

\$1.50

# Washington Apple Pi



Volume 3

March 1981

Number 3

## Highlights

HERE, SO SLOWLY, COMES THE ///  
CP/M FOR THE APPLE  
FLAVORS: LITTLE TIDBITS

## In This Issue

	Page
MEMBERSHIP INFORMATION	1
EVENT QUEUE	1
EDITORIAL	1
MORE ON PASCAL 1.1 - BERNIE URBAN	1
MINUTES	2
NOTICES	2
SIGNEWS	3
LETTER TO VISINEWS - WALTON FRANCIS	3
QUESTIONS, QUESTIONS, QUESTIONS - MARK L. CROSBY	4
CLASSIFIEDS	5
WASHINGTON APPLE DIGEST - MIKE LEAVITT	6
HERE, SO SLOWLY, COMES THE /// - MARK CHEREN	8
A PAGE FROM THE STACK: LIBRARIAN'S CORNER - DAVE MORGANSTEIN	9
CP/M FOR THE APPLE - DAVID NEUMANN	10
FLAVORS: LITTLE TIDBITS - BURTON S. CHAMBERS III	17
MASTER CATALOG 48K ENHANCEMENTS - DONALD E. KAHLER	22
MORE TIPS ON THE MX-80 - R. J. DECKER	22
IAC BULLETIN - FEBRUARY	23
IAC BULLETIN - MARCH	24
APPLE SOFTWARE PRODUCT NOTE	25

# ComputerLand®

We know small computers.



**apple® computer**  
Sales and Service

ComputerLand/Tyson's Corner  
8411 Old Courthouse Road at Rt. 123  
893-0424

# OFFICERS & STAFF EDITORIAL

President -Bernard Urban (301) 229-3458  
Vice President -Rich Wasserstrom (703) 893-9147  
Treasurer -Bob Peck (301) 621-2719  
Secretary -Dana Schwartz (301) 725-6281  
Members-at-Large -Mark Crosby (202) 488-1980  
-Sandy Greenfarb (301) 674-5982  
-Hersch Pilloff (301) 292-3100  
Immed.Past Pres.-John L. Moon ABBS Sysop  
Editor -Bernard Urban (above)  
Associate Editor -Rich Wasserstrom (above)  
-Genevie Urban (301) 229-3458  
Librarians:  
Disk Mail Order -David Morganstein (301) 972-4263  
Disk Pick-up -Bill Bowie (301) 924-3455  
Group Purchases -Howard Lefkowitz (301) 649-3373  
Membership -Tom Jones (301) 460-8773  
SIG Chairmen:  
ASMSIG -Jim Rose (301) 730-2230  
EDSIG -Chuck Philipp (301) 924-2354  
NEWSIG -Paul Hoffman (301) 831-7433  
Pascal SIG -Tom Woteki (202) 547-0984  
SIGAMES -Al Gass (703) 371-3560  
SIG/DISABLED -Curt Robbins ABBS WAP428  
8805 Barnsley Court  
Laurel, MD 20811

Washington Apple Pi  
P. O. Box 34511  
Washington, D.C. 20034  
(301) 621-2719

ABBS (301) 983-9317

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 \$18.00 per calendar year. If you would like to join, please call the club phone and leave your name and address, or write to the PO Box above. A membership application will be mailed to you. &

## EVENT QUEUE

Washington Apple Pi meets on the 4th Saturday of each month at 9:30 AM, at George Washington University, usually in Building C, on G Street at 23rd Street, NW. (To be sure of the exact location call the club phone or ABBS during the week of the meeting.) The March meeting will be on March 28 and the April meeting will be on April 25.

The Executive Board meets on the 2nd Wednesday evening of each month. All members are welcome to attend. Details will be on the club phone and ABBS, or call the President at 229-3458.

NOVAPPLE meets on the 2nd Saturday of the month at 1:00 PM at Kings Park Library on Burke Lake Road in Fairfax County; and on the 4th Thursday of the month at 7:30 PM at Computerland of Tysons Corner.

Mike Cornblith of Apple Computer Inc. will speak at the April 11 meeting of NOVAPPLE (time and address above). He will answer questions from the floor on APPLE and its future. All members of NOVAPPLE and Washington Apple Pi are invited to attend this very special meeting.

In the March issue of the IAC Bulletin (reproduced together with February's, beginning on page 23 of this newsletter) I have drawn attention to bit and nybble copy programs and the ethics of their use. Those of you who attended last month's meeting will recall our discussion on the subject. Please read the Bulletin and if available read Val Golding's editorial in February's Call-A.P.P.L.E., and Robert Tripp's editorial in the March issue of Micro. Both express sentiments opposing their use. Unfortunately I have not seen anything in print other than ad copy for the software which defends their use and presents the other side of the story. I ask you to think hard on this. Then check off your responses to the questions I have raised. Let me hear from you so that I can discuss it more intelligently with the Directors and Officers of IAC. &

## MORE ON PASCAL 1.1

by Bernie Urban

You may have seen and heard conflicting stories on how to get your Pascal 1.1 Update. I had high hopes of setting the record straight and published what I thought was the definitive information in the January issue of the IAC News Bulletin (it was reprinted in the February issue of Washington Apple Pi). Apparently I too have led you astray.

Dave Escoffery, Product Manager, Languages, Apple Computer Inc., called and said I had committed a no-no. We agreed to publish Apple's Software Product Note on this (which went out February 15 to all Apple dealers) in the next issue of the IAC Bulletin. March's Bulletin had already gone to press and should have reached you by now (it, together with February's issue, is reprinted elsewhere in this newsletter). Rather than wait till April, I decided to publish the "official word" in this issue of Washington Apple Pi. The Pi goes to all IAC Officers and Directors and to all the user groups within my (East Coast) region.

Please help get the word out by sending copies of the Note (or some surrogate) to all clubs which may not have gotten the word. On pages 25 - 27 you will find the Software Product Note and a portion of the note that went to Pascal owners of record. Also, I have excerpted (to omit promotional language) the letter sent by Apple to Fortran owners of record. This letter provides information on the FORTFIX program which can be used to correct some errors in the current release. &

# MINUTES

## EXECUTIVE BOARD MEETING

The Washington Apple Pi Executive Board meeting of February 11, 1981 was called to order at 7:00 PM at the home of the President, with eight persons in attendance.

A motion was passed to allow an applying school, below college level with a computer and computer club, to obtain a reduced membership rate (with an adult club advisor as the WAP member of record). A discussion of the number of newsletters and promotional material to be printed was followed by an announcement that the club had been given 'dealer status' by several vendors for group purchases. The Board encouraged the Membership Chairman to print a first edition of the Membership Directory, and to proceed with the processing of the membership cards. It was also decided to seek a Volunteer Coordinator to better utilize the available talents in the club.

The Board decided to pay for the ABBS monthly phone expenses and to move the club voice phone to the home of the Secretary. 'Hot-line' questions will be directed to the SIG group chairmen who agree to accept such questions. It was announced that a volunteer, Steve Hadley, had been found to edit a compilation of the 'Best of WAP', and the Board voiced their encouragement.

The meeting was adjourned at 9:50 PM.

## GENERAL MONTHLY MEETING

The Washington Apple Pi meeting of February 28, 1981 was called to order at 9:30 AM by the President with approximately 210 persons in attendance.

The new club phone number was announced, and a presentation was made to WAP's 500th member, David Moses. The President then surveyed the opinions of members on a 'universal copy' program and whether the INTERNATIONAL APPLE CORE should accept advertising for this program in the Apple Orchard.

The main presentation was twofold. Dr. Elaine Eckels (625-7626) gave a talk and asked for assistance in a project to aid the elderly. Rusty Luhring gave his observations and advice on setting up a business in the software development field.

The meeting was adjourned to SIG meetings at 11:10 AM.

Dana J. Schwartz, Secretary

✻

APPLES  
INTRODUCED IN  
THE U.S. IN 1629  
NOW COMPRISE  
OVER 7,000  
VARIETIES



# Notices

## NEW CLUB PHONE NUMBER \*\*\*\*\*

Please note that the club phone has changed to (301) 621-2719. It is located in the home of our Secretary, Dana Schwartz. Messages left on the club phone are directed to the appropriate person for handling.

## FOREIGN CLUB MEMBERS \*\*\*\*\*

Since we now have several foreign members of Washington Apple Pi, the following policy has been established. Foreign remittances must be in U.S. funds drawn on a U.S. bank. This means that if you get either a check or money order in U.S. funds, make sure it is payable through a U.S. bank. Many foreign banks do have U.S. correspondents or agency branches of their own, but some do not. The latter are subject to collection charges amounting to a considerable percentage. In the future, any draft not payable through a U.S. bank will be returned to the sender. So please be sure that any draft or money order you purchase abroad is not only in U.S. dollars but is payable through a U.S. bank.

## QUESTIONS, QUESTIONS, QUESTIONS \*\*\*\*\*

If you have any technical questions about the APPLE and its use, please send them to Mark Crosby, so that he can include the questions and their answers in his monthly column. Use the club P.O. Box or the club phone. When using the club phone, we will try wherever possible to answer your questions on the spot. If not we will refer you to others who may have the answer. But we are most interested in capturing your questions and their answers to be published in the newsletter for the benefit of all.

## WE NEED NEWSLETTER ARTICLES \*\*\*\*\*

We need your support in writing articles for the newsletter. Our monthly publication is probably our most important communications forum. For the past few months we have had a wealth of articles, but lately this has begun to slow up. With over 500 members, we should have an overflow of articles. Don't be shy - we need articles from and for all levels of expertise. Anything from a small programming tip to a dissertation. Please!!!!

## MEMBERSHIP STATISTICS \*\*\*\*\*

As of this writing, we have assigned WAP #586. However, about 80 members have not renewed for 1981. So that leaves us at just above the 500 mark. We don't like losing "old" members, some who have been with us from the beginning, but we can no longer carry on the rolls any unpaid memberships.

✻

# SIG-NEWS

SIGAMES is the special interest group of computer hobbyists interested in using their APPLES for entertainment. The main meeting of this group is held at a location announced at and following the Washington Apple Pi monthly meeting.

A tutorial on fixing the APPLE II's game paddles will be given by Jim Eatherly at this month's meeting. The information supplied concerning how to solder will also be of use to the hardware novice in completing the SIGAMES joysticks project.

Leading off the meeting will be a review of a new APPLE game, "SPACE EGGS".

---

PIG, the Pascal Interest Group meets on the third Thursday of each month at 7:30PM at the Uniformed Services University of the Health Services, Bldg. A, Room A2054 (2nd floor), near the National Medical Center at 4301 Jones Bridge Road, Bethesda, MD.

---

EDSIG will meet on the 4th Saturday immediately after the regular meeting of Washington Apple Pi.

---

NEWSIG will meet just after the regular Washington Apple Pi meeting. Questions regarding 13 sector vs. 16 sector disks, how to get the APPLE up and going, etc. will be answered.

The introduction to Washington Apple Pi will be held during the regular business meeting. This will be for people who have never been to a WAP meeting before. Its purpose is to tell the new members about us, what we do, how to buy the library disks, what the SIGs are all about, etc.

---

Greenapples, our SIG for young people will meet during the regular Washington Apple Pi meeting. After a discussion and planning session, the members will be accompanied by an adult to the APPLE room in the School of Engineering.

#####

Due to the Spring holidays (during which your editors will be away) there is an early deadline for the April newsletter. Please have your articles in by no later than April 10 so that we can leave the finished copy at the printers on April 15.

#####

The following is a letter written by one of our members which may be of interest to VisiCalc users.

March 9, 1981

Mr. Wes Thomas  
VISINEWS  
Box 341  
Kings Park, NY 11754

Dear Mr. Thomas:

Enclosed are 10 checks for \$12 each from members of the Washington Apple Pi club, for charter subscriptions to VISINEWS at the club rate. Our club has over 500 members, and I am confident that dozens more will be interested in becoming subscribers. Pursuant to our discussion, I am notifying club members that they can subscribe directly at the club rate by providing you their club identification number with their subscription applications.

There are probably many others who, like myself, have a love/hate relationship with VisiCalc, which is on the one hand arguably the single most brilliant and useful piece of microcomputer software ever developed (for me, indispensable), and on the other hand one of the most arrogantly cold-blooded marketing efforts ever seen. I hope that your newsletter will feel free to operate in a spirit of constructive criticism of Personal Software, with as many suggestions for improvements in VisiCalc as kudos for its achievements.

Specifically, I would like to know when, if ever, we can expect an APPLE II version of VisiCalc which has the extra features of the PET and Hewlett-Packard versions (e.g., internal rate of return, merger of VisiCalc files), plus operation with an 80-column upper and lower case card, plus a serious data base management interface (in which category I do NOT count CCA).

In the meantime, I hope that you publish as many user tips as possible on using VisiCalc, innovative ways to stretch its capabilities, and reviews of VisiCalc related software, such as Computer Station's Visilist and Progressive Software's VU #3. Your first issue sounds exciting, and I will encourage our club members to provide you with VisiCalc applications. Your offer to serve as a clearinghouse for sale of VisiCalc worksheets sounds great, but I hope that you find as many people interested in free contributions to fellow users, and publish these as well.

If you wish, feel free to print this letter - maybe Personal Software will respond to your newsletter even if they won't answer the letters of individual owners. Best of luck.

Sincerely,

Walton Francis

# QUESTIONS, QUESTIONS, QUESTIONS

by Mark L. Crosby

THIS COLUMN NEEDS YOUR SUPPORT!! Any and all items will be considered (that means we'll print it regardless). SIG groups are requested to send interesting items for publication. I suspect many of you already know the answers - so how about sharing some of them with us!

Thank you

(End of Editorial)

Q. How can I disconnect DOS so that input can be taken from the keyboard. I wish to have a file open while alternately taking input from the keyboard and writing to the disk.

A. The easiest method I know of allows DOS to do most of the work. First you must open your file using standard DOS commands. Whenever you wish to take input from the keyboard issue the BASIC statement "PR#0:IN#0". This will disconnect DOS permitting the use of "PRINT" and "INPUT" commands to obtain keyboard-entered data. When you are ready to write data to the disk, issue the BASIC statement "CALL 1002". This calls a machine-language subroutine that reconnects DOS as it was when you left it. A sample program might look like this:

```
10 D$ = CHR$(4)
20 PRINT D$"MON I,O,C"
30 PRINT D$"OPEN FILE"
40 PRINT D$"WRITE FILE"
50 PRINT "START OF FILE"
60 PR# 0: IN# 0
70 PRINT "WHAT STATEMENTS TO WRITE?"
80 INPUT A$
90 IF A$ = "" THEN 130
100 CALL 1002
110 PRINT A$
120 GOTO 60
130 CALL 1002
140 PRINT D$"OPEN FILE"
150 PRINT D$"READ FILE"
160 ON ERR GOTO 190
170 INPUT A$
180 GOTO 170
190 POKE 216,0
200 PRINT D$"CLOSE"
210 END
```

Try entering several statements totalling more than 256 characters in all. This will cause the DOS file buffer to empty itself onto the disk periodically.

Q. How can I protect the HI-RES pages from a big Applesoft program and its variables?

A. Depending on the program size and number of variables, it may be suitable to move your program above either or both of the HI-RES pages using this method:

```
POKE 103,1
POKE 104,64
POKE 16384,0
LOAD (YOUR PROGRAM)
```

Substitute 96 for 64 and 24576 for 16384 to protect both page 1 and 2.

Q. How do you convert the Apple's negative numbers to positive ones, i.e., "CALL -958"?

A. Add +65536 to the number to find its positive counterpart. -958 becomes 64578.

Q. What does the BASICS DISK do when using DOS 3.3 and the Language Card?

A. Simply speaking, the BASICS DISK will check your system's power-on language (Integer or Applesoft) and load the other so you can use both (in the Language Card). This will set you up with DOS 3.2. DOS 3.3 disks will boot normally without the BASICS DISK. The 3.3 Master will load your Language Card with the alternate BASIC as well. If you have DOS 3.3 but do not have the Language Card, using the BASICS DISK will set you up for A DOS 3.2 boot.

Q. I need a fast print-using capability while using Applesoft. Do you know of any programs that will do it?

A. The best I know of was written by R. M. Mottola and published in Nibble Volume 1 Number 6 with erratta in Volume 2 Number 1. The print-using allows printing of numbers in binary, hex or decimal format and justification of floating- and fixed-point numbers, string justification, and fixed-point with commas and/or leading/trailing characters. A copy of this excellent utility (which also contains subroutines to print repeating characters, Hi-Res screen select, ring bell, clear ON ERR condition and/or re-patch the stack after a GOSUB, clear to end-of-line or page) will soon be in the club library. Commercial rights are retained by the author.

Q. What is the correct method of tabbing while using a printer and an Apple Parallel Card.

A. All Apple Cards use this standard: PRINT TAB (5);"HELLO";TAB (20);"GOODBYE" or POKE 36,5: PRINT "HELLO";: POKE 36,20: PRINT "GOODBYE". HTAB does not function through the cards. An additional note: you should specify the line length to the Parallel Card by sending a CTRL-I followed by a number (line length) and the letter "N" followed by a carriage return. This can be accomplished within a program as: PRINT CHR\$(9)"132N". Numbers greater than 40 will turn off the video output. If your printer has expanded printing, bear in mind the tabbing is relative to character density e.g., TAB (20) is not the same absolute position for normal and expanded print. The same is true, naturally, for different pitch.

contd.

Q. I have DOS 3.3 and the renumber program doesn't renumber properly. I remember from WAP Vol 2 Number 4 April 1980 a change to prevent renumbering numbers that follow the multiplication sign (\*). How do I change 3.3?

A. Here are the fixes you need:

FOR RAM APPLESOFT 3.3

```
LOAD RENUMBER
POKE 14316,172
POKE 14317,171
SAVE RENUMBER
```

FOR ROM APPLESOFT 3.3

```
(same load/save as above)
POKE 4789,172
POKE 4790,171
```

FOR RAM APPLESOFT 3.2

```
(same load/save as above)
POKE 14342,172
POKE 14343,171
```

FOR ROM APPLESOFT 3.2

```
(same load/save as above)
POKE 4815,172
POKE 4816,171
```

**CORRECTION:** In a previous issue someone asked how to cause a program to continue without getting a "NOT DIRECT COMMAND ERROR". After a break, type "POKE 51,0: CONT" <CR> or "POKE 51,0: GOTO n" (line number) <CR>.

## CLASSIFIEDS

FOR SALE: Terminet 300 tractor fed printer. Reconditioned and in excellent condition with all manuals. \$800. Call S. A. Merritt, (804) 293-8024.

FOR SALE: Oscilloscope - Tektronix 511AD with manual; 10MHz single trace. \$85. Call Guy Black, (703) 691-0625.

FOR SALE: Integer Firmware Card. \$125 or best offer. Call Steve Hadley, home (301) 831-5353, work (301) 251-8205.

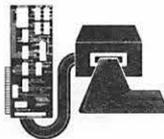
FOR SALE: Mountain Hardware Apple Clock, \$165; Mountain Hardware Computalker, \$165. Take both for \$300. Call David Morganstein, home (301) 972-4263, work (301) 251-8215.

### CORRECTION TO FEBRUARY ISSUE

Please note that the highly recommended User's Manual for the MX-80 printer by David A. Lien may still be available free from Epson. If you choose to obtain a copy from Parsons Pilchar (not Pilchard), please send \$3.00 (not \$2.00 as reported in the February issue).

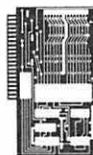
# Measure, Control, Document...

Computer Systems and Peripherals designed to coordinate with the Apple Computers that let you measure, monitor, regulate, control and document with laboratory precision, at affordable cost.



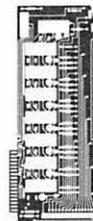
#### AI13 Analog Input System (A/D,12bits)

Measure voltages with 0.024% precision  
Monitor 16 separate inputs under program control  
Choose from 8 voltage ranges under program control; 100mV to 10V  
Make a reading in 20 microseconds



#### AI02 Analog Input System (A/D,8bits)

Measure voltages with 0.4% precision  
Monitor 16 separate inputs under program control  
Match most sensors with a standard 5 volt range  
Signal Conditioning System SC14 available



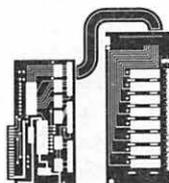
#### A003 Analog Output System (D/A, 8 bits)

Control voltage levels from a simple program  
Conversion in less than 3  $\mu$ sec.  
Standard 0 to 10 V Range  
Jumpers select 0 to 5V or 5 to 5V  
2, 4, or 8 channel models  
Each channel has its own memory  
Each channel has range, offset adjustments



#### DI09 Digital Interface

Monitor or control 32 circuits in any combination  
Measure time intervals or count pulses  
Plug into BCD, parallel, or switch closure instrument outputs



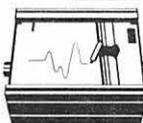
#### UI16 Isolated Power Interface

Control 110VAC circuits from a program  
Operate DC relays or solenoids  
Receive logic pulses in noisy environments  
Avoid interference and grounding problems in large systems  
Start with one circuit - expand to dozens



#### PR12 Intelligent Printer System

Clear, readable printing on standard paper  
Intermix graphics and text of three sizes  
A true printing system including intelligent interface, and a complete manual.



#### PL12 Intelligent Plotter System

Produce detailed pen drawings from a program  
8 x 10 Plotting surface with .005" resolution  
Automatically plot from Basic arrays and strings  
A true plotting system including intelligent interface and complete manual.

**Interactive Structures** has been producing laboratory systems and interfaces for Apple Computer since 1977. Our technology is in use in thousands of applications - from our own Spacelab instrumentation system to automobile engine research and woodstove analysis.

Contact us for the details on affordable laboratory computing.

 **Interactive Structures, Inc.**  
P.O. Box 404  
Bala Cynwyd, PA 19004  
215-667-4713

# WASHINGTON APPLE DIGEST

by Mike Leavitt

BYTE, December 1980

Summary of Adventure Games Reviewed  
Mike Leavitt

I usually don't bite on BYTE; I find it too oriented towards bit-pushers and prom-burners. Occasionally I like a Jerry Pournelle column and the ads, but that's it. The December issue, however, focuses on adventure games, does it extremely well, and is of much more general interest than most issues.

Three feature articles provide overviews of the basic kinds of adventures now available. Bob Liddel's "On the Road to Adventure" provides a very thoughtful and useful perspective on adventures; characterizing the computer as providing you with a "puppet, the narrator inside the program who resides in the world of your Adventure" as well as with the background and computational facilities to decide on outcomes. The article describes the Scott Adams adventures and describes each of the current releases; Greg Hassett (the thirteen-year-old Massachusetts prodigy) has his works reviewed positively; and a host of smaller game manufacturers are discussed.

"If you haven't Zorked, you haven't gamed" is the message of David Lebling's piece. Lebling and three other co-conspirators built Zork based on the original Crowther-Woods Adventure, but put a lot of computer-science-informed thought into the process to create an efficient, tight, incredibly frustrating tour-de-force Underground Empire. The clues are obscure; the villains enjoyable, the puzzles impossible (until, by pure, dumb, luck you pull them out). Lebling describes how to build adventure games right.

Jon Freeman's piece on "Character Variation in Role-Playing Games" is not strictly on Adventures; rather, it describes the kind of automated dungeons&dragons that his company (Automated Simulations) produces. The article describes the six personal factors (strength, dexterity, constitution, ego, intelligence, and intuition) that affect the play of their characters as they proceed to battle monsters, steal treasure, and rescue maidens while avoiding being killed. The games are good fun, and the character variation is a central factor in the enjoyment of one of their products.

The other major piece is Pournelle's "User's Column." He manages to review ma-

yor microcomputer languages (and a few minor but useful ones) and the major adventure games in the same column and have it all hang together. Both parts are excellent, one might even say definitive, surveys, and must be considered required reading for both the novice and the expert. Novices shouldn't lay down their first nickel on a game before reading Dr. P's review; experts should read it to make sure they know what they're talking about.

Specific game reviews abound. Most of Softtalk's adventure-related national best sellers seem to have been covered: Dungeon Campaign, Morloc's Tower, Odyssey, Microsoft Adventure, and even the Computer Bismarck game (an adventure by dispensation, only).

Finally, listings of two adventures are presented in their entirety: Scott Adams's Pirate's Adventure (Level II Basic for the Trash-80), and Teri Li's Lost Dutchman's Gold (Applesoft Basic).

BYTE did it right. If they're sold out at your neighborhood computer store, find a library or a trusting friend, and borrow it, read it, and remember it.

INFOWORLD, 1/19/81

Apple Grapples with Problems. (News analysis) The Apple III business machine is not likely to do very well in the marketplace. It's biggest problem is that it doesn't come with much software on its own, and people aren't going to buy a III and then run it in emulation mode. Its long production-caused delays won't help the situation, either. Good sensible analysis of some of Apple's problems. We should all hope they get their problems under control.

Program Protection for Apple Disks (Software review, intermediate) "'Program Protection' makes Apple disks uncopyable by normal means. The system functions primarily by putting modified DOS on the uncopyable disk. The system is extremely easy to use." Personally, I feel that this system is an abomination generator. But I understand why folks do it. Somehow, there's got to be a better way.

INFOWORLD, 2/2/81

Real-Time Clock Makes Apple Tick. (Product Announcement) "West Side Electronics announced SUPERCLOCK II a new Real Time Clock for the Apple II computer. The clock board plugs into any slot of the computer and provides time and calendar information as well as interrupt capabilities . . . from BASIC, Applesoft, and Pascal."

contd.

ASCII Express II from SDS (Southwestern Data Systems, P. O. Box 582, Santee, CA 92071) Software Review. Intermediate. "The finest program for Apple data communications is certainly the ASCII Express written by one of the founders of the Apple Bulletin Board Systems, Bill Blue . . . works best with a DC Hayes Micromodem II modem card . . . the Apple functions as a 'brilliant terminal' . . . will dial a number for you, sign you on a system, draw information from the system or transmit information from files, provide special 'customized' responses, and do it all with one or two keystrokes." This looks like the one all Apple communicators have been waiting for. Read the review first (or a company-supplied description of capabilities) but for good communications with remote machines, this seems to do it all.

INFOWORLD, 2/16/81

Applesoft Typing Tutor. Microsoft has announced an Applesoft version of the Typing Tutor (Integer has been out for a while), that requires at least 32K and one disk drive. This will permit very well-paced instruction and drills through the use of high-precision keyboard sampling.

The Apple III Appears: Some First Impressions. The article contains substantial amounts of description, much of which could have been taken from the spec sheet. The author is not satisfied with the Qume printer driver, or the lack of compatibility of some of the hardware interface capability. Aside from that, little new was offered.

What's Behind Apple's SDS. This editorial and several letters to the editor continue the discussion of Apple's policy with regard to the "Special Delivery Software." The issue is Apple's lack of support for this product line, as well as the protection of the diskettes. The critics don't want to accept the "caveat emptor" label and other conditions firmly and visibly attached to the product; the sellers don't seem to care about the ways that their policies make their product much less useful and attractive than it otherwise might be. A plague on both their houses. (MRL)

Apple World Program. Elementary Software Review. "With the Apple World program, written by Paul Lutus, you can create, modify, and display three-dimensional color images on Apple II computers. When an image is on the screen you can move your viewpoint close, further away, around, or through the image. . . . You can also move in closer and peer through a keyhole or move way back and see the house from what looks like miles away." This is recommended as quite useful, inexpensive, but definitely for advanced users, because of unsatisfactory documentation.

Computer Paradise Lost. The author feels that the originators of the Apple Computer blew a great opportunity to substantially change the way computing is done, by treating the Apple II like a computer. The author wishes that it had been treated like an electronic black box that could do many different things, like replace stock exchanges and banks. The author confuses the purposes of private enterprise with the purposes of government: private enterprise makes a buck by creating demand and then filling it; government spends a buck by failing to recognize demand, and then stifling it.

INFOWORLD, 3/2/81

Class Scheduling and Grade Reporting. Elementary software review. "The Class-scheduling and Grading system, written by Gerald Gonderinger and distributed by Charles Mann & Associates, is a new tool for teachers and school administrators . . . that automatically schedules students and reports their grades. The \$640.00 system is written in BASIC and requires a 48K Apple with two disk drives." The reviewer finds the package difficult to use, but probably worth while, considering the alternatives.

AppleGraph. This product announcement describes a microcomputer software package for general purpose plotting of data in a variety of formats for use by the business, professional, and research decision-maker.

Microsoft's Z80 Softcard. Intermediate hardware review. "The SoftCard is a fascinating piece of hardware. A quick glance at the schematic for the add-on board shows that it really only consists of decoding circuitry--necessary to allow the card to read the signals on the Apple's bus--and a Z80A chip. The board is the size of an Applesoft board and only sparsely populated with circuitry. . . . The Z80 SoftCard 'takes over' all processing of information when a CP/M diskette has been booted. All input and output (I/O) routines are still handled by the 6502." The reviewer really likes the \$365 product that has as its major use, the ability to run the WordStar word processing system. Now we Apple owners need not choose between the beauty, elegance, and sheer delight of our Apples, and the crass commercial advantages of CP/M-based software: we can now have both. &

(DISCLAIMER: Opinions expressed in articles published in this newsletter represent those of the authors and not necessarily those of Washington Apple Pi.) &

# HERE, SO SLOWLY COMES THE /// by Mark Cheren

Last October, after two months of looking into various microcomputers, I decided to buy an APPLE ///. My primary use will be word processing, but I'm also interested in doing some bookkeeping, budget planning, and a small amount of software development. It seems worth sharing what has been happening for the benefit of those contemplating a similar purchase and those with APPLE IIs who are just plain curious.

I ordered an \$8,000 system, complete with Qume Sprint Printer and an external drive. I put my deposit down and proceeded to wait. In December, I learned that to run the word processing system, I would need the full 128K and that would cost \$500 more than the price released by Apple earlier for the full word processing package.

On January 15th, my first APPLE /// came. It took a while for my dealer to get my brand new APPLE /// to work. The mother board needed to be seated properly, and somehow that wasn't near so easy as it should have been. Finally, it started operating. After three days of further testing, I took it home.

I got a few more surprises right off. I had learned in December that the word processing software wouldn't be available until June, but I wasn't quite prepared for the fact that there was still no communications driver (you can't use a modem yet), no workable RF adapter for color TV hookup, no fair quality color TV for less than \$700, and no good quality color TV for less than \$2000.

APPLE II emulation is quite a bit more cumbersome and limited than one might expect. It is necessary to choose either Applesoft or Integer on the way in. (You can't switch between them without rebooting.) It's impossible to use APPLE II Pascal. And you need to load APPLE II emulation everytime you turn the computer on. In fact, everything you do needs a driver booted first, usually with a language, like Business Basic, or eventually APPLE /// Pascal.

The fact that everything will be able to be continually improved through software updates doesn't quite convince you when you think of APPLE II firmware coupled with Autostart ROM. APPLE ///'s Autostart ROM just barely makes this constant booting livable.

Oh, something else about emulation and your old software: no paddles or joystick yet... And about that communications driver, when it comes, the D.C. Hayes Micromodem currently on the market won't work with it. D.C. Hayes says they don't

presently have any plans to make one that will. They, and so many others, are waiting to see if the APPLE /// will make it.

So, have I made a mistake? Frankly, I don't know yet. A week after I brought my new APPLE /// home, I started losing memory and then it stopped altogether. It took a month to get my second APPLE ///, but only one day for it to start malfunctioning - this time the internal disk drive. Supposedly, they've isolated the problems, a combination of insufficient fastening of boards and poor quality control arrangements. They think they have it under control and my third APPLE /// is on the way straight from California.

All of the foregoing give some reasons to go slowly before plunging into an APPLE ///. But let me tell you a bit on the positive side - why, after all this, I still haven't passed judgment on the thing. I'll admit that I have a lot more respect for APPLE IIs these days. But there is the following.

If I tell you that every key is repeatable, some with an extra fast mode, that may sound like a dandy extra, but how often do you really want a string of C's or S's or T's. OK, but what about repeatable Cntrl-T's, repeatable Cntrl-anythings or Escape-anythings. And to that add the four directional arrows, the extra function keys (open Apple and closed Apple), alpha lock (like the shift lock on a regular typewriter keyboard for everything except numbers), and you begin to have a very powerful little machine just through the keyboard innovations. Once you've played with them a bit, you aren't in such a hurry to leave them behind.

But what I'm holding out for is the hunch that Word Painter, the yet to be unveiled APPLE /// word processing software, is really going to be as good as Apple keeps saying. I'm sure hoping, as Walton Francis recommended in the February issue of Washington Apple Pi, that it is a "visual" approach that they have taken. And that it really is a dream to use - professional quality word processing.

I'm expecting that along with the quality control problem, they'll clean up most of the other problems, and that the software and more hardware will come. But it will all probably come, if indeed it does as I hope and expect, rather slowly.

## FIX FOR ELECTRONIC FILE CABINET

The following tip was retrieved from a message left on the ABBS by Steve Hadley:

When using ELECTRONIC FILE CABINET on Library Disk 22, make this change:

Delete line 1600  
Renumber line 1595 to 1600

# A PAGE FROM THE STACK: by Dave LIBRARIAN'S CORNER Morganstein

Greetings from the Library staff...only one new disk in the offing this month, a new games disk. Featured is an exceptional Hi-res Arcade game entitled ICBM. This fast action game was written by Sam Spade Consultants and was sent freely to APPLE interest groups around the country. Our thanks to Sam and the boys!!! Several other good games can be found, including Haunted House from CALL -A.P.P.L.E. and several Hi-res card games.

Help Wanted. Feels like many volunteers are needed to keep up with the current work load. Right now, it may take 6 - 8 weeks from the time an order is received before a package goes out in the mail. To speed this up we would like some volunteers for the following:

- Be responsible for organizing the incoming orders. That is, keep a count of what disks are needed and contact the reproduction crew to get the copies made.
- Participate in disk reproduction. Pick up blanks and labels, and make the required number of copies of specific volumes. Deliver the copies to the reproduction organizer.
- Mail orders. Pick up orders and copies. Complete mailing labels, stuff mailing envelopes and boxes, and deliver to the post office.

ANY VOLUNTEERS OUT THERE ????????

Seems like the new disk sales policies are making for a more peaceful Saturday morning meeting. The Library staff has been well stocked to cover on-the-spot purchases, although most people have been pre-ordering and assuring themselves that their choices will be available. Stopping sales after the meeting has allowed the staff to get to the valuable SIG groups of interest. I hope that members can make it to the meeting before 9:30 to pick up orders.

## New Commercial Software of Note -

Phantom Five.  
Another fabulous Hi-res game by Nasir of Sirius Software. You find yourself piloting a jet fighter, passing over a colorful quilt of targets below you. You can move your plane left and right and drop bombs with the button. The bombs fall away in realistic 3-D to plunge earthward and explode. Meanwhile, Ack-ack fires on the ground below and puffs of white explode around you on the screen. If you're hit, you fall away in a cloud to be replaced up to five times by reinforcements. Periodically, the scene changes to solid sky blue, out of which approach the enemy planes. You fire a stream of bullets and perhaps hit them before they hit you. If not, your screen explodes

into fragments. Congratulations to Nasir!!!

(As a footnote, I must mention that Nasir's Cyber Strike is too hard for me. I asked two arcade experts, Steve and Helena, to try it and they also agree. The only way to succeed at that one is to have the Force on your side. Skill seems not to help. What was your reaction?)

Rebel Force (Computer Conflict by SSI). what I look for in a good game is: the ability to make the game more difficult as I master it; Hi-res color; and speed. The people at Strategic Simulations, Inc. have some of the best war gaming, Hi-res stuff around. However, Bismarck and Ambush take more time that I want to spend. Rebel Force has a nice quick pace. For those familiar with Avalon Hill games, the Computer Conflict package will appear as a somewhat pared down version, occupying a smaller, though manageable, grid. The object of the game is for you to take and hold the town at the right side of the screen. Your ten divisions begin at the left and move at your discretion. As you contact the computer's forces the battle odds are computed, with the results noted. Different colored Hexagons imply different terrain, resulting in movement penalties and battle advantages. Try it, you'll like it!!!!

ABS (Muse).

This popular arcade game may drive you to purchase a joystick. (It did me...) With it you can control the target site of your five ABM's. These are used to defend your cities from the incoming streamers from above, the attacking missiles. Casually, you position the site ahead of and below the oncoming missiles and press the button. Up flashes your ABM, poof disappears the missile. Simple? Hah!! Just wait till the missile MIRV's and now there are six...then more appear. Suddenly the sky is full of attacking missiles...good luck...

Super Disk Copy III (Sensible Software). Long awaited, this update provides greatly expanded capabilities to the user. Not only able to copy, but copy in many ways. It can do the usual track at a time copy for the 2-drive owner, or the active sectors copy for 1-drive systems. It can also reconfigure the program sectors for faster loads and saves. It works in both 3.2 AND 3.3. It can fix various disk woes and cause imbedded control characters to appear upon CATALOG. A useful utility.

Next month we hope to review Zork, the long awaited super-adventure, and Galactic Attack, the first Pascal Hi-res game we've seen...it comes on a run-time package that works in a 48K machine and does not require the language system. The first of many such programs that you can expect to see (Apple's Tax Planner works this way).

⊗

# CP/M FOR THE APPLE

by David Neumann

This article is divided into four parts:

- 1) General information on the SoftCard package
- 2) A description of the CP/M system
- 3) A description of some of the major programs
- 4) General comments and comparisons between CP/M and APPLE systems

The style used is mainly one of listing the main features of the system to show what is available, and to give brief descriptions of each part. Since the main reason for getting the SoftCard is to be able to run CP/M and programs available only with CP/M, a large part of the article will be devoted to the CP/M system and associated programs. The last part of the article attempts to list some good and bad points, and to make some comparisons.

The books that I used for reference are:

- 1) The two volumes supplied by Microsoft
- 2) The CP/M Handbook With MP/M by Rodney Zaks

## GENERAL

The following items are included in the SoftCard package:

- 1) Microsoft Z-80 Soft Card
- 2) CP/M Version 2.2
- 3) Microsoft Basic Interpreter Version 5.0
- 4) Two volumes of documentation

The following programs are included with the system:

- |             |   |
|-------------|---|
| a) APDOS    | - transfers text or binary files from DOS to CP/M   |
| b) ASM      | - 8080 assembler  |
| c) CONFIGIO | - used to configure the system  |
| d) COPY     | - copies disks  |
| e) DDT      | - Dynamic Debugging Tool for interactive testing and debugging of 8080 assembly language programs |
| f) DOWNLOAD | - allows CP/M files to be transferred from another CP/M system                                    |
| g) DUMP     | - hexadecimal display of file   |
| h) ED       | - text editor   |
| i) FORMAT   | - formats disks   |
| j) LOAD     | - converts .HEX file to an executable .COM file   |
| k) MBASIC   | - Microsoft Basic extended to support lo-res, sound and game controls                             |
| l) PIP      | - Peripheral Interchange Program to transfer files between disks and devices                      |
| m) STAT     | - general status information on files, disks and device assignments                               |
| n) SUBMIT   | - input lines with parameter substitution executed from disk file                                 |
| o) XSUB     | - used with SUBMIT to allow character input from disk file  |

The following programs are available only on the 16 sector disk:

- |           |  |
|-----------|--|
| a) CPM56  | - updates CP/M system to use the language card memory    |
| b) GBASIC | - same as MBASIC, but also supports hi-res graphics      |
| c) RW13   | - allows 16 sector CP/M system to access 13 sector disks |

CP/M (Control Program/Microprocessors) is an operating system written for use with the 8080 and Z-80 microprocessors. Both 16 sector and 13 sector versions are supplied. With the language card, you have up to a 56K system.

The Z-80 addresses are different from the 6502 addresses because the standard CP/M system requires certain fixed low memory addresses. Therefore, it was necessary to remap the address space so that CP/M could have contiguous memory from 0 to DFFF without accessing the 6502 page 0 and stack or the APPLE peripheral area. The effective Z-80 clock rate is 2.041 MHz.

## SYSTEM

CP/M is logically divided into:

- |         |   |
|---------|---|
| a) BIOS | - Basic I/O System (hardware dependent) |
| b) BDOS | - Basic Disk Operation System           |
| c) CCP  | - Console Command Processor             |
| d) TPA  | - Transient Program Area                |

The BIOS provides the primitive operations necessary to access the disk drives and to interface to standard peripherals. It can be tailored to particular hardware by patching in user written software. The system is partitioned into distinct modules, so that the standard system can easily be modified to non-standard hardware by changing only the BIOS.

The BDOS provides disk management, disk allocation strategies and dynamic file construction. There are the following primitive functions:

- a) search - look for file by name
- b) open - open file
- c) close - close file
- d) rename - change name of file
- e) read - read record
- f) write - write record
- g) select - select particular disk drive

The CCP provides symbolic interface between the console and the remainder of the CP/M system. It reads and processes commands. The command line given to the CCP has a program name and file names or parameters.

The TPA holds programs which are loaded from disk under command of the CCP. Transient commands are loaded from the currently logged disk and executed in the TPA. Additional functions or commands can easily be defined by the user. The TPA is the area where non-resident operating system commands and user programs are executed.

CP/M has the following built in commands:

- a) DIR - directory of disk
- b) ERA - erase file or group of files
- c) REN - rename file
- d) TYPE - display file on terminal
- e) SAVE - writes pages of memory to disk
- f) USER - changes logical area within same directory

The standard modifications to the system are made by changing the I/O Configuration Block. The five primary functions of it are:

- 1) console cursor addressing and screen function interface
- 2) redefinition of keyboard characters
- 3) support of non-standard peripheral devices and I/O software
- 4) calling of 6502 subroutines
- 5) indication of presence and location of peripheral cards

The screen functions are handled by the software and hardware screen function tables. These tables translate the characters sent by the software into the characters expected by the hardware. This allows terminal independent software to be written.

The user can redefine up to 6 keyboard characters. This allows you to enter characters not defined for the APPLE keyboard. The standard APPLE escape editing functions, however, are not available. The user can also write and load in his own special purpose I/O driver software. Three blocks of 128 bytes each are reserved in the system for this purpose.

The logical I/O devices are assigned to physical I/O devices by means of the IOBYTE. This can be changed by the STAT program or by a user program dynamically.

SUBMIT allows CP/M commands to be batched together for automatic processing. The submitted file must have the .SUB extension. The prototype command file has \$n to stand for actual parameters on the SUBMIT command. SUBMIT substitutes parameters and creates a file of commands called \$\$\$SUB. When the system reboots at the end of the SUBMIT, CCP reads the file instead of the console. The last command in a .SUB file can initiate another .SUB file, thus allowing chained batch commands. Submitted files are only acted upon when they appear in drive A. This allows a file to be created in another drive, and execution is delayed until that disk is rebooted in drive A. The XSUB command extends the SUBMIT command by allowing character input from the .SUB file rather than from the console.

One of the ways in which CP/M programs are made compatible between different versions of the system and different computers is to use standard function numbers. If all I/O operations use these standard function numbers, then a program can run on any standard CP/M system. A list of all of the currently available functions is:

- 0 - system reset
- 1 - console input
- 2 - console output
- 3 - reader input           reader, punch and list are logical devices and can be
- 4 - punch output         redirected to any physical device
- 5 - list output

6 - direct console I/O	bypasses control character functions
7 - get I/O byte	
8 - set I/O byte	logical to physical device assignment
9 - string string	
10 - read console buffer	
11 - get console status	checks for character typed
12 - return version number	allows checking for version dependent functions such as random I/O
13 - reset disk system	disk change without reboot
14 - select disk	default disk
15 - open file	
16 - close file	
17 - search for first	scan directory for file match
18 - search for next	find next wildcard match
19 - delete file	
20 - read sequential	
21 - write sequential	
22 - make file	create and open a new file
23 - rename file	
24 - return login vector	gives on-line disk drives
25 - return current disk	
26 - set DMA address	buffer address for disk read or write
27 - get addr (alloc)	amount of remaining storage for disk
28 - write protect disk	
29 - get R/O vector	which disks are marked as read only
30 - get file attributes	
31 - get addr (disk parms)	
32 - set/get user code	
33 - read random	
34 - write random	
35 - compute file size	for random files
36 - set random record	

## PROGRAMS

This section will describe several of the major programs that come with the system. No attempt is made to give a complete description. A general idea of the workings of each program is given, and some of its commands. These commands in some cases were selected to illustrate features that are not available with standard APPLE programs.

ED is the CP/M system context editor. It has a number of commands for character string searching, replacement and insertion. There can be approximately 5000 characters in memory, but the file can be easily paged through the work area. Data is appended from the source file into the work area. This data can then be displayed, altered and written from the work area back to disk. ED creates an intermediate work file and renames the original as .BAK upon completion, and the work file is given the original name.

The memory buffer is organized as a sequence of lines. A line can be of indefinite length, and is terminated by a carriage return and line feed. There is an imaginary character pointer (CP) to indicate the character position within a line. Lines are put into the memory buffer with the A command to append from the source file, or with the I command to insert new lines.

Direction in the buffer is specified by + for forward toward the end of the buffer and - for backwards toward the beginning of the buffer.

Some commands are:

- 1) B - move to beginning or end of buffer
- 2) nC - move the CP n characters
- 3) nD - delete n characters
- 4) nK - kill (remove) n characters
- 5) nL - move the CP n lines
- 6) nT - type n lines
- 7) n - equivalent to nLT, move n lines and type single line

The text search and alteration commands are:

- 1) nFs - find the nth occurrence of string s
- 2) nSsd - substitutes the next n occurrences of string s with string d
- 3) nJcde - the juxtaposition command will repeat n times to find the c string, insert the d string after it, then delete all characters up to but not including the e string

You can include source from other files with the R command. The file must have a file type of LIB. There can be repetitive execution of commands with the M macro

command. Any string of ED commands can be repeated. Block moves are accomplished by using the X transfer command. Lines are written to a temporary file which can then be read in with the R command.

The newer versions of ED allows absolute line numbers. They are printed after using the V command. The colon is used to indicate absolute line numbers as follows:

- 1) n: - absolute line number n
- 2) :n - from the current line to line number n
- 3) m:n - from absolute line m to absolute line n

ASM is the CP/M 8080 assembler. It is a two-pass assembler that produces a print file and a code file which is in Intel hex format. It is then necessary to use the LOAD command to convert this .HEX file into an executable .COM file.

The arithmetic and logical operators allowed are:

+,-,\*,/,MOD  
NOT,AND,OR,XOR  
SHL,SHR - shift left and shift right

The assembler directives are:

- a) ORG - set program or data origin
- b) END - end program
- c) EQU - numeric equate
- d) SET - numeric set used with conditional assembly
- e) IF - begin conditional assembly
- f) ENDIF - end of conditional assembly
- g) DB - define data bytes
- h) DW - define data words
- i) DS - define data storage area

Some of the more interesting aspects of 8080 assembly instructions are:

- a) conditional call instructions
- b) conditional return instructions
- c) 16 bit load, store and add instructions
- d) no relative jumps
- e) eight 8 bit registers which may be accessed in pairs for 16 bit values

The Dynamic Debugging Tool (DDT) has the following commands:

A - enter assembly language mnemonics with operands  
D - display memory in hex or ASCII  
F - fill memory with constant data  
G - begin execution with optional breakpoints  
I - set up standard input file control block  
L - list memory using assembler mnemonics  
M - move a memory segment  
R - read program from disk  
S - substitute memory values  
T - trace program execution  
U - untraced program monitoring  
X - examine and optionally alter the CPU state

The trace displays the registers and the instruction mnemonic. Tracing is discontinued at the interface to CP/M and resumes after the return from CP/M. This allows I/O operations to run in real time. The trace mode is 500 times slower than real time.

PIP is the CP/M Peripheral Interchange Program that implements basic media conversion operations. It has a destination parameter which can be either a file or a peripheral device. The source parameter can be a series of one or more files or devices. This allows for concatenation of files. Wildcard file names can be used. A default name is assumed for PIP if no filename is given for the destination. There are two special devices (INP: and OUT:) which can be used to patch in your own input and output routines. An area is reserved in PIP for user routines for INP: and OUT:.

Some of the more interesting options are:

- a) Dn - delete characters past column n; used to truncate long lines
- b) H - hex data transfer checks for proper Intel hex format
- c) N - add line numbers
- d) Pn - page ejects every n lines
- e) Ss & Qs - start and quit copying when string s is found  
allows copying partial files
- f) Tn - extend tabs to every nth column
- g) L & U - translate to all lower or upper case
- h) V - verify data by rereading

Microsoft BASIC (or BASIC-80) Version 5.0 comes in two versions. MBASIC includes all of the standard Applesoft extensions from low-resolution graphics to sound and cursor control. GBASIC has these plus high-resolution graphics. GBASIC is available only on the 16 sector disk.

Some of the features of Microsoft BASIC that are not found in Applesoft are:

- a) CHAIN and COMMON - call another BASIC program and pass variable to it
- b) CALL - call 6502 or Z-80 assembly language subroutines or Fortran subroutines
- c) PRINT USING - formats output with asterisk fill, floating dollar sign, scientific notation, trailing sign and comma insertion.
- d) disk I/O statements - separate disk statements instead of PRINT "ctrl-D"
- e) WHILE/WEND - structured statements
- f) EDIT commands - edit program lines
- g) AUTO and RENUM - automatic line numbering and renumbering of programs
- h) IF...THEN...ELSE - handles negative case
- i) ANSI compatibility
- j) compilability - separate BASIC compiler is available it will compile to Z-80 machine code
- k) powerful date types - integer, single and double precision 16 digit as opposed to 9 digit Applesoft
- l) added string functns- INSTR, HEX\$, OCT\$, STRING\$ and direct assignment of substrings with MID\$
- m) added operators - AND, OR, XOR, IMP and EQV, integer divide and MOD
- n) user defined functns- may have multiple arguments
- o) protected files - BASIC programs may be saved in protected binary format

Four new features were added to Microsoft BASIC to take advantage of APPLE characteristics:

- 1) BUTTON - for paddle buttons
- 2) BEEP - generates tone
- 3) HSCRN - determines if point plotted on hi-res screen
- 4) VPOS - returns vertical cursor position

Applesoft enhancements that are also found in Microsoft BASIC are:

- |           |            |           |         |
|-----------|------------|-----------|---------|
| a) GR     | b) COLOR   | c) PLOT   | d) VLIN |
| e) HLIN   | f) SCRN    | g) POP    | h) HGR  |
| i) HCOLOR | j) HPLOT   | k) TEXT   | l) HTAB |
| m) VTAB   | n) INVERSE | o) NORMAL | p) PDL  |

Applesoft features that are not supported are:

- |                           |           |           |                     |
|---------------------------|-----------|-----------|---------------------|
| a) FLASH                  | b) SHLOAD | c) XDRAW  | d) ESC A,B,C,D edit |
| e) STORE                  | f) DRAW   | g) RECALL | h) SCALE            |
| i) IN#                    | j) PR#    | k) ROT    | l) HIMEM...LOMEM    |
| m) Cassette LOAD and SAVE |           |           |                     |

There are four types of variables:

- 1) string
- 2) integer
- 3) single precision real
- 4) double precision real

User defined functions may be of type numeric or string.

Some of the more interesting commands and statements are:

- a) AUTO - generates line numbers automatically
- b) CALL name - can call 6502 or Z-80 assembly language routines (or any (argument list) Microsoft compiler generated code) and pass arguments
- c) CHAIN (MERGE) - calls another program and passes variables; ALL says all filename variables are passed; start line tells where to start start line,(ALL) execution; MERGE allows a subroutine to be brought in as an (DELETE range) overlay; DELETE deletes lines to free space when done.
- d) CLEAR - set all variables to 0 and all strings to null optionally can set end of memory and stack space
- e) COMMON - Passes variables to a CHAINED program must be used if ALL is not on MERGE
- f) DEFINT/SNG/DBL/STR - declares variables that begin with specified range of letters to be of specified type
- g) DEF USR - specify starting address for up to 10 different assembly language routines; you can pass parameters to these routines

- h) EDIT - edits BASIC line with the following functions:
  - 1) moving the cursor
  - 2) inserting text
  - 3) deleting text
  - 4) finding text
  - 5) replacing text
  - 6) ending and starting edit mode
- i) ERASE - eliminates specified arrays from the program arrays; may be redimensioned or the space can be used for other purposes
- j) ERR and ERL - two variables which contain the err code and the line number where the err was detected
- k) ERROR - simulates the occurrence of a BASIC-80 error or allows the user to define his own error codes
- l) FIELD - allocates space for variables in a random file buffer
- m) FILES - prints names of files on disk like a DIR command was issued
- n) GR mode,color - initializes lo-res graphics mode; tells if bottom 4 lines are text; color is used to fill screen
- o) IF..THEN..ELSE - standard IF..THEN but negative or ELSE clause is also possible
- p) LINE INPUT - input entire line to string variable without use of delimiters
- q) LLIST AND LPRINT - lists and prints to line printer
- r) LSET AND RSET - used to move data to a random file buffer; left or right justifies; can also be used with non-fielded strings
- s) MERGE - merge disk file with program in memory
- t) OPTION BASE - set minimum value for subscripts to 0 or 1
- u) PRINT USING - formatted output
  - 1) first or first n characters of string
  - 2) right justify number in field
  - 3) floating dollar sign
  - 4) comma every 3 digits
- v) RENUM - renumber program lines
- w) SAVE - saves programs in three possible formats
  - 1) default is compressed binary
  - 2) ASCII format is required for some types of disk access such as merge
  - 3) encoded binary format is protected so that any attempt to list or edit it will fail
- x) SWAP - exchange two variables
- y) WIDTH - sets line width and screen height of terminal

Some of the more interesting BASIC-80 functions are:

- a) CVI,CVS,CVD - convert string values to numeric values used to convert random disk buffer
- b) HEX\$ - returns a string which is the hexadecimal value of the decimal argument
- c) INSTR - searches for first occurrence of one string in another string and returns position where found
- d) MKI\$,MKS\$,MKD\$ - convert numeric values to string values used for random disk buffer
- e) OCT\$ - returns a string which is the octal value of the decimal argument
- f) SPACE\$ - returns a string of spaces
- g) STRING\$ - duplicates a given character
- h) USR - calls user assembly language routine and passes parameters
- i) VARPTR - returns the address of the parameter; this address can be passed to a user assembly language routine with USR

#### COMMENTS AND COMPARISONS

Some of the comparisons in this section are with the language card or Pascal system of APPLE. Since we are essentially comparing BASIC systems, this may seem unfair, but CP/M compares favorably in many cases. Since I have been using mainly Pascal lately, I am more familiar with this system. This may account for any cases where I presented BASIC-80 as having features that are not present in Applesoft. Another thing to notice is that the comparisons are with Applesoft and not Integer BASIC.

Some of the things that I like about CP/M and BASIC-80 are:

- 1) ease of using hex numbers
  - in most cases numbers can be represented in hex
- 2) ease of modifying the system
  - CP/M is organized so that it can be more easily modified
  - a program is supplied that will modify the system
  - special I/O drivers are part of the system
  - no kludge of patching BIOS as with Pascal
- 3) logical structure of system

- the parts and interfaces between the parts are well defined  
the system is well documented  
physical devices are equated to logical devices and can be easily changed
- 4) powerful BASIC
    - there are many new functions available
    - most of the Applesoft extensions are available
    - 16 digit precision as opposed to 9 with Applesoft
    - BASIC-80 built-in editor is somewhat like Neil Konzen's Program Line Editor
  - 5) parameters on command line
    - file names and other parameters may be passed when the program is called
    - there is no need for separate prompts from within the program
  - 6) parameters in CALL statement
    - parameters can be passed to assembly language routines on the call line
    - there is no need to POKE them in specific locations and then call the routine
  - 7) the SUBMIT statement has parameters
    - this is much more powerful than the EXEC statement
    - parameters can be substituted when the file is submitted
    - the file submitted can be created with ED, no special programs are needed
    - input to the submitted program can come from the submitted file
  - 8) much CP/M format software is available
    - many business related programs are available
    - many languages and compilers are available
    - these compilers produce Z-80 machine code
    - are there any 6502 compilers???

Some of the things that I don't like about CP/M and BASIC-80 are:

- 1) only 8080 assembler
  - it is a Z-80 processor, so Z-80 assembler should be part of system
  - 8080 assembler to keep system compatible with other CP/M systems
  - separate Z-80 assembler available "soon"
- 2) shape tables are not implemented
  - only the basic plotting routines are implemented for hi-res
- 3) BASIC-80 is quite large
  - MBASIC - 26,483 free bytes in 56K system
  - GBASIC - 17,393 free bytes in 56K system

For people interested in assembly language, the Z-80 has many interesting features. Once an assembler in APPLE format becomes available, the following things can be explored:

- 1) two register sets
  - the 8080 has eight 8 bit registers and the Z-80 has 16
  - eight of the registers can be active at one time
  - the two sets of registers can be exchanged with two instructions, thus changing "environments"
  - the registers can be used in pairs as 16 bit registers
- 2) relative jumps
  - 8080 has no relative jumping
- 3) no store instructions
  - all mnemonics are expressed as loads
  - you load memory with a register LD (nn),A
  - this is a notational convention that takes getting used to
- 4) bit addressing
  - there are instructions to set, reset and test a bit
- 5) block instructions
  - there are block transfer, search and I/O instructions
  - a register is loaded with a repeat count and the instruction is executed that number of times
- 6) 16 bit instructions
  - DJNZ - decrement and jump if not zero instruction
  - load register with loop count and use one other instruction to implement loop control

---

#### FFT FOLLOWUP

by Scott A. Merritt

I was delighted by Bruce F. Field's FFT subroutine article in the September 1980 WAP newsletter. It is fast and easy to use but contains an error in the multiplication subroutine. The CLC instruction at location \$4567 should be moved to after the ASL MCD instruction, so that the MSB of MCD does not get added to the multiplier byte, MPR. The correct sequence of instruction reads:

```
$4567 PHP      SAVE RES SIGN ON STACK
      ASL MCD   REMOVE SIGN FROM MCD
      CLC
```

This correction increases the dynamic range of the transform from 15 dB to about 30 dB.

# FLAVORS: LITTLE TIDBITS

by Burton S. Chambers, III

(The flavors chosen for each tidbit are not necessarily an indication of content.)

## NECTARINE: Apple FORTRAN Problems

Applications requiring the use of Apple FORTRAN will need careful attention. Significant problems exist with certain aspects of the current release, some of which were brought to my attention by a coworker. The problem involved interfacing FORTRAN to the newly released Pascal operating system. Because I was asked to help out, I contacted Paul Sand, who asked me if I had turned in my warranty card for FORTRAN. Oops, is that really important? Sure enough Apple had recently (February) sent out a letter to registered owners of FORTRAN, which is found elsewhere in this issue, describing the "FORTFIX". Paul came through and provided me with this copy. However, I was suspicious that the "fix" wouldn't clear up the problems my friend at work was having. Naturally, I ended up calling the Apple Hot-Line. A discussion with Jo Kellner, clarified what the "fix" did. But... not all is well in Applesville. Jo passed on the following information about known bugs in Apple FORTRAN.

1. Labeled COMMON: a maximum of seven references will yield correct results. Beyond that, errors will occur. (If this is accurate, avoid use of labeled COMMON until Apple is sure they have fixed the problem)

2. Passing REAL variables from Pascal to FORTRAN will not work correctly.

3. FORTRAN will not run correctly with revision [1.1] of the Pascal operating system, unless each FORTRAN code file is modified by a special program, ie. FORTFIX.

(I only have first hand knowledge of #3, and was very surprised to learn of #1, since I have used labeled COMMON and wasn't aware of problems, although I must admit I program in Pascal most of the time.)

## PEAR: Bilingual Programs.

Notwithstanding the aforementioned problems, I recently had an opportunity to attempt a marriage of two FORTRAN UNITS with a Pascal program. I finally gave up because the run-time system for FORTRAN hogged too much core, probably with stuff I wasn't using. Nevertheless, I learned something you may someday need to know (but not for a 4000 line program). When you use a FORTRAN UNIT in a Pascal program, you have a problem if one of those routines define a TYPE not a scalar. For example, if you pass characters to a subroutine in the FORTRAN UNIT, a new type is set up, and if the number of characters in the passed variable is 6, the type identifier happens to be ALFA06. The apparent catch 22 is that you can't define such a type in your program, because the UNIT's declarations are considered before the programs. The fix may be obvious to you, but to me I had to think a while. The solution is:

```
(*S+*) (* UNITS need swappings mode *)  
UNIT defalfa6;
```

```
INTERFACE  
TYPE alfa06 = PACKED ARRAY [1..6] OF CHAR;  
PROCEDURE NOTHING;
```

contd.

```

IMPLEMENTATION
  PROCEDURE NOTHING;
  BEGIN
  END;

```

```

BEGIN
END.

```

This UNIT then must be used in the Pascal program before the FORTRAN UNIT is used, for example:

```

Program Pascal;
Uses
  (*$U DEFALFA.CODE*)
  defalfa6,          (* THE DUMMY UNIT *)
  (*$U MY.LIBRARY*)
  fortrana;          (* THE FORTRAN UNIT *)

TYPE whatever = INTEGER;
(* Shows where your program declares its types *)
BEGIN
  WRITELN('An example program')
END.

```

PLUM: Capturing APPLESOFT files as TEXT files.

There may be times when you may wish to capture your APPLESOFT programs in a TEXT file to be EXECed in later, or sent over phone lines, or whatever. If the lines are long, however, you may have a difficult time putting them into memory. The problem is the 255 byte limitation being exceeded because your lines were already long, and LIST added a pile of blanks. Well another friend in the office needed a hand, so I offered to help. Much pain and embarrassment. Close to 8 hours went down the old rathole. The job in Pascal would have taken no more than one-half hour. So I have decided to show you the result (muffled laughter) and hope you find some use for it. Please, someone courageous that likes programming in BASIC, write a decent text editor specifically for APPLESOFT.

```

1  REM  PRELIMINARY MARCH 1981          |      COMMENTS
2  REM  BURTON S.  CHAMBERS III        |      -----
3  CLEAR :                             |
   GOTO 30                             |
4  PRINT Q$;" [Y,N]";:                | The subroutines are
   GET AN$:                            | kept near the front
   IF AN$ < > "Y" AND AN$ < > "N" THEN | for speed. Note that
   PRINT :                             | this program is not
   GOTO 4                               | very fast.
5  PRINT :                             |
   RETURN :                             | A general yes & no
   REM  ASK A QUESTION                 | routine.
6  PRINT D$;:                          |
   RETURN :                             | Trying to keep out
   REM  TURN OFF READ OR WRITE        | extraneous returns
7  PRINT CD$;W$;FO$:                  | and making sure DOS
   RETURN :                             | commands are seen !
   REM  WRITE                          | WRITE OUTPUT file.

```

```

8 PRINT CD$;R$;FI$;
  RETURN ;
  REM READ

9 A$ = ""
10 GOSUB 6:
  GOSUB 8

11 GET CH$:
  IF LEN (CH$) < = 0 THEN
    GOTO 11:
  REM GET ALL CHARS EXCEPT NULLS
12 IF (Y$ = "Y") AND
  ( ASC (CH$)< 32) AND
  (CH$ < > CR$) THEN
    GOTO 11
13 IF (Y$ = "Y") AND
  ( ASC (CH$)> 127) AND
  ( ASC (CH$) < 32+ 128) AND
  (CH$ < > RC$) THEN
    GOTO 11
14 IF CH$ < > " " THEN
    GOTO 18
15 IF BB$ < > "Y" THEN
    GOTO 18
16 IF QQ$ = "Y" OR NOT QU THEN
    GOTO 11
17 GOTO 20
18 IF CH$ < > CHR$ (34) THEN
    GOTO 20
19 QU = NOT QU
20 IF CH$ < > CR$ THEN
  A$ = A$ + CH$:
  GOTO 11
21 GOSUB 6:
  GOSUB 7
22 IF LEN (A$) < = 0 THEN
  PRINT :
  GOTO 10
23 PRINT T$;A$:
  GOTO 9
24 GOSUB 6:
  PRINT
25 IF LEN (A$) > 0 THEN
  GOSUB 7:
  PRINT T$;A$
26 GOSUB 6
27 PRINT CD$;C$;FI$
28 PRINT CD$;C$;FO$
29 END

30 REM =====
31 REM TEXT FILE CLEANSER
32 REM INITIALIZATION
33 REM =====

34 TEXT :
  HOME :
  VTAB 15:
  HTAB 10

```

```

|
|
| READ INPUT file.
|
| The loop over each
| output line.
|
| Start reading a
| line.
|
| Get rid of nulls
| (don't use ASC)
|
| Skip control chars
|
|
| Ditto with > 127
|
|
| Is it a blank?
|
| Do we skip them?
|
| Is it in a quote?
|
|
| Is it a quote?
|
| Yes, so toggle
|
| Otherwise, add
| to line.
|
| Ready to write.
|
|
| Write it out!
| and start next
| line.
|
| An error! Which
| should be END
| of Data. I do
| not bother to
| check, instead
| I just close
| the files and
| exit.
|

```

```

35 INVERSE ;
   PRINT " TEXT FILE CLEANSER ";;
   NORMAL
36 PRINT ;
   PRINT ;
   FLASH ;
   PRINT "PRELIMINARY";;
   NORMAL ;
   PRINT " MARCH 1981 B. CHAMBERS "
37 PRINT ;
   PRINT " FILE NAME OF INPUT FILE ? " :
   INPUT " ";FI$
38 PRINT ;
   PRINT " FILE NAME OF OUTPUT FILE ? " :
   INPUT " ";FO$
39 PRINT
40 Q$ = " DO YOU WANT CONTROL CHARACTERS REMOVED?":
   GOSUB 4:
   Y$ = AN$
41 Q$ = " DO YOU WANT BLANKS REMOVED?":
   GOSUB 4:
   BB$ = AN$
42 Q$ = " HOW ABOUT INSIDE QUOTES?":
   GOSUB 4:
   QQ$ = AN$
43 D$ = CHR$ (4);
   REM CTRL-D
44 O$ = "OPEN "
45 C$ = "CLOSE "
46 W$ = "WRITE "
47 R$ = "READ "
48 CR$ = CHR$ (13)
49 RC$ = CHR$ (13 + 128)
50 T$ = " " :
   REM A BLANK
51 CD$ = CR$ + D$
52 PRINT CD$;"MONC,I,O":
   REM LEAVE IN NO MATTER WHAT!
53 PRINT CD$;O$;FI$:
   PRINT CD$;O$;FO$:
   REM OPEN FILES
54 ONERR GOTO 24
55 GOTO 9

56 REM =====
57 REM PRELIMINARY 1981
58 REM BY B. CHAMBERS
59 REM =====
60 REM GOOD COMPUTING !

```

Incidentally, the above APPLESOFT program was copied onto a Pascal TEXT file and then edited with the Pascal EDITOR. I wish that I could report that I have such a program to do this on one Apple, but alas, such is not the case. I called another Apple, that conveniently was operating in APPLESOFT in the remote control mode (using a D. C. Hayes Micromodem), and my Apple was using the Pascal Operating System using Datacomm by Hayes Microcomputer. Pairs of WAP members, who use Micromodems, could do the same thing.

TANGERINE: Datacomm

At this point, I am a little sun-shy, but I'll stick my neck out one more time. Hayes Microcomputer has just started shipping Datacomm. It should either be in the stores by the time you read this, or all sold out. If this statement ends up not being true, I intend never to mention that program again.

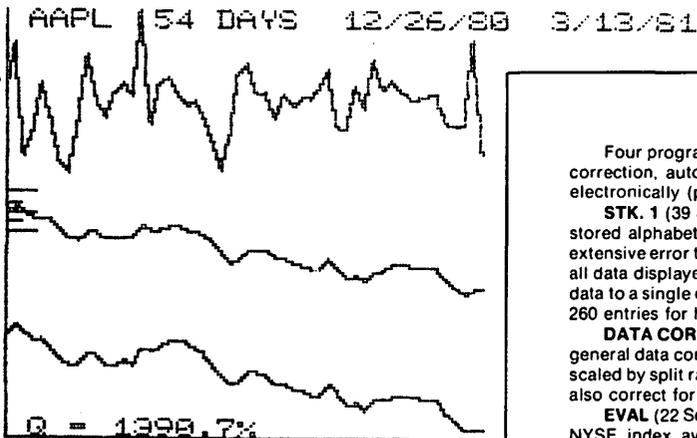
My employer recently purchased an Epson MX-80 printer for use with one of the Apples that are in the office. I had an opportunity to try it out, and wanted to pass on to you some information that I haven't seen explicitly stated in any of the good reviews published to date in this magazine. First, this printer is manufactured in Japan, and looks like the Japanese are very serious about competing with U. S. companies. Naturally, everyone wants to save their money, but what is going to be the long term effect on our economy. The printer works well, but was originally set up for the TRS-80 it would seem, at least as far as graphics go. With the Apple, you can't access the graphics characters, even though they are there (somewhere) and are printed out for the printer test.

A previous review indicated that a graphics package is going to be made available in the near future. I personally don't like to invest in something that has a feature you need, until that feature is actually available.

On the surface it would seem to be a good investment, since the printer does provide very good copy for a relatively modest price. Furthermore, it has a disposable head that is advertised to cost \$30 to replace. This seems like a good idea, but how much does it cost for ribbons, what is their availability, and how long is the motor expected last? I hope everyone will share any failures they have with new products as well as the successes. And I also hope the Epson holds up under the torture we are liable to give it.

Good computing !

♣



★ ★ ★ **STOCK MARKET UTILITIES** ★ ★ ★  
4 STOCK MARKET PROGRAMS ON DISK

Four programs provide a complete programming system for entry and storage of stock data, data correction, autoscaling Hi-Res graphical display of performance, and building historical data files electronically (program to download data not included). **Now with HARD COPY GRAPHICS.**

**STK. 1** (39 Sectors) provides complete utilities for manual entry of stock data. **Features:** names stored alphabetically by exchange, easy addition and deletion of names, automatic prompting and extensive error trapping for data entry (date, volume, price), numerous entry points for data correction, all data displayed prior to updating stock files with further option for data correction, input historical data to a single data file, display contents of individual stock files from disk, option to reduce files to last 260 entries for high-res graphics. All data files are fully accessible.

**DATA CORRECTOR** (31 Sectors) used to correct and rewrite stock data files. **Features:** option for general data correction - correct any entry, option for stock splits - all prices and volumes prior to split scaled by split ratio (transaction dollars constant) to provide continuous momentum and price curves, also correct for incomplete updating due, for example, to a power outage.

**EVAL** (22 Sectors) provides comparative evaluation of stock performance. **Features:** synchronizes NYSE index ave with first stock entry, option to evaluate all stocks automatically or just one, simultaneous high-res display of momentum, price, and price relative to NYSE index, auto scaling graphics, numerical figure of merit for performance relative to NYSE index ave.

**MIROQ** (12 Sectors) is used to build historical data files electronically by converting downloaded stock price data obtained from Compuserve's Micro-Quote financial data base to data files compatible with these programs.

Programs written by H. S. PILLOFF.  
Requires Apple II,™ ROM Applesoft,™ 48K and Disk (DOS 3.2)

★ ★ ★ **ELECTRONIC STOCK PACKAGE** ★ ★ ★

A complete system including password and programs for accessing the Dow Jones Stock Quote Reporter (contains more than 6000 daily stock prices). Current rates permit nightly updating for about 1.5¢ per stock.

Downloading programs provide for auto dialing, logging on, retrieving daily data (prev. close, open, high, low, close, volume) for up to 200 stocks stored in easily edited file, disconnecting from system, and then writing data to a single file on the user's disk. Data can then be displayed or printed.

Conversion programs read this disk file, formats data (M/D/Y/VOL/FNL), and automatically updates each individual stock file. Format is fully compatible with STOCK MARKET UTILITY PROGRAMS.

Requires Apple II/II+,™ Applesoft,™ 48K, Disk (DOS 3.2 or 3.3), and D.C. Hayes Micromodem II™.

Electronic Stock Package (includes Dow Jones password) \$80.00  
Stock Market Utility Programs \$59.95

**COMING SOON!! Black-Scholes Stock Option Evaluation Programs**  
*Specializing in Investment Software*

Foreign orders add \$10 for shipping. **H&H SCIENTIFIC** VISA/Mastercard Accepted

13507 Pendleton Street, Oxon Hill, MD 20022 Tel (301) 292-3100

Apple II/II+, and Applesoft are trademarks of Apple Computer, Inc. Micromodem II is a trade mark of D.C. Hayes Assoc., Inc.

# MASTER CATALOG 48K ENHANCEMENTS by Donald E. Kahler

The excellent 'MAS.CAT. 48K REVISED' on club disk 22 would not print two or three columns properly on my printer because I have a serial card, and the TAB or POKE 36 functions do not work as intended with this card. However, the SPC function does work. So I made the changes listed below in the program, and in the process tried to make it a little more elegant by right-justifying the record numbers, so that everything stays lined up whether there are 1, 2 or 3 digits in the record number.

Also, the DELETE, VOLUME function wasn't working for me. The problem was apparently in line 3032, and since my tired old brain couldn't successfully analyze that code, I replaced it with lines 3030 and 3032, which even I can understand.

```

3030 IF LEN(VD$) < 2 THEN VD$="00"+VD$
3032 IF LEN(VD$) < 3 THEN VD$="0"+VD$

4135 HD$ = "NO. VOL TYPE    PRO
      GRAM
      "
4137 PRINT HD$;: IF LP$="Y" THEN PRINT
      HD$;HD$;
4138 PRINT
4140 IF N < 0 THEN 4305
4142 PK = 42
4150 FOR I = FI TO LA
4151 IF NA$(I) = "DELETE" GOTO 4260
4153 IF I < 10 THEN TA = 2
4154 IF I > 9 THEN TA = 1
4155 IF I > 99 THEN TA = 0
4156 PRINT SPC(TA);
4157 PRINT I;" ";NA$(I);:SY = LEN(NA$(I)):
      IF (I + NE) > 99 THEN SY = SY + 1
4158 IF LP$ = "N" GOTO 4180
4160 IF LA > I + NE THEN PRINT SPC
      (PK - SY);I + NE;" ";NA$(I + NE);:
      SY = LEN (NA$(I + NE)) + 1

4170 IF LA > I + 2 * NE THEN PRINT
      SPC(PK - SY);I + 2 * NE;" ";
      NA$(I + 2 * NE);

```

## MORE TIPS ON THE MX-80 by R. J. Decker

I read the note about the Epson MX-80 printer by Mike Kramer in the February newsletter with interest, since I have had an MX-80 for many months now. I thought that some clarification of the problems mentioned would be of interest.

The problem with the horizontal tabbing arises because, while the printer uses CHR\$(9) as a tab command, the Epson interface card (following the example of the Apple parallel interface card) uses

Ctrl-I as a printer card command flag. (Ctrl-I = CHR\$(9)). Therefore when a horizontal tab, CHR\$(9), is sent to the printer from a program, the interface card intercepts it and attempts to execute the printer card command. This leads to strange looking printouts. The cleanest fix is to issue a command to change the printer control character. This is done by sending: Ctrl-I Ctrl-x to the printer card (the x can be any character you select). This will cause the Ctrl-x to be the command flag instead of the Ctrl-I. I prefer this way of dealing with the problem because I hate POKES and I still have the horizontal tab capability without having to bother about column numbers in my programs. (If I cared about column numbers I would use TAB() - see below.)

The interface card I received with my printer was originally designed for the Epson TX-80, but Epson assured me that it would work fine with the MX-80. This is true as long as you don't want to print any of the graphics symbols available. The card as supplied has a jumper at P4 which grounds the D8 line to the printer. This causes all character codes in the range of hex 80-FF to be sent as 00-7F. Graphic symbols are in the range A0-DF. The graphic symbols will just not print. The fix is to move the jumper from P4 to M4. Caution: be wary of software which may try to print characters in screen format (top bit set)!!

The last problem is that the TAB() keyword in a PRINT statement will not work properly if the printer line width is set using the Ctrl-I printer command. Without this fix, the program statement:  
100 PRINT "A";TAB(20);"B";TAB(40);"C"  
will print the C at column 60 instead of column 40. I found out about this fix (an IAC ApNote) from G. Robbins. It should work for serial, parallel, silenttype and comm card interfaces. To use: CALL 944 instead of PR#s initially and CALL 954 for all subsequent PR#s.

```

03B0-  A9 04    LDA #$04
03B2-  20 95 FE JSR $FE95
03B5-  A9 8D    LDA #$8D
03B7-  20 ED FD JSR $FDED
03BA-  AP C5    LDA #$C5
03BC-  85 36    STA $36
03BE-  A9 03    LDA #$03
03C0-  85 37    STA $37
03C2-  4C EA 03 JMP $03EA
03C5-  20 aa bb JSR normal entry
03C8-  48      PHA
03C9-  AD cc dd LDA column
03CC-  85 24    STA $24
03CE-  68      PLA
03CF-  60      RTS

```

```

Serial: aa=7 bb=C0+slot# cc=F8+slot# dd=5
Par:    aa=2 bb=C0+slot# cc=F8+slot# dd=7
Silent: aa=7 bb=C0+slot# cc=4      dd=CF
Comm:   aa=7 bb=3      cc=2      dd=3

```

PIGs like APPLES.



# BULLETIN

P. O. BOX 976, DALY CITY, CALIFORNIA 94017 USA

We are in the throes of refining long range goals for the IAC and objectives for 1981 and subsequent years. Got any favorites? Get them into your friendly Director (or to Jerry Vitt) by no later than February end. We intend to have them approved by the Board and available for your information by annual meeting time in Chicago.

Speaking of which, it will be held in the Marriott Hotel located near O'Hare on May 2 and 3. This is the weekend preceeding the National Computer Conference. The IAC annual meeting will include the general business meeting and seminar sessions to be given by Apple experts from member clubs and from Apple Computer Inc. Plan to attend.

On the move...

Scott Knaster, one of our Board Directors for Region 3 has joined Dakin5 Corporation.

Product Poop...

Apple Pilot which uses Pascal 1.1 runs fast and has four editors - Text, Music, Graphics and Character Set Generator. Worth a look, particularly by educators.

Apple Computer Inc. is coming out with its own modem. Rumored to have touchtone or pulse dialing capability and dial tone or busy signal detection. Numerous features above and beyond the D.C. Hayes.

Miscellaneous trade papers have reported that Apple Computer Inc. has signed a contract with Seagate Technology (formerly Shugart) for a large order of 5 1/4 Winchester disk drives (5 - 6 m bytes). No information re pricing, availability, etc.

More hot air - there are now two sources for fans to coddle your treasured peripheral cards. One reported on last month is from the M&R folks - the other is from a company called MRE in Chicago. Both are priced around \$50.

New Lipson light pen now available which works with 80-character Videx card.

Caution!

The NEC jolly green monitor for the APPLE III will clobber your disks. Plastic non-shielding is the culprit.

Apple Orchard contributors, please take note. The pay rates for accepted copy has been increased to \$60/page for a standard article and \$75/page for a feature/lead article. Help us make the Orchard the best Apple magazine going. Remember, please mark your copy "For the Orchard".

Reminder...

Jerry Vitt has his IAC Bulletin Board System up and running. Post your club meetings, items of interest, check in, leave messages or just browse. (214)369-0427.

Bernie Urban, Ed.  
February 2, 1981

23



## APPLE ORCHARD SUBSCRIPTIONS

P. O. BOX 1493 BEAVERTON, OR 97075, USA

The International Apple Core will make individual subscriptions to "The Apple Orchard" available commencing with Volume 1, Number 3.

NAME \_\_\_\_\_

STREET \_\_\_\_\_

CITY \_\_\_\_\_ STATE \_\_\_\_\_ ZIP \_\_\_\_\_

COUNTRY \_\_\_\_\_

Annual Subscription Rate: \$10.00 per year (Four issues - Published Quarterly)  
First Class Postage: \$5.00 per year additional (required for Canada, Mexico, APO, and FPO addresses)  
Overseas and other foreign air mail postage (required): \$10.00 per year additional  
TOTAL REMITTANCE ENCLOSED: \$(USA) \_\_\_\_\_

Make check or money order payable to "International Apple Core" and return with this form to:

APPLE ORCHARD SUBSCRIPTIONS  
P.O. Box 1493  
Beaverton, OR 97075, U.S.A.

Attention:

On Sunday, February 22, A.B.S. Computer Services of Olympia, Wash. was robbed and its owners, Henry and Laverne Rumberger, were killed. The following items (listed with serial numbers) were taken:

1. APPLE III, 128K 001065
2. APPLE II, 48K A2SA 152415
3. APPLE II, A2M030 11892
4. APPLE II, 48K w/controller 95802
5. DISK II w/controller 215227
6. BMC Monitor 65101709
7. Sony KB1276 TV 503527
8. Centronics Printer 12349
9. 2 IDS 440 Printers 315 and 316
10. Stockwatch 15N 5980

If you have any information on these items, call Detective Jones of the Olympia Police Department.

What do you think?

This is a knotty problem - it won't go away, so let's all think on it. By now most of you should be aware of Locksmith (and others of its type). It is a nybble copy software package available for the APPLE II. It enables its purchaser to copy most any APPLE II diskette and, based on what I've heard, does so with few exceptions (e.g., itself). I do not intend to pass judgment on it nor whether there is a need for it. I do know that software firms are threatening to withdraw their ads from publications which advertise it. Its proponents argue that backups of expensive diskettes are an absolute necessity, if only because of the time required to obtain replacements for clobbered or worn-out originals. Also, supporters argue that copies are needed where a firm has purchased one copy but needs several for use by its staff. Opponents argue that it panders to the greed and something-for-nothing instincts within all of us by giving us the means to pirate copies for ourselves, for our friends or for profit.

The IAC must face this issue squarely. We need your opinions. We have already gone on record to be against software piracy. But is the purchaser and user of the Locksmith a pirate? What constitutes piracy? Take some time at your next meeting to discuss this issue with your members and tell us where your club comes out. Then answer the following:

Should the IAC

- |   |                |
|---|----------------|
| 1. condone its sale and use?                              | Yes ___ No ___ |
| 2. advertise the Locksmith in the Orchard?                | Yes ___ No ___ |
| 3. do nothing?  | Yes ___ No ___ |
| 4. refuse ad space in the Orchard?                        | Yes ___ No ___ |
| 5. express an official position against its sale and use? | Yes ___ No ___ |

It deserves time at the annual meeting in Chicago but May is too far off. Let's get feedback from all of you soon. Results can be presented at the meeting, or if necessary we can schedule an open forum to cover it more effectively.

Bernie Urban, Ed.  
March 5, 1981

Some bad/good news.

The bad, Steve Wozniak - the Woz - co-inventor of the APPLE, crashed his private plane a few weeks ago. Both he and his fiancée, Candy, suffered severe lacerations of the face. He also suffered a concussion and partial lapse of memory. He had just been transferred to El Camino Hospital in Mountain View, California, when I called him. He was quite concerned about Candy. He was overwhelmed by the concern and well wishes expressed by all the "APPLE people". He says he intends to fly again. The good news, both have been released from the hospital, and the prognosis is good.

Here's what new IAC member clubs should be getting:

- The constitution and by-laws of the IAC.
- Identification of all IAC Board Members with their areas of responsibility, all Officers and Committee Chairmen.
- Information pertaining to the IAC Apple Orchard - how to subscribe, how to get back issues, current methods for submitting articles, page rates for accepted articles, etc.
- A complete set of IAC meeting minutes.
- The entire set of ApNotes issued to date and all succeeding submissions.
- The next IAC software disk and all succeeding disks.

New:

- The latest version (??) of the APPLE II allows for easy modification by you (beware of voiding warranties) to enable the use of the shift key to get capital letters. Also, it seems to provide a socket for the SupR-term (80 column) board. Plus other goodies.
- Computer Data Services is marketing a Memory Management System which expands your APPLE by 10.5K of programmable memory by relocating DOS onto a 16K RAM expansion board language card for use in slot 0. Retail for less than \$50.
- More power to ya. M&R has come out with a beefed up power supply. The SupR switched external power supply looks just like a disk drive and supplies 60 watts, 6 amp at 5 volts or 1 amp at  $\pm 12$  volts, etc. Will retail for approximately \$295.

As of Monday February 23, Apple Inc. has established two dealer assistance lines which cover approximately three fourths of the U. S. (Alaska included). This service is to dealers only but APPLE owners can tap in through their local dealers in those areas presently covered. Currently this means owners and dealers within and north of Fresno, California, eastward to but not including Mississippi and Tennessee. The territory jogs over to include Kentucky, West Virginia and Ohio. Next in line will probably be the East Coast and the remainder by about May 1.

SOFTWARE PRODUCT NOTE

APPLE FORTRAN  
PASCAL 1.1 REVISION

February 1981



10260 Bandley Drive  
Cupertino, California 95014  
(408) 996-1010

January 1 1981

Since the Apple Pascal 1.1 Revision Kit was first announced in Apple Service Bulletin #21 and in your October 15 mailing, a certain amount of confusion has somehow surfaced among Pascal customers. Last month Apple sent a direct letter to all Language System owners of record informing them of the upgraded software and documentation. Unfortunately since only 10 percent of the warranty cards have been returned, most users remain unnotified. Obviously we cannot stress enough the importance of urging your customers to send in their warranty cards and Software License Agreements.

Those users who did receive the letter were informed of the added features and the two updated manuals. Also, the new automatic method of configuring additional device drivers was discussed. Complete information on this Attach feature is being distributed through International Apple Core member user groups.

Please remember that customers who purchased a Language System on or after August 1, 1980 are considered to be within the Warranty period and are entitled to receive the Pascal 1.1 Revision Kit free of charge. All customers who have purchased a Language System prior to that date must pay the list price of \$60.00. In either case the customer must return to you their original Pascal system diskettes.

Apple FORTRAN has been released for over six months. Like any such sophisticated software package, it has gone through a "shake-down" period. Recently Apple sent an additional letter to all registered Apple FORTRAN users. Again it was impossible to reach a significant percentage of the installed software base. Consequently you will find copies of this letter attached should you know of any FORTRAN user who did not receive theirs in the mail.

The letter describes a remarkably small number of problems we felt the user should know about. Most serious is the absence of certain information not provided by the FORTRAN compiler to Pascal to support intrinsic units. To remedy this situation a program listing of a routine called FORTFIX was supplied. It takes less than half an hour to enter and compile the program. If, however, this is an inconvenience to your customer, we offer two alternatives. First, the FORTFIX code file will be distributed by International Apple Core member user groups. Secondly, Apple will make available to you, on request, a copy of the same diskette.

Dear Apple Language System Owner:

We are mailing this direct letter to inform you of the recently upgraded Apple Pascal software and documentation, now available from Apple Level 1 Service Centers. Not only have many problems reported in the original version of Pascal been corrected, several new features have been added. In particular:

- A chaining facility allows one program to tell the system which program to execute next. A string of up to eighty (80) characters can be passed between the programs.
- A new option at the command level allows you to M(ake an EXEC file -- saving console input to be subsequently executed. This allows the powerful capability of using a file to initiate any sequence of tasks you would normally have to enter from the keyboard. These files can be of any length.
- New I/O routines give faster disk access.
- Larger programs can be compiled.
- An operating system swapping option allows you to maximize the space available in the Apple's memory. With swapping on, the maximum user program space is increased by an extra 1100 words.
- Sixteen regular segments are now available to the user program -- six more than the previous version. Up to 26 segments are available if 10 of these segments are intrinsic units.
- Regular units can now use intrinsic units.
- A capability is offered to shift the Apple keyboard into lowercase (and back into uppercase) at any time, and to cause uppercase letters to be displayed on the screen in reverse video to distinguish them from lowercase letters.
- The Editor has a S(ave to original file name option on exit. This minimizes the number of disk Krunches needed to consolate files.
- All code files generated on a previous Apple Pascal 1.0 system will run on the new 1.1 software without recompilation. Text files are likewise compatible. Please note that 1.0 programs and device drivers that PEEK and POKE parts of system memory may not work with the 1.1 revision.

Advanced users who have an indepth knowledge about Pascal I/O will be glad to hear of an automatic method of configuring additional device drivers. This method requires that drivers be written following certain rules and incorporated using the special programs ATTACHUD.CODE and SYSTEM.ATTACH. For this revision, users interested in the Attach feature can obtain complete documentation through International Apple Core member user groups.

All Language System owners will be interested to know that two new manuals, the Apple Pascal Operating System Manual and the Apple Pascal Language Reference Manual, replace the original white-cover reference manual. These two new documents provide detailed examples and information on using the Pascal language and its operating environment. In addition, a third manual, the nine-page Apple Pascal Update, summarizes differences between the original product release and the new revision. If you would like to see more information on the new revision, each dealer has been sent a copy of this Update Manual.

The Pascal 1.1 Revision Kit contains four Pascal diskettes (APPLE0, APPLE1, APPLE2 and APPLE3) along with the manuals mentioned above and their addenda. If your Language System is covered by either 90-day or Extended Warranty, then the Revision Kit is yours free!

SPECIAL NOTE: Apple will consider any Language System purchased on or after August 1, 1980 as being under warranty for this revision only.

If you are under warranty, simply take proof of purchase date or Extended Warranty number to the nearest Level 1 Service Center and place your order for Service Part Number 652-0196. If you are not under warranty, there will be a \$60 charge for the Revision Kit.

In order to receive the Revision Kit, your original four Apple Pascal 1.0 diskettes must be returned. Please do NOT send any diskettes to Apple. Your Level 1 Service Center will be prepared to handle replacements most efficiently.

If you have any questions, see your Apple Dealer.

February 1981

Dear Apple FORTRAN User:

David S. Escoffery  
DAVID S. ESCOFFERY  
Product Manager, Languages

As befalls any software of this scope and magnitude, "bugs" tend to surface after initial market introduction--no matter how extensive a test program was instituted. Listed below are several such problem areas we have identified and feel you should know about. Most of these errors arise only when combining FORTRAN modules with Pascal modules. Please note that although the last problem listed (and described in detail) applies primarily to FORTRAN code files which are compiled and executed under the 1.1 revision of Pascal, we still recommend that you apply the FORTFIX program to "all" FORTRAN code files.

1. The first problem is more documentation related than anything else. Whereas an asterisk (\*) specifies the Apple console to FORTRAN Input/Output operations, its use within the Pascal Operating System is entirely different. To be specific, and depending upon circumstance, an asterisk may be used in the Pascal Operating System in file size specification, in specification for SYSTEM LIBRARY and, most commonly, as a means to specify the volume name of the system or boot diskette.

WHEN USING THE FORTRAN COMPILER, DO NOT TYPE AN ASTERISK (\*) IN RESPONSE TO THE "LISTING?" PROMPT. The system will appear to compile, but will damage your FORTRAN disks and severely cripple the operating system. If you do want the listing to appear on your video monitor, then the proper response should be "CONSOLE:" not an asterisk.

2. A FORTRAN host program calling a Pascal real subroutine with a VAR parameter may cause a Stack Overflow.

3. Pascal intrinsic units will not work with FORTRAN unless they have unit numbers greater than 15. If you want to have your FORTRAN program use Pascal intrinsic units, you must remember this.

4. Certain information needed by Pascal is not put into the code files generated by the FORTRAN compiler. WITH THE 1.1 PASCAL SYSTEM, USER DEFINED PASCAL INTRINSIC UNITS WILL NOT WORK WITH FORTRAN, NOR WILL BUILT-IN PASCAL INTRINSIC UNITS WORK ALL OF THE TIME WITH FORTRAN, UNLESS THE FOLLOWING POST-PROCESSOR IS APPLIED TO THE CODE FILES.

The attached program listing (PROGRAM FORTFIX) should be typed in and compiled by the Pascal compiler. Save the executable code file. Then after compiling any FORTRAN unit or host program, run FORTFIX on the resulting code file. Even if the code file does not use any intrinsic units or was compiled under Pascal 1.0, still run FORTFIX on the program. This will fill in the necessary missing information in the code files.

When you execute FORTFIX, it will first ask you for the file name. You should type in the name, including the volume the file is on and suffix .CODE, of the FORTRAN host code file or unit you want to fix. If there is a problem opening the code file, you will be given an error message.

It will then ask you if the FORTRAN code file uses any intrinsic units. If you answer no, by typing an 'N', FORTFIX will write out the corrected portion of the code file and give you a completion message. If you answer yes, by typing a 'Y', FORTFIX will ask you to enter the intrinsic unit numbers which the code file uses. The system intrinsic units have the following unit numbers:

TURTLGRAPHICS	20 & 21 (It has a code and a data segment)
APPLESTUFF	22
CHAINSTUFF	28
TRANSCEND	29

If you use any Pascal intrinsic units other than those supplied by the system, enter their numbers as well. The intrinsic unit number is the number following 'CODE' in the intrinsic unit definition in the Pascal source to the intrinsic unit. If the intrinsic unit has a data segment, the number following 'DATA' must also be entered. Type '0' when you are done and FORTFIX will finish its processing.

please be assured that each of these problems will be fixed in a forthcoming FORTRAN maintenance release. In the mean time, it does not take long to enter and compile the FORTFIX program. However, if this is inconvenient and you can afford the time, arrangements may be made with your local dealer, as Apple will be making the FORTFIX code file available to dealers who request it. Additionally, International Apple Core member user groups will be sent copies of the code file.

FORTFIX Program

```

(*$I-*)
PROGRAM FORTFIX;
CONST
  MAXSEG=31;
  MAXSLOT=15;
TYPE
  SEGRANGE=0..MAXSEG;
  SEGDIRANGE=0..MAXSLOT;
  MTYPES=(UNDEF,PCODEMOST,PCODELEAST,PDP11,M8080,
    Z80,GA440,M6502,M6800,TI9900);
  REVISIONS=(NONAPPLE,ONEZERO,ONEONE,FUTURE1,FUTURE2,FUTURE3,
    FUTURE4,FUTURE5);
  SEGSET=SET OF SEGRANGE;
  SEGDIRREC=RECORD
    FILLER1:PACKED ARRAY[0..255] OF CHAR;
    SEGINFO:ARRAY[SEGDIRANGE] OF
      PACKED RECORD
        SEGNO:0..255;
        MACHTYPE:MTYPES;
        FILLER:0..1;
        MAJORREVISION:REVISIONS;
      END;
    INTSEGSET:SEGSET;
    FILLER2:ARRAY[0..109] OF INTEGER
  END (* SEGDIRREC *);
VAR
  SEGDIR:SEGDIRREC;
  F:FILE;
FUNCTION YESNO:BOOLEAN;
  VAR CH:CHAR;
BEGIN
  REPEAT
    WRITE(' Y(es or N(o:');
    READ(CH);
    Writeln
  UNTIL CH IN['Y','N','y','n'];
  YESNO:=CH IN['Y','y']
END;
(* To keep FORTFIX short, all errors abort *)
PROCEDURE ERROR(MESSAGE:STRING);
BEGIN
  Writeln;
  WRITE('==>ERROR ',MESSAGE);
  WRITE(' RETURN to abort:');
  READLN;
  EXIT(FORTFIX)
END;
PROCEDURE INIT;
  VAR FCFNAME:STRING;
BEGIN
  WRITE('Name of Fortran code file:');
  READLN(FCFNAME);
  RESET(F,FCFNAME);
  IF IORESULT<>0 THEN
    ERROR('opening code file');
  IF BLOCKREAD(F,SEGDIR,1,0)<>1 THEN
    ERROR('reading the segment dictionary');
END;

```

27

```

PROCEDURE FIXMAJVER;
  VAR SLOT:INTEGER;
BEGIN
  WITH SEGDIR DO
    BEGIN
      FOR SLOT:=0 TO MAXSLOT DO (* Fix major-version problem *)
        WITH SEGINFO[SLOT] DO
          IF (SEGNO<>0) AND (MACHTYPE=UNDEF) AND (MAJORREVISION=NONAPPLE) THEN
            BEGIN
              MACHTYPE:=PCODELEAST;
              MAJORREVISION:=ONEZERO
            END;
          END;
        END;
      END;
END;
PROCEDURE FIXINTRINSEGSET;
  VAR IUNIT:INTEGER;
BEGIN
  WITH SEGDIR DO
    BEGIN
      Writeln('Does this FORTRAN code file use any');
      WRITE('intrinsic units ');
      IF YESNO THEN (* Fill in INTSEGSET *)
        BEGIN
          IUNIT:=7;
          Writeln('Enter unit numbers in the range7..31');
          Writeln('Enter a number out of that range when you are done');
          Writeln;
          INTSEGSET:=[];
          WHILE IUNIT IN [7..31] DO
            BEGIN
              WRITE('INTRINSIC UNIT #:');
              READ(IUNIT);
              IF IORESULT<>0 THEN
                ERROR('reading an intrinsic unit #');
              IF IUNIT IN [7..31] THEN
                INTSEGSET:=INTSEGSET+[IUNIT]
            END
          END
        END
      END
    END
  END;
PROCEDURE FINISHUP;
BEGIN
  IF BLOCKWRITE(F,SEGDIR,1,0)<>1 THEN
    ERROR('writing segment dictionary');
  Writeln;
  Writeln('FORTFIX finished correctly')
END;
BEGIN
  INIT;
  FIXMAJVER;
  FIXINTRINSEGSET;
  FINISHUP;
END.

```

 **apple computer inc.**  
 10260 Bantley Drive  
 Cupertino, California 95014  
 (408) 996-1010

*David S. Escoffery*  
 DAVID S. ESCOFFERY  
 Languages Product Manager

-----  
 WASHINGTON APPLE PI  
 MAIL ORDER FORM  
 -----

Washington Apple Pi now has a program library, and disks are available for purchase by anyone. The price to members is \$5.00 per disk and \$8.00 to non-members. These disks are chock full of exceptional programs - the utilities are especially useful. The games are some of the best - not just simple and uninteresting ones. You may pick them up at any meeting or have them mailed for \$2.00 per disk additional. (If you order five or more the additional charge will be \$10.00 total.) They will come in a protective foam diskette mailer.

PROGRAM DISKETTES

Members:           \$5.00 picked up at meeting  
                   \$7.00 mailed to you (for the first five, remainder at \$5.00)

Non-members:   \$8.00 per disk picked up at meeting  
                   \$10.00 mailed to you (for the first five, remainder at \$8.00)

- |           |                    |     |             |                      |     |
|-----------|--------------------|-----|-------------|----------------------|-----|
| Volume 1  | Utilities I        | ( ) | Volume 29   | Utilities VIII       | ( ) |
| Volume 2  | Utilities II       | ( ) | Volume 30   | Games X              | ( ) |
| Volume 3  | Games I            | ( ) | Volume 31   | Plot Utilities       | ( ) |
| Volume 4  | Games II           | ( ) | Volume 32   | Games XI             | ( ) |
| Volume 5  | Games III          | ( ) | Volume 33   | Accounting           | ( ) |
| Volume 6  | Games IV           | ( ) | Volume 34   | Solar Tutor          | ( ) |
| Volume 7  | Games V            | ( ) | Volume 35   | Garden Management    | ( ) |
| Volume 8  | Utilities III      | ( ) | Volume 100  | DOS 3.3 Utilities A  | ( ) |
| Volume 9  | Educational I      | ( ) | Volume 180  | Dungeon Designer     | ( ) |
| Volume 10 | Math/Science       | ( ) | Volume 181  | Beginner's Cave      | ( ) |
| Volume 11 | Graphics I         | ( ) | *Volume 182 | Lair of Minotaur     | ( ) |
| Volume 12 | Games VI           | ( ) | *Volume 183 | Cave of the mind     | ( ) |
| Volume 13 | Games              | ( ) | *Volume 184 | Zyphur Riverventure  | ( ) |
| Volume 14 | IAC Utilities IV   | ( ) | *Volume 185 | Castle of Doom       | ( ) |
| Volume 15 | Games VII          | ( ) | *Volume 186 | Death Star           | ( ) |
| Volume 16 | Utilities V        | ( ) | *Volume 187 | Devil's Tomb         | ( ) |
| Volume 17 | Graphics II        | ( ) | *Volume 188 | Caves of Treas. Isl. | ( ) |
| Volume 18 | Educational II     | ( ) | *Volume 189 | Furioso              | ( ) |
| Volume 19 | Communications     | ( ) | *Volume 190 | The Magic Kingdom    | ( ) |
| Volume 20 | Music              | ( ) | *Volume 191 | The Tomb of Molinar  | ( ) |
| Volume 21 | Apple Orchard      | ( ) | *Volume 192 | Lost Island of Apple | ( ) |
| Volume 22 | Utilities VI       | ( ) |             |                      |     |
| Volume 23 | Games VIII         | ( ) |             |                      |     |
| Volume 24 | Games IX           | ( ) |             |                      |     |
| Volume 25 | Utilities VII      | ( ) |             |                      |     |
| Volume 26 | Stocks/Investments | ( ) |             |                      |     |
| Volume 27 | Math               | ( ) |             |                      |     |
| Volume 28 | Planetfinder       | ( ) |             |                      |     |

\*Vol. 181 required with these disks.

-----  
 TOTAL ORDER = \$                   -----

Check here if you want these shipped---

NOTE: PLEASE ALLOW 6 - 8 WEEKS FOR MAILING.

NAME                   -----  
 ADDRESS               -----  
 CITY, STATE, ZIP      -----  
 TELEPHONE             -----

Membership No.(1st three digits after WAP on mailing label) -----

Make checks payable to "Washington Apple Pi"

Send order to:           Washington Apple Pi- ATTN: Librarian  
                           PO Box 34511  
                           Washington, DC 20034



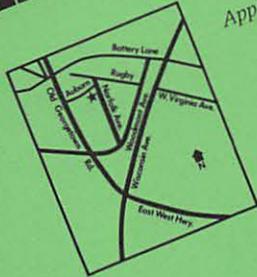
**JUST OPENED**



We are pleased to announce the opening of **Bethesda Computers**, to specialize in the sale of microcomputers, software, peripheral equipment & accessories for use in home, business and educational environments.

Major lines will include Apple, Atari and Hewlett Packard. Please drop by for a

"hands-on" demonstration and professional analysis of your computer needs



Hours  
10 am - 6 pm Monday thru Saturday  
Open till 9 pm on Thursday  
8020 Norfolk Ave., Bethesda, MD 20814  
(301) 657-1992