

A2-CentralTM

formerly
Open-Apple

June 1991
Vol. 7, No. 5

ISSN 0885-4017

newsstand price: \$2.50

photocopy charge per page: \$0.15

A journal and exchange of Apple II discoveries

Apple releases "The Apple II"

The May 1991 Apple User Group mailing includes a paper titled **The Apple II**. The first paragraph of the introduction is the kind of statement Apple II users like to see from the company: "No company is more closely identified with the birth and growth of the personal computer industry than Apple Computer, Inc. As the company's first flagship product, the Apple II set a standard and has since played a leading role in personal computing, becoming the longest-lived personal computer line available on the market today." Now *that's* a thought you can base an advertising campaign on.

The paper lists comments from several educators regarding the utility of the Apple II, comments regarding the original vision of the Apple II as created by Wozniak and Jobs, a timeline of the Apple II's evolution from its birth in 1977 to the introduction of HyperCard IIgs at the end of last year (strangely, AppleWorks was missing from the event list), a listing of the product family (currently, that is the Apple IIe and IIgs) and their features, a list of support organizations inside and outside of Apple (user groups, the new Apple II Bulletin Board on AppleLink, APDA, Apple's Software Guides, and Apple's Office of Special Education), a list of some innovative uses of the Apple II, and a section on Apple II and Macintosh coexistence which contained the statement "Apple's customers are finding that both the Apple II and Macintosh computers provide excellent value, and that in environments calling for multiple computers, both products work well together for complimentary tasks." (We can attest to that ourselves).

This is the type of mature marketing approach that Apple II users would like to see taken to the mass media. Let's hope *The Apple II* is an omen of things to come. Soon.

A newsletter for Apple II sysops. "Sysop" is the abbreviation for the title "system operator", a person responsible for administering a computer bulletin board system. Douglas Gramzow is offering one free issue of a new bi-monthly publication for Apple II sysops. The first issue is to be published in June, the cost will be 6 issues (1 year) for \$10, 3 issues for \$6, or one issue for \$2. Write to *II Sysops*, c/o Douglas Gramzow, P.O. Box 720, Eldersburg, Md. 21874, 301-549-2584.

Asphyxia is a monthly Apple II publication distributed by Thomas Mavroudis; the first issue was seen being distributed via on-line services. If you'd like to know more about *Asphyxia* or feel you have something to contribute, write to *Asphyxia* care of Thomas Mavroudis, P.O. Box 592, Syosset, N.Y. 11791, America Online T.MAVROUDIS, GENIE T.MAVROUDIS1.

The Communicator is an Apple II newsletter compiled by educators concerned with information regarding aid for the visually impaired. It's published every other month at a subscription price of \$10 per year from *The Communicator*, Rt. 4, Box 263, Hillsville, VA 24343-8047.

Want some BASIC Help? This new program from InSite Software provides a "help" command for Applesoft (ProDOS) that brings up a menu page with 18 categories of help ("Applesoft Errors", "Graphics", and so on). Selecting a category takes you to a series of information screens containing commands, general information, and tips that you can page through. When you've found the item you need, make a note and escape back to the main menu for more searches or to Applesoft to continue programming.

Can't remember which HCOLOR number corresponds to green? It's here. Need to know a CALL to wait for a keypress? It's here. Not everything in the Applesoft manuals is here, of course, but it may save you from stopping and hunting for a manual when you'd rather be programming.

One of the menu options is a search function that lets you specify an Applesoft keyword or text to search for in the help screens. Once the item is found, you can jump to the next occurrence or the next page with an occurrence, quit to perform another search, or "drop out" of the search to page through the current section.

Another item will report information about the BASIC program in memory, including current HIMEM: and LOMEM: settings, the location of variable tables, program size, and free space.

BASIC Help did load and run with *Program Writer's Editor.LC*. (You run *BASIC Help* first, then *Editor.LC*.)

This looks like a very useful program for those who are trying to learn Applesoft interactively or who (like ourselves) can't always put their hands on the manual when they need to check the use of an Applesoft command; a sort of on-line "PEEKs and POKEs" chart. *BASIC Help* is \$29.95 plus \$2.50 shipping (Arizona residents add 5% sales tax) from InSite Software, P.O. Box 3778, Flagstaff, Ari. 86003-3778. It requires an Apple IIe, IIc, or IIgs with at least 128K of memory and an 80-column display.

Technology and Learning lists several Apple II programs among the best of educational software for 1990. Among the top seven were four programs available for the Apple II: AppleWorks 3.0,



"...AND YOU'RE CLAIMING THAT THE SOURCE CODE USED FOR THE MICROSOFT CORP. GUI, WAS ACTUALLY AUTHORED BY YOU AS THE GUMBY USER INTERFACE?!"

National Geographic Kids Network; Hello! and Acid Rain (Ilgs), GTV: A Geographic Perspective on American History (Ilgs and Macintosh), and Scholastic Software's HyperScreen (Apple II and MS-DOS).

Among other Apple II programs listed in an additional list of 29 Awards of Excellence: Math Blaster Mystery (Davidson and Associates), Data Insights (Sunburst Communications), Transformations (Sunburst), Hop To It (Sunburst), Writer's Helper 3.0 (Conduit), Spell It Plus! (Davidson), Revolution '76 (Britannica Software), Time Patterns Tool Kit (Tom Snyder Productions), The Playroom (Brøderbund

Software), Bannermania (Brøderbund), Color 'N' Canvas (Wings for Learning), Peterson's Financial Aid Services 1990 (Peterson's Guides), S.M.A.R.T. Choices (Tom Snyder Productions), First Categories for the Apple Ilgs (Laureate Learning Systems), Just a Little Lie (Sunburst), Reading Realities, Elementary Series (Teacher Support Software), Where in Time is Carmen SanDiego? (Brøderbund), USA Geograph (MECC), International Inspirer (Tom Snyder), Playing with Science: Motion (Sunburst), Second Voyage of the Mimi (Wings for Learning), and The New Print Shop (Brøderbund).



Ask (or tell) Uncle DOS

Just to reassure anyone who needs it, every issue of **A2-Central** is read by at least three people. Still, mistakes creep past (normally unintentionally) and for those Murphy doesn't reveal to us right after the issue has been mailed we depend on our readers to point out any problems.

One of the people I usually try to get a copy of the newsletter to is Matt Deatherage, who normally works for Apple II Developer Technical Support but occasionally spends his own time reading our material for technical errors. Being close to Apple, Matt catches the types of things that we may not see as inherently evil but that may lead to misunderstandings or occasionally outright misconceptions. Unfortunately, given our abnormally short lead time sometimes the newsletter doesn't make the cycle from the editor to Matt and back in time. I wanted to acknowledge Matt's "silent" help, the type of unselfish contribution to the quality of the Apple II that many people at Apple make. And also I wanted a sneaky lead-in to a few comments Matt made about the May issue:

To simplify the issue of termination on the Apple interfaces: if you have a **Rev. C SCSI** interface with **one device**, that device should be terminated. If you have **more than one device**, the **first** and **last** device connected to the interface should be terminated. If you have the **High Speed SCSI Card**, then **only the last device** connected (for one or more devices) should be terminated. In each case, at least one of the devices on the bus should supply terminator power. (If you are using a High-Speed SCSI Card, you may find that an unterminated drive on a short (about two foot) SCSI cable may work; if the drive doesn't work on another card, don't forget to check the termination.)

Regarding the use of text page 2 on the Ilgs, the ROM 03 motherboard does support hardware shadowing of page 2; this is enabled when you select to enable the Alternate Display Mode in the Control Panel (on the older Ilgs, this option activated some code that simulated hardware shadowing by actually moving the page 2 data from fast RAM in the Ilgs bank

0 memory to the 1 MHz video display memory in bank \$E0).

Given that the answer to "Orphans abroad (and here)" regarding the UniDisk 3.5 driver for the original IIC motherboard said the overwriting of RAM memory used by AppleTalk won't be a problem on a IIC, we should clarify that that only means that AppleTalk code isn't likely to be residing there. It could still be a problem for Apple since that memory is reserved for expansion of the ProDOS operating system, and there might be other things Apple would like to do with that memory. Programs that intend to stay compatible with all configurations of ProDOS (a good thing for a ProDOS program to be) should not be using that memory. ProDOS 8 Technical Note #26 ("Polite use of Auxiliary Memory") does mention that **if /RAM is present** when your program starts up, the "reserved" areas above \$800 in auxiliary memory (including the language card space) are actually available. If /RAM is not enabled, that indicates system software may be using those "reserved" areas (for the same reason, you shouldn't re-enable /RAM if you find it disconnected when your program starts).

Finally, Matt pointed out that you have to be very careful in changing the access attributes on your AppleShare network volume if users may be booting over AppleShare ("Learn to 'share'", p. 7.31). The USERS directory carries a subdirectory named for each user that may be used for INIT files (in a SETUP folder), user information, and so on; access to this directory could be stymied by locking access to all folders except the user's personal folder.

Walker Archer of Quality Computers let us know there is a way to deal selectively with preventing AppleWorks users from getting lost on the network. John Link has written a new version of **SuperPatch** designed specifically for use with the network version of AppleWorks. Among other options similar to the new **SuperPatch 7.0** (\$34.95), it will allow canceling the use of the open-Apple-< and open-apple-> keys to move in and out of directories, thereby locking a user into a specific subdirectory. **SuperPatchNet** is \$79.95 (this includes a license to use the product on all the file servers in a single school) from Quality Computers, P.O. Box 665, St. Clair Shores, Mich. 48080, 313-774-7200 or FAX 313-774-2698.

James P. Davis wrote regarding "Questioned SANE-ity" to point out that (a) line 1250 on page 7.28 should have ended in GOTO 1390, (b) the "2^52" in line 1760 on page 7.29 should have been 2^52 (at least that was only a REM statement!), (c) the conditional in line 1780 should be "IF ((E) AND (NOT F))" (line 1780 and 1790 use the same test as listed), and (d) the "2^52" in lines 1780 and 1920 should be "2^52". To add insult to injury, I can't locate the original program file I used to generate that listing. Excuse me while I go flatten my forehead on a wall...—DJD

Keeping tab on HyperCard

I am very much indebted to you for your sample scripting on page 7.12 of the March issue. Up until then I had assumed that text could not be imported into a HyperCard field. I figured out that the text file involved must have a hard return at the end of each line so the "read until return" loop will copy one line to build each new card. There were 37 lines in my text file and when I tried to run your script all I got were 37 blank cards.

This did not surprise me too much as I did not understand the "offset (tab, it)" script (There's a lot about HyperTalk I don't understand.) However, I modified your script as follows and got the entire text into one card field:

```
on mouseUp
  ask file "File to import?" with "xxxxxxx"
  if it is empty then exit mouseUp
  put it into fileName
  open file fileName
  repeat with x=1 to 37
    read from file fileName until return
    put it into line x of card field 1 of card 1
  end repeat
  close file fileName
end mouseUp
```

Obviously this script is less sophisticated than yours as it is necessary to predetermine the number of lines in the text file. However, I do understand it and it worked for me.

All of the above is just a comment; I do have a question about HyperTalk which is driving me up a wall. If you create a card (for example) with 40 card fields, before any entries are made the tab key will take you through the fields in sequence according to their assigned number. However, when you add a script to a card field, the tab does not proceed to the next numbered card field. Example: into card field 14 you enter "44" and into card field 15 you enter 22, then in card field 16 you create a script which says "Put card field 14 + card field 15 into me". That part works fine, and card field 16 now contains the sum "66", but the cursor has now disappeared and the next time you hit "tab" on the keyboard nothing happens. If you hit "tab" again the cursor now appears in card field 1 instead of card field 17 where you wanted it. Perhaps there is some command you can put in the script for card field 16 which will make the cursor go to the next card field, but I haven't been able to find it. HELP!

Also, it is good that they included an animation feature in HyperTalk but it works so slowly between cards as to be ludicrous. As far as I can tell there is no way to speed up the animation(???)

Barney Woodruff
Camp Springs, Md.

SSSi is offering a special pre-release purchase price on Deskpak v2.0. Deskpak is a set of Ilgs New Desk Accessories that includes: SysFX (installs NDA's, CDA's Fonts, initialization files, drivers and CDEVs without rebooting, where possible; it also allows you to select which of these, except fonts, are "inactive"), FileTools (disk and file utilities), Multi-Calc (calculator with functions for scientific, engineering, and programming applications), Reminders (appointment calendar and alarm), DeskWrite (word processor), SuperScrap (clipboard/scrap manager), DataDesk (miniature data base), Blackout

(modular screen blanker), GoMacro (a macro NDA), and Deskpak Help.

If you order Deskpak (due in midsummer) now, SSSI will give you almost 50% off the suggested retail of \$79.95 and sell v2.0 to you for \$39.95. They'll even throw in free shipping within the U.S. (outside the U.S., add \$5). Contact SSSI, Inc., 4612 North Landing Drive NE, Marietta, Ga. 30066, 404-928-4388.—DJD

By all means, find your own way to accomplish things; in HyperTalk, like Applesoft, there may be several ways to accomplish a similar purpose. Being able to work out new solutions on your own is the indication that you're making the computer work the way you want it to, rather than using it as a toaster (stick in program, get out result). When you learn to program (be it with macros, HyperTalk, BASIC, C, or Pascal) you are beginning to control your destiny. Otherwise, you're confined by the whims of the programmer.

To clarify what may be a misunderstanding, the file data you read into a HyperCard field doesn't have to be separated by hard carriage returns; it doesn't even have to be a **text** file. The parser in our example used a tab-delimited data file created from an AppleWorks 3.0 database; tabs separate fields within a record and hard carriage returns indicate the end of a record. But we could have changed the script to use other methods for a different kind of file (I've used this to read records from a CD-ROM file to display in fields in a HyperCard stack).

A HyperCard field can hold up to 30000 characters, but a HyperCard Ilgs variable can hold as many characters as will fit into memory (page 155 of the **HyperCard Script Language Guide** says you can only read strings up to 16384 characters; that's an error). So if you want to read in the whole file, go for it.

I tried adding two fields into a third field as you suggested and got the same type of result. Apparently the selectedChunk (the location of the selected text or text insertion point in a field) gets set to "empty" (null) when the script is executed to assign a value to a field. That's no big thing; it might even be expected since you might consider the work in your current field to be "done" when the script starts executing.

It's easy enough to have it your way, since the selectedChunk is a value that can be requested from HyperCard. I used mouseEnter to trigger my script for field 16 (to cause the addition to take place, you drag the mouse cursor into field 16) and my script for field 16 looked like this:

```
on mouseEnter
    put the selectedChunk into ourSelection
    put card field 14 + card field 15 into me
    select ourSelection
end mouseEnter
```

This script saves the selectedChunk to our variable ourSelection before the "put" command invalidates the selection, then re-selects the save selection afterward. So the same text that was selected before the "put" remains selected afterward.

It's safe to say that none of us here at **A2-Central** consider ourselves to be "experts" on HyperTalk, though we're trying to get up to speed. We've found some experts that were willing to let us talk them into doing a Hyper-

Card Ilgs publication for us, though, and you'll see our announcement of the birth of **Script-Central** in this month's catalog.

HyperCard's animation is on the slow side; it's intended to allow the addition of movement for display effects, but isn't intended to be "animation" in the sense of an arcade game. If you've seen HyperCard on the Mac, you'll see that it doesn't exactly rip along, either. The only way we know of to speed up script-driven animation beyond the basic speed of HyperCard's interpreter is to speed up the computer's processor.

Again (trying to keep up the idea that HyperCard and **HyperStudio** do have different features) if you are interested in fast animation you may find that **HyperStudio** is better suited to that particular task. **HyperStudio** implements animation as a specific function within the program's button options whereas HyperCard handles it under script control (meaning the animation control loop can't proceed faster than the HyperCard interpreter can push it along). Such a specialized animation routine could be added to HyperCard via an XCMD, but we don't know of one that is currently available.—DJD

New address

I typed in your program (**Open-Apple**, p.2.83 and 2.93) to check my SuperExpanderE which has 768K. Your programs recognized only 128K. Any fix for this program?

Adalbert Goetz
Waynesboro, Pa.

Try using \$C071 (49265) instead of \$C073 (49267). Jerry Kindall at Quality Computer tipped us off that the **SuperExpanderE** uses a different bank selection address than some of the other auxiliary memory style expansion cards.—DJD

Fee paid?

Like most Apple owners, I am bewildered by Apple's schizophrenic behavior towards the II series. I mean, free system software updates, free tech notes, the Video Overlay Card, the DMA SCSI card, and now HyperCard on the one hand; the bald statement that the Ilgs is the last Apple II, the lack of Apple II promotion, etc., on the other...

Shareware is another area of concern. It's wonderful stuff and I love the people who are publishing it, but are they getting their money? It has to be obvious that if they don't the supply will dry up. We have a severe problem here as do most non-US users, I imagine, in getting the money to them. It can cost almost as much, if not more, to use bank drafts for small amounts as the shareware costs. Not everybody has a Visa or MasterCard and most shareware authors don't accept it anyway. I hear that AUSOM is looking at collecting the shareware fees for the stuff they distribute and sending it on to the authors. KAOS, the user group for which I am

Ilgs librarian, intends to do the same. It would help us if there was a matching organization in the US to distribute the money on our behalf. There would be an expense involved but the increased shareware fees should do more than cover that. The radio amateurs manage to run their QSL bureaus; surely computer people can manage something the same.

Peter Maloney
Hawthorn, Vic.

We're not familiar with how radio amateurs deal with their similar problems, but it would be interesting to hear suggestions of how to deal with this dilemma. We've heard suggestions that a company collect and distribute fees, but given the normally low amount requested as tender for shareware fees, additional transaction costs in a commercial endeavor to transfer fees may end up dissuading shareware payments.

Many interesting products seem to be appearing as shareware; paying fees to these authors is a good way to establish the viability of the Apple II market by convincing them (and even publishers, who may consider better shareware products to be "competition" in their markets) that there is money to be made in developing and improving Apple II products.—DJD

Tool 33

Ilgs System Software 5.0.4 includes a Tool #33; it's not listed in the **Apple Ilgs Toolbox Reference** volumes 1, 2, or 3. Can you tell me what it is and where I can get more info about it?

F. L. Stewart
Stewart Systems
London, Ont.

Tool 33 is the toolset for the Video Overlay Card; it's included with the system software but only has utility if you have the Video Overlay Card installed. The toolset is documented in the **Apple II Video Overlay Card Development Kit v1.1** from APDA (#A0221LL/B, \$35); the kit is a Class 1 (non-beta) product so can be ordered without a membership fee to APDA.—DJD

Alternate CD-ROMs

Does the Toshiba CD-ROM drive require a special driver like the NEC or will it work with the Apple-supplied Ilgs drivers?

Is CD-ROM mastering software available for the Ilgs and Mac platforms?

R. Payne
Noble Park, Vic.

We were told the Trantor driver for the Ilgs only supports the NEC drives, though support of others was being investigated; you may want to keep in touch with Digital Data Express

(13636 S. Western Avenue, Suite 28, Blue Island, Ill. 60406, 708-389-7744) regarding compatibility with other drives.

The **RamFast** v2.0 ROM update supports several SCSI CD-ROM drives without the use of special drivers.

Apple's technical publication, **develop**, had an article about CD-ROM mastering for the Apple II and Mac in issue number 3 (July 1990); back issues are available from APDA. You can also get more information from Apple Developer Services. If you have a concept for a product, it would probably be very wise to submit an application to become an Apple Associate (\$150 for Apple II; \$300 for both Apple II and Macintosh) or Apple Partner (\$300 for Apple II; \$600 for both).—DJJ

Praise for Plus Works

I'm writing this on an Apple II Plus using AppleWorks 3.0. My Apple II Plus has a **Ramex 128** board (Omega Microware), a **Smarterm** board (80 column capability from Advanced Logic Systems), a **Zip Chip**, and a **Martek** keyboard (has open/closed apple keys connected through the game port connector). The AppleWorks 3.0 has been modified by Norwich Data Services' **Plus Works III** with a patch provided by Norwich to activate the open/closed apple keys on the Martek keyboard.

I prefer this Apple II Plus setup for any necessary quick work over my 386SX clone. Perhaps some of your readers have an older Apple or Franklin hanging about and don't realize that it can be made to run AppleWorks 3.0 (original keyboards work fine and you don't need a **Zip Chip** but memory must exceed 64K and the ability to bring up an 80 column screen is required, as I understand it). I suggest such readers contact Norwich Data Services Ltd., P.O. Box 356, East Norwich, N.Y. 11732-0356, 516-922-9584.

I can only praise Norwich for their help in making my Apple II Plus run AppleWorks, particularly Richard Hajdu, Vice President. A couple of years ago when trying to get AppleWorks 2.0 up with **Plus Works II** Richard fielded my several phone calls with expertise (for example, my **Ramex** board had to be moved to slot 0, the **Zip Chip** set to not cached for slot 0 and the **Zip Chip** set to normal speed for the slot containing the **Smarterm** board). This time he was very accomodating in supplying the patch for the **Martek** keyboard.

It was a pleasure to do business with Norwich and thus a pleasure to write this. I also must you at **A2-Central** for making it with only 7,000 subscribers. You must be horribly efficient.

George Waaser
Wantagh, N.Y.

We try to be efficient. 7,000 subscribers isn't enough for us to go out and hire 30 employees and several specialists to locate and test products, but it is enough to swamp us with other work. That's why we appreciate having "reports from the field" like yours.—DJJ

Applied Engineering support

A development occurred in the Apple II community which I certainly hope will not set a precedent. Applied Engineering Incorporated (AE) has made three changes to its customer support which will adversely affect its customers and the Apple II community as a whole. I would

hope the Apple II community would respond with letters to its president documenting our displeasure with his company's new direction. I would expect an appropriate decrease in AE sales.

Applied Engineering has been a leader in the Apple II hardware arena for nearly a decade. Its founder, Dan Pote, built the company from a garage operation to the position it holds today, a leader in Apple II hardware manufacturing. It is my understanding AE is second in Apple II hardware sales only to Apple Computer Inc. itself. It is the manufacturer of a long list of favorites such as: **Ramfactor**, **RamWorks**, **RamKeeper**, **GS-RAM**, **GS-RAM Plus**, **TransWarp**, **TransWarp GS**, **Slot-Mover**, **PC-Transporter**, **DataLink**, **Serial Pro**, **TimeMaster H.O.**, **Z-80 Plus**, **Sonic Blaster**, **Audio Animator**, the **Vulcan**, and the list goes on and on.

Those of us in the Apple Corps of Dallas have long been fans of Applied Engineering because of its great customer support. It carried a 5-year warranty on most of its hardware. Its local office provided easy access to the technical staff. Phone calls to the technical support line were helpful and handled by knowledgeable technicians. That has apparently come to an end.

Recently there was a change in management at AE which has had great impact on customer support. The first change was the technical support staff. A great number of the technicians either left on their own or were laid off. The telephone number for technical support was changed to a 1-900 number having a rate of \$1.50 per minute. It is also my understanding that the standard 5-year warranty has been reduced to one year. Discussions with past employees of Applied Engineering indicate these changes are due in part to increased support of the Mac and Amiga lines. This business decision was reported to be in response to the declining support of the Apple II line by Apple Computers Incorporated.

We Apple II supporters must make our feelings known on this subject. Members of my user group will be writing letters to Applied Engineering's new president Bob Carroll, running editorials in our club newsletter, and most importantly showing our feelings through product sales. Showing this sort of response on a nationwide basis is also very important. If we allow one company to do this without suffering in the profit and loss area, other companies will follow Applied's lead. If you are inclined to support this cause, I recommend you send a letter expressing your feelings to:

Bob Carroll
President
Applied Engineering, Inc.
P. O. Box 5100
Carrollton, TX 75011

Mike Sample
Ilg SIG leader
Apple Corps of Dallas
Plano, Texas

We'd prefer not to second-guess Applied Engineering's position on this, so we've obtained permission to reprint comments they send to their customers on this subject from Patrick Gallagher, Vice President of Sales and Marketing:

"Over the past couple of years, we have continued to expand the Apple II Technical Support organization and still have not been able to handle the increased volume of calls.

This situation has occurred at a time when costs continue to increase and Apple II revenues are diminishing. This is a significant reason why all the other major hardware vendors have left the Apple II market.

"For 11 years AE has designed, developed, manufactured, and marketed over 100 products for the Apple II. We continue to support the Apple II market and are committed to its success. However, this would not be possible if we did not offset the tremendous support costs and provide a vehicle for customers who truly have AE technical issues to reach our technical support staff.

"After several months of research, we decided to follow a growing trend in the industry and implement a 900 number. Some of the major companies who have implemented a 900 program include, Microsoft, Lotus, Compaq, and Softwarehouse. Additionally, the big three distributors in the reseller channel, Ingram Micro, Merisel, and Tech Data, have initiated fee-based technical support programs.

"The goal is not to make the Apple II Technical Support department a profit center but to help offset some of the financial burden and allow us to continue providing support to the Apple II market. The \$1.50 per minute charge is one of the lowest in the industry and includes the customer long-distance charges.

"...I recognize that many people who received free technical consultation over the years will question fee-related services. We sincerely believe we are doing the right thing to keep AE financially viable, thus allowing the Apple II market to continue through the 1990's...."

The edited portions of the letter contain situation-specific advice, and also mentions two alternative sources of support: the dealer that sold the product (AE sells 92% of its products through dealers), or (for modem owners) the AE Bulletin Board System at 214-241-6677.

The strongest objection we've heard is that the use of a 900 number is confined to Apple II users while Applied Engineering's new product lines for the Amiga and Macintosh do not have this limitation. Mr. Gallagher indicates the reason for this is that Macintosh sales are currently strong enough to cover the additional cost of the non-900 (but also non-800) support line for their Mac products, where the current revenue for Apple II products is not enough to absorb the amount of Apple II support requested.

The reduction in the warranty term for current product sales is another sign that Applied Engineering may be trying to cut cost overhead, but realistically we doubt the change will be felt in most cases. Our experience is that a product that does not fail in the first year is unlikely to fail except due to extreme age (we have some interfaces over ten years old) or some type of physical damage not covered by warranty.—DJJ

Graphic, sound file formats

I have some questions about file formats. There seems to be a plethora of graphics file formats (low-res, high-res, double high-res, \$C0, \$C1, Print Shop, etc). Now I'm seeing **The New Print Shop**, **Print Shop GS**, and **GIF** formats. Where do these fit in? Are these last compatible

with other Apple II graphics? If not, can they be converted using *The Graphic Exchange* or some other program?

Along the same lines, what about sound file formats? I've noticed the GEnie Apple II library "ACER Sound Files". What programs are these compatible with? Are there other filetypes for sound files?

Bob Allan
Limuru, Kenya

There are generally three ways that file format standards are defined.

First are standards that arise from a single popular product. The original **Print Shop** graphic format became a "standard" because many people used the program and exchanged files.

Next are manufacturer standards that are documented so that others can use them. Apple has originated several of these formats for data types it knows its systems will be using.

Next are "computer independent" formats agreed upon for the conversion of data. DIF was an attempt at this for record-based data (see "What's the DIF", **A2-Central**, April 1989, p. 5.17-20).

Apple publishes File Type Notes documenting file formats which have been cleared through Apple Developer Technical Support (DTS). A developer can document their file format and request a filetype from DTS; if the file format is documented and is used in a **shipping** program (it doesn't have to be commercial, but the program does have to be "real" and in distribution so a filetype provided for it isn't wasted) Apple will include the information regarding the file format and the originating company in a File Type Note. This is a wonderful way to consolidate information on file formats for posterity.

Not all manufacturers will go this route; they may elect to develop a format independently (sometimes in the desire to maintain a proprietary advantage by use of a special, incompatible format). This is thankfully becoming a rare occurrence. But if the file format is not documented in the Apple II File Type Notes, then it must be assumed the manufacturer hasn't sent the file format information to DTS for publication and assignment.

There are several graphics file formats due to differing opinions on what the "best" way is to store a graphic for an intended purpose. Apple proposed a few itself, and as new (and possibly better) ways of storing information are thought up new formats may be added. Right now, the most popular techniques for storing Apple IIgs graphics images are probably as an unpacked screen image (type \$C1, auxiliary type \$0000) and in Apple Preferred Format where the picture data can be compressed.

Most of the "computer independent" graphics formats are pretty inefficient when compared to computer-specific storage techniques, so it's unlikely you'll see a standard like CompuServe's GIF (Graphics Interchange Format) become the "default" standard for a manufacturer. But utilities to convert such independent formats are usually developed for most popular systems.

Two products that allow converting several graphics formats are **The Graphic Exchange** (\$49.95 from Roger Wagner Publishing, 619-442-0522), and **SuperConvert** (\$39.95 from Seven Hills Software, 904-575-0566 or 800-

627-3836). Roger Wagner is currently working on an enhancement disk to **The Graphic Exchange** (version 4.2 or later) that will provide converters for several additional formats including GIF files; the newest version of **SuperConvert** has GIF support included.

Sound formats include computer-specific formats such as MIDISynth records on the IIgs, but there are also independent formats such as Audio IFF (for digitally "sampled" sounds), Audio IFF-C (the "C" stands for "compressed"), and MIDI formats (both formats have been allocated Apple II File Type designations). We know less about this area, and products for interconversion are not as frequently seen. Apple does officially recommend that all sound applications read and write AIFF-C files (filetype \$DB, aux type \$0001).

You may have seen the term ACE, which is an abbreviation for the Audio Compression/Expansion toolset that has been supplied with IIgs System Software version 5.0.5 and later. ACE provides a standard system tool designed specifically for compression of digitized sound (and, of course, expanding it again for later use). ACER is a sound utility application; the name is often used to refer to files stored in the ACE-compressed format the application uses. However, we didn't find a file type assignment for ACER in our current listings.—DJD

Block access and interleave

I want to be able to load a sector, any sector, from a ProDOS device into memory at, say, \$2000 and be able to manipulate it, all from within BASIC. The manipulation I can handle, but I don't know how to load the sector from disk.

Brian Noller
Wagga Wagga, N.S.W.

All you need to read blocks is to execute a read block MLI call. A simple routine you can CALL from Applesoft looks like this:

```
0300: LDA    #000
0302: STA    $031A
0305: JSR    $BF00      ProDOS MLI entry
0308: DC     H'80'      read_block function
0309: DC     A'$031C     parameter block
030B: BCC    $0311      no error returned
030D: STA    $031A      save error code
0310: CLC              for Applesoft
0311: RTS              back to BASIC

031A: DS     1          byte for error code
031C: DC     H'03'      parameters in list
031D: DC     A2'$2000'  buffer address
031F: DC     I2'$0000'  block to read
```

The buffer address is the location in memory of 512 bytes of space to allow reading the block; I used \$2000 since this is unlikely to interfere with a simple BASIC program even though it is sitting in the middle of unprotected memory where various entities (an HGR command or encroaching variable tables) could overwrite it. For a more rigorous method you'd want to protect the buffer by tucking it under LOMEM; or above HIMEM.

You can POKE your own buffer address or block number. To change from reading to writing blocks, POKE the MLI command code with \$81 (the parameter block is the same for reading and writing).

As a sample program, I'd like to redo a reply to "Disk Interleave Info" (**A2-Central**, January 1991, p. 6.96) regarding the detection of the

interleave of a 3.5 disk. I discovered an old message from Dave Lyons (now a member of the Apple II System Software group) indicating that on the IIgs the interleave of a disk can be found in memory after reading a block from the disk. **This is not officially documented or supported**; it works on the ROM 01 and ROM 03 IIgs but there is **no** guarantee it would work if there was another ROM revision. Assuming we take that risk, all we need to check the interleave (after first verifying we have a IIgs) is our read_block call and another short routine to grab a byte from the extended IIgs memory.

To detect the IIgs, we check that \$FBB3 (-1101 in decimal) contains 6, which indicates the computer is a IIe, IIc, or IIgs. Then we use the following code:

```
0300: PHP              save status
0301: SEC              get ready
0302: JSR    $FE1F      IIgs ID routine in ROM
0305: ROR    $030A      saves carry in high bit
0308: PLS              recover status
0309: RTS              back to BASIC
030A: DS     1          byte for carry status
```

The routine at \$FE1F in the IIgs ROM will clear the carry (and also returns other information such as the ROM version which we don't check here); on the other systems it just returns with the carry unaltered.

The memory location we need to examine for the interleave is \$E1/OF31; we just grab that byte and put it where BASIC can PEEK it:

```
0312: LDA    E1OF31
0316: STA    $031B
0319: RTS
```

This routine was tucked in as part of the "read_block" routine in the Applesoft program that follows; if the block is read without error, we now branch to this routine at \$312 instead of the RTS at \$311. (In case you're wondering, the IIgs's 65816 doesn't have to be in "native" 16-bit mode to use long addressing; that particular operation is available even from the "65C02 emulation" mode.)

First, we check for the needed computer (a IIgs):

```
1000 REM == get interleave on IIgs ==
1010 TEXT : HOME
1020 IF PEEK ( - 1101 ) = 6 THEN GOSUB 1220:
      CALL 768:IIGS = ( PEEK (778) < 128 )
1030 IF NOT IIGS THEN PRINT "Interleave test
      won't work.": STOP
```

Next we install the read_block code, including the routine to grab the interleave byte:

```
1040 GOSUB 1250: REM install our read block routine
```

Then we loop through instructions that ask for a disk, reads the block (via CALL 768), checks for a (non-zero) MLI error code, and (if no error) grabs the interleave byte and reorts the interleave found. The byte containing the interleave also identifies whether the disk is one- or two-sided, so we might as well report that, too:

```
1050 TEXT : HOME
1060 INPUT "Insert a disk and press <Return>: ";A$
1070 CALL 768
1080 IF PEEK (794) THEN PRINT "MLI error #";
      PEEK (794): GOTO 1160
1090 BYTE = PEEK (795)
```

```

1100 IL = BYTE - INT (BYTE / 16) * 16: REM interleave
1110 DS = NOT ( INT (BYTE / 64) * 2 = INT (BYTE /
32)): REM double sided?
1120 PRINT
1130 PRINT "This disk's interleave is ";IL
1140 IF (DS) THEN PRINT "This is a double-sided
disk": GOTO 1160
1150 PRINT "This is a single-sided disk"
1160 PRINT
1170 PRINT "Another? (Y/N) ";
1180 GET AS$: IF AS$ < > "Y" AND AS$ < > "y" AND
AS$ < > "N" AND AS$ < > "n" THEN GOTO 1180
1190 PRINT : IF AS$ = "Y" OR AS$ = "y" THEN GOTO 1050
1200 END

```

There are two short subroutines; one to install the "check for IIgs" code, and another (which overwrites the first) to perform the read_block request:

```

1220 REM — install check for IIgs ML —
1230 FOR EYE = 768 TO 778: READ THUMB: POKE EYE,THUMB:
NEXT EYE: RETURN
1240 DATA 8,56,32,31,254,110,10,3,40,96,0
1250 REM — Install read block ML —
1260 FOR EYE = 768 TO 801: READ THUMB: POKE EYE,THUMB:
NEXT EYE: RETURN
1270 DATA 169,0,141,26,3,32,0,191,128,28,3,
144,5,141,26,3,24,96,175,49,15,225,141,
27,3,96,0,0,3,80,0,32,0,0

```

If you think you might want to POKE other routines into the same area, remember that the READ statement always continues at the next DATA item following the last one read. The two subroutines above work because we only use them once, and in the sequence of the DATA.—DJJ

FORMAT II help

In the January, 1990 issue of **A2-Central** (5.95), you had a letter requesting information on transferring data base records created with Kensington's **FORMAT II** program to Appleworks. **FORMAT II** (the DOS 3.3 based version) uses RWTS directly to store records on disk, making normal file-based utility programs ineffective in dealing with them.

Faced with converting over 30,000 records myself, I have succeeded in crafting a procedure that will transfer the records.

Fairly simple, it requires 4 pokes in the BASIC loader program, and a few other 'minor' setup procedures. This allows the records to be 'printed' to a sequential access text file for importation into Appleworks, after, of course, being converted to ProDOS.

I won't bore you with the particulars (unless you really **want** to know), but I would be happy to share this information with anyone who is interested, as it is a 'whale' of a time saver.

By the way, although I prefer Appleworks database manager for large record files, the word processing module of **FORMAT II**, for 1 and 2 page letters, memos, etc, continues to be the best I have ever used on any computer, and I will continue to use it. Although Kensington no longer publishes it, it is still being sold and supported by the British company that wrote it. Yes, there is a ProDOS version available, which we now use, but, like **AppleWriter** on the IIe, there is no support for memory greater than 128K. For that, you need the MS-DOS version. Sound familiar?

Hugh Hood
Box 154308
Waco, Texas 76715-4308

Time is Money fix

It has come to our attention that there is a small problem with *Time is Money*. Since the original programmers have long since retired to raising sheep in New Zealand (well, actually they're still around, but haven't had time to revise the program) we've asked them for a work-around to this problem. Here we go:

The problem: When you carry forward from 1989 to 1990, the dates on the new user disk all say 1991, not 1990. The Oracle informs us that the mistake was made by the programmers in the Master Disk program, not the main *Time is Money* user program.

The solution: We hope this is not too complicated, but it does work. You will need your *Time is Money* Master Disk, the disk from which you want to carry forward, and two blank disks before you start. You should have at least two backups of the disk from which you want to carry forward.

We will be juggling 3 user disks here, so we will call them OLDDISK, NEWDISK, and TEMPDISK to help keep things clear. OLDDISK is your old user disk with your current data (make sure you have at least two backups of this disk). NEWDISK is the new user disk to which you will carry forward. TEMPDISK is a second, temporary, user disk which you will create that lets you create the magic of fixing the dates.

Take a deep breath...

1. Make a new user disk (NEWDISK) with 1990 as the start year for Book 1. The start year for Book 2 can be either 1990 or 1991. The names for each book can be whatever you want.

2. Carry forward from your old user disk (OLDDISK) to Book 1 on the new user disk (NEWDISK) that you made in step 1. The carry forward date must be 01/01/90.

3. Make a second new user disk (TEMPDISK) where the start year for Book 1 is 1991. The start year for Book 2 does not matter. The names for each book can be whatever you want them to be.

4. Follow the instructions in this step carefully: Carry forward from Book 1 on your new data disk (NEWDISK) to Book 1 on the second new data disk (TEMPDISK). The carry forward date must be 01/01/91. As this operation proceeds, the status line should indicate that you have the same number of transactions on both the new user disk (NEWDISK) and the second new user disk (TEMPDISK). If you don't have the same number of transactions on both disks, you have probably entered the wrong date.

5. Boot the second new user disk (TEMPDISK).

6. Remove the second new user disk (TEMPDISK) from the disk drive.

7. Insert the new user disk (NEWDISK) in the drive from which you just removed the second new user disk (TEMPDISK).

8. From the main menu, select B (for Backup and Exit).

9. From the Backup and Exit menu, select S to save your data to the new user disk (NEWDISK). **DO NOT EXIT FROM TIME IS MONEY.**

10. Remove the new user disk from the drive (NEWDISK) and turn off your Apple II. The operation is now complete. Label this disk appropriately as it is your new user disk.

11. You may destroy (or otherwise reuse) the second new user disk (TEMPDISK). It is no longer needed.

12. Remember to make at least two backups of your new user disk (NEWDISK).

Bruce Rosenblum
Turning Point Software, Inc.
230 Western Avenue
Boston, Mass. 02134
(617) 782-4877

The path not taken

Recently I attempted to upgrade the IIgs system software on my hard drive from version 5.0.2 to 5.0.4. This was reasonably easy to do. However, it had an effect I did not anticipate. I use an 8-bit selector called *EasyDrive* (from Quality Computers). When I go from the selector to *WordPerfect* for the IIgs, version 2.1, it overrides the default setting for the data file location. That is, instead of giving me the path "hard.1/wp.data" when I go to retrieve a file it consistently defaults on its own to "hard.2/", the second partition on my 40 megabyte hard drive. When I launch *WordPerfect* from the Finder, I get the correct default prefix for my data files. It seems obvious to me that the problem lies in the switch between the 8-bit selector and the 16-bit program. What is happening here?

Both the selector and all *WordPerfect* files (system and data) reside in the partition "hard.1". Why does the operating system feel compelled to look to the second partition for my data files? How (and why) does the system override the default data prefix that is part of the defaults file for the program?

Andrew Klimas
Randalltown, Md.

This was driving many users crazy, and credit for finally tracking it down goes to developer Shawn Quick. Shawn's explanation mentioned the normal GS/OS file selection tool *Standard File*, but the same type of thing could occur in other circumstances.

When a device (driver) is restarted by booting or by re-entering GS/OS from ProDOS 8, it prepares to send a notification that a new disk may have been inserted (this is done so that GS/OS will know that any disk in the drive will need to be read). If the driver is not called by another application before *Standard File* is called then that notification is left pending.

When *Standard File* is called it installs its notification vector and then builds its device list by calling each device driver, checking for block devices and those with removable media. When a device driver for a device that supports removable media is called, it will notice that the internal "disk switched" flag is set and so calls GS/OS to tell it so. GS/OS in turn calls all the installed notification vectors so they can react to the disk switched event. This includes *Standard File*, which then assumes something has changed. So *Standard File* tries to read the "new" directory from the "switched" device. If no disk is there, *Standard File* grabs the last available device in the device list, reads its file list, and displays this instead. In your case, you just got bumped from hard.1 to hard.2.

Standard File intentionally works this way (jumping over to check a switched disk) so that inserting a new 3.5 disk will automatically cause the new disk's file list to appear in the *Standard File* window (this makes it easier to check a series of floppies for the file you seek

without constantly tabbing through all the on-line devices). The problem is the initially pending notification after the driver is first restarted.

You don't see this problem when launching programs from the Finder since the Finder is constantly polling removable media devices and will therefore "clear" the initial notification. Shawn has written a permanent initialization file called **SF.Fix** that...well, let's let Shawn explain:

"When a driver is started up, whether by a cold boot or return to GS/OS from P8. It holds a pending 'disk switched' that can not be returned to the operating system, until some of the driver's code has been executed. What my INIT does, is simply insert a notification procedure into the system that causes all removable devices to execute some code (`_DStatus I believe`) any time GS/OS is started or restarted. This causes those devices to tell GS/OS that the disk has been switched and will 'eat' the pending disk switched. Once this has happened, everything in the system is back in sync, and SF (and everything else) works properly. You may also see from this, that my INIT only is executed on GS/OS starts and restarts... it is never used again. Some people had commented to me that they didn't like the fact that it was called anytime SF was called or when an application is launched... it's really not, **only** on start/restart. So you may point out that the INIT does **not** patch anything in the system, nor slow it down, etc. It's hardly ever used."

Other applications would need to follow this model; Glen Bredon has already modified **ProSel-8** (which can launch GS/OS applications if GS/OS was booted) to take this into account.

We've also been frustrated by the use of the data prefix (GS/OS prefix #8) in that we seldom seem to leave it set where we want for certain applications like **AppleWorks GS**. For some of these programs, we'd like the program to use the same path by default each time we enter the program (such as classic AppleWorks "default disk"). We've seen a new shareware utility called **DataPath NDA** that allows you to specify a default data path for your applications; it's \$10 from Bill Tudor, 1220 Gerling Street, Schenectady, N.Y. 12308.—DJJ

SCSI-2 and you

I'm in the process of putting together a hard drive for my IIGs. I thought all I needed to look for was a SCSI drive and I'd be fine. Now, however, I'm seeing drive mechanisms advertised as SCSI, SCSI-2, and Mac compatible SCSI. What's the difference? Is there anything else I have to be careful of in making the decision?

Daryl C. Morgan
Turlock, Calif.

The word "standard" in the computer industry has historically seemed to mean a set of design criteria that looked absolute in theory but that was often modified (some manufacturers say "enhanced") in practice. Like many other standards, the SCSI specification was supposed to define a set of rules that would allow manufacturers to design equipment that would "plug and play" with each other's products. In practice, either non-conformance to the standard or non-conformance **within** the standard (sometimes it's hard to nail the lid down on creativity) would keep two devices from working together.

The reference work on SCSI is the Small Computer System Interface (SCSI) Specification (ANSI X3.131-1986), available from the American National Standards Institute, 1430 Broadway, New York, N.Y. 10018, (202) 642-4900 (sales department). Manufacturer's references we've seen are pricey (we've seen a Fujitsu manual that hovers around the \$45 mark), but we did pick up an NCR pamphlet titled **SCSI: Understanding the Small Computer System Interface** (ISBN 0-13-79855-8) which "only" ran \$19.95 for about 70 pages of information. Sarcasm aside, it appears to be a very concise, useful technical overview of the SCSI standard (it does not include specific information such as descriptions of individual commands). It also lists general proposals for extensions of the current standard to SCSI-2 and SCSI-3.

The original SCSI standard has actually worked pretty well; it is relatively unusual to hear of compatibility problems that keep two SCSI devices from working together. Sometimes differences in the exact implementation do cause problems, though, so as SCSI is becoming more widely used the impetus to insure compatibility is becoming more prominent.

SCSI-2 attempts to remove some of the "design options" that could allow incompatibility among devices from different vendors following the original specification. We asked Chris Adams at Chinook Technology and Matt Gulick at Apple Computer about the newer standards; they confirmed that they shouldn't hurt (and should help, by confirming the adherence to standards) compatibility with true SCSI interfaces which are currently available for the Apple II. There are also some enhancements (such as provisions for higher-speed transfer methods) which don't benefit current Apple II systems.

"Macintosh SCSI" could indicate a couple of things that would not be likely to interfere with use of the drive on a IIGs. First would be that the drive has been tested on and works on a Macintosh. It may also indicate that the drive includes the new terminators required for the high-speed SCSI bus of the Mac IIfx (see Macintosh Technical Note #273). But a third possibility is that the disk **requires a special Mac driver** to simulate full SCSI compatibility, and that will keep the drive from working on a IIGs.—DJJ

A different view

In regard to the letter in **A2-Central**, p. 7.30, your readers may be interested in knowing that the View routine in **Change-a-File** (found on GEnie) will read control characters in a file or directory.

Harold D. Portnoy
Bloomfield Hills, Mich.

Change-a-File is a shareware utility that allows converting AppleWorks 3.0 word processor files to AppleWorks 2.x format, interconverting AppleWorks word processor files and standard ProDOS text files, filtering line feeds from text files, adding line feeds after carriage returns, massaging AWP and ADB files whose internal structure has been damaged so that they can be loaded into AppleWorks (the extent of the recovery depends on the extent of the damage), changing file types and auxiliary types, and viewing the contents of files.

The View option is useful for checking the

contents of text or AppleWorks files before using the conversion utilities, but it can be used to view nearly any type of file (we ran into a few types of files, such as DIR files, where nothing appeared on the screen).

The recovery options work by verifying the header for the damaged AppleWorks file; if the program can recover from any detected errors it will rebuild the file, otherwise options are provided to try and extract as much information as possible and place it in a new file. The program can deal with bad data blocks within a file, but cannot recover a file with a damaged directory or index blocks (index blocks indicate where the file is stored on disk). Harold recommends using another utility in such a case; his own **Ressurrection** program is designed to try to recover files where the directory has been damaged. After processing with **Ressurrection**, you can try feeding the "restored" files to **Change-a-File's** data recovery options.

Harold has been spreading these files by BBS; after you download them, you will find that some options in the programs are not available. Payment of a \$10 fee gets you the password to unlock the files for full functionality.—DJJ

Apple's StyleWriter printer

OK, the situation is like this: Apple goes and creates a new and affordable inkjet printer—the StyleWriter. It connects to the standard serial printer port and is driven by TrueType (Apple's own idea of PostScript-incompatible, of course). It (the printer) costs about \$500, and its printouts look just as good (with text at least) as those done with a LaserWriter II. The problem? According to my local unfriendly Authorized Apple Dealer, this printer **only works with the Mac**. If I wanted to use the printer with the IIGs I would need a GS/OS device driver, of which Apple has yet to announce or make available.

I have snooped around the various BBS's and found many assorted printer drivers that have been written. I guess what I am asking you guys is this: if you see **any** StyleWriter/TrueType driver around, should it be commercial or otherwise, please let us know.

We (the Apple II community) have the computer, the page layout software, and the fonts. Give us a TrueType driver, and we will have what we have been waiting for: laser quality printouts at a dot matrix price.

Gregory J. Betzel
River Falls, Wisc.

The StyleWriter is not a panacea; it's quoted print speed is **half a page per minute**. Also, the StyleWriter does not use a page description language like **PostScript**; that means graphics images would be rendered as bitmaps, not as the smooth graphics that can be rendered on the **PostScript** LaserWriters.

The StyleWriter's print engine specifications sound suspiciously similar to those for the Canon **BubbleJet** printers: thermal ink jet technology with a print resolution of 360 dots per inch (that's slightly higher than the 300 dpi of many common ink jet and laser printers). We checked with Vitesse's John Pothier, who says they have had some reports of good results using **Harmonie's** Epson LQ-series

driver with the **BubbleJet 10e** which has an LQ emulation mode.

The StyleWriter's major advantage would be the support of TrueType font technology. TrueType fonts are scalable like the LaserWriter's Adobe **PostScript** fonts, but have the advantage (to Apple) of representing a standard that Apple, in conjunction with Microsoft, controls.

TrueType is not yet supported by the IIGs printer drivers, so let's grapple with that. Apple had not released the complete technical specifications of the StyleWriter yet, so it was not possible for a third party company to develop drivers prior to its release. If Apple wants to keep TrueType under proprietary control, the only source for a driver may be Apple.

The StyleWriter also doesn't just accept and print ASCII data; characters are printed in conjunction with the TrueType technology of rendering scalable fonts, and graphics are imaged **within the host computer** and sent to the printer as compressed bitmaps. Given that the printer does not accept ASCII data, using it with 8-bit software such as AppleWorks may be out of the question. Even the IIGs System Software is currently not prepared to deal with all of this.

We expect Apple to be committed to the use of TrueType fonts in the future, so TrueType may be supported on the IIGs in some future system software update. No specific future IIGs System Software features have been discussed, but we hope this is under consideration; the cost of the StyleWriter is certainly low enough to be competitive with third-party alter-

natives.—DJJ

Apple II vs. Mac

I was a charter subscriber to **Open-Apple** and have been a faithful subscriber ever since (taking the disk version since it was released). However, despite your cute little post card and numerous reminders, I have pretty well made my mind up not to renew my subscription when it expired with the April issue.

Well, I've thought about it some more, and have decided to stick with it another year. **But** I wanted to convey to you why I was finally going to let it die.

I am really rather sick of hearing very vocal members of the Apple II user family constantly screaming about the death of the Apple II line and how Apple has maligned them by trying to kill it. It was bad enough having to wade through that dung here on GEnie (until I finally permanently ignored all of the categories which contained such tripe to cut my bill) trying to capture the few true gems of messages interspersed. It was a final straw, though, to have to read it in **A2-Central**. I've been associated with the Apple II for many years (not as long as you, though) and in my honest opinion, the computer's biggest enemy has been this group of lamenters, standing at the wailing wall, crying and shredding their garments for the past few years.

Even though Apple did not actively and publicly **push** the II line (and the IIGs in particular), this lack of attention really did it relatively little harm. New stuff was slow for the IIGs, **but** it did come faster than it ever did for the Mac. It was just taking time for us "die-hard" CLI (command line interpreter) users to get over the GUI (graphics user interface) prejudice and start learning how to use the new capabilities of the family. Major manufacturers were **not** deserting the series. **Then** the wailing started, and continued, and continued, and ... Everywhere a software or hardware developer turned, they were told that the Apple II (and specifically the IIGs) was dying or dead. Constantly. They soon started to conclude that it must be true and **then** they started abandoning the machine and the series. **Who** was telling them this? **Not** Apple, it was the hard core, chest beating, mostly respected independent programmers, hackers, and "gurus" who were telling them this. So, it **must** be true. These attitudes and actions by outstanding community leaders such as yourself (and/or your organization) have done more damage and chased away more developers than Apple could ever have wished for. The lament has become a self-fulfilling prophecy. I think it's time for a major attitude adjustment on the part of the "Apple II community". That is why I've decided to go another year with **A2-Central**, because there seemed to be a more positive and constructive approach in the last two issues or so (like the **A2-Central** Catalog of March—though that banner **could** have been done in a much more positive way).

There are a lot of Apple II owners like me out there. (I know, I run a BBS and have been a "community leader and guru" on the local scene for many years.) I own multiple units of multiple versions of this great computer. I also have Macintosh's (from a laptop up through the IIfx) either of my own or at my office. I also extensively use MS-DOS machines. They're **not** mutually exclusive! I "like" them all (well, I tolerate the Messy-DOS stuff). I'm not tired of the II computers, especially the IIGs which I have

found superior to the low-end Macs and I can't afford IIfx's for my home. But, I **am** sick and tired of the attitudes being preached by the "user community" leaders and so are a lot of other owners. People don't go to user group meetings and conventions or pay hard earned money for subscriptions and online services just to sit around and hear the "pros" tell them how bad they and their computer of choice have it. They want the positive. Many Apple II series owners out there never realized how miserable and unhappy they were until some friendly newsletter editor or online guru enlightened them. Up until that point, they mistakenly believed that the computer they owned and loved was meeting their needs! How about accentuating the positive for a few years? You may be surprised at the level of hardware and software support that still exists and might even be regenerated! There are several other brands of computers which are much "deader" than the Apple II series, which have fewer deployed units, which may not even be in production any more much less enjoy manufacturer's support, and yet which enjoy extremely good after-market support. The only clear difference that I see with them is that their users aren't running around telling the world what fools they are and how terrible their computers are.

Well, I hope I've said enough on the subject to maybe start an attitude adjustment process. Thanks for reading this and for all of your past positive contributions to the growth and maturing of the Apple II line. I sincerely hope you'll be able to help extend its "life" even further.

Scott Galbraith
Monrovia, Md.

There's a wide variety of opinion on this issue in the Apple II community, and several people here, including our publisher Tom Weishaar and our head GEnie sysop, Chet Day, are cheerleaders for your point of view. They've felt for months that we should stop expending emotional energy worrying about what Apple did or didn't do and get on with enjoying our computers.

*Do Apple II owners dwell overly much on their problems with Apple Computer? Probably so. Users seem to expect Apple to do **everything**, and Apple is not that large a company. There are many aspects in which the Apple II user community has **not** rallied to support the machine. Apple II engineers have generally given us great tools to make the IIGs useful and there are some immensely useful products on the market that are not being exploited to their full potential. Mac users took HyperCard and wrote their own small business software; Apple has now given similar opportunities to us.*

*The success of accentuating the positive does depend on Apple's **visible** commitment to **promote and support** the Apple II. User evangelism can't succeed when the parent company invests heavily in advertising that works against them, and it is important to make others understand Apple's contribution to the weakening of the Apple II market. Apple expended massive corporate resources, including a lot of Apple II dollars, to entrench the Macintosh in the public's mind. Apple II users have a right to expect more from the company than it has done for the Apple II and especially the IIGs.—DJJ*

A2-Central™

© Copyright 1991 by
Resource-Central, Inc.

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

Publisher:

Tom Weishaar

Editor:

Dennis Doms

with help from:

Sally Dwyer
Jay Jennings
Denise Shaffer

Dean Esmay
Jeff Neuer
Tom Vanderpool

Joyce Hammond
Ellen Rosenberg
Jean Weishaar

A2-Central—ditled **Open-Apple** through January, 1989—has been published monthly since January 1986. 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.

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

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

ISSN 0885-4017

GEnie mail: A2-CENTRAL

Voice: 913-469-6502

Printed in the U.S.A.

Fax: 913-469-6507