Washington Apple Pi

$2

Highlights

6502 ARCHITECTURE
A BEGINNER'S GUIDE TO THE
DOS SYSTEM MASTER DISK
REVIEW OF THE PASCAL DISKS
AT YOUR FIRST PI MEETING

In This Issue

ComputerLand®
Tysons Corner

apple computer
Authorized Dealer

8411 Old Courthouse Road
(at Route 123)
Vienna, Virginia 22180
Sales (703) 893-0424
Service (703) 893-0114

FALL TUTORIALS !

1. AN INTRODUCTION TO THE APPLE: MICROCOMPUTER FUNDAMENTALS 9-12:15
*   Including: Binary, Hex and ascii systems*
*   Bits, Bytes, nibbles, rams and roms*
*   The Memory Map and the Monitor*
*   Basic Programming*
*   The Disk Operating System - the CATALOG, VTOC, DOS*
*   and memory usage*

   AND

2. APPLESOFT BASIC PROGRAMMING   1:15-4:30
*   Including: The BASIC Language, variables and program control*
*   Input and Print*
*   Limitations and Restrictions*
*   Reading and Writing Files, EXEC'ing*
*   Graphics, Music and Miscellaneous*
*   Changing Program Control*

The tutorials will be held on two consecutive Saturdays, October 9
and 16 at the Washingtonian Motel on Shady Grove Road in South
Gaithersburg. There will be a limit of 24 participants at each
tutorial. The tutorial will be taught by David Morganstein.

If you wish to attend one or both tutorials, please fill out the
form below. Enclose a check or money order made out to L.
Morganstein Consultants for the appropriate fee. You will receive
confirmation of registration and a list of suggested manuals and
articles to read within two weeks. Mail form to:

                    L. MORGANSTEIN CONSULTANTS
                    13329 Woodruff Court
                    Germantown, Maryland 20874

_____

NAME _____ HOME PHONE # _____

ADDRESS _____ WORK PHONE # _____

        _____ WAP #   _____

I will/will not bring and be responsible for an Apple, Disk Drive
and Monitor.

(Pls. Ck.) _____ Intro. to the Apple _____ Applesoft Basic

Each tutorial - with computer - WAP member $45. Nonmember $60.
             without computer   "   " $60.   " $80.

Please feel free to let us know if there are topics you would like
to see included in the tutorials that would help you be able to get
more use from your Apple, or use it in more creative ways.

OFFICERS & STAFF

President	-David Morganstein	(301)	972-4263
Vice President	-Dana Schwartz	(301)	725-6281
Treasurer	-Edward Myerson	(703)	751-7997
Secretary	-Paula Benson	(202)	546-0076
Members-at-Large	-Bernie Benson	(202)	546-0076
	-Bruce Field	(301)	340-7038
	-Nancy Philipp	(301)	924-2354
	-Bernie Urban	(301)	229-3458
	-Jesse Wagstaff	(301)	937-4215
	-Rich Wasserstrom		
Editor	-Bernie Urban	(above)	
Associate Editor	-Betsy Harriman	(202)	363-5963
	-Richard Gibson	(703)	521-0800
	-Jane Mason	(301)	365-3888
	-Genevie Urban	(301)	229-3458

Newsletter Staff:

Store Distrbtn.	-Bill Hershey	(301)	588-1992
	-Dave Martin		
Advertising	-Eli Argon	(301)	229-4229

Columnists:

Applesoft	-James T DeMay Jr	(301)	779-4632
Assembly Lang.	-Bill Schultheis	(703)	538-4575
Pascal	-Jim Harvison	(301)	593-2993
Q & A	-Bruce Field	(301)	340-7038
SIGames	-Donn Hoffmann	(202)	966-2616
VisiCalc	-Walt Francis	(202)	966-5742
Group Purchases	-Rich Wasserstrom	(703)	448-0984
	(7:00 - 9:30 PM, Mon - Fri)		
Head Librarian	-Gordon Stubbs	(703)	750-0224

Library Staff: -Ben Acton, Dave Weikert,
 -Van Kozak, Gabor Laufer,
 -Sarah Lavilla, Michael Leavitt,
 -Jerry & Dave Waller

New Disks	-Jill and Vance Giboney		
Disk Documen.	-Boris Levine	(301)	229-5730
	-Tony Untied	(301)	241-8678
Arrangements	-Jim Carpenter	(301)	371-5263
Demonstrations	-Julie Oliver	(202)	543-7408
	-Harry Misuriello	(202)	543-7408
General Counsel	-Jim Burger (Shaw, Pittman,		
	Potts & Trowbridge)		
	daytime -	(202)	822-1093
Membership	-Dana Schwartz	(301)	621-2719
Program	-Ernie Forman	(703)	241-1216
	-Tom Woteki	(202)	547-0984
Rules&Elections	-Bob Platt	(202)	223-1588
SYSOP	-Tom Warrick	(202)	333-4000
Tutorials	-Steve Stern	(301)	881-2543
	-Joe Silverman	(301)	384-1285
Volunteer Coord.	-Boris Levine	(301)	229-5730

SIG Chairmen:

Appleseeds	-David Stern	(301)	881-2543
APPLE ///	-Charles Dow	(301)	460-3911
ASMSIG	-Robert Palus	(301)	776-3075
Business	-John New	(301)	577-7959
CP/M	-Dave Neumann	(301)	776-6133
EDSIG	-Peter Combes	(301)	871-1455
FORTH SIG	-Bill Wurzel	(301)	593-2258
LOGO	-Nancy Strange	(703)	691-1619
NEWSIG	-Bernie Benson	(202)	546-0076
Pascal (PIG)	-Tom Woteki	(202)	547-0984
SIGGAMES	-Jim Eatherly	(202)	232-6046
SIG/DISABLED	-Curt Robbins	ABBS WAP428	
(or write)	8805 Barnsley Court		
	Laurel, MD 20708		
(or phone)	-John Molineaux	(301)	341-7391
Telecomm. SIG	-George Kinal	(202)	863-6489

EDITORIAL

A "special issue" this month and, in my opinion, timely. You will notice elsewhere in this issue David Morganstein's comments on our 2000th member. Many of these good folks are new to 1) the APPLE, 2) computers and 3) the WAP. So how better to serve them than to write introductory articles on a host of items. To whit, Bob Platt's excellent review of Apple, Inc.'s System Master Disk, Bill Schultheis' 6502 Architecture article, a thumbnail sketch of FORTH as presented by Bruce Field and Bill Wurzell and reported by Paul Benson, and the announcement of the new Applications Demo Program by Harry Misuriello. For newcomers to the WAP we have Jim Harvison's Review of the WAP PIG Library Disks, Jim DeMay's review of our monthly happenings at USUHS, and our new Bylaws honed to perfection by Bob Platt.

And lots more, like the excellent Q&A by Bruce Field and the tips on how to best the computer in Wizardry and Cannonball Blitz. For all you new members who are also involved with teaching, take a close reading of the EDSIG and LOGOSIG columns by Nancy Strange. But why waste your time reading this. Check it out for yourself. ↵

EVENT QUEUE

Washington Apple Pi meets on the 4th Saturday of each month at the Uniformed Services University of the Health Sciences (USUHS), Building B, 4301 Jones Bridge Road, Bethesda, MD, on the campus of the National Naval Medical Center. Sales, library transactions, newsletter pickup, etc. are from 8:30 - 10:00 AM. From 9:00 to 10:00 AM there will be an informal "Help" session in the auditorium. The main meeting starts promptly at 10:00, at which time all sales and services close so that those volunteers can attend the meeting.

PLEASE NOTE: Due to scheduling difficulties the October meeting will be on the 5th Saturday, October 30.

Following are the speakers and topics for the next few months:

October 30 - Bob Platt & Dick Hodder
             PILOT
           - Demo-MCPS Science Teachers

November 27- (Tentative) Mark Crosby
             Graphics
           - Demo-Versawriter

December 18- Swap Meet

GENERAL INFORMATION

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

Membership dues for Washington Apple Pi are $18.00 per year, beginning in the month joined. If you would like to join, please call the club phone and leave your name and address, or write to the P.O. Box above. A membership application will be mailed to you.

Subscriptions to the Washington Apple Pi Newsletter are not available. The newsletter is distributed as a benefit of membership.

CLASSIFIEDS

FOR SALE - Olympia ESW102 daisy wheel printer, 17 cps, bidirectional and can select proportional spacing. Three print wheels and manual included. Used less than three months. Must sell. Call Bill Statsky, 363-7159, with best offer. I paid $1500.

FOR SALE - The following APPLE II items:
CPS Multifunction card w/Pascal disk--$135
  (serial, parallel & real-time clock)
VisiCalc 3.3 (3.2 also included)------$125
Wizard & Princess Adventure----------$ 15
Tiny Pascal (Programma)--------------$ 20
Three Mile Island (Muse, orig.vers.)--$ 20
All software is on original disks and in-cludes documentation. Contact Susan Zakar or Joe Zakar, (301) 672-2889.   🍎

MINUTES

GENERAL MONTHLY MEETING

The monthly meeting of Washington Apple Pi, Ltd. was held as usual at USUHS on Aug. 28, at 10 AM. From the editor: there are a few new columnists; more would be appreciated. Office space is still being sought. The president solicited program topics for the monthly presentations. A new SIG on tele-communications gathered for the first time this month. A listing of items for Group Purchase has been printed in the newslet-ter. The September Tutorial will be on personal finance. The August issue of BYTE contains an editorial on Congressional action pertaining to microcomputers. The Executive Board will meet on Sept. 8 at the Urbans with an agenda including the Bylaws.

The presentation was given by Bruce Field and Bill Wurzell on the language FORTH.

SUMMARY OF SEPTEMBER WAP BOARD MEETING

The Executive Board of Washington Apple Pi, Ltd. met on September 8 at the home of the Urbans. The items discussed include: a proposal for Apple floor mats, computer faire, hands-on demonstrations at meetings from 9-10:00, structure of general meeting - no SIGS during presentation, rules for ABBS, office plans, request for applicants, request for help to train teachers, editor is encouraged to reject advertising for selling public domain disks for profit. Bylaws of incorporation were examined and accepted.   🍎

JOB MART

HELP WANTED

Position open at Naval Medical Research Institute, Bethesda, MD. GS-7/9 pro-grammer/biomedical technologist. Work involves operating cell sorter and inter-facing to various computers. Contact Dr. C.P. Reynolds, (202) 295-5828 or 295-1121.

High school student who is proficient in Basic programming and wants to earn extra money. Contact Judy Priven, (301) 869-3728.   🍎
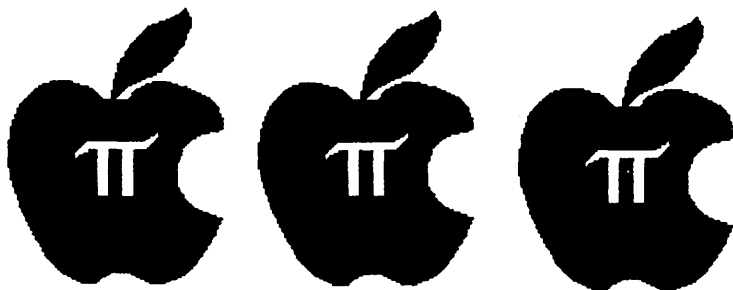
I'm attached to my Apple

SIG-NEWS

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

PIG, the Pascal Interest Group, meets on the third Thursday of each month at 7:30 PM at the Uniformed Services University of the Health Sciences, Bldg. A, Room 2054 (2nd floor), on the campus of the National Naval Medical Center at 4301 Jones Bridge Road, Bethesda, MD.

EDSIG - the education special interest group - will meet on Tuesday, October 12 at 7:30 PM in Lecture Room A, Building A, USUHS. For details of this and other meetings, see the EDSIG page elesewhere in this issue.

ASMSIG meets immediately after the regular Washington Apple Pi meeting.

The APPLE /// SIG meets on the second Thursday of the month at 7:30 PM. The meeting place alternates between the Walter Reed Medical Center and Universal Computers.

NEWSIG will meet just after the regular Washington Apple Pi meeting. We will answer questions and try to help new owners get their systems up and running. We will also explain how our club operates.

The following members have agreed to answer questions over the phone when someone gets stuck and needs help between meetings:

    Bob Chesley   560-0120
    Paul Hoffman   831-7433
    Sarah Lavilla   926-6355
    Boris Levine   229-5730
    Steve Sondag   281-5392

Another SIG has been added - the Telecommunications SIG. For information call George Kinal, 863-6489

LIFE SCIENCE SPECIAL INTEREST GROUP

By Paula Benson

Anyone interested in forming a Life Science Special Interest Group please contact Paula Benson at home, 546-0076. We want to exchange information and explore the possibilities of using the APPLE in clinical lab and research environments.

PRESIDENT'S CORNER

by David Morganstein

2K AND GROWING - I met a member at our August meeting. While I can't remember his name, I'll not forget his number. It was WAP2000! Welcome aboard!!!

AN OFFICE - By the time you read this, we should have selected an Office location. If so, this issue will contain an address and map to get you there, as well as the hours when you can visit. The next step will be to find part-time help and to obtain office supplies and equipment. With tables, chairs and shelving, Jess Wagstaff and Paul Hoffman can begin to get our library into shape. Contact me if you have any suggestions or offerings that can help to get things moving.

PERSONAL NOTES - Many thanks to Jim Graham for his able assistance this past year in supervising the distribution of newsletters to local stores. Jim will be in school this year and unable to continue in this capacity. Bill Hershey, Jim's second in command, will continue to arrange deliveries. He will be joined by Dave Martin in this effort.

Betsy Harriman proved herself an able Editor for the July and August issues, while Bernie & Gena Urban were vacationing. Betsy, your efforts were much appreciated by avid WAP readers.

Dave Weikert who has worked generously on the Disk Library staff (Gordon's Troops) for some time is contributing even more by taking responsibility for assembling and transporting half of the library to the monthly meetings. Thanks, Dave.

LIBRARY EXCHANGE - At the last Board meeting, a formal statement of a library exchange policy was made. This policy has been followed by our current software editors, Vance & Jill Giboney and our previous one (yours truly). However, to insure that Apple clubs around the country are aware of this policy, let me repeat it.

WAP is pleased to exchange disks assembled by existing clubs on a straight one-for-one basis. These exchanges benefit members of both clubs. For new clubs just starting out we will offer a limited two-for-one exchange to help get the ball rolling. If a club has no library but does have a newsletter demonstrating regular activities, we will advance several disks with the understanding that software will be returned in due course. Since our library is one source of revenue for us, we do not routinely "give-it-away" but will gladly trade for new programs generated by other clubs.

TUTORIALS - Keep you eye out for new tutorials to be offered by the WAP. In October, our SYSOP, Tom Warrick, will be holding a machine language tutorial. Tom will be aiming his course at the novice and after a few sessions, he will no doubt turn out some fine machine language programmers!

NEW WAP COLUMNISTS - Bill Schultheis has volunteered to do a column on machine language programming. Bill will alternate articles for the newly initiated with ones for the alchemists amongst us.

Jim Harvison, Pascal librarian, will be adding more Pascal articles to our magazine. (Rumor has it that Dr. Wo may be getting a bit more time to put together some of his much-missed Pascal words of wisdom, as well. Let's hope the rumor is accurate!)

COMPUTER FAIRE - Our first volunteer to organize a Faire, Nick Santelli, was incapacitated and had to withdraw his offer. Let's hope that our second volunteer, Al Weiner, does not suffer a similar fate. Al is thinking that the Faire might become a regular function but first must find a suitable location with ample lighting and power outlets. The Faire would present an opportunity to vendors of Apple related products to advertise and sell. A nominal fee would be charged to vendors and attendees to cover costs of rental or advertising.

USE OF THE USUHS FACILITIES - A reminder that advertising and sales should not be conducted at our monthly meetings. The USUHS has permitted us access to their excellent and much appreciated facilities on the grounds that the WAP is a non-profit organization dedicated to educating Apple owners. It is the WAP, not commercial vendors, who have use of and responsibility for the facilities. If you see someone passing out or taping up advertisements, please ask them to respect our understanding with the USUHS. Encourage them to call Al Weiner and help organize the Faire.

DEMOS AT MONTHLY MEETINGS - Starting with the October meeting, we hope to have regular monthly demonstrations of software and hardware products. These demos will be conducted in the cafeteria during the 9:00 to 10:00 hour. The October demos will be provided by the Montgomery County school system and will feature novel "home brew" equipment for demonstrating scientific principles. Julie Oliver and Harry Misuriello will be organizing the demos and would like your help in giving demos or suggesting products.

IMPORTANT NOTE: these demos must be given by purchasers of products and not by vendors!!!   ⑤

###############

The October meeting will be on the 5th Saturday, instead of the 4th. The date is October 30. Mark it on your calendar.

###############

A PAGE FROM THE STACK
by Jill and Vance Giboney

This month we're adding two new disks to the library. The first is Library Disk 121 - ABBS. This disk contains John Moon's ABBS program, which served as WAP's ABBS until Tom Warrick's recent modifications and enhancements. Note that this is not a turnkey system and must be configured for your system and purposes. To aid anyone interested in doing this we're also including on the disk John Moon's article "Write Your Own Bulletin Board System" from the May 1982 newsletter.

The second disk we're adding is the latest from the IAC, another disk of miscellaneous software, IAC #17, which is WAP Library Disk 122. This one doesn't seem to be a contribution from a single user's group and, in fact, contains many programs that have long been in the WAP library (and some more than once, we have to admit). A listing of the contents of IAC #17 is:

AUTOMATIC BINGO	BATTLE OF NUMBERS
BEGINNING	BINGO
BINGO CARD	BLACK JACK
COLOR MATH	COMPUTER ART
CRYPTOGRAM	DATA DISPLAY AND SAVE
FAST MEM TEST	HANGMAN
HANGMAN 2	HELLO
INSTRUCTIONS	LAS VEGAS SLOT MACHINE
LIFE	MOON LANDER
MUSIC	NEVADA CRAPS
NJ LOTTERY	PHILA. COLOR ORGAN
PHILA. PINBALL	RANDOM LINES
SKUNK	SLOTTS
SLOW MEM TEST	STARWARS RESCUE
STATES AND CAPITALS	THE HART PIANO
TITLE DEMO	TWINE (TYPE GR FIRST)
TWO CIRCLES	WINGS (TYPE GR FIRST)

And while we're on the subject of the IAC we'd like to report that after rumors of an IAC library of over 800 disks filtered back from Applefest we contacted the IAC librarian to see how and if we could get hold of some of these disks. We've now been informed that the entire 855+ disk library of the IAC is THEORETICALLY available to us. The only way we can obtain an IAC disk, however, is to trade them a disk that isn't in their library. This means, of course, that only a small fraction of the library is actually accessible to us (and to other user groups) since even the most optimistic of us have doubts about our chances of coming up with 800 new disks of public domain software. We will, however, exchange disks with the IAC as much as we can, as if they were another user group. This means, of course, that your contributions are especially useful since they not only add to our library but allow us to exchange our library disks for disks from other libraries.

On behalf of all of us (library staff and the club as a whole) we'd like to thank all those who have been contributing programs for the library.

(Ed. Note: See also comments re IAC disk library in "IAC Corner" elsewhere in this issue.)   ☜

TIDBITS FROM NEWSIG
by Bernie Benson

Welcome to all the new members who have joined WAP over the summer. I think I know how you will be spending a lot of time this fall. I was quite busy with my APPLE this summer but unfortunately not using it to write this column. I hope to do better. The club had a very good summer: elections, membership over 2000, new special interest groups, and possible office space, to name a few highlights.

The purpose of this column is to provide useful information to the new home computer owner. I have owned an APPLE II for two years now and have worked with computers for much longer. I am not in touch with many of the small annoying problems the new computer owner faces. Therefore, I would like to invite new APPLE owners to write to this column describing any problems or situations you encountered as a new user, and any solutions or recommendations you came up with for those problems.

Just a couple of lines may be a great help to someone buying their first computer just before or after you did. The simplest solutions are often the best. I will see that any mail addressed to this column sent to the WAP P. O. Box is included in a future newsletter.

Remember that questions should still be sent to the Q & A column, and you are encouraged to do so. I must say again to the new members that the newsletter and the club works best when people of all backgrounds and experiences share ideas and information. I think there is a lot we can learn from each other.   ☜

DEALER'S CORNER

Computers Plus, Inc., located near Springfield Mall in Northern Virginia, is offering a variety of classes in September and October on the APPLE. These include Orientation to the System, VisiCalc, Basic Programming and Wordstar. Call Nancy Kelly, Training Coordinator, at 922-7850 for details and to register.   ☜

SIGAMES NEWS

by Donn Hoffman

Sigames is the Washington Apple Pi special interest group concerned with two of the most popular applications: games and graphics. We meet every month at 10:30, during and following the regular WAP meeting in Building C, Lecture Room D of USUHS. All are welcome. The agenda is flexible; the meeting usually starts with a discussion session where attendees trade tips and field queries. This is followed by the main presentation (if there is one) after which newly released games are reviewed. At some point during every meeting, we play several minutes of a continuing adventure – Ulysses and the Golden Fleece.

LAST MONTH

In August Sigames was visited by representatives of Avalon Hill. They displayed their wares and entertained questions. I was, unfortunately, out of town, but I have been told that they gave a splendid presentation and pledged to return sometime after Christmas. The response to the appeal for volunteers to assist in the planning and execution of the monthly meeting was exceedingly poor. One would not expect such anomie from what is usually such an enthusiastic group. Well, there's still time!! If you would like to give a presentation/ lecture or demonstrate your favorite game, call Jim Eatherly at (202) 232-6046. Your active participation is vital!

THIS MONTH & BEYOND – * CONTESTS *

This is a kickoff month for two events: Jim Eatherly will begin his monthly series on APPLE graphics and animation, and the first round of the arcade-game tournament will be held. The contest will run for three months – September, October, and November. Each month players will test their gaming prowess against that of their opponents in an unannounced game of speed, action, and dexterity. Players will be allowed three minutes of play apiece, their place to be determined by a combination of raw score and level attained. There will be two entrance categories : ages 8 to 12 and 13 to 16. One prize (a copy of the game played) will be awarded each month to the champion of the respective age groups. In November, a grand prize will be awarded – the game Time Zone. Many thanks to Columbia Computer Systems for their generous donation of the contest prizes.

For our more cerebral members, there exists the programming competition. You are challenged to create (with the help of a Hi-res animation package – e.g. The Arcade Machine, Graphics Magician) an arcade-type game that embodies all the qualities that make a great vid game : an original scenario, challenge (and the chance to win) at all levels, attractive graphics, etcetera. All submissions will be donated to the club library and form the first Sigames diskette in a long while. To the winner of the programming contest : an as yet unannounced graphics utility package.

** T * I * P * S **

As promised, TIPS is back with Wizardry Character Enhancement Techniques. To take advantage of the following procedures, you will need : 1) a Wizardry diskette, 2) a nibble copier (Locksmith 4.0, Nibbles Away][) 3) a sector editor (Zap, NA][, Inspector). All this equipment will permit you to change your character's attributes where they are stored on the disk. If you have the new version of Wizardry, simply use the M(ake scenario utility option. The FIRST and MOST IMPORTANT step is to MAKE A BACKUP !!!! Do not use your original scenario diskette. With The Locksmith 4.0 the procedure is as follows :

        TRACKS 0-9 NORMAL
        TRACKS F-22 NORMAL
        TRACKS A-E SYNC : parm 36=01
                    or
     TRACK 0 : parms 36=01, 21=02, 46=96
     TRACKS 1-22 SYNC : parm 36=00

Do not continue until you have made a backup!

1) Create a character with a name nine letters long. This is the character you will edit, and this also gives you a chance to make sure the backup is good.

2) Boot up whatever sector editor you are going to use. Locate your character by his/her name's ASCII values. The first character is stored on track $1D, sector $D. The rest of the characters may be found on tracks $1D, $1E, and $1F. Remember that all byte offsets are given in dec, values in hex.

3) Your character's password is stored starting at byte 8 (using the end of the character's name as the reference point).

4) At byte 24 the character's race is recorded. The values for race follow; any other values will produce 'NORACE'.

    $01 = HUMAN $02 = ELF $03 = DWARF
    $04 = GNOME $05 = HOBBIT

5) The value of byte 26 determines your character's class. These are the values:

    $00 = FIGHTER $01 = MAGE $02 = PRIEST
    $03 = THIEF $04 = BISHOP $05 = SAMURAI
    $06 = LORD $07 = NINJA

6) At byte 30 you will find your character's age.

contd.

7) Byte 32 holds your alignment. Any value other than those given will yield 'NOALIGN-MENT'.

   $01 = GOOD $02 = NEUTRAL $03 = EVIL

8) Change bytes 34 through 40 to $FF and all your character's attributes will be 31.

9) Bytes 42 through 49 hold your cash. Change them all to $FF and your character will have one hundred million, ten thousand and one gold pieces.

10) At byte 122 your level will be found. Change 122 to $FF and byte 123 to $7F and your character will be level 37,767. Do NOT exceed this level.

11) Byte 124 is used to store your hit points. Change it to $FF and byte 125 to $7F to obtain 32,767 hit points. Do the same for bytes 126 and 127, they are the status bytes.

12) The 114th through 117th byte after the name signify your character's experience points. If you wish (for cosmetic or esthetic reasons) change these four bytes to $FF. As your character has already obtained the maximum level of 32,767, actual experience points are meaningless.

13) Your spell book resides from bytes 128 to 163. Change bytes 128 through 135 to $FF and you will be able to cast both mage and priest spells (regardless of class). Starting with byte 136, enter $FF $7F fourteen times (total entry of 28 bytes). This will provide your character with 32,767 spell points for each spell level of both spell classes.

After you have made all the changes DOUBLE CHECK CAREFULLY and write them back to disk. Have fun with your super-humanoid characters !!

These notes are, for the most part, the product of the labors of Jim Reich, Adam Levine, and Jon Feldman, the authors of APPLESOURCE, from which these notes were transcribed.

TIPS will be back again next month with info on how to shoplift from Boltac's Trading Post. If you know any cheats, bugs, or tricks, call me at (202) 966-2616.

(Ed. Note: A reminder that WAP policy does not sanction the use of copiers for illegal purposes.)   ⬥

CANNONBALL BLITZ TIPS
by Jeff Bruner

The following tips will help you survive the third level, which I find the hardest. When you get to the platform just before the one which bounces the cannonballs, wait for a ball to come bouncing by, then quickly turn around (towards the previous platform), back up as close as possible (one foot will be in the air) and stay there. This is a safe spot. As long as you stay as close to the edge as possible, the cannonballs cannot hit you.

Now, as soon as the ball bounces downward, turn around, walk to the left edge, and time your "up" to the next level so that your head is in motion to the left just under the high arc of the approaching bouncing ball. You will not make it the first time, so keep trying and don't get upset. Cannonball Blitz is a very difficult game. I praise Mr. Lubeck for creating the challenging level of difficulty.

Back to where we were, if you're still alive you should be at the next level where there are no safe spots and you will have to do the jump-under-the-arc routine again in order to get to the next level. Once you are there, simply climb the ladder.

(This article was found in The SOURCE and I would like to thank the man who must have spent countless hours perfecting this method, although his identity is unknown.)

⬥

WAP HOTLINE

General

	Robert Fretwell	971-2621
	Dave Harvey	527-2704
	Robert Martin	498-6074

APPLE SSC	Bernie Benson	546-0076

Communications Packages

Data Capture	Howard Simkowitz	882-4645
VISITERM	Steve Wildstrom	933-7728

Compilers

Expediter	Peter Rosden	229-2288

DB Master	Dave Einhorn	593-8420

Disk Drives

(other)	Fred Schulz	223-1397

Games

	Jim Eatherly	232-6046
	Donn Hoffmann	966-2616

Graphics	Bill Schultheis	538-4575

Languages (A=Applesoft, I=Integer, P=Pascal, M=Machine)

A	Peter Combes	871-1455
A,I	Jeff Dillon	422-6458
A	Mark Pankin	370-9219
A,I,P,M	Bill Schultheis	538-4575
A,I,M	Richard Untied	241-8678
P	Robert Fretwell	971-2621

Mem. Expansion	Fred Schulz	223-1397

Modems-Telecom.

	Ben Acton	428-3605
	Tom Nebiker (216)	867-7463
ASCII Express	Dave Harvey	527-2704
Hayes Smartmod.	Bernie Benson	546-0076
XTALK CP/M Com.	Bernie Benson	546-0076

Operating Systems

APPLE DOS	Richard Untied	241-8678
CP/M	Robert Fretwell	971-2621

Paddles	Tom Riley (eve.)	340-9432

Pers. Fil. Sys.	Ben Ryan	469-6457

Printers

General	Walt Francis	966-5742
Anderson Jac.	Bill Etue	620-2103
IDS 460	Jeff Stetekluh	521-4882
MX-80	Jeff Dillon	422-6458
Silentype	Bruce Field	340-7038

Time-Sharing

	Chuck Reinbrecht	299-6810
	Dave Harvey	527-2704

VisiCalc	Walt Francis	966-5742

Word Processors

	Walt Francis	966-5742
Apple Writer	Bruce Field	340-7038
Data Perfect	Leon Raesly	460-0754
Letter Perfect	Leon Raesly	460-0754
Screen Wrtr.II	Peter Combes	871-1455
Supertext	Peter Rosden	229-2288
Word Proc. III	Leon Raesly	460-0754
WordStar	David Inouye	422-8926

IAC CORNER

by Bernie Urban

Three items this time.

● The Apple Tech Notes have been mailed to all current member clubs of the IAC. My copy has a stack of notes over 1 1/2" thick on 8 1/2 x 11" loose leaf pages. The source of this information is Apple Computer, Inc., and is a successor to Apple Answer Books 1 and 2. They realized that a way had to be found to update information generated as a byproduct of their hotline service and responses to other mail and phone inquiries which would obviate the need for replacing entire books every six months. They developed "Tech Notes" as an easily expanded and updated answer to this need.

I found the numbering system somewhat disconcerting at first, but understandable after some study. The notes are organized by major subject and by document. Four digits are assigned to the subject and three to the specific document. Two additional are assigned to page numbers. Thus we have, for example, 3700.000.01 referring to page 1 of document 000 (the index document) of subject 3700 which is DOS. This particular index refers to 34 documents on DOS ranging in subject content from Speeding up I/O, through Hiding Things Above the DOS Buffers and Phone List Printer Problems, to Errata - The DOS Manual.

I found typos and other errors in my copy but am not surprised...these are typical first time problems to be rectified by mailouts of errata and change sheets on a quarterly basis to subscribers to the service. Club members are able to buy copies at the computer dealer of their choice. But remember to first ask for the $7.50 discount coupon from your club IAC representative or other officer.

Both Bob Ramsdell (the other IAC Regional Director, Eastern Region) and I concluded that IAC should initiate a companion volume using the same numbering system which would draw from information gleaned from hotlines run by the member clubs, IAC and wherever else as appropriate. What do you think?

● We really do need to focus on improving communication both ways between member clubs and IAC. Each club has been sent a memo which asks for someone to be designated the IAC liason. That person would be responsible for reporting all IAC matters to his club. S/he would receive all IAC mailings and correspondence including disks. This would of course also cover nominations, elections and other important IAC events. Help us to help you.

I have been asked to help clear up any misunderstandings concerning the IAC software library and policies regarding access to them. Still working on it. Hope to get back to you next month on that. ⚘

APPLICATIONS DEMO
PROGRAM by Harry Misuriello

A new monthly meeting activity to demonstrate hardware and software applications using the APPLE computer will begin in October. These demos will be held in the cafeteria from 9 to 10 A.M. Here's what has been lined up so far:

OCTOBER: Interfacing lab experiments through the Apple game I/O connector. Presented by Montgomery County Public School System Science Teachers.

NOVEMBER: Using the "Versawriter" for fun and profit. Presented by David Morganstein and Harry Misuriello.

The MCPS science teachers (staff to be announced) plan to have 3-5 APPLE systems on hand to conduct physics, chemistry and biology experiments using sensors to measure temperature, motion and light among other things. Remarkably, all this data is gathered through hardware connected to the game I/O connector! Don't miss this one as club members were consultants on the development of the system.

The Versawriter, featured in November, is a relatively low-cost digitizer that is connected to the game I/O. With its two disks of applications software, it can handle almost any graphics task. David Morganstein will show how his firm uses it to draw statistical maps, color code them for data and run them off on a Prism color printer through a Pkaso interface board. Harry Misuriello will demonstrate shape table creation, editing and manipulation, as well as the Versawriter's other software features. Pictures will be run off on a NEC 8023-A printer with a Pkaso interface.

Given enough volunteers and member interest, future Demo programs will demonstrate a wide variety of products such as voice I/O boards, co-processors (6809,8086,etc.), real time clocks, ramcards, printers with interface cards and homebrew concoctions. We also hope to demonstrate more graphics packages, data base programs and other software. Besides learning more about the APPLE (and its seemingly boundless versatility) Pi members will get the opportunity to examine and evaluate a product or application in a more natural setting than a retail store. As with every Apple Pi effort, it's volunteers that make things happen. If you want to demo your favorite application, call Harry Misuriello or Julia Oliver at (202) 543-7408 or contact us at the monthly meeting.

6502 ARCHITECTURE

by William Schultheis

If you want to write Apple Assembly language programs, you have to know how the 6502 processor works. This means that you have to know both the architecture and the instruction set. What does a computer have to do with architecture? Computer scientists use the term "architecture" to describe the component parts of a computer and how they fit together. These are not physical parts like chips and wires but the logical parts of the computer that hold and operate on data. It is much easier to explain what this means by examples, so lets get to the point and start with the Apple.

Any computer, the Apple included, can be considered as having three logical parts. These are the processor, the memory, and the input/output system. In this review we will concentrate on the processor and the memory. In the Apple, the input/output looks like a special feature of memory, so we will not be missing much if we leave it for a later article. You probably already have a general idea of how the Apple memory is organized. Memory consists of 64K (actually 65536) locations, each one of which contains eight bits, a number from 0 to 255, or an ASCII character, depending on how you want to look at it. Not all 65536 of the locations will be used in your Apple, but from the standpoint of the system architecture that doesn't matter.

Each memory location has an address. The addresses go from 0 to 65536. It turns out to be more useful to use their hexadecimal equivalents which go from $0000 to $FFFF. (If you are going to use assembly language you are going to have to get used to thinking in hex, so I am going to use it for the rest of this article without any apologies.)

The 6502 treats memory as 256 pages, each containing 256 memory locations. We refer to the pages by their hex numbers: starting with the zero page and ending with the $FF page. Within each page the memory locations are similarly numbered from $0 to $FF. So the location $803, where most programs start, is position $3 in the $800 page. Some pages have special purposes which we will be learning about. The 256 locations in page zero have some very useful characteristics which we will cover in a later article. The $100 page is used as the stack - to be revisited later. The $200 page is used by the monitor to read input from the keyboard. The four pages starting at $400 are used for the text screen and low resolution graphics. The four pages starting at $800 are used for the second text and lo-res graphics screen if they are being used. Most of the time programs start in the $800 page. Hi-res graphics screen one occupies 32 pages numbered $20 to $3F. Screen two, if used, extends from pages $40 to $5F. If you have a 32K Apple, your useful memory continues up through page $7F; otherwise it goes

right up through $BF. DOS hovers in the top 42 pages of this range. We have now come to the end of RAM, the part of the memory which you can store things into.

The sixteen pages from $C0 to $CF are used for input and output. As far as the processor is concerned it is just normal memory. In practice these locations are wired up to the keyboard, paddles, speaker, cassette I/O, switches controlling graphics and text modes, and devices plugged into the eight peripheral slots.

Memory pages from $D0 through $F7 contain BASIC, and the last eight pages of memory, $F8 to $FF, contain the monitor. If you have a firmware card or 16K memory card, it contains a duplicate set of memory locations for the pages starting at $D0. Since these have the same addresses as regular memory, the processor cannot tell them apart. These alternate memory banks are selected by switches in the $C0 page.

That is enough about memory (probably more than enough) for now. Lets get on to the processor that gives the Apple its smarts.

The processor action takes place in registers. Registers are like memory locations in that they hold data. Most of the 6502 registers are the same size as memory cells, eight bits holding values from 0 to 255. The 6502 operates on data in registers, moves data between registers, moves data between registers and memory, and does a few operations on data in memory. The configuration of registers and memory and their connecting data paths make up the 6502 architecture. (Remember the architecture?)
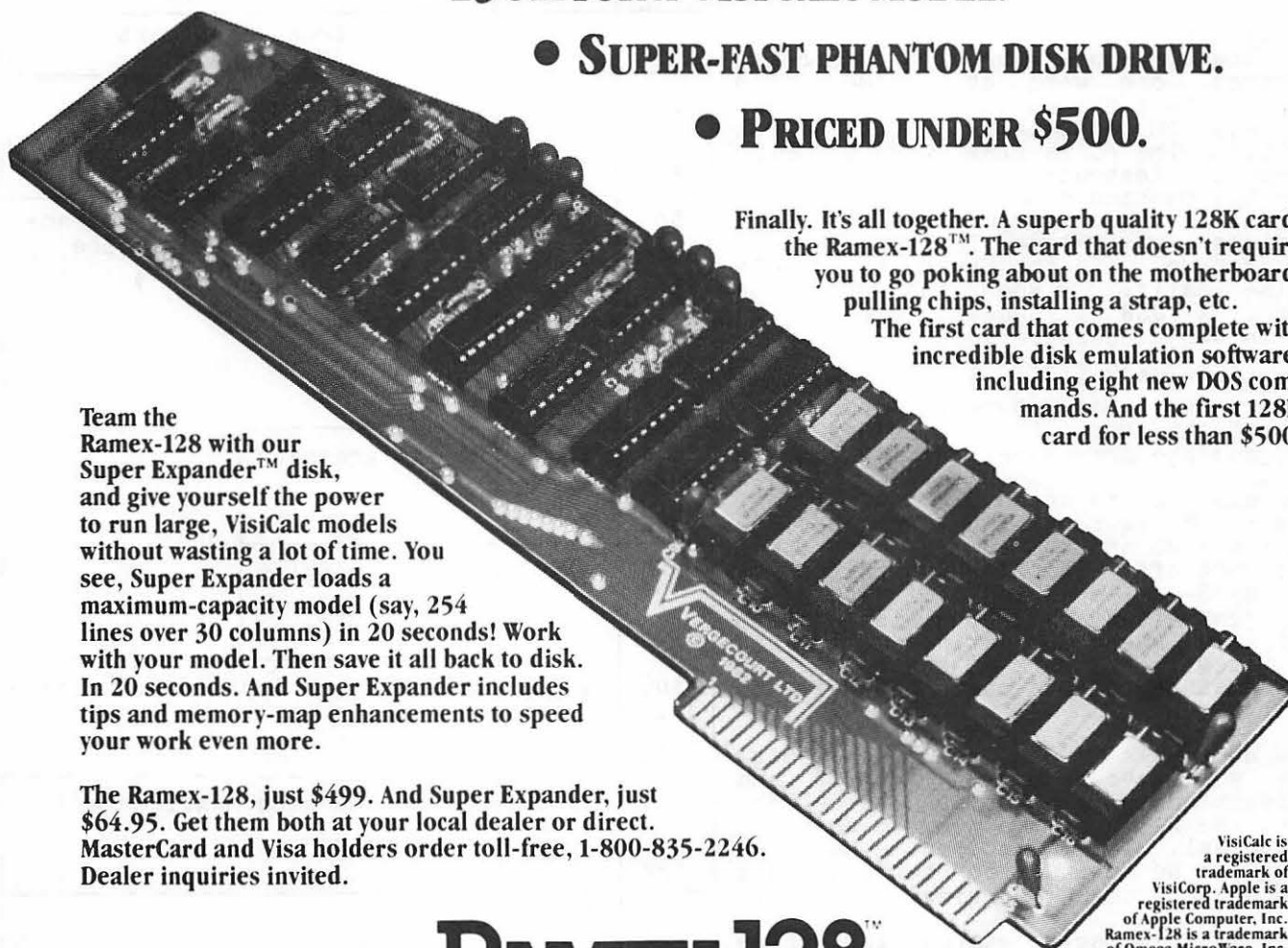
The configuration is pictured in Figure 2. Registers are shown as small boxes. Memory is the big box on the right, and data paths are arrows connecting the boxes. Each register is labelled with its name. The cryptic three letter words in the boxes are the names of instructions which the processor can use to operate on the contents of those boxes. The names beside the arrows are the names of the instructions that move data between boxes. The names in parentheses bring data in from memory but do not change the contents of the registers; they change the value of bits in the P-reg.

Now lets take a quick tour of the registers. The busiest register is the A-reg, often called the accumulator. This is where the 6502 does arithmetic and logical operations. The LDA and STA instructions transfer data between the A-reg and memory. The PHA and PLA instructions transfer data between the A-reg and the part of memory used for the stack. (Remember the stack at $100?) The S-reg automatically keeps track of the next stack location to use. The ADC instruction adds the contents of a memory location to the A-reg. Similarly the SBC,

contd.

AND, ORA, and EOR instructions perform subtraction and logical operations. The result of each operation stays in the A-reg (which is probably why we call it an accumulator.) The ASL, LSR, ROL, and ROR instructions shift or rotate the bits in the A-reg. Bits shifted out of the A-reg wind up in the C bit of the P-reg.

The X-reg and Y-reg are used mainly for counters and for indexing. The LDX and STX instructions and their Y-reg counterparts move data between these registers and memory. The INX, INY, DEX, and DEY instructions add and subtract one from the contents of these registers. The CPX and CPY instructions compare the register contents with a location in memory. (This is necessary in order to check for the end of an index loop.) The TAX, TAY, TXA, and TYA instructions move data between these registers and the A-reg.

The PC is the only 16-bit register in the 6502. The processor works by fetching instructions, one word at a time, from memory. The PC contains the address of the memory location containing the next instruction. The PC is reset when a JMP or Bxx (branch) instruction is executed. (This is the machine equivalent of a GOTO.) The processor then proceeds to fetch instructions from the new locaton. The PC works together with the S-reg to handle subroutine calls. When the processor encounters a JSR instruction the PC is stored in two successive locations of the stack. The S-reg indicates where in the stack to store the data and steps down after each word is stored. Then the address of the subroutine is loaded into the PC, and the processor starts fetching instructions from the new location. When the processor picks up an RTS instruction, it uses the S-reg to pick up the previous value of the PC and the program continues where it left off. (Notice that this works only if the S-reg is at the same setting as it was just after the JSR. This happens automatically when subroutine calls are nested. It will not work if the subroutine pushes something on the stack and fails to remove it.)

We have already said what the S-reg does. Normally you leave it alone. The TXS and TSX instructions will transfer data between S-reg and the X-reg if you need to do something unusual. This sometimes is used so the X-reg can be an index for fetching data from the stack without changing the S-reg.

The P-reg contains seven status bits; these indicate the results of arithmetic operations and other more obscure conditions. The P-reg can be saved on the stack and restored later with the PHP and PLP instructions.

The INC and DEC instructions allow memory locations to be incremented and decremented by one without using a register. Similarly the ASL, LSR, ROL, and ROR instructions handle shifting and rotation. Note that even though no registers are used, the contents of the memory location is copied into the processor, operated on there and stored back again.

This completes our tour of the 6502. I hope it will make it a little easier for you to understand how assembly language works.

MEMORY MAP			
PAGE		USE	
No.	Addr.		
$0	$0	Zero Page - Registers, temp storage, Monitor, DOS, Basic	
$1	$100	Stack	
$2	$200	Monitor keyboard input	
$3	$300	Low part free. High part - Monitor, DOS, Basic	
$4 . . $7	$400	Text and lo-res graphics	
$8 . . $11	$800	Second text & lo-res graphics	Most programs start here ↓
$12 . . $1F			
$20 . . $3F	$2000	Hi-res graphics Page 1	
$40 . . $5F	$4000	Hi-res graphics Page 2	
$60 . . $95	$6000		↑ Applesoft Strings
$96 . . . $BF	$9600	DOS on 48K with Maxfiles = 3 DOS	
$C0 . . $CF	$C000	I/O Pages	
$D0 . . $F7	$D000	Basic ROM	$D000 $D000 Pascal Pascal, BIOS etc. 16K RAM
$F8 . . $FF	$F800	Monitor ROM	card

FIGURE 1

contd.

P-REG

| N | V | | B | D | I | Z | C |

•PHP •PLP

STACK

•JSR •RTS, RTI

PC •JMP, JSR •Bxx

S-REG •TXS •TSX

Y-REG •STY •INY, DEY (CPY)

•TYA •TAY

A-REG •PHA •PLA •STA •LDA
•ASL, LSR •ADC, SBC
•ROL, ROR •AND, ORA, EOR (CMP) (BIT)

•TAK •TKA

X-REG •STX •INX, DEX •LDX (CPX)

MEMORY
O-PAGE
STACK

•INC, DEC
•ASL, LSR
•ROL, ROR
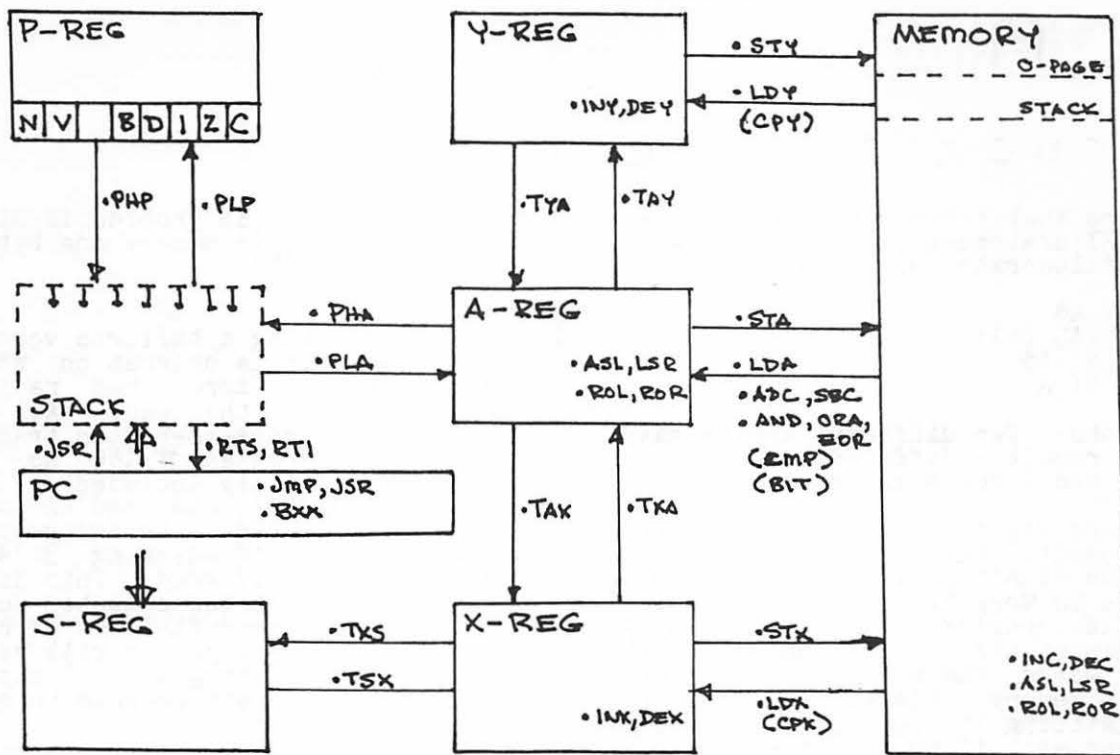
FIGURE 2   6502 ARCHITECTURE

Q & A

by Bruce F. Field

Q. I've heard that there are some catches in the GET statement, and the following seems to illustrate the point:

```
120 GET A$
130 A= VAL (A$)
140 PRINT A$
150 PRINT A
```

Any comments? Two different Apples gave the same result - different values for A$ and A, and I guess yours will too.

A. Well, there may be some catches in the GET statement, but this isn't one of them. You didn't say, but your example only fails to work if you haven't booted DOS. The problem actually is a combination of the way the Applesoft VAL function works and where strings are stored in memory. First, for those of you not sitting in front of your Apples at the moment, if you run the program and enter say 2, A$ is printed as 2, but A is printed as 2.22222222E+16.

The problem is that the string A$ is stored at the very top of RAM memory ($BFFF) if DOS isn't in the Apple. When the VAL function evaluates the string it does so on a character-by-character basis. That is, the first character in the string is checked to see if it is a digit, if it is the second character is temporarily replaced by zero then a routine is called to convert the first character into a floating point number. The zero in the second character acts as a delimiter so only the first character is converted. This process continues for each character in the string.

This method fails when the string ends at the top of memory and a zero cannot be stored in the next location beyond the end of the string. (Although an instruction can be executed telling the Apple to store a zero at $C000 the hardware ignores it.) When the first character of the string is converted to a number there is no zero in the next character to stop the conversion so the routine continues to read successive locations ($C000 to $C00F) which happen to be the keyboard addresses. If a GET statement was used just prior to the VAL function the last number typed in is left in the keyboard, so successive reads to addresses $C000 to $C00F just read the keyboard 16 more times; which is why you get the number 2.22222222E+16. When the keyboard strobe address is "read" the character obtained doesn't look like a number and the string to number conversion process stops. For the same reason using an INPUT statement does work. After the INPUT is executed a carriage return character is left in the keyboard and since this is not a number the number conversion process stops at the end of the string.

The way to fix this problem is simply to move HIMEM down in memory one byte, that is, HIMEM: 49151.

Q. I've been making a tailored version of the great circle program on WAP disk 106, which is more ham radio than scientific, by the way. At several points the degrees-to-radian transformation is entered as $\pi/180$. As in the past, I've usually included a line such as RX = .0174533, and used RX wherever the degree to radian transformation was needed instead of entering 3.14159/180 every time it occurred. This is easier programming, but the computer does need to seek out the definition of RX. My question: which approach will yield the faster running program? (I suppose I could concoct a test program to check it out.)

A. Handwritten at the bottom of the letter was the following - "I DID IT - see attached!", and he did. The answer is that it is faster to define RX at the beginning of the program than to use 3.14159/180. There are two reasons for this. First 3.14159/180 requires that the machine perform a division every time it is used, this takes time. So, it would be faster to use .0174533 instead of 3.14159/180. Secondly it takes Applesoft less time to look up a variable name and find it's value than it does to interpret the ASCII characters .0174533 and convert them to a number. (For this example conversion requires six multiplications.) Also, to increase program speed you should put frequently used variables at the beginning of your program. When Applesoft stores variables it does so in the order in which they appear in the program. When looking for a variable, Applesoft starts at the beginning of the list and searchs for the desired name. If frequently used variables are at the beginning of the list the average search time is reduced.

The author of this question also had a second question, and that is why when he ran the comparison programs he got slightly different answers. He speculated that there are rounding differences. This is true, but the rounding errors are not with the program but with the programmer. 3.14159/180 does not exactly equal .0174533. If this is corrected the answers agree to at least nine decimal places which is the precision of Applesoft.

Q. I recently started to have problems with my Apple. It would flash letters all over the screen, wipe out programs, and once it beeped for five minutes straight. This started to happen when we installed a ceiling fan, a micro-wave

oven, a stove, and a cook-top, all of which are on the ground floor. For now (and when the problems started) the Apple is on the top floor. Could it be that my Micro-wave or something else is interfering with my Apple and what can I do about it?

A. Yes it is possible that other electrical equipment can affect your Apple. If this is what is happening it is probably because something is putting electrical interference (spikes) on the power line. The simplest thing to do in this case is to buy one of the surge suppressors that sell for around $10 and plug your Apple into this.

However we cannot neglect the possiblity that your problems are unrelated to the installation of the new equipment. Poor electrical contacts inside the Apple between the leads on the chips and the chip sockets, and any interface boards and their sockets may also cause strange things to happen. It would be a good idea to gently push down on all the chips (with the power off!) to reseat them in their sockets. Also carefully remove and replace all the boards you have plugged into the expansion slots. This may improve any poor contacts.

As far as the microwave oven affecting the Apple, except for the above mentioned problem of spikes on the power line, there should be no problem. If the microwave field is intense enough to affect the Apple you would be turned into a fried ham hock when using it.

Q. My co-worker keeps tying up our Apple with a inventory control program with a very slow sort program in it. This program is about two years old and I can not see the code so I don't know exactly what it is doing. It takes about 20 minutes to sort 1000 four digit integers into ascending order.

The fastest way I know to do a sort like this is to place the numbers in order one digit at a time starting with the least significant digit and working to the most significant. This approach is fastest if done with the number in binary but it requires the use of an "AND" command on binary numbers. Applesoft has only very weak "AND" and "OR" commands which I do not think will work.

I have been told that CP/M would do this job in a minute or so but this means new hardware and software. I would appreciate your suggestions for doing this in Applesoft and your idea of a reasonable time.

A. First to address your question of using CP/M. CP/M is an operating system and as such cannot be used to do sorts or perform application related tasks. You would have to use a language, such as Basic, that runs under CP/M and rewrite your program in this language. No guarantees that it will run much faster.

Sorts in Applesoft are slow no matter

what algorithm you use. The best way to speed up the sort is to use machine language. Fortunately Alan Hill has written an excellent machine language sort program called Ampersort. This program originally appeared in Micro magazine a few years ago and was recently re-published in the book Micro on the Apple, Volume 2 which is available in the local computer stores. This routine is designed to be used from Applesoft via the & symbol and the user need not know or care how the machine language works. I wrote a test program to sort 1000 four digit integers using Ampersort and it took 76 seconds for the sort.

Q. Are programs for NIBBLE magazine (apparently copyrighted by MICRO-SPARC) ineligible for contribution to the WAP library?

A. Yes, these programs are copyrighted and should not be included in the library. The publishers position is that if the user buys a copy of the magazine and types in the program he has "earned" the right to use it. The publishers have reserved the right to distribute the program in machine readable form.

Q. I have installed a Paymar lower case adapter. I suspect this is causing the space bar to print reverse apostrophes between words or whenever the space bar is hit when using Apple Writer. Other keys, (I believe punctuation keys) are also printing lower case incorrect letters instead of what would be expected. Is this a correctable situation? I also have a 16K RAM card installed. Could this be part of the problem?

A. The RAM card has nothing to do with the problem. Apple Writer writes directly to the screen with some characters that are interpreted differently by the Paymar adapter. It is a simple matter to modify Apple Writer to correct this. Part of the following modification came from a note in the Apple Orchard a few years ago from Lou Rivas. I have included a modification to allow the use of the shift key. The shift key should be modified according to the instructions accompanying the Paymar adapter. Then either the ESC key can be used to enter upper case, or the shift key can be used. One conflict is in entering @, ^, and] which are shift-P, shift-N, and shift-M respectively. To enter these characters use the ESC key to get into upper case mode and then type shift-P, shift-N, or shift-M. The following modification applies only to Applewriter versions 1.0 and 1.1.

```
UNLOCK TEDITOR
BLOAD TEDITOR

CALL -151  (to enter the monitor)

0811:8D 10 C0 4C 48 18
0AE6:20 64 18
1549:20 6B 18
1848:C9 C0 90 09 24 0B F0 05
```

EDSIG NEWS

by Nancy Strange

EDSIG CALENDAR

DATE: October 12, 1982
TIME: 7:30 PM
PLACE: USUHS, Bldg. A, Lecture Rm A
TOPIC: "Turtle Talk" - LOGO Feedback

The October meeting will be a special feedback session by LOGOSIG for EDSIG. All those with LOGO experience are invited to come and share. Anyone who has been using any APPLE version of LOGO is welcome to contact Nancy Strange (691-1619) to discuss this opportunity.

SEPTEMBER MEETING REPORT

The EDSIG meeting was privileged to have C. Dianne Martin return with her co-author Rachelle S. Heller for a more indepth look into computer literacy and the use of computers in education. Their book "Bits 'n Bytes about Computing - A Computer Literacy Primer" - is especially designed for the classroom teacher with little or no computer experience.

As outlined in the August 82 WAP Newsletter EDSIG column, Computer Literacy has two major levels: Awareness of computer impact and Functional Knowledge of computers. Until recently it was considered enough to just be computer literate to the first level, where you did not even have to have access to a computer. However, today the second level with its requirement for computer use is becoming more the standard for computer literacy, in that more and more jobs are requiring the ability to be a computer user (i.e., grocery check-out clerk, analyst, or secretary).

Ms. Martin discussed the three roles of the computer as those found in Robert Taylor's book, "Tutor, Tool, Tutee". It "tutors" with CAI, CMI, and Drill and Practice programs, where previously learned material is reinforced. It acts as a "tool" when it is used as a word processor, editor, statistical analyzer, calculator, and filing system. It plays the "tutee" role when it is instructed by student-written programs and when the student teaches the computer (i.e. LOGO). Each of these were discussed, keeping in mind the differing roles of the three participants (teacher, computer, and student) and why each has its place in education.

Of particular interest was the discussion of an on-going study by HUMMRO regarding the strands of computer knowledge and when each should be taught to children.

Some of the teaching techniques shared were:

A Computer Simulation Game
Human Binary Computer Exercise

Time Line for Computation
Use of Computer for Poetry and Art
Case Study of Computer Installation
Word Processing

Ms. Heller addressed teaching programming techniques and the need for the teacher to be aware of different learning styles of their students. Each child is different and must be taught to enhance the strengths and overcome the weaknesses.

The tips and techniques she shared were:

Present one instruction/command at a time as a tool.
Have a Computer Wall with different commands on each brick as a reinforcement tool.
Create a Class Computer Dictionary and encourage each child to create their own (especially with LOGO).
Have Sample Programs to pass around.
Have and rotate around the class
  THE EXPERT OF THE WEEK
  Honor Position
Schedule the students' use of the computer.
Limit the number of kids on a computer to 2.
Limit the team time to 30 minutes per week (15 min. each).
Allow students to barter their time.
Let each child have a disk to keep own programs.

Each teacher should keep in mind that it is best to not always "teach" the student verbatim what to do on the computer, but rather to lead them through problem solving (ask the student what they are trying to do) whenever possible. Sometimes, this may mean an extra effort of pulling the idea/design from the student and helping the student over the hurdle, but it's worth it. Mathematics and science educators are already stressing that in the 80's it will be the PROCESS of how we solve problems that will be more important than the getting of the END PRODUCT ("the right answer").

The use of the computer in education for these fields, as well as social science, language arts, industrial arts, art, etc., is becoming more and more common-place. How can we help others to become computer literate? What new technologies and software concepts will be created? It looks like an exciting time ahead for the TEACHER and the STUDENT.

DIANNE LORENZ contributes the following:

Classes and Conferences

If you haven't signed up for a class for yourself, your spouse, or your children, here are some last minute ideas, but hurry, as most are beginning very soon. Don't forget the Apple Pi Tutorials which are

contd.

described at the end of this issue.

....LEARNING WORKS: Pick up a schedule at your library or call 657-4488. Short classes on various aspects of computers.

....OPEN UNIVERSITY: Schedule of short courses also available at the library or call 966-9606.

....Montgomery County Adult Education Classes: Taught at various area high schools. Call 942-8304.

....Montgomery College Continuing Education: Most classes held in the evening at the Rockville campus. Call 279-5188.

....ELF: Classes for adults and children - LOGO and BASIC. See September issue for full description or call 530-3299 (day) or 493-9696 (evening).

....The Computer Workshop: Continuing classes and short courses in all aspects of computing. See September issue for details or call 468-2399.

Fall Meetings and Shows

....Mid-Atlantic Computer Show and Office Equipment Exposition: October 28-31 at the Armory/Starplex in Washington. Contact Computer Expositions,Inc. at 800-368-2066.

....Second Annual Symposium on Small Computers in the Arts: Topics include computer graphics and animation, computer-automated sculpture, choreography, designs, and computer-generated music. October 15-17 in Philadelphia, PA. Write POB 1954, Philadelphia, PA 19105.

....Sixth Annual Symposium on Computer Applications in Medical Care. October 30-November 2 at the Sheraton Washington Hotel in Washington, DC. Call Bruce Blum at GWU for more infomation - 676-4285.   ⑤

contd. from pg 17

```
1850:2C 63 C0 10 3C 60
1864:20 78 18 91
1868:28 C8 60 C9 A0 90 06 20
1870:01 15 20 78 18 4C F6 FD
1878:C9 E0 90 02 49 40 C9 C0
1880:90 02 09 20 C9 40 B0 08
1888:C9 20 B0 02 09 40 09 80
1890:60 C9 DE D0 02 A9 CE C9
1898:C0 D0 02 A9 D0 C9 DD D0
18A0:02 A9 CD 48 A9 00 85 0B
18A8:68 60
```

BSAVE TEDITOR,A$803,L$10A8
LOCK TEDITOR

Thanks to Tom Warrick and Eli Argon for help with this month's column.   ⑤

LOGOSIG TURTLE TRACKS

by Nancy Strange

The newest special interest group in Washington Apple Pi, LOGOSIG, was created in July to gather together those WAP members who are interested in LOGO. The LOGOSIG will meet monthly at 12:30 PM on the Saturday of the Wap meeting. Today we will meet in Bldg. A, Lecture Room A, and will be working on a LOGO Project together. This exercise will give everyone an exposure to some of the LOGO language, structures and concepts; problem solving; and project design. We will have four APPLEs and lots of people eager to use LOGO. Anyone who has LOGO for the APPLE or is interested in this new language and its impact on the use of the microcomputer in homes and schools is invited to attend.

Starting with the October meeting the LOGOSIG will change location to the very nearby Bethesda Country Day School at 5616 Beech Avenue, Bethesda, MD, just off Old Georgetown Road, which has better access to APPLEs and a better setting for on-hands opportunities. A map will be provided in the November WAP Newsletter.

There are currently at least three versions of LOGO available for the APPLE: KRELL'S LOGO, TERRAPIN LOGO (MIT Version) and APPLE LOGO (Logo Computer System Version). The LOGOSIG wishes to be a meeting place for all of these - and others to come - so that we might come together, share and exchange information, ideas and procedures. There will be presentations and demonstrations like other SIGs, but we look forward to having as many on-hands LOGO experiences as possible.

The first major activity of LOGOSIG will be to develop a LOGO tutorial for the LOGOSIG members, which should begin in October. The emphasis of the tutorial will be using LOGO and reaching as many different levels of LOGO knowledge as possible, with the more advanced leading smaller groups.

PRINT PROCEDURES FOR APPLE LOGO PICTURE

The following LOGO TOOLS (utility procedures) were obtained from the Computer Learning Center in New York and from LOGOSIG members. If you have successfully dumped LOGO graphics from Hi-res Screen 1 to other printers or by other means, please contact the LOGOSIG so we can share it with others.

FOR THE SILENTTYPE PRINTER:

```
TO PRINTPIC
.PRINTER 1
.DEPOSIT 65536 - 12524 0
.DEPOSIT 65536 - 12528 7
PRINT CHAR 17
.PRINTER 0
END
```

FOR DOT MATRIX PRINTER WITH GRAPHICS INTERFACE (Grappler or Pkaso):

```
TO PRINTPIC
.PRINTER 1
(TYPE CHAR 9 "G CHAR 13)
. PRINTER 0
END
```

Note: - "GD replaces "G for some printers
- SETSCRUNCH 1 will help the appearance of the printout

FOR DOT MATRIX PRINTER WITH SOFTWARE GRAPHICS INTERFACE (Graftrix):

Run LOGO Proc to get picture on screen
.BPT (Puts you in monitor - prompt appears)
Insert graphics dump disk (i.e., Graftrix or Computer Station Enhanced Graphics Pkg Dump)
6 (Ctrl-P) (Return) - (Boots your disk process with normal dump program)

Note: - Unfortunately you can't reenter LOGO without rebooting the LOGO Language Disk.
- Pictures are not as clear as original, some static lines.

(Anyone who can improve this situation is asked to let us know.)

LOGO BUYS

For information regarding special purchases of APPLE LOGO, contact me, 691-1619.

WANTED!!!

1. LOGO ISSUE OF BYTE: (BYTE Magazine's Language Issue, Aug 82, Vol 7, No 8).

Anyone having copies they are no longer interested in keeping are invited to donate them to the LOGOSIG. The LOGOSIG would like to create a lending library of materials for LOGO enthusiasts, and believes that this issue will be very helpful to the new LOGO user.

2. PROCS AND TOOLS: Your created or typed-in LOGO Procedures or Tools (Utility Procedures) are wanted for the LOGO Exchange.

3. BUG WATCHERS: If you have found bugs in any of the LOGO versions please document and indicate version and bug. These may include discoveries as well as problems.

4. TIPS & QUESTIONS: All Tips and Questions regarding LOGO are invited.

5. INFORMATION ON ARTICLES AND BOOKS: If you have discovered articles and books on LOGO and LOGO concepts please let us know. All contributions to the lending library are welcome. They will be put to good use.

6. TRAINING OPPORTUNITIES INFORMATION.

7. INFORMATION ON HARDWARE AND SOFTWARE ENHANCEMENTS.

8.   YOU!!!!!!
        All Ages,
            Trained LOGO Users
              LOGO Teachers
                LOGO Parents
                  LOGO Users
                    LOGO Kids

We will try to share as much of the above WANTED items as we can. Please feel free to contact any of us or me, 691-1619, and remember us when you see L-O-G-O.

MICROCOMPUTER SUPPLIES

MINI FLOPPIES

	Single side/ single density	
SYNCOM "Ectype"		22.00
3M SCOTCH		25.99

	single side/double density	
VERBATIM "DataLife"		28.50
KYBE "Accutrack"		28.50
3M SCOTCH		31.50
SYNCOM "Ectype"		28.50

CARTRIDGE RIBBONS

New film for Ricoh Printer	each	4.75
Recycled Ricoh film ribbons	each	2.95
DIABLO Hytype II black cloth	doz.	44.75
DIABLO High capacity film	doz.	46.50
EPSON MX 70/80	each	9.50
IDS "Paper Tiger"	each	10.75
NEC Spinwriter film or cloth	doz.	66.00
NEC compatible film ribbons	doz.	48.00
QUME Black film or cloth	doz.	33.95

CONTINUOUS FORM PAPER

size	stk no.	wt.	sheets	price
9 1/2 x 11	6910PP	15#	3500	27.25
9 1/2 x 11	6910PP(20)	20#	2700	25.95
9 1/2 x 11	6910HSPP	18#	3000	26.40
14 7/8 x 11	641	15#	3000	32.19
14 7/8 x 11	6410(20)	20#	2400	31.75

MODEM

Lexicon LX-11 300 baud acoustic		145.00
Signalman 300 baud		99.00

ROBINS(Roberts Information Services, Inc.) (703) 560-5900 (800)336-0198
    8306 Hilltop Road Fairfax, VA 22031
(across from the Post Office in Merrifield)
Monday-Friday 9 to 5   Saturday 10 to 2
We ship by Capital Messengers and UPS

Come Forth with FORTH
by Paula Benson and
Bruce F. Field

The August presentation at WAP was on the language FORTH. Bruce Field started off by explaining the language in general. Bill Wurzel continued by speaking specifically on FIGFORTH. Following is a summary of the highlights of the presentation. It is an overview and not intended to be a user's guide to FORTH.

The language FORTH was originally developed for control applications, using integer arithmetic. It is not a numbers cruncher. It is a medium level language ranging between low level Assemblers and high level Fortran. It is easy to learn and program. Its characteristic of being modular is great for developing and debugging programs. Code is written in small pieces (words) and then joined together into a large program. FORTH is a compiled language, so it runs fast. Yet it is interactive too. Another nice feature of this language is that it is extendable. The program progresses by defining new "words" which then become part of the permanent vocabulary. The user can build a class of words tailored to the particular application for which the program will be used.

The flip side of the extendability coin is that there are many different versions of FORTH because of its versatility and ability to add words so easily. FIGFORTH was developed in an attempt to standardize the language. GRAFORTH was developed to converse with the computer about graphics.

FORMAT

Manipulations are done on a push down stack. As numbers are entered, they are put at the top of the stack. Commands are available to move entries around on the stack.

Type	Stack	Action
1	1---	1 is entered
2	21--	1 is pushed down
over	121-	copy 1st to 3rd
3	3121	add 3 to stack
swap	1321	1st with 2nd
drop	321-	rid of 1st entry

Unfortunately, there is nothing to tell what is in a stack. Numbers can be printed or entered in many different bases including hexadecimal. The base must be specified. Sixteen bit arithmetic functions are easily done.

STRUCTURE

There are no GOTO's in this language. Words are defined in terms of previously defined words. Modules (words) are constructed and then combined together. It is a threaded language with a small interpreter. As each word is executed, the computer goes in and finds the code for that word. Only 2 bytes are needed for each word identifying the address where the code can be found. For this reason the language runs fast and only 7K bytes are needed for the basic system. Some systems have an assembler, so assembly language can be integrated into the code.

FEATURES

FORTH is transportable between machines. The native machine code can be redefined in the machine being used and all successive words will run on the new machine. Though Apple DOS has a catalog capability, this is not available when using this language, and text file capability is not supported. It has a rudimentary indexing capability for itself, though. FORTH has a self-contained operating system. Execution time is fast since only integer arithmetic is used. It is 5 to 10 times faster than Applesoft. The FORTH language is easy to learn because it is just a matter of defining words. It has an embedded assembler for the user that wants to code in assembly language.

There are a few problems with this system. There is no floating point, unless specifically designed to include it, e.g. FORTH 79 has this as an extra cost item. There is little error checking at run time. It does not protect the program from the user. The source programs are hard to read without numerous comment statements. Parameters are passed along the stack unseen by the user.

Below is a comparison of various versions of FORTH. The columns indicate whether that version has an editor, assembler, is Apple DOS compatible, has floating point, and price.

COMPARISONS

VERSION	ED	ASM	APCOM	FP	PR($)
Figforth	y	n	n	n	5
Superforth	n	n		n	50
Forth II	y	y	y	n	75
Transforth	y	n	y	only	75
Forth 79	y	y	n	$40	100
Forth V17	y	n	n	n	140
Microspeed	y	y	y	y	495

FIGFORTH

A few comments on this modification of the basic language. Again the code progresses by defining words. Old definitions are used up to where they are redefined, but the same letters can be used to mean different commands. A warning will be given if the word is already in the dictionary. A line oriented editor vocabulary could have an "I" for insert. A Forth vocabulary could use "I" for input. The vocabularies are at different parts of the stack and so are called at different times. VLIST gives a listing of words in the dictionary.

contd.

Words can be entered from keyboard or disk. The latter feature allows debugging of each module, storing on disk, then compiling them into one program. The lines can be up to 64 characters long. Decimal or hex can be used with decimal as the default. The first line of each screen should be used as a comment for what's on that screen.

REFERENCES

Starting Forth - Brody
Byte - Aug. '80
Dr. Dobb's Journal - Sept. '81
Various library disks
Graforth - Aug. '82 WAP   ⑤

LETTER TO THE EDITOR

I read the recent review of the "Bag of Tricks" program in the August issue and wish to relate some of my own experiences with the program.

As a result of that review I went out and bought the program and also sent away for the backup disk. Little did I realize at the time that there was more than one hidden meaning in the title of the program.

When I got around to running the program the disk failed to boot. Instead the screen displayed "LOAD ERROR-REBOOT". I wrote Quality Software about the problem and enclosed my disk. A few weeks later I received a new copy of the program plus the backup copy I had ordered. Both disks failed to boot. My first thought was that it was related to an item mentioned in the current issue of Byte magazine where they explained that a problem with Apple LOGO not booting was traced to a bug in the P6A Prom on the Apple controller card. However, this was checked out and it was not the cause of my problem. Although I have bought many programs lately, most of them on protected disks and have experienced no problems, I decided that maybe the disk drive was out of alignment. So I had an Apple dealer align my disk drive. Brought the drive home - the disks stil wouldn't boot.

I've now taken my drive to another dealer (whose shop has a better reputation) for alignment, and hope this will take care of the problem. This time I am giving them the program disks so they can check the drives after the alignment.

I am relating all of the above to let the potential buyer of "Bag of Tricks" know that if they buy the program they should be prepared to experience the same expenditure of time, dollars and aggravation. I suspect that the protection method used to protect the disk from copying has been a factor, if not the single factor in the failure to boot. As more and more software manufacturers devise new ways to protect their programs, the poor users can expect to have the same or similar problems that I have experienced.

Dave Harvey
⑤

COMMERCIAL LIST by James T. Demay Jr.

I would like to support the businesses and professionals that belong to Washington Apple Pi. However, to my knowledge, there is no readily available information concerning who does belong, what type of business they manage, and where to contact them. To remedy this situation, I am attempting to compile a list containing name and type of business, WAP#, address, and phone #'s of business or professions owned and/or operated by club members. Letters should be addressed to me in care of the club post office requesting to be included in the commercial members list, or leave a message for me (WAP342) on the ABBS.

This proposed list would be a service not only to the businesses, but especially to the membership at large. I for one would feel more comfortable contracting a service from a member rather than a non-member. I realize that this is no guarantee of a good job, but if members were to get a raw deal from one of the businesses listed, the offending organization could be investigated, and if guilty, removed from the list.

The list will be made available to all members who send a self addressed stamped business size envelope to me in care of the club post office box. As always, your comments and suggestions are solicited. ⑤

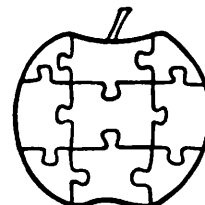# A BEGINNER'S GUIDE TO THE DOS SYSTEM MASTER DISK by Robert C. Platt

We are frequently asked at NEWSIG meetings about the programs on the Disk Operating System (DOS) System Master Disk and their apparent lack of documentation. In order to unveil the mystery from these valuable programs, this article will provide a step by step guide to their use.

The DOS System Master disk is distributed free with the purchase of every Apple disk control card. Unlike most diskettes, it does not have a notch cut on it's side, which makes it "write protected." This means that the computer can read or copy files from this disk but cannot erase or modify the files on the disk. (see p. 36-37 of DOS Manual) Although you cannot accidentally erase files on this disk, you should still make a back-up copy of the disk to avoid losing programs due to physical disk damage.

The programs on the System Master Disk not only perform valuable functions, they are provided to serve as programming models. You should not only practice using them, but you should also list them and study how they use DOS commands in print statements as well as Basic programming techniques.

BOOTING THE MASTER DISK

When you first turn on your Apple, its RAM memory is empty. On an Apple II+, a program which is built in (the Autostart ROM) automatically starts reading from the disk in slot 6, drive 1. This process is called "booting" DOS. If the DOS System Master Disk is in this drive, then a special version of the DOS is read into memory, from the disk. This version is special because it does not assume that your Apple has any given amount of RAM memory. The program first checks to see how much RAM is contained in your Apple, and then reads in the DOS program into the uppermost part of RAM memory (to leave you the most room for your Basic programs.) DOS remains there until you turn off your computer or use the "PR#6" command to repeat the boot process.

After DOS is read from the System Master Disk, an Applesoft Basic program named HELLO is read from the System Master Disk. This program prints out the message, "DOS VERSION 3.3 04/15/80 APPLE II STANDARD SYSTEM MASTER" at the top of the screen and then checks to see if your Apple has a 16K memory card installed in slot 0. If you have such a card, then the HELLO program will read the file named INTBASIC from the System Master Disk into the 16K card. This will allow you to use both Applesoft Basic and Integer Basic programs. You will see the message, "(LOADING INTEGER INTO LANGUAGE CARD)." If you do not have a 16K card, no message will appear. (1)

ANIMALS

This game program illustrates the use of random access files and a programming technique called "binary tree searches." (See p. 37 of DOS Manual) In order to use ANIMALS, it must be first copied onto another disk which is NOT write protected.

The object of the game is to pick an animal for the Apple to identify. The program will ask a series of yes/no questions about the animal, and based upon your answers, it will guess which animal you picked. If it guesses wrong, it will ask you to teach it a question which it can use to distinguish the animal it guessed from the one you had in mind. The program grows smarter with each game. Its knowledge is saved between sessions on disk, but you have the option of erasing its acquired knowledge with the NEWFILE command.

Behind the scenes, the ANIMAL program is building a random access disk file named ANIMALSFILE. This file contains the series of questions with information on what to do with a yes or no answer. Based upon a yes or no answer, the file either contains another question or an animal name to guess. Whenever a guess is wrong, the program replaces the animal name with a new question in the file. One branch of the new question then points to the name of the animal which the user had in mind, and the other branch points to the name of the animal which the computer had guessed. The process of adding new questions can be repeated until you run out of space on the disk or until the file is deleted by the user typing in NEWFILE.

APPLE PROMS - data file used by RANDOM APPLESOFT

This program performs the same function as the HELLO program on Apple II computers which have Integer Basic in ROM instead of Applesoft Basic. Although the program is called Applesoft, it is written in Integer Basic and will read Applesoft Basic onto a 16K memory card, if one is present in slot 0. Owners of Apple II's can copy this program onto other disks and specify that it is the first program to be run. This program is not useful on an Apple II+ which has Applesoft Basic in ROM memory.

APPLEVISION

APPLEVISION is a sample Integer Basic program illustrating Hi-res graphics, animation and music.

BIORHYTHM

BIORHYTHM is an Integer Basic program which uses Lo-res color graphics to plot three biorhythm curves for a given month. The program asks your name and birthdate. It repeatedly asks for a projection date and gives the curves for the 30 day period following that date. The plots are hard to distinguish in the absence of a color TV or monitor. The date input routine is a good example of protecting a program by care-

contd.

BASIS 108
THE ALTERNATIVE

Hardware and Software compatible with Apple II'

COMPATIBILITY

The BASIS 108 dual processor microcomputer system has features and capabilities not found in systems many times more costly. But more importantly, the system is compatible with the largest library of microcomputer software and plug-in peripherals available today. It's design and organization permit the BASIS 108 to be compatible with most programs and peripherals originally developed for use with the Apple II TM microcomputer, including CP/M² based applications software packages.

THE KEY . . . . COMPATIBILITY
THE ALTERNATIVE . . . . BASIS 108.

DUAL PROCESSORS

The BASIS 108 is equipped with both 6502 and Z80 microprocessors, and up to 128K or RAM. As the 6502 and Z80 can only address a 64K memory area, the RAM is split into two banks, and bankswitched as required during processing. With dual processors and bank-switched RAM capability, the system offers tremendous utility for computing tasks at all levels. The BASIS 108 will fit your needs; for personal use, professional use, or complex business applications that require more computing power.

MEMORY

Memory capacity should never be a problem with the BASIS 108, with more built-in RAM than any comparible microcomputer system (64K to 128K). The system has a 2K monitor ROM, plus five sockets for additional ROM's or EPROM's of 2K each, for a total of 10K of user ROM space. The central processing unit (CPU) allows internal mounting of two 5¼ inch floppy disk drives for a total storage capacity of more than one-half megabyte.

SYSTEM PACKAGING

The BASIS 108 is a single-board microcomputer system, configured with a detached keyboard and separate CPU. This arrangement makes it easy to configure various work station set-ups.

CPU

The cast-aluminum CPU housing is designed for use in any environment. It's design provides excellent shielding characteristics, eliminating all concerns about radiated interference (RFI, EMI, etc.).

Within this housing are the single-board CPU, six APPLE compatible peripheral slots (2-7), power supply, and space for mounting two floppy disk drives. All system inputs and outputs are located on the unit's rear panel. Included are:

—Power Input
—Two Utility Power Outlets
—A DB-25 Connector for Keyboard I/O

- —Two DB-25 connectors for system I/O
- —Three blank Spaces for DB-25's
- —The Composite Video Connectors
- —A DB-9 Connector for RGB Video Out
- —A DIN Connector for Cassette In/Out
- —A Ribbon Cable Entry Slot

BASIS 108

Characters
- —80 Horiz. X 40 Vert., plus 4 Lines of Text, 80 Characters
- —280 Horiz. X 160 Vert., plus 4 Lines of Text, 40 Characters
- —280 Horiz. X 160 Vert., plus 4 Lines of Text, 80 Characters

KEYBOARD

The keyboard for the BASIS 108 features an exceptionally low profile design that is both attractive and functional. It is just over one inch in height and weighs about five pounds. The keyboard is connected to the CPU with a coiled cable approximately five feet in length, allowing maximum operator freedom and convenience. The keyboard consists of 100 keys that are separated according to function into 4 groups:

- —Standard ANSI Typewriter Keyboard
- —Numeric Keypad
- —Cursor Control Block
- —Programmable Function Keys

The keyboard output is sent to the motherboard where it is decoded by a ROM programmed with mapping and character set information. This arrangement permits tremendous keyboard flexibility. By simply exchanging the standard ROM for one with different programming, you may have any keyboard configuration or set of characters (language) desired. When more sophistication in keyboard handling is desirable, it is possible to select a keyboard interrupt mode of operation.

System packaging for the BASIS 108 is attractive, practical, and complete with built-in capability for system expansion. As your system grows, it will remain a compact, self-contained package instead of the awkward proliferation of boxes, cables, and adaptors that are characteristic of most microcomputer systems.

VIDEO

The BASIS 108 offers a wide range of video display capabilities. The system provides Normal, Inverse, and Blinking displays, two Text modes, and three Color Graphic display modes.

TEXT DISPLAY
- —40 Colums X 24 Lines
- —80 Columns X 24 Lines

GRAPHICS DISPLAY
- —40 Horiz. X 48 Vert.; 15 Colors
- —80 Horiz. X 48 Vert.; 15 Colors
- —280 Horiz. X 192 Vert.; 6 Colors

The BASIS 108 has three separate video outputs to accommodate any display mode. These outputs are:
- —Composite Black and White Video
- —Composite Color Video
- —RGB Video Output

The system also allows soft-switch selection of four character sets, permitting tremendous display flexibility.

The BASIS 108 offers compatibility, and more . . . a lot more. More computing power. More memory. More utility. More of everything you want in a microcomputer system . . . for a lot less.

STANDARD FEATURES

- 6502 and Z80 microprocessors
- 64K RAM Expandable to 128K
- 2K System ROM, plus sockets for 10K additional user ROM
- RGB and Composite (NTSC) Video
- Selectable 80 or 40 Column Text display
- High Resolution Graphics: Six Colors, 280X192 or 280X160 with four lines of text
- Color Graphic Mode: 15 Colors 40X48 or 80X48; 40X40 or 80X40 with four lines of text
- 8 bit parallel output
- RS-232c Serial I/O
- Detached Keyboard: All standard keyboard functions, Upper/Lower case characters, Numeric Keypad, Cursor Control Block, and 15 programmable function keys.
- Built-in mounting for two 5¼ inch floppy disk drives
- Six Apple II compatible slots for plug-in peripherals
- Game paddle I/O
- Built-in loudspeaker
- Two accessory power outlets
- Mounting slots for up to six DB-25 I/O plugs

Hardware and Software compatible with Apple II TM

FUTURE TECHNOLOGY CORPORATION

1735 TWENTIETH STREET, N.W.
WASHINGTON, D.C. 20009 U.S.A.

(202) 785-1599

Call now to schedule your demonstration!

fully testing input ("bomb-proofing").

BOOT13

The Apple can write disks in two different formats. Disk control cards sold after August 1980 usually write to disk with 16 sectors on each track (DOS version 3.3), while older disks wrote only 13 sectors per track (DOS version 3.2). Even though the new DOS 3.3 can write more information on a disk, many older disk (such as WAP library disk volumes 1-40) were recorded with a 13 sector format. Chances are that your disk expects the 16 sector format, and you must take special measures to use a 13 sector disk. The easiest way to handle a 13 sector disk is to copy its contents onto a 16 sector disk using the MUFFIN program described below. Once you convert the disk using MUFFIN, you never have to use the 13 sector disk again. However, certain copy-protected disks cannot be converted using MUFFIN, or perhaps you don't want to bother keeping a 16 sector version of a disk. For these cases, the BOOT13 program allows you to boot a 13 sector disk.

To use this program, put a disk with a copy of BOOT13 in your drive and enter "BRUN BOOT13." You will then be asked which slot to boot from. At this point, put your 13 sector disk in drive 1, and press return for slot 6 or type a different slot number. A new version of DOS will be read from your 13 sector disk, which overwrites the DOS 3.3 which you have in memory. As a result, even if you have two disk drives, the Apple will expect 13 sector disks in both drives. You will not be able to use a 16 sector disk until you use the "PR#6" command to boot a version of DOS 3.3 from a 16 sector disk.

The BASICS disk which came with your disk control card performs the same function as BOOT13. (See DOS Manual at p. 182.)

BRIAN'S THEME

This Applesoft Basic program illustrates the use of the HPLOT command to draw lines on the hi res screen. The program draws a series of lines from a point chosen at random near the center of the screen to evenly spaced points along the screen's edge. By plotting both black and white lines closely together, BRIAN'S THEME creates interesting interference patterns. These patterns appear blue and green on color monitors as a result of the placement of colored phosphors on the TV screen. To end the program, press the escape key.

CHAIN

This machine language program allows an Applesoft Basic program to load a second Applesoft Basic program from disk without erasing the values stored in the first program's variables. This allows you to break up a program which is too large to fit into RAM memory at one time. Detailed instructions on using CHAIN are on p. 106-107 of the DOS Manual.

COLOR DEMO

This Integer Basic program illustrates the 15 colors available through the Apple's low resolution graphics. The program can be used to adjust the color setting on your TV or color monitor, before running more dramatic programs. (Who can tell whether a space invader is supposed to have a green face?) The Applesoft Basic version is called COLOR DEMOSOFT. (See p. 18-19 of Applesoft Tutorial.)

COPY - COPYA

This program is a fast way to make a backup copy of an entire disk. (See DOS Manual p. 38-39.) To use it, type "RUN COPY" if you have Integer Basic, or "RUN COPYA" if you have Applesoft Basic. (Both versions call in a machine language program called COPY.OBJ0 to do the job.) You will be asked which slot and drive to use for the source and destination disk. (If you only have one disk drive, specify slot 6, drive 1 as the answer to both sets of questions and the program will tell you when to insert or change disks.) As a safety precaution, put a write-protect tab on your source disk before using the program. Your destination disk (the one receiving the information) need not be INITialized before using the program. All of the information previously stored on the destination disk is lost.

If you have RUN the program, but want to stop without copying a disk, type Ctrl-C when asked for a slot and drive and then type "POKE 34,0" when you return to Basic.

EXEC DEMO

EXEC files are a valuable tool for storing a series of DOS commands (e.g. LOAD, RUN, etc.) for later automatic execution. EXEC files can also be used to edit Basic programs on a text editor and then reenter them into Basic. The EXEC DEMO builds a text file named DO'ER and then uses it to create, save, list and run an Applesoft Basic program. (See p. 74-75 of DOS Manual.)

FID (FILE DEVELOPER)

This is the most important utility included on the DOS System Master Disk. You may want to have copies of FID on several of your other disks for convenience. FID allows you to copy selected files from one disk to another, find out how much space is left on a disk or delete groups of files.

Detailed instructions on using FID are on p. 183-189 of the DOS Manual. FID has one tricky feature for Apples with more than one disk drive. When you first perform an operation, such as cataloging a disk, FID will ask for a slot and drive so that it will know which disk drive to use. FID will continue to use that drive until you select option 7 RESET SLOT AND DRIVE. After using this option, you will be asked for a new slot and drive when performing your next operation.

LITTLE BRICKOUT

This Applesoft Basic game uses the game control paddle to move a paddle in a classic "Breakout" game. The game can be played on either a black and white or color screen (although the color effects are

contd.

nicer.) The program's speed is due to its extensive use of PEEKs and POKEs, so it is a difficult program for a beginner to decipher. Bonus points (accompanied by interesting sound effects) are awarded for removing all the bricks before using up all of the balls.

MAKE TEXT

This sample Applesoft Basic program illustrates the creation of a sequential text file. Its listing is on p. 64-65 of the DOS Manual. You should put a non-write protected disk in your disk drive before responding to the "WHAT FILE NAME?" prompt.

Pick a file name which has not yet been used on the disk (or the file with that name will be deleted.)

MASTER CREATE

Every disk which you initialize has its own copy of the DOS. When you boot a disk, the copy of DOS is read from the disk into RAM memory. The copy of DOS which is recorded at the time you use an INIT command is called the "slave" version, because it is customized to run on a machine with at least the same amount of RAM. These days, most Apples have at least 48K of RAM, so "slave" DOS disks do not pose a problem. However, if you ever want to give a disk to someone who may not have 48K of RAM, you can use the MASTER CREATE program to put a version of DOS on a disk which will work on smaller RAM Apples. (See p. 44-46 for instructions.) MASTER CREATE can also be used to change the name of the disk's greeting program.

MUFFIN

This machine language program is used to copy files from a 13 sector disk to a 16 sector disk. (See BOOT13 above.) You need to have an initialized 16 sector disk to serve as a destination ready before using MUFFIN. To use it, type "BRUN MUFFIN" and then give the slot and drive of the source and destination disks. You may designate one or more files to be copied. See p. 192-194 of DOS Manual for details.

PHONE LIST

This Applesoft Basic program allows you to store, retrieve and print up to 150 names and phone numbers. The program illustrates a technique of using POKE statements to store data in the program itself. Before using the program, copy it to a non-write protected disk. As you create new entries, they will automatically appear in the DATA statements at the start of the program. You may wish to enter your phone numbers in alphabetical order, as they will be displayed in the order entered. When you select the menu option for exiting the program, a SAVE PHONE LIST command is executed to store your updated list. (The program will also work with a tape cassette by issuing a CSAVE command.)

The program assumes that your printer is in slot 1. To use a different slot, change program line 410.

RANDOM

Perhaps the most powerful data storage method on the Apple is Random Access Disk files. RANDOM, an Applesoft Basic program, uses this method in a sample inventory program. Instructions for using RANDOM are on p. 86-88 of the DOS Manual. RANDOM requires a non-write protected disk to operate, so it gives you the option at the start of the program to insert an initialized disk to which it will copy itself and its data file.

RENUMBER

Applesoft Basic programs use line numbers assigned by the programmer. Usually, programmers will number by 5's or 10's to allow new lines to be inserted in between old ones. If you didn't leave enough line numbers for your inserts, or if you wish to combine two Applesoft Basic programs together into a single program, RENUMBER can help.

RENUMBER will assign new line numbers according to your specifications. It will also read through your program and revise all the line number references found in LIST, GOTO, GOSUB and IF statements. Line numbers in REM statements are not changed.

RENUMBER assumes that you want to renumber the entire program and that the new line numbers should increase by 10's, unless you tell it otherwise. So typing "&" followed by a carriage return will assign new line numbers 10, 20, 30 etc. to your entire program. To change RENUMBER's actions, you can set a number of parameters:

F - first new line number to be assigned.
I - increment between new line numbers.
S - the first line of the range of lines to be renumbered.
E - the last line of the range to be renumbered.

In addition, RENUMBER has commands to combine two Basic programs (Hold and Merge) and to convert ROM Applesoft programs to RAM and visa versa (Convert.)

To use RENUMBER, type "RUN RENUMBER". This will load a special machine language program into memory, but will also set Basic pointers so that typing or loading Basic programs will not overwrite RENUMBER. RENUMBER will remain in memory until you boot a new disk or perform some other special operation. RENUMBER commands can then be executed while editing Basic programs by typing a & followed by the RENUMBER command in response to a] Applesoft Basic prompt.

To test whether RENUMBER is still in memory, type "& I 0" from Basic. If you get the message "INCREMENT = 0", then RENUMBER is probably still in memory. If not, then RUN RENUMBER again.

Specific instructions on using RENUMBER can be reviewed by RUNning RENUMBER INSTRUCTIONS. (2)

Two cautions about using RENUMBER. Do not press Reset during a renumbering operation, or your Basic program may be left half done and unusable. Also, RENUMBER has its

REVIEW OF THE PASCAL INTEREST GROUP LIBRARY
by James S. Harvison

The PIG library was last reviewed by Paul Sand in the May 1981 issue of WAP. At that time we had three disks. The library has now grown to nine disks with almost enough contributions to make another. I encourage all Pascalers to share your work with us. Contributors are entitled to a disk of their choice. Please send them to James S. Harvison (PIG Librarian), 307 Lexington Drive, Silver Spring, MD 20901, or you can contact me via the club BBS (WAP455), the Source (TCD563), Micronet (70320,165) or 301-593-2993.

Following are the reviews of all nine PIG disks. The first three are repeated from the May 1981 review. PIG: 1-4 was compiled by Paul Sand; 6 and 7 by Howard Lefkowitz; 5, 8 and 9 by James Harvison.

===

LIBRARY DISK PIG1:

The following is a description of the programs available on this disk. Each program is provided in source code form. The purpose of this minimal documentation is to explain the use of those programs that have no internal directions. I hope you enjoy the use of these routines and that everybody comes up with contributions to next month's disk. Grateful thanks to this month's contributors.

DSPCHARSET - (Bill Wurzel) This program reads the data in SYSTEM.CHARSET, which holds the graphics character set used by the Pascal turtlegraphics routines. It displays any of those characters on the text screen. When started, the program displays character '0' on the screen. To see another character, type '#', <cr>, <the character number>, <cr>. When done, simply type '%', <cr> and the program ends. Note that Bill's program keeps only one block of the two in SYSTEM.CHARSET in memory at once, so when you type in a number greater than 64, the second block is read in. There is no provision to go back to the first block.

DEFCHARS - (Bill Wurzel) A logical sequel to the preceding program, DEFCHARS allows you to generate new graphics character sets. When the program is started, SYSTEM.CHARSET is read from drive #4: and copied to a file NEW.CHARSET on drive #5:. (Obviously, those with one drive will have to tinker with this code before using it.) All editing of the character set is done on a 7 x 8 dot matrix. Periods signify off bits, plus signs designate on bits. One creates a new character by moving the cursor through the dot matrix, turning the individual bits on or off as necessary. Single key commands are easy to master: The keys W, E, R, S, F, X, C, V form a cursor pad, moving the cursor in the direction corresponding to the location of the key in the pad. For expample, E moves the cursor up, V right and down. The D, B, and M keys

are mode-changing keys. D causes bits landed on by the cursor to be turned on, B causes them to be turned off, and the M key causes no change.

Finally, when the character is finished, type '#' (if you want to edit more characters) or '%' (if you want to exit the program). The program will ask for the number of the character you just defined, and save the new definition in NEW.CHARSET. Then, depending on whether you typed '#' or '%' the program will allow you to enter another character or exit.

CHECKBOOK - (Tom Woteki) Tom's checkbook program was presented at the November meeting, and here it is in machine-readable form. It is a program to analyze your checking account cashflow on a monthly basis. The program is (more or less) self explanatory; running it should pose no problem for you smart guys.

MINIFILER - (Tom Woteki) This program duplicates three functions of the UCSD Filer: L)ist Directory, K)runch, and Z)ero Directory. Unlike the Filer, however, you have the source code of this program. From perusing Tom's program, you can learn all sorts of interesting stuff about how directories and files are maintained on your disks. I expect a lot of useful utility programs next month based on the ideas here.

PEEKPOKE - (Michael Hartman) Here are two familiar routines from good old Basic: peek(addr: integer) returns an integer with the contents of memory location addr. Poke(addr, value: integer) puts value in the memory location specified by addr.

Mike has written these routines in a unit so they can be incorporated into your library.

FILEDUMP - (Paul Sand) This is a simple program that allows you to see what is really in those files on your disks. It dumps any file in hex and ASCII, block by block. When started, it asks for a file name. If the file is found, it will display the first half of the first block. Use the arrow keys to display other blocks: back-arrow displays the previous block, forward arrow displays the next block. Control - C gets you out of the program.

TIMERSTUFF - (Paul Sand) These routines are offered to anyone with a Mountain Hardware Apple Clock. They are meant to be used in the program development process to find out where your big, complex program is spending all its time. You can specify timing of up to twenty different portions of code and report at any point the total time recorded by each timer and the number of times each timer was turned on. ('You mean this procedure is called TWENTY THOUSAND TIMES?!'). Procedures included in this unit are:
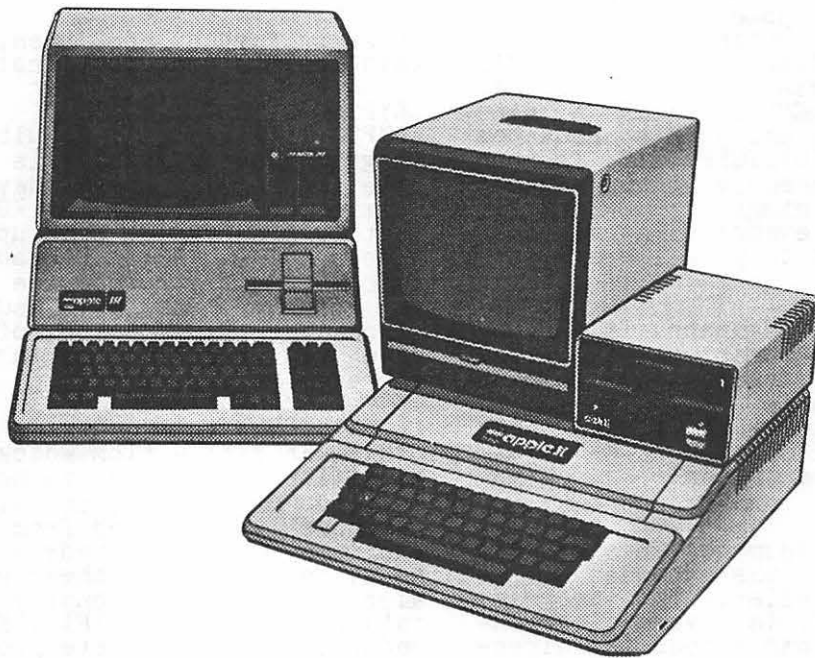
contd.

inittimers - sets everything up, zeroes timers.

starttimer(i) - starts timer #i

stoptimer(i) - stops timer #i

reporttimers - outputs a table showing the cumulative times, etc.

PRINT - (Paul Sand) Another simple program, but I use it more than any other. All it does is print out a series of text files on the printer, skipping over page breaks and numbering each page. When started the program will ask for a file to print. When done, it will ask for another. When done, just reply with a <cr> to this request.

BIOSUNIT.TEXT
BIOSDEMO.TEXT
BIOSDOC.TEXT
BIOSSTUFF.TEXT - (David Neumann) These files contain the results of David's attempts to figure out how to make Pascal programs do the neat things that are not obvious; inverse characters, horizontal scrolling, etc. Note that the version on this disk works for Apple Pascal version 1.0 only. The same routines for version 1.1 are on library disk PIG3:. David's documentation for thse routines is in BIOSDOC.TEXT. BIOSDEMO.TEXT is a nice demonstration of the things his routines allow you to do. BIOSSTUFF.TEXT is the real stuff - 6502 assembly language that does all those things. Finally, BIOSUNIT.TEXT is an example Pascal host unit that could be put in your library.

MENU.TEXT
MENUDOC.TEXT - (David Neumann) This program will detect the "runable" code files on your disks, display them, and allow you to choose which one to run by typing a single letter command. Very nice. Uses routines in BIOSTUFF above. MENU.TEXT contains David's documentation.

DIR.TEXT
DIRDOC.TEXT - (David Neumann) This program essentially duplicates the extended directory command in the Filer. The program, contained in DIR.TEXT, is a very clear explanation of how one can access the directory from within a program. DIRDOC.TEXT is David's documentation.

CRECHARSET.TEXT
CRECHARDOC.TEXT - (David Neumann) This program, like DEFCHARS above, allows one to alter the graphics character set. See which one you like better, sports fans. CRECHARSET.TEXT is the program, CRECHARDOC. TEXT is the documentation.

DOODLER.TEXT
SERENDIP.TEXT
CUBE.TEXT - (Bill Schultheis) These are three excellent graphics programs. DOODLER is a fast and pretty random drawing routine. SERENDIP is a dot-pattern generator, and CUBE displays a semi-animated cube undergoing rotations, with hidden lines.

(Librarian's Note: SERENDIP is based on an algorithm given in the August 1977 BYTE, in which the first advertisements for a com-

puter called "APPLE II" appeared. A 48K machine "using the new 16K RAM chips" went for the tidy sum of $2638. Ah, the good old days.)

INTDAT.TEXT - (   ) This file is a collection of procedures to convert strings to integers, data strings to integers, and integers back to data strings. This allows one to input data as a string, using inter-line editing, and then convert to numbers. Good idea!

==

LIBRARY DISK PIG2:

The following is a brief description of each of the files found on this disk. The contributions from the members have been excellent, and I'm sure the users will find a lot of interesting stuff on this disk. I was unable to identify the contributors of some of the programs on this disk. To these anonymoose: Please let me know who you are, so I can give you proper credit in future editions of this disk. I apologize for my sloppy recordkeeping and failing memory, and hope you guys don't have easily offended egos.

Without further ado, then, let us examine this month's contributions:

CRYPTODOC.TEXT
CRYPTO.TEXT - (Bill Schulteis) The CRYPTO program does cryptanalysis on a level suitable for solving the Sunday Post's magazine puzzle or decoding any KGB transmissions that you happen to pick up on your short-wave. Just kidding. Read CRYPTODOC.TEXT before you try to compile or run CRYPTO - it requires that David Neumann's BIOSSTUFF (from the PIG1: disk, of course. Where have you been?) be installed in your library.

FILER.LIB
MINIFILER.TEXT - (Tom Woteki) This is Tom's Minifiler described to us at the January meeting. Briefly, it allows filer-like commands to be used from inside your own programs. Tom has made a real achievement here, not only in sheer programming, but also in his exceptionally clean implementation. Note that MINIFILER.TEXT is a procedure, not a complete program. FILER.LIB is a library file that can be used from another program.

PAGEDUMPER.CODE - (Tom Woteki) Dumps out sucessive pages of your memory in ASCII. You can move both forward and backward in memory, discovering who knows what.

IOUNITA.TEXT
IOUNITB.TEXT - (Anonymous) These two files contain routines written as intrinsic units that allow your program to do terminal-independent screen I/O by reading terminal data from SYSTEM.MISCINFO. Most, if not all routines are from APPLE3:DISKIO.TEXT, but this is the right idea - terminal dependencies should not be written into your own programs, but put into the library.

LIFE.TURTL.TEXT
LORES.USER.TEXT
LORES.UNIT.TEXT

contd.

LIFE.LORES.TEXT
LIFE.INFO.TEXT
LORES.INFO.TEXT - (Michael Hartman) A series of programs and units that implement Conway's Life, both in turtlegraphics (LIFE.TURTL.TEXT) and lo-res graphics (the others). LIFE.TURTL.TEXT needs to have PEEKPOKE, given in PIG1:, inserted in your system library. LORES.UNIT.TEXT is the Pascal host for the 6502 Assembler code in LORES.USER.TEXT, which is documented in LORES.INFO.TEXT. LIFE.LORES.TEXT is Conway's Life implemented in lo-res, which uses Mike's lo-res unit. It is documented in LIFE.INFO.TEXT. I sure hope I got all that straight, Mike.

DIS1.TEXT - (William Wurzel) This is Bill's P-Code disassembler, which he described at the December 1980 meeting. Obviously, this can be of great use in studies of how the compiler works and how you can write code that takes less space and runs faster. Use is straightforward - good luck!

PLOTPOURRI.TEXT - (Anonymous) A nifty program to do 3-D contour plotting, you might compare it to the Basic version available from the club library. It has a provision to label the plot with the equation of the function plotted, complete with super- and subscripts! As an added attraction, it also has a routine to print the whole thing on a Paper Tiger (440 or 445). Fantastic!

RNDSPIRO.TEXT - (Anonymous) A graphics program that draws spirograph-type displays. Very pretty! (Use a color TV.)

LORES.TEXT
PLOT.TEXT
ANDROMEDA.TEXT
DANMAC.TEXT - (David Neumann) What do you know? Another lo-res plotting package! LORES.TEXT is the Pascal host for the 6502 code found in PLOT.TEXT and DANMAC.TEXT. Once you have that together, ANDROMEDA.TEXT is a program that uses the unit.

==

LIBRARY DISK PIG3:

We again have a disk of useful programs, submitted by our talented membership or filched from other folk.

BIOSDOC.TEXT
BIOSDEMO.TEXT
BIOSSTUFF.TEXT - (David Neumann, Bill Shultheis) These programs also appear on PIG1:, as contributed by David Neumann. Bill Schultheis has modified the assembly language in BIOSSTUFF.TEXT to reflect the changes made by Apple in Pascal version 1.1. I've duplicated the unchanged files here as well. BIOSDOC.TEXT is David's original documentation. BIOSDEMO.TEXT is his demonstration of the various neat things his routines allow you to do: horizontal scrolling of the text page, inverse video characters, text window definition, and other wonders you thought you could never do in Pascal. BIOSUNIT.TEXT is the Pascal unit suitable for inclusion in the system library.

UNASM.INFO.TEXT
XUNASM.CODE - (Bill Schultheis) This is a 6502 disassembler program that will dis-

play contents of memory as 6502 instructions, if possible. Supplied in code form only - check it out. UNASM.INFO.TEXT is the file containing Bill's documentation.

X.TRACE.TEXT
X.SPY.TEXT - (Bill Schultheis) These files will show you how to figure out just where your program is taking you in memory. X.SPY.TEXT is the Pascal demo program that turns on tracing for a single Pascal writeln statement. X.TRACE.TEXT is 6502 Assembler code that contains routines needed by SPY. Bill almost makes me wish I knew Assembly language. Almost.

SETUPMX80.TEXT - (Burt Chambers) As one of the proud owners of an Epson MX-80 printer, I can guarantee a warm spot in my heart for anyone who writes software for it. This program is essentially a routine to output those special characters to the MX-80 to allow it to do compressed print and/or overstriking (for darker print). Recommended to all you MX-80 owners. It also poses a challenge to those of you who have other printers: can you provide us with equivalent programs for your own machine, or (better yet) can you write a more general program that would allow a user to define his own printer commands?

BURTS-DUMP.TEXT - (Burt Chambers) An improved version of Tom Woteki's memory dump program (provided on PIG2:) Check it out - it may show you where your variables have vanished to.

EIGHTQ.TEXT - (Paul Sand and Somebody Else) This program solves the famous Eight Queens Problem - How can you place eight queens on a chessboard so that no two queens attack each other? I found this program on a San Francisco Apple Core Disk; the documentation stated it came from Washington Apple Pi! It wouldn't quite run succesfully as furnished, so I made some pretty extensive changes to it. The pretty recursive algorithm used is taken from Algorithms + Data Structures = Programs by Niklaus Wirth. The exceptionally nice graphics and sound generation were in the original version, however, and I hope whoever it is that wrote it will identify himself (or herself).

DATE.TEXT - (Jeffery Sue, SF Apple Core) A nice routine to extract the "Last Booted" date from a disk.

MASTERCA.TEXT - (Steve Lloyd, SF Apple Core) - This program creates a "master catalog" file from many disk directories. It could use some fixing up, guys. Although our brother from California, has the right idea, we can do better. Next month?

TIGER.UNIT.CODE
TIGER.TEXT
SCRNBYT.TEXT - (Sue, Sue, and Gustafson, SF Apple Core) These routines allow a Hires screen dump to the venerable Paper Tiger 440/445 printer. TIGER.UNIT.CODE is a unit suitable for inclusion in the system library. TIGER.TEXT is the Pascal source of the unit. SCRNBYT.TEXT is an Assembly language interface to read the hires screen.

PRINTSET.TEXT - (Paul Norris, SF Apple Core) A program to throw the software switches on the Centronics 737 printer. See comments above.

MASTERMI.TEXT - (Paul Norris, SF Apple Core) This program plays Mastermind. I think it cheats.

PILOT.TEXT - (George Golden, SF Apple Core) This program is from the July 1980 Byte. It will take a Pilot program and translate it into a Pascal program.

TWOD.TEXT - (David Cheng, SF Apple Core) This is, despite the name, a three-d plotting program. Not quite as neat as David Neumann's on PIG2:, but still worth a look if you're interested.

PRETTYPAS.TEXT - (David Cheng, SF Apple Core) A Pascal program formatter. It converts Pascal reserved words and single letter varaibles into upper case, and some other stuff. See the program for more information.

BLIZZARD.CODE - (Bob Doran, SF Apple Core) No source file, but one of the prettiest graphics programs I've seen. Will be pleasant to look at next July.

GROCERY.TEXT
MASTER.TEXT - (Jeff Sue, SF Apple Core) Have you ever wanted to use your computer in your everyday life? If so, why? Never mind, you could do worse than to look at this program. Impress the checkout girl at the Safeway with your computer-generated grocery list! A gem of a program as well - techniques used here are very nice.

==

LIBRARY DISK PIG4:

The following is a description of the programs found on this disk. As usual, we have a random assortment of utilities, applications, games, graphics, hardware interfaces and a small kitchen sink on track 23, sector 4. (Thought we didn't have room for it, but we squeezed it in.) Due to the sink's presence, I don't have space on this disk for my usual rambling comments. (Please stop cheering, guys.) So I'll keep my editorializing to a minimum.

CASSETTE.TEXT
TAPERTN.TEXT - (Brian Pierce) CASSETTE.TEXT is the Pascal program, which uses the assembly language contained in TAPERTN.TEXT. The purpose of this program is to enable transfer of data and programs on cassette tape between two computers that may not be able to communicate in any other manner. Brian's explanation of the program is contained in XMITTAL.TEXT.

XREF.TEXT - (Brian Pierce) This program will generate a cross-reference listing of your Pascal source file. It will handle "included" files correctly. If the amount of input exceeds the memory available it will use a workfile on disk. It will not cross-reference Pascal keywords (but could, if you changed it.) Brian's documentation for XREF is also in XMITTAL.TEXT.

PASCALZAP.TEXT - (?) This program, typed in from the Jan. '81 Call-Apple, (and modified from there), allows one to read, write, and modify individual blocks on the disk in the second drive. This has to be the ultimate disk patcher - but does anyone actually use a disk patcher?

TFORMAT.TEXT
STRINSTUFF.TEXT
FORMAT.DOC.TEXT - (Tom Woteki) International fame has not gone to Tom's head, folks. Here is his latest project, a text formatter. TFORMAT.TEXT is the program itself, STRINSTUFF.TEXT is a unit containing routines used by the formatter, and his documentation is in FORMAT.DOC.TEXT. Enough said - you would do well to check this out.

FILESYS.TEXT - (Larry Johnson) This is a program from the new book "Pascal Programming for the Apple" by T. G. Lewis. It is an example of a database implemented by using "B-Trees" and an index file. It isn't really suitable for any serious use, but could offer helpful hints for anyone out there considering storing large amounts of data. For documentation, I suggest you buy the book, which isn't all that bad considering that the author doesn't understand the difference between TEXT and INTERACTIVE files.

OTHELLO.TEXT - (Paul Sand) My pride and joy - my first large Pascal program. You play the game of Othello against the computer. Since the program itself has no directions, here are a few tips: When the computer asks for "lookahead", I wouldn't recommend anything higher than two. (It is slow!) When picking your move, the forward and back arrows move the star to the locations of your possible moves. When it lands on a square you like, press the "P" key to move there. Be warned - this program will probably only beat you if you don't know the game very well. If you don't know how to play Othello at all, you should probably learn before you try this program.

PRETTY.TEXT - (Paul Sand) I typed this program in from the book "Pascal With Style" by Ledgard. It will turn a poorly-formatted Pascal source file into a nicely-formatted one (by the author's standards.) Check out the book for any documentation you might need on this.

SIERPINSKI.TEXT - (Paul Sand) This little graphics ditty is a space-filling curve, like the HILBERT program on APPLE3:. The algorithm is from Wirth's "Algorithms + Data Structures = Programs" and all that had to be added was the graphics for the Apple. Very pretty, I think.

COMPARE.TEXT - (USUS) This program is available on one of the disks from USUS, but since its original source is the "Pascal News", and the program's author (according to the comments in the program) does not restrict non-profit distribution, I think it's fair to distribute it outside USUS. This is a program that will find differences in two text files; I've found a program like this to be useful when I've got more than one version of a program and can't remember how they differ. And the program itself is very instructive.
                                              contd.

```
==========================================
```

LIBRARY DISK PIG5:

This fifth disk in the Pascal library is composed of 17 files from two submitters. The first 13 are by David Neumann where he provides several units to allow reading and writing to DOS formatted disks. The last four files are (to my knowledge) submitted anonymously and provide for an excellent program to make use of the Epson printers enhancement capabilities.

Librarian's Note: The library copy of the file "PRINT_FILE.TEXT" has a bad spot in the middle. If the submitter of these programs sees this, please contact the PIG librarian to correct this problem. Your anonymity will be maintained or corrected as you wish.

DOSIOGLO.TEXT
DOSIOGLO.CODE
DOSIOGEN.TEXT
DOSIOGEN.CODE
DOSIOIN.TEXT
DOSIOIN.CODE
DOSIOOUT.TEXT
DOSIOOUT.CODE
DOSXFER.TEXT
CAT.TEXT
CAT.CODE
DOS.DOC.TEXT - (David Neuman) These files implement file transfer to and from DOS formatted disks. DOSXFER.TEXT and CAT.TEXT are demonstration programs to show how to use the procedures. DOS.DOC.TEXT is a brief documentation file about the procedures implemented.

MX80SAMPLE.TEXT
PRINTINFO.TEXT
PRINT.TEXT
PRINT_FILE.TEXT - (   )
PRINT_FILE.TEXT and PRINT.TEXT are the source files for a program that makes full use of the print enhancement capabilities of the MX80 (or MX100). By including commands within your text file (such as {E} which turns on double printing) you can selectively turn on or off features of the printer. PRINTINFO.TEXT is the documentation file for the program and MX80SAMPLE.TEXT is a sample text file using the features of the program.

```
==========================================
```

LIBRARY DISK PIG6:

The first "GEM" on this disk is a Master File Catalog Program provided by John F. Long. It is based on the Ed. Heyman article in Byte, May 1981. John has added options for printer output and page by page screen output as well as some other improvements. This program includes "FIRSTPART.TEXT" and "MAINCAT.TEXT".

Next we have different version of the same Byte article by George Hart via Jim Harvison. This has many changes and is not compatible with the original article. It is much faster and perhaps someone will take the best of both programs and "CREATE" the SUPER master catalog. The files for this program include "CAT1, CAT2, CAT3 AND JUST PLAIN OLD CAT".

ADDRESS.TEXT is an address calculator submitted by Roger Kaufman.

Michael Hartman submitted the SPIROGRAPH program. Very relaxing and not too bad for a demonstration.

Next we have a set of HIRES units for the Paper Tiger sent in by Ed Knepley. These are based upon earlier WAP newsletter articles by the infamous Dr. Wo. The procedures (1) Dump the screen to the Paper Tiger, (2) Save the screen to disk and (3) Load a screen from disk are provided. We also have the ability to use page 2 with these units.

Finally a Pascal Memory Utility submitted by Bill Wurzel. This routine allows you to examine, disassemble and change memory. It is based upon an article by Ron DeGroat in the August issue of CALL A.P.P.L.E. Bill has made modifications to accommodate the Smarterm board.

```
==========================================
```

LIBRARY DISK PIG7:

Thanks to David Newman for the LISP and Clock routines. Dave has provided the "essence of a LISP interpreter". As he points out, it needs to be extended. So get busy out there. This means YOU!

The clock routines are for the CCS 7224 clock/calendar card. They allow the CCS card to function essentially the same as a Mountain Hardware card. Two extra routines are provided for setting the clock and calendar and to allow automatic setting of the date when booting.

The balance of the files are from the Dallas Apple Corps - Fort Worth Apple User Group, and are described in README.TEXT.

```
==========================================
```

LIBRARY DISK PIG8:

Contributions on this disk are from a potpourri of sources. Tom Woteki, the USUS library with modifications by Brian Pierce and Howard Lefkowitz, and via an exchange program, the Dallas Apple Corps and Fort Worth Apple Users Group.

PUFFIN.CODE
PUFFIN.TEXT
TRANSFER.TEXT - (Tom Woteki (Dr. Wo)) This program, featured and well documented in the September 1981 Washington Apple Pi, is an excellent program to move DOS files, text or binary to the Pascal system. I use it myself for moving HIRES files created under DOS to Pascal. I can then put the "pictures" together for a fast moving slide show. TRANSFER.TEXT is a part of the source file which is (*I included) in the PUFFIN.TEXT source.

PROSE.CODE
PROSE.DOC1.TEXT
PROSE.DOC2.TEXT - (USUS Library modified by Brian Pierce) an exceptional text formatter altho somewhat slow. It is provided here in CODE form only due to some concern about whether or not the source is in the public domain. The DOC files fully explain its

use. The code version has been fixed and enhanced by Brian Pierce.

HOLIDAY.TEXT
TWINKLE.TEXT
MUSIC.BOX
AULDLANG
RAUCOUS
JINGLE.BLS - (Jim Harman Fort Worth Apple User Group) Makes your Apple sing and hum (Well if you listen closely it already hums). Compile these and give them a try. Excellent demo's and use of the NOTE procedure in the APPLESTUFF UNIT of the system library.

===

LIBRARY DISK PIG9:

This month's contributions are from just two people, yet we almost fill the whole disk. In particular, we are most grateful to John C. Stephenson for the contribution of his CRT unit - a major effort. We hope these contributions will motivate the rest of you to dig through your own programs and share them with us. John is a Washington Apple Pi member (WAP591) even though he lives in New Jersey. I hope that other distant members of WAP will follow his example and take the time to contribute to us as well.

CRT - (John Stephenson) This unit supports extensive filtering of keyboard input, screen template use and all in all should make your program "crash" proof (at least so far as input goes). It is comparable to what is provided in commercial versions costing from $75 on up. This is one you should definitely have a use for. Documentation is contained in READTHIS.TEXT. All files except the last two on the disc have something to do with this UNIT or a demo of it.

CPMTOPAS - (Dave Neumann) Dave has provided us owners of a Z80 card and CP/M (running UCSD on it of course) to transfer files from the CP/M system to Apple Pascal format. CPMTOPAS.TEXT is the source. Dave also has the source for a program to move files from Apple Pascal to CP/M format. This program is available thru the WAP CP/M users group. ❦

---

contd. from pg 29

machine language subroutine hidden on the last line of its program, so don't edit the last line in Applesoft.

RETRIEVE TEXT

This Applesoft Basic program illustrates reading sequential text files from disk. The program reads the file created by MAKE TEXT. See p. 65 of DOS Manual for a listing.

REFERENCES

(1) Creative Computing, Vol 7, No. 5, p. 200 (May 1981)

(2) Softalk, Vol. 2, No. 11, p. 152-56 (July 1982)

APPLESEEDS CORNER

APPLESEEDS OFFICERS

President   - David Stern
Vice-President - Brian Van Flandern
Editor   - David Stern (Peter Thomas
                 will soon be starting)
Librarian   - Mike Forman

APPLESEEDS

Appleseeds is a group of kids who meet during the monthly WAP meeting. Our meetings consist of a speaker (first 30 min.), presentations, and discussions of general points of interest. If you have special knowledge in a topic relating to the APPLE and would like to give a presentation for Appleseeds, please contact David Stern at 881-2543.

ELECTIONS

Elections were held last month, and an editor was elected. The editor, Peter Thomas (not yet working on the newsletter) will soon be working with a co-editor (not yet elected). The offices of president, vice president, and librarian remain the same.

The job of the editor is as follows:

1) Collect articles from Appleseeds members.

2) Compile the articles into a newsletter (using a word processor).

3) Get the newsletter to the president two (2) weeks before the Appleseed meeting.

Also, it has been decided by our group that it will be necessary to start the collection of dues. Starting with the October meeting (at the end of September), each Appleseed member will be required to pay fifty cents per year membership fees. The money accumulated for Appleseeds will go toward the purchase of Demo Disk, a disk full of demonstrations of the new Apple games. Because Demo Disk is public domain software, each Appleseed member will receive a copy of the disk. In addition to Demo Disk, Softdisk magazette will also be ordered for Appleseeds. Softdisk is a double sided disk containing useful (and sometimes fun) programs. Softdisk is only $10 for the first issue and $5 for each subsequent issue when the previous disk is returned. More on this topic later, or call David Stern.

HIGH SCORE LISTING

The high score list in this newsletter started just two months ago, and already it has tripled in size and variety. The response which I received concerning the list was incredible! I was receiving about five phone calls and two letters in the mail per day with new scores! But there are still a lot of games without listings,

so if you have a game which is not listed or you have bettered one of these scores, call David Stern.

HIGH SCORE LIST

GAME	SCORE	APPLESEED
Alien Rain	592,080	Steve Green
ICBM Strike	107,352	S.C.D.
Bug Attack	57,530	David Hecker
Bandits	33,135	David Stern
Cannonball Blitz	136,210	Mike Forman
Ribbit	14,170	David Stern
ABM	26,000	S.C.D.
Wormwall	3,573	David Stern
Sneakers	47,810	Andrew Kaplan
Laf Pak:		
Creepy Corridors	3,530	Andrew Kaplan
Apple Zap	31,580	David Stern
Space Race	9,900	David Stern
Mine Field	4,872	Andrew Kaplan
Zero Grav Pinball	23,500	David Stern
Mouskattack	11,080	Danny Roth
Juggler	62,530	Jacqui Stern
Snoggle	22,640	Robert Riddell
Marauder	153,100	David Stern
Taxman	93,320	Mike Forman

GAMING TIPS

These tips are compiled from suggestions of many Appleseed members, so no names are given. If you have been sending in tips, keep up the good work. If you would like to send in tips, until a co-editor has been elected, call David Stern.
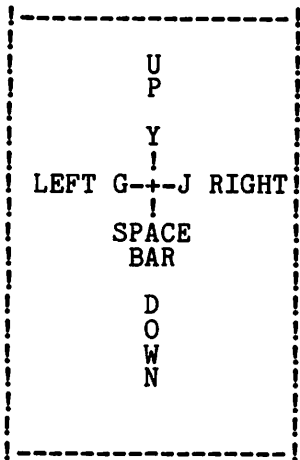
LAFPAK

CREEPY CORRIDORS:

The main tactic in this game is to pick up all the diamonds first, then find a corridor which is pretty straight, get at one end and shoot all the monsters (early levels ONLY). Then leave the room. Make sure NOT TO SHOOT ALL THE MONSTERS BEFORE GRABBING ALL DIAMONDS! THIS WILL CAUSE THE REMAINING DIAMONDS TO DISAPPEAR AND THE DOORWAY TO APPEAR. Diamonds are 100 pts. each, and those points can be very valuable. In the later levels (those levels with the monster who looks like a blimp), your best strategy is to grab all the diamonds and leave as QUICKLY AS POSSIBLE. The blimp guys CANNOT BE KILLED, so don't try!

APPLE ZAP:

The keys which you select to control your turret are very important. The A, Z, -> and <- keys are okay for short games, but have been found to be too clumsy in a long game. We suggest the following configuration:

contd.

```
!------------------!
!                  !
!        U         !
!        P         !
!                  !
!        Y         !
!        !         !
! LEFT G-+-J RIGHT !
!        !         !
!      SPACE       !
!       BAR        !
!                  !
!        D         !
!        O         !
!        W         !
!        N         !
!                  !
!                  !
!------------------!
```
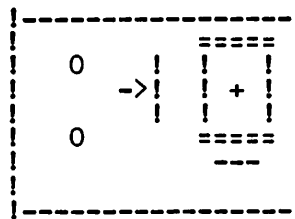
This setup allows your fingers to relax and also showed the best response time. The fire key will be used alot, so it should be a key near the repeat key. ('P' has shown to be the best).

Once the game starts, and for the first 10 or 15 thousand points, hold the fire and the repeat down together. Let go only when changing directions. This way if a ship has fired a laser at you, both the laser AND the ship will be de-destroyed.

SPACE RACE:

There are not very many tips for this one. We have found that if you are using a TG

Products joystick, the following trick works very well.

```
!------------------!
!              =====!
!   O    ->! ! ! !  !
!         ! ! + ! !  !
!   O     ! ! ! !  !
!              =====!
!               ---  !
!                  !
!------------------!
```

Adjust the marked control so that when you let go of the ship, it flies to the center of the screen. This will come in handy when you are near the top of the screen. When you are up there, let go of the joystick and let it guide itself through the opening.

MINE FIELD:

The screen setup is not random. On one screen, six shots will clear the whole screen. It is not advisable to stop next to a large bomb, even if you don't plan on shooting it. The chain reaction caused by your shot could come back to your area. If you can, shooting the guy who runs across will give you a lot of points. &

ATTENDING YOUR FIRST PI MEETING

by James T. Demay Jr.

Two of the major benefits of belonging to Washington Apple Pi, in my opinion, are the exchange of ideas and the sharing of experiences. The ABBS (Apple Bulletin Board System run by WAP) offers an acceptable forum for short messages, but the monthly club meetings are an experience in themselves. If you have not yet attended, I would like to give you an idea of what goes on and how to get the most from your Saturday morning.

I always try to arrive at USUHS by 8 o'clock. The main meeting doesn't start till 10, but if you don't get there early, you will be depriving yourself of the best part of the meeting. The two hours between 8 and 10 are used to full advantage. It is at this time the WAP newsletter is distributed, club library disks are picked up, the club store sells software and hardware items, and new members sign up. This all happens in the lobby while Bruce Field, Tom Warrick and a few other knowledgable Apple owners conduct an informal question and answer forum in the main meeting room, actually a lecture hall for the USUHS students.

If you arrive any time after 8:30, you will be met by what appears to be the serenity of a prospering beehive. Look closely toward the center, and you will see Paula and/or Bernie Benson answering some new member's questions under the information sign. To your right, there is a wall of display cases to which the club library catalog has been fastened for your perusal. Look closely at this for you will find many exciting and useful programs almost free for the asking. I say almost free because the club charges only $5.00 per disk to club members. Along the opposite wall there are usually three tables set up, each manned by several volunteers. The closest table to the elevators is manned by Gordon Stubbs and his staff of volunteers. This is the place to order and pay for library disks, pick up back issues of the newsletter, and renew your membership. Your name and WAP membership number will be required by the receipt writer to conclude these transactions. If possible, please pay by check. Having the check made out in advance to Washington Apple Pi, including your membership number, will shorten your time in the always long lines.

The receipt writer will take your check, or cash, and give you two copies of the receipt. If you paid for library disks, you will want to move to the table to the right. Here you give one copy of the receipt to the volunteer behind the table full of disks. S/he will select the disks indicated by the receipt and give them to you.

The third table is the place to pick up your latest copy of the WAP newsletter. Here your zip code and name are required by Gena and Bernie Urban and the other volunteers behind the piles of newsletters. There are three lines divided by zip codes. You can determine which line to enter by the signs above each volunteer. When you get to the front of the line, tell the volunteer your zip code, and then your name. S/he will find your mailing label, and affix it to a copy of the newsletter. If you have not done so, examine your mailing label. It consists of four lines. The first is your name. The second line is important since it has your WAP membership number followed by a slash (/) and the expiration date of your membership. Sending a check to cover your renewal a month before this date will insure that you do not miss an issue of the newsletter and that you will receive the full benefits of the club. The last two lines of the label are your home address. The magazines which are not claimed at the meetings will be mailed to members at club expense.

If you have been reading carefully, you will have noticed the recurrence of one word in particular: VOLUNTEER. This was no accident. All of the people you see here (and many you don't see) are working for you for free. There is no paid staff (yet). The joy of sharing in and being a part of this evergrowing group of dedicated individuals is all the reward they receive. I like to think that by just being near so many talented and intelligent individuals, some of those desirable qualities will "rub off" on me. If you have the time and feel the urge to help, ask anyone behind a table to direct you to David Morganstein, the club President and general overseer for these Saturday morning get-togethers. He is always looking for volunteers to help with something.
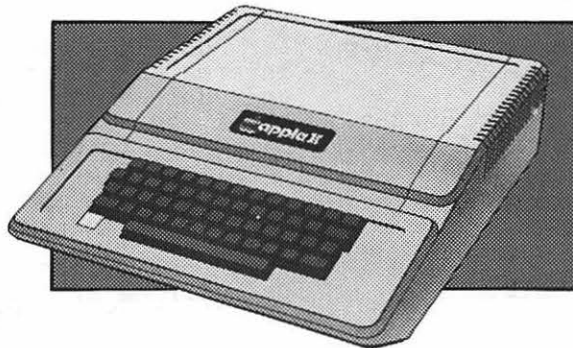
Moving on to the next table, you will see any number of software and hardware items for sale. This is the club store. Rich Wasserstrom, head of group purchases, may be able to save you some money on your next Apple add-on. Behind this table too are volunteers. If you have ordered something from the club store, they will be happy to take your money or your order. Note that the items purchased from the club store must be paid for at this table, or by mail at the club post office box. The receipt writers at the first table will not be able to accept your payment for these items. So don't get in the wrong line by mistake.

The final table is manned by Dana Schwartz, the Vice-President. He is accepting membership applications and checks from new members. This is where you will find out just how many members there are. As of the August meeting there were over 2000.

The activity in the lobby slows as the 10 o'clock meeting time nears. This is because the more experienced meeting goers know that all of the seats are taken by 9:45 or so. Unless you like standing in the aisles, I would advise an early entry

WAP INCORPORATES AND ADOPTS NEW BYLAWS

by Robert C. Platt

Last summer a membership poll indicated that the membership almost unanimously approved incorporating WAP. The Board has taken the first steps toward incorporation by registering with the Maryland Secretary of State and adopting a set of Bylaws.

Unfortunately, the Bylaws for our new corporation "Washington Apple Pi, Ltd." are more legalistic than those of our old club. Some of the passages were copied out of the Maryland Corporation Law. But don't let the legalisms prevent you from reading what will be the key document to WAP's new structure.

Although the new WAP Ltd is born, the Board of Directors seeks membership approval of the incorporation and these new Bylaws in the form of permission to transfer our club's assets to the new corporation. We plan to present the new Bylaws and take this formal vote at the October meeting.

Most of the new Bylaws are based on our old club Bylaws. However, there are many new provisions. We are calling the Newsletter the "WAP Journal" to reflect its professional quality. Criteria for forming new SIGS are included for the first time. Plus corporate boiler plate on indemnification, notice of meetings, and tax exempt status now clutter what used to be a readable document.

We owe a debt of thanks to Jim Burger, Susan Freund and the law firm of Shaw, Pittman, Potts & Trowbridge for donating their time to work on our incorporation project.

BYLAWS OF
WASHINGTON APPLE PI, LTD.

ARTICLE I - NAME

The name of the corporation is Washington Apple Pi, Ltd. (hereinafter referred to as "WAP" or the "Corporation").

ARTICLE II - PURPOSES

WAP is organized for such charitable and educational purposes as may qualify it for exemption from federal income tax under Section 501(c)(3) of the Internal Revenue Code of 1954, as amended (or the corresponding provision of any future United States internal revenue law). More specifically, such purposes include, but are not limited to, mutual learning and education of members of the public who share an interest in computers.

ARTICLE III - PROHIBITED ACTIVITIES

No part of the net earning of the Corporation shall inure to the benefit of or be distributable to its members, directors, officers, or other private persons, except that the Corporation shall be authorized and empowered to pay reasonable compensa-

tion for services rendered and to make payments and distributions in furtherance of the purposes set forth in Article II hereof. No substantial part of the activities of the Corporation shall be the carrying on of propaganda, or otherwise attempting to influence legislation, and the Corporation shall not participate in, or intervene in (including the publication or distribution of statements), any political campaign on behalf of any candidate for public office. The Corporation shall not carry on any other activities not permitted to be carried on (a) by a corporation exempt from federal income tax under Section 501(c)(3) of the Internal Revenue Code of 1954, as amended (or the corresponding provision of any future United States internal revenue law) or (b) by a corporation, contributions to which are deductible under Section 170 of the Internal Revenue Code of 1954, as amended (or the corresponding provision of any future United States internal revenue law).

No member, director, or officer of the Corporation shall be financially interested, directly or indirectly, in any agreement relating to the operations conducted by the Corporation, nor in any transaction for furnishing services, facilities, or supplies to the Corporation for compensation, unless the fact of such interest be known to the Board of Directors and unless such agreement or transaction shall be authorized by the Directors who have no interest, direct or indirect, in such agreement or transaction.

ARTICLE IV - OFFICES

The post office address of the the principal office of the Corporation is P.O. Box 34511, Bethesda Maryland 20817. The Corporation may maintain additional offices at other places as the Board of Directors chooses to designate.

ARTICLE V - MEMBERSHIP

SECTION 1. CLASSES OF MEMBERSHIP. There shall be three classes of membership: individual, family and educational institution. Members of each class shall have equal voting powers. Any member of the immediate household holding a family membership shall be entitled to all privileges of membership, except that the family membership is entitled to a single WAP Journal and a single vote. Institutional members shall exercise the privileges of membership through a single individual.

SECTION 2. ELIGIBILITY. Membership in the Corporation is open to any person or non-profit educational institution interested in computers. New members may be required to pay an initiation fee if prescribed by the Board of Directors.

SECTION 3. DUES. The annual dues for each class of membership shall be determined by

contd.

the Board of Directors.

SECTION 4. LIMITATION OF AUTHORITY. Except as otherwise specified in these Bylaws, no member shall act in the name of the Corporation or take any action which would bind the Corporation.

SECTION 5. RESIGNATION. Any member may resign from membership in the Corporation by submitting a written resignation to the Secretary of the Corporation, but members shall not be entitled to a refund of any portion of the annual dues paid prior to such resignation.

SECTION 6. EXPULSION. A member may be expelled from membership in the Corporation for (a) failure to pay the annual dues for more than thirty (30) days after the due date thereof, or (b) engaging in activities prohibited by these Bylaws or any rule or practice adopted by the Board of Directors. Before any member may be expelled for engaging in prohibited activities, written charges specifying the alleged cause shall be filed with the Secretary and a copy thereof shall be served on the member charged and he shall be given the opportunity to correct such conduct. If such member continues to engage in prohibited activities, he may be expelled at any meeting of the Board of Directors at which there is a quorum by the affirmative vote of two-thirds of those present and voting.

ARTICLE VI - MEMBERSHIP MEETINGS

SECTION 1. REGULAR MEETINGS. Regular Membership Meetings shall be held on the fourth Saturday of each month, unless otherwise determined by the Board of Directors. Notice of these meetings, stating the date and hour of the meeting and the place where it is to be held, shall be by publication in the WAP Journal.

SECTION 2. SPECIAL MEETINGS. Special Membership Meetings may be called by: the President, the Board of Directors, or a signed petition of fifty (50) members. Notice of a Special Membership Meeting, stating the date and hour of the meeting and the place where it is to be held, shall be given by publication in the WAP Journal or by a mailing to all members and shall state the purpose of the meeting. No business other than that stated in the notice shall be transacted.

SECTION 3. QUORUM. One hundred members in good standing must be present at a Regular or Special Membership Meeting in order to conduct business.

SECTION 4. PROXIES. No member may vote by proxy at a Regular or Special Membership Meeting.

ARTICLE VII - BOARD OF DIRECTORS

SECTION 1. POWERS. The business and affairs of WAP shall be managed by its Board of Directors, which may exercise all powers of the Corporation and perform all lawful acts and things which not by law, the Article of Incorporation, or these Bylaws directed or required to be exercised or performed by or are conferred upon or reserved to the members.

SECTION 2. COMPOSITION. The Board of Directors shall consist of 13 members: the President, the Vice-President, the Secretary, the Treasurer, the Past-President, the Editor of the WAP Journal, the Head Librarian, and six Directors-at-Large. The Directors-at-Large shall be elected annually pursuant to Article IX hereof. The Past-President shall be the individual who last served as the President; provided, however, that if an individual is entitled to serve on the Board of Directors as an Officer or Director-at-Large, he shall not also serve as Past-President. In that case, the Past-President shall be the predecessor to the individual who last served as the President. If an individual is entitled to serve on the Board of Directors by virtue of election or appointment to more than one position, then such individual shall be entitled to only one vote and the second Director position shall remain vacant.

SECTION 3. TERM. Directors shall serve for a one year term beginning on July 1.

SECTION 4. REMOVAL. A Director may be removed at any time, with or without cause, either at a Regular Membership Meeting or at a special meeting called for such purpose.

SECTION 5. VACANCIES. Upon the resignation or removal of a Director-at-Large, the Board of Directors may, by majority vote, appoint a member to serve for the remainder of the Director's term.

SECTION 6. REGULAR MEETINGS. Regular meetings of the Board of Directors shall be held monthly, unless otherwise determined by the Board of Directors. Regular meetings shall be held, at such time and place as may be determined by the Board of Directors, without further notice.

SECTION 7. SPECIAL MEETINGS. Special meetings of the Board of Directors may be called by the President, the Vice-President or two or more Directors on seven (7) days notice to each Director, if such notice is delivered personally, by telegram or by telephone; or on fourteen (14) days notice if sent by mail or published in the WAP Journal. No business other than that stated in such notice shall be conducted.

SECTION 8. CONDUCT OF MEETINGS. The President, or in his absence the Vice-President, shall preside at meetings of the Board of Directors. The Secretary of the Corporation, or such other person as the presiding officer may appoint, shall act as Secretary of the meetings. A majority of the Directors currently in office (not counting vacancies) shall constitute a quorum. Meetings of the Board of Directors shall be open to all the Corporation's members, except that upon majority vote, the Board of Directors may enter executive session to transact business which the Board determines to be confidential.

SECTION 9. ACTION BY CONSENT OR BY TELEPHONE. Any action required or permitted to be taken at any meeting of the Board of Directors may be taken without a meeting if a written consent to such action is signed by all members of the Board of Directors and such written consent is filed with the
contd.

minutes of the proceedings of the Board. The Board of Directors may participate in a meeting by means of a conference telephone or similar communications equipment by means of which all Directors participating in the meeting can hear each other at the same time. Participation by such means shall constitute presence at such meeting.

ARTICLE VIII - OFFICERS

SECTION 1. DESIGNATION. The officers of the Corporation shall be the President, Vice-President, Secretary, Treasurer, Editor of the Journal, Head Librarian, Bulletin Board Operator, and Director of Group Purchases. The President, Vice-President, Secretary and Treasurer shall be elected annually by the members pursuant to Article IX hereof.

SECTION 2. PRESIDENT. The President shall be the chief executive officer of the Corporation, and shall have general charge of the business, affairs and property of the Corporation with general supervision over its other officers and agents. The President shall preside at all Membership and Board of Directors meetings and shall see that all resolutions of the Board of Directors are carried into effect. The President shall have the general powers and duties of supervision and management usually vested in the office of president of a corporation. The President shall submit an annual report describing the past year's activities of the Corporation to the membership, either in person or through the WAP Journal.

SECTION 3. VICE-PRESIDENT. The Vice-President shall, in the absence of the President, or in the event of the President's disability, perform the duties and exercise the powers of the President. The Vice-President shall also serve as the administrative assistant, oversee and coordinate the efforts of all committees and special interest groups, and perform other duties designated by the President or the Board of Directors.

SECTION 4. SECRETARY. The Secretary shall give, or cause to be given, notice of Special Membership Meetings and of Special Board of Directors meetings. The Secretary shall keep the minutes of the meetings of the members and the Board of Directors. The Secretary shall send copies of the minutes of all meetings to the Board of Directors and shall also see that the books, reports, statements, and all other documents required by law are properly kept and filed. The Secretary shall perform such other duties as may be designated by the President or the Board of Directors. The Secretary may appoint one or more assistants.

SECTION 5. TREASURER. The Treasurer shall have custody of the corporate funds and other valuable effects, shall keep full and accurate accounts of receipts and disbursements in books belonging to the Corporation and shall deposit all monies and other valuable effects in the name and to the credit of the Corporation in such depositories as may be designated by the Board of Directors. The Treasurer shall have the authority to disburse checks as provided in

Article XII hereof. The Treasurer shall maintain a system of internal fiscal control and shall report regularly to the Board of Directors on the expenses and financial condition of the Corporation. The Treasurer shall perform such other duties as may be designated by the President or the Board of Directors. The Treasurer may appoint one or more assistants.

SECTION 6. EDITOR OF THE JOURNAL. The President shall appoint an Editor of the WAP Journal, who, subject to the general policy direction of the Board of Directors, shall be responsible for its editorial and advertising content and publication on a regular basis. The editor may appoint one or more assistants.

SECTION 7. HEAD LIBRARIAN. The President shall appoint a Head Librarian, who shall organize the Corporation's collection of programs, computer media, books and periodical collections, subject to the general policy direction of the Board of Directors. The Head Librarian may appoint one or more assistants.

SECTION 8. DIRECTOR OF GROUP PURCHASES. The President shall appoint a Director of Group Purchases who shall maintain a close and continuing interest in computer products and accessories. The Director of Group Purchases, subject to the general policy direction of the Board of Directors, shall negotiate arrangements for the purchase of such equipment by members. The Director of Group Purchases may sign checks on behalf of the Corporation in amounts not in excess of $500. The Director of Group Purchases may appoint one or more assistants.

SECTION 9. BULLETIN BOARD OPERATOR. The President shall appoint a Bulletin Board Operator who shall maintain the Corporation's community bulletin board system, subject to the general policy direction of the Board of Directors. The Bulletin Board Operator may appoint one or more assistants.

SECTION 10. VACANCIES. If the office of President, Vice-President, Secretary or Treasurer becomes vacant for any reason, the vacancy shall be filled by the Board of Directors. If the office of Editor of the Journal, Head Librarian, Bulletin Board Operator or Director of Group Purchases becomes vacant for any reason, the vacancy shall be filled by the President. Any officer elected or appointed to fill a vacancy shall hold office until the election and qualification or appointment of his successor.

ARTICLE IX - ELECTIONS

SECTION 1. NOMINATIONS. Nominations for President, Vice-president, Secretary, Treasurer, and Director-at-Large shall be submitted during the month of April. The nominations may be made at the April Regular Membership Meeting or sent by mail to the Secretary through the the Corporation's office.

SECTION 2. ELIGIBILITY. Only members in good standing may be candidates for elective office. An individual may be a
contd.

candidate for more than one office, except that no one person may be a candidate for President and Secretary. Any dispute regarding the eligibility of a candidate shall be resolved by the Board of Directors.

SECTION 3. BALLOTING. An election shall be completed during the month of June. Only members in good standing as of May 2 may vote in the election. A ballot shall be mailed or hand-delivered to each member in good standing. Each ballot may cast one vote for each office, except that up to six votes (noncumulative) may be cast for Director-at-Large. Prior to each annual election, the Board of Directors shall establish a deadline for the receipt of ballots.

SECTION 4. RESULTS. In the event that any candidate for Director-at-Large is also elected as an officer, such candidate shall not be eligible for election as a Director-at-Large. The six eligible candidates for Director-at-Large with the greatest number of votes shall be elected. The election tellers shall certify the results of the election to the membership prior to July 1 of each year.

SECTION 5. INTERIM OFFICERS AND DIRECTORS-AT-LARGE. Effective as of the date of the adoption of these Bylaws, the President, Vice-President, Secretary, Treasurer, Past-President, Editor of the Journal, Head Librarian, Bulletin Board Operator, Director of Group Purchases and Directors-at-Large of the Corporation shall be those presently serving as such officers and directors of Washington Apple Pi, an unincorporated association. Such officers and directors shall serve until elections are held in 1983 or until their successors have been chosen and shall qualify.

ARTICLE X - SPECIAL INTEREST GROUPS

SECTION 1. FORMATION. The Board of Directors may establish special interest groups (SIGS) to provide a forum for exploring topics of special concern to a portion of the membership. Among the factors to be considered by the Board of Directors in recognizing a new special interest group are: whether the group conducts regularly scheduled meetings which it announces in the WAP Journal and whether it has at least five members.

SECTION 2. ORGANIZATION. Each special interest group shall elect its own Chairman from among its members and shall publish an account of its activities in the WAP Journal.

SECTION 3. LIMITATION OF AUTHORITY. No special interest group shall act in the name of or take any action which would bind the Corporation.

ARTICLE XI - COMMITTEES

SECTION 1. AUDIT COMMITTEE. The Audit Committee shall be appointed by the Board of Directors to select the Corporation's accounting firm and to review the financial records and reports of the Corporation. The President, Treasurer, Head Librarian, Director of Group Purchases, or other officer receiving or disbursing funds of the Corporation may not serve as a member of the Audit Committee.

SECTION 2. OTHER COMMITTEES. The Board of Directors may appoint other standing or special committees.

ARTICLE XII - FISCAL CONTROL

SECTION 1. DISBURSEMENTS. Disbursements over $25.00 shall be made only by check. Any disbursement over $5.00 shall be supported by voucher or receipt. All checks, drafts notes and evidence of indebtedness of the Corporation shall be signed by the Treasurer and either the President or the Vice-President. However, the Director of Group Purchases and the Treasurer shall each have authority to sign checks in an amount not to exceed $500.00.

SECTION 2. ANNUAL BUDGET. Prior to the commencement of each fiscal year, The Board of Directors shall prepare an estimated budget for the following fiscal year for the approval of a majority of those present and voting at a Membership Meeting.

SECTION 3. FISCAL YEAR. The fiscal year of WAP shall be from July 1 to June 30.

SECTION 4. CAPITAL EXPENDITURES. Capital expenditures in excess of $1,000.00 must be approved by a majority of those present and voting at a Membership Meeting. Approval of a line item in the annual budget shall constitute membership approval of a capital expenditure.

ARTICLE XIII - NOTICE

SECTION 1. FORM OF NOTICE. Whenever under the provision of law, the Articles of Incorporation or these Bylaws, notice is required to be given to any Director or member, such notice may be given by publication in the WAP Journal or in writing, by mail addressed to such Director, officer or member, at his post office address as it appears on the records of the Corporation. Such notice shall be deemed to be given at the time it is deposited in the United States mail. Notice may also be given personally, or by telephone or telegram.

SECTION 2. WAIVER. Whenever notice is required under the provision of law, the Articles of Incorporation, or these Bylaws, a written waiver of the notice, signed by the person entitled to the notice, whether before or after the fact, shall be deemed to be the eqivalent of such notice. Any member or Director who attends a meeting, without protesting the commencement of the meeting or the lack of notice shall be conclusively deemed to have waived notice of such meeting.

ARTICLE XIV - INDEMNIFICATION.

The Corporation shall, to the extent legally permissible, indemnify and hold harmless any person serving or who has served as a director, officer, or duly authorized agent of the Corporation against all liabilities and expenses, including amounts paid in satisfaction of judgments, in compromise or as fines and penalties, and counsel fees, reasonably incurred by him in connection

contd.

with the defense or disposition of any claim, action, suit or other proceeding whether civil or criminal, in which he may be involved or with which he may be threatened, while serving or thereafter, by reason of his being or having been such a director, officer or agent, except with respect to any matter as to which he shall have been adjudicated in any proceeding not to have acted in good faith in the reasonable belief that his action was in the best interests of the Corporation.

Expenses, including counsel fees, reasonably incurred by any such director, officer, or agent in connection with the defense or disposition of any such claim, action, suit or other proceeding may be paid from time to time by the Corporation in advance of the final disposition thereof upon receipt of an undertaking by such individual to repay the Corporation the amounts so paid if it ultimately determined that _indemnification of such expenses is not authorized herein. The right of indemnification hereby provided shall not be exclusive of or affect any other rights to which any such director, officer, or agent may be entitled. As used in this Article, the terms "director", "officer", and "agent" include their respective heirs, executors and administrators.

The Corporation may, at the election of the Board of Directors, purchase and maintain insurance on behalf of any person who is or was a director, officer, employee or agent of the Corporation or who is or was serving at the request of the Corporation as a director, officer, or agent of another corporation, partnership, joint venture, trust, or other enterprise against any liability asserted against him and incurred by him in or arising out of his position, whether or not the Corporation would be obligated or empowered to indemnify him against such liability under this Article XIV.

ARTICLE XV - AMENDMENT

The Board of Directors or the membership may amend these Bylaws at any Regular Meeting by an affirmative two-thirds majority of those present and voting. Notice of all Bylaw amendments shall be published in the WAP Journal.   ᘐ

into the auditorium.

The main meeting is conducted, covering club business, and discussing relevant topics, after which there is usually a presentation about a subject of interest to the majority of members. These presentations are interesting, often humourous and always informative.

The show is not over yet. When this part of the meeting is over, the SIGs (Special Interest Groups) wander off to separate rooms of the USUHS building to conduct their own mini-meetings. For a listing of the many SIGs and their chairmen see the first page in any WAP newsletter.

Do yourself a favor; attend the next meeting. You won't be disappointed. After attending a meeting or two, you may be interested in volunteering your services. If you are, we can use the help!   ᘐ

APPLICANTS SOUGHT

Washington Apple Pi wants to fill two part-time positions, described below. If you are interested, please forward a description of your relevant experience to:

David Morganstein
13329 Woodruff Ct.
Germantown, Md. 20874.

Please include any ideas you may have about the performance of the position and desired hourly compensation. Applications for either position should be received by October 20, 1982.

1. The first job has two primary responsibilities: maintaining the WAP office, and word processing preparation for the newsletter. The job is expected to require 25 to 30 hours per week and may be filled by one or more individuals. The applicant(s) must be familiar with the use of an APPLE computer.

● Maintaining the WAP Office. We desire to keep the Office open for at least 20 hours per week, to follow a fixed schedule agreed to by the accepted applicant - the schedule should include at least one evening. This person will be responsible for opening, maintaining and closing the Office and insuring its security and cleanliness while open. S/he will review correspondence received, forwarding it to the appropriate individual. S/he will fill requests for membership information and for back issues of periodicals. S/he will answer the phone and must be able to assist callers with WAP-related questions or put them in contact with someone should the question be appropriate for a hotline response. S/he will maintain the membership roster using the APPLE computer, making additions and corrections as needed. S/he will print mailing labels, as needed. S/he should become experienced in bulk mail regulations and will arrange required bulk mailings, including that of the monthly newsletter. S/he will coordinate reproduction of materials needed, such as copy for the orientation package sent to new members. Further, s/he will work with the Treasurer to insure that billings are sent periodically. One additional responsibility will be to assist the periodicals Librarian in keeping the library organized.

● Word-processing for Newsletter. In addition to the above, s/he must be familiar with a wide variety of word processing software, such as Screen Writer II, Supertext and Apple Writer. S/he will receive copy for the magazine and must format it for inclusion in the newsletter. The copy may come in either hardcopy or machine readable form. S/he will be responsible to the Editor for this activity.

2. Magazine Lay-out. To assist the WAP in the preparation of the newsletter, a person is needed to perform the job of laying out the magazine contents, and will be responsible to the Editor. This person should have prior experience in assembling a periodical. This task is expected to require 40-50 hours of work per month.   ᘐ

SOFT VIEWS: NEW RELEASES

by David Morganstein

WATSON (Omega MicroWare, Inc., 222 So. Riverside Plaza, Chicago, Il. 60606, (312) 648-1944). When the Inspector needs help, who does he call on? Why Dr. WATSON, of course! With this companion title, the people who brought you Locksmith and the Inspector bring another set of utilities for general disk exploration. These utilities include: disassembly of machine code with ASCII display, free sector count added to their disk Map display, reconstruct the VTOC, free up tracks 1 and 2 for storage (kill DOS), display of a selected file's sectors, moving forward or backward (can easily locate blown sectors of an inaccessible file), compare two disks sector by sector, and several printer options for obtaining hard copy of the displayed information. You can even add up to 9 machine language commands of your own! WATSON runs at $D000 and cannot work without the Inspector (which resides at $D800). The pair of programs can either be merged with your INTBASIC file (for those with memory cards) or can come on ROM for your Motherboard or Integer ROM card. A brief but quite adequate 16-page manual is included. A command summary card is provided which includes the Inspector's features, as well. Nice touch! I find the functions of the Inspector more valuable than the new additions of WATSON but they are quite handy. $49.95

Printographer (Southwestern Data Systems, 10159-I Mission Gorge Road, Santee, CA. 92071 (714) 562-3670). Need a Hi-res graphics dump routine, one that allows cropping of pictures? The Printographer comes equipped to service a wide variety (seventeen models) of printers (including my NEC 8023-A) and many common interface cards (not my PKASO card!). However, the program allows you to provide your own software driver if your interface is not among the eight interface drivers they provide. In addition, there is a method of describing your printer's characteristics, if it is not on the list provided. After configuring the system for your equipment, the easy to follow menu permits you to load a Hi-res screen, edit a Hi-res screen or exit. Editing features include cropping to any sub-rectangle or constructing a diamond or oval border around the image. At print time you can select: inverse or normal image, magnification from 1 to 9 times, horizontal or vertical format and indentation for horizontal shifting. For programmers, routines are provided to incorporate Hi-res dumps into your own software, a nice touch! A clearly written, 27-page manual is provided. While the disk is protected, SDS uses their novel copy system which allows you to make three copies for back-up purposes. (In fact, they urge you not to use the original, rather to back it up first).

To be completely candid, the description of how to incorporate your own interface driver was not quite accurate. After examining the machine code, I spoke with the author, Stephen Billard. He confirmed that an errata sheet included in the newer packages explained the discrepancy. While the discussion of how to write printer or interface drivers is a bit tricky, Stephen indicated that SDS preferred to be contacted by consumers to insure that they were able to get things working. The ability to use the Printographer routines in your own program and to write drivers for virtually any printer/interface card makes this package an excellent idea for anyone needing a graphics dump program. $49.95
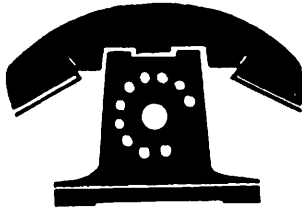
Rendezvous (Edu-Ware, Box 22222, Agoura, CA. 91301) When I discovered that Rendezvous was written by Wes Huntress at JPL (Jet Propulsion Labs), I knew that this space flight simulation was bound to be realistic. Wes' previous simulation, Saturn Navigator was sold by Sub-Logic, who created the 3-D graphics package used by Wes to generate the view of Saturn out the space ship window. I am not sure if the same package is used in Rendezvous, but the simulation of take off, rendezvous with an orbital space craft and docking is excellent. The 21 page "Operations Manual" that accompanies the simulation is well organized and should make it easy for even novice space pilots to lift-off. (Getting into correct orbit and docking will take a little work!) The simulation contains four phases which can be run in order, as they would happen, or a choice made to practice a particular feature of the flight. The four phases are: liftoff, orbit, approach and alignment/docking. Liftoff includes a Hi-res display of your ship on the launch pad. As you lift off, your task is to attain orbit using the minimum fuel. You will see your first stage booster drop away at the proper moment. After liftoff, the view changes to allow viewing your current path around the earth. Mid-flight corrections are needed to attain an orbit that will cross that of the orbiting vehicle with which you will dock. The third phase finds you viewing the orbital vehicle during approach and again changing course and velocity. In the last stage you will see the vehicle approach as you align for the docking manuever. You will need to use all fingers on both hands to accomplish the final step...good luck (I needed it!!!) $34.95   ✪

****

The October meeting will be on the 5th Saturday, instead of the 4th. The date is October 30. Mark it on your calendar.

****

QUESTIONNAIRE

We have received some interest in establishing a "consumer advocate" group within Washington Apple Pi. In order to establish the need and framework of such an organization, we would like your comments on the following items:

1. Hardware: Have you ever had serious difficulties with any dealers in purchasing hardware products for your APPLE II?

          _____yes   _____no

   Comments _____

   _____

2. Service: Have you ever had serious difficulties with any dealers in obtaining service on any hardware products?

          _____yes   _____no

   Comments _____

   _____

3. Software: Have you ever had any serious difficulties with any dealers or manufacturers in obtaining software or in getting advice on software use?

          _____yes   _____no

   Comments _____

   _____

4. Which of the following functions do you think would be of service to the club membership?

	yes	no	no opinion
mediate customer/dealer disputes	____	____	____
provide guidance on filing complaints with government agencies/ courts	____	____	____
publish list of local dealers	____	____	____
publish list of complaints of members about hardware and software	____	____	____
publish list of local hardware repair facilities and the peripherals they service	____	____	____

5. Any additional comments? _____

   _____

   _____

6. Please return questionnaire to:
          Washington Apple Pi
          Attn. Mark Pankin   Phone: (703) 370-9219 (Home)
          P. O. Box 34511   (703) 243-2210 (Work)
          Bethesda, MD 20817

WASHINGTON APPLE PI MAIL ORDER FORM

Washington Apple Pi has a program library, and disks are available for purchase. You may pick them up at the monthly 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.) If you plan to pick them up at the meeting, the order must be received by the 1st of the month, and must be paid in advance.

PROGRAM DISKETTES:
Picked up at meeting: Members $ 5.00; Non-members $ 8.00.
Mailed to you (1st 5):   Members $ 7.00; Non-members $10.00.
     "   (remainder)   " $ 5.00;   " $ 8.00

DOS 3.2
Volume   1 Utilities I   ( ) *Volume 188 Caves of Treas. Island ( )
Volume   2 Utilities II   ( ) *Volume 189 Furioso   ( )
Volume   3 Games I   ( ) *Volume 190 The Magic Kingdom   ( )
Volume   4 Games II   ( ) *Volume 191 The Tomb of Molinar   ( )
Volume   5 Games III   ( ) *Volume 192 Lost Island of Apple   ( )
Volume   6 Games IV   ( ) *Volume 193 The Abductor's Quarters( )
Volume   7 Games V   ( ) *Volume 194 The Quest for Trezore ( )
Volume   8 Utilities III   ( ) *Volume 195 The Underground City   ( )
Volume   9 Educational I   ( ) DOS 3.3
Volume 10 Math/Science   ( )   Volume 100 Utilities A   ( )
Volume 11 Graphics I   ( )   Volume 101 Utilities B   ( )
Volume 12 Games VI   ( )   Volume 102 Games A   ( )
Volume 13 Games   ( )   Volume 103 Merry Christmas   ( )
Volume 14 IAC Utilities IV   ( )   Volume 104 Business A   ( )
Volume 15 Games VII   ( )   Volume 105 FIG-FORTH/Utilities   ( )
Volume 16 Utilities V   ( )   Volume 106 Science Engineering   ( )
Volume 17 Graphics II   ( )   Volume 107 Games B   ( )
Volume 18 Educational II   ( )   Volume 108 IAC 10 (Graphics)   ( )
Volume 19 Communications   ( )   Volume 109 IAC 11 (Applesoft Tut.)( )
Volume 20 Music   ( )   Volume 110 Personal/Education   ( )
Volume 21 Apple Orchard   ( )   Volume 111 Games C   ( )
Volume 22 Utilities VI   ( )   Volume 112 Utilities C   ( )
Volume 23 Games VIII   ( )   Volume 113 Business B   ( )
Volume 24 Games IX   ( )   Volume 114A Adventure - INIT   ( )
Volume 25 Utilities VII   ( )   Volume 114B Adventure   ( )
Volume 26 Stocks/Investments   ( )   Volume 115 IAC 12/13 Miscellaneous( )
Volume 27 Math   ( )   Volume 116 IAC 14 Micromodem II   ( )
Volume 28 Planetfinder   ( )   Volume 117 Picture Packer   ( )
Volume 29 Utilities VIII   ( )   Volume 118 Utilities D   ( )
Volume 30 Games X   ( )   Volume 119 IAC 15 Miscellaneous   ( )
Volume 31 Plot Utilities   ( )   Volume 120 IAC 16 Miscellaneous   ( )
Volume 32 Games XI   ( )   Volume 121 ABBS Program   ( )
Volume 33 Accounting   ( )   Volume 122 IAC 17 Miscellaneous   ( )
Volume 34 Solar Tutor   ( )
Volume 35 Garden Managment   ( )
Volume 36 Games XII   ( )
Volume 37 Utilities IX   ( )   Volume 300 Pascal PIG0:ATTACH BIOS( )
Volume 38 Games XIII   ( )   Volume 301   PIG1:   ( )
Volume 39 IAC VII   ( )   Volume 302   PIG2:   ( )
Volume 40 IAC VIII   ( )   Volume 303   PIG3:   ( )
Volume 180 Dungeon Designer   ( )   Volume 304   PIG4:   ( )
Volume 181 Beginners Cave   ( )   Volume 305   PIG5:   ( )
*Volume 182 Lair of Minotaur   ( )   Volume 306   PIG6:   ( )
*Volume 183 Cave of the Mind   ( )   Volume 307   PIG7:   ( )
*Volume 184 Zyphur Riverventure   ( )   Volume 308   PIG8:   ( )
*Volume 185 Castle of Doom   ( )   Volume 309   PIG9:   ( )
*Volume 186 Death Star   ( )
*Volume 187 Devil's Tomb   ( )   Volume   EDSIG1 (Elem. Math)   ( )
                                          Volume   TUTORIAL (Adv.VisiCalc)( )
* Volume 181 required with these disks.
                                           Total Order = $_____

Check here if you want these shipped ____ (NOTE: ALLOW 6 TO 8 WEEKS FOR MAILING.)

NAME   _____

ADDRESS   _____

CITY, STATE ZIP _____

TELEPHONE   _____WAP MEMBERSHIP NO._____

Make checks payable to "Washington Apple Pi".
Send order to: Washington Apple Pi - ATTN: Librarian
                      P.O. Box 34511
                      Bethesda, MD 20817

WAP TUTORIAL REGISTRATION

A WAP tutorial on Assembly Language will be taught by Tom Warrick, who is our ABBS System Operator. It will be held on Saturday, October 16, 1982, 9:00 AM to 12:30 PM, in room 2017A at USUHS On Jones Bridge Road (on the campus of the Bethesda Naval Medical Center) in Bethesda, MD. Check the club phone and ABBS for any change in details.

The fee is $15 if you bring your APPLE; $30 if you do not.

Please add $5 to the fee if you are not a WAP member.

Name _____

Daytime Phone _____   Evening Phone _____

Check One;
  I will bring an Apple, etc. (fee $15, WAP Membership No. ____) Non-member + $5 ___
  I will not bring an Apple   (fee $30, WAP Membership No. ____) Non-member + $5 ___

Please return application and check made out to "Washington Apple Pi" to:

Steve Stern
6921 Old Stage Road
Rockville, MD 20852

INDEX TO ADVERTISERS