mini'app'les

apple computer user group newsletter

MARCH 1979 (ALSO FEBRUARY 1979) ISSUE

EDITED BY

Daniel B.Buchler
13516 Grand Avenue South
Burnsville
Minnesota   55337
612-890-5051

ANNOUNCEMENTS:

MARCH 14th USERS' GROUP MEETING at 7:30 p.m. at Computerland

1. CHUCK BOODY (873-2227) will give a talk and demonstration
   entitled:-

   MUSIC & THE APPLE

   Chuck helped design a music synthesizer board which works in
   conjunction with an external amplifier and speaker system.
   According to one listener, it sounds like a "symphony orchestra".
   The board provides frequency control (notes), 3 voices, loudness
   including Attack and release and some degree of wave shape
   control. Talk will be application oriented.

2. The Business meeting will be confined to a discussion on dues,
   candidates for office, functions of the officers, etc for a
   formalized MINI'APP'LES users group. See minutes and other
   items in this newsletter.

APRIL 11th USERS' GROUP MEETING

1. ELECTION OF OFFICERS FOR MINI'APP'LES

AN APPEAL FROM THE EDITOR

Included under APPLE PUGETSOUND PROGRAM LIBRARY EXCHNAGE are several original
articles and letters to the editor, all of which were compiled
by that users' group. WE CAN DO THE SAME. This editor was exceptionally busy
earning a living during the past 2 months and was unable to do any original work.
We need you, the MINI'APP'LES users to provide material. All of you who program
must have some ideas and/or have developed some interesting techniques, programs
or hardware. Let us know about those things. Let others have the benefit of
your experience ! Please drop a line to this editor. We need your input !

PLEASE RETURN YOUR COMPLETED USER SURVEY IF YOU HAVE'NT YET DONE SC.

MINUTES OF FEB 12th USERS' MEETING

1. Approximately 22 users attended this meeting.
2. This editor pointed out that the current method of financing the users' group through a $1 charge for a program from the user contributed bank was inadequate. It was suggested that a $7 to $10 annual dues payment be established. A show of hands indicated unanimous approval.
3. Discussion on a formal users' group organization with a board, president, etc. When the users group was first started in the summer of 78, such an organization was turned down in favor of a simple unstructured group.. By a show of hands it was evident that the majority now favored a more formal organization.
4. It was agreed that the business portion of the March users' meeting would be devoted to a discussion of a formal users' group. Subjects to be discussed would include financing and candidates for offices of the proposed formal organisation. It will also be necessary to define the offices and their function. (see elsewhere in this newsletter)
5. It was agreed that the business portion of the April meeting be devoted to an election for offices of the users' group.

Following the above discussions, Stan Brooks gave a very enlightening talk and demonstration about interfacing to the Game I/O port of an Apple.

WHY WE MUST CHARGE DUES FOR MEMBERSHIP IN MINI'APP'LES.

Projected annual expenditures:

•Newsletter printing	$432
assume: 150 copies/issue	
8 pages/issue	
3 cents/page (currently printed by Honeywell	
at a cost of 2cents per page- but that source	
might not be available in the future.	
Instyprint charges around 3cents per sheet	
in quantities of 200)	
•Newsletter mailing 120@ 15 cents/issue	$216
•Newsletter mailing labels 150 x 12	18
•Purchase of programs for user bank. assume one $25	
every 3 months.	$100
•Floppy disks for storing user contributed programs	25
•Subscriptions to other user groups' newsletters. Assume	
6 at $10 each	60
•Rental of facilities for extraordinary meetings. Assume	
2 per year at $50 each.	$100
Expenditures Total	$951

Projected Income

•20 Out of town subscriptions	@ $7 per annum	$140
•100 Local members	@ $10 per annum	$1000
	Income Total	$1140

The above projections assume free distribution of programs to local members. Some users' groups make up disks or tapes of new programs and sell them to their members at cost. The current method of selling user programs to members for $1 has yielded an income of about $70 against costs of about $120 leaving Computerland $50 in the hole.

NEWSLETTER DATING:

Maybe you are wondering what happened to the February newsletter. Well there was'nt one- but don't get mad. Up_til now the Feb newsletter would have been written in Feb and published in March. That is sort of inconsistent with the way the publications business works. So the Feb newsletter became the March newsletter which was written (mostly) in February and published in March. Perhaps we should call it the April newsletter ?

WOZPAK

As stated in the last two issues of the newsletter, we are trying to purchase a copy of the WOZPAK to be used as a reproduction master. WOZPAK is a set of notes collected by Wozniak of Apple pertaining to the design and programming of the Apple II. There are copies of these notes floating around in several West coast users groups. The guys from Apple who visited us in Jan suggested that we could get a copy from Jade Products of California. Unfortunately this was a false lead. Puget Sound Program Library Exchange have announced the plan to make a set available in March. The price will be $20. It will be approximately 300 pages in length. If we can get permission to reproduce, the price to those who have signed up will be about $7. If anyone else who has not already signed up wants a copy, please let me know immediately.

NEWSLETTER MAILING LABELS

In the past Computerland have kindly assisted us in addressing the newsletter. This was done by hand on a time available basis and consequently was very prone to error. Chuck Thiesfeld has volunteered to print Mailing Labels using a Disk based Mailing List Program which the users' group will purchase for $20. We had originally planned to write a program ourselves but on seeing a demonstration of the commercially available program, it was decided that the investment could be justified (another reason for a formal MINI'APP'LES).

USERS GROUP OFFICERS DUTIES.

This is a subject for discussion at the next users group meeting. However the following might provoke some thought. - these are just ideas:-

President: Presides over the board of officers and board members if any
                  Organizes monthly meeting
                  Reviews newsletter draft

Secretary: Interface with other user groups
                  Writes the minutes of the meeting which are published in newsletter

Program Editor-Basic:
                  Review Basic programs for user bank
                  Review Basic program articles for publication in user newsletter
                  Maintain User bank and keep it up to date

Program Editor Assembly:
                  Same functions as above except for machine programs.

The Newsletter Editor could be a separate board function or a parallel function assumed by a board member. The Newsletter editor
                  publishes the newsletter
                  keeps the books (since most of the costs are newsletter related
              unless an office of treasurer is defined)

APPLE PUGETSOUND PROGRAM LIBRARY EXCHANGE (A.P.P.L.E.)

A.P.P.L.E. is the users' group for the Seattle, Washington area. They claim to be the largest group in the country with over 600 members spread throughout the USA and elsewhere. Their newsletter is superb. We are exchanging newsletters with them. If you want to be an individual member, you can subscribe for $7.50 plus a $2.50 application fee. Their address is

> 6708 39th Avenue S.W.
> Seattle, Washington, 98136

A.P.P.L.E. is in the process of compiling all their 1978 monthly publications into a single volume which will be called "PEEKing at Call - A.P.P.L.E " Price will be $8.50 and is supposedly available now from the above address.

A.P.P.L.E. has given us permission to extract from their newsletter provided we give credit. The following articles are from their Jan 79 newsletter. Program listings have been omitted to conserve space. Copies of such listing are available for a few cents from this Editor

CHAINING PROGRAMS IN APPLESOFT

There now exists a method whereby user programs written in Applesoft can chain between programs and retain all variable values. Note that when chaining between programs, all variables are kept.

To use: At the beginning of your program, put in the following three POKEs:
POKE 1013, 76: POKE 1014, 16: POKE 1015, 2

Note that these POKEs only have to be done once; they could be enclosed in your HELLO program, since they never need to be rePOKEd.

When you wish to chain to another program, use the following two steps:
PRINT CHRS(4); "BLOAD CHAIN, AS210"
& "program name"

The first step loads the chaining ability into the computer; the second step actually does the chaining. These two steps should be directly next to one another in your program.

Val,

In working with the disk, I found the file handling program to be of interest. Here are a few of my observations on files, relating to that program.

The "documentation" (underlined) says that in program control, the DOS commands maybe deactivated by the next printing of 'control D', even if it is a null type of entry. I notice that the listed program has several of these null Print DS (for control D) lines. I have found that this seems to be unnecessary and an extra program step. The 'control D' entry used to CLOSE the file after writing or reading will do the same thing.

I have several programs running with the disk files and have not had any problems so far, by not including a null 'control D'.

Also, I think I like the way Apple does rather than the way Apple says to wit:
Disk documentation says to let DS = 'control D' and then to use PRINT DS in the program control, i.e.:
10 PRINT DS; "WRITE FILE"; ",R";n

However, in their Animals program, they initialize somewhat as follows:
10 OFS="' 'control D' OPEN FILE, Ln"
20 RFS="' 'control D' READ FILE, R"
Then to use, you go
30 PRINT OFS
30 PRINT RFS;n

I have used this in my own programs and it works very well. Much better, I think than the method that they describe.

Jim Alguire
P. O. Box 6756
Ketchikan, Alaska 99901

WRITE - A.P.P.L.E.

Dear CALL-APPLE,

In the last issue of CALL APPLE, two methods of avoiding the "end of disc" error were suggested. A third, most useful after a program has been debugged, is to utilize the "ON ERR GO TO" command. "ONERRGOTO 10000" can be written at the beginning of any section that may cause an "end of disc" error. Then write

10000 PRINT (CONTROL) D; CLOSE
10010 CALL - 936: VTAB12:PRINT "END OF DATA ON DISC"
10020 FORN = 1 TO 1000: NEXT N
10030 POKE 216,0: REM CLEARS ERROR FLAG
10040 GO TO (any statement you wish)

The advantages of this method are its simplicity and flexibility. Whenever an "end of disc" error is encountered, the program, instead of bombing out, jumps to 10000 to let you know that you've gone too far, then jumps to any statement you desire to pick up the program again.

I continue to enjoy CALL-APPLE. With regard to the running debate on the level of technical sophistication of the articles, I don't mind the complex ones as long as they assume I know virtually nothing about computers.

I'm looking forward to whatever you're going to be doing for professionals (mentioned a few issues back) e.g. doctors and lawyers.

Marc Schwartz, M.D.
Editor, MICRO-PSYCH
26 Trumbull Street
New Haven, Connecticut 06511

CONVERT WITHOUT DISK

If you have an Apple Com Card and a printer with tape punch and read you can convert Integer Basic to Applesoft by punching tape from an Integer program listing and reading back into Applesoft.

Ron's current offering should delight many disk-o-philes. "Split Catalog" is designed to be used as a boot-up program. When your catalog exceeds 18 lines, it will be displayed in two columns, avoiding the need to hit a key for the next page. "Disk to Tape Transfer" and "Disk to Disk Transfer" are two similar and related programs. The former allows you to select by number X programs which will be transferred to tape in one continuous operation. The latter permits you to also designate by number X programs to be transferred from one disk to another, using only one drive.

EXTRACTS FROM A.P.P.L.E. JAN NEWSLETTER

HIGH CRIMES (and how to commit them)

by Val Golding

Have you ever tried to enter a program from a listing in Call-Apple or another magazine, only to find you get a ***SYNTAX ERR message when you try to enter a line that reads, for example: 100 DEL 0,300?

We will show you here how to commit a "crime" and get away with it! We use the word "illegal" because the DEL command and certain others (in Integer Basic) are illegal under program control, and as originally designed, may be used as direct commands only. (A direct command is one which may be executed from the keyboard only.) The reason these commands are illegal from within a program is that if not properly used, they can damage or destroy the program they are used in.

The so-called "illegal" commands include: LOAD, SAVE, RUN, DEL,   NEW, CLR, HIMEM:, and LOMEM:. We do not agree with Apple on the necessity of restricting the use of these commands to direct use, as many of them are handy in a program. Your Apple is a very powerful computer, and we believe that you should not have to do anything (such as manually set HIMEM:) that your computer can do for you.

The art of placing these commands in your program is known as "hand-patching" a program. Again, you must have some knowledge of how Integer Basic works, and for this we refer you to the article "Apple II Basic Structure" in the May Call-Apple, by S. Wozniak. It is also contained in Wozniak's "Renum-Append" article in the Wozpak. You will also need to be familiar with the Integer Basic "tokens", a chart of which appeared in the October Call-Apple.

Our "Programmer's Workshop" program will be very helpful to this project, but is not essential. The Basic procedure for entering these commands is to substitute an illegal command for a legal one already contained in the program. For example: you wish to set HIMEM: within the program. Enter a statement that reads "0 PRINT 16384". Then go into monitor and modify (or hand-patch) so that this line will end up as "0 HIMEM: 16384". One caution: HIMEM: or LOMEM: should always be the first line of a program.

The simplest way to handle this is to enter all lines you wish to modify before you enter the balance of the program. This will make the Basic lines to be changed easy to locate in memory. First you must determine where your program begins in memory. You can do this with the following statement, entered directly from the keyboard:

PRINT PEEK (202) + PEEK (203) * 256

This will give you, in decimal form, the address of the first basic line in memory, which must then be converted to hex. If you have a 32K machine and entered line 0 as shown, the answer would be 32760 (without disk up). This converts to 7FF8 in hex.

Now hit reset to go into monitor and type 7FF8. 7FFF and hit return to display a "hex dump" of those locations. This is what you will see:

7FF8 - 08 00 00 62 B1 00 40 01

If you have done your homework, you will know at this point that 08 is the number of bytes in the statement, 00 00 is the line number and 62 is the hex byte for PRINT. Now, counting up from 7FF8, you find the address of the 62 byte is 7FFB (Remember, A comes after 9 in hex). This is where we want to do the actual patch. 10 is the hex token for HIMEM: so type 7FFB: 10 and hit return. Now Control C back to Basic and LIST and you will find that PRINT now reads HIMEM:.

This same procedure may also be followed to create an illegal line number such as 65535. In Integer Basic, numerical values are limited to the range of ± 32767, since the high bit of the high order byte is reserved as a sign bit. In the case of a line number, Apple views this as an "unsigned logical constant", meaning there is no requirement for a sign bit in a line number, as all line numbers have only positive values. Thus we may go into memory and modify a line number to a value > 32767.

Looking at memory, you would find line 32767 displayed as FF 7F. By changing the high order byte from 7F to FF, we will have changed our line number to 65535.

In examining the Apple Integer Basic Token chart, you will find in many cases command names and punctuation appear multiple times. This is because different useage of a command calls for a different token. One case in point is the PRINT command, tokens 61, 62 and 63. They are defined, respectively, as: PRINT (The following character string), PRINT (The following variable's value) and PRINT (Scroll text up one line).

Similarly, in changing the line:
"120 PRINT 0, 300" to read "120 DEL 0,300", we must make two patches in memory. Not only must the 62 (PRINT) be changed to a 09 (DEL), but the 49 (,) must be changed to a 0A (,), since the DEL command requires its own comma.

To find the correct useage of a given token, just enter a Basic statement using that token (if legal) then examine it in memory. The EXAM routine in our Workshop series will do that for you automatically. Experiment. You can not damage your Apple in this manner; only your program. Patch in a NEW command as the last line of your program, run it and watch the program "self destruct".

CRIME PAYS!

MINI'APP'LES USER CONTRIBUTED PROGRAM BANK(Note one floppy disk had an error at time of writing newsletter)

```
I  002  MINI APP LES
I  043  APPLESOFT
I  024  EVIL
I  022  CREATES H8192
I  004  TEST
I  006  MATH GAMES
A  005  GRAVITY/ORBIT DEMO
I  008  RENUM/APPEND INST.
B  003  RENUM/APPEND
I  005  MUSICPIC
I  001  MARKGR
I  089  ATOM 24K
B  006  HIKES
B  002  KIN6
I  011  DISC TO DISC TRANSFER
I  026  APPLESTAND
A  036  KINGDOM
I  008  FLY
I  004  DISC SPACE HELLO
I  003  COLOR MAGIC
I  05   SPELL H16384
A  012  LUNAR LANDER
I  016  I DARE YOU
I  007  AIRCRAFT GUNNER
I  035  MATH GAMES
B  003  SPACE.DOC
I  005  DISCHELLO
A  008  HOME IMPROVEMENT
A  005  GRAVITY ORBIT
I  008  T.E.INSTRUCTIONS
B  005  TEXT EDIT
```

COPIES OF CONTACT 1, 2 & 3

For those of you who signed up for copies of the Apple Company Newsletter CONTACT, editions 1, 2 & 3, these have been reproduced and may be picked up at Computerland. Price is 60 cents for the set of 3. Anyoneelse desiring copies please contact the editor.

SOFTWARE ETC IN STOCK AT COMPUTERLAND

Computerland thought that the following items might be of interest to Apple users. All are in stock

SC Assembler- as reviewed in Feb Micro magazine.	Tape version	$25
	Disk version	$35
XPLO - High level programming language. Supplied by PJR Boyle of Colorado. Advertised elsewhere		$35
APPLESOFT in ROM		$200
PROGRAMMERS AID in ROM		$50
APPLE II MAIL - A disk based information retrieval system Supplied by SOFTWARE HOUSE Inc.		20
16K Memory Expansion		$120
IP125/225 Brighter Writer Printer		838/988
Dust covers for the Apple II- Vinyl , color co-ordinated		$9.95
Mountain Hardware Apple Clock board (1 only)		199

Computerland has just published a price list of personal computing products for Apple II and PET. These are available at the store.

page 6