

\$2

# Washington Apple Pi



The Journal of Washington Apple Pi, Ltd.

Volume 6

May 1984

Number 5

## Highlights

LOGO VIRUS LogoSIG News  
 MAC ↔ APPLE ][ FILE TRANSFER  
 PROGRAM LISTING FORMATTER I Am What I Am ...  
 USING WPL TO CUSTOMIZE  
 APPLEWRITER //e

## In This Issue

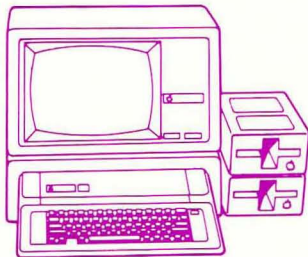
Officers & Staff, Editorial . . . . .	3	Hardware Helpers . . . . .	31
President's Corner . . . . . David Morganstein	4	I Am What I Am . . . . . John A. Love III	34
Event Queue, General Information . . . . .	4	Softviews . . . . . David Morganstein	46
Minutes, Congratulations, SIGNews . . . . .	5	Anchor Mark 12 Review Update . . . George V. Kinal	47
WAP Hotline . . . . .	6	Auto-Repeat Dialing . . . . . George V. Kinal	48
Apple Teas . . . . .	7	Print Master Deluxe: A Review . . . . Bob Oringel	48
Q & A . . . . . Bruce F. Field	8	Guide to General Disk Utilities . . . . Bob Trexler	49
Educational Software . . . . . Raymond H. Hartjen	12	Turbo Pascal:JRT's Promise Fulfilled..Bill Seeley	51
EDSIG News . . . . . Peter Combes	12	The Forth Column . . . . . Kevin Nealon	52
DisabledSIG News . . . . . Jay Thal	13	Stuffing Com-Ware ][ . . . . . J. T. DeMay Jr	53
A Meeting With Laura Meyers . . . . . Jay Thal	14	The Hotline Calls Back - Apple Writer // . . . . .	54
LOGOSIG News . . . . . Nancy C. Strange	16	Using WPL to Customize AW //e . . . . Dianne Lorenz	55
Classifieds, Job Mart . . . . .	19	Magic Window II & AW // . . . . . Anne F. Moeller	56
A Page from the Stack . . . . . Robert C. Platt	20	What is a Program Review? . . . . . Bob Oringel	57
SIG Mac News . . . . . Ellen L. Bouwkamp	22	A Letter to the Pres/Ed . . . . . Fulton Armstrong	58
Macintosh <-> Apple ][ Transfer..Donald C. Schmitt	24	Transferring Binary Files . . . . . C. Swift, Prop.	60
Macintosh Speed Comparisons . .Richard D. Norling	26	Author Guidelines . . . . .	60
Speed of Macintosh Basic . . . . .		Converting CP/M Files to DOS . . . George V. Kinal	61
David Morganstein & Dave Weikert	28	WAP Tutorial Registration . . . . .	62
International SIG . . . . .	28	Disketeria and Magazine Order Forms . . . . .	62,64
Modula 2: Input/Output . . . . . Robert C. Platt	30	Index to Advertisers . . . . .	64

# SOFTWARE CENTRE INTERNATIONAL SOFTWARE CENTRE

Computer Productivity Sale!  
Apple Computer  
Memory Expansion!

CP/M  
GOLD  
CARD™

 DIGITAL  
RESEARCH®  
We make computers work.



WINTERGREEN PLAZA  
UPPER LEVEL  
833H ROCKVILLE PIKE  
301-424-3024

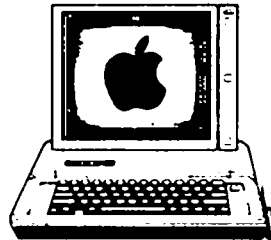
SPRINGFIELD PLAZA III  
7020 COMMERCE ST.  
SPRINGFIELD, VA 22150  
703-455-3202

The  
Original  
Software  
Store™



This ad was composed  
on an Apple p/c and  
converted into type  
via modem.

**PUT  
YOUR**



**APPLE  
AND MERCURY PRESS  
TO WORK  
ON YOUR PUBLICATION  
NEEDS**

- TELECOMMUNICATIONS ● COMPUTER SERVICE ● EDITING ●
- DESIGN ● LAYOUT ● ADVERTISING TRAFFIC/PRODUCTION ●
- COMPOSITION ● PRINTING ● MAILING ● PROMOTION
- MEMBERSHIP DEVELOPMENT ● AD SALES/MARKET  
CONSULTATION ● MAIL/MEMBERSHIP LIST MAINTENANCE

**MERCURY PRESS**

PUBLISHING SERVICES

- Books ● Brochures ● Magazines ● Tabloids ● Directories

*Please Call: Suzanne Walters/Michael Stephens*

**301 770-6177**

# A OPTIONAL C OPTIONAL m p l e C o m OPTIONAL m u n i c OPTIONAL a t i o n s S y OPTIONAL t e m ? ?

Many modem manufactures have offered you complete communications systems for the Apple computer. Hayes offers you the Micromodem II. A complete communications system without the software. Zoom telephonics offers you the networker. A complete communications system without the auto-dialing.

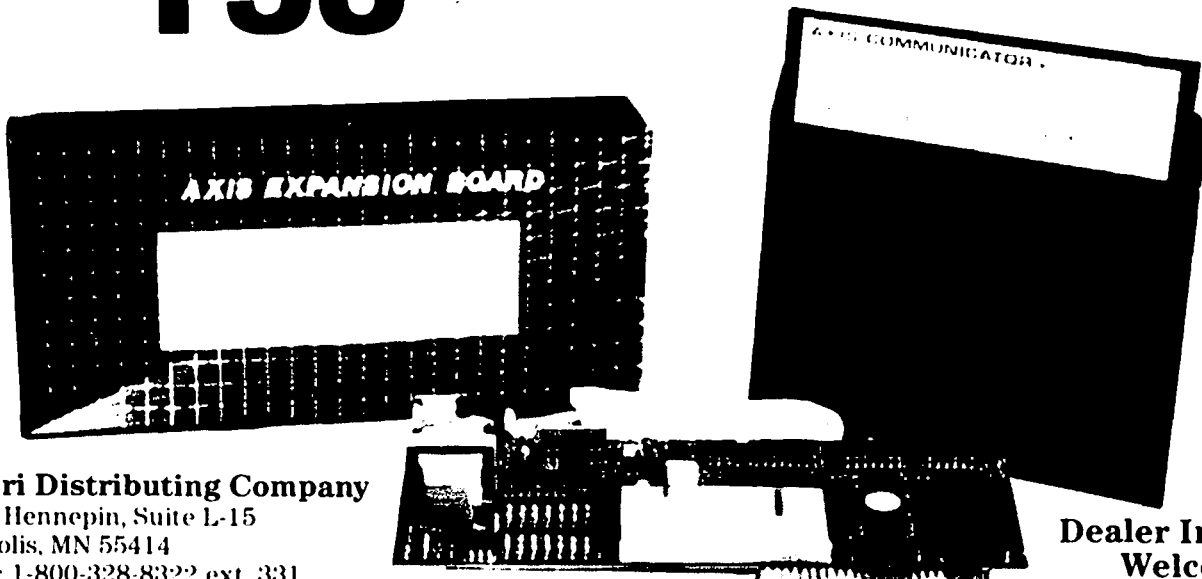
We offer you the **Axis APS modem**, our interpretation of what a complete communications system should be. Built with our dedication to quality, the Axis modem features:

1. **Micromodem II Software Compatability**
2. Auto-Dial
3. Auto-Answer
4. 1.5K Firmware Program (for stand-alone operation)
- \* 5. **The Axis Communicator Software**
6. Stand-Alone Operation (no need for a serial interface)
7. Full/Half Duplex Operation
8. Serial Asynchronous Binary Data with options of 7 or 8 Data Bits, 1 or 2 Stop Bits, Odd, Even or No Parity
9. 300 BPS
10. Bell System 103 Compatability
11. 1.2W Power Consumption
12. Documentation with Programming examples
13. User Friendliness (installation & use in less than 5 minutes)
14. A One (1) Year Warranty

\* The Axis Communicator includes: Auto-call, Answer and Pickup, Basic File Transfer, Binary File Transfer and Text File Transfer.

# \$195<sup>00</sup>

**MICROMODEM II  
SOFTWARE  
COMPATIBLE**



**Centauri Distributing Company**  
416 East Hennepin, Suite L-15  
Minneapolis, MN 55414  
Toll Free: 1-800-328-8322 ext. 331  
In Minnesota: (612) 379-5312  
Telex: 820800

**Dealer Inquiries  
Welcome**

Micromodem II is a Trademark of Hayes Microcomputer Products, Inc. Apple is a registered Trademark of Apple Computers, Inc. Networker is a Trademark of Zoom Telephonics, Inc. Axis, Axis Communicator are Trademarks of Centauri Distributing Co., Inc.

# OFFICERS & STAFF

President	- David Morganstein	(301) 972-4263	Membership	- Dana Schwartz	(301) 654-8060
Vice President	- Dana Schwartz	(301) 725-6281	Program	- Cara Cira	(301) 468-6118
Treasurer	- Edward Myerson	(703) 759-5479	Publicity Chairman	- Hunter Alexander	(703) 820-8304
Secretary	- Sara Lavilla	(301) 926-6355	Public Relations	- Lee Raesly	(301) 460-0754
Directors	- Bernie Benson	(301) 951-5294	Rules & Elections	- Bob Platt	(202) 223-1588
	- Peter Combes	(301) 445-6188	School Coordinators:		
	- Bruce Field	(301) 340-7038	Virginia	- Barbara Larson	(703) 451-9373
	- Nancy Philipp	(301) 924-2354		- Nancy Strange	(703) 691-1619
	- Gordon Stubbs	(703) 750-0224	Mont. Co. MD	- Margie Stearns	(301) 657-2353
	- Rich Wasserstrom	(703) 893-9147	Pr. Geo. Co. MD	- Conrad Fleck	(301) 699-8200
Editor	- Bernie Urban	(301) 229-3458	Special Publcitns.	- Betsy Harriman	(202) 363-5963
Associate Editors	- Betsy Harriman	(202) 363-5963	SYSOP	- Tom Warrick	(301) 656-4389
	- Genevie Urban	(301) 229-3458	Tutorials	- Steve Stern	(301) 881-2543
				- Leon Raesly	(301) 460-0754
Journal Staff:			Volunteer Coord.	- Boris Levine	(301) 229-5730
Store Distribtn.	- Ray Hobbs	(301) 490-7484	SIG Coordinator	- Bob Platt	(202) 223-1588
Columnists:			SIG Chairmen:		
Applesoft	- James T DeMay Jr.	(301) 779-4632	Appleseeds	- David Stern	(301) 881-2543
DisabledSIG	- Jay Thal	(202) 244-3649	Apple ///	- Jerry Chandler &	(703) 790-1651
EDSIG	- Peter Combes	(301) 445-6188		- Bill Hershey	(301) 588-1992
LOGOSIG	- Nancy Strange	(703) 691-1619	ASMSIG	- Ray Hobbs	(301) 490-7484
Q & A	- Bruce Field	(301) 340-7038	CESIG	- John Kapke/	(301) 694-5968
Telecomm	- Dave Harvey	(703) 527-2704		- Roy Rosfeld	(301) 340-7962
VisiCalc	- Walt Francis	(202) 966-5742	CP/M	- Charles Franklin	(301) 997-9138
Group Purchases	- Rich Wasserstrom	*(301) 654-8060	DISABLEDSIG	- Curt Robbins	ABBS WAP428 or
	*(Call Kevin at office on M,W,Th,F)				write 6114 Quebec Place, Berwyn Hgts. MD 20740
Disketeria Staff:	- John Malcolm	(301) 384-1070			(or phone)
	- Dave Weikert, Joy Aso, Bob Hicks,		EDSIG	- John Molineaux	(301) 341-7391
	- Ed Lang, Jim & Nancy Little,		FORTH SIG	- Peter Combes	(301) 871-1455
	- Terry Prudden, Bruce Redding,		Games SIG	- Kevin Nealon	(703) 280-1136
	- Gordon Stubbs		LAWSIG	- Jim Eatherly	(202) 667-2287
New Disks	- Bob Platt	(202) 223-1588	LISASIG	- Charles Field	(202) 265-4040
Pascal Lib.	- Mike Hartman	(301) 445-1583	LOGOSIG	- Gordon Stubbs	(703) 750-0224
CP/M Lib.	- Kit Leithiser	(703) 569-8037	SIG Mac	- Nancy Strange	(703) 691-1619
Head Reading Lib.	- Walt Francis	(202) 966-5742	NEWSIG	- Tom Warrick	(301) 656-4389
Apple Tea Coord.	- Paula Benson	(202) 546-0076	Pascal (PIG)	- Bernie Benson	(301) 951-5294
Arrangements	- Jim Taylor	(301) 926-7869		- Harry Bishop/	(703) 931-4937
Demonstrations	- Signe Larson	(703) 524-4541	STOCKSIG	- Jim Harvison	(703) 593-2993
General Counsel	- Jim Burger (Shaw, Pittman, Potts		Telecomm. SIG	- Robert Wood	(703) 893-9591
	& Trowbridge) day	(202) 822-1093		- 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

Copyright 1984, Washington Apple Pi, Ltd.

## EDITORIAL

The end of an era is upon us. A.P.P.L.E. member 0001 is stepping down from his position as editor of the illustrious Call-A.P.P.L.E. journal, which has been published for some 6+ years. He also is stepping down from his position on the A.P.P.L.E. hotline which he has manned over the same years and to which he has devoted long hours and many weekends. He has provided many solutions to problems posed by Apple neophyte and professional alike. A patient man, filled with good humor and possessed with a gnomish grin, Val Golding is pulling up stakes, leaving old friends and relations behind in Seattle, Washington and starting anew in San Diego. He has joined forces with Roger Wagner (formerly of Southwestern Data Systems and now Roger Wagner Publishing, Inc.), and has taken on responsibility for editing a new "Apple magazine for kids 8 to 80" entitled Apple's Apprentice. We have had many fun filled occasions to work with him in the past, and we look forward to other opportunities to do so again. All the best, Val, in your endeavors and in your new life.

-----

An early report on the outcome of our special April 21, Future of the Pi meeting. While the turnout was somewhat disappointing (approximately 30 folks), those who were there certainly made up for it by entering into the spirit of the meeting. Time was too short. We could have spent ten hours on the subject but cut it off after two. I speak only for myself in what follows. There were some surprises but in my opinion we left with a very positive feeling about the results. I came away with a dozen new ideas. More about that from others later. ...Ed &

# PRESIDENT'S CORNER

by David Morganstein

**ELECTIONS.** Bob Kosciesza spent quite a bit of time contacting people to find candidates for WAP offices. Thanks, Bob, for your contribution. The election will be held at the end of June. At the April Board meeting the topic of expanding the Exec. Board was raised. The need for two vice-presidents with clearly defined responsibilities and for an additional board member was discussed and the expansion approved. The ballot will, as in years past, contain a number of issues relevant to our future budget and activities. You will read more about this in the June issue of the Journal.

**WAP Number 9998.** No, we're not there yet, at least not in total members. This special number was allocated to a member from Great Britain to enable our bulletin board to go International. If you want to leave a message for England, contact WP9998 on the ABBS.

**HOT-LINE VOLUNTEERS.** The phone just rang with a request for help with Quickfile. We don't have a hot-liner for it. Anybody out there who can handle questions? A few other recently requested items include: Supercalc, the Apple Imagewriter printer, the C language and the Word Juggler word processor.

**CALL-A.P.P.L.E. UPDATE.** Dick Hubert, President of Call-A.P.P.L.E., called our attention to the letter of agreement he had sent us after we had verbally established the membership special last spring. The letter did state that the agreement was to last only until December 1983. Our mistake and apologies to Call-A.P.P.L.E. for failing to note this. They added the cut-off date in their letter because they expected to increase their membership fee, as they did, at the beginning of the year. Dick indicated that a repeat offer of some kind will be made again shortly.

**MANY THANKS.** We want to thank Cara Cirra for all her efforts in arranging our monthly programs. She has put together some very educational, interesting and entertaining sessions. We note with mostly sadness (but some excitement) the formation of a Connecticut Hotline. Lofty Becker is returning to a law teaching position in Hartford and has (jokingly) offered to start a local hotline for our members in that area to cut down on phone expenses. While we will miss his personal contribution at meetings, we hope to continue to receive his excellent and much appreciated input on our printed pages. Best of luck, to you, Lofty.

**MOST RECENT VERSION.** Look in this issue for a listing of the most recent version numbers of well known commercial products. It occurred to us that many people who have sent in the registration card accompanying their commercial software do not receive notice from the company of updates. (I thought of this regarding my Microsoft Multiplan for the Mac, reputed to have a few "quirks", having read somewhere that there is already an update out there. True or False? Read Mac Notes to find out.) If you find that you have a more recent version number than we show, please call or write us with the number. We will revise the list and republish it next month. We will repeat this every several months and see if you find it a useful way to keep in touch with the "evolution" of that commercial program you bought.

**MACINTOSH.** Tom Warrick has accepted the leadership of our exciting SIG MAC. We hope to be a source of news

and help for those members trying to overcome the information shortage in which they find themselves. Keep your eye on these pages for news, tips and opportunities. Our application form is being changed to recognize what we think will be a large group of soon-to-be owners. Rich Wasserstrom has indicated that interest in a group buy is rapidly reaching the one hundred mark. Our ability to obtain a good price will depend almost entirely on quantity once the Mac becomes readily available. If you can hold out a bit, we may be able to get substantial savings.

**SECRETARY RECEPTIONIST.** We have received three applicants for the secretary-receptionist position. If you are interested in applying for this half-time job to answer phones and assist in other office activities, please send us a brief resume and your salary requirements. Our budget allows for compensation in the five to seven dollar range but can be adjusted somewhat to meet the applicants experience. ☞

## 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:

May	26	- DOS, David Morganstein
June	23	- Garage Sale

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. ☞

## GENERAL INFORMATION

Apple user groups may reprint without prior permission any portion of the contents herein, provided proper author, title and publication credits are given.

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	- 10 AM to 1 PM (except meeting Sat)
	- 12:00 to 3:30 PM (meeting Sat only)

# MINUTES

## SUMMARY OF MARCH EXECUTIVE BOARD MEETING

The Executive Board of Washington Apple Pi, Ltd. met on March 14, 1984 at the WAP office. John Malcolm was directed to form a committee to select and announce the winner of the new name for the disk library. The Board authorized the purchase of a phone answering machine and a phone line for the purpose of providing group purchase information. The need to provide specialized new member information (disk and handbook) was discussed. A letter of concern to the IAC was reviewed. Bob Platt reported on his communication with a publisher for the book, Pascal for the People. It was announced that Richard Langston will organize the taping of meetings and tutorials. The problems associated with cable TV's interest in the video tapes were discussed. The outlines for the "future of WAP" and the budget update were reviewed. A report on workmen's compensation for the office staff was presented. The treasurer was directed to investigate the status of officers' and directors' liability insurance. Randy Paisley of Frederick Computer Club reported on their needs and wishes to affiliate with WAP. Jay Thal announced the next meeting of Disabled SIG which will feature Laura Meyers and her work with computers and children without language. The feasibility of serving as a clearinghouse for software and hardware for people with special needs was discussed. A request from the Philadelphia Apple users group to distribute our new member disk was discussed. Other announcements included a government organized EXPO in April, WETA's series on computer literacy, and the election procedure. The Board approved the purchase of a Corona hard disk for the ABBS.

## MARCH GENERAL MEETING

Washington Apple Pi, Ltd. met at USUHS on March 24, 1984 at 10:00 AM. President David Morganstein presided. Members were reminded to behave responsibly at USUHS; clean up, replace chairs and equipment, etc. A volunteer was found to finish tabulating the questionnaire data. Persons interested in serving as officers and directors next year were asked to submit their names for the June election. The special meeting on April 21 to discuss the future of Washington Apple Pi was announced. Applications are still being taken for the parttime position of office secretary/receptionist. An announcement was made that WETA is seeking mentors for its computer tutorial program. Members were invited to donate unwanted commercial programs to the commercial software library. It was announced that hardware clinic advisors are listed in the WAP Journal. Volunteers were sought to host Apple teas. Two winners of the disk software library naming contest were announced: C. Swift, Prop. for the name 'Disketeria', and Gerald Michalski for the slogan, 'Software for Creative Living'. Summer tutorials were announced and requests for future tutorials were noted. Group purchase reported, and suggestions for up-front group purchases were taken. Topics for future meetings were solicited. Questions for Bruce Field's Q & A column were collected. &

# CONGRATULATIONS TO THE WINNERS!

There were two winners in the contest to rename the Washington Apple Pi disk-sales activities. C. Swift, Prop. submitted the winning name, "DISKETERIA", and Gerald P. Michalski submitted a slogan which just couldn't be ignored, "Software for Creative Living". Congratulations to both winners, and our thanks to all the members who submitted entries. &

# SIGNEWS

APPLE /// SIG meets on the second Thursday of the month at 7:30 PM. The next meeting will be on May 10 at Walter Reed Institute of Research. From 16th Street entrance go 3/4 around circle. Go in North entrance of the 4-story brick building on your right and ask the guard for Room 3092.

APPLESEEDS is the special interest group for our younger members. They meet during the regular WAP meeting.

ASMSIG, our SIG for assembly language programming, has chosen a new name to reflect its new emphasis - the Program Interface SIG. The PISIG will focus on how to write assembly language routines that will hook into Applesoft programs, DOS, ProDOS, and Pascal. The PISIG will also work with SIG Mac to explore assembly language programming on the MC68000 and the Macintosh. For further details, including information on tutorials, call Ray Hobbs at (301) 490-7484. (Given the fact that all of our new SIG names are one letter away from the names of foods, it's obvious that SIGs only name themselves when people are hungry.)

CESIG is the new special interest group of computer entrepreneurs. They meet after the monthly WAP meeting at the club office.

DISABLEDSIG - See "DISABLEDSIG News" elsewhere in this issue.

EDSIG - the education special interest group - see the EDSIG Page elsewhere in this issue.

FORTH SIG has been revived. See "The Forth Column" elsewhere in this issue.

LAWSIG usually meets in downtown Washington, D.C. at noon once a month. For information call Charles G. Field, Chairman, 265-4040, or Jim Burger, 822-1093.

LOGOSIG meets monthly at 12:45 after the regular WAP meeting at the Barrie School, 13500 Layhill Road, Silver Spring, MD. See "LOGOSIG News" elsewhere in this issue.

NEWSIG will meet just after the regular Washington Apple Pi meeting and on Thursday evenings from 7:30-9:00 PM in the office. We will answer questions and try to help new owners get their systems up and running. We will also explain how our club operates.

PIG, the Pascal Interest Group, meets on the third Thursday of each month at 7:30 PM at the Club Office.

SIGAMES is the special interest group of computer hobbyists interested in using their APPLES for entertainment. They meet immediately following the monthly meeting of Washington Apple Pi.

SIG Mac - see SIG Mac News elsewhere in this issue.

STOCKSIG meetings are on the second Thursday at 8:00 PM at the WAP office. Call Robert Wood, (703) 893-9591.

Telecomm SIG usually meets after the regular WAP meeting. &

# 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	Math/ O.R. Applns.	Mark Pankin (703) 524-0937
Accounting Packages		Operating Systems	
Accountant(Dec.Sup.)	Mark Pankin (703) 524-0937	Apple DOS	Richard Langston (301) 258-9865 Richard Untied (703) 241-8678
Home Accountant	Leon Raesly * (301) 460-0754	CP/M	Robert Fretwell (703) 971-2621
APPLE SSC	Bernie Benson (301) 951-5294	ProDOS	Richard Langston (301) 258-9865
Apple TechNotes	Ed Schenker (301) 977-7349 Lance Bell (703) 550-9064	Paddles	Tom Riley (eve.) (301) 340-9432
Basis 108	Loftus Becker (202) 338-5217	Pers. Filing Sys.	Bill Etue (703) 620-2103 Ben Ryan (301) 469-6457 Jenny Spevak (202) 362-3887
Communications Packages and Modems-Telecom.		Printers	
Anchor Mark 12	George Kinal(7-10)(202) 546-7270 Jeremy Parker (301) 229-2578	General	Walt Francis (202) 966-5742 Bill Etue (703) 620-2103
Apple CAT II	Ben Acton (301) 428-3605	Anderson Jacobson	Leon Raesly * (301) 460-0754
ASCII Express	Dave Harvey (703) 527-2704	Apple Dot Matrix	Joan B. Dunham * (301) 585-0989
BIZCOMP Modem	Jeremy Parker (301) 229-2578	Daisywriter 2000	Henry Greene (202) 363-1797
General	Tom Nebiker (216) 867-7463 Tom Warrick (301) 656-4389	IDS 460	Jeff Stetekluh (703) 521-4882
Hayes Smartmodem	Bernie Benson (301) 951-5294	MX-80	Jeff Dillon (301) 434-0405
Micromodem	Tom Warrick (301) 656-4389	NEC 8023	Bill Mark (301) 779-8938
Omninet	Tom Vier (1-6 PM) (703) 860-0083	Okidata	Fred Feer (703) 978-7724 Scott Rullman (301) 779-5714
Source & Transcend	Jerry Michalski (703) 442-8835	Silentype	Bruce Field (301) 340-7038
VISITERM	Steve Wildstrom (301) 564-0039	Spreadsheets	Leon Raesly * (301) 460-0754 Walt Francis (202) 966-5742
XTALK CP/M Comm.	Bernie Benson (301) 951-5294	Lotus 1-2-3	Walt Francis (202) 966-5742
Corvus Hard Disk	Tom Vier (1-6 PM) (703) 860-0083	Multiplan	Terry Prudden (301) 933-3065
Expediter Compiler	Peter Rosden (301) 229-2288	VisiCalc	Walt Francis (202) 966-5742 Leon Raesly * (301) 460-0754 Leon Raesly * (301) 460-0754
Data Bases		Spreadsheet 2.0	
dBase II	Loftus Becker (202) 338-5217 Paul Bublitz (301) 261-4124	(MagiCalc)	
DB Master	John Staples (703) 759-3461 Doug Daje (301) 868-5487	Statistical Packages	Jim Carpenter (301) 371-5263 Mark Pankin (703) 524-0937
Data Perfect	Dave Einhorn (301) 593-8420	Stock Market	Robert Wood (703) 893-9591
Data Factory	Leon Raesly * (301) 460-0754	Tax Preparer-H.Soft	Leon Raesly * (301) 460-0754
General Manager	Bob Schmidt (301) 736-4698 Norman Bernache (301) 277-6935	Time-Sharing	Dave Harvey (703) 527-2704
InfoMaster	Leon Raesly * (301) 460-0754	Word Processors	Walt Francis (202) 966-5742 Doug Daje (301) 868-5487
List Handler	Doug Daje (301) 868-5487 Jon Vaupel (301) 977-3054	Apple Writer II	Dianne Lorenz (301) 530-7881 Leon Raesly * (301) 460-0754 Tom Warrick (301) 656-4389
Graphics	Bill Schultheis (703) 538-4575	Executive Secretary	Louis Biggie (202) 296-1280 Henry Donahoe (202) 298-9107
Languages (A=Applesoft, I=Integer, P=Pascal, M=Machine		Format II	Neil Muncy (301) 251-9330
A	Peter Combes (301) 445-6188	Gutenberg	Cara Cira (301) 468-6118
A,I	Jeff Dillon (301) 422-6458	Letter Perfect	Leon Raesly * (301) 460-0754
A	Richard Langston (301) 258-9865	Magic Window and II	Joyce C. Little (301) 321-2989
A	Mark Pankin (703) 524-0937	Peach Text	Carl Eisen (703) 354-4837
A	Leon Raesly * (301) 460-0754	PIE Writer/Apple PIE	Jim Graham (703) 643-1848
A,I,P,M	Bill Schultheis (703) 538-4575	ScreenWriter II	Peter Combes (301) 445-6188
A,I,M	Richard Untied (703) 241-8678	Supertext II	Doug Daje (301) 868-5487 Peter Rosden (301) 229-2288
M	Raymond Hobbs (301) 490-7484	Word Handler	Jon Vaupel (301) 977-3054
P	Dottie Acton (301) 428-3605	Word Star	Christopher Romero(703) 471-1949 Christopher Romero(703) 471-1949
P	Donn Hoffman * (202) 966-2616		
LOGO	Ron Murray (eve.) (202) 328-3553		
LISP	Fred Naef (703) 471-1479		
Macintosh Computer	Tom Warrick (301) 656-4389 Donald Schmitt (717) 334-3265		

\*Calls up until midnight are ok.



# APPLES TEAS

WAP is conducting small discussion groups around the D.C. area. Below you will find a list of volunteers who will be holding sessions in the next few months. These will be informal and last about 2 hours. If you would like to attend, call the host and register. Each host will limit the size according to his/her wishes and will let you know if there is room in that particular tea. Hopefully, there will be ample numbers of sessions to meet the interest.

Session leaders may be needed if the host does not feel knowledgeable about the Apple. If you would like to volunteer your help or be a session leader, contact Paula Benson.

## APPLE TEA SCHEDULE

Host(ess)	Area	Phone	Date/Time	Topic
David Einhorn	Wheaton/Kemp Mill	(301) 593-8420	May 24/8:00-10:00	DB Master
Joe McAllister	Fredericksburg, VA	(703) 786-6577	Arrange with host	
Sharon Eagle	Oxon Hill	(301) 839-3316	Arrange with hostess	
Randy Zittel	Woodbridge, VA	(703) 491-5493	Arrange with host	

# TIC Computer Camp

MOUNT VERNON COLLEGE

CALL FOR BROCHURE: 241-5542

## Quality Computer Instruction



**Programming Languages**  
taught at three levels each  
(beginning, intermediate, advanced)

- LOGO
- BASIC
- Pascal
- Machine/Assembly

**Computer Applications,**  
such as

- typing, word processing
- data management
- music generation
- graphics, animation
- voice synthesis
- electronic hardware projects



**The Fun of  
Going All  
Out in  
Athletics**



# Q & A

by Bruce F. Field



Last month I answered a question about underlining with Apple Writer II and an Okidata printer. The problem was that if the printer underlining feature was turned on, and the words to be underlined continued from the right side of one line to the next line, the underline continued through the left margin. The solution is to use the backslash command in Apple Writer rather than sending the "underline on" command to the printer. Right after sending the column in for publication I received several more letters describing exactly the same problem. My solution still stands for all printers except the Okidata. Unfortunately the Okidata refuses to recognize the backspace character which is part of what Apple Writer uses to underline when the backslash command is given. If the left margin command of Apple Writer is used (i.e. .lmXX), then when Apple Writer reaches the end of a line it sends XX spaces at the beginning of the next line. If the backslash command of Apple Writer is used, Apple Writer is smart enough to know not to underline the spaces. On the other hand if the printer underline feature is turned on, the spaces look like any other character to the printer and are underlined. If you can't use the backslash command (as with the Okidata) another solution is to always use a left margin of 0 with Apple Writer (.lm0) and instead send the proper command to the printer to set the left margin as desired. For the Okidata printer the left margin command is <ESC>%Cnnn, where nnn is the number of dot columns for the left margin. For a left margin equivalent to about 10 characters, nnn should be 080. Leading zeros are required.

I have received a letter from Robert Knight of Guilderland, New York, regarding his experience with keyboard problems, that is worth passing along.

"...(I have had a problem), specifically either multi-printing of a letter on a single keypress or no action at all on a keypress. I too have been to the dealer on three occasions for this problem and have been charged for one decoder board, the repair of a decoder board, and finally a recheck at no cost. I took the bull by the horns and took the keyboard off when I seemed to find that the reseating of the connector appeared to solve the problem temporarily. Cleaning the connector contacts seemed to improve the condition for awhile but did not solve the problem.

I checked every solder connection on the printed circuit side of the keyboard and I found three broken connections, two on keys and one on the wires that lie on the board. I soldered these connections about six months ago and have not had any problems ever since. Before paying for a decoder, I would recommend getting a 10X magnifying lens and carefully check those solder joints."

Q. I use the Pascal language with my Apple ][+ and EPSON MX-100 printer. I have a CCS 7710B asynchronous serial interface but cannot print from Pascal. Can you advise me?

A. Pascal should automatically recognize the CCS card as a serial interface if it is plugged into slots 1, 2, or 3. To be used for the printer output function of Pascal the card must be in slot 1. You should have no problem if this is the case. In the Pascal system, slot 2 is reserved for a modem, and slot 3 is used for a terminal (or an 80-column card).

Q. My problem concerns the Pascal error #253, "procedure too long". When I write a program in Pascal, everything is fine until it gets long, then I get the "procedure too long" error even if there is no procedure in it. When I get the error I can eliminate it by reducing the size of the program. Often I can make it run just by taking out 1 or 2 characters. I have seen programs many times longer than mine so the real problem isn't length. Any help you can give me with this problem would be appreciated.

A. Apple Pascal has an internal limitation of 1200 words (2400 bytes) of object code per procedure. Your entire program is a procedure and is causing the "too long procedure" error. The solution is to break up your program into a number of smaller procedures. You can have a lot of small procedures as long as each procedure is smaller than 1200 words of object code. (Object code is what the system produces on compilation of the source code that you type in.)

Q. I have the new Macintosh computer and Imagewriter printer. The printer works fine when I use the high resolution character printing mode from MacWrite, but any other combination, such as using MacPaint, produces garbage on the printer.

A. Some early model printers were shipped with a defective cable. Pin 20 was clipped off on the printer end with the expectation that the XON/XOFF protocol, rather than DTR, would be used with the printer. See your Apple dealer to get a replacement cable.

Q. I have an Apple ][ (Integer Basic in ROM), and a 16K card, generally used to contain Floating-point Basic. I've noticed that several programs on the market specify that one needs an Apple ][+ in order to run their software. More importantly, ProDOS does not support Integer Basic. What alternatives do I have? For instance, can I purchase the Floating-point ROM set, and simply substitute them for the Integer ROMs? Or, can I purchase a Floating-point Firmware card and still be able to use my 16K card? And, who sells it?

A. You can plug the Floating-point Basic (Applesoft) ROMs right into the motherboard of your ][ without any other modifications and have the functional equivalent of an Apple ][+. If you purchase a firmware card I still recommend that you put the Applesoft ROMs on the motherboard, and if you wish you can put the Integer Basic ROMs on the card. And, yes you can use both the firmware card and the 16K card. The best way to do this is to put the 16K card in slot 0 and put the firmware card in another slot. Slot 4 is a good choice because the switch on the back of the card can fit through the slot in the case. For more information on using the firmware card see Dave Morganstein's article "Using Firmware Card in Slot 4", which appeared in the November 1981 issue of the Washington Apple Pi Journal and in the May 1982 issue of Call-A.P.P.L.E. Since the Applesoft ROMs are copyrighted you will have to get them from an authorized Apple dealer.

contd.

# MICROMASTER™

## The MICROPOWER™ System for Applesoft Programmers

Applesoft is easier to use than most computer languages. But Applesoft is missing commands like "print using" found in other versions of BASIC. MicroMaster lets you keep the ease of programming in Applesoft, while adding powerful new commands. A few of them are:

**&PRINT** - a versatile "print using" command that formats numbers quickly and easily. It remembers the last format you gave it; and allows you to use the full syntax of the PRINT command (including strings, SPC(), TAB(), commas and semicolons).

**&INPUT** - "input anything" reads strings -- including commas, colons, quotation marks and other punctuation -- from the keyboard or a disk text file. When INPUTing to a numeric variable, you can type the number, or any valid Applesoft formula.

**&TAB, &CEOP, &CEOL** - Tab, clear to end of page, and clear to end of line commands that work on both 80 and 40-column screens.

**&STORE, &RECALL** - STORE and RECALL values of any array with binary disk file speed. Much faster than using text file commands.

**&SUBSTR** - finds one string inside another string at machine language speed.

**&UNDIM, &SWITCH** - undimension or switch arrays without clearing other variables.

**&FRE** - fast routine collects Applesoft string garbage in an instant.

**&** - "run anything" command knows from the file name you give it whether the file should be RUN, BRUN, or EXECed.

**&APEEK, &APOKE, &MPOKE** - APEEK and APOKE commands handle 2-byte values, and MPOKE makes it easier to poke machine language routines into successive memory locations.

**&GOTO, &GOSUB** - GOTO and GOSUB to variable names instead of meaningless line numbers.

**&ERRMS** - print error messages without halting program execution.

**&MERGE** - a merge that works on machine language as well as Applesoft programs.

### HOW MICROMASTER WORKS:

MicroMaster operates a subroutine library in your computer's memory outside your program. You can use the subroutines we provide, and also add your own subroutines written in either Applesoft or machine language. In addition to calling any library subroutine by name, your program can at any time delete subroutines from the library, load more subroutines into the library from disk, or rename subroutines.

You can use MicroMaster as a library for subroutines you use in your programs (without having to add the code to each program). And you can divide your Applesoft program into segments, and execute each segment as a library routine. For example, execute your program's start-up segment as a library subroutine, and then delete it to make room in memory for large arrays or the later segments of your long program.

MicroMaster works with Applesoft on an Apple II+ or //e with at least 48K of RAM, and DOS 3.3 in its normal location. It works alone, or jointly with your favorite editor (e.g., GPLE or CRAE) and programming tools (like RENUMBER or APA).

MicroMaster brings you over 50 commands! No copy protection to keep you from using it with all your programs. Full after-purchase support, including a periodic newsletter. MicroMaster is available now for \$75 (add \$3 shipping and handling, Maryland residents add 5% sales tax).

### MICROPOWER, LTD.

Call (301) 495-2933 today  
We take VISA & MasterCard

1100 East-West Highway  
Silver Spring, MD 20910

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

Q. How can one write a format file to change parameters (e.g. change margins) for the Imagewriter printer?

A. A format file as defined by Apple, contains control codes that set up the printer for specific jobs. It is not a different kind of file but rather a text file that can be used with a word processor or even a Basic program. When you turn on your printer (any printer, not just the Imagewriter) it sets certain default printing characteristics, i.e. 6 lines per inch vertically, 10 characters per inch horizontally, unemphasized or regular characters, left margin at zero, etc. Suppose for example that you are using your word processor to type letters, and you always want the printer to print in the 12 character per inch mode with a horizontal tab at column 50. You could keep the appropriate control codes in a file that loads into your word processor. If you keep the codes in a separate file you won't have to type them in each time, just load the file, print it and presto your printer is initialized. Or, you could write a short Applesoft program to setup the printer and then this would be your format file. The following program will set the Imagewriter to 12 characters/inch with a tab at column 50.

```
100 PRINT CHR$(4)"PR#1" : REM FOR PRINTER IN SLOT 1
110 PRINT CHR$(27)"E" : REM SET 12 LPI
120 PRINT CHR$(27)"(050." : REM SET TAB
130 PRINT CHR$(4)"PR#0" : REM TURN PRINTER OFF
```

While we're on the subject a short tutorial on printer control codes might be worthwhile. I get a fair number of questions of the type, "How can I set my printer to ---- from Applesoft" or "...from my word processor". To operate the advanced features on most printers you need to send one or more control codes and possibly a character or two to the printer. In the above example to set the printer to 12 lines per inch the manual says to send ESCAPE E. In the Applesoft Reference Manual (also in the printer manual) the ESCAPE control code is found to be ASCII code 27. Thus to send ESCAPE E from Applesoft PRINT CHR\$(27)"E". Sometimes the printer manual will just give you the code number but not the name. This is no problem from Applesoft; the CHR\$ function works with the number not the name. Most word processors however have a special method for embedding control characters in the text. For Apple Writer //e you first press Ctrl-V; then any control character you type, ESCAPE for example, will be entered into the text rather than being treated as an Apple Writer command. (ESCAPE is the key labelled ESC on the Apple keyboard. Apple Writer II works the same way as Apple Writer //e for all control characters except ESCAPE. ESCAPE must be pressed 3 times from Apple Writer II to enter it once into your text.) You must then press Ctrl-V a second time after you have typed the control character to get back to the regular text mode. For the above example you would type this sequence in Apple Writer //e: Ctrl-V, ESCAPE, Ctrl-V, E. This is fine until you run across a printer manual that says to do underlining send the following characters, CHR\$(27)"X". What they are telling you is what the Applesoft command is to set the printer, not what you should type into your word processor. The CHR\$ is an Applesoft function that prints the ASCII code enclosed in the parentheses. You need to find how to print code 27 from your word processor. This will probably require that you look the code up in an ASCII table (see your Applesoft reference manual) to find what keys to press on the keyboard. In this case we already know that code 27 is ESCAPE. For Apple Writer //e you would type: Ctrl-V, ESCAPE, Ctrl-V, X, with no spaces or punctuation between the

characters.

Q. Should the disk drive door be left open or closed when not in use?

A. I am sure that no matter what I say someone out there will take the opposite viewpoint. Seriously though, if you leave your computer on all the time and leave a disk in the drive with the door closed, then if there is a power failure when the computer comes back on it will boot DOS instead of turning on the drive and running forever. On the other hand if you are referring to closing the door when the computer is turned off I am of the opinion that in a reasonably clean environment the disk drive door should be left open. Closing the door with no disk in the drive lets the pressure pad which is coated with oxide residue and other garbage come in contact with the read/write head.

Q. Why can't I do double Hi-res graphics on my ][+. From all I have seen it requires extra memory, but not 64K of it. Is there some way to superimpose both HGR and HGR2 screens at once or preventing the Apple from skipping each other line?

A. In addition to the extra memory in the //e to do double Hi-res, the graphics circuits have been modified. There are actually two Hi-res 1 screens and two Hi-res 2 screens in the //e (with the 64K extended memory) and both Hi-res 1 (or Hi-res 2) screens can be displayed simultaneously to produce 560 X 192 resolution graphics. In your ][+ the circuits are not designed to display both Hi-res screens at the same time, it's one or the other.

However some interesting double Hi-res effects can be done on the ][+ if you realize that turning on the color-shift-bit (the eighth bit) shifts the dots on a monochrome monitor a half-bit position to the right. If you were to plot on Hi-res screen 1 with HCOLOR=3 (or 1 or 2) and plot on Hi-res screen 2 with HCOLOR=7 (or 5 or 6) and then rapidly switch the screen being displayed on the monitor back and forth between 1 and 2 (this would have to be done from machine language) you would have 560 X 192 resolution.

Q. I am trying to interface an Apple /// to the Telex I Service. (Telex I assumes that the terminal is a Teletype using 5-hole punch tape.) I have written the necessary software to communicate with Telex I in its "5-level Code", but I am unable to get the Apple ///'s RS232 Driver to send 5 data bits or find a modem that can work with 5 data bits. Any suggestions?

A. First let's deal with the modem problem. The electrical signals generated by a computer are not compatible with the telephone lines. A modem is simply a circuit to convert the computer signals into tones that can be sent over the telephone. The number of data bits, stop bits, or whatever are of no concern to the modem. You do have to worry about the speed with which the data is sent; however Teletypes communicate at (yawn) 110 baud so that any 300 baud modem (the most popular kind) is more than fast enough for you.

Your problem boils down to not being able to get the Apple /// to output 5 data bits and this is a limitation of the Apple /// RS232 Driver. One way around this would be to buy an RS232 serial card capable of 5 bit operation, but as far as I know such a card doesn't exist. Another solution is to talk to the Bizcomp modem people, they have a soft-

contd.

ware program written for the II that uses the game I/O port to drive the modem and does not require any serial card. It might be possible to modify this for use with the III.

- Q. In regard to the BSAVE command, the length parameter can not exceed 32767. If a file exceeds this length it must be split into 2 files and then saved separately. Is there any way to get around this restriction so the file can be saved under a single name?
- A. Sure, change memory location \$A964 in DOS from \$7F to \$FF. From Applesoft this can be done with POKE 43364,255. This changes the range attribute table in DOS to allow binary files as large as 65535 bytes. &



**TYSONS CORNER CENTER'S**  
**HomeInform**  
 Information on Store Names, Sales, Events, Restaurants, Theatres, Gifts, Metrobus Schedules and Much More

**Computer Access Number**  
**(703) 893-0874**

**COMPUTERS + EDUCATION**  
**= YES! BOOKSHOP**

• TYPING SKILLS • SPEED READING • MATH SKILLS

**450 TITLES OF EDUCATIONAL SOFTWARE**  
**+1400 TITLES OF MICROCOMPUTER BOOKS**  
**1850 REASONS TO VISIT THE YES! BOOKSHOP!**

• GEOGRAPHY • LANGUAGE TUTORIALS • S.A.T. REVIEWS

**DEMO ANY PROGRAM IN STOCK ON OUR APPLE IIe!**  
**YES! BOOKSHOP...1035 31st Street, NW...Washington, DC 20007**  
**Bookshop: (202) 338-7874 Mail Orders: (202) 338-2727**  
**VISA, MC, CHOICE, AMEX ACCEPTED (\$15 MIN.)**  
**Open 10-7 daily, until 9 on Thursdays.**

*Response to 1st ad was terrific.  
 Let's keep #49.95 price 1 more month.  
 HB; PWS.*

introducing

## RECORD MASTER

THE COMPLETE DATA BASE PROGRAM FOR  
 APPLE COMPUTERS - ONLY \$49.95

RECORD MASTER is a full-featured data base program. Use it for addresses, bills, collections, employee data, inventory, investments, sales, student records, or any other data which need order and control.

COMPARE FEATURES	RECORD MASTER	OTHER
750 records - 38 fields	✓	
100 characters per field	✓	
5 field types - alpha, date, numeric, dollar, computed	✓	
Search: and/or, <, =, >, not =	✓	
Wildcard & substring search	✓	
Bulk change/update	✓	
Duplicate record check	✓	
Complete report formatting and print capability	✓	
Mailing labels - up to 5 across 3- and 4-line addresses	✓	
Sort 10 fields simultaneously	✓	
Ascending or descending sorts	✓	
Arithmetic functions: count, average, +, -, x, /	✓	
Complete Editor: add, change, delete fields. Merge and pack files	✓	
Password protection	✓	
64-page detailed manual; indexed	✓	
Unprotected disk	✓	
Menu driven, error trapped	✓	
DOS commands. Standard text files	✓	
No disk swapping. 1 or 2 drives	✓	
Multiple files on a disk	✓	
Easy to learn and use	✓	
PRICE	\$49.95	

And there's much more! RECORD MASTER, under development and test for two years, has all the features of programs costing hundreds of dollars. Yet, you can order it now at the low, introductory price of \$49.95, plus \$1.50 for shipping/handling. Maryland residents add 5% sales tax. For the Apple II+/e. Requires at least 48K.

send check or money order to

384-7875

**BRIDGET SOFTWARE CO.**  
**1309 CANYON RD.**  
**SILVER SPRING, MD 20904**

# EDUCATIONAL SOFTWARE THAT WORKS

by Raymond H. Hartjen

You have read about software exchanges, and you are undoubtedly aware of the high cost of commercial software. Now there is another source of educational software that is low in cost and carries with it the assurance, through statistical documentation, that it has made a significant difference in the educational progress of children. The name of this resource is the National Diffusion Network (NDN). It is supported with your tax dollars and is part of the U.S. Department of Education. The primary function of the NDN is to identify high quality educational programs and assist interested schools in the process of locating and adopting programs that have a proven track record. These validated programs are available to any school in the USA at a cost that one Principal termed, "almost free". The NDN's most repeated goal is to eliminate the reinvention of the wheel.

Today the NDN has funded a dozen computer software programs to enable the originating schools to disseminate (market) their materials to any interested public and private school in the nation. In addition, another group of twenty schools with validated programs has recently completed applications to compete for funding in order that they may also disseminate their materials. The current cluster of 12 computer software programs includes one in computer literacy for middle school students, seven in various dimensions of basic skills, three in the administrative use of computers, and one in special education.

There are several dimensions of the NDN that distinguish it from other sources of software. Most of the NDN programs involve much more than a disk and a student manual. They represent complete teaching systems. Some include teacher guides, student work books, validated pre and post tests, wall charts, etc. Some are limited to a specific computer; others can use any computer available. Some programs are even adapted to meet local school needs. In almost all cases, the school or district must receive training in the various dimensions and components of the program. Follow-up services are also provided to help the school through the start-up periods and afterwards when the school is faced with teacher turnover problems.

This might sound rather burdensome, but an example, for instance, Computeronics, will help to enlighten you. This program was created by a group of talented educators in Tallahassee, Florida. It consists of twenty-five hours of training in Problem Solving with Computers and fifteen hours of training in Computers in Society. (The times noted refer to student course hours.) The training time for teachers is two full days. This time is spent working through the student materials and computer exercises. In order for the training to be effective, the site must have at least one computer for every three teacher trainees. The number of trainees cannot exceed thirty teachers. The two days of inservice is designed to provide an uninitiated teacher with sufficient experience in computer programming and the student materials to enable her/him to feel comfortable leading her students through the program. The recommended classroom ratio of computers to students is eight to one. The Computeronics program is not dependent on any specific computer to function but may, in fact, be used with any computer that can be programmed in BASIC. Educators who have had experience with other computer literacy training comment that the Computeronics training far

surpasses other programs they have attended.

Finally, the NDN has set in place personnel in each state to assist in the search and acquisition of these programs. State Facilitators have been funded to provide service to the educators of their respective states on the model of the agricultural extension agent. In states where there has been a great amount of interest in any particular program, the State Facilitator arranges to have a local educator trained as a certified trainer in that program, thus eliminating the need to arrange for the travel of trainers from the originating site. For instance, Maryland has its own trainer for the Computeronics program. This certified trainer has a strong background in computers and has conducted numerous workshops within the state, in adjoining states, and as far away as Illinois.

The National Diffusion Network offers a unique alternative for the acquisition of well developed computer software curriculum materials and programs for running an effective school. The materials are low in cost and have been demonstrated to be effective in teaching students.

Information about the NDN may be obtained from Dr. Raymond H. Hartjen, the Maryland State Facilitator and Executive Director of Educational Alternatives, Inc. A brochure describing the twelve validated computer technology programs is also available from Educational Alternatives, Inc., P.O. Box 265, Port Tobacco, MD 20677. Please send \$1.00 to cover postage and handling.

## EDSIG NEWS by Peter Combes

### EDSIG Calendar

Tuesday, May 1, at 7.30 p.m.

"Applications of Commercial Software in Education" -- Betty Manchak and Gloria Seelman of the Montgomery County Public Schools System discuss various types of programs available in MCPS, and how they can be used in classrooms and in management.

Tuesday, June 5, at 7.30 p.m.

"Are There Computers After Feminism?" Mary Brown chairs a discussion of:

- the differences in how girls and boys approach computers.
- how to interest women and girls in using computers.

Come prepared to discuss your experiences.

All 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.

# DISABLEDSIG NEWS

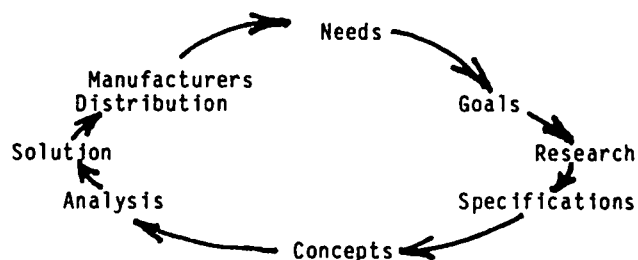
by Jay Thal



\*\*\*\*\*

DISABLEDSIG MAY MEETING  
 THURSDAY, MAY 10, 1984, 7:00 P.M.  
 Chevy Chase Community Center  
 Connecticut Ave. & McKinley St., NW, D.C.

\*\*\*\*\*



In part it seemed like a convocation of speech therapists, but there were also engineers, computer experts, teachers, and parents of learning disabled children at the DisabledSIG's April 12 meeting. It was one of our largest meetings ever, due in part to the interest engendered by the earlier meeting with Laura Meyers (reviewed elsewhere in the Pi), and the speaker of the evening.

Dr. Sally Blackstone, of Baltimore's John F. Kennedy Institute spoke on the subject: Prescribing Communication Aids for Handicapped People. Though the flexible Apple ][ is often one of the considered aids, good practice and the application of engineering models take precedence.

Dr. Blackstone sees the computer as a device which can aid communication, but not a true prosthesis which can overcome a deficit. After all, as an augmentive device, especially through speech synthesis, it can only reach speeds of about 25 words per minute - slow compared to the spoken word. It also fails in that it cannot send the social signals, like body language can, which repair communication breakdowns. Important, too, in prescribing a computer as an aid are factors such as: communicative dissonance (where needs outweigh abilities), environmental considerations (including family stress with the technology), portability, durability, and cost.

Dr. Blackstone shuns the myth that the prescription of a communication system means that one has given up efforts to gain speech. To the contrary, the aid gives power and the power allows the patient to reach deeper within to often attain speech. Also discounted is the myth that there is one best system - and patients often have two or more aids prescribed for given settings.

Prescription is an attempt to find appropriate technologies and the evaluation must take into account the person, the environment, and the system. Factors in personal evaluation include: motor and cognitive abilities; visual and hearing discrimination; vocational goals and leisure activities; the state of dissonance; and, the person's strengths. Environmental considerations are: the family situation; who are the communication partners; funding; and community resources.

Finally the system must be reviewed in terms of the types of inputs and outputs needed, the availability of software, power supply, and whether to use a dedicated system or an off-the-shelf and modified one.

### EVALUATION MODEL

Obviously included in the loop shown below are avenues for feedback, since the patient and technology are constantly changing. Also, product and funding availability factors change, and when you actually try things out they may not work the way the brainstorming sessions thought they might.

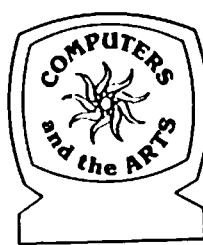
In the example Dr. Blackstone presented that night the Apple was not the immediate choice - it lacked the portability needed, and cost too much. The immediate choice was a Sharp Memowriter - the Apple was targeted for future use as a workstation when funding and other environmental considerations fall into place.

Dr. Blackstone recommends that persons interested in the process of evaluation read Greg Vanderheiden's article in the Summer 1983 issue of Rehabilitation World. She can be reached at the John F. Kennedy Institute, 707 N. Broadway, Baltimore, MD 21205, phone (301) 522-5450.

### AN ADDITION TO OUR LIBRARY

Dr. Theresa Rosegrant, of Arizona State University's Department of Elementary Education has sent us democopies of her Talking Screen Textwriter Program. The program is a word processor for children using an Apple and an Echo II speech synthesizer. Dr. Rosegrant is known for her work with exceptional children, some of which was described in the February 1984 issue of the Smithsonian.


We have placed one of the demo-programs with a school for special education for evaluation. The results of the evaluation will appear in a later issue of the WAP Journal. The other copy will be in our office for review. Further information regarding the program can be obtained by writing: Computing Adventures, Ltd., P.O.Box 15565, Phoenix, Arizona 85060.




## A Unique Summer Experience For Your Child

Session I July 2-13  
 Session II July 16-27

Computer Programming in BASIC and LOGO  
 Computer Graphics and Music  
 Word Processing  
 Daily Swimming and Sports  
 Art, Music, and Creative Writing



- BETHESDA
- SILVER SPRING
- GAITHERSBURG
- ARLINGTON



986-4929

# A MEETING WITH LAURA MEYERS

by Jay Thal

On April 3, 1984, the Education and Disabled SIGs held a joint meeting. And, what a meeting it was! Those of you who missed it, missed a presentation of great importance. Moreover, you missed Laura Meyers.

"So who is Laura Meyers?", you ask. Too easily someone might answer: "She's a PhD in the field of linguistics who works with disabled children." But she is more than that - part prophet, part iconoclast, she is achieving success teaching children who have been labeled unteachable.

Conventional wisdom sees the disabled as a breed apart, struck by some, often inexplicable, fate that has effected them physically, mentally, or emotionally. Conventional methodologies try first to remedy the perceived problem, make the patient whole so that they can reenter "our" world. But, since the disabled cannot regenerate limbs or make up for some of their deficits, society generally views them as second class citizens. A large part of disabilities is a social construction - we think of people in casts differently than we do people in braces!

And the loss is never so tragic as when we label a child disabled - the tag lasts a lifetime. Abilities are obscured by the apparent problems. Expecting less, we demand less and the child is never challenged to his or her full potential. Many professionals, therapists, and families in turn succumb to the myths.

"So what", you say, "does this bit of philosophical musing have to do with Laura Meyers?" Well, let me tell you. Laura Meyers believes, along with some others, that ALL children develop along certain age-appropriate patterns. While it may be an oversimplification, disabled children are no different. It is just that their disabilities add a dissonance to their lives, and the lives of those around them, and well meaning "help" further retards their development. To overcome that dissonance Meyers uses microcomputers. She works with infants; children with Down's syndrome; cerebral palsy; and, other medical involvements. She uses speech synthesis for children without language. And, instead of waiting for the children to independently develop their physical skills so that they can be trusted on a sensitive machine like a computer, she finds that the children's physical controls improve BECAUSE they are using the computer.

The computer gives the children control over their environment - it gives them speech, and Meyers says: "They understand immediately that the computer is giving them control over their own meaning." But it does not end with synthesized speech. Children who had few utterances, of whom questions were raised regarding their cognition of words, start talking. Given the power, the children strive for power beyond that which the computer can give them - the power suppressed within them. It's the "power principle": the intervention empowers the learner to perform meaningful projects that could not be done without it. (1)

Once the suddenly learned skill is internalized, the computer can be "discarded" for that particular purpose. It is a principle which Meyers refers to as Scaffolding: a supportive tool that extends the child's skills, thereby allowing the learner to accomplish a task not otherwise possible. (2)

And, its more than just theory. We saw it in operation through videotaped replays of some of Meyers'

sessions. Meyers uses an Apple ][, with an Echo II speech synthesizer and a touch sensitive board. Technologically, it's nothing new but the approach and the critical software element are. With the assistance of Anthony Villano and Jim Day, both programmers, Meyers developed software called "Programs for Early Acquisition of Language" (PEAL).

"Child's play" is an often misunderstood term. It is the work of children, the learning of childhood. PEAL applies the power of play to teach children who cannot play in conventional ways. Play is meaningful, it is a preparation - doctor, explorer, and (unfortunately) warrior - even emptying the contents of mother's purse. Based upon the dynamics of culture, PEAL applies the added principle of Cultural Resonance. (3)

With these and other principles in mind, Meyers and PEAL turn the computer into a unique language prosthesis which provides:

1. ACCESS - through special keyboards & switches.
2. SPEECH - for non-oral or the speech-delayed.
3. READING - reinforced with speech for non-readers.
4. WRITING - letters and words voiced as typed.
5. PRINTING - hard copy for those who can't control a pencil.

For those of us who have been seeking software for children who learn differently, PEAL appears to have secured the beachhead. It is now for Meyers and others to push the front lines forward. Needless to say, this writer, as well as the rest of the large audience, was impressed. Laura Meyers can be reached, for further information, by writing her at PEAL Software, 1725 Promenade, Suite 323, Santa Monica, CA 90401.

Several recent articles on Meyers are:

Trachtman, P., "Putting computers into the hands of children without language", Smithsonian, February 1984.

Ray, C., "Breaking the Silence", Softalk, April 1984.

References:

- (1) Papert, S., Mindstorms; Children, Computers and Powerful Ideas, Basic Books, 1980
- (2) Greenfield, P. and Smith J., The Structure of Communication in Early Language Development, Academic Press, 1976
- (3) Papert, ibid.

Additional References:

Gliedman, J. and Roth, W., The Unexpected Minority: Handicapped Children in America, Harcourt Brace Jovanovich, 1980.

Meyers, L. (Guest Ed.) "Augmenting Language Skills with Microcomputers", Seminars in Speech and Language, Vol. 5, No. 1, March 1984.

Meyers, L. "Unique Contributions of Microcomputers to Language Intervention with Handicapped Children", ibid.

Silverman, F., Communication for the Speechless, Prentice-Hall, 1980.



# PLAN AHEAD

## See...Shop...Save

# Washington, DC Computer Showcase Expo



## SET ASIDE THESE DATES

**May 3-6, 1984 • Washington, DC Convention Center**

Thursday, Friday and Saturday — 10 am-7 pm • Sunday — 10 am-6 pm

COMPUTER SHOWCASE EXPO... the computer show designed specifically to help local area business owners, corporate managers and professionals do business more efficiently and profitably through the utilization of personal computers and small business systems.

### Hundreds of Personal Computer Products on Display

Come. Compare. Buy. Learn from the experts. Leading local, regional and national suppliers exhibit a broad selection of the computers, related products, small business systems and services you want and need. **SAVE HUNDREDS OF DOLLARS ON "SHOW SPECIALS"** at Washington, DC's **COMPUTER SHOWCASE EXPO!**

**ADMISSION: Adults \$7.50 — Children (accompanied by an Adult): \$2.50**  
(Bring this ad to the Box Office to receive \$2.50 off the Regular Adult Admission price of \$7.50)



**THE SMALL COMPUTER COLLEGE™**... Informative seminars presented to help potential buyers identify needs in order to purchase computer systems and software.



Register to be eligible to win a personal computer.



For more information  
**(305) 484-6800**

Presented by THE INTERFACE GROUP, Inc., World's leading producer of computer conferences and expositions.  
300 First Avenue, Needham, MA 02194 Telephone (617) 449-6600 Regional Office: 4700 North State Road 7, Suite 121 Fort Lauderdale, FL 33319 (305) 484-6800

# LOGOSIG NEWS

by Nancy C. Strange



## TURTLE TRACKS

### LOGOSIG SCHEDULE

April 28 -- Logo Project Teams  
May 5 -- LOGO IN OUR SCHOOLS  
(Time Change, 9:30-12:00)  
June 23 -- to be announced  
July 28 -- LOGO CELEBRATION

### LOGOSIG HOLDS ROUND-TABLE DISCUSSION FOR TEACHERS

The LOGOSIG is sponsoring a special meeting "LOGO in Our Schools" on May 5, 1984. Representatives of the area public schools will make up our panel, who will discuss how they are using computers and Logo.

### LOGO IN ARLINGTON (by Jo Winkelmann)

LOGO experiences in Arlington, VA are so new that currently no rumblings are heard to "grade" the process. LOGO programs are as diverse as the staff. Educational leadership needed to begin staff development courses was provided by the Elementary Principals Task Force on Computers and a parent. To date, 49 staff members (pre-school, elementary, junior high and high school teachers, librarians, teachers aides, etc.) have been introduced to LOGO in a three-hour graduate level course taught by C. Diane Martin and Rachelle S. Heller. The model is designed to encourage staff to present LOGO to children as they themselves learned the philosophies and the language.

Each of the 19 elementary schools has Apple //e computers and APPLE LOGO. At least one teacher from 18 elementary schools has participated in INTRODUCTION TO LOGO FOR EDUCATORS. LOGO is also available at the Career Center on hard disk with a Corvus/Omninet system.

For the most part, computers are mobile (on carts) and computer instruction occurs within the classroom setting. LOGO has been used to reinforce principles of mathematics as well as to learn poetry. LOGO is also found outside the classroom being used in creative ways; a dedicated librarian provides LOGO experiences before and after school for children.

Currently in the planning stages is a summer LOGO workshop to be held at Taylor Elementary School for 30 staff and 60 children who will be entering grades 5 or 6 in the Fall of 1984.

Children are motivated and staff are enthused about the potential of LOGO as a teaching tool.

### LOGO VIRUS by Ron Murray

Ron writes the first of a multi-part article on Logo Virus concepts. In it he shares how to use the DEFINE and NAME Logo commands to Change, Insert, Copy, and Delete elements from a list and therefore from procedures. The reader is encouraged to try his FOR THE READER assignment (the answer will be given next month). Anyone interested in learning more should come to the LOGOSIG meeting and attend Ron's advanced project team.

Logo Virus, or: I Have A Code In My Nose (part I)

This month's article is not derived from random

thoughts collected while watching "The A Team" on television, but from ideas accreted while recovering from a bad head cold.

### THE SITUATION

In the midst of antihistamines and kleenex, I began to think about how viruses (the source of my misery) work their wiles on other organisms. Viruses have been referred to as "parasites at the genetic level". When a virus enters a cell, it breaks down, releasing its chromosome(s) into the cytoplasm of the unfortunate cell. This chromosome serves as a template to direct the synthesis of new viruses. The virus manages this feat by providing a new set of instructions to the machinery responsible for the functioning and reproduction of a cell. The infected cell manufactures copies of the virus rather than doing what it's supposed to do. This creates problems for the cell as well as for the organism of which the cell is a part. The study of viruses provides an interesting view into genetics and provides an opportunity for us to think about creating, sending and destroying instructions in a particularly Logo fashion.

A computer program can also be seen as a set of instructions that eventually produce a desired result, much like the sequences of nucleic acids in DNA (DeoxyriboNucleic Acid) eventually direct operations within a cell. When we write and run our procedures we spend some time in making sure that the procedure does exactly what we want it to, with few side effects or surprises. Because this creative process is under our control, we usually don't have to worry whether or not our efforts will be subverted by some smart aleck process that changes our procedures to suit its purposes. That doesn't mean we should not think about how such a thing could happen, or construct procedures that would do so. Pursuing this line will teach us a little about writing procedures that modify themselves or other procedures, and that can work behind the scenes to produce subtle or fatal "mutations" of previously defined procedures.

We will try to borrow some ideas from genetics, and apply them to a "Logo virus" microworld.

### THINKING ABOUT THE TOOL

Think of a procedure as a set of instructions that tell the computer how to accomplish a certain task. A "Logo virus" would be a procedure that somehow alters the instructions of another procedure, for whatever purpose. Our virus might accomplish this by:

**DELETING INSTRUCTIONS** - The virus might cause the loss of specific or random instructions in the procedure, also yielding "good" or "bad" results.

**COPYING INSTRUCTIONS FROM ONE PLACE TO ANOTHER** - Though this is not necessarily something a virus might do, there is evidence that copying somehow takes place. Let's include it.

**INJECTING A NEW SET OF INSTRUCTIONS** - If the virus could place a new set of instructions in the procedure, all sorts of unexpected things could result, even the production of new "viruses".

**CHANGING CERTAIN INSTRUCTIONS** - The virus might change one or several instructions; though this might not

contd.

immediately produce new viruses, it might predispose the procedure to a successful attack later. It would produce a "better" or "worse" procedure, according to some criteria we will set.

With Logo, we can create an "environment" in which the four activities described can occur, giving the impression of a "Logo virus" infecting certain other procedures in the environment.

#### SOURCES OF IDEAS

**BIOLOGY TEXTBOOKS** - A short visit to the nearest biology text is recommended for those of us who have been out of school for a while. I happen to have on my bookshelf another unreadable book titled "the Molecular Biology of the Gene" by Mr. DNA himself, James D. Watson. In such books one can read about the genetic code, the mechanism of infection by viruses, and about genetics in general. They are good sources of ideas, some of which can be "expressed" by Logo procedures.

**LOGO REFERENCE MANUAL** - A brief review of the Logo Language Reference Manual presents us with a Logo command called `DEFINE :NAME :LIST`, and an operation called `TEXT :NAME`. They act to turn lists into procedures (and vice versa) in a way that will allow us to create a Logo virus.

To better explain these two operations, we will first construct a simple procedure:

```
TO DOUBLE :NUMBER
  PRINT [HI, MOM]
  PRINT :NUMBER * 2
END
```

This procedure requires a number input and prints out the phrase "hi, mom", followed by two times the number that was input. If we typed:

```
DOUBLE 3
```

the result would be:

```
HI, MOM
6
```

When we define procedures we usually find ourselves using the Logo editor. Another way to define procedures would be to use the command called `DEFINE :NAME :LIST`. `DEFINE` requires two inputs. The first input is the name of the procedure to be defined, and is preceded by a quotation mark ("). The second input is a special type of list: the first element of the list is a list of the inputs, with no dots (:) before their names. If there are no inputs, the empty list [] is used. Each subsequent element of the second input is a list consisting of one line of the procedure definition. This list does not contain `END`, since `END` is not part of the procedure definition.

This is how we would create the procedure `DOUBLE` using the `DEFINE` command:

```
DEFINE "DOUBLE [ [NUMBER] [PRINT [HI, MOM]] [PRINT :NUMBER * 2] ]
```

As mentioned, the first input ("DOUBLE"), is the name of the procedure. The inputs to the procedure are the first element in the procedure definition list, with each line in the procedure enclosed in brackets.

What we have now is a command that lets us define procedures using list processing activities that we have used on other occasions. If we want to have a previously defined procedure placed in this special and useful list form indicated above, we need only use the `TEXT` operation. The text operation requires the

name of a procedure as an input and returns a list that is organized in exactly the same way as the second input to `DEFINE`.

So if we type `SHOW TEXT "DOUBLE`, we get:

```
[ [NUMBER] [PRINT [HI, MOM]] [PRINT :NUMBER * 2] ]
```

`SHOW` gives us a result with the right number of brackets around it. Compare this result with the second input of `DEFINE` above.

Now that we know how to change lists into procedures and procedures into lists we can turn to the "virus" problem. It is most useful to think of a procedure as a simple type of list. Procedures that we develop for operations on lists in general will therefore work for lists that just happen to be definitions of procedures.

#### GENERAL LIST PROCESSING SPECULATIONS: (1)

##### ● HOW DO WE DELETE ELEMENTS FROM A LIST

We can construct a procedure that will delete an element from a given position within a list. `DELETE :POSITION :LIST` will return a list from which the element located at location `:POSITION` of a given list is deleted. For example:

```
PRINT DELETE 3 [A B C D E]
```

returns the list

```
[A B D E]
```

##### ● HOW DO WE COPY ELEMENTS FROM A LIST

We can construct a procedure that will copy a word or list from a given place within a list. `COPY :POSITION :SIZE :LIST` will return a list consisting of `:SIZE` elements, starting at location `:POSITION` and moving to the right.

```
PRINT COPY 2 4 [A B C D E F G]
```

returns the list

```
[B C D E]
```

##### ● HOW DO WE INSERT A WORD OR LIST INTO A LIST

We can construct a procedure that will insert an element in a given position within a list. `INSERT :ELEMENT :POSITION :LIST` will insert a word or list (:element) at a given location (:POSITION) inside a of a given list.

```
PRINT INSERT [TEST IT] 4 [A B C D E F G]
```

returns the list

```
[A B C [test it] D E F G]
```

##### ● HOW DO WE CHANGE AN ELEMENT OF A LIST

We can construct a procedure that will change an element in a given place within a list. `CHANGE :POSITION.LIST :ELEMENT.LIST :LIST` requires three lists as inputs. The first list `:POSITION.LIST` indicates the locations in the given list where there are to be changes made. The second list, `:ELEMENT.LIST` contains the elements that are to be placed in the locations indicated by `:POSITION.LIST`. They must be in the same order as in `:POSITION.LIST`. The third list, `:LIST`, is the list to be changed.

```
PRINT CHANGE [1 3 5] [Z Y X] [A B C D E F G H]
```

contd.

therefore returns the list

```
[Z B Y D X F G H]
```

We now have operations that let us delete, copy, insert, and change elements of Logo lists. When we combine these operations with DEFINE and TEXT, we can write procedures that will modify other procedures in the Logo workspace.

EXAMPLES: We will use the procedure DOUBLE :NUMBER above.

DELETE PORTIONS OF OTHER PROCEDURES -

DEFINE "DOUBLE DELETE 2 TEXT "DOUBLE  
changes the definition of DOUBLE :NUMBER to:

```
TO DOUBLE :NUMBER
  PRINT :NUMBER * 2
END
```

It does so by deleting the second element in DOUBLE's list of instructions, and redefining the remainder as DOUBLE :NUMBER.

COPY PORTIONS OF PROCEDURES -

DEFINE "DOUBLE COPY 1 2 TEXT "DOUBLE  
changes the definition of DOUBLE :NUMBER to:

```
TO DOUBLE :NUMBER
  PRINT [HI, MOM]
END
```

It does so by copying only the first and second elements from DOUBLE's definition list, and redefining that list as DOUBLE.

INSERT PORTION OF PROCEDURE -

DEFINE "DOUBLE INSERT [PRINT [APPLE PI]] 2 TEXT  
"DOUBLE  
changes the definition of DOUBLE to:

```
TO DOUBLE :NUMBER
  PRINT [APPLE PI]
  PRINT [HI, MOM]
  PRINT :NUMBER * 2
END
```

It does so by inserting the list [PRINT [APPLE PI]] in location 2 of DOUBLE's definition list, then redefining the whole thing as DOUBLE.

DEFINE "DOUBLE CHANGE [3] [ PRINT :NUMBER \* 3] TEXT  
"DOUBLE  
changes the definition to:

```
TO DOUBLE :NUMBER
  PRINT [HI, MOM]
  PRINT :NUMBER * 3
END
```

It does so by changing the third element in DOUBLE's definition list and defining the result as DOUBLE.

FOR THE READER!

Figure out this new definition of DOUBLE:

DEFINE "DOUBLE (INSERT (BF TEXT DOUBLE) 3 TEXT  
"DOUBLE)

IMPLICATIONS AND FUTURE ACTIVITY

These procedures hint at Logo's symbol manipulating power in ways not even mentioned in elementary or intermediate texts or classes. The type of problem is

not a typical "computing" problem, like calculating square roots to 1000 or drawing ellipses on the screen. The possible results are not even predictable, but the general outcomes are known. This type of problem is one which cries out for list processing.

Try to do the equivalent of just those four procedures above in any other language (LISP would be cheating). When you come down off the wall, take another look at Logo and come on over to our SIG meeting. Being able to represent definitions of procedures as lists and being able to modify these definition lists with other procedures offers tremendous possibilities for the creation, modification and destruction of procedures by other procedures. The examples above did not even use regular list processing operations, such as first, butfirst, etc.

Play with these procedures for a while and get a feel for their possibilities; the next article will concern itself with the creation of the virus itself, and with the development of a procedure in which it will operate.

LOGO FEATURES USED:

NEW COMMANDS AND OPERATIONS

```
DEFINE
TEXT
```

RECURSION

LIST PROCESSING

References:

- (1) The procedures introduced here will be discussed in detail by the advanced Logo group during this month's Logosig meeting.)
- (2) Apple Logo Language Reference Manual.
- (3) The Molecular Biology of the Gene. James D. Watson. W.A. Benjamin, 1970.

LISTINGS

```
PPROP ".SYSTEM "BURY "TRUE
TO CHANGE :POSITION.LIST :ELEMENT.LIST :LIST
  IF EMPTY :LIST [OP []]
  IF EMPTY :POSITION.LIST [OP :LIST]
  IF EMPTY :ELEMENT.LIST [OP :LIST]
  MAKE "X :LIST
  CHAN :POSITION.LIST :ELEMENT.LIST :LIST
  OP :X
END
```

```
TO COPY :POSITION :SIZE :LIST
  IF :POSITION < 1 [OP []]
  IF :SIZE < 1 [OP []]
  IF (:POSITION + :SIZE) > COUNT :LIST [OP []]
  IF EMPTY :LIST [OP []]
  OP COP :POSITION :SIZE :LIST
END
```

```
TO COP :POSITION :SIZE :LIST
  IF :SIZE < 1 [OP []]
  OP FPUT (ITEM :POSITION :LIST) (COP :POSITION + 1
  :SIZE - 1 :LIST)
END
```

```
TO INSERT :ELEMENT :POSITION :LIST
  IF EMPTY :LIST [OP []]
  IF :POSITION = 1 [OP FPUT :ELEMENT (INSERT :ELEMENT
  :POSITION - 1 :LIST)] [OP FPUT FIRST :LIST (INSERT
  :ELEMENT :POSITION - 1 BF :LIST)]
END
```

contd.

# JOB MART

## POSITION WANTED

DOCUMENTATION WRITING SERVICE - for computer program authors whose programming skills are not matched by American Standard English writing skills. Fees commensurate with complexity, but with compassion as well. Author of two published textbooks, magazine articles and program reviews. Call Bob, 262-1355, day or evening.

## HELP WANTED

PARTTIME TYPIST. Name your hours in a convenient N. Arlington location (near Chain Bridge). Must be a good typist, preferably familiar with Apple CP/M and Word Star, or will train. Call 241-5542.

MINUTEWARE, publisher of the Minute Manual for Apple Writer //e and ][+, is looking for someone to do WPL programming. The WPL programming will be included in a WPL Minute Manual product. Must be familiar with WPL, work fast, and be willing to work for royalties based on sales of the product. Call Jim Pirisino, Columbia, MD (301) 995-1166.

UNIX/C PROGRAMMERS/INSTRUCTORS wanted. Please call KRACO, 762-3333.

GRAPHICS SPECIALIST wanted to incorporate sound and animation techniques for educational games. Call Judy Priven at 320-2829.

BYTE Magazine's Book Review Editor, Margaret Cook, is looking for book reviewers. Reviewers get paid \$100 for four to five pages on a technical work. If you would like to be considered, send her an example of your published work, with a resume. If there is a particular book you would like to review, send her that information also. 70 Main Street, Peterborough, NH 03458.

```
TO DELETE :POSITION :LIST
IF EMPTY :LIST [OP ]
IF :POSITION = 1 [OP DELETE :POSITION - 1 BF :LIST]
[OP FPUT (FIRST :LIST) DELETE :POSITION - 1 BF :LIST]
END
```

```
TO CHAN :POSITION.LIST :ELEMENT.LIST :LIST
IF EMPTY :POSITION.LIST [STOP]
IF EMPTY :ELEMENT.LIST [STOP]
LOCAL "NUM
MAKE "NUM FIRST :POSITION.LIST
MAKE "X DELETE :NUM :LIST
MAKE "X INSERT (FIRST :ELEMENT.LIST) :NUM :X
CHAN (BF :POSITION.LIST) (BF :ELEMENT.LIST) :X
END
```

```
TO DOUBLE :NUMBER
PRINT [HI,MOM]
PRINT :NUMBER * 2
END
```

⊗

# CLASSIFIEDS

DONATIONS WANTED. Apple ][ or ][+ for Garrison Forest School. Call Gregory Cukor at (301) 363-1500. Will provide receipt for income tax purposes.

WANTED. Apple parallel interface board and cable for Paper Tiger IDS-440. Ray Thelwell, 972-1614.

FOR SALE. AJ 830/831 printer, 30 cps, 14 7/8 in. carriage. Printer will backspace, underline and double strike. Includes tractor feed, two built-in shelves, 10 black ribbons, 6 colored ribbons (red, brown and blue), 9 different print wheels. Valued at \$1077 current cost. Has been under maintenance contract. \$800 or best offer. Also includes one month free service (by AJ) at your home. Call Lee Raesly, 460-0754.

FOR SALE. TI Programmer electronic calculator with power supply, instructions, carrying case. In mint condition. Make offer. Also, boat 2-way VHF radio, Standard Horizon 78, new in carton. Asking \$400 or best offer. Call Bob, 262-1355, day or evening.

SOLD MY //e, now selling peripherals, programs, books, magazines. Quadram 64K buffer, \$200. Vista 6 megabyte cartridge disk system, \$750. PCPI 6 mhz CP/M system, 128K extra RAM, many programs, \$550. dBase II, \$250. Apple Pascal, \$150. Much more. Lofty Becker (202) 338-5217, evenings and weekends.

FOR SALE. VisiCalc, advanced version for //e. Never used, still factory sealed in plastic, \$200. Bill Cavanaugh, (301) 258-6812 days, and (301) 977-3660 evenings.

FOR SALE. Grappler+ graphics card, like new (bought Xmas '83, got //e with a second one and don't need two). With manuals, etc. Extra bonus: graphics demo software disk. List \$175, will sell \$110 or best offer. Bud Gropper, 384-9150 (home) or 724-7631 (work).

⊗

## CONSULTING & TEACHING



DATA BASE • WORD PROCESSING  
BUSINESS PROGRAMS • ACCOUNTING PROGRAMS  
BUSINESS SYSTEMS

• OPERATION PROCEDURES WRITTEN •

PERSONAL TUTORING  
\*CLASSES TAUGHT FOR YOU\*  
OFFICE CENTERED TRAINING

*Staff Hand-Holding a Specialty!*

Leon H. Rasely, L.C.S.W. **LRO** Inc.  
460-0754

# A PAGE FROM THE STACK

by Robert C. Platt

As Spring arrives, the WAP software library continues to search for new additions. As you can see from the order form at the back of the Journal, we take every conceivable type of program - including Apple III and Macintosh programs. Remember, for each disk of software that you donate, you get a free library disk. So, if your Spring cleaning extends to straightening up your disks, donate copies of your creations to the WAP! I will be at the "Disketeria", our sales table, from 9-10 before the monthly meetings to discuss possible contributions.

This month's column covers adapting WAP library programs to run on your printer, donation procedures, and corrections to the inventory program on Disk 104.

## PRINTER CONTROLS

Most of the programs in our library will operate on Epson printers. Having never owned an Epson, I sympathize with callers who find that the programs cause their printers to do bizarre things. The following chart will help you convert library programs to run on your printer:

Printer Function	Epson	Okidata	Prowriter Imagewriter Apple DMP
17 cpi	CHR\$(15)	CHR\$(29)	CHR\$(27);"Q"
Normal 10cpi	CHR\$(18)	CHR\$(30)	CHR\$(27);"N"
Emphasis	Esc;"E"	Esc;"T"	Esc;"!"
Emphasis Off	Esc;"F"	Esc;"I"	Esc;CHR\$(34)
Double width	CHR\$(14)	CHR\$(31)	CHR\$(14)
Single width	CHR\$(20)	CHR\$(30)	CHR\$(15)

Please note that in this table, Esc means CHR\$(27).

## LIBRARY ACQUISITIONS

Given the number of calls I get on the subject, you may be interested in how programs are added to WAP's collection. First, we ask you to submit the program on a floppy disk (or a 3 1/2" disk for Macsoftware.) Your submission need not be original, but we ask that you identify its source. For example, you might have a useful program that you typed in from a magazine. Some magazines like Softalk have generously given us permission to distribute their programs. But other magazines, such as Nibble, who actively sell disks to accompany their listings, are generally unwilling to give us permission. In general, we also try to avoid disks purchased from other clubs. The reason for this is that we need their permission to distribute their materials, and we already have an exchange arrangement with most of the major Apple user groups. But if you have made an improvement on either a program from the WAP library or from some other club, we welcome it.

When you submit your contribution, we ask you to fill out a simple questionnaire telling us about the programs and also to sign a copyright license which gives WAP permission to distribute your material. (We've just redesigned these forms on the Mac to improve their appearance.)

All submissions are routed to a librarian for review. I review the DOS and ProDOS contributions, Mike Hartman handles Pascal, and Kit Leithiser and Dave Neumann get CP/M disks. Tony Anderson and Ron Grognet have volunteered to review Macsoftware. The reviewer uses

the following criteria:

- 1) Is it already in the library?
- 2) Does the program run?
- 3) If there are bugs, can they be easily fixed?
- 4) Does the program have a copyright notice? Do we have permission to distribute the program?

I also try to give priority to programs which accompany articles in the WAP Journal, so as to save members the trouble of rekeying from Journal listings. In general, WAP has not been very selective in deciding which programs to include in the library. Nor can we guarantee that our quick check of a program means that it is bug free.

That is where you can help. If you have a problem with a library program, please fill out an "Error Report Form." You will find the form in the back of your New Members Reference Book or at the WAP Office. If you give us enough details, we will contact the program's author and try to get the program fixed.

Remember that we are trying to expand the scope of the library and to fill up theme disks. For example, I am currently trying to fill a "music" disk and a "game" disk. Help us by donating!

## ADVENTURE BUGS

We are fortunate to have a version of the original Adventure game in our library (Disks 114A and 114B). I have never played the game, but I get rave reviews from people who enjoy such games. The problem is that our most recent disks appear to have a variety of errors, either "out of data" or missing Ctrl-D's in DOS commands. If anyone has a functioning or corrected copy of Disks 114A and 114B, please call me so that I can correct this problem.

## VANISHING DISK?

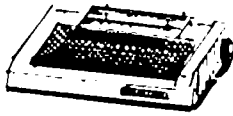
Last month, I reported on the debut of Spreadsheet template Disks 90-92. However, Walt Francis' Visi-Column inadvertently included a writeup of Disk 93, the next installment in the series. That disk is still being processed and will be added to the library as soon as possible, probably at the end of May.

## DISK 104: INVENTORY

Perhaps the most unusual bug reported to date came from Bob McDevitt, who reports that the Inventory program on Disk 104 does not contain any disk input/output instructions. Sure enough, all disk operations are expressed as comments, rather than PRINT statements with DOS commands. I supplied the required DOS commands, and have the start of a working program. However, this program obviously was not created for the Apple ][. (Maybe it was typed from a magazine?) As a result, even with the new DOS commands, a user will want to customize the screen and printer display formats. If someone comes up with a slick format, please donate it to the library. &

**SPECIAL!**  
**Limited Time Only**  
**Gemini-10X**

Incredible print quality, 120 cps, friction and tractor, dot graphics.



**\$289.00**



**THE MICRO CONNECTION**  
**LOWEST PRICES AROUND! HERE'S WHY:**

**... We Don't Play Games with Your Money!**

1. 3 STORE LOCATIONS
2. LARGE VOLUME BUYING
3. LOW OVERHEAD
4. SUPPORT & SERVICE

**SOFTWARE SPECIALS**

*Prices Good thru April 15, 1984*

Tax Advantage — Apple . . . . \$60	Tax Wizard — Apple . . . . . \$49
Tax Preparer — Apple . . . . . \$192	Tax Manager — Apple . . . . . \$153
Home Accountant — Apple . \$59	The Handler . . . . . \$110

**COMPUTER HARDWARE**

<b>• DISK DRIVES</b>	
Disk Drive — Rana	\$279
Dana Disk Drive — Apple	\$239
10 MB Hard Disk — Corvus	\$2,295
<b>• MODEMS</b>	
H. Micromodem IIe, II+	\$269
Anchor — MK XII	\$399
Hayes S/M — 300	\$225
Apple Cat II	\$269
Smart Model 1200 — Hayes	\$499
<b>• SOFTWARE</b>	
Zaxon	\$32
Master Type	\$32
Multiplan	\$189
One-On-One	\$34
Sensible Speller	\$99
Ascii Pro	\$109
Koala Pad	\$99
Flight Sim II	\$43
Flip n' File — 5 1/4"	\$19

<b>• COMPUTER SYSTEMS</b>	
Franklin Ace 1200 Sys.	\$1,899
Corona PC-2	\$2,695
Franklin Fam. Pac	\$1,099
<b>• PRINTERS</b>	
Gemini Delta	Call
Epson FX80 (160 cps)	Call
Epson 12x80	Call
Gemini 15	Call
Okidata 92P	\$449
Epson FX100 (160 cps)	Call
<b>• ACCESSORIES</b>	
System Saver	\$69
Applicard + Wordstar	\$399
16K RAM Card (M/Soft)	\$78
Computer Desk	\$125
80 col. crd. (Videx)	\$250
Micro-Buffer II — 16K	\$179
Comp. Paper (9 1/2 x 11)	\$19
Kraft Joystick	\$49

<b>• DISKETTES</b>	
Wabash (SS/SD)	\$20
Verbatim (DS/DD) (10)	\$40
<b>• PRINTERS</b>	
Gemini 15X — 120 cps	Call
Grappler + (No Buffer)	\$130
Par. Interface	\$79
Grappler + 16K Buffer	\$199
Okidata 93P	\$729
<b>• PRINTERS (Ltr. Qual.)</b>	
Gemini Prototype	Call
Silver Reed — Exp 550	\$599
F-10 (35 cps)	\$1,195
<b>• MONITORS</b>	
Gorilla GRN — 12"	\$99
Gorilla — Amber 12"	\$125
AMDEK 300G — GRN	\$159
AMDEK 300A — Amber	\$165
Color 1 13" (AMDEK)	\$299

All orders subject to product availability. Merchandise cash prices listed — credit cards 3% extra.

**3 FRIENDLY LOCATIONS TO SERVE YOU**

Store Hours: 10 AM to 6 PM Monday thru Saturday

**DON'T OVERPAY — CALL THE DISCOUNTERS AT:**

**THE MICRO CONNECTION**

320-A MAPLE AVENUE, EAST  
 VIENNA, VA 22180  
**281-1866**

6192 FRANCONIA RD.  
 ALEXANDRIA, VA 22310  
**971-5800**

8228 FENTON STREET  
 SILVER SPRING, MD 20910  
**495-5960**

# SIG MAC NEWS

by Ellen L. Bouwkamp



SIG Mac's second meeting opened with a decision to continue to hold the principal meeting after the main meeting and a second monthly meeting on the first Thursday of each month at the Washington Apple Pi office, beginning in May. This meeting will be devoted to programming. Tom Warrick was elected chairman; Ellen Bouwkamp, columnist; and Tony Anderson and Ron Grognat as disk librarians.

The three books available about the Macintosh are deemed by persons who have read them to be valuable primarily to those who are considering the purchase of a Mac. Current Macintosh owners were advised not to purchase them.

The SIG voted to request from Apple Computer, Inc., a free copy of Inside Macintosh, which describes the Mac's ROM routines. If this request is refused, the SIG will request that Washington Apple Pi purchase a copy for the club library. It is hoped that this will encourage WAP members to write public domain software for the Mac. Currently there are no reference works for the Mac other than Inside Macintosh. The manual gives lots of pictures and a little text but is poorly indexed.

As of the meeting, only MacWrite, MacPaint, Microsoft BASIC and Microsoft Multiplan are available to Mac owners. Sometime this summer a 68000 assembly language assembler/debugger for the Mac will be released. MacBASIC is reviewed in Macintosh! Complete by Doug Clapp. MacPascal, an interpretive Pascal, is also a future programmer's tool.

There are several Apple Certified Software Developers in WAP. They are planning to get together to explore the possibility of a cooperative that will enable members to buy and sell items that WAP cannot buy or sell. For more information, contact Bob Platt at 223-1588.

Michael Esserman, (301) 277-6582, is interested in being contacted by anyone who can help adapt the Mac and its software so that blind persons can use it.

There are two released version of Microsoft's Multiplan, with release dates of January 23 and January 31. The former should be regarded as unreliable, and even the latter has serious bugs in it. To find out the date of a particular version, open the disk, select the Multiplan icon and use the "Get Info" to see when the file was created. InfoWorld rated Multiplan as "excellent."

Microsoft BASIC programs can be edited using MacWrite, then saved as text for execution.

Compuserve's Micronet Apple User Group's (MAUG) Mac/Lisa software and Mac/Lisa hardware are excellent current sources of information about the Mac. SIG Mac will attempt to obtain some of the public domain software for the disk library.

Undocumented features of the Macintosh:

COMMAND-SHIFT-1 will eject the disk. By inference, COMMAND-SHIFT-2 is believed to eject the disk in the external disk drive.

The reason file names cannot have a ":" is because "2:" will access the second disk drive.

There is a "Seattle" font, which is 10 point and appears only in some Microsoft programs.

To rebuild a trashed disk directory, insert the disk while pressing COMMAND-OPTION. The machine will go into disk recovery mode to recover files. Alternatively, use the RESET button on the left side of the machine. This is documented in the manual but is not easy to find.

Using the Font Editor, it is possible to save up to 75K of disk space by removing unnecessary fonts.

Creative Logic has announced three versions of FORTH for the Mac. Version 1 will be \$150; Version 2, which will give the user more control over the Mac, is \$250; Version 3 provides a complete tool kit and includes a license to make 250 copies, but it costs \$2,500.

All Sony 3 1/2-inch disk drives Apple gets are currently being used for Macs and Lisas. The extra drive for the Mac will probably not be available until June or later. Davong and Techmar are each reported to be producing hard disks for the Mac. The Mac upgrade to 512K will probably be available in 1985 by exchanging boards out of 128K Macs.

Hewlett-Packard 3 1/2-inch disks are media-compatible with Mac's disks but are more expensive than Apple's and can be used in a pinch.

The meeting closed with a discussion of the advantages of a Mac over other computers. Members reported that women and girls in their households are using the Mac much more than other computers. One Mac owner, new to computers, called his machine "user seductive." Will Mac be a hit because it attracts computerphobics??

## FREE!

**Get your FREE copy  
of the new IDS catalog of  
ergonomically-designed  
Personal Computer furniture.**



PREVEL  
Highest Quality  
at Discounted  
Prices

It's the one "single source" of its kind, with quality selections of every description from more than 20 of the most respected makers in the business. **All at a discount off manufacturer's suggested retail prices!** If you haven't received your copy, call (202) 333-7502. Or write:

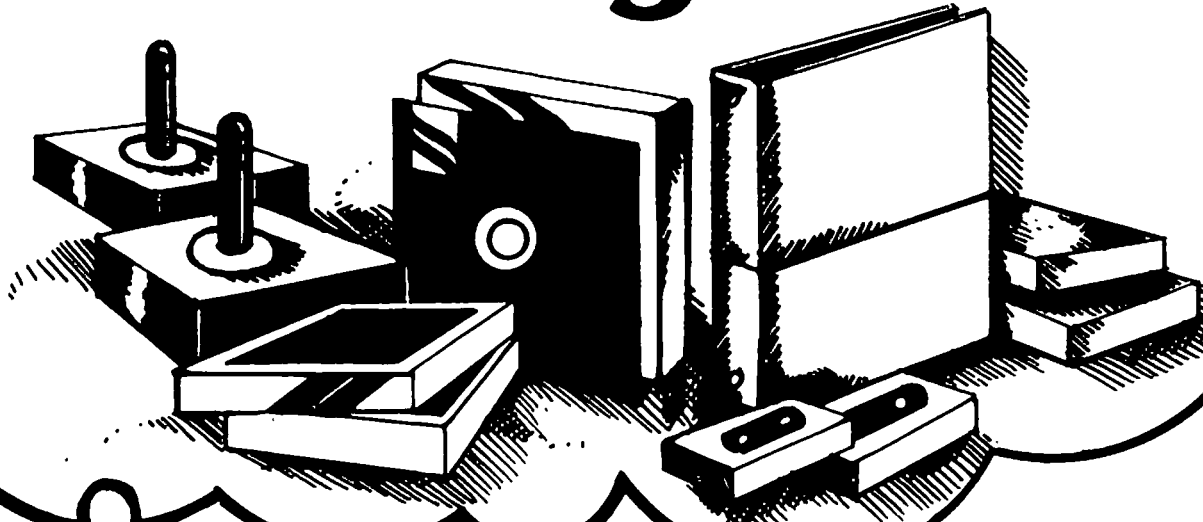


# IDS

**Interior Design Systems**  
3641 St. Mary's Place N.W.  
Washington, DC 20007



***“I felt like a kid  
in a candy store.”***



That's what most people say on their first visit to Software City. Racks of software for business, home, education and recreation. Ready for browsing. Every one discounted. Come See. Come Save!

***Programs, peripherals, disks and accessories, plus a complete selection of books and magazines.***

**SOFTWARE ALWAYS DISCOUNTED**

***Software  
City***<sup>®</sup>

**11621 Nebel Street • Bethesda, MD 20852  
(301) 468-1001**

# MACINTOSH <-> APPLE ][ FILE TRANSFER

by Donald C. Schmitt

What do you do when you want to use an existing spreadsheet or Basic program that you have in Apple ][ format on your new Macintosh? You don't have two modems to transfer the data, but you do have a serial card in the Apple ][. What follows may be just the answer.

I have had an Apple ][ for about 1 1/2 years and have recently purchased a Macintosh to accompany it. I had developed some spreadsheets on the Apple ][ that I would hate to type in again and thought there must be an easier way. With an Apple Super Serial Card in the Apple ][ and a serial port on the Mac, I figured a connection could be made that would provide the answer. I have little experience with serial communications, but I gave it a shot.

The primary reason for all this madness was so that I could use on my Mac some existing spreadsheet models and Macintosh programs I downloaded from Compuserve via my Apple ][. I had previously transferred several models from VisiCalc to Multiplan using the Multiplan "Other" mode under "Transfer-Options". And, after loading these models into Multiplan, I saved them again, but this time as a "Symbolic" file under "Transfer-Options" and onto a separate DOS 3.3 initialized disk.

In order to start this project you will need to accumulate several things; all are easy to obtain and use. The "shopping list" will include:

#### Hardware:

- Super Serial Card or equivalent
- DB 25 male connector (SSC connector)
- DB 9 male connector (Mac connector)
- 7 (or more) conductor cable
- Soldering iron/solder
- Wire stripper/screwdriver

#### Software:

- ASCII Express Pro (or other Apple ][ terminal software for Mac)
- MacTEP terminal software for Mac (A+ Magazine or Compuserve MAUG)
- Blank initialized Mac and Apple ][ disks

The steps involve building a cable configured to the two physically different serial connections (see Fig. 1). Next, you will have to type in, or in some way obtain, a copy of Dennis Brothers MacTEP terminal program for the Macintosh. This free program has appeared in several versions on Compuserve (GO PCS-51 then area XA4), and more recently in the May issue of A+ Magazine on page 94. You will also need an Apple ][ terminal package of some variety. I used ASCII Pro and configured it to my serial card and to send/receive at 1200 baud, 8 bits-no parity-1 stop bit. I turned off most of ASCII Pro's special features and used only those essential for basic communication. Finally, I set the serial card to communications mode (see Fig. 2).

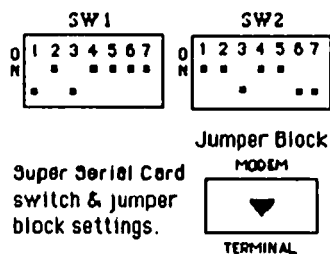
After you have assembled the cable, plugged it into the proper jacks and set the serial card switches as indicated, you are about ready to start communicating. Get your version of MacTEP and set the baud rate to 1200. If you have the A+ Magazine version you will have to change the first four switches of SW1 on the Super Serial Card to ON OFF OFF ON so it will operate at 300 baud, as this is the speed that is preset on

Figure 1

DB 9 Macintosh	Function	DB 25 Apple //
3	Signal Ground	7
9	Receive Data	3
6	Data Terminal Ready(DTR)	20
7	Data Carrier Detect (DCD)	8
8	Shield Ground	1
5	Transmit Data	2
1	Frame Ground	1

Cable conn. between Mac & the Apple //

Figure 2



this version of MacTEP. You will also have to change your Apple ][ terminal program to 300 baud. Now boot up your configured Apple ][ terminal program. If all is connected properly and your software is running properly you should be in communication between the two computers. If you have this link you are ready to upload/download files. If not, check over all the previous steps. This method worked the first time I attempted it and I am generally not that lucky.

If your communication link is established and working properly, you are ready to transfer a file to the Mac. Choose the "Send" option of your terminal program on the Apple ][ and take the steps necessary to prepare it to transfer your Multiplan "Symbolic" or ASCII text file. Choose the CMD-R command of MacTEP and enter a file name of your choice. Push the required key to send the file from your Apple ][ and you should see the file scrolling happily by on the Mac. When the file is finished being sent, again press the CMD-R key on the Mac to finish the transfer. When you boot up Multiplan on the Mac, choose the "File" window and "Open" option. Load the new file into the spreadsheet using the "Sy1k" mode. This procedure has worked everytime for me with only minor cosmetic changes having to be made to the spreadsheet before using the normal Mac Multiplan "Save" method.

The same procedure for transferring Basic programs is used. You should first use a word processor capable of reading ASCII text files to edit your downloaded program from the Apple ][, so that it contains only the program lines. Save these programs under individual file names on a separate disk. You would "Send" these files from the Apple ][ in the same manner used for the Multiplan transfer. MacTEP has a special command for receiving Basic files. Instead of using CMD-R, as you would for standard text, you would sub-

contd.

stitute CMD-8 at the start and finish of the file to properly receive the Basic code. Again, this has worked flawlessly and I have saved hours of typing.

It is also very possible to send files from the Mac to the Apple ][. This would be of interest to someone who developed, for instance, a spreadsheet or text file on the Mac and now needs it to be used on the Apple ][. You would merely reverse the send/receive procedures.

Since I am not an expert in data communications I was quite happy with the results I have obtained. I think it is possible that this project is within the realm of any Apple owner who is beyond the neophyte stage and who is not afraid to "tinker with the mysteries of the machine". I found it quite interesting.

If you have a problem you may drop me a note or call and I will try to help (a SASE would be appreciated). My address is P.O. Box 14, Gettysburg, PA 17325; phone 717-334-3265, 9:30 PM - 11:00 PM. You may also be able to get the aid you require from members of the WAP Telecomm SIG or possibly from a computer dealer who has staff that works with data communications.

Happy Mac'ing .....



## \$\$\$ DISCOUNT PRICES \$\$\$

### Monitors

Amdek Color 1 .....	\$292
Amdek RGB Color II.....	\$425
NEC 1260 (Green) .....	\$120
NEC Color (Composite) .....	\$305

Synetix Disk Emulator 294 K.	\$510
Grappler Printer Card (Specify Printer).....	\$130
Buffered Grappler Plus.....	\$182
Amdek Digital Plotter .....	\$600
Z-80 Card .....	\$130

### Modems

D. C. Hayes Micromodem II e .....	\$245
D. C. Hayes 300 Baud Smartmodem.....	\$220
D. C. Hayes 1200 Baud Smartmodem....	\$500

Softcard Premium System IIe .....	\$350
Format II.....	\$125
Wordstar.....	\$250
Wordstar with Applicard....	\$325
PBL Personal Investor .....	\$110
Saturn Accelerator II .....	CALL
Videx Videoterm.....	\$215
E-Z Ledger (Highlands) .....	\$45
E-Z Invoice (Highlands) .....	\$45

### Printers

Toshiba P1350 .....	\$1550
Transtar 120P.....	\$470
NEC 8023 .....	\$385
NEC 3510 Spinwriter .....	\$1425
IDS Microprism 480.....	\$480
Okidata Micro 82 A .....	\$380
Okidata Micro 83 A .....	\$565
Okidata Micro 92 .....	\$445
Okidata Micro 93 .....	\$700
Prowriter 8510 A .....	\$390
Epson FX80 .....	\$545
Epson FX100 .....	\$710

Stock Option Analysis Program (H & H Scientific).....	\$250
Stock Option Scanner (H & H Scientific).....	\$350
d Base II.....	\$440
Dow Jones Analyzer.....	\$290
Dow Jones Manager.....	\$250
LJK Letter Perfect.....	\$105
Visicalc .....	\$185
Microsoft Multiplan .....	\$200
General Manager .....	\$120
Screenwriter II .....	\$100
Sensible Speller .....	\$100

# ROBINS Inc.



THE SMALL COMPUTER  
SUPPLIES STORE  
SHOP BY PHONE  
WE SHIP BY UPS  
PRICES SLASHED  
ON RIBBONS!

[B=brand C=compatible]

Epson FX/MX/RX 80 [C] .....	3.95
Epson MX/FX 100 [C] .....	9.95
Prowriter/NEC 8023 [C] .....	3.95
Diablo Hytype 2 Film [C] .....	3.50
Diablo Hytype 2 Film [B] .....	4.25
QUME Film [C] .....	1.90
QUME Film [B] .....	2.80
QUME IV Film [C] .....	3.45
NEC 5500/7700 Film [C] .....	3.30
NEC 3500/2000 Film [B] .....	11.90

#### RELOAD YOUR OWN!

Diablo, QUME, NEC 7700 Film .....	1.50
Epson 80, NEC 8023, Itoh .....	1.95
Epson 100.....	3.95

#### PAPER

Clean Edge 18 & 20 ...	\$26.95
Labels 3-1/2x15/16 5k .....	16.00

Hours: Monday thru Friday 9-5, Saturday 10-2

6 Outlet Surge and Spike  
Suppressor..... \$4.95

#### PRINT WHEELS

Diablo Plastic .....	\$6.40
Diablo Metal .....	\$41.50/49.95
Diablo 620 .....	\$11.40
QUME .....	\$6.40/7.90
Smith-Corona TP-1 .....	\$4.90
Olivetti .....	\$15.50
Olympia EC .....	\$26.50
Brother .....	\$24.95
Daisywheel II .....	\$27.50

Head Cleaning Kit .....

#### FLOPPY DISKS

5-1/4 ss/dd Sentinel .....	\$22.50
Kybe, 3M, Verbatim .....	\$26.50
Brown ds/dd .....	\$29.95
Brown ds/(96 tpi) .....	\$39.95

**560-5900**

8304 Hilltop Road  
Fairfax, Virginia 22031  
(Merrifield Area)



VISA/MC (Add 3%), money order, certified check.  
Prices subject to change. Shipping/handling \$5.  
MD residents add 5% tax. Mail order only.

CALL FOR ITEMS NOT LISTED  
GOV'T PURCHASE ORDERS ACCEPTED

## RAMADA COMPUTER PRODUCTS

A Division of H & H Scientific

13507 Pendleton St.  
Ft. Washington, MD 20744  
Tel. (301) 292-2958

# MACINTOSH BASIC SPEED COMPARISONS

by Richard D. Norling

By the middle of this summer, Macintosh owners will be able to choose between two different versions of the BASIC language. One, Microsoft's MS-BASIC for Macintosh, is available now at a retail price of \$150. The other, called Macintosh BASIC (or MacBASIC for short), is being prepared by Apple Computer for release sometime this summer at a rumored price of \$99.

The Microsoft product is pretty much a standard MS-BASIC, translated to operate in windows on the Macintosh. It contains the usual MS-BASIC EDIT (line edit) and RENUM (renumber) commands. But it does not evidence, in my opinion, any use of imagination to redesign a product so that it would use the full capabilities of the Macintosh and its user interface.

From all reports, Apple's MacBASIC will include as part of the standard package a full-screen Macintosh editor (the equivalent of GPLE and then some), and a dynamic debugger that displays variable values and breakpoints. In addition, MacBASIC will have the ability to run several programs at once, and will have an expanded command set that includes many control structures similar to those in Pascal.

The April issue of Byte Magazine contains an article by Scot Kamins that describes Apple's Macintosh BASIC. Towards the end of the article, he presents 6 short benchmark programs and the times it takes Macintosh BASIC to execute them. The times presented in the article didn't mean anything to me without having something to which to compare them. So I decided to try the 6 programs with Microsoft's MS-BASIC on the Macintosh.

The results of the speed comparison between the two versions of BASIC on the Macintosh are dramatic, as you can see. Microsoft's version varied from 2 to 4 times slower than Apple's version, depending on the particular test involved. The times for Apple's version are from the Byte article. The Eratosthenes Sieve program did not fit within the work space normally allowed by Microsoft BASIC until I typed the command CLEAR,25000 to take memory away from the Macintosh desk accessories. Listings of the test programs used are on page 328 of the April, 1984 Byte. Each test involves a loop with 5000 iterations except for Eratosthenes Sieve, which was dimensioned to find 1899 prime numbers.

By now I had become involved in the comparison process, so I decided to add a comparison with Applesoft BASIC running on an Apple ][+. The last column of the table shows those results. The programs were used with the least amount of change necessary to allow them to run in Applesoft. I did not run test number 5 in Applesoft because Applesoft does not have a separate "integer divide" command.

Observant readers will notice that humble ol' Applesoft was faster than Microsoft's MS-BASIC on test number 4, long divisions. One possible explanation for the slowness of the MS-BASIC divisions could be that it is keeping track of a greater number of significant digits than Applesoft. But the time estimate of 31 seconds shown for MS-BASIC is for single precision (4 bytes) data, and using double precision (8 bytes) adds only 1 second to the MS-BASIC time. Applesoft is handling 5 bytes of data. I can only surmise that the division procedures used by Applesoft are much more efficient than those used by Microsoft

in MS-BASIC.

Besides noting, as we have, that Apple's MacBASIC will run much faster on the Macintosh than the Microsoft BASIC now available, it is also interesting to compare the speed of Apple's MacBASIC with Applesoft. By looking at the results of test 6, we see that at least some BASIC programs will be able to run ten times faster on the Mac than on the Apple ][+. Does this give anyone another reason to get a Macintosh?

## BASIC SPEED COMPARISONS IN SECONDS

	Macintosh		II+
	Apple MacBASIC	Microsoft MS-BASIC	Applesoft
1. Empty GOSUBs	3.0	8	15.5
2. Empty FOR/NEXT loop	1.5	5	7
3. Midstring function	9.0	21	33
4. Real divides	18.0	31	30
5. Integer divides	3.0	13	--
6. Eratosthenes Sieve	31.5	121	327

Ⓔ

Why are you using diskettes with black jackets?  
Because you have no choice?  
Now you do.

SUNSTAR offers you high quality diskettes with the advantage of color. And they are guaranteed. Forever.

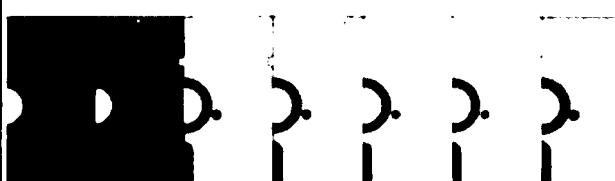
**SUNSTAR**  
Premium Diskettes in Color

And as an added bonus—at no extra cost—SUNSTAR diskettes are packaged in a handsome and durable polyvinyl library file.

Choose your colors: red, burgundy, yellow, green, dark blue, light blue, orange, gray, dark brown, light brown.

---

Order now using VISA or MasterCard.  
Call our 24-hour, toll free number: (800) 227-3800 Ext. 61.



# Operant Systems

## -- HARDWARE --

<b>DISK DRIVES--</b>	
MicroSci A2 drive (100% Apple-compatible Shugart 390)...	235
Rana Systems Elite One drive (40 track, 163K).....	259
Elite Two (40 track, double side, 326K)....	369
Elite Three (80 track, double side, 652K)....	449
TEAC Thinline drive (40 track, 100% Apple compatible)...	259
Davong, Corvus, and Corona Winchester drives.....(call)	
<b>CP/M &amp; 6502C SYSTEMS--</b>	
Applicard (6 Mhz Z-80, 64K to 192K RAM, 70-col video)...	249
Microsoft Softcard J/c (Z-80, 80 col & 64K on one card)...	325
Softcard (includes CP/M 2.2 and MBASIC).....	225
DR Gold Card (6 Mhz, 80-col, 64-192K, CP/M 3.0, CBASIC)...	345
ALS CP/M Plus System (6 Mhz Z-80, 64K RAM, CP/M 3.0)...	275
Titan Systems Accelerator J/C (3.6 Mhz 6502C processor)...	425
Speed Decon (6502C high-speed coprocessor).....	239
<b>MONITORS--</b>	
Adek 300G (12" green anti-glare screen, 18Mhz).....	145
300A (12" amber anti-glare screen, 18Mhz).....	155
Color I (40-column text/color graphics).....	289
NEC JB-1201/1205 (green/amber 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
<b>MODEMS--</b>	
Hayes Micromodem J/c (tone dialing/speaker/Smartcom I)...	239
Smartmodem 900 (300 baud, RS-232, direct-connect)...	210
Smartmodem 1200 (300/1200 baud, as above).....	495
Novation Apple-Cat II (w/ Comware; 1200 baud capable)...	235
Smart-Cat 300 (RS-232, direct-connect).....	175
Smart-Cat 300/1200 (as above).....	369
J-Cat 300 (miniature RS-232, direct-connect)....	105
55M ModemCard (300 baud, audio monitor, emulates Hayes)...	235
US Robotics Password (300/1200 baud, auto-dial/answer)...	355
Anchor Automation Mark 12 (300/1200 baud, RS-232).....	269
Volkmodem (300 baud, RS-232).....	59
<b>PRINTERS--</b>	
Epson FX-80 (160 cps, tractor/single sheet, graphics)...	489
FX-100 (wide carriage version of the above).....	679
RX-80F/T (100 cps tractor/single sheet, graphics)...	379
RX-100 (wide carriage version of the above).....	539
Okidata 92 (160 cps, graphics, BEST print for price)...	439
93 (132 column version of above).....	715
Toshiba 1340 (144 cps draft, BEST letter-quality matrix)...	799
ID5 Prism-80 (200 cps, buffer, correspondence quality)...	999
Prism-132 (wide-carriage version of above).....	1149
Gemini 10X (120 cps, tractor/single sheet, graphics)...	289
15X (wide-carriage, same features as Epson RX)....	410
Itoh F-10 Starwriter (40 cps daisywheel, Qume compat)...	1099
A-10 (18 cps version of the above).....	559
Transtar 130 (18 cps daisywheel, Diablo compatible)....	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
<b>INTERFACES &amp; BUFFERS--</b>	
Phaso/U printer interface (superior graphics & more!!)...	129
ShuffleBuffer (32K—128K w/cut & paste/serial and par)...	235
Crappler+ printer interface (parallel w/ graphics).....	120
Buffered Crappler+ (16K to 64K buffer plus graphics)...	169
CPS Multifunction (serial, parallel, and clock).....	165
Wizard BPO (16K-32K buffered parallel w/ graphics).....	135
IPI (graphics, text dup, page formatting).....	69
CCS 7710A (RS-232 serial for printers & modems).....	109
<b>VIDEO &amp; KEYBOARD--</b>	
Videx Ultratera (160 column/48 row display !!).....	259
Videtersa (80-column w/ softswitch & inverse)....	209
Wespercorp Wizard-80 (softswitch & inverse built-in)...	155
ALS Seartera II (80 columns at a bargain price).....	129
MicroSci 80-col card w/64K RAM for J/c.....	125
Keytronic KB-200 low-profile detached keyboard.....	235
<b>MEMORY EXPANSION--</b>	
Wespercorp Wizard 16K RAM card (2-yr warranty).....	69
Titan Systems 128K RAM card.....	359

## -- SOFTWARE --

<b>WORD PROCESSING--</b>	
Wordstar 9.3 (including 6 Mhz Z-80 Applicard !!!)...	299
ScreenWriter II (70 col display, spooling, more!!)...	85
Super-Text Professional.....	69
Bank Street Writer or Speller.....	49
pfs: Write J/c.....	85
Word Handler + List Handler + Spell Handler.....	109
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)...	119
Sensible Speller IV (checks ANY file type).....	89
The Word Plus (super spelling checker for CP/M)....	109
<b>SPREADSHEETS &amp; FORTUNE-TELLERS--</b>	
Multiplan (state-of-the-art spreadsheet).....	135
VisiCalc (J/C+ and J/c versions available).....	165
SuperCalc 2 (powerful CP/M spreadsheet).....	179
<b>INFORMATION MANAGEMENT--</b>	
dBASE II (w/ ZIP screen generator).....	409
Quickcode (dBASE program generator).....	185
The General Manager 2.0.....	149
Thinktank (electronic thought organizer).....	99
pfs: File, Report, or Graph.....	85
Infostar (including 6 Mhz Z-80 Applicard !!!).....	299
Data Perfect.....	99
DB Master 4.0 (latest version).....	229
<b>BUSINESS &amp; ACCOUNTING--</b>	
BPI Accounting (AR/AP/PAYROLL/INVENTORY) ea module:	249
Peachtree Peachpat (GL/AR/AP)..... all three:	229
Dollars & Sense (home accounting + graphics).....	69
The Accountant (double-entry bookkeeping system)...	99
Home Accountant.....	55
<b>COMMUNICATIONS--</b>	
Ascii Express Professional (for DOS 3.3).....	89
Z-term Professional (for CP/M).....	99
Data-Capture J/c.....	65
CompuServe Starter Kit (password/user guide).....	29
<b>LANGUAGES--</b>	
Einstein Applesoft Compiler.....	85
Softtech UCSD p-System.....	425
Digital Research Pascal/MT+.....	249
Microsoft Fortran-80.....	129
BOS C compiler.....	115
Microation FORTH-79.....	79
Terrapin Logo.....	99
<b>COOS &amp; ENDS--</b>	
Kensington System Saver Fan (w/surge protector)....	66
Kraft Joystick / TG Joystick.....	85/40
Trace SoundTrap (silence your printer).....	99
Hockingboard (sound synthesis w/speech option).....(call)	
Wildcard Plus (hardware-based copy unprotection)...	129
Locksmith 5.0 (...the best gets better!).....	85
KoalaPad Graphics Tablet (with graphics software)...	89
Maxell, Dysan, TDK, Memorex, Wabash disks.....	from 20

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 OILLON 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.

# THE SPEED OF MICROSOFT BASIC ON THE MAC

by David Morganstein and Dave Weikert

Macintosh owners and prospective buyers are considering what they can do with Apple's recent arrival. Besides the Paint and Write programs provided, there are only a few applications packages available. Since the Mac does not come with any built-in language, a disappointment to us, the first piece of software many will consider is a language to allow you to program it. The only currently available choice is Microsoft Basic, the well-known and widely-used standard.

In this brief note, we provide some speed comparisons, since our first experiences with the Mac version made us realize that, while faster than other BASIC alternatives, it was not as fast as we think it should be. Last month, a note indicated that Microsoft's Multiplan appeared to be nine times faster on the Mac than in Apple CP/M. Their BASIC appears to be little more than twice as fast as the CP/M version running at the standard 2MHz clock speed. This indicates that a faster Z80 card using Microsoft BASIC, running at 6MHz would complete tasks more quickly than the Macintosh version. We will be in touch with Microsoft to inquire why this is so as it seems surprising.

In the table below you will find the results (in seconds) of five tests. The first two compare the speeds for writing and reading numbers. Sequential files were used for the test. The next comparison is of a bubble sort of numbers which had been randomly generated. The fourth test was a simple empty loop. The last test was to add the first "k" digits. All versions tested got the same, correct, answers! The first three tests, which depended on random numbers, were repeated several times to insure that the accuracy of the results did not depend upon the numbers generated.

For comparison, we ran the tests against: the CP/M version running on a standard Z80 card, Applesoft, and

a Corona (IBM clone) with both floppy and hard disks. While the results indicate the Mac implementation of Microsoft BASIC is generally faster than the others, it is not as fast as we had hoped.

On another dimension, be aware that the Microsoft BASIC initially is configured for a rather small 14K workspace. You can have as much as 30K of workspace memory by giving up some of the Mac's desk-top features.

Before deciding on a language, you might want to examine MacBasic, now rumored to be scheduled for release in July. A thorough review is found in the April issue of BYTE.

	MAC	CP/M 2MHZ	A'SOFT	CORONA FLOPPY	HARD DISK
1. WRITE TO FILE					
100 NUMBERS	6	6	7	4	4
250 "	8	9	10	7	6
500 "	11	17	14	11	10
2. READ FROM FILE					
100 NUMBERS	1	4	7	3	2
250 NUMBERS	3	13	14	7	6
500 "	5	25	26	12	11
3. BUBBLE SORT					
50 NUMBERS	12	27	20	16	
75 "	25	60	47	38	
100 "	44	110	79	66	
4. LOOP OF 5000	5	12	7	6	
5. ADD FIRST K DIGITS					
1000	3	6.5	4	4	
5000	13	31.5	19	18	⊕

## COMMUNITY COMPUTER FAIR

There will be a Community Computer Fair on Saturday, May 12, 1984 at the Mount Rainier Elementary School, 4011 32nd Street, Mt. Rainier, Prince Georges County, MD. The hours are from 10:00 AM to 3:00 PM, and the admission is \$1.25.

Anyone interested in volunteering to demonstrate their educational software at this fair should contact Conrad Fleck at 699-8200 (day) or 445-2315 (evening). ⊕

## APOLOGY

In the March 1983 issue of the Washington Apple Pi Journal we erroneously listed the author of "Relabeling the Apple Keyboard" as Barry Slotnick. The correct name of the author is Barry Skulnick. Our apologies for this. We try very hard to insure the accuracy of author name and spelling, but somehow this escaped us.

## INTERNATIONAL SIG

The suggestion has been made to establish an International SIG within Washington Apple Pi that would coordinate the exchange of information with similar groups in other countries, particularly those in which computer use is not widespread. Such a group would work principally through nationals of the countries concerned who live in, work in, or frequently visit Washington, DC, and who are interested in furthering the use in their countries of microcomputers produced by any manufacturer.

Means of exchanging information could include operation of international computer bulletin board systems and personal contacts through people traveling to other countries. An organizational meeting will be set up in the near future. Interested persons should contact Gabriel Roth at (202) 676-8060 or Bernie Urban at the WAP office (301) 654-8060. ⊕

## CLINTON COMPUTER LEARNING CENTER

### Classes scheduled for April through June

**APPLE //e INTRODUCTORY COURSE.** 4 hrs. \$100. Week night classes 6-8pm and alternate Saturdays. 9am-1pm. Learn the basic operating procedures of the world's most popular small computer! Course includes three program disks.

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

**LOGO FOR TEACHERS.** 10 hrs. \$100. May 2, 9, 16, 23 & 30. 6-8pm. Workshop for teachers with no experience with this programming language for children. Beginning concepts, and practical methods for teaching LOGO in classroom.

**APPLEWRITER IIe.** 2 1/2 hrs. \$46. April 24, May 29 or June 26. 6-8:30pm. 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 hrs. \$65. May 7, 14, 21. 6pm-8pm. Hands-on instruction for computer novices in this simple programming language. Classroom exercises and optional homework assignments.

**APPLESOFT BASIC PROGRAMMING II.** 6 hrs. \$65. June 4, 11, 18. 6pm-8pm. Learn new Basic commands and additional programming techniques. Classroom exercises and homework assignments. Prerequisite: Basic Programming I or equivalent.

**INTRODUCTION TO WORDSTAR.** 2 hrs. \$40. April 4 or June 6. 6pm-8pm. Learn how to create documents quickly and easily by using this popular word processing system.

**ADVANCED WORDSTAR.** 4 hrs. \$80. April 5 & 12 or June 13 & 20. 6pm-8pm. Experiment with more advanced Wordstar commands, and learn file merging techniques. Prerequisite: Wordstar I or equivalent.

**MICROSOFT BASIC PROGRAMMING I.** 6 hrs. \$65. May 17, 24, 31. 6pm-8pm. Hands-on instruction in basic programming for CP/M-based computer users. Classroom exercises and optional homework assignments.

**MICROSOFT BASIC PROGRAMMING II.** 6 hrs. \$65. June 7, 14, 21. 6pm-8pm. Learn additional programming techniques in CP/M Microsoft Basic. Classroom exercises and homework assignments. Prerequisite: Microsoft Basic Programming I or equivalent.

**COMPUTER CAMP FOR KIDS.** 10 hrs. \$110. June 2, 9, 16, 23, 30. 2-4pm. Optional Lab with instructor, 4-5pm. Introduce your child to the world of small computers! Hands-on programming and instruction in practical applications.

**INTRODUCTION TO SPREADSHEETS.** 2 hrs. \$40. April 26 or June 28. 6-8pm. Create useful financial models for your business using two popular spreadsheet programs: VisiCalc and Multiplan.

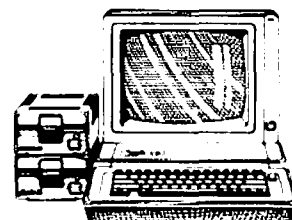
**OVERVIEW OF THREE SOFTWARE SOLUTIONS.** 2 hrs. \$30. April 2, 9, 23, May 8, 15, 22, June 5, 12, 19. 6pm-8pm. This two hour overview course is for computer owners who would like to see what the capabilities of their computer are. This hands-on class demonstrates three major applications: Word Processing, Spreadsheets and Databases.

The Learning Center also provides custom-designed instruction on-site or in our classroom. Course discounts available with system purchases. For details and inquires, contact us at either one of our two locations: Clinton, Maryland; 301/856-2500 or Alexandria, Virginia; 703/838-9718.



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

(301) 856-2500



# MODULA 2: INPUT/OUTPUT

by Robert C. Platt

Our fifth chapter in the Modula tutorial will focus on Input and Output, the vital part of the language which was "left out" by its creator, Niklaus Wirth.

## MODULA IMPLEMENTATIONS

But first some preliminary matters. This tutorial covers the Volition System implementation of Modula for the Apple II and III. (Volition plans to market Modula for the Macintosh this fall.) Six other Modula compilers are on the market, although most do not run on the Apple. The most promising is The Modula Research Institute's compiler for the IBM PC which sells for only \$40. Rumors abound that MRI is currently planning a comparably-priced Apple version.

The fact that seven different vendors are currently selling Modula systems makes a presentation of input/output difficult, as these folks have yet to agree on a standard. So while they continue to negotiate the "official" Modula input/output library, I'll stick with presenting Volition's approach, which is very faithful to Wirth's book PROGRAMMING IN MODULA-2.

## FOR FURTHER INFORMATION

If this series has succeeded in wheting your interest in Modula, consider several sources of further details. First, WAP's Pascal Interest Group (PIG) will answer your Modula questions at its monthly meetings. Also, the PIG is preparing its first Modula disk for the club library, which includes the flash card drill program from last month. Next, look at a copy of the Journal of Pascal, Ada & Modula-2, which recently changed its name and expanded its coverage of the language. Finally, consider buying a book or two on Modula. At present, the only book on the market is "Programming in Modula-2" by Niklaus Wirth (1982 Springer-Verlag pp. 176). This book is the official bible of Modula, but it suffers from the fact that English is not Wirth's native tongue. (Also, a revised third edition is at the printers, so you might want to wait for it.)

I predict that by fall, the bookstores will be flooded with new Modula books. Even Jerry Pournelle has threatened to write one! Of course, if you prefer the straightforward, down-to-earth style of this tutorial over Jerry's, that option will be available to you as well.

## MODULA TEXT FILES

As with Pascal and Basic, Modula supports text files. Remember, a text file is a series of characters divided up into lines by a carriage return (CR) character. The advantage of using text files is that they can be created or modified by text editor/word processor programs and can be routed to the printer for printing.

In the flash card drill program presented last month, a text editor is used to type in the questions, expected answer, and computer responses. Somehow the Modula program must find and read in the text file. Here's how it's done.

Remember that the file is stored on a floppy disk. When the file was stored, it was given a name, which we call the "file name." That file is like an airplane with an assigned flight number. But if you have ever tried to catch a flight at the airport, you know

that both the flight number and gate number are needed to find an airplane. In our analogy, the "gate" where the airplane parks to unload is called a "text stream." Sometimes the text stream is hooked up with one file, other times the stream has a different file. By giving flights and gates different sets of names, the airport has the flexibility to avoid building a separate gate for every different flight. Well, the same advantage comes from naming files and text streams separately.

## BRINGING IN THE AIRPORT

All the routines to bring our text file in for a landing are stored in a library module called "Texts". Lines 11-13 of the flash card drill program call these routines into our program. By using the FROM...IMPORT command, we avoid having to qualify each reference to these routines with the module name. We can write "WriteLn" rather than having to write "Texts.WriteLn."

The drill program assumes that the student knows the file name of his currently assigned drill. Lines 50-54 read in that name and opens the file. (Our plane has landed!) You may want to revise this section of the program to give the student a menu of available drills.

Line 55 then attempts to connect the file to our text stream named "t." Connect is a function in the Texts module which returns a value to tell us whether the file has been successfully connected to the text stream. If the value equals "TextOK" then the connection was made successfully. (Our plan is hooked to the gate.)

All following reference to the information in the file will be to the text stream "t" not to our file "f". For example, line 69 reads the next line from the text stream (until the next CR) and stores it in a string named "line."

```
ReadLn(t,line)
```

To check if we have read up to the end of the text stream, we call a procedure that returns a Boolean value called EOT (for end of text). If EOT(t) is true, then the drill is over and our program is done.

## CLEANING UP

At the end, line 118 tests for EOT again and EXITs the program. Modula automatically closes all files at the end of the program. But if we wanted to be perfectly proper about things, we could have called "Disconnect" to decouple our text stream from our file and then closed our file.

I should note that module Texts is not the only way to read text files. Another module, InOut, will achieve the same results without using the intermediary device of text streams. In fact, the Texts module routines call the InOut routines for their actual input/output operations.

(Ed. Note: What happens if you're at the wrong airport?)



# HARDWARE HELPERS

If you are having hardware problems with your Apple and/or peripheral equipment, the following persons have agreed to help. 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.

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

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

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

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

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

Pete Jones  
1121 N. Arlington Blvd.  
N. Arlington, VA 22209  
(703) 430-1606 (7-10)

Bob Kosciesza  
2301 Douglas Court  
Silver Spring, MD 20902  
(301) 933-1896 (till 10)

Mark Pankin  
1018 North Cleveland St.  
Arlington, VA 22201  
(703) 524-0937 (till 10)

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

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

Dave Weikert  
17700 Mill Creek Drive  
Derwood, MD 20855  
(301) 926-4461 (7-10 except  
(Thurs. and weekends)

APOLLO ELECTRONICS, INC.  
11525 COLT TERRACE  
SILVER SPRING, MD. 20902

\*\*\*\*\* BEST DISCOUNTERS ON \*\*\*\*\*  
HARDWARE, SOFTWARE & ACCESSORIES

#### HARDWARE:

MICRO-SCI A2 DISK DRIVE W/O CONT.	\$235
MICRO-SCI A2 DISK DRIVE WITH CONT.	\$310
NEC 8023A LINE PRINTER	\$379
OKIDATA ML-92 LINE PRINTER	\$430
TRANSTAR 120P L/Q PRINTER	\$435

#### SOFTWARE:

BANK STREET WRITER	\$ 45
FROGGER	\$ 15
LIST HANDLER	\$ 30
SNACK ATTACK	\$ 19
WORD HANDLER	\$ 35

#### ACCESSORIES:

BASF 5 1/4" DISKETTE S/S D/D BOX 10	\$ 23
FLIP-N-FILE	\$ 18
KRAFT JOYSTICK	\$ 37

#### TO ORDER:

\*\*\*\*\* PLEASE CALL \*\*\*\*\*  
649-7446 M-F 9 AM - 5 PM  
933-1849 AFTER 6 & WEEKENDS

## MINUTE MANUALS

Don't waste your valuable time trying to get your software to do what it is supposed to be able to do. Who can you turn to in your hour of need - the answer is probably in the Minute Manual.

**Minute Manual For Apple Writer IIe** - A lot of support for \$7.95. Beginners guide to word processing, step by step instructions for the basic and advanced procedures. Over 60 pages of problem solving procedures on formatting and printing. Specific procedures for Epson, Gemini, Apple, Prowriter, NEC, Okidata.

**Glossary Disk For Apple Writer IIe and II+** - Easy to use - Automatically load glossary from menu. Select Apple Writer IIe or II+. Select Epson FX/MX/RX, Gemini 10/10X, Prowriter, Apple DMP/Imagewriter, NEC 8023A, or Okidata 92/83A. Then turn on or off any print style with a single keystroke. \$14.95.

**Minute Manual For Apple Writer II+ \$7.95**  
**Minute Manual For DB Master \$12.95**

Available at COMM CENTER, B. Dalton, Sidney Kramers (H St.), VF Assoc., Logical Choice, Towson Computer, Computers Unlimited, Westminster Computers & more, or send check + \$1 S/H to Minuteware, P.O. Box 2392, Columbia, MD 21045 (301) 995-1166.



## Paragon Technologies, Inc.

offers classes in our IBM PC laboratory  
and Apple IIe laboratory in McLean:

- Getting Started With The Micro Computer
- VISICALC, LOTUS 1, 2, 3
- d Base II
- WordStar
- Programming in BASIC
- Computer assisted SAT preparation

All classes are hands-on with one person per computer.

Call  
556-9659

or write to us:  
P.O. Box 6128  
McLean, Virginia 22106

**WHEN YOU'RE  
LOOKING FOR**



**LOW PRICES, IN-STOCK MERCHANDISE, EXPERTISE**



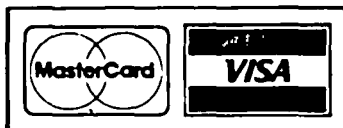
**CALL**

**OUR FULL-TIME STAFF**

**at**

**(202) 364-0273**

- VISA, MC add 3% (include card no. and expiration date)
- Allow 3 weeks for personal check to clear.
- COD's require 10% deposit—all COD charges to be paid by customer.
- Prices subject to change without notice—all items subject to availability.
- Money order, certified check, cashier's check or bank wire deposit accepted.
- Include full telephone number on all mail orders
- Include \$2.00 for postage and handling per software shipment (UPS).
- DC residents add 6% tax (if tax exempt, include number).



Description	List \$	Sale \$
<b>NEW GAMES</b>		
Tunnel Terror	29.95	21.95
Forel	29.95	21.95
Curse of Ra	19.95	14.95
Danger	19.95	14.95
Serpentine	34.95	26.95
Choplifter	34.95	26.95
Pig Pen	29.95	21.95
Death Race 82	29.95	21.95
Deadly Secrets	34.95	26.95
Hardhat Noah	29.95	21.95
Sea Fox	29.95	21.95
Cross Country Rallye	29.95	21.95
Space Cadet	34.95	26.95
Crazy Mazey	29.95	21.95
Mars Cars	29.95	21.95
Freefall	29.95	21.95
Ming's Challenge	34.95	26.95
Ultima II	59.95	44.95

**New Gebell**

Phaser Fire	29.95	21.95
High Orbit	29.95	21.95

**New SSI**

Cytron Master	39.95	29.95
Galactic Gladiators	39.95	29.95
Guadalcanal Campaign	59.95	49.95

**Old Favorites**

Borg	29.95	21.95
Cannonball Blitz	34.95	26.95
Deadline	49.95	39.95
Marauder	34.95	26.95
Ruski Duck	34.95	26.95
Snack Attack	29.95	21.95
Threshold	39.95	29.95

**UTILITIES**

Southwester Data		
Munch-a-Bug	49.95	39.95
The Routine Machine	64.95	49.95
Doubletime printer	149.95	119.95
On-Line Systems		
The Artist	79.95	59.95
Insoft		
GraForth II	75.00	59.95

**CALL  
for information  
about our**

**MONTHLY SPECIALS**

Description	List \$	Sale \$
<b>HARDWARE</b>		
<b>Disk Drives</b>		
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
5 1/4" floppy drive	335.00	249.95
Controller Card	65.00	49.95
<b>Monitors</b>		
BMC 12A: 15MHz	99.95	79.95
AMDEK		
Video 300G	199.00	149.95
Video 300A	210.00	149.95
<b>Dot Matrix Printers</b>		
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
<b>Daisy Wheel Printers</b>		
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	895.00	659.95
**Best Buy**		
<b>Surge Protection</b>		
Kensington		
System Saver	89.95	69.95
<b>Also available</b>		
<b>Modems</b> <b>Interfaces</b> <b>Disk drives</b> <b>Accessories</b> <b>Terminals</b> <b>Peripherals</b> <b>Joysticks</b> <b>Paddles</b> <b>Keypads</b>		
and, hundreds more!		

Description	List \$	Sale \$
<b>BUSINESS</b>		
<b>Data Managers</b>		
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
Versalform II	389.00	249.95
Visifile	250.00	169.95
<b>Accounting</b>		
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
<b>Financial Modeling &amp; Planning</b>		
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!		
<b>New Specials</b>		
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		
Gripper 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.**

code (again, in the Accumulator, at least initially) is a candidate for a program line number. It has got to pass the critical tests in lines #74-79 also as I've pointed out earlier.

The label "LNNUM" at line #83 begins testing for a number because what if there are only three digits in the program line number? The fourth character input would still be  $\leq 5$  away from the (possibly) indented left margin, but this time it's a blank! So, we must still test for the digits 0 thru 9. If a number, then we save it in the array "LNNUM" for printing after all those blanks and underlines I addressed earlier. This saving procedure begins on line #90. I store the line number's length twice because I change "LENNUM" later on; however, "LONNUM" remains fixed throughout the printing of the entire program line.

You'll notice in the module labelled "STORENUM" that I re-initialized "WORDCHAR". That makes sense because all the while the program line number has been rolling in, this variable has been incremented; so I've got to reset it to 0 to begin a new word after I'm done with the line #. What about "PREBLK"? I've re-initialized it also, but what is it? For now, I'll keep you in suspense. At the end of "STORENUM" I output the non-printing CTRL-A because I don't want to print the line number until later. This is the first of many times I alter the content of the Accumulator - so now you see why I don't restore the A-register when I'm "DONE".

The next test labelled "QUOT" simply tests for quotation marks and, if found, increments the counter "NUMQTE". The second quote negates the presence of the first as explained in the comment portion of the listing. Looking for quotes is important because, as I said earlier, I do not test for keywords when between quotes. Finally, I JUMP to "STOREWD" wherein I begin storing the characters of a soon-to-be complete word. See line #211.

The next test looks for blanks. Remember that a blank is one of my word delimiters except for the first blank that follows either a "HIMEM" or a "LOMEM" (see lines #144-148). Toward the beginning of the module, "EMPTY", I set up two tests to ignore the blanks that immediately follow your program line number. The first test quantifies if "WORDCHAR" = 1 which it must be if this blank follows your line number. The second test is to see if "LINECHAR" is within "LONNUM + SUB" units of the (possibly) indented left margin. While this line number was being stored, "SUB" was set equal to 3 (see line #102). If the line number is 3 digits long, for example, then "LINECHAR" must be  $\leq 5$  units from the left margin to accommodate up to 2 leading blanks. "SUB" is used as a parameter because it is reset to 4 later (line #270) to accommodate the possibility of a multiple instruction line. Look closely at one of your printed listings wherein colons are used to separate instructions. See the blank that follows some of the colons. This means that effectively there are sometimes a total of 3 blanks following the program line number, that is, when I put the separate instructions on different lines. Lines #137, 138 insure that the first non-blank character of each separate instruction begins with the same count. If the blank I've found isn't trailing your line number, then I store this blank as the last character of the word in the sub-module "COMPLETE" as well as set the word's length (line #153). Notice here, and anywhere I just store characters in "WORD", that I decrement "LINECHAR". This because "LINECHAR" is the # of characters printed on a line. You'll see that I do not print out "WORD" until I hit one of those three delimiters I addressed earlier. I finally print out the stored complete word via the coupled subroutines "LINE" and "OUTWD" to be described later.

I promised earlier I would address the variable "PREBLK". Well, every time I detect a leading blank after the program line number I naturally increment "PREBLK" (line #139). But, why??? The answer has to do with that string of alternating blanks & underlines addressed earlier, in particular, their numeric count. When I get to printing out that string, I will totally eliminate your suspense.

The next test looks for a colon. Remember, the colon can't be in a REMark or between quotation marks, so I look for these conditions first. I've gotten into the habit of using a colon to separate logic modules in my Applesoft programs instead of a REMark followed by nothing. When I entered this colon, I typed " :<CR>". So, if I universally insert another <CR> after this colon as well, thinking that it separates multiple instructions, then I will generate a cumulative triple space here - the first <CR> because of the colon and two more because of the <CR> inside " :<CR>". The latter <CR> signals the end of the line, so later on I will double space on top of the first <CR> - making a total of three. Whoops !!! So, line #166 begins the test to determine if I have this unique use of a colon, namely, one standing by its lonesome.

Still on the subject of colons, I don't want a <CR> inserted following "HIMEM:" or "LOMEM:", so I guard against that possibility beginning with line #178. "Betwixt and between" I squeeze in storing the colon at the end of the now-completed word, assuming of course that the colon passes all the aforementioned tests, such as not in "HIMEM:". I set the word length and execute "LINE" and "OUTWD" in succession just as I did in the previously mentioned case of a blank.

One last thing on colons. I've used line #165 to count colons, or at least to see if this count is  $\geq 1$ . Why? Because I rely on a colon to tell me when the next instruction is coming so I can quickly go to the next line. Well, what if I have " . . . ABC: A1 = B"? For sure the "1" is not a line # even though its  $\leq 5$  spaces from the left margin. Go back to lines #78,79 to refresh your memory on this issue!

So far I've been talking alot about <CR>s. Guess what my next test looks for? I add the extra <CR> for double spacing between complete program lines as previously discussed. I also reset the colon counter ("NUMCLN") as well as the "IF--THEN" flag, "DENT3". Remember, I indent one more unit (five blanks) upon detection of a "THEN" that follows its partner, "IF". Therefore, when a <CR> eventually appears following the "THEN", both the flag "DENT3" and the # of indent units "DENT" must be decremented to get us back to the left margin existing prior to the appearance of "IF".

Lines #211-216 store each character as the word is being built. Next, the paired subroutines, "LINE" and "OUTWD", follow. Jointly, they are responsible for transcribing the completed words onto paper.

"LINE" first tests the amount of remaining line space. If ample, a return via "EXIT" occurs, thus expediting the ensuing JUMP to output the word via "OUTWD". The entire "ample" test encompasses lines #220-232. If not ample, I branch to "NEXTLN" and do the alternating blanks and underlines "trick". After I finally print out your program line number against the right margin, I change "LENNUM" to zero (line #268) so that your line number will not be printed out on the second and succeeding lines of print occupied by the same program statement. Now that "LENNUM" is 0, the comparison of line #255 will negate your line number being printed on these succeeding lines. Lines #263-267 initialize "LINECHAR" on these succeeding lines with the same value it had at the beginning of the initial line.

Onto the module "OUTWD" at line #276. See - I do skip  
contd.

the keyword tests when the stored word is part of a REMark or between quotes. It looks like I finally get to print the stored word, beginning with line #281. When the last character of a complete word is a <CR>, I re-initialize only the REMark and quote flags because the "HIMEM"/"LOMEM" flag is automatically reset with the first blank after the immediately-succeeding colon (see line #146). If all of your quotes are evenly paired, then its flag is automatically reset at line #116. But, if your final instruction in a given program line looks like . . . PRINT "ABC <CR>, Applesoft BASIC won't hiccup. So, just in case your program has this scene, I better reset the quote flag with the <CR> output from the text file. Notice I did wait to reset this quote flag until after the word was printed. This is because of lines #278, 279. The comment section of the assembly listing leads you through the code remaining in this module.

Lines #320-380 depict the assembly of the Put MaCro (PMC) instructions that attend to the keyword comparisons. Notice at the end of each module that we JUMP to a unique indenting routine that addresses, for example, indenting of "FOR--NEXT" loops. More on these indenting routines later.

The <CR> module that follows on line #390 accomplishes two things. First, it prints the <CR>; second, it counts lines printed per page for pagination control. At the very, very end I naturally indent (if required), so I'm ready to print the ensuing words on the next line. The routine "INDENT" is a generalized indenting routine that I will describe shortly.

Well, here comes that nifty 25-byte routine developed by the great BEAGLE BROTHERS. Let's look at it closely! First:

1. INVERSE letters have their left-most bits 6 & 7 clear (= 0).
2. FLASHing letters have bit 7 clear.
3. CTRL letters have bit 6 clear.

For example,

1. NORMAL "A" = %11000001 = \$C1
2. INVERSE "A" = %00000001 = \$01
3. FLASHing "A" = %01000001 = \$41
4. CTRL "A" = %10000001 = \$81

and so forth for "B" thru "Z".

What about numbers?

1. INVERSE #s, like letters, have their left-most bits 6 & 7 clear.
2. FLASHing #s have bit 7 clear, but it's already clear. What now? Well, notice that a FLASHing letter = INVERSE letter + \$40. It looks like this relationship also works on numbers.
3. CTRL #s = NORMAL #s.

Finally, we must address all those "special" characters such as the ampersand, = and ! Look up all these non-letters and non-numbers and you will soon see that the following almost universal rules apply, at least when the characters exist (CTRL-! doesn't, for example, just as a CTRL-7 doesn't):

1. NORMAL : NORMAL
2. INVERSE : clear bits 6 & 7
3. FLASHing : INVERSE + \$40
4. CTRL : either INVERSE + \$80 or FLASHing + \$40

There are only two exceptions to these rules: the

question mark and the equals sign. Please double-check me here to be sure.

Now onto the "CTRL" routine by the BEAGLE BROTHERS. Looking at your handy-dandy chart of ASCII character codes, keep in mind that the only desired result is to convert CTRL letters to lower case. So,

1. Look for the characters with the lowest ASCII code. These characters are the INVERSE letters, all < \$20. So, "CTRL" immediately performs this comparison and if < \$20, bits 6 & 7 are set, thus converting INVERSE letters to NORMAL. By tracing "CTRL", you see that you will then quickly exit with now-NORMAL letters.
2. Then, we look for FLASHing letters and INVERSE numbers, along with some of the above-mentioned INVERSE and CTRL "special" characters; that is, all characters whose ASCII code is between \$20 and \$5F, inclusively. If in this range, bit 7 is set, thus converting them to NORMAL which leads to the same quick exit out of "CTRL". Note that some CTRL "special" characters are also converted to NORMAL. Once again, however, we are only interested in CTRL letters.
3. Next, we look for the remaining non-CTRL letters, which category includes ASCII character codes between \$60 and \$7F, inclusively. These encompass all the FLASHing #s & "special" characters, along with all remaining INVERSE "special" characters.
4. Now, for the finale !!! We are presented with the final category of ASCII codes between \$80 and \$9F, inclusively. This looks like the CTRL letters we're interested in, along with the remaining few CTRL "special" characters (like a CTRL-~, but what does a lower case CTRL-~ look like?). Well, line #423 converts all these CTRL letters to lower case. Phew!

The "PRINTOUT" module simply performs a warm start to much-earlier-mentioned \$C102, protecting the A, X & Y registers in the process. Following this are the keyword indenting modules:

1. With a "FOR" stored, I indent as explained earlier.
2. With a "NEXT" stored, I first test to see if this "NEXT" follows a "THEN" as exemplified by Statement #410 in my Applesoft Driver. If and only if it does not, then I test the "FOR" flag ("DENT1"). If already 0, then execute the "RTS". What?? More "NEXT"s than "FOR"s? Look at my Applesoft Driver. Only the "IF--THEN" syntax permits 3 "NEXT"s and only 1 "FOR" as an example.
3. With an "IF", I set its flag.
4. With its matching "THEN", I must print the alternating blanks and underlines, proceed to the next line and then indent (if required). Believe it or not, the module "INDENT4" gets all three accomplished.

The next module "INDENT" serves as the generalized indenting routine producing 5 leading blanks for each "DENT" unit. Of course, I also have to increment "LINECHAR" accordingly.

Since I began this "small" article with an explanation of the module "CHARO", I guess I'm at "DONE". If you do not have an Assembler, feel free to use my menu-driven Machine language program editor to enter "FORMAT". This editor is written entirely in Applesoft and was published in the March 1984 issue of Washington Apple Pi Journal.

contd.

```

9125: DO 3E 77      BNE QUOT      ; or after a ":"
9127: AD FO 95 78      LDA NUMCLN    ; (mult. instr.
912A: DO 39 79      BNE QUOT      ; on same line).

81 *****

912C: AD FE 95 83  LINNUM  LDA ASAVE      ; "LINECHAR"
912F: CD C3 95 84      CMP ZERO      ; <= 5. If = #,
9132: 90 05 85      BCC NOTNUM    ; store. If not,
9134: CD C7 95 86      CMP CLN       ; set "WORDCHAR"
9137: 90 03 87      BCC STORENUM ; = 1 for the
9139: 4C 65 91 88  *  NOTNUM  JMP QUOT      ; next word.
89
913C: AD F8 95 90  STORENUM LDA LINECHAR ; Store len. of
913F: 38 91          SEC          ; number in
9140: ED C1 95 92      SBC INIT      ; "LENNUM" &
9143: AA 93          TAX          ; line number
9144: 8E 77 95 94      STX LENNUM    ; in "LNNUM".
9147: 8E 78 95 95      STX LONNUM
914A: AD FE 95 96      LDA ASAVE
914D: 9D 71 95 97      STA LNNUM-1, X
89
9152: 8D C0 95 99      STA PREBLK
9155: A9 00 100      LDA #0
9157: 8D F7 95 101     STA WORDCHAR,X
915A: A9 03 102      LDA #3
915C: 8D 79 95 103     STA SUB
915F: AD C5 95 104     LDA CTRLA    ; Output "CTRL-A"
9162: 4C 63 95 105     JMP DONE     ; to get next char

107 *****

9165: AD FE 95 109  QUOT    LDA ASAVE      ; A quote (") ?
9168: CD C8 95 110      CMP QUOTE     ; If so, test
916B: DO 12 111      BNE EMPTY    ; for number
916D: EE F3 95 112      INC NUMQTE   ; of quotes.
9170: AE F3 95 113      LDX NUMQTE   ; If = 2, then
9173: EO 01 114      CPX #1       ; reset to 0.
9175: FO 05 115      BEQ QTEOUT
9177: A2 00 116      LDX #0
9179: 8E F3 95 117      STX NUMQTE
917C: 4C 54 92 118  QTEOUT  JMP STOREWD

120 *****

917F: CD C4 95 122  EMPTY   CMP BLANK     ; A blank ?
9182: DO 55 123      BNE COLON
9184: AE F7 95 124  TEST1   LDX WORDCHAR ; Ignore blanks
9187: EO 02 125      CPX #2       ; that follow
9189: 90 03 126      BCC TEST2    ; line number.
918B: 4C B6 91 127      JMP ELEPH    ; Use 2 tests.
918E: AD F8 95 128  TEST2   LDA LINECHAR
9191: 38 129          SEC
9192: ED C1 95 130      SBC INIT
9195: ED 78 95 131      SBC LONNUM
9198: ED 79 95 132      SBC SUB
919B: B0 19 133      BCS ELEPH
919D: AE 79 95 134      LDX SUB
91A0: EO 04 135      CPX #4
91A2: 90 06 136      BCC EMPTY1
91A4: CE F8 95 137      DEC LINECHAR
91A7: CE C0 95 138      DEC PREBLK
91AA: EE C0 95 139  EMPTY1  INC PREBLK   ; Count blanks.
91AD: CE F7 95 140      DEC WORDCHAR
91B0: AD C5 95 141      LDA CTRLA
91B3: 4C 63 95 142      JMP DONE
91B6: AD C4 95 143  ELEPH   LDA BLANK     ; If MEM. flag
91B9: AE F1 95 144      LDX MEMORY   ; set, reset &
91BC: FO 08 145      BEQ COMPLETE ; store. Then,
91BE: A2 00 146      LDX #0       ; word ends with
91C0: 8E F1 95 147      STX MEMORY   ; 2nd blank.
91C3: 4C 54 92 148      JMP STOREWD
91C6: AC F7 95 149  COMPLETE LDY WORDCHAR ; If no MEM flag,
91C9: 88 150          DEY          ; 1st blank shows
91CA: 99 7A 95 151      STA WORD,Y   ; end of word.
91CD: CE F8 95 152      DEC LINECHAR
91D0: 8C B0 95 153      STY WORDLEN ; Set "WORDLEN".
91D3: 20 66 92 154      JSR LINE
91D6: 4C E4 92 155      JMP OUTWD

157 *****

91D9: CD C7 95 159  COLON   CMP CLN       ; A ":" ?
91DC: DO 51 160      BNE CRET
91DE: AE F2 95 161      LDX NUMRMK   ; A ":" shows
91E1: DO 71 162      BNE STOREWD ; end of word
; if not in a

91E3: AE F3 95 163      LDX NUMQTE   ; REMark or
91E6: DO 6C 164      BNE STOREWD ; between " ".
91E8: EE FO 95 165      INC NUMCLN
91EB: AD F8 95 166      LDA LINECHAR ; If colon
91EE: 38 167          SEC          ; follows line
91EF: ED C1 95 168      SBC INIT     ; #, do not
91F2: ED 78 95 169      SBC LONNUM   ; add <CR>.
91F5: E9 03 170      SBC #3
91F7: 90 30 171      BCC COLON1
91F9: AC F7 95 172      LDY WORDCHAR ; Otherwise, add
91FC: 88 173          DEY          ; ":" & <CR>.
91FD: AD C7 95 174      LDA CLN
9200: 99 7A 95 175      STA WORD,Y
9203: 8C B0 95 176      STY WORDLEN ; Set "WORDLEN".
9206: CE F8 95 177      DEC LINECHAR
9209: 20 50 93 178      JSR HIGH     ; Do not add <CR>
920C: AC BD 95 179      LDY WORDLEN ; after colon in
920F: AE F1 95 180      LDX MEMORY   ; "HIMEM:"/"LOMEM:".
9212: DO OF 181      BNE HILO
9214: AD C6 95 182  ADDCR   LDA CR        ; If all rqmts.
9217: 99 78 95 183      STA WORD+1,Y ; are met, then
921A: EE F7 95 184      INC WORDCHAR ; add a <CR>
921D: 20 66 92 185      JSR LINE     ; after the
9220: 4C E4 92 186      JMP OUTWD    ; trailing
9223: AD C5 95 187  HILO   LDA CTRLA    ; colon.
9226: 4C 63 95 188      JMP DONE
9229: AD C7 95 189  COLON1 LDA CLN
922C: 4C 54 92 190      JMP STOREWD

192 *****

922F: CD C6 95 194  CRET    CMP CR        ; <CR> ?
9232: DO 20 195      BNE STOREWD ; (end of
9234: AC F7 95 196      LDY WORDCHAR ; line).
9237: 88 197          DEY          ; Store <CR>.
9238: 99 7A 95 198      STA WORD,Y
923B: 8C B0 95 199      STY WORDLEN ; Set "WORDLEN".
923E: CE F8 95 200      DEC LINECHAR
9241: A2 00 201      LDX #0
9243: 8E FO 95 202      STX NUMCLN  ; Reset ":" flag.
9246: AE F6 95 203      LDX DENT3   ; Reset "IF-THEN"
9249: FO 06 204      BEQ CRET1   ; indent flag.
924B: CE F4 95 205      DEC DENT
924E: CE F6 95 206      DEC DENT3
9251: 4C 14 92 207  CRET1   JMP ADDCR    ; Add 2nd <CR>.

209 *****

9254: 20 CD 94 211  STOREWD JSR CTRL     ; Convert CTRL char.
9257: AE F7 95 212      LDX WORDCHAR
925A: 9D 79 95 213      STA WORD-1,X ; Store in "WORD"
925D: CE F8 95 214      DEC LINECHAR
9260: AD C5 95 215      LDA CTRLA
9263: 4C 63 95 216      JMP DONE

218 *****

9266: AD FA 95 220  LINE    LDA WIDE     ; Quantify the
9269: 38 221          SEC          ; remaining
926A: ED F8 95 222      SBC LINECHAR ; line space.
926D: 8D BF 95 223      STA NUMBLK   ; If ample,
9270: ED BD 95 224      SBC WORDLEN ; print word.
9273: A8 225          TAY
9274: 90 0E 226      BCC NEXTLN
9276: AE BD 95 227      LDX WORDLEN
9279: BD 7A 95 228      LDA WORD,X
927C: CD C6 95 229      CMP CR
927F: FO 62 230      BEQ EXIT
9281: 88 231          DEY
9282: 10 5F 232      BPL EXIT
9284: AD BF 95 233  NEXTLN  LDA NUMBLK   ; If not ample,
9287: 18 234          CLC          ; go to next
9288: 6D C0 95 235      ADC PREBLK   ; line.
928B: AA 236          TAX
928C: CA 237          DEX
928D: FO 22 238      BEQ SPACE
928F: A9 FF 239      LDA #$$F    ; First, print
9291: 8D C2 95 240      STA COUNTER ; (WIDE-LINECHAR
9294: AC 77 95 241  COUNT   LDY LENNUM   ; +PREBLK-1)
9297: FO OF 242      BEQ BLK     ; alternating
9299: EE C2 95 243      INC COUNTER ; blanks and
929C: AD C2 95 244      LDA COUNTER ; underlines.
929F: 4A 245          LSR
92A0: 90 06 246      BCC BLK
92A2: AD C9 95 247  DASH   LDA UNDER
92A5: 4C AB 92 248      JMP BEFORE

contd.

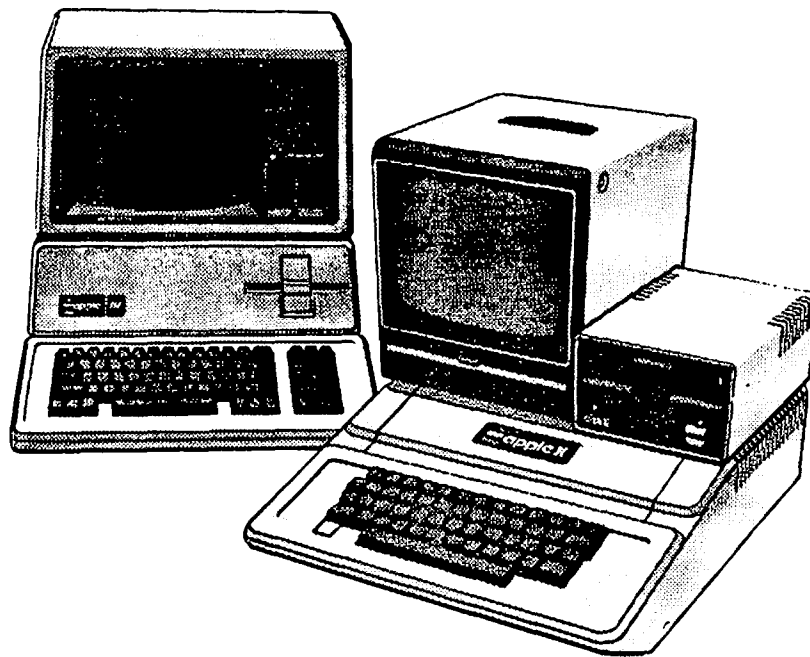
```

# There's only one place to buy apples.

AT FREDERICK COMPUTER PRODUCTS !!!

As an Authorized Apple Dealer we offer a full selection of both Apple Hardware and Software at low, low prices.

Also, as an Authorized Service Center we provide fast and efficient service on all Apple products.



## **FREDERICK COMPUTER PRODUCTS, INC.**

Microcomputer Systems And Peripherals

5726 INDUSTRY LANE-FREDERICK, MD.- (301) 694-8884

 **apple computer**  
® Authorized Dealer

```

92A8: AD C4 95 249 BLK LDA BLANK
92AB: 20 E6 94 250 BEFORE JSR PRINTOUT
92AE: CA 251 DEX
92AF: D0 E3 252 BNE COUNT
92B1: AD C4 95 253 SPACE LDA BLANK ; 2nd, print
92B4: 20 E6 94 254 JSR PRINTOUT ; blank. Then
92B7: EC 77 95 255 NUM CPX LENNUM ; line #.
92BA: B0 0A 256 BCS NEWLN
92BC: B0 72 95 257 LDA LNUM,X
92BF: 20 E6 94 258 JSR PRINTOUT
92C2: E8 259 INX
92C3: 4C B7 92 260 JMP NUM
92C6: AD C6 95 261 NEWLN LDA CR ; Jump to next
92C9: 20 A3 94 262 JSR <CR> ; line to
92CC: AD F8 95 263 LDA LINECHAR ; print the
92CF: 18 264 CLC ; word.
92D0: 6D 78 95 265 ADC LONNUM
92D3: 6D C0 95 266 ADC PREBLK
92D6: 8D F8 95 267 STA LINECHAR
92D9: A9 00 268 LDA #0
92DB: 8D 77 95 269 STA LENNUM
92DE: A9 04 270 LDA #4
92E0: 8D 79 95 271 STA SUB
92E3: 60 272 EXIT RTS

274 *****
92E4: AE F2 95 276 OUTWD LDX NUMRMK ; If in REMark,
92E7: D0 08 277 BNE OVERKEY ; or inside " ".
92E9: AE F3 95 278 LDX NUMQTE ; skip test for
92EC: D0 03 279 BNE OVERKEY ; a keyword.
92EE: 20 B4 93 280 KEY JSR REMARK ; Key words ?
92F1: A0 00 281 OVERKEY LDY #0 ; Print word.
92F3: B9 7A 95 282 NEXTCHR LDA WORD,Y
92F6: CD C6 95 283 CMP CR
92F9: D0 37 284 BNE PRT
92FB: AD FA 95 285 LDA WIDE
92FE: 38 286 SEC
92FF: ED F8 95 287 SBC LINECHAR
9302: ED BD 95 288 SBC WORDLEN
9305: 8D BF 95 289 STA NUMBLK
9308: AE BD 95 290 LDX WORDLEN
930B: BD 7A 95 291 LDA WORD,X
930E: CD C6 95 292 CMP CR
9311: F0 03 293 BEQ REINIT
9313: CE BF 95 294 DEC NUMBLK
9316: A2 00 295 REINIT LDX #0 ; Re-initialize
9318: 8E F2 95 296 STX NUMRMK ; flags.
931B: 8E F3 95 297 STX NUMQTE
931E: 20 B4 92 298 JSR NEXTLN
9321: AE F7 95 299 LDX WORDCHAR
9324: 8D 78 95 300 LDA WORD-2,X
9327: CD C6 95 301 CMP CR
932A: D0 19 302 BNE OVER
932C: 20 A3 94 303 JSR <CR> ; Double space
932F: 4C 45 93 304 JMP OVER ; at end of
9332: 20 E6 94 305 PRT JSR PRINTOUT ; the line.
9335: C8 306 INY
9336: CC F7 95 307 CPY WORDCHAR
9339: 90 B8 308 BCC NEXTCHR
933B: 18 309 NEXTWD CLC ; Incr. "LINECHAR"
933C: AD F8 95 310 LDA LINECHAR ; to incl. length
933F: 6D F7 95 311 ADC WORDCHAR ; of word.
9342: 8D F8 95 312 STA LINECHAR
9345: A2 00 313 OVER LDX #0 ; Re-initialize
9347: 8E F7 95 314 STX WORDCHAR ; "WORDCHAR".
934A: AD C5 95 315 LDA CTRLA
934D: 4C 63 95 316 JMP DONE

9350: A2 D9 320 HIGH PMC KEYWD/MEM1; Keywd "HIMEM".
9352: 8E BE 95 320 LDX #*Y" ; Compare your
9355: A2 FF 320 STX ANS ; listed word
9357: A0 00 320 LDX #*FF ; to this key
9359: B9 CA 95 320 LDY #0 ; word. This
935C: CD BD 95 320 LDA MEM1,Y ; comparison is
935F: D0 0F 320 CMP WORDLEN ; accomplished
9361: E8 320 BNE NO ; by 1st checking
9362: C8 320 NEXTLTR INX ; 2 lengths.
9363: EC BD 95 320 INY ; If =, then
9366: F0 0D 320 CPX WORDLEN ; each char.
9368: B9 CA 95 320 BEQ OUT ; is checked in
936B: DD 7A 95 320 LDA MEM1,Y ; the loop =
936E: F0 F1 320 CMP WORD,X ; the label,
BEQ NEXTLTR ; "NEXTLTR".
LDX #*N" ; "ANS" gives
STX ANS ; the result.
EOM ; End of Macro.

9370: A2 CE 320 NO LDX #*N" ; "ANS" gives
9372: 8E BE 95 320 STX ANS ; the result.
320 OUT EOM ; End of Macro.

9375: AE BE 95 322 LDX ANS
9378: E0 D9 323 CPX #*Y"
937A: D0 06 324 BNE LOW
937C: A0 01 325 LDY #1 ; Set "HIMEM"
937E: 8C F1 95 326 STY MEMORY ; flag.
9381: 60 327 RTS

329 *****
9382: A2 D9 331 LOW PMC KEYWD/MEM2; Keywd "LOMEM".
9384: 83 BE 95 331 LDX #*Y" ; Compare your
9387: A2 FF 331 STX ANS ; listed word
9389: A0 00 331 LDX #*FF ; to this key
938B: B9 D0 95 331 LDY #0 ; word. This
938E: CD BD 95 331 LDA MEM2,Y ; comparison is
9391: D0 0F 331 CMP WORDLEN ; accomplished
9393: E8 331 BNE NO ; by 1st checking
9394: C8 331 NEXTLTR INX ; 2 lengths.
9395: EC BD 95 331 INY ; If =, then
9398: F0 0D 331 CPX WORDLEN ; each char.
9399: B9 D0 95 331 BEQ OUT ; is checked in
939D: DD 7A 95 331 LDA MEM2,Y ; the loop =
93A0: F0 F1 331 CMP WORD,X ; the label,
93A0: F0 F1 331 BEQ NEXTLTR ; "NEXTLTR".
9370: A2 CE 331 NO LDX #*N" ; "ANS" gives
9372: 8E BE 95 331 STX ANS ; the result.
331 OUT EOM ; End of Macro.

93A7: AE BE 95 333 LDX ANS
93AA: E0 D9 334 CPX #*Y"
93AC: D0 05 335 BNE BACK
93AE: A0 01 336 LDY #1 ; Set "LOMEM"
93B0: 8C F1 95 337 STY MEMORY ; flag.
93B3: 60 338 BACK RTS

340 *****
93B4: A2 D9 342 REMARK PMC KEYWD/RMK ; Keywd "REM".
93B6: 8E BE 95 342 LDX #*Y" ; Compare your
93B9: A2 FF 342 STX ANS ; listed word
93BB: A0 00 342 LDX #*FF ; to this key
93BD: B9 D6 95 342 LDY #0 ; word. This
93C0: CD BD 95 342 LDA RMK,Y ; comparison is
93C3: D0 0F 342 CMP WORDLEN ; accomplished
93C5: E8 342 BNE NO ; by 1st checking
93C6: C8 342 NEXTLTR INX ; 2 lengths.
93C7: EC BD 95 342 INY ; If =, then
93CA: F0 0D 342 CPX WORDLEN ; each char.
93CC: B9 D6 95 342 BEQ OUT ; is checked in
93CF: DD 7A 95 342 LDA RMK,Y ; the loop =
93D2: F0 F1 342 CMP WORD,X ; the label,
93D4: A2 CE 342 NO BEQ NEXTLTR ; "NEXTLTR".
93D6: 8E BE 95 342 LDX #*N" ; "ANS" gives
342 OUT STX ANS ; the result.
EOM ; End of Macro.

9309: AE BE 95 344 LDX ANS
93DC: E0 D9 345 CPX #*Y"
93DE: D0 06 346 BNE FOR
93E0: A0 01 347 LDY #1 ; Set "REM"
93E2: 8C F2 95 348 STY NUMRMK ; flag.
93E5: 60 349 RTS

351 *****
93E6: A2 D9 353 FOR PMC KEYWD/KEY1; Keywd "FOR".
93E8: 8E BE 95 353 LDX #*Y" ; Compare your
93EB: A2 FF 353 STX ANS ; listed word
93ED: A0 00 353 LDX #*FF ; to this key
93EF: B9 DA 95 353 LDY #0 ; word. This
93F2: CD BD 95 353 LDA KEY1,Y ; comparison is
93F5: D0 0F 353 CMP WORDLEN ; accomplished
93F7: E8 353 BNE NO ; by 1st checking
93F8: C8 353 NEXTLTR INX ; 2 lengths.
93F9: EC BD 95 353 INY ; If =, then
93FC: F0 0D 353 CPX WORDLEN ; each char.
93FE: B9 DA 95 353 BEQ OUT ; is checked in
9401: DD 7A 95 353 LDA KEY1,Y ; the loop =
9404: F0 F1 353 CMP WORD,X ; the label,
9406: A2 CE 353 NO BEQ NEXTLTR ; "NEXTLTR".
9408: 8E BE 95 353 LDX #*N" ; "ANS" gives
STX ANS ; the result.

contd.

```



# FONT DOWNLOADER & EDITOR

for Apple II+, IIe & Franklin Computers

DOWNLOAD CUSTOM FONTS & CHARACTERS TO EITHER OF THESE PRINTERS: EPSON FX-80, OKIDATA 92/93 C.I.TOH PROWRITER 8510AP, APPLE DOT MATRIX

COMPATIBLE WITH WORD PROCESSORS CAPABLE OF SENDING PRINTER CONTROL CODES

## MENU DRIVEN SOFTWARE

FONT DOWNLOADER & EDITOR  
COPYRIGHT 1983 RAK-WARE

- 0 . . . EXIT
- 1 . . . DOWNLOAD FONT
- 2 . . . EDIT FONT
- 3 . . . RUN DEMO
- 4 . . . INSTRUCTIONS
- 5 . . . CONFIGURE I/F (DUMB)

**FONT EDITOR**

! " # \$ % & ' ( ) * +	
, - . / 0 1 2 3 4 5 6 7	
8 9 : ; < = > ? @ A B C	
D E F G H I J K L M N O	
P Q R S T U V W X Y Z [	
\ ] ^ _ ` a b c d e f g	
h i j k l m n o p q r s	
t u v w x y z {   }	

- <ESC> ENTERS EDIT MODE
- J I K CHARACTER SELECT
- M
- <RTN> MOVES CURSOR TO NEXT CHAR
- R REPLACE CHAR - ENTER EDIT
- L LOAD FONT FROM DISK
- S SAVE FONT TO DISK
- D DOWNLOAD FONT TO PRINTER
- T TOGGLE SOUND ON/OFF

RAK-WARE (201)325-1885  
41 RALPH ROAD, W.ORANGE, NJ 07052

\$39 + \$3 S/H  
SPECIFY PRINTER WITH ORDER

# TRI-STATE SYSTEMS

## DISCOUNT APPLE SOFTWARE

	Retail	Our Price		Retail	Our Price
Lode Runner.....	34.95	25.95	The Accountant.....	129.00	89.95
Zork I, II, or III.....	39.95	29.50	ASC II Express: The Professional.	129.95	89.95
Zaxxon.....	39.95	27.50	Screenwriter II.....	129.95	89.95
Choplifter.....	34.95	25.95	Master Type.....	39.95	27.95
Kraft Joystick.....	49.95	39.95	Bank Street Writer.....	69.95	49.95
Exodus: Ultima III.....	54.95	38.95	Word Star.....	495.00	289.95
Wizardry.....	49.95	37.95	Galactic Adventures.....	59.95	42.95
Wizardry II-Knight of Diamonds.....	34.95	25.95	Home Accountant.....	74.95	56.95
Wizardry III-Legacy of Lylqaymn.....	39.95	29.95	Verbatim (10 disks) 5 1/4".....	42.00	26.95

Mail orders to:

**TRI-STATE SYSTEMS**

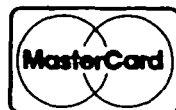
P.O. Box 2544  
Springfield, VA 22152

or call: (703) 569-9363

Hours: 9AM-6PM LEAVE MESSAGE  
7PM-8PM CALLBACK



No credit card penalty!



- 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.

```

353 OUT      EOM      ; End of Macro.
940B: AE BE 95 355    LDX ANS
940E: E0 D9 356      CPX #*Y"
9410: D0 03 357      BNE NEXT
9412: 4C FC 94 358    JMP INDENT1

```

```

360 *****
362 NEXT      PMC KEYWD/KEY2; Keywd "NEXT".
9415: A2 D9 362      LDX #*Y"      ; Compare your
9417: 8E BE 95 362    STX ANS        ; listed word
941A: A2 FF 362      LDX #$$$FF    ; to this key
941C: A0 00 362      LDY #0        ; word. This
941E: B9 DE 95 362    LDA KEY2,Y    ; comparison is
9421: CD BD 95 362    CMP WORDLEN  ; accomplished
9424: D0 0F 362      BNE NO        ; by 1st checking
9426: E8 362        NEXTLTR INX      ; 2 lengths.
9427: C8 362        INY          ; If =, then
9428: EC BD 95 362    CPX WORDLEN  ; each char.
942B: F0 OD 362      BEQ OUT      ; is checked in
942D: B9 DE 95 362    LDA KEY2,Y    ; the loop =
9430: DD 7A 95 362    CMP WORD,X   ; the label,
9433: F0 F1 362      BEQ NEXTLTR  ; "NEXTLTR".
9435: A2 CE 362      NO          LDX #*N"    ; "ANS" gives
9437: 8E BE 95 362    STX ANS      ; the result.
362 OUT      EOM      ; End of Macro.

```

```

943A: AE BE 95 364    LDX ANS
943D: E0 D9 365      CPX #*Y"
943F: D0 03 366      BNE IF
9441: 4C 05 95 367    JMP INDENT2

```

```

369 *****
371 IF      PMC KEYWD/KEY3; Keywd "IF".
9444: A2 D9 371      LDX #*Y"      ; Compare your
9446: 8E BE 95 371    STX ANS        ; listed word
9449: A2 FF 371      LDX #$$$FF    ; to this key
944B: A0 00 371      LDY #0        ; word. This
944D: B9 E3 95 371    LDA KEY3,Y    ; comparison is
9450: CD BD 95 371    CMP WORDLEN  ; accomplished
9453: D0 0F 371      BNE NO        ; by 1st checking
9455: E8 371        NEXTLTR INX      ; 2 lengths.
9456: C8 371        INY          ; If =, then,
9457: EC BD 95 371    CPX WORDLEN  ; each char.
945A: F0 OD 371      BEQ OUT      ; is checked in
945C: B9 E3 95 371    LDA KEY3,Y    ; the loop =
945F: DD 7A 95 371    CMP WORD,X   ; the label,
9462: F0 F1 371      BEQ NEXTLTR  ; "NEXTLTR".
9464: A2 CE 371      NO          LDX #*N"    ; "ANS" gives
9466: 8E BE 95 371    STX ANS      ; the result.
371 OUT      EOM      ; End of Macro.

```

```

9469: AE BE 95 373    LDX ANS
946C: E0 D9 374      CPX #*Y"
946E: D0 03 375      BNE THEN
9470: 4C 16 95 376    JMP INDENT3

```

```

378 *****
380 THEN    PMC KEYWD/KEY4; Keywd "THEN".
9473: A2 D9 380      LDX #*Y"      ; Compare your
9475: 8E BE 95 380    STX ANS        ; listed word
9478: A2 FF 380      LDX #$$$FF    ; to this key
947A: A0 00 380      LDY #0        ; word. This
947C: B9 E6 95 380    LDA KEY4,Y    ; comparison is
947F: CD BD 95 380    CMP WORDLEN  ; accomplished
9482: D0 0F 380      BNE NO        ; by 1st checking
9484: E8 380        NEXTLTR INX      ; 2 lengths.
9485: C8 380        INY          ; If =, then
9486: EC BD 95 380    CPX WORDLEN  ; each char.
9489: F0 OD 380      BEQ OUT      ; is checked in
948B: B9 E6 95 380    LDA KEY4,Y    ; the loop =
948E: DD 7A 95 380    CMP WORD,X   ; the label,
9491: F0 F1 380      BEQ NEXTLTR  ; "NEXTLTR".
9493: A2 CE 380      NO          LDX #*N"    ; "ANS" gives
9495: 8E BE 95 380    STX ANS      ; the result.
380 OUT      EOM      ; End of Macro.

```

```

9498: AE BE 95 382    LDX ANS
949B: E0 D9 383      CPX #*Y"
949D: D0 03 384      BNE RTN
949F: 4C 1A 95 385    JMP INDENT4
94A2: 60 386 RTN      RTS

```

; Not any keyword.

```

388 *****
94A3: 20 E6 94 390 <CR> JSR PRINTOUT ; Output <CR>.
94A6: EE F9 95 391    INC PAGELL   ; Up line count.
94A9: AD F9 95 392    LDA PAGELL   ; Compare with
94AC: 38 393        SEC          ; 11/pg. input
94AD: ED FB 95 394    SBC LINESPP ; from Driver.
94B0: 90 13 395      BCC NOTFULL ; If new count
396 *

```

```

94B2: AD FC 95 397 FULL LDA FIRST    ; is < the
94B5: 20 E6 94 398    JSR PRINTOUT ; desired 11/pg.
94B8: AD FD 95 399    LDA SECOND   ; page "NOTFULL".
94BB: F0 03 400      BEQ RESETPG  ; If page full,
94BD: 20 E6 94 401    JSR PRINTOUT ; paginate.
402 *
94C0: A2 00 403 RESETPG LDX #0      ; When new page,
94C2: 8E F9 95 404    STX PAGELL   ; reset count.
405 *
94C5: A2 00 406 NOTFULL LDX #0      ; New page or not,
94C7: 8E F8 95 407    STX LINECHAR ; reset "LINECHAR"
94CA: 4C 31 95 408    JMP INDENT   ; & set indent.

```

```

410 *****
94CD: C9 20 412 CTRL    CMP #$$$20   ; Converts CTRL
94CF: B0 02 413        BCS INVERSE ; to lower case.
94D1: 09 C0 414        ORA #X1100000; INV--NORMAL.
94D3: C9 60 415 INVERSE CMP #$$$60
94D5: B0 02 416        BCS FLASH
94D7: 09 80 417        ORA #X1000000; INV/FLASH--NORM.
94D9: C9 80 418 FLASH  CMP #$$$80
94DB: B0 02 419        BCS CONTRL
94DD: 69 40 420        ADC #64     ; FLASH--NORMAL.
94DF: C9 A0 421 CONTRL CMP #$$$A0
94E1: B0 02 422        BCS LOWER  ; Already NORMAL.
94E3: 09 60 423        ORA #X01100000; CTRL--lower case.
94E5: 60 424 LOWER    RTS

```

```

426 *****
94E6: 8D 6F 95 428 PRINTOUT STA ATEMP ; Save registers.
94E9: 8E 70 95 429    STX XTEMP
94EC: 8C 71 95 430    STY YTEMP
94EF: 20 02 C1 431    JSR PRINTER ; Print Accum.
94F2: AD 6F 95 432    LDA ATEMP   ; Retrieve reg.
94F5: AE 70 95 433    LDX XTEMP
94F8: AC 71 95 434    LDY YTEMP
94FB: 60 435        RTS

```

```

437 *****
94FC: EE F5 95 439 INDENT1 INC DENT1   ; Keywd "FOR".
94FF: EE F4 95 440    INC DENT
9502: 4C 31 95 441    JMP INDENT  ; Indent.
9505: AD F6 95 443 INDENT2 LDA DENT3   ; Keywd "NEXT".
9508: D0 0B 444        BNE IND2    ; If after "THEN",
950A: AD F5 95 445    LDA DENT1   ; no indent.
950D: F0 06 446        BEQ IND2
950F: CE F5 95 447    DEC DENT1  ; For nested
9512: CE F4 95 448    DEC DENT   ; loops.
9515: 60 449 IND2     RTS

```

```

9516: EE F6 95 451 INDENT3 INC DENT3   ; Keywd "IF".
9519: 60 452        RTS
951A: 18 454 INDENT4 CLC          ; Keywd "THEN".
951B: AD F5 95 455    LDA DENT1
951E: 6D F6 95 456    ADC DENT3
9521: 8D F4 95 457    STA DENT3
9524: AD FA 95 458    LDA WIDE   ; Go to next
9527: 38 459        SEC          ; line before
9528: ED F8 95 460    SBC LINECHAR ; indenting.
952B: 8D BF 95 461    STA NUMBLK
952E: 4C 84 92 462    JMP NEXTLN

```

```

464 *****
9531: AE F4 95 466 INDENT LDX DENT   ; This indents
9534: F0 17 467        BEQ RET     ; prop. to "DENT".
468 *
9536: AD C4 95 469 NXTDNT LDA BLANK  ; One "DENT"
9539: A0 05 470        LDY #5     ; = 5 blanks.
471 *
953B: 20 E6 94 472 NXTBL JSR PRINTOUT
953E: 88 473        DEY

```

contd.

```

953F: DO FA 474 BNE NXTBL
9541: 18 475 CLC
9542: AD F8 95 476 LDA LINECHAR ; Increase
9545: 69 05 477 ADC #5 ; "LINECHAR"
9547: 80 F8 95 478 STA LINECHAR ; by 5 for
954A: CA 479 DEX ; each "DENT".
954B: DO E9 480 BNE NXTDNT

```

```

954D: 60 481 *
482 RET RTS ; Return.

```

484 \*\*\*\*\*

```

954E: A2 00 486 CHAR0 LDX #0 ; Compute
9550: 8E C1 95 487 STX INIT ; "LINECHAR"
9553: 8A 488 TXA ; at beginning
9554: AE F4 95 489 LDX DENT ; of line.
9557: FO 06 490 BEQ FINISH
9559: 18 491 CLC
955A: 69 05 492 NEXT0 ADC #5
955C: CA 493 DEX
955D: DO FB 494 BNE NEXT0
955F: 8D C1 95 495 FINISH STA INIT
9562: 60 496 RTS

```

498 \*\*\*\*\*

```

9563: 8D 10 C0 500 DONE STA STROBE ; Clear keybd.
9566: AE FF 95 501 LDX XSAVE ; Load back
9569: AC 00 96 502 LDY YSAVE ; X & Y reg.
956C: 4C 02 C1 503 JMP PRINTER ; Re-transmit.

```

505 \*\*\*\*\* DATA \*\*\*\*\*

```

507 ATEMP DS 1
508 XTEMP DS 1
509 YTEMP DS 1

```

```

511 LNUM DS 5
512 LENNUM DS 1
513 LONNUM DS 1
514 SUB DS 1
515 WORD DS 67
516 WORDLEN DS 1
517 ANS DS 1
518 NUMBLK DS 1
519 PREBLK DS 1
520 INIT DS 1
521 COUNTER DS 1

```

```

95C3: B0 523 ZERO HEX 80
95C4: A0 524 BLANK HEX A0
95C5: 81 525 CTRLA HEX 81
95C6: 8D 526 CR HEX 8D
95C7: BA 527 CLN HEX BA
95C8: A2 528 QUOTE HEX A2
95C9: DF 529 UNDER HEX DF

```

```

95CA: 05 531 MEM1 HEX 05
95CB: C8 C9 CD 532 ASC "HMEM"
95D0: 05 533 MEM2 HEX 05
95D1: CC CF CD 534 ASC "LOMEM"
95D6: 03 535 RMK HEX 03
95D7: D2 C5 CD 536 ASC "REM"
95DA: 03 537 KEY1 HEX 03
95DB: 04 538 KEY2 ASC "FOR"
95DE: 04 539 KEY2 HEX 04
95DF: CE C5 D8 540 ASC "NEXT"
95E3: 02 541 KEY3 HEX 02
95E4: C9 C6 542 ASC "IF"
95E6: 04 543 KEY4 HEX 04
95E7: D4 C8 C5 544 ASC "THEN"

```

546 \*\*\*\*\* End of Data \*\*\*\*\*

548 \*\*\*\*\* End of "FORMAT" \*\*\*\*\*

--End Assembly, 1259 bytes, Errors: 0

## CALL FOR ALL YOUR COMPUTER NEEDS

Printers:

C. Itoh ProWriter .....	345.00
ProWriter II .....	600.00
"Hot Dot" 180 CPS .....	475.00
180 CPS, 132 column ..	675.00
Color, 180 CPS .....	600.00
Okidata Microline 92 .....	429.00
Epson RX-80 .....	320.00
RX-80 F/T .....	385.00
FX-80 F/T .....	485.00
FX-100 .....	700.00

Accessories:

SuperSprite Graphics Pkg. ...	325.00
NETWORX Filtered Power Tree ..	54.00
Taxan Amber Monitor .....	135.00
Videx Function Strip .....	58.00
TG Track Ball .....	40.00
Wico Joysticks (3-way Deluxe) ..	35.00
Koala Pad Touch Tablet .....	89.50
Micro-Sci Disk Drive, A2 .....	235.00
FingerPrint (MX, RX Series) ..	45.00
(FX Series) .....	52.50

Grappler + .....	122.50
Buffered Grappler + .....	175.00
PKASO .....	122.50
SuffleBuffer (64K) .....	275.00
SuffleBuffer (128K) .....	350.00
TG Joystick (Iie) .....	45.00
Hayes Mach III Joystick (Iie) ..	41.25
Quick-Vis .....	14.00
ASTROFIRE (Rapid Fire) .....	12.00
AFPLEsurance Diagnostic Controller Card .....	80.00
Verbatim diskettes (10 packs) SS/DD .....	23.00
DS/DD .....	32.00
Holiday Packs (10 disks in plastic case, head cleaning kit) SS/DD .....	26.00
DS/DD .....	35.00

In stock word processing, utilities,  
and game specials:

Epson Printer Utility (condensed, italic, boldface, etc.) ...	30.00
The Artist .....	55.00
Super-Text 40/80 (II, II+) .....	55.00
Visidex .....	150.00
Desk Top Plan II .....	135.00
Ampersoft .....	19.00
Lexicom .....	19.00
V.I.P. II .....	12.00
Key Perfect .....	12.00
A.M.P. .....	10.00
Critical Mass .....	20.00
Epoch .....	19.00
Three Mile Island .....	20.00
Free Fall .....	16.00
Frogger .....	19.00
Mouskattack .....	19.00
Lunar Leaper .....	16.00
Apple-Oids .....	17.00
Firebug .....	16.00
Creature Creator .....	25.00
Spectre .....	17.00
Buzzard Bait .....	25.00
Sammy Lightfoot .....	18.00

COMPUTER DEN Ltd.  
11716 KEMP MILL ROAD  
SILVER SPRING, MARYLAND 20902  
301-649-6868

Call for an appointment or to order  
Sunday - Thursday, 10 A.M. - 9 P.M.  
Free delivery to Crystal City  
UPS shipping \$4.00 extra  
Quantity Discounts Available

# SOFTVIEWS

by David Morganstein

C.I.A. (Confidential Information Advisors). You might think the title implies a data base package. The small, unobtrusive ad for this product found in many magazines indicates you will obtain five sophisticated disk utilities and a 60,000+ word guide to the Apple disk. I found the ad to be quite accurate. Most owners of Beneath Apple DOS, an exceptional and well known reference to DOS 3.3, applaud "Beneath" as the most informative reference on the subject of Apple DOS. I suspect many owners of "Beneath" have or will obtain the CIA as a marvelous complement. The five utilities provided are: Tricky Dick (the shell program), the Linguist, the Code Breaker and the Tracer (subroutines which are called from Tricky Dick, only one of which can be loaded in memory at any one time) and the Tracker, a separate program.

Tricky Dick is a general purpose ZAP program for reading, examining, modifying and writing sectors. While it performs just about all of the functions found in other sector editors, it also has several new options which you will find useful. A single key-stroke provides you with a help screen, a function common to all the supplied programs. Not only can you disassemble machine code, you can LIST Applesoft OR Integer Basic programs found in the sector without exiting from Tricky Dick! If you are attempting to modify a commercial program you bought which is supplied in protected form, Tricky Dick allows you to enter non-standard sector marks which may allow you to access your software.

The Linguist is the nibble editor of the package. Its one very novel and useful feature is the ability to decode the address field into the volume number, track and sector. This decoding can be done in any of the three often-used nibble formats. Having examined the nibbles that comprise a sector, you can jump to TD to examine the decoded bytes they represent. One criticism of this routine is that re-entry from Tricky Dick forces a re-read of track zero. This inconvenience can be defeated through an easy to apply modification which limits the track read to a single selected track. Of course this modification must be undone to read other tracks.

The Code Breaker routine is used to modify the "byte translate table" found in the RWTS (read/write track/sector) subroutine in DOS. Fear not if the previous sentence sounded like computerese, the CIA documentation will make it clear! This part of the CIA is used to encode or decode non-standard versions of DOS.

The Tracer is somewhat misnamed. It should be called the Finder (with apologies to Mac!) It is used to locate all sectors which appear to be Track sectors, or Catalog sectors, or the VTOC (Volume Table of Contents). In addition it can search the entire disk for a user-supplied string or it can verify that sectors can be read with user-supplied sector marks.

I found the Tracker to be the least useful program. It is instructive but, I believe, of little value beyond that. Once run, it will provide you with a screen display of the location of the disk drive arm, that is, the current track and sector being accessed. While this can teach you what happens when a standard DOS program or file is accessed, it can not be easily made a permanent part of your screen display for use with non-standard DOS formats where it might be helpful to learn what is happening to the disk drive.

Given the value of the software, I must add that the accompanying manual is practically worth having on its own! While a fair part of the 127 pages is dedicated to using the programs, the instructions are done in a tutorial fashion. You will learn about DOS formatting and storage as you become acquainted with the routines. Two of the chapters are strictly about DOS. You might be able to discover much of the information with a thorough reading of Beneath Apple DOS, but some of it was new and interesting to me. The authors style is humorous and easy to read. My only complaint was the lack of an index (what they call an index is a table of contents). Unlike many products associated with making archival copies of your purchased software, the CIA is not copy-protected. In fact, the disk even contains a couple of "protected" tracks which you use for practice. All of the programs can be moved to a standard DOS 3.3 formatted disk. All in all, a valuable buy. Golden Delicious Software LTD., 350 Fifth Ave., Suite 330B, New York, N.Y. 10001. Price \$65.00

Pen-Pal. This low-end entry in the word processing field has a few nice features but may not meet the needs of those with serious applications. It is fairly easy to learn because it does not have a large number of special control functions. This is a mixed blessing since many such features can be quite handy. I found the cursor control keys to be awkward to use: scroll left is Ctrl-Q, scroll right is Ctrl-E, scroll up is Ctrl-W and down is Ctrl-X.

PP does not permit you to use an 80 column board but does give you an eighty column "feel" via a left and right half of the screen viewed in 40 column chunks. You can scroll the window left or right one character at a time, or you can jump 27 characters at a time. While this does the job, I found it confusing since there is some overlap of characters and my eye took a while to sort out what I was seeing.

The program uses a typical format of two operating modes: one for text Entry/Edit and one for Display to let you see the formatted result. However, after edit or entry, you must enter a command which causes the program to re-format the text to view the result in an organized display.

To format output, PP has a small number of embedded characters. These allow for: a new page, titling, line spacing and margin adjustment, as well as line centering and right justification. To use additional formatting capabilities of your printer, you can define up to four control character sequences which can be embedded within the text.

The documentation consists of a 71-page reference manual and several reference cards. A useful tutorial begins the documentation. You will find a "how to get started" section and a reference section, both understandable. Another feature of PP's documentation is on-screen help menus. There are five of these which can be called up to describe program features. Unfortunately, they are only available when the program is configured for 48K systems (which allow about 22K of text, enough for about eight single spaced pages). The menus can not be used with a 64K system which expands the capacity to about 12 pages.

Pen-Pal can read and write both text and binary files,  
contd. on pg 47

# ANCHOR MARK 12 REVIEW UPDATE

## by George Kinal

Apparently my timely review of the Anchor Automation Mark 12 300/1200 baud "smartmodem" has been pretty popular. I've received letters and phone calls from all over the country. Would you believe that WAP has members in California and in New Orleans?

As other WAP members and I have continued to use the Mark 12, several additional "features" have been uncovered. Here are the most prominent:

- On 300 baud signals that are too strong, the Mark 12 tends to garble the received data sporadically. The most annoying example of this is our own ABBS. This only happens from some telephone exchanges. Presumably, it will go away when the ABBS uses a Mark 12 as well. Anchor originally claimed this was a firmware problem, but a new PROM failed to change the situation. The signal can be reduced in strength by taking your phone off the hook once connection has been made.
- The Anchor modem does not support (send out) the so-called "break" signal. The break is used by some main-frame computers as a means of signalling interruption to an ongoing process, something like Ctrl-C on the Apple and on many BBSs. Even though your mainframe supports the break, you may not really need it for most applications. It comes in handy if you are listing a long file and need to pause or cancel further listing. For on-line editing, line-at-a-time data entry, and the like, the break is rarely important.
- The tone dialing process does not generate the tones for the asterisk (\*) or the pound sign (#). These are usually not required for normal telephone call dialing, but are used by some internal systems (PABXs) and accessory equipment such as programmable dialers.
- Early versions of the Mark 12 would sometimes get hung up in a software (firmware) loop and refuse to respond to commands. One WAP member reported that the Mark 12 would sometimes refuse to hang up a call (normally accomplished with the +++ ATZ sequence). I have noted that sometimes after a mistaken connection, the modem will refuse to respond. On the ABBS, the first attempt to use a Mark 12 was not successful, because a call would come in too soon after the completion of the previous call, and the modem would then refuse to respond correctly to commands. At the time of writing, Anchor is supposed to be updating the firmware in the ABBS modem to cure this problem.

These various problems are admittedly annoying, though not serious, in my opinion. With the 2 year warranty, I would expect that Anchor will revise the firmware to fix many of these problems. Some of these deficiencies represent cost-saving measures by Anchor; why do you think the Mark 12 can sell for half the price of the Hayes product? And even the Hayes has some cost-saving design choices built into it. For example, the Hayes cannot operate in synchronous mode like a real Bell 212A. For most Apple ][-/// owners, that doesn't matter. But many Lisa, Macintosh, and IBM-PC owners may require synchronous links at some point.

So, I would still say that for the average home/small business computer owner who wants to call BBSs, the Source, CompuServe, etc., or transfer text files to a

remote location, the Anchor Mark 12 has all the essential features at a very attractive price (now, as low as \$225!). If you MUST have RJ 12/13 multiline capability, the "break" feature, the ability to dial a \* or # tone, you'll have to find another product. If you're interested in a modem for a BBS, I suggest you wait until we confirm that the firmware exchange will cure the Mark 12's propensity to sometimes "go South" and refuse to respond to commands or properly answer the telephone line. ☺

Softviews contd. from pg 46

providing useful flexibility. The program allows for direct transfer of text files via modem, a feature I have not yet tried. It provides for a lower case chip option and allows you to use a joy-stick as a "mouse" to move the cursor around. It does not allow for footnotes, constructing an index or writing form letters. Howard W. Sams & Co., 4300 W. 64nd St., Indianapolis, IN 46202. Price \$59.95.

ABSTAT. This CP/M based program is a very easy to use, general purpose, statistics package. The descriptive statistics and analyses which can be obtained are of the type found in most introductory textbooks. The two-way ANOVA (analysis of variance) is somewhat limited in that it only allows one observation per cell. A disconcerting fact is that the ANOVA and regression options require different file organizations. Once you choose a "command", the program provides you with options within that choice. The package handled all of my keyboard errors in stride by responding with an intelligible message. Unlike other programs I have used, it never rudely kicked me out of the package. Help in the form of a clear statement of what could be done next was always available by typing "?". It is possible to back out of commands by typing "!".

Data management and processing capabilities are extensive and easy to use with one bothersome requirement: you must specify the total number of variables in your data set at the time it is created. This number must be large enough to allow for additional variables that may be added at a later time. If this number is exceeded, a larger, empty data set must be created and the old data transferred to it. While this operation is easy, it should not be the responsibility of the user. While your data file can contain up to 64 variables, the total number of data items (variables times records) is 2,000.

The 94 page manual, while somewhat brief, is thorough. You benefit from both a table of contents and an index. It contains a sample of the output for most options and a description of the statistical computations employed. Unfortunately, there are no technical references. Anderson-Bell, Box 191, Canon City, CO., 81212. Phone (303) 275-1661. Price: \$395.00. ☺

# AUTO-REPEAT DIALING WITH AE-PRO AND ANCHOR MARK 12 by George Kinal

Users who have switched from an "internal" modem, or who are new to the use of AE-Pro Macros, often don't know how to set up the program to auto-dial. The AE-Pro manual has a section on the Hayes Smartmodem which describes how to auto-dial using a macro, but there is no mention of repeated dialing. A letter was published in Softalk recently on how to do this, but it was full of typographical errors. Further, the Anchor Mark 12 is different from the Hayes Smartmodem in that it provides some additional response codes (like "BUSY"), and has no DIP switches.

Here, as a public service (?) is how to perform repeated auto-dial (dial until answered) with this hardware/software combination. (Yes, I know, the phone company will probably put out a contract on me if too many people all start clogging up the network with thousands of call attempts to busy numbers like our ABBS...) To use the following Macros, your interface MUST have the carrier detect on at all times. (See my recent WAP article on this subject). The reason for this is simple: the CD must be on so that AE-Pro can receive the result codes from the modem.

The simplest way to repeat auto-dial is with the following macro:

```
**AT DP 9868085'%Y\L1  
(DT for tone dialing instead of the "DP").
```

Remember that the backslash is available on the Apple ][ by prefixing the "/". This simple approach has several deficiencies, especially with pulse dialing. Sometimes the ABBS is down, or a mis-connection occurs. Then, instead of "BUSY", the modem sends back "NO CARRIER." Also, when connection has been made, you must "bail out" using ESCape or RESET to actually get on line.

The following pair of linked conditional macros gets around these problems, and also does log-on as well, once connection has been established:

```
Macro 1: ****AT DP 9868085'~YC2\L1
```

---

## PRINT MASTER DELUXE: A Review

by Bob Oringel

The Print Master Deluxe program, by David Foard of Soft Byte, Box 5701, Forest Park, Dayton OH 45405, is actually two programs on the same unprotected disk. They are the Print Master program and the Check Master program.

The Print Master program has several sub-programs: File Maker, which creates mailing lists; File Changer, which changes those lists; Labels, which prints labels and also the label lists; Check Writer, which simply writes checks; Envelopes, which addresses envelopes; Arranger, which combines list files of up to 400 entries; Print & Save, a label add-on feature; Storage Disk Maker, a disk initialization program for list files; and File Fixer, which the documentation says, "will help, maybe, if a file won't load, or if the program acts funny". The author includes a Sample File for practice, and a State Names file, for State standard two letter abbreviations.

```
Macro 2:  
'~TR1\EH*'G*G*G*G*G*'G**'*****WP9999.PSWD'\EF  
(WP9999.PSWD is your WAP ABBS ID  
number and password, of course)
```

Here is an explanation of these macro elements:

- \*\*\*\* - I find that some delay is necessary for reliability
- AT DP 9868085' - Commands modem to dial.
- ~ - conditional macro symbol (available on Apple ][ by prefixing the = .)
- Y - The "Y" of "BUSY" causes the macro to continue to the \L1 portion.
- C2 - The "C" of "NO CARRIER" OR from "CONNECT" causes a jump to macro number 2.
- ' - an initial carriage return
- T - The "T" of "CONNECT" causes the macro to continue, starting with the \ .
- R1 - The "R" in "NO CARRIER" causes a jump back to macro 1
- \EH - switch to half duplex temporarily.
- \* - delay
- 'G\*G etc - Send a bunch of bells, which the Apple emits as beeps since it is in half duplex. This alerts the operator that a connection has been established. NOTE: enter the BELL character by prefixing, i.e. type CTRL-W, then CTRL-G.
- ' - the carriage return requested by the ABBS
- WP9999.PSWD' -log-on sequence with carriage return
- \EF - Return system to full-duplex mode

The Check Master program is a checkbook reconciliation program which, together with Check Writer, provides checking account control for a family or for a small business.

I have tried most of the program options and they work as promised. This is a good pair of programs but its 10-page, 5 1/2" x 8 1/2", photo-offset of dot matrix documentation leaves a great deal to be desired. The author is obviously a fine programmer, but he should invest in a spelling checker program, having written the documentation. There are spelling errors on the disk as well. The purchaser buys the documentation as well as the disk, and that purchaser is entitled to standard American English, in a clearly written, clearly printed text, devoid of abbreviations, and including an index.

# A USER'S GUIDE TO GENERAL DISK UTILITIES

by Bob Trexler

The February 1984 issue of the WAP Journal contains a review of the programs on WAP Disk 37: Utilities IX. Among these is a program called, "General Disk Utilities." I have used this program, (internally documented as "DISK UTILITY PROGRAM, by J. Garbor, 1979") for several years as my "el cheapo" disk snooper and repair aid. Because I have never seen any documentation of this program, I have had to experiment with it to find out how it works. While I do not pretend to know it as well as J. Garbor does, I know how to use it and I thought that some words from me might encourage others to try to use it too.

Now onward towards using GDU, as I will call it for short! The program is written in Integer Basic, so you will have to get Integer up before it will run for you. Let me first summarize what GDU will give you:

1. The contents of a given sector of a given track.
2. A graphic display of used and free sectors.
3. A directory which reveals the statistics on a specified file (track/sector list, address, length, type, and so forth.)
4. A nibble editor that permits entering data, then writing directly to a specific sector on a given track.

I have to admit that once I became familiar with the program, I was thoroughly impressed with its usefulness.

## USES

I use GDU mostly when I have trouble with a disk. Sometimes a file (usually a text file) will not load into my word processor. If I really want that file, I break out GDU and look for the trouble. Usually the trouble has to do with being unable to read one of the sectors shown in the track/sector (t/s) list. Something has happened that scrambles the pointers so that the whole of the file cannot be loaded, and an I/O message is returned to me. That's when I know it is time to try GDU.

First, I have to find the file. If I know the exact FILENAME, I will use the directory to find the location of the track/sector list. (If I don't know the exact name but guess at it, GDU will tell me that the file can't be found, if I have guessed wrong.) So the safest bet is to examine the catalog. This is found in track \$11, starting in sector \$0F. I switch to mode 2 to get the whole sector displayed, and also switch to format "A" so that I can read the titles of the files.

In case you have not actually seen sector by sector screen dumps of your files (programs) in the Hex or ASCII formats, you are in for somewhat of a surprise. In the hexadecimal notation, they seem to be pretty much massed hex digits. In ASCII format, you will find then to be a bewildering mixture of flashing, normal and inverse characters and symbols that seem to have very little to do with what you know is in the file. And, oh yes, there is plain old English text mixed in as well. In any event, for you to understand what DOS does, I recommend your getting a copy of the excellent manual called Beneath Apple DOS by Don Worth and Pieter Lechner, available from Quality Software.

This manual is invaluable towards an understanding of the contents and formatting of DOS files. (I will have some occasion to refer to certain illustrations in it, but that's not the reason to buy it.)

In Beneath Apple DOS, figure 4.4 shows the layout of the typical catalog sector. The important thing to know about this layout for our present purpose is that the first three bytes before the file's name are the track and sector where the file's track/sector list may be found, and the file-type code. For example,

```
BYTE==>00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F
          00 00 11 0E 00 00 00 00 00 00 00 12 0F 01 C8 C5 CC
          10 CC (etc...)
          20
```

## First 17 Bytes of a Catalog Sector

The contents of the byte at 0A is 12. This is the track on which this file's track/sector list will be found. The next byte, at 0B, is 0F. This is the sector. At byte 0C is 02, which is the file-type code for an "unlocked Applesoft" program. At 0D, 0E and 0F are the codes for the first three alpha characters in the FILENAME: H, E, L.

Under the conditions I described, in alpha mode, I would not see the contents of 0A, 0B or 0C. Instead, I would see something else, and frankly I have not yet puzzled out what the symbols mean (they could be flashing semicolons or brackets or other things). However, I could be able to read the plain English text title of the file, and therefore be able to locate it. After I locate it, by switching back to the hex mode I can then read the location of the track/sector list. Next, I go to the track/sector list to see what the trouble can be, if any, with the list itself. This is usually determined by my starting at the beginning of the list and attempting to examine all sectors sequentially. The keys "N" and "L" will let me scroll forward or backward. However, to get to a specific sector, I must ask for it. (You know that DOS doesn't always put content in successive sectors.) You will recall that the T/S list contains the track number and sector number of where DOS has stored the contents of the file. This list is often sequential, but it doesn't have to be. Once I have found the sector that cannot be loaded (even GDU will not be able to load it) I can then copy the other sectors to another disk and salvage most of the file.

## UNDELETING A DELETED FILE

You may recall that when you delete a file, the contents of the file remain intact until the disk drive attempts a SAVE following the deletion. In this case, even though you know the FILENAME, typing it does no good because DOS has overwritten the track number of the T/S list in the catalog. What DOS does is to write "FF" into the location of the track data for the file's track/sector list, making it impossible for the Apple to find the track/sector list, and therefore the file. DOS allocates 30 characters for a FILENAME. If your FILENAME is shorter than that, DOS pads the remainder with "A0's". The last 2 bytes contain the length of the file in LO byte HI byte order. If your file was 2 sectors long, these bytes would be 02 00. A file 2 sectors long that was not deleted would show at its end "...A0 A0 02 00". However, a deleted file

contd.

would show "A0 A0 TT 02 00". The "TT" represents the track of the track/sector list, which DOS has written here when it changed the TT to FF at the front end of the catalog entry. It is possible to undelete the file through simply changing the "FF" back to the track number of the track/sector list, and changing the TT at the end of the catalog entry back to A0. Once you have done that the file's name reappears in the catalog and the program is available again. (Of course, there are "undelete" or file recovery programs that are easier to use, but I think it's nice to know how to do this myself.)

## COMMANDS

After running the program, you will be directed to press the space bar. Unless you wish to examine the disk that contains the GDU program, this is the time to replace that disk with the one you wish to examine. (This is also the time to put in a disk that you don't care very much about, so that you can play with it freely. You can do no harm until you deliberately enable the write capability, however.)

Having inserted your scratch disk, press the space bar. The drive runs and the program reads and displays the contents of track 00, sector 00 and gives you the hex number of the disk if you have initialized it with a volume number; otherwise you will see the classic \$FE, hex for 254. The default condition has the following:

Mode 1, Format H, Input and Output radix is H, Initial track and sector are 00,0. Help available by pressing "H".

After pressing H the following commands are displayed to you:

```
R(ead)   W(rite)  E(dit)   F(ormat)  M(ode)
I(nrad) O(utrad) N(ext)  L(ast)   T(est)
D(irectory) X(exit)
```

The following additional commands will not fit in the space provided for the help commands but are also active:

P(arams) C(hange)

In addition, when you are in edit, the following commands are available:

```
I/J/K/M (Move cursor) U(home) O(set pointer)
R(otate ASCII)
F(ormat) D(eposit)
C(designate slot+ enable or disable write)
```

Following is an annotated description of the user commands.

R: You will be asked the track number. Enter it and press return. Then you may either specify a sector number or hit return. Return accepts the present sector; then you may specify the byte at which the display is to start. After that last return, the drive will start and 128 bytes of the requested sector will be displayed in hex notation (mode 1).

W: You will be asked the track number, and sector start numbers. If you have not enabled the write, you will receive an error message, and there will be no writing to the disk.

E: Edit. In edit the cursor becomes a pointer (>) which you may move using the I/J/K/M commands.

U sends the pointer to the upper left (home) position.

O lets you enter the desired location at which to put the pointer, rather than move it there manually.

R permits selecting among three different ASCII formats; normal, inverse, or flashing.

D converts the selected byte into two question marks ?? and permits depositing the desired HEX value in that location.

C lets you select the slot and drive which enables or disables (the default) the write capability.

W activates the drive and writes the contents of a given sector to a track and sector that you specified.

F: Format. Select either hex (H) or alpha (A).

M: Mode. View either 128 bytes (1) or 256 bytes (2).

I: Inrad. Like its sibling Outrad, Inrad lets you specify the track/sector data in either hex or decimal values.

O: Outrad. (See Inrad.)

N: Next. N reads the next sector to the screen.

L: Last. L reads the last sector to the screen.

D: Directory. This is a mode in itself.

F displays the descriptive data for the FILENAME you specify. Includes track/sector list location, length, address, filetype.

D deletes the specified FILENAME.

S provides a map of the in-use and free sectors on the disk.

x returns to main program.

T: Test. This will fill a given sector with a test pattern.

X: eXit. Returns you to Integer Basic.

P: Params. This command displays the present status.

C: Change. This command lets you specify the slot and drive you wish to activate, and to enable a write, or disable it. Default is disabled.

## CONCLUSION

I have found General Disk Utility to be a practical tool for aiding my recovery of otherwise lost files. But, more importantly, I have found it to be a means by which I could learn more about DOS and its tricks. Together with Beneath Apple DOS, the pair makes a good combination. ~~WHILE IT IS TRUE~~, this program cannot match the power of commercial nibble copiers, I think it deserves more attention, and I believe anyone can profit from some experience with it. &



# TURBO PASCAL: JRT's Promise

## Fulfilled

### by Bill Seeley

A year or so ago, quite a stir was raised in CP/M circles when JRT Pascal was announced for \$29. Many people saw this as an opportunity to learn about Pascal without changing operating systems, and without paying an arm and a leg for one of the 'professional' CP/M based Pascals (\$400 and up). JRT has since gone out of business, and rumor has it that its late, lamented version of Pascal, while worth \$29, was not without its price in quirkiness and unreliability. With the recent announcement by Borland International of 'Turbo Pascal' for \$49.95, it looks as if someone has gone and done it again. The same questions which were asked about JRT Pascal will be asked again of Turbo Pascal: at this bargain basement price, what are the limitations... what's the catch? After working with Turbo Pascal for about a month, I am happy to report that the limitations are few and I haven't as yet found any 'gotchas'. I can only wonder how Borland can market such an excellent product for so little. Let me count the superlatives!

The feature which initially sold me (which I found out not from the ads but from a review in 'Microsystems') is Turbo's provision of a full-screen editor which completely supports Word Star's editing commands. I am teaching my son Pascal, and since he already knows Word Star and the rudiments of CP/M, I figured that this product would eliminate the need for him to learn another editor and set of operating systems commands (e.g. the UCSD p-System which we also own), and would allow him to concentrate on the Pascal language itself. In fact the Turbo editor works almost exactly like Word Star (but alas provides no help!), which by itself could justify the purchase of this product. More importantly, the editor is integrated quite nicely with the compiler and the rest of the development system. To start the system, you type 'turbo' and in a few seconds you are given the option of loading an error message file (a space saving feature) and are then presented with Turbo's main menu. The menu commands are all accessible with single keystrokes and the switching between functions is instantaneous, since the entire development system resides in memory.

After declaring a workfile with the 'W' command, you enter the editor by simply pressing 'E'. When you are ready to compile the program, you exit the editor with 'KD' and simply type 'C' ('KD does not save the file to disk as in Word Star - the 'S' command on the main menu is used for this). If there are any syntax errors, the compiler stops at the first one and displays an error message (or error number if you opted not to load the messages at the beginning). If you press the ESC key at this point you will be back in the editor immediately, with the cursor positioned to the offending statement. When all of the syntax errors have been corrected you can run the program by pressing 'R'. If any changes to the source code have been made since the last compile, Turbo will re-compile it before commencing execution. If a run-time error (I/O error, range error, etc.) is encountered, Turbo will again allow you to re-enter the editor to correct the statement in error.

Turbo Pascal has additional features which facilitate the compilation of programs too large to fit in the workspace. The source program can be broken down into segments (e.g. procedures, standard record layouts, etc.) which can be stored in separate disk files and then included with compiler directives embedded in the main program. During compilation of a main program,

if Turbo finds an error in an include file, it will enable you to immediately enter the editor to make corrections. When you exit the editor, Turbo will automatically re-save the include file on disk and then re-load the main program for the next compilation attempt.

Turbo provides 3 forms of output for the native mode object code it generates. The first is output directly to memory, where the code can be immediately executed with the 'R' command. This is suitable for the development of relatively small programs and procedures to be included in main programs. Secondly, object code may be output to a CP/M COM file with automatic linkage of the Turbo run-time library. If a run-time error occurs in a COM file being run from CP/M, Turbo informs you of the program counter value at the time of the abort. If you re-enter the menu system, and specify the main program and the above P register value, Turbo will automatically load the main program followed by any necessary include files and then place you in the editor positioned to the statement which caused the problem. Finally, object code may be output to a chain file. Chain files (designated with extension .CHN) do not include the run-time library and may be invoked at execution time via the Turbo command CHAIN. CHAIN is analogous to a GOTO and neither parameter passing nor subsequent return to the caller is possible. Information may be passed to chained modules via global variables declared at the beginning of the caller and all callees (preferably via common include files). Regrettably, Turbo makes no provision for the development or linkage of object libraries. Parameterized external procedures declared at absolute address locations are supported, but Turbo provides no way to link them into the object file. The only way this feature would be useable is via a machine language interface to resolve parameter addresses, and the use of DDT to link them - not a very practical solution.

Perhaps Turbo Pascal's most impressive feature is its speed. The ads cite compilation and link speeds which are approximately 1/100 that of the IBM Pascal compiler and MT Microsystems's Pascal MT+, both of which also generate native code. A similar claim of an up to 3 to 1 advantage in execution speed is made for a CPU intensive benchmark program. How accurate these claims are, I cannot say. One example I can offer are the statistics for the most recent compilation of a program I am currently developing on an Apple //e equipped with a 6 mhz Appli-Card. The single pass Turbo compiler translated 600 lines of source code, consisting of a main program and 7 disk-based include files, to a CP/M COM file containing native Z80 machine code in 18 seconds. When compiling and running small programs in memory, Turbo takes on more of the flavor of an interpreter, with the execution phase seeming to commence almost instantaneously. Perhaps it is merely Turbo's well thought out, easy to use development system which makes the compiler appear so fast!

Thus far I have said little about Borland's implementation of the Pascal language itself. Turbo Pascal claims to adhere very closely to Standard Pascal as defined by Jensen and Wirth in their 'User Manual and Report'. Additionally it provides numerous non-standard extensions which give CP/M users virtually complete control of all CP/M functions as well as direct access to machine level functions. Among these

contd.

are the ability to make BIOS calls, to manipulate absolute address locations, and to directly access I/O ports. Among Turbo's more intriguing extensions is the INLINE statement which allows the embedding of hex machine code directly in a Pascal source program. Of course the use of all of these features renders a program less transportable to other systems. On the subject of transportability, Borland International provides native code versions of Turbo Pascal for the following operating systems: IBM PC DOS, Apple CP/M, MS DOS, CP/M 86, CCP/M 86, and CP/M 80. Since Borland's documentation carefully delineates operating system dependent extensions, programmers can easily avoid these if they wish to develop source code which is transportable to the above environments. One potential problem is the fact that the CP/M 80 version is written in and generates Z80 machine code, which will not run on all CP/M systems, e.g. 8 bit Heath/Zenith and CompuPro systems using the 8080/8085 CPU. The decision to use Z80 machine code was apparently a compromise based on the Z80's more powerful instruction set and the fact that the majority of CP/M implementations (including Apples) use the Z80 CPU.

Apple users may be more concerned about how closely Turbo Pascal resembles Apple's implementation of the UCSD p-System version. Not being a connoisseur of UCSD Pascal I can make only a few observations on this subject. Turbo Pascal of course does not provide high-level language support for the Apple's graphics functions a la turtlegraphics. However, since Turbo does permit 'peeking' and 'poking', routines could be developed for directly manipulating the Apple's graphics pages. Other differences are the elimination of the GET and PUT procedures and the provision of extensions to READ, READLN, WRITE, and WRITELN which enable the latter procedures to handle both text and fixed length files. To test the difficulty one might have in converting a UCSD Pascal program to Turbo Pascal, my wife keyed in a 6 page, non-graphics game program (HANGMAN) from Herget and Kalash's 'Apple Pascal Games' using the Turbo editor. It took us approximately 15 minutes to make the conversions necessary to get it running properly under Turbo Pascal. The major changes were the substitution of READs and WRITEs for GETs, PUTs and file variable assignments, plus a minor rewrite of the randomization routine.

One final kudo: the Reference Manual. In addition to all of the above, Borland throws in a 254 page, bound manual providing complete documentation of the editor, the development system, and the compiler directives, plus a thorough description of the Pascal language (beginners will still require a tutorial however). The manual is well typeset, clearly written and concise, and contains numerous examples to illustrate the various language features. It is extensively indexed, and provides over a dozen appendices, which include the language syntax in Backus-Naur Form, as well as lists of all compiler, run-time, and I/O error messages. The description of the terminal installation procedure which must be performed before the full-screen editor can be used is quite straightforward. Turbo Pascal provides support for over 30 standard terminal types, plus the capability for custom terminal definition if necessary. While Turbo Pascal did not directly support any of the Apple's 80 column cards at the time I installed it, many 80 column cards can emulate common terminal types which are in turn recognized by Turbo. In the worst case, the user would have to answer a series of prompts from the installation program defining the particular 80 column card's escape sequences for things like cursor positioning, clearing the screen, etc. With the Appli-Card I got the best results by first selecting a Televideo 920 driver with the CP/M INSTALL program, and by then specifying the latter terminal type to Turbo's installation program.

In short this is a highly professional piece of software offered at an incredibly low price (their introductory package even includes the source code for a limited, but functional spreadsheet which they call MICROCALC!) How they're able to market a compiler of this quality, this cheaply I don't know. How much longer they'll continue this giveaway is anybody's guess.

## THE FORTH COLUMN

by Kevin Nealon

On April 7th the Forth SIG had its second meeting with 16 people in attendance, which just goes to show that you can't keep a good computer language down. The main topic of the meeting was the installation of the Forth full screen editor (WAP Vol. 701) on the club's Fig-Forth78 (Vol. 105). It seems there was an error in the original instructions Bob Platt gave in his fine article on this subject in the May 1983 WAP Journal (an errata listing was given in the June 1983 Journal). To repeat that errata, it should be 73 load, not 21 load, after you have put Vol. 700 into drive 1. For those of you you have a recent version of Vol. 701, there seems to be a defect in this volume. Hopefully, in a couple of month's we will have a new version in the Disketeria, which will have all the modifications needed to install it, and will be able to accomodate Videx and //e 80-column cards. When this is accomplished we will describe in this column how to install the new editor, its use, and a few other facts you'll want to know.

In the meantime, if you are interested in learning about this fascinating language there are two texts that the SIG recommends. Starting Forth by Leo Brodie is an excellent beginning text, a good mix of humor and work problems. The other is the two-volume set Forth Fundamentals by C. McCabe, in which he explains some of the advanced areas in a more complete way and gives many example programs. The Forth SIG plans to use both of these texts during the tutorial meetings.

The next Forth SIG meeting will be held at the office on May 19 at 1:00 PM. Planned topics are: how the new screen editor is coming along, just what is a stack anyway, and am I insulting an ethnic group when I say Reversed Polish Notation.

# STUFFING COM-WARE ][

by J. T. DeMay Jr.

Usually COM-WARE ][ (the communications package that comes with the Novation Apple-Cat) is limited to storing 26 phone numbers in the PHONE LIST file, one for every letter of the alphabet. This may seem enough to most people, but there will always be someone who wants to push the machine to its limits and beyond. For those of us who are not able to live with the 26 number limit, here is a procedure for adding many times the original 26 numbers. The general idea is to create several files of phone numbers of the same format as the PHONE LIST file; then whenever a new set of phone numbers is wanted, just use the <D> DOS COMMAND from the main menu to BLOAD the new file over the old file. There are a few potential problems using this procedure, but with a little caution they can be avoided.

Decide which phone numbers you would like in each phone list file and then choose a name for each file. Pick something that will remind you of the actual numbers stored in the file. For this example, let's assume there are two groups of phone numbers you would like to use with COM-WARE ][; FRIENDS and BUSINESS and each has 20 entries. Here is the procedure:

First make a copy of your COM-WARE ][ disk and put the original in a safe place. Then BOOT the copy of COM-WARE ][. Choose <D> from the main menu to enter a DOS COMMAND. COM-WARE ][ will respond with DOS COMMAND: You should then enter RENAME PHONE LIST,OLD PHONE LIST and press <return>. This will save a copy of the PHONE LIST file that COM-WARE ][ uses to store the phone numbers.

Choose <A> to select AUTO-DIAL, then choose <L> to DIAL NUMBER FROM LIST. We're not going to call anyone; just modify the PHONE LIST file in memory by using the submenu under the <L> command. Select the <#> DELETE NUMBER FROM LIST option and remove all the numbers you don't want included in this file. Let's do the FRIENDS file first. After you have deleted all the numbers you don't want in the FRIENDS file, enter any new numbers desired. Up to 26 numbers can be saved in each file.

Choose the <\$> SAVE LIST TO DISK option. This will save the file in memory as PHONE LIST. We must now RENAME the file FRIENDS. This is easily done by pressing <ESC> to get back to the main menu, then choose <D> to enter a DOS command. COM-WARE ][ will respond with DOS COMMAND: Enter RENAME PHONE LIST, FRIENDS and press return. Now whenever the FRIENDS phone numbers are desired, just select <D> from the main menu, and then enter BLOAD FRIENDS,A\$2A4A and press <return>. Dial the numbers as you normally would.

To build the BUSINESS phone list, get back to the DIAL NUMBER FROM LIST menu by pressing <A> then <L>. Use the <#> option to delete all the numbers from the PHONE LIST file in memory. Press the <ESC> key to get to the main menu again. Choose the <D> option again. When COM-WARE ][ responds with DOS COMMAND: enter BLOAD OLD PHONE LIST,A\$2A4A and press <return>. This will LOAD the original phone numbers file. Now delete the phone numbers you saved in the FRIENDS file using the same procedure we used earlier. Enter the new numbers you would like to keep in the BUSINESS file. When finished, choose the <\$> option to SAVE LIST TO DISK. Once the file has been saved, it must be renamed BUSINESS. Do this with the <D> command from the main menu.

It is important to remember that every time COM-WARE ][ saves a group of numbers using the <\$> command, it saves it as PHONE LIST. If we are going to fool COM-WARE ][ into thinking there are only 26 numbers available, we must RENAME our files each time they are saved. It is also necessary to have a file called PHONE LIST on the COM-WARE ][ disk because it is the file which COM-WARE ][ looks for when it is BOOTed.

You may add as many phone number files as you like. The only limit is the available space on the disk. Once the COM-WARE ][ disk has been BOOTed, it can safely be removed from the drive and replaced with another one. If you have two or more drives, the <D> DOS COMMAND works just like DOS 3.3. Just add a ,D2 to reference the disk in drive two. COM-WARE ][ will read from or write to the disk last accessed. This should allow ample room for as many phone numbers as anyone needs.

One last step is required. We must create a file called PHONE LIST for COM-WARE ][ to use when BOOTing. I would suggest deleting all the numbers in the PHONE LIST file in memory, and entering the names of the files we have stored the actual phone numbers in; in this case FRIENDS and BUSINESS. This will serve as a reminder of which names you used to save the numbers. If you have more than a few files, this could be very helpful. When you are satisfied with these names, choose the <\$> option and save the PHONE LIST file to the disk.

The next time your COM-WARE ][ disk is BOOTed, the names of the files which have the phone numbers will be listed under the <L> submenu. Any of these files can be accessed by BLOADing them from the main menu using the <D> option with the address parameter ,A\$2A4A. (This is the address to use with COM-WARE ][ REVISION 5.0-3. and a 64K APPLE ][e, it may be different for other configurations).

If you would like to use PHONE LIST PRINTER to keep track of all those phone numbers, the following changes must be made:

a. If you already have a copy of PHONE LIST PRINTER, LOAD it into memory. It is available from the DOWNLOAD section of the WAP ABBS, or from the February WAP Journal.

b. Enter the following lines:

```
275 PRINT:PRINT "ENTER PHONE LIST FILE
NAME, OR PRESS [RETURN] TO USE
<PHONE LIST>: ":PRINT : PRINT
";:FOR I=1TO20:PRINT
CHR$(8);:NEXT:INPUT"";NA$
```

```
277 IF NA$="" THEN NA$="PHONE LIST"
```

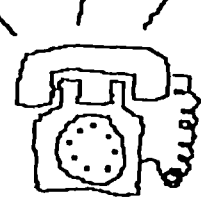
```
280 PRINT D$;"BLOAD";NA$;" ,A";AD
```

```
320 IF P$="P" THEN POKE 36,4:PRINT
"PHONE #";:POKE 36,16:PRINT
"PASSWORD";:POKE 36,44:PRINT
"SYSTEM NAME";:POKE 36,59:PRINT
"I/O FILE NAME":FOR L=1TO80:
PRINT "-";:NEXT
```

c. Save PHONE LIST PRINTER.1

contd. on pg 57

# THE HOTLINE CALLS BACK... APPLEWRITER //



(Contributing hotliners: Doug Daje, Dianne Lorenz, Lee Raesly, and Tom Warrick.)

Members of the Apple Writer hotline are banding together to return your calls en masse. Each month this column will address the most often asked questions in a specific hotline area. The Apple Writer // group is inaugurating this column with answers to our top ten questions. You will notice how many questions relate to printing problems. We all agreed that the majority of our calls relate to printers; therefore, with your help we are starting a library disk of glossaries of print commands to be used with Apple Writer. If you have developed a glossary for your printer, please send it to the Apple Pi office, c/o Apple Writer Library Disk, on disk or hard copy.

Q: When I use the backslash to underline as the Apple Writer manual suggests, I get little dashes between the l e t t e r s. How come?

A: The backslash enables underlining on a letter quality printer only. To underline on a dot matrix printer, you will need to embed the printer's command to begin and end underlining using Ctrl-V. Most printers use an ESC followed by a letter or number. For example, the Prowriter, NEC 8023, and Apple Dot Matrix begin underlining with ESC X and end with ESC Y. This information can be found in your printer manual.

Q: How can I find and replace returns in my text? Nothing happens when I use the usual delimiter, /, and the right bracket, ], to indicate a return.

A: On page 65 of the Apple Writer //e manual is an interesting chart that will answer your question. There is no way to represent a RETURN using the normal delimiter, /. However, if you select another delimiter, the ! for example, follow the line across to the Carriage Return column and you will see that you can represent a RETURN with the pound sign, #. To remove unwanted returns from your text, respond to the Find command with !#!, i.e., replace every return with nothing. You will have to reinsert the returns that marked the ends of paragraphs, as this command removes all returns.

Q: When using the backslash to underline on a letter quality printer, the / symbol creates a space. If a period follows the underlined word, it doesn't look too good floating out in space.

A: Most printers have a command to backspace. Several printers use Ctrl-H. Embed this command after the closing underline command and before the period to close the unwanted space. You won't have this problem with a dot matrix printer because the command to underline does not create a blank space.

Q: I'm trying to send printer controls to my dot matrix printer, but it just isn't listening. Don't you use Ctrl-V to embed the CHR\$ number that you need?

A: No. PRINT CHR\$(number) can be a printer control string, but is sent from BASIC, not Apple Writer. The letter that corresponds to the ASCII code is what you want to send to your printer, usually

prefaced by ESC, e.g., if the BASIC command for compressed print is PRINT CHR\$(81), then you would send an ESC Q from Apple Writer. The ASC(Q) = 81. Most printer manuals have an ASCII code chart in the appendix.

Q: Why do footnotes and page numbers not appear on the final page?

A: Whenever you use page numbering or footnotes, the last item in your text must be .FF, which causes a form feed or page throw and forces the errant footnotes and page number to print.

Q: I need to use the backslash, \, in my text, but it causes everything to be underlined that follows it.

A: The \ symbol is defined as the underline token in the print menu. If you need to use this character in your text, (for instance to answer this question), change the underline token in the same way you change any other parameter in the print menu, by typing UT and whatever symbol you want to represent underlining or just UT and return to eliminate the underline symbol temporarily. (Note: On the NEC 8023, the \ is used to represent one of the superscripted numbers in the Greek alphabet. It is impossible to put this number into a print glossary unless you redefine the underline token.)

Q: In using the vertical format commands, TM, BM, PL, and PI, I cannot get the number of printed lines per page that I want. What's the secret?

A: Contrary to the way it sounds, that you would specify the number of printed lines to be the number of lines of print per page, you actually add up the top margin, bottom margin, header, footer, and footnotes and put that number in the print menu for # of printed lines. These must all add up to less than the page interval which is usually set for 66. Any difference between printed lines and page interval will be made up with blank lines.

Q: Is there a way to get more than 128 characters per footnote in Apple Writer?

A: Yes. When you are approaching 128 characters, close the footnote, add a space and open another footnote. You can go on with your footnote in this way up to 1024 characters...probably enough for anyone. &

# USING WPL TO CUSTOMIZE

## APPLEWRITER //E

by Dianne Lorenz



WPL, or Word Processing Language, is a programming language utility that is part of Apple Writer //E. With WPL you can write little programs that do nifty things for you when you are using Apple Writer. Most people are so happy when they finally get comfortable with Apple Writer that they don't look beyond it to the accompanying WPL manual. And if they do, one look at a WPL program is enough to convince the novice that it's something they can live without. Your Apple Writer disk has several programs written in WPL on it already. These programs are accessed by the print/program command, Ctrl-P. WPL is the program part of this command. Explanations of the programs you already have and how to run them can be found in the Apple Writer manual. The following programs illustrate some of the simple functions that WPL can perform. More sophisticated programs will be illustrated in future issues. If you are already programming in WPL and have some helpful utilities, please send them to this column for publication.

### MULTIPRINT

A function noticeably absent from the Apple Writer print menu is the option to print out additional copies of the text in memory. Type in the program below exactly as is and save it under the name of Multiprint. When you need multiple copies, load the document into memory and type Ctrl-P, DO MULTIPRINT. An explanation of the program follows the listing.

```
START PIN Please enter the number of copies = $a
      PSX
PRINT PNP
      PSX + 1
      PAS(x) = $b
      PCS /$a/$b/
      PGO QUIT
      PGO PRINT
QUIT PQT
```

All commands that are part of the WPL language are preceded by a P. You will recognize that all commands that are embedded or are reached via the print/program menu are WPL commands. Therefore, to print in WPL, the command is PNP. WPL follows the same format and execution logic as assembly language. Here's what's happening in the above program, line by line.

LINE #1: START is a label that is optional. Labels are mandatory if you want to get back to a line for a loop or repeat. The labels PRINT and QUIT are not optional as you will see as we progress. PIN is an input statement. Whatever is answered to the question, "Please enter the number of copies?", will be stored in the variable \$a. Yes, \$a, not a\$ as in BASIC...just to keep you on your toes. There are only 8 string WPL variables, lower case \$a - \$d and upper case \$A - \$D. Lower and upper case x, y, and z are the six numeric variables.

LINE #2: PSX 0 is equivalent to set X to 0 or LET X = 0.

LINE #3: PRINT is a label. PNP tells the printer to print whatever is in memory at this time; hopefully it is the document of which you want multiple copies.

LINE #4: PSX + 1 is equivalent to LET X = X + 1.

LINE #5: PAS(X)=\$b is equivalent to LET \$b = X. This

statement assigns the value of the numeric variable X to the string \$b so that the following comparison can be made.

LINE #6: PCS /\$a/\$b/ compares the value of the two strings \$a, the original input string of the number of desired copies and \$b, the value of the numeric variable X which is the number of times the program has executed or printed out the contents of memory. Now a familiar logic branch occurs. If the comparison evaluates to true, i.e., \$a = \$b then the next statement is executed just like an If/Then statement. If the comparison evaluates to false, then the next statement is ignored and the one following it is executed.

LINE #7: PGO QUIT sends the program to the statement labeled QUIT if the comparison statement evaluates to true.

LINE #8: PGO PRINT sends the program to the statement labeled PRINT. It will be executed each time the comparison statement in LINE #6 evaluates to false. This forms the loop that allows the program to continue until it has printed out your text in memory the number of times that you requested.

LINE #9: QUIT is a label. PQT ends the program.

Look for an interactive merge program that will allow you to merge a base letter or text with names and addresses in the next WPL column....

### SAVE YOUR APPLE II/II+ with a MACRO FUNCTION CARD ADDS CAPABILITIES, KEEPS COMPATABILITY

#### GIVES STANDARD APPLE COMPUTERS 3 KEYBOARDS!

1. Standard keyboard
2. 70 FIXED MACRO function keys
3. 120+ USER DEFINABLE function keys

USER DEFINED KEYS MAY BE STORED AS DISK FILES  
OVER 700 CHARACTERS MAY BE USER DEFINED  
APPLESOFT AND CP/M FIXED MACRO KEYS PROVIDED  
PROVIDES YOU WITH A 128 KEY TYPE-AHEAD BUFFER  
GREAT FOR CUSTOMIZING KEYBOARDS TO SOFTWARE  
COMES WITH MACRO CARD, SOFTWARE AND MANUAL  
MACRO KEYS ARE TRANSPARENT TO SOFTWARE  
EASY TO INSTALL. EASY TO USE. POWERFUL!

Best of all - the price: ONLY \$75!+  
Complete with 90 day warranty.

SYSTEMS SERVICES CO.  
1125 SHANNON PLACE  
HERNDON VA 22070  
(703) 435-3896

#### DEALERS WANTED

\* \$60 to qualified handicapped persons.  
Virginia residents add 4% sales tax.  
Add \$1.50 for shipping and handling.

# MAGIC WINDOW II AND APPLE- WRITER //e: A Teacher's Review

by Anne F. Moeller

Magic Window II and Apple Writer //e word processing programs approach the job of computer-assisted typing from two quite different directions. They both have their strong points, and each has its fans.

Apple Writer //e is designed for use with Apple Computers only. If you try to boot it up on the Franklin ACE, you will receive a message to this effect. Its main drawback is speed. A good typist would be very frustrated with Apple Writer's limitation of 65 words per minute. Typing above this speed, the program drops characters. That, alone, precludes its use by serious word processors.

Apple Writer //e has the usual wraparound feature one would expect from a word processor. It is easy to delete words, a bit awkward to strike over. Inserting material is very easy - just place the cursor where you want the new material and type. All lines to the right will make way for the new material. There is an "express" cursor feature that can be very handy for getting around the editor. Just hold down the closed apple while pressing the arrow. Apple Writer allows for easy and quick pasting and glueing of words and paragraphs.

The main problems with Apple Writer arise from the fact that formatting is done separately from data entry. The typist enters data into the editor without regard for spacing, margins and paragraph indentations. All formatting is done through the printer setup menu or with embedded commands. This means that the copy produced on the printer can look quite different from the way it looks in the editor. The operator can preview the printout by printing to the screen, however.

Even so, things get frustrating with Apple Writer when the typist wants to do anything more than type simple paragraphs. Any columns must be typed correctly the first time. Additions or deletions rearrange the spacing, but this will not be apparent to the typist until the material is printed. Then, it is a game of find-the-invisible-problem. Arranging two columns with one left-justified and another, opposite, right-justified, can be very tricky.

Apple Writer comes with a very impressive feature for the producer of term papers - it will automatically print the footnotes at the bottom of the correct page - theoretically. The wraparound feature is inactive in the footnote mode, and no more than about 60 strokes can be included in a footnote line. There is no way other than counting strokes to tell when to return, since the editor has no corresponding margins. Even then, the footnote feature may not work correctly, try as you may to appease the footnote spirits in Apple Writer.

Every footnote requires a superscript. In Apple Writer any special feature calls for a very complex string of commands unless you are using an Apple printer. For example, with an Epson FX-80 printer, the following commands are required to print a superscripted number one: <Ctrl>V, ESCape, S, 0, 1, <Ctrl>V, <Ctrl>V, ESCape, T, <Ctrl>V.

Printer commands are much simpler with Magic Window II. This is because the program adapts itself to any equipment the user may have. A "driver diskette" configures the program when it is booted up. The driver

diskette holds the configuration information and the files created by the user. Making a driver diskette is very simple - just answer three questions about what kind of equipment you use. If necessary, the disk will be initialized at the same time.

With Magic Window, what you see is what you get. Margins and spacing are set before typing begins - just like with a typewriter. The user can see how the material will look as it is being entered. Columns can be edited with ease. It is easy to arrange material in complex displays, and change it without concern that it will print as desired.

In order to add material with Magic Window, the user must make room. Press <Ctrl>T to split the line where material is to be added, and type. Space for the added lines will be made as typing continues. Then, lines can be closed up by using <Ctrl>G (for glue). Of course, Magic Window has a wraparound feature and "cut and paste".

Magic Window seems to have no limitation on speed of typing. If the user has a good keyboard, (Apple //e and Franklin ACE are both fine), Magic Window is just as efficient to type with as an IBM typewriter - even at speeds close to 100 words per minute!

Magic Window goes under disguise as FRANKLIN ACE-WRITER. This is the version I use. By comparing the manuals for both, I determined that Magic Window and ACEWRITER are the same program. They are both published by Artsci, Inc., and written by Bill Depew. On this basis, I approved an order for Magic Window II word processors for use in the Computer Applications class that I teach. The programs are, indeed, very similar. The files are completely compatible (and stored with a .MW following the file name in both cases). Two commands are different: To leave the editor with ACEWRITER, press ESCape; with Magic Window press <Ctrl>A and @. The command to insert a line is different, as well. Otherwise, they are basically the same program.

Both Magic Window and Apple Writer have features that allow the user to select excerpts from stored material to load into memory. With Magic Window, this must be planned for, since the material must have been stored as an "unformatted file". With Apple Writer, no such planning is required; however, the careless user will find that there is no automatic memory clearing when a new file (or portion thereof) is added to memory. Magic Window will clear memory between loading formatted files (with a warning and your approval, of course).

Apple Writer provides a programmed form letter feature that can use Quickfile records for addresses. There are some other automatic features available, as well, but they are designed for business applications seldom required by the personal computer user.

For the serious typist Magic Window seems to be a better bet than Apple Writer. Being able to type as fast as I like and see how material will appear before it is printed are features I consider requirements. Apple Writer can be a useful program for the less proficient typist, however, especially if it is used mainly for straight typing - no fancy stuff. The fancy features in Apple Writer seem designed more to please a programmer than to provide tools for the

contd.

office, because the advantages of the special features are outweighed by the disadvantages of the speed limitations. (In other words, if the typist is good enough to work as a word processor in an office, the speed limitations of Apple Writer //e will be more frustrating than the special features are useful.) That is why I feel these two programs approach word processing from two quite different directions. Apple Writer is a programmer's program; Magic Window is a typist's program.

(Ed. Note: The author is a high school teacher of Computer Applications. This article is based on experiences gained in training students in word processing on the Apple //e.)

Com-Ware ][ contd. from pg 53

While we are talking about the Novation Apple-Cat, I would like to attempt to explain another "mystery". You may have noticed a SYNTAX ERROR right before COM-WARE ][ dialed a number from the list. This is what I think happens: When COM-WARE ][ tries to dial a number, it first looks to see the I/O FILE name associated with the number requested. It then appends this filename to a BLOAD instruction which it sends to DOS telling DOS to read the information from the disk. If the file exists, then DOS will BLOAD it into memory and change the TERMINAL characteristics accordingly. Since most of the systems we call use the same characteristics, there is no I/O FILE associated with these phone numbers. When COM-WARE ][ appends the null string to the BLOAD instruction, DOS complains with a SYNTAX ERROR. However COM-WARE ][ recovers nicely. In fact, you may not have noticed the SYNTAX ERROR at all.

As always if you have any comments or suggestions I can be contacted through the WAP office, or the phone number in the front of the WAP Journal.

## WHAT IS A PROGRAM REVIEW?

by Bob Dringel

After having several of my reviews of computer programs published in these pages, someone has finally asked: What is a review? Are there review criteria? Is there bias? What are the qualifications necessary to review programs? (Or, "Who are you to review computer programs?")

My subjective answers to the questions raised above follow. A computer program review is someone's judgment of that program. Hopefully, the reviewer has tested the program to ascertain whether it meets whatever criterion are set up. Mostly, the criteria is, does the program do what it says or its advertising says it will do?

Does the reviewer write from a biased position? Of course, he does. Other similar programs have shaped his opinion. In my case, fragmented or otherwise flawed documentation sours my estimations. Hopefully, again, the reviewer is not also the program's author, or a relative, or even an employee of the program's publisher. That would constitute unfair bias.

And as to the qualifications of the reviewer, they are, in my estimation, twofold. The reviewer must be a program user, but not necessarily a computer software or hardware expert. The reviewer must also be more facile with the American English language than he is with either high or low level computer languages. Modesty prohibits comments on my personal facility in that communication medium.

## PRINTER SPECIALS

DM = dot matrix, LQ = letter quality  
10/15" = carriage, parallel type

- Best DM print quality: Oki-Data 92A, 160 cps, DM10 \$429, DM15 \$765
- Best repair record: Prowriter 8510, 120 cps, DM10 \$349
- Best Epson-compatible: Gemini-10X, 120 cps, DM10 \$289, DM15 \$429
- Most features: Delta-10, 160 cps, DM10 \$479, DM15 \$599
- 7-color model: Transtar 315, 50 cps, DM10 \$462
- Fast Letter Quality: Printmaster F-10, 55 cps, LQ15 \$1395
- With keyboard & keypad: AJ831, 30 cps, LQ15 \$749
- Portable Letter Quality: Transtar 120, 14 cps, LQ13 \$449
- INTERFACE CARDS** for Apples: standard \$49, graphics \$75

## VF ASSOCIATES

6327 Western Ave. NW, Wash. DC 20015  
(202) 363-1313

Highest Quality, Lifetime Guarantee!

## DISKETTES

**\$1.65** 5¼" soft-sectored, hub ring, envelopes, double density, double-sided on APPLE drives -- 100 for \$155, 100 single-sided for \$149.

Hard plastic stand-up 10-diskette carrying cases \$2.75 each, 4 for \$10 (beige, black, blue, green, grey, red, yellow). Smoked-plastic flip-top 75 diskette file cases, \$19.50. Heavy-duty nibbling tool, \$22.

## Disk Drives

**\$199** 100% APPLE-compatible, 40-track, full-size, Siemens type quality drives, with manufacturer's 1-year warranty. Controller card, \$65.

COD & VISA/Master Card orders welcome. Add \$4 for shipping & handling (only \$2 for orders under \$50) plus 6% sales tax for DC residents. Send for our catalog.

## VF ASSOCIATES

Western Ave., N.W., Wash., D.C. 20015  
(202) 363-1313

# A LETTER TO THE PRES/ED

Dear President and Editor,

Since you have now twice in the WAP Journal explored the reasons for the limited member response to your open invitation to brainstorm on the direction the Pi should pursue in the future, I thought I'd write to share an idea or two as a relatively new member.

I joined WAP about four months ago, after seeing your table at the computer show at the Washington Convention Center. I have attended all but one plenary meeting since then. I have not attended a NEWSIG meeting, partly because it was canceled the first Saturday I attended the general meeting (garage sale day) and partly because the evening meetings at the Pi office have not been convenient for me (I live in Arlington and often work till 7:00 p.m.).

Perhaps all of my perceptions of the WAP would be clearer, and maybe better, if I'd gone to a NEWSIG meeting. Perhaps not. Either way, please take any comment I might offer in that perspective. I'm not an expert on how organizations are supposed to work, nor on how WAP has worked in the past or is supposed to be working now. I'm writing out of a fascination (and appreciation) for the dynamics of a group like this and because your editorials, as well as several of your comments during the business part of meetings, have given me the impression that you feel there's something wrong.

In the April Journal, you asked, "What has happened to the concept of participative management, alias democracy?" Later, you observed how "surprisingly few" members attend the weekly NEWSIG meetings. While I can appreciate the stylistically exaggerated tone of your comments, I myself as a newcomer have been surprised by how much is done by strikingly few people in WAP. Frankly, I'd expected to see a lot more "grass-roots" action, a lot more comradeship among members in general, particularly because of the "self-help" image one ascribes to any "users' group." Simply put, rightly or wrongly, I was expecting more of a sense of community.

I would be quick to admit that such an expectation may be unrealistic in a big metropolitan area like this. But given the technical, exciting, almost history-making nature of this interest we in WAP have in common, as well as the almost absolute chaos in the hard- and software markets today, I think it's understandable that a newcomer such as myself would hope for some sense of community.

To be honest, I have to concede that I have not done all I could to immerse myself in the group. As I said above, I have not attended any SIG meetings. All I've done is attend the plenary sessions, try to strike up conversation with the person(s) sitting near me, and volunteer to deliver copies of the Journal to a store near my home. In the future, I'd be happy to help out with the writing, editing or distribution of the Journal.

Nonetheless, I'd like to pursue your points about member involvement further, first, with a couple observations:

WAP membership is made up of some of the most diverse and sophisticated people in this country. Not only are they potentially important people in politics, science, medicine, education, business, journalism, etc.; they are, by virtue of their interest in computers, probably among the most innovative and forward-thinking in their areas.

The sophistication and diversity of this membership must put a burden on the Pi that other clubs do not have to endure. Our fellow members are potentially quite demanding - and perhaps a little hard to please - but they might not be accustomed to getting involved in the tedious detail of projects.

All other differences between WAP members aside, it seems that there are two types of people in WAP: those who have almost unlimited resources and can afford to pant and drool at the sight of any new hard- or software, and those who must make do with limited resources and want to focus on learning the best way to get their older, simpler machines to do more at little or no expense. It's the difference between the guy who sees the Pi as a forum for discussing new products and the guy who wants to learn neat, little tricks that are going to help him squeeze a little more service out of his Apple ][+ without buying something new. This dichotomy in the membership is sure to have some effect on the level of member involvement.

I can't say how successful Pi activities have been at reaching a balance between these two groups' desires. You would have a much more balanced perspective on this than I would, but at several meetings I have felt that discussions not only focused a lot on new things, but on expensive new things.

Perhaps I'm wrong, but I personally think that a number of demonstrations could be done without a lot of talk about new products. In fact, it's often just as exciting to learn a new trick with old software than to buy something new. During the last meeting, I read Bruce Field's Q & A explanation of using the embedded command "[V], left-arrow, [V]" to compensate for Apple Writer ]['s underlining problem. For me, trying it out - and seeing that it worked - was as much fun as booting up new software.

By no means do I think that WAP discussions should be limited to such topics. The utility of the organization would certainly decrease rapidly if it were, since people are always going to be attracted to new stuff. I myself oversee the purchase of computers and computer-related equipment in my office and need to know the "big picture" of what's new in the microcomputer industry and marketplace.

It was interesting to see the response to the various ideas for future meetings tossed out at the last meeting. While topics that went beyond the basic home computer setup - such as "coprocessors," which in themselves can be quite an investment - were loudly approved by the group, those on ways to make better use of equipment - such as advanced Applesoft programming and a disk drive clinic - were only lukewarmly received. This indicates that WAP has been quite successful at reaching the balance between new and old that the majority of members would like and that financially limited members are in the minority.

I mention all this not necessarily because I think that a change in policy is necessary for the rebirth of democracy. I just wanted you to know that up in those chairs there sit some of us who would be more active in the club, including at an organizational level, if the club discussion did not focus so much on the investment-intensive aspect of our hobby.

In such a technical area as microcomputers, the individuals who know the most are very often going to tend, wittingly or not, to intimidate the less-informed. It's a fact of life in all groups that a hierarchy develops, but the technical nature of com-

contd.



puters aggravates the situation and creates sharp differences between "insiders," who are the sought-after aristocracy, and "outsiders." It seems the latter group is eager to talk with the former but not with other "outsiders."

The two or three gentlemen who hold the 9:00-10:00 Q & A session each meeting have done much to close this gap. But, still, they can't change human nature; some Pi members have made thorough knowledge of the Apple a pillar of their egos, and there's no way (and, perhaps, no reason) for them to share their knowledge, in English, with the less informed.

Given all this, I conclude that there is nothing that really needs to be, can be or should be done to substantially change the overall structure, orientation and activities of the Washington Apple Pi. In my opinion, any major change would diminish the WAP's effectiveness as an organization devoted to serving persons from throughout the Washington metro area who own all kinds of Apples and Apple-related equipment. To break membership into smaller geographic areas or to narrow the group's orientation (for example, away from the old ][+ and toward Apple's newer, sexier machines) would alienate such a great number of members that I wonder if the Pi as we know it would survive.

On the other hand, it seems to me that some sort of change on the individual member level may be in order. I'm referring mostly to the creation of a slightly more "grassroots" or "open" environment, an ambience that would be a little easier for non-expert, limited-resources newcomer to penetrate.

Of course, the club leadership cannot and should not dictate what goes on between members. But you and the other officers can help (at least for new members) define what the Pi is: Is it a consumer group that helps warn members about shoddy products and misrepresentation in advertising? Is it a forum for members to promote products, some of which they have a financial interest in? Is it an educational group to research programming tricks, develop new software applications for the handicapped and other special-interest users? Is it to share information about emerging trends in the microcomputer industry in general? Perhaps the answer to this is written in the Apple Pi's charter, which I have not read, but it would not hurt to remind the membership of it.

Well, little did I know my few "observations" would go so long. My apology for being so uneconomical in my presentation. As I said above, all I wanted to do was work out a few ideas I had about why I, as a newcomer, have felt that, though the Washington Apple Pi is good, it could be better.

Fulton T. Armstrong

Ⓔ

# FREE SHIPPING

VERBATIM DATALIFE disk(10 packs in plastic case. SS, DD)(@23 for 100 packs or more)	24
MITAC MATE-I disk drive(100% APPLE compatible, SHUGART 390 mechanism)	179
SYNCO 12" amber monitor(18MHz)	109
Z-80A card(4 MHz)	75
80 column card	80
16K RAM card	45
Parallel printer card(cable included)	59
EPSON MX80 F/T printer	409
EPSON FX80 printer	525
MITAC 12" amber monitor(20 MHz)	119
ZENITH 12" green monitor (ZVM-123)	89
AMDEK color I plus 13" monitor	289
MITAC professional joystick (metal case with 2 buttons, 360° cursor control, auto centering & 2 fine trimers)	22
MITAC AD-10 WINCHESTER 10 MB with controller	1099

MICRO STAR COMPANY  
P. O. BOX 2307  
COLUMBIA, MD 21045  
(301)730-7172  
M - F 10:00 - 7:00  
SAT. 9:30 - 1:00

Terms: Add \$1.00 handling fee per order. MD residents add 5% tax. Personal or company checks allow 2 weeks to clear. COD accepted by M.O. or cashier check only(add \$1.65 COD charge). Prices are subject to change without notice. Q'ty is subject to availability. MC & VISA customers add 3.5% surcharge. Manufacturer or MSC warranty provided.

# TRANSFERRING BINARY FILES

by C. Swift, Prop.

When I finally got it into my head that a Binary File isn't magic, I started taking more interest in learning about this fascinating and powerful aspect of programming. No longer do I tremble when a CATALOG shows a "B" listing instead of the safe and comfortable "A". I smile smugly and think (rightly or wrongly), "I know what you are!"

Right there in a B file is located the data needed to print a hi-res picture ... or reproduce a long document ... form instructions for the computer ... or just list information. How versatile!

Then came the moment when I wanted to transfer a B file from one disk to another. GADS! It wasn't enough to LOAD and then SAVE. It wasn't even enough to BLOAD and then BSAVE. It wasn't that easy at all! It was resistant to my earnest hopes and wishes. (Of course, my ][+ treasure knew what it needed all along, but it wasn't talking...)

AHA! B files have addresses and lengths. Now who would have ever thought of such a thing? But there it was in the literature, big as life, and chortling that I had never noticed it. "Come play with numbers", it taunted me. And so I fumbled and struggled, and, to my ever-lasting surprise, with some success! Ah, the happy thought of never again having to be snubbed by a B file.

The pleasure of knowing how to transfer a Binary file from one disk to another was the well-known joy forever. But it was tedious and prone to my constant proclivity toward typing in the wrong digits at just the wrong time. The solution? Since my Apple was doing this to me, why not let it do the work for me?

Thereby, the following program. It is designed for my one disk drive, and goodness knows whether it is any good for anything other than by modest 48K, but it works for me, and that's just what I want.

```
1  REM TRANSFERRING BINARY FILES
5  REM BY C. SWIFT, PROP.
10 ONERR GOTO 1000 : REM I MAKE LOTS OF MISTAKES!
20 HOME : PRINT SPC (6); : INVERSE : PRINT
   "TRANSFERRING BINARY FILES " : NORMAL
30 VTAB 5 : PRINT "--INSERT ORIGINAL DISK"
40 VTAB 7 : HTAB 1 : PRINT "--PRESS <RETURN>
   FOR CATALOG"; : GET Z$
50 IF Z$ < > CHR$ (13) THEN 40
60 PRINT : PRINT CHR$ (4) "CATALOG"
70 PRINT : PRINT : PRINT
80 INPUT "NAME OF BINARY FILE: "; FILE$
90 PRINT : PRINT
100 PRINT "PRESS <RETURN> TO " CHR$ (34) "BLOAD"
    CHR$ (34) " " "FILE$
110 VTAB 23 : HTAB 20 : GET Z$
120 IF Z$ < > CHR$ (13) THEN 110
130 PRINT : PRINT : FLASH : PRINT "LOADING..."; :
    NORMAL : PRINT " " "FILE$
140 PRINT : PRINT CHR$ (4) "BLOAD" FILE$
150 HOME : VTAB 5
160 PRINT FILE$ " BLOADED" : PRINT : PRINT
170 PRINT "INSERT NEW DISK..."
180 PRINT : PRINT : PRINT "PRESS <RETURN> TO "
    CHR$ (34) "BSAVE" CHR$ (34) " " "FILE$
190 VTAB 18 : HTAB 20 : GET Z$ : IF Z$ < >
    CHR$ (13) THEN 190
200 PRINT : PRINT : INVERSE : PRINT "BSAVING "; :
    NORMAL : PRINT " " "FILE$
```

```
210 PRINT : PRINT CHR$ (4) "BSAVE"FILE$ " ,A"
    PEEK (43635) * 256 + PEEK (43634) " ,L"
    PEEK (43617) * 256 + PEEK (43616)
220 HOME : VTAB 5
230 PRINT "BINARY FILE "FILE$" HAS BEEN LOADED
    ON NEW DISK"
240 VTAB 12 : PRINT "PRESS <RETURN> TO REPEAT..."
250 PRINT : PRINT "PRESS <ESC> TO END PROGRAM..."
260 VTAB 22 : HTAB 19 : PRINT "< >"
270 VTAB 22 : HTAB 20 : GET Z$ : IF Z$ <> CHR$ (13)
    AND Z$ < > CHR$ (27) THEN 270
280 IF Z$ = CHR$ (13) THEN 10
290 HOME : END

1000 REM ERROR CONTROL
1010 HOME : VTAB 12 : PRINT "WHOOOPS! --- TRY AGAIN"
1020 FOR I = 1 TO 1400 : NEXT
1030 GOTO 10
```

## AUTHOR GUIDELINES

We have had quite a few requests for guidelines for authors of articles in the Washington Apple Pi Journal, so perhaps the following will help you to help us.

First and foremost, we want your articles - on any subject pertinent to the use of microcomputers - in any size, shape or form. If you do not use a word processor, we will type from handwritten or typed copy, provided that you give us sufficient time to meet our deadlines, e.g. it should arrive at the office by the 5th of the month. But, if you do use a word processor, and your article is more than a page or so, please submit it to us on a disk. We will return the disk to you (true, we have been a bit slow in doing this, but we will try harder).

We can accept most any file produced by an Apple ][ or //e computer. We use ScreenWriter which will accept any text file directly. If your masterpiece is in a binary file format, we can easily convert that, with a pass through one of our handy-dandy utilities, to a text file. If you use Pascal, we can "Huffin" your file into a DOS text file. We can convert CP/M files to DOS files (here, we prefer a file that does not have justifying spaces and carriage returns after each line, but we're not fussy about this).

We prefer to print all the articles and listings on our letter quality printer, so that there is a uniformity of print. If you have listings, send them on a disk also. We can convert Applesoft listings to a DOS text file, as well as Pascal and CP/M listings. Machine language listings may present a bit of a problem, so you might need to consult with us about that.

If you have special commands from your word processor, i.e. underlining, centering, paragraphing, etc., don't be concerned about removing them. We can convert them to commands for ScreenWriter.

Our regular format is 54 characters per column line, but you do not need to format for this. Just give it to us in whatever line width you prefer. The only thing we need to know is where the paragraphs end.

contd. on pg 62

# CONVERTING CP/M FILES TO DOS

by George V. Kinal

Washington Apple Pi club library disk #402 contains on it a program entitled "APLXFR.COM" which permits the conversion of Apple CP/M text files to DOS 3.3 format, similar to Microsoft's "APUOS" utility that converts in the other direction. Also on this disk is a file named "APLXFR.DOC", which is supposed to have the instructions for using the utility program.

I encountered several difficulties in applying this utility. First of all, the operating instructions are not accurate. They imply that you can designate the disk drive associated with each file, for example D1:Applefilename=B:cpmfile.ext. That does not appear to be true. The drives should NOT be specified in the command. Instead, the file to be converted must be in Drive 2 (or Drive B:, under CP/M terminology); the target DOS 3.3 disk MUST be in Drive 1. Then simply name the files, as follows: Applefilename=CPMFile.ext. I have updated the instructions and submitted them to the Club Library; hopefully a revision will prevent this confusion in the future.

APLXFR is a CP/M ".COM" program. But when you get all done setting it up, it creates a utility that runs under DOS 3.3!!! In other words, the final product can be used by Apple owners without a Z-80 board! If you think about it, this can be of great use in word processing and in communications. Now, you can accept disks prepared using CP/M word- or text-processors, and read them under a DOS processor or communications package. In fact, programs written for Microsoft BASIC use are stored as text files (they usually have the filename extension .BAS). These programs can then be transferred into DOS format, and edited for use under APPLESOFT. The procedure given in APLXFR for reaching the desired end result is a little arcane. The resulting 6502 binary code is supposed to be BSAVED at \$6000, with the name CPMXFER suggested. Then, an EXEC file to BRUN CPMXFER, but at location \$900, is advised. This is all unnecessary. Instead, I simply BSAVED the code with A\$6000, then BLOADED it with A\$900, and finally BSAVED it with the more descriptive name CP/M TO DOS. This final program has been submitted to the Club (DOS) Library; one simply enters BRUN CP/M TO DOS to use it. And remember, you don't need CP/M to use it, though you do need two drives in slot 6.

So far, so good. But the biggest glitch in using this utility, even if you have CP/M, is that you must know exactly the name of the file to be converted. If you make a mistake, you'll have to re-boot with CP/M, read the DIRectory, then boot up DOS and try again. If you don't have CP/M, the utility is useless to you if the file name is not known perfectly. Now comes the fun part.... I have written a program called CP/M DIR. READER. It allows you to read the Disk Directory of a CP/M disk using DOS! The program is written in Applesoft, so it's a bit sluggish for casual use, but for the semi-emergency situation described above it does the job. Now even without a CP/M card, you'll be able to determine which files on a CP/M disk are likely to be text files (these will include files with the extensions .DOC, .TXT, and .BAS; files with .COM will definitely NOT be text files), and convert those that are text to DOS format. CP/M DIR. READER has likewise been submitted to the Club Library.

One last observation. Both of these programs have a few quirks which I haven't bothered to find fixes for. They may give you strange error messages if not run

directly from a cold boot of DOS 3.3 Master. In particular, if you run the CP/M TO DOS first, then, realizing that you have the file name wrong, do a RESET, and RUN CP/M DIR. READER, you will get a FORMULA TOO LONG error, apparently because the binary code has messed up some pointers for Applesoft.

## LISTING OF CP/M DIR. READER

```
10  REM  READ CP/M DIRECTORY
20  REM  UNDER DOS 3.3
30  REM  BY G. KINAL
40  REM  WASH. APPLE PI 1983
50  REM  * * * * *
60  H1 = 38400
70  HIMEM: (H1 - 256)
80  HOME
90  PRINT : PRINT
100 DIM FM$(48): REM  FILE NAMES
110 :
120 REM  * * * * *
130 REM  POKE RWTS ROUTINE
140 POKE 47084,3
150 POKE 47092,1
160 POKE 47083,0
170 POKE 47091,0
180 LOC = H1 - 255: REM  RWTS BUFFER START
190 POKE 47088,LOC - INT (LOC / 256) * 256
200 POKE 47089,INT (LOC / 256)
210 POKE 768,32: POKE 769,227
220 POKE 770,3: POKE 771,76
230 POKE 772,217: POKE 773,3
240 PRINT "READ CP/M DISK DIRECTORY"
250 PRINT
260 INPUT "SLOT 6, DRIVE # ? (1 OR 2)...";DN
270 IF NOT ((DN = 1) OR (DN = 2)) THEN GOTO 260
280 POKE 43624,DN: POKE 47082,DN
290 PRINT : PRINT
300 PRINT "PLEASE STAND BY !"
310 :
320 REM  READ CATALOG/DIRECTORY
330 FOR SK = 0 TO 15 STEP 3
340 POKE 47085,SK
350 CALL 768
360 FOR GP = 1 TO 8
370 NX = GP + (8 * (SK / 3))
380 FM$ = ""
390 FOR CC = ((GP - 1) * 32 + 1) TO ((GP - 1) * 32
+ 13)
400 FC$ = CHR$ ( PEEK (LOC + CC - 1))
410 FM$ = FM$ + FC$
420 NEXT CC
430 FM$(NX) = FM$
440 NEXT GP
450 NEXT SK
460 REM  END OF STAGE 1
470 :
480 REM  FILTER VALID FILES
490 FOR NN = 1 TO 48
500 IF FM$ = "" THEN GOTO 580
510 IF ASC (FM$(NN)) = 229 THEN GOTO 580:
REM  DELETED OR EMPTY ENTRY
520 FM$(NN) = MID$ (FM$(NN),2,11): REM  STRIP OFF
LEADING NULL
530 IF ASC (FM$(NN)) < 65 THEN GOTO 580:
REM  BELOW "A"
540 IF ASC (FM$(NN)) > 90 THEN GOTO 580:
REM  AFTER "Z"
550 REM  .....ELSE... :
560 FM$(NN) = LEFT$ (FM$(NN),8) + "." + MID$
(FM$(NN),9,3): REM  VALID FILE NAME + EXT.
```

contd. on pg 62

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 was given in the February 1984 WAP Journal.

No tutorials - INTRODUCTION TO APPLE COMPUTER HARDWARE ( ) Tutorials may be  
 are being - HOW TO USE YOUR APPLE SOFTWARE ( ) scheduled for June  
 scheduled - BEGINNING BASIC ( ) if there is sufficient  
 for May - BITS, BYTES, NIBBLES ( ) interest. Call the office.

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 at the office on Saturday mornings, 9:30 AM to 12:00 Noon. Please register in advance.

dBase II - Lee Raesly, May 5 PASCAL - Robert Platt (4 sessions) Jun 2, 16, 30, Jul 7  
 ( ) \$10 with Apple, member ( ) \$15 with, non-member ( ) \$40 with Apple, member ( ) \$60 with, non-member  
 ( ) \$15 w/o Apple, member ( ) \$20 w/o, non-member ( ) \$60 w/o Apple, member ( ) \$80 w/o, non-member

INTRO. TO SPREADSHEETS - Lee Raesly, July 14 VISIPILOT & APPLE PLOT - Lee Raesly, Aug. 11  
 ( ) \$10 with Apple, member ( ) \$15 with, non-member ( ) \$10 with Apple, member ( ) \$15 with, non-member  
 ( ) \$15 w/o Apple, member ( ) \$20 w/o, non-member ( ) \$15 w/o Apple, member ( ) \$20 w/o, non-member

The following tutorials are planned but no exact date yet:

Word Star - Bernie Benson, Sept.; VisiCalc - Roy Rosfeld, Nov.

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 \$ \_\_\_\_\_

CP/M to DOS contd. from pg 61

```
570 GOTO 590
580 FM$(NN) = "": REM NOT A VALID FILE
590 NEXT
600 REM END OF STAGE 2
610 :
620 REM PRINT DIRECTORY
630 HOME
640 FOR I = 1 TO 48
650 IF FM$(I) < > "" THEN PRINT FM$(I); SPC( 8);
660 NEXT I
670 POKE 43626,6: POKE 47081,96: REM RESTORE SLOT 6
680 END
```

Author Guidelines contd. from pg 60

It is always comforting to have a hard copy of your article accompany the disk file, but this is not necessary if you find it inconvenient to do so.

Again, these guidelines are not meant to be mandatory or to discourage anyone from submitting articles that are not on disk. All in all, Washington Apple Pi members seem to have a wealth of information which they are willing to take the time to share with others, and this is what enables us to continue to produce an excellent Journal.

WASHINGTON APPLE PI DISKETTERIA MAIL ORDER FORM  
Software for Creative Living

Disks from Washington Apple Pi's Disketeria are available for purchase. This form is only for ordering disks that you want mailed to you. Disks purchased at the office or monthly meetings are \$5.00 each.

GRAM 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.

- |   |   |  |
|---|---|--|
| <p>DOS 3.2</p> <ul style="list-style-type: none"> <li>( ) Volume 1 Utilities I</li> <li>( ) Volume 2 Utilities II</li> <li>( ) Volume 3 Games I</li> <li>( ) Volume 4 Games II</li> <li>( ) Volume 5 Games III</li> <li>( ) Volume 6 Games IV</li> <li>( ) Volume 7 Games V</li> <li>( ) Volume 8 Utilities III</li> <li>( ) Volume 9 Educational I</li> <li>( ) Volume 10 Math/Science</li> <li>( ) Volume 11 Graphics I</li> <li>( ) Volume 12 Games VI</li> <li>( ) Volume 13 Games</li> <li>( ) Volume 14 IAC Utilities IV</li> <li>( ) Volume 15 Games VII</li> <li>( ) Volume 16 Utilities V</li> <li>( ) Volume 17 Graphics II</li> <li>( ) Volume 18 Educational II</li> <li>( ) Volume 19 Communications</li> <li>( ) Volume 20 Music</li> <li>( ) Volume 21 Apple Orchard</li> <li>( ) Volume 22 Utilities VI</li> <li>( ) Volume 23 Games VIII</li> <li>( ) Volume 24 Games IX</li> <li>( ) Volume 25 Utilities VII</li> <li>( ) Volume 26 Stocks/Investments</li> <li>( ) Volume 27 Math</li> <li>( ) Volume 28 Planetfinder</li> <li>( ) Volume 29 Utilities VIII</li> <li>( ) Volume 30 Games X</li> <li>( ) Volume 31 Plot Utilities</li> <li>( ) Volume 32 Games XI</li> <li>( ) Volume 33 Accounting</li> <li>( ) Volume 34 Solar Tutor</li> <li>( ) Volume 35 Garden Management</li> <li>( ) Volume 36 Games XII</li> <li>( ) Volume 37 Utilities IX</li> <li>( ) Volume 38 Games XIII</li> <li>( ) Volume 39 IAC VII</li> <li>( ) Volume 40 IAC VIII</li> </ul> <p>DOS 3.3</p> <ul style="list-style-type: none"> <li>( ) Volume 41 IAC 25 Mach. Lang. Util.</li> <li>( ) Volume 42 One Key DOS ***</li> <li>( ) Volume 43 IAC 29 Utilities H</li> <li>( ) Volume 44 Utilities I</li> <li>( ) Volume 90 Spreadsheet C Genl. Bus.</li> <li>( ) Volume 91 Spreadsheet D Investment</li> <li>( ) Volume 92 Spreadsheet E Bus. Recd.</li> <li>( ) Volume 100 Utilities A</li> <li>( ) Volume 101 Utilities B</li> <li>( ) Volume 102 Games A</li> <li>( ) Volume 103 Merry Christmas</li> <li>( ) Volume 104 Business A</li> </ul> | <p>DOS 3.3 contd.</p> <ul style="list-style-type: none"> <li>( ) Volume 105 FIG-FORTH/Utilities</li> <li>( ) Volume 106 Science Engineering</li> <li>( ) Volume 107 Games B</li> <li>( ) Volume 108 IAC 10 (Graphics)</li> <li>( ) Volume 109 IAC 11 (Applesoft Tutorial)</li> <li>( ) Volume 110 Personal/Education</li> <li>( ) Volume 111 Games C</li> <li>( ) Volume 112 Utilities C</li> <li>( ) Volume 113 Business B</li> <li>( ) Volume 115 IAC 12/13 Misc.</li> <li>( ) Volume 116 IAC 14 MicromodemII</li> <li>( ) Volume 117 Picture Packer</li> <li>( ) Volume 118 Utilities D</li> <li>( ) Volume 119 IAC 15 Misc.</li> <li>( ) Volume 120 IAC 16 Misc.</li> <li>( ) Volume 121 WAPABBS 1.1 Doc. **</li> <li>( ) Volume 122 IAC 17 Misc.</li> <li>( ) Volume 123 French Vocabulary</li> <li>( ) Volume 124 Utilities E</li> <li>( ) Volume 125 IAC 18 Misc.</li> <li>( ) Volume 126 Sights and Sounds</li> <li>( ) Volume 127 Math/Science</li> <li>( ) Volume 128 Games D</li> <li>( ) Volume 129 GLAQ</li> <li>( ) Volume 130 Diversi-DOS ***</li> <li>( ) Volume 131 Personal/Educ. 2</li> <li>( ) Volume 132 IAC 19 - Utilities F</li> <li>( ) Volume 133 IAC 20 - Hybrid (Pascal &amp; DOS 3.3)</li> <li>( ) Volume 134 New Members Disk</li> <li>( ) Volume 135 WAPABBS 1.1 Disk 1 **</li> <li>( ) Volume 136 WAPABBS 1.1 Disk 2 **</li> <li>( ) Volume 137 IAC 21 Spreadsheet A</li> <li>( ) Volume 138 IAC 23 Utilities G</li> <li>( ) Volume 139 IAC 24 Education 3</li> <li>( ) Volume 140 Education 4</li> <li>( ) Volume 141 Special Data Bases</li> <li>( ) Volume 142 IAC 28 Pinball Games</li> <li>( ) Volume 143 Sports</li> <li>( ) Volume 144 IAC 27 Applesoft Prog.</li> <li>( ) Volume 145 Apple Logo Tool Kit</li> <li>( ) Volume 146 Apple Logo Sample Prog.</li> <li>( ) Volume 147 Logo Documentation</li> <li>( ) Volume 148 Adventure - Disk 1} Both</li> <li>( ) Volume 149 Adventure - Disk 2} Req.</li> <li>( ) Volume 150 EDSIG1 (Elem. Math)</li> <li>( ) Volume 151 1983 Tax Template</li> <li>( ) Volume 152 IAC 31 Miscellaneous</li> </ul> <p>Eamon Series</p> <ul style="list-style-type: none"> <li>( ) Volume 180 Dungeon Designer</li> <li>( ) Volume 181 Beginners Cave</li> <li>( ) *Volume 182 Lair of Minotaur</li> </ul> | <p>Eamon contd.</p> <ul style="list-style-type: none"> <li>( ) *Volume 183 Cave of the Mind</li> <li>( ) *Volume 184 Zyphur Riverventure</li> <li>( ) *Volume 185 Castle of Doom</li> <li>( ) *Volume 186 Death Star</li> <li>( ) *Volume 187 Devil's Tomb</li> <li>( ) *Volume 188 Caves of Treas. Isl.</li> <li>( ) *Volume 189 Furioso</li> <li>( ) *Volume 190 The Magic Kingdom</li> <li>( ) *Volume 191 The Tomb of Molinar</li> <li>( ) *Volume 192 Lost Isl. of Apple</li> <li>( ) *Volume 193 Abductor's Quarters</li> <li>( ) *Volume 194 Quest for Trezore</li> <li>( ) *Volume 195 Underground City</li> <li>( ) *Volume 196 Merlin's Castle</li> <li>( ) *Volume 197 Horgrath Castle</li> <li>( ) *Volume 198 Deathtrap</li> <li>( ) *Volume 199 The Black Death</li> <li>( ) *Volume 200 The Temple of Ngurct</li> <li>( ) *Volume 201 Black Mountain</li> <li>( ) *Volume 202 Nuclear Nightmare</li> </ul> <p>Pascal</p> <ul style="list-style-type: none"> <li>( ) Volume 300 PIGO: ATTACH/BIOS</li> <li>( ) Volume 301 PIG1:</li> <li>( ) Volume 302 PIG2:</li> <li>( ) Volume 303 PIG3:</li> <li>( ) Volume 304 PIG4:</li> <li>( ) Volume 305 PIG5:</li> <li>( ) Volume 306 PIG6:</li> <li>( ) Volume 307 PIG7:</li> <li>( ) Volume 308 PIG8:</li> <li>( ) Volume 309 PIG9:</li> <li>( ) Volume 310 PIG10:</li> <li>( ) Volume 311 PIG11:</li> <li>( ) Volume 312 PIG12:</li> </ul> <p>See also Volume 133</p> <p>CP/M</p> <ul style="list-style-type: none"> <li>( ) Volume 401</li> <li>( ) Volume 402</li> <li>( ) Volume 403</li> <li>( ) Volume 404</li> <li>( ) Volume 405</li> <li>( ) Volume 406 ZCPR2 Install</li> <li>( ) Volume 407 ZCPR2 Documentation</li> <li>( ) Volume 408 ZCPR2 Utilities</li> </ul> <p>Forth</p> <ul style="list-style-type: none"> <li>( ) Volume 700 Assembler/Disassembler</li> <li>( ) Volume 701 Full Screen Editor</li> <li>( ) Volume 702 GoForth</li> </ul> <p>See also Volume 105</p> |
|---|---|--|

\* Volume 181 required with these disks. \*\* Vols. 121, 135, 136 must be purchased together.  
\*\*\* 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. Disketeria  
8227 Woodmont Avenue, Suite 201  
Bethesda, MD 20814  
DATE \_\_\_\_\_

WASHINGTON APPLE PI MAGAZINE MAIL ORDER FORM

If you would like to order back issues of the Washington Apple Pi newsletter, "Inside Apple Pi" (a compendium of our 1979 and 1980 articles), or the New Member Reference Book and/or disk, please check the appropriate box(es) below. If you order 5 or more back issues we will pay the postage (U.S. only); for 1 to 4 issues add \$.50 postage each. The price for "Inside Apple Pi" and the "New Member Reference Packet" (or book alone) includes U.S. Postage.

- \$0.50 October 1980 Volume 2, No. 10
- \$0.50 June 1981 Volume 3, No. 6
- \$0.50 Sept. 1981 Volume 3, No. 8
- \$0.50 October 1981 Volume 3, No. 9
- \$0.50 Nov. 1981 Volume 3, No. 10
- \$0.50 Dec. 1981 Volume 3, No. 11
- \$2.00 January 1983 Volume 5, No. 1
- \$2.00 Feb. 1983 Volume 5, No. 2
- \$2.00 Mar. 1983 Volume 5, No. 3
- \$2.00 May 1983 Volume 5, No. 5
- \$2.00 June 1983 Volume 5, No. 6
- \$2.00 July 1983 Volume 5, No. 7
- \$2.00 August 1983 Volume 5, No. 8
- \$2.00 Sept. 1983 Volume 5, No. 9
- \$2.00 Oct. 1983 Volume 5, No. 10
- \$2.00 Nov. 1983 Volume 5, No. 11
- \$2.00 Dec. 1983 Volume 5, No. 12
- \$1.00 January 1982 Volume 4, No. 1
- \$1.00 Feb. 1982 Volume 4, No. 2
- \$1.00 April 1982 Volume 4, No. 4
- \$1.00 May 1982 Volume 4, No. 5
- \$1.00 June 1982 Volume 4, No. 6
- \$1.00 July 1982 Volume 4, No. 7
- \$1.00 August 1982 Volume 4, No. 8
- \$1.00 Sept. 1982 Volume 4, No. 9
- \$1.00 October 1982 Volume 4, No. 10
- \$1.00 Nov. 1982 Volume 4, No. 11
- \$1.00 Dec. 1982 Volume 4, No. 12
- \$2.00 Jan. 1984 Volume 6, No. 1
- \$2.00 Feb. 1984 Volume 6, No. 2
- \$2.00 Mar. 1984 Volume 6, No. 3
- \$2.00 Apr. 1984 Volume 6, No. 4
- \$4.50 Inside Apple Pi \*
- \$4.00 New Member Reference Book \*
- \$8.50 New Member Reference Book & Disk \*

\* Includes postage.

Total No. of issues \_\_\_ at \$0.50 = \$ \_\_\_\_\_  
 Total No. of issues \_\_\_ at \$1.00 = \$ \_\_\_\_\_  
 Total No. of issues \_\_\_ at \$2.00 = \$ \_\_\_\_\_  
 Inside Apple Pi at \$4.50 = \$ \_\_\_\_\_  
 New Member Ref. Book & Disk at \$8.50 = \$ \_\_\_\_\_  
 New Member Ref. Book at \$4.00 = \$ \_\_\_\_\_  
 Postage, if any: = \$ \_\_\_\_\_  
 U.S. 1 to 4 back issues add \$.50 each  
 Foreign add \$1.50 for each back issue  
 add \$2.50 for Inside Apple Pi

TOTAL AMOUNT ENCLOSED \$ \_\_\_\_\_

NAME \_\_\_\_\_  
 ADDRESS \_\_\_\_\_  
 CITY, STATE, ZIP \_\_\_\_\_

Make check payable and send to:  
 Washington Apple Pi, Ltd.  
 Attn. Magazines  
 8227 Woodmont Avenue, Suite 201  
 Bethesda, MD 20814

INDEX TO ADVERTISERS

Anderson Jacobson . . . . .	Back Cover	MinuteWare . . . . .	31
Apollo Electronics Inc. . . . .	31	Operant Systems . . . . .	27
Bridget Software Co . . . . .	11	Paragon Technologies Inc . . . . .	31
Centauri Distributing Co. . . . .	2	Rak-Ware . . . . .	43
Clinton Computer . . . . .	29	Ramada Computer Products . . . . .	25
Columbia Computer Systems . . . . .	32, 33	Robins Inc. . . . .	25
The Interface Group, Inc. . . . .	15	Software Centre . . . . .	Inside Front
Computer Den Ltd. . . . .	45	Software City . . . . .	23
ELF . . . . .	13	Sunstar . . . . .	26
Frederick Computer Products . . . . .	41	Systems Services Co . . . . .	55
HLA Computers . . . . .	Inside Back	TIC Computer Camp . . . . .	7
LRO Inc . . . . .	19	Tri-State Systems . . . . .	43
Mercury Press . . . . .	1	Tyson's Corner Center . . . . .	11
Micro Connection . . . . .	21	VF Associates . . . . .	57
MicroPower Ltd . . . . .	9	Yes! Bookshop . . . . .	11
Micro Star Co . . . . .	59		

THE HLA TRAVELING STORE

## CURB SERVICE FOR THE BUSY BUSINESS EXECUTIVE.



### ROAD TEST A COMPUTER IN YOUR OWN PARKING LOT.

Talk about service! It's our store-on-wheels that brings a selection of the best office computers right to your door . . . saving you valuable time and giving your employees a chance to help pick the equipment they will operate. You can test the computers while you're only seconds away from your office. We'll show you off-the-shelf hardware and software and tell you how our professional programmers will custom design software especially for your needs. Ask us about our installation, warranty, service agreements and training program for your entire staff.

You can pick and choose from a wide variety of personal computers. We have them all: The versatile and powerful IBM . . . the affordable and easily expandable Apple for the office, home or classroom . . . the Compaq you can take with you, wherever . . . the functionally designed Corvus Concept with its dual-orientation and full page word processor . . . Digital's high performance Rainbow and flexible Professional 350 . . . the powerful Sage for the computer programmer . . . the HP 150 Touch Screen . . . the multi-user Altos.

To make an appointment with the HLA Traveling Store, call 301-345-8170

GSA schedule  
345-8170

# HLA COMPUTERS

5700-J Sunnyside Avenue  
Beltsville, Maryland 20705  
(301) 345-1123

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

BULK RATE  
U. S. POSTAGE  
**PAID**  
PERMIT # 5389  
Silver Spring, MD  
20910

ADDRESS CORRECTION REQUESTED



A REFURBISHED DAISY WHEEL TERMINAL  
FOR PERSONAL COMPUTER USERS AND SMALL BUSINESSES.

## Now you can have letter-quality printing and professional features for just ~~\$995\*~~ **\$712.00**

AJ daisy wheel printer terminals are renowned for exceptional performance, high reliability, and applications versatility. Now you can have all this for only \$995\* in our special limited offer.

- 30 cps *letter-quality* printing
- Changeable type faces
- Full ASCII keyboard with numeric pad
- High resolution X-Y plotting
- Complete electronic forms control
- 128-character buffer
- Asynchronous RS-232 interface
- Printwheel, ribbon cartridge, and cable included
- 30-day parts/labor warranty



And you can choose from a list of options including forms tractor, pin-feed platen, paper trays, side shelves, extra printwheels, APL keyboard and 2K buffer.

Call your nearest AJ regional office for details: San Jose, CA (408) 946-2900; Rosemont, IL (312) 671-7155; Hackensack, NJ (201) 488-2525. Or check the phone book for the number of your local AJ sales/service office.

\*Price excludes options and is subject to change without notice. Model shown includes certain options. Offer available only in the contiguous U.S.

**ANDERSON  
JACOBSON**

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.