

\$1.50

Washington Apple Pi



Volume 2

December 1980

Number 12

Highlights

BLAISE AWAY
LINEAR SCALING IN
HI-RES GRAPHICS
DANA'S HEX LOADER
PRINTING OUT THE
HI-RES SCREEN

In This Issue

| | Page |
|--|------|
| MEMBERSHIP INFORMATION, EVENT QUEUE, EDITORIAL, SIG-NEWS MINUTES | 1 |
| LETTER TO THE EDITOR - SUSAN M. ZAKAR | 2 |
| ELECTRONIC COMMUNICATION SYSTEM: A REVIEW - STEVE HADLEY | 3 |
| CLASSIFIEDS | 3 |
| WASHINGTON APPLE DIGEST - DAVE EFRON | 4 |
| A REVIEW OF THE EPSON MX-80 PRINTER - BILL WURZEL | 5 |
| COMMENTS ON PRETTYLISTING - STEPHEN E. BACH | 6 |
| BLAISE AWAY - DR. WO | 7 |
| A PAGE FROM THE STACK: LIBRARIAN'S CORNER - DAVE MORGANSTEIN | 12 |
| THE ABBS, FROM ONE USER'S VIEWPOINT - GENEVIE URBAN | 12 |
| FLAVORS: LITTLE TIDBITS - BURTON S. CHAMBERS III | 13 |
| COMMENTS ON VISICALC - WALT FRANCIS | 14 |
| CONTEST SEEKS COMPUTER AID FOR THE HANDICAPPED: A REPRINT | 16 |
| LINEAR SCALING IN HI-RES GRAPHICS - BILL WURZEL | 17 |
| A BUG IN THE TOOLKIT ASSEMBLER - BILL WURZEL | 18 |
| DANA'S HEX LOADER - DANA J. SCHWARTZ | 18 |
| CHANGES FOR DOS 3.3 - DANA J. SCHWARTZ | 20 |
| QUESTIONS, QUESTIONS, QUESTIONS - MARK L. CROSBY | 21 |
| PRINTING OUT THE HI-RES SCREEN - HOWIE MITCHELL | 22 |
| EQUATIONS FOR (SOME OF) THE BESSEL GRAPHS - HOWIE MITCHELL | 24 |
| COLUMN-FORMATTER FOR APPLE PI - HOWIE MITCHELL | 27 |
| INFORMATION IN THE POST-GUTENBERG ERA: A REPRINT | 29 |
| THE DREADED COMPUTER: A REPRINT | 31 |

ComputerLand[®]

We know small computers.



 **apple[®] computer**
Sales and Service

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

OFFICERS & STAFF

President -Bernard Urban (301) 229-3458
 Vice President -Rich Wasserstrom (703) 893-9147
 Treasurer -Bob Peck (301) 468-2305
 Secretary -Dana Schwartz (301) 621-2894
 Members-at-Large -Mark Crosby (202) 488-1980
 -Sandy Greenfarb (301) 674-5982
 -Hersch Pilloff (301) 292-3100
 Editor -Bernard Urban (above)
 Associate Editor -Rich Wasserstrom (above)
 -Genevie Urban (301) 229-3458
 Librarian -David Morganstein (301) 972-4263
 Group Purchases -Howard Lefkowitz (301) 649-3373
 Membership Chmn. -Tom Jones (301) 460-8773
 SIG Chairmen:
 ASMSIG -Jim Rose (301) 730-2230
 EDSIG -Chuck Philipp (301) 924-2354
 NEWSIG -Al Weiner (301) 468-0663
 Pascal -Tom Woteki (202) 547-0984
 SIGAMES -Al Gass (703) 371-3560

Washington Apple Pi
 P. O. Box 34511
 Washington, D.C. 20034
 (301) 468-2305

ABBS (301) 983-9317

Membership dues for Washington Apple Pi will be changing for the calendar year beginning January 1981. The Executive Board is recommending an increase to \$18.00 per year, but this is subject to approval by the membership at the December meeting. Renewal information, based on this proposal, is included at the back of this newsletter. New applicants should start their membership with the new year. A revised membership form will be available soon. Please request a form by calling our club phone and leaving a message with your name and address, or by writing us at the above address.

EVENT QUEUE

Because of the holidays, the December meeting of Washington Apple Pi will be on the third Saturday, December 20 (our regular meeting date is the fourth Saturday), at 9:30 AM at George Washington University, usually in Building C on G Street at 23rd Street, NW. (To be sure of the exact location call the club phone or ABBS the week of December 15.)

The business meeting will include a vote on membership dues for 1981 and on various items of the budget. Plans are being made for a social gathering after the formal meeting, with refreshments and a chance to say happy holidays to your fellow club members.

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

NOVAPPLE meets on the 2nd Saturday of the month at 1:00 PM at Kings Park Library on Burke Lake Road in Fairfax County. The December meeting is on the 13th and the January meeting will be on the 10th.

EDITORIAL

May you all have pleasure, peace and satisfaction in this holiday season. Enjoy your APPLES.

ABBS NOTES

If you want to sign onto the Washington Apple Pi ABBS, please call the club phone, 468-2305, and leave your name and WAP number. This message will be forwarded to John Moon who will take care of signing you up. Previously, we had asked that you send a postcard, but we think that this will be a better method.

SIG-NEWS

SIGAMES is a group of people sharing a mutual interest in developing and using games for the APPLE and what they contribute to the learning process. The general meeting of this group is held at a location announced at and following the Washington Apple Pi monthly meeting.

There was a great deal of enthusiasm shown by members at last month's SIGAMES meeting for constructing their own joy sticks. The December meeting will be devoted to the design and implementation considerations for this project. If anyone is interested in constructing APPLE II joy sticks as a group project, they should attend the December meeting of this group. Now is the time to get involved so that design requirements and supply purchases can be coordinated. If you cannot attend the SIGAMES December meeting, call Bill Bowie at (301) 924-3455 to let us know of your interest. Bill Bowie and Brian Dormer will be chairing this project.

Alban Gass

The Pascal SIG meets on the third Thursday of each month at 7:30PM. Programs thus far have consisted of round table discussions and presentations by members on topics of mutual interest. November's meeting centered on a checkbook filer program and the use of pointers in the program. In December Bill Wurzel will lead the discussion and January will feature Burt Chambers.

All members of Washington Apple Pi are cordially invited to attend. Watch the club's ABBS or call me for info--547-0984.
 Dr. Wo

ASMSIG (Assembly Language SIG) will meet on Monday December 15 at 7:30 PM at the University of Maryland Computer Science Center. The topic is the APPLE Assembly Language Subroutine Library.

NEWSIG, the new user's SIG, will meet after the main WAP meeting on the third Saturday of December at GWU.

MINUTES

EXECUTIVE BOARD MEETING

The Executive Board meeting of November 12, 1980 was called to order at 7:15 PM at the home of the President, with 12 persons in attendance.

The President elicited comments on a proposed statement of club directions to be included in the November newsletter. Comments from those present were reflected in the final version. In related matters, the Vice President will oversee the formation of the financial system, Howard Lefkowitz will pursue the establishment of a documentation library at a public place, and available courses will be announced at the next regular meeting to complete details of enrollment.

The December meeting was set for the third Saturday instead of the fourth due to holiday conflicts. It was decided that there was a need for a revised advertising rate schedule for the newsletter, and a policy for overseas mailing rates. There was great enthusiasm for the "new look" of the newsletter, and the Editor was encouraged to continue as funds permit.

In an effort to reduce distractions created by sales at the regular meetings, the Board decided to stop all sales during the formal presentation, and also to offer a financial incentive to prepay by mail disk orders picked up at the meetings.

The meeting was adjourned at 10:45 PM.

GENERAL MONTHLY MEETING

The Washington Apple Pi meeting of November 22, 1980 was called to order at 9:35AM by the President, with approximately 105 members in attendance.

The majority of the business session was spent in reviewing the contents of the President's Editorial in the November newsletter. The club's future plans in such areas as ABBS, the Newsletter, and the Finance Committee were discussed.

A copy of the 1981 proposed budget was distributed to the membership for their study. Discussion and vote will be held at the December meeting.

After several announcements, the meeting was turned over to Stuart Milner, and Chuck and Nancy Philipp and their children, for an excellent series of discussions on APPLE education applications.

Dana J. Schwartz, Secretary

LETTER TO THE EDITOR

from Susan M. Zakar

This is a letter to the general membership of Washington Apple Pi concerning the extremely discourteous treatment of our speakers at the November meeting. The disrespect shown to them was unforgivable.

Instead of cooperating with the speakers' request to divide into interest areas for the session, fully 75 percent of the membership either left the meeting, began transacting club and/or personal business, or just created a disturbance by talking. I wish I could say that this was discouraged by the officers present, but unfortunately they were in some cases the instigators of the problem. Library programs were being sold, group purchases were being distributed, and so on. Certainly the membership could not have done more to humiliate and embarrass the speakers, who had spent a great amount of time and effort to prepare for the session. For those few who had the common courtesy to listen to them, the session was highly interesting and time spent wisely. Unfortunately the disruptive conduct of the majority hindered even their best efforts to communicate all that had been prepared. As the outgoing Program Chairman, I could only express sincere apologies to all of them, and share their justified anger at the situation. How will the next Program Chairman expect to have speakers participate in the session if the thanks they get is inattention and disruption? What reputation will Apple Pi have if this continues?

I hope that those involved in this sham will rethink their actions for coming sessions. I would also hope that the Board of Apple Pi and its membership will consider a revamp of meeting schedule. From here on, the Speaker(s) should be the FIRST order of business, and the ONLY order of business conducted at that time. NO OTHER CLUB BUSINESS has any right to be conducted while the speaker is presenting his or her topic. To that end, I might suggest the following agenda for club meetings:

- From 9:00 to 9:45 - library and group purchases.
- From 10:00 to 10:45 - start meeting and introduce speaker.
- From 10:45 on - business meeting.

I hope the membership will consider such a proposal.

To those who were not involved in the disruption last meeting, thank you. The accusations in this letter were not directed at you in any way, except perhaps to enlist your support in bringing the problems to a reasonable end.

President's Note: I agree with Susan. It was a rather poor performance on the part of many of the attendees, but I feel that I must take the blame. The meeting was lengthy, and probably boring to most. When I finally turned the meeting over to the guest speakers, people took the opportunity to leave, pick up diskettes and group purchases, meet with their friends, or listen to the lectures. We intend to change all this: Susan's ideas are good. Also, see Dave's PAGE FROM THE STACK. Never again!

ELECTRONIC COMMUNICATION SYSTEM

Ed Magnin, of the Telephone Software Connection, has come up with a new and, I think, exciting APPLE/Modem application. Of course, the concept of telecommunications is far from new or unique. The list of modem users within our own club is growing at an amazing rate, as are the various BBS's across the country. So what makes one system a stand out is the quality of its 'bells and whistles'. Ed's system, 'Electronic Communications System', gets all A's from this user.

Right up front, let me say that the use of the system is not free. There is a subscription fee and you must establish an account (American Express, Master Charge, or Visa) before going on-line. This fact, that it costs, seems to have caused several people to turn away without looking any further. Hence, this article.

The charges are \$3.75 per hour (minimum of \$6.25 per month) and a storage charge of \$0.04 per 2K per month. That, of course, is for non-peak hours, i.e., after 6PM Monday through Friday and all day Saturday, Sunday and Holidays. (The peak hours rate is \$16.50 per hour!)

So what do you get for your hard earned dollar? Well, for openers, you are placed in direct contact with other APPLE owners from all over the country (oh, yes, and a few TRS-80 and PET owners). That is a vast resource to draw from for the exchange of ideas, problems, opinions, etc.! Also, you are not limited to just discussing APPLES. The current list of special interest bulletin boards accessed through the ECS system includes:

| | |
|------------------|--------------------|
| Amateur Radio | Modem Questions |
| APPLE Questions | Open Forum |
| Atari | Photography |
| Financial | Practice |
| Games | Product Reviews |
| General Interest | Science Fiction |
| Legal | S100 |
| Mail Wanted | Telecommunications |
| Medical | TRS80 |

Also, there are about twenty State bulletin boards and the list is growing every week.

When you first sign on the system you will have a 'Welcome' letter waiting in your box. This is from Ed, or 'ADMIN' as he likes to call himself on-line. The letter will give you a brief introduction and directions on how to access a tutorial for the system. Since the network is so all encompassing, you can imagine that it is fairly complex and leaves the novice a bit bewildered. I sent off several 'Help!' messages to ADMIN and received the answers within seconds!

As I grew more accustomed to the whys and wherefores of the system, I ventured out to several bulletin boards and left messages (rather timid messages, as I wasn't convinced that I was not typing

a review by
Steve Hadley

into a void, and was feeling a little foolish!). When next I logged on, I found ten messages waiting in my box! These ranged from a review of a new data base manager I had inquired about, sent from a dentist in Seattle, WA, to a geneologist in California wanting to know if there were any Quakers in my family tree! It was on the system that I learned that a POKE 1784+Slot,0 will disable the micro-modem's lower to upper case translation.

Finally, added to the fact that there is a lot of information to be gleaned out there...it is also just plain fun!

So, if the idea of paying for this service bothers you, then this may not be the system for you. As for me, after investing a couple of thousand in this little toy, the idea of a few dollars more to help get the most out of it seems like an excellent investment.

For further information:

Call 347-1200 at either 110 or 300 baud. After you are connected, hit RETURN twice. When it asks "TERMINAL=", hit RETURN again. When it says "@", type "MAIL". When it asks "User Name?", type "NEW/TSC". When it asks "Password?", type "NEW/TSC" again. ☺

CLASSIFIEDS

FOR SALE: Lear-Siegler ADM-3A "dumb" terminal, six months old, perfect condition, lower case option. \$575. Call "Reds", (301) 666-1398.

FOR SALE: APPLE II + 48K, integer card, one disk drive, serial I/O card, communications card, CRT stand, carrying case, extended warranty, Sony 12" TV, over \$600 worth of software and games. Retail cost, \$3,500; will sacrifice for \$2,300. (Hardware only, \$1,850; Sony TV only, \$275; Software only, \$400.) Call Kirk Balcom, (703) 385-8389.

FOR SALE: 48K APPLE II, Applesoft card and DOS 3.3; D.C. Hayes modem; B and W TV; small library, including Appewriter and great games; literature and magazines; plus a few extras. ENTIRE system \$2200 cash! Call Scooter, weekdays after 4 PM, (301) 725-6251.

FOR SALE: IDS 440G Paper Tiger printer - never used, \$750. Call Raphael Thelwell, (301) 972-1614, evenings.

"It has been written that the great challenge in this world is not so much where we stand as the direction we are moving. To reach our port, we must sail sometimes with the wind, sometimes against it. But we must sail, not drift or lie at anchor."
--Ted Kennedy, Fall 1980.

WASHINGTON APPLE DIGEST

by Dave Efron

CREATIVE COMPUTING, November 1980

Computer Ambush, p28. Review of this software game that simulates infantry maneuvers. The author feels that the first product by Strategic Simulations, Inc. (Computer Bismarck), is a better game. Price not given.

Effective Documentation, p30. Describes the necessary components of a good documentation package: An introduction, quick-reference guide, detailed description of the program operation and use, and technical description of the program's design and programming.

Sorting, p35. This paper describes a method for fast sorting.

Systems analysis, p43. This article describes the process of systems analysis.

Interactive System Design, p56. A complicated message is presented by the author, dealing a lot with philosophies of computer system design. A number of generic and specific systems products are described in terms of their "virtuality." Advanced level of understanding is required.

Software Authoring, p64. Creative Computing gives advice to programmers thinking about marketing their work through this magazine. The programs ought to be original (i.e. not another Star-game), challenging, free of program errors, and ample instructions for use must be given. Listings are not accepted -- only tapes or discs.

Apple Graphics, p66. Part III of a series. Lesson and discussion for beginners.

Actor Languages, p74. Continuation of last month's article.

String Data Entry, p102. Using GET function as an alternative to INPUT. Elementary.

INFOWORLD, 10-13-80

Stock Market. The article deals with the

general popularity of micros on the stock market especially noting the reaction to the ads that Apple took to publicize its recent stock offerings. The response to the ad was good, with the minor exception of disagreement with Woz's and Jobs' claim to have invented the personal computer.

Apple Operating Systems. The history of operating systems for the Apple II computer has been one of cleverly designed patchwork solutions to problems that should have been foreseen. Indications are, however, that Apple has learned from the exercise, and is designing a sound operating system for the Apple III and future products. The article reviews in detail the progress from the non-disk system through DOS 3.1, 3.2, UCSD Pascal, and 3.3. The Apple III SOS is seen as the natural cumulation of the development.

Interlude, software review. Elementary. Interlude is the Apple II and TRS-80 program that suggests sexual activity given the preferences and attitudes of the participants. The heart of the program is the booklet that accompanies it, for that is where the actual activities are described (with a few exceptions).

Chem Lab Simulations, software review. From High Technology, two college-level chemistry lab simulations are reviewed. They are considered useful in high school and even in some elementary schools. The first simulation is three titration experiments; the second is two gas behavior demonstrations. The reviewer feels that both are very valuable for teachers.

The Voice, software review. From Muse software. For anyone who wants to add speech to Apple programs without using hardware, this interactive editor is a practical alternative. Users can build a vocabulary of words and phrases for use in BASIC programs. Editor programs are provided for both versions of Apple Basic. The evaluation is too brief to be really useful. A demo before purchase would be in order. ☺

* * * * * RING OUT THE OLD, RING IN THE NEW * * * * *

A REVIEW OF THE EPSON MX-80 PRINTER

by Bill Wurzel

The thing that impresses me most about the Epson MX-80 printer (besides the price) is the print image. The 9x9 dot-matrix impact printhead produces attractive, tack-sharp characters with true descenders. Unlike too many other printers I've seen, each print dot has the same intensity, giving a neat, balanced character. The printer, with the parallel interface option, retails locally for \$645!

PRINTHEAD: The printhead consists of a 9x9 dot matrix with an advertised life expectancy of 100 million characters. If and when it dies, you just replace it! Changing the printhead just requires moving one lever -- replacement printheads run around \$30. Character sizes are 40, 66, 80 and 132 characters per line and are software selectable. There are also dipswitches that permit you to select American, French, German, British and Japanese (Katakana) character sets - but you have to remove the base to get at the switches. The printing is bidirectional and the CPU controlling the printhead stepping motor has shortest-distance seeking logic. Print speed runs around 80 CPS. Printhead overheating doesn't seem to be a problem - I've run the printer continuously for over one hour without difficulty.

GRAPHICS: Although the MX-80 has its own "graphics" character set composed of all possible combinations of a six-cell rectangle, it does not yet support Apple HIRES graphics. According to an Epson representative I spoke to, however, they're working on a special PROM to give the MX-80 this capability. The PROM should be out in a quarter or two and is field installable.

PAPER TRANSPORT: The MX-80 supports pin-feed paper only and has a carriage width adjustable from 4 to 10 inches. Both left and right pin tractors are movable. The tractors are driven by a stepper motor so the paper may be advanced in increments as small as 1/72 of an inch (exactly the distance between dots!) under software control. Under different software control, you may select line spacing of 6 or 8 lines per inch. A special forms-thickness control adjusts the distance between the printhead and the paper to accommodate forms up to 3-ply.

INTERFACING: Although the MX-80 is available with serial interfacing, Epson seems to prefer marketing the parallel model. The advantages of parallel over serial interfacing are:

- 1) The software driver is usually shorter so in limited PROM space you may get more control features.
- 2) For the same money, parallel interfaces may be driven much faster than serial.
- 3) They tend to be less expensive. On the other hand, 1) Serial interfacing by and large gives greater interdevice compatibility (i.e., you can drive many different printers with the same serial interface).
- 2) Serial interfacing lets you drive the printer through the game I/O connector, thus saving a card slot (and the cost of a serial card!) and 3) The cable is much easier to make and so standard that it usually comes with the serial card.

So parallel interfacing can be a pain in the neck - but Epson markets its own Apple parallel interface card, complete with cable for \$120. This is the card I bought and it works fine with both Basics, Pascal (no mods to BIOS necessary) and the Z-80 CP/M card - with one exception: the ctl-I feature (set character size) doesn't work. According to Epson, it works okay for them and I may have a bad PROM. But it's such a small problem that I haven't bothered to do anything about it: instead of ctl-I132N, for instance, you can just type ?CHR\$(15) and get exactly the same result.

According to Epson, the parallel interface is 100% Centronics compatible. Either the Apple Centronics parallel interface or the one made by California Computer Systems (Model 7720B - less expensive) should work.

MANUAL CONTROLS AND INDICATORS: Besides the power switch, the MX-80 has manual controls for on/offline, linefeed and formfeed. The formfeed is slow, moving at linefeed speed (although with continuous motion). The printer assumes the paper is at top of form when it is powered on. The lines per page defaults to 66 but may be software set to any lower number. Indicators are pretty standard: POWER-ON, READY, NO PAPER and ONLINE.

SPECIAL FEATURES: The following features are available under software control:

TABBING-Any number of vertical and/or horizontal tabs may be set and executed.

BELL-The ASCII bell character actuates an irritating three-second buzzer, guaranteed to get your attention from the next room. Furthermore, you get a 30-second buzzer when the printer falls into error status - like being out of paper. Fortunately, the buzzer can be deactivated by dipswitch!

OUT-OF-PAPER OVERRIDE-Ordinarily, you can't print when the OUT OF PAPER indicator is on.

EMPHASIZED CHARACTERS-In this mode, each line is printed twice, giving the text a much more professional look. Correspondence printed in this mode looks terrific!

BOLD CHARACTERS-Words or individual letters printed in bold format are printed a second time after the paper has advanced 1/216 inch. I think this is a better technique for emphasis than the "emphasized mode" above.

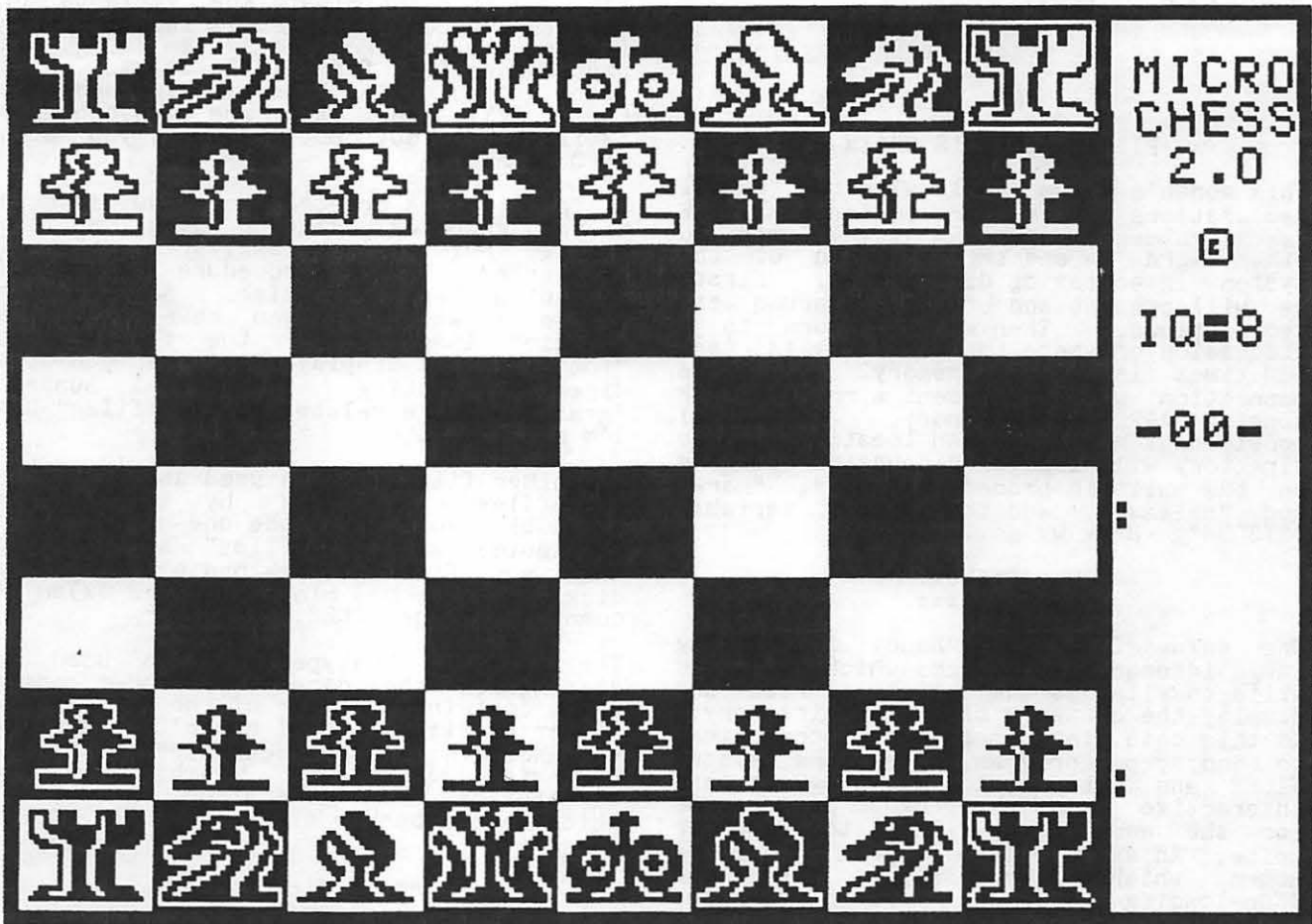
DIMENSIONS-14.7x12.0x4.2 inches

Considering the above features, the print quality and disposable printhead of the MX-80, \$645 seems like a good price. The only two stores in the area that carry the MX-80 that I know of are COMPUTER AGE on Georgia Ave. in Silver Spring and THE PROGRAM STORE on Wisconsin Ave. in North D.C.

COMMENTS ON PRETTYLISTING

by Stephen E. Bach

Thanks to Paul A. Sand and WAP for his Prettylisting program. For those, like me, without Applesoft in ROM, the 53456 in line 63000 should be changed to 2256. Then it works just fine. Let's see this program in assembly language!



'MICROCHESS'

BLAISE AWAY

by Dr. Wo

THE CASE OF THE DISAPPEARING DIRECTORY

-OR-

PICKING APPLE'S BRAIN

This month's column deals with the Pascal declarations for the directories of APPLE Pascal blocked volumes - disk volumes in other words - and the operation of the system in accessing directories. First, we will present and briefly discuss the declarations. Then we will turn to a discussion of where the directory is (and sometimes isn't) in memory. In this connection we will present a routine for dumping APPLE's memory. Additional consideration of the RAM location of the directory will lead to discussion touching on the built-in procedures "new", "mark" and "release", and the global variable "SYSCOM". Here we go...

Why Bother?

=====

One thing I've found handy in writing large interactive programs which read and write to files is the ability to read and display the contents of a disk directory. In this case, large means too inconvenient to end my own program, enter the system FILER and return to my program; and interactive means the program prompts me for the name of the file(s) to read or write. An example is a program to drive a modem which also allows up- and down-loading of files. In this case the last thing one wants is to log off a mainframe just to find out what the name of a forgotten file is.

When I first got started on this problem I tried reading the directory blocks of a disk to figure out what is stored there. I made some headway, with the help of David Neumann of Apple Pi, but quickly got frustrated. Therefore, I gave the Apple Hotline a call. When I came upon a stone wall ("That stuff's confidential") I got irate and wrote to Apple and anybody else I could think of. Finally, a friendly voice from Apple got in touch and informed me of the existence of a Technical Note entitled "APPLE Pascal System Disk Directory Structure." This note is available from Apple or from your club if it is a member of the INTERNATIONAL APPLE CORE. It turns out to be just what the Dr. ordered - nothing less than the full declaration for APPLE Pascal disk directories!

The Declarations

=====

The full declarations are reproduced below. There are several items to note in studying the declarations.

One is the discipline used: Constants and types are given descriptive names (well, sort of anyway) and, with only a few exceptions, the fields in the TYPE "direntry" are declared in terms of

previously declared types. This makes for easy reading, reference and modification.

A second noteworthy feature is the declaration for the TYPE "filekind" which is used to discriminate the variants of the TYPE "direntry". Recall the directory listing you get from the system FILER, in particular the far right column which lists kind of file. The filekind messages CODE, TEXT and DATA are undoubtedly familiar to you. But when have you seen INFO or FOTO?

A fotofile is created when you save a hi-res graphics image to disk, so the message FOTO will be familiar to you if you tried the procedure "savefoto" presented here last time. Somehow the system knows when you have a hi-res graphics image, sets the filekind to "fotofile" and displays FOTO when you do a directory listing. Try it! I suspect "graffiles" are related to "fotofiles" but I'm not positive.

Two other filekinds are used as follows: "infofiles" are used by the system DEBUGGER (you know, the one that isn't available) and "xdskfiles" are created when you find and mark bad blocks on a disk using the bad block scan and x)amine commands in the FILER.

The filekind "untypedfile" is used to distinguish the directory header entry which contains the name of the volume, the number of files etc. I haven't been able to uncover what a "securedir" is. The name is suggestive. Perhaps secure directories are maintained by systems which keep a backup directory. The APPLE doesn't.

Dynamic Directories

=====

If you have observed the behavior of the FILER when listing a directory you know that sometimes the system reads a disk file before doing the listing and sometimes it proceeds directly to the listing. This suggests that a copy of the directory of the most recently accessed disk is maintained in memory. This turns out to be partly true. In fact, space for directories is dynamically allocated, and the system's knowledge of directories is intimately related to the built-in procedures "new", "mark" and "release". This means the address of the system directory can vary, and even that there may not be a directory in memory - but only if the aforementioned procedures are invoked. Assuming we don't invoke them, where do we find the directory?

Memories, Memories

=====

Listed below is a program for passing through APPLE's memory. It has two options: flipping through pages sequentially and viewing a selected page. Each page is displayed as an 8x32 matrix. If a byte holds a printable ASCII code, the corresponding character is displayed; if not, a period is displayed. Refer to the procedure "display" and note the use of the variant record technique discussed here last time. We use it here to point at desired pages of memory.

Type the program in and let me share some memories with you. If you take a look at page 13 (\$0D) you'll see the directory laid out beginning somewhere on line 3. Specifically, you'll see the name of the volume, say APPLE1, from which you called the program "pagedumper"; the string begins at line 3 position 25 which corresponds to memory address 3449. Following this you'll see a series of strings corresponding to the names of files stored on the disk.

Now look at the declarations for a directory entry so we can count bytes and find out exactly where the directory starts. The declaration includes allocations for the fields "difirstblk" and "dlastblk", both of which require 2 bytes. Then comes the tag field "dfkind" and, for the zero-th entry of the directory, "filler1". Because this is a packed record, the combination of the latter two fields requires only 2 bytes. Then comes "dvid", the name of the volume, whose first byte contains the length of the volume name.

Counting back a total of 7 bytes from the address 3449 computed above, we conclude that the directory begins at memory address 3442, which we will refer to as "dramaddress" of the directory.(!)

Out of Sight, Out of Mind
 =====

The drawback to all of this is the intransience of the directory. It's time to think about the global variable SYSCOM. It will help for this discussion if you will open your APPLE Pascal Reference Manual to Section 3.4.2, page 201, which deals with communications between the operating system and the hardware.

SYSCOM links the operating system and the P-machine, and one of the fields in this record is "GDIRP" which is a pointer to the most recent directory read in unless dynamic memory allocation has occurred. Turning to page 209 we see that the P-machine instructions MRK, NEW and RLS (corresponding to the aforementioned built-in Pascal procedures) all cause the system to forget the location of the directory by causing the assignment GDIRP:=nil. A corollary observation is that the address of the directory can vary. If only we could access GDIRP, we could always find the current directory. The key to this is to find out where SYSCOM lives.

According to the BIOS (Basic I-O Subsystem) listing for the APPLE (available from Apple or your club) the RAM location 248 (\$F8) contains a 2 byte pointer to the SYSCOM area. we can easily determine this address, call it "syscombase". Reading the information on page 201 literally, and allocating 2 bytes per field suggests that the RAM address of gdirp is syscombase+8. The programs "findgdirp" listed below verifies these hypotheses. Running the program returns the following values:

```
syscombase= -16930
address of gdirp field= -16922
address of directory= 3442
```

Higher Memories
 =====

You might try exploring other parts of APPLE's memory. There's a lot to see! Try pages 169 and 170 for example.

Page 169, line 0, position 25 contains the system prompt line. Following this is a series of strings which looks very much like a table of devices and volumes on line. Turn to page 260 in your reference manual. Looking at the memory display starting at line 4, position 21, you should see CONSOLE, SYSTEM, a blank space corresponding to the unused device #3, then the names of the volumes in drives 4 and 5 and so on up the line.

Now turn to memory page 170. Starting on line 1, position 6, is a set of strings which looks like a table giving the volumes on which the system programs reside.

Fading Memories
 =====

The point of all this discussion was to learn a little bit about the way the system operates. It's not to say that the best way to access directories from user programs is to look for them in memory. Being about to fade away, I'll leave with the promise of discussing this further and presenting a mini-filer procedure callable from user programs as my next effort.

For now-----

BLAISE AWAY!!!!!!!



Blaise Pascal.

Apple Pascal System Disk Directory Structure

=====

Supplied by Apple Computer, Inc.
Dated 1 June 1980

```
CONST maxdir=77;(maximum number of entries in directory)
      vidleng=7;(number of characters in volume id)
      tidleng=15;(number of characters in title id)
      fblksize=512;(standard disk block length)
      dirblk=2;(directory starts at this disk block address)
```

```
TYPE daterec=PACKED RECORD
      month:0..12;(0 implies meaningless date)
      day:0..31;
      year:0..100;(100 implies dated volume is temporary)
END;
```

```
(volume id)
vid=string[vidleng];
```

```
dirrange=0..maxdir;
```

```
(title id)
tid=string[tidleng];
```

```
filekind=(untypedfile,xskfile,codefile,textfile,infofile,datafile,
          srafile,photofile,securedir);
```

```
(directory layout)
```

```
direntry=PACKED RECORD
      dfirstblk:integer;(first physical disk address)
      dlastblk:integer;(points at block following last used block)
      CASE dfkind:filekind OF
        securedir,untypedfile:
          (only in dir[0], this is volume info)
          (filler1:0..2048;(13 bits for downward compatibility ??)
           dvid:vid;
           deovblk:integer;(number of blocks in this volume)
           dnumfiles:dirrange;(number of files in directory)
           dloadtime:integer;(time of last access ??)
           dlastboot:daterec;(most recent date setting)
        xskfile,codefile,textfile,infofile,
        datafile,srafile,photofile:
          (filler2:0..1024;(12 bits for downward compatibility)
           status:BOOLEAN;(for filer wildcards)
           dtid:tid;(title of file)
           dlastbys:1..fblksize;(numbr of bytes in file's last block)
           daccess:daterec;(date last modified)
      END;
```

```
directory=ARRAY [dirrange] OF direntry;
```

```
--- end of declarations ---
```

contd.

```

PROGRAM pasedumper;
CONST cleol=29;
      cleos=11;
      radix=256;
VAR messaa:STRING;
    ch:CHAR;
    paaenumber:INTEGER;

PROCEDURE prompt;
BEGIN
  unitclear(1);
  gotoxy(0,0);
  write(chr(cleol));
  write('View File, Q)uit?');
END;

PROCEDURE getpaaenum(VAR mess:STRING;VAR paaenum:INTEGER);
BEGIN
  write(chr(cleos));
  write(mess);
  readln(paaenum);
  IF NOT (paaenum IN [0..255]) THEN paaenum:=0;
END;

PROCEDURE display(paaenumber:INTEGER);
CONST iopaae=206;
TYPE buffer=PACKED ARRAY[0..7,0..31] OF CHAR;
      memorypaae=RECORD CASE BOOLEAN OF
        TRUE:(addr:INTEGER);
        FALSE:(contents:buffer);
      END;
VAR mpaae:memorypaae;
    okset:SET OF CHAR;
    baseaddress,linenumber,ccount:INTEGER;
    ch:CHAR;
BEGIN
  okset:=[chr(32)..chr(126)];
  gotoxy(0,5);
  write(chr(cleol));
  writeln('Writing paae number ',paaenumber);
  IF paaenumber<(radix DIV 2) THEN baseaddress:=radix*paaenumber
    ELSE baseaddress:=radix*(paaenumber-radix);
  writeln;
  write(chr(cleol));
  writeln('Base address= ',baseaddress);
  writeln;
  IF paaenumber=iopaae
  THEN writeln('This is the i-o paae. No display.')
  ELSE BEGIN
    writeln('  0   (0-15)   F 0   (16-31)   F');
    writeln('=====');
    mpaae.addr:=baseaddress;
    FOR linenumber:=0 TO 7 DO
      BEGIN
        write(linenumber,' ');
        FOR ccount:=0 TO 15 DO
          BEGIN
            ch:=mpaae.contents[linenumber,ccount];
            IF ch IN okset THEN write(ch)
              ELSE write(' ');
          END;
        writeln;
      END;
  END;

```

contd.

```

    END;
    write(' ');
    FOR chcount:=16 TO 31 DO
    BEGIN
        ch:=mpage.contents[linenumber,chcount];
        IF ch IN okset THEN write(ch)
        ELSE write(' ');
    END;
    writeln;
    END;
END;

```

```

PROCEDURE viewpage;
BEGIN
    write(chr(cleas));
    gotoxy(0,5);
    message:='Enter page number (decimal)..';
    getpagenum(message,pagenumber);
    display(pagenumber);
END;

```

```

PROCEDURE flippages;
CONST esc=27;
      downctrl=8;
      upctrl=21;
VAR lastpage:INTEGER;
    ch,escape,up,down:CHAR;
BEGIN
    escape:=chr(esc);
    up:=chr(upctrl);
    down:=chr(downctrl);
    gotoxy(0,5);
    message:='Enter starting page (decimal)..';
    getpagenum(message,pagenumber);
    gotoxy(0,3);
    write(chr(cleas));
    writeln('Flipping pages');
    display(pagenumber);
    writeln;
    writeln('Press <ESC> to stop flipping,');
    writeln('left arrow to go to low memory,');
    writeln('right arrow to go to high memory. ');
    REPEAT
        read(keyboard,ch);
        IF ch IN [up,down] THEN
            BEGIN
                IF ch=up THEN pagenumber:=(pagenumber+1) MOD radix;
                ELSE IF pagenumber=0 THEN pagenumber:=255;
                ELSE pagenumber:=(pagenumber-1) MOD radix;
                display(pagenumber);
                gotoxy(0,23);
            END;
        UNTIL ch=escape;
    END;

```

```

BEGIN
    message:='';
    ch:=' ';
    pagenumber:=0;

```

```

page(output);
REPEAT
    prompt;
    read(ch);
    CASE ch OF
        'v','V':viewpage;
        'f','F':flippages;
    END;
    UNTIL ch IN ['0','a'];
page(output);
gotoxy(0,5);
writeln('That''s all folks!');
END.

```

```

PROGRAM findsdire;
CONST ptrtosyscom=248;
      offset=8;
VAR syscombase,sdire:INTEGER;

FUNCTION wordpeek(wordaddr:INTEGER):INTEGER;
TYPE wordunion=RECORD CASE BOOLEAN OF
    TRUE:(addr:INTEGER);
    FALSE:(contents:INTEGER);
END;
VAR word:wordunion;
BEGIN
    word.addr:=wordaddr;
    wordpeek:=word.contents;
END;

BEGIN
    page(output);
    gotoxy(0,5);

    syscombase:=wordpeek(ptrtosyscom);
    writeln('The address of the SYSCOM');
    writeln('record is: ',syscombase);
    writeln;

```

```

    writeln('The address of the GDIRP');
    writeln('field of SYSCOM is: ',syscombase+offset);
    writeln;

    sdire:=wordpeek(syscombase+offset);
    writeln('The address of the current');
    writeln('directory is: ',sdire);
END.

```

A PAGE FROM THE STACK

Librarian's Corner

by Dave Morganstein

I am delighted to report that the official Library Committee has doubled in size! With the superhuman help of Bill Bowie (recently interviewed in the Washington Post), we have managed to copy, label, check, address, mail and distribute so many disks that we are only three months behind ... Seriously, Bill's help is making this task "doable".

Donations have also picked up and many of our newer disks were contributed entirely by members. Bill Schultheis, Bruce Field, Michael Thomas, Jim Reilly and Fred Sharp are just a few of the much appreciated donors. Below I'll review some of the newer additions. Keep those disks and listings coming!!

First, let's talk about growing pains. A major problem in my book is disk distribution at the meetings. For one thing, this is often a noisy and distracting process. The meeting and program suffers whenever this distribution takes place. If we wait until after the meeting, the "distributor" has no opportunity to join in on the small group discussions. One solution is to sell ONLY before the meeting, say from 8:45 to 9:30. Several members have volunteered to be available then. Perhaps we can try this.

Regarding sales, let me again urge that you mail in your order so that we can copy what is needed and can assemble a package with your name on it, for pickup at the meeting. Spur of the moment purchases, while lucrative for the club, cause the long lines while folks decide which of the available disks they want. Pickup of assembled packages goes very quickly. I would even suggest a price disincentive of maybe a dollar for purchases at the meeting. (Sounds like Federal economics, huh?)

WHAT'S NEW FOR YOU

Disk 22 - a "must-have" utility disk. Contains a revised Master Catalog providing a three-column list of programs. DANA'S HEX LOADER for easy entry of hex dumps. NIFFUM and FID are 3.2 versions of Apple's utilities from the new DOS 3.3 disks (thank you, Sandy Greenfarb!!!). FID (file developer) permits easy copying of all types of files, cataloging, locking, unlocking, deleting and disk free space. FAST INIT allows initializing of DOS 3.2 disks in record time. PROGRAM DEVELOPMENT PACKAGE is a set of useful Applesoft functions (from the Call -A.P.P.L.E workshop).

Disk 29 - IAC UTILITIES 2. A few goodies from the International, including Sandy's REM STRIPPER, a BOLD PRINT program for the Silentype and a CATALOG MANAGEMENT PROGRAM.

Disk 30 - IAC GAMES 1. Some good hi/lo res games from the Baked Apple Users Group. See a detailed review in last month's newsletter.

Disk 31 - CPLOT from W. Schultheis. A marvelous donation from Bill. Contains both Applesoft and Integer versions. Allows you to plot characters on the Hires pages.

MISC. INFO.: THE VERSAWRITER

I have spent some time recently, playing with this wonderful APPLE peripheral. What you get for the entry fee of \$250 is a "machine" and two disks of software. A goodly part of the price covers the excellent programs provided. Anyone keen on, but not wealthy enough to buy, a graphics tablet will be interested in the Versawriter.

Physically it is a plastic slab with a double hinged arm. Potentiometers, similar to those found in the game paddles, form the hinges. At the end of the arm, a plastic lens is used to center the arm over the desired point. As the arm is moved the two potentiometers are varied. Plugged into the game I/O connector, the Versawriter sends two numbers to the APPLE. These numbers can be translated into X and Y coordinates, after which much magic can be accomplished.

The software supplied lets you do all of the following:

- . Create, load, save, display and modify a Hires page image.
- . Put text on the Hires page.
- . Draw curves and straight lines.
- . Fill in closed surfaces with over 100 (yes 100) different color quilts.
- . Compute area and distance from maps.
- . Define and save shape tables.

If this sounds a bit like an ad for a Vegematic, remember, the Versawriter costs more ...

THE ABBS, FROM ONE USER'S VIEWPOINT

by Genevie Urban

Those of you who have modems and haven't signed on to the Washington Apple Pi ABBS, you don't know what you're missing! Not only is it full of information about the club and what's going on, it's an easy convenient way to reach a lot of people when you need help or a question answered. It also provides a way to get in touch with someone without worrying whether you've interrupted his dinner with a phone call. In addition, there's something very exciting about an electronic bulletin board - the wave of the future, perhaps.

John Moon is doing a fantastic job keeping the bulletin board up-to-date and incorporating changes suggested by all of us.

I, for one, am having a lot of fun using the ABBS. The more people that sign on, the more fun it will be!

FLAVORS: LITTLE TIDBITS

by Burton S. Chambers III

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

LEMON: Bug in Apple Fortran ?

This piece of news falls under the heading of gossip, since I have not tested for the "reported bug" myself. Nevertheless, I am passing the information on because it came from one of the helpful Apple Hot-Line people. I have been told that the current release does not support unformatted sequential files. Hopefully some reader, who needs this capability, will test it and let the rest of us know whether a problem really exists.

GRAPE: Pascal File Names

Legal characters for Apple Pascal filenames are given on Page 13 of the reference manual. At the risk of having future problems, I have been using the character plus sign (+), which seems to work well.

PEAR: Pascal Filer

Did you know that you can give the filer a list of commands in some cases? Get in the filer, type an "E" followed by RETURN (for an extended directory list), then "#4,#1,#4,#1" followed by RETURN. If you have a printer, substitute a "6" for a "1", or if you have two disc files, substitute a "5" for a "4". Apparently, the filer recognizes the comma as a delimiter in the keyboard buffer.

BANANA: Error 407

If you have been using Apple Pascal for large programs, you may have seen this one. It comes after you have compiled your program error-free, and then wipes the code-file. Much distress. The last time I talked to Apple about this, they said it was some bug that was not understood. Coincidentally, Apple Fortran has a problem compiling the text file containing your source code if extra characters exist at the end, however a way around the problem has been suggested, which always seems to work. "Hmm, what if you do the same thing in Pascal?", I thought. So, I have been religiously ending my source file with the standard last statement, "END.", followed by ONE and only ONE RETURN. I don't assert that this works every time, BUT I haven't had an error 407 since I have been following this practice (knock on wood). I would appreciate feedback on this one.

ORANGE: Duplicate Directories in Pascal

For those of you that want to go to the bother of setting up and maintaining back-up directories, you can create the

space for them as follows. Format a BLANK diskette and then set into the Filer and press "Z" to Zero the newly formatted diskette. When asked "DESTROY THAT DISK?", respond with "Y". When asked "DUPLICATE DIRECTORIES?", respond "Y". The other questions should be answered in the normal manner (see reference manual, pg.38). You will have to do the updating and maintainings, since Apple Pascal doesn't support duplicate directories. I don't maintain duplicate directories, instead, the important stuff gets backed-up three levels. This method of making duplicate directories is used by a friend of mine to save himself when his diskettes get wiped out by the system. (P.S. I haven't had that problem.)

LIME: Maintenance of YOUR system

Besides cleaning disc heads (using professional disc cleaners), is there any other maintenance needed for the Apple and its peripherals? Sorry, I don't know, I am just asking the question.


CHERRY: IAC Annotes

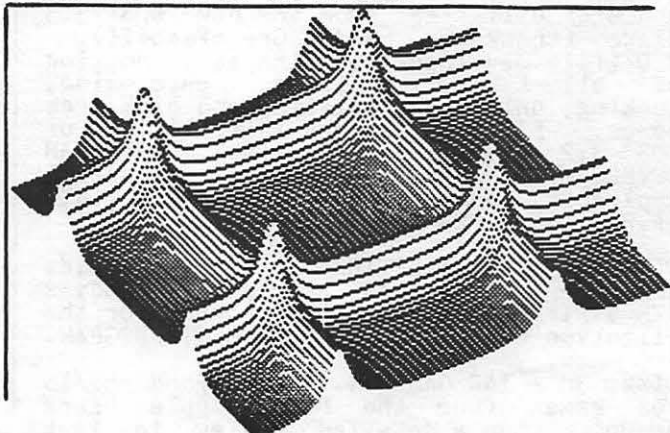
As some of you are aware, Washington Apple Pi receives the International Apple Core (IAC) APNOTES. These are notes provided by Apple computer users, and for the most part are of interest to us all. However, to copy 125 pages or so of goodies for each individual member would be costly. Maybe a better approach is to publish (in the magazine) the table of contents, and let volunteers review individual articles. With the reviews in hand, members could request copies of articles that may interest them for say 10 cents a page to defray copying costs.

PLUM: IAC Note G10 (Pascal BIOS)

I believe Table III is in error (Page G10-4 and G10-5) under the Column "RAM LOCATION". Examples of the correct entries are:

| VECTOR | RAM LOCATION |
|---------------|--------------|
| CONSOLE READ | FF77 |
| CONSOLE WRITE | FF80 |
| CONSOLE INIT | FF89 |

Briefly, you need only subtract 4 from each location in the Annote. 



COMMENTS ON VISICALC

Ed. Note: Walt Francis asked that the following information preface the letter below, which he has shared with the club. For those readers who want more background information on VisiCalc, there is a good nuts and bolts review in the August, 1980 issue of Creative Computing, and a good, short conceptual review in the "Interactive Systems" article in the November, 1980 issue. Walt says thank you to Jim Manley and Chuck Reinbrecht for their helpful advice, and hopes that there is sufficient interest for either a VisiCalc or interactive software special interest group. He recommends purchase of VisiCalc (retail price \$150) by anyone with even a hint of an excuse for analyzing data, systems analysis, table generation, or any application involving simulation and modeling. Although VisiCalc is commonly explained and demonstrated in terms of financial planning, its uses are virtually limitless. It could be used, for example, for regression analysis and other statistical applications with only a few hours of work to lay our format of output and to enter the actual statistical formulas.

PERSONAL SOFTWARE INC.
592 Weddell Drive
Sunnyvale, California 94086

Dear Sirs:

As an extremely satisfied VisiCalc user, I am writing you to make several suggestions for future improvements in VisiCalc. My particular application involves use of VisiCalc in the preparation of a book on health insurance plans for Federal employees (see enclosure 1). In writing this book, I had to perform a series of detailed actuarial calculations on each of 124 different insurance plans, calculating for each plan expectation of each of seven types of cost for each of seven levels of expense for each of five family sizes. For each plan, there are approximately one dozen variables (premium, dues, deductibles, coinsurance rate by type of expense, etc.) which generally differ among plans. My approach to this problem was to set it up as a simulation problem, in which each plan represented a different run of the same basic series of equations (see enclosure 2 for a sample run from the system). The simulation for each health plan requires about 450 VisiCalc statements and occupies about 10K of RAM. The simulation runs on all 124 health plans therefore involve over 1.2 megabytes of disk storage on my Apple, though this could have been reduced to about 1 megabyte had I been less generous in my use of blank spaces for readability.

After these calculations were performed for each plan, the results were then entered on a separate table (also using VisiCalc), ranking plans by average cost for each family size. These are the cost tables published in the book in a 10 page segment, starting on page 20. Several other tables in the book were also prepared using VisiCalc, in effect as a word processor for tables.

As a result of my experience, in which I was able to successfully handle five times as many plans as I had done by hand calculator in the previous year's edition (which only covered plans available in the D.C. area), in less total time, I have become a fanatic admirer of VisiCalc, which is a brilliant piece of software. My only alternative would have been to prepare a program from scratch, a job far beyond my neophyte computer skills and which would have cost thousands of dollars at commercial rates. Moreover, some of the calculations for particular plans could not use my standard simulation, and my format had to be modified individually for these plans. As a result, I am not sure that my system could have been created from scratch by any programmer without being so large as to exceed the capacity of either the Apple or any conceivable budget.

It is in the light of this success that I offer the following suggestions for improvements.

First, VisiCalc needs a hard-copy LIST capability. While most uses would not involve the sheer size and complexity of mine, I am surely not alone in using VisiCalc for substantial number crunching. Many of my 450 statements are algebraically complex. As VisiCalc is now configured, checking the accuracy of each formula was tedious in the extreme. Had I been able to print out the statement used by each cell, I would have saved hours of work by using hard copy rather than the TV screen. I recently discovered that list programs can be purchased from other vendors, and I bought and used successfully Computer Station's Visilist. Why on earth shouldn't VisiCalc have come with this capability built in, if necessary on a second disk?

Second, VisiCalc could use an upper/lower case capability. As I am sure you have heard many times, VisiCalc is virtually a word processor. Unfortunately, when a table produced by VisiCalc cannot be used directly without first retyping it, because the table requires distinguishing upper and lower case characters, much of the potential utility of the system is lost. In my case, you will notice that many of the plan titles used in the book require just this distinction. I appreciate that upper/lower case is an expensive and hardware intensive problem, but surely there is a way you can make VisiCalc use this capability where the computer already has it (either for the Apple III or for the Dan Paymar chip, for example).

Third, there are few things more tedious than sorting by hand (see the cost tables for the problem I faced, and imagine trying to shuffle 30-odd pieces of paper around 10 times, and then type each table from scratch). Since each row heading in each of my tables is the same, the approach I used was to type the first table, save it, load it, change all the numbers in each row, and put the rows in the proper ranking order by Replicate/Delete or (a belated discovery) Move. This gave me the second table, and repeating the process gave me each succeeding table. Surely an automatic SORT function, using size of each result in a given column, would be at least as valuable to many users as some of the more exotic VisiCalc functions.

Fourth, a system by which selected portions of several individual files (in my case, 124 files) could be saved into a new file would be a great convenience when results of one file are needed to create a new file. My cost tables had to be created from scratch, even though each piece of data on them, including plan name and other identifiers, existed already in a VisiCalc file. The idea is to be able to select particular cells, store them in a separate memory location, then clear the screen and load a new file, store the same selected items along with the first batch, etc. I recently discovered that it is possible to do something like this using the CCA Data Management System you sell. However, the CCA system is ludicrously inadequate for this (it is probably wonderful for mailing lists and other uses unrelated to VisiCalc), since it requires a series of steps far more time consuming than typing results into a new file--first reformatting the VisiCalc sheet to remove the blank lines which make it readable, then converting the VisiCalc file to a DOS file, then converting the DOS file to the first type of CCA file (an unbelievably slow step), then converting to the second type of CCA file, etc. Frankly, I don't know why you even bothered to give CCA a VisiCalc interface capability so unwieldy. In addition, CCA

contd.

itself is extremely difficult to master, as I am sure you have heard many times--so difficult that I am not sure after several hours of work whether it is even possible to do what I need using CCA. Why don't you simply create a program tailored to VisiCalc, and as user friendly, to add this capability?

Fifth, your manual needs several easy to add features to improve its usefulness. It is, as you know, a superb manual--clear, easy to follow, etc. Nonetheless, the lack of a real index is sometimes aggravating. Related to the lack of an index is the need for a section describing briefly the rules and purpose of each function, and where to find it discussed in the text. Your reference card is close to this, but not enough. Thirdly, a section describing how VisiCalc functions can be creatively combined to achieve particular purposes, the shortcuts developed by experienced users, etc., would be most useful. For example, even as VisiCalc stands now, it is possible to merge a small number of files by clever formatting and Saving one file on top of another. Finally, while I can understand why you might not want to plug competitor's products, surely it is not too much to ask that your manual at least mention that products such as Visilist and CCA exist, and preferably describe their usefulness.

I suspect that several of these ideas are impractical given the RAM limitations of micro-computers, but even two or three of them would add greatly to the already superb abilities of VisiCalc.

I look forward to hearing your reactions or, better yet, receiving a VisiCalc update in time for next year's edition of my book.

Sincerely yours,

Milton Francis

Washington Business

The following is reprinted from the
December 1, 1980 Washington Post.

Contest Seeks Computer Aid for the Handicapped

By Scott Chase
Washington Post Staff Writer

Baltimore's Johns Hopkins University, with grant aid from the National Science Foundation and the Radio Shack division of Tandy Corp., last week announced a nationwide contest search to inspire new inventions that apply personal computers to the needs of handicapped people.

The grand prize in the competition is \$10,000. Last day for submission of entries is June 30, 1981. The ultimate objective of the contest is new inventions or concepts that handicapped people can use to improve their quality of life.

Paul L. Hazan, director of the university's Personal Computing to Aid the Handicapped project, said the contest is a challenge to people to use their "conceptual skills" in creating practical aids based on

current computer technology that will help an individual or group with a handicap. All categories of handicaps, including mental disorders, can be addressed by potential inventors.

"Just as important will be the opportunity provided the inventors and developers to make contact and form partnerships with the handicapped in a way that can lead to wide acceptance and the use of the new computing technology," Hazan said.

Orientation meetings are being scheduled at major rehabilitation centers throughout the country in an effort to bring together contest entrants, handicapped individuals and professional rehabilitation workers. Special presentations are planned at chapter meetings of the Association for Computing Machinery, the Institute of Electrical and

Electronics Engineers, and private personal computer clubs.

Individual contest participants will be provided with an inventor kit that includes official entry forms for submission and evaluation of their computer systems or concepts. Judging will be done on three levels of competition—computer professional, amateur and full-time student.

In addition to the \$10,000 grand prize, 15 personal computer systems, several \$1,000 awards and many additional prizes will be awarded. Winners will be invited to Washington to participate in awards ceremonies and receive their prizes. All invention rights will remain with the contestant.

For more information on the contest, contact Personal Computing to Aid the Handicapped, The Johns Hopkins University, P.O. Box 670, Laurel, Md. 20810.

LINEAR SCALING IN HI-RES GRAPHICS

by Bill Wurzel

The distinction between logical and physical is frequently encountered in data processing. Roughly speaking, physical refers to hardware and logical refers to software. For example, a program may work with records that are 100 bytes long. As far as the program (and frequently the programmer) is concerned, the records are 100 bytes long. The DOS, however, may store this record as 256-byte physical records. Somewhere, somehow some piece of code is going to have to translate logical quantities into physical quantities.

As a second example, consider the "carriage return" character. A "logical carriage return" is some character or group of characters which actually make the carriage return. Somewhere, in some piece of code, the logical carriage return must be translated into a physical carriage return. Logical quantities must usually be translated into physical quantities. Sometimes this translation is done by system software (as in the first example above - and the user may not even know that this particular logical/physical distinction exists!) and sometimes it must be done by the user's program (as may be the case in the second example).

A programmer who wants to use APPLE Hires graphics to do any kind of graphing discovers that he has a logical to physical translation problem facing him right off the bat: he must translate the logical units of the quantities to be graphed into the physical units of Hires graphics. The physical graphics units range from 0 to 279 in the horizontal direction (x-axis) and from 0 to 191 in the vertical direction (y-axis). The point (0,0) appears in the left upper corner of the screen. X-values increase to the right and y-values increase down the screen. Thus, for example, the coordinates (120,86), (160,86), (120,106), (160,106) define - in physical units - the four corners of a rectangle in the middle of the screen.

Now let's say I want to draw a connected line graph of my car's gas mileage by months for the past year. Then the x-variable will be month of the year and will take on whole number values from 1 to 12 inclusive. The y-variable will be gas mileage expressed in miles per gallon of gas and will take on values between (say, for instance) 18 and 23. Furthermore, let's say I want the x-axis to go from physical position 20 to 260 and the y-axis to go from 150 to 30. This gives us the following physical and logical intervals:

| PHYSICAL | LOGICAL |
|-------------|------------|
| x: (20,260) | x: (1,12) |
| y: (150,30) | y: (18,23) |

Now let's consider the x-axis first. As x assumes the logical values 1 through 12, we want the x-coordinate of the Hires cursor to assume physical values of 20 to 260 in such a way that ratios are

preserved. In other words, when x (the month) is one third the way between 1 and 12, we want the cursor to be one third the way between 20 and 260 in the horizontal direction.

Algebraically, then, what we want is the equation of the straight line connecting the points (1,20) and (12,260). Reaching way back into high school algebra, we remember with some pain that this equation can be written:

$$XP = (260-20) / (12-1) * (XL-1) + 20$$

where XL is the x value in the logical scale (i.e. month of the year) and XP is the x value of the cursor. Just as a check, we can see by inspection that XP = 20 when XL = 1 and that XP = 260 when XL = 12.

By exactly the same sort of thinking, we derive the equation for transforming logical y (YL, our gas mileage) into physical y (YP, the y-coordinate of the cursor):

$$YP = (30-150) / (23-18) * (YL-18) + 150$$

These functions can easily be implemented in Applesoft using the one-line "DEF FN" statement. If we define the logical x-interval as (XL,XH) and the physical x-interval as (XA,XB), and similarly for the y's, then we can write the following two Applesoft function statements:

```
DEF FN (X) = (XB-XA)/(XH-XL)*(X-XL)+XA
DEF FNA (Y) = (YB-YA)/(YH-YL)*(Y-YL)+YA
```

Then, with all the above variables properly initialized, we can plot a point on the x-y plane corresponding to (month #3, 20 mpg) by writing:

```
H PLOT FN A(3), FN B(20)
```

It is now very easy to change our logical and/or physical plotting intervals. If, for example, we want to graph gas mileages from 15 to 25 mpg, we just change the values of variables YL and YH and all the appropriate scaling is taken care of. Similarly, if we want to change the y-axis on the screen to go from 130 to 10, we just change the values of YA and YB. Or, it might just happen that we want to enter many different sets of gas mileages one after the other and graph each one. In this case, we could let the program find the minimum and maximum value of each data set and assign these values to YL and YH. Still further, we could graph four different sets of gas mileage data in each quadrant of the screen by merely changing the physical interval variables accordingly. The above schema thus allows the computer to make the translation from logical to physical graph space, enabling us to concentrate on WHAT we want to graph rather than HOW we are going to graph it!

The following program illustrates what we've talked about. It draws the graph discussed above, makes the appropriate tic marks and then graphs twelve "gas mileages." If you fool around with the values of XA, XB, YA and YB you can move the graph all over the screen!

LIST

```
10 DIM M(12)
20 XA = 20:XB = 260:XL = 1:XH = 1
  2
30 YA = 150:YB = 30:YL = 18:YH =
  23
40 DEF FN A(X) = (XB - XA) / (X
  H - XL) * (X - XL) + XA
50 DEF FN B(Y) = (YB - YA) / (Y
  H - YL) * (Y - YL) + YA
60 FOR I = 1 TO 12: READ M(I): NEXT

70 HGR2
80 HCOLOR= 3
90 HPLOT FN A(1), FN B(23)
100 HPLOT TO FN A(1), FN B(18)
  TO FN A(12), FN B(18)
110 GOSUB 500
120 HPLOT FN A(1), FN B(M(1))
130 FOR I = 2 TO 12
140 HPLOT TO FN A(I), FN B(M(I)
  )
150 NEXT
160 END
500 REM DRAW THE TIC MARKS
510 FOR I = XL TO XH
520 HPLOT FN A(I), FN B(18) TO
  FN A(I), FN B(18) - 3
530 NEXT I
540 FOR J = YL TO YH
550 HPLOT FN A(1), FN B(J) TO FN
  A(1) + 3, FN B(J)
560 NEXT J
570 RETURN
999 DATA 19,20.0,20,21,19,20,5,
  22,22.5,20,21,22,20
```

J

Ⓞ

A BUG IN THE TOOLKIT ASSEMBLER

by Bill Wurzel

The 6502 Assembler that comes with the Applesoft Toolkit is a powerful editor/assembler capable of assembling large programs. Unfortunately, it incorrectly assembles instructions using the indexed indirect addressing mode when the associated zero-page address is a literal: 1) the opcode is incorrect; 2) a three-byte zero-page instruction is assembled; 3) the assembler's internal instruction counter is bumped by a hexadecimal 60. For example, you get the following code

```
0100 9D 15 00 STA ($15,X)
0162 ...
```

instead of the correct

```
0100 81 15 STA ($15,X)
0102 ...
```

Needless to say, this can cause you a boatload of grief! The easiest way to get around the problem is to equate the zero-page address to a symbol -- then everything works fine.

I phoned the Hotline people to ask about a fix -- sadly, they were not aware of the bug. So, as soon as I can locate one of their special "bug reporting forms" I'll report the bug. In the meantime, beware. Ⓞ

DANA'S HEX LOADER

by Dana J. Schwartz

Most assembler routines are published today with a hexadecimal dump which can be directly POKE'd into memory. This routine was my attempt to simplify the entering of hex code and data, and to ease the detection and correction of the errors which are bound to occur.

When the program is run, it will list the addresses of memory which it is using. If the regions to be loaded overlap these portions, HIMEM: and/or LOMEM: should be altered to move the program to another area of memory.

After a starting address is entered, the user merely has to type the hex characters to be stored, with no need for intervening spaces or returns. Each POKE is performed immediately after the value is typed. If an error is made, simply move the cursor through the memory display using the left and right arrows, and then reenter the data at the appropriate spot. Hitting the space bar will duplicate the data from the previous location, thus allowing the rapid entering of a series of identical bytes. The program is terminated by a Return, which also generates a composite dump of the memory just altered.

Note the use of the "Scroll Down" routine (WAP September 1980) in lines 2140-2160.

>LIST

```
10 DIM H$(16),B$(4),C$(2)
20 H$="0123456789ABCDEF"
30 G$="": REM CTRL-G
40 UP= PEEK (205)+2: IF PEEK (
  203)<UP+4 THEN UP=-1
50 CALL -936
60 VTAB 8: TAB 10: PRINT "DANA'S HE
  X LOADER": VTAB 12
70 PRINT " RIGHT =FORWARD"
80 PRINT " LEFT =BACK"
90 PRINT " SPACE =REPEAT"
100 PRINT " RETURN=TERMINATE/DUMP"

110 C1= PEEK (75)/16+1:C2= PEEK
  (75) MOD 16+1:C3= PEEK (74)
  /16+1:C4= PEEK (74) MOD 16+
  1
```

contd.

```

120 VTAB 18: PRINT "STORAGE $";
    H$(C1,C1);H$(C2,C2);H$(C3,C3)
    ;H$(C4,C4);"-";
130 IF UP#-1 THEN 150
140 C1= PEEK (205)/16+1:C2= PEEK
    (205) MOD 16+1: PRINT H$(C1,
    C1);H$(C2,C2);"FF (NO SCROLL DO
    WN)": GOTO 160
150 C1=(UP+3)/16+1:C2=(UP+3) MOD
    16: PRINT H$(C1,C1);H$(C2,C2)
    ;"CF"
160 C1= PEEK (203)/16+1:C2= PEEK
    (203) MOD 16+1:C3= PEEK (202
    )/16+1:C4= PEEK (202) MOD 16
    +1
170 PRINT "PROGRAM $";H$(C1,C1)
    ;H$(C2,C2);H$(C3,C3);H$(C4,
    C4);"-";
180 C1= PEEK (77)/16+1:C2= PEEK
    (77) MOD 16+1:C3= PEEK (76)
    /16+1:C4= PEEK (76) MOD 16+
    1
190 PRINT H$(C1,C1);H$(C2,C2);H$
    (C3,C3);H$(C4,C4)
200 VTAB 22: INPUT "START(HEX)?"
    ,B$
210 IF LEN(B$)=0 THEN 50
220 BASE=0:EXT=0
230 FOR I=1 TO LEN(B$)
240 FOR HEX=1 TO 16
250 IF H$(HEX,HEX)=B$(I,I) THEN
    280
260 NEXT HEX
270 PRINT G$;: GOTO 50
280 IF LEN(B$)#4 OR I#1 OR HEX<
    9 THEN 300
290 HEX=HEX-8:EXT=1
300 BASE=BASE*16+(HEX-1)
310 NEXT I
320 STRTL=BASE MOD 256:STRTH=BASE/
    256+EXT*128
330 CALL -936
500 C1=BASE/4096+1+EXT*8:C2=(BASE MOD
    4096)/256+1:C3=(BASE MOD 256)/16+
    1:C4=(BASE MOD 16)+1
510 TAB 1: PRINT H$(C1,C1);H$(C2,
    C2);H$(C3,C3);H$(C4,C4);": "
    ;: CALL -868
520 HERE=BASE-32767*EXT-EXT
530 C$="": GOSUB 1000
540 IF LEN(C$)#2 THEN 800
550 CH=0
560 FOR I=1 TO 2
570 FOR HEX=1 TO 16
580 IF H$(HEX,HEX)=C$(I,I) THEN
    600: REM C$ ALWAYS VALID
590 NEXT HEX
600 CH=CH*16+HEX-1
610 NEXT I
620 POKE HERE,CH: REM <-THIS IS IT!
630 IF BASE#32767 THEN 650
640 BASE=-1:EXT=1
650 BASE=BASE+1: GOTO 500
800 TAB 1: CALL -958
810 BASE=BASE-1
820 IF BASE>=0 THEN 860
830 IF EXT=1 THEN 850
840 BASE=0: GOTO 860
850 BASE=32767:EXT=0
860 POKE 60,STRTL: POKE 61,STRTH:
    POKE 62,BASE MOD 256: POKE
    63,BASE/256+EXT*128
870 CALL -589: PRINT
880 END
1000 KB= PEEK (-16384): IF KB<127
    THEN 1000: POKE -16368,0
1010 IF KB=136 THEN 2000: REM LEFT
    ARROW
1020 IF KB=141 THEN RETURN : REM C/R
1030 IF KB=149 THEN 3000: REM RIGHT
    ARROW
1040 IF KB=160 AND C$="" THEN 4000
    : REM SPACE
1050 KB=KB- ASC("0")
1060 IF ((KB>=0 AND KB<=9) OR (KB>=
    17 AND KB<=22)) AND (EXT=0 OR
    BASE<16384) THEN 1080
1070 PRINT G$;: GOTO 1000
1080 IF KB>9 THEN KB=KB-7: REM A-F
1090 C$( LEN(C$)+1)=H$(KB+1,KB+1
    )
1100 TAB 6: PRINT C$;
1110 IF LEN(C$)<2 THEN 1000
1120 PRINT
1130 RETURN
2000 IF C$="" THEN 2020
2010 TAB 6: PRINT " ";C$="": GOTO
    1000
2020 BASE=BASE-1
2030 IF BASE>=0 THEN 2070
2040 IF EXT#0 THEN 2060
2050 BASE=0: PRINT G$;:C$="": GOTO
    1000
2060 BASE=32767:EXT=0
2070 VAL= PEEK (HERE)
2080 C1=VAL/16+1:C2=VAL MOD 16+1
2090 TAB 6: PRINT H$(C1,C1);H$(C2,
    C2);
2100 IF PEEK (37)=0 THEN 2120
2110 VTAB ( PEEK (37)): POP : GOTO
    500
2120 IF UP#-1 THEN 2140
2130 CALL -936: POP : GOTO 500
2140 POKE 60,0: POKE 61,4: POKE
    62,174: POKE 63,7: POKE 66,
    0: POKE 67,UP: CALL -468
2150 POKE 60,0: POKE 61,UP: POKE
    62,86: POKE 63,UP+3: POKE 66
    ,128: POKE 67,4: CALL -468

```

contd.

```

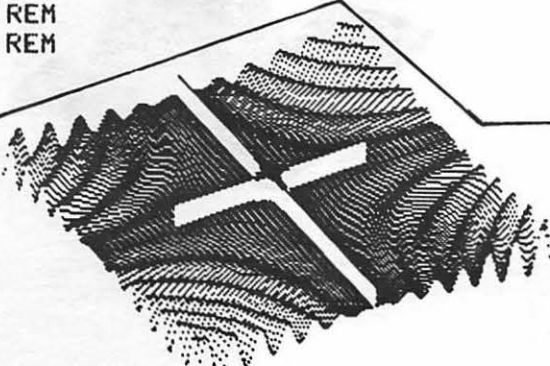
2160 POKE 60,128: POKE 61,UP+3: POKE
      62,174: POKE 63,UP+3: POKE
      66,40: POKE 67,4: CALL -468
2170 POP : GOTO 500
3000 IF BASE#32767 THEN 3040
3010 IF EXT=0 THEN 3030
3020 PRINT G$: POP : GOTO 500
3030 BASE=-1:EXT=1
3040 VAL= PEEK (HERE)
3050 C1=VAL/16+1:C2=VAL MOD 16+1

3060 TAB 6: PRINT H$(C1,C1);H$(C2,
      C2)
3070 BASE=BASE+1
3080 POP : GOTO 500
4000 IF EXT=0 THEN 4040
4010 IF BASE>16383 THEN 4050
4020 IF HERE>=-32767 THEN 4060
4030 CR= PEEK (32767): GOTO 4070

4040 IF BASE#0 THEN 4060
4050 PRINT G$: POP : GOTO 500
4060 CR= PEEK (HERE-1)
4070 C1=CR/16+1:C2=CR MOD 16+1
4080 C$=H$(C1,C1):C$(2)=H$(C2,C2)

4090 PRINT C$
4100 RETURN
10000 REM *****
10010 REM *
10020 REM *      DANA'S HEX LOADER *
10030 REM *
10040 REM *      BY DANA J. SCHWARTZ *
10050 REM *
10060 REM *      7/25/80 *
10070 REM *
10080 REM *****
10090 REM
10100 REM 0-999 MAIN ROUTINE:
10110 REM      0-499 GET BASE
10120 REM      500-799 MAIN POKE LOOP
10130 REM      800-999 FINAL DUMP
10140 REM
10150 REM 1000-4999 KEYIN SUBR:
10160 REM      1000-1999 GET HEX CHAR
10170 REM      2000-2999 BACKUP
10180 REM      3000-3999 FORWARD
10190 REM      4000-4999 REPEAT
10200 REM
10210 REM

```



THE "OLD RUFFLED CROSS"

CHANGES FOR DOS 3.3

by Dana J. Schwartz

FILEMOVER: (3.2 OR 3.3 compatability)

Replace the following lines:

```

2250 IF TR>2 AND TR<35 AND TR<>17
      AND SE<MAXS THEN 2270
6000 TR=17:SE=0: GOSUB 5000:TR= PEEK
      (2305):SE= PEEK (2306):MAXS=
      PEEK (2357): GOSUB 5000: REM DI
      RECTORY

```

SAVE TAPE: (3.3 version only)

Routine SVTP:

Delete lines 190-270.

Add the following lines:

```

190 FOR SE=15 TO 8 STEP -1
195 IF B1>128 THEN 220: REM NOT USED

200 GOSUB 2000: REM READ SECTOR
205 BUF=BUF+1: IF BUF<TOP THEN
      220
210 GOSUB 1000: REM WRITE TAPE
215 BUF=18
220 B1=(B1*2) MOD 256
225 NEXT SE
230 FOR SE=7 TO 0 STEP -1
235 IF B2>128 THEN 260: REM NOT USED

240 GOSUB 2000: REM READ SECTOR
245 BUF=BUF+1: IF BUF<TOP THEN
      220
250 GOSUB 1000: REM WRITE TAPE
255 BUF=18
260 B2=(B2*2) MOD 256
270 NEXT SE

```

Routine RESTP:

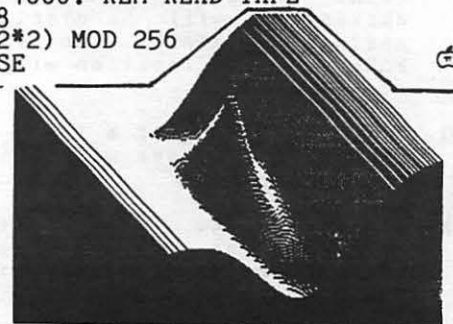
Delete lines 160-250.

Add the following lines:

```

160 FOR SE=15 TO 8 STEP -1
165 IF B1>128 THEN 190: REM NOT SAVE
      D
170 GOSUB 2000: REM WRITE SECTOR
175 BUF=BUF+1: IF BUF<TOP THEN
      190
180 GOSUB 1000: REM READ TAPE
185 BUF=18
190 B1=(B1*2) MOD 256
195 NEXT SE
200 FOR SE=7 TO 0 STEP -1
205 IF B2>128 THEN 240: REM NOT SAVE
      D
210 GOSUB 2000: REM WRITE SECTOR
215 BUF=BUF+1: IF BUF<TOP THEN
      240
220 GOSUB 1000: REM READ TAPE
230 BUF=18
240 B2=(B2*2) MOD 256
250 NEXT SE

```



'NORMAL DISTRIBUTION CURVES'

QUESTIONS, QUESTIONS, QUESTIONS

edited by Mark L. Crosby

their normal version to include software that will display lower case. Should be available during the 1st Quarter of 1981.

Q. I'd like to be able to turn on the TRACE command and still be able to use the disk system for I/O. Is that possible?

A. Yes it is. Instead of using only a CONTROL-D for disk commands add a carriage return before it in this manner: (This is for Applesoft)

```
10 D$ = CHR$(13) + CHR$(4)
```

Because TRACE printing can leave you anywhere on the screen the carriage return will permit all disk commands to be executed beginning in column #1. Therefore they will work properly.

Q. Is it possible to change a disk volume # without re-initting the disk? I have the Disk Aide program and changed byte 6 in track \$11 sector \$0 and it reads back OK, but it still has the old volume # when you boot it!

A. You will need to use D.O.S. 3.1 for Disk Aide to work properly. When it asks for track and sector # you must enter the decimal equivalent (\$11 is 17) not the hex value.

Q. Does anyone know of a reference source listing the PEEK and POKE commands that are available for the apple II.

A. One of the more complete lists was published in WAP Vol 1 No. 3 April 1979 pages 7-8. for Applesoft, the 1st issue of the APPLE ORCHARD has a very complete list of internal entry points. (If someone knows of a more complete list please let us know!).

Q. I am looking for a supplier of Floppy Armour - any ideas?

A. Couldn't find them locally. Guess you'll have to mail order from: SQUARE ONE, 614 - EIGHTEENTH AVENUE, MENLO PARK, CA 94025. (Floppy Armour is their trademark).

Q. I have a file on my disk with a control character in the name. I don't know what it is. How can I find out?

A. If you use DOS 3.2, POKE 251 into the location specified: 10647 (16K), 27031 (32K) or -22121 (48K). Now all control characters will be displayed when you perform a CATALOG. To return to normal POKE the same location with 253.

Q. I am looking for a lower case display generator for use with EasyWriter - any suggestions?

A. According to Information Unlimited Software, Inc. their professional version (for 80-column video cards) supports lower case display. They are rewriting

Q. I want to know if the Low-Res graphics screen can be cleared to a color?

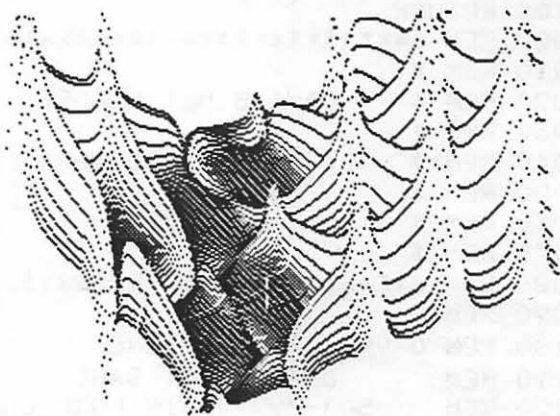
A. Here is a short program that will do just that:

FOR INTEGER BASIC:

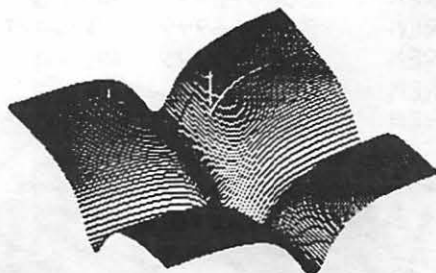
```
10 GR
20 CALL -936: INPUT "COLOR VALUE FOR SCREEN",C
30 POKE 1024,C + C * 16
40 POKE 60,0: POKE 61,4: POKE 62,254
   POKE 63,7
50 POKE 66,1: POKE 67,4: CALL -468
60 GOTO 20
```

FOR APPLESOFT:

```
5 POKE 768,216: POKE 769,160: POKE
770,0: POKE 771,76: POKE 772,44: POKE
773,254
10 GR
20 CALL -936: INPUT "COLOR VALUE FOR SCREEN ?";C
30 POKE 1024,C + C * 16
40 POKE 60,0: POKE 61,4: POKE 62,254:
   POKE 63,7
50 POKE 66,1: POKE 67,4: CALL 768
60 GOTO 20
```



FOREST



TULIP

PRINTING OUT THE HI-RES SCREEN

by Howie Mitchell

THE HIGH-RESOLUTION SCREEN IS AFFECTED BY QUANTITIES STORED IN A SERIES OF MEMORY LOCATIONS (#8192 THROUGH #16384, FOR 'PAGE 1'). THE FIRST 40 LOCATIONS (#8192 THROUGH #8231) INFLUENCE THE TOP LINE, WITH EACH LOCATION CONTROLLING 7 POINTS ON THE SCREEN.

THE SCREEN MEMORIES TURN ON POINTS ON THE SCREEN IN WHAT MIGHT BE CALLED A 'BACKWARDS-BINARY-BIT' MANNER.

FOR EXAMPLE:

| NO. STORED | BAC'RDS BIN. | SCREEN EFFECT |
|------------|--------------|---------------|
| 1 | 1000000 | . |
| 2 | 0100000 | . |
| 3 | 1100000 | .. |
| 4 | 0010000 | . |
| 5 | 1010000 | .. |
| 6 | 0110000 | .. |
| 7 | 1110000 | ... |
| 19 | 1100100 | .. . |
| 127 | 1111111 | |

WHEN WE GET THROUGH WITH MEMORY LOCATIONS AFFECTING THE TOP LINE, WE ARE IN FOR A SURPRISE, BECAUSE LOCATION #8232 CONTROLS THE SCREEN ROUGHLY MID-WAY DOWN. MEMORIES #8232 THROUGH #8271 HANDLE THAT (APPROX.) MIDDLE LINE, AND THEN MEMORY LOCATIONS #8272 - #8311 AFFECT A LINE TOWARDS THE BOTTOM OF THE SCREEN!

MEMORY LOCATION #8320 OFFERS A FURTHER SURPRISE, BECAUSE IT BEGINS A LINE ON THE SCREEN WHICH IS 8 LINES BELOW THE VERY TOP ONE. WHEN THAT LINE IS ENDED, WE JUMP DOWN TO 8 LINES BELOW THE PREVIOUSLY MENTIONED 'MIDDLE LINE', AND WITH THAT ONE COMPLETED, WE THEN SEE THE SCREEN AFFECTED 8 LINES BELOW THE ONE NEAR THE BOTTOM OF THE SCREEN.

THIS PATTERN CONTINUES UNTIL WE GET TO A MEMORY LOCATION WHICH IS 1024 GREATER THAN THE (INITIAL) LOCATION #8192.

HERE, AT LOCATION #9216 (8192+1024), WE SEE THE SCREEN AFFECTED JUST ONE LINE DOWN FROM THE TOP LINE!

FROM HERE ON, THE BASIC PATTERN REPEATS, WITH THE SMALLER INCREMENTS IN MEMORY LOCATIONS RESULTING IN GREATER 'JUMPS' BETWEEN SCREEN AREAS AFFECTED.

PROGRAM TO FILL IN HIRES SCREEN, LINE-BY-LINE:

```

40 FOR LEVEL = 8192 TO 8192 + 80
   STEP 40
45 FOR SUB = 0 TO 7 * 128 STEP 1
   28
50 FOR SSUB = 0 TO 7 * 1024 STEP
   1024
55 FOR INC = 0 TO 39
60 POKE LEVEL + SUB + SSUB + INC
   ,127
65 NEXT INC,SSUB,SUB,LEVEL
70 REM
80: 'LEVEL' CONTROLS THE
   LARGEST INCREMENT (C 1/3 OF
   THE SCREEN), SUB MAKES THE 8
   LINE JUMPS, AND SSUB THE 1-
   LINE JUMPS. 'INC' SCANS A-
   CROSS A SINGLE LINE. THE RE-
   SULT IS THAT THE SCREEN IS
   FILLED IN LINE-BY-LINE!!

```

* * *

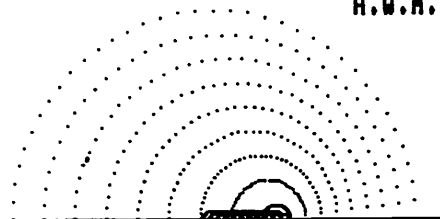
THIS DISCUSSION IS OFFERED AS A CLARIFYING STATEMENT FOR THE HIRES PRINTOUT PROGRAM (FOR THE ANADIX DP-9501).

HAVING ACQUIRED ONE OF THESE ELEGANT PRINTERS, I WAS AT A LOSS AS TO HOW TO USE THE PAPER-TIGER MACHINE-LANGUAGE PROGRAM, AND WAS ALSO INSPIRED TO TRY MY HAND AT WRITING A PROGRAM FOR THE SAME PURPOSE.

WHEN IN THE GRAPHICS MODE, THIS PRINTER PUTS OUT BINARY-PATTERNED DOTS FOR CHARACTERS IN THE RANGE OF ASCII 64 THROUGH 127. THE LETTER 'A' IS EQUIVALENT TO BINARY 1 (ONE DOT), AND '@' IS LIKE BINARY 0 (A TINY SPACE).

THE PROGRAM PEEKS AT THE HIRES SCREEN MEMORY LOCATIONS, AND TRANSLATES EACH AMOUNT INTO A STRING OF A'S AND @'S, SO THAT THE RESULTING DOT PATTERNS MATCH THE POINTS LIGHTED ON THE SCREEN.

H.W.M.



'DOPPLER EFFECT'

contd.

THIS IS AN FP BASIC PROGRAM WHICH WORKS QUITE WELL FOR THE ANADEX DP-9501 PRINTER, FOR PRINTING OUT THE HIRES SCREEN:

```

10 LOMEM: 4 * 16 ^ 3: REM
    SET LOMEM TO $4000 TO
    PROTECT HIRES PAGE 1.

15 DIM SCAN$(128)
20 TEXT : HOME
30 PRINT "*** ANADEX GRAPHICS PL
    OTTER ***          -----
    -----": PRINT
40 PRINT " I CAN MAGNIFY YOUR P
    ICTURE:"; PRINT : PRINT " (
    UP TO 3X FOR 8 INCH WIDE PAP
    ER.);" : PRINT
45 PRINT "*** NOTE ***": PRINT :
    PRINT " FOR 3X MAGNIFICATI
    ON, AND 8-INCH WIDE PAPER, I
    WILL PRINT THE PICTURE IN T
    WO PORTIONS.": PRINT : PRINT

47 IF MAG > 0 THEN PRINT "(MAGN
    IFICATION FROM PREVIOUS RUN
    = ";MAG;"X)": PRINT : GOTO 5
    5
50 INPUT "MAGNIFICATION DESIRED
    (1-3) ? ";MAG: IF (MAG < 1 OR
    MAG > 3) THEN PRINT CHR$(
    7): GOTO 50
55 PRINT : INPUT "NAME OF PICTUR
    E (TO BE PRINTED WITH UN- DE
    RLINE UNDER PICTURE): ";PICN
    AME$
60 PRINT
65 IF SCAN$(1) < > "" THEN GOTO
    230: REM : SCAN$ CONTENTS
    INTACT, SO SKIP DOING IT ALL
    OVER AGAIN !

70 TIME$(1) = "INITIALIZING: -
    ONE MOMENT PLEASE -
    (PICTURE PRINTOUT
    WILL TAKE A
    BOUT 5 MINUTES.)"
72 TIME$(2) = "INITIALIZING: - TH
    IS TAKES 30 SECONDS -
    (PICTURE PRINTOUT
    WILL TAKE A
    BOUT 10 MINUTES.)"
75 TIME$(3) = "INITIALIZING: - TH
    IS TAKES 45 SECONDS -
    (PICTURE PRINTOUT
    TAKES: 20 MIN
    . FOR 1ST 'PIECE';
    10 MIN. FOR 2ND.)"

```

```

80 PRINT TIME$(MAG): PRINT
100 FOR A = 0 TO 1: FOR B = 0 TO
    1: FOR C = 0 TO 1: FOR D = 0
    TO 1: FOR E = 0 TO 1: FOR F
    = 0 TO 1: FOR G = 0 TO 1
110 SCAN$(COUNT) = CHR$( G + 64)
    + CHR$( F + 64) + CHR$( E
    + 64) + CHR$( D + 64) + CHR$(
    C + 64) + CHR$( B + 64) +
    CHR$( A + 64)
120 REM *****
    * HERE, WE ARE BUILDING UP *
    * "REVERSE BINARY" STRINGS *
    * FOR THE PRINTER. "B"=0, *
    * AND "A"=1. EXAMPLE: *
    * SCAN$(19)="AAB@A@@" *
    *****
130 COUNT = COUNT + 1
140 NEXT G,F,E,D,C,B,A
150 IF MAG = 1 THEN GOTO 210
160 REM *****
    * IN LINES #170-200, WE *
    * "MAGNIFY" BY DUPLICATING *
    * EACH CHARACTER IN SCAN$ *
    * A NUMBER OF TIMES EQUAL *
    * TO THE MAGNIFICATION. LA-*
    * TER, WE DUPLICATE-PRINT! *
    *****
170 FOR STRING = 0 TO 127:MAG$ =
    ""
180 FOR CHAR = 1 TO LEN (SCAN$(
    STRING)): FOR M = 1 TO MAG:M
    AG$ = MAG$ + MID$( SCAN$(ST
    RING),CHAR,1): NEXT M,CHAR
190 SCAN$(STRING) = MAG$
200 NEXT STRING
210 PRINT "*** DONE ***"
220 PRINT CHR$( 7)
230 PRINT " PRINTER READY (Y/N)
    ? ";: GET ANS$
240 IF ANS$ < > "Y" THEN PRINT
    : PRINT "(PROGRAM EN
    DED.)": END
250 PR# 1
255 PRINT : REM THIS IS VERY IM-
    PORTANT TO DO AFTER PR#1;
    THE ANADEX WON'T GET INTO
    THE GRAPHICAL MOOD WITHOUT
    IT!
260 PRINT CHR$( 27);"7080";: REM
    SET PRINT WIDTH TO 8 INCHES.

270 PRINT CHR$( 28);: REM
    ENTER GRAPHICS MODE.

280 FOR LEVEL = 8192 TO 8192 + 8
    0 STEP 40

```

```

290 FOR SUB = 0 TO 7 * 128 STEP
    128
300 FOR SSUB = 0 TO 7 * 1024 STEP
    1024
310 FOR M = 1 TO MAG
320 PRINT ";015";: REM
    INDENT 15 DOT POSITIONS.

330 FOR INC = 0 + K TO 39: REM
    K IS USED FOR PRINTING 2ND
    HALF OF PICTURE, IF IT'S
    TOO BIG TO FIT THE PAPER!
    (SEE COMMENT; LINE #405.)

340 MEM = PEEK (LEVEL + SUB + SS
    UB + INC):MEM = MEM - 128 *
    (MEM > 127)
350 PRINT SCAN$(MEM);
360 NEXT INC: PRINT "1";: REM :
    DROP DOWN 1 DOT POSITION.
370 NEXT M
380 REM *****
    * THE "M-LOOP" DUPLICATES *
    * THE LINES FOR VERTICAL *
    * MAGNIFICATION. HORIZONTAL*
    * MAGNIFICATION WAS TAKEN *
    * CARE OF BY DUPLICATE *
    * CHARACTERS IN SCAN$'S. *
    *****
390 NEXT SSUB,SUB,LEVEL
400 REM
50: 'LEVEL' CONTROLS THE
    LARGEST INCREMENT (C 1/3 OF
    THE SCREEN), SUB MAKES THE 8
    LINE JUMPS, AND SSUB THE 1-
    LINE JUMPS. 'INC' SCANS A-
    CROSS A SINGLE LINE. THE RE-
    SULT IS THAT THE SCREEN IS
    SCANNED LINE-BY-LINE!!

402 IF MAG < 3 OR FLAG = 1 THEN
    410: REM SKIP THE 2-PIECE
    PRINTING IF IT FITS (MAG<3)
    OR IF SECOND PIECE ALREADY
    PRINTED (FLAG=1).

403 PRINT CHR$( 29); CHR$( 18):
    PRINT CHR$( 14); CHR$( 30)
    ;:L = LEN (PICNAME$):H = (5
    5 - L) / 2: HTAB H: PRINT PI
    CNAME$: PR# 0:PICNAME$ = "":
    REM : EXIT GRAPHICS, PRINT
    PICTURE NAME IN DOUBLE WIDTH
    MODE (55 CHR. SPAN PICTURE
    WIDTH.)

404 FLAG = 1: HOME : PRINT CHR$(
    7): VTAB 10: PRINT "NOW, PL
    EASE ADJUST YOUR PRINTER PAP

```

ER SO THE REMAINING HALF-P
ICTURE WILL BE PRINTED ON
A FULL SHEET OF PAPER." : PRINT

EQUATIONS FOR (SOME OF) THE BESSEL GRAPHS

```

405 REM IN THE NEXT LINE, K IS
    SET TO 20, TO REDUCE THE HI-
    RES SCAN TO THE RIGHT HALF
    OF THE SCREEN (SEE LINE
    #330.)

406 INPUT "(PRESS 'RETURN' WHEN
    READY.)";HOLD$: HOME : PR# 1
    : PRINT : PRINT CHR$(28):K
    = 20: GOTO 280

410 PRINT CHR$(29); CHR$(7): REM
    EXIT GRAPHICS AND RING BELL.

415 IF PICNAME$ ( ) "" THEN PRINT
    CHR$(18):L = LEN (PI$):H =
    (MAG * 40 - L) / 2: HTAB H: PRINT
    CHR$(30);PICNAME$:

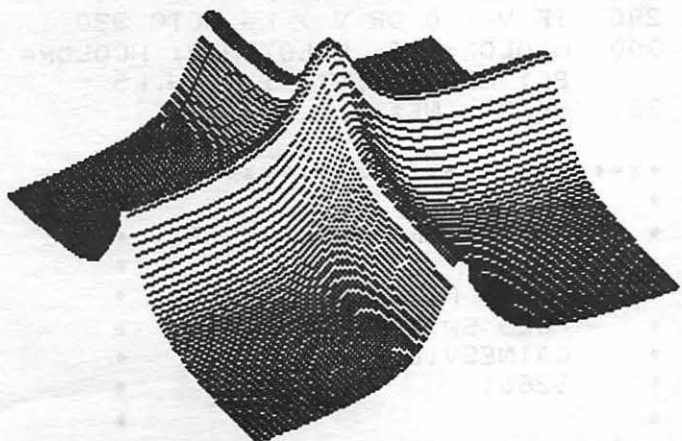
420 PR# 0
425 TEXT : HOME : VTAB 10
430 PRINT "VOILA !! HOW'S THAT ?
    ?"
440 PRINT : PRINT "TYPE GOTO 20
    FOR ANOTHER PLOT (AT SAME M
    AGNIFICATION, ETC.)."
  
```

```

26000 REM *****
    *
    * HOWIE MITCHELL *
    * 408 JACKSON AVE. *
    * LEXINGTON, VA. 24450 *
    * OCTOBER 1980 *
    *
    *****
  
```

```

27000 REM *****
    *
    * THIS PROGRAM IS WRITTEN *
    * FOR THE ANADIX DP-9501 *
    * PRINTER, BUT SHOULD WORK *
    * WITH THE "PAPER TIGER". *
    *
    *****
  
```



'COZY ROOFTOP'

```

3 REM *****
    *
    * PLOTPOURRI *
    * S COTTRELL *
    * SEPT. 1979 *
    *
    *****

30 X1 = - 6:X2 = 7:Y1 = - 6:Y2 =
    6:ZS = 10:RZ = 2:PC = 3:BC =
    0
  
```

```

*****
* THE 'OLD RUFFLED CROSS' EQUATION *
*****
  
```

```

35 DEF FN Z(X) = SIN (X * Y) *
    X * Y / 36 + ( ABS (X * Y) <
    ABS (DX) / 2)

36 REM
37 REM X1,X2=LOWR,UPR LIM,X
38 REM Y1,Y2=LOWR,UPR LIM,Y
39 REM RZ=RESOLUTION
40 REM ZS=VERTICAL SCALE
41 REM PC=PLOT COLOR
42 REM BC=BACKGROUND COLOR
50 REM ** THE 'OLD RUFFLED CROSS
    EQUATION **
  
```

```

130 NY = RZ * 87:NX = RZ * 50
140 HGR : POKE - 16302,0
150 DX = (X2 - X1) / NX:DY = (Y2 -
    Y1) / NY
160 H0 = 0
170 K1 = 58:K2 = Y2 - Y1:K3 = 90
172 K4 = 174:K5 = 191
200 FOR X = X1 TO X2 STEP DX
220 H0 = (X - X1) / (X2 - X1) * 1
    00:NH = - 1
240 FOR Y = Y1 TO Y2 STEP DY
250 ZZ = FN Z(X)
260 V = - ZZ * ZS - (Y - Y1) / K
    2 * K1 + K3 + H0
270 NH = NH + 1
280 H = H0 + NH * K4 / NY
290 IF V < 0 OR V > K5 GOTO 330
300 HCOLOR= PC: H PLOT H,V: HCOLOR=
    BC: H PLOT H,V + 1 TO H,K5
330 NEXT : NEXT
  
```

```

*****
*
* EQUATION AND PRINTING BY: *
*
* HOWIE MITCHELL *
* 7823 SW. 55TH PLACE *
* GAINESVILLE, FLORIDA *
* 32601 *
*
*****
  
```

```

3 REM *****
*
* PLOTPOURRI *
* S COTTRELL *
* SEPT. 1979 *
*
*****
30 X1 = - 4:X2 = 1:Y1 = - 4:Y2 =
1:ZS = 3:RZ = 2:PC = 3:BC =
0

*****
* 'COZY ROOFTOP' EQUATION *
*****

35 DEF FN Z(X) = 1 / ( SIN (X) +
1.08) + 1 / ( SIN (Y) + 1.08
) + (( ABS ( SIN (X)) > .995
) OR ( ABS ( SIN (Y)) > .995
)) * 2

36 REM
37 REM X1,X2=LOWR,UPR LIM,X
38 REM Y1,Y2=LOWR,UPR LIM,Y
39 REM RZ=RESOLUTION
40 REM ZS=VERTICAL SCALE
41 REM PC=PLOT COLOR
42 REM BC=BACKGROUND COLOR
50 REM ** COZY ROOFTOP EQUATION
**

130 NY = RZ * 87:NX = RZ * 50
140 HGR : POKE - 16302,0
150 DX = (X2 - X1) / NX:DY = (Y2 -
Y1) / NY

160 H0 = 0
170 K1 = 58:K2 = Y2 - Y1:K3 = 90
172 K4 = 174:K5 = 191
200 FOR X = X1 TO X2 STEP DX
220 H0 = (X - X1) / (X2 - X1) * 1
00:NH = - 1
240 FOR Y = Y1 TO Y2 STEP DY
250 ZZ = FN Z(X)
260 V = - ZZ * ZS - (Y - Y1) / K
2 * K1 + K3 + H0
270 NH = NH + 1
280 H = H0 + NH * K4 / NY
290 IF V < 0 OR V > K5 GOTO 330
300 HCOLOR= PC: H PLOT H,V: HCOLOR=
BC: H PLOT H,V + 1 TO H,K5
330 NEXT : NEXT

*****
*
* EQUATION AND PRINTING BY: *
*
* HOWIE MITCHELL *
* 7823 SW. 55TH PLACE *
* GAINESVILLE, FLORIDA *
* 32601 *
*
*****

```

```

3 REM *****
*
* PLOTPOURRI *
* S COTTRELL *
* SEPT. 1979 *
*
*****
30 X1 = - 4:X2 = .5:Y1 = - 4:Y2
= .5:ZS = 3:RZ = 2:PC = 3:B
C = 0

*****
* 'TULIP' EQUATION *
*****

35 DEF FN Z(X) = 10 - 1 / ( SIN
(X) + 1.08) - 1 / ( SIN (Y) +
1.08)

36 REM
37 REM X1,X2=LOWR,UPR LIM,
X
38 REM Y1,Y2=LOWR,UPR LIM,
Y
39 REM RZ=RESOLUTION
40 REM ZS=VERTICAL SCALE
41 REM PC=PLOT COLOR
42 REM BC=BACKGROUND COLOR

50 REM ** TULIP EQUATION **
130 NY = RZ * 87:NX = RZ * 50
140 HGR : POKE - 16302,0
150 DX = (X2 - X1) / NX:DY = (Y2 -
Y1) / NY

160 H0 = 0
170 K1 = 58:K2 = Y2 - Y1:K3 = 90
172 K4 = 174:K5 = 191
200 FOR X = X1 TO X2 STEP DX
220 H0 = (X - X1) / (X2 - X1) * 1
00:NH = - 1
240 FOR Y = Y1 TO Y2 STEP DY
250 ZZ = FN Z(X)
260 V = - ZZ * ZS - (Y - Y1) / K
2 * K1 + K3 + H0
270 NH = NH + 1
280 H = H0 + NH * K4 / NY
290 IF V < 0 OR V > K5 GOTO 330
300 HCOLOR= PC: H PLOT H,V: HCOLOR=
BC: H PLOT H,V + 1 TO H,K5
330 NEXT : NEXT

*****
*
* EQUATION AND PRINTING BY: *
*
* HOWIE MITCHELL *
* 7823 SW. 55TH PLACE *
* GAINESVILLE, FLORIDA *
* 32601 *
*
*****

```

```

3 REM *****
*
* PLOTPOURRI *
* S COTTRELL *
* SEPT. 1979 *
*
*****
30 X1 = - 4:X2 = 8:Y1 = - 4:Y2 =
8:ZS = 2:RZ = 2:PC = 3:BC =
0

*****
* 'CHRISTMAS TREES' EQUATION *
*****

35 DEF FN Z(X) = 1 / ( SIN (X) +
1.08) + 1 / ( SIN (Y) + 1.08
)
36 REM
37 REM X1,X2=LOWR,UPR LIM,X
38 REM Y1,Y2=LOWR,UPR LIM,Y
39 REM RZ=RESOLUTION
40 REM ZS=VERTICAL SCALE
41 REM PC=PLOT COLOR
42 REM BC=BACKGROUND COLOR
50 REM ** CHRISTMAS TREES EQUATI
ON **
130 NY = RZ * 87:NX = RZ * 50
140 HGR : POKE - 16302,0
150 DX = (X2 - X1) / NX:DY = (Y2 -
Y1) / NY
160 H0 = 0
170 K1 = 58:K2 = Y2 - Y1:K3 = 90
172 K4 = 174:K5 = 191
200 FOR X = X1 TO X2 STEP DX
220 H0 = (X - X1) / (X2 - X1) * 1
00:NH = - 1
240 FOR Y = Y1 TO Y2 STEP DY
250 ZZ = FN Z(X)
260 V = - ZZ * ZS - (Y - Y1) / K
2 * K1 + K3 + H0
270 NH = NH + 1
280 H = H0 + NH * K4 / NY
290 IF V < 0 OR V > K5 GOTO 330
300 HCOLOR= PC: H PLOT H,V: HCOLOR=
BC: H PLOT H,V + 1 TO H,K5
330 NEXT : NEXT

```

```

*****
*
* EQUATION AND PRINTING BY: *
*
* HOWIE MITCHELL *
* 7823 SW. 55TH PLACE *
* GAINESVILLE, FLORIDA *
* 32601 *
*
*****

```

```

3 REM *****
*
* PLOTPOURRI *
* S COTTRELL *
* SEPT. 1979 *
*
*****
30 X1 = 5.5:X2 = - 9:Y1 = 12:Y2
= - 16:ZS = .4:RZ = 2:PC =
3:BC = 0

*****
* 'FOREST' EQUATION *
*****

35 DEF FN Z(X) = ( SGN (X) / ( SIN
(X) + 1.08) - .8 * SGN (Y) -
.8 * SGN (X) + SGN (Y) / (
SIN (Y) + 1.08)) * (X + Y)
36 REM
37 REM X1,X2=LOWR,UPR LIM,X
38 REM Y1,Y2=LOWR,UPR LIM,Y
39 REM RZ=RESOLUTION
40 REM ZS=VERTICAL SCALE
41 REM PC=PLOT COLOR
42 REM BC=BACKGROUND COLOR
50 REM ** THE 'OLD RUFFLED CROSS
' EQUATION **
130 NY = RZ * 87:NX = RZ * 50
140 HGR : POKE - 16302,0
150 DX = (X2 - X1) / NX:DY = (Y2 -
Y1) / NY
160 H0 = 0
170 K1 = 58:K2 = Y2 - Y1:K3 = 90
172 K4 = 174:K5 = 191
200 FOR X = X1 TO X2 STEP DX
220 H0 = (X - X1) / (X2 - X1) * 1
00:NH = - 1
240 FOR Y = Y1 TO Y2 STEP DY
250 ZZ = FN Z(X)
260 V = - ZZ * ZS - (Y - Y1) / K
2 * K1 + K3 + H0
270 NH = NH + 1
280 H = H0 + NH * K4 / NY
290 IF V < 0 OR V > K5 GOTO 330
300 HCOLOR= PC: H PLOT H,V: HCOLOR=
BC: H PLOT H,V + 1 TO H,K5
330 NEXT : NEXT

```

```

*****
*
* EQUATION AND PRINTING BY: *
*
* HOWIE MITCHELL *
* 7823 SW. 55TH PLACE *
* GAINESVILLE, FLORIDA *
* 32601 *
*
*****

```

COLUMN-FORMATTER by Howie Mitchell

```

5  REM *****
   * 1-3 COLUMN TEXT FORMATTER*
   *****

10  TEXT : HOME
20  PRINT "*** ANADEX 1 - 3 COLUMN
    FORMATTER ***"
    -----": PRINT

30  PRINT "THIS PROGRAM ACCEPTS TEXT
    IN DATA LINES, AND THEN PRINTS
    THEM OUT IN A 1, 2, OR 3-COLUMN
    FORMAT (WITH APPROPRIATE CHARACTER
    SIZE).": PRINT

40  PRINT "A CENTERED, UNDERLINED,
    DOUBLE-SIZED TITLE WILL BE
    PRINTED IF DESIRED."

45  PRINT
50  PRINT "MENU:"; PRINT "----": PRINT

60  PRINT " 1. DATA FORMAT.": PRINT
    : PRINT " 2. PRINT OUT TEXT
    .": PRINT

70  PRINT : PRINT "WHICH ? "; GET
    ANS: PRINT ANS: IF ANS < 1 OR
    ANS > 3 THEN PRINT : PRINT
    "(PROGRAM ENDED.)": VTAB 23:
    END

80  ON ANS GOTO 200,300
200  REM *****
     *      DATA FORMATTER      *
     *****

210  DAYTA = 1000
215  Q$ = CHR$ ( ASC (" ") - 1)
217  HOME : HTAB 15: PRINT "*** TEXT
    ***": PRINT
220  FOR DAYTA = DAYTA TO DAYTA +
    30 STEP 5
225  PRINT "]" ; DAYTA ; "DATA" ; Q$ ; FOR
    N = 1 TO 30: PRINT CHR$ (95
    ) ; NEXT N: PRINT "3COL/
    2COL/"; : FOR N = 46 TO 75: PRINT
    CHR$ (95) ; NEXT N: PRINT "
    1COL/"

230  NEXT DAYTA
235  PRINT "]" DAYTA = " ; DAYTA ; " : GOT
    O 215 (MORE LINES)";

240  VTAB 2
299  END
300  REM *****
     *      PRINT TEXT      *
     *****

305  TEXT : HOME
310  INPUT " HOW MANY COLUMNS FOR
    TEXT (1-3) ? "; COLS: PRINT

315  INPUT " TEXT TITLE ? "; TITLE
    $: PR# 1
320  IF COLS = 2 THEN PRINT CHR$
    (25) ; CHARS = 12.5 * 4 - 5: C
    PL = 12.5 * 4
325  IF COLS = 3 THEN PRINT CHR$
    (26) ; CHARS = 15 * 8 / 3 - 5
    : CPL = 15 * 4
330  IF COLS = 1 THEN PRINT CHR$
    (18) ; CPL = 40
332  REM *****
     * CPL = # CHAR. PER LINE *
     * FOR TITLE. *
     * CHARS = # CHAR. PER LINE *
     * OF TEXT. *
     *****

335  READ LINE$: IF LINE$ ( ) "E
    ND OF DATA" THEN INDEX = IND
    EX + 1: GOTO 335
337  IF INDEX = 0 THEN 398: REM :
    NO TEXT, SO QUIT.
340  DIM SEN$(INDEX + (COLS - 1) *
    60): RESTORE
345  FOR N = 1 TO INDEX: READ SEN
    $(N): NEXT N
350  IF TITLE$ ( ) "" THEN L = LEN
    (TITLE$): PRINT CHR$ (14) ; HTAB
    (CPL - L) / 2: PRINT CHR$ (
    30) ; TITLE$: PRINT

352  N = 1
355  K = 57: FOR N = N TO N + 56
357  IF N > INDEX THEN 398
360  PRINT SEN$(N) ; IF COLS > 1 THEN
    FOR SKIP = 1 TO COLS - 1: POKE
    36, (CHARS + 5) * SKIP: PRINT
    SEN$(N + K * SKIP) ; NEXT SK
    IP
365  PRINT : NEXT N: N = N - 1 + 5
    8 * (COLS - 1) + (COLS = 1)
370  IF N < INDEX THEN PRINT CHR$
    (12) : GOTO 355

398  PR# 0
399  END
999  REM *****
     *      DATA FOLLOWS      *
     *****

10000  DATA END OF DATA
25000  REM *****
     *
     * HOWIE MITCHELL *
     * 7823 SW. 55TH PLACE *
     * GAINESVILLE, FLA. 32601 *
     * NOVEMBER, 1980 *
     *
     *****

```

contd.

SAMPLES OF 2 & 3 COLUMN FORMATS:

HERE ARE SOME STORIES YOU MIGHT ENJOY:

FROM: CHINESE FABLES.

A BLESSING IN DISGUISE.

ONCE THERE WAS AN OLD MAN WHO HAD A HORSE. ONE DAY HIS HORSE DISAPPEARED, AND HIS FRIENDS ALL CAME TO CONSOLE HIM FOR HIS LOSS. BUT THE OLD MAN SAID:

"HOW CAN YOU TELL THAT IT WAS NOT A LUCKY OMEN?"

HERE ARE SOME STORIES YOU MIGHT ENJOY:

FROM: CHINESE FABLES.

A BLESSING IN DISGUISE.

ONCE THERE WAS AN OLD MAN WHO HAD A HORSE. ONE DAY HIS HORSE DISAPPEARED, AND HIS FRIENDS ALL CAME TO CONSOLE HIM FOR HIS LOSS. BUT THE OLD MAN SAID:

"HOW CAN YOU TELL THAT IT WAS NOT A LUCKY OMEN?"

AFTER SEVERAL MONTHS, THE LOST HORSE RETURNED, BRINGING WITH HIM ANOTHER HORSE. THE OLD MAN'S FRIENDS WHO HEARD OF HIS HAPPY NEWS CAME TO OFFER THEIR CONGRATULATIONS AND THE OLD MAN REMARKED:

"HOW CAN YOU TELL THAT IT WAS NOT A BAD OMEN?"

BY AND BY HIS SON FORMED THE HABIT OF RIDING ON THE HORSE AS A HOBBY, AND ONE DAY HE FELL OFF AND BROKE HIS LEG. THE FRIENDS AGAIN CALLED AND EXPRESSED THEIR SORROW. THE OLD MAN SAID:

"HOW CAN YOU TELL THAT IT WAS NOT A GOOD OMEN?"

A YEAR LATER AN ORDER WAS GIVEN THAT ALL THE YOUNG MEN BE MADE TO JOIN THE ARMY AND FIGHT THE TARTERS. THE CRIPPLED SON, OWING TO HIS LAMENESS, WAS OF COURSE SPARED.

SO THE OLD MAN SAID TO HIS FRIENDS:

"NO, THEY ARE GONE."

"THE TONGUE IS PRESERVED BY REASON OF ITS SOFTNESS, WHILE THE TEETH ARE DESTROYED BY REASON OF THEIR HARDNESS. IS THAT NOT SO?" LAO-TZE NODDED IN ASSENT.

THE SOVEREIGN.

"SHANG," QUERIED CONFUCIUS OF TZE-HSIA, "CAN YOU TELL ME IN WHAT MANNER THE SOVEREIGN OF A STATE IS A SOVEREIGN?"

"HUMAN AFFAIRS ARE ALWAYS CHANGEABLE. HOW CAN YOU TELL WHETHER WHAT TODAY IS A MISERY MAY NOT TURN OUT TO BE A BLESSING TOMORROW?"

THE VIRTUE OF MECKNESS.

OPENING HIS MOUTH SO AS TO SHOW HIS TONGUE, CHANG TS'ING ASKED LAO-TZE, "IS MY TONGUE STILL THERE?"

"YES."

"ARE MY TEETH ALSO INTACT?"

"NO, THEY ARE GONE."

"THE TONGUE IS PRESERVED BY REASON OF ITS SOFTNESS, WHILE THE TEETH ARE DESTROYED BY REASON OF THEIR HARDNESS. IS THAT NOT SO?" LAO-TZE NODDED IN ASSENT.

THE SOVEREIGN.

"SHANG," QUERIED CONFUCIUS OF TZE-HSIA, "CAN YOU TELL ME IN WHAT MANNER THE SOVEREIGN OF A STATE IS A SOVEREIGN?"

"THE FISH OUT OF WATER MUST PERISH," REPLIED THE DISCIPLE, "WHEREAS THE WATER WITHOUT THE FISH CONTINUES TO BE THE SAME ELEMENT."

"YOU KNOW IT, SHANG, I CAN SEE," RETURNED THE MASTER.

LAUGH WITH OTHERS.

A BLIND MAN WAS IN THE COMPANY OF OTHERS. WHEN HIS COMPANIONS SAW SOMETHING FUNNY, THEY LAUGHED, AND THE BLIND MAN LAUGHED, TOO.

WHEN THEY ASKED HIM WHY HE WAS LAUGHING, THE BLIND MAN REPLIED, "SINCE YOU LAUGH, THERE MUST BE SOMETHING WORTH LAUGHING AT. CAN YOU BE CHEATING ME?"

A COMPASSIONATE MAN.

A MAN ONCE CAUGHT A TURTLE. HE WANTED TO MAKE IT INTO SOUP, BUT UNWILLING TO BE ACCUSED OF TAKING LIFE, HE BOILED A PANFUL OF WATER AND, PLACING A ROD OVER THE PAN, SAID TO THE TURTLE: "IF YOU CAN GET ACROSS THE PAN ON THE ROD I WILL SET YOU FREE."

THE TURTLE DID NOT QUITE UNDERSTAND THE INTENTION OF THE MAN, BUT HE DID NOT WANT TO DIE. SO SUMMONING UP ALL HIS WILL POWER, HE ACCOMPLISHED THE IMPOSSIBLE.

"WELL DONE," SAID THE MAN. "BUT NOW TRY IT AGAIN."

NEVER TOO LATE TO LEARN.

"THOUGH SEVENTY," SAID DUKE PING OF CHIN, "I STILL WANT TO BE A STUDENT. BUT I AM AFRAID IT IS TOO LATE."

"WHY SHOULD YOU NOT BE LIKE A CANDLE?" RETORTED SHIH-KUANG.

"DOES A SERVANT DARE TO MAKE FUN OF HIS SOVEREIGN?"

"HOW DARE I, WHO AM BUT YOUR BLIND SERVANT, MOCK AT MY MASTER? BUT, IN MY OPINION, TO BE A STUDENT IN YOUTH IS LIKE THE FIRST RAYS OF THE RISING SUN. TO BE STUDIOUS IN MIDDLE AGE IS LIKE THE DAZZLING LIGHT OF THE SUN AT NOON. TO BE STUDIOUS IN OLD AGE REMINDS ONE OF THE LIGHT OF A CANDLE. NOW, BETWEEN THE LIGHT OF A CANDLE AND UTTER DARKNESS, WHICH IS TO BE PREFERRED?"

Information in the Post-Gutenberg Era

Technological advances in the past few years have brought major changes in research, storage, and dissemination of information. Under Fund sponsorship, Anthony Smith investigated newspapers—leading consumers of the new technology—in France, Germany, Japan, the United Kingdom, and the United States. Mr. Smith's work, presented here in a condensed article, has just been published by Oxford University Press as Goodbye Gutenberg: The Newspaper Revolution of the 1980's.

FOR FIVE CENTURIES the printing press spread its influence over the whole of mankind: successive refinements came, but none changed the notion of disseminating information through multiplication of copies. Now the self-contradictions at the heart of Gutenberg's system are becoming evident. There is too much research for scientists to sift through and absorb, too much advice for politicians and executives to ponder; even the most remote specialist finds it difficult to keep abreast of bibliographical and abstract material that piles up unread, unfiled.

What will make the contribution of the 1980s so different from that of the 1970s is that the new word-processing and telecommunications devices are becoming much broader in scope. They are moving from internal message and filing systems to general instruments, available in cheap, flexible, and accessible as well as specialized and sophisticated form. On top of that these new devices are becoming interconnected worldwide.

The Newspaper Revolution

The newspaper industry has gone farther than any other traditional industry in adopting the new equipment, not as a way of self-displacement, but as a means of reducing its production costs. In the past 20 years or so, the newspaper industry has discovered that its chief advantages have all turned against it. Its public has become more suburban and therefore more expensive to reach and more diverse in its information needs. Labor and newsprint have become costly.

In the 1890s the mass newspaper was a one-cent miracle; in the 1980s no one would trouble to invent it, but it

exists and is still enormously profitable if managed on modern lines.

Newspapers are, moreover, the nodal points for general information in all western societies and because they possess a large capacity to collect large quantities of information (only a tiny proportion of which appears in print), they may well play new roles in the near future. Instead of providing a package of generalized material to many households that require only small segments of what they get, newspapers may choose to slim down and send out selected categories of information through the new domestic devices. They could simultaneously increase business by printing some material that now remains unused but is potentially saleable to small pockets of customers.

Only in the very long run might the newspaper cease to exist in its traditional garb. A newspaper electronically transmitted to the home has been predicted since the 1930s, and could come about one day as a result of the interaction of several new devices, but the newspaper—almost in the form that we know it now—is likely to remain around at least until the end of the century. Nonetheless the new internal electronics of the newspaper industry act as an intermediary between the specialized data base and the domestic user.

The newspaper industry in the United States and Japan already has been substantially altered in this way, and that of Europe will be transformed in the next decade.

New Concepts of Information

From the welter of pioneering practice it is possible to see how these new technologies are reshaping concepts of information. First, they treat the public as individuals, instead of supplying a farrago of information to the whole waiting audience; the individual reader makes the initial choice, like a contemporary copy editor summoning the newspaper's stories from the computer store.

Second, the new system tends to abolish distinctions between audiences of one and audiences of many. The typewriter and the printing process merge. Word processors and computer terminals can send messages to an individual and publish for a larger audience at the same time. Quite new forms of discourse and collective composition become possible in the new age and already exist in embryo among groups of specialized colleagues.

Third, a new kind of knowledge worker is emerging from the ancient profession of librarianship: a technician who is rather more than a glorified filing clerk and is an expert in searching and manipulating data bases. This new kind of expert is already at work in some of the learned professions—the modern lawyer's office may draw upon a number of data bases, as well as general information centers such as the New York Times Information Service, and a new kind of legal clerk is emerging who can use these electronic files as forensic tools, offering lawyers far more subtle and individualized research assistance than before.

Fourth, in the new systems distance is not an important cost factor; satellite networks handle bulk text at costs so low that the only important factors relate to the amount of traffic and the number of destinations rather than the transmission distance. In the "paperless society," it is not so much the paper as the transportation of paper that is reduced or eliminated. What is essential, however, to the post-Gutenberg era, is that selection of the product is made by the receiver rather than the publisher, and printing is accomplished after the physical journey.

Impact on Government and Business

The two areas of society likely to undergo the most profound but subtle transformation from the new systems are government and business.

A modern government consists of layer upon layer of advisory expertise, human information-processing that cocoons decisionmaking. Every national policy is forced through a bureaucratic funnel made necessary by the complexity and inter-connectedness of decisions, which makes the final judgment harder and harder to reach.

There exists, of course, no electronic substitute for wisdom, but several important starts are being made to introduce the texts of politics and government to the computer. Both the White House and the Library of Congress use computerized systems for tracking legislation and key contemporary issues. The White House computer can indicate where mutually conflicting objectives (e.g., lowering inflation and lowering unemployment, or saving energy and providing new jobs in certain geographical areas) are at work in the nation or the Administration.

In the course of time, the computer could help the politician claw back some of the power that has been dispersed into the enveloping bureaucracy. The new systems should, in any case, help greatly to simplify and reduce the flow of paper, reduce the sheer numbers of processing

and administrative staff, and enable those in charge to track the progress of policy.

Perhaps the most important general principle entailed in many of the new computerized methods for transmitting text is the by-passing of institutions and layers of hierarchy that have traditionally been concerned with processing information. Wire services can send material directly to readers without passing through newspaper editors. Upper management can channel information to junior staff without passing it through middle management. Pressure groups and lobbies can transmit material to their constituencies without having to persuade newspapers and magazines to carry it. Electronic systems are natural by-passers and corner-cutters.

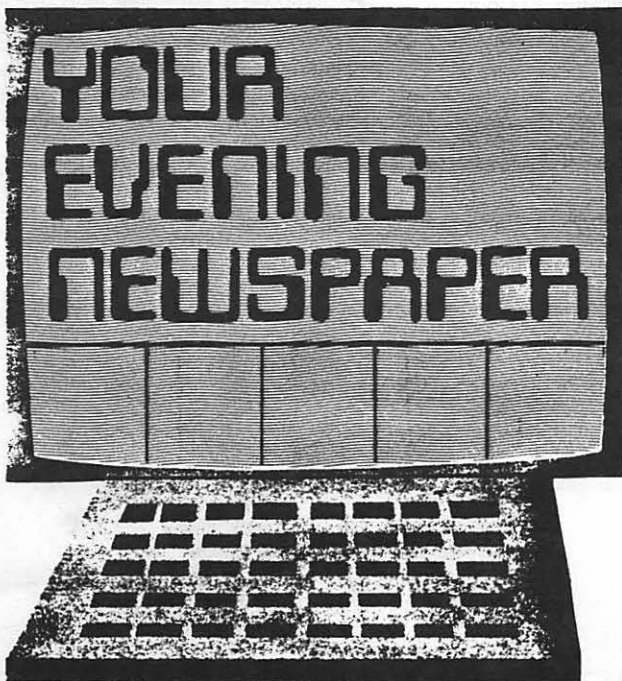
Revolution Brings Problems

The text revolution will, however, bring with it a whole series of new problems of domination and dependence, far more than did the first decades of cinema, radio, and television. Although much of the technology has been developed in Western Europe, Japan, and elsewhere, and although the research and expertise are far from being exclusively American, the major market for computers and much of the terminal equipment is in thrall to U.S. capital. The domestic airlines of several Eastern European countries operate their own bookings system through a computer stationed on U.S. soil. A Swedish fire hazard data base is held in Florida. Many Canadian payrolls are calculated by computers south of the border.

At the same time the computer market of most western nations is controlled mainly by IBM, despite the growth of other suppliers. Inevitably, questions of sovereignty and ultimate control will arise and become politically sensitive matters. A high degree of international interdependence is developing, and so dispersed will storage become that data bases of even the most powerful societies will be vulnerable. Satellite and fiber networks are becoming so complex, and the collective vulnerability so great, that no society can hope to function without considering the interests of others. The dominant supplier will find itself in a position of stewardship, rather like the librarian of one of the great manuscript collections of the world, protector of a common store of knowledge.

Nonetheless, one should be prepared for intense conflict to break out from time to time, particularly between Third World societies and the rest, over the ways in which the constitutional system and prevailing information doctrine of the United States tend to foreclose upon the national decisionmaking of other societies.

One of the great long-term achievements of Gutenberg was to break up the international information system dependent upon Latin, and open up the circulation of information in the vernacular; information became more subject to national and local controls. Today, the computer



ON CIRCUIT AVENUE AND ALL ABOUT THE TOWN

By LOUISE ALDRICH BUGBEE

The Dreaded Computer

Yesterday I got a letter from the Kelley House with a return address in Dearborn, Mich. Actually, the letter was from a nice lady who evidently lives and works in Dearborn and vacations at the Kelley House. I won't mention her name or her business address because I suspect she works with computers and computers are vicious, spiteful things and might take revenge for her agreement with me about them.

She writes: "Never cease war on computers. You've got to keep after them. In addition to all their other drawbacks, one breath of too warm or too cold air brings on fever, chills or swooning, which ever can complicate our lives to the greatest degree. The fanned and fainting ladies of Edwardian fiction are breathtakingly down-to-earth and robust compared to the computers of today."

It is wonderful to find someone who understands. This nice lady would never tell me I get too emotional about a mechanical thing which can neither think nor feel and never makes mistakes except when the human operator involved makes a human error.

I never was emotional about computers, or any other tool or piece of machinery invented for the service of humans. If they work and save time and energy, I use them. If they complicate my life and frustrate me, I junk them. I respect them about as much as I respect my paycheck. There is no emotion involved but if it is something I need and intend to use, I don't get careless with it.

What I'm critical and vocal about is the enthusiasm and awe, almost amounting to worship, that the humans who invent, work with and play with these new tools give computers.

Computers will change business, they say, and government and even the quality of human life, our way of thinking and living and even our attitudes. Tools change nothing of any importance. The most even the good ones can do is give humans more time to think and any change, for better or worse, is brought about by the way the humans use their additional time.

It seems to me that the ones most enthusiastic about computers have no idea about using the time wisely. They use it admiring computers and figuring out what more the computers can do.

When I ask, "In what way will this improve mankind? Will it make people better, kinder, more intelligent, more inclined toward peace and understanding?" They tell me how computers can be used in business for making money or for winning wars. One guy even told me that computers could change the voting system. Elections could be held over television and we could vote in our own homes by pushing a button. When I asked if this would make the average voter more aware of what was going on, more intelligent, more interested in good government, I was told that wasn't the point.

Good or Evil

Ah well, it was ever thus. Any tool ever invented can be used for good or evil. Humankind shouldn't expect anything mechanical to improve them. We have to do that ourselves. We are rather slow about it and when I seem to get emotional about machines I'm only jealous and fearful because they distract the human mind from working on this rather important task of improving humanity.

I console myself speculating that the invention of the wheel may have caused a lot of excitement and big talk before they got the edges rounded off so it could be used on a wheelbarrow to lighten the load humans toiled.

My new friend of the Kelley House and Dearborn is right though. Where computers are concerned, "You've got to keep after them." Not long ago a man I know phoned for boat reservations several weeks in advance of the intended trip. He was told that they couldn't make reservations that far in advance. The computer had broken down and until it was fixed they were taking only immediate reservations because the work had to be done by hand. That's a small, unimportant thing but it scared me. If, before we get the computers working consistently in warm or cold weather and without dependence on some plug inserted into a source of power, we depend upon them to such an extent that we stop work when they do, if we can't solve problems with our own brains and do the work with our own hands, civilization could stop with a jolt when the tools we use to run it break down or lose their power.

A few weeks ago Nancy Billings, an antique dealer, was here looking over the collection of used furniture in the house. The most valuable, according to Nancy, was a highchair I thought was worthless to anyone else but valuable to me because I knew the story of it. My grandfather had made it for my father using an axe, a plane and a jackknife. Other tools had been invented at the time, saws and such, but Granddad didn't have them handy. The finished product is primitive. I've always liked it because it is family. Now I value it more. I'll keep it as a symbol and a reminder. Also as a warning not to depend on tools and mechanical things too much. Humans should keep the ability to carve out a highchair, or a life, without all the latest tools.

Of course the memory bank of computers would be useful, for some people. I'd forget which button to push for any bit of information I wanted to remember.

(Ed. Note: This presents an interesting opinion of computers from a "non-believer". Is there anyone who wishes to comment? We know, for instance, that an APPLE II is used by Mother Earth News, who seem to be down-to-earth folks trying to help humankind.)

and telecommunications are bringing about a new internationalization of information, despite the heavy dependence upon American corporations. Here lies another aspect of the reversal of Gutenbergian logic.

The technology that produces text is the most crucial sustaining technology in any civilization. Its transformation in the late 20th century forebodes a slow upheaval in the fundamental institutions of administration, business, and education—everywhere that information is stored and exchanged.

ANTHONY SMITH



Apply apples

WASHINGTON APPLE PI
MAIL ORDER FORM

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

PROGRAM DISKETTES

Members: \$5.00 picked up at meeting
 \$7.00 mailed to you...

Non-members: \$8.00 per disk picked up at meeting
 \$10.00 mailed to you...

| | | | | | |
|-----------|------------------|-----|-------------|---------------------|-----|
| Volume 1 | Utilities I | () | Volume 22 | Utilities VI | () |
| Volume 2 | Utilities II | () | Volume 23 | Games VIII | () |
| Volume 3 | Games I | () | Volume 24 | Games IX | () |
| Volume 4 | Games II | () | Volume 25 | Utilities | () |
| Volume 5 | Games III | () | Volume 26 | Stocks/Investments | () |
| Volume 6 | Games IV | () | Volume 27 | Math | () |
| Volume 7 | Games V | () | Volume 28 | Planetfinder | () |
| Volume 8 | Utilities III | () | Volume 29 | Utilities | () |
| Volume 9 | Educational I | () | Volume 30 | Games | () |
| Volume 10 | Math/Science | () | Volume 31 | Plot Utilities | () |
| Volume 11 | Graphics I | () | Volume 180 | Dungeon Designer | () |
| Volume 12 | Games VI | () | Volume 181 | Beginner's Cave | () |
| Volume 13 | Games | () | *Volume 182 | Lair of Minotaur | () |
| Volume 14 | IAC Utilities IV | () | *Volume 183 | Cave of the Mind | () |
| Volume 15 | Games VII | () | *Volume 184 | Zyphur Riverventure | () |
| Volume 16 | Utilities V | () | *Volume 185 | Castle of Doom | () |
| Volume 17 | Graphics II | () | *Volume 186 | Death Star | () |
| Volume 18 | Educational II | () | *Volume 187 | Devil's Tomb | () |
| Volume 19 | Communications | () | *Vol. 181 | required with these | |
| Volume 20 | Music | () | disks. | | |
| Volume 21 | Apple Orchard | () | | | |

 TOTAL ORDER = \$ -----

Check here if you want these shipped---

NAME -----

ADDRESS -----

CITY, STATE, ZIP -----

TELEPHONE -----

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

Make checks payable to "Washington Apple Pi"

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

GROUP PURCHASE PLAN

In order to more effectively plan for club group purchases we would like to have some idea of your needs. Please check the items of interest and either bring the form to the next meeting or mail it to:

Howard Lefkowitz
11508 Colt Terrace
Silver Spring, Maryland 20902

When we receive sufficient interest in an item, where the quantities qualify for group purchase, we will be in touch with you. A one-third deposit is needed for all items before we can place an order.

Hardware

Floppy Disk: _____ ; Disk with Controller _____
Language System _____ ; DOS 3.3 _____
Applesoft Card _____ ; Integer Basic Card _____
Parallel Card _____ ; Centronics Card _____
Serial Card _____ ; Communications Card _____
Prototype Card _____ ; _____
New Apple 80 Column, Lower Case Card _____
Silentype Thermal Printer _____
IDS Paper Tiger Printer 440 _____, 460 _____
Hayes Micro Modem _____
Z-80 Microsoft System _____
ALF Music Synthesizer _____
Clock/Calendar Card _____
Graphics Tablet _____
CRT Monitors Sanyo 9" _____ Sanyo 12" _____
SOROC IQ 12 _____
Other _____

Software

Apple FORTRAN _____ ; PILOT _____
Apple Writer _____ ; Apple PLOT _____
Apple Post _____ ; DOS 3.3 Tool Kit _____
Visicalc _____ ; Desk Top Plan _____
CCA Data Mgmt. _____ ; Dakin-5 3.3 Utilities _____
Other _____



**INTERNATIONAL
APPLE CORE**
TM

**APPLE
ORCHARD
SUBSCRIPTIONS**

P. O. BOX 2227 SEATTLE, WASHINGTON 98111, USA

The International Apple Core will make individual subscriptions to "The Apple Orchard" available commencing with Volume I, Number 2 to be published in September, 1980.

NAME _____

STREET _____

CITY _____ STATE _____ ZIP _____

COUNTRY _____

Annual Subscription Rate: \$10.00 per year

First Class Postage: \$5.00 per year additional (required for Canada, Mexico, APO, and FPO addresses)

Overseas and other foreign air mail postage (required): \$10.00 per year additional

TOTAL REMITTANCE ENCLOSED: \$(USA) _____

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

Apple Orchard Subscriptions
P.O. Box 2227

7/7/80

Seattle, Washington, USA 98111

MESA, INC.

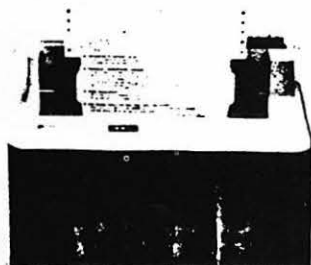
OFFERS EXCLUSIVE DISCOUNT PRICING TO APPLE PI MEMBERS!

BECAUSE OF YOUR QUALIFIED MEMBERSHIP
IN APPLE PI, YOU CAN OBTAIN SIGNIFICANT
DISCOUNTS ON THE PURCHASE OF:

INTEGRAL DATA SYSTEM'S "PAPER TIGER" PRINTERS

- * MESA, INC. IS AN AUTHORIZED
DISTRIBUTOR FOR IDS
- * SERVICE - AUTHORIZED DEPOT
SERVICE CENTER - SPARE PARTS &
FACTORY TRAINED STAFF

**TO PLACE AN ORDER, SIMPLY CALL
MS. LESLIE HARE AT 301/948-4350
WITH APPROPRIATE PROOF OF MEMBERSHIP



Integral Data Systems, Inc.

MESA, INC.

16021 INDUSTRIAL DRIVE
GAITHERSBURG, MD 20760
(301) 948-4350