65816 • FOUR MEGABYTES • BUILT-IN HARDDISK • VLSI
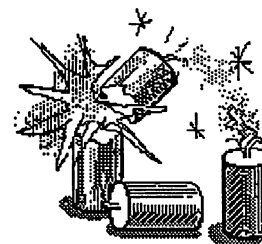COLOR GRAPHICS • MIDI INTERFACE • MEGABYTE FLOPPY
LOW COST LASER PRINTERS • DIRECT MEMORY ACCESS •
EXPANSION SLOTS • FLOATING POINT HARDWARE • SCSI
PAGESIZE SCREEN • 68020 • CD ROMS • BUILT-IN FAN •
BUILT-IN MODEM • DILITHIUM ... OWERED • AI •
REFINED USER INTERF... ...ALE INTEGRATION
ROM SOFTWARE • PORTAB... ...E ROM • 68081 • RISC
MORE MEMORY • MODULAR ... ANSION SLOTS • DMA
ARTIFICIAL ... ...APPLE //
CD ROMS ... E SERVER
65816 • ... ...UTPUT
PORTABLE... ...AY ... ...OLOR
DMA • CRAY XMP ... ...MODEM
MORE R... ...CLF

Contents

About the Cover

What will Apples' elves bring us next? With all the rumors we've heard lately about new products, all bets are off as to when, what, and how much. But rest assured that some of the items on the cover will materialize.

Editors

Tom Engle
Chris Flick

Overview

The Houston Area Apple Users Group provides a forum for the exchange of ideas and information about Apple Computer products and their associated hardware and software. Meetings, newsletters, and special interest groups are just a few of the ways this forum is maintained. Below are listings of the club officials and descriptions of the groups' functions.

Club Officials

President
Robin Cox   778-1635
First Vice President
Tom Dillon   665-6502
Second Vice President
Michael Conway   495-2292
Secretary
John Marek   782-3898
Treasurer
Jack Cowart   467-4215
Directors
Steve Bass   847-4407
Jeanne Boucher   723-9519
Dick Fairman   723-9619
Mike Kramer   358-6687
Dick Lee   821-1298
Bill Muhlhausen   668-3963
Past President
Michael Levinson   270-0137
Senior Advisor
Rudge Allen   622-3979
Membership
Midge Kocen   270-6246
***Apple Barrel* Editors**
Tom Engle   774-9055
Chris Flick   363-3153
Publicity Coordinator
Dennis Bilbe   HotLine
IAC Representative
Ina Levinson   270-0137
Vendor Coordinator
Jeanne Boucher   723-9519
Program Coordinators
Mike Kramer   358-6687
Steve Bass   847-4407

The club officials meet on the first Thursday of the month at the Marriott Astrodome Hotel at 7 pm. These meetings are open to the membership.

Membership Information

A one-year membership to HAAUG is available for thirty dollars and includes the Starter Kit, a subscription to the *Apple Barrel,* and a membership card giving access to all the club's services. Memberships are available at the Answer Desk during the meeting, through brochures located at local dealers, or by calling the HAAUG HotLine and having a brochure mailed to you. Renewals are twenty dollars per year.

Meeting Dates

The HAAUG monthly meetings take place at the Marriott-Astrodome Hotel, 2100 South Braeswood at Greenbriar. The meetings are scheduled for the third Saturday of the month and begin at 11 am. All new members should attend the New Member Orientation.

Meeting Schedule

11:00	New Member Orientation
12:00	HAAUG Club Business
12:30	Main Presentations
1:30	Special Interest Groups
2:30	Special Interest Groups
3:30	Special Interest Groups

HAAUG HotLine (713) 522-2179

The HAAUG Hot Line provides a means for members to learn of meeting topics, news, etc, and to obtain answers to computer-related questions. If you want a return call, leave your name, telephone number, and membership number. If the question pertains to a particular commercial product, please check the list of volunteer specialists in the back of this issue.

Special Interest Groups

SIGs focus on one topic of common interest during these meetings within a meeting. SIG meeting locations and times are contained in the meeting handout.

SIG Leaders

Apple // SIG Coordinator
Tom Dillon   665-6502
Appleworks - Beginning
TBA
AppleWorks-Intermediate
George Pierce   526-5103
Advanced Topics
Bill Hensley   980-4993
Jim Record   353-1119
Basic Programming
Robin Cox   778-1635
Business
Rudge Allen   622-3979
CP/M
Kip Reiner   463-7074
Education
Dick Lee   821-1298
Games
Thomas Carson   861-5425
John Newell   531-9139
Graphics
Dick Fairman   723-9619
Green Apples
Richard Goss   463-0640
Hardware
Mark Stevens   341-8197
Home Use
Mike Stoops
David Jaschke   461-2450
Pascal
Richard Lemay   749-3135

Mac SIG Coordinator
Steve Bass   847-4407
Business
Gary Saxton   479-9525
Programming
Mike Martin   333-6405
Desktop Publishing
Tom Engle   774-9055
Cleland Early   941-7247
Tips & Techniques
Steve Bass   847-4407

Public Domain Software Libraries

The Public Domain Libraries are divided into two sections: Macintosh and Apple //.

Macintosh Library

The Macintosh library contains over ninety disks of public domain software. Copies may be purchased at the meeting for five dollars apiece. In addition, there are two directory disks which may be copied for free. These disks contain the listings of what is on the other disks. Also, the latest disk submissions are available for copying.

Macintosh Librarian
Larry Saphier   496-0507

Apple // Library

The Apple // library may be copied at the meeting for one dollar per side. Books are available at the meeting describing what is available in the library.

Apple // Librarian
John Marek   782-3898

Hardcopy Library

The Hardcopy Library maintains an archive of back issues of the *Apple Barrel*, issues of other magazines, and other paper material. It is open to the membership and is available at the main meeting. At present, no provisions for loaning the material have been made.

Hardcopy Librarian
Les Brown   795-2741

"HAAUG Heaven" Bulletin Board System

HAAUG Heaven is HAAUG's computerized Bulletin Board System. The BBS's telephone number is (713) 664-9873 and operates 24 hours a day. The recommended protocal setting is 8-N-1-Full.

Writer's Guidelines

The HAAUG *Apple Barrel* exists to provide the membership with timely information about their Apple computers. We at the *Barrel* hope that readers will write articles on subjects such as reviews, computer techniques, and casual observations of computerdom. For those of you interested in writing for the *Barrel* (hopefully most of you), we have established helpful guidelines to follow when writing and submitting articles.

1. Try to include some form of graphic or table which helps to explain points made in the article. The saying that a picture is worth a thousand words certainly applies here.

2. For those including graphics and tables, please make references to them within your text. This will help both the reader understand what you are saying and assist us in the paste-up of the article.

3. Include references to other sources of information if possible. By doing so, the reader is directed to other useful material beyond that of the actual text.

4. For Apple // users, we will accept text created either in *AppleWriter* or in the *AppleWorks* word processor. We prefer ProDOS, but can handle DOS 3.3 as well.

5. For Macintosh users, we accept text created in either *MacWrite* or *Word*. Please do not include graphics within the actual text. Graphics accompanying the articles should be saved as *MacPaint*, *MacDraw*, or *Chart* files.

6. Articles should be provided on disk as well as printed hardcopy. The hardcopy is primarily used to allow us to see where you as the author place emphasis. (A backup copy for yourself would be advisable)

7. Article submissions can be given to either of the editors or left at the Apple Answer Desk. Include both your name and phone number with the article to help us communicate with you if problems arise.

We hope that these guidelines will not only help you but also make it easier for us to provide a more consistent and classy newsletter.

Advertising

Rates

Inside Covers	$100
Full Page	$80
Half Page	$60
Third Page	$40
Business Card	$15
Other	$5 per column inch

Ads are to be prepaid, on disk, and received by the third Saturday of the month before publication. Contact either of the Editors for further information.
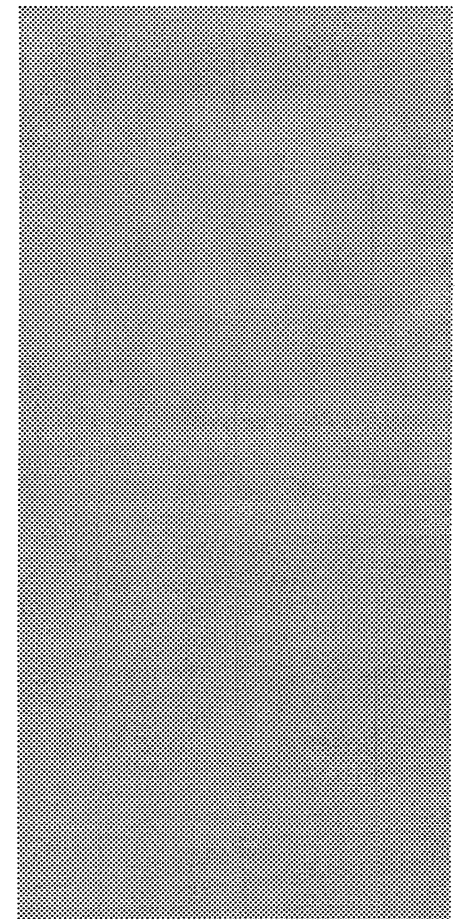
Reminder!

The July swap meeting is on a Fourth Saturday (the 26th) at the Marriott.

The August meeting takes place at the Astro-Village Inn on the 16th.

"All right, I'll write!"

As this month's *Apple Barrel* nears completion and as the other editor skims through and sets the format for the issue, I thought I would take some time to write my editorial. Like the first *Barrel*, publishing the second monthly issue has proven to be a challenge. Therefore I would like to talk to the readership about one way that they can really help to insure more monthly *Barrels*. Write!

We have several writers now that have, are, and will continue to write articles. However, since this club is composed of you the members, we would like to see input from you. Unlike the nationally published magazines that you pick up in the bookstore or at the newsstand, this newsletter is written by and for clubmembers. That fact is important in several ways. Through the *Barrel*, you can voice your opinions and explain new techniques you have just discovered. The newsletter can provide a forum for discussions about topics ranging from programming languages to getting the most out of *AppleWorks* or *Excel*. It is also an ideal place to pose questions and to get them answered. Furthermore since most magazines tend to get facts to its readers approximately three months after the fact, the *Barrel*, by being published locally as well as monthly, can use the knowledge of the general membership to publish facts which are both timely and relevant.

I will speak for the other editor by stating what we would like to see in the *Barrel*. The *Barrel* has the potential to be a wonderful source of information. It would be nice to see the time when the average reader on having a problem with his computer or software might say or think, "I saw something about that in the *Barrel*!" Since the problems and experiences of the average reader are generally not unique to him, feel free to dash off a few lines or paragraphs explaining tips, techniques, or

solutions. If you have questions, pose them to the *Barrel*. We will try to get the questions before the membership so other users can help. If you use a product and would like to let more members know about it, write a review of it. Remember that the more people that use and know about a product, the more likely there will be someone to answer a question about it. (For those interested in reviewing a product, a formal review committee has been established and will coordinate the review of products with members. Contact Vice President Mike Conway for more details.)

By talking to the membership at large, we are showing our desire for articles from all users—new users as well as seasoned veterans. Because many readers are inexperienced with the computer, they generally want to read about the problems other people are having and about the solutions to those

---

The Barrel has the potential to be a wonderful source of information.

---

same problems. Don't feel shy if you get confused in a certain program or have difficulties computing. You are not alone! Your questions don't get answered until you pose them. We also seek the tips and advice that more experienced users can give to the membership. Even if you just want to express your sheer delight at using your computer from morning to night, go right ahead.

Although many people buy computers for home use, many users bought their computers to use in a professional capacity. If you did this, consider writing about your discoveries concerning the computer use in your business.

Other professionals might have useful and creative tips helping you use your investment that much better.

Last, a great source of information is the experienced user. Sooner or later, computer owners will find that one user who always seems to have the answers to their questions and to know the latest news. If you have found such a source, encourage him or her to write an article. Just as you have benefited from his or her advice, let the club do so as well.

As this editorial draws to a close, I would like to make this appeal to the membership. To publish a newsletter that you can look forward to receiving each month, I just want you to consider the following...WRITE!

—Chris Flick

Note: Beginning with the July Issue, a question and answer column will appear in the *Barrel*. Please submit your questions at the Answer Desk during the meetings or mail them to:

Apple Barrel Q&A
PO Box 610150
Houston, TX 77208
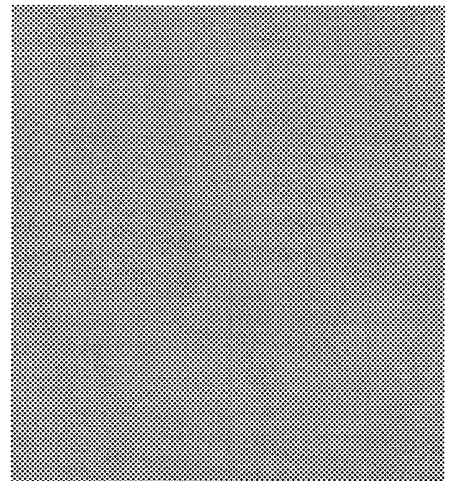
It Should Have Been 200...

Like many scientists I do a substantial amount of writing. Four years ago I decided to buy my first personal computer so that I could do my writing at home during evenings and weekends, freeing up more time during the day to spend in the lab with my graduate students. I was not new to computers, having learned to program in ALGOL some twenty years earlier, but I was totally unprepared to enter the world of microcomputers.

My choice was limited. There was hardly any software available for the recently-introduced IBM-PC so I opted for an Apple /// which seemed destined to replace the less-than-adequate Apple][. The Apple ///, I was told, would run most of the Apple][software and with its SOS operating system and 128K of RAM was bigger and better in every respect. The salesman assured me that the *Apple Writer ///* software would satisfy all of my word processing requirements.

Perhaps I should have told him that I planned to process words like "bis(deuteromethylthio)tris(trimethyl-silyloxy)dihydrophenanthrene" which sometimes look a little odd on the page if they cannot be hyphenated. The *Apple Writer ///* software lacked soft hyphens and many other features that are almost mandatory for technical typing.

Fortunately, the latest issue of *Personal Computing* (August, 1982) had a good summary of the word processing software then available for microcomputers, and I was able to discover that *Spellbinder* had more desirable features than any of the other packages. This was the software that I had to buy, but it was not that simple. *Spellbinder* ran only under the CP/M operating system and Apple had not yet introduced the SoftCard /// System with its Z-80 card. The salesman assured me that the release of this accessory was imminent, so I called each of the major computer stores every Saturday morning

(for five or six weeks) until it arrived in Houston. All that I needed then was the software. It was disappointing to find that none of the local computer stores was familiar with this panacea. A phone call to *Spellbinder's* publisher (Lexisoft, Davis, CA) revealed that they had only one dealer in Houston; an unemployed engineer working out of his home. He was more than willing to deliver the software in person, but it took him three whole evenings to modify it to make it compatible with Apple /// CP/M.

I struggled with *Spellbinder* for a couple of weeks. Typing and printing simple text was easy, but the commands required to unleash the advanced features that I needed were almost impossible to fathom. The manual was the size of a telephone directory, but it was written in gibberish ("FT: Fetch YT Table values and install in text as .yt statement" and similar cryptic statements). Like much of the software from that era there were no instructions on how to actually use it.

I soon switched to *WordStar* because, even though it was less sophisticated than *Spellbinder* and its manual was no more intelligible, there were several well-written books available to guide me through its intricacies. Again I had to track down someone who could patch it so that it could be used with Apple /// CP/M, but this was less of a problem. *WordStar* works well with my NEC 3510 Spinwriter printer (which is ideal for combining symbols and plain text), and the system has paid for itself many times over. I frequently use *Access ///* to dial up remote computers and download bibliographic data, but I have never been able to figure out how to convert SOS files to CP/M to insert the data into *WordStar* files.

Throughout the two-month period that I was struggling to assemble a

functioning word processing system I must have spoken to salesmen (and a few saleswomen) at more than a dozen computer stores, yet not one of them mentioned HAAUG. When I acquired my first Macintosh (I now have two—Plus at home, 512K at work) a little over a year ago I was considerably more fortunate. A colleague introduced me to HAAUG, and I learned more about the Macintosh at my first meeting than I had ever had the opportunity to learn about the Apple ///. I joined HAAUG immediately and became the proud recipient of membership card number 2000. It should have been 200...

(Now can anyone explain how to transfer partially-written manuscripts from my Apple /// CP/M files to the Macintosh without a major investment in new Apple /// software??)

—Brian S. Middleditch
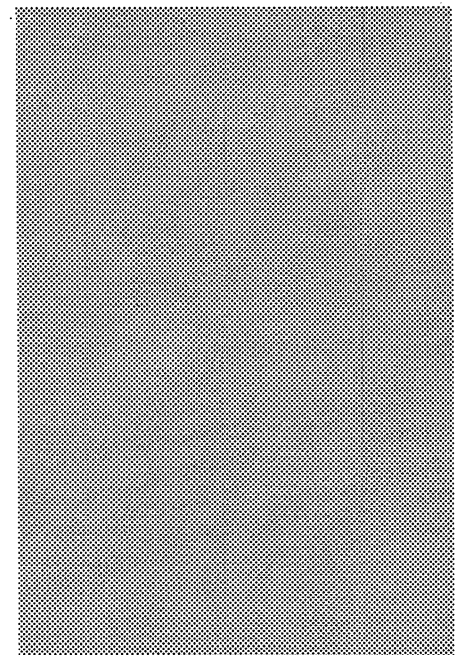
Random Thoughts

What Support?

A few months ago, Apple Computer announced that it would be dropping about 25% of its authorized dealers. Supposedly one of the reasons for this move was that the revised dealer network would provide "better support" for computer buyers. Computer columnists often assert that buyers should be willing to pay higher prices for computers and software at computer stores because the stores provide support for the user. I view this assertion somewhat skeptically because my experiences with dealers have been decidedly mixed.

I bought my 512K Macintosh at Businessland over a year ago. Businessland wasn't my first choice for a dealer. I originally planned to buy from the Computerland in Pasadena. This store is one of the least impressive places to computer shop. The day I first arrived in the store, all the salesmen were sitting around smoking cigarettes and playing with computers. I must have stood around for ten minutes before someone, without standing, asked me if he could help me. Even then, the salesman acted as if I were an annoyance rather than a customer. When I asked for a Macintosh demo, the salesman scowled and moved over to a Mac were he proceeded to give a perfunctory demonstration. When I asked to see a sample print-out from the ImageWriter, he discovered that the printer wasn't even connected to the computer. Yes, great support.

I should have written that Computerland off, but the price the salesman finally quoted me was $200 less than I'd seen anywhere else. I got a written quote and went and arranged a loan. About ten days later, I returned with my checkbook to buy my Mac, but the guy I'd talked to was on vacation. "Oh, no," the new salesman said, "the price is $200 more than that."

Next stop, ComputerCraft—money in hand. The salesman here was much friendlier and more knowledgeable and enthusiastic about the Mac, but for some reason ComputerCraft insists on sticking a plastic ComputerCraft logo on the front of all the computers they sell. To some people this probably isn't a problem, but to me it is (I once almost refused delivery of a car I ordered because the dealer initially wouldn't agree to not rivet a dealer advertisement on the rear of my new car). Could you get me one without the ComputerCraft ad, I asked. No way; can't be done. They're put on at the warehouse. Yes, great support.

Businessland. They had one in stock, no plastic Businessland logo stuck to the front, thank goodness. The salesman was friendly, but he didn't know much about Macs. For example, he told me that the Programmer's Switch (brown plastic, left side) was a "disk stand." A few weeks later, I read

> *For example, he told me that the Programmer's Switch...was a "disk stand"*

in a computer publication that Apple would be distributing a free system and *MacWrite/MacPaint* software upgrade. I took my disks by Businessland to check on the upgrade. My salesman was there, but he didn't remember me, nor did he know anything about software upgrades. If I brought my official disks in, could I get the upgrades? Yes, if you bring your purchase receipt so we know you bought it here, and if there's no charge. Great support. I quit driving across town to visit that dealer.

My subsequent dealings with the Southpoint ComputerCraft, the store which is closest to my home, have been much more satisfactory, but there are still annoying gaps. Unlike Businessland, ComputerCraft cheerfully upgraded my software without any questions when the upgrades finally arrived. ComputerCraft now has a Mac specialist, who makes it his business to know what goes on in the Mac world. Yet, when I had my Mac power supply replaced, the Mac came back with the screen image misadjusted.

After four ComputerCraft tries at getting it right and three new power supplies, I finally opened my Mac up and fixed the problem myself (a simple matter of turning a couple of adjustment potentiometers). The technicians know how to use test instruments, but they don't know how a Mac screen is supposed to appear. During this ordeal, I had my honesty questioned several times. When I first brought my Mac in for repairs, the salesperson wrote the wrong serial number on the repair order. When I returned it, I was accused of substituting a second Mac for the one I originally had repaired. This happened three times, and the supervisor out at the ComputerCraft Tech Center was downright nasty. Great support.

In relating these experiences, I don't mean to suggest that all dealers are venal or incompetent, but I have to wonder what Apple and the computer columnists are talking about when they exault support. I've had one free software upgrade in 15 months, but is that worth the $500 or so difference between the official Mac price and the gray market price? I doubt it. When they talk about support, do they mean repair service? I don't think the computer stores lose money on repairs, and they get reimbursed for warranty repairs anyway. Are we talking about friendly, knowledgeable salespeople? I read *InfoWorld, Macworld, MacUser, Macazine, MacTutor, A+, Byte,* and *Personal Computing.* The salespeople

usually know less than I do. Why don't they read these magazines; after all, it's my hobby, their job? They certainly don't know enough about Mac software to justify paying them list price for software you can mail order at a 30-50% discount.

Salespeople in the chain computer stores want that big score. They want to sell a dozen computers to an accounting firm or a school, not one to an individual. That's why the salespeople are always on the phone in the back room and why you have to wait so long to get waited on when you want to buy something. You can't really blame them for that, however; the effort to sell ten Macs at a time is probably the same as the effort to sell one, and most of them work on some sort of commission. Nevertheless, I don't see what these dealers do for me that is worth an inflated initial price.

Purchasers should have the option of doing without this overvalued support. I'd pay five or ten dollars for five minutes of a salesman's time and a software upgrade, if I could save $500 up front. I'd like to have the option of

---

> ... *"computer*
> *literacy" isn't much*
> *use without plain*
> *old "literacy."*

---

buying an Apple computer at Best Products or through mail-order. You can buy well-engineered Sony televisions and stereo equipment through a number of channels, but Apple computers (the ones with generous 90-day warranties, anyway) can be bought only through some 2500 (soon 2000) authorized dealers.

As far as I'm concerned, "support" is a bogus issue. Apple limits the number of dealers because they want to restrict supply and thus keep prices and margins high. Since dealers provide repair service, Apple doesn't need to maintain its own repair centers (Sony and GE both have them). When Apple provides free (or cheap) Mac system software upgrades, it's not because Apple is a benevolent company, but because every Mac in the field is an

advertisement. Better system software running on 500,000 Macs will make additional sales more likely.

Ironically, I suspect that Apple's dealer reduction will probably result in decreased rather than increased support for users. As I understand it, Apple's main criteria for deciding which dealers get dumped is sales volume. The stores with really good support, the Mom-and-Pop computer stores run by enthusiasts, are thus more likely to get dropped than big volume chains like Businessland or Computerland which are generally indifferent to the needs of single-computer customers.

For my part, I'll continue to mail-order software and only buy stuff at computer stores when I must. I feel no particular obligation to subsidize undeserving computer store operations in the name of something as ephemeral as "support."

Software Poor?

When the fifteenth of the month rolls around, my computer magazines start to arrive in the mail. Typically, when I open a magazine, I begin to read the ads to see what's new in software. Pretty soon, I begin to salivate and gnash my teeth. Unfortunately, I'm poor; I can't afford all this stuff. Alas, I just don't have enough software.

I mean, I've got a database management system *(Double Helix)*, a word processor *(MacWrite)*, a page processor *(ReadySetGo)*, a couple of painting programs *(MacPaint* and *FullPaint)*, a drawing program *(MacDraw)*, a spreadsheet *(Crunch)*, an outline processor *(ThinkTank 512)*, a spelling checker *(MacSpell+)*, a programming language *(MacPascal)*, and a bunch of utilities and games. I've got 20 disks of shareware or public domain stuff.

Sounds like a lot? Piffle. What about statistics programs? I don't have one. Clip art? Nada. A 3-D graphics program? Nyet. A communications program? Nein. An accounting package? No. A chart drawing program? Non. Languages? I have one, an interpreter; I want a compiler. Actually, I want several compilers. *Lightspeed C, TML Pascal, ExperLisp, ExperLogo, Mac Modula-2.* I want all of them. Games? I have eleven; there must be

hundreds. I want all of them. *Factfinder?* I want it. Doug Clapp's *Word Tools?* I need them.

I have a hard time believing that Mac software sales are as bad as John C. Dvorak says they are.

Looks nice, reads terrible, or good software, bad results.

A while back, I attended one day of a two-day Apple expo at the University of Houston. The focus was on the Macintosh in education. The exposition was sparsely attended. Worse, some interesting lectures were given to almost empty side rooms. One of the lectures was by a member of the Courseware Development Group at Dartmouth College.

Dartmouth, a member of the Apple University Consortium, strongly encourages its students to buy personal computers. Most of Dartmouth's dorm rooms are wired for AppleTalk. Some 80% of Dartmouth's 1984 freshman class bought computers.

I asked the Dartmouth speaker, whose name I can't recall, whether the faculty had noticed any appreciable improvement in student writing skills that could be attributed to the presence of so many computers on campus. He replied that while papers tended to be neater, the quality of thought and expression hadn't improved tremendously.

Clearly, computers by themselves aren't enough to improve the state of education; "computer literacy" isn't much use without plain old "literacy." Incoherent gibberish, remains incoherent whether it's printed on a LaserWriter or by hand on a Big Chief tablet. Turgid business reports remain turgid even if the reports are embellished by *Excel* spreadsheets and charts.

Similarly, Apple's new emphasis on desktop publishing as a major market niche for the Macintosh suggests that all managers need is a Mac and a Laserwriter and they can fire their graphic artists and produce professional-looking publications. Look around at some of the results, and you'll see that this is a vain hope. For example, I receive a newsletter called *Helix User News* which looks

amateurish even though it's produced using *PageMaker* and printed on a LaserWriter. For a counter example, look at *MacTutor*, which looks terrific, even though the design is very conservative.

What I'd like to suggest is that if someone out there is considering a personal publishing project, he should do some research: brush up on his grammar and composition skills and study graphic design a little.

For people who need help with their writing, I'd like to suggest a couple of useful books. Karen Elizabeth Gordon's book, *The Transitive Vampire*, is the neatest grammar book I've yet encountered. Gordon has a wicked sense of humor, and *The Transitive Vampire* is both entertaining and informative. Hands down, the best book on writing I've ever seen is *Writing with Style: Conversations on the Art of Writing* by John Trimble. Trimble was a professor at the University of Texas when I was in English graduate school ten years ago. *Writing with Style* is clear and non-dogmatic. Trimble's writing is so good, it's like he's standing there talking to you, which is of course why the book is subtitled *Conversations*.

While I know a little about writing, I am strictly a neophyte graphic designer. Nevertheless, I'd like to recommend several books that I think will help you make your publication look better. The first book is called *Do-It-Yourself Graphic Design*, published by Facts On File. While the title is rather mundane, the book itself is excellent, particularly the chapters on designing books and magazines. Much of the book is irrelevant for people using page-composition software, but you'll want to read it all so you'll appreciate *PageMaker* or *ReadySetGo* even more. Scribners has published a whole series of books called the *Manual of Graphic Techniques*. Volume 3 in the series contains more information about page layout, typefaces, and grids.

Grids are an important graphic design concept. Most professional publications are built around a grid, which is the underlying structure of the page design. The grid provides unity for a publication; though the actual page layouts may vary, the grid remains the

same and the publication maintains a consistent look. Everybody who designs a publication should know something about grids. Allen Hurlburt has written a whole book on grids called, curiously enough, *the grid: A modular system for the design and production of newspapers, magazines and books*.

As far as I know, all five of these books are still in print.

Copy Protection

Franz Kafka would appreciate copy-protection; so would Yossarian, of Joseph Heller's *Catch-22*. The current copy protection situation is as absurd as anything in existential literature. Software publishers spend a whole lot of money improving copy-protection schemes, which people like Central Point Software promptly defeat. Then Central Point sells copy programs to users, and the whole cycle begins again. As far as I can tell, the only people who benefit are the people who create and defeat copy-protection schemes.

I thought of this because yesterday I finally received a copy of *FullPaint*, a fine new drawing program from Ann Arbor Softworks which I ordered about a month ago. *Copy II Mac*, version 4.5 wouldn't copy *FullPaint*, so what did I do? Simple, I dropped an $18 check to Central Point for the latest version of *Copy II Mac*. I'm sorry, but there's no way I'm going to insert the master disk every time I fire up the *FullPaint* program, nor am I going to use the original disk, nor am I going to pay $15 for a back-up. As far as I'm concerned all three alternatives are unreasonable. I'm suffering, and I bought (sorry, licensed) the program.

So what has Ann Arbor Softworks accomplished by copy-protecting *FullPaint*? Nothing constructive. Eventually, I'll be able to make a back-up copy which won't require a key disk or cost $15. I'm mad at *FullPaint's* publisher; I don't like *FullPaint* as much as I would if it weren't copy protected. The legitimate user and the publisher screw each other; thieves will copy it anyway. Copy protection is dumb.

Fortunately, most of what software I own is not copy-protected. Publishers who ship unprotected software earn

points with me. For the rest, *Copy II Mac* will defeat most schemes; some programs can even be completely deprotected using patch programs. I don't object to software publishers protecting themselves from thievery, but they need to find a way which won't punish legitimate users. Odesta Corporation has a clever alternative to traditional copy-protection—my copy of *Double Helix* is stamped with a serial number and my name. You can bet I'd think twice before giving that program to anyone.

What's even more absurd is that though some software publishers copy-protect their products, they cynically position themselves to benefit from software piracy. If you doubt this, go down to your bookstore and look at a copy of a $22.95, 695-page book called *Excel in Business*. Thumb through it. *Excel in Business* is very thorough, and even includes a tutorial. No question about it—*Excel in Business* is nothing more than a user's manual for *Excel* thieves. The publisher? Microsoft Press.

—Cleland Early

The Cauzin SoftStrip Reader

A number of years ago I was intrigued by the idea promoted by *Byte* magazine that program listings could be printed in bar code in their magazine and entered into a computer using a bar code reader. They even published construction articles on how to build crude bar code readers. The idea never got off the ground. When advertisements recently appeared for a relatively low cost product called the Cauzin Softstrip System, my interest was immediately aroused. When *Nibble Mac* magazine began printing their program listings in the Softstrip format I concluded that it was time to investigate. I ordered a Softstrip System.

Versions of the Cauzin Softstrip System are available for the Apple //, all versions of the Mac, and (ugh) the IBM. The system includes a reader, a power cube, a cradle for protecting and storing the reader, and an accessory kit. The accessory kit includes a cable, software on disk, a manual, and a sampler library of software on strips. The Mac sampler includes a variety of programs, documents, and *MacPaint* pictures such as *Amortization, Secrets of the Finder, Visible Engine, Icon Maker, Excel Macros*, and others from publishers such as Addison Wesley, Nibble, Family Computing, Osborne/-McGraw-Hill, and Wiley. Most of the programs in the sampler are written in Microsoft *BASIC* or Macintosh *Pascal* and require that the user have those languages to try them. Most of the Apple // programs are written in Applesoft.

The Softstrip System is based on an optical scanning device that reads encoded information in the form of a pattern of black and white rectangles printed on paper and saves the information as a file on disk. An example of a Softstrip is shown on the left of the dialog box in Figure 1. The reader is a 15-inch long tubular device that is aligned over softstrips with a moving optical scanning device inside that interfaces to the computer's serial port. Since the reader is externally powered, it will work with the Macintosh Plus, although the circular 8-pin adaptor is not included.

The softstrip containing the information looks much like a seismic log used in oil and gas exploration and can contain text, binary data, pictures, or digitized sound. Up to 5500 bytes of information can be contained on each strip in the highest data density. The information is read into the computer by aligning the reader on the softstrip and telling the computer to read.

Sophisticated error detection and correction techniques are used to assure accurate reading of information. Depending on the amount of information represented, one or more strips may be required. Reading of a document from a softstrip is simple, quick, and painless. For purposes of illustration, the process on a Mac will be described. Operation of the Apple // or IBM should be similar.

Connecting the hardware on a Mac involves plugging the interface cable into a telephone style modular connector on the reader and to either the printer or modem port on the back of the Mac. After booting the disk, the reader application called *Cauzcom* is run. The screen in Figure 1 appears and instructs you to align the first strip and click on the "Read Strip" button when ready.

A menu option is provided to permit designation of which serial port is being used. The selected port can be saved as a default. Another menu option permits selection of file conversion method. As a default MacBinary files are automatically saved in MacBinary format. Other formats are saved as text files. Files created for the Apple // or IBM are stored as text files on the Mac, which provides a simple way to transfer files from those sysems. This means that softstrips published in Cauzin's ads in *PCWeek* can be read into the Mac. A Macintosh conversion mode is included for future expansion, whatever that means. You can also specify no conversion or conversion to text.

Alignment involves centering a black dot near one end of the Softstrip in a guide on the side of the reader and aligning a bar at the other end of the Softstrip with the edge of the reader. The strips are numbered to facilitate reading them in sequence. An alert dialog is displayed if an attempt is made to read the strips out of sequence.

As each strip is read, the arrow pointing to the Softstrip in the dialog box in Figure 1 indicates the progress. As soon as the first strip is read, the name of the file, its type, and its size in bytes are shown on the screen. If more than one strip is involved, instructions are given to align the reader over strip 2, 3, etc. until all strips are read. As each strip is read, its contents are written to the program disk as a temporary file. When all strips have been read, a standard dialog box appears asking for a file name and permitting the file to be stored on another disk. The temporary file is automatically deleted.

Cauzin's ads state that they will publish low cost software ($3 to $20) on paper. Among the first of these are Mac and Apple // versions of *Stripper*, a $20 program to permit printing Softstrips on the ImageWriter in the lowest density. A test of a 7000 byte compiled Lisa *Pascal* program required eight strips. An attempt was made to transfer *MacPaint* to strips, but the effort was terminated when a message on the screen advised that 68 strips would be required. Time to process and print strips seemed to be limited by the printing speed of the printer. Cauzin will soon release a LaserWriter version that should be able to print most if not all of the higher densities.

Tests were run on Cauzin's advertisements in *Nibble Mac*, on strips in the sampler provided with the system, and on a homegrown, compiled Lisa *Pascal* program which had been printed using the *Stripper* program. A few read errors were encountered reading strips printed in one of Cauzin's ads and in a *Nibble Mac* article which prevented reading those strips. At the suggestion of my wife, I read in the short strip on a Cauzin business card. To our surprise, it contained Cauzin's name and address along with the names of all those involved in developing the product.

Cauzin's response to customer problems is outstanding. The first unit (Mac version) is flawless, and support is outstanding. If you don't like to type, don't have the time to type, don't want to look for the bugs introduced when typing, etc., the Softstrip System is just what the doctor ordered. What remains to be seen is whether enough interest is generated in the product for it to become a standard. With *Nibble Mac, Mac User*, and a number of major book publishers committing to publishing Softstrips in their publications, I think it will make it.

Cost of the Cauzin Softstrip System is $200 including choice of accessory kits. Additional kits are $20 including cable and software, permitting
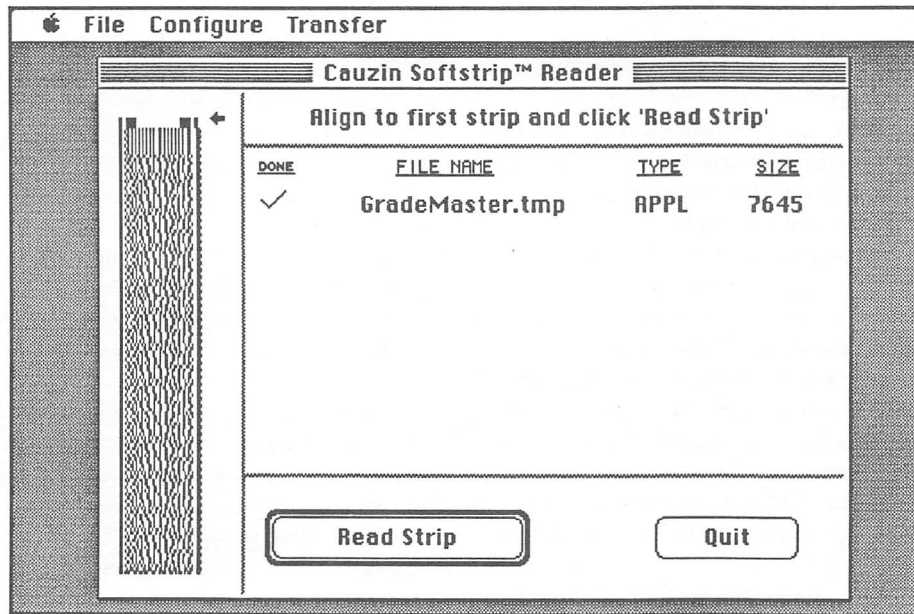

Figure 1. Sample Dialog Box

received would not read, giving alignment errors. It was possible to move the reader around and cause an undocumented bad cable error message to be displayed. I called Cauzin technical support, who, expressing surprise at the undocumented message and promised to ship a new cable immediately. The next day, my wife was able to get the reader to read part of a strip before getting an error by rocking the reader. Inspection revealed that the body of the reader was twisted along the long axis. A call to Cauzin resulted in an apology, a new reader and accessory kit, and a "no postage" return label. The replacement unit worked flawlessly.

My overall impression of the product? Cost is reasonable, hardware quality is good (if not warped), software

use of the reader with more than one type of computer. The Cauzin Softstrip System can be purchased direct from Cauzin at 835 South Main Street, Waterbury, CT, 06706, 203-573-0150, 1-800-5333-7323, or from local dealers.

—*Mike Kramer*

AppleWorks Spreadsheet Tips

With the invention of the spreadsheet concept the usefulness of the personal computer has increased far beyond what was ever imagined in the days when a good knowledge of BASIC programming was required for even the simplest applications. Today by using the spreadsheet portion of *AppleWorks*, anyone can organize his personal affairs, make quick calculations and save the results, run "What If?" scenarios and surprise-surprise to the IBM crowd, even run a fair-sized business.

In this column this month and in the months to come I will lead you through the *AppleWorks* spreadsheet and reveal tips that may be helpful to both the beginning user and the advanced user. If any reader has suggestions or questions I will be happy to share them with you.

Compared to a word-processor where the user writes on a blank sheet of paper, the spreadsheet is a collection of rows and columns very similar to a large piece of yellow ledger paper. The intersection of each row and column is called a cell and can be identified by the column letter and row number ie., A1, M201, AB56 etc. The useful size of the spreadsheet in *AppleWorks* or any other program is determined by the amount of memory or desktop that you have available. It is probable that you will eventually add memory to your computer so that you can do bigger spreadsheets.

Each cell may contain a label (text) or a value (number or formula). Any entry starting with a number or one of many of the arithmetical symbols (i.e. +, (, -) are values. Mixed entries starting with numbers but also containing text such as 28 April 1986, are not considered values and will cause the dreaded "beep"! To handle this situation it is necessary to convert your entry to a label. The easiest way to do this is to precede the entry with quotation marks, or to hit P and Delete

then retype your entry. P and Delete will blank a cell if you follow with Spacebar and Return.

Hitting the Spacebar as the first entry in a cell will also cause a beep. Spacebar, in addition to most of the symbols above the numbers (even the $), and other similar symbols are neither labels nor values. You must get into the label mode first, using quotes or P followed by Delete, then type these characters.

In the lower left corner of the screen *AppleWorks* will tell you the cell you are in and whether you are in the label or value (number) mode.

Standard settings for the spreadsheet are set using Open-Apple V (Values). Here you set the number of decimal places, use of commas, column widths, frequency and order of recalculation and whether you want text left-justified, right-justified or centered. Columns are nine characters wide unless you set them otherwise. The current standard settings can be viewed by pressing Open-Apple ?.

The spreadsheet will recalculate after every value entry. This becomes a pain in the neck if your spreadsheet is of any size. Using Open-Apple V, then pressing R (Recalculate) F (Frequency) and M (Manual) turns off the automatic recalculation feature. When the spreadsheet is on manual pressing Open-Apple K will cause the spreadsheet to reKalculate.

If you want an area of the spreadsheet to be different than the standard, press Open-Apple L (Layout). Open-Apple L will change a single cell or a group of cells. When setting the decimal point it will only work on entries already on the spreadsheet if you pick Rows or Columns. If you pick Entry or Block it will apply to an entry that has not yet been entered. If the Layout is different than standard, the screen will alert you down in the lower left corner with a message such as "F3"

when you are on the cell in question. (F3 means three decimal places.)

When you goof, and who doesn't, there are several ways to recover. Pressing Escape will remove the entry and pressing Delete will remove the last character as long as you haven't hit Return. After hitting Return you can press Open-Apple B and blank out the entry or you can press Open-Apple U and undo or correct the entry or you can just start over. I use all three methods more often than I care to admit!!

When using Open-Apple U to correct text be aware that it only corrects the text in the one cell. It does not continue on to the adjoining cells. If your correction changes the length of the text then it is better to retype from the left edge of the cell and overlap as necessary.

Open-Apple U is very useful in correcting formulas. If your correction won't work, the computer will beep when you hit return and not accept the correction. It is also possible by using Open-Apple U to add a few more characters to a formula that is too long for the machine to accept the first time.

One of the most useful features of the spreadsheet is the ability to copy text and values by using Open-Apple C. This feature will be covered next month.

Final tip—in the spreadsheet mode there is no difference between the Open-Apple and Closed-Apple. Use either one as you see fit.

—John B. Slack
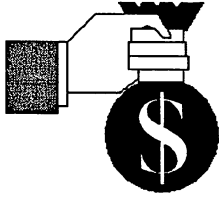
Announcing

FINANCIAL PRO

The newest financial template for use with AppleWorks. Enter the realm of high finance!! By using the **FINANCIAL PRO** spreadsheet templates with AppleWorks you can perform the same calculations that once were only done by bankers, loan officers, and sophisticated investors.

FINANCIAL PRO templates are easy to use. Once the financial data is entered the results are immediately available. By using the powerful AppleWorks spreadsheet, comparisons and "What If" scenarios can be presented.

Here are only some of the 21 templates that are currently included with **FINANCIAL PRO:**

Loan Payment Size & Interest	Depreciation Schedule
Balloon Payment	Depreciation Rate
Loan Payoff	Salvage Value
Mortgage Schedule	Future Value
Initial Investment	Minimum Investment
Annuity (Both ways)	Discounted Cash Flow
Future Value of Regular Deposits	Discount Paper
Maximum Regular Withdrawals	Effective Interest
Minimum Required Deposits	Nominal Interest

Plus the most widely used spreadsheet of all-Checkbook!!

By using the Checkbook template you never habe to worry about checks that are out of sequence. Amounts of deposits and withdrqwals can be corrected at any time and **FINANCIAL PRO** will automatically recalculate your balance. By using account codes and the soritng feature of the AppleWorks spreadsheet it is easy to categorize your transactions for tax purposes, budgeting, etc.

FINANCIAL PRO will soon be marketed nationally and will cost $37.95.
Now for a limited time HAAUG Members can buy **FINANCIAL PRO** for only $20.00.

Send check or money order to-
    Pro Products
    1179 Bournewood
    Sugar Land, TX 77478
    Phone 713-491-1747 after 6:00 PM

BASIC Under ProDOS

This month I'll be demonstrating how to store information in sequential text files on a ProDOS disk.

Sequential Text Files

Storing information in a sequential text file is much like writing a message in the sand on a beach. You can write and write, never starting a new line, and just continue on down the beach. However there is a problem with writing in this method. How do you break this long stream of words into smaller sections?

Well, in English we seperate words with spaces and sentences with periods. Often in poetry, editors will seperate lines with /'s, such as "Mary had a little lamb/Its fleece was white as snow". These are merely ways of breaking these streams of information into more manageable chunks. These same tricks are used in text files. Let us look at the program.

Line 280 asks ProDOS to OPEN a sequential text file called "SEQ.FILE" on the disk. If the file already exists then this command merely tells ProDOS that this is the file with which you wish to work. If "SEQ.FILE" does not exist then ProDOS will create it. The WRITE command in line 290 instructs ProDOS to divert all PRINT statements to this file. That means that subsequent PRINT statements will print information into the file instead of printing information on the screen. Now things get interesting, line 330 puts the words "FIRST FIELD" into the file. The file now looks like...

FIRST FIELD«

The "«" symbolizes a carriage return; most print statements are terminated with a carriage return. It is the RETURN character that we use to break the information up into sections, which we call "fields". Thus the RETURN characters are used much like

the /'s in poetry.

Line 340 adds another field to the file.

FIRST FIELD«SECOND FIELD«

Now things get a little trickier. Some PRINT statements work differently. PRINT statements that have a semicolon after them are **not** terminated with a RETURN character. Therefore the two print statements of line 350 add only one field to the file.

FIRST FIELD«SECOND FIELD«FIRST HALF SECOND HALF«

Notice that there is no RETURN character after the words "FIRST HALF" since its PRINT statement ends with a semicolon. There is, however, a RETURN character after the words "SECOND HALF".

"But wait, there's more!"

Just as sentences can be broken down into phrases, a field can be broken into several smaller sections called "elements". Each element in a field is seperated by a comma. Thus, line 360 creates a fourth field that is comprised of two elements.

FIRST FIELD«SECOND FIELD«FIRST HALF SECOND HALF«ELEMENT ONE, ELEMENT TWO«

To some degree, elements are just smaller fields; we use the commas to break up groups of information. Why do we even worry about breaking up information in a file? Mainly to assist us in reading the information back from the file. We'll get to this later.

Please note: What goes up must come down or...close what you open. When we are finished placing information into the file, we must close it with a CLOSE command. Often the information is not written to the disk

until you close the file.

Back to READing...

Let's skip to line 480 to see how we open a file up and read the information in it. Again we OPEN the file but this time we follow it with a READ command. The READ command instructs ProDOS to divert all INPUT statements from the screen and keyboard to the file. This means that subsequent INPUT's will fetch their information from the file either a field or an element at a time. This allows us to store the information from a file in many variables, each containing a field or element.

The first input statement (Line 500) will set the variable A$ equal to the first field. Thus A$ will equal "FIRST FIELD". Accordingly, line 510 will fetch the next field or element of information setting B$ equal to "SECOND FIELD" and line 520 will set C$ equal to "FIRST HALF SECOND HALF". Now for the elements. Line 530 will set D1$ equal to the first field or element (in this case an element); thus D1$ will equal "FIRST ELEMENT" and then D2$ will be set to "SECOND ELEMENT". If we were to issue another INPUT statement then we would get an error. You cannot read more fields than there are in the file. Again, line 540 instructs ProDOS to CLOSE the file when you are finished reading it.

Onward through the fog...

There are two main disadvantages with sequential text files. One is reading just the nth field. You must read all of the previous fields to know when you have reached the nth field. Secondly, it is difficult to replace fields with different information. For instance, say I had a file—

ONE«TWO«THREE«FOUR«

SEQ.MAKER Listing

```
100   REM
110   REM   |
120   REM   |   Program: SEQ.MAKER
130   REM   |
140   REM   |   Written by Dr. Arthur S. Cranston, Jr.
150   REM   |
160   REM   |   Purpose: The program creates a sequential
170   REM   |   text file called SEQ.FILE and puts
180   REM   |   4 lines of information in it and then reads
190   REM   |   it back and prints it on the screen.
200   REM   |
210   REM
220   HOME
230   PRINT "Writing information to disk. (Nonsense)"
240   REM   ==================================================
250   REM      Create a Sequential TEXT file.
260   REM   ==================================================
270   REM
280   PRINT   CHR$ (4);"OPEN SEQ.FILE"
290   PRINT   CHR$ (4);"WRITE SEQ.FILE"
300   REM
310   REM      Put 4 lines of information into the file.
320   REM
330   PRINT "FIRST FIELD"
340   PRINT "SECOND FIELD"
350   PRINT "FIRST HALF ";:PRINT "SECOND HALF"
360   PRINT "ELEMENT ONE, ELEMENT TWO"
370   PRINT   CHR$ (4);"CLOSE SEQ.FILE"
380   REM   --------------------------------------------------
390   PRINT "The information has been written to disk."
400   INPUT "Hit RETURN to continue ";N$
410   PRINT
420   PRINT "Reading information from disk"
430   REM
440   REM   ==================================================
450   REM      Read in information from SEQ.FILE
460   REM   ==================================================
470   REM
480   PRINT   CHR$ (4);"OPEN SEQ.FILE"
490   PRINT   CHR$ (4);"READ SEQ.FILE"
500   INPUT A$
510   INPUT B$
520   INPUT C$
530   INPUT D1$,D2$
540   PRINT   CHR$ (4);"CLOSE SEQ.FILE"
550   REM   --------------------------------------------------
560   PRINT
570   PRINT "The contents of SEQ.FILE are..."
580   PRINT
590   PRINT A$: PRINT B$: PRINT C$: PRINT D1$:PRINT D2$
600   PRINT
610   PRINT "The END"
620   END
```

and I wanted to replace "TWO" in the second field with "FIVE". If I merely replaced "TWO" with "FIVE" I would get

ONE«FIVE«HREE«FOUR«

which alters the information in the third field. I would still have a file with four fields, but the third field now contains "HREE". This is because the word "FIVE" is longer than "TWO". Now what if I wanted to replace "ONE" with "1"? The file would look like

1«E«FIVE«HREE«FOUR«

Now I have accidently added another field. For programs that need to change information in this manner there is another useful text file called a Random Access Text file which I hope to cover next month. Until then...

—A. S. Cranston, Jr.

ASCII Express Pro Macros and the HAAUG Heaven BBS

ASCII Express Pro is probably the single most versatile communications and terminal program available for the Apple II series of computers. One of its strengths is its macro capability. Macros are sets of predetermined commands, instructions, or text for use with the program or the remote bulletin board.

The most common use of macros is for automatic logon and logoff of bulletin board systems. This article will provide you with a set of *ASCII Express Pro* macros to log on and off of the club's *HAAUG Heaven* bulletin board. It assumes that you are already familiar with and using *ASCII Express Pro* but are entering your logon sequence by hand each time you call in.

To start, boot *ASCII Express Pro* and run the *Install* program. Press F from the main menu to set up the Macro Action Characters. Select J and change the "String Search Begin Char" from "<" to "{". Likewise change the "String Search End Char" from ">" to "}". Press X to exit to the main menu and S to save the changes and run *ASCII Express Pro*. This step is necessary because *HAAUG Heaven* uses the ">" character to prompt for ID and Password.

Note that after making this change, any time the manual says to use "<" and ">" in a macro, use "{" and "}", respectively instead. Likewise, you will need to change any existing macros that use the old characters.

Now from the *ASCII Express Pro* main screen, enter the command mode and press U to enter the macro definition area. If you are using an internal modem such as a Hayes Micromodem, then press P to enter the phone number. Enter:

```
+664-9873!1 <CR>
```

This tells *ASCII Express Pro* to turn on the modem speaker if it has

one, dial the number, and then go to macro #1. Remember, this step is not for users who have external modems and serial ports (Apple //c) or serial cards.

Next press D to go to the Edit/Define Macros sub menu. If you did not set up the phone number above, press "0" and enter:

```
ATX2DT664-9873'\U1 <CR>
```

This macro is for external Hayes Smartcom compatible modem users. It tells the modem to use the extended (X2) response set and to touch tone dial 664-9873 (leave out the T in DT if you do not have tone dialing). The "'\U1" tells *ASCII Express Pro* to send a carriage return after the phone number and go to macro #1.

Now press "1" and enter:

```
{Heaven} {-->}(your ID#)'{--
>}(your password) <CR>
```

Substitute your correct ID# and password for the "(your ID#)" and "(your password)", respectively. This macro waits until "Heaven" is transmitted by *HAAUG Heaven*, sends a

space to skip the *HAAUG Heaven* equipment list, etc., waits for "-->", transmits you ID#, waits for the next "-->", and sends your password. This macro will complete the log on procedure automatically.

Press "9" to enter the log off macro and enter:

```
T'{sure ?}*Y'{NO CARRIER}*A
TH <CR>
```

Skip the {NO CARRIER}*ATH if you are not using an external modem. This macro transmits a T <CR>, waits for the "Are you sure ?" question, transmits a Y <CR>, waits for an external modem's NO CARRIER message and transmits ATH to be sure the external modem hangs up the phone.

In summary, the macro menu should look like this for external modems:

```
#0: ATX2DT664-9873'\U1
#1: {Heaven} {-->}ID'{--
    >}Password
#2:
#3:
#4:
#5:
#6:
#7:
#8:
#9: T'{sure ?}*Y'{NO CARRIE
    R}*ATH
#;:
#::
```

Press <CR> to return to the main Macro menu. Now press W to save the new macro to disk. Enter a file name of your choice at the prompt.

The last step is to add the new macro to the Macro Library. Press X to return to the main *ASCII Express Pro* prompt, again enter command mode and press Y to start the Editor. At the line number prompt >0 type:

```
.GMACRO.LIB <CR>
```

to load the MACRO.LIB file from disk. Decide what letter you want to use to identify the *HAAUG Heaven* macro (H is a good choice!) and type the following:

```
H/HAAUG Heaven/(the name
```

you used to save the macro) <CR>

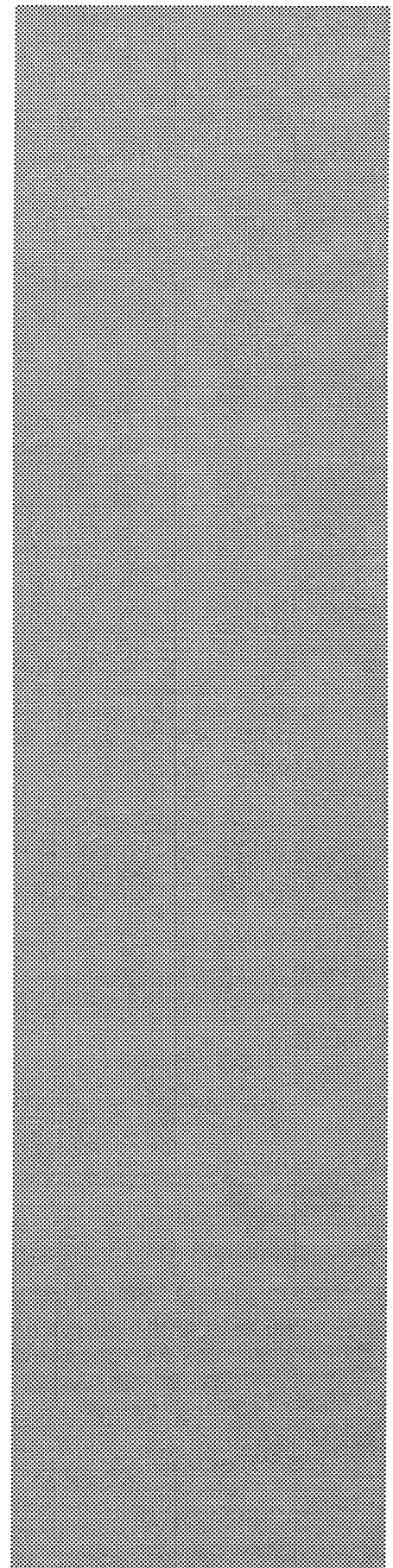Now type:

```
.pMACRO.LIB
```

to save the file back to disk, and .X to leave the Editor, and we're done!

Activate the macro from command mode by pressing <CR> at the +> prompt and then pressing H to load the macro. Now press <Ctrl>-W 0 (external modem users) athout any help from you. To log off, type <Ctrl>-W 9 from the *HAAUG Heaven* main command line.

Good luck and have fun!

—Rick Oshlo

Apple //

Apple Access	Patrick Hennessey	342-0643
AppleWorks	Jack Cowart	467-4215
AppleWriter //	Robin Cox	778-1635
AppleWriter ///	Shorty Hirsekron	697-9797
CAD-Apple	Dick Fairman	723-9619
dBase //	H. Helm	529-7405
Data Factory	Dick Fairman	723-9619
Dollars and Sense	Leon Cesazini	467-3650
DowJones Market Analyzer	Ed Hurley	497-8877
Home Accountant	Rick Oshlo	578-8530
IAC Calc	Chris Kravits	320-0381
Letter Perfect	Bill Muhlhausen	668-3963
List Handler	Rick Kravits	320-0381
MCI Mail	Patrick Hennessey	342-0643
Multiplan	Mackie Brown	688-8655
	Jack Bailie	440-5963
	Glen Ray	359-2103
Palantir	Re Kramer	358-6687
PFS	Jack Cowart	467-4215
	Re Kramer	358-6687
	Ed Haymes	522-1863
Pie Writer	Ray Fielding	358-4217
Quickfile	Jack Cowart	467-4215
ScreenWriter //	Dick Fairman	723-9619
Space Eggs	T. Sloan Engle	774-9055
Time Is Money	Dick Fairman	723-9619
VersaForm ///	Shorty Hirsekron	697-9797
Wordstar	Ed Haymes	522-1863
Word Handler	Rick Kravits	320-0381

Would you like to be a specialist? Leave your name and phone number at the Answer Desk during the meeting or mail it to the *Apple Barrel*.

Macintosh

Aztec C	Mike Martin	486-9169
DB Master	Mike Conway	495-2292
Dollars & Sense	Gerald L. Penn	498-0079
	Greg New	568-0290
Filevision	Mike Martin	486-9169
Helix	H. Helm	529-7405
Mac-SIG	Steve Bass	847-4407
MacDraw	Billy Jacobs	480-4410
MacDraft	Billy Jacobs	480-4410
MacPaint	Bill Hailey	682-5147
MacWrite	Bill Hailey	682-5147
Megamax C	Chris Flick	363-3153
Microsoft BASIC	Bill Hailey	682-5147
Microsoft Chart	Mike Martin	486-9169
Microsoft File	Gerald l. Penn	498-0079
	Greg New	568-0290
	Mike Martin	486-9169
Microsoft Multiplan	Mike Conway	495-2292
	Peter Lemettais	523-1390
Microsoft Word	Virginia Conway	495-2292
OverVue	H. Helm	529-7405
TML Pascal	Grady Beaird	529-8420
Tekalike	Mike Martin	486-9169
XLisp	Mike Martin	486-9169

Hardware

Abatron Optical Character Reader	H. Helm	529-7405
Mac XL	Mike Conway	495-2292

Macintosh Plus with KanjiTalk Introduced

Apple Computer Japan has introduced the Japanese Macintosh Plus personal computer complete with a state-of-the-art Japanese operating system.

Two years in development, the KanjiTalk operating system gives Japanese users access to the three traditional Japanese alphabets plus the English alphabet.

"KanjiTalk is unequaled in the Japanese marketplace because of Macintosh's ease-of-use, advanced features and powerful graphics," according to Alexander van Eyck, General Manager, Apple Japan.

"KanjiTalk offers immediate access to a powerful library of Macintosh software which can be easily translated (localized) from its original language to Japanese. Perhaps more important, KanjiTalk also provides a development environment which allows local software developers to create Japanese software products," van Eyck adds.

Within a month, Apple Japan will begin to ship fully localized versions of Apple software, including *MacPaint*, *MacDraw* and *MacProject*.

EgWord, a powerful Japanese word processing program from ErgoSoft, is available at launch.

A localized version of Microsoft *Excel*, the powerful integrated spreadsheet program, will be available in July.

Historically, the Japanese personal computer market has been driven by word processing applications. Now the emphasis is beginning to shift towards spreadsheets and data bases, according to van Eyck.

"From a timing point of view, we are announcing the right product at the right time. The market is demanding powerful, diverse software solutions. Thanks to KanjiTalk, third-party developers can localize and ship those solutions in a fraction of the normal time. We have a unique window of opportunity," van Eyck adds.

The suggested retail price for the Macintosh Plus with KanjiTalk is 648,000 yen. Apple has no immediate plans to market the KanjiTalk Macintosh outside Japan.

Macintosh in Japan

The 128K Macintosh (English version) was first introduced in Japan in April, 1984. In May, 1985, Apple Japan introduced a katakana (phonetic) keyboard for the 512K Macintosh.

EgWord, a powerful word processing package was introduced in September, 1985, by ErgoSoft.

Several other third parties have also introduced various other versions of Japanese Macintosh software.

Apple Japan is offering a variety of upgrade programs for existing Macintosh owners.

KanjiTalk Features

Since KanjiTalk supports Japanese language input and output at an OS level, selected software developed in the United States and Europe can be easily localized in Japanese. In addition, new software which follows Apple's development guidelines can also be localized for Japan.
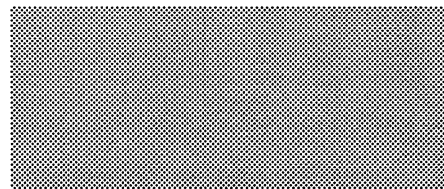
KanjiTalk application tools also facilitate the conversion of application software from English into Japanese.

Japanese software developers can easily access and utilize KanjiTalk development tools to develop unique Japanese software products.

Accurate sentence-level conversion Through phonetic input, the Macintosh can execute kanji conversions, making it possible to convert up to 40 characters at once.

KanjiTalk features three separate dictionaries with different conversion levels—a basic dictionary, a special dictionary and a users' dictionary. The basic dictionary maintains more than 35,000 words.

Through a unique design, the KanjiTalk utilities are built into the Macintosh's Desk Accessories so any function can be used any time without turning off (or rebooting) the software.

Apple to Work with National Geographic and LucasFilm

Cupertino, California. June 2, 1986.

Apple Computer, Inc. today announced it has entered into a technical collaboration with the National Geographic Society and Lucasfilm Ltd. to explore the effective use of optical technologies such as videodiscs and compact disks in education. These devices can store the resources of a small library, provide students and educators alike with access to vast amounts of information—including still and moving images—and give them individual control over how they use the material.

National Geographic Society is a respected publisher of magazines, books and other educational materials and Lucasfilm is a leading entertainment production company with advanced technical and design capabilities.

The collaborative effort is designed to explore ways to use the immense storage capabilities of optical media to bring together such teaching tools as text, film, video, audio and software and provide educators with powerful, meaningful and useful new ways to teach, deliver information and promote learning.

"We are extremely pleased to align our research efforts in the area of optical technology with such respected and innovative groups as National Geographic and Lucasfilm," said John Sculley, Apple's chief executive officer and chairman of the board. "It's our belief that personal computer technology can greatly enrich the way educators teach and the way students learn. And with that belief comes the responsibility to explore new ways to expand the usefulness of computer technology. This collaborative effort provides Apple, National Geographic Society, and Lucasfilm with a unique opportunity to learn how combined technologies and media can enhance the educational process."

According to Apple, the collaboration is part of the company's ongoing commitment to work in partnership with educational publishers, innovative developers and others to address the needs of educators and enrich the quality of education.

Apple and Kinko's Release Courseware Catalog

Cupertino, California. June 25, 1986.

Apple Computer, Inc. and Kinko's Copies, of Santa Barbara, Calif., today announced the availability of the first Kinko's Academic Courseware Exchange catalog. This resource guide offers the higher education community information and a way to purchase university-developed educational software for the Macintosh and Apple // computers at prices students can afford and in the volume that universities require.

The first edition of the bi-annual catalog currently lists 30 academic software programs with more than 100 programs expected to be listed in the fall edition. It is available at any of Kinko's 300 stores that serve more than 500 colleges and universities in the United States. The Academic Courseware Exchange will enable teachers to assign software to their classes as easily as they assign textbooks. Students can purchase software programs at the Kinko's store for between $7 to $30.

"The Academic Courseware Exchange is a highly innovative program for distributing university-developed software," said Bud Colligan, Apple's manager of higher education marketing. "For the first time, campuses nationwide can easily share the results of on-going software development work at leading colleges and universities. There are hundreds of quality programs currently in development, and future catalogs will include even more of this emerging software base. Because university developers can now easily distribute

their software while earning royalties, we anticipate that this program will accelerate the development of software on campus and will play a leading role in helping schools to more fully integrate computers into university curricula."

The first catalog includes such programs as *Animated Waves and Particles*, a program from the physics department at the University of Tennessee that simulates wave and pulse behavior and other aspects of physics; and *The Would-be Gentleman* from Stanford University, a program that simulates economic and social life in 17th century France.

"We've already had a phenomenal response to our initial direct mailing to more than 150,000 faculty members," said Keith Lawrenz, Kinko's director of the Academic Courseware Exchange. "We think the catalog will soon become an indispensable tool in university curriculum design."

Kinko's has been serving the academic community for 15 years, assisting faculty with such programs as Professor Publishing, which enables professors to inexpensively duplicate and distribute supplementary course materials. Kinko's also offers a service for publishing specialized textbooks as well as a desktop publishing service using Apple's Macintosh and LaserWriter printer.

Institutions and individuals interested in either purchasing courseware or distributing their courseware through the Academic Courseware Exchange should contact their local Kinko's or Kinko's courseware director at (800) 235-6919.
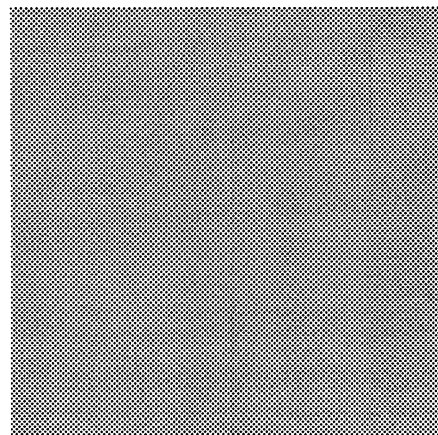
New Advanced Topics SIG

The Apple // Advanced Topics SIG held its organizational meeting at the May HAAUG meeting. This is not the first time that HAAUG has had an Advanced Topics SIG. Several years ago, however, the group gradually dissolved as many of our more advanced members gravitated to the Mac. Robin Cox, our current HAAUG president, believes there should be a forum for the exchange of information and ideas among those in the Apple // community who have gone beyond the basics. If you think that might be you, plan to join us at the next meeting.

I will be co-chairing the SIG along with Jim Record. He and I work together as professional software developers at an engineering software company. We both have Apple //e's at home, which we primarily own for the fun of having a personal computer. Not that we're above using our computers for an occasional worthwhile task, it's just that there are so many other interesting things they can do.

The kindred spirits who gathered at the first meeting discussed what direction this new SIG should take. Three principal areas of interest emerged: advanced programming techniques and languages; learning to use the full power of some of the popular Apple // application packages; and keeping abreast of the many new products available for the Apple //. Most of the presentations we will be enjoying in the Advanced Topics SIG will be given by members, so I encourage any of you that may have a topic you have looked into to volunteer to share it with the group.

We will be spending a lot of time improving our assembly language skills. We will also be learning about other languages, such as C. (Yes, Virginia, there is a decent C compiler for the Apple //. Several, in fact.) At the June meeting, Jim and I will present a discussion of accessing ProDOS functions from assembly language. Those of you who may have had past experience with programs written under DOS 3.3 will find the techniques for reading and writing files under ProDOS refreshingly simple and well documented. Also, at a future meeting Richard Lemay will present a discussion of how to write and use ampersand (&) routines from within Applesoft Basic. The ampersand technique is used to add many sophisticated capabilities to Applesoft that make it easy to write powerful, fast programs in Basic.

Most of us are also interested in keeping abreast of the new hardware available for the Apple //. One of the most exciting things about the Apple // is all those slots (eat your heart out Mac-ites). Just in the last several months, I have heard of 10 and 20 megabyte hard disk drives that install directly inside of your Apple //. There are now multifunction cards, accelerator cards, voice and music synthesis cards, home controller cards, and much more available for modest prices. And in the latest round of the RamWars, Applied Engineering is advertising memory expansion up to 16 megabytes! A few years ago I would have been impressed to hear of an IBM mainframe computer with that much memory.

Beyond the tutorials and the product reviews that make up so much of the typical SIG fare, however, I would particularly like to see the Advanced Topics SIG provide the opportunity for HAAUG programmers to demonstrate some of their pet programming projects. The Apple // is a fun computer to own because people have written such a variety of programs for it. And the power and performance that people have coaxed out of the Apple // is amazing. The basic horsepower of a standard Apple // isn't any different than it was six or eight years ago. But look how much more impressive the animated graphics of a recent game program is than that of games written several years ago. Software for the IBM-PC seems so dreary in comparison. How many new database programs can you hear about before you want to cry "Uncle"? The Apple //, perhaps because of its broader range of uses (in the home, in education, and in games, as well as in business), inspires so much more creativity. Of course there is always a place for a better compiler or word processor. But a really clever utility or an offbeat program is what interests me. Jim and I have several of our own that we would like to share with the group, and I hope that some of you do, too.

So if you're interested in finding out what makes your Apple // tick, or you want to hear about interesting new applications for the Apple //, or you would like to show off your spiffiest new program to an appreciative audience, be sure to join us next month at the Advanced Topics SIG.

—Bill Hensley

Games SIG

The Games SIG convenes at every HAAUG meeting. Our purpose is to let members see and discuss a variety of games and to help in playing and purchasing decisions. A typical meeting consists of about two game presentations by various SIG members. We look at arcade, adventure, war, and sports games. In the past few months games covered included *Kampfgruppe, Winter Games, Transylvania, Decision in the Desert, Black Cauldron, King's Quest, Silent Service, Lordlings of Yore, Captain Goodnight, Skyfox, Sundog,* and *Hardball.*

Another important function of the meeting is to let gamers pose questions to the group about specific game problems they are having. Usually someone has played the same game and can be of assistance. Although, we have had a *Bard's Tale* player whose questions have gone unanswered for two months. Don't let this man die in the dungeons! Come out with your suggestions to our next meeting. If you have a favorite game you would like to present, give the Games SIG leaders a call and we'll reserve a spot for you.

—Thomas Carson
—John Newell

Business SIG

The Business SIG was formed to be an information exchange for people using Macs in "business" applications. This is, indeed, a very broad charter but the topics, meetings, and discussions will ebb and flow according to the needs, interest, and resources we can muster. Recent meeting topics included "one write" accounting systems, business graphics, and spreadsheets. As for the future, we welcome your requests and participation.

Charter

We are here to help people who want to use their Mac as a tool to increase productivity in a work environment. Our backgrounds range from part-time, business in the spare bedroom types to *Fortune 50* middle managers, to the full-time, self-employed.

The key to achieving this goal is the product demonstrations and ensuing discussions at the monthly meetings. Sharing a few days experience of trying to make a product productive in a real live business is worth much more than volumes of "reviews" in the commercial press.

Past Meetings

The Business SIG has been organized for three months now and software reviewed and discussed so far include *Dollars & Sense*, Microsoft *Chart* and Microsoft spreadsheets *(Multiplan* and *Excel)*.

Each meeting had a similar format. There was a short, broadbrush overview followed by a fairly detailed review and demonstration of a few of the programs special features. The demonstration evolves into a fairly free-form question, answer, and comment period.

Many feel the question and answer period is the most productive of all because the discussions get down to the age old business approach of how to solve problems. Of coures the initial comments center around the current demonstration. Discussions of the product's strengths and weaknesses in absolute terms soon leads to a comparison of competitive products.

Future

As new products are brought to the marketplace, we still solicit early purchasers to share their experiences.

Also, as requests and comments on one particular product or family of products becomes frequent, that product will become a regular topic on the agenda. Perhaps, if interest is really high, a SIG for that product will be formed.

Closing

If you are trying to improve your productivity using your Mac, you qualify. So come on out. Exchange ideas, ask questions, join in the challenge of doing our jobs smarter every day.

—Gary Saxton
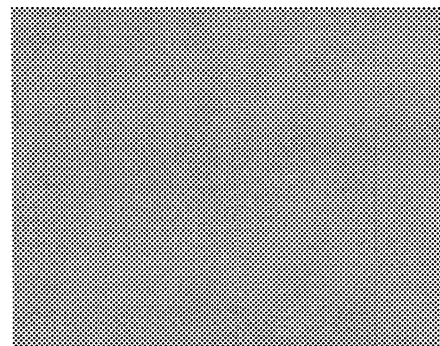
Education SIG
The SIG For All
Of Us

Education SIG? Oh! That's for teachers. Well, yes, but that's not all: teachers, administrators, concerned parents, businessmen, interested students, and last but not least, all of us who still want to learn. The one real objective of HAAUG's Education SIG is to have members conditioned to ask questions. Why? Because when you're asking questions, we know what you're interested in and are given an idea of its importance to you. More than that, we want the Education SIG to be a place where you can get answers, express ideas, and even air an occasional gripe.

The May meeting was encouraging and the attendance was much improved. Next month, I expect the attendance to double! Why, you ask? Because everyone who attended in May promised

they would bring a friend in June. Don't forget your assignment or you go directly to the principal's office, without passing the snack bar...and if you happen to be the principal you're really in for it...would you believe a 1000 word essay on your school's use of computers?

Seriously, we did have an interesting meeting in May, even if it was only to pin down a few of the topics for future presentations and to answer a few questions. In June we will have presentations by Bill McCluskey on devices to assist the handicapped use computers and by Ann Petrillo on evaluating educational software. Of course, we will still be looking for input on what the Ed. SIG can do for you!

Future meeting topics will include reviews of educational software, e.g., software used by local area schools, and what's recommended and available to the public from local stores. We will also cover what's in the HAAUG Public Domain Library for Apple II and

Macintosh. Programs for educators will also be demonstrated, i.e., gradebooks, attendance, test construction, spreadsheets, and word processors. New software employing AI (Artificial Intelligence) as a teaching aid will be presented. The possibilities for this SIG are both exciting and infinite.

Whether you are an educator looking for something to make your job easier or a way to stimulate the interest of your students, or a parent who needs to know what computer software is available to help your child or maybe even learn about something new yourself, I hope you'll allow HAAUG's Ed. SIG to help.

Be sure to check the meeting schedule for the new earlier time for the meeting of the Ed. SIG. See you there!

—Dick Lee

FullPaint Examined

FullPaint is essentially an updated version of *MacPaint*. Anyone who has worked with *MacPaint* will feel comfortable almost immediately upon double-clicking the *FullPaint* icon. All of the usual *MacPaint* characteristics are there: FatBits, Goodies, and so on. *FullPaint* will even open *MacPaint* documents. However, *FullPaint* adds most of the features *MacPaint* users have wanted. *FullPaint* is like *MacPaint*, *Click-Art Effects*, and *Paint Cutter* rolled into one.

FullPaint permits the user to work with more than one document at a time. Up to four may be open simultaneously; thus, cutting and pasting between documents is greatly simplified. *FullPaint's* tool palette is almost identical to *MacPaint's*; so is the pattern palette. Both of the palettes can be moved around the screen or hidden.

At the bottom of the tool palette, you'll find three extra icons. These allow you to choose whether you want the active window to take up the full screen, the full screen with a menu bar, or a conventional window on the desk top. In this last mode, the painting window is completely resizable and can be scrolled using scroll bars or the grabber hand. Unlike *MacPaint*, you don't have to select "Show Page" from a menu to scroll without the grabber hand (though you can do it that way if you want). In a sense, the *FullPaint* interface corresponds more closely to the Macintosh standard than does *MacPaint*.

Fullpaint will scroll when a tool reaches the edge of the window; therefore, using the marquee tool, it's possible to select items that are bigger than the screen for cutting and pasting. *MacPaint's* inability to select objects bigger than the fixed-size *MacPaint* window was one of that program's biggest shortcomings.

The program also contains special effects tools similar to those contained in *Click-Art Effects*. Using the marquee, you can select and manipulate objects. You may rotate pictures in one degree increments, skew (slant) the object, distort it, or create pseudo 3D effects. You can also print pictures at up to 400% of their usual size.

Other welcome features of the program include pop-up tracking rulers, like those available in *MacDraw*, and MouseSpot, which displays the current exact position of the cursor hot-spot. Another feature, MouseCrawl lets you adjust the mouse tracking, so the cursor, well, crawls. Mousecrawl supposedly allows the artist much more precise control of whatever tool is in use; however, when I tried it, I found the cursor more difficult to position.

The *FullPaint* disk also contains *LaserPrint*, a neat utility that splits *FullPaint* documents so you can print high-resolution (150 or 300 dots per inch) documents on the LaserWriter. You can also use *LaserPrint* and *FullPaint* to create poster-sized drawings.

In general, I've been pleased with *FullPaint*. The extensions to *MacPaint* make the program much more powerful than the original. I have noticed a few problems, however. The lasso tool doesn't always tighten around the object it surrounds; sometimes the lasso also selects a portion of the white background. The scroll bars also work rather lethargically. *FullPaint* is a much larger program than its predecessor. *FullPaint* consumes 175K of disk space while *MacPaint* only requires 60K. Unfortunately, the program employs a tough-nut copy-protection scheme (which fortunately *Copy II Mac 5.3* can crack; however, *Copy II Mac 4.5* won't work). I hate to see an otherwise fine program marred by pointless, annoying copy-protection.

The manual claims that *FullPaint* is *Switcher* compatible, but you'll have to reconfigure the size resource using the *Switcher* "Configure then Install..." option (see "Fixing *FullPaint*"). The program requires a 384K partition; this shouldn't be too much of a problem for users with a megabyte or more of RAM, but the rest of us will probably have to forget about using *FullPaint* with *Switcher*.

FullPaint comes with a size resource, but for some reason the size resource was only set for 376K. Therefore, the program won't work right. Every time I switched into *FullPaint*, the program asked for the key disk. Curiously, though the program asked for the key disk, it failed to eject the copy. I ejected the copy manually, then inserted the original. After the computer spit out the original, and I reinserted the copy, I got a dialog box saying "*FullPaint* will only run on a 512K or larger Macintosh." If the size resource is adjusted to 384K, the program works fine. This is a relatively easy procedure, but it shouldn't be the user's responsibility.

The *FullPaint* manual pays tribute to Bill Atkinson, the creator of *MacPaint*, for "pointing the way." No wonder. *FullPaint* is so much like *MacPaint* that Ann Arbor Softworks must have at least gotten permission from Apple to use elements of *MacPaint's* user interface, such as the tool icons. I wouldn't be surprised if underneath *FullPaint* lurks some of Bill Atkinson's very own source code or if Apple gets a portion of the profits from its sale. *FullPaint* doesn't exactly make *MacPaint* obsolete, but now that *MacPaint* is no longer bundled with Macintoshes (*MacPaint* now costs $125), it doesn't make much sense to pay 25 or 30 extra dollars ($60 if you mail order) for a less capable product.

—*Cleland Early*

Fixing *FullPaint*

The *FullPaint* manual says that the program requires a 384K partition to run under *Switcher*, but *Switcher* automatically assigns a too-small 376K when you load *FullPaint* into *Switcher*. Fortunately, Andy Hertzfeld, the author of *Switcher*, has provided a means of correcting this shortcoming. Before installing *FullPaint* in *Switcher*, you'll need to reconfigure *FullPaint*. Double-click on *Switcher* to start the program. When the wristwatch disappears, pull down the *Switcher* menu and select "Configure then Install..." Figure 1 shows the dialog box you should get. To reconfigure *FullPaint*, change Minimum Memory Size from 376 to 384 and click on the "Permanent" button. That's all there is to it.

Note that you should not change the Preferred Memory Size even though *Switcher* displays the same number in that box and the Minimum Memory Size box. *FullPaint* actually prefers 512K. I don't know why Switcher displays the wrong number (the same thing happens to other programs where 512K is preferred), but I'd guess that you'd need more than
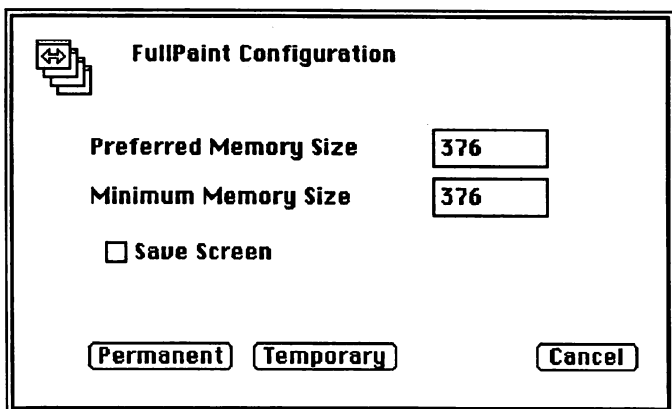
Figure 1. The Configuration Dialog Box

512K of RAM for 512 to appear in the Preferred Memory box. This same procedure can be used to add SIZE resources to programs that don't have them. Most new programs already have SIZE resources. Older programs may require more than the default 128K, but you have to configure them manually the first time. For example, *MacPaint* prefers 179K, but will run in 128K. Once you've created a SIZE resource using "Configure then Install...," *Switcher* will assign the right amount of memory automatically. *Switcher* actually tricks the program into thinking it's running in a full 512K.

Switcher is a reasonably smart program. When you install a program in *Switcher*, the program examines the SIZE resource to determine how much memory a program prefers and how much it needs. Then *Switcher* finds how much memory is available. If available memory is larger than the

preferred memory allocation, *Switcher* will assign the preferred memory size. If available memory is smaller than the preferred size, but bigger than the minimum memory size,
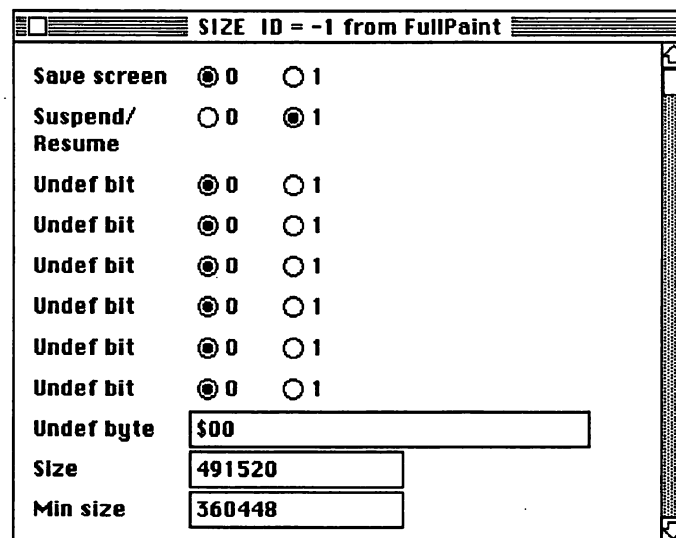

Figure 2. SIZE Resource from *FullPaint*

then *Switcher* assigns the minimum.

Figure 2 shows what a size resource looks like if you care to examine one using *ResEdit 1.04D*. When I first looked at the "Min size" number here, I was a little confused, but then I remembered that 1K actually equals 1,024 not 1,000. Dividing 352,256 by 1,024, you get 344K, which is considerably less than both 376K or 384K. I was confused again. However, I looked at the SIZE resources of some other programs and found that this number was consistently 32K smaller than it should be; don't ask me why, and Hertzfeld says this is normal but doesn't really explain why, in *Inside Switcher*. For example, *ThinkTank 512* requires 256K to run under *Switcher*, but its "Min size" was given as 229,376, which works out to 224K. If you want the "Min size" to equal 384K instead of 376K, you must actually subtract 32K from 384K. Then you multiply the result (352K) times 1024 and get 360,448. Thus, the "Min size" for *FullPaint* should read 360448, not 352256.

Adventurous souls can use *ResEdit* to create or change SIZE resources, but it's certainly much easier to do the same thing from within *Switcher*.

—*Cleland Early*

Macintosh Programming: A Step-by-Step Approach

Step 1—The Beginning

As everyone is aware, the Macintosh is a unique type of computer having capabilities beyond what were imagined only a few years ago. Because of this, the Macintosh requires different programming techniques compared to conventional computers. "User Friendly" programs require effort and thought from the programmmer in order for the programs to work correctly. The purpose of this series is to teach you the "how to" behind programming the Mac in easy-to-digest portions. For expert programmers this may move a bit slow, but it stresses fundamentals that are necessary for all types of programming. Hopefully, by the end of this series, you should be able to generate a full-blown, double-clickable application utilizing all the Macintosh's features in the way intended by Apple.

Of course, for any course like this it is important for you to have the necessary equipment and reference materials before starting. Besides a Macintosh 512, Macintosh Plus, or Macintosh XL, you will need some sort of programming language. *Pascal, Forth, C,* or the *68000 Development System* will allow you to program from scratch. MS-*Basic* (Version 2.0 or greater) and *MacPascal* will allow you to try most of these features but you will not be able to make stand alone applications. The key features to look for in the documentation are if all Toolbox calls are available and if the langauage can create stand-alone applications. The more Toolbox calls available, the more control you can have over your program content. I'll be using the *Macintosh Development System* on the MacXL (*LisaPascal*). The concepts and Toolbox calls should be similar regardless of the language used.

In addition, a number of program editors put out by Apple and other programmers are very helpful and we'll use them in future articles to help understand what is going on within the Mac and your program. These programs are either public domain or shareware and most of them are available in the HAAUG public domain library. Here's the list; *Fedit, ResEdit, Redit, Uriah Heap* (a desk accessory) & *Dialog Creator*. In addition, you should have the *MacPaint* application and plenty of blank disks. **It is important for you to have your example program on a diskette that has nothing irreplaceable on it; some programming errors can have a disasterous effect on disk data.** Each programming diskette should have its own system folder on it, so it is the only diskette in the machine when you test your program.

In terms of books, *Inside Macintosh* is a must. It can be purchased as a hard cover (Volumes I-III) or each volume separately in soft cover. Alternatively, you might be able to find an earlier copy from a developer that is loose bound and is also acceptable to use. The other books to obtain are the two volumes of *Macintosh Revealed* by Stephen Chernicoff (Hayden Books). Between *Inside Macintosh* and *Macintosh Revealed*, 99% of your programming questions can be answered; the other 1% comes down to trial and error or what other programmers can tell you.

Now we can now begin to program. However, this does not mean that we can immediately sit in front of the computer and begin typing code. First, we have to determine what we are trying to accomplish with the program. To do this, write down on a piece of paper what you would like to do with this program. For example, you could write "I would like to write a program that would chart stock prices on the screen and do some analysis of the data." Next, under this statement, you should write down all the key features

that your program will have (i.e. "I should be able to have multiple windows with each window having a different company displayed"). Take a little time to think about what you want or don't want in your program. This will allow you to be more efficient in developing your program and makes sure that it has all the features you want.

After you are satisfied with your list you can then begin to map out your program. Start out by using a program like *MacWrite* or any application similar to what you wish to develop. When in the application, analyze all the features that you will have to include within your program (i.e. menu bars, windows, sound, dialog boxes, etc.). Next, press the Command, Shift, and 3 keys simultaneously at various points of your evaluation to make a few *MacPaint* documents of what you see. This will form the basis of your program. When you leave the program, you will see the *MacPaint* documents "Screen 0", "Screen 1", etc. on your diskette.

Then, by using *MacPaint*, you can modify these screens showing how your program would look when it is completed. This "storyboard" approach will allow you to create the exact image of your program without typing a single line of program code. Every window, menu bar item, dialog box and special feature should be drawn and documented in this way. You should print the screens out and keep them together as templates to help you create your program. You can also use them to check your program as its being written and make sure that it looks exactly how you visualized it. Later on you may find that it needs to be modified, but this technique provides a starting point for your programming.

Assignment

Before the next article, you should

determine what program you wish to develop, the features you wish it to have, and develop the "storyboard" of your program with as many features shown as possible. In addition, read Chapter 2 of *Inside Macintosh*, Volume 1, called "Macintosh User Interface Guidelines." This will help you in determining how your program should look. The "Next Step" in the next issue will be to give the general outline of a Macintosh program and talk about the concept of an event driven computer. Please refer any questions you may have to the Editors of the *Apple Barrel* or leave them at the Answer Desk with my name on it and I'll do the best I can to answer them.

—Mike Conway

Clear Lake Research Libraries Beef Up *BASIC*

This is a review of a set of assembly routines from Clear Lake Research that add power to Microsoft *BASIC*. If you prefer to program in *BASIC*, but don't because you require access to resource files or MacinTalk, then don't even bother to read this but go ahead and buy the complete set of CLR Libraries. If you don't live with *Inside Macintosh* then read on.

I must first apoligize, however, as there is no way I could list all the options provided by the CLR Libraries.

No piece of software is perfect but those that sell well are often the target of "support programs". These are programs that try to improve upon the original by adding neglected features or by offering alternative approaches to a problem. These features may be ones left out of the original program by mistake (poor research) or there may have been a decision to simplify things for the user by presenting only a few options. When Microsoft released their *BASIC* (version 1.0) for the Macintosh they were guilty of rushing a product to market. Unfortunately, it was the only language available at the time so a lot of good work was done with this language by necessity. The programmers at Clear Lake Research cut their teeth during this period by using "peeks" and "pokes" to gain access to areas in the Macintosh restricted by Microsoft *BASIC*. When Microsoft upgraded their *BASIC* to version 2.0 they addressed some of the shortcomings of their earlier versions as well as providing a way that assembly language routines could be called directly from the language. This addition paved the way for Clear Lake Research to bring to us these areas that they had tapped. CLR has produced three "libraries" designed to expand the options available from *BASIC* for your programs.

The CLR *ToolLib* provides access to almost all of the Quickdraw Toolbox routines that Microsoft omitted. These are the ROM calls for drawing pretty pictures on the screen. There are too many routines to list here, too many to even count, but some of the areas covered include point manipulation (LocalToGlobal!), rectangles (Empty-Rect!), polygons, regions (DiffRgn!), mapping, clipping (extremely helpful), text (TextWidth!), and resources. Resources are the files that most commercial programs have that contain all the information on cursors, icons, patterns, pictures, text strings, and dialog boxes! There is even a routine to allow the creation and use of scroll bars (with text only).

MathStatLib reflects the professional origin of Clear Lake Research. It is a set of routines that are very useful in mathematical data analysis (there are also a few string operations). There are sorting routines, matrix operations (inversion!), and simple statistics (standard deviation, sums, and t-tests to name a few). As a graduate student in Biochemisty I found these routines particularly useful at work. All of these math routines can be written directly in *BASIC*, but the CLR routines are faster because they are coded in assembly. Unfortunately, the majority of the tools in *MathStatLib* can only be used with the binary version of *BASIC*. If you don't care which version you use then obviously this is no problem; however, it can be a large irritation if you have an existing program in the decimal version that you have to first convert to the binary version. In CLR's defense, the problem arises from the difference in the internal representation of floating-point numbers in the two interpreters. Furthermore, *ToolLib* and *SpeechLib* don't have this restriction.

Some of this review may sound familiar. About a year ago CLR gave a presentation at our club meeting. Something new since then is the CLR *SpeechLib* which gives the power of speech to your *BASIC* programs through access to Apple's MacinTalk routines. These are the same routines used in the recently released game of *Orbiter* from Spectrum Holobyte, Inc.! You can control the pitch, rate, and volume of the voice. The audio quality of the spoken words can be improved with phonemic spellings. This particular library is a lot of fun and easy to use.

All this power comes with a price (besides the retail price which is reasonable). Integrating these tools with your program is extremely easy, just include the statement "Library 'name of the library' " in your program (as long as the library is actually on your disk!). The manuals are well thought out and clearly written. There is even a troubleshooting guide to help track down errors. I have had no trouble implementing any of their routines. So what could be difficult? Microsoft left out many of the QuickDraw ROM calls not just because Microsoft was lazy but also because those ROM calls applied to concepts unique to the Macintosh. Data structures such as resource files and regions are ideas that even advanced programmers sometimes have difficulty with. Futhermore, using these ROM calls often requires manipulation of pointers and handles as well as some knowledge of the way the Mac's memory is divided and used. These are not typical requirements for a *BASIC* programmer. Those of us who have owned our Macs since "the beginning" are familiar with this discussion, as it contains the reasons most developers gave for postponed software releases. Of course, there are several good books on the market nowadays to help the beginner.

Any investment of your time will be well worth the effort. With the Clear Lake Libraries you will be able take your programs where no *BASIC* programmers have gone before and any knowledge you gain will be worth its

weight in gold if you decide to use a compiled language such as *Pascal* because the concepts are the same. There is a lot to be said for programming in *BASIC*. It is quick and easy. With the additional flexibility provided by these routines it can be even more rewarding. Finally, for those of you who need a little more reassurance that Microsoft *BASIC* and these libraries are a serious combination, *MacTutor* (the Macintosh magazine for programmers) has started to run articles on programming tips with the Clear Lake Research routines. For the impatient, using *SpeechLib* isn't difficult. Most *BASIC* programmers will have their Macs talking (swearing?) in under 5 minutes.

So where can you get the best thing to happen to *BASIC*? Unfortunately, they aren't out in the stores. You can either send away for them or order them by phone. The three libraries are $35.00 each, two for $50.00, or all three for just $85.00, plus $3.00 for shipping and handling. Of course Visa or MasterCard is acceptable. To order call 1-800-835-2246, x199 or write to Clear Lake Research, 5353 Dora Street, Suite 7, Houston, Tx., 77005. Microsoft *BASIC* is required.

—Rick Light

Flash!!!

Disk Caching Not Supported by Older Programs

MOST programs do not use the RAM cache feature supported by the new 128K ROM's. This is due to the fact that there is a special indicator on files that must be set for caching to occur. Since most programs came out before the indicator existed, it is not properly set on most files.

The indicator on a file may be set by using the "Get Info" command in ResEdit (versions 1.0 or later) and clicking on the cache bit option. When set correctly it should look like the screen at right.

Info for file FullPaint

File: FullPaint

Type: APPL   Creator: PNT

☐ Locked   ☐ Invisible   ☑ Bundle   ☐ System
☐ On Desk   ☐ Bozo   ☐ Busy   ☐ Changed
☒ **Cached**   ☐ Shared   ☑ Inited
☐ Always switch   ☐ Never switch

☐ File Busy ☐ File Lock   ☐ File Protect

Created   4/19/86 3:27:38 PM

Modified 6/11/86 11:02:43 AM

Resource fork size = 174857 bytes
Data fork size = 0 bytes

In the Public Domain: The HAAUG Trough!

One of the most active sections of any computer user group is its library of public domain software. Find the crowd and chances are it's either the vendor's tables or the copy desk. The Mac section of HAAUG is no exception and has a public domain library of close to 105 disks. This amounts to over 3,000 separate files. Like all public domain collections, there is duplication of some files and a wide variety of value and quality. Admittedly, some files are downright worthless. However, like so many other things, one man's trash is another man's treasure. For this reason we have attempted to make it all available to our members. We have had to be careful to screen copyrighted software from some of our donations. Nothing can hurt a user group's reputation like allowing this material to stay in its library. But that's another issue. As many of you have heard, we are attempting to reorganize our collection into categories, i.e., games, fonts, desk accessories, etc. With luck, this collection should be available later this summer.

Having worked at the "copy desk" for the HAAUG Mac Public Domain Library for over a year now, I have been asked many questions about our collection of public domain software. I must honestly say that sometimes my answers haven't been too helpful. With the great number of disks it's hard to know what's what. On top of this, I still claim a Mac-Novice Status. In spite of my status, maybe this article will help some HAAUG members to get an idea of our library's contents and allow me to rectify some of the misconceptions about the Mac Public Domain Library.

First and most important misconception—public domain software is all "freeware", right? WRONG! In reality, very little, if any of the really good stuff is free. Tell me something I didn't already know, Dick?

Second important misconception—"I paid HAAUG $5 for this disk of public domain software, so I don't owe anyone else!" NOT SO! Some of the best software in the public domain is distributed as "shareware". This means the authors are doing us a favor and, because of lower distribution costs, can provide us quality software at reasonable prices. HAAUG requests you to pay the authors if you use their product.

Third important misconception—"OK guys, I'll buy disk HMS-37 from HAAUG and you can all copy it from me, right?" WRONG. Gee, don't we have a great meeting place and the big monitors and projectors are nice, right? The *Apple Barrel* doesn't cost much to print, right? Your annual dues pay for all these things, right? WRONG. The public domain library is an income source for the club and helps pay for all the things we enjoy, so please support your club by buying your own copy of the club's public domain software.

Fourth misconception—"Dick, tell me what public domain disks (programs) I can use in my business?" My standard answer is NONE! You may not agree, but to me my business is my living and I won't take unnecessary risks by using software that lacks real documentation, supplier support and warranties. OK, I know some software we can get from retail outlets have the same problem. For this reason I rely on reputable dealers and good reviews to select business software.

So I don't use public domain software for my business, but that doesn't mean I don't use it at all. On the contrary, I have spent hours enjoying desk accessories, fonts, music, speech programs, and playing games that reside in the public domain. Yet when I'm asked by new members what public domain software they should buy, it's difficult to answer. One problem is that a disk may contain only one or two programs that are interesting or useful. I think the best approach to choosing public domain software is to decide what interests you most and ask for recommendations on the best in its catagory, e.g., fonts, games, utilities, etc. I also recommend checking out computer magazines and seeing what they say is "hot". At least that's what I did until I surveyed the last six months of the five Mac magazines I take and found only one had really addressed public domain software in any depth at all. RECOMMENDED: "Open To The Public" by Neil L. Shapiro which appeared in *MacUser*, February, 1986, pp. 110-115. With thousands of public domain programs, I find it hard to believe more public domain software reviews are not published.

So to rectify the aforementioned oversight, and with apologies to the *Apple Barrel* editors and to authors of excellent public domain software that I have omitted or overlooked, here are my TOP 25:

GAMES

Missle—HMS #58
Destroy the incoming missles before they get you...they will, you know.

Dungeon of Doom—HMS #101
Other reviewers rate this as a good adventure game but, being accident prone, I may never know.

Canfield—HMS #98
A solitaire game that's hard to cheat (or was that beat)...maybe both.

Core War—HMS #73
Recommended for programmers.

Bomber—HMS #74
Try it you'll like it!

Megaroids—HMS #42

Written in MegaMax C...fast and good.

COMMUNICATIONS

Red Ryder—HMS #81

A really great shareware terminal program.

Packit—HMS #58

Compresses and decompresses files for telecommunications. In common use by most BBS's.

Binhex—HMS #64

Codes & decodes programs for telecommunication upload/download.

FONTS

Boston Two—HMS #75

Good practical font that has excellent clarity.

UTILITIES

Switcher—HMS #99

A must-have for 512K & Mac Plus owners.

ResEdit—HMS #51

Apple's Resource Editor (copyright but uploaded with permission).

Rescue —HMS #72

Recovers most "unopenable" *MacWrite* documents.

FEDIT—HMS #63

The most powerful disk-editor and file-recovery program in public domain.

DESK ACCESORIES

MockPackage—HMS #69

Consists of *MockTerminal*, *MockWrite*, & *MockPrinter*.

MockChart—HMS #74

A desk accessory that creates line, bar, stacked, and hi-lo charts of your data.

PaintMover—HMS #73

Another great gift from Bill Atkinson, for use with *MacPaint*.

CRAY 2A & 2D—HMS #New

A super desk calculator.

APPLICATIONS

QuickFile—HMS #73

Database-Create your own fast rolodex file, thanks to Bill Atkinson.

Banner—HMS #14

Create banners with large letters on the ImageWriter. Printing takes longer than our waitress at the pizza place.

MassCopier—HMS #New

Disk copy program for 512k Mac...fast 30-second copies.

MultiReg—HMS #New

Powerful statistics program (be sure and get the documentation files, too).

MUSIC & SPEECH

MacinTalk—HMS #52

A speech synthesizer...limited but fun.

LANGUAGES

Modula 2—HMS #46

If you're into programming, you might want to check this out.

"New"—In the Mac Public Domain Library but not assigned HMS# at the time this article was written. See disks greater than HMS #100.
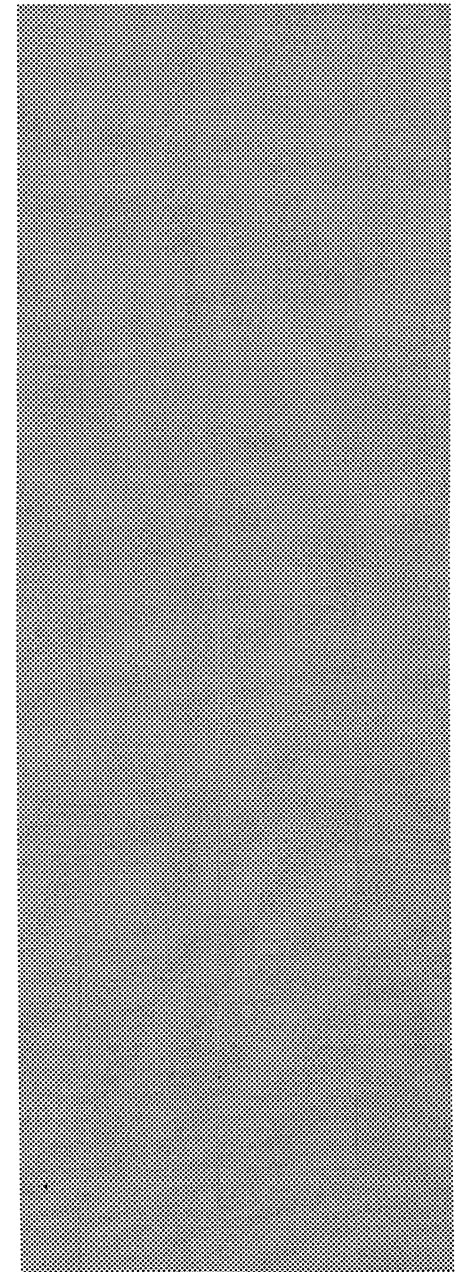
Don't forget, our public domain library depends on member contributions of material downloaded from bulletin-boards and other user groups. If you run across a really good public domain program that doesn't appear on our catalog disk, let us copy it at the regular meeting. Normally HAAUG obtains three to five new disks per meeting. However, last month for the first time since the Mac Group was formed, no new disks were received. Hopefully this will not become a trend. Have you contributed anything lately?

This article should also point out the real need for reviews of public domain software. If you have a favorite public domain program, I encourage you to write a review of about a quarter of a page long. These reviews would be very helpful to our other members and great material for an Apple Barrel column..."The HAAUG Trough"? How about it? Sit down now and review your favorite program. These reviews may be given to Chris Flick, Tom Engle, or me.

I hope this article will be of assistance in helping new HAAUG members find some interesting public domain software.

—Dick Lee

Macintosh Software Updates

Program	Latest Version	Notes
Aztec C	1.06g	C+
BatteryPak	1.1	C
Boston Font II	2.0	S+
Calendar Maker	1.2	S
ChipWits	1.1	C
Choose Printer	1.5	A
Chooser	2.2	A+
ColorChart	1.3	C
ColorPrint	2.01	C
Communique	2.0	?
ConCode	1.3	S
ConcertWare	3.0	C+
Consulair C	4.5	C+
Control Panel	2.0	A+
Copy II Mac	5.3	C+
Crunch	1.0	C
Day Keeper	1.55	C+
Deluxe Music Construction Set	-	C+
DiskInfo	1.43	S+
Dollars & Sense	1.3	C
Edit	2.0d1	A+
Excel	1.01	C+
ExperLISP	1.04	C
ExperLogo	1.1	C
Factfinder	1.1	C
Fedit	4.0	A
Finder (MFS)	4.1	A
Finder (HFS)	5.3	A+
Flight Simulator	1.0	C+
FONTastic	2.6	C+
Font/DA Mover	3.2	A+
Font Display	3.1	S
FreeTerm	1.8	P
Front Desk	1.0	C
Gato	1.3	C
Hard Disk Util	1.21	C
Hayden Speller	1.2C	C
Helix	2.0r5	C
Home Accountant	1.03	C
ImageWriter	2.3	A+
ImageWriter AppleTalkDriver	2.3	A+
Installer	2.1	A+
Interlace	1.0	C
Jazz	1.0	C+
Just Text	1.0i	C+

Program	Latest Version	Notes
LaserPrep	3.1	A+
LaserWriter	3.1	A+
LAZlife	2.0c	S
MacDraw	1.9	AC+
MacDraft	-	C
Mach	11.1	C
Macintosh Pascal	2.0	AC+
MacLabeler	2.0	C
MacPaint	1.5	AC+
MacProject	1.0	AC
MacSpell +	1.07	C
MacTerminal	2.0	AC+
MacWrite	4.5	AC+
MacZap	4.1	C
MCS	1.0A	S
MCS Editor	1.0	P
MCS Edit	1.0	P
Microphone	1.0	C+
MockChart	1.0	S
MockPackage	4.2a	S+
Mouse Exchange	1.4	C+
MS Basic	2.1	C
MS Chart	1.00	C
MS File	1.01	C
MS Fortran	2.1	C
MS Word	1.05	C+
MS Multiplan	1.1	C+
MusicWorks	-	C+
Neon	1.5	C
Omnis 3	3B.10	C
Other...	1.6	S
OverVUE	2.0C	C
Princeton Font	2.0hm	S+
PackIt II	1.0	S+
Paint Cutter	-	C
PageMaker	1.2	C+
QUED	1.3	C+
Quick& Dirty #1	1.5	C
Quick& Dirty #2	1.0	C+
RamStart	1.21	P
ReadySetGo	2.1	C+
Red Ryder	8.0	S+
Redit	1.2	A+
ResEdit	1.0d7	A+
Smartcom II	2.2A	C+
Statworks	1.2	C
Switcher	4.9	A+

Program	Latest Version	Notes
System	3.2	A+
TermWorks	1.2	S
ThinkTank 512	1.2	C
Throw Paint	1.0	P
TimeBase	1.6	C
TurboCharger	1.11	C
12-C FinancialCalc	1.4	C+
Vco	1.3	S+
VersaTerm	2,1	C
VideoWorks	-	C+
VMCO	1.5C	F+
WayStation	2.2	S+
YAPU	3.2	S

Special thanks to Dr. William Linsley at the University of Houston for submitting this download from the INFO.MAC BBS.

Legend

A=Apple Product
C=Commercial Product
S=Shareware
P=Public Domain
+= MacPlus Compatible

Encode the Hidden Message
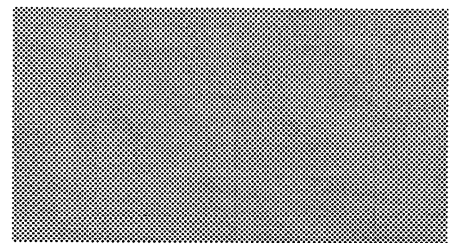
HEY MAC

Now you can print your Macintosh documents on Kinko's own Apple Macintosh Plus computer and LaserWriter Plus printer.

This page shows how amazing 90,000 dots per square inch can look. ImageWriters can't even compete.

Come in today to test drive the LaserWriter.

kinko's®

Great Copies • Great People
11152 WESTHEIMER (AT WILCREST)
(713) 977-2666
OPEN 24 HOURS A DAY, EVERYDAY

1986 Meeting Dates

———— • ————

July 26, 1986
August 16, 1986 [†]
September 20, 1986
October 18, 1986
November 15, 1986
December 20, 1986

[†](at the AstroVillage Inn)