Open-Apple™

October 1986
Vol. 2, No. 9

ISSN 0885-4017
newstand price: $2.00
per page photocopy charge: $0.15

Releasing the power to everyone.

New $999 Apple IIgs arrives; upgrade for IIe is $499

Apple announced the release of the Apple IIgs (graphics/sound), the new high-end member of the Apple II family, on September 15. The suggested retail price of the new machine, without monitor or disk drives, is $999. Apple also announced an updated version of the Apple IIc that accepts an Apple IIc Memory Expansion Card. The price of the IIc remained unchanged at $940. The price of the IIe, including 128K of RAM, was reduced 12 per cent to $829.

In addition, Apple said that it would provide, beginning in January, a $499 upgrade kit for the Apple IIe that will convert it into a IIgs. Apple also announced new monitors, disk drives, a II-family SCSI interface card for hard disks, two new versions of ProDOS, a new version of AppleWorks, and a 15 per cent price reduction for its two Macintosh computers and its LaserWriter printer, which is now priced at $4,999.

The IIgs is the machine many of us have been waiting for. The original Apple II of Wozniak is at the heart of this machine. It can boot and run DOS 3.3 programs, our old friend Applesoft is in there, and it has seven peripheral slots just like the ones on the original Apple II. Most of the peripheral cards and most of the software you already own will work and run on the IIgs.

You may not need any of your old peripheral cards, however. In addition to the seven slots, the IIgs has a back panel much like that of the IIc. In order to outfit a IIe with all the built-in stuff of the IIgs you'd have to add two Super Serial Cards, an accelerator, a clock card, a mouse card, and two disk interface cards.

At that point, though, your IIe would still be missing the 65816 microprocessor of the IIgs, the ability to add and make direct use of up to 8 megabytes of RAM and 1 megabyte of ROM, two new full-color super high-res graphics modes (a 320 x 200 mode that supports 16 colors per line and 256 colors per screen out of 4,096 possible colors and a 640 x 200 mode that supports 4 colors per line at any pixel position or 16 colors per line with limits on color positions within the line), built-in analog RGB video, the most advanced sound-generation system available on any general-purpose computer, and AppleTalk local-area-network capability.

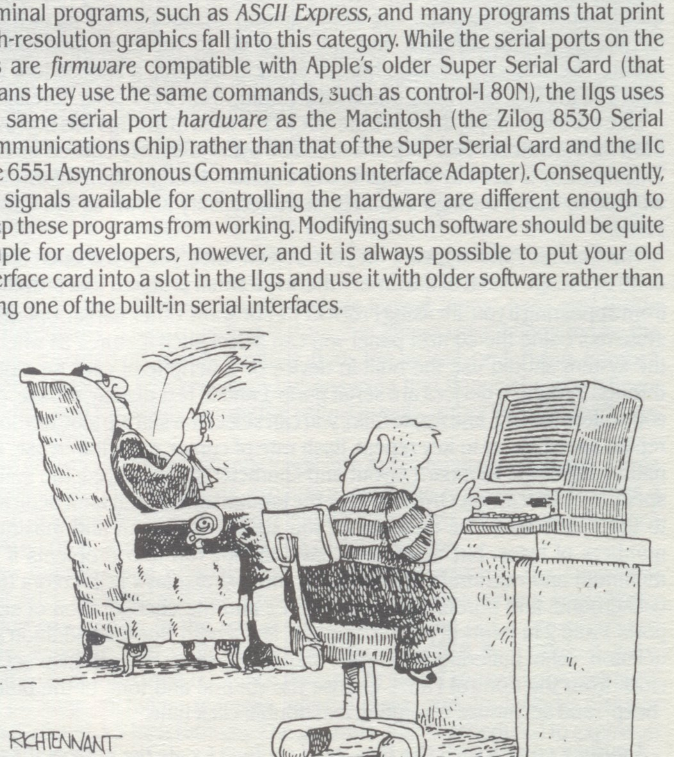
You would also be missing most of the 128K of ROM-based software built into the IIgs—the IIgs "toolbox." The tools in the toolbox include a memory manager, which should make it possible to have several different programs and desk accessories loaded into the machine at the same time and switch between them; Quickdraw II, which includes routines for drawing lines, rectangles, arcs, pixel images, and text characters on the super high-res screen; a set of sound tools, which include a MIDI-like note-control interface that is supposed to make it simple to create applications that accept input from external MIDI-equipped keyboards; a set of math tools, including Apple's IEEE standard, extended-precision, floating-point SANE package and a package of integer math routines; text tools that allow even advanced software written specifically for the IIgs to use the standard 80-column text screen; and, as could be expected from Apple, mouse-interface tools that provide a standard way for programmers to set-up and activate pull-down menus, windows, dialog boxes, and other such stuff on the super high-res screen.

Those of you eager to upgrade from an Apple IIe should be aware of a few details, however. The IIgs does not have a slot compatible with the IIe's auxiliary memory slot. Memory/RGB cards such as Applied Engineering's RamWorks and Checkmate's MultiRam don't work in the IIgs. (Slot-based memory cards such AE's RamFactor and Apple's own memory expansion card work just fine, although their RAM isn't directly addressable—it works best as a RAMdisk—just as on the IIe.) The IIgs does have a special slot for memory cards (used to expand the system up to its maximum of 8 megabytes of RAM and 1 megabyte of ROM) but that slot is different from the one on the IIe.

Next, the RGB signal created by the IIgs is an analog signal. The RGB signal used heretofore on Apple IIs is a digital signal. Consequently, that $600 RGB monitor you bought for your IIe won't work on a IIgs either, bobo.

In addition, most multi-function cards that pretend to be in several slots at once probably won't work on the IIgs. Where the IIe and IIc have two 64K banks of memory, with the $C000-$CFFF addresses of each bank mapped into a single I/O area, the IIgs can have many "banks" of memory, all with addresses in the $C000-$CFFF range, that have nothing to do with I/O. Peripheral cards must use the slot "select" signals to determine whether an address in the $C000-$CFFF range is meant for them. Multi-function cards that pretend to be in several slots at once tend not to use these select signals, however.

Another problem occurs with programs that avoid serial card firmware and instead directly manipulate serial card hardware locations. Most terminal programs, such as *ASCII Express*, and many programs that print high-resolution graphics fall into this category. While the serial ports on the IIgs are *firmware* compatible with Apple's older Super Serial Card (that means they use the same commands, such as control-I 80N), the IIgs uses the same serial port *hardware* as the Macintosh (the Zilog 8530 Serial Communications Chip) rather than that of the Super Serial Card and the IIc (the 6551 Asynchronous Communications Interface Adapter). Consequently, the signals available for controlling the hardware are different enough to keep these programs from working. Modifying such software should be quite simple for developers, however, and it is always possible to put your old interface card into a slot in the IIgs and use it with older software rather than using one of the built-in serial interfaces.



"THE KIDS AT SCHOOL SAY THAT 'gs' STANDS FOR GASSÉE/SCULLEY."

The two newly announced versions of ProDOS are called ProDOS 8 and ProDOS 16. The versions of these currently being shipped with the IIgs are preliminary. A solid version of ProDOS 8 should be released before the end of the year. The real ProDOS 16 won't be available until 1987.

ProDOS 8 will run on all members of the II family with 64K of memory and will be the current version of ProDOS for 6502-based programs when it is released. It is very similar to ProDOS 1.1.1, except for some minor bug fixes and for modifications that allow it to read the built-in clock on the IIgs. Under ProDOS 8 you will find two RAMdisks built into the IIgs—the standard ProDOS /RAM volume that uses the auxiliary 64K bank and an Apple Memory Card-like /RAM5 that uses as much of the extra memory available on the IIgs as you would like to allot to it.

ProDOS 16 will be a 65816-code kernel that will run only on the IIgs. It will be significantly different from earlier versions of ProDOS. It has no global page. It will eventually allow an unlimited number of open files and will support any number of online devices, according to current plans. (Do you hear AppleTalk talking?) Meanwhile, the "ProDOS 16" currently being shipped has the right interface and acts appropriately most of the time, but the nut of the kernel is the old 6502 ProDOS. This particular version is known to developers who have been working on IIgs software as "ProDOS 12." Like the real ProDOS 16 that is still upcoming, it accepts calls from and can read and write to memory in any of the 256 possible IIgs memory banks. It uses the IIgs memory manager intensively and does no memory management of its own.

ProDOS 16, of course, will be used only by new software that takes full advantage of the 16-bit 65816 microprocessor in the IIgs. Programs written to run on earlier IIs as well as the IIgs will use ProDOS 8 and the 8-bit 6502-mode of the 65816 (or DOS 3.3 or Apple Pascal or even CP/M).

The clock and the advanced sound system built into the IIgs are somewhat different from previous designs in that neither of them is "in a slot." As mentioned earlier, one of the primary differences between ProDOS 1.1.1 and ProDOS 8 is that the latter can read the IIgs clock. The easiest way for ProDOS 8 programmers to learn the time is to ask ProDOS. ProDOS 16, on the other hand, won't give you the time of day. ProDOS 16 programs read the IIgs clock by using one of the tools in the toolbox.

Likewise, the only way to use the advanced sound capabilities of the IIgs is through the toolbox. The sound system, incidentally, uses a digital sound synthesizer chip made by Ensoniq, a company well-known for its professional-quality music synthesizers. The sound system includes 64K of its own RAM for storing sound samples and can produce 15 voices of monophonic sound. The chip can emulate human speech, however, a speech tool isn't built into the IIgs. Let's hope one is forthcoming from third-party developers.

Like the serial chips, the real-time clock chip used in the IIgs was borrowed from the Macintosh. This chip includes 256 bytes of battery-backed up RAM. This memory is used by the IIgs to store system configuration information entered by users through a built-in desk accessory known as the Control Panel.

The Control Panel appears on your screen when you press open-apple/control/escape (most of the time—there is a small problem that prevents it from appearing if you are using ProDOS 1.1.1 or some other, earlier, operating systems). Using the control panel you can select such features as whether the system should use the built-in device or a peripheral card for slots 1 through 7 (built-in devices are serial ports 1 and 2, text display, mouse, disk controllers 1 and 2, and AppleTalk); you can select the startup slot; keyboard repeat speed, delay to key repeat, flash rate of cursor, caps/lowercase, and nationality of the keyboard layout and character set displayed; the system speed (1 MHz or an effective 2.5 MHz); the amount of RAM you want dedicated to the /RAM5 disk (the controls let you specify minimum and maximum numbers of bytes, but developers have already run into problems if the minimum and maximum aren't equal); the text display's RGB colors (text, background, and border are controllable); and the configuration of serial ports 1 and 2 in terms of printer/modem, line length, line feed addition and deletion, echo, buffering, baud rate, and data format. You can also set the clock from the Control Panel, choose the volume and tone of the built-in "beep", and set mouse resolution and double-click time.

A mouse comes with the IIgs, but I couldn't locate the person at Apple who knows for sure whether a mouse will be included in the upgrade kit for IIes; the upgrade kit definitely doesn't include a new keyboard, but uses your existing one. The IIgs keyboard, on the other hand, is detached from the main system unit—a first for the Apple II family. The mouse plugs into the keyboard, which, in turn, plugs into the system unit by means of a thin wire that ends in a telephone-like modular plug. This connection to the system unit is called the "desktop bus." Additional input devices, such as joysticks and, we hope, bar code readers, can be added to the desktop bus in a daisy chain.

Apple II Family Identification Bytes

	$FBB3	$FBC0	$FBBF	$FB1E
Apple II	$38			
Apple II-Plus	$EA			$AD
Apple III (emulation mode)	$EA			$8A
Apple IIe (original)	$06	$EA	$00	
Apple IIe (enhanced)	$06	$E0	$00 *	
Apple IIgs	$06	$E0	$00 *	
Apple IIc (original)	$06	$00	$FF	
Apple IIc (3.5 ROMs)	$06	$00	$00	
Apple IIc (version 3)	$06	$00	$03	

* to distinguish the IIgs from the enhanced IIe:

```
     SEC
     JSR $FE1F
     BCC IIgs
     BCS IIe
```

The system unit is a simple box-like affair with a power switch and lots of connectors on the back. It's meant to be stored underneath your monitor. The slots are inside, of course. Unlike most other 3-piece computers, the disk drives are separate from, rather than built into, the system unit. This box is more reminiscent of a IIe with the keyboard chopped off than of the traditional 3-piece computer. The system unit comes with 128K of ROM and 256K of RAM (plus the sound system's 64K plus the clock's 256 bytes).

The IIgs 80-key keyboard, like many of the new peripherals announced by Apple, is designed to work on either a IIgs or a Macintosh. The Apple II people finally won an internal corporate battle—the cursor-key layout matches that of the IIe and IIc rather than that of the Macintosh. There was a small price to pay for this—the solid-apple key is now called the "option" key, as on the Macintosh. The open-apple key is still there, is larger, and also has the Macintosh propeller on it. The return key is larger and L shaped; it displaced the backslash key, which moved to where solid-apple used to be. Solid-apple...er,...the option key...moved to where tilde is on IIe and IIc keyboards, tilde got a space underneath the X and C keys but the space bar had to be shortened to fit it in. Open-apple resides underneath the Z and X keys, approximately where it and the power-on light are on the IIe and IIc. On the right end of the keyboard there is an 18 key numeric keypad with 10 digit keys, four math keys, an equal key, a decimal key, enter (same as return), and clear (control-X).

Though Apple didn't do quite as well as Video Technology (makers of the Lazer 128) in designing a II-compatible keyboard layout, they whipped Franklin badly and have come up with a layout we can live with. The only key that will move around on people who are using both the IIgs and IIes and IIcs will be solid-apple/option. Some application programs, such as *MacroWorks*, *AutoWorks*, and *Pinpoint* make heavy use of this key; most programs, however, never use it.

If you aren't satisfied with the IIgs keyboard, incidentally, a company called Datadesk International has already announced what it calls "the first professional keyboard for Apple computers." It has 101 keys and plugs right into the desktop bus. (Turbo-ADB Keyboard, $149.95, Datadesk International, 7650 Haskell Ave, Suite A, Van Nuys, CA 91406 818-780-1673.)

The built-in Monitor firmware is one area of the IIgs that really shines, particularly in comparison to the Macintosh. The Monitor on the IIgs does everything previous Monitors could do, plus stuff such as searching through memory for a byte pattern up to 236 bytes long, making hex/decimal conversions, and performing 32-bit math. Screen dumps include both hex and ASCII values. The new Monitor supports all the features of the 65816, including 16-bit registers and 24-bit addresses.

The Monitor includes both a mini-assembler and a (L)ist command (disassembler) that can handle all 256 of the 65816's opcodes. The disassembler even properly expands ProDOS 8 and ProDOS 16 MLI calls, showing command numbers and parameter-list pointers on separate lines.

Under ProDOS 8, the IIgs supports Applesoft, as always, and all other available ProDOS-based languages and assemblers. The IIgs can also run DOS 3.3 and its languages and assemblers as well as Apple Pascal. It is also compatible with most CP/M cards and their languages and assemblers.

Under ProDOS 16 Apple is attempting to create a single programming environment known as the "IIgs Programmer's Workshop." Programs in the workshop package are executed from an outer "shell," which works kind of like the Applesoft prompt without Applesoft behind it. You type in "shell" commands, just as you would Applesoft or Basic.system commands, to display directories, to copy, move, delete or otherwise manipulate files and subdirectories, and to execute other workshop programs. The shell can also respond to file-based EXEC commands.

Other workshop programs are an editor, a debugger, and a set of utility programs. At the core of the workshop are Pascal and C compilers and an assembler based on *ORCA/M.* The Pascal offered by Apple is a variant of its Pascal for the Macintosh rather than a variant of the UCSD Pascal previously used in the Apple II family. Other languages are expected to be added by Apple, third parties, or both.

The dominant feature of the programming workshop is that the assembler and all workshop compilers produce object code in a standard format. A program called the workshop linker can combine these object code modules in any combination and can throw in routines from library files for good measure. The output of the linker goes into what are called "load files." Finally, a part of ProDOS 16 called the "system loader" actually executes the load files. Segments of the load file can be "dynamic," which means they will be brought into memory only when needed. Other segments can be "static," these must remain in memory while the program is running.

Details about the price and release date of the programmer's workshop are still unavailable. Early versions will very likely be obtainable through the Apple Developers and Programmers Association, however (see the September 1986 ***Open-Apple***, page 2.57.)

Apple also plans to produce a whole series of technical manuals for the IIgs—none of these are currently available in other than preliminary form. Again, being a member of the ADPA is probably the best way to obtain these manuals. Currently planned titles are *Technical Introduction to the Apple IIgs, Apple IIgs Hardware Reference, Apple IIgs Firmware Reference, Programmer's Introduction to the Apple IIgs, ProDOS 8 Reference, ProDOS 16 Reference, Apple IIgs Toolbox Reference* (two volumes), *Apple IIgs Programmer's Workshop Reference, Apple IIgs Workshop Assembler Reference, Apple IIgs Workshop C Reference,* and *Apple IIgs Workshop Pascal Reference.*

Here are a few final details about the IIgs I couldn't seem to fit in elsewhere in this article: the exterior color of the IIgs is a new Apple-wide designer color called platinum; if you put three or more cards in the IIgs system unit Apple thinks you should also add a fan, and is selling one for $49; both the 9-pin game connector from the IIe and IIc and the II-Plus/IIe internal 16-pin game connector have been included on the IIgs; the cassette plugs are gone; the first 50,000 units sold will have Steve Wozniak's "Woz" signature and the words "limited edition" stamped on the outside of the case.

All that barely scratches the surface of the IIgs. It's going to be around for a long time, however, and it will be a long time before we learn its limits, so I guess we don't have to digest the whole thing this month.

The new version of the Apple IIc is much like the previous two versions. The big difference is that the main logic board has been changed so that it will accept a memory expansion card that fits inside the IIc case. The card comes with 256K of memory and is expandable to 1 megabyte. The 256K card costs $269. A piece of good news is that if you own a previous version of the IIc and you buy the memory card, Apple will upgrade your old IIc at no additional charge.

We haven't yet heard whether the new logic board works with third-party add-ons for the IIc such as memory expansion, CP/M, and 65816 cards. It should be noted that Apple's card adds memory in the directly-inaddressable style of the Apple Memory Card rather than in the bank-switched style of third-party cards.

From the user's perspective Apple's new card looks like a RAMdisk in slot 4. One incompatibility problem with the new IIc is that slot 4 is where the IIc mouse has always been before. It's been moved to slot 7. Mouse-based software that assumes the mouse is in slot 4 won't work on the new IIc. The lesson to programmers is to *always* do a slot scan to identify the location of devices—don't assume anything, even on the IIc.

Another interesting problem is that when you do a slot scan of the new IIc, it *always* appears to have the memory card installed, whether it's actually there or not. This is because the ROM part of the card is built into the IIc, as with all other IIc devices. The memory card itself carries just the RAM. Apple tech support is advising developers to make a STATUS call to the protocol converter to determine how many blocks of memory (if any) are available on the expansion card.

The second version of the IIc, the one released for 3.5 inch drive support, prints the message "AppleTalk Offline" in response to a PR#7. That version of the IIc actually has some non-functional AppleTalk firmware in the slot 7 ROM space. This was removed from IIc version 3.

Apple's new disk drives include a 20-megabyte SCSI hard drive, a new 3.5 inch drive compatible only with the IIgs and the Macintosh, and a "new" 5.25 inch drive in a platinum-colored case.

The Apple Hard Disk 20SC will sell for $1,299 and can be used with any slotted Apple II or the Macintosh Plus. The most important element of this announcement is the $129 Apple II SCSI card, which brings the relatively low-priced world of SCSI devices to the Apple II kingdom. The new card can handle up to five standard SCSI (Small Computer Systems Interface, pronounced "scuzzy") devices.

The Apple 3.5 Drive is an 800K, 3.5 inch floppy different from, but priced exactly the same as ($399), Apple's UniDisk 3.5. The UniDisk 3.5 is "intelligent"—it has a 6502 microprocessor inside. The Apple 3.5 Drive is "dumb," like the drives on the Macintosh. The Apple IIgs can use either kind of drive. The IIc and the IIe can use only the UniDisk. The Macintosh can use only the dumb drives (unless you take the UniDisk 3.5 apart and disconnect the wires that make it intelligent). If you are buying 3.5 drives, the UniDisk is by far the better deal at these prices. (Or take a 50 per cent discount and buy Central Point Software's $195 + $6 shipping, $35 shipping outside North America, 3.5 inch dumb drive—9700 SW Capitol Hwy, #100, Portland OR 97219 503-244-5782.)

The new Apple 5.25 drive is the same color as the IIgs and is incredibly overpriced. Generic 5.25 drives are widely available for about 60 per cent less. Apple's II-family pricing strategy seems be to keep the old-timers happy and to keep customers coming through the door by selling new computers at a relatively low price, while tacking an institution/new-user tax onto Apple peripherals. Few sophisticated, independent users are going to buy these high-priced drives, but most new buyers and institutions will be afraid of going the generic route. It's a happy strategy for the kind of people who subscribe to ***Open-Apple***.

Apple's new monitors include the 12 inch AppleColor RGB Monitor at $499, the 12 inch Apple Monochrome Monitor at $129, and the 14 inch AppleColor Composite Monitor at $379. The RGB and monochrome monitors are suggested for use with the IIgs. The monochrome monitor has a black and white screen intended to mimic the Macintosh. Apple doesn't recommend its new composite monitor for use with IIgs super high-resolution graphics; apparently the resolution, at 560 x 192 pixels, isn't high enough. However, the screen is slightly larger, the price slightly lower, and the exterior slightly more platinum than Apple's previous composite monitors.

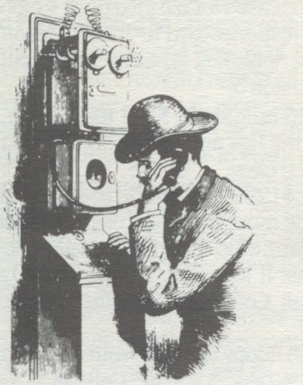
No, the new version of AppleWorks doesn't have mouse support, and it's not written in 65816 code. It does require at least 128K of RAM. New features are a built-in mail merge facility that allows users to combine information in a database file with information in a word processor file, several spreadsheet enhancements, and better support for Apple memory cards.

The spreadsheet enhancements include logical "ands" and "ors" inside @IF statements; a new rounding feature that allows calculations to use the visible, rounded-off number rather than the actual underlying number, thereby providing less precision but ensuring that financial spreadsheets "add-up"; and the ability to copy the *values* in cells, rather than the *formulas,* to the clipboard.

The new version automatically loads into Apple memory expansion cards at startup. With a 1-megabyte expansion card, the new version provides a 1,012K desktop. Word processing documents can have 7,250 lines, databases 6,350 records, and spreadsheets 10K of data per row.

New manuals and tutorials have also been developed. The suggested retail price remains at $250. Current owners of any previous AppleWorks version can update to the new version 2.0 by filling out a pre-addressed mailer available from Apple dealers and enclosing $50, their original disks, and the cover from their reference manual. In return you will get a complete new package, including manuals, tutorial disks, and AppleWorks disks in both 5.25 and 3.5 inch formats. The upgrade offer is scheduled to expire on April 30, 1987.

Only one thing was missing from Apple's September announcements. No mention was made of a trade-in program for the Apple II-Plus and Apple III. Apple president John Sculley strongly hinted such a program would be available when speaking at the Apple World Conference last January; let's hope the program slipped through the cracks rather than being swept under the table.



Ask (or tell) Uncle DOS

ImageWriter exonerated

I was surprised to read in the September 1986 ***Open-Apple*** (page 2.59) that I couldn't use my "standard" one-up, fanfold pressure sensitive labels with my ImageWriter. I've been doing it for years.

It's a standard ImageWriter (purchased in early 1984). And the Avery labels I use are standard (stock number 4145) and were easily available at any number of places the last time I bought some.

As for rolling the labels backwards—I've been doing that for years also with no problem (the Avery labels do stick to their backing better than some I've seen). I just have to make sure the pressure drive is off.

Hope this may help other ImageWriter owners.

Clay Wispell
Arlington, Texas

My apologies to the ImageWriter. It's Apple's earlier Dot Matrix Printer that can't accommodate standard one-up labels. In most respects the earlier printer works like an ImageWriter, but Apple should be congratulated for solving one of the DMP's most notorious problems in the ImageWriter design. (If only I had noticed that before—we could have been using cheaper labels around here for months already.)

One-uped

I got your September issue today and your note on page 2.59 reminded me of a friend's problem. As you so accurately stated, "standard one-wide pressure sensitive labels can't be used with several of the printers commonly found hooked to Apple IIs." I am using an Okidata with tractor and bottom feed because of this problem. I can roll forward and backward with ease and no labels stick in the printer's platen. Okidata has some disadvantages, however.

But back to my friend's problem. He is a professional printer with a rather large print-shop business. He has a large mailing list and wanted a dedicated printer just for mailing lists, etc. He had seen my Apple II-Plus and the software I use: *Bulk Mailer* (Satorie Software, 5507 Woodlawn Ave. N., Seattle, WA 98103 206-633-1469). On an Apple II with one disk drive it can hold 1,200 labels or up to 2,400 on two disk drives, or up to 32,000 with a hard disk. It has some very good search, duplicate name deletion, and sorting capabilities since it is disk oriented. With a typical list of 500 labels, it can sort by alphabet, zip code, or "X-code" (sort by type, etc.) in about 5 to 7 minutes, and printing is as fast as your printer will go. A print buffer helps to continue using the Apple while printing.

My friend learned several lessons with the purchase of his Apple IIe. He bought *PFS File* from the local Apple dealer as that was what they understood then and had in stock. Of course they sold him their printer as well. His only intended purpose at that time was for mailing labels and he had already bought a truck load (really!) of standard one-up labels *which he had shown to salesman* who called at his print shop to make the sale. To his amazement and ire, the labels would not fit his printer once delivered and working. Being of a mechanical mind, and with many years at "fixing" print shop equipment to meet printing deadlines and stay in business, he looked to himself (not Apple) for his solution. He had already called the local Apple dealer to be told that they had checked and that there was no solution.

So he took his Apple printer apart, used a soldering iron to remove some of the plastic on the back of the tractor feed holders (just enough so they would come close enough to grasp his one-up label pin-holes), and reassembled the printer. As of last week, over two years later, the printer was still pumping out labels without missing a stroke.

What did the local Apple dealer say? Of course he said it *invalidated the warranty*, but that was over two years ago, "and what does a salesman know from printers?", my friend asked. I made Polaroid photos of his printer disassembled and in operation and sent them to Cupertino for their comment. What did I hear? You got it! I am still waiting for a reply. But there was a bright spot in all this. I am still buying labels for my own mailing list use from my friend and he expanded his printing business to include micro-perf paper, diskettes, and labels a lot cheaper than even by bulk order by mail at "discount" rates.

Wayne Neighbors
Florence, S.C

Disks (and us) saved

In response to your question about disk recovery services in the September 1986 issue—I have offered a disk recovery service, intermittently, for several years. I say intermittently since I have never pursued it as an end to itself. However, if you would like to pass on my address, please do so. The telephone number above is always available for messages during the day.

I generally have quoted $25 per hour, and cannot currently recall having to charge more than $25. If a blown disk is going to take longer, I talk with the customer and list the options available, and their likelihood for success.

Two of the most common disk problems I have encountered are saving an empty file over a previous file (which my wife seems to specialize in, using *Apple Writer*) and the VTOC becoming trashed. Fortunately, each is (usually) easily correctable.

Guy Neill
Neill Enterprises
739 Ashland Terrace, Suite 203
Chattanooga, TN 37415
615-870-8340

You asked if anyone has been providing a disk recovery service. I have been doing this for the last couple of years. I charge anywhere from $5 to $20 per disk depending on what needs to be recovered, how quickly it needs to be recovered, and how long it takes to recover it.

In addition to your basic software like Bag of Tricks 2 and Locksmith, I also use EDD 4 Plus, which is a hardware system used to make archival back-up copies. Anybody who wants to have their disks fixed can write:

Michael de la Maza
The Apple Surgeon
10 Tumbleweed
Irvine, CA 92715

Polaroid (2020 Swift Drive, Oakbrook, IL. 60521, 312-954-1600) offers a disk recovery service for free, but only for Polaroid brand diskettes. It is part of their Gray Jacket Diskette's warranty.

Polaroid's David James recently told me the recovery process takes around 2 hours per diskette, but they have spent up to 5 hours on a single diskette. They recover data from 70 to 75 diskettes per month. They clean the diskette first to get rid of all the surface contaminants, then they read the diskette with a special diskette duplicator machine that uses data recovery software written by Polaroid personnel.

The data recovery is extensive—they read as much of the data as possible from the diskette and then rebuild the missing parts. This includes rebuilding directories, bit maps, and file structures that have been damaged. This means that if a sector halfway through an AppleWorks database file is glitched, then they try to make sure that the rest of the good data can be read by AppleWorks by putting dummy information in place of the missing data. James said that Polaroid supports over 100 different machines and operating systems with this recovery system.

I asked if Polaroid was selling this system and he said they weren't. The only way to get their service is with their brand of diskettes.

Jim Luther
Kansas City, Mo.

Of mice and macros

I don't think you've heaped enough praise on Beagle Brother's *MacroWorks*. It seems to me to bring AppleWorks up to as near perfection as a mortal could dare hope for. I don't know whether you are interested in publishing macros in ***Open-Apple*** but I thought you might at least be interested in how I am using *MacroWorks*.

One of the neatest things I have done with it so far is to write a macro that works with a specially set up database file of addresses. When I'm writing letters, I load my address database and a standard letter format onto the desktop, find the person who I want to write to, put a check against that record, then hit solid-apple-&. This macro selects the checked record, prints it to the clipboard, and then dumps it into the right position in the letter, all in a matter of about two seconds.

The ability, through *MacroWorks*, to use the mouse in AppleWorks is also a great feature. I share your deep reservations about the Macintosh-style desktop (I use a Mac at work, and hate the beast), but the ability to use a mouse when and if I feel like it is great. The best use of the mouse is when creating customized database layouts (in single record format or when creating a label format). Use the mouse to put the cursor on the first letter of the category, hold down the open-apple key, and use the mouse to take the category where you want it on the screen. It beats the pants off of using the cursor.

I'm also looking for help in a couple of areas. The first is that my Apple IIc seems to be developing a degree of keybounce while I'm using AppleWorks. I used to own a Tandy Model I, and boy did that have keybounce. This problem is nowhere near as severe, but it is annoying. I also have a niggling feeling that it is not a straightforward hardware problem, but something a bit more mysterious—I mean, occasionally I hit an 'a' and get something like 'azxa'. Or I get a '0' (that's a zero) appearing in the middle of my typing. I'm not exactly a ham-fisted typist—I can touch type at around 80 words a minute. Am I going too fast for AppleWorks? Perhaps some of your other readers with IIcs could tell me if they've experienced anything similar.

The second area is really just wishful thinking. We Australians use the British date format: day-month-year. Somehow, we kind of think that's more logical than your month-day-year format. But AppleWorks (along with most U.S. software) insists on making me enter the date in the U.S. format. Some programs (including the ProDOS CATALOG command) remove all ambiguity by spelling out the month, e.g. 29-AUG-86. But it's confusing to me to be forced by AppleWorks to enter 8/29/86, when all my background is telling me that this must mean the eighth day of the 29th month of 1986.

Is there any fix for this?

David Grigg
Research, VIC Australia

__Open-Apple__ is always interested in good AppleWorks macro tips.

Both Dennis and I seem to remember other reports of IIc keyboard problems, but can find nothing specific except an article in the April 1985 __inCider__ (page 61) about the world's fastest typist, Barbara Blackburn. She switched to using the IIc exclusively because on U.S. models the keyboard switch brings up the Dvorak keyboard layout. (On international models this switch is used to select either the standard U.S. layout or the standard layout in the country of sale.) Blackburn's record is 150 words per minute for 50 minutes and she reportedly can do bursts of 190 words per minute. Consequently, we don't think your problems are inherent in the machine's hardware, as was the case with the Tandy, but can't tell you much more. Can anyone else help with this?

Your point about date formats is a good one that software authors should give more consideration to. I think the best solution is software that allows the user to choose how dates should be formatted, rather than forcing anyone to use what's unfamiliar. In my own case, for example, I prefer what AppleWorks does with dates to what ProDOS does, but the day-first format is an alternative that should always be supported by software.

AutoWorks #1

I was pleased to see the fine people at Software Touch get some mention for the really super programs they put out, but I was surprised that an avid AppleWorks user like yourself did not give more space and praise to *AutoWorks*. I cannot remember what it was like using AppleWorks before *AutoWorks*, but I know I could never use the program again without *AutoWorks*.

Its macro and mail merge functions are so flexible and easy to use that even a novice like me can easily install the *AutoWorks* modification to AppleWorks. The program has literally saved me hours by automating repetitive steps. Two members of our local AppleWorks S.I.G. are currently using the *AutoWorks* "Read Disk" function to put the names of all the files in the 28 public domain T.A.W.U.G disks into a data base file. Then they are adding short descriptions of each file to the data base. It will then be used as a reference source by our members.

I made the mistake of recommending *MacroWorks* to our S.I.G. members when it first came out, but I have now changed my recommendation to *AutoWorks* without qualification.

I also use *FontWorks* from The Software Touch to print my AppleWorks files sideways without having to first Print-To-Disk and convert to DOS 3.3. FontWorks does the whole job effortlessly and reads AppleWorks spreadsheet files directly.

Thomas E. Militello, M.D.
Chairman, AppleWorks S.I.G.
Original Apple Corps
Los Angeles, Calif.

So much new stuff passes through our mailbox every month that it's hard to keep up. However, once I got time to take a close look at __AutoWorks__, it took only another second to realize, as you have, that it's the top AppleWorks add-on package of the moment. Its big advantage over __MacroWorks__ is that a separate "compilation" step isn't necessary to change existing macros. You can update one or all of the __AutoWorks__ macros from within AppleWorks at any time. Installation, as you mention, is a snap, and the mail merge function also looks terrific. It will be interesting to see how it compares to the new mail merge built into AppleWorks 2.0 (It will also be interesting to see whether __MacroWorks__ and AutoWorks have to be modified to work with version 2.0.)

Trading inserts

The last couple of issues you've inserted advertising for commercial software packages inside the **Open-Apple** envelope. Are these inserts endorsements or paid advertising? I don't want **Open-Apple** to be like my bills from the oil companies, where I have to wade through a lot of garbage to get to the important stuff. However, any company that offers to let you try their product free as Spectral Graphics does deserves support.

Richard Halliburton
73277,3327

The inserts U.S. subscribers have been finding in their envelopes are neither endorsements nor paid advertising. We have been testing the idea of carrying inserts to generate advertising revenue (we pay for a full ounce of first class postage but the newsletter weighs only half an ounce), but have stumbled upon a better arrangement for both us and the "advertiser." That is to simply trade inserts.

We will carry inserts for selected Apple II-related products and the producers of those products will, in turn, place inserts about __Open-Apple__ in their packages. This provides a very cost-effective way for a small publication (us) and other small Apple II-related companies to reach potential customers. Anyone interested in trading inserts with us should contact __Open-Apple__ business consultant Rich Barger at 816-942-2228 for the full details.

The ultimate modification

I have made several modifications to Basic.system 1.1 and to ProDOS 1.1.1 to eliminate bugs and improve performance. The modifications are:

1. Patch Basic.system to permit BSAVEing to non-binary files without first CREATEing them (Mark Jackson, ***Apple Assembly Line***, July 1985, page 30).

2. Fix the Basic.system bug in the CHAIN command (Peter Meyer, ***Call -A.P.P.L.E.***, June 1985, page 49; Tom Weishaar, ***Open-Apple***, April 1986, page 2.20).

3. Fix the parameter problem in the Basic.system BSAVE command (Sandy Mossberg, ***Nibble***, August 1986, pages 93-105).

4. Replace the ProDOS QUIT code with a program selector (Bob Sander-Cederelof, ***Apple Assembly Line***, July 1986, pages 2-18).

I would like to be able to distinguish my modified PRODOS and BASIC.SYSTEM files from the original Apple-supplied files. Is it a violation of the Apple copyright if I modify the sign-on strings in PRODOS (located at $24E3-$24F6 when PRODOS is BLOADed at $2000) and BASIC.SYSTEM (located at $2273-$2282 when BASIC.SYSTEM is BLOADed at $2000) so that they indicate that the modifications have been performed?

Robert C. Moore
Laurel, Md.

There is nothing in the copyright law that prevents you from modifying an original copy of a copyrighted work. The copyright law only prohibits making copies of the original, whether modified or not.

Even within the bounds of the copyright law, however, you can make copies of copyrighted works for certain specific purposes deemed "fair use." This includes archival copies. I don't know of any legal precedents that determine the bounds of "fair use" for copyrighted computer software, but I suspect even a jury of software developers would agree that putting modified copies of ProDOS on all your personal disks is "fair use."

A PFS-AppleWorks cable

I have moved a lot of PFS file data to AppleWorks by cabling two computers together and printing the PFS files into *Apple Writer* (ProDOS), using *Apple Writer's* communication mode. This mode is a little awkward and clunky, but a 48K buffer is darn nice compared to most communication programs. I then cleaned up the data a bit while in *Apple Writer*, saved it off in text files, and loaded it into AppleWorks.

It took a little patience, but with over 3200 records in the correspondence file, retyping was out of the question.

John Noell
Eugene, OR

Password privacy

I am writing a program that should have a password entrance. My problem is that when I have the user enter the password, it appears on the screen. How can I replace it with a series of asterisks from Applesoft?

Tom Firon

Try this:

```
10 PRINT "ENTER PASSWORD: " : PW$=""
20 GET A$ : PRINT "*";
30 IF A$ < > CHR$(13) THEN PW$=PW$+A$ : GOTO 20
40 PRINT
```

Space Coast surfaces

Around the first of the year we started using our first Challenger 20 hard disk from Space Coast Systems in Titusville, Fla. Everything was fine and then Space Coast suddenly disappeared. What happened?

Curt N. Wade
San Diego, Calif.

According to a message Dennis found on a local bulletin board, Space Coast Systems went bankrupt. However, some former employees are trying to continue to support the products through a user *group. Get in touch with them at Space Coast Systems User Group, P.O. Drawer 2767, Titusville, FL 32781.*

3.5 questions

I am thinking about buying the new 800K 3.5 drive announced by Central Point Software as a compact backup for my Sider, and for those times when I have too little work to warrant powering up the hard disk.

Do you know of any problems with booting AppleWorks 1.2 from a 3.5 inch disk?

For easy backups of the 3.5 disks themselves, I want to use a disk copier that accesses my 1-meg RAMWorks II for 1-pass copies. Have you actually used such a program?

Tom Smith
Fort Vancouver, Wash.

Central Point's new 3.5 drive and controller look very good on paper. For $285 you can get one 3.5 drive and a controller that will accept either a second 3.5 or a standard 5.25 inch drive (or even two 5.25s). For the 5.25 connector, the controller will accept either old-style 20-pin cables or the newer DB-19 plugs. In addition, the controller works not only with UniDisk 3.5-type drives (which contain an internal microprocessor), but also with Macintosh-type 3.5 inch drives (which don't). (Central Point Software, 9700 SW Capitol Hwy, Suite 100, Portland, OR 97219)

AppleWorks 1.2 works just fine with a 3.5 inch drive, as long as you are willing to access the device by pathname. As a hard disk user, you should be very familiar with this. The later AppleWorks 1.3 does two things with 3.5 inch disks that version 1.2 can't do—it allows you to select the root directory of a 3.5 inch disk by slot and drive and it will format 3.5 inch disks.

*Both Glen Bredon's **ProSel** and Bill Basham's **Diversi-Copy** include fast disk copy routines for 3.5 inch disks that will use all available memory. I have used both and do not hesitate to recommend either (**ProSel**, $40, Glen Bredon, 521 State Road, Princeton, NJ 08540; **Diversi-Copy**, $30, Bill Basham, 34880 Bunker Hill, Farmington, MI 48018.)*

I'm running as fast as I can

I recently purchased an Apple UniDisk 3.5 for my IIe but I could not get the device to be recognized. The NO DEVICE CONNECTED message appeared when I tried booting from the slot into which the UniDisk controller was inserted.

By trial and error, I found out that my MCT SpeeDemon was the culprit. When the SpeeDemon was disabled, the UniDisk could be accessed. Any reason for this incompatibility?

I have a Laser 128 at the office, and would like to add a UniDisk to that unit. Would there be any problem connecting an Apple UniDisk to the Laser 128?

Roger Rodriques, MD
Burr Ridge, Ill.

There is a jumper you can put on the MCT SpeeDemon that automatically slows it down when accessing devices in slots 4 and 5; use it and your problem will be solved. The SpeeDemon manual has more information. There seem to be far more devices that don't work reliably at an accelerator's full speed than there are devices that do.

The Apple UniDisk uses firmware on its controller card for the IIe; similar firmware is installed in the ROMs of newer IIcs. If you buy a UniDisk for an older IIc, Apple will change the motherboard for free so that you can get the new ROMs. Obviously, you can't plug the UniDisk into the disk drive connector on the back of a Lazer 128, as you can on a IIc, because the Lazer doesn't have these updated ROMs.

On the other hand, you can use the Lazer's peripheral slot for a IIe-type UniDisk controller card. You might want to check out the 3.5 inch drive from Central Point itself, which was mentioned in the last letter.

DOS 3.5

I am very interested in a DiversiDOS- or ProntoDOS-like operating system that will work with Apple's 3.5 inch UniDisk. I also want to be able to move this version of DOS to language card memory.

B. Webster
Hollywood, Calif.

DOS 3.3 enhancement packages for the UniDisk 3.5 have been quite popular with software developers—we know of seven packages and suspect there are more.

*None of them that we know of include either a fast DOS or a DOS mover. However, you can execute HELLO PRONTO-DOS from the Beagle Bros **Pronto-DOS** disk to add speed to all of them that we've tried (brun HELLO PRONTO-DOS after booting the 3.5 enhanced DOS as described in reference to the Sider in our June 1986 issue, pages 2.47-48).*

*Moving the multi-modified DOS to language card memory is more difficult. The program DOS-UP on the **ProntoDOS** disk will move only two of the 3.5 enhanced DOS versions that we know of—the latest version of Gary Little's **AmDOS** and Glen Bredon's **UnoDOS**.*

*Little's program is at the top of our list of recommended buys. At $15 ($20 Canadian) it is priced right. In addition to being compatible with both HELLO PRONTO-DOS and DOS-UP, it is fully compatible with DOS 3.3 partitions on hard drives and on Apple-style RAM cards. It organizes the 3.5 inch disk as two 400K volumes. If you have two 3.5 devices, the first appears as drives 1 and 3, the second as drives 2 and 4. **AmDOS** includes patches for FID that allow it to use 4 drives and that fix a problem FID has with using sectors 16 through 31 on 400K disks. (**AmDOS**, $15, Gary Little, 131 Water St, #210, Vancouver, BC Canada V6B 4M3 604-681-2051.)*

*Bredon's **UnoDOS** is a part of his **ProSel** package. Unlike the other packages listed here, **UnoDOS** never devotes more than 200K of a 3.5 inch disk to DOS 3.3. The remaining 600K are always left for ProDOS. The primary purpose of **UnoDOS** is to allow both DOS 3.3 and ProDOS programs to be stored on the same disk. **ProSel** then provides a single menu that you can use to execute any of the programs in either operating system. Bredon doesn't include any patches for FID, but then his program doesn't need any. Each **UnoDOS** disk has just one DOS 3.3 volume on it and the high numbered sectors aren't used. (**ProSel**, $40, Glen Bredon, 521 State Road, Princeton, NJ 08540 609-924-5976.)*

*The reason **AmDOS** and **UnoDOS** can be moved to the language card is that all of the patches they make to DOS 3.3 reside within the DOS image. In order to make room for these patches, the programs remove the DOS INIT command and use its space. Thus, to actually initialize a 3.5 inch disk with either of these packages you must use a separate formatting program.*

*Of the remaining packages, two give 3.5 inch drives the ability to work not only with DOS 3.3, but also Apple's Pascal operating system. One of these is Nordic Software's **ProFix 2.1**. The other, Cirtech's **Uni-Mate**, even turns a UniDisk 3.5 into a 784K CP/M drive. These two packages put the DOS 3.3 UniDisk drivers in the normally-empty bank of language card memory and thus require 64K machines. **Uni-Mate**, however, includes a second program that puts the drivers between the DOS buffers and HIMEM on 48K machines. Unfortunately, none of these locations is appropriate for DOS mover utilities such as DOS-UP. Caution—the 48K **Uni-Mate** version appears to move to the language card without a hitch, but it will eventually crash when the disk drivers, which get left behind in the lower 48K, are overwritten by Applesoft strings.*

***Uni-Mate** is the only program that brings all four of the popular Apple II operating systems to the 3.5 inch drive. It supports Pascal versions 1.1 and 1.2 and Microsoft Softcard CP/M versions 2.20B and 2.23. Because it doesn't overwrite the INIT command, **Uni-Mate** can format 3.5 inch disks. It calls the two DOS volumes on your first 3.5 inch device drives 1 and 2; a second UniDisk is referenced as drives 3 and 4. (**Uni-Mate**, £28, Cirtech Ltd, Currie Road Industrial Estates, Galashiels, Selkirkshire, Scotland TD1 2BP 0896-57790, SourceMail AAH555.)*

***ProFix 2.1** brings DOS 3.3 and Pascal support not only to 3.5 inch drives, but also to any ProDOS-compatible mass storage device. It allows these devices to be partitioned into 140K, 280K, or 400K volumes in your choice of operating systems. It also allows smaller drives to be "linked" to create one large volume. It includes backup and restore utilities for all volume sizes. (**ProFix 1.2**, $69.95, Nordic Software Inc, 4910 Dudley St, Lincoln, NE 68504 800-228-0417.)*

*Another entry is Meacom's **UniDOS**. I haven't actually seen a copy of this one, but I'm fond of the drawings in Meacom's ads, which render an image of a person who bears a striking resemblance to Uncle DOS. (**UniDOS**, $35, Meacom, PO Box 272591, Houston, TX 77277 713-526-5706.)*

*The granddaddy of the genre is MicroSparc's **UniDOS 3.3**. It was the first UniDisk 3.5 enhancement package on the market. It's similar to Little's **AmDOS**, except that it places the UniDisk drivers between DOS and its buffers and consequently can't be moved by any of the DOS-move utilities I know about. (**UniDOS 3.3**, $49.95, MicroSPARC, 45 Winthrop St, Concord, MA 01742.)*

*Finally, the most educational, entertaining, and cheapest DOS 3.3/UniDisk 3.5 enhancement package of all is the article "DOS 3.3 for the UniDisk 3.5", by Bill Morgan, which appeared in the May 1986 **Apple Assembly Line** pages 2-15. Morgan's version fits completely within the DOS 3.3 image and should be movable by DOS-UP. (**Apple Assembly Line**, $18/yr, S-C Software, PO Box 280300, Dallas, TX 75228 214-324-2050.)*

*One last thing we should mention about these programs is that they are not necessarily compatible with each other. Files on a 3.5 inch DOS 3.3 disk initialized by one package may appear scrambled or non-existent when the disk is read by another of the packages. Bredon's **UnoDOS**, for example, doesn't put the DOS 3.3 catalog in the same blocks that **Uni-Mate** and **UniDOS 3.3** use. The DOS 3.3 track/sector to ProDOS block mapping also seems to vary slightly from package to package.*

Hybrid 3.5 disks

In a recent issue of **Open-Apple** you discussed a method for making a disk readable by both DOS 3.3 and ProDOS. I was wondering if it would be possible to do the same thing with the UniDisk 3.5.

It seems to me that if you could reserve one side of the disk for each system it would greatly facilitate transfers between the two. I have little experience in this area, but it seems that all you would have to do is tell each system that the other half of the disk is full. I don't know the proper pokes to do this, and would greatly appreciate your ideas on this matter.

Jim Parker
(76327,171)

*As mentioned in the last letter, Glen Bredon's **UnoDOS** divides 3.5 inch disks into a 200K DOS 3.3 section and a 600K ProDOS section.*

It doesn't facilitate transfers between DOS 3.3 and ProDOS however, because all the conversion programs I know of (CONVERT, System Utilities, Copy II Plus) are ProDOS based. These programs only see the ProDOS portion of the disk.

For example, the Sider hard disk is another hybrid DOS 3.3/ProDOS device. But to move stuff from one partition to the other requires that you first move the DOS 3.3 files to another drive. ProDOS-based conversion programs simply can't read the DOS 3.3 portion of the Sider. A DOS 3.3-based conversion program might be able to read hybrid disks, but probably not, and I don't know of such a thing, anyhow.

FISHEAD 3.5

I have a program that allows me to continue using my old DOS 3.3 programs with the UniDisk 3.5. I have to use DOS 3.3 because my data is in random-access text files and ProDOS (or at least CONVERT) refuses to handle them. Is there any hope ProDOS will be upgraded to handle random-access text files?

In the meantime I have to live with the old DOS. However, I do not know of any fast and good file-copying programs that can access slot 4, drives 1-4. There is one excellent modification to FID, provided some years ago by Rana (which has ceased to exist): *RFC*. Unfortunately, it accesses only slots 5 and 6. I must use slot 4 for the UniDisk because my other slots are taken. Is there a way to change *RFC* so that it will work with slot 4 as well?

Stefan Soltysik
Pacific Palisades, Calif.

*While it's true that CONVERT can't handle random access files, ProDOS itself can do everything with them that DOS 3.3 can, and then some. ProDOS versions of Central Point Software's **Copy II Plus** can move your existing random-access files to ProDOS. However, as mentioned in the last letter, you'll have to move them back onto floppies first; **Copy II Plus** can't read 3.5 inch disks formatted for DOS 3.3.*

*There are a number of people around who know how to modify FID, unfortunately, none of them work for **Open-Apple**. **Nibble** has published two nice articles on modifying FID, "FID Plus" by Joe Humphrey (April 1984, page 73) and "Vigilant FID", by Donald W. Miller, Jr. (June 1985, page 126). Neither do what you want, however. Humphrey shows how to eliminate the necessity of pressing return throughout the program and how to change the commands from numbers to mnemonic symbols; Miller shows how to move FID to the language card and run it with a new DOS 3.3 FID command. I'm pretty sure modifying FID to accept more than 2 drives would be pretty simple. Another modification DOS 3.3 high-capacity drive users would like to see is the ability to specify volume numbers. Then there's the patch mentioned earlier to make FID work with 400K disks. Do we have any readers out there who can send us some patches?*

Incidentally, here's some FID lore you may be interested in. People have wondered through the ages what FID stands for. Apple's DOS 3.3 manuals claim it's (fi)le (d)eveloper, which sounds more appropriate for something that comes with a camera than with a computer. Rich Williams, the Apple employee who wrote FID, recently admitted that the original name of the program was FISHEAD (it had some mnemonic meaning which has since been lost). Apple's marketing people said that would never do and asked him to rename it. So he called it FID and the marketing wizards were happy. What he didn't tell them was that FID means (F)ishead (I)n (D)isguise.

Another road to 800K

About the time I had half-way decided to purchase a Sider, along came this EquiDisk-Plus device that can put 1.6 megabytes of storage on line (8 megabytes on a box of disks) and can work equally well with DOS 3.3, ProDOS, Pascal, and CP/M 3.0. Have you seen this disk? Is it for real? Where does this leave the hard disk industry?

Larry T. Green
Mayfield, Ky

The EquiDisk-Plus ($499, H&M Disk Drives, 1101 E Pacifico, Anaheim, Calif. 92805 714-385-1146) consists of two half-height high-capacity 5.25 inch floppy disk drive mechanisms in a cabinet designed to fit between your computer and monitor. The unit includes a surge suppressor, three extra power outlets, a power switch, and a power supply for the drives. The package also includes a small installation manual and an interface card.

Installation software is provided for MicroSoft's SoftCard versions 2.20B (56K) and 2.23 (60K) of CP/M 2.2, PCPI version 2.0 of CP/M 2.2, Advanced Logic System's version of CP/M 2.2 for the Z-Engine (not their CP/M 3.0 card), and for ProDOS and DOS 3.3. Read those ads carefully! The ads don't mention Pascal, and the disk doesn't support it.

Dennis installed the software for DOS 3.3, ProDOS, and the PCPI version of CP/M on a unit H&M loaned to the local Apple user group. He reports installation was straightforward with the help of the manual. The program disks include on-disk HELP files and the programs are menu-driven.

H&M's ads note that the minimum system configuration is a 64K II-Plus or IIe with one standard drive. You must have a standard drive because the EquiDisk can't read Apple disks. In addition, the EquiDisk configures itself for the various operating systems by loading a software driver stored on an Apple-format disk. Since no driver is loaded at startup, you can't boot from the EquiDisk.

The EquiDisk drives create disks of 800K (each drive; a total of 1.6 megabytes for the pair). For ProDOS, you get a single 800K volume per drive. For CP/M, each disk also is formatted as a single CP/M volume of 800K capacity; the normal assignments are as CP/M drives C: and D:. For DOS 3.3, each disk is divided into two 400K DOS volumes differentiated as drives 1 through 4.

In terms of speed, the EquiDisk transfers files at about the same speed as standard Apple floppies. In a few situations, such as reading CP/M files, the EquiDisk is faster; in others, such as reading ProDOS files, the EquiDisk is slower.

You ask for a comparison of the EquiDisk and the Sider. The Sider is much faster and has more total on-line capacity. The EquiDisk, on the other hand, has removable media.

This comparison doesn't get at the real strength of the EquiDisk, however. The EquiDisk uses a recording technique known as MFM (modified frequency modulation). This is totally incompatible with the GCR (group code recording) method used by Apple-compatible floppy drives. However, the use of MFM allows the EquiDisk to read disks formatted for a Kaypro (DSDD, 48 TPI) CP/M system. With the addition of one of the widely available CP/M programs that convert one machine's disk format to another, the EquiDisk can read a multitude of CP/M formats and transfer the files onto Apple floppies in your standard Apple drive.

Moreover, reading MS-DOS disks is technically feasible. H&M has a driver for MS-DOS under development. As mentioned earlier, the bad part of having to load the EquiDisk with software is that you can't boot from it; the good part is that you can theoretically load it with software to read any MFM-type disk.

The real comparison is Apple's 3.5 inch 800K solution to the EquiDisk's 5.25 inch 800K solution. If CP/M and MS-DOS conversions are of no concern to you, the smaller, sturdier, bootable, and more expensive 3.5 inch disks may have the advantage. If you are in a situation where you also use CP/M or MS-DOS computers, on the other hand, the EquiDisk not only costs less than Apple's drives, it also has the potential to solve some major incompatibility hassles.

The IBM connection

Many of us use IBMs and compatibles at work. Some information on how to transfer files, gain similar features, etc. would help.

Peter R. Nurse
Tunichannock, Penn.

*First of all, let's make sure everyone realizes that you can't transfer AppleWorks to your IBM or **Lotus 1·2·3** to your Apple and expect it to run. On the other hand, while you can't transfer program files, what you can transfer back and forth is data. In general, however, the data you transfer is going to have to be in standard ASCII text files. This means that both your Apple II and IBM programs are going to have to be able to read and write standard ASCII text files. If you don't have such programs on both computers, trying to transfer files back and forth is going to be nothing but an exercise in frustration. But let's assume standard ASCII text files are just exactly what you want to transfer....*

If the EquiDisk-Plus people mentioned in the last letter get their MS-DOS drivers working, the EquiDisk will be the easiest tool we know of for moving data between Apple IIs and IBMs and compatibles. It sure seems like there ought to be a half dozen devices on the market that allow Apples to use IBM-formatted disks, but we don't know of any others.

*From the other direction, there's a device for IBM and compatible disk drives called Apple Turnover that allows those machines to use 5.25-inch Apple-format disks. A local user who has one says it's great; unfortunately, the company that made the device, Vertex Systems in Los Angeles, has gone out of business. But the latest issue of **Call ·A.P.P.L.E.** (September 1986, page 52) has a few paragraphs* about a similar device called Matchpoint-PC, which consists of a half-size interface card, software, and cables that hook to the IBM's floppy drives. I don't know anything else about this product but I did call the company to make sure **they** were still in business, and they say Matchpoint-PC is selling every bit as well as I would have expected it to. It allows IBMs and compatibles to read and write on DOS 3.3, ProDOS, SOS, and Apple CP/M disks. (Matchpoint-PC, $195, MicroSolutions, 125 S Fourth St, DeKalb, IL 60115 815-756-3411.)

A paper-based approach to compatibility is the Cauzin Softstrip System (Cauzin Systems, 835 South Main St, Waterbury, CT 06706 203-573-0150). Cauzin sells a program called the **Stripper** that will print DOS 3.3 text files onto paper in a special format that can be read by their Softstrip reader. You take the sheets of paper to an IBM or compatible equipped with a reader and read them in—in the process the files are effectively converted from one system to the other. Cauzin has another **Stripper** program that allows your IBM to print Softstrips so you can move data back to your Apple. If you are willing to carry the reader back and forth, you'll need one reader with an accessory kit for your Apple ($199), an extra accessory kit for your IBM ($19.95), and two **Stripper** programs ($19.95 each).

Open-Apple

is written, edited, published, and

© Copyright 1986 by
Tom Weishaar

Business Consultant Richard Barger
Technical Consultant Dennis Doms
Circulation Manager Sally Tally

Most rights reserved. All programs published in **Open-Apple** are public domain and may be copied and distributed without charge (most are available in the MAUG library on CompuServe). Apple user groups and significant others may obtain permission to reprint articles from time to time by specific written request. Requests and other editorial material, including letters to Uncle DOS, should be sent to:

Open-Apple
P.O. Box 7651
Overland Park, Kansas 66207 U.S.A.

ISSN 0885-4017. Published monthly since January 1985. World-wide prices (in U.S. dollars; airmail delivery included at no additional charge): $24 for 1 year; $44 for 2 years; $60 for 3 years. All back issues are currently available for $2 each; a bound, indexed edition of Volume 1 is $14.95. Index mailed with the February issue. Please send all subscription-related correspondence to:

Open-Apple
P.O. Box 6331
Syracuse, N.Y. 13217 U.S.A.

Subscribers in Australia and New Zealand should send subscription correspondence to **Open-Apple**, c/o Cybernetic Research Ltd, 576 Malvern Road, Prahran, Vic. 3181, AUSTRALIA.

Open-Apple is available on disk for speech synthesizer users from Speech Enterprises, P.O. Box 7986, Houston, Texas 77270 (713-461-1666).

Unlike most commerical software, **Open-Apple** is sold in an unprotected format for your convenience. You are encourgaged to make back-up archival copies or easy-to-read enlarged copies for your own use without charge. You may also copy **Open-Apple** 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 **Open-Apple** is useful and correct, although drivel and mistakes are included from time to time, usually unintentionally. Unsatisified subscribers may return issues within 180 days of delivery for a full refund. Please include a note from your parents or children confirming that all archival copies have been destroyed. The unfullilled portion of any paid subscription will be refunded on request. MY LIABILTY 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.

Open-Apple is neither affiliated with nor responsible for the debts of Apple Computer, Inc.; "tinaja questing" is a trademark of Don Lancaster.

Source Mail: TCF238 CompuServe: 70120,202

As I mentioned back in March (page 2.14), it seems to me the biggest market for the Cauzin Softstrip system is people who need a practical and efficient way to distribute relatively small amounts of digital data from one point to many other points. What I didn't realize then is that an additional advantage of the Cauzin system is that the end-points can all be different kinds of computers. This is a tremendous advantage over distribution by disk, where you'd have to produce disks in several different formats and keep track of who gets what.

Today's most commonly-used method of moving files back and forth between IBMs and Apples is with communication programs. You connect your Apple and IBM serial ports together with either a "null modem" cable or with two modems and a phone line. Startup a program such as **ASCII Express** on your Apple and a program such as **PC-TALK** on your IBM. Put one or the other system into remote mode and use the other computer to tell the first computer to send files to you or to receive files from you. That's all it takes.

Dennis has also used an Apple II hooked to a UNIX system (UNIX is an operating system like ProDOS or MS-DOS; the computer involved was a DEC PDP 11/23 minicomputer) to upload and download ASCII data files. He set up **ASCII Express** to make the Apple appear to be a DEC VT-52 terminal. Then he told UNIX to send a file to the fake VT-52 and he captured it in **AE**'s copy buffer. And he has opened a file on UNIX and then told **AE** to send one he had stored on his own disk—UNIX thought he was typing it in.

I have done the same thing to transfer some of my wife's research data into an IBM mainframe at the university where she teaches. The data was originally entered into an Apple II spreadsheet program—a far better data entry, checking, and editing tool than anything available on the mainframe.

This particular mainframe, on the other hand, has a statistical analysis package called SAS that does lots of things that simply can't be done with today's Apple II statistics programs.

After asking the advice of the mainframe gurus and finding out that it was impossible to transfer data from a lowly Apple II spreadsheet to SAS, I set up **ASCII Express** so that it turned my mild-mannered Apple into a big dumb ADM-31 mainframe terminal. On the mainframe end of the modem connection there was a protocol converter that answered the phone, found out I was an ADM-31, and made the mainframe think I was an IBM 3270 terminal. This is roughly equivalent to translating English into Hindi by way of Japanese, but it worked.

The **ASCII Express** documentation on how to do this is quite murky, as usual. After hours of reading and experimenting, I figured out you should begin by skipping the enticing chapter in the **AE** manual called "Terminal Emulation" and move directly to the description of the (S)et Terminal Parameters command in the chapter on macros (page 198 in my manual). In order to emulate a mainframe terminal your Apple must be able to display 80 columns. What the documentation calls your "output terminal" turns out to be your Apple. Make the current output terminal choice 5, in deference to the little parenthetical addendum about "80 col. bds." after that choice.

The "input terminal" is what you want your Apple to appear to be to the computer or protocol converter on the other end of the line. **ASCII Express** can make you look like an Adds Regent, Viewpoint, Hazeltine 1500 or 1510, Televideo 912, ADM-3A or 31, Soroc IQ-120, DEC VT-52, IBM 3101, or a Heath H19.

Some of these terminals have keys that don't appear on your Apple keyboard. What these special keys (such as CLEAR, NEWLINE, and RESET) produce is control-codes and escape sequences. You can duplicate all of these with your Apple; finding out what they are is either a matter of finding a technician at the receiving end who is willing to look it up for you or of doing a lot of experimenting. In my situation, I figured out by experimentation that control-C was CLEAR, control-I was NEWLINE, and control-R was RESET, but it was necessary to cajole a mainframe technician into photocopying the ADM-31 page in his protocol converter manual for me to discover that I could fake the function keys with nothing more than ESC and a number.

Normally you enter information into this particular IBM mainframe using a very fancy line editor called XEDIT. Next it was necessary to figure out exactly what keys XEDIT expects you to type, put those keystrokes in a file, then have **ASCII Express** send them. XEDIT thought we were typing them in. Essentially we were EXECing them into XEDIT.

We saved the spreadsheet data into DIF files. I wrote a short program that read the DIF files (see "DOStalk" in the February 1984 **Softalk**, page 65, for the required routines) and converted them into the keystrokes XEDIT expected.

When the files were ready, the next step was to use the Apple to call the mainframe, logon, startup XEDIT, and tell it we wanted to input some data. When XEDIT was ready for us to begin typing, we pressed "control-Q" to tell **ASCII Express** to wake up, then "S" to tell it we wanted to send a file. It asked for the file's name, asked whether this was to be a (S)tandard or (P)rotocol transfer (to which we answered "standard"), and asked for a prompt to look for.

XEDIT wants a newline character (control-I) at the end of each line and a return at the end of each page (20 lines). It could chew 20 lines of data as fast as **ASCII Express** could send them (at 1200 baud), but at the end of each page it needed several seconds to swallow. We eventually figured out we could trick **ASCII Express** by telling it to look for a "\" prompt from the IBM. If you specify a prompt, **AE** stops after every return character it sends and waits for the receiving computer to indicate it's ready by sending the prompt. We could see the data going out; every time it stopped we counted to 10 and then pressed the backslash key on the Apple's keyboard. **ASCII Express** thought that was the IBM talking and sent another page of 20 lines.

To ensure the integrity of the data my wife calculated means, minimums, and maximums for each data set with the spreadsheet and again with SAS after the numbers had been transferred to the mainframe. Everything checked out. The mainframe boys still don't know what hit them.

For better or worse

Help! I fought with my wife for two years ("everyone who gets a computer ends up with it in a closet"). I won, but now the only time I get to use it is 2 AM when she is asleep and isn't typing madly away. What do you recommend?

Greg van Hartesvelt
Marquette, Mich.

Suggest to her that the two of you get an Apple IIgs. My experience is that the fights over the second and later computers get successively more mellow. Spout the one-person-at-least-one-computer philosophy. The Apple IIgs I have on order will be our family's sixth Apple and it just occurred to me I didn't even ask my wife if we could get one.