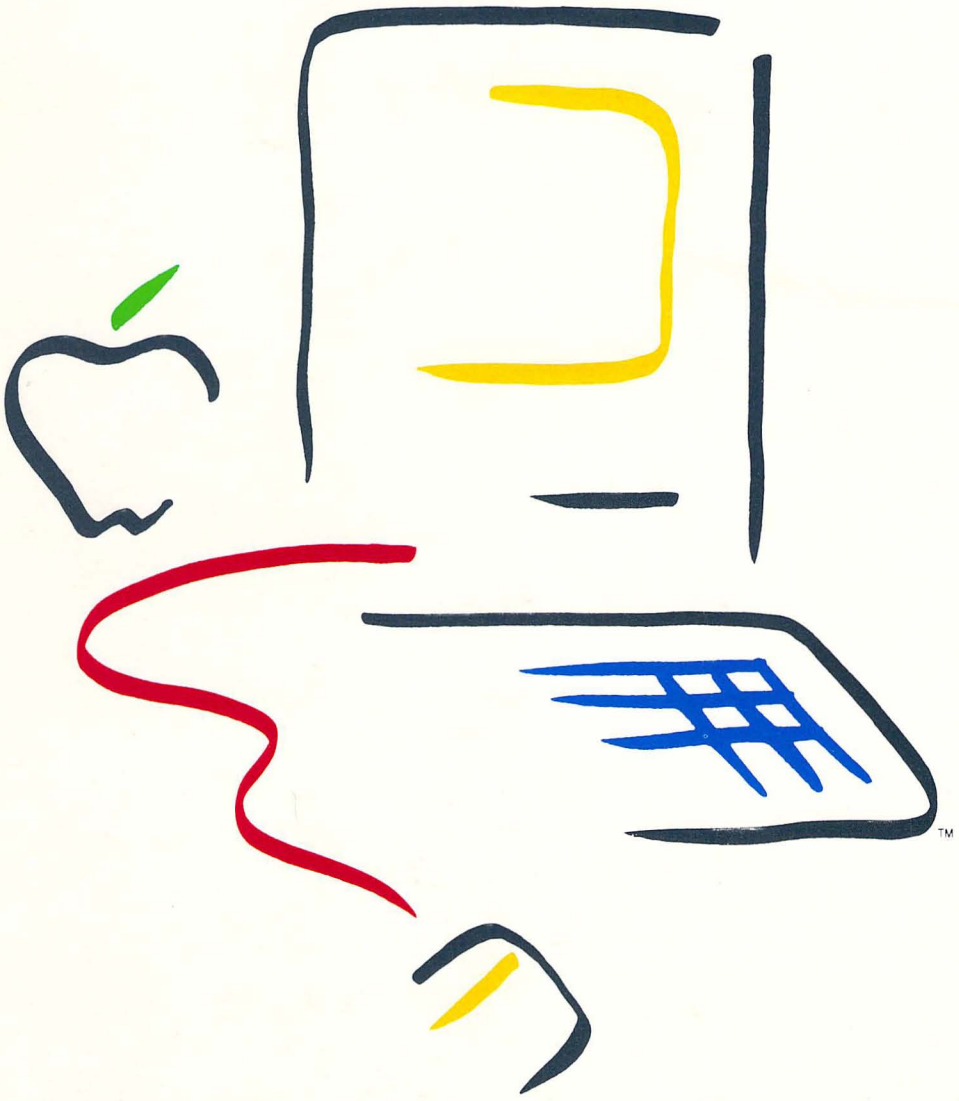
Washington Apple Pi

Volume 6 February 1984 Number 2

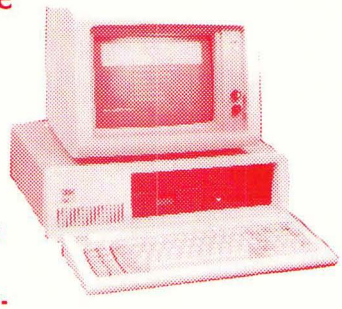
Highlights

Macintosh



We have them all.

HLA Computers is about the only place in the Washington area where you can pick and choose from a wide variety of personal computers. We have them all: the versatile and powerful Apple® for the office, home or classroom . . . the affordable and easily expandable IBM for the professional . . . the Osborne™ you can take with you, wherever . . . the functionally designed Corvus Concept with its dual-orientation display and a full page word processor . . . Digital's high performance Rainbow and flexible Professional 350 . . . the incredible, powerful Sage for the computer programmer . . . the multi-user Altos . . . Xerox . . . and the Corvus Local Area Net (LAN) 64 workstation system.



And then some.

Off-the-shelf hardware and software for all of these computers and our professional expertise to custom design software especially for your needs. Stop in and take your pick.

Apple & IBM
National Account Dealer

GSA Schedule
345-8170

HLA COMPUTERS

5700-J Sunnyside Avenue
Beltsville, Maryland 20705

(301) 345-1123

What makes Macintosh tick.

THE MACINTOSH: A First Look
MACSOFTWARE
THE MACINTOSH AS VIEWED BY AN ENGINEER
WHITHER APPLE?

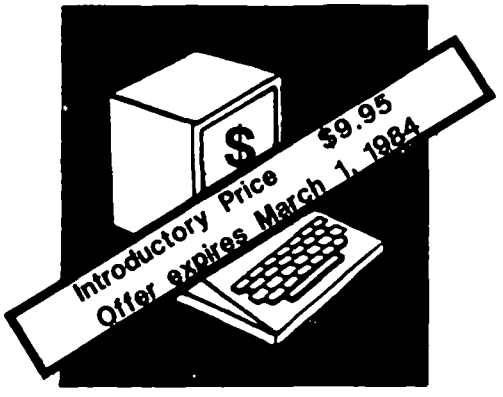
In This Issue

Hy-Quality Products from HYTEK LTD.

•New from the Publishing Division...

SELLING YOUR SOFTWARE

The Personal Computerist's Guide
to Fame and Fortune
.....maybe!



HYTEK LTD.

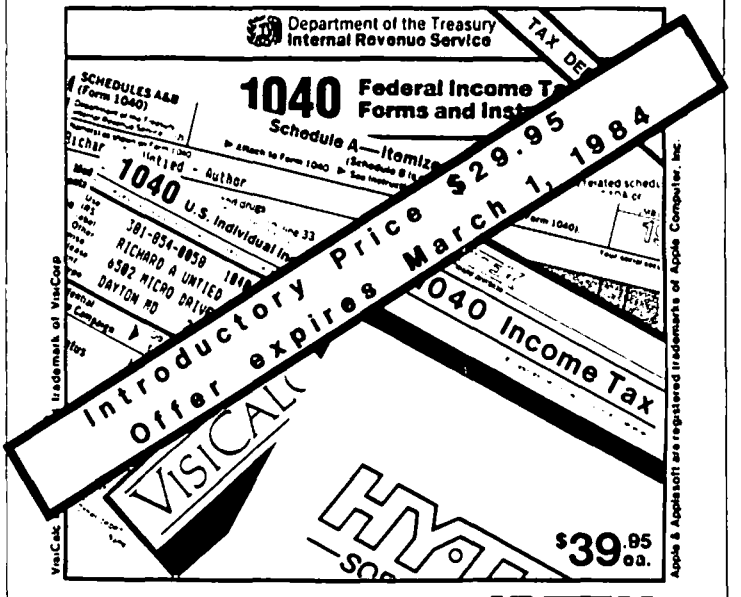
$19.95

•New from the Software Division...
•(by W.A.P. member Richard A. Untied)

1040 Income Tax

****

VisiCalc Template

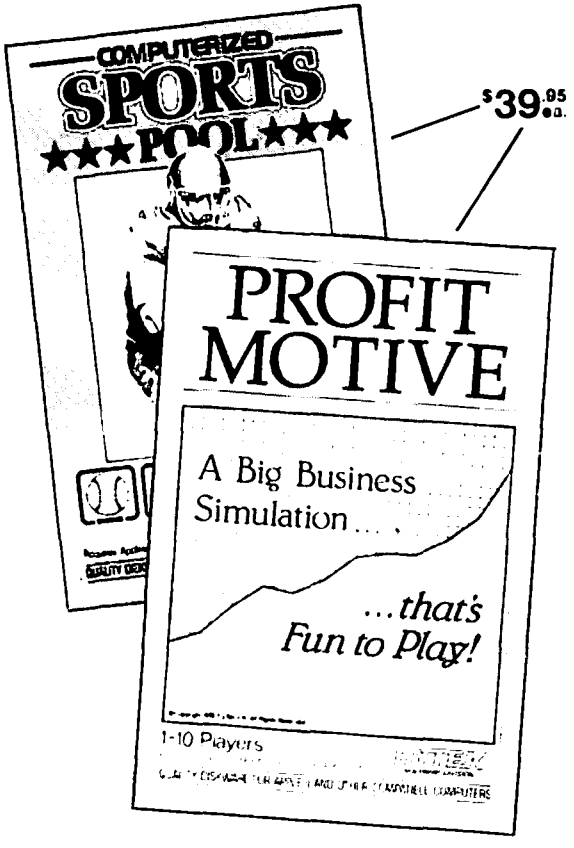


Requires: VisiCalc

HY·TEK LTD. — SOFTWARE DIVISION

QUALITY DISKWARE FOR APPLE II AND OTHER COMPATIBLE COMPUTERS

•Other software products...



$39.95 ea.

Contact your local dealer!

Dealer inquiries invited!

HY·TEK LTD.

Corporate Offices
6502 Micro Drive · Dayton, MD 21036
(301) 854-0058

OFFICERS & STAFF

President	- David Morganstein	(301) 972-4263
Vice President	- Dana Schwartz	(301) 725-6281
Treasurer	- Edward Myerson	(703) 759-5479
Secretary	- Sarah Lavilla	(301) 926-6355
Directors	- Bernie Benson	(202) 546-0076
	- Peter Combes	(301) 871-1455
	- Bruce Field	(301) 340-7038
	- Nancy Philipp	(301) 924-2354
	- Gordon Stubbs	(703) 750-0224
	- Rich Wasserstrom	(703) 893-9147
Editor	- Bernie Urban	(301) 229-3458
Associate Editors	- Betsy Harriman	(202) 363-5963
	- Genevie Urban	(301) 229-3458
Journal Staff:		
Store Distrbtn.	- Ray Hobbs	(301) 490-7484
Columnists:		
Applesoft	- James T DeMay Jr.	(301) 779-4632
DisabledSIG	- Jay Thal	(202) 244-3649
EDSIG	- Peter Combes	(301) 871-1455
LOGOSIG	- Nancy Strange	(703) 691-1619
Q & A	- Bruce Field	(301) 340-7038
Telecomm	- Dave Harvey	(703) 527-2704
VisiCalc	- Walt Francis	(202) 966-5742
Group Purchases	- Rich Wasserstrom	*(301) 654-8060
	*(Call Kevin at office on M,W,Th,F)	
Head Disk Lib.	- John Malcolm	(301) 384-1070
Library Staff:	- Dave Weikert, Bob Hicks, Ed Lang,	
	- Jim & Nancy Little, Terry Prudden,	
	- Bruce Redding, Gordon Stubbs	
New Disks	- Bob Platt	(202) 223-1588
Pascal Lib.	- Mike Hartman	(301) 445-1583
CP/M Lib.	- Kit Leithiser	(703) 569-8037
Head Reading Lib.	- Walt Francis	(202) 966-5742
Apple Tea Coord.	- Paula Benson	(202) 546-0076
Arrangements	- Jim Taylor	(301) 926-7869
Demonstrations	- Signe Larson	(703) 524-4541
General Counsel	- Jim Burger (Shaw, Pittman, Potts & Trowbridge) day	(202) 822-1093

Membership	- Dana Schwartz	(301) 654-8060
Program	- Cara Cira	(301) 468-6118
Publicity Chairman	- Hunter Alexander	(703) 820-8304
Public Relations	- Lee Raesly	(301) 460-0754
Rules & Elections	- Bob Platt	(202) 223-1588
School Coordinators:		
Virginia	- Barbara Larson	(703) 451-9373
	- Nancy Strange	(703) 691-1619
Mont. Co. MD	- Margie Stearns	(301) 657-2353
Pr. Geo. Co. MD	- Conrad Fleck	(301) 699-8200
Special Publctns.	- Betsy Harriman	(202) 363-5963
SYSOP	- Tom Warrick	(301) 656-4389
Tutorials	- Steve Stern	(301) 881-2543
	- Leon Raesly	(301) 460-0754
Volunteer Coord.	- Boris Levine	(301) 229-5730
SIG Coordinator	- Bob Platt	(202) 223-1588
SIG Chairmen:		
Appleseeds	- David Stern	(301) 881-2543
Apple ///	- Bill Hershey	(301) 588-1992
ASMSIG	- Ray Hobbs	(301) 490-7484
CESIG	- John Kapke/	(301) 694-5968
	- Roy Rosfeld	(301) 340-7962
CP/M	- Charles Franklin	(301) 997-9138
DISABLEDSIG	- Curt Robbins	ABBS WAP428 or
write 8805 Barnsley Court, Laurel, MD 20708		
(or phone)	- John Molineaux	(301) 341-7391
EDSIG	- Peter Combes	(301) 871-1455
FORTH SIG	- Kevin Nealon	(703) 280-1136
Home Control SIG	- Russ Eberhart	(301) 596-4995
LAWSIG	- Charles Field	(202) 265-4040
LISASIG	- Gordon Stubbs	(703) 750-0224
LOGOSIG	- Nancy Strange	(703) 691-1619
NEWSIG	- Bernie Benson	(202) 546-0076
Pascal (PIG)	- Harry Bishop/	(703) 931-4937
	- Jim Harvison	(703) 593-2993
SIGAMES	- Jim Eatherly	(202) 667-2287
STOCKSIG	- Robert Wood	(703) 893-9591
Telecomm. SIG	- George Kinal	(202) 546-7270

Washington Apple Pi, Ltd.
8227 Woodmont Avenue, Suite 201
Bethesda, MD 20814
Office (301) 654-8060

ABBS (301) 986-8085

EDITORIAL

I agree with Tom Pittman of San Jose, CA who wrote in the January 30th issue of Infoworld, "I am disturbed by a recent trend in your evaluations to fix on the needs of the novice user. Perhaps you ought to divide the Ease of Use score into two parts: ease of learning for first-time users and long term ease (after 50 hours of use). The latter score is much more important to professionals who must switch between different software products...." Products such as Word Star, ScreenWriter //e, Gutenberg Word Processor, DB Master, etc. have all been criticized at one time or another for being difficult to learn. But, folks, sometimes learning a subject is difficult. Have you ever tried to teach a youngster English grammar? I submit that once learned a software product takes on a new value, a value which may be greater or lesser than that which initial impressions have imparted to the student. Right on, Tom!

But what of the chore of learning difficult software? Can't something be done which is different from the publication of manuals written in arcane hieroglyphics? Yes indeed! Witness the article in this issue by Tom Warrick on the Macintosh and the one by Bob Oringel, "Two Programs About Programs". Software and hardware developers have begun to develop audio cassette tutorials which augment the owners manual. Sometimes diskettes are also included which provide hands-on use of the software while being coached by the cassette. This to me is a good first step for providing a means for lessening the chore of learning a difficult subject. I believe that all software developers should be encouraged to prepare, or contract out the preparation of, teaching aids which provide (at perhaps additional cost) audio cues to diskette tutorials.

VIDEO TAPE OF WOZ VISIT

Washington Apple Pi hopes to have a vhs format video tape of Woz's visit and introduction to the Macintosh, which would be available for loan to Apple Users Groups. Borrowers would be requested to leave a deposit while they use the tape to show to their members. For details, please call the WAP office, (301) 654-8060.

EVENT QUEUE

Washington Apple Pi meets on the 4th Saturday (usually) of each month at the Uniformed Services University of the Health Sciences (USUHS), Building B, 4301 Jones Bridge Road, Bethesda, MD, on the campus of the National Naval Medical Center. Library transactions, journal pickup, memberships, etc. are from 8:45 - 10:00 AM. From 9:00 to 10:00 AM there is an informal "Help" session in the auditorium. The main meeting starts promptly at 10:00, at which time all sales and services close so that volunteers can attend the meeting. A sign interpreter and reserved seating are provided for the hearing impaired.

Following are dates and topics for upcoming months:

February	25	- Personal Instrumentation - Tom Riley
March	24	- Information Services - George Kinal and Bill Wood
April	28	- Memory Boards
May	26	-
June	23	- Garage Sale

Demonstrations for February 25 are: UltraTerm Board & VisiCalc Preboot by Bob Schmidt, and Kaola Pad by Nancy Harvey.

The Executive Board of Washington Apple Pi meets on the second Wednesday of each month at 7:30 PM at the office. All members are welcome to attend. Call the office for any changes.

CLASSIFIEDS

FOR SALE: Apple][+ compatible computer, 64K, upper/lower case, 6502 based. Runs all Apple software. New with 90-day warranty, $520. Also Scriptwriter II software; never used; requires Pascal, $10. Call Doug Schneider, 937-4108.

FOR SALE: Apple][+, starter equivalent. 64K, ALS 80 column card, 1 disk drive, Monitor III. Call weekends (301) 587-3735, weekdays (215) 898-4320.

FOR SALE: Epson MX-80F/T printer. Includes Graftrax+ PROMs, tractor feed and manual. Excellent condition. $320. Contact Jim at 533-0601 (evenings).

FOR SALE: Epson MX-70 printer, $200 or best offer. At 80 cps this capable little printer gives you text AND graphics, adjustable tractor feed, 40 or 80 columns and will accomodate carbons. Print routine to generate true lower case descenders; shipping included. Call (617) 584-8927.

FOR SALE: Videx UltraTerm. Only one month old, used very little - have not yet sent in warranty card. Asking $225. Call Kevin Nealon at (703) 280-1136.

FOR SALE: New MicroSci disk drive. $150/best offer. Call Steve (703) 250-5979.

WANTED: Used modem (preferably Hayes or Novation) and used printer interface card. Card must be compatible with Epson MX-80. Call Mike Beauregard, evenings, (301) 864-4063.

MINUTES

SUMMARY OF DECEMBER EXECUTIVE BOARD MEETING

The Executive Board of WAP, Ltd. met on December 14, 1983 at the WAP office. The form letter for school principals who desire WAP software was reviewed. A lease-purchase agreement is being secured on the Ricoh 3050 copy machine for the office. The possibility of video taping Steve Wozniak's visit was discussed and up to $1000 was authorized for the rental of a larger auditorium for that event. A request was presented by the Housing Opportunity Commission for volunteers to promote computer literacy for disadvantaged youth in Montgomery County. Other areas covered were the 1984 office rent, a phone for the tutorial room, streamlining the group purchase calls to the office, sale of back issues of the Journal at the Garage Sale, and a $250 limit on office purchases to be left to the discretion of the office managers, and another limit of $500 requiring the approval of the President or Vice-President.

DECEMBER GENERAL MEETING

WAP, Ltd. met at USUHS on December 17, 1983, at 10:00 AM. President David Morganstein presided. A season's greeting from Emily was presented. Members were again reminded to adhere to the parking restrictions and not to wander through the USUHS facility. Members were asked to indicate probable attendance at the January meeting on a handout form. Starting in February, 6 or 7 volunteers coordinated by Tom DeMay will help in diagnosing (not fixing) computer hardware problems using club software utilities. A coordinator of demonstration projects in the cafeteria is needed. The office is now open on Thursday evenings, and NEWSIG meets there each Thursday evening. The members favored the purchase of 5 or 6 monitors to be used at the office for tutorials. Appleseeds will meet in the cafeteria. Tutorial offerings were announced, and a tutorial on intermediate-advanced graphics was requested. Topic additions to the Hotline were solicited. A new telephone number for the ABBS was announced. Group purchase reported. Members were asked to direct technical questions to the Hotline rather than to group purchase. Little interest was expressed in having group purchase carry Apple compatible computers for $600 or less. Volunteers were sought to assist in a pilot project sponsored by the Housing Opportunity Commission at Gaithersburg High School. Some rumors about Apple Inc. were aired. Tom DeMay offered to write a rumors column.

SUMMARY OF THE JANUARY EXECUTIVE BOARD MEETING

The Executive Board of WAP, Ltd. met on January 11, 1984 at the WAP office. Reports were made on the response for software from area achools, the office rental agreement, group purchase time spent by Kevin Nealon at the office, sales of WAP Journals in area stores, a give-away IBM system, and feature articles for the February Journal. An idea for networking Board members and active volunteers through an ABBS was accepted for trial, and details of the January meeting were discussed. The Board agreed to spend $300 for diagnostic software/hardware to be used by Tom DeMay and his computer diagnostic team.

MEMBERSHIP IN CALL-A.P.P.L.E.

Membership in Call-A.P.P.L.E., a users group in Kent, Washington, is available at a discount to WAP members. The one-time application fee is $5 instead of $25, and the annual dues of $20 remain the same, making a total of $25 for the first year. Call the WAP office for further details and an application blank.

PRESIDENT'S CORNER
by David Morganstein

THANKS TO THE WOZ. To the man who made it all possible, thanks again! We enjoyed your visit and your talk. We almost hated to take you away from Cupertino and the marvelous new Apple][improvements you are working on. You can tell how enthusiastic and supportive we are about your handiwork from the incredible turn-out we had to hang on your every word. While we hold no "hostility" toward other manufacturers products, we have been waiting for the announcements from Apple that would put us back in the ball game again. Go get 'em!!

SOFTWARE FOR SCHOOLS. Almost a dozen schools have contacted us requesting software from our public domain library. If your school has Apples, have your principal complete our request form and send in an unopened box of blanks. We will fill requests for ten free selections promptly.

HELP FOR ROTTEN APPLES. Tom DeMay has arranged a ten person team of volunteers to help diagnose those infrequent but frustrating cases of Apple-itis. These folks will take calls and open their home during specified hours to other members with hardware problems. Please read Tom's article for the names, numbers and times these folks are available. To Tom and his volunteers, thanks for providing us with this help!

HAVE YOU WRITTEN ANY GOOD PROGRAMS LATELY? We recently discussed allowing program authors to submit articles and source code about their handiwork. The article may contain a note that a machine readable copy could be obtained from the author for a small fee, thereby providing some compensation for your efforts. If you are such an author, we await your contribution. Put your address at the end of your article with a note that a blank disk and self-addressed mailer should be accompanied by a three dollar handling fee.

GARAGE SALE A SUCCESS. The USUHS cafeteria was never so crowded. It was shoulder to shoulder. Most bringers of used items seemed to leave almost empty handed. We distributed over 600 copies of the Journal, so you know the attendance was unusually high. Joe Fuchs, organizer, again showed his administrative talents by making the sale a success. Thanks to you, Joe, and to your many volunteers who helped coordinate the activity.

APPLE TEAS. We are going to organize Apple teas for this Spring. If you would be willing to host a small group of six to twelve WAP members for a discussion group in your home, please contact Paula Benson. You needn't be the resident Apple Alchemist. We can find "experts" to answer questions. The teas are a means of recapturing the informal feeling we used to have back in the "good old days" when the mailing list fit on an 8 1/2" by 11" page.

WHAT'S A 65802? Many of the 1.5 million Apple][,][+ and //e owners must often wonder if there is a future for their computing machine. With the continued switch to 16 bit computers, we wonder if our little Apple has any future. Interesting news has appeared in two articles (one in BYTE and one in Infoworld) which described a recently announced 65816 microprocessor chip. This little beauty is upward compatible with the 6502 (that is, it recognizes all of the 6502 instructions, as well as, performs some new miracles). It is a true 16 bit chip (unlike the 8088 used in the IBM PC) in that it has a 16 bit data bus (the 8088 with its eight data lines can only move an 8 bit byte at one time). This new chip is rumored to be able to address 16 megabytes of memory, just like the 68000 chip used in the LISA (16 times more memory than can be addressed by the 8088 chip). This 65816 is not all that relevant to us Apple owners. What is, however, is a "baby brother", the 65802. This little IC is pin-compatible with our 6502 and has most of the other nice features, like faster operation and gobs of memory addresses. By pin compatible we mean you could conceivably remove your 6502 and insert the new chip in its place (and make a number of other important hardware changes) and just maybe have a revitalized monster computer!!!. While there is no guarantee that it will happen, it's exciting to consider. And now that the WOZ is back in Cupertino, who could not help but speculate...

ENTER THE MAC. The waiting is over, the word is out and the word is good. Apple's new business machine has been unveiled and it has an impressive appearance. With built-in video, movable keyboard, 128K of memory and built-in 3 1/2" disk drive with 400K storage in single-sided mode, it looks as though the MAC can solve many problems. The internal ROMs contain a lot of programs including a word processor. Just turn it on and its ready to go. The price is surprisingly low for the features (well under competing products like the IBM PC). We are looking for a MACSIG chairman!

WAP HOTLINE

Have a problem? The following club members have agreed to help. PLEASE, keep in mind that the people listed are VOLUNTEERS. Respect all telephone restrictions, where listed, and no calls after 10:00 PM except where indicated. Users of the Hotline are reminded that calls regarding commercial software packages should be limited to those you have purchased. Please do not call about copied software for which you have no documentation. If the person called has a telephone answering machine, and your call is not returned, don't assume that he did not try to return your call - perhaps you were not home. Try again.

General	Dave Harvey		(703) 527-2704
	Robert Martin		(301) 498-6074
APPLE SSC	Bernie Benson		(202) 546-0076
Apple TechNotes	Ed Schenker		(301) 977-7349
	Lance Bell		(703) 550-9064
Basis 108	Loftus Becker		(202) 338-5217
Communications Packages and Modems-Telecom.			
Anchor Mark 12	George Kinal		(202) 546-7270
Anchor Sig. & BIZCOMP Modem	Jeremy Parker		(301) 229-2578
Apple CAT II	Ben Acton		(301) 428-3605
ASCII Express	Dave Harvey		(703) 527-2704
CompuServe	Jerry Michalski		(703) 442-8835
General	Tom Nebiker		(216) 867-7463
	Tom Warrick		(301) 656-4389
Hayes Smartmodem	Bernie Benson		(202) 546-0076
Micromodem	Tom Warrick		(301) 656-4389
Omninet	Tom Vier (1-6 PM)		(703) 860-0083
Source & Transcend	Jerry Michalski		(703) 442-8835
VISITERM	Steve Wildstrom		(301) 564-0039
XTALK CP/M Comm.	Bernie Benson		(202) 546-0076
Corvus Hard Disk	Tom Vier (1-6 PM)		(703) 860-0083
Expediter Compiler	Peter Rosden		(301) 229-2288
Data Bases			
dBase II	Loftus Becker		(202) 338-5217
	Paul Bublitz		(301) 261-4124
	John Staples		(703) 759-3461
DB Master	Doug Daje		(301) 868-5487
	Dave Einhorn		(301) 593-8420
Data Perfect	Leon Raesly	*	(301) 460-0754
Data Factory	Bob Schmidt		(301) 736-4698
General Manager	Loftus Becker		(202) 338-5217
	Leon Raesly	*	(301) 460-0754
InfoMaster	Doug Daje		(301) 868-5487
List Handler	Jon Vaupel		(301) 977-3054
Graphics	Bill Schultheis		(703) 538-4575
Home Accountant	Leon Raesly	*	(301) 460-0754
Languages (A=Applesoft, I=Integer, P=Pascal, M=Machine			
A	Peter Combes		(301) 871-1455
A,I	Jeff Dillon		(301) 422-6458
A	Richard Langston		(301) 258-9865
A	Mark Pankin		(703) 370-9219
A	Leon Raesly	*	(301) 460-0754
A,I,P,M	Bill Schultheis		(703) 538-4575
A,I,M	Richard Untied		(703) 241-8678
P	Dottie Acton		(301) 428-3605
P	Donn Hoffman	*	(202) 966-2616
Languages, contd.			
LOGO	Ron Murray (eve.)		(202) 328-3553
P-Lisp	Fred Naef		(703) 471-1479
Multiplan	Terry Prudden		(301) 933-3065
Operating Systems			
Apple DOS	Richard Langston		(301) 258-9865
	Richard Untied		(703) 241-8678
CP/M	Robert Fretwell		(703) 971-2621
Paddles	Tom Riley (eve.)		(301) 340-9432
Pers. Filing Sys.	Bill Etue		(703) 620-2103
	Ben Ryan		(301) 469-6457
	Jenny Spevak		(202) 362-3887
Printers			
General	Walt Francis		(202) 966-5742
Anderson Jacobson	Bill Etue		(703) 620-2103
	Leon Raesly	*	(301) 460-0754
Apple Dot Matrix	Joan B. Dunham	*	(301) 585-0989
Daisywriter 2000	Henry Greene		(202) 363-1797
IDS 460	Jeff Stetekluh		(703) 521-4882
MX-80	Jeff Dillon		(301) 422-6458
Okidata	Scott Rullman		(301) 779-5714
Silentype	Bruce Field		(301) 340-7038
Spreadsheet 2.0	Leon Raesly	*	(301) 460-0754
Statistical Packages	Jim Carpenter		(301) 371-5263
Stock Market	Robert Wood		(703) 893-9591
Tax Preparer-H.Soft	Leon Raesly	*	(301) 460-0754
Time-Sharing	Dave Harvey		(703) 527-2704
VisiCalc	Walt Francis		(202) 966-5742
	Leon Raesly	*	(301) 460-0754
Word Processors	Walt Francis		(202) 966-5742
Apple Writer II	Doug Daje		(301) 868-5487
	Dianne Lorenz		(301) 530-7881
	Leon Raesly	*	(301) 460-0754
	Tom Warrick		(301) 656-4389
Gutenberg	Neil Muncy		(301) 251-9330
Letter Perfect	Cara Cira		(301) 468-6118
	Leon Raesly		(301) 460-0754
Magic Window and II	Joyce C. Little		(301) 321-2989
Peach Text	Carl Eisen		(703) 354-4837
PIE Writer/Apple PIE	Jim Graham		(703) 643-1848
ScreenWriter II	Peter Combes		(301) 871-1455
Supertext II	Doug Daje		(301) 868-5487
	Peter Rosden		(301) 229-2288
Word Handler	Jon Vaupel		(301) 977-3054

*Calls up until midnight are ok.

GENERAL INFORMATION

Membership dues for Washington Apple Pi are $25.00 for the first year and 18.00 per year thereafter, beginning in the month joined. If you would like to join, please call the club office or write to the office address. A membership application will be mailed to you. Subscriptions to the Washington Apple Pi Journal are not available. The Journal is distributed as a benefit of membership.

Current Office hours are:

Mon, Wed, Fri - 10 AM to 2:30 PM
Tue - 12:30 to 2:30 PM & 7 to 9:30 PM
Thurs - 10 AM to 2:30 PM & 7 to 9:30 PM
Sat - 12 to 4 PM (meeting Sat only)

THE MACINTOSH: A First Look

by Thomas S. Warrick

To describe Apple's new Macintosh computer in a sentence, use this: The Macintosh is a Lisa at an Apple][price. That says a great deal, and it is intended to. Apple's Lisa computer, introduced a year ago, offered software that won the 1983 Software of the Year award from the prestigous microcomputer newsweekly InfoWorld. Macintosh must be regarded as the odds-on favorite to win the Hardware of the Year award for 1984.

This article will cover (i) the Macintosh hardware, (ii) the Macintosh operating system, and (iii) the Apple software that comes with it. Other articles that follow will describe third-party software and the languages available to use on the Macintosh, the importance of Macintosh to Apple Computer, Inc., and the failings of the Macintosh as a "hacker's machine."

The list price for the Macintosh -- which is already coming to be known by the abbreviation "Mac" -- is $2,495. Most users will want to add the Apple Imagewriter printer, a revision of the Apple Dot Matrix printer that is intended to print Lisa and Macintosh images much faster than the old Apple DMP. The Imagewriter will be sold with the Mac for $495. Other accessories include a numeric keypad ($129), a carrying case ($99), and a security kit to discourage theft ($49). Apple will sell disks for the Mac in boxes of 10 for $49, which is about half of the standard list price for such disks. The Mac comes with Apple's standard 90-day warranty, and the "Apple Care" extended warranty will also be available, although the price is not yet known. The Mac will come initially with two software packages, MacWrite and MacPaint, which are word processing and graphics programs. After the initial offering period, these programs will sell as a set for $195.

One issue of overriding concern to potential purchasers is the availability of the Mac. On or about January 24, Apple dealers around the country were shipped three units each. One will be for display, and the other two will be for sale. That is all that dealers will receive until February, and shipments are likely to trickle in for at least four months. The Mac will be in scarce supply for about nine months or so, and waiting lists are likely to be long. Apple is planning a massive advertising campaign beginning during the Winter Olympics, so interested buyers should place orders early.

The Macintosh is made in a single factory in Fremont, California. When the factory reaches full production, expected to be in June of this year, it should be capable of producing 1,000 Macintoshes in an eight-hour period. Apple has insisted on "zero defects" parts, and quality control is likely to be strict, in order to avoid the initial problems that plagued the Apple ///. But strict quality control, however desirable, also means that production delays can arise. Most purchasers will decide that Mac will be worth the wait. (Ed. Note: This factory was specially constructed for manufacturing Macs and features extensive robotics. Apparently only about 100 employees will be needed to attain the 1 every 27 seconds goal, including packaging, etc. We learned that Steve Jobs visited "all" manufacturing facilities in the area to learn how they manufactured their computers, including Apple clones.)

I. THE MACINTOSH HARDWARE

The Macintosh looks like no other microcomputer on the market. It comes in three parts: the main unit, which encloses a monitor and a disk drive, a detachable keyboard, and a mouse. The main unit of the Mac is the same beige color as all other Apple computers. The unit is very small: 13.5 inches high, 9.7 wide and 10.9 inches deep. By comparison, the Apple //e with a Monitor II on top is 15.2 inches high, 15 inches wide and 17.5 inches deep. Measurements are deceptive, however, until you include the detachable keyboard, which is about 12 inches by 5 inches, and an area large enough for the mouse to run around in. More will be said about both the keyboard and the mouse below, but even adding the area required to use both devices, the "footprint" of the Mac is smaller than that of Lisa and no bigger than that of an Apple //e with a disk drive off to one side.

Visible from the outside front of the main unit is a 9-inch (diagonal measure) black-and-white monitor and a Sony 3 1/2-inch single-sided disk drive. The monitor produces the same black characters on a white background as are produced by Lisa. Many -- but not all -- industrial design engineers maintain that this is the easiest type of monitor to work with. The monitor has a resolution of 512 dots by 342 dots, compared to the 280x192 resolution of the Apple][. Purchasers of the Mac should keep in mind that the brightness of the monitors varies significantly from individual unit to individual unit. Of the four units this writer saw, the brightest picture seemed twice as bright as the dimmest, even with both brightness controls adjusted to maximum brightness. This does not affect the utility of the machine for most people, although people with vision problems or people who must use the Mac against a window that looks out over a bright area should insist on testing out their particular Macintosh before taking it out of the store. Dealers will probably not be aware of this problem. Color is not available, nor is there any jack or port to which a color monitor could be added.

The Sony 3 1/2-inch disk drive uses single-sided disks with a capacity of 400K. Although Apple has made no formal announcement, double-sided disk drives will probably be available later this year. An external 3 1/2-inch drive will be available from Apple. Apple is not planning to build a hard disk for the Mac, although at least two hardware vendors will have them out this year. Wise buyers not in the disk copying business will probably stay away from the single-sided Apple drives as a second drive.

The Sony disk is in itself a marked improvement over the old 5 1/4-inch versions now used by all Apple computer owners. Not only does the disk contain almost three times the storage capacity of standard Apple-formatted disks in a disk with only 44 percent of the area, the Sony disk is encased in a rigid plastic shell that resists bending, thereby eliminating the need for separate paper dust jackets. Perhaps more importantly, when the disk is not actually in the drive, a metal slide covers the hole in the jacket through which data is read and written. Thus there is no easy way for dirty fingers to deposit dirt or oil on the surface of the disk. The disks can be carried easily in shirt pockets without fear of damaging data.

The top of the Mac conceals a convenient handle not

contd.

unlike the handle on desk telephones. The Mac weighs less than 20 pounds, and is truly a transportable computer. It will not fit under an airline seat, but should fit in the overhead rack of most airplanes.

The rear of the Mac is fairly simple. Most notable -- and disappointing -- is that there are no slots for peripheral devices. The Mac is, as noted in a companion article in this issue, not a "hacker's machine." Peripheral devices will be connected to one of five ports in the back. From left to right, as you look at the rear of the Mac, they are:

1. The mouse connector, a DB-9 connector like the one that the new Apple paddles plug into on the //e.

2. An external disk drive connector, a DB-19 connector.

3. An RS-232 serial interface for the printer, a DB-9 connector. At present, the only printer that will take advantage of the graphics power of the Mac is the new Apple Imagewriter. The speed of the Imagewriter driven by the Mac rivals the best parallel printers. Apple boasts that the Imagewriter will print at 180 characters a second. Other printers will no doubt soon be available, but the offered price and features for the Imagewriter, $495 if bought with the Mac, make it a recommended buy.

4. An RS-232/RS-422 serial port, a DB-9 connector. "RS-232" is the protocol used by many printers and modems to communicate with their host computers. This particular serial port is intended primarily for use with modems. The "RS-422" protocol is used by Apple's new networking system, AppleBus. AppleBus will be a very inexpensive way for any Apple computer to share data or expensive peripherals.

5. A mini-phono plug jack so that you can plug your stereo system into your Mac. The Mac has a built-in polyphonic synthesizer with four voices. A built-in speaker is adequate for producing pleasant tones, but more complex music and even speech requires more than a small speaker.

The rear panel also has a notch for a security plug. Apple will for $49 sell you a security kit that includes two steel plugs, one for the main unit and another for the keyboard. The plugs are attached to a steel cable like that of a bicycle security system.

The on-off switch on the Mac is a true on-off switch, in that it actually turns the machine on and off. This differs from the Lisa's "On/Off" switch, which is merely requests the machine to turn itself on or off. One example of the ergonomics of the Mac is that, while the on/off switch is on the rear panel, the side panel has a rough texture except at the vertical height of the on/off switch, thus making it easy to find without groping around the back.

The rear panel is attached to the chassis by two screws that require a non-standard, star-shaped screwdriver to disassemble. Apple is clearly trying to discourage users from entering the inside of the Mac. One of the most interesting touches of the Mac is therefore inaccessible to purchasers. The signature of each of the members of the principal design team of the Mac has been molded into the inside of the rear panel.

One of the most novel features of the Mac is the Reset key. There is no reset key on the keyboard itself. Instead, there is a little one-piece plastic assembly that plugs into the left side of the Mac. Pressing one switch on this assembly generates a reset signal to the system software. Pressing the other switch generates a non-maskable interrupt (NMI) to the hardware. This inexpensive plastic device can be removed manually for those applications, such as in classrooms, where the user should not be able to reset the machine. The utility of a NMI switch is somewhat uncertain, but it may be useful as a secondary reset key. While the author could not test these keys in operation, it may be that the "reset" key is actually the equivalent of a Control-C key on most computers, i.e., a way of stopping execution of a program. The NMI key would therefore be used as a true hardware reset command.

The keyboard of the Mac is detachable. It is connected to the main unit by means of a cord absolutely identical (except in length) to the one used by most telephones to connect the handset to the main unit of the telephone. The keyboard is very similar to the newest Apple //e keyboard. The feel is soft, without any tactile feedback. It must be regarded as somewhat of a disappointment to all IBM Selectric users. The letters are in the upper left hand corner of the keys.

What must come as a shock to many experienced computer users is that there is no control key. The place of the control key on the //e keyboard is taken up by a caps lock key similar in function to the well-designed caps lock key on the //e. The bottom row that on most keyboards holds the space bar contains in the Mac a space bar and four other keys. Two are called "Option" keys. One has what looks like a four-leaf clover or fan symbol on it, and is called "Feature". The fourth is entitled "Enter". In a way, all of these can serve like control keys. It appears from the software available during pre-release testing that the Feature key may simply be a disguised control key. If so, its placement is somewhat awkward.

The most noticeable innovation with the Lisa hardware was the introduction of the mouse. This has been carried over to the Mac, although the button on the mouse has been improved. The mouse on the Mac is an object about the size of a cigarette pack with a single button on top. The mouse is connected to the Mac by means of a thin cord. The mouse has inside it a simple ball that detects motion. (Picture a trackball turned upside down.) The mouse is connected to a pointer on the Mac's screen and movement of the mouse is therefore duplicated on the screen. Thus, to move the pointer left, move the mouse to the left.

The jury is still out on whether the mouse is the easist way to use a computer, or whether function keys or a touch-sensitive screen will prevail. One wag -- who makes a device that competes with mouse technology -- has said that the mouse is a perfect device for users with three hands. There is much truth to this. Using a mouse takes some getting used to, particularly for users of computers with many function keys. After a while, however, using the mouse will likely become second nature for every application except word processing. But most software writers will recognize that users will not always want to use a mouse. Well-written programs will therefore also recognize control characters from the keyboard, using the Feature or Option key.

The interior of the Mac is, as has been noted above, inaccessible to users without a special tool. The contents of the interior are, however, very important to the users of any computer system.

The Mac comes today with 128K of random access memory (RAM), using 64K chips like those used in the Apple //e. (Each chip holds one bit of an eight bit byte, therefore eight 64K chips are required to store 64K bytes of memory. The term "64K chips" should more accurately be "64K bit chips.") When new chips capable of storing 256K bits come on the market later

contd.

this year, the Mac will be upgradable to 512K of RAM.

The operating system for the Mac, including the disk operating system and the graphics generator, is contained in 64K of read-only memory (ROM). Users need not fear that bugs will be uncorrectable, as programs actually use ROM routines through vectors in RAM. Thus if it is necessary to patch a routine in ROM, the patch can be installed in RAM.

The motherboard of the Mac is an incredibly tiny 80 square inches. The designer of the board is reputed to have been with Apple for one-seventh of his life: three years. Right. That makes him twenty-one.

All chips are soldered directly into the motherboard rather than being inserted into sockets. This eliminates one of the principal sources of trouble with microcomputers, particularly transportable microcomputers, but it makes repair much more difficult. When Macs require repair, Apple has instructed its dealers to swap boards rather than try to diagnose the cause of the problem. This policy, which also exists with regard to the //e, will decrease the cost of labor for repairs but may increase significantly the cost of parts. Inquire at your dealer's what the cost of out-of-warranty repairs will be. Finding places to have Macs repaired should not be a problem, as most licensed Apple repair facilities should be willing to repair Macintoshes.

The Mac hardware, in addition to the four-voice synthesizer mentioned above, also includes a built in clock and calendar chip. A small battery is supposed to keep the time up to date for up to two years even if the power is off.

The heart, or brain, of any microcomputer is the microprocessor, the central processing unit of the computer. The Mac's is at this time a Motorola 68000. The 68000 is called by Apple a 32-bit microprocessor, but that is true only to the same extent that the IBM PC's 8088 is a 16-bit microprocessor. The data paths on the 68000 move 16 bits (two bytes) of data at a time, compared to the 8 bits at a time moved by the microprocessors in the Apple][and the IBM PC. The instruction set, however, uses 32 bit instructions, giving the microprocessor the capabilities of minicomputers that five years ago required a box the size of a large desk. Later this year, Motorola and others should announce the 68020, which will be the same as the 68000 except with a 32 bit data bus. The 68000 has a theoretical capacity of addressing 16 megabytes of RAM. Without slots, however, there is no way a user of the Mac will be able to address more than one-half a megabyte of RAM even after the 256K bit chips come out later this year.

The clock rate on the 68000 is 8 megaHertz, which is twice that of the original Lisa and eight times that of the Apple][family. This understates the power of the 68000, however, because the instruction set is much more powerful. For example, multiplication on the 6502 used in the Apple][requires a series of additions and bit shifts. Multiplication on the 68000 can be done by one instruction. Apple says that it will release an Assembler/Debugger for the Mac some time in the third quarter of 1984.

II. THE SYSTEM ENVIRONMENT

But even though the hardware of the Mac is state-of-the-art for its price, the real beauty of the Mac is the way it works, its "system environment." Those who have seen Apple's Lisa will recognize Mac's indebtedness to the "Star" developed by Xerox's Palo Alto Research Center. The Mac uses windows (subdivisions within the screen image that can store different things), icons (small graphic images to represent files or tasks), and a pointer on the screen for the user to indicate to the computer the task the user desires to perform.

The proper way to approach using the Mac, as with the Lisa, is the metaphor of the desktop. One of the most important and useful features of the operating environment is that you can overlay different applications on the "desktop" provided by the screen. Thus, you can start to write a report, stop to work on a spreadsheet, convert the spreadsheet into a graph, put away the graph program (but not the graph itself), put away the spreadsheet, and insert the graph directly into the report. This is an example of "integrated software," one of the cliches of 1983-84, but its status as a cliche belies its utility.

This software technology on the Lisa was named the Software of the Year for 1983 by InfoWorld, and the Mac software developers have solved many of the flaws present in the Lisa software. Graphic images move much more quickly on the Mac. For example, on both the Lisa and the Mac the user would see windows shrinking to a point as a file was closed. In the Lisa, this process took long enough to be distracting and to make the user think, "Why waste the machine's time showing me that? Why not just close the darned file?" The process has been accelerated in the Mac to the point where it's just long enough for the user to reorient his or her thinking for the next task.

III. APPLE' SOFTWARE FOR THE MACINTOSH

With the purchase of a Mac, one gets a system master disk called the "Journal." This disk has system utilities: not only the standard kind that allows you to copy files, but also system utilities unlike any seen elsewhere but on a Lisa. For example, one system utility that you can call up in the middle of many programs is a reverse Polish notation calculator. Thus if in the middle of a word processing application you need to know the sum of a column of numbers too short to justify the use of a spreadsheet, you can call up the calculator to do the work for you, then resume the word processing application.

One of the most innovative features of the Mac, and one that is certain to be copied by other manufacturers, is the combination of a tutorial cassette and a tutorial program on the system master disk. The standard audio cassette is played on a regular cassette player, and it coaches the user through a computer program that shows off the capabilities of the Mac, explains how to use it, and gives the novice user an opportunity to play with the machine. The lessons are slow enough to be of real use to the novice, although too slow for the expert Lisa user. Nevertheless, all purchasers will want to sit through a session with the cassette just to hear the piano music that accompanies the instructor's words.

Apple's literature and advertising for the Mac plays heavily on the user's manual. It will be fairly short, but with lots of pictures and the graphics you would expect since it appears that the manual was written on a Macintosh. A quick look at a preliminary version of the manual reveals it to be similar in scope and quality to the Apple //e user's manual.

Note that the user's manual is not a reference manual or a programming manual. The reference manuals comprise three full three-ring notebooks and are as detailed as you would expect from having seen the new Apple //e reference manual. Few will be surprised to learn that the reference manuals cost extra.

Mac comes with no programming languages built-in, although BASIC, Pascal and Forth will be available almost immediately from third parties, with Logo and

contd.

FORTRAN due out some time in 1984. Languages and third-party software are discussed in another article appearing in this issue.

For an introductory period, purchasers of the Mac will get two programs from Apple, MacWrite and MacPaint. MacWrite is a word processing program very similar to LisaWrite, although the screen display of MacWrite can keep up with a 60-word a minute typist whereas LisaWrite could not. While only a preliminary version of MacWrite was available for review, it appears that, given the power of the Mac, MacWrite is an indifferent word processing program that will quickly be supplanted by other, better word processors, most notably Microsoft's Word, which will be out in the third quarter of 1984. That is not to say that MacWrite is a bad word processing program overall, only that it should be much, much better. The power of the 68000 and the Mac operating environment is such that it should be possible to write a word processing program that would not only beat Word Star but would rival the best standalone word processors such as NBI, IBM's Displaywriter and the late Dictaphone Dual Display.

In terms of features, MacWrite appears to lack many useful features found in other word processing programs, but frankly this is probably be due to the ignorance of this reviewer. One would be hard-pressed to presume that one could even begin to use the features of any good word processing program without a moment's study of a manual. Yet one of the testaments to the Macintosh system environment is that it is easy to write programs that are very easy to use. A "menu bar," common to virtually all Macintosh applications, contains some of the basic functions of any word processor, such as File, Edit, Search, Format and Style. Use of these functions is usually by means of pull-down windows activated by the mouse.

MacWrite's principal advantage is that what you see is what you get, i.e., text will be printed on paper as you see it on the screen. You can change character fonts, type size and spacing. The ability to see text appear on the screen in either single, double or triple spacing is what sets MacWrite apart from many other word processors.

MacWrite is relatively easy to use, being more difficult than Bank Street Writer but much less difficult than Word Star, for example. The what-you-see-is-what-you-get feature is useful if your output is to the Apple Imagewriter dot matrix printer, but it remains to be seen whether and how well Mac can drive letter-quality printers. Further, at least in the version this writer saw, all characters appeared in proportional spacing, which makes setting up columnar text difficult and makes judging spacing next to impossible. Literature for the Mac, however, claims that users can design their own typefaces. If these user-designed typefaces can be used with proprietary programs such as MacWrite, this problem will be alleviated.

Experienced typists will find that no matter how much they use and practice with the mouse, it will slow them down in some applications. Use of the Feature and Option keys in conjunction with regular letter keys is permitted in MacWrite and will avoid this problem.

MacPaint is a program that is not likely to be supplanted in the near future, if at all. It is a superlative graphics generator, giving the commercial artist all of the tools necessary to generate high-quality graphics. Others in need of high-quality graphics include architects, draftsmen, and doodlers of all ages. There are only two drawbacks to MacPaint, which are due to the Macintosh: no color, and graphics output is limited to a dot matrix printer. The second will likely be overcome as packages come out to drive color printers and plotters. The former is not likely to be solved easily.

Users will recognize MacPaint as an improved version of LisaDraw.

Three other programs that came bundled with the Lisa have no Macintosh analogue as of today. MacProject, due to be released later this year, will duplicate and enhance LisaProject, which was an innovator in project management software. LisaCalc and LisaGraph appear so far to have no Mac version. This statement is based on Apple's promotional literature, which shows no such programs as MacCalc or MacGraph. Instead, it shows Microsoft's Multiplan and Microsoft Chart. The former is a spreadsheet regarded by some as inferior to VisiCalc and vastly inferior to Lotus 1-2-3. Other companies will soon introduce their own integrated spreadsheets for the Mac by autumn of this year. However, Lotus 1-2-3 will not be among them until the 512K Macintosh upgrade is available.

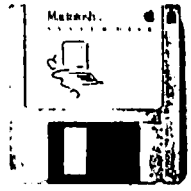
IV. CONCLUSION

The Macintosh is, as those closely associated with it will admit, not for everyone. It will certainly not supplant the Apple //e, particularly for games or education. Mac is meant for people in professional and managerial occupations who work with information. For people in those areas, which include most of the members of Washington Apple Pi, the Mac is an excellent piece of hardware. What made both the Apple][family and the IBM PC into successes, however, was third-party software for a wide variety of applications. The Macintosh system environment should make such software easy to use. Ease of use will be the major selling point of the Mac after the novelty aspects wear off.

Should you buy now or wait? From a hardware standpoint, there are three reasons for waiting. Later in this year, the Mac will have 512K of RAM, not just 128K. Also, the single-sided built-in disk drive will be replaced by a double-sided disk drive. Finally, the 68000 microprocessor will likely be replaced by a 68020. Ask dealers which if any of these upgrades they or Apple will make at no additional charge or for the cost of parts only.

But as for me, I'm buying a Macintosh. Now.

Postscript on the name "Macintosh": Apple admits that it is a misspelling. The dictionary admits of three possible spellings of the word. "Mackintosh" is a waterproof jacket, or the fabric used in such a jacket, designed by Charles Macintosh (1766-1843), a Scottish inventor. "Macintosh" is a variant for "mackintosh." The apple with a similar name is "McIntosh," named for John McIntosh, a Canadian who first discovered and cultivated in in 1796. The name of the computer, according to the story I've heard, came from a meeting of the design group, who were discussing their favorite variety of (edible) apples. More people in the group preferred McIntoshes, which was agreed upon as the name of the computer. Somehow the name was erroneously recorded as "Macintosh," which in any event looks better on a computer than either of the other two variations, even the "correct" one.



Macintosh's 400K 3½" disk.

MacSoftware: More Than Fast Food

by Robert C. Platt

A diversity of software is the principal strength of the Apple II, and Apple acknowledges that getting software houses committed to producing software for the Macintosh is critical to its success. After a "test drive" of the Mac and hearing Apple's plans, software availability appears to be the Achilles' heel of the Mac.

THE GOOD

Apple has been cultivating independent software developers and has made Mac's available to several of them for more than a year. Hence, Apple's promotional material features the Presidents of Microsoft, Lotus and Software Publishing Corp. smiling and proclaiming the Mac as a "whole new software standard." Bill Gates of Microsoft predicts that 50 percent of that firm's 1984 revenues will come from the Mac.

THE BAD

Given the growing base of 68000 software already developed for the Lisa and the Sage computers, one would expect a flood of software on announcement date. However, the Mac is presently limited to 128K of RAM, far less than that available on other 68000 machines. Further, as indicated in the accompanying article, Mac's operating system (OS) is fixed on 64K of ROM, and is accessible through 460 different system calls. This leads to a cohesive "user interface." (That is, most programs make you feel that you are using a Lisa.) Yet, it provides a major obstacle to transporting programs from other 68000 machines. None of the three operating systems popular on 68000 machines, CP/M-68K, UCSD p-System, or Unix have been announced for the Mac. Nor does Apple itself plan to market these systems as it has for the II and the Lisa.

Given Mac's unique OS, most commercial software development both within and outside Apple has been done with a Pascal compiler that Apple originally developed for the Lisa. This compiler generates machine language with calls to Mac's OS for graphics, input/output and memory management. (Ref. 1) The Mac Pascal software development system (which is the equivalent to the Lisa WorkShop system) will be sold publicly in the second quarter of '84.

THE UGLY

With the absence of a hard disk and only 128K, Apple is trying to target the Mac for that portion of the office market that requires less data manipulation than the Lisa. Apple's promotional video tapes proclaim, "The market for the Macintosh is the desk." and the goal of the Mac is to "create a product that will improve the life at the desk." When asked to distinguish the Lisa market, the main points were the absence of a hard disk, other operating systems and the software such as accounting packages that would use these features.

THE GOODIES

However, third party software and hardware vendors will expand the Mac beyond Apple's targeted market, and I predict will turn the Mac into the biggest phenomenon since the II. Most notably, Tecmar plans to offer a removable Winchester hard disk and expansion chassis by second quarter 1984. Devong, Microtek and Prometheus have also announced hardware accessories.

The most impressive software package that I tried was Instant Pascal and Instant Basic by Think Technologies. These products are designed for learning these languages. The Mac screen is divided into two windows with the source program entered into the left window and output displayed on the right. After entering a short Pascal program, I used the mouse to select the "check" option from the menu. Within a second, my Pascal program had been checked for syntax errors and all keywords had been converted into a script font. When a syntax error is detected, a comment noting the nature of the error is automatically inserted at the trouble spot. Upon selecting "go" from the menu, an interpreter executed the program as fast as any Basic interpreter that I have seen on the //e. These extremely user friendly Pascal and Basic implementations will be very popular.

If you don't like instant languages, Apple will sell Microsoft's "GeeWhiz" (GW) Basic for $150. This implementation has been popular on the IBM PC and clones. For you Forth-fans, Creative Solutions has announced a full feature Forth development system.

Apple also promises to market MacTerminal in February. (Or you can wait for terminal packages from Dilithium Software, Hayes, Creative Solutions, or Winterhalter.) Apple will sell MacProject and MacDraw (two Lisa converts) in the second quarter of 1984. Finally, Logo for the Mac will be available from Apple in the third quater of '84.

RECREATION

A number of software publishers have planned Mac versions. You will have Wizardry, Sargon III, Zork et al from Infocom, and Millionaire, Barron and other money games from Blue Chip Software. Penguin will convert its game line, and its Graphic Magician for debuts in the first quarter of '84. Other vendors promise graphics utilities (which are a breeze with Mac's OS.)

OLD FRIENDS

Many other Apple software packages will be converted. The databases include: dBase II, Friday!, Incredible Jack, Habadex, MPL/database, DBMaster, Microsoft file, and pfs:file. Firms planning accounting packages include: BPI systems and Continental Software's The Home Accountant. To insure that desk-workers remain psyched up, a number of firms are planning business simulation and of course, Human Edge Software's psych-them-out strategy programs. Finally, three developers will offer typing tutor programs (although you will be surprised at how many commands can be entered quickly with a mouse.)

It will be interesting to see how many of these old friends will take on a Lisa-like cast. (The Multiplan for the Mac was very un-Lisa-like.) With Mac's OS, a mouse and menu interface become easy to implement. Whether these programs will change their formats is up to each developer. Yet, one feature will not change: I doubt that any of these programs will offer multitasking or concurrent processing. Mac's OS generally supports only one program running at a time. The only exception to this are "desk accessories" which are short routines that will project a clock or calculator on a portion of the screen while your main program is running. However, given the growing number of products which support concurrent processing (Ref. 2), I

contd. on pg 17

THE MACINTOSH AS VIEWED BY AN ENGINEER

by Tom Riley

THE MACINTOSH SCREEN

"My eyes feel like sanded granite", Charles A. Lindbergh, Orly Airport, Paris, 1927.

The screen of the Macintosh is a monochrome (blue-white), nine-inch diagonal cathode ray tube (CRT). It is mounted in a box together with the motherboard and disk drive; the keyboard is an independent unit and can be positioned in front of the computer to suit the user. The screen is tilted back slightly from the vertical and its center is about 10 inches above the desk top. The only control is a brightness knob, but at the brightest setting the picture is still a bit dim and somewhat blue rather than a true white.

In the standard software we checked out the background color was blue-white and the printing appeared in various shades of blue-black. The work area was divided into many blocks or screens which contained a variety of print fonts, icons, and lines.

I was never able to read the screen well; within an hour my head was aching, and within two hours my eyes were watering so badly that I could not continue using the machine. I don't have good vision, but I have never had difficulties of this magnitude with any other CRT. I am nearsighted and wear bifocals with a correction for astigmatism, but I am able to spend hours at my Apple][, using an old 12-inch green monitor, without any eye fatigue. I have had difficulty reading the small monitors on Osbornes and Kaypros; amber screens in particular give me trouble.

The Macintosh has no external video jack, but the dealer should be able to add one. A larger monitor could be used with the jack, but I don't think that this alone will solve the eyestrain problem, and besides the two units would clutter up the user's desk. The instructors at the demo didn't know of any way to inverse the screen to white on black, which might provide some relief.

As I see it, the problems with the Macintosh screen are multiple: its small size, the enormous variety of fonts and icons, the poor contrast, and the spatial relationship of the sceen to the eye. None of these problems are easily addressed, and the frozen hardware design doesn't give the user much room to experiment. The excessive number of type styles utilized in the standard software made it difficult for me to read. This problem could be addressed only by a massive software rewrite.

By spinning the CRT yoke 180 degrees and cutting a hole in my desk, I think I could bring the small screen close enough to the keyboard so that both would fall in the visual field covered by my bifocals. I am certain this approach will not be popular, might overheat the unit, would cost me my favorite desk drawer, and would create a real knee bumper. The accompanying sketch shows a special stand that would eliminate the need for the hole in the desk.

I would recommend that anyone evaluating a Macintosh spend several hours on the machine in the dealer's showroom. This is especially important if the purchase of several units for an office is being considered, and people other than the buyer will be using them. All potential users should spend real time on this machine, particularly those who have experienced visual fatigue working with other terminals or word processors. I don't know what percentage of workers will have significant eye strain problems with this unit, but I will be surprised if this is not a fairly common complaint.

MAC AND THE HACKER

The Apple][is the true offspring of the Apple I. Both are hackers' machines through and through. Their open designs permit access to the software and the hardware. Even the main buses, the key control and data circuits of the microprocessor, are accessible. The Macintosh is not a member of this family at all. It is related to machines like the Hewlett-Packard 80 and 100 series, which are very pretty packages indeed, but whose hardware is locked away from the user.

Even if you could get the special tool required to open the Macintosh case, internal access wouldn't do you any good. There are no available sockets or slots inside. Heavy modifications would be required to access the main bus. But remember, just cracking the box voids the warranty.

This locked design is fine if what you want is a powerful desktop machine to do exactly the same type tasks that tens of thousands of other people are doing. You can buy the hardware package and the software packages and you are off and running. The simplicity and speed of the Macintosh are most impressive in this context.

If you want to write a few programs of your own, the Macintosh can accommodate you, too. Basic, Pascal, and Forth packages will be available soon at reasonable prices. You may, however, have to spend considerable time learning the graphics protocols and mouse functions to make your programs consistent with the standards.

There are five jacks on the back of the case (mouse, disk, RS232, RS422, and sound) which will permit the Macintosh to interface with a wide variety of other machines. These machines (for example, a modem and the Apple][) will not be under the control of the Macintosh 68000, but will only be talking to it. This is a major change from the design philosophy of the Apple][.

The disk and RS422 ports look especially powerful. Although I am basing this opinion on limited pre-release information, I think they will certainly attract many eager designers of addon devices. Home construction versions of modems and special mice for the handicapped will undoubtedly be developed. I look forward to getting full specifications for all the ports.

The remarkable low price, $2495, has been achieved by state-of-the-art automatic manufacturing processes. A brand new plant was built to assemble only the Macintosh. This approach will keep the price down and put a great many units in the field very quickly, but it may also make changes in the hardware design extremely difficult to implement. Also, the attachments for the Macintosh might not have the advantages of such mass production techniques and so will either be relatively expensive or will not measure up to the sophistication of the Macintosh.

contd.

The Macintosh is in no way related to the Apple][. It is a beautiful, powerful, inexpensive machine, but the hardware appears to be not easily personalized. Speaking as a confirmed hacker, this is just not my machine.

THE MACINTOSH AS AN ENGINEERING WORK STATION

The Macintosh is an extremely powerful machine designed for "desk" workers. ("...anyone who handles, collects, distributes, interprets, organizes, files, comprehends, generates, duplicates, or otherwise works with information...We've narrowed it down to anyone who sits at a desk." Excerpt from the Macintosh promotional brochure.) Ok, I'm a desk worker much of the time, so how well does the Macintosh suit my needs?

In my present job I design and supervise the construction and installation of computer-based test equipment for the manufacture of photovoltaic systems. I am using an old Apple][+ for a work station, but to get it to handle all the tasks required for the job I will have to add some hardware (CP/M card, 80 column card, etc.) and purchase more software, for example, the Guidepost project manager. These items will cost several thousand dollars. Would the Macintosh be a better buy for almost the same money?

At my work station I write reports and memos, work up budgets, manage projects, reduce test data, and run tests and experiments. I would like to be able to draw illustrations for my reports at the work station, so this would be an attractive feature in any computer I evaluate for engineering work. Let's look at these functions one by one in a comparison of the Apple][and the the Macintosh in handling the same tasks.

My report writing requires a plain vanilla word processing program. I am now using ScreenWriter][, a satisfactory program in most respects. The ScreenWriter manual, however, is poorly organized, and I find it difficult to locate the answers to my questions about the program. The procedure for inserting mathematical equations is so clumsy that I usually write them in by hand. The drawings I use in my reports are usually handled as separate pages, not as insertions in the text, and can easily be generated by a separate program.

In the brief demonstration we saw, MacWriter was an impressive word processor. The response of this program was strikingly fast. But I didn't like the heavy dependence of the program on the mouse to execute commands; I am a fast typist, and constantly removing my hand from the keyboard breaks the rhythm. And I didn't see any mention of a spelling correction feature, which I desperately need. But MacWriter is definitely an improvement over ScreenWriter][, given its speed and special features.

A simple spread sheet suffices for the budget preparation I have to do. Since the procedure must be uniform throughout the company, we avoid the need for any fancy features. VisiCalc on the Apple][and Multiplan on the Macintosh can both fulfill the requirements of this task with ease.

Project planning is a problem area for me at present. My Basic language Critical Path Management (PERT/CPM) software is difficult to update and does not have graphics. The best available replacement, Guidepost, runs in CP/M. I could go ahead and purchase a Z80 card and the program, but haven't, because Guidepost is not particularly good at handling the large number of small projects I have to deal with. The Macintosh planner, MacProject, was not available for review. From its description I would expect it to be the type intended for use by executives and quite powerful. Executive-style planners, however, are not always ideal for engineering work. Even so, I will give MacProject the edge here because of its consistency with the other standard programs.

The task of general calculation and data reduction requires writing programs of moderate complexity, using the heaviest math I can lay my hands on. Besides being necessary for the engineering tasks, this work with mathematics is important to me because it separates the engineers from the administrators and electronic technicians. I now do these programs in Applesoft, which is slow and unstructured. I could add Pascal to the Apple at a reasonable cost for more logical programming and faster execution.

The Macintosh will soon have Basic, Pascal, and Forth packages, as well as a scientific calculation package. We were able to review the Basic briefly and were impressed by its speed. The heavy computation power of the 68000 will leave the Apple][in the dust in this application. For me, this is the most attractive feature of the Macintosh.

Finally, I use my work station for carrying out experiments. This involves adding an analog-to-digital (A/D) card and a clock to the Apple][and writing my own software. An example of an industrial experiment would be determining the thermal time constant of a photovoltaic cell. The experiment requires two 10-bit voltage measurments at uniform 0.1 second intervals. The Apple addons cost $1000 and do an admirable job.

The Macintosh, by contrast, has no card slots and the main bus can't be reached, so the A/D would have to be a stand-alone system with the ability to communicate with the Mac. This appears to be an expensive and clumsy application for the Mactinosh, and such a sys-system will probably not be available for some time. The Apple][wins hands down here.

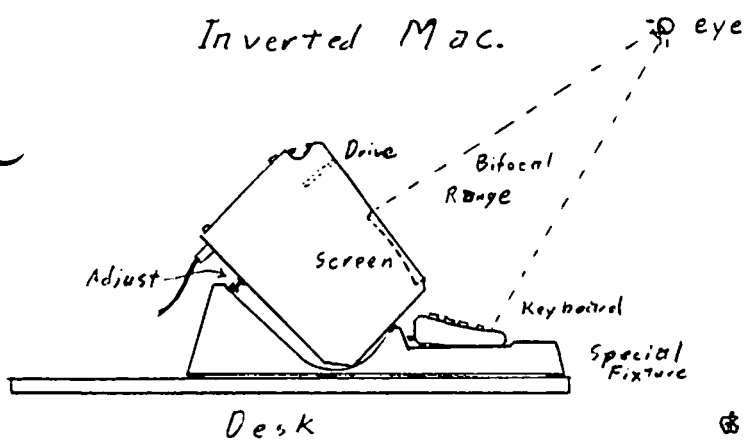
The graphic capabilities of the Apple][+ are acceptable for drawing bar charts and plotting equations. I draw the rest of my graphics for reports by using mechanical drawing techniques; I never use anything but black on white graphics since my work must be photocopied. The examples of work done with the MacPaint program appear completely inappropriate for my purposes. I have never seen the type of freehand sketch work shown in the promotional literature used as illustrations in a technical report. I don't think anyone but a trained artist could do this type of illustration well enough to risk putting a signature on it and distributing it around the office. MacPaint does let you use lettering within the graphics screen, and this is an excellent feature. In addition, the display is 512 by 342 pixels, a marked improvement over the Apple][. Overall, however, I am unconvinced that MacPaint would be a valuable or even practical part of a work station in most offices.

Although the Macintosh is a more powerful computer than the Apple][, I still favor my old machine. The two critical factors that determine this choice are the limited capability for expanding the Mac's hardware and the small screen, which is exceedingly difficult for me to read.

I hope that we will see a lot more technical data on the Macintosh this Spring and will be able to find ways around its straightjacket hardware design.

Macintosh's 32-bit microprocessor

contd.



MacSoftware contd. from pg 14

suspect that this will change by the time that the upgrade to 512K RAM memory is available.

In sum, software on the Mac is off to a promising start, although you may have to wait another six months before products become generally available.

REFERENCES:

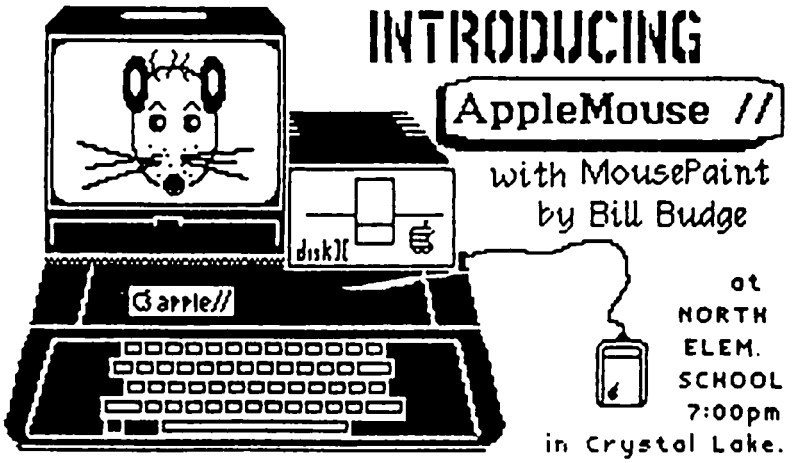
(1) Hoffman, "Lisa Pascal", USUS News, Vol. 11 p. 40 (Dec. 1983).

(2) Ahl, "Microsoft Windows", Creative Computing p. 74 (February 1984)

COURTESY OF APPLEPEOPLE

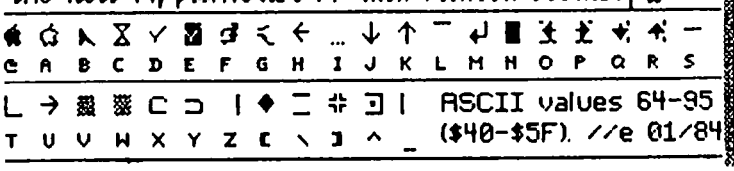
APPLE COMPUTER USER'S GROUP

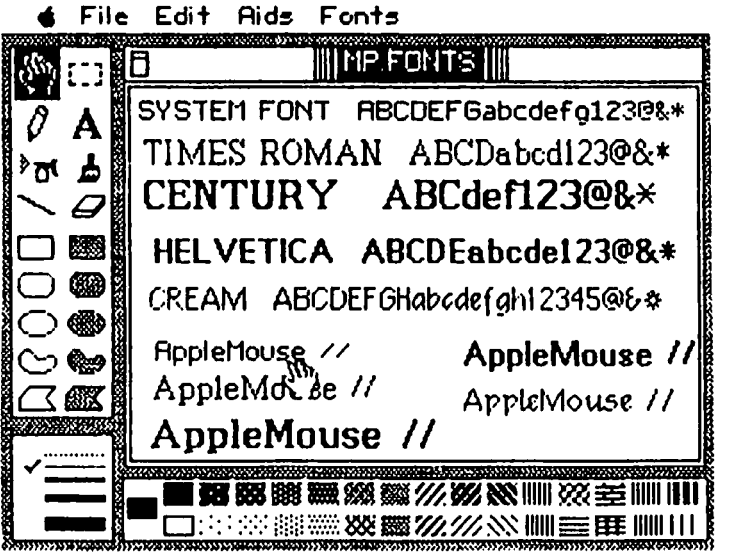
MEETING - January 16, 1984

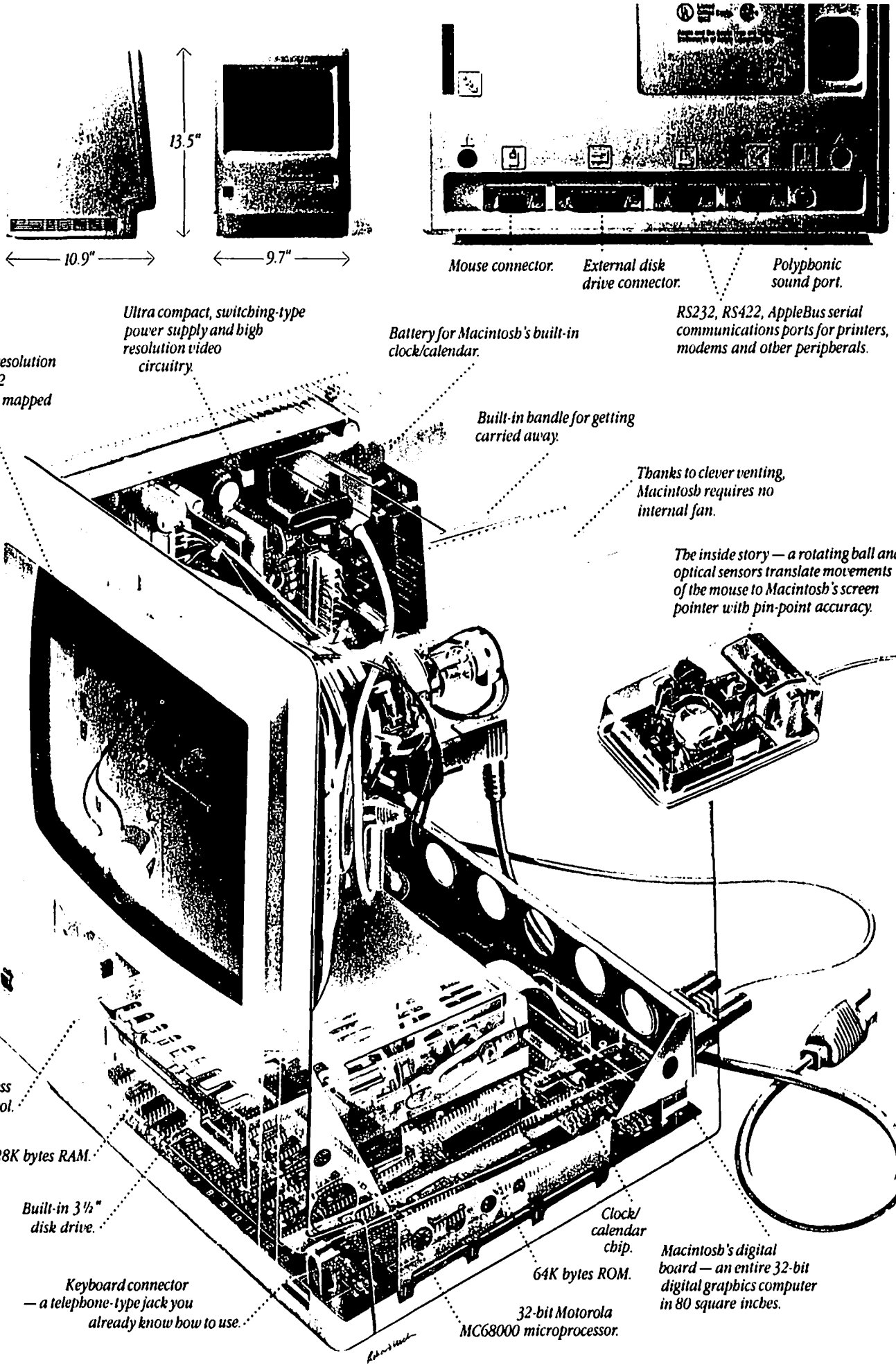


Apple announces a new ROM for use with AppleMouse //

Apple will soon begin shipping //e's with a new character generator ROM. ASCII values 64-95 will be changed to the graphic symbols shown below. These formerly INVERSE UPPERCASE values are also found at 0-31, so nothing is really lost. This change is to facilitate the new AppleMouse // and related items.






13.5"
10.9"
9.7"
Mouse connector.
External disk drive connector.
Polyphonic sound port.
RS232, RS422, AppleBus serial communications ports for printers, modems and other peripherals.
Ultra compact, switching-type power supply and high resolution video circuitry.
Battery for Macintosh's built-in clock/calendar.
9" high resolution 512 x 342 pixel bit-mapped display.
Built-in handle for getting carried away.
Thanks to clever venting, Macintosh requires no internal fan.
The inside story — a rotating ball and optical sensors translate movements of the mouse to Macintosh's screen pointer with pin-point accuracy.
Brightness control.
128K bytes RAM.
Built-in 3½" disk drive.
Keyboard connector — a telephone-type jack you already know how to use.
32-bit Motorola MC68000 microprocessor.
64K bytes ROM.
Clock/ calendar chip.
Macintosh's digital board — an entire 32-bit digital graphics computer in 80 square inches.

WHITHER APPLE?

by Bernie Urban

Admidst the hype and hoopla with which Apple precedes the introduction of the Macintosh, several things become evident to this writer. Apple is staking out an image for its corporation and line of microcomputers which stands in stark contrast to IBM and its appeal to conservative, safe thinking. Apple Computer Incorporated was the brainchild of two young men who had been affected by (and had possibly been part of) the counter culture of the sixties. Here were two guys who in their youth, among other pursuits, got their kicks building "blue boxes" which they used to break into Ma Bell's telephone system for the purpose of making long distance calls all over the world. One unsubstantiated story tells of Woz's almost successfully consumated conversation with the Pope while masquerading as Henry Kissinger.

The riches which have come from Apple's great success have mellowed their thinking and have channeled their exuberance into more acceptable social minded avenues. To whit, Apple appears to be making a statement that it's OK to be a woman architect, a programmer or other professional, usually the province of males; it's OK to be young and a free thinker. Apple through its ads is reaching out to the youth and young adults of the baby boom. Beginning with the Winter Olympics, Apple is mounting a major campaign which will take the IBM PC head on and feature youth and upbeat sounds. (Dealers were subjected to the theme song from Flashdance with changed lyrics addressed to Apple and sung by Irene Cara.)

Apple has in the recent past supported racing events in Europe and the U.S.; it features a hot air balloon replete with, Apple logo; it takes pride in the fact that one of its chief developers of the Mac Operating System is 21 years old. It is in effect courting the generation which is soon to assume the reins of business and government. Through its grant programs, Apple is reaching out to help educate the young and to strengthen the social organizations within communities. It is in the process of developing an image of youthful purpose, exuberance and of social concern. It is apparently gambling on its projection of an image of a firm which provides state of the art technology both in hardware and software.

And what of the rest of us? Has Apple deserted the million or more Apple][, //e, /// and Lisa pioneers which have made Apple Computer Inc. and the two Steves what they are? A resounding NO!

Simultaneously with the announcement of the Mac on January 24 comes announcements of interest to us "old timers":

• Starting with the Lisa. The Lisa has been rebuilt and now comes standard with a 400 KB, 3 1/2 disk drive which is identical to the Mac's. They have redesigned the face of the older Lisas to incorporate the new drive and will by early Spring install a facility to run the Macintosh software. The Lisa will be sold in three versions, starting with the Lisa 2 at $3495 which comes equipped with 1/2 MB of main memory. Lisa 2+5 at $4495 is a 2 with an external 5 MB hard disk drive. For $1000 more, Apple offers the Lisa 2/10. Here the external drive of the 2 is replaced by a built-in 10 MB hard disk drive. Other cost items include an additional 1/2 MB of memory, UNIX and MS-DOS, and the Lisa Office System. Apple has reworked the Lisa software to speed up operating system and disk performance and claims that all Lisa 2's can load and save documents 50 to 200 percent faster as a result of this. Many software improvements which have been incorporated into the Mac will undoubtedly be installed on the Lisa systems.

• The Apple /// Plus. Here we find three major changes:

1. A built-in clock/calendar which is battery operated and is probably the same device incorporated into the Mac;

2. A switchable high resolution text display mode which provides double the resolution of the usual text display; and

3. A keyboard identical to that of the //e with the exception that the old /// 13-key numeric keypad will reappear beside it. (Does this mean that sometime soon //e owners will be able to house their motherboard in an Apple case equipped with numeric keypad?)

Further, the disk file format has been changed to that of ProDOS, the new Apple //e operating system. Text disks can now be processed on either system. /// E-Z Pieces Integrated Software (Hiba Systems Inc.'s answer to Lotus 1-2-3) is a new item for the /// and is virtually identical to the Appleworks system which is available for //e owners. Apple states that they are also developing the Apple Mouse /// which should be available in the near future as an option for /// Plus owners.

• And the //e. Yes, Virginia, there is an Apple Mouse // which was developed by Woz himself after others could not get the physical componentry down to manageable size. What comes with the Mouse? Why, MousePaint, a program almost identical to that of MacPaint. And ProDOS, the new //e standard operating system provides total file compatability with the /// Sophisticated Operating System (SOS). This opens up possibilites for adding the ProFile hard disk drive to the //e. Peculiarly, Apple is introducing DuoDisk which includes two 140 KB disk drives in one case about the size of the ProFile cabinet, which they encourage //e owners to place between the computer and the monitor. Apple claims it gives "the entire system a clean space-efficient modular look". I say that it looks awkward, and should as a minimum incorporate the Sony 3 1/2 drives. Perhaps the Woz shall enlighten us about these drives and Apple's reasoning ...

So, there it is - some very significant and some not so significant announcements regarding the Mac, the Lisa, the /// Plus and the //e. Apple intends to mount a major advertising and sales campaign during the next 100 days. Apple also appears to be courting software developers by decentralizing to regional and field offices responsibility for conducting orientation briefings for systems developers on their developers tool kit for the Mac. This bodes well for Apple and the enthusiasts who will be, I am sure, closely scrutinizing this new member of the Apple line.

MicroMaster™

the MicroPower System for Applesoft Programmers

Now Applesoft Can Be Fast, Easy, and Flexible For All Applesoft Programmers

From the wizards of MicroPower comes an enhanced Applesoft with all the convenience and capability of the "big" BASICs; and an easy, natural way of adding the speed and power of machine language to the Applesoft programs you write -- without having to know machine language. The MicroMaster library contains over 50 commands -- more than twice the number offered with any other package! Applesoft enhancements include a PRINT USING that is fully integrated with all Applesoft PRINT features, and actually remembers your formats; INPUT command accepts any string (including commas), and evaluates numeric formulas, as well as accepting numeric data. Other features include fast STORE and RECALL of full arrays, SUBSTRING searches and more!

MicroMaster does it all for you! Simply choose the commands you want and let MicroMaster make them available to your program. You call them by name, and can even rename them to whatever you want. And by the way, MicroMaster also eliminates the need for different RUN, BRUN, and EXEC commands .... simply mention the name and MicroMaster is smart enough to figure out what you want.

Commands include:

ADD	APEEK	APOKE	BEEP	BLEN
BOOT	CAT	CEOL	CEOP	CLEAR
CRAE	DEL	DL	DLIST	ERRMS
FIND	FIXS	FIX	FLEN	FP
FRE	GOSUB	GOTO	GPLE	HELP
HOLD	INPUT	LOAD	MAP	MEM
MENU	MERGE	MON	MPOKE	NEW
OFF	P	PRINT	RECALL	REN
RESET	RESTORE	RETURN	RUN	SAVE
SECTORS	SSQR	STOP	STORE	SUBSTR
SWITCH	TAB	TONE	TRUN	UNDIM
W	WAIT	*	hex/decimal	

A Revolutionary Operating System with Overlay Capability For Experienced Applesoft Users

Do NOT confuse MicroMaster with programs that simply play with the '&' vector. MicroMaster is a totally unique concept -- a co-resident operating system and dynamic subroutine library. MicroMaster allows you to create custom systems...easily. With MicroMaster, jobs that would be a real challenge to experienced machine language programmers can be easily done by a few simple commands, without machine language!

No more CHAINing for large programs. Only MicroMaster has full overlay capability -- just like large main-frame computers! Swap subroutines in and out and modify your operating system while your program is executing! Swap entire operating systems in and out while your program is executing -- just try that one, BIG BLUE! And the overlaying is done with simple LOAD and DELete commands ... actually much EASIER than CHAINing.

MicroMaster allows you to structure Applesoft programs using external subroutine libraries. You'll find that structuring programs into subroutines and modules makes programming easier and faster. In addition, you'll find that you're able to "borrow" subroutines from other programs more often, which will make programming even faster. MicroMaster also has a full error handling and trace-back capability -- just like on the large computers.

Those who write in machine language will find MicroMaster's ability to automatically relocate binary programs an added bonus! Simply write your program and tell MicroMaster to load it. Machine language programs do NOT have to be written in relocatable code (What a terrible restriction!), and relocation dictionaries are not used.

MicroMaster will make a difference in the speed and power of your programs, as well as make them easier to write. MicroMaster is available now for only $75 (plus $3 shipping & handling, Maryland residents add 5% sales tax). We even include a medium speed DOS and 2 MENU generating programs for FREE! Consistent with the MicroPower policy, the MicroMaster system disk has no annoying copy protection schemes. All files are fully copyable and listable.

Call today to order: (301) 495 - 2933
MasterCard & VISA accepted

MicroPower, Ltd. ◆ 1100 East-West Highway ◆ Silver Spring, Maryland 20910

MicroMaster and MicroPower are trademarks of MicroPower, Ltd.
Apple and Applesoft are registered trademarks of Apple Computer, Inc.

A 68000 BIT CO-PROCESSOR FOR YOUR APPLE

by Peter E. Rosden

Many club members have reached the point where either their faithful Apple][‘s, although reliable machines, are just too slow for the complicated, time-consuming and repetitive operations they wish their computers to perform, or they are anxious to expand their horizons and learn to use more powerful computers. Unfortunately, the rumor mill has repeatedly misinformed us that better, faster, "inexpensive" (viz. LISA ha-ha) machines from the "next generation" will be available "next month" or "in the fall". I fit both categories. After months of waiting, groaning, checking out possibilities and looking at new computers, I finally decided (about one year ago) that the best way to achieve an interim quantum leap in speed and stay acquainted with the newest technology was to move up to a 16 or 32 bit machine.

Before discussing my board and the chip specifically, some readers might like to know in general terms what a 16 bit microprocessor (more specifically the Motorola 68000) would do for them compared to an 8 bit microprocessor. The 6502 microprocessor is an 8 bit microprocessor. This means that its data bus (that's what carries data back and forth from memory) has 8 parallel switchable lines which can be in an "on" or "off" state to represent information. As a result, the data bus for this processor can be used to serially transmit 256 (2 raised to the 8th power) different single pieces of information, such as punctuation, letters and numbers. What happens is that each piece of information is translated by the computer after you type it on your keyboard into a number using a generally agreed upon standard which, for the Apple, is the ASCII character set where, for example, the letter "A" equals 193, the letter "a" equals 225 and a "?" equals 191. The same holds true for the internal registers of the 6502 processor (the X,Y and ACC registers discussed so much by assembly language programmers) which are used to manipulate data and perform operations like adding, subtracting and comparing the pieces of information. Most of these 256 possible pieces of information are taken up by numbers representing upper and lower case letters, digits from 0 to 9, punctuation marks, and special control codes. In order to move just the letter "a" from memory into the processor, all of the space in the 6502 data bus must be used to represent the number "225". That's why only one letter can be moved at a time. There are basically two ways (without getting truly sophisticated and erudite) to move more than one piece of information at a time with an 8 bit data bus.

First, you could change the ASCII character code to eliminate a lot of the 256 pieces of information so that, for instance, all of the pieces of information you were interested in could be represented in only 5 bits of every byte, giving you 32 (2 raised to the 5th power) number of pieces of info with 3 free bits left over. This is called packing and is done in some systems such as IBM's. Obviously, you give up a lot of flexibility this way.

The second way is to enlarge the size of your data bus and your internal registers to 16 or 32 or even 64 bits so they can carry more information every time they perform an operation like fetching data from memory via the data bus or comparing a string of information in memory. This is the approach taken by the 68000 processor which has a 16 bit data bus and 32 bit internal registers. By comparison, the big mainframes generally have 64 bit data bus and internal registers. That's one reason why the microcomputer industry is making such inroads into mainframe sales territory. The result of the wider data bus and internal registers is to make it possible to directly address 65,536 bytes (2 raised to the 16th power) of memory. In the same way you can access only 256 bytes of memory with the 6502 using "zero page addressing". That rids you of the frustrating problem of tracking and counting all of the information in memory in blocks of 256 bytes. Of final enormous interest is the inclusion in the Motorola 68000 processor of 8 separate data registers which are each 32 bits wide. This enables each one to hold 4 bytes of information and allows the performance of operations such as comparisons on all 4 bytes simultaneously. It also enables the holding of numbers which are repetitively needed throughout execution of a program. Also included are 7 separate 32 bit wide address registers which are very convenient to use for holding the starting addresses of frequently accessed blocks of memory, thereby eliminating the need to load those addresses from memory locations each time they are needed.

Hopefully, this simplistic narrative will help you understand what a 16 bit processor will do for you: faster access to a far greater range of memory; retrieval of a greater amount of data in less time; and a truly astounding number of data and address registers made available to access various parts of memory and manipulate larger amounts of data with greater speed. As a further comparison, while the 6502 has an 8 bit data bus and 8 bit internal registers, the Intel 8088 used by the IBM PC has an 8 bit data bus (so much for all those ads about a 16 bit machine!) and 16 bit internal registers. A new 68020 from Motorola has a 32 bit data bus and 32 bit internal registers. (P.S. all software developed for the 68000 is totally transportable to the new 68020). Moreover, while the 6502 runs at a once respectable clockspeed of 1 Megahertz, the 8088 can do about 5 MHz and the Motorola 68000 is available at 12.5 MHz in the newest board discussed below. A 16 MHz version is coming. The difference in processing speed is absolutely incredible. One graphics demo that came with the board I bought showed a program that took 1 hour in Applesoft to plot and took 10 seconds in 68000 assembly language.

Another nice feature of the 68000 is that it is very easy to program in assembly language compared to the 6502, and a whole lot easier than the Intel 8088 or 8086. Enough of abstractions, let's get back to dollars, software and service.

My investigations at the time convinced me that 32 bit machines were out of my league. The only one available was from Hewlitt Packard at a mere $30,000 for a CPU and some software. However, 16 bit microprocessors did exist and were being used. The question was whether to buy an entirely new computer or to seek an add-on board for my Apple. The only feasible choice in the first instance was the Pascal-based Sage IV at some $3500, little software and a relatively slow 68000 processor implementation. If I followed the second course I could still await a fully operable 68000 system, yet be able to develop software in the

contd.

interim period. After further searching, I concluded that the best choice on the market at minimum cost was a Dtack Grounded board from Digital Acoustics, a company located in Santa Ana, Calif.

Alas, a few words of warning are in order - this company is annoyingly uncommunicative about their product. Some of you may have seen the WAP letter to the editor's column some months ago which included a short epistle written to me by this company in response to a letter from me requesting detailed information on their board, a request which I still consider reasonable since the board requires an investment of between $600 and $1100, depending on the memory configuration chosen. Basically, they informed me that if I wasn't interested in subscribing to their newsletter to obtain technical information, I could get lost. Needless to say, I was furious and continued my quest for other 68000 or 16 bit machines of the Intel (80286 or 8086) or National Semiconductor (16032) variety. All to no avail. In desperation, I finally subscribed to Digital Acoustics' newsletter called "Dtack Grounded, The Journal of Simple 68000 Systems."

Surprise!! - the newsletter was extremely informative, (biased, to put it mildly, in favor of the Motorola 68000 and Digital Acoustics' product). This newsletter is a fantastic way to keep up with the leading edge of practical 16 bit and beyond microcomputer developments. A year's subscription now runs $15 in the U.S. for about 6 issues (publication, in keeping with the offbeat personality of the company, is a little irregular). The company is located at 1415 E. McFadden, Suite F, Santa Ana, CA 92705. I strongly recommend that anyone interested in 16 bit processors, whether or not limited to Motorola, purchase back issues commencing with Number 1 (there are now approximately 27 issues). In so doing, you will be able to trace the developments that have occurred in this area and will be current on the Who's Who of the computer world, complete with inside rumors, backstabbings, tongue-in-cheek humor and some incredible insolence.

On to the board itself - More Warning! - if you purchase this board, at least for now, you must be willing to learn 68000 assembler. There is no high level language (Basic, Pascal, etc.) yet available for it, although one software company (Phase Zero in Phoenix, Arix.) is developing a Basic for it that will run about 100 times faster than Applesoft Basic. Be aware that Phase Zero is almost as uncommunicative as DTACK, but, so far, I have found their products satisfactory. Also if you are interested in 68000 assembler, you should purchase 2 books: "68000 Assembly Language Programming" by Gerry Kane, Osborne/McGraw-Hill 1981 ($17.00) and "The 68000: Principles and Programming" by Leo J. Scanlon, Blacksburg Continuing Education Series 1981 ($15.00). Phase Zero also markets for $40 an assembler which works with DOS Toolkit to provide functioning 68000 code. I have the assembler and it works very well. The DTACK newsletter will give you further detailed information. Another problem you will encounter with the Board is its inability to provide a simple usable interface with the Apple for on-screen display when frequently required, at least as far as I am concerned. This is due to the absence of an operating system that can make the 68000 truly the main processor and turn the 6502 into just a device for handling input and output. Phase Zero is now (as of Nov. 1983) marketing a miniassembler which I purchased that should help eliminate the problem, but, frankly, I haven't had time to work with it yet.

What else is new at DTACK? - you can purchase presently a board (basic price $795 for 128K of RAM) with up to 1 megabyte of RAM ($1995), and a graphics board is in the works. They are also about to start marketing a mathematics "co-processor" for the 68000 based on the Natl. Semiconductor 16081 (about $350) which should run significantly faster than the Intel 8086/8087 variety.

What other solutions are now available to the 68000 machine problem? - Apple is coming out with the McIntosh in January, or at least there will be a demo of it in DC on January 28. About the Sage, talk to Tom Woteki. He has one. I just saw a demonstration of the Fujitsu Micro 16S. It is now an Intel 8086-based machine which I wouldn't touch, but the company already has priced a 68000 based 12.5 MHz version available in about 6 months (have you heard that before?) at about $4300 retail, including RGB monitor, two 320K disk drives and a bundle of software. The problem with earlier marketing is, of course, lack of software. Even old-fashioned, slow to change, expensive IBM is finally coming up with a 68000-based machine - it's rumored to be a version of the IBM/XT 370 which is supposed to give you a desktop mainframe capability by running parallel 68000's and an 8087. I have tried to get more info on this to pass on, but I just get shifted from one "marketing rep" (salesman) to another.

Well, that's about all I have to report for now. If anyone else has or gets a DTACK board, get in touch and we can help each other work out the bugs!

1001 BINARY TALES or Getting Down to the Level of the Apple

by Raymond Hobbs

This to-be-seen-from-time-to-time column is aimed at the aspiring Assembly Language programmer, and will contain tidbits for the novice and intermediate level programmer (you experts can fend for yourselves - you've gotten along fine this far, haven't you?). If you have a problem with some aspect of Assembly Language programming, just drop a line to me at the office, and I'll try to handle it.

This month's installment starts right at the beginning. One might call it "The Care and Feeding of Assembly Language Programmers" (to keep alive the mystique of some dark, symbiotic relationship between the AL programmer and the computer), but I think that it's high time we put that myth to rest and showed up Assembly Language programming for the straightforward (but not trivial) exercise that it is. So I choose to call this article "Getting Down to the Level of the Apple".

The 6502 microprocessor (the chip that drives the Apple) is capable of performing less than 60 operations, which fall into about 10 categories (move memory, add, subtract, compare bits, shift bits, change bits, set bit, clear bit, skip forward/backward in the program, do nothing and stop). These turn out to be rather rudimentary operations - for example, an IF....THEN type of operation is way beyond the capability of a single-instruction operation for the 6502, and must be simulated by an entire subroutine. This is, in fact, what the Applesoft interpreter does when it encounters an IF....THEN - it calls upon a machine language subroutine. The point is, the CPU (the computer chip) operates on a rudimentary level, while we humans like to operate on a more complex level. This is because we can keep track of several things simultaneously, but the computer can only keep track of one thing at a time (it can't walk and chew gum at the same time). Running the risk of belaboring the point, let's look at a concrete example of human-type programming and computer-type programming:

HUMAN-TYPE PROGRAM	TRANSLATION
10 FOR X=1 TO 10	Do the following operation 10 times
20 A=A+B	Add value B to whatever value A is at the moment
30 NEXT	continue until finished

COMPUTER-TYPE PROGRAM		TRANSLATION
LOOP	LDX #$00	Set counter to zero
	LDA A	Move value A into the arithmetic area
	ADC B	Add value B to it
	STA A	Save the result as value A
	INX	Add one to the counter
	CPX #$0A	Is the counter ten yet?
	BNE LOOP	If it isn't, go back and repeat.

Three things about computer-type programs become apparent from the above comparison:

1. The computer has to be reminded what counter it is using;

2. The arithmetic operation has to be explained in detail to the computer;

3. The program is twice as long (because the instructions are much less complex).

It now becomes plain that Assembly Language programming is really just a matter of "getting down to the level" of the computer (we might imagine a parent-toddler relationship). This is, in fact, where the terms "high-level" and "low-level" languages come from, and now we will take a closer look at these levels.

Level one, the most rudimentary, is that which the computer understands. It is a binary code, and while perfectly understandable to the computer, it is rather unpalatable to humans, even when represented in hexadecimal format. We use it on occasion, when program routines or data must be passed directly into the innards of the Apple from the monitor, for example. Binary programs were the original method of software programming (in the early days, the only method available), and are termed "first generation" language programs.

Level two represents a step up the language ladder, and is a mnemonic representation of binary programming. The translation is performed on a one-to-one basis; that is, one mnemonic code is translated to one binary byte. Since the computer is unable to understand the mnemonic code, the program is assembled from mnemonic (source) code to binary (object) code. The binary code is usually shown in its hexadecimal representation. Thus the term "assembly language", referring to second-generation languages. The particular assembly language we will be primarily concerned with (but not limited to) in upcoming articles is the 6502 Assembly Language.

Level three refers to those third-generation languages which most of us are familiar with, such as BASIC, FORTRAN, PASCAL and so forth. These languages are much more readable (to humans) and the bulk of all programming efforts employ third-generation languages. Such languages fall into sub-categories dependent upon the manner in which the code reaches the computer for action. The familiar APPLESOFT BASIC, for example, is an INTERPRETED language; that is, each instruction is examined by the interpreter (located in ROM), parsed for grammatical correctness and then executed (by a first-generation subroutine in ROM or in DOS) and then forgotten. When an interpreted language program loops back, this process is repeated, of course, which is one of the reasons that BASIC programs execute so slowly on the Apple. By the way, you may have noticed that occasionally, when you are testing one of your BASIC programs, the program will run for a considerable length of time, then crash because of a syntax error. You did not find out that there was an error before trying to run the program because the computer did not examine the code until it tried to interpret it during execution!

contd.

Languages such as PASCAL are called COMPILED languages. Compiling is a process, similar to assembly, through which a source code (PASCAL, for example) is translated into a (more or less) machine-readable code (called P-CODE, in the case of PASCAL) and placed into an executable file, called an object file. All the grammatical checking is done during the compiling process and therefore no syntactical errors can occur at run time. Although the compilation process is analagous to assembly, the translation is not necessarily done on a one-to-one basis. Therefore, one PASCAL instruction (for example) can generate an entire machine-language subroutine.

You might think that it would be easier to bypass an assembly language in favor of a compiled language, to take advantage of the ability to generate many machine-language instructions for just one instruction written in the higher level language. However, with this ability comes some amount of generalization in the process of creating a program. Assembly language, being at virtually the lowest level of interface between the programmer and the computer, allows the code to be tailored to the problem in a very precise fashion. Thus, coding in Assembly Language will normally produce a faster-running, more compact program. (Ed. Note: This is why Apple programmers have spent so many person years of effort to create the operating system for the Mac in assembly language.)

The next level up (fourth generation) comes with program generators. These may be loosely defined as programs which take data plus English-type commands as input and generate executable program modules as output. Although (to the best of my knowledge) no true fourth-generation languages exist for the Apple, there are some products that come pretty close. LISP-like languages could be considered proto-fourth generation, except that the input and output must be within the defined boundaries of the language itself (similar to the FORTH "dictionary"). Products like Dbase II take data and command input and create executable output, but within very restricted parameters.

The computer industry has been moving toward the next level (fifth generation), which would entail self-generating programs, or a true learning process for computers. If that capability comes about for micros in my lifetime, I will send my Apple out to get a job and buy me a LISA. But then, a self-teaching computer would probably join a union. Oh, well.

To get back to level two, you probably see now that Assembly Language programming is really just a matter of dropping down a level to talk more directly to the computer, and reminding the 6502 to keep track of all the details.

In upcoming articles, we will be returning to 6502 Assembly Language, getting down to the level of the computer. The Assembly Language SIG has been at work putting together reviews of assemblers themselves. We have also been developing some useful applications tools. We invite all of you to attend the novice or the intermediate tutorials, which are held on alternate Wednesday evenings at the WAP office. Our SIG meetings are in the cafeteria, following the main meeting. For details, call the ASMSIG Chairman. If you have a question about Assembly Language, drop by a meeting, or write a card to the ASMSIG, and we will tackle your question in this column.

MODULA 2: Part II

by Robert C. Platt

Last month we discussed Volition System's Modula 2 compiler and worked through a very simple program. This month I will begin to present a flash-card drill program. Because this program is a more complex example, it is divided into two separate procedures.

MODULES VS. PROCS

Many programming languages allow you to divide your program into separate building blocks. In this manner, a series of instructions need only appear once, but can be incorporated at a variety of different places in your program. Most languages call these building blocks "procedures" or "subroutines." However, Modula 2 features two different types of building blocks: procedures and modules. This is the principle difference between Modula 2 and its predecessor language, Pascal.

In Modula 2, MODULES are like books in a library, and PROCEDURES are like chapters in those books. Most modules are made up of more than one procedure, although a module need not have any procedures in it. Procedures aid in displaying the inherent structure of a program and in breaking down the task of writing programs into manageable pieces. Using procedures can eliminate duplication of code and make debugging programs easier.

Our flash card drill program needs a procedure to display messages on the screen. Look at the following example:

```
PROCEDURE show(x:msg; xu:INTEGER);

VAR j:INTEGER;

BEGIN
  IF xu=0 THEN
    IF A#EA THEN
      WriteString(output,'NO, TRY AGAIN') END
    ELSE
      FOR j:=1 to xu do
        WriteString(output,x[j]);
        WriteLn(output) END (* for *)
    END (* ELSE *)
END show;
```

The first line is called the "procedure heading." It consists of the word PROCEDURE followed by the name of the procedure and a list of parameter values which will be given to the procedure to act upon. In this case, the procedure is named "show" and will be given a message and an integer number as parameters. Everything between the word PROCEDURE and the END statement containing the procedure's name are a part of the definition of the procedure.

VARIABLE NAMES

Modula 2 allows a programer to give names to pieces of data that a program acts upon. Because the value stored by the computer as data can change during the execution of a program, these items are called "variables." Wirth calls the names which a programmer invents for his variables, procedures and modules "identifiers." Any sequence of characters which begin with a letter can be used as an identifier. However, you must be consistent with the capitalization of your variable names, and there are 40 keywords (such as BEGIN and END) which have a special meaning and cannot be used as identifiers.

All variables in Modula 2 have certain characteristics which indicate whether they contain numeric data or strings of text, or even combinations of both. Hence we say that every variable has a TYPE. For example, we see that the variable "xu" is of type INTEGER, which means that only whole numbers will be stored there. Modula 2 allows you to create your own data types. In this example, I have invented a new TYPE called "msg", and the variable "x" has this data type. In our example, variables x and xu are taken from the program which calls "show." (That's why x and xu are parameters listed as part of the procedure heading.) However, show also has its own secret variable named "j" which is of type INTEGER.

BRANCHING

The purpose of our sample procedure is to take a message and display it on the screen. A message consists of several lines, with each line stored in a separate string. Variable x has the message and xu has the number of lines. If there are no lines in the message, procedure show will display "NO, TRY AGAIN." in those cases where the student has entered a wrong answer.

The IF statement in Modula 2 is designed to branch the program to different sequences of steps depending upon whether a given test is true or false. (It is like the IF statement in Pascal and BASIC, except that its syntax is easier to use.) In the statement:

```
IF A#EA THEN WriteString(output,'NO, TRY AGAIN') END
```

the computer will check if the value stored in the variable A (which has been defined somewhere outside of procedure show) equals the value in variable EA. The symbol # means "not equal" in Modula 2. If the values are not the same, then all statements between the words "THEN" and "END" will be executed. If the values are equal, then this part of the program will be skipped. The IF ... THEN ... END sequence is called a one-way branch because any statements following the word END will be obeyed regardless of the truth of the condition following the word IF. Modula 2 also features a two-way branch in the form of IF (a condition) THEN (statement sequence if the condition is true) ELSE (statement sequence if the condition is false) END.

Here, if variable xu has zero stored as its value, then one path will be taken, otherwise, the path will be skipped and the message will be displayed.

Other IF statements can be nested within the branching path of a given IF statement. For example, only if there is no message (xu=0) and if the student types a wrong answer (A not = EA), do we want to display "NO, TRY AGAIN."

FOR LOOPS

As with BASIC and Pascal, Modula 2 has a FOR statement to repeat a given sequence of statements a specified number of times. Since we have xu number of lines in our message, we want to repeat a sequence of steps until all of the lines in the message are displayed. The statement, "FOR j:=1 to xu do" causes all following statements to be repeated until the following END

contd.

statement. These statements will first be executed with j set to 1, then to 2, and so on until the last time j will have the value that is stored in xu. In Modula 2, the variable that is changed by the FOR statement (called the "loop index") can be changed in ways other than by just adding one each time. For example, "FOR j:=1 TO 11 BY 2" will count all the odd numbers between 1 and 11. "FOR j:=10 TO 1 BY -1" will count backwards from 10.

Two statements are repeated with our loop. The first displays the next line of the message on the screen, and the second places a carriage return on the screen so that the next line will appear below the previous ones. Unlike Pascal where END statements correspond only to BEGIN statements, in Modula 2 a number of different commands require the use of the keyword END as a part of their syntax. The word END is a required part of the FOR statement to mark the end of a loop. Here, the word END following "WriteLn(output)" matches with the FOR statement and marks the end of our loop. (Remember "(* FOR *)" is a comment that helps us see which statement the END belongs to.)

The next END statement belongs to the ELSE and marks the end of the branch which is executed only if xu does not equal 0. Finally, the third END statement marks the end of procedure show. The Modula 2 compiler matches each END statement with the closest statement that requires an END, but which has not yet been matched up. The indentation of the program's lines helps you see this structure.

Procedure show is just a part of a 120 line long program. Next month you will see show in the context of the complete drill program.

IMPLEMENTING AN EXPERT PROGRAM

by Frederick E. Naef

For several months, I have been working with the language LISP, and believe that my experiences may be of interest to other members of Washington Apple Pi. In the process, I have succeeded in implementing an expert program that can be used for a variety of applications.

WHAT IS LISP?

LISP (LISt Processing) was invented by John McCarthy, and has become the best-known of the artificial intelligence languages. It is used extensively in the field of artificial intelligence because it is easy to learn, precise, and unambiguous. It is an interpreted language that is designed to manipulate symbols, and is therefore an excellent developmental language for the type of symbol-manipulation operations that are so prevalent in AI programs.

WHAT IS P-LISP?

P-LISP is a LISP interpreter written for the 6502 microprocessor used in the Apple II. It is supplied by GNOSIS, a Division of Pegasys Systems, Inc. of Philadelphia. I have been using version 3.0, but later versions are available, and a new release will be available soon for the Apple //e. The program runs well and the documentation, if augmented with additional references, is sufficient to permit the user to master the language and operation.

The program requires a disk drive and at least 48K of RAM. Since LISP consumes large quantities of memory for internal operations, the program periodically initiates a garbage collection routine to recover available memory. This routine takes time, and therefore the more memory you have installed, the faster the program will run, because less garbage collection is required.

The best way to learn is to type in the functions and other lists provided by one of the references, and then debug until it runs. The P-LISP system has excellent trace and debugging facilities, and is very resilient and user tolerant.

WHAT IS AN EXPERT PROGRAM?

Within the field of Artificial Intelligence (AI, which might be described better as automated reasoning), there is a subdivision known as expert programs. While the objective of AI is to develop software that emulates the cognitive or thinking processes of the human mind, an expert program extends this by employing several attribute databases, and a set of "if-then" rules, to emulate the diagnostic and deductive reasoning performed by a human expert when solving a problem. The power of an expert program is that it can combine the expertise of many experts, and can condense that wisdom into a single, enduring piece of software.

Expert programs have been written to perform medical diagnoses, to configure equipment installations, and to interpret mass spectograph measurements, among other applications. While the technique works best on a rules-oriented problem it can be used on unstructured problems as a decision making assistant to a human operator. In this case, the expert provides "what if" options, which permit the human operator to make decisions from a more enlightened perspective.

The extraction of the expertise from the human experts, and the writing of the "if-then" rules, has led to a new profession known as knowledge engineering. The knowledge engineer fills a crucial role in developing an expert program because he or she must be able to coax the sometimes reluctant expert to reveal the special secrets of the target skill, and then to convert that information into a set of rules that replicate the process.

THE ANIMAL EXPERT TOY

An excellent text on the language is LISP, by Winston and Horn (Ref. 1), in which Chapter 18 is devoted to expert systems. The book presents most of the functions required for a working expert program, and provides some of the databases and rules for a toy program that deduces which animal the user has in mind. Once the deduction is complete, the program can explain how it came to its conclusions, and why it needed certain data.

There are two modes of operation. The program can be made to deduce solutions by forward chaining, in which case it must be provided in advance with a set of attributes that describe the target animal. On the other hand, the program can diagnose the problem by backward chaining, in which it engages the operator in an interactive dialogue as it verifies relevant facts about the animal.

Unfortunately, the functions in Ref. 1 are written in a dialect called MACLISP, which runs on a DEC PDP10 computer. This dialect must be converted to P-LISP in order to run on the Apple II. An additional problem occurs because the book has several omissions and errors which must be corrected.

SPECIAL FUNCTIONS REQUIRED

One of the strengths of LISP is the ability to define functions that are not offered by the interpreter. In the case of P-LISP, the number of inherent functions is kept small in order to reduce the size of the interpreter, and thus to make more memory available for faster processing operations. To implement the animals expert, it was necessary to write functions for MEMBER, CADAR, CADR, CADDR, CADDDR, and CDADDR. Suitable functions are described in Ref. 1, or can be provided by the author.

The book is not clear about how one passes the attributes of the target animal to the expert program. A typical list should be typed as follows:

```
(SETQ PENGUIN '((ANIMAL HAS FEATHERS) (ANIMAL LAYS
EGGS) (ANIMAL DOES NOT FLY) (ANIMAL SWIMS) (ANIMAL
IS BLACK AND WHITE)))
```

OTHER MODIFICATIONS

The creation of the rules and attributes databases requires an editor capable of producing text files that can be EXECed into the expert program. I used the editor from the AEPro telecommunications program, but any word processor that produces text files could be used. A typical rule should be typed as follows:

```
(SETQ RULES '(((RULE ONE (IF (ANIMAL HAS HAIR))
(THEN ANIMAL IS MAMMAL))) (RULE TWO ... etc.etc.
....))))
```

contd.

There are several errors and omissions in the material provided in Ref. 1. The first is the absence of the function TRYRULE+, which should be typed in as follows:

```
(DEFINE(TRYRULE+(LAMBDA(RULE)
   (AND(TESTIF+ RULE)
       (USETHEN RULE)))))
```

Another error occurs in the function STEPFORWARD which must be corrected as follows:

```
(DEFINE(STEPFORWARD(LAMBDA()
     (PROG(RULELIST)
          (SETQ RULELIST RULES)
          LOOP
          (COND((NULL RULELIST)(RETURN NIL))
               ((TRYRULE(CAR RULELIST))
               (RETURN T)))
          (SETQ RULELIST(CDR RULELIST))
          (GO LOOP)))))
```

WHAT NEXT?

Once the expert functions are installed and debugged using the animals databases, one can develop a variety of other applications that are unique to each user's interest. At this time, I am developing the rules, and creating an attributes database on the members of the United States Senate, in order to have an expert rpogram that will deduce and explain how the Senate will vote on a piece of legislation. This assumes that the Senators vote according to their historical attributes, and that the user correctly defines the attributes of the legislation.

While most expert programs are being developed to run on large computers, or dedicated LISP machines, there is a challenge in developing meaningful expert programs to run on personal computers with limited memory. Branching or nested experts may solve that problem - in which groups of rules and/or portions of databases are selectively moved into RAM where they can be worked sequentially to solve a problem that is too large to handle in a single pass. The program will not be fast, but the results can be equivalent to those obtained from large systems.

CONCLUSION

P-LISP on the Apple II is not as powerful as other dialects running on dedicated LISP machines, but it does provide a good opportunity to learn LISP, and to learn about expert programs. In addition, small but useful programs can be developed to run on the Apple II.

REFERENCES

1) Winston, Patrick H. and Horn, Berthold K.P., LISP, Addison-Wesley Publishing Company, 1981.

2) Cherry, Steven, P-LISP User's Manual, Gnosis/ Pegasys Systems, 1982.

3) Tracton, Ken, Programmer's Guide to LISP, Tab Books Inc., 1980.

FEEDING AT THE TROUGH: News From the PIG

by Michael Hartman

This month I again have some news from the world of Pascal. Sources include Pascal Interest Group (PIG) meetings, the national USUS meeting, the MUSUS bulletin board, and loose talk on the street. (I'm still waiting for my first hot tip by phone.) Topics range from happenings within USUS to the wonders of Apple /// Pascal.

USUS National Meeting and Member Services

The UCSD p-System User's Society (USUS) held its semi-annual meeting in Washington on October 14-16, 1983. Approximately 200 people attended, including several PIG members, and a fun and informative time was had by all. There were several first-rate presentations and panel discussions. Tom DeMarco talked about data-flow programming and the potential of Modula-2; Stephen Pickett explained how he sped up NCI's p-machine interpreter (now I know why I despise assembly language programming--I've seldom seen anything more devious); Joel McCormack presented an introduction to Modula-2; and Bob Peterson offered a tutorial on advanced topics in UCSD Pascal and the p-System. The panel discussions focused on two often opposing aspects of computer software: marketing issues and technical questions. The vendor panel included representatives of Apple, Sage Computer, SofTech Microsystems, NCI, and Volition Systems. The most striking thing about this session was how satisfied the Apple Pascal users are in comparison with users of UCSD version IV. However, SofTech appears to be making a considerable effort to clean up their act, and all p-System users can only wish them well. The closing session was the Expert Q & A, and it featured some of the people who developed the p-System at UCSD. (Did you know that the Filer keeps track of wild-card transfers by setting marker bits in the source disk's directory? I trust that you do care!)

Another big development at the USUS meeting was the election of PIG's own Jim Harvison as USUS secretary. Jim succeeds retiring PIG-founder Tom Woteki, who is now busy with his start-up p-System software firm. Jim inherits the secretary's TI 990 minicomputer, which is the size of a refrigerator and will serve to keep his den warm this winter. We wish both Jim and Tom the best in their new endeavors. Also in the close-to-home department, the first USUS Local User's Group has been founded by the USUS members of Washington and Baltimore. This group includes users of Apple, Sage, IBM, and various other computers. The Washington/Baltimore LUG meets on the second Wednesday of odd-numbered months, and all p-System users are welcome. For further information, contact Bill Hershey or me (on the WAP BBS or at the phone numbers listed in the masthead).

For those of you who joined USUS in the past year and had been wondering where the magazine was, you should have received issues 9/10 (combined) and 11 by now. Various problems, which have been corrected, delayed the printing and mailing of 9/10 by about six months. Related difficulties also led to the loss of some names from the USUS membership roster. If you are a USUS member and have not received these issues of the magazine, send a letter to USUS including some proof of membership (probably a photocopy of your cancelled check). You will be restored to the roster and mailed the back issues you are missing. To further insure that nothing of this nature happens again, the new office of vice-president for member services was created, and Eli Willner is its first occupant. Eli, Jim Harvison, and the other officers are working hard to make USUS more reliable and responsive. PIG now has immediate access to another USUS membership benefit, its program library. This consists of over twenty volumes (two Apple disks per volume) of UCSD Pascal software in source form. Come to a PIG meeting for further details.

More on Apple Pascal 1.2

Last time I discussed some of the new features of Apple Pascal 1.2 for the Apple][family, scheduled for release early this year. At the October USUS meeting, Chris Jewell of Apple Computer provided a great deal of additional information. Chris is marketing director for Apple][and /// Pascal, and also an expert in the technical area. He put to rest one possible worry by stating that there would be complete 1.1 object code compatibility. Also, one of my pet peeves will be taken care of--UnitStatus for the keyboard (Console: and Systerm:) will check for a keypress instead of just looking in the type-ahead buffer. And there will be BIOS support for the "apple" keys on the //e. Those with a 128K //e will also get some goodies seen previously in Apple /// Pascal. The number of segments permitted will be increased from 32 to 64 (especially useful for an expanded SYSTEM.LIBRARY on a high-capacity disk). And the compiler will support "library" files, where you can store a collection of Regular Units related to a specific application. The bad news about 1.2 is that while all your "foreign" device drivers will still work, the current version of SYSTEM.ATTACH is not compatible with 1.2. The new version of ATTACH will probably not be available until one or two months after 1.2 is released. (This will not affect most Pascal users, but those of us with big RAMdisks will be leaving 1.2 on the shelf until then.)

For those contemplating a purchase of Apple Pascal and now wondering whether to wait for 1.2, my advice is to go ahead and buy now. When Apple released version 1.1, it provided free upgrades to everyone who had purchased Apple Pascal in the previous four months. And considering their generous policy of free Revision-B motherboard upgrades for the //e, I'd expect them to do no less this time. (But please note that I have no information from Apple on whether or not they will do so.) And if 1.2 is delayed, you can get an awful lot of use out of Pascal in the interim. For current owners of Apple Pascal 1.1, Apple will be providing upgrades for about $50.

SofTech Announces UCSD IV.13 for the Apple][

Also at the October USUS meeting, SofTech Microsystems announced the release of version IV.13 of the UCSD p-System for the Apple][. This version includes several bug fixes, assorted enhancements, and revised and reorganized documentation. Most importantly, support for the extended memory of the Apple //e is provided. This will speed up the system somewhat and may make the use of version IV feasible on the Apple. One of the things I was most surprised to learn in October was that on a 64K IV.x p-System, the code pool sits between the stack and the heap. This is done to permit more flexible residency requirements for code

contd.

segments. However, if the stack or heap expands into the pool, its entire contents are shifted in memory! Needless to say, this can have a detrimental effect on throughput. With a full 128K of memory, the code will have its own dedicated 64K, and more segments (especially more of the OS) will be able to reside in memory instead of rolling in from disk. SofTech stated that IV.13 would be available for the Apple in December; I don't know if it's actually out yet.

There are other, more effective ways to speed up version IV on an Apple][, if that's what you want to run for enhanced portability or other reasons. The most straightforward is an Accelerator][from Titan Technologies (formerly Saturn Systems). This will speed up your Apple by better than a factor of 3, which I would guess should make IV.x faster than Apple Pascal at standard speed. (See Ed Knepley's article on the Accelerator in the October 1983 WAP Journal.) For those with a //e, Titan has stated that the Accelerator //e will be available this month (the current one can't support the //e's extended memory). Another solution is to add a 68000 or 8086 co-processor card to your Apple. Most of the ones I've seen advertised feature an implementation of the IV.x p-System.

What about Apple /// Pascal?

I've long been convinced that the Apple /// is a much better machine than it is widely regarded to be, and discussions at the USUS meeting brought the wonders of Apple /// Pascal to my attention. One of the nicest features of Pascal on the /// is the amount of memory available. There is a full 64K for data, and on a 256K machine there is considerably more than that for code. This makes it the roomiest version of the p-System available (from Apple or SofTech). And the Pascal language itself has several extensions which make programming easier. There are two new variable types, WordStream and ByteStream, which make possible conformant-array procedure parameters (this is similar to the way Wirth added them in Modula-2). Also, the CASE statement has the very useful OTHERWISE clause. Plus the compiler supports conditional compilation using compile-time variables, expression evaluation, and an IF-THEN-ELSE structure (great for debugging or for porting code to a][). The console driver on the Apple /// is unmatched, offering extensive support for windowing and for text I/O to the graphics-mode screen. All this, and almost complete compatibility with Apple][Pascal! As David Ramsey points out in his discussion of Apple /// Pascal in the November/December 1983 issue of the "Journal of Pascal and Ada", the next time SofTech thinks about redesigning the p-System, they could do much worse than to emulate what Apple has done on the ///.

Apple itself has been paying a lot more attention to the /// these days, and this has resulted in several new Pascal products for the machine. The long-awaited "Apple /// Pascal Technical Reference Manual" was published a few months ago. It contains detailed descriptions of the Apple /// p-machine, code file formats, Pascal programming techniques, and methods for interfacing with the SOS system. The book also comes with a disk of utility and example programs. Later this year, Apple will be releasing the Pascal Development Toolkit for the ///. This will include a cross-referencer, a pretty-printer, and (heavens to Betsy) a debugger! Apple also released version 1.1 of Pascal for the /// last summer. I believe it included only minor enhancements, including some new Editor commands, but I really don't know very much about it. (Thanks to Ro Lutz-Nagey and Bill Hershey for much of the information in this section.)

PIG Meetings Revisited

One of the things I've often wondered about is how many people there are in WAP who use Pascal. From the sales of the PIG library disks, I know that it's many more than we've ever seen at meetings. (I also wonder how many people own Pascal and don't use it, for whatever reasons.) We'd like you to come and share your knowledge and experiences with us. Also, if you have a particularly intractable program bug, feel free to bring it in for some help. But please bring the current listing of the program and a copy of the source and code files on disk. It's very hard to debug a program without seeing it, and I hate to think of all the time that's been wasted by people using an old listing for debugging. Over the next few months, PIG will have a number of interesting presentations. These will include screen windows and double-wide lores graphics on the //e, a Pascal to Modula-2 source code translator, and the design and implementation of a text formatter.

New PIG12: Library Disk

Once again, I have the description of a new PIG library disk to pad out the column. This time we have another collection of Pascal programs from around the country. TRACKER and WEIGH-IN are substantial applications programs for tracking stock prices and weight, respectively; DIABLO is a two-part utility for Diablo daisy-wheel printers; and MORSECODE is for all of you ham radio enthusiasts. I'd like to encourage the contribution of more programs which perform tasks useful to people other than computer programmers. PIG12: is now available from WAP through the usual channels.

TRACK.INFO.TEXT
TRACK4.TEXT
UTRACK.TEXT
TRACKALSO.TEXT
TRAKCOMP.TEXT
TRAKGRAF.TEXT
UENTRIES.TEXT
TRACKER.CODE
UENTRIES.CODE - Jerry Crawford

TRACKER is a complex program for tracking the prices of stocks or other speculative investments. It proproduces graphs and listings of each item's price history, and also makes performance comparisons among the investments. Beware of option 5, "Create new set of files": this can cause the data files on your disk to become inaccessible. TRACK4 is the main program. UTRACK, TRACKALSO, TRAKCOMP, and TRAKGRAF are regular UNITS which are used by TRACK4. UENTRIES is a general-purpose intrinsic UNIT for filtering console input. TRACK.INFO contains some background information about the program. Because of the large number of program components, a compiled and linked code file has been included as TRACKER.CODE. It expects to find UENTRIES.CODE installed in your SYSTEM.LIBRARY. One note for those who have more than two disk drives: there is a bug in TurtleGraphics which causes it to look for SYSTEM.CHARSET only on units 4 and 5 rather than on all blocked units. If your graphs aren't labelled, put a copy of SYSTEM.CHARSET on your disk in drive 4 or 5.

DIABLO-DOC.TEXT
DIABLO.TEXT
MINIPUB.TEXT
DIAPLOT.TEXT
PLOTTEST.TEXT - Paul H. Kimpel

This Diablo printer interface package came from the San Antonio Appleseed User Group via an exchange with the Fort Worth Apple User Group/Dallas Apple Corps.

contd.

Lee Meador of the DAC described it as follows: "This is a couple of units, some test programs, and documentation that is quite thorough. The purpose of these is to implement control of a Diablo 1600 series printer via interface card in slot 1. Some Qume printers and some NEC Spinwriters use a Diablo interface also. You have functions to print lines, paginate, wait for letterhead entry, print subscripts, double strike characters, underscore automatically, and a bunch more. The plotting package will let you at least plot straight lines." DIABLO-DOC is the documentation of the programs. DIABLO is the text printing unit, which is exercised by the mini-formatter MINIPUB. DIAPLOT is the plotting unit-in-progress. Its functions are demonstrated by PLOTTEST.

WEIGH.INFO.TEXT
WEIGHIN.TEXT
MKWTDATA.TEXT - John Stokes

Have you been scarfing down too much junk food during those late-night debugging sessions? (Yes, it's true - Real Programmers just love Double-Stuf Oreo cookies!) Well, this program will help you keep track of your gains and losses on the fat front. Weigh-In lets you enter your weight every day and gives you a graph of your weight over a two-month period. The program keeps one year's worth of data on file, and this also can be reviewed. Weigh-In additionally keeps track of your high and low weights, and lets you set a goal to measure your weight against. Editing of the file (to correct mistakes, no cheating!) is also supported. WEIGHIN is the main program which you use to enter, display, and edit your weight data. MKWTDATA is a utility program which must be run once to initialize the weight data file. WEIGH.INFO is John's very clear documentation. There are a couple of interesting "tricks" in WEIGHIN. John pulls the current date out of memory (your boot disk might not be on-line), and he also peeks at a memory location to determine if your computer has an eighty-column card. He uses this last information to control the centering of the program menu.

MORSECODE.TEXT - Brian Cuthie and Jim Harvison

MORSECODE is a program for use by ham radio enthusiasts to practice their Morse code receiving skills. It reads a specified text file from disk and transmits it in Morse code over the Apple's built-in speaker. The program asks you for two parameters: speed and spacing. Speed controls the length of the dots and dashes; 1 is the fastest and anything greater than 15 isn't useful. Spacing dictates the pause between characters, and is scaled by the speed parameter. Its value for proper code formation is between what 2 and 3 give you. The program needs a few improvements still. It should be possible to specify the speed in words per minute, and the spacing between words is too short. MORSECODE was originally written by Brian Cuthie for the IBM-PC (bleah!) in Microsoft's Pascal. Jim Harvison modified it to work under Apple Pascal. Brian used a compact encoding technique to define the dot-dash sequence for each character. It suggests the presence of bit-manipulation functions in Microsoft's Pascal (I'm still not comfortable with how Apple Pascal uses, ignores, or clears the "irrelevant" bits in Boolean variables).

Thanks again to the contributors, and please keep sending your programs in. We're still waiting for our first contribution from another country (either that or one from each of the 50 states!). If you have a contribution for the library or news for the column, please contact me at (301) 445-1583, on the WAP BBS (WAP284), or on CompuServe [73075,1171]. Contributions may be mailed to me in care of the club office.

TELECOMM SIGNEWS

by

George V. Kinal

The December meeting of the Telecommunications SIG was held after the "garage sale" and distress auction activities on December 17. The lateness of the hour, coupled with a "Skins" game, contributed to sparse attendance. For the half-dozen members who did stay, a wide range of topics was covered. The Chairman relayed current and pending changes to the ABBS:

1. New telephone number, 986-8085, is via an electronic exchange, not the ancient crossbar of the old number, and provides two telephone lines. The signal quality via this new number should be better, and the electronic busy signal should not give false carrier detects to smart modems.

2. The message file space allows for 255 messages, instead of the previous 138. This change has been implemented.

3. The software and hardware will shortly be converted to use the Anchor Mark 12 modem and modified CCS 7710 interface. This will allow calls at either 300 or 1200 bps, automatically switched.

4. Installation of a lower case chip will allow the inclusion of lower case in bulletins and other messages. This modification has been implemented.

5. Changes to support two lines (calls) simultaneously will be made Real Soon Now. Actually, the software to properly support two simultaneous 1200 bps calls on an Apple][represents a truly ambitious undertaking, and when (not if !) this happens it will be a true first among ABBSs.

Other matters reviewed by the SIG included status of the Group Purchase order for Anchor Mark 12s, and the identification of another source for $59 serial interface ("Comm.") cards. I will post an announcement on the ABBS concerning the adequacy of a PROM modification to be supplied by Anchor to cure a shortcoming in early versions of the Mark 12, whereby they were too sensitive to signals at higher than normal strength (the most prominent of which was our own ABBS on the old 657 telephone number!). Mark 12 owners who experience garbage characters when calling certain modems are advised to contact Anchor for a return authorization; the modems will be upgraded with the new PROM under Anchor's two year warranty, and returned by UPS Blue Label.

SOFTRONICS SOFTERM 2 COMMUNICATIONS PROGRAM: A Review

by Bob Oringel

When an Apple computer owner looks for a communications program to drive a modem, and s/he is considering Softronics, "Softerm 2", the questions uppermost in mind are, "What should a communication, or terminal program do for its user ?", and "How well does Softerm 2 perform those functions?".

The primary functions of such a program are, in my estimation:

1. To contact and communicate with a BBS (bulletin board) or with a remote data base like The Source or Dow Jones;

2. To send and/or retrieve data files, hopefully in large quantities, via telephone, between Apple computers;

3. To provide the user with a bunch of useful shortcut devices like keyboard macros, auto-dialing (if your modem cooperates), easy printing to screen or disk, and the like;

4. In the case of Softerm 2, to emulate, or look like the same brand of computer terminal, as a host computer that you wish to contact.

And how well does Softerm 2 perform? While it falls short in describing a clear methodology for accessing a BBS (Item #1), it performs admirably on all of the items that I was able to check. Alas, my D-Cat modem has no auto-dial facility. In performance then, four out of four, is not a bad average.

A communications program, one that connects an Apple computer to the "outside world", necessarily must have very clearly written documentation. Else the would-be communicator will not be able understand its methodology. The documentation for Softerm 2 is clearly and professionally written, and is indeed comprehensive. It is typeset and professionally printed, in looseleaf form. It is formidable, a volume that is 3/4" thick. It has an index, and it is packaged in a very handsome binder.

BUT, nowhere, does it take the novice by the hand and say, "here, in simple, one-two-three, procedure, is how to initiate a call to a BBS or to a database". The information is there, but you must dig it out. The conclusion that one must come to then, is that Softerm 2's documentation is not written for the novice. For this reviewer, that is a serious fault. We all begin as novices.

Softerm 2 includes two disks, a System Disk and a System configuration Disk. Each has programs on both sides of the disk. The disks are not copy protected. They could not be, as the program must be configured to the user's equipment (modem, interface cards). We give Softerm a big plus on this score anyway. The program's segments are menu driven, and many program segment elements default to the most common usage, which simplifies user operation.

Softerm 2's features include: Compatibility with Apple DOS, CP/M, and Pascal files; Printer support, although I was not able to find a way to adjust paper margins within the program; Automatic dialing from a user generated built-in phone book file, unchecked in this review, but a super feature to have if your modem supports it; Terminal mode line capture to printer, to disk or simultaneously to both; A FORTRAN 77 source program (unchecked in this review) to provide Softrans protocol compatibility with any host computer; support for a profusion of printer and modem interface devices; and an Online Update Service for Softerm program updates, enhancements and correspondence with users.

The Softerm package includes yet another card for the Apple, which is connected by cable, to a three-key keypad. These keys are used like shift keys, in combination with the Apple keyboard to perform Softerm functions.

PHONE LIST PRINTER

by J. Tom DeMay, Jr.

I remember reading stories about the beginnings of the railroad industry. There were many railroad companies and each railroad set its own standards. This included such things as the size and shape of the rails, and even more important, the distance between the rails. These non-standards were responsible for considerable difficulties and expense in the transportation of goods across railroad boundries. The engines and rolling stock of one railroad could not be used on the tracks of another railroad because of the different "gauge" (distance between the rails). Freight cars from "connecting" railroads were parked side-by-side on specially built tracks so that goods shipped across railroad boundries could be unloaded, then reloaded onto cars of the other railroad before they could be delivered to their final destination.

That's where the computer industry is right now. Computers can't talk directly to other computers without special electronics and/or software. Even worse, programs that run on the same computer can't usually use the same data files. Several attempts have been made to correct these problems. Computers can be connected together thru networks such as ETHERNET (tm). DIF (tm) files were developed to facilitate data transfer between programs. But the modem has had the biggest effect on the problem of transporting data between computers.

I have been using a NOVATION APPLECAT][for some time now, and I have few complaints. Included with the modem is COMWARE][, a very good terminal program. There are very few features missing from this program. Two that come to mind are; 1. REDIAL FOREVER - this would instruct the computer to continually dial a number until it was answered. 2. TIMED DIALING - dial a number at a preset time. The latter probably was not included because it would involve the use of a clock card.

COMWARE][permits storing and automatic dialing of 26 different numbers of up to 36 digits each. The manual says 56 but 36 is more accurate and should be enough to cover most situations. A little-known fact is that passwords can also be included with these numbers. Just enter the phone number as usual, then enter several spaces (I use four), then enter your password. The key is that COMWARE][tries to dial the first 36 characters in the number field. If the characters are not numbers or control characters (+ wait for additional dial tone, or @ pause for a two second delay) COMWARE][will ignore them. Then when a number is dialed, the password will be printed on the screen. Be sure not to include the digits (0-9) in your password. If the password you want to enter does contain numbers, just spell them. It may be necessary to use only enough of the spelling to remind you of the correct password. The simple substitution of an O for a zero will also work. The password printed will not be sent to the distant computer, but only serves as a reminder to the operator.

The directory and passwords are stored on the disk as a binary file called PHONE LIST. There is no provision for obtaining a paper copy of the directory, and that brings me to the purpose of this article. I have written a basic program to load the PHONE LIST file, and print it to the screen or, optionally, to a printer. Just enter the program as presented here, or if you can wait, it will be included on a WAP library disk in the near future. Check the WAP ABBS, it may be included in the DOWNLOAD section.

The operation of the program is straightforward. If you type it in from the Journal, please check your typing before RUNning it. It will work under most circumstances. There is a problem when trying to RUN this program with Diversi-Dos (tm) version 3-c and GPLE (tm) configured to use the APPLE at the same time. If you avoid this combination all will work well.

If your printer is in other than slot #1, you will want to change the "PR#1" in line #300 to "PR#*" where the * represents your slot #. The pokes in line #320 serve to position the column labels on the paper copy. These may be replaced with whatever is required to convince your printer of the proper location for these labels.

I hope this has been informative and will enhance your enjoyment of an already outstanding product. As always, if you have any comments, corrections, or suggestions, I can be contacted through the WAP office, or the telephone number listed in the front of the WAP Journal.

Listing

```
100  REM ********************
110  REM *                  *
120  REM *    PHONE LIST    *
130  REM *     PRINTER      *
140  REM *                  *
150  REM *   BY TOM DEMAY   *
160  REM *     (C) 1983     *
170  REM *    WASHINGTON    *
180  REM *     APPLE PI     *
190  REM *                  *
200  REM ********************
210  TEXT : HOME : PRINT  TAB( 12)"COMWARE ][ PLP":
     PRINT
220  PRINT "  THIS PROGRAM WILL PRINT THE DIRECTORY
     NUMBERS AND NAMES SAVED BY COMWARE ][,  THE
     COMMUNICATIONS SOFTWARE DISTRIBUTED BY NOVATION
     WITH THEIR APPLECAT MODEM."
230  PRINT : PRINT
240  D$ =  CHR$ (4)
250  AD = 10000
260  PRINT "PRESS [ESC] TO EXIT OR INSERT COMWARE
     ][DISK AND PRESS A KEY...";: GET A$: PRINT A$:
     PRINT
270  IF A$ =  CHR$ (27) THEN  END
280  PRINT D$"BLOAD PHONE LIST,A";AD
290  PRINT : PRINT
300  PRINT "PRESS [P] TO PRINT A PAPER COPY OR ANY
     OTHER KEY TO PRINT TO THE SCREEN...";: GET P$:
     PRINT P$: IF P$ = "P" THEN  PRINT  CHR$ (4)"PR#1"
310  HOME
320  IF P$ = "P" THEN  POKE 36,4: PRINT "PHONE #";:
     POKE 36,16: PRINT "PASSWORD";: POKE 36,44: PRINT
     "SYSTEM NAME": FOR L = 1 TO 80: PRINT "-";: NEXT
330  FOR I = 1 TO 26
340  IF  PEEK (AD + (71 * I)) = 26 THEN 450
350  PRINT "<"; CHR$ (I + 64);"> ";
360  FOR J = 0 TO 70
370  IF J = 36 THEN  PRINT "    ";
380  I$ =  CHR$ ( PEEK (J + K + AD))
390  PRINT I$;
400  NEXT
410  IF P$ <  > "P" THEN  IF I = 10 OR I = 20 THEN
     VTAB 24: HTAB 1: PRINT "PRESS ANY KEY...";: GET
```

contd.

```
    A$: PRINT A$: HOME
420  K = K + 71
430  PRINT
440  NEXT
450  PRINT D$"PR#0"
460  REM

65535 REM **********************
65535 REM *                    *
65535 REM *  IF YOU HAVE ANY   *
65535 REM *  COMMENTS, FIXES   *
65535 REM *  PLEASE  CONTACT   *
65535 REM *  J.T. DEMAY, JR.   *
65535 REM * 4524 TUCKERMAN ST *
65535 REM *   RIVERDALE, MD    *
65535 REM *       20737        *
65535 REM *                    *
65535 REM **********************
```


JOB MART

HELP WANTED

TUTOR NEEDED: I have an Apple][+, 2 disk drives, Okidata 92 printer, Micromodem //e, ScreenWriter][and PFS. Need help in utilizing this hardware/software combination to its fullest. Contact Margaret Daneberg, 983-9488.

SMALL EDUCATIONAL SOFTWARE DEVELOPER located on Wisconsin Avenue in Washington, DC needs a full-time Apple programmer. Call days 966-3686.

MANAGEMENT ASSOCIATES, an Executive Search Firm, has a client with an immediate need for Systems Analysts, Systems Programmers, Programmer Analysts, and Programmers for a new Systems Development Group being established in a large local commercial company. Work will start on an IBM 4341, then develop to an IBM 4381. Requirements are 3 year's experience in those positions with a working knowledge of several of the following: MVS, JES2, ISPF, JCL, TSO, CICS/DMS, ACF/VTAM, Utilities, and SNA. Call Wendell at 894-7850.

POSITIONS WANTED

Do you need personal help on Apple programs or applications? We'll help with almost anything, including custom data bases, real-time control systems, hardware problems. Diversified experience, reasonable rates. Call Steve Knowles, 767-3010.

EDSIG NEWS
by Peter Combes

EDSIG Calendar

Tuesday, February 7th at 7.30 p.m.

Subject: "Does LOGO Deliver?"

EDSIG meetings are held in the Auditorium, Building B, of the Uniformed Services University of the Health Sciences, on the campus of the National Naval Medical Center, 4301 Jones Bridge Road, Bethesda, MD.

Meeting Reports

Tuesday, January 2nd at 7.30 p.m.

"Microcomputers in Montgomery County" - Chuck Philipp.

21 years ago, Chuck Philipp entered the Montgomery County Public Schools System as a science teacher. The IBM 7094 was then a "state of the art" machine. Chuck taught one group of eighth graders how to do simple programs in FORTRAN, as part of a science project. From handing in the punched cards, it took one and a half weeks to get the results. In the late sixties, MCPS piloted a Computer Assisted Instruction project, on a mainframe computer at Einstein High School. The goal was to produce some drill and practice programs, some low level simulations, and some "electronic page turning". The project was funded over a five year period, and in terms of its goals was reasonably successful. Problems were associated with the cost, the physical limitation that terminals could only be up to 1100 feet from the computer, and the problem of overbright (Ed. Note: overbright ??!!) students who "broke" into the inner files of the computer.

As Chairman of the Science Department at Churchill High School, Chuck was involved in a computer club that used time-sharing terminals - at a cost of $12.82 per month - that, courtesy of one of the parents, was soon on line to MIT with LISP and other languages. At the same time, a teacher at Walter Johnson High School started a course in FORTRAN.

By 1972/73, it was obvious that high school students did very well on computers, so well that the school system became seriously concerned about the equity of the situation, since some students obviously received much more facilities than others. As a result, MCPS supplied terminals to every high school in the county, connected to an IBM 370 mainframe computer. One disadvantage of this system was that, again, the software technology could not keep the kids out of important files. Ohio Scientific microcomputers were distributed to high schools to give the kids something safer to work with. These were in fact sent to mathematics departments, which kept professional ownership of the OSI's.

By the late 1970's a different educational strategy was emerging. Lots of games were developed by the kids. There was lots of "programming to teach programming", and mathematics departments used the machines to teach a 3-4 week course in support of either Algebra or Geometry. All this was clearly different from the orginal concepts of CAI.

In 1978, two staff people in the Gifted and Talented Department of MCPS acquired PET computers. These were made available to kids without formal instruction, and it was found that some students were attacking 6502 assembly language before the end of the 6th grade. There was no teaching problem and no curriculum problem - the kids just got on with it. Soon they were performing better than either their teachers or their parents expected.

MCPS found that its emphasis began to shift away from traditional CAI.

HUMRRO obtained a grant from the National Science Foundation to develop a framework for K-8 Computer Literacy, and MCPS had part of this grant with HUMRRO. One major outcome was the book "My Students Use Computers" by Beverly Hunter, published by Reston. Some 75 Montgomery County school teachers are named in the book as participants.

The aims of the project were:

"To provide students with computer related skills, knowledge and attitudes which they will need to function effectively in our increasingly computer oriented, information based society.

To use computer technology effectively and efficiently to support and improve the teaching/learning process across all disciplines at all instructional levels, with both regular and special needs pupils."

One severe conceptual problem in the teaching of computer-related skills is that the current student force will be at work at the turn of the century, and we cannot predict what computers will be like then. So what can we teach? Obviously, we cannot teach "Computer Science" based on the technology of today. To take an extreme example, it would be unreasonable to spend a great deal of class time teaching the more obscure PEEKS and POKES of the Apple][. The need is to develop fundamental concepts of the human-machine interface that will always be valid.

MCPS observes that not all students are as at home with computers as others are. Many minority students have not developed confidence in using computers. Some very bright young people who learn programming very quickly, nevertheless fail to focus on any other kind of skill.

Currently, MCPS looks at Computer Literacy Education from two aspects. "Awareness", which is developed in K-8, looks at fundamental concepts and applications in society. "Use" includes Computer Managed Instruction, Computer Assisted Instruction, Utilities and Programming, the computer as a functional tool in mathematics, Business, Social Studies, Science, English, and other subjects.

The only current CMI is in mathematics, which is partly on a mainframe computer, and partly on Apples. There is little CAI, apart from some at a few schools such as the Takoma Park Junior High School. The use of utilities is growing; a recent example was the use of software to attach mark sense card readers to the Apples.

All the school media centers have fully configured Apples attached to the Dialog data base, a commercial data base that allows efficient literature searches. In Social Studies, there is some simulation, and some use of data bases. In Science, each Senior High

contd.

School has 3-4 Apples, attached to hardware, for reading data and starting the analysis. (This has been demonstrated both at an early EDSIG meeting, and at a main Pi meeting.) Most science teachers allow their pupils to write programs to "massage" data.

The Edison Career Center has an IBM 4321 mini-computer, and 15 IBM terminals, together with IBM PC's, a Wang system, and a few NBI machines. Other technical centers, connected to satellite centers, are planned.

There are not many formal projects in English teaching, but several teachers have their own projects. In one school, journalism classes are using word processors, modems, and computerized typesetting. In Vocational and Technical Education, there will shortly be a Computer Assisted Design system in place. Sherwood High School and a number of other schools in the Eastern side of Montgomery County are developing the use of micros in Music.

Montgomery County sees instructional materials, equipment updates, staff training, and the curriculum all developing together in a "bootstrap" operation, but Chuck pointed out that a county that covers 500 square miles, has 90,000 students and 7,000 professional staff people for curriculum and instruction, and has a budget of a third of a billion dollars, cannot change overnight. Currently it has 1100 microcomputers, and in 4-5 years will have nearly four thousand.

Chuck listed examples of courses being developed for junior and senior high Schools, and In-Service courses being developed for teachers.

Vigorous discussion on such topics as ways of funding, types of micros, and what to do about the apparent "sexism" of computer literacy kept this fascinating EDSIG meeting running well past the usual closing time.

LOGOSIG NEWS by Nancy C. Strange

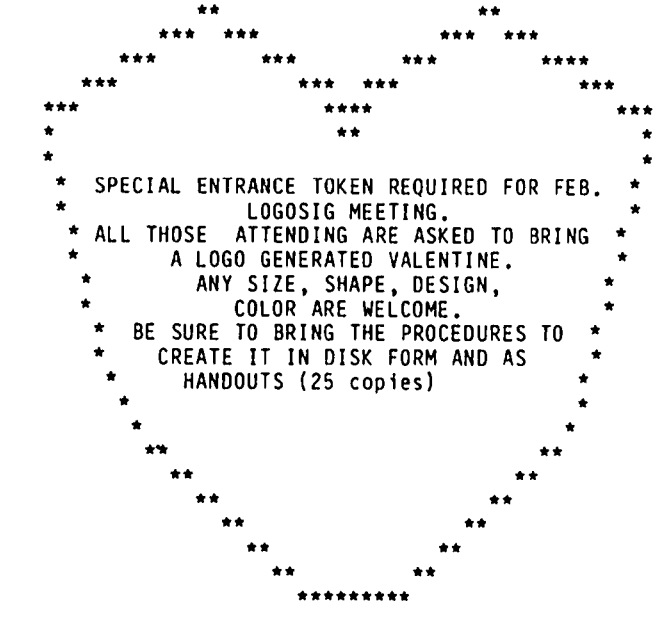
TURTLE TRACKS

JANUARY 28 LOGOSIG MEETING

```
**************************************************
*                                                *
* THERE WILL BE NO LOGOSIG MEETING FOR JANUARY DUE *
*   TO THE SPECIAL MEETING WITH WOZ AND MAC.     *
*                                                *
**************************************************
```

FEBRUARY 23rd LOGOSIG MEETING

The upcoming February LOGOSIG Meeting will be at our regular location: Barrie School, 13500 Layhill Rd., Silver Spring, MD. (For instructions call Dave Weaver at 871-6200 or see a previous issue of the WAP Newsletter with a map.) June Wright of the University of Maryland will present the TASMAN ROBOT. If you have not seen "Shelly??" in action you will want to be there. If you have, come and meet it up close.



The LOGOSIG is open to all LOGOs. If you are using LOGO on any machine and want to come, bring your machine and share -- we welcome you. There are now APPLE LOGO, TERRAPIN LOGO, ATARI LOGO, COMMODORE LOGO and SPRITE LOGO regularly in attendance.

```
##################################################
#                                                #
# NOTE:  LOGOSIG SNOW POLICY -- If there is ANY  #
#        accumulation of snow, the LOGOSIG meeting #
#        will be postponed to the following Satur- #
#        day at 12:45 (Barrie schedule permitting. #
#                                                #
##################################################
```

Description		List $	Sale $
HARDWARE			
Disk Drives			
Micro Sci			
A35 w/o Controller		395.00	269.95
A35 w/Controller		495.00	349.95
Rana			
Elite One		379.00	269.95
Elite One Plus		519.00	349.95
Controller		145.00	89.95
Quentin			
500 - 5mb hard disk		1,995.00	1,449.00
5¼" floppy drive		335.00	249.95
Controller Card		65.00	49.95
Monitors			
BMC 12A: 15MHz		99.95	79.95
AMDEK			
Video 300G		199.00	149.95
Video 300A		210.00	149.95
Dot Matrix Printers			
Epson			
MX-80		549.00	359.95
FX-80		699.00	539.95
Mx-100		845.00	629.95
IDS			
Prism 132 w/all options		1,699.00	1,495.00
MicroPrism		799.00	529.95
NEC 8023		645.00	449.95
Okidata			
92		699.00	499.95
93		1,199.00	844.95
Daisy Wheel Printers			
C. Itoh			
Starwriter: 40cps		1,895.00	1,249.00
Prowriter: 55cps		2,395.00	1,479.00
Smith Corona TP-1		895.00	599.95
Transtar 130	"Best Buy"	895.00	659.95
Surge Protection			
Kensington			
System Saver		89.95	69.95

Also available

Modems
Interfaces
Disk drives
Accessories
Terminals
Peripherals
Joysticks
Paddles
Keypads

and, hundreds more!

Description	List $	Sale $
BUSINESS		
Data Managers		
dBase II with ZIP	700.00	399.95
DB Master	229.00	149.95
Utility pac #1	99.00	74.95
Utility pac #2	99.00	74.95
Stat pac	99.00	74.95
General Manager	229.00	149.95
InfoStar (Replaces DataStar)	495.00	299.95
PFS: File	125.00	84.95
Report	125.00	84.95
Graph	125.00	84.95
Versaform II	389.00	249.95
Visifile	250.00	169.95
Accounting		
Accounting Plus II +	995.00	599.95
II e	995.00	599.95
Business Bookkeeping System	395.00	239.95
E-Z Ledger	60.00	44.95
General Ledger w/Payables	495.00	379.95
Home Accountant	74.95	44.95
Payroll	395.00	294.95
Receivables	495.00	379.95
Financial Modeling & Planning		
Asset Manager	200.00	149.95
Budget Planner	150.00	119.95
Business Forecasting Model for VisiCalc	100.00	79.95
CalcStar	195.00	114.95
Multiplan	275.00	169.95
SuperCalc	295.00	199.95
VisiCalc II +	250.00	169.95
II e	250.00	169.95

and much, much more!

New Specials		
Davong 5mb hard disk drive	1,995.00	1,449.00
Includes cables, card and software		
Microtek		
Parallel Printer Card	130.00	89.95
Pkaso	165.00	134.95
Micro-Professor MPF-II	399.00	319.95
Apple Compatible portable!!		
Mind II	695.00	599.95
CSI slimline disk drive	295.00	239.95
Orange Micro		
Grappler Plus	165.00	129.95
Bufferboard	175.00	134.95
Syscom II		
95% Apple II Plus Compatible	895.00	659.95

COLUMBIA COMPUTER SYSTEMS

OVER 12,000 PRODUCTS AVAILABLE FOR YOUR APPLE

The largest & oldest supplier of Computers, Peripherals, & Software
in the Washington, D.C. Area

Mail your orders to: P.O. BOX 40813, Washington, DC 20016
or call (202) 364-0273 any time, day or night

Office Hours: 10:00am - 6:00pm M-F & 10:00am - 2:00pm Sat.

SOFTVIEWS

by David Morganstein

Chess 7.0. My first experience with a chess program for micros was an assembly language program written for the KIM microcomputer which I purchased with the aim of making it work on an OSI computer. The package was called Microchess and came on a punched paper tape. While I did get it to work, I am glad to report that things have gotten easier since then. Many Apple owners have purchased a much improved Microchess or Sargon II program. Several recent additions to the field have taken chess playing a long way beyond the capabilities of these early efforts. Chess 7.0 is written by Larry Atkin who is known for his main-frame chess programs which have won international competitions. This Apple program does a lot more than just play chess.

It is a useful package for teaching someone how to play. It can suggest moves, show squares which can be attacked by any piece, show squares which are attacking any piece and several other handy options. You can adjourn a game at any point and save it to disk for later use. The package comes with 40 interesting games, many classics. These can be stepped through for study using a handy replay feature. You can watch Chess 7.0 think, ask its advise for your move, or get it to play itself.

You indicate your moves by using the keyboard or paddles to select a piece which is to be moved. Chess then steps through the legal moves which can be made by that piece and you select the one you want. Personally, I would have liked the more traditional option of entering the algebraic notation designating my move; I think that is easier and faster. However, this technique is good for the learner since it shows all possible legal moves.

How well does it play? Better than I, which is not saying a great deal. You can select any of seventeen levels of play. Nine of the levels are time limited from one second per move (level 0) to eight to twenty-four minutes per move (level 8). There is an infinite time limit if you want Chess to keep looking for the best move. This might be useful for those who play chess by mail. Six of the levels are depth limited, that is, Chess 7.0 explores all moves to the given depth of play (number of moves ahead). These range from one to eight half moves, called plys. The newest programs, like Chess 7.0 and Sargon III think while you think. Thus, if you select the move the algorithm expected, the program is one step ahead of you!! Chess 7.0 has an "opening book" of 7000 positions. If you remain in the book, Chess will move automatically without "thinking".

I have very few criticisms of the package or the extensive 64 page manual. A nice option would have been to permit the game moves or a given position to be printed. Odesta, 930 Pitner, Evanston, Ill. 60202. Price $59.95

Amperware. If you have written a long program in Applesoft, you no doubt have been dismayed by the lack of certain handy functions found in other languages or systems. If your program does disk accesses, you may have experienced the "long" delays associated with DOS activity. There are commercially available subroutines which add many valuable features to Applesoft. A lot of these packages are implemented with a series of & routines. Amperware, one such package, provides routines which fall into the following four categories: disk, I/O, screen and cursor and miscellaneous.

DOS enhancements include the ability to &WRITE and &READ whole arrays of variables with a single command at much increased speeds. The same speed increases apply to I/O of individual variables, as well. While the files created by AW can not be accessed with standard disk INPUT and WRITE commands, this is not a great limitation unless you need to communicate data with commercial programs requiring the standard format. There is an &SPACE command which returns a count of the number of blocks remaining on the specified diskette.

My greatest heartache with Applesoft is its lack of a PRINT USING (formatting) option. The standard PRINT statement produces terrible results for computed real numbers and does not allow for fancy formatting. AW provides many handy options including the specifying of the number of decimal points and inserting commas every three real digits to the left of the decimal. An &INPUT command allows for editing upon data entry.

The screen and cursor commands are somewhat less valuable. The more useful are an &MOVE command which sends the cursor to the row and column specified (like the Pascal GOTOXY), and an &CLEAR PAGE or &CLEAR LINE which clear the current display window or line.

Under the miscellaneous category we find the ability to GOTO or GOSUB to computed line numbers, handy programming extensions, not recommended by the structured programmer. The most useful options are the sort and find commands. &SRT can be used to sort a single dimensioned array, or a keyfile can be used to create an indexed sort. Multikey sorting is also possible using several key arrays. The &FIND command can be used to search a string array for a substring at machine language speeds. Both sort and find commands can be selective, beginning and ending at specified locations in the array. Another useful option is the ability to selectively erase an array, clearing up needed memory space.

The package is easy to use. The software resides from $84A0 to $9CFF and the manual clearly explains how to activate it. Scientific Software Products, Inc., 3171 Donald Ave., Indianapolis, IN. 46224. (317) 299-0467. Price $49.95 The manufacturer is offering user groups a substantial discount for group purchases. If you are interested in buying this package, call the WAP office. If we get a half dozen buyers, we will order it for you.

Locksmith 5.0. A completely revised version of the classic nibble copy back-up program is available. The new features include: a text editor for entering Locksmith Programming Language commands; a graphic quick scan of a diskette to uncover patterns; 16 sector utilities for DOS 3.3 disks including fast disk copying and verifying, formatting selected tracks and comparing two disks; improved back-up algorithms which do automatic self-sync determination and nibble counting. The package can read and write on quarter and three-quarter track boundaries. The update includes a new 140 page manual which carefully explains most features of the new version. It includes a brief history of disk protection schemes. The program no longer resides entirely in memory. For some options, the original disk must be reinserted so that additional code can be read.

contd.

I found the manual brief in its discussion of the Programming Language. The user will have to do a bit of studying (pun intended) to figure our what is going on and how to use the program. Competing products include a discussion of the technical issues such as self-sync nibbles, nibble counting, data encoding and formats, etc. This manual contains very little discussion of these subjects and what is there is sprinkled throughout the manual.

The Omega people claim the new package is capable of backing-up all existing software. Even if this is true, a user must realize that the process is not automatic, anymore than it is with competing products. A user must still delve deeply into the methods of data encoding and storage to make successful, archival back-ups of many commercial products. However, for the knowledgeable, this version is one of the most powerful tools available. Omega Microware, Inc., 222 So. Riverside Plaza, Chicago, ILL. 60606 (312) 648-1715. Price: New $99.95, Update for 4.x owners $24.95

DISABLED SIGNEWS

by Jay Thal

* * * * * * * * * * * * * *

DISABLEDSIG FEBRUARY MEETING
THURSDAY, FEBRUARY 9, 1984, 7:00 P.M.
Chevy Chase Community Center
Connecticut Ave. & McKinley St., NW, D.C.

* * * * * * * * * * * * * * * * *

AN OPEN LETTER TO STEVE WOZNIAK

Dear Steve,

On behalf of the DISABLEDSIG, and those others who use your invention to aid the handicapped, I am writing to enlist your assistance.

One of the fundamental reasons I bought my Apple over three years ago was to try to use this technology to assist a learning disabled son. It was my feeling, echoed by others, that that hunk of silicon, metal, and plastic could open up new vistas. The keyboard provided a fertile playground for my son to explore. The computer was patient, non-threatening, and a tool for education. It has proved to be all those things. But it still could do more.

My son first learned to spell his name at the Apple's keyboard. He found enjoyment through some of the software. He was limited though by the fact that I'm neither a professional programmer nor an educator.

But, this is not just my son's story, or that of other children with learning disabilities. It is the story of a large segment of our population which is physically, emotionally, or educationally handicapped. The hardware has been developing faster than the software. The needed communications links between those working at making the technology accessible and useful and the disabled have also been developing at a slow pace.

When you last visited Washington Apple Pi, in May 1980, you made a chance remark after the meeting. You said that you would like to give a computer to every disabled person in the country. A visionary's statement -- but you had already proved to be a visionary by developing an industry.

It would be nice if every handicapped person had access to a computer -- if he or she could utilize it. Right now that is not practical, but it could be in the future. Many people are working at that goal. Many more than you, or I, or those doing the work, know about. Of course there's TRACE, and Johns Hopkins, and the Council for Exceptional Children, and CLOSING THE GAP, etc. but the gaps in the communication link need to be closed.

Apple produced a short resource guide in early 1981 entitled: PERSONAL COMPUTERS FOR THE PHYSICALLY DISABLED. It needs updating and expansion. The Apple Education Foundation, once, produced THE APPLE JOURNAL OF COURSEWARE REVIEW. There needs to be more, with some focus on applications for the handicapped and software. Apple and its Foundation can provide focus; they could establish a clearinghouse; they could provide further funding. You needn't give away thousands of Apples. Coordination is what is needed, communications and education, as well. Then, maybe the Apples.

Thanks for hearing me out.

Sincerely,
Jay M. Thal

UNDELETING A FILE

by Rudie Slaughter

When DOS 3.3 executes a delete file command, it does not actually delete the file. Instead it destroys the file's track/sector list pointer, flags the catalog entry as deleted, and marks the sectors used by the file as unallocated storage space. The file name still exists in the disk directory, only the catalog command ignores files flagged as deleted. There are many public domain file recovery programs available for restoring Apple files that have been inadvertently deleted from a disk. They usually work by rewriting the file directory entry of a deleted file, erasing the deleted file flag and restoring the file's track/sector list pointer. Programs of this type must be used before any additional files are saved to the disk, as the new file names will overwrite the deleted file names. Although the name of the deleted file is usually properly restored by these programs, and everything that shows up at the user level in the disk's catalog appears normal, there is often a hidden trap for the unwary that has fatal consequences. When a file is deleted by DOS, the sectors that were allocated to it are marked as free in the Volume Table of Contents (VTOC). Many 'undelete' programs do not restore the marking of the file's sectors to indicate that they are again in use. Thus, when something new is saved to the disk, DOS may overwrite the sectors used by the 'restored' file because it sees the sectors as unallocated storage space. Once the sectors of a file are overwritten, they are gone forever. The probability of this overwriting is inversely related to the free space left on the disk. Thus, if the 'recovered' file is on a fairly empty disk, the overwrite and loss of the file may not occur until several additional files are saved, perhaps weeks or months later. Consult the sections on VTOC and track bit maps in Appendix C of the DOS manual for further details.

There are two solutions to the problem of not restoring the sector allocation map. The first solution is to, upon exiting the 'undelete' program, immediately LOAD the 'restored' files into memory (DOS can read the file if nothing has been saved on top of it) and then SAVE it to the disk again. The SAVE command may put the file back on the disk in a different physical location, but this should not affect the typical user. This solution to the problem involves two additional disk access commands and for large files can take some time. The second, more elegant, solution is to have the 'undelete' program fix up the VTOC whenever it recovers a deleted file.

FILE SALVADOR is such a program. The coding is my own, but the inspiration for the some of the program logic is from other 'undelete' programs. I am indebted to Worth and Lechner's book, Beneath Apple DOS for providing guidance to the DOS file manager subroutines used in the program. FILE SALVADOR will work with DOS 3.2, DOS 3.3, and any enhanced or customized DOS that preserves the standard entry points used in line 1 of the program. These DOS versions expect to find a standard VTOC on track $11 at sector $0. This VTOC can describe a disk with up to 32 sectors per track and 49 tracks per disk. Thus, FILE SALVADOR will work with 40 track drives, and some double-sided drives, such as the RANA Elite Two. It will not work with disks that have 80 tracks as it does not know where to find the track bit maps for tracks 50-79. Although the program does not check that these limitations are met, this can be remedied by the addition of a statement or two after the VTOC is read in line 17 to insure that PEEK(VB+52)<50 and PEEK(VB+53)<33..

The program can be used to access any disks in drives connected to the last active disk controller. It has no provision for changing drive controller slots. In order to do so, with the program in memory, issue a CATALOG, S# command to the access the alternate slot and then RUN the program. Or add another prompt similar to the one that asks for the drive number (DOS stores the current slot number multiplied by 16 at FW+38). The program could be easily changed into 6502 assembler code, but the small increase in speed is probably not worth it, as the program only takes a few seconds to search an entire disk directory for deleted files.

Listing

```
 0 REM    FILE SALVADOR
 1 FW = 46545:FS = 45975:DW = 45111:VR = 45047:DB =
   46267:VB = 46011: GOTO 15
 2 C = C * 4 + 59 + VB:B = B + 32 -  PEEK (VB + 53):
   C = C -  INT (B / 8):B = B - 8 *  INT (B / 8):
   POKE C, PEEK (C) - 2 ^ B: RETURN
 3 POKE DW + 10,1
 4 POKE FS,T: POKE FS + 1,S: CALL DW: POKE DW + 10,2:
   RETURN
 5 F = F + 1:F$ = "": FOR J = I + 3 TO I + 31:F$ =
   F$ +  CHR$ ( PEEK (J)): NEXT J:J = 31
 6 IF  ASC ( RIGHT$ (F$,1)) = 160 THEN J = J - 1:F$ =
   LEFT$ (F$,J): GOTO 6
 7 PRINT : PRINT "RESTORE "F$"? Y"; CHR$ (8);: GET A$:
   IF A$ =  CHR$ (13) THEN A$ = "Y"
 8 PRINT A$: IF A$ <  > "Y" THEN  RETURN
 9 POKE I, PEEK (I + 32): POKE I + 32,160: GOSUB 4:
   U = S:V = T:T =  PEEK (I):S =  PEEK (I + 1)
10 C = T:B = S: GOSUB 2
11 GOSUB 3: FOR J = 12 + DB TO 254 + DB STEP 2:
   C =  PEEK (J):B =  PEEK (J + 1): IF  NOT (B + C)
   THEN J = 254 + DB: GOTO 13
12 GOSUB 2
13 NEXT J:T =  PEEK (DB + 1):S =  PEEK (DB + 2):
   IF T + S THEN 10
14 CALL VR + 4: PRINT F$;": RESTORED": PRINT :S = U:
   T = V: GOSUB 3: RETURN
15 TEXT : HOME : NORMAL : PRINT "RESURRECT DEAD FILES
   ON DISK IN DRIVE "; PEEK (FW + 39); CHR$ (8);:
   GET A$: IF A$ =  CHR$ (13) THEN A$ =  CHR$ ( PEEK
   (FW + 39) + 48)
16 PRINT A$: IF A$ < "1" OR A$ >  CHR$ ( PEEK (43355)
   + 48) THEN  PRINT  CHR$ (7);: GOTO 15
17 POKE FW + 39, VAL (A$):F = 0: CALL VR:T =  PEEK
   (VB + 1):S =  PEEK (VB + 2)
18 GOSUB 3: FOR I = 11 + DB TO 221 + DB STEP 35:
   IF  PEEK (I) +  PEEK (I + 1) = 0 THEN I = 221 + DB:
   NEXT I: GOTO 21
19 IF  PEEK (I) = 255 THEN  GOSUB 5
20 NEXT I:T =  PEEK (DB + 1):S =  PEEK (DB + 2):
   IF T + S THEN 18
21 IF F = 0 THEN  PRINT : PRINT "NO DELETED FILES
   FOUND, V."; PEEK (VB + 6);", D"; PEEK (FW + 39)
22 PRINT "PRESS ANY KEY TO CONTINUE, ESC TO END";:
   GET A$: IF A$ <  >  CHR$ (27) THEN 15
23 END
63999  REM **********************
     *  PUBLIC DOMAIN PROGRAM   *
     *                          *
     *  RUDIE W. SLAUGHTER, III *
     *  5341 ESSEX COURT #51    *
     *  ALEXANDRIA, VA    22311 *
     ****************************
```

NOTES FROM THE SYSOP

Washington Apple Pi members whose membership numbers are listed below have not used the club Apple Bulletin Board System (ABBS) for the past year and will have their passwords deleted after the February meeting. Members may, of course, request to keep or change their passwords at any time by contacting the ABBS System Operator (SYSOP) Tom Warrick through the office at 654-8060 or at home at 656-4389. If your number is not listed here, you need take no action.

More than 790 members have passwords. Between 55 and 60 members call in each day.

Membership numbers may be found on the second line of the label attached to the back of the Washington Apple Pi Journal. The membership number follows the "WAP#" and precedes the virgule ("/"). The last four digits are the year and month of expiration of membership.

Members whose passwords will be deleted unless they contact Tom Warrick at 654-8060 or 656-4389 are:

304	827	935	1142	1375	1386	1454	1586	1714
1874	1896	1953	2064	2129	2220	2294	2419	2428
2453	2580	2887	2892	2922	2972	2973	2976	3006
3088	3137	3219	3293	3347	3372	3404	3415	3457

PROGRAM PREVIEWS

by Cara Cira

February 25, 1983 - Personal Instrumentation and the Apple - Tom Riley

Would you like to get some insights into using your Apple for a science fair project? Curious about scientific research applications or industrial research and development with your Apple? Tom Riley will be taking us through the different levels in using personal instrumentation and how it applies to the scientific method in today's changing technology as well as future trends. The Quad Thermometer will be discussed in relation to experimental design, instrument calibration, data logging, data reduction, report generation and graphics generation.

Tom received his Bachelor's degree from the University of Houston in 1969 in Electrical Engineering. He is currently designing computer controlled testing equipment for a leading solar energy firm.

March 24 - Information Services. George Kinal and Bill Cook have put together a program representing different services available to the computer owner. Multi-function information services, stock market information, and research information sources will be discussed using a panel format.

April 28 - Memory Boards. For all the things you've wanted to know about memory boards - this one's for you. By membership request.

May 26 -

June 23 - Garage Sale.

TWO PROGRAMS ABOUT PROGRAMS:
A Review by Bob Oringel

A couple of years ago, when I purchased my Apple][+, I received my earliest training in its operation from a three-audio-cassette course entitled, "How to Operate the Apple Two Plus", by FlipTrack Learning Systems, of Glen Elyn, IL.

I have just immersed myself in two additional instructional programs by FlipTrack. The first is "How to Use Apple Writer", and the second is "How to Use VisiCalc". Each of these instructional programs consist of three or four audio cassettes, and a printed User's Guide. The user must have an audio cassette player, NOT hooked up to the computer. The FlipTrack option on the audio tapes suggests to the listener that from time to time, on a tone signal, he flip the cassette over and go to another tape track for additional information on a subject, in-depth information for specific users, or for useful optional quizzes on the material just presented. From the vantage point of a former Communications teacher, this is an excellent method of instruction.

Inasmuch as this reviewer has used Apple Writer II as his primary word processing program in the writing of two published textbooks, the decision to look at, or more properly listen to, "How to Use Apple Writer" came easily. A couple of very mild complaints at the outset: The audio cassettes were clearly duplicated on a high-speed duplicator. There is then the slightest hint of over-modulation, which sounds like echo on the tape. Also, the tape in the cassette tended at first to hang up, until gently manually freed with a pencil. These minor things aside, I jumped into the "Apple Writer" progam. It is a three cassette course which begins with a review of the disk initialization process and then proceeds to booting Apple Writer, and discusses the program's menus. Its very thorough presentation at first causes some confusion. It describes the use of Apple Writer II on the][+, //e and ///. The listener quickly learns however to "tune out" references to everything but his computer.

The course continues in a most friendly way to describe and explain the construction of text files; saving and loading files; picking up, moving and retrieving portions of files; using Tabs, Find and Replace, Glossaries, and all the other things that make Apple Writer II such a gem to anyone who creates text files. The description of Apple Writer's print program and its use was made very clear on the cassettes. FlipTrack winds up the course with a fairly weak description of WPL, Apple's Word Processing Language which permits, among other things, the construction of personalized form letters. I would have like to have heard more about WPL.

I must admit here that I picked up a few pointers, and I had thought I already knew Apple Writer intimately. The printed documentation, The User's Guide of 46 pages including a reference index, is well done and follows the audio in excellent fashion. I would recommend "How to Use Apple Writer" most enthusiastically to newcomers to word processing.

And now to "How to Use VisiCalc". I examined this course from the viewpoint of someone who has rarely used a spreadsheet program. The four audio cassettes covered the entering of information such as labels, value and formulas into the spreadsheet cells; building a VisiCalc model worksheet - in this case a cashflow worksheet; and it followed through with the analysis of results and the use of advanced worksheet techniques. Along the way it very thoroughly covered Replication, a concept which formerly confused me but no longer does. In carrying the practice cashflow worksheet throughout the course, it covered all the VisiCalc commands by describing their functions and leading the learner through the use of each command.

In the analysis phase, "How to Use VisiCalc" carefully leads the listener through the use of split screen "windows", synchronized scrolling, using Insert and Delete, Moving, and Built-In Functions. Saving to a printer as well as to disk is covered. The advanced techniques cassette describes worksheet recalculation, the recognition of forward and circular references, using the NVP, Choose and Lookup functions, and concludes with the methodology for making a VisiCalc template. A very thorough and clearly presented course indeed. This course too has a 55 page text with index, which follows the audio presentation. And once again, I would recommend "How to Use VisiCalc" by FlipTrack Learning Systems to anyone with interest in learning VisiCalc.

HARDVIEWS

by David Morganstein

VIEWMASTER 80. What gives your Apple an 80 column display, looks like a Videx board to Pascal and CP/M and costs about half the price of the Videx board? You will not be surprised if I answer "Viewmaster 80" from Applied Engineering. The VM is easier to install than the Videx board, as the soft-switch feature (automatic shifting between 40 and 80 column modes) is built in to the card. You need only insert the card in slot 3, connect your monitor to the card and run a patch cord from the card to the Apple monitor output. That's it! No ICs to pull and extra wires to connect. Whats more, Applied Engineering gives a three year warranty on their products. Another positive aspect of the card is the use of clearly marked, off-the-shelf, integrated circuit chips. For those capable of trouble-shooting a hardware problem, this will be an asset.

The card comes with a brief twelve page manual containing much less information than found in manuals from competing hardware. While it is probably possible to design a new character set and burn it into one of the on-board EPROMs, no mention of this is made. However, a user will have little trouble making the card perform in the DOS, Pascal or CP/M operating systems. Applied Engineering, Box 470301, Dallas, TX. 75247. Price $169.00.

Accelerator II. Ed Knepley's article on Speeding Up Your Apple in the October 1983 issue whet my appetite for a faster Apple. Gordon Stubbs graciously lent me his Accelerator II card so that I could watch the bytes fly faster. Indeed they do! The card contains a 6502B, a version of the 6502 chip which can run 3.6 times faster than the standard microprocessor in your Apple. Since the Apple's RAM is not designed to run that fast, the Accelerator II contains 64K of higher speed memory, as well. (I am waiting to hear the details about a possible competitor, the Speedemon, that alleges the Apple's memory chips can run faster.)

The Accelerator, like the Z-80 board, can be plugged into any slot. There are a few limitations about its slot position as it relates to the slot of various peripheral cards. First, the Accelerator needs to be in a slot to the left of a Z-80 board, if one is present. Second, all cards that are time dependent (like MODEM cards, disk controllers and video digitizers) must be in slots numbered four and higher. This presents a small problem for UCSD Pascal users since that system is looking for the MODEM in slot 2, its "traditional" home. I should point out that the Accelerator will be of no benefit to CP/M users. In fact, it must be "disabled" in order to access a Z-80 board.

Using the Accelerator is very easy, just turn on your computer and it is up and running. You will have to make a decision about the way in which the program you are going to run uses memory. The Accelerator must know if a standard 48K RAM is to be used, if a 16K RAM card is needed or if the motherboard ROMs are needed. You convey these decisions to the Accelerator via a pre-boot process. If you do not know the answers to these questions, Titan Technologies (formerly Saturn Systems...soon to be Universal Utilities...just kidding Titan!) has included a list of several hundred commonly used programs and the required pre-boot process. All in all, it gives your Apple a tremendous boost of capability.

I must be critical of one design decision. Most of the chips on board the card are labelled with a Titan five digit serial number and not the IC descriptor. If you have trouble with the card your only option is to return it for repair. There is no schematic or documentation on what to do if a problem arises. For many, this would be an unacceptable situation, especially for an expensive item like this one. In the current, fluid world of computer manufacturing, who can be sure that there will be support for a product a year from now. Titan Technologies, 310 W. Ann, Ann Arbor, MI. 48104. Price $599.00

TAX PREPARATION WITH COMPUTERS

by Roy Rosfeld

Last year (Pi Journal April 1983) I discussed how you can use your computer to prepare your tax schedules and forms and I gave you two sample schedules that are acceptable to the IRS. Since our membership has grown dramatically since last year, it appears prudent to repeat some of what I said in that article. I referred there to Rev. Proc. 83-9, IRB 1983-6, 2/7/83, page 4; and Rev. Proc 83-10, IRB 1983-10, 2/7/83, page 23. There will be a new Rev. Proc. issued this year but the date is uncertain. I have received an advance copy and the schedules are quite similar to last year. There are, of course, changes to reflect law changes, such as the 5% medical expense limitation and the 10% credit for working couples. Otherwise, the schedules look almost identical. You may order the new Rev. Proc. from the IRS or you may go to your local library and copy whatever portions you need. At any rate, read the material carefully since there are some fairly rigid guidelines. I will try to highlight some of the more important ones below.

I have prepared and presented to the club librarian an updated disk which contains the data listed below:

1. Schedule A - Itemized Deductions
2. Schedule B - Dividend and Interest
3. Schedule C - Business Income and Expenses
4. Schedule D - Capital Gains and Losses
5. Schedule W - Working Couple Deduction
6. Schedule SE- Self-Employment Tax Computation
7. The complete Federal form 1040 model with Maryland.
8. Tax tables (including a new four dimension table)
9. Tax worksheet plus form 1040

The schedules can be run on a 64K machine, but the complete models will require something more. If you have a 64K system you will have about 34K worth of memory after you boot up VisiCalc. The Worksheet model requires about 54K and the complete model in the same vicinity. I tried to condense the data but this loses the benefits of having the computer do all the work for you. The ideal memory configuration is around 128K to be able to create a truly complete and automatic tax model. I have "scattered" some of the tax tables so that you can experiment with creating your own model by adding segments as your system will permit.

As for the Rev. Procs themselves, there are some guidelines. You cannot computer produce the form 1040 itself (with certain exceptions not relevant here). You can computer-generate and submit the one listed above with the following caveats. You are supposed to use a dollar format such as (a) 0,000,000.00 or (b) 000,000.00, but since I haven't found a way for VisiCalc to do this I have ignored the commas (so far I haven't had any inquiries). There is also a requirement that computer-generated substitute forms be submitted on quality lined, machine paper, but lined, color-barred paper is also acceptable. If your printer has underlining capability, you may underline every other text line on the form. The line must be continuous and extend from the left text margin to the right text margin and be the same length for all lines on the page.

You must also have margins at least as large as the official forms themselves. This translates to about a one-half to one-quarter inch margin across the top, bottom, and both sides. When printing, just be sure that your form is centered so that the margins will reflect enough space. The guidelines call for spacing of 6 lines to the inch.

You are permitted on Schedules A and B to use supporting statements to give details of numerous entries, but they may not be used as a substitute for a collection of lines. If you have many items of dividends or interest, for example, you may show them below as "STATEMENT A", list them, total them and then put the total in the proper place on the schedule, for example, on line 4C on Schedule A.

Your forms should be assembled according to the number in the upper right hand corner of the particular form. For example, Schedule A is number 07 and Schedule B is number 08.

The IRS strongly urges you to round off the figures to the whole dollar amounts; however, although you may omit printing the cents, they still want the decimal point and 00's shown.

That's about all, sports fans! Hope the disk and this little dissertation is of some help to you.

TRI-STATE SYSTEMS

DISCOUNT APPLE SOFTWARE

MONTHLY SPECIALS

	Retail	Our Price		Retail	Our Price
Zaxxon	39.95	27.50	Screen Writer II	129.95	89.95
Choplifter	34.95	24.50	Master Type	39.95	27.95
Kraft Joysticks	64.95	46.95	Bank Street Writer	69.95	49.95
Choplifter & Kraft Joystick	99.90	67.95	Snack Attack	29.95	19.95
Exodus: Ultima III	54.95	38.95	Galactic Adventures	59.95	42.95
Frogger	34.95	24.50	Doss Boss & Utility City	53.50	37.95
Lode Runner	34.95	24.50	Verbatim (10 disks) 5¼"	42.00	26.95

GAMES

	Retail	Our Price
Wizardry	49.95	37.95
Wizardry II-Knight of Diamonds	34.95	25.95
Wizardry III-Legacy of Llylgamyn	39.95	29.95
Wizardry I & II	84.90	59.95
Castle Wolfenstein	29.95	20.95
Thunderbombs	19.95	14.95
Transylvania	19.95	14.95
Spy's Demise	19.95	14.95
Pie Man	19.95	14.95
Serpentine	34.95	25.95
Zork I, II, or III	39.95	29.50
Star Blazer	31.95	22.95
Repton	39.95	29.50
Pinball Construction Set	39.95	29.50
Deadline	49.95	37.95
Aztec	39.95	29.95
Cytron Masters	39.95	29.50
Starcross	39.95	29.50
Prisoner 2	32.95	24.95
Robot War	39.95	29.50
Apple Panic	29.95	20.95
Battle for Normandy	39.95	29.50
Laf Pak	34.95	25.95
Sammy Lightfoot	37.95	27.95
David's Midnight Magic	34.95	24.95
Ultima I	39.95	29.95
Ultima II	59.95	44.95

UTILITIES

	Retail	Our Price
Home Accountant	74.95	56.95
Sensible Speller	125.00	89.95
ASC II Express: The Professional	129.95	94.95
Transend I	89.00	69.95
Visicalc, Visifile, or Visidex	250.00	169.95
Multiplan	275.00	169.95
PFS: File, Report, or Graph	125.00	89.95
dBase II (CP/M)	700.00	414.95
The Incredible Jack	179.00	134.95
The Accountant	129.00	94.95
DB Master	229.00	149.95
Snooper Troopers I or II	44.95	32.95
Facemaker (Spinnaker)	34.95	25.95
The Graphics Magician	59.95	44.95
Beagle Basic or Touble-Take	34.95	25.95

HARDWARE

Microsoft 16K RAM Card	99.95	69.95
Rana Elite I Disk Drive	355.00	279.95
Hayes Micromodem II w/term Program	409.95	289.95
TG Select-a-port	59.95	44.95
System Saver Fan	89.95	69.95
Grappler II + ($179 w/16K Buffer)	175.00	129.95
Videx Ultraterm	379.00	279.95
Amdek Color I Monitor	379.00	294.95
Amdek 300A (Amber) Monitor	199.00	159.95
Okidata Micro 92 Printer	699.00	495.00

ALL ORDERS SUBJECT TO AVAILABILITY OF MERCHANDISE

Mail orders to:

TRI-STATE SYSTEMS

P.O. Box 2544
Springfield, VA 22152

or call: (703) 569-9363

VISA

No credit card penalty!

MasterCard

- We accept VISA or Mastercard (include card # and expiration date), personal checks (allow 2 weeks), or money orders
- VA residents add 4% sales tax
- Include $2.00 for shipping
- All software products on disk
- All prices subject to change without notice
- **We carry a large line of discount software. Write us for our free catalog.**

THANK YOU FOR YOUR ORDER!

Q & A

by Bruce F. Field

Jerry Battenhouse's letter in the November 83 WRITE-A.P.P.L.E. column has done wonders for my mail. For those of you who missed it, it was a continuation of the discussion started in this column about doing superscripts and subscripts with the C. Itoh 8510AP and Apple DMP printers. Jerry correctly pointed out that with the Apple printer it is not possible to produce superscripts and subscripts by moving the printer platen up and down in the middle of a line. Whenever a linefeed command is sent, the printer also performs a carriage return. It is possible however to only move the platen using the C. Itoh printer. First put the printer in the incremental print mode by sending "ESC [" to the printer; then forward and reverse linefeeds may be accomplished without a carriage return. This is one major difference between the C. Itoh and the Apple printer. This limitation rules out using most word processors to move the platen up and down for superscripts and subscripts. The Gutenberg word processor (and I assume the Gutenberg Jr.) can do superscripts and subscripts with either the Apple DMP or the C. Itoh 8510AP. I am not exactly sure how they work, as there are two ways of getting around the above limitation with the DMP. It is possible to print all the superscripts first, then the main text line and then all the subscripts. This would require three passes for the line. Or, an alternate character set can be downloaded into the Apple printer (true also for the 8510 if it has the extra buffer). In this case small characters on a 5 by 5 matrix can be built to print at the top of the line or the bottom. They don't really extend beyond the regular text but because of their size are taken for superscripts or subscripts. This is the technique that EPSON uses with their MX-80 and Graphtrax Plus option. There is one program I know of to download characters, FONT DOWNLOADER & EDITOR by Rak-Ware and it is advertised in most of the Apple magazines.

Q. This is mostly a question about preventive medicine since the darn thing has never failed in six months or so of use, but ???? Hex dumping an Applesoft program is very instructive. To make it easier, I wrote the following TEXT file:

```
PRINT "TURN ON THE PRINTER AND PRESS SOMETHING"
FOR X=1 TO 128 : X=PEEK(49152) : NEXT :
      POKE 49168,0
PR#1
PRINT
PRINT POKE 60,PEEK(103) : POKE 61,PEEK(104) :
      POKE 62,PEEK(175) : POKE 63,PEEK(176) :
      CALL -589
PRINT
PR#0
```

All that needs to be done is LOAD <programname> and EXEC APHID ("APHID" is the name I saved the TEXT file under -- would you believe "Applesoft Program Hex Instructional Dump"?).

As I have said, it has worked 100% for me for many months. But, I read recently in WHAT'S WHERE IN THE APPLE that the Y-register has to be initialized to 0 before the CALL to address -589. (This would be easy to do: insert the three commands CALL -182 : POKE 71,0 : CALL -193 in front of the CALL -589.) Is this really necessary in light of my success? Why, or why not? Is it possible that the Y-register will automatically have the right value in this context? If so, how can I (primarily an Applesoft programmer) know what is safe and what is not?

A. Let's start at the beginning and see what happens. The routine you call at -589 is a monitor routine that takes the starting and ending addresses from locations 60,61 and 63,64 which you have set to the beginning and end of your Applesoft program. It is not true that the the Y-register must be set to zero for this to work properly; the first thing the routine does is to set both the X and Y registers to their proper values. It appears that WHAT'S WHERE IN THE APPLE is incorrect on this point.

It is not generally reasonable to assume that any of the registers will be set to the same thing every time you call a machine language routine; what they contain will depend on the previous sequence of instructions. If we assume that the Y-register does indeed have to be set to something specific, your CALL -182 etc. sequence will not work. The problem is that between the CALL -193 and the CALL -589 that follows, Applesoft regains control and immediately changes the Y-register (also the A and X-registers). If you wish to set a 6502 register there is no simple universal call to do this from Applesoft. One way is to write a short machine language program that sets the register and then jumps to the machine language routine you really want to run. Then you can call your new routine from Applesoft.

As for your last question, the only way you can be sure that the registers are set properly is to explicitly do so or learn how to read assembly language programs (the former is less painful).

Q. I am using Apple Writer II to print a manuscript but no matter how I set the top and bottom margins the first page always comes out with the wrong number of lines. Is there a bug in Apple Writer?

A. The problem is undoubtedly that you are changing the top and/or bottom margins from within Apple Writer using the (.) dot commands. By the time you execute the .tm (top margin) command it is already too late to do anything about the top margin on the first page. Apple Writer uses the default print values in the Print Command Menu until dot commands are encountered in the text. To make the first page come out correctly set the default print values to the same values that the dot commands specify. This procedure is explained in the Print section of the Apple Writer manual.

This problem also shows up if a header is used and the left margin is changed at the beginning of the file. The header on the first page is printed according to the default print values and may not line up properly with the remaining text. The header continues using the value for the left margin that was set for the first page, so changing the left margin on succeeding pages will not affect the header.

Q. I have the //e with the extended 80-column card and Word Star version 3.01P. I'd like to use the auxiliary memory as a pseudo-ramcard in which to stash one or more of the Word Star COM and OVR files, thereby freeing disk space and hopefully

contd.

improving Word Star's rather geriatric performance on the Apple. How do I access the additional memory under CP/M, and what patches to Word Star and changes to CP/M are necessary to give ramcard-like results?

A. (What I know about CP/M will fit in a nutshell so Murray Arnow was kind enough to answer this question.)

This requires an involved reply. I will briefly state that there is no simple way to alter the BIOS to use the extended 80-column card memory to become a pseudo-disk. I seem to recall that there may be commercial packages available to perform the required patching, but I am unable to give any recommendations. There is a simple way to speed up Word Star. Create a Word Star disk by first formatting a new disk and then PIPing over the Word Star files in the following sequence: first the Word Star overlay files and then the WS.COM file. The Word Star files should appear first in the disk directory. If the latter is done, then disk access time can be significantly shortened.

Q. I would like to change file name used for disk boot up without initializing a new disk. Is there some way to do this?

A. As you know, when a disk is initialized a greeting program (usually named HELLO, although it doesn't have to be) is saved on the disk. The easiest way to boot up with a different greeting program is to rename or delete the original and save the new program under the same name as the original. If for some reason you actually want to change the name, you can do this too if you have a disk zap program. This is a program that allows you to read any individual sector on the disk, modify it, and write it back to disk. The greeting program name is stored on track 1, sector 9, starting at the $70th location. Read this sector into memory, change the name to what you want, and write it back to disk. It's a good idea to make a backup before doing this, as it is fairly easy to make a disastrous mistake and permanently wipe out some useful information.

Q. When drawing on the hi-res screen with HCOLOR=5 the X-coordinates with X=13+14 i (0<=i<=19, e.g. 13, 27, 41, 55, ...; I will call these peculiar) seem to be somehow unique. If I plot a vertical line at a peculiar and another one at a normal position, the first one is darker than the second one. I am able to observe this on my monochrome and RGB equipment.

```
Try : HGR : HCOLOR=5
      HPLOT 13,0 TO 13,100
      HPLOT 43,0 TO 43,100
```

If I plot a third line to the left of the peculiar line, nothing strange will happen. But a third line just to the right (HPLOT 15,0 to 15,100) will make the peculiar line brighten up. Do you know, whether the above can be seen on all Apple IIs, or is there something wrong with my Apple?

A. There is nothing wrong with your Apple; this is "normal" for all Apple IIs and //es. Your peculiar lines result whenever your plot in the last dot position of an even byte (this is what you call a peculiar line) and the next byte to the right has the color-shift-bit set opposite of the color-shift-bit in the even byte. Each byte in the graphics memory has 8 bits, 7 of which are displayed to the screen, and the 8th is the color-shift-bit. If the color-shift-bit is off (HCOLORs 0 through 3) then the next 7 dots can be used to produce either green, violet, or white. If the color-shift-bit is turned on (HCOLORs 4 through 7) then the next 7 dots can be used to produce either orange, blue, or white. If the even byte has the color-shift-bit on and the next byte has the color-shift-bit off the line is fainter than normal. When you plot a line in the next byte that turns the color-shift-bit on, even HCOLOR=4 which plots a black line, the peculiar line returns to normal. Conversely if the even byte has the color-shift-bit off and the next byte has the color-shift-bit on, the peculiar line will be brighter than normal. I can't explain exactly why this happens, but it appears to be a bug in the Apple graphics hardware.

An excellent article, Color Me Apple by M. A. Iannce, appeared in the November 1982 issue of Call-A.P.P.L.E. that explains most of the hi-res graphics idiosyncrasies in layman's terms. I recommend this article to those of you who want to know how your Apple really generates colors.

VOLUNTEERS NEEDED

A pilot project to provide economically disadvantaged Montgomery County youngsters with access to microcomputers outside of the regular school day will start in January, 1984. Elementary age students will use the Gaithersburg High School computer center two afternoons a week from about 3:00 to 5:30 PM.

The project is jointly sponsored by the Housing Opportunities Commission (HOC), the Montgomery County Public Libraries, and the school system. The goals of the project are to:

- develop computer use skills,
- promote use of computer technology to support learning in and out of school, and
- make students aware of the role of computers in a rapidly changing society.

Ms. Ruth Spector of HOC attended an Apple Pi Board of Directors meeting and described how volunteers would use the computer literacy curriculum of the school system to help students identify projects and activities. Volunteers will be in the role of mentors, generally helping individual students by coaching, tutoring and demonstrating hardware and software.

Volunteers should have skill in the use of the Apple //e. Epson FX-80 printers will be used with the APIC-G graphics interface card. All machines have two disk drives and 80 column green screens. In addition, one of the 15 student stations has a D.C. Hayes Micromodem. Three color monitors are also available. Knowledge about word processing and other kinds of applications software is desirable. Poten- tials mentors should enjoy working with children on a one-to-one basis and be able to put together creative demonstrations for small groups.

Interested members of Apple Pi should contact Ruth Spector (933-9750), 10400 Detrick Avenue, Kensington, MD 20895.

HARDWARE HELPERS

by J. Tom DeMay, Jr.

Have you ever had a problem with your APPLE? Was it a hardware or software problem? Maybe it was just a "cockpit error" (a less offensive way of telling someone that you didn't know what you were doing). Maybe you are not sure of the difference between hardware and software. The easiest way to differentiate is: "If you can touch it, it's HARDWARE. If you can LOAD, RUN, or SAVE it, it's SOFTWARE." As in most cases there is a gray area. It is something that is SOFTWARE and also fits my definiton of HARDWARE. It is called FIRMWARE, and consists of software permanently (or semi-permanently) saved on a special type of memory chip called a ROM (Read Only Memory), PROM (Programmable ROM), or EPROM (Eraseable PROM).

If you have difficulties with your APPLE, the first thing to do is to determine which category best defines the problem: Hardware, Software, or "Cockpit Error". The latter is most easily corrected. "If all else fails READ THE MANUAL" is one of my favorite expressions because it is so true. We all share that feeling of excitement when we purchase a new program (software), or hardware device. I must admit the reluctance to READ THE MANUAL first. I would rather GET IT UP AND RUNNING. Reading the manual before connecting a piece of hardware, or before BOOTing a new software package can save you considerable aggravation. Re-reading the manual occasionally is also a good idea. You may find something that meant little at the first reading, but is important to you now. A good example is the SCREENWRITER][manual. There are many options available, and re-reading the manual has kindled more than one idea.

The next step is determined by which type of problem you are having. If it seems to be software, try the program on a friend's computer. Be sure to use one which has the same configuration as yours, or the results may be misleading. Software problems can cause all sorts of strange things to happen. It is a good idea to cycle the power (Turn it off. Wait a few seconds. Turn it on again.), then BOOT the program before attempting troubleshooting. Residual software problems (those leftover from a previously loaded program) can be difficult to diagnose. You may find that you have been "chasing balloons". There may not be a problem at all. If you suspect the program you are attempting to RUN will not work with some leftover setup, cycling the power will reset the computer to its original state. This is different from pressing the RESET button which just performs a warm boot, and may not clear the problem.

A hardware problem can be just as tricky. There are several things to try to correct a hardware problem. The first and most obvious is to turn off the power and verify that all connections are intact. Check the power plug at the wall outlet and at the rear of the APPLE. Don't forget the outlet strip, or surge protector if you are using one. Turn off the power and remove the cover from the top of the APPLE. Touch the power supply with your fingers. It is the big metal box to your left. This will drain the potentially hazardous static charge your body may have picked up while walking across the floor. This is especially important in the winter when the air in most homes is less humid. The shock you experience when touching a metal door knob, while unpleasant to you, may be deadly to your APPLE. The body cannot detect a static discharge less than 3 or 4 thousand volts. Therefore, if you were to feel a shock when touching an integrated circuit, there is a good possibility that the chip will be destroyed. Static discharge is one of the biggest enemies of electronic circuits, especially considering that they are designed to work on from one to twelve volts and static charges may accumulate in excess of 50,000 volts. Touching the power supply will harmlessly dissipate any charges. An occasional misting of the area around your APPLE with an anti-static spray will help to lessen the possibility of an accidental static discharge.

Now that you have dissipated the static charge, carefully remove the cards from their slots one at a time. Apply a small amount of liquid contact cleaner to the contacts at the edge of the card and wipe it gently with a lint-free cloth. If you don't have contact cleaner, then tape head cleaner, or alcohol will do in a pinch. Please do not use rubbing alcohol. Some of these contain oil and will do your APPLE more harm than good. Some people suggest using a clean pencil eraser to remove the oxidation from these contacts. If you do, use a light touch. Some of the gold may also be removed. Gently insert the cards.

While the top is off, closely examine the integrated circuits (black rectangles with lots of feet). Some of them are soldered in, and some are inserted in sockets. These have a tendency to work loose from the expanding and contracting of the sockets because of heat. A firm, steady pressure may reseat those that have worked loose.

Replace the cover and BOOT the APPLE as you normally would. Try the program or procedure you were having a problem with. You may be in for a pleasant surprise. If this dosen't work, WAP may have just what you need: a "HARDWARE HELPER". Several members have offered their services as diagnosticians. Look through the accompaning list to find the one closest to you. Give him a call to arrange a mutually agreeable time for you to meet. He may suggest that you take your APPLE to his house, or he may wish to visit your "sick" APPLE. He will have a series of diagnostic programs, supplied by WAP, which can be used to determine the cause of the problem. It will be at the discretion of the HARDWARE HELPER just how involved he becomes. He may only suggest things for you to do, or he may actively assist in cleaning, removing or replacing parts.

We intend to maintain a database of problems and solutions. It will be analyzed to evaluate the HARDWARE HELPER project as well as give some idea of the most common APPLE problems and solutions.

If your area is not included in the list of "HARDWARE HELPERS", and you feel "comfortable" inside an APPLE, please call me and we can arrange for you to help. I can be contacted via the WAP ABBS, through the WAP office, or by phone at (301) 779-4632. If I can help don't hesitate to call. Washington Apple Pi invites your participation.

Washington Apple Pi - HARDWARE HELPERS

Mark Pankin
1018 North Cleveland Street
Arlington, VA 22201
703/524-0937
hours (till 10)

Dave Weikert
17700 Mill Creek Drive
Derwood, MD 20855
301/926-4461
hours (7-10 except
Thursday & weekends)

contd.

Jim Taylor
16821 Briardale Road
Derwood, MD 20855
301/926-7869
hours (till 10)

Lyman Hewins
Route 2, Box 26
Leonardtown, MD 20650
301/475-9563
hours (till 11)

Tilghman Broaddus
Rt 1, Box 246
Mechanicsville, VA 23001
804/779-2553
hours (till 10)

Pete Jones
1121 N. Arlington Blvd.
N. Arlington, VA 22209
703/527-5464
hours (7-10)

J.T. (Tom) DeMay, Jr.
4524 Tuckerman Street
Riverdale, MD 20737
301/779-4632
hours (till 11)

Bruce Field
1402 Grandin Avenue
Rockville, MD 20851
301/340-7038
hours (till 10)

Richard Rowell
1906 Valley Stream Drive
Rockville, MD 20851
301/770-5260 hours (7-11)
202/651-5816 hours (9-4)

Gene Cartier
6026 Haverhill Court
Springfield, VA 22152
703/569-8450
hours (till 10)

GAME CONTROLLER TESTS

by Rudie Slaughter

Here's a nifty little two-line program that provides for a really quick way to test whatever is connected to the analog inputs, $C064-C067, the flag inputs, $C061-C063, and the cassette input, $C060. The analog inputs and flag inputs are usually connected by means of the game controller port to either paddles or a joystick. The common method of obtaining a fourth pushbutton by wiring it to the cassette input port is recognized by the program. The program does not involve any DOS tricks, hi-res, or sound; only a little bit of Boolean algebra to turn inverse on and off. The program provides for an inelegant exit as the keyboard strobe is not cleared when the program is terminated. Being as short as it is, the program does not need machine language coding to keep up with input changes.

To enter the program in only two lines, type it in exactly as shown (in 40 col. mode), substituting a blank for each underscore. The line two strings must be in CAPS only, although Apple //e user's can enter lower case in the line one strings.

```
]1TEXT:HOME:?TAB(12)"Game_Controller_Test":?:?__

Inverse_indicates_corresponding_buttonis_pushed_

in.Cassette_input_substitutesfor_Button_3._Inverse

_and_Paddle=255_areno_input_defaults._Any_key_exits

test.":FORI=1TO5:A$=A$+CHR$(8):NEXT

]2POKE-16368,0:VTAB10:FORI=0TO2:POKE50,(255-192*(PEEK

(-16287+I)>127)):?"__PADDLE_"I"_READS_____"A$;PDL

(I):?:FORJ=1TO10:NEXTJ,I:POKE50,(255-192*PEEK(-16288)

>127)):?"__PADDLE_3_READS_____"A$;PDL(3):NORMAL:IF

PEEK(-16384)<128THEN2
```

MONTHLY TUTORIAL OUTLINE

The following outlines are used in our four monthly beginning tutorials, which are given on Tuesday evenings at the office, from 7:30 to 9:00 PM. A registration form is in the back of this issue.

SESSION 1. Introduction to Apple Computer Hardware.

A. Welcome to the World of Apple

1. Hooking it up: disk drives, printers TVs & RF modulators, games paddles
2. Inserting & removing cards
3. The keyboard

B. Handling Floppy Disks

1. Reading a catalog
2. Running a program
3. Using the back of the disk

C. Handy Features

1. Added memory boards
2. Eighty columns
3. Lower Case
4. Fans

SESSION 2. How to Use Your Apple Software

A. Booting a Disk: PR#6 and Friends
1. The Autostart ROM
2. The Disk Drive
3. Run the "HELLO" Program
4. The Worst That Could Happen: There Goes $2.90 Down the Drain

B. CATALOG: What's on This Disk?

1. The File Name (30 characters)
2. Locked or Unlocked?
3. File Types: A, I, B, T, R
4. File Size I: Sectors

C. Other Important DOS Commands

1. General DOS Syntax
   a. The command word
   b. The filename
   c. Other information (e.g. new filename, loading/saving address
   d. Slot, drive, volume
   e. Must start at the "left margin"
2. INIT filename: Initializing a Disk
3. LOAD filename: for Applesoft and Integer
4. RUN filename
5. SAVE filename
6. BRUN, BLOAD and BSAVE
7. DELETE filename
8. LOCK and UNLOCK
9. Text Files: a Very Brief Introduction

D. Useful Utilities

1. FID: File Developer
2. COPYA
3. BOOT 13 and MUFFIN
4. INTBASIC and INTEGER BASIC-DISK
5. The WAP New Member Disk
6. Disk Recovery Programs

E. For Further Reading

1. The DOS Manual(s)
2. Beneath Apple DOS
3. All About DOS

SESSION 3. Welcome to Applesoft Basic

A. What is Programming?
B. The PRINT Statement
C. Variables
   1. $ means string
   2. % means integer
D. INPUT Statement
E. Arrays
F. Immediate Mode vs. Stored Program
   1. Line numbers
   2. LIST, NEW, DEL
G. Changing a Program Line
H. RUNning a Program
   1. RUN
   2. CONT
   3. CLEAR
   4. END
I. Looping
J. Branching
K. Subroutines
L. Limitations and Restrictions
   1. Precision
   2. Garbage collections
   3. PEEK, POKE, CALL & and USR
M. Advanced Memory Usage

SESSION 4. Bits, Bytes, Nibbles

This tutorial is intended to describe what goes on in the machine underneath BASIC. Assembly language is not going to be taught but we will show how assembly/machine language programs can be entered from printed listings. Several commands of the Monitor program in the Apple will be explained.

A. Binary and Hexadecimal Number Systems

1. What is a bit, byte, or nibble?
2. Adding and subtracting binary and hex
3. Converting binary and hex to decimal
4. Using the Monitor to add hex numbers

B. The ASCII Character System

1. What is ASCII?
2. Converting control codes to decimal or hex

C. Using the Monitor Program

1. Examining and changing memory
2. Entering machine language programs
3. Listing machine language programs
4. The difference between assembly and machine language

D. Memory Map or Where Is It?

1. Text and Hires graphics pages
2. Free areas for machine language programs or shape tables
3. How a language card works; what it's good for

E. Using Applesoft Efficiently

1. How Applesoft is stored
2. How to use LOMEM and HIMEM
3. What an interpreter is and why it is so slow
4. How DOS works with Applesoft (or why you shouldn't use PR#1)

WAP TUTORIALS
by Stephen M. Stern

This year we are offering two types of tutorials: beginning level tutorials on the "A,B,C's" of the Apple and more advanced tutorials on specific topics and software packages.

The beginning tutorials are given several weeks a month at the club office in Bethesda on Tuesday evenings from 7:30 to 9:00 PM. A detailed outline of these tutorials and a registration form appear elsewhere in this issue.

WAP will offer five advanced tutorials between February and June of this year. Topics range from VisiCalc (Feb), VisiPlot (Feb), Advanced Applesoft (Mar), CP/M (April), dBase II (May) and Pascal (June). Detailed outlines for the next two tutorials appear below. Outlines for subsequent tutorials will appear in later issues. These tutorials will be held at the club offices in Bethesda on Wednesday evenings and Saturday mornings as specified on the registration forms in the back of this issue.

ATTENDANCE AT THESE TUTORIALS IS LIMITED TO 14 PER SESSION! LAST MONTH WE HAD TO TURN PEOPLE AWAY -- PLEASE REGISTER IN ADVANCE. Priority will be given to members over non-members and to attendees bringing their Apples over those without Apples. Attendees are advised to bring their Apples if they wish to maximize the learning experience. WAP will not provide Apples for the use of attendees, nor can refunds be provided (unless WAP cancels the tutorial).

Remember the person leading these advanced tutorials is an unpaid volunteer. If you believe you can contribute by leading a tutorial, please contact Steve Stern, Lee Raesly or the club office.

TOPIC: VISIPLOT/VISITREND - SAT, FEB 18, 9:30 AM-Noon
INSTRUCTOR: Lee Raesly

1) Introduction - trend analyses and plots

2) Visiplot
- -available plots
- -keyboard entry
- -plots and overlays
- -saving data to disc
- -saving plots to disc

3)Trend analysis
- -keyboard entry
- -trend analysis
- -saving data to disc
- -saving trend analysis to disc

4) Loading data fron Visicalc & other files

5) Printing and printer drivers

TOPIC: ADVANCED APPLESOFT - SAT, MAR 10, 9:30 AM-NOON
INSTRUCTOR: Richard Untied

This tutorial will include discussion of program design and structure, performance and testing, input/output and file handling techniques. The tutorial is aimed at those who have been using Basic for a while, and would like to learn additional techniques and approaches to typical programming problems. You will be encouraged to share your problems and experiences.

I. Introduction
   - A. Scope of the tutorial

II. Program Development Concepts
   - A. Functional requirements
   - B. Performance requirements
   - C. Top Down design
   - D. Modular approach
   - E. Tradeoffs
   - F. Documentation
   - G. Testing

III. Techniques
   - A. Making it faster
   - B. Making it smaller
   - C. Memory management
   - D. Text line input
   - E. File handling
   - F. ONERR error handling
   - G. Debugging

IV. Helpful Utilities
   - A. PLE and GPLE
   - B. Cross-reference
   - C. Optimizers
   - D. Compilers
   - E. Amper this-and-that

V. Discussion
   - A. Develop a short program from scratch.
   - B. Question and Answer session

ULTIMA II TIPS

by Nicholas G. Carter

ULTIMA II is an outstanding adventure, but it is almost too challenging for the player, like myself, who does not have an excess of time and an infinite amount of patience. I do not appreciate having to spend quite a bit of time and thought creating a character and then have to wait through the long boot of the adventure only to have him wiped out before he can get to the nearest town. Thus, I began my search for ways to improve my chances of survival so that I could enjoy this superb adventure without being in constant fear of a wipe-out. What follows is a guide to some of the internal codes; if you have a disk zap utility you can change your player master (your own, not the original) to suit the challenge you would like from ULTIMA II.

Get out your utility (I used BAG-OF-TRICKS) and take a look at Track $04 Sector $08 of your player master. Here is where ULTIMA II stores all of the current characteristics and scores of your player. This is actually an image of the vector in memory which the program uses continuously during the game as certain fields are used to indicate situations from which you cannot save the adventure. The first thing you should notice is the name of your character. This is in bytes 4-12, in hex format, and should be visible if your utility provides the ASCII translation. In byte 14 you will find the sex of your character, CD for male and C6 for female. Bytes 15 and 16 contain the type and race of the character in the order presented in the ULTIMA II handbook with codes ranging from 0 to 3. Byte 17 indicates the era in which the character is currently adventuring; 0=Legends, 1=Pangea, 2=BC, 3=AD, 4=Aftermath. Unless your character is very experienced, do not put zero in this byte!

Byte 18 is interesting. It tells the program whether the character is in a village, a town, or in the countryside. As you can only save the adventure when your character is in the countryside (byte 18=0), putting down another number here will result in your starting up inside a town which will disappear when you leave it.

Bytes 19 through 1E are the basic characteristics: strength, agility, stamina, charisma, wisdom, and intelligence. The values are decimal from 0 to 99. Watch out that you do not put these values too high, otherwise when your character legitimately earns an increase, he will exceed 99 and be put back to zero! Bytes 1F and 20 are the hit points, again in decimal form. Bytes 21 to 23 contain the amount of food available in hundredths of units, in decimal form. Experience and gold are located in bytes 24 and 25 and bytes 26 and 27.

The next thing the program needs to know is where on the map your charcter was when you saved the scenario. This is done with bytes 28 and 29. The first one contains the X coordinate measured from 0, the eastern end of the Bering Strait, around the map to 3F, the value at the western end. Similarly, byte 29 contains the Y coordinate ranging from the top of the world to the bottom.

The weapon your character has "ready" is indicated in byte 2F, with the codes ranging from 1 (=Dagger) to 9 (=Quicksword). His armour is shown in byte 30, with the codes ranging from 1 (=Cloth) to 6 (=Power). The spell is in byte 31 and ranges from 1 (=lights) to 9 (=kill). Bytes 32 to 34 indicate the number of torches, keys, and tools in his possession.

The weapons themselves are shown, decimal counts of numbers owned, in bytes 45 to 4D. Similarly armour is contained in bytes 65 to 6A, spells in bytes 85 to 8D, and finally the particular objects your character has picked up, including those needed to save leg paralysis, view, enter and ignite the rocket, and operate the airplane are in bytes A4 to B3. I leave it to your adventurer to figure out what they are and what they do.

The information presented above is nowhere near sufficient for your character to be able to go out and quickly win the game. There is a lot more that you need to discover and learn before you can tackle Minax! It will however enable you to give him enough of an advantage so that you will not be in constant danger of being wiped out every time you venture forth into the world of ULTIMA II.

THE 2% SOLUTION

by Peter Trinder

You may be interested to hear that I seem to have solved a little problem that has foxed me for some little while. When the //e came out I passed my][+ on to my son's school, and then found that one of the programs that I had used quite often didn't work on the //e. This was the enhancement for HIGHER TEXT called HTE from C & H Video of Hershey, PA. It did work but when I tried to center justify a line with Control J the cursor took off on a tour of the screen beyond my control! C & H were very nice about it, but couldn't provide a solution by re-writing, so I was stuck until we found this answer, which I have sent to C & H for their assistance.

The answer is to fool the //e into thinking it's a][+ for the duration. Take a spare disk, put FPBASIC from the][+ DOS 3.3 System Master on it and add the following HELLO:

```
10 PRINT CHR$(4); "BLOAD FPBASIC, A$D000
20 LCK = 12 * 4096 + 8 * 16 : REM $C080
30 POKE LCK,0 : POKE LCK,0
```

This loads the old Apple][ROM data into the top 16K so that programs which follow will refer to it rather than to the //e ROMs.

I didn't incorporate this HELLO onto the HTE disk because there isn't room for the FPBASIC and all the FONTS, so to RUN HTE just boot the above HELLO and the change disks and RUN HELLO on the HTE. What could be simpler! This, I am sure, could be the solution to many other users problems. I certainly hope so.

(Ed. Note: Peter is one of our several members from England. Thanks for the contribution.)

— Operant Systems —

-- HARDWARE --

DISK DRIVES--
- MicroSci A2 drive (100% Apple-compatible Shugart 390).. 239
- Rana Systems Elite One drive (40 track, 163K)..... 265
- Elite Two (40 track, double side, 326K)... 425
- Elite Three (80 track, double side, 652K). 549
- TEAC Thinline drive (40 track, 100% Apple compatible).. 259
- Indus GT half-height drive (40 track, Apple compatible) 249
- Davong, Corvus, and Corona Winchester drives.....(call)

CP/M SYSTEMS--
- Applicard (6 Mhz Z-80, 64K to 192K RAM, 70-col video).. 259
- Microsoft Softcard IIe (Z-80, 80 col & 64K on one card) 335
- Softcard (includes CP/M 2.2 and MBASIC)..... 229
- ALS CP/M Plus System (6 Mhz Z-80, 64K RAM, CP/M 3.0)... 275
- Titan Systems Accelerator IIC (3.6 Mhz 6502 processor).. 445

MONITORS--
- Amdek 300G (12" green anti-glare screen, 18Mhz)..... 145
- 300A (12" amber anti-glare screen, 18Mhz)..... 155
- Color I (40-column text/color graphics)..... 295
- NEC JB-1201 (12" green anti-glare screen, 20Mhz) 159
- JB-1260 (12" green, 15Mhz, best value for money)... 110
- USI PI-2 (12" green anti-glare screen, 20 Mhz)..... 145
- PI-3 (12" amber anti-glare screen, 20 Mhz)..... 155

MODEMS--
- Hayes Micromodem IIe (tone dialing/speaker/Smartcom I). 225
- Smartmodem 300 (300 baud, RS-232, direct-connect) 199
- Smartmodem 1200 (300/1200 baud, as above)..... 479
- Novation Apple-Cat II (w/ Conware; 1200 baud capable).. 245
- Smart-Cat 300 (RS-232, direct-connect)..... 175
- Smart-Cat 300/1200 (as above)..... 389
- J-Cat 300 (miniature RS-232, direct-connect).. 105
- SSM ModemCard (300 baud, audio monitor, emulates Hayes) 235
- US Robotics Password (300/1200 baud, auto-dial/answer). 355
- Anchor Automation Volksmodem (300 baud, RS-232)..... 59
- Mark XII (300/1200 baud, RS-232)..... 259

PRINTERS--
- Epson FX-80 (160 cps, tractor/single sheet, graphics).. 545
- FX-100 (wide carriage version of the above)..... 739
- RX-80F/T (100 cps tractor/single sheet, graphics) 409
- MX-100 (wide carriage version of the above)..... 575
- Okidata 92 (160 cps, graphics, BEST print for price)... 439
- 93 (132 column version of above)..... 715
- IDS Microprism (110 cps, prop spacing, excellent print) 479
- Prism-80 (200 cps, buffer, correspondence quality). 999
- Prism-132 (wide-carriage version of above)..... 1149
- NEC 8023 (120 cps, tractor/sheet, same as Prowriter)... 385
- Gemini 10X (120 cps, tractor/single sheet, graphics)... 315
- 15X (wide-carriage, same features as Epson MX).. 425
- Itoh F-10 Starwriter (40 cps daisywheel, Qume-compat). 1099
- Transtar 140 (40 cps daisywheel, same as Itoh F-10)..(call)
- 130 (18 cps daisywheel, BEST low-cost unit)... 615
- Silver-Reed 500 (14 cps daisywheel--goodbye typewriter) 439
- NEC 7710 (55 cps daisywheel, built like a tank)..... 1975
- Diablo 620 (25 cps daisywheel)..... 950
- Qume Sprint 11/40 (40 cps daisywheel)..... 1349

INTERFACES & BUFFERS--
- Pkaso printer interface (parallel w/ superior graphics) 135
- ShuffleBuffer (8K to 128K w/merge-print & much more).(call)
- Grappler+ printer interface (parallel w/ graphics)..... 120
- Buffered Grappler+ (up to 64K buffer plus graphics).... 169
- CPS Multifunction (serial, parallel, and clock)..... 165
- Wizard SPO (16K-32K buffered parallel w/ graphics)..... 135
- IPI (graphics, text dump, page formatting)..... 69
- CCS 7710A (RS-232 serial for printers & modems)..... 115

VIDEO & KEYBOARD--
- Videx Ultraterm (160 column/48 row display !!!)..... 259
- Videoterm (80-column w/ softswitch & inverse).... 215
- Wespercorp Wizard-80 (softswitch & inverse built-in)... 155
- ALS Smarterm II (80 columns at a bargain price)..... 129
- MicroSci 80-col card w/64K RAM for IIe..... 125
- Keytronic KB-200 low-profile detached keyboard..... 235

MEMORY EXPANSION--
- Wespercorp Wizard 16K RAM card (2-yr warranty)..... 63
- Titan Systems 128K RAM card..... 375

-- SOFTWARE --

WORD PROCESSING--
- Wordstar 3.3 (including 6 Mhz Z-80 Applicard !!!).. 399
- Wordstar 3.3 (program alone)..... 299
- ScreenWriter II (70 col display, spooling, more!).. 85
- Super-Text Professional..... 69
- Bank Street Writer..... 49
- PFS: Write IIe..... 85
- Word Handler + List Handler + Spell Handler..... 95
- HomeWord (easy-to-use icon-based system)..... 39
- Perfect Writer (edit/view multiple files in CP/M).. 175
- Incredible Jack (word processing, data base, calc) 135
- Sensible Speller IV (checks ANY file type)..... 89
- The Word Plus (super spelling checker for CP/M).... 119

SPREADSHEETS & FORTUNE-TELLERS--
- Multiplan (state-of-the-art spreadsheet)..... 169
- VisiCalc (IIC+ and IIe versions available)..... 165
- SuperCalc 2 (powerful CP/M spreadsheet)..... 189

INFORMATION MANAGEMENT--
- dBASE II (w/ ZIP screen generator)..... 409
- Quickcode (dBASE program generator)..... 189
- dUTIL (time-savers for the dBASE programmer) 70
- The General Manager 2.0..... 149
- DB Master (new version 4)..... 219
- PFS: File, Report, or Graph..... 85
- Infostar (including 6 Mhz Z-80 Applicard !!!)..... 399
- Data Perfect..... 99
- VisiFile..... 165
- VersaForm (business forms processor/database)..... 199

BUSINESS & ACCOUNTING--
- BPI Accounting (AR/AP/PAYROLL/INVENTORY) ea module: 249
- Peachtree Peachpak (GL/AR/AP)..... all three: 229
- Dollar & Sense..... 69
- The Accountant (double-entry bookkeeping system)... 99
- Home Accountant..... 55

COMMUNICATIONS--
- Ascii Express Professional (for DOS 3.3)..... 95
- Z-term Professional (for CP/M)..... 99
- Data-Capture IIe..... 69

LANGUAGES--
- Softech UCSD p-System..... 425
- Digital Research Pascal/MT+..... 249
- Microsoft Fortran-80..... 129
- BDS C compiler..... 115
- Micromotion FORTH-79..... 79
- Terrapin Logo..... 109

ODDS & ENDS--
- Kensington System Saver Fan (w/surge protector).... 66
- TG Joystick / Kraft Joystick..... 43/47
- Trace SoundTrap (acoustic enclosure for printers).. 99
- Wildcard (hardware antidote for copy protection)... 99
- Locksmith 5.0 (...the best gets better!)..... 79
- KoalaPad Graphics Tablet (with graphics software).. 95
- Verbatim, Maxell, Dysan, TDK, Wabash disks..... from 20
- Computer furniture (tables, printer stands, etc) (call)

Call for prices of items not listed.
We can supply HUNDREDS of other accessories, programs, and computer systems--most WITHIN 24 HOURS.
--) PLEASE COMPARE OUR PRICES (--
If you find a lower price, give us a chance to beat it.
Feel free to call for answers to technical questions

TO ORDER: Call JEFF DILLON at (301) 434-0405
or
Mail cashiers/certified check or M.O. to:

OPERANT SYSTEMS, 7676 New Hampshire Ave, Suite 312
Langley Park, Md 20783

Md. residents add 5% tax. UPS shipping is available.
All items carry full manufacturer's warranties and are fully tested by us prior to delivery.

WINTER DISK ROUNDUP Contd.

by Robert C. Platt

Here is the third installment of the Documentation Team's efforts. Included are a brief description of each program on the disk, known bugs, and hints on using the disk. Documentation for disks 12, 13, 19, 23, 25, 27, 100, 102, 105 and 122 appeared in the November WAP Journal; and documentation for disks 11, 16, 20, 34, 106, 107, 127, 132 and PIG7 in the December Journal. Each program has been given a unique serial number which is enclosed in brackets {}. The number is not a part of the program file name on the library disk. You can buy these disk at our meetings, the WAP office, or by mail using the form in the back of this issue.

DISK 24: GAMES IX
(Reviewed by V. Khera)

An (I) indicates an Integer BASIC program.

{24.1} CIPHER-TEXT - A self-prompting introduction to text ciphering.

{24.2} SUPER STAR TREK - Destroy the Klingons. Text, instructions included.

{24.3} ANIMALS FILE MOVER - The ANIMALS guessing game on the DOS System Master Disk (now called the Sample Program Disk) generates a file called ANIMALSFILE. This program will move that file to another disk.

{24.4} LIFE - A mathematical pattern game. Lo-res, but very quick graphics. Uses the LIFE.OBJ binary file. (I).

{24.5} ALIEN - A text game to hunt and kill an alien.

{24.6} ARTSHOW - A shape table demo.

{24.7} BLIZZARD - Catch the falling snowflakes in the lores graphics game. The game stops after 5 misses. The graphics are a bit jumpy. (I).

{24.8} MASTERMIND 2 - A text version of classic game.

{24.9} DOGBITE - You are the mailman trying to deliver ten packages.

{24.10} SAVE THE WORLD - Lo-res shoot at the moving target game. (I).

{24.11} LO*RES LIFE - Similar to LIFE.

{24.12} MAZE - Rearrange cards according to a set of complicated rules.

{24.13} BISLEY - Similar to MAZE but different rules.

{24.14} SUPER HOCKEY - A text hockey game. Instructions are included.

{24.15} ONE MAN PONG - Play against the computer which has a tremendous advantage. (I find the graphics jumpy and suspect that the computer cheats.) (I).

DISK 37: UTILITIES IX
(Reviewed by Ross Lenet)

With a couple of exceptions, the programs on this disk would be of interest primarily to programmers. A (N) indicates that the program is also found on Disk 134, New Members Disk. An (I) notes Integer BASIC programs.

{37.1} CATALOG MANAGEMENT - Intended as a greeting program, this Applesoft program displays a menu that loads or runs a program or displays a description of any program on a disk. (N)

{37.2} CATALOG MANAGEMENT-EDIT - Is used in conjunction with the previous program. This program creates or changes the program descriptions. (N).

{37.3} FLAW BITS - Can set user-defined "flaw bits," unset flaw bits, and perform similar functions. (I).

{37.4} GENERAL DISK UTILITIES - This poorly documented program is a disk read/zap utility that also performs other functions. (I).

{37.5} INTEGER RENUMBER - A self-explanatory program that renumbers programs written in Integer BASIC. (I).

{37.6} PRETTYLIST - This machine language program produces an attractively formatted listing of Applesoft programs. For details, BRUN PRETTYLIST INFO.

{37.7} PRINT...USING DEMO - Many versions of BASIC include a PRINT USING command, by which both numbers and strings can be printed in various formats. Although Applesoft does not include such a command, you can use the & command to call machine language routines to perform the same function.

{37.8} PROGRAM SECTORS MAP - Displays on the screen or printer the sectors associated with each disk file. An easy method to control the speed of screen output would have been nice. There seems to be a slight bug in this program: it does not pick up the last program in the catalog. (I).

{37.9} RESTORE - This program will restore a deleted disk file if it has not since been overwritten by newer files. All you need to do is specify the name of the deleted file. (I).

{37.10} SUPER FILE CABINET - This Applesoft program drives a small data base and features a fast machine language sort and other niceties. There is no on-screen help, however, if you are confused as to what to do at any given point. See WAP Journal December 1983 p. 60 for details. (N).

WAP.UTILITIES IX - This is the greeting program which displays the catalog.

DISK 38: GAMES XIII
(Reviewed by Doug Waters)

This disk contains a variety of games including one text adventure.

{38.1} JOURNEY TO CENTER EARTH - A text adventure game. You chose the direction of travel (N, S, E, W) and the screen tells where you are.

{38.2} COLOR STARTREK - Lo-res paddle-driven game that pits you against the Klingons. You must state the angle of fire to destroy the enemies, so it's a good drill on estimating angles.

{38.3} INVADERS - Similar to Space Invaders game. Paddles required. Features 10 skill levels. Your lazer shoots a 3 point valve on invaders, and your power charge decreases. Features cute instructions and a funny ending.

{38.4} LE MANS FOR TWO - A lo-res road race for two players. Two paddles required. Hard to see on a

contd.

green & black monitor. Game features 3 skill levels and takes points off when your car leaves the road. A final score is displayed for each car.

{38.5} INSULTS - A silly random insult generator that claims 40,960,000 insult combinations. Each insult is a one or two line sentence. Some words are considered R+ rated.

{38.6} YELLOW BRICK ROAD - A cute lo-res game. Good for developing hand-eye coordination.

{38.7} WIPE OFF - A lo-res version on the Brickout theme. Paddles needed. Choose colors, three levels of difficulty. Use the paddles to keep the ball from leaving its box. Five chances to succeed.

{38.8} LE MANS FOR ONE - The object is to move your car without crashing through the course from the top to the bottom of the screen. Three minutes are allowed to play and you can complete the course more than once. Your number of completions and crashes are recorded. Ten skill levels. Paddles required. As the skill levels increase, the road gets narrower.

{38.9} GUNFIGHTERS - (by Don Graham) Use the keyboard to make your gun shoot the bad guy. Ten increasingly faster draws. If you draw first, you are considered to be cheating. [Such is Apple etiquette in the Old West.]

{38.10} PIG LATIN - Translates your input into Igpay Atinlay. OK for the novice learning the language.

{38.11} A DISK SUMMARY - A menu to select program descriptions or to run a program on this disk.

DISK 103: MERRY CHRISTMAS
(Reviewed by John Malcolm)

This disk is for the holiday season. Most of the programs, including the HELLO program are in Integer Basic. To start, BRUN CAT for an automated catalog. Then select INTEGER BASIC-DISK to load the Integer Basic interpreter into your RAM memory. The musical programs are hard to get to from the menu, but with perserverance you can play them.

{103.1} WAP-MERRY XMAS - Seasons' greetings scroll up the screen.

{103.2} CHRISTMAS TREE - Lets you decorate and display a Christmas tree with lights that flash and glow.

{103.3} XMAS TREE - A tree with flashing lights.

{103.4} CHRISTMAS - A series of holiday pictures.

{103.5} HOLIDAY GREETING PAK - (from Call A.P.P.L.E.) Nine seasonal graphics.

{103.6} JOY TO THE WORLD - This timeless carol is played through your tinny Apple speaker.

{103.7} DECK THE HALLS - See "Joy to the World."

{103.8} XMAS TREE & MSG - A decorated tree and presents slowly materialize on your screen, followed by a holiday message.

APPLE-MUSIC SUBROUTINES - could not get it to run.

RUN ME FIRST - A WAP greeting program.

DISK 120: IAC 16 MISCELLANEOUS
(Reviewed by Ron Bernstein)

This disk from the IAC contains a varied assortment of programs including utilities, business, games, and educational programs.

{120.1} ALGEBRA 1 - This program plots linear equations of the form Y = mX + b on the hi-res screen.

{120.2} APPLEWRITER READER - This program will display or print the contents of binary files containing text produced with old versions of Apple Writer.

{120.3} DAN'S OPTION WRITER - Use this program to estimate the commission and possible return on investments in stocks and options. When your Apple talks, people listen.

{120.4} DAN'S SWITCH - This handy utility will switch the position of files in your disk's catalog. It's a nice way to keep large catalogs organized and readable.

{120.5} GAMBLER - The computer picks a number from 1 to 10. You and your friends get one chance to guess.

{120.6} MEMORY GAME - A game for one or more players. The computer displays nonsensical strings of characters. If you remember correctly, the next one will be longer.

{120.7} PRODUCTION OF LIGHT - This educational program uses hi-res animation to show how light is produced, at the atomic level.

{120.8} RUTHER - This program uses the Apple's graphics to re-create a famous experiment by Ernest Rutherford.

{120.9} SHIP CAPTAIN CREW - A game of chance with hi-res dice. The instructions are the best part. Requires the binary file H-DICE, included.

{120.10} SIMON - An Apple version of the well known electronic game. Features multiple difficulty levels and elaborate sound effects.

{120.11} SPELLING TEST - This educational program briefly displays words and prompts the user to spell them correctly. It uses the included text file, SPELLING LIST, or make up your own.

{120.12} STATIC ELECTRICITY - This is a brief, interactive tutorial on static electricity, featuring animated graphics.

{120.13} TEXT.APNOTE-B2 - A binary-as-text file with info of interest to Apple users. Use APPLEWRITER READER to read it.

{120.14} THE GREEN APPLES - On my copy, this Applesoft program was 7 empty sectors.

{120.15} TOM'S DISK MODIFIER - This is a super disk-zap utility that displays any sector on a disk in hex, ASCII, or disassembled format. You can make any changes you want and write them back to disk, or just peek around. Very fast and very nice.

{120.16} TOM'S MATH DRILL - This drill in simple arithmetic for 1 to 5 players lets you see how soft your mind has become after years of using that pocket calculator.

{120.17} TOM'S TRASH COMPACTOR - This utility removes holes in your disk's catalog left by deleted files.

{120.18} TWO DRIVE COPY - This program permits copying a range of sectors from one disk to another. Two drives required.

{120.19} DOS POINTERS RWDISC - These machine language routines must be used by a program on this disk. If you figure out which, you win any prize you choose to purchase.

PLANETFINDER FIX (WAP VOL. 28)

by Michael Plett

The Washington Apple Pi library disk Vol. 28 (Planetfinder) contains a program to compute the positions of the planets for any time within 300 years of the present time. After obtaining this program and reading the self-contained instructions, I was still confused about the system of time (Greenwich, Eastern standard, etc.) required for input. Cross checking the results with The American Ephemeris and Nautical Almanac, I knew there was some problem since I could not reproduce any of the positions in it. Therefore I was forced to locate the primary reference article, "Low-precision Formulae for Planetary Positions" (Van Flandern and Pulkkinen, Astr. J. Supp., 41:391-411, 1979 Nov) coauthored by fellow member of the WAP T. VanFlandern. There I found that Greenwich Mean Time was acceptable, but Eastern Standard Time was not. Fortunately the article also included a detailed example worked out for 0000GMT June 28, 1969. Running the program against that date and time quickly revealed serious discrepancies. The debugging effort which followed uncovered errors in the Julian date conversion, errors in the coefficients in the code, and errors in text file data. If you implement the following simple fixes, you will restore the program at least to the point to where it will agree with the primary reference.

Before applying the following fixes, make a copy of the disk in case you make an error. Next load the Applesoft routine PP5. This is the main routine of Planetfinder. (If you have an Apple II+, you can get around the Integer Basic HELLO program by just running PP5.) Add the following lines to fix the Julian date/time problems.

```
 97  TI=T: REM SAVE INPUT TIME FOR LATER DISPLAY
 98  IT=INT(T/100): REM DETERMINE NUMBER OF HOURS
     SINCE MIONIGHT GMT
 99  T=IT+(T-IT*100)/60: REM ADD FRACTION OF TIME
     REPRESENTED BY MIN.
100 J=367*Y-INT(7*(Y+INT((M+9)/12))/4)-INT(3*
    (INT((Y+INT((M-9)/7))/100)+1)/4)+INT(275*M/9)
    +D+1721029+(T-12)/24 : REM COMPUTE JULIAN DAY
129 T=LIT
```

To fix the coefficients, add the following lines.

```
 380 ARG(28) = .725368 + .00001672092 * T
 420 ARG(33) = .357355 + .00001104864 * T
1127 L = ARG(24) * 360:D = 19.24877: GOTO 1040
```

If you delete line 1050, the planet declinations will be printed in the customary range +90 degrees to -90 degrees. Before you finish, I recommend you add the following line so that whenever you run Planetfinder you will know you have the right version.

```
3032 PRINT TAB (11)"REV. FEB. 1983"
```

Be sure to save this version of PP5 before doing anything else.

The next step is to modify the text files and although it appears tricky it is really simple. The Planetfinder disk has a program on it to simplify the process. There are seven files that need to be fixed. With two exceptions only one fix is needed. When run, the program, APPEND, will ask for the name of the file to be fixed. It will load the file and then ask which line is to be fixed. It will then display the line and ask if it is the right one. Each line is actually 6 to 12 numbers which are mostly single digit integers. The display however is compacted; you will have to reenter each number individually. To simplify this process, the data below will first give you the original compacted line and the corrected line with the numbers separated by commas.

CHANGE FILE	LINE	FROM	TO
MOONB	10	-30001-20010	-30,0,0,0,1,-2,0,0,1,0
MOONV	0	.3995580001010000	.39558,0,0,0,1,0,1,0,0,0
MOONW	44	-5E-06002202000	-5E-05,0,0,2,2,0,2,0,0,0
MARSU	11	1.07E-03010110-110	-1.07E-03,0,1,0,1,1,0,-1,1,0
JUPITERV	22	1.9E-05110001100	1.9E-04,1,1,0,0,0,1,1,0,0
JUPITERU	9	-1.119E-03010010000	-1.19E-03,0,1,0,0,1,0,0,0,0
JUPITERU	12	9.8E-0500010-12-50	9.8E-04,0,0,0,1,0,-1,2,-5,0
SATURNU	10	1.168E-030000010-10	1.68E-03,0,0,0,0,0,1,0,-1,0
SATURNU	11	-.0153010000020	-.00153,0,1,0,0,0,0,0,2,0

After you have made these changes and saved them to the disk, run the updated version of PP5 for the test time and date of 0,6,28,1969. To be sure everything is OK check the results against the table below. The column labeled Original Value is the result from the uncorrected Planetfinder. The column labeled Reference Value is from the reference article. The column labeled Corrected Value is from the corrected Planetfinder. The column labeled AE&NA Value is from the American Ephemeris and Nautical Almanac as quoted in the article. The angles are quoted in degrees and decimal fractions as output by Planetfinder except for the right ascension which is quoted in hours, minutes, and seconds. Because of the nature of the error in the Julian date calculation, the original results are fortuitously close to the correct ones. Also note that the distance quoted for the Moon is actually in units of Earth radii, not Astronomical Units as stated.

In summary, once the program is fixed, it should provide a convenient way of obtaining planetary positions accurate to one minute (1/60 degree.)

PLANET	PARAMETER	ORIGINAL VALUE	REFERENCE VALUE	CORRECTED VALUE	AE&NA VALUE
Sun	Lambda	96.0276	96.1272	96.1272	96.1272
	Beta	0.0	0.0	0.0	-0.0003
	Distance	1.01664	1.01665	1.01665	1.01659
	Declin.	23.3075	23.3030	23.3029	23.3039
	Rt.Asc.	6h26m14s	6h26m41s	6h26m40s	6h26m40s
	Distance	1.01662	1.01665	1.01662	1.01659
Moon	Lambda	248.352	249.917	249.918	249.917
	Beta	-4.8085	-4.8517	-4.8516	-4.8525
	Distance	56.6162	56.5554	56.5554	56.5524
	Declin.	333.286	-26.738	-26.738	-26.741
	Rt.Asc.	16h22m59s	16h29m54s	16h29m54s	16h29m54s
	Distance	56.6165	56.5554	56.5556	56.5524
Mercury	Lambda	340.825	341.264	341.264	341.266
	Beta	-6.4561	-6.4353	-6.4352	-6.4380
	Distance	0.37934	0.37873	0.37873	0.37869
	Declin.	19.8871	19.9211	19.9207	19.9211
	Rt.Asc.	4h53m13s	4h53m45s	4h53m45s	4h53m43s
	Distance	0.92246	0.92470	0.92471	0.92477
Venus	Lambda	326.206	326.371	326.371	326.368
	Beta	-3.1850	-3.1883	-3.1884	-3.1894
	Distance	0.72810	0.72810	0.72810	0.72808
	Declin.	15.0680	15.0956	15.0957	15.0967
	Rt.Asc.	3h16m6s	3h16m32s	3h16m31s	3h16m30s
	Distance	0.78546	0.78630	0.78630	0.78630
Mars	Lambda	265.022	265.081	265.081	265.082
	Beta	-1.0820	-1.0836	-1.0836	-1.0814
	Distance	1.46468	1.46455	1.46455	1.46463
	Declin.	336.411	-23.729	-23.729	-23.719
	Rt.Asc.	15h59m49s	15h58m41s	15h58m41s	15h58m43s
	Distance	0.50888	0.50631	0.50631	0.50638

Jupiter	Lambda	188.562	188.569	188.569	188.568
	Beta	1.3055	1.3056	1.3055	1.3056
	Distance	5.45232	5.45233	5.45233	5.45265
	Declin.	2.0126	2.0142	2.0146	2.0108
	Rt.Asc.	11h54m24s	11h54m27s	11h54m27s	11h54m27s
	Distance	5.50117	5.50323	5.50323	5.50390
Saturn	Lambda	31.0750	31.0786	31.0787	31.0811
	Beta	-2.4667	-2.4667	-2.4667	-2.4667
	Distance	9.26187	9.26188	9.26184	9.26281
	Declin.	11.4449	11.4778	11.4763	11.4775
	Rt.Asc.	2h19m50s	2h19m53s	2h19m52s	2h19m54s
	Distance	9.76074	9.73510	9.73503	9.73504
Uranus	Lambda	183.228	183.224	183.225	183.227
	Beta	0.7267	0.7267	0.7267	0.7292
	Distance	18.3066	18.3077	18.3077	18.3086
	Declin.	0.6378	0.6392	0.6374	0.6425
	Rt.Asc.	12h1m23s	12h1m21s	12h1m22s	12h1m22s
	Distance	113.872	18.3846	18.3848	18.3881
Neptune	Lambda	237.064	237.583	237.583	237.581
	Beta	1.7074	1.7028	1.7028	1.7194
	Distance	30.2418	30.3266	30.3266	30.3234
	Declin.	342.485	-17.640	-17.640	-17.643
	Rt.Asc.	15h35m43s	15h37m54s	15h37m53s	15h37m53s
	Distance	29.4566	29.5375	29.5375	29.5354
Pluto	Lambda	174.451	174.468	174.452	174.429
	Beta	15.5950	15.5950	15.5950	15.5586
	Distance	31.8308	31.8308	31.8308	31.7720
	Declin.	17.0854	17.0842	17.0841	17.0961
	Rt.Asc.	11h57m47s	11h47m53s	11h57m48s	11h58m1s
	Distance	32.0060	32.0076	32.0076	31.9861

A PAGE FROM THE STACK
by Robert C. Platt

With all of the excitement surrounding the announcement of ProDOS and the Macintosh, it's important to remember that the WAP library stands ready to support your exploratory efforts on these new products. I will be at the disk sales desk before the monthly meeting and would like to hear about your latest efforts. Please keep those contributions coming! For every disk of software that you donate, you will receive a free library disk in exchange.

Given the Wozfest, no library disks will be sold at the January meeting, although you can still purchase disks at the Office or through the mail using the order form at the back of the Journal. Hence, we are holding back on new disk debuts until February with two exceptions: a revised Edsig 1 Elementary Math and Disk 151: 1983 Tax Templates.

ALF MUSIC FILES

Bernie Benson is compiling a disk of song files for playing on the ALF Music Synthesizer Board. Submit a file through the usual donation procedures and get a free library disk in exchange. Contact Bernie at 546-0076 or me for details.

ON-GOING PROJECTS

Thanks are due to Fred Edwards and William Herrick for their work on converting our DOS 3.2 disks into 3.3 format. Also, WAP thanks Doug Schneider and Susan Gowin for volunteering to prepare a subject index of our disk library. Finally, thanks to Walt Francis for assembling spreadsheet template disks.

If you can volunteer some time to work on improving the library, please call me at 223-1588.

DISK 110: ERRATA

File CLASS GRADES on Disk 110 erroneously uses an APPEND command instead of OPEN in lines 5190, 5350, 5430, 5530, 5610, 5740, 5810 and 7030. Change the word APPEND in each of these lines to read OPEN. For example, line 5430 should read:

```
5430 PRINT D$;"OPEN G%";A;",L4"
```

Note that a correct version of this program is called GRADE MAINTENANCE on Disk 141. Thanks to Miriam Levine for reporting this error.

DISK 151: 1983 TAX TEMPLATES

The worst part of my post-Christmas mail was the Form 1040 from the IRS. The best part was Roy Rosfeld's tax templates which will work with the major spreadsheet programs such as Visicalc and THE Speadsheet 2.0. As shown by the companion article in this issue, Roy has worked hard at making the template easy to use. It's excellent, and I for one plan to use it for my 1983 tax return. Because of the need to get this disk to our members as quickly as possible, this disk is being sold at the Office from an inventory stocked by volunteer copiers. So check its availability before making a special trip to the Office to purchase a copy.

DISK 150: EDSIG 1 - ELEMENTARY MATH

We are reissuing this disk with added documentation. To view the summary printed below, type "BRUN DOCUMENTATION".

WAP's Education Special Interest Group has assembled these programs in a convenient format. When each program is completed, the menu display program "HELLO II" is automatically run.

{150.1} COLOR MATH - (from the San Francisco Apple Core) presents random addition, subtraction, multiplication and division drill problems using lo-res graphics. Problems continue until the student enters -99 as an answer.

{150.2} MUSIC MATH - (by P. Schumacher Dec. 1977 Kilobaud, adapted by Charles L. Hearn) addition, subtraction, multiplication and division drills using numbers between 4 and 9. Your score is presented after each set of 10 problems. Correct answers are rewarded with music.

{150.3} LONG DIVISION - the student provides a division problem and watches the Apple solve it step-by-step. An optional Slow Speed feature allows you to call out the intermediate steps before the Apple displays the step on the screen.

{150.4} BAR GRAPH - the user supplies data points, the range of data, and the number of data items and a multicolor bar graph is produced on the lo-res page.

{150.5} COIN TOSSING and HEADS/BINOMIAL - user selects the number of coins to be tossed. TAILS or HEADS is displayed on the screen for each toss, and totals and a ratio are reported at the end of the experiment.

{150.6} PRIME NUMBERS - displays all prime numbers between 2 and 32766. The program pauses several times to ask if it should continue. A brute-force trial of all previous prime divisors is implemented using the Integer Basic MOD operator.

{150.7} ELEMENTARY MATH - a lo-res graphics addition drill. Two rows of dots are displayed, and the student is asked to add the numbers represented by the dots. User has the option of specifying the number of incorrect answers permitted before moving on to another problem.

{150.8} DENOMINATORS - (by Dennis Ward) finds the greatest common divisor of two numbers provided by the user. Uses Euclid's method of repeated subtractions. Stems from Apple Vol. 3 No. 4 p. 7 (April 1980).

{150.9} FRACTIONS - (by Dale Weberg) has three subprograms to (1) reduce fractions, (2) convert improper fractions into mixed numbers and (3) arithmetic drills with proper fractions, improper fractions and mixed numbers. An excellent implementation featuring hi-res numbers, "work space" to record intermediate calculations and polite error messages.

LAST MINUTE NOTE: We are also introducing PIG12: this month. See the review in the PIG column elsewhere in this issue.

WASHINGTON APPLE PI DISK LIBRARY MAIL ORDER FORM

Washington Apple Pi's library disks are available for purchase. This form is only for ordering disks that you want mailed to you.

PROGRAM DISKETTES:

Mailed to you (1st 5) - Members $ 6.00 ea.; Non-members $ 9.00 ea.
" (remainder) - Members $ 5.00 ea.; Non-members $ 8.00 ea.

DOS 3.2

( ) Volume 1 Utilities I
( ) Volume 2 Utilities II
( ) Volume 3 Games I
( ) Volume 4 Games II
( ) Volume 5 Games III
( ) Volume 6 Games IV
( ) Volume 7 Games V
( ) Volume 8 Utilities III
( ) Volume 9 Educational I
( ) Volume 10 Math/Science
( ) Volume 11 Graphics I
( ) Volume 12 Games VI
( ) Volume 13 Games
( ) Volume 14 IAC Utilities IV
( ) Volume 15 Games VII
( ) Volume 16 Utilities V
( ) Volume 17 Graphics II
( ) Volume 18 Educational II
( ) Volume 19 Communications
( ) Volume 20 Music
( ) Volume 21 Apple Orchard
( ) Volume 22 Utilities VI
( ) Volume 23 Games VIII
( ) Volume 24 Games IX
( ) Volume 25 Utilities VII
( ) Volume 26 Stocks/Investments
( ) Volume 27 Math
( ) Volume 28 Planetfinder
( ) Volume 29 Utilities VIII
( ) Volume 30 Games X
( ) Volume 31 Plot Utilities
( ) Volume 32 Games XI
( ) Volume 33 Accounting
( ) Volume 34 Solar Tutor
( ) Volume 35 Garden Management
( ) Volume 36 Games XII
( ) Volume 37 Utilities IX
( ) Volume 38 Games XIII
( ) Volume 39 IAC VII
( ) Volume 40 IAC VIII

DOS 3.3

( ) Volume 41 IAC 25 Mach. Lang. Util.
( ) Volume 42 One Key DOS ***
( ) Volume 43 IAC 29 Utilities H
( ) Volume 100 Utilities A
( ) Volume 101 Utilities B
( ) Volume 102 Games A
( ) Volume 103 Merry Christmas
( ) Volume 104 Business A
( ) Volume 105 FIG-FORTH/Utilities
( ) Volume 106 Science Engineering
( ) Volume 107 Games B

DOS 3.3 contd.

( ) Volume 108 IAC 10 (Graphics)
( ) Volume 109 IAC 11 (Applesoft Tutorial)
( ) Volume 110 Personal/Education
( ) Volume 111 Games C
( ) Volume 112 Utilities C
( ) Volume 113 Business B
( ) Volume 114A Adventure-INIT **
( ) Volume 114B Adventure **
( ) Volume 115 IAC 12/13 Misc.
( ) Volume 116 IAC 14 MicromodemII
( ) Volume 117 Picture Packer
( ) Volume 118 Utilities D
( ) Volume 119 IAC 15 Misc.
( ) Volume 120 IAC 16 Misc.
( ) Volume 121 ABBS Program
( ) Volume 122 IAC 17 Misc.
( ) Volume 123 French Vocabulary
( ) Volume 124 Utilities E
( ) Volume 125 IAC 18 Misc.
( ) Volume 126 Sights and Sounds
( ) Volume 127 Math/Science
( ) Volume 128 Games D
( ) Volume 129 GLAQ
( ) Volume 130 Diversi-DOS ***
( ) Volume 131 Personal/Educ. 2
( ) Volume 132 IAC 19 - Utilities F
( ) Volume 133 IAC 20 - Hybrid
( ) Volume 134 New Members Disk
( ) Volume 135 WAP ABBS-Part 1 **
( ) Volume 136 WAP ABBS-Part 2 **
( ) Volume 137 IAC 21 Spreadsheet A
( ) Volume 138 IAC 23 Utilities G
( ) Volume 139 IAC 24 Education 3
( ) Volume 140 Education 4
( ) Volume 141 Special Data Bases
( ) Volume 142 IAC 28 Pinball Games
( ) Volume 143 Sports
( ) Volume 144 IAC 27 Applesoft Prog.
( ) Volume 145 Apple Logo Tool Kit
( ) Volume 146 Apple Logo Sample Prog.
( ) Volume 147 Logo Documentation
( ) Volume 150 EDSIG1 (Elem. Math)
( ) Volume 151 1983 Tax Templates

Eamon Series

( ) Volume 180 Dungeon Designer
( ) Volume 181 Beginners Cave
( )*Volume 182 Lair of Minotaur
( )*Volume 183 Cave of the Mind
( )*Volume 184 Zyphur Riverventure
( )*Volume 185 Castle of Doom
( )*Volume 186 Death Star

Eamon contd.

( )*Volume 187 Devil's Tomb
( )*Volume 188 Caves of Treas.Isl.
( )*Volume 189 Furioso
( )*Volume 190 The Magic Kingdom
( )*Volume 191 The Tomb of Molinar
( )*Volume 192 Lost Isl. of Apple
( )*Volume 193 Abductor's Quarters
( )*Volume 194 Quest for Trezore
( )*Volume 195 Underground City
( )*Volume 196 Merlin's Castle
( )*Volume 197 Horgrath Castle
( )*Volume 198 Deathtrap
( )*Volume 199 The Black Death
( )*Volume 200 The Temple of Ngurct
( )*Volume 201 Black Mountain
( )*Volume 202 Nuclear Nightmare

Pascal

( ) Volume 300 PIG0: ATTACH/BIOS
( ) Volume 301 PIG1:
( ) Volume 302 PIG2:
( ) Volume 303 PIG3:
( ) Volume 304 PIG4:
( ) Volume 305 PIG5:
( ) Volume 306 PIG6:
( ) Volume 307 PIG7:
( ) Volume 308 PIG8:
( ) Volume 309 PIG9:
( ) Volume 310 PIG10:
( ) Volume 311 PIG11:
( ) Volume 312 PIG12:
See also Volume 133

CP/M

( ) Volume 401
( ) Volume 402
( ) Volume 403
( ) Volume 404
( ) Volume 405
( ) Volume 406 ZCPR2 Install
( ) Volume 407 ZCPR2 Documentation
( ) Volume 408 ZCPR2 Utilities

Forth

( ) Volume 700 Assembler/Disassembler
( ) Volume 701 Full Screen Editor
( ) Volume 702 GoForth
See also Volume 105

* Volume 181 required with these disks. **Both disks required.
*** For examination only. Use of this disk requires sending money to the author ($30 for Diversi-Dos and $9 for One Key Dos.)

(NOTE: ALLOW 2 TO 3 WEEKS FOR MAILING.)

Total Order = ___ disks. $_____

NAME _____

ADDRESS _____

CITY, STATE ZIP _____

TELEPHONE _____ WAP MEMBERSHIP NO._____

Make check payable and send to:

Washington Apple Pi, Ltd.
Attn. Librarian
8227 Woodmont Avenue, Suite 201
Bethesda, MD 20814

WAP TUTORIAL REGISTRATION

The following four WAP tutorials are being offered on Tuesday evenings from 7:30 to 9:00 PM, at the office, 8227 Woodmont Avenue, Room 202, Bethesda, MD. (The tutorials start promptly at 7:30; if you bring your computer please arrive 15 minutes early to set up.) You may sign up for any or all of the series. They are designed for the "beginner" and will be repeated monthly. A detailed outline of the tutorials is given elsewhere in this issue.

( ) February 7 - INTRODUCTION TO APPLE COMPUTER HARDWARE ( ) March 6

( ) February 14 - HOW TO USE YOUR APPLE SOFTWARE ( ) March 13

( ) February 21 - BEGINNING BASIC ( ) March 20

( ) February 28 - BITS, BYTES, NIBBLES ( ) March 27

The fee for each tutorial is $10.00 with an Apple, monitor and disk drive, $15.00 without. Please note that WAP does not have equipment for you to use; if you do not bring your own, you will have to look over someone's shoulder.

__ Tutorials at $10.00 (with equipment)
__ Tutorials at $15.00 (without equipment)

The following "non-regular" tutorials are being offered (also at the office).

VISICALC - Lee Raesly, Sat. AM Feb. 11 9:30-12:00 (Almost full - check with office.)

( ) $10 with Apple, member ( ) $15 with, non-member
( ) $15 w/o Apple, member ( ) $20 w/o, non-member

VISIPLOT - Lee Raesly, Sat. AM Feb. 18 9:30-12:00

( ) $10 with Apple, member ( ) $15 with, non-member
( ) $15 w/o Apple, member ( ) $20 w/o, non-member

ADVANCED APPLESOFT - Richard Untied, Sat. AM Mar. 10 9:30-12:00

( ) $10 with Apple, member ( ) $15 with, non-member
( ) $15 w/o Apple, member ( ) $20 w/o, non-member

CP/M - Bob Fretwell, Sat. AM Apr. 14, 9:30-12:00

( ) $10 with Apple, member ( ) $15 with, non-member
( ) $15 w/o Apple, member ( ) $20 w/o, non-member

dBase II - Lee Raesly, Sat. AM, May 5, 9:30-12:00

( ) $10 with Apple, member ( ) $15 with, non-member
( ) $15 w/o Apple, member ( ) $20 w/o, non-member

The following tutorials are planned but no exact date yet:

Pascal - Pascal Sig - June.

Please check the desired tutorials and return this form with the proper fee(s) payable to Washington Apple Pi, Ltd. to:

Washington Apple Pi, Ltd.
Attn. Tutorials
8227 Woodmont Avenue, Suite 201
Bethesda, MD 20814

Name _____

Daytime Phone _____ Evening Phone _____ Total Enclosed $ _____

INDEX TO ADVERTISERS

GET YOUR HANDS-ON A COMPUTER!

at the
CLINTON COMPUTER LEARNING CENTER

Classes scheduled for January thru March

NEW APPLE OWNERS COURSE. 4 1/2 hours. $100. Alternate Saturdays. 9am-1:30pm. Learn the basic operating procedures of the world's most popular small computer! Course includes five program disks.

NEW CP/M OWNERS COURSE. 4 hours. $100. Alternate Saturdays. 9am-1pm. Hands-on instruction in this standard operating system. A must for new owners of CP/M-based computers.

LOGO I FOR TEACHERS. 14 hours. $125. Jan.11, 18, 25, Feb. 1, 8, 15, & 22. 6:30-8:30pm. Gain experience using LOGO Graphics. Learn how to apply the microcomputer in the classroom and to incorporate its use into an existing curriculum. (The cost of this course is $100 if you sign up for both LOGO courses.)

LOGO II FOR TEACHERS. 14 hours. $125. Feb. 29, Mar.7, 14, 21, 28, April 4, 11. 6:30-8:30pm. Learn to use LOGO for activities with numbers, words, and lists. This class is a follow-up to "LOGO I for Teachers." (The cost of this course is $100 if you sign up for both LOGO courses.)

APPLEWRITER IIe. 2 1/2 hours. $46. Jan.17, Feb. 21, & Mar. 20. 6:00-8:30pm. Understand what a word processing system is and how it works. Learn how to create/revise letters and documents quickly and easily.

APPLESOFT BASIC PROGRAMMING I. 6 hours. $65. Call for details. Hands-on instruction for computer novices in this simple programming language. Classroom exercises and optional homework assignments.

APPLESOFT BASIC PROGRAMMING II. 6 hours. $65. Call for details. Learn new Basic commands and additional programming techniques. Classroom exercises and homeword assignments. Prerequisite: Basic Programming I or equivalent.

INTRODUCTION TO WORDSTAR. 2 hours. $40. Jan. 24, Feb. 7, March 6. 6:30-8:30pm. Learn how to create documents quickly and easily by using this popular word processing system.

ADVANCED WORDSTAR. 2 hours. $40. Jan. 31, Feb. 14, March 13. 6:30-8:30pm. Experiment with more advanced Wordstar commands, and learn file merging techniques. Prerequisite: Wordstar I or equivalent.

MICROSOFT BASIC PROGRAMMING I. 6 hours. $65. Call for details. Hands-on instruction on the features of Microsoft Basic for the computer novice. Classroom exercises and optional homework assignments.

MICROSOFT BASIC PROGRAMMING II. 6 hours. $65. Call for details. Learn additional programming techniques in Microsoft Basic. Classroom exercises and homework assignments. Prerequisite: Microsoft Basic Programming I or equivalent.

COMPUTER CAMP FOR KIDS. 10 hours. $110. Jan. 7, 14, 21, 28 & Feb 4. 2:00-4:00pm. Optional Lab with instructor, 4:00-5:00pm. Introduce your child to the world of small computers! Hands-on programming and instruction in practical applications.

CHOOSING APPLE EDUCATIONAL SOFTWARE. 2 hours. $20. Jan 16. 6:30-8:30pm. Discussion and demor strations of educational software. (Price of class may be credited toward purchase of educational software in our retail store)

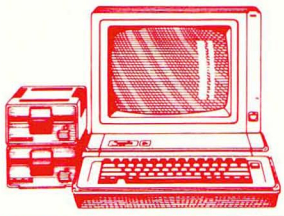
The Learning Center also provides custom-designed instruction on-site or in our classroom. Course discounts available with system purchases. Contact us at 301/856-2500 for details and inquiries.

Clinton Computer

apple computer
®Authorized Dealer & Service Center

Beltway Exit 7A
Branch Ave. (Rt. 5) at
6443 Old Alex. Ferry Rd.
Clinton, Md. 20735

(301) 856-2500



WASHINGTON APPLE PI, LTD.
8227 Woodmont Avenue, Suite 201
Bethesda, MD 20814

BULK RATE
U. S. POSTAGE
PAID
PERMIT #101
Glen Echo, MD
20812

ADDRESS CORRECTION REQUESTED

FOR WAP MEMBERS ONLY. You can now buy an AJ 831 for **$712.** This includes a 90-day full warranty with any repairs being done at your home or office. Please call Sean Belanger at 840-5700 for details.