A2-Central™

formerly Open-Apple

September 1991
Vol.7, No.8

ISSN 0885-4017
newstand price: $2.50
photocopy charge per page: $0.15

A journal and exchange of Apple II discoveries

KansasFest 1991

This year's ***A2-Central Summer Conference*** (often referred to as "KansasFest") had a different flavor than previous years. The conference itself was much the same, but the environment surrounding it provided a different background.

Apple's announcements. Apple Computer sent fewer engineers this year than last year, when many new products were being discussed. Some of those new projects have been released to developers (the MIDISynth tools) or the general public (HyperCard IIgs, the Apple High-Speed SCSI Interface), and the future of some (Jim Mensch's animation toolset) are not clear at present.

Apple's new product introductions this year again attack several criticisms of the Apple II. System Software 6.0 (due out sometime before the end of this year) adds more speed to the IIgs tools, a remarkably improved Finder, and three new File System Translators (FSTs). Other new products are an improved HyperCard IIgs version 1.1, and for the IIe and IIgs a new Ethernet peripheral card and the anticipated SuperDrive interface card.

System 6.0 will ship on ***five*** 800K disks. New programs will include an archiver (effectively a backup program that archives files on any block device medium from floppy disks to hard disks), drivers for Apple's tape drive and flatbed scanner, and the components of Universal Access (sticky keys, mouse keys, CloseView, and the Video Keyboard). The three new FSTs will provide for read-only access to Apple DOS3.3 and Apple Pascal disk formats and full read and write access for Macintosh HFS volumes.

The HyperCard revisions will add some features for enhanced use of color, better compatibility with Mac HyperCard 2.x (including "xwindows", the ability to create floating windows through HyperTalk) and a media integration stack using a new media control toolset.

The SuperDrive card is the anticipated interface to allow connecting the high-density Apple drive to a IIe or IIgs. In addition to the Apple 3.5's 800K (and rarely used 400K) disk support, with appropriate software the SuperDrive can read and write disks using 1.44 megabyte and 720K formats. The System 6.0 HFS FST will be able to handle the Mac HFS file system on 800K and 1.44 megabyte disk, but no mention was made of support for the MS-DOS 720K and 1.44 megabyte formats.

The Ethernet card adds support for that popular networking protocol to the IIe and IIgs. One of the motivations for the card was that it will have better performance than the current LocalTalk interfaces; the card is expected to be very reasonably priced.

None of these products were available to show in "final" form as of KansasFest, so detailed comments would be speculative. We'll hold off on a complete review until each of these new products is actually shipped, but we think Apple II owners will see a lot of new potential and realization in these offerings.

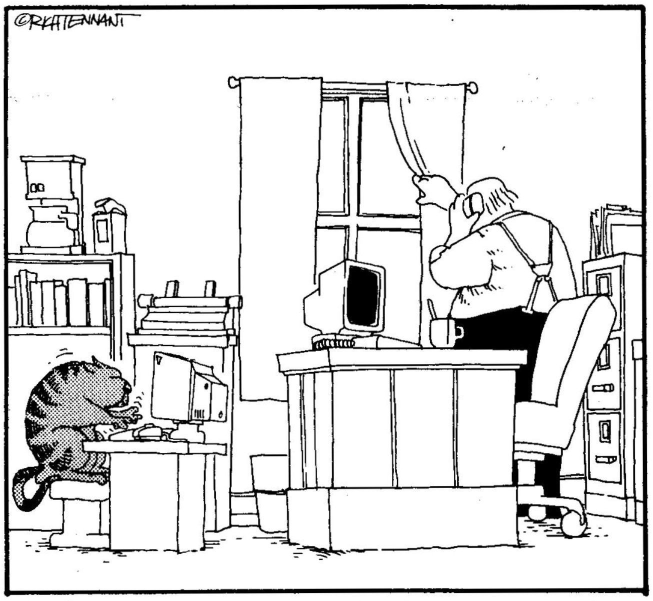
Roger Wagner's keynote address featured a history of hypermedia which Roger set into action and left to run as he wandered offstage. The history began with Bob Bishop's classic AppleVision, done in black and white on the original Apple II. Progressive screens enhanced the AppleVision image using subsequent incarnations of Apple II graphics (single high-resolution, double high-resolution, and the IIgs's Super High-Resolution modes). Finally, thanks to a laserdisc player under *HyperStudio*'s control and a video overlay card, Roger's image appeared within the television's screen and spoke to the audience, completing the introduction before turning the presentation back to Roger (returning from offstage). Presentations no longer require a human to be present, even if they are part of the presentation.

Rogers's major thrust was the capability and suitability of the IIgs as a hypermedia engine, demonstrating some of the things that had been achieved in *HyperStudio* and enhancements that were being planned. The features of the IIgs—multiple expansion slots, built-in ports, television-style (NTSC-compatible) video output, color graphics, a built-in sound chip capable of sound and speech digitization and speech—are capabilities making the IIgs the ideal computer for hypermedia. Roger used *HyperStudio* for his examples (it's his program and the hypermedia tool he's likely to be the most pleased and familiar with), but his message overall is that the IIgs does have great potential in areas that it is ideally suited for, and that potential has not been fully tapped.

An example of misinterpreting the IIgs's capabilities is the impression that the 200 line vertical resolution is a limitation. In one perspective, it is; the Commodore Amiga, Apple Macintosh, and multitudinous MS-DOS compatibles all support greater vertical resolutions. But the Macintosh and MS-DOS machines have no inherent video output capability, and video overlay peripherals for these machines cost 5 to 10 times the price of the Apple II Video Overlay Card. If you want a personally affordable and maintainable hypermedia machine, the IIgs and Amiga are the leading contenders, and the Amiga has the flicker problems in 400-line mode that IIgs engineers have pointed to as a reason why the IIgs currently does not support such a mode (the Video Overlay Card does function in a currently unsupported 400-line mode, but the flicker is annoying).

Roger also emphasized that hypermedia can be seen as a developer environment both for writing programs using the hypermedia program, and for creating tools, graphics, sound, and so on to use as enhancements to the environment.

New releases outside of Apple. This year's KansasFest had a two-day Apple Central Expo following it where many Apple II exhibitors displayed



"YEAH, I'VE FINISHED REVIEWING THE MONITOR, KEYBOARD, CPU AND PRINTER, AND I'M JUST FINISHING UP MY REVIEW OF THE MOUSE RIGHT NOW.

their wares. Some new "soon to be released" items were also shown. The following is a sampling of what we saw; we'll save in-depth coverage for future issues. (The ***A2-Central*** crew was too busy running its portion of KansasFest to see everything that was going on.)

Accudraw (Kitchen Sink Software, 903 Knebworth Court, Westerville, Ohio 43081, 614-891-2111) is a CAD-type drawing program for IIe, IIc and IIgs (also Laser 128) computers. It supports printing documents to many dot matrix printers (using Beagle Bros' *Triple-Dump* drivers), even tiling printouts to allow printing a document image several times larger than a single sheet of paper. Kitchen Sink also has many add-ons available , such as symbols for architecture and electronics and project drawings. (If you are interested in CAD software that outputs to a plotter, Kitchen Sink's older *CADDRAW* program still performs this function.)

Brian Walker of the Learning Performance Corporation, 2850 Metro Drive, Suite 413, Minn. 55425-9880, 612-851-3250 or 800-926-3279, demonstrated LPC's *LanPro* networking products. An Apple II (a IIe will work as well as a IIgs) is used as the server which you can boot from (a IIgs workstation can even boot into GS/OS) and use as a common repository for GS/OS, ProDOS and DOS 3.3 programs. Additional peripherals including hard disks, floppies, and printers can be shared. LPC's *EasyShare IIx* customizeable menuing system was also demonstrated in combination with *LanPro.*

Phil Shapiro of Balloons Software was demonstrating some of his award-winning software, but when we visited he actually used most of his time to show us a product other than his own: *StoryWorks* from Teacher's Idea and Information Exchange (TI & IE), P.O. Box 6229, Lincoln, Neb. 68506, 402-483-6987. *StoryWorks* is a program that allows you to use AppleWorks to create quizzes and tutorials with menus, hypertext linkages between words, and sound effects. If you don't have an 80-column card or like large text, *StoryWorks* can format your text to use a 30-column mode on the graphics screen. *StoryWorks* is $49.95 from TI & IE and will run on any Apple II with at least 64K and AppleSoft in ROM.

Integer BASIC is back thanks to the Byte Works (4700 Irving Blvd. N.W., Suite 207, Albuquerque, N.M. 87114, 505-898-8183) as the *ORCA/Integer BASIC* package. The package includes a compiler that runs under the *ORCA/M* IIgs shell and creates GS/OS executable files from Integer BASIC (ASCII text) source code. The compiled code uses the IIgs super high-resolution screen to simulate the older Apple II graphics modes used by Integer BASIC. Of course, quirks such as embedded machine language and some types of hardware manipulation (PEEKs and POKEs) are not supported, but the compiler does give the first Integer BASIC support we're aware of under the ProDOS (well, GS/OS) disk format. If all this seems too weird to you, consider that the real intent of this project was to provide a tutorial in compiler writing, and all source code (in a mixture of *ORCA/Pascal* and assembly language) for the compiler is provided. It was definitely the Apple II hacker hit of the show. If you order before September 30, 1991, Byte Works will throw in a free disk of *Integer BASIC Classics* culled from the public domain.

DreamWorld (P.O Box 830, Iowa City, Iowa 52244-0830, 319-338-6491) was demonstrating a more complete version of its IIgs *DreamPaint* ($99.95) software that was announced last year (see "Miscellanea", October 1990). Steven Chiang showed us the draft copy of the manual and hopes to have the product shipping within a few weeks. In addition to many standard paint features, *DreamPaint* allows editing graphics using 256 and 3200 color modes.

Micol Systems (9 Lynch Road, toronto, Ont. M2J 2V6, Canada) was demonstrating their revised version of *Micol Advanced BASIC for the IIe and IIc* which now includes windowing routines that can be used to enhance the user interface. Also at the Micol booth Vladimir Federov was demonstrating the *Liberty Card*, an interface designed to allow connecting MS-DOS drives to a IIe or IIgs for use as inexpensive ProDOS storage and for transferring data files to and from MS-DOS disks.

Roger Wagner's booth was doing its usual brisk business with demonstrations of *HyperStudio* features. Roger has a new "Neat HyperStuff" catalog that showcases several products from various companies; it's one of Roger's ways of making sure good products are publicized.

Procyon, Inc., 1005 N. Kingshighway, Suite 309, Cape Giradeau, Mo. 63701, 314-334-7078 was demonstrating a Unix-like multitasking development environment for the IIgs (still in development).

A new line of new internal hard disks for the IIgs was being shown by Econ Technologies, P.O. Box 195356, Winter Springs, Fla. 32719, 407-365-4209. The drives replace the internal power supply of the IIgs and unlike some internal drives use your standard SCSI interface. In fact, if you want to add your own hard disk mechanism, Econ sells just the power supply and drive housing for $299. Prices go up to $1099 for a 200 megabyte subsystem. (These prices do not include a SCSI interface card to connect the drive to your computer.)

Zip Technology (5602 W. Slauson Avenue, Suite #190, Culver City, Calif. 90230, 213-337-1313) representative David Fein showed up for the entire conference and was offering older version 1.01 *Zip GSX* to developers at drastically reduced prices during the colleges and conference sessions. The offer was also made to upgrade these cards to the current version 1.02 for an additional fee. The primary limitations of the 1.01 are compatibility problems with some hardware configurations and the inability to upgrade to versions with speeds over 8 MHz, but the price was very attractive for a basic accelerator.

An eclectic but interesting hardware product was an Apple II (II Plus, IIe or IIgs) peripheral card that provides high–speed (8 millisecond conversion time) 12–bit analog–to–digital conversions (software selectable from 8 analog inputs), a 12–bit digital–to–analog converter, and bitwise (digital) input and output. This device can be used to interface various sensors to the Apple II; one use being demonstrated was a real–time oscilloscope simulation. The *Advanced Interfacing Board II* is $260 from Sunset Laboratory, 2017 19th Avenue, Forest Grove, Ore. 97116, 503-357-5151.

The Big Red Computer Club (423 Norfolk Avenue, Norfolk, Neb. 68701-5234, 402-379-4680) has announced they will be picking up distribution of Electronic Arts's IIe and IIgs software titles; including *The Immortal, Keef the Thief, The Bard's Tale, Instant Synthesizer, Chuck Yeager's Advanced Flight Trainer,* and *Skate or Die.*

Claris (5201 Patrick Henry Drive, Box 58168, Santa Clara, Calif. 95052-8168, 408-987-7000) had a booth at the Expo, but had a rough time of it due to some confusion about the nature of the show. The Claris representative attending had not come prepared for the fact that the Expo was going to have an Apple II flavor; thinking the emphasis would be Macintosh, the demonstration machine she brought was a IIsi. On the second day of the show handouts were also in the booth for Apple II products, but by then some cretin had decided that the quickest way to justice was to damage the files on her hard disk. Folks, we're all acutely sensitive to how we're treated by some retrograde Macintosh users and companies; but showing such complete disrespect for someone's **personal** situation and property is inexcusable under **any** circumstances.

As it turns out, one of Claris's Mac products does have a positive feature for Apple II users: *MacWrite II* (soon to be enhanced to *MacWrite Pro*) includes Claris's new Xtnd technology for adding file translation capabilities directly into programs. *MacWrite II*'s file importing capabilities include translators for AppleWorks and *AppleWorks GS,* which provides a simple way to get files from an Apple II onto a Mac without losing all your document formatting. Unfortunately, the current *MacWrite* only supports exporting to AppleWorks 2.0 format, but maybe Claris could be gently persuaded to add AppleWorks 3.0, *AppleWorks* GS, and other formats.

Last year's KansasFest was Apple's show, with the roll-out of the major changes in system software and new hardware capabilities provided by the High-Speed SCSI Card and Video Overlay Card. This year, more of the hoopla was from companies taking advantage of new and rediscovered features of the Apple II, and Apple could step back a bit. That leaves the part of Apple that still has to get into gear...—DJD

Apple's developer message

The last two years continue to smolder in the hearts and minds of Apple II zealots. Apple keeps rolling out wonderful enhancements for Apple II owners while steadfastly refusing to let much of the rest of the world know about them. As a result, developers feel all dressed up with nowhere to go; you can bring a product to market, but you can't generate sales.

Apple II advertising stalled again. For Apple II owners, Apple as a company moves frustratingly slowly. It also seems that whenever Apple is caught in a fiscal "crunch", the first funds to be jettisoned somehow cripple any resurgence of the Apple II. In the spring of 1990, John Sculley emphasized the allocation of 20 per cent more research and development funding for the Apple II and we have seen many engineering improvements: System 5.x, the High-Speed SCSI Interface, the Video Overlay Card, and the KansasFest announcements. What we have yet to see is visible improvements in

Apple II support at Apple dealers (or to actually *implement* alternative methods of distribution) or to promote the Apple II outside of those already committed to it.

With the formation of the Apple II Business Unit last year, hopes were high that promotions would appear by this fall. This would indicate Apple had an interest in providing a more viable marketing environment for Apple II developers and customers by restoring some prestige to the machine in the public eye. Unfortunately, the first message Rob Barnes had for developers in a luncheon speech at KansasFest was that "it doesn't look as if we're going to have Christmas this year". The interpretation is that after two years of apparent appeasement, Apple ***again*** won't get a fall promotion for the Apple II launched and what was historically a strong period for Apple II sales will again be missed.

These decisions are out of the Business Unit's hands since they operate as a part of Apple USA, which allocates the funds. With shipments of Macintosh systems up 60 per cent and net revenue up only12 per cent compared to the third quarter last year, Apple took a $53.1 million dollar loss in this year's third quarter. This can hardly be blamed on the Apple II; why should the Apple II Business Unit (and Apple II customers) be penalized for it by cutting ***their*** funds?

There is a pitch being made to the new head of the Apple Consumer Products Division for positioning of an Apple II product, but Rob Barnes expressed doubts that anything could be implemented by this fall. If not, next spring (when schools start shopping for systems) would be the earliest opportunity; in other words, after years of asking for a little time, Apple wants a little more time.

The Apple II Business Unit has apparently not been idle, however. One indicator of Apple's direction is the content of their quarterly mailing to user groups. The July User Group Mailing had more Apple II content than most, but still fell short in a few significant areas (such as the user group video, which again had no Apple II segments).

A color brochure advertising Apple's educational solutions had the IIgs prominently displayed. The promotion of hypermedia was the major content of the brochure, with HyperCard IIgs and the Video Overlay Card getting the most space. The Macintosh Classic, LC and HyperCard were also included, but their role in the brochure appeared secondary.

Another brochure in newsletter form was dedicated solely to Apple II software, listing many products in several categories for the Apple II and IIgs. Many contact addresses for companies supporting the Apple II were listed, along with their products and prices.

We hope these brochures are also distributed to new markets in an attempt to convince those other than current Apple II diehards that the machine is viable and that Apple Computer, Inc. still supports the computer. The advertising brochure also mentions a specially priced educator's IIgs system package with two megabytes of memory, 3.5 drive, color monitor, and 40 megabyte hard disk with the System Software and HyperCard IIgs installed; we'd like to see this bundle advertised to a broader market.

Apple believes the current place to make money on the Apple II is in 8-bit educational software. Rob Barnes points out that a number of schools are still buying IIe instead of IIgs systems because they can buy more systems for the same amount of money. But what really tips the balance is that over 50% of the Macintosh LC systems being sold are using Apple II Cards. Apple relates this as the truth they have to present for struggling Apple II developers. But we remain skeptical of these arguments for several reasons.

Even if over half of the Mac LC systems sold have IIe cards, one has to believe that schools are buying these to run **existing** IIe programs. We hesitate to believe that a Mac owner would continue to buy software to run under a less–than–ideal emulation when a native Mac application may be available (and, unlike the dearth of attention given the IIgs, Apple is obviously pumping developers and promoting new Mac products to make ***sure*** Mac applications are going to be available). Encouraging 8-bit development helps to freeze the fate of the Apple II at the pre-IIgs level, which makes it much easier for Apple to sell Macintoshes. We've noticed Apple is fond of referring to the Apple II as "ten year old technology", and exposing Macintosh owners to IIe software does nothing to remove that onus. So it would seem advisable to us to remove the IIe Card from the numbers.

Additionally, some developers (including ourselves and John Wrenholt of the Big Red Computer Club) disagree with the assessment of the desire for 8-bit versus 16-bit software. Our own product sales tell us that IIgs products sell ***much*** better than 8-bit products.

Apple was more than willing to entice developers toward the Macintosh when its software sales were negligible; we remember that it took only a few months to revise the Mac 128K to 512K when new programs weren't appearing. Apple later also took to packaging HyperCard with new Macs, and selling newer and more powerful configurations capable of running it. Apple prefers to forget (or somehow excuse) these exceptions when discussing options for reviving Apple II sales. Apple should be encouraging developers to move toward 16-bit products for the IIgs (as well as the Macintosh) as they approach the limits of the 8-bit development environment on the Apple II.

None of this is to say that the advice for developers to consider 8-bit programs was offered out of anything but concern for their financial well-being. But Apple continues to not do enough to promote Apple II sales, and it should never expect its Apple II customer or developers to fail to look for duplicity in its motives.—DJD

Miscellanea

Apple has scheduled a nationwide live 90-minute telecast of UG-TV (User Group Television) for September 25, 1991. The broadcast will include on–air presentations by Apple Product Managers and Engineers as well as a live interactive question and answer session with participating sites.

In order to see the telecast, you'll need to find a satellite downlink site with an "aimable" dish that can receive the appropriate C–band or Ku–band signal. The informational sheet that came in the user group mailing suggests appointing a user group "broadcast coordinator" to check with local colleges and businesses that may have a site available for use or rental.

If you have your own dish, here's the technical information for reception:

Ku-Band Satellite Information:
Satellite: SBS-6
Transponder: 6
Polarity: Vertical
Downlink Frequency: 11847.5 Mhz
Audio Sub-carriers: 6.2 & 6.8

C-Band Satellite Information:
Satellite: Weststar4
Transponder: 20
Polarity: Vertical
Downlink Frequency: 4100 Mhz
Audio Sub-carriers: 6.2 & 6.8

The broadcast is scheduled for 9:30 PM Eastern Daylight Time; Apple will be broadcasting a test signal for about an hour before the actual broadcast starts so that you can test and tune your reception. (Please don't ask us for details; we don't do satellites.)

Interested user groups should contact USER.GROUPS on AppleLink or Apple UGC on America Online before September 1, 1991 (yes, we know; we got caught at deadline) with the name of their broadcast coordinator and technical details if they have

Apple and IBM have announced a technology partnership. Their aim is to create new "open software platforms" for the 1990's. Four areas of understanding were outlined in a letter of intent:

- A joint venture for object-oriented software. The new system software platform will be based on object-oriented technology and be able to run on a range of processors. This will be facilitated by the formation of a jointly-owned software company that will be independently managed. The software will be offered for both IBM and Apple computers, and also marketed for use by other vendors.
- The integration of the Apple Macintosh into IBM's system environment. This will be accomplished by the development of communications and networking products to extend the ability of the Mac to operate in the IBM environment, and enhancements to AIX (IBM's Unix operating system environment) to combine IBM's open systems with Macintosh applications. The enhanced AIX will offer Macintosh and OSF/Motif user interfaces.
- Apple plans to adopt future single-chip versions of IBM's RS/6000 POWER RISC (Reduced Instruction Set Computer) architecture for future Macintosh computers. The new processor (called POWER PC) will be designed and manufactured by IBM and Motorola. Motorola will serve as a supplier to Apple, IBM, and other systems vendors.

- A common multimedia platform for the industry. Apple and IBM plan to create and license platform-independent environments toward this end.

Apple and IBM expect to have contracts that define the actual terms later this year, and expect products to reach the market over the next two to three years.

APDA, Apple's mail-order source for developer tools and documents, is eliminating its annual subscription fee for those who order directly from APDA. *APDALog*, APDA's previous magazine and catalog, will be replaced by the new 144-page *APDA Tools Catalog.* Current customers will receive a copy of the new catalog automatically; new customers may request a copy from APDA, Apple Computer, Inc., 20525 Mariani Avenue, M/S 33-G, Cupertino, Calif. 95014, 408-562-3910.

To order pre-release products through APDA, customers need to sign and return a Terms and Conditions form found in the catalog.

With the Apple IIe Card out there, there's a new twist on identifying Apple II models. For most users who just need to find the system type and ROM revision, the following table (from information in the May 1991 revision of Apple II Miscellaneous Technical Note #7) gives the ROM identification bytes:

Machine/ROM rev.	$FBB3	$FB1E	$FBC0	$FBDD	$FBBE	$FBBF
Apple II	$38		($60)			($2F)
Apple II Plus	$EA	$AD	($EA)			($EA)
Apple III (emulation)	$EA	$8A				
Apple IIe	$06		$EA			($C1)
Apple IIe (enhanced)	$06		$E0			($00)
Apple IIe Card	$06		$E0	$02	$00	
Apple IIc	$06		$00			$FF
Apple IIc (3.5 ROM)	$06		$00			$00
Apple IIc (Org. Mem. Exp.)	$06		$00			$03
Apple IIc (Rev. Mem. Exp.))	$06		$00			$04
Apple IIc Plus	$06		$00			$05

The non-parenthetical values are sufficient for identification. Values in parentheses are for comparison; you might consider them "additional confirmation".

For the IIgs, check the identification bytes for the enhanced IIe; if they match, you need to use a ROM-based routine which determines whether the machine is a IIgs and also returns detailed system identification information. (To just detect the IIgs, do a SEC and JSR $FE1F; if the carry is clear when $FE1F returns, you're on a IIgs.).

As an example of narrowing things down from a BASIC program:

```
1000  REM == ID Apple II system BASICally ==
1010 I1 = 64435: REM $FBB3
1020 I2 = 64286: REM $FB1E
1030 I3 = 64448: REM $FBC0
1040 I4 = 64477: REM $FBDD
1050 I5 = 64446: REM $FBBE
1060 I6 = 64447: REM $FBBF
1070  IF  PEEK (I1) = 56 THEN ID$ = "APPLE II"
1080  IF ( PEEK (I1) = 234) AND ( PEEK (I2) = 173) THEN ID$ = "APPLE II PLUS"
1090  IF ( PEEK (I1) = 234) AND ( PEEK (I2) = 138) THEN ID$ = "APPLE III (EMULATION)"
1100  IF  PEEK (I1) < > 6 THEN  GOTO 2000: REM not IIe/c/IIgs
1110  IF  PEEK (I3) = 234 THEN ID$ = "Apple IIe (not enhanced)"
1120  IF  PEEK (I3) < > 224 THEN  GOTO 1200: REM not enhanced IIe-type
1130 ID$ = "Apple IIe (enhanced)"
1140  IF ( PEEK (I4) = 2) AND ( PEEK (I5) = 0) THEN ID$ = "Apple IIe Card (Mac LC)": GOTO
      1400

1145  REM == next is test for IIgs ==
1150  FOR I = 768 TO 782: READ X: POKE I,X: NEXT I
1160  DATA 169,0,141,14,3,56,32,31,254,110,14,3,24,96,0
1170  CALL 768: IF  PEEK (782) < 127 THEN ID$ = "Apple IIgs"
1180  GOTO 1270: REM skip IIc stuff

1200  REM == come here if possibly IIc ==
1210  IF  PEEK (I3) < > 0 THEN  GOTO 2000
1220  IF  PEEK (I6) = 255 THEN ID$ = "Apple IIc (original ROM)"
1230  IF  PEEK (I6) = 0 THEN ID$ = "Apple IIc (UniDisk ROM)"
1240  IF  PEEK (I6) = 3 THEN ID$ = "Apple IIc (first memory expansion ROM)"
1250  IF  PEEK (I6) = 4 THEN ID$ = "Apple IIc (second memory expansion ROM)"
1260  IF  PEEK (I6) = 5 THEN ID$ = "Apple IIc Plus"

1270  REM == print answer ==
1280  PRINT "This computer says it is an ";ID$;"."
1290  END
2000  REM == come here if we're confused ==
2010 ID$ = "Can't identify. You're sure this is an Apple?"
```

Apple II Miscellaneous Technical Note #2 contains a complete assembly language routine for identifying Apple II family configurations.

Meanwhile, Apple IIe Technical Note #10: The Apple IIe Card for the Macintosh LC, covers differences between the IIe Card's environment and that of an actual IIe. There are eight pages of information, but among the items of general interest:

The IIe Card's control panel allows the user to vary the keyboard parameters delay until repeat, repeat rate, and type ahead. Programs that can't deal with the various settings (which are also available on the IIgs) need to notify the user of possible conflicts in their documentation. Also, you can't make assumptions about the keyboard layout since there are various ADB (Apple Desktop Bus) keyboard layouts available and you don't know which may be attached to the computer. Finally (as you might have guessed) the single-wire shift-key modification is not implemented in the IIe Card's hardware.

The IIe game port signals on the IIe Card's 9-pin adapter are restricted to those available on the similar 9-pin game connector on the back panels of the IIe. Additional signals available on the IIe's internal game port are not implemented.

Also as no surprise, there is no cassette port and associated softswitches are absent. Also, two other IIe softswitch functions are missing: $C07E (IOUDIS when written to, RDIOUDIS when read from) and $C07F (IOUEN; reading data from this location may be possible in future IIe Card revisions). The VBL detection softswitch at $C019 functions at its normal 60 Hz, but since it is not synchronized with the Mac video using VBL detection may not result in flicker-free displays.

Some of the Apple II monitor routines respond differently, but most differences are minor. Two exceptions affecting timing: the duration of the BELL routine is no longer nominally 0.1 seconds (it uses the Mac bell generation, which can vary pursuant to user preferences), and the PREAD routine does not compensate for the fast (1.9 MHz) processor speed. (With the proliferation of accelerated Apple II systems, these problems are not confined to the IIe Card.)

Also, there is a "trap" mechanism involving access to certain memory locations (the display memory except for the screenhole RAM, the slot I/O address space, the annunciators, the paddle reset trigger, and the display softswitches unless the slot 6 disk motor is on) to allow the IIe Startup program on the Mac LC to interact with what's happening in the IIe Card. When the "trap" occurs, IIe Card operation is (very) temporarily suspended while the IIe Startup application handles things; this can also affect timing.

There are also example programs for controlling the IIe Card's speed from software and notes on the "virtual devices" which can be assigned to the IIe Card's "slots".

Most applications should run fine on the IIe card, but if you are developing an application that doesn't seem to operate correctly then it's time to check out the details in this technical note.

KingWood Micro Software, a small company emphasizing AppleWorks accessories in their product lines, is moving. Their new address is KingWood Micro Software, attn: B. Cadieux, 2018 Oak Dew Drive, San Antonio, Texas 78232 (effective September 1, 1991).

The company also sends out a product newsletter, ***Texas II***. For a catalog and a year's subscription to the newsletter (including back issues), send $1 to KingWood Micro Software. The $1 is refundable with a product purchase.

Apple II software from the American Institute of Small Business takes you step-by-step through 221 questions relating to the establishment and evaluation of a business plan. Answering the questions allows the program to assess whether to enter a line of business, or whether to expand or diversify an existing business. The program determines its response by evaluating the responses to questions regarding the user's personality, educational and vocational history, and financial background.

In addition, the program prepares a business plan for the user with narrative and financial documents including balance sheets, profit and loss, and cash flow statements. A completed sample business plan is also presented.

The package sells for $125 from The American Institute of Small Business, 7515 Wayzata Blvd., Minneapolis, Minn. 55426, 800-328-2906

(FAX 800-545-7020).

"Our name in vain" department. A rumor has been circulating about a possible new IIgs. Rob Barnes has confirmed this rumor at KansasFest (he confirmed it as being absolutely ***false***), but the other disturbing aspect of the rumor is that our name was associated with it through the guise of "an ***A2-Central*** editor (Chris)".

Those of you who read the newsletter regularly know that the name "Chris" has never appeared in the masthead (that box on the back page where we put most of the publication's legal and production information). We don't know where the rumor got started, but we'd like the record to be set straight that ***A2-Central*** had nothing to do with it. Even though the rumor is false, our good-faith relationship with Apple depends on Apple's confidence that we will not "leak" confidential information if and when we do have it. What we can say usually makes it into our publications when appropriate.—DJD

JEM's *TotalControl*

The good thing about AppleWorks has always been that it does many basic things well. The beautiful thing is that if it doesn't do precisely what you want, someone often comes around with a way to do it.

AppleWorks's database has been its strongest feature not because it was inherently "powerful" but because it was easy for most users to understand and apply to their problems. The database module worked on a simple "flat file" model (all information kept in a single file of records) that suited the needs of most users.

Users wanting to generate reports using relational methods (the capability of importing and exporting information from several data files working in combination) were given some help with *TimeOut ReportWriter*. But some users also wish to add a "forms" approach to organizing their data where the methods of data entry and lookup can be tailored to a specific application. JEM Software's new *TotalControl* patches the AppleWorks 3.0 database to add this capability.

TotalControl installs by running the BASIC "STARTUP" program, choosing the "Install TotalControl" option, and telling the program where your copy of AppleWorks is located if it isn't in the root directory of an on-line volume. If you use 5.25 disks, you'll also be asked to add the SEG.WP (word processor segment) file to your AppleWorks program disk since *TotalControl* can import data from word processor files.

TotalControl* adds several new commands to the database.** The first and most important is the Option (open-apple-O) command to allow setting ***TotalControl options for the database you are currently using.

The main option menu contains the following: *Modify rules* lets you redefine any rules for entering data (including entry masks, importing of data, or formulas) that have already been set; *Define new rules* allows creating the rules anew; *Cancel rules* lets you...ah, you know; *Change categories* lets you pick a category (field) to define without having to exit the options menu; *change lock status* lets you lock (prevent modification) or unlock the contents of a field, *Release TC memory* causes *TotalControl* to unload its segments from RAM in case you need more AppleWorks desktop space (if you do this, *TotalControl* will have to reload its segments when it is called into play again); *Set preferences* allows you to change some aspects of *TotalControl*'s operation (whether imports are case sensitive, whether you can edit fields controlled by formulas, whether to require a file to be loaded along with the database currently being used, and so on) to suit you; and *Print all rules* lets you dump a printout of the rule definitions to your open-apple-H (hardcopy) printer.

Under *TotalControl*, a database field (category) can be defined to contain text only, numbers only, characters determined by a *mask* (a formatted entry field, such as "xxx-xxx-xxxx" for a U.S. area code and phone number), information taken from a *glossary* , data *imported* from another database or spreadsheet on the desktop, or a *formula*.. You can also define whether information in the field is of a particular case (as entered, UPPERCASE, lowercase, or Each Word Capitalized) and limit the minimum and maximum number of characters that can be entered.

Using *TotalControl* is a matter of creating the data files you wish to use and then entering the rules you want to connect the files with. Rules can be the entry restrictions or masks you want to use to allow data entry in a field, formulas to enter data into calculated fields, or the relationships between files used to import data.

As an example of a data input mask, assume you wanted to force the entry of a valid (U.S.) phone number into a field. You bring up the "Options" menu with open-apple-O, choose "Modify Rules", and select "Mask". Then you work through the options: "Right" or "Left" justified (we used "Left"), Auto-Return (we picked "yes", which causes a return to be issued automatically when the mask is filled), and "Must fill" (we set this to "yes", which means the mask must be bypassed or completely filled before you can leave the category). Finally, we define the mask itself to consist of three numeric characters, a dash, three more numeric characters, another dash, and four more numeric characters (in addition to "numeric" characters, you can also specify "text" or "anything" characters; any literal characters you type will appear as formatting characters). A printout of the mask looks like "###-###-####" (on screen it looks different as MouseText characters replace the "#" characters).

When you enter that field of the database record, the mask will appear and you enter your values.As you type, *TotalControl* will sound the error bell if you try to enter an illegal character (like typing "s" for a numeric character) and will skip over the formatting characters. If you don't fill the "Must fill" entry before hitting "return", *TotalControl* will also beep and make you re-enter the category's data..

Glossaries are suitable for creating lists of possible entries for a field, and also for doing lookups to fill an entry from a second database. For example, *TotalControl*'s sample files include a "States" database file which contains the full names of the 50 states followed by their postal abbreviations. You can have a category in another database automatically filled with the correct abbreviation by giving the category a glossary rule to create a list from the category "Name" (the state's full name) in the "States" database and return the contents of the field "Abbrev" (the postal abbreviation) for the name selected from the list. When you move to the category to be filled the identifier "Glossary" appears in the middle of the lower AppleWorks status line; that tells you you've entered a field defined to use glossary rules. You can then type open-apple-G to bring up the glossary list; assuming the "States" file is available on the desktop a list of the contents of the "Name" field for the selected records (in the rules, you can select a numbered range used to build the list) will appear at the right edge of the screen. You can scroll through this list to pick the state you want; *TotalControl* will look up the corresponding "Abbrev" (abbreviation) entry and insert it into the category.

There is also a lookup feature in the glossary. If you enter the glossary field and type a sequence of characters followed by a return, *TotalControl* will scan though the specified range of "Name" fields of the second database until it finds the first matching entry (matches can be on the complete name or only on a unique pattern at the start of the name; you decide). This way, you can type "alab" and the abbreviation for Alabama (AL) will show up in the category.

Import allows you to export data from another database or spreadsheet file on the desktop to the current file. You'd normally use import to fill several other categories based on a key expressed in a category.

Imagine your products are coded with a product number that uniquely indexes their appearance in a second database, which also contains the (human readable) product name, price, and possibly other information. Once the product number is entered in a category in the first database, *TotalControl*'s import feature can be used to load the product name and price from the second base automatically as a way of facilitating product orders.

The word processor lookup allows you to create a long description in a word processor document (also on the desktop) linked to a key category. When you issue the open-apple-W command with the cursor in that category the corresponding long description will be displayed.

Of course, these features can be combined and interwoven between sets of files.

The weaknesses of *TotalControl* are the weaknesses of AppleWorks's file limits, plus a few others. The Glossary command is limited to use database lists of about 2000 records. Exporting data from the "controlling" database to other files is not automatic; you need to move to the other file and define an import (or glossary) operation and update the file from within itself. *UltraMacros* can go a long way in automating the control of several interactive files.

If you need an actual report generator, *TimeOut ReportWriter* is still the solution. But for $60 (retail), JEM's *TotalControl* opens the door to turning AppleWorks into a credible forms system for the home or small business. JEM Software, 7578 Lamar Court, Arvada, Colo. 80003.—DJD

Ask (or tell) Uncle DOS

Within moments of the time we sent the August issue to the printer, Mark Munz from Beagle Bros called to let us know he had checked with Alan Bird, author of the ***Beagle Compiler****. Mark said Alan confirmed that checking the contents of zero page location $B7 is the recommended way to see if the BASIC program is running as (****Beagle Compiler****) compiled code, so our example in "Beagle Compiler and "&""will stand up. Mark mentioned the way to check from BASIC is to see if PEEK(183) equals 172; if so, assume the program is running compiled.*

Also, in the same article, there is a complete word missing from the third sentence of the second paragraph of the reply. It should start "Checking that location will verify...".

What next?

Regarding your outstanding article in the August ***A2-Central***: I'm a nonprogrammer (except for a few months of PEEKing and POKEing around) who enjoys controlling some of the power of the IIgs through hypermedia that I'm compelled to learn to program within a more "traditional" sense. What can the high level languages offer that current hypermedia either doesn't now or won't in the future?

Bill Lynn
Northfield, Conn.

With any language you're constrained to using the command set that's available. It's hard to quantify, but basically hypermedia and the currently associated programming languages are designed to allow users to write certain application-style programs. You can use hypermedia programs to create presentation graphics, address books, educational software, and so on. These types of programs tend to fall within the realm of the "construction set" that hypermedia offers you: visual design and logical connection of elements such as buttons, fields, and so on.

There are other things to computer life than applications, though: things like system software, utilities, and so on. Traditional programming languages are, to varying degrees, more "general purpose". It would be difficult to write a printer driver in HyperTalk, for example, because the language is pretty well insulated from knowing low-level details of the computer hardware. You wouldn't write GS/OS in HyperTalk.

You might also find trying to simulate some types of applications using HyperTalk to be difficult because the language is based on a specific universe: a model of buttons, fields, cards, stacks, and so on. Designing a word processor based on this model would be difficult because a word processing environment has a more specialized purpose for which it needs to be. optimized without having to carry the broader-based hypermedia environment along.

Both ***HyperStudio*** *and* ***HyperCard*** *acknowledge that they aren't complete unto themselves and have provisions for adding external commands written in other languages. These external functions can actually take the form of applications themselves; maybe in the future a hypermedia "shell" will become the programming environment.*

Think of hypermedia as another development environment, but one that has stepped distinctly toward making program development accessible to ***anyone*** *who can think logically. Once you've gained experience in the logical basis of programming, moving to a new language becomes less of a hurdle.—DJD*

VCR Plus "nifty"?

My mother is one of those hardware illiterates, and she just got a *VCR Plus*. The problem is, she cannot figure out how to setup the *VCR Plus* either. Oh well.

Tim Tobin
Carson, Calif.

I read with interest your August 1991 tirade against the *VCR Plus* ("The People vs. Programming", page 7.49). It was a measured, rational tirade; well thought out—but methinks a tirade nonetheless. The *VCR Plus* is a nifty gadget, and admittedly a step away from user-friendliness; but a *change* in technology is not a *trend* in technology (witness the Apple III and the cordless IBM keyboard). This little box fills a niche where the human-computer interface is still mysterious; most of us understand phone numbers and zip codes, so this is a comfortable, if arcane, shortcut. When the technology for giving commands to our TV and VCR improves (be it by keyboard, mouse, or voice) the *VCR Plus* will fade into oblivion (and you'll get your programming highlights back in the newspaper).

In my lectures on approaching computers, I point out a two-box organizational chart: The top big box reads "YOU", and the bottom, smaller box reads "COMPUTER". I think you and I agree on that. However, in the American Instant-Fix culture, people do not see shortcuts as giving up any power. When we programmers make a natural-language shortcut, it will win; but we haven't for TVs and VCRs. So, we still have some work to do.

P.S. Thanks to all of your subscribers who have registered their copy of *Milestones 2000*. I received checks from ***A2-Central*** users on four continents, and a local elementary school for homeless children is $1000 richer. By the way, it has four IIgs's, no Macs, and no MS-DOSs.

Ken Franklin
Puyallup, WA

(Ken's ***MileStones*** *is a shareware program that was included on the April 1991 issue of* ***A2-Central On Disk****.)*

One has to ask the price of "ease of use". Telephone numbers are a good example; it would be easier to be able to dial a friend by name than to add the additional layer of correlating an obscure number with the name. There are devices for the phone system that let you do this, which is a positive step. Devices that take the information content in the reverse direction are counterproductive if they become "standards". It is obviously in the financial interest of the manufacturers of the ***VCR Plus*** *to see that happen with their solution, and it appears they've had some effect already.*

To put it closer to a computer user's perspective, the question is whether you would use a spreadsheet program that required you to enter "35" instead of "@AVG" for a function name (if "35" were a "token" for the formula). Obviously, that would not have been a good start toward a "standard", and though it did not become one it would have been reasonable to be critical of a company that attempted to make it one. With the ***VCR Plus****, we are similarly tied to entering numbers whose meaning can't be obvious to the user.*

Although I see the advantages of more obscure computer languages like C and assembly language, I celebrate languages like BASIC and HyperTalk because they allow many users to understand and solve their own programming problems. Surely methods that release the power to average users are better than those that create a new type of "priesthood"...—DJD

Slower ProFile

I am looking for a disk optimizer. I own an Apple IIgs with one megabyte of RAM and an old 5 megabyte ProFile drive. Lately, the ProFile drive response is becoming sluggish. It takes a long time to boot up and the screen refreshes whenever you click on a folder takes longer. Can you recommend a low-cost disk optimizer and backup/restore package for my Apple?

A. Mahmood Merican
Singapore

We'll talk about optimizers first, but stay tuned: your problem may indicate something more dangerous than fragmented files.

There are two major sets of disk utilities for the IIgs: ***ProSel-16*** *from Glen Bredon (521 State Road, Princeton, N.J. 08540) and* ***Salvation*** *(now* ***Salvation Bakkup****) from Vitesse (13909/2A Amar Road, La Puente, Calif. 91746, 818-813-1270). We roughly compared the backup programs in "Backup Only", March 1990, pp 6.14-15).*

Both companies also have optimizers; ***ProSel-16****'s is built-in as part of the utilities package; Vitesse (which sells the* ***Salvation*** *modules individually, or as a specially priced* ***Salvation Supreme*** *package) sells its package as* ***Renaissance****.*

The purpose of an optimizer is to arrange the individual blocks making up the files on a disk so that the blocks belonging to each file tend to lie in physical proximity to each other. This allows the disk head to make fewer passes across the surface of the disk media to read a file, which makes file access quicker. Although you might think a simple file-by-file backup and restore operation would "optimize" the disk, and optimizer can do further fine-tuning of the disk organization.

ProSel-16*'s optimizer has two modes: "complete" and "turbo". Both modes work through the disk locating the blocks used by each file. Then, starting at the beginning of the storage area of the disk (the lowest block numbers available for file storage once the volume directory and bit map blocks have been assigned) each file's blocks are rewritten in sequence (****ProSel-16*** *actually does this in* *stages; handling the larger "tree" files last). The result is a disk where each file is contiguous on the disk and its blocks can be read sequentially.*

Before optimizing the files, the "Complete" mode will also group all directory blocks and sequence them at the very beginning of the disk. This puts all directory entries within one set of contiguous blocks, so that when issuing "catalog" or directory management commands all the relevant directories will be as closely grouped as possible.

***Renaissance**'s optimizer also works its way through the disk making the files contiguous, but does not group the directories at the beginning of the volume. The major cosmetic difference between **Renaissance** and **ProSel-16** is that Vitesse's products use the Apple Desktop user interface, while **ProSel-16** sticks to a more traditional text-menu interface.*

*With either product you **must** have a backup of your hard disk before you optimize; if the optimization should fail (due to a disk error or other problem) before completion the disk may be left "scrambled" between the original and optimized form.*

Optimization may help performance if the files on your disk exist mostly as isolated fragments (hackers say the disk is highly "fragmented"). But if the disk also takes a while to spin up to speed and boot, it may indicate a hardware problem in the drive itself. Such a problem might be a failing power supply in the drive (makes it harder for the drive mechanism to achieve and maintain constant speed), or a slow failure in the disk's low-level formatting that is causing the operating system to have to make several attempts to re-read data blocks. If the disk performance doesn't improve after optimization, it may be time to have the hardware checked.

Normally we'd suggest trying a low-level format of the hard disk to attempt to cure the latter problem but we don't know of an easy way for a user to reformat a ProFile.—DJD

AppleWorks tips

I purchased an Apple II Plus back in 1979. It was "loaded", and I was impressed with all that it could do. Later I purchased a IIe, then a second IIe. Today the kids play with the II Plus; I use the two IIe's in my business. Apple wants me to buy a Mac. I've considered upgrading to the IIgs, but have decided to buy nothing new for the following reasons:

- Apple Computer's total lack of commitment to the Apple II line.
- The "marriage" between Apple and IBM may make *everything* obsolete.
- My IIe (with a Zip Chip) is faster than a IIgs.
- With AppleWorks 3.0, *TimeOut*, and the IIe's, I can do everything that I need to do.

Here's a setup which may interest some readers. I have my IIe's connected to a VAX 4000 mainframe at the office, which allows printing from AppleWorks on a very fast (20 pages a minute, a page every three seconds!) Talaris laser printer, using any and all of its fonts.

For those whose office uses a laser printer hooked up to a mainframe you may want to try it. Set up a new printer in AppleWorks, choose the "print to disk or another Apple" option, and enter the necessary control sequences for the printer (found in the laser printer manual). When the document is ready for printing, AppleWorks 3.0 asks for the path name and prints a text file on the disk (or RAM disk) complete with the sequences necessary for the laser printer.

For home I use a modem to dial the mainframe and from my desk at the office I use a Super Serial Card (19,200 baud!) connected directly to a server port on the VAX. For communications I use Kermit with VT100 emulation (downloaded from CompuServe). Upload the AppleWorks text file (containing print control sequences) to the VAX (Kermit must also be installed on the VAX) as a text file, and with a simple VAX PRINT command the file will be sent to the laser printer with beautiful results (this letter is an example).

In setting this up, I discovered some interesting things about AppleWorks:

- after printing the last page of the document, AppleWorks automatically reverts back to 10 characters per inch (CPI). If there is a 10 CPI printer code installed in AppleWorks, AppleWorks sends this code sequence after the last form feed to reset the printer to 10 CPI, which may result in the printing of a blank page. I leave 10 CPI set to nothing to avoid this.
- if there are no CPI printer codes entered in the printer setup, AppleWorks will not do page breaks correctly for various CPI settings (other than 10 CPI). Use a non-printing character (such as Ctrl-G) in the CPI setup if you plan to use more than one font for a given CPI. Since the laser printer has many different fonts, do not set up CPU commands to choose a font, but the control sequence for the font in the document itself (like at the beginning of the document). Control characters are easily entered in AppleWorks 3.0 by using a "special code" with the necessary control sequence followed by whatever is necessary for a particular font after the special code caret ("^"). This allows selection of any font as well as a change of fonts within the document.

Terrell Smith
Madison, Wisc.

Basically, the best reason for staying with the IIe is exactly the last reason you gave: if it does what you want it to, why change? (We'd consider the other reasons to be debatable, but let's move on...)

The need for a 10 CPI printer code may explain why we sometimes get letters from readers wondering why their pages don't "break" at the right place. We use ImageWriter (or compatible) printers here, so any problems we have are usually direct artifacts of the program; we don't have to customize settings.

Of course, we should all remember that AppleWorks has to place calculated page breaks by counting the lengths of lines, and without knowing how many characters are to be printed on a line it has no way to figure the accurate count. If you want the program to be "smart", you have to give it enough information to do your thinking for you.—DJD

No scroll

A quick Applesoft programming question: I have a menu that ends with an input statement on the twenty-fourth line of the screen. If the user inputs an error, the program goes to a line that just CALLs -868 to clear the line and resumes at the twenty-fourth line via VTAB 24.

What I'm trying to do is have it do it **without** scrolling up one line after the input. If I put it on line 23 it works, but as soon as I hit return on line 24 it scrolls. I have to use return rather than GET or a one character PEEK because the menu choices go up to "10" (two characters). Any help would be appreciated.

Daryl Morgan
Turlock, Calif.

*You can't really stop the scroll from occurring since you've turned the screen control over to the Apple firmware (via Applesoft's INPUT statement). When you hit <return> on the bottom line, the screen **will** scroll; that's just the way the firmware works.*

There are several possible workarounds. You could reformat the menu items (if possible) to use less than 24 lines (maybe you can do two columns of five if the items are short, for example). You could rewrite the menu handler to use some other mechanism; I've always liked the ones where you use the cursor keys to highlight an item and then select the item with the "return" key.

*The quickest fix is to change the text window to allow **only** the last line of the screen to scroll. Using the **Applesoft Programmer's Reference Manual** for details of the text window storage in zero page, something like:*

```
<get to where you want to use bottom line)

1000 TP=PEEK(34):REM get and save top edge
1010 BM=PEEK(35):REM get and save bottom edge
1020 POKE 34,23:REM "top" is line above line you want to
     use
1030 POKE 35,24:REM "bottom" is bottom line (24)
1040 VTAB 24:REM better move cursor to window!

<do your question in here…when you're ready to "unfreeze"
     upper display, then…>

1100 POKE 34,TP:POKE 35,BM:REM don't forget to fix window

<on with the program>
```

should work; the screen still scrolls when you hit "return", but since the scrollable area is only one line high you only have to rewrite the prompt line, not the whole screen.—DJD

SuperWorks "wish list"

In the last issue Rick Pedley asked about *SuperWorks*. I spent thirty days trying to like the program, however it's a case of "Nice try, but..."...

The programmers made a real attempt at translating the AppleWorks integration that we all love to the messy-DOS environment. All the commands work exactly right, the screens look almost like "home" and the desktop metaphor is identical. The only problem is that it's AppleWorks 2.*x* that has been translated.

The word processor is the best module; it is good enough for almost any small business application. **But** the translation is not perfect: Control-L and Control B- do the "underline" and "bold" printer options but Control-C doesn't "center". There is no spellchecker.

The spreadsheet is just okay. Missing are the vital (to me) block operations and *TimeOut SpreadTools*. There is 3-D capability, but only for files on the desktop—there's no way to access disk files.

The database is extremely limited as was the AppleWorks database before *ReportWorks*. I have always used *DB Master* as my database, regularly exporting files to AppleWorks database files to use as a quick desktop reference. In 1991 a business that relies on a flat-file database is either duplicating a lot of effort or has little use for a database. In the latter case this program will suffice.

The macro capability is good, but not in the league of *UltraMacros*—but then, what is? I am not a big user of macros, but have developed some that really cut down on my workload. (For example, importing the file from *DB Master* is a one-key operation.) Without other *TimeOut* modules, especially *SuperFonts,* I feel naked, too, but I understand this omission.

The communications module is elementary at best. No scripting nor communications specific macros are available. This module is only really useful as an import tool. It is easy to import AppleWorks files to *SuperWorks* either by modem or, as I did, via Super Serial Card direct to the serial interface on my Zeos 386 notebook computer. You can also use files from other IBM programs directly without translation. There are two problems with this setup; first there is no way to return the files to AppleWorks when you are done with the IBM environment. Worse yet, there is no way to get files from *SuperWorks* to any other IBM format (*Lotus 1-2-3, dBASE, WordPerfect*) except via an ASCII transfer, and that does not meet my requirements.

Worst of all, the program is protected by a hardware protection key. This little monster looks like a null modem, but sticks into (and out of) your printer port. It only works if the printer is turned on—and is easily knocked out if you don't screw it in. That makes my notebook computer not fit in my briefcase, and dangerous on airplanes!

When I called the company for an RMA the young lady I spoke to only addressed the problem of the spellchecker—she told me they are trying to license one for the next *SuperWorks* incarnation—but had no information on the other drawbacks.

A2-Central™

© Copyright 1991 by
Resource-Central, Inc.

Publisher: **Tom Weishaar**
Editor: **Dennis Doms**

with help from:
Sally Dwyer **Dean Esmay** **Joyce Hammond**
Jay Jennings **Jeff Neuer** **Ellen Rosenberg**
Denise Shaffer **Tom Vanderpool** **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 six 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.

ISSN 0885-4017
Printed in the U.S.A.
GEnie mail: A2-CENTRAL
Voice: 913-469-6502
Fax: 913-469-6507

As I said before, "Nice try, guys, but it's not AppleWorks 3.0" and without the *TimeOut* series this program just cannot cut it in business. It might be useful to someone needing a way to cross over to IBM with minimal readjustment, but then all the really good stuff that is available to IBMers would be locked out.

Derek Sutton
(via FAX)

*Tom Vanderpool received **SuperWorks** and demonstrated it to us in the office. We were impressed with the similarity to AppleWorks, but agree with the reservations about the current lack of add-on enhancements and the hardware protection "key". Our primary interest in the product is that it is significantly more AppleWorks-like than other MS-DOS products for Apple II users who may also find themselves working on MS-DOS machines.—DJD*

More power to you

When does an Apple IIgs need a heavy-duty power supply like Applied Engineering makes? I've got an Apple IIgs with 1.25 megabytes, and a 5.25 controller card in slot 6, with two 5.25 drives and two 3.5 drives, as well as an Epson printer.

Jack B. Powell
Beaver Creek, Ohio

Usually, power supply problems will evidence themselves in several ways. If you start having problems that you can't trace to a specific item of hardware or software (like crashes during "Reset", or intermittent disk errors not traceable to another source), suspect the power supply might be a culprit.

The first solution to try may be a fan (internal or external); cooling the interior of the computer lowers the demands on the power supply. Electronic circuits become harder to push electrons through as they get warmer, so the hotter the computer runs the harder the power supply must work (that's why Apple recommends a fan if you have two or more peripheral cards installed in your IIgs). If you wonder whether cooling might help your system, try setting it somewhere where you can operate the IIgs with the cover off (be sure to put it someplace where nothing can fall in it!) for an average work day; if formerly chronic problems disappear then a fan is probably in order.

Peripheral cards draw power from your computer, as dô the devices they power, so each of your drives puts an additional load on the supply. Four is within the limits Apple recommends, but you might want to consider looking for a cable to daisy-chain the 5.25 drives off the back of your 3.5's and therefore eliminate the 5.25 controller card, which draws power itself. (The internal 5.25 circuitry is already drawing power; you might as well get your money's worth out of it.)

Peripheral devices which have their own power cables, such as printers and external hard disks, don't draw much (if any) of their power from the computer. Outside of any slot-based interface card they use, these devices's drain on the system should be negligible.

*As always, the answer is "if it ain't broke, don't fix it". We have systems here with four megabytes of memory and several peripheral cards that work fine with the standard IIgs power supply and a fan. If you have high-demand cards, like more RAM, a **PC Transporter**, or a **TransWarp GS** you may want to consider the heavier supply as an option, but it may not be necessary.—DJD*

Modem–to–modem

Is it possible to transfer data from one computer to another via modem without a telephone line? I would like to interconnect several computers via telephone wiring using two conductors, but without the need for an actual telephone-to-telephone connection.

Elonard Lanigan
Browns Valley, Calif.

We've transferred files between two modems with a phone wire connection, but without connection to the phone network itself. You need two computers, each with a modem and communications software. Connect the modems to the computers as normal. (Most modems are Hayes command set compatible, so we'll use those commands in our examples.)

Most modems have two phone jacks; one to connect to the line that brings phone service into your house (usually marked "Line" on the modem) and one to allow "daisy-chaining" a phone to the modem (usually marked "Phone" on the modem). We connected the modems using the "Phone" connection on the first and the "Line" connection on the second. We also disconnected the first system from the phone service (either unplug the service line from the modem, or use a line cutoff switch available from Radio Shack).

If one of the communications programs supports a "host" mode that allows others to call your computer from a remote location, activate the host mode and set up the communications parameters (baud rate and so on) on the host. Set up the communications software on your "remote" computer as if you were going to "dial" the host computer.

Some modems will attempt to detect a dial tone before dialing. Since we're not on a normal phone line you'll need to disable that detection on the "remote" system; one Hayes command to do this is "AT X0". (See your modem manual for details on the supported command set.)

Now you can dial in from the "remote" system. Enter "AT D" (no phone number!) and it should activate the phone line and wait for the other system. The "host", which has been waiting, should "pick up", issue a carrier, and start communications with the "remote" system.

You'll be limited to the baud rate of the slower of the two modems, but the connected systems should act just as they would if you "dialed up" over the phone system. We have used this method to transfer smaller files between computers; for larger files, sometimes it's quicker to actually connect them with a null modem cable so that the maximum baud rates of the serial ports can be used.

You would be limited to two computers using this connection at any time (that's the way modems work), though you might be able to "chain" several computers along the same line (we've never tried connecting more than three in a row). This is not a replacement for a network, but it may be a serviceable way to share smaller files.—DJD