and one 5.25 drive. When launching an application from the new GS/OS system disk, why must I swap disks ("please insert System Disk") in the 3.5 drive? What file could I move or copy to my application disk to allow a nice one-shot start-up from the desktop?

Mark Bolduc
Albany, N.Y.

Moving files from the system disk to the application disk won't help. GS/OS expects the disk you boot from, which we refer to here as the System Disk (capital S, capital D), to be available at all times. If the System Disk is not online when GS/OS needs it, GS/OS will ask you for it, as you've discovered. There's no way around this.

The only solutions are to figure out how to

squeeze your application software onto the remaining space on the System Disk or onto a 5.25 disk for your other drive. Or, you can buy another storage device. At the present time we lean toward solving the storage problem by putting together your own SCSI hard drive—see last month's "Self-made hard drives", page 5.28 and previous articles.

Apple VAR program

I have designed a piece of automated test equipment using an Apple IIe computer. I would like to market this device as a system—computer, drives, monitor, and software all in one package.

What sort of licensing agreement would I need with Apple in order to sell the software and hardware? I can go with DOS 3.3 or ProDOS. What sort of costs are we talking about?

Roy Roesel
San Jose, Calif.

If you want to include one of Apple's operating systems (DOS 3.3, ProDOS, or GS/OS) on your disks, you have to get a license to do so. These have an annual fee of \$50 (DOS 3.3 or ProDOS) or \$75 (GS/OS). For more information and the required forms, contact:

Apple Software Licensing
Apple Computer, Inc.
10101 N. DeAnza Blvd, M/S 28-B
Cupertino, CA 95014
408-973-4667

You don't need any kind of license to sell Apple hardware. You'd probably like some way to buy it directly from Apple at a discount, however. Apple used to have a Value Added Reseller, or VAR, program for this, however, we haven't heard much about that program lately. (The minimum purchases required were far above what most start-up businesses are interested in, anyhow.) I tried to contact Apple's VAR people on industrial AppleLink to learn the program's current rules and status, but they're not answering their mail.

ProDOS 8 changes

In a relatively short period of time we have progressed from ProDOS 1.1.1 through 1.2, 1.3, 1.4 up to the current 1.7. As I was installing ProSel on my hard disk I noticed the ProDOS 1.4B that Glen Bredon uses, with an explanation that he has fixed a bug in it. I'm sure I'm not the only one of your readers who wonders just what differences have been made in the various versions and whether the latest version is really the best.

Robert Sutherland
Toronto, Ont.

Apple includes a 20-page document with ProDOS 8 when you order an update through APDA that documents the changes since version 1.1.1. Here are a few of the major ones.

Versions 1.2 and 1.3 included the following changes: initialization of an auxiliary page stack pointer; some changes in the handling of a peripheral card in slot 3 (on a IIgs, the control panel setting is consulted); interrupt status is not altered on booting (ProDOS 1.1.1 always turned interrupts off); some MLI calls (ON_LINE, OPEN, READ) were modified or fixed; an enhanced QUIT call was implemented for the IIgs; the handling of index blocks

during a DELETE was changed; a bug in the /RAM driver affecting a block read of block 7 was fixed; a bug in the ProDOS interrupt handler was fixed; unclaimed interrupts are buffered so that a single spurious (unclaimed) interrupt won't crash the system; SmartPort remapping from slot 5 to slot 2 was added so that four drives could be supported by SmartPort devices in slot 5; the Disk II driver was changed in an unsuccessful attempt to kill the ProDOS track 0 bug; support for an AppleTalk ATINIT file was added; a ligs clock driver was added; the clock driver's look-up table for the current year was updated; among other things.

Version 1.4 really fixed the ProDOS track 0 bug and got rid of a 65C02 instruction included in version 1.3 that made that version dangerous on a II-Plus or unenhanced IIe. Version 1.4 also changed the handling of the MLIAC-TIVE flag—Glen Bredon's version 1.4B changes this back so it matches previous versions.

In version 1.5 the MLIACTIVE change was removed; interrupts are disabled during a status call; a bug involving writing using too large a positioning value was eliminated; and escape is cleared from the keyboard when booting (now plainly implemented for the sake of the IIc-Plus, which you can slow down to "normal" by pressing escape after open-apple/control/reset).

No update information for versions 1.6 and 1.7 are included yet, but we understand changes were made to give better support to AppleTalk.

Sider stuff, cont.

Category 11, Topic 15

Message 1 Sat Apr 22, 1989

D. LARSEN2

at 11:17 PDT

The February **A2-Central** has a letter telling about a replacement power supplies for 10 Meg Sider I's. The 'PS-ASTEC' from JDR Micro Devices was a perfect fit electrically and physically and it only cost \$25 as compared to \$150 from First Class Peripherals.'

Unfortunately, JDR seems to have discontinued this power supply. Does anyone have any ideas for an alternate source? I have a friend who would prefer to spend less than \$150 for a power supply if possible.

Category 11, Topic 15

Message 2 Sat May 06, 1989

C. PETERSON5

at 01:45 PDT

I cannot tell a lie. I wrote that letter to **A2-Central** on the Sider power supply replacement. JDR may not call it the PS-ASTEC now, but I have their latest catalog (have them send you one), and they have a PS-37 supply (37 watts), \$24.95 + \$5.50 s&h, that looks like the one I got for my Sider. Just a new name. My old, dead, Sider power supply was 4.25" x 6.312" and delivered unknown amps at +5V and +12V. The PS-37 is 4" x 6.3" and seems to have plenty of power for the job (+5V @ 2.5amps, +12V @ 2amps).

My Sider I is still humming 4 months later. JDR has another supply, PS-194, that is 4.125" x 6.5" and delivers even more power, at \$19.95. But the PS-37 looks like what I used.

Extra notes: there is a small amount of modification necessary. I had to connect the common ground of the supply to the Sider power supply case, and I bent the case just a smidgen to make it support the new supply better, but it was all very easy. Once you see how the existing supply fits, you'll see what I mean. I removed the stand-off grounding post from the old supply case and used that to ground the new supply at a different spot.

I assume you have figured how to get the Sider case apart. If you can do that, you should have little problem with the rest. And I bought connectors to plug into the ones in the supply for about \$1.00 or so. AND, you need to know that the wires in the Sider are as follows: black is ground, red is +5V, and orange is +12V. I got that info direct from First Class Peripherals. I'm not an experienced hobby fellow, but I was able to make it all work pretty easy. And to save over \$100, it was worth it. Good luck, and let me know if you try it, and how it works out.

Cheers.

Craig Peterson

Santa Monica, Calif.

JDR Microdevices, 2233 Branham Lane, San Jose, CA 95124 408-866-6200, 800-538-5000.

Modem hacking

What is there about my Hayes Micromodem IIe (300 baud) that makes it operate only at 300 baud? Could I replace a chip or other component that would upgrade its speed to 1200 or 2400 baud? Interestingly enough, I don't feel the same way about upgrading any other part of my IIe system, but that seldom-used Micromodem board has just sat there in slot 2 for years, not really earning its keep, and I would like, if possible, to upgrade it myself to keep it up to

the state of the art, without the expense inherent in purchasing a new one. Hayes Microcomputer Products company notwithstanding, do you or your colleagues have any suggestions regarding rebuilding/upgrading the speed of a modem?

Bob Oringel
Bowie, Md.

We're not of aware of any way to upgrade it, though our magic isn't very strong when it comes to hardware. We trust that if any of our subscribers know of a way that would cost less than the price of a new internal modem or an external modem and interface card they'll let us know. There must be hundreds of these modems sitting on shelves around the country.

CDA keystrokes

I have a bit of technical information for anyone writing a Classic Desk Accessory. Classic accessories are usually written like text-based applications and they usually poll the keyboard directly for input. This can be a problem when a Classic Desk Accessory is started from an event-driven desktop application. In this case the event manager is also polling the keyboard when keyboard interrupts occur. This can create a race condition where input is received by whichever routine polls the keyboard first after a key is pressed.

If the event manager's keyboard interrupt handler polls the keyboard before the Classic Desk Accessory gets around to it, the CDA may not see the key stroke at all. This is a subtle problem because the user usually blames himself for not typing correctly and simply hits the key again; if the CDA gets the keystroke correctly on the second try the user's feeling that the keystroke was mistyped is reinforced.

Of course, it's not the user's problem at all, but a problem with the way the CDA was written. The correct thing for the CDA to do is see if the event manager is active. If it is, then keystrokes should be obtained from the event manager. This is what the Visit Monitor CDA does.

A second method is to preserve the state of keyboard interrupts, disable them during execution of the CDA, and restore them before quitting. This allows the old tried and true method of direct keyboard polling to be used while insuring that keystrokes aren't missed.

Ray Montagne
Cupertino, Calif.

Move, Pick, Store

Using the Monitor's move command (page 3.27) has its problems: one has to remember where the start, end, and destination addresses must appear in the command and, if the destination range overlaps the range to be moved, there are strange results.

Under ProDOS, an easier method of doing the move is to use the RAMdisk. Use BSAVE to save the original range to /RAM and then use BLOAD to load the range to the destination address.

Minor point: In both the PICK and STORE routines in your January 1989 issue (page 4.96), the combination of CLC and ROR (at bytes \$2008 and \$2036) can be replaced by a single LSR. This results in a saving of one byte in each routine. This may not be much, but sometimes every byte counts.

Riaz Seedat
Natal, South Africa

A2-Central™

© Copyright 1989 by
A2-Central

Most rights reserved. All programs published in **A2-Central** are public domain and may be copied and distributed without charge. Apple user groups and significant others may obtain permission to reprint articles from time to time by specific written request.

Written, edited, and published by:

Tom Weishaar

with help from:

Dennis Doms	Sally Dwyer	Dianne Plumberg
Tom Vanderpool	Steve Kelly	Joyce Hammond
Dean Esmay	Jean Weishaar	

A2-Central, titled *Open-Apple* through January, 1989—has been published monthly since January 1985. World-wide prices (in U.S. dollars; airmail delivery included at no additional charge): \$28 for 1 year; \$54 for 2 years; \$78 for 3 years. All back issues are currently available for \$2 each; bound, indexed editions of our first four volumes are \$14.95 each. Volumes end with the January issue; an index for the prior volume is included with the February issue.

The full text of each issue of **A2-Central** is available on 3.5 disks, along with a selection of the best new public domain and shareware files and programs, for \$84 a year (newsletter and disk combined). Single disks are \$10. Please send all correspondence to:

A2-Central
P.O. Box 11250

Overland Park, Kansas 66207 U.S.A.

A2-Central is sold in an unprotected format for your convenience. You are encouraged to make back-up archival copies or easy-to-read enlarged copies for your own use without charge. You may also copy **A2-Central** for distribution to others. The distribution fee is 15 cents per page per copy distributed.

WARRANTY AND LIMITATION OF LIABILITY. I warrant that most of the information in **A2-Central** is useful and correct, although drivels and mistakes are included from time to time, usually unintentionally. Unsatisfied subscribers may cancel their subscription at any time and receive a full refund of their last subscription payment. The unfiled portion of any paid subscription will be refunded even to satisfied subscribers upon request. MY LIABILITY FOR ERRORS AND OMISSIONS IS LIMITED TO THIS PUBLICATION'S PURCHASE PRICE. In no case shall I or my contributors be liable for any incidental or consequential damages, nor for ANY damages in excess of the fees paid by a subscriber.

ISSN 0885-4017
Printed in the U.S.A.

Genie mail: **A2-CENTRAL**
913-469-6502