VIPER

September 1984
Volume 6, Number 1

Journal of the VIP Hobby Computer Assn.

The VIPER was founded by ARESCO, Inc. in June 1978

****

VIPER's Final Issue

This issue will be the last for VIPER. A detailed explanation is in the Editorial pages. The most recent editorial material is on 6.01.01.

Older material has been left in since it had already been printed. Due to the split printing, I goofed and labeled two sets of pages as 6.01.12.

Contents

Editorial

September 20, 1984

Dear VIPHCA members:

This issue of VIPER has been a long time a-coming and I'm sorry that it has taken so long. I'm also grateful for everyone's patience-- not one of you has seen fit to take me to task for not getting VIPER into the mail. Thank you again. However, I'm sure most of you realize that VIPER has run its course. As mentioned in other parts of this issue, membership has declined to the point where it would be very difficult to keep a good newsletter going.

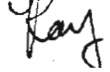
I've left most of the material for this issue intact, so some of it was written quite some time ago. And I thought I'd leave in the letters which I received at renewal time. I also received some very nice material from Mike Weigert which I rushed to include in this issue. Mike sent in a tape (VIP) of his programs and I'll be happy to send a copy to any of you who might be interested.
Just send in $2 and I'll mail out a cassette with the Operating System program and the Globe program.

I'm also going to refund $10 of the $12 dues for this year. About $1.25 will be used to prepare and mail this issue of VIPER and the other 75 cents should cover incidental expenses in winding down the operation, handling any additional correspondence, tax returns, etc. (Yes, even our friends at the IRS keep an eye on VIPHCA, to make sure that we're really a non-profit group.)

Also, if any of you are interested in back issues or copies of the VP-710 game book, now is the time to drop me a line. I expect that I'll serve as a focus point for any VIP related querries in the future, so this is as good a place as any to direct a question. Can't guarantee an answer for you, but I'll do what I can.

Thank you all once again for you kind and welcome support over the years. We have all enjoyed much and learned even more. And that will no doubt be VIPER's greatest legacy.

Yours,
Raymond C. Sills, Director, VIPHCA



Well Gang this is the start of Year 6 for VIPER. Unfortunately, I don't think that the fates have good things in store for us. Right now we have had only 21 members renew membership for this year. That's not a very large membership base for a computer organization. Especially since we used to have hundreds of members. And I'm not sure that we'll be able to make a go of it, at least not as we have in the past. I still have some un-published material here, but it would be better if there was new stuff coming in. Paul Piescik has released his Cuddly Software programs into the public domain, so we are free to distribute copies of those programs. I have most of the Cuddly programs here and I'll send you a copy of the programs and the supporting documentation if you will send me $3 to cover the cost of copying, the tape cassette and shipping. I also still have copies of the RCA VP-710 Game book and cassettes of the VP-710 programs. I also have an original copy or two of Tom Swan's entire PIPS series. If anyone is interested in that stuff, drop me a line.

Interestingly, most of the people who returned the questionaire from the last issue, felt that VIPHCA should keep going, but perhaps in conjunction with supporting another computer. And it seems that a lot of you are now owners of the Commodore 64. That's not very surprising, considering the features of that machine and its very reasonable price. But those of you who are machine language programmers and considering getting the C-64 are probably going to feel a bit cramped using the 6502 processor. There's only six registers in the 6502 and only one is a 16 bit register. On the other hand, there's a lot a support for the C-64 from the manufacturer, independent hardware/software companies, and a half dozen or so magazines. (Compute, Compute's Gazette, Commander, Ahoy, Home Computer, and Run, just to name a few. They are all very good, but Gazette is probably the largest of them. The "innards" and operating system of the C-64 is very well documented and there are no "secrets" kept by the manufacturer.

Here are the results of the questionaire: (in some cases, a respondent did not make a choice, so the totals will not be equal to the 21 returned questionaires.

Question 1. (Should VIPHCA continue) 19 Yes 0 No 2 Not sure.
Question 2. (support other computers) 9 Yes 4 No 8 Not sure.
Question 3. (which?) VIC-20, C-64, CoCo, ELF are mentioned.
Question 4. (fewer pages) 15 Yes 3 No 3 Not sure.
Question 5. (more frequent) 12 Yes 3 No 3 Not sure.
Question 6. (higher dues) 10 Yes 5 No 5 Not sure.

****

Editor's analysis

I'm pleased that almost everyone who replied felt that VIPHCA should continue. Of course, it's a "loaded" question since those who felt the other way, did not reply or renew. Looks like people see the problem in supporting the VIP, but still have a warm spot in their hearts for the VIP/1802. And most of you feel you can live with a smaller VIPER, but would like to have it more often. And I am pleasantly surprised that that many of you would even be moderately willing to pay higher dues. But relax, since no changes contemplated for now.

You might wonder what type of computer gear yours truly has---after all, I am a computer junkie by my own admission. So here's the list:

1. Radio Shack Color Computer (the original "D" version upgraded to 32K, Disk, etc.) It is being used to prepare this text, which is printed on an Okidata 82A printer.
2. Radio Shack MC-10, a gift from Santa.
3. RCA VIP (2 of them) 4K plus 2 4K RAM boards, Simple Sound, Super Sound, 4-chan Super Sound, Tiny BASIC, VP-601 ASCII KB
4. Simclair ZX-81
5. Timex TS-1000
6. Timex TS2068
7. Microace 2K
8. Commodore VIC-20 (2) with 16K RAM board, Rabbit cart, C2N cassette, modem, joystick, Cardco 3-slot, HES Mon, HES Forth, Kantronics interface, Hamsoft, Hamtext.
9. Commodore C-64, C2N cassette, 1541 Disk, Hes Forth, Magic Voice.
10. Radio Shack Videotex terminal unit, 16K - looks like a CoCo.
11. Heathkit ET-3400 Microprocessor trainer - uses Motorola 6800 uP.

I guess that's quite a list, but probably not a world's record! The only reason I mention it is in case any of you might have one or more of the above systems and wish to include it in the perview of VIPER.

It'll not be easy to find a lot of material for VIP machines and a lot of you have mentioned in your questionaires that the ol' VIP is not getting a lot of use these days. One of our members suggested that we print letters from members regarding what each of us is doing with our systems. Sort of a church newsletter for computers. There was quite a bit of that back in the early days of VIPER and it was especially important during those years when the VIP was popular. And it's still a good idea even if we brach out to encompass other computers.

I'm going to send each of you a list with the names and addresses of past VIPHCA members, in case there is someone near you whom you might like to contact about 1802 type projects.

Comments from Members' Questionaires:

I have a Commodore 64 with about 20 disks loaded with programs. I am looking for some in-depth information about the C-64. I have had my C-64 for about 6 months.
---- Frank C. Awtrey, Fayetteville, GA

****

Dear Ray:
I know what you are going thru with VIPER, it is probably very similar to the decisions I had to make regarding offering VIP add-ons. Here are som of my thoughts, for what they're worth:
I look upon VIPER more as an "1802" instructor, than a VIP magazine. I owe VIPER for most of the knowledge I have about ML programming and interfacing the 1802.
I think we can say, although it is still popular in industrial applications, that the 1802 has seen it's day. CMOS versions of CPUs with more common architecture are countering the 1802's advantages.
I would like to see VIPER adopt a newer CPU; I suggest the 6502 because it is used in so many low priced computers, and give it the same treatment as the 1802. I have probably subscribed to and let lapse ten computer magazines in the last two years. They all seem to cater to "appliance operators." I think there is a need for a journal that teaches how to make that computer do what you want.
I realize that it is easy for me to sit here and say what I like and not have to take the chances or do the work.
I hope you get good response to your questionaire and that they indicate a clear goal.
-----Jerry Krizek, West Covina, CA

****

I have a VIC-20 but still spend as much time on my first olove, the RCA VIP. I recently interfaced a Brother CE-50 with an IF-50 interface module to both the VIP and the VIC.
-----Bill Fisher, Armonk, NY

****

Your comments reflect the view that the VIP owners are using these machines as mainframes for serious office or business use. We are a hobby club. This is just like a sport. The people who are interested are here for the fun. (You bet! RS) We have to stir new interest and find solid ground. Editorial comments column might help.
----- Bruce Konek, Walnutport, PA

****

More Comments:

I really have mixed emotions. I'm loyal to the community, but time does go on. I now have a Kaypro, H-89, TRS-80 Model 100, Sharp 1500 & 1500A, so I really don't "hack" any more with VIP COSMAC. If Tom Swan got interested in new field, I'd go with that!--he's good!

-----Glen Merritt, Mt. Laurel, NJ

(Ed. note: Some of Tom Swan's main interests these days are in the field of languages, mainly Pascal.
Tom has written several fine books with business programs for Pascal users, published by Hayden.)

****

Hi Ray-

I think one possibility is to encourage articles of a general nature that would apply to all computers. Perhaps we could generate a response with press releases to magazines. I'd be in favor of a marketing effort to tell people about our Hobby Computer club. I'm sure we could attract members.

Rather than pick one or two computers, whether PCjr., Apple, whatever -- let the mebers write about what they want. In other words, let the membership decide by voting with their contributions, the future direction of VIPHCA. You said it yourself-- because of the fluctuations in the micro business, systems will come and go. But just because we have different computers doesn't mean we don't have common interests!

And I mean this sincerely, although I have been pleased to see PIPS IV finally in print, I do not think the VIPER should be a Tom Swan publication. I'm getting tired of seeing (?) my own voice. Let's see contributions from everybody!

-----Tom Swan, Columbia, MD

****

Probably should start articles on how to tie in the VIP to other comouters: Atari, Apple, Commodore, etc. Also discuss machine codes of these various machines. (Most magazines manly deal with BASIC programs.) --and how to write games, etc. One idea is to have the Atari printer list out VIP programs for hard copy, or store VIP programs on the Atari disk drive.

-----Robert Casey, Oradell, NJ

****

On the VIPER, it is obvious that the VIP is on it's way out. You really have few choices. Either throw in the towel or expand to other systems. Hopefully keep publishing some 1802 related info.

The two articles on the VIP that I wrote for 73 magazine have never been published. They paid for them, so I can('t give them to you, but I suggest that you write to 73 magazine and request permission to publish in VIPER. You can point out that you can't affort to pay for them, but you could offer a free full page ad in each issue which an article is published.

-----George Gadbois, Lancaster, PA

And still more comments:
Articles published to date have been a big educational assist, inspirational, --- and fun!
-----Gene Landes, Des Plaines, IL

****

As I have 2 Sinclair ZX80s and a newly acquired 6809 Radio Shack CC II, I'd welcome the broadened base. I still maintain a Studio II with memory and have tought myself 1802 machine language with the help of Tom Swan books. Am still interested!
-----Tom Keene, Austin, TX

****

I appreciate the good work. Hope you can continue.
-----William Perry, Mission Viejo, CA

****

For my interests, the personal computers like IBM, Apple, Commodore and Heath are better. The VIP served a good educational purpose. I am lukewarm on supporting a discontinued product.
-----C. Spencer Powell, Evanston, IL

****

I have a Sinclair Timex 1000 and a 64K TRS-80 Color Computer. The VIP is getting little attention. I have used it recently to burn an EPROM, that's all. I'm trying to use K6AEP software and RTM circuit boards to interfact the CoCo for SSTV and FAX. My main Z80 microcomputer (Digital Group) has been retired.
-----Al Crisson, Lewes, DE

EDITOR'S REMARKS:

This is a pretty good sampling of opinion from the membership. It appears that most of you would like to have VIPER continue, even at a reduced or changed format and moving on to support other computers. So how about some more ideas and progress reports from you guys out there and I'll print them here in the VIPER!

73,

Ray

We had some classified ads requested by members after VIPER 5.06 went out, and I listing them here, but I think it only fair to return to those members the payment they made for the ads, since we have so few current members.

FOR SALE
****

2	14EPM bare board w/data	$15
1	8K RAM bare board w/data	20
1	8K RAM assemb/tested	60
1	VP-550 Super Sound w/book	30
6	VP-701 F.P. BASIC manual for 14EPM board	2

All prices include shipping. CA residents add tax.

G.J. Krezek
722 N. Morada Ave.
West Covina, CA 91790

FOR SALE
****

Know anyone who wants a VIP? I'm selling a 4K VIP with ASCII Keyboard, Simple and Super Sound boards, homemade color board, Tiny BASIC ROM Board, 2-card exoander board, power supply, cassette player, plenty of games and programs on cassette, manuals with documentation, and three years of VIPER newsletter. Over $550 value, selling for $290 or any reasonable offer. Send check to:

Nathan Gooen
168 Pond St.
Sharon, MA 02067
phone: 617-784-2771

This is a true CHIP-8 Assembler. It will run on any machine with Microsoft BASIC and a disk drive. You may need to make slight modifications, however. This version will run directly on a H-89 with MBASIC and one drive, or on a TRS-80 Model I or III with one disk. You could probably even make it run on Full 1802 Basic with a few mods.

The program reads in the file for the first pass, and generates the symbol table and addresses. The second pass does the actual opcode decoding and file output. The opecodes and pseudo-opcodes are detailed in Figure 1-1. In addition to using the predefined opcodes, you may also generate your own constant (ie, no variable inclusion) opcodes as labels. For example:

```
TVON:   EQU     ØØ4B    ;TV On instruction
START:  TVON            ;Turn TV on
```

Byte definitions using the BYTE pseudo always generate one word of code. In other words, BYTE 5 will generate ØØØ5. This ensures that the CHIP-8 count never becomes odd or incorrect.

Labels occur only in the first eight characters of the line. They always end with a colon, and always need one space or tab after the colon to delimit the opcode. Simple arithmetic tells us, then, that labels may be up to six characters in length. Labels may be used wherever addresses or data is desired. If only one byte of data is needed, the LSD of the label's address will be used. In the case of an address, labels may be substituted in all cases. For example,

```
START:  LET     I=VARS
        LOAD    V1
HALT:   GOTO    HALT

VARS:   BYTE    8Ø53
```

RCA's comments, with slight modifications, will work for input to the assembler. You may wish to define the data blocks as labels on even-numbered addresses for easy reference.

Semi-colons are used as comment delimiters, as is standard with any assembler in existance except RCA's. If you want, you may change all the semicolons to double periods.

The variables used in the program are:

LO	current location counter
F	Flags whether data bytes were printed
IN$	current input string
A$	first argument
B$	second argument
SYM$()	symbol table
LO()	location table

If you have any questions or comments, please send them to VIPER or to me. My address is: William Lindley, 21 Hancock St., Bedford, MA Ø173Ø.

CHIP-8 ASSEMBLER

Format:

<label> opcode <operand> <comment>

Opcodes:

May be any one of the following opcodes and Pseudo-ops.

opcode	operand	assembles to
GOTO	<addr>	1MMM
GOSUB	<addr>	2MMM
RET		ØØEE
CALL	<addr>	ØMMM
SKEQ	<var>,<kk>	3XKK
SKEQ	<var>,<var>	5XYØ
SKEQ	<var>,KEY	EX9E
SKNE	<var>,<kk>	4XKK
SKNE	<var>,<var>	9XYØ
SKNE	<var>,KEY	EX9E
LET	<var>=kk	6XKK
LET	<var>=TIME	FX07
LET	TIME=<var>	FX15
LET	TONE=<var>	FX18
LET	<var>=KEY	FXØA
LET	I=<addr>	AMMM
RAND	<var>,kk	CXKK
ADD	<var>,kk	7XKK
ADD	I,<var>	FX1E
ADD	<var>,<var>	8XY4
SUB	<var>,<var>	8XY5
COPY	<varX>,<varY>	8XYØ
OR	<var>,<var>	8XY1
AND	<var>,<var>	8XY2
XOR	<var>,<var>	8XY3
SHR	<var>,<var>	8XY6
SHL	<var>,<var>	8XYE
PATT	<var>	FX29
DPAT	<var>	FX33
SHOW	N @ VxVy	DXYN
ERASE		ØØEØ
LOAD	<var>	FX65
SAVE	<var>	FX55

Pseudo-Opcodes

BYTE <four nybbles> NNNN

EQU Equates symbol with location specified.
Examples:

```
DPAGE:  EQU  Ø3ØØ                DPAGE=Ø3ØØ
SAME:   EQU  DPAGE               SAME=Ø3ØØ
```

ORG continues to assemble source code at a different address.

Arguments

<var> is a variable name.
Examples: V1,V2

<addr> is a hex location or s symbol
Examples:

```
        GOTO    DPAGE
        GOTO    Ø1ØØ
```

<kk> is a data byte
Example:

```
        SKEQ    V1,5F
```

A symbol may also be used. In this case, only the lower byte of the label's address is used. Example:

```
        SKEQ    V1,DPAGE
```

will skip if V1 equals the LSD of DPAGE.

```
10 REM                        CHIP-8 ASSEMBLER      by  William Lindley
15 CLEAR 5000
20 PRINT "Input Filename ->";:LINE INPUT F1$
30 PRINT "Output Filename ->";:LINE INPUT F2$
35 DIM SYM$(50),LO(50):LO=512:SY=0
40 REM PASS ONE FOR FILE
45 OPEN "I",1,F1$
50 IF EOF(1) THEN 1000:REM Second Pass for File
55 LINE INPUT #1,IN$
60 IF LEFT$(IN$,1)=";" OR IN$="" THEN 50
65 IF INSTR(IN$,";") THEN IN$=LEFT$(IN$,INSTR(IN$,";"))
66 IF INSTR(IN$,"EQU") THEN 200
67 IF INSTR(IN$,"ORG") THEN 300
70 IF INSTR(IN$,":") THEN GOSUB 100
75 N=INSTR(IN$,CHR$(32)):IF N THEN IN$=LEFT$(IN$,N-1)+MID$(IN$,N+1):GOTO 75
77 N=INSTR(IN$,CHR$(9)):IF N THEN IN$=LEFT$(IN$,N-1)+MID$(IN$,N+1):GOTO 75
78 IF IN$="" THEN 50
80 LO=LO+2:GOTO 50
100 REM PUT SYMBOL IN TABLE
110 A$=LEFT$(IN$,INSTR(IN$,";")-1)
120 SYM=SYM+1:SYM$(SYM)=A$:LO(SYM)=LO:RETURN
200 REM EQUATE HANDLING
205 IF INSTR(IN$,";") THEN IN$=LEFT$(IN$,INSTR(IN$,";")-1)
210 N=INSTR(IN$,"EQU"):A$=LEFT$(IN$,N-1):B$=MID$(IN$,N+4)
220 IF LEFT$(B$,1)="*" THEN AD=LO:GOTO 240
230 HX$=B$:GOSUB 10000:AD=HX:REM CONVERT TO DECIMAL
240 SYM=SYM+1:SYM$(SYM)=A$:LO(SYM)=AD:GOTO 50
300 REM ORIGIN HANDLING
310 N=INSTR(IN$,"ORG"):B$=MID$(IN$,N+4)
315 Q=INSTR(B$,CHR$(32)):Q1=INSTR(B$,CHR$(9)):IF Q OR Q1 THEN B$=LEFT$(B$,Q+Q1-1
320 HX$=B$:GOSUB 10000:LO=HX:PRINT HX$,B$,HX:GOTO 50
1000 CLOSE:OPEN "I",1,F1$:OPEN "O",2,F2$:LO=512
1005 PRINT "Symbol Table:":PRINT
1010 FOR X=1 TO SYM:PRINT SYM$(X),HEX$(LO(X)):NEXT:PRINT
1020 PRINT "-------------- Second Pass --------------"
1030 IF EOF(1) THEN CLOSE:END ELSE LINE INPUT #1,IN$
1035 IF F=0 THEN N$=HEX$(LO):N$=STRING$(4-LEN(N$),"0")+N$:PRINT #2,N$,;:F=-1
1040 IF LEFT$(IN$,1)=CHR$(32) THEN IN$=MID$(IN$,2):GOTO 1040
1045 IF LEFT$(IN$,1)=CHR$(9) THEN IN$=MID$(IN$,2):GOTO 1045
1046 N=INSTR(IN$,";"):IF N>2 THEN IN$=LEFT$(IN$,N-2) ELSE IF N>0 THEN 1030
1050 N=INSTR(IN$,":"):IF N>0 AND N<8 THEN IN$=MID$(IN$,N+1):GOTO 1040
1053 IF IN$="" THEN 1030
1055 IF LEFT$(IN$,3)="ORG" THEN HX$=MID$(IN$,5):GOSUB 10000:LO=HX:GOTO 1030
1060 IF LEFT$(IN$,3)="EQU" THEN 1030
1065 IF LEFT$(IN$,4)="BYTE" THEN 3200
1070 AP=INSTR(IN$,CHR$(32)):IF AP=0 THEN AP=INSTR(IN$,CHR$(9))
1075 F=0
1080 RESTORE:FOR X=1 TO 22:READ Q$:IF LEFT$(IN$,LEN(Q$))=Q$ THEN 1100 ELSE NEXT
1090 FOR X=1 TO SYM:IF LEFT$(IN$,LEN(SYM$(X)))=IN$ THEN 3300 ELSE NEXT
1095 PRINT "? Illegal Opcode: "IN$:F=-1:GOTO 1030
1100 ON X GOTO 2000,2500,3000,3100,3500,4000,4500,5000,5100,6000,6100
1110 X=X-12
1120 ON X GOTO 6200,6300,6400,6500,6700,6800,7000,6900,7500,8000
2000 REM GOTO HANDLING
2010 HX$=MID$(IN$,6):GOSUB 10000
2020 O$="1"+MID$(HX$,2):PRINT #2,O$:LO=LO+2:GOTO 1030
2500 REM GOSUB HANDLING
2510 HX$=MID$(IN$,7):GOSUB 10000
2520 O$="2"+RIGHT$(HX$,3):PRINT #2,O$:LO=LO+1:GOTO 1030 ; REM  Line 3000 is next
```

6.01.11

```
3000 REM RETURN HANDLING
3010 PRINT #2,"00EE":LO=LO+1:GOTO 1030
3100 REM MACHING LANGUAGE CALL HANDLING
3110 HX$=MID$(IN$,6):GOSUB 10000
3120 PRINT #2,"0";MID$(HX$,2):LO=LO+2:GOTO 1030
3200 REM BYTE HANDLING
3210 F=0:HX$=MID$(IN$,6):GOSUB 10000
3220 PRINT #2,HX$:LO=LO+2:GOTO 1030
3300 N$=HEX$(LO(X)):N$=STRING$(4-LEN(N$),"0")+N$:PRINT #2,N$:LO=LO+2:GOTO 1030
3500 REM SKIP IF EQUAL HANDLING
3510 B$=MID$(IN$,6):IF LEFT$(B$,1)<>"V" THEN ER=2:GOSUB 9000:GOTO 1030
3520 V$=MID$(B$,2,1):IN$=MID$(B$,4)
3530 IF IN$="KEY" THEN PRINT #2,"E"V$"9E":LO=LO+2:GOTO 1030
3540 IF LEFT$(IN$,1)="V" THEN PRINT #2,"5"V$MID$(IN$,2)"0":LO=LO+2:GOTO 1030
3545 IF LEN(IN$)<2 THEN IN$="0"+IN$
3550 PRINT #2,"3"V$IN$:LO=LO+2:GOTO 1030
4000 REM SKIP IF NOT EQUAL
4010 B$=MID$(IN$,6):IF LEFT$(B$,1)<>"V" THEN ER=2:GOSUB 9000:LO=LO+2:GOTO 1030
4020 V$=MID$(B$,2,1):IN$=MID$(B$,4)
4030 IF IN$="KEY" THEN PRINT #2,"E"V$"A1":LO=LO+2:GOTO 1030
4040 IF LEFT$(IN$,1)="V" THEN PRINT #2,"9"V$MID$(IN$,2)"0":LO=LO+2:GOTO 1030
4045 IF LEN(IN$)<2 THEN IN$="0"+IN$
4050 PRINT #2,"4"V$IN$:LO=LO+2:GOTO 1030
4500 REM LET HANDLING
4505 IN$=MID$(IN$,5)
4510 N=INSTR(IN$,"="):IF N=0 THEN ER=2:GOSUB 9000:LO=LO+2:GOTO 1030
4520 A$=LEFT$(IN$,N-1):B$=MID$(IN$,N+1)
4530 IF A$="TONE" THEN PRINT #2,"F"MID$(B$,2)"18":LO=LO+2:GOTO 1030
4540 IF A$="TIME" THEN PRINT #2,"F"MID$(B$,2)"15":LO=LO+2:GOTO 1030
4550 IF B$="TIME" THEN PRINT #2,"F"MID$(A$,2)"07":LO=LO+2:GOTO 1030
4560 IF A$="I" THEN HX$=B$:GOSUB 10000:PRINT #2,"A"RIGHT$(HX$,3):LO=LO+2:GOTO 1030
4570 HX$=B$:GOSUB 10000:PRINT #2,"6"MID$(A$,2)LEFT$(HX$,2):LO=LO+1:GOTO 1030
5000 REM RAND HANDLING
5005 IN$=MID$(IN$,6)
5010 N=INSTR(IN$,","):IF N=0 THEN ER=2:GOSUB 9000:LO=LO+1:GOTO 1030
5020 A$=LEFT$(IN$,N-1):B$=MID$(IN$,N+1)
5030 HX$=B$:GOSUB 10000:PRINT #2,"C"MID$(A$,2,1)RIGHT$(HX$,2):LO=LO+1:GOTO 1030
5100 REM ADD HANDLING
5110 IN$=MID$(IN$,5):N=INSTR(IN$,","):A$=LEFT$(IN$,N-1):B$=MID$(IN$,N+1)
5115 IF A$="I" THEN PRINT #2,"F"MID$(B$,1,1)"1E":LO=LO+1:GOTO 1030
5120 IF LEFT$(B$,1)="V" THEN PRINT #2,"8"MID$(A$,2,1)MID$(B$,2,1)"4":GOTO 5140
5130 HX$=B$:GOSUB 10000:PRINT #2,"7"MID$(A$,2,1)RIGHT$(HX$,2)
5140 LO=LO+2:GOTO 1030
6000 REM SUB
6010 IN$=MID$(IN$,5):N=INSTR(IN$,","):A$=LEFT$(IN$,N-1):B$=MID$(IN$,N+1)
6020 PRINT #2,"8"MID$(A$,2,1)MID$(B$,2,1)"5":LO=LO+2:GOTO 1030
6100 REM COPY
6110 IN$=MID$(IN$,6):N=INSTR(IN$,","):A$=LEFT$(IN$,N-1):B$=MID$(IN$,N+1)
6120 PRINT #2,"8"MID$(A$,2,1)MID$(B$,2,1)"0":LO=LO+2:GOTO 1030
6200 REM OR
6210 IN$=MID$(IN$,5):N=INSTR(IN$,","):A$=LEFT$(IN$,N-1):B$=MID$(IN$,N+1)
6220 PRINT #2,"8"MID$(A$,2,1)MID$(B$,2,1)"1":LO=LO+2:GOTO 1030
6300 REM AND
6310 IN$=MID$(IN$,5):N=INSTR(IN$,","):A$=LEFT$(IN$,N-1):B$=MID$(IN$,N+1)
6320 PRINT #2,"8"MID$(A$,2,1)MID$(B$,2,1)"2":LO=LO+2:GOTO 1030
    :REM Line 6400 is next
```

6.01.12

```
 6400 REM XOR
 6410 IN$=MID$(IN$,5):N=INSTR(IN$,","):A$=LEFT$(IN$,N-1):B$=MID$(IN$,N+1)
 6420 PRINT #2,"8"MID$(A$,2,1)MID$(B$,2,1)"3":LO=LO+2:GOTO 1030
 6500 REM SHR
 6510 IN$=MID$(IN$,5):N=INSTR(IN$,","):A$=LEFT$(IN$,N-1):B$=MID$(IN$,N+1)
 6520 PRINT #2,"8"MID$(A$,2,1)MID$(B$,2,1)"6":LO=LO+2:GOTO 1030
 6600 REM SHL
 6610 IN$=MID$(IN$,5):N=INSTR(IN$,","):A$=LEFT$(IN$,N-1):B$=MID$(IN$,N+1)
 6620 PRINT #2,"8"MID$(A$,2,1)MID$(B$,2,1)"E":LO=LO+2:GOTO 1030
 6700 REM PATT
 6710 IN$=MID$(IN$,6)
 6720 PRINT #2,"F"MID$(IN$,2,1)"29":LO=LO+2:GOTO 1030
 6800 REM DPAT
 6810 IN$=MID$(IN$,6)
 6820 PRINT #2,"F"MID$(IN$,2,1)"33":LO=LO+2:GOTO 1030
 6900 REM ERASE
 6910 PRINT #2,"00E0":LO=LO+2:GOTO 1030
 7000 REM DISP
 7010 IN$=MID$(IN$,5):N=INSTR(IN$,CHR$(64)):A$=LEFT$(IN$,N-1):B$=MID$(IN$,N+1
 7020 PRINT #2,"D"MID$(A$,2,1)MID$(B$,4,1)MID$(B$,2,1):LO=LO+2:GOTO 1030
 7500 REM LOAD
 7510 IN$=MID$(IN$,6)
 7520 PRINT #2,"F"MID$(IN$,2,1)"65":LO=LO+2:GOTO 1030
 8000 REM SAVE
 8010 IN$=MID$(IN$,6)
 8020 PRINT #2,"F"MID$(IN$,2,1)"55":LO=LO+2:GOTO 1030
 9000 PRINT #2,"ERROR "ER:RETURN
10000 REM CONVERT HEXADECIMAL TO DECIMAL
10010 FOR X=1 TO SYM:IF HX$=SYM$(X) THEN HX=LO(X):GOTO 10045 ELSE NEXT
10020 HX=0:FOR X=1 TO LEN(HX$)
10030 B$=LEFT$(HX$,1):IF B$<"A" THEN H1=VAL(B$) ELSE H1=ASC(B$)-55
10040 HX=HX*16+H1:IF LEN(HX$)>1 THEN HX$=MID$(HX$,2):NEXT
10045 HX$=HEX$(HX)
10050 IF LEN(HX$)<>4 THEN HX$="0"+HX$:GOTO 10050 ELSE RETURN
19000 DATA "GOTO","GOSUB","RET","CALL"
19010 DATA "SKEQ","SKNE","LET","RAND","ADD","SUB","COPY","OR"
19020 DATA "AND","XOR","SHR","SHL","PATT","DPAT","SHOW","ERASE"
19030 DATA "LOAD","SAVE"
19040 END
```

6.01.13

Dear VIPER:

I am a reader of your magizine ever since it came out. I owned a VIP for about four years now and have most of the publications on it. I have written quite a few machine language 0rograms for the VIP which I would like to share with you and your readers.

First I would like to share some little tid-bits of information that I have discovered. One of the things that I discovered before it was published in your magizine was the 2N instruction can be used to stop your program instead of the short branch to itself. I am not sure if this was published but I accidently discovered that a D2 located at 0000 is excellent for entering programs. This prevents any program from being executed until the D2 is removed. The D2 causes the VIP to return to the operating system as if key C was pressed. This will stop you from losing pieces of programs that you entered to one little slipped finger when you did not press key C. The author once lost an E page Chip-8 program that was typed in but not checked for errors to an accidental flip of the switch which took hours to enter (such a large mistake will not happen again !!!. It only takes one time like this to learn what not to do.). There was an article discussing how to stop the display interference caused by three cycle instructions when the video was turned on. Some programers suggested to turn off the video but when you do not want the veiwer to notice the video to disappear this is not the solution. Tom Swan suggested that you would initilize another register as the program counter and use it to get around the problem of long branches. Still another method is to use Standard Call and Return but if you never plan to return to where you called from this may present some problems. I discovered another method that you can use that does not require the use of a sometimes unavailible register. I call it the Sneeky Long Branch (SNLBR). The SNLBR is based on a tricky manipulation of the program counter. If you want to make a long branch somewhere using only two cycle instructions here is one other way to do it. First load the high page in D (the 1802 accumulator) then short branch to the desired low byte minus one and put the contents of D in the high part of the program counter. It looks something like this:

```
Somewhere on page YY:          Please note that YY, XX, and ZZ are
        F8      LDI            arbitrary bytes.
        XX      page that you are branching to
        30      BR
        ZZ      destination on page XX minus one

Adress
YYZZ    BN      PHI -- N is register P and after the instruction is exe-
                cuted the next instruction is executed at XX(ZZ+1)
```

This method also has its draw backs in that YYZZ may not be a free location. You may also want to preceede the BN instruction by a 38 (SKP). There is one other method which workes some of the time but unfortunately not always which is to do a long branch and immediately follow it with a C4 (NOP). The trick is to execute this at the right time after a vidio interrupt so that the disturbance in the sink does not show up on the screen.

Now to write about the programs that I promised. I have written three machine language programs which are Debug, Globe II, and Display Type Out. I have written revisions to Basic so that we could store it on EPROMS with serial output routines to a teletype which can be called by one key press from the operating system without loading a boot. I modified the operating system to accept a byte for the number of pages in the tape read and write routines which makes it possible to manipulate as much as FF pages. The operating system is also equipped with a move data routine which can move memory anywhere. I also wrote input output routines for Tom Swan's Assembler Disassembler the input is from a parallel keyboard and the output is a teletype. I will provide listings for the operating system (the operating system also includes UT4 which we mainly use for memory type outs such as in this article), the Basic input and output routines for a teletype, Globe II, and Display Type Out. The listings follow.

Thank You,

Michael Weigert

Michael Weigert

Anybody that has any questions can write to:

United States Naval Academy 35th Company
Annapolis, Maryland 21412

I give VIPER the rite to publish any or all of the information enclosed in this letter.

Michael Weigert

Michael Weigert

6.01.13

The operating system has many features. It still does everything that the normal operating system does but also has the debugging program incorporated in it, UT4, Move Data, Basic, and a Go To Routine as well. The instructions for using it are key C gets you into the normal operating system where you have the options of Tape Read, Tape Write, Memory Read, Memory Write, and an added function Go To (key C instead of keys B, F, A, and 0) which will execute a machine language program with R(0) as the program counter at the adress keyed in. When you flip the switch with 4 pressed you enter UT4, with key E pressed (E is a mnemonic for Error) you enter the Debug program, with key B pressed you enter Basic, and with key D pressed you enter the Move Data routine.

The operating system on our machine is located at F000-F1FF. The added features discribbed above are located starting at F200. The Debug program is located at F600-FA16 and UT4 is located at F400-F5FF. My dad rewired the VIP with the help of his co-workers at RCA. If there are any questions about our hardware my dad would be glad to answer them. The adress is: Here is a listing of our operating system.

Fred Weigert
59 West Patricia Road
Holland, Pa. 18966

```
?1F000 200
F000  F8F0 B2F8 03A2 E2D2 C0F2 9999 F8FF A1F3;
F010  EFB1 F8AA 5101 FBAA 3222 9199 043B 22B1;
F020  3012 36FC C0F2 00FF E1F3 0073 81FB AF3A;
F030  299F8 D273 F399F 5181 A091 B0F8 CFA1 D073;
F040  2020 40FF 0120 50FB 323A 3E92 B3F8 51A3;
F050  D390 B23B BDF8 F1B1 B4B5 B7BA BCF8 46A1;
F060  F8AF A2F8 DDA4 F3C6 A5F3 BAA7 F8A1 ACC0;
F070  F21A D7D7 D7B6 D7D7 D7A6 D4DC BE32 F4FB;
F080  0A32 EFC0 F27D 6199E FB0B 32C2 99EFB 0F3A;
F090  8FF8 6FAC F340 B993 F6DC 2999 3A97 F810;
F0A0  A7F8 03A9 46B7 93FE DC36 3AAD 2E97 F6B7;
F0B0  DC29 8393A AD17 87F6 DC3E 3A9E DC69 26D4;
F0C0  30C0 F833 ACF8 0AB9 DC33 C529 9993A C3DC;
F0D0  3BCF F309 A9A7 9776 B729 DC89 3AD6 87F6;
F0E0  33E3 7B97 5616 8163A CF2E 8E3A CF30 BDDC;
F0F0  16D4 30EF D7D7 D756 D416 30F4 620D 3028;

F100  3039 222A 3E20 2434 2623 2E18 141C 1012;
F110  F030 F030 F030 8030 F050 7050 F050 5050;
F120  F030 F010 F030 F093 F093 F010 F010 F090;
F130  F090 9090 F010 1010 1060 2020 2070 A0A0;
F140  F020 207A 4270 2273 2252 C419 F300 A09B;
F150  B0E2 E280 E2E2 20A0 E220 A0E2 20A0 3C53;
F160  9332 67AB 2B3B B333 3243 7B23 3044 D3F3;
F170  0A3B 76F3 2017 7BBF FF01 3A73 396E 7A9F;
F180  3078 D3F3 103D 853D 8FFF 013A 3717 9CFE;
F190  3590 3082 D3E2 9CAF 2F22 8F52 62E2 E23E;
F1A0  98F3 04A3 8331A A4F3 04A3 36A7 3831 AA3F;
F1B0  FA0F 5230 9400 0000 00D3 DCFE FEFE FEAE;
F1C0  DC8E F130 B9D4 AA0A AAF3 05AF 4A5D 3DFC;
F1D0  03AD 2F3F 3ACC 8DFC D9AD 30C5 D322 0673;
F1E0  8673 9652 F306 AEF3 D8AD 02F6 F6F6 F6D5;
F1F0  42FA 0FD5 8EF6 AE32 DC3B EA1D 1D30 EA01
*
```

6.01.14

```
?MF200 300
F200  620B C4C4 3EAC 90A0 F810 B0E0 D062 0DC4;
F210  C436 1790 A0E0 D0C0 F028 3626 E362 0CE2;
F220  6ADC F3F0 3071 E26A DCD7 D7D7 B6D7 D7D7;
F230  A6D4 9673 8673 D7D7 D7B6 D7D7 D7A6 D496;
F240  7386 73D7 D7D7 B6D7 D7D7 A6D4 F3DC A49B;
F250  B4F8 0054 1484 3262 FA07 3A51 84FA 04A4;
F260  3051 1272 A472 B472 A5F0 B536 3276 2645;
F270  33B3 5414 306B 963A 6EF8 04A3 23C4 9EFB;
F280  0C3A 9196 B036 A0E3 6100 92B1 F3FF A1E0;
F290  D0D7 D7D7 AE22 C0F0 8664 0062 04C4 C43E;
F2A0  A790 A0F3 F4B0 D062 0CC0 F00C 620E C4C4;
F2B0  3E0D 900A 0 F8F6 B0D0 FFFF FFFF FFFF FFFF;
F2C0  FFFF FFFF FFFF FFFF FFFF FFFF FFFF FFFF;
F2D0  FFFF FFFF FFFF FFFF FFFF FFFF FFFF FFFF;
F2E0  FFFF FFFF FFFF FFFF FFFF FFFF FFFF FFFF;
F2F0  FFFF FFFF FFFF FFFF FFFF FFFF FFFF FFFF;

F300  FFFF FFFF FFFF FFFF FFFF FFFF FFFF FFFF;
F310  FFFF FFFF FFFF FFFF FFFF FFFF FFFF FFFF;
F320  FFFF FFFF FFFF FFFF FFFF FFFF FFFF FFFF;
F330  FFFF FFFF FFFF FFFF FFFF FFFF FFFF FFFF;
F340  FFFF FFFF FFFF FFFF FFFF FFFF FFFF FFFF;
F350  FFFF FFFF FFFF FFFF FFFF FFFF FFFF FFFF;
F360  FFFF FFFF FFFF FFFF FFFF FFFF FFFF FFFF;
F370  FFFF FFFF FFFF FFFF FFFF FFFF FFFF FFFF;
F380  FFFF FFFF FFFF FFFF FFFF FFFF FFFF FFFF;
F390  FFFF FFFF FFFF FFFF FFFF FFFF FFFF FFFF;
F3A0  FFFF FFFF FFFF FFFF FFFF FFFF FFFF FFFF;
F3B0  FFFF FFFF FFFF FFFF FFFF FFFF FFFF FFFF;
F3C0  FFFF FFFF FFFF FFFF FFFF FFFF FFFF FFFF;
F3D0  FFFF FFFF FFFF FFFF FFFF FFFF FFFF FFFF;
F3E0  FFFF FFFF FFFF FFFF FFFF FFFF FFFF FFFF;
F3F0  FFFF FFFF FFFF FFFF FFFF FFFF FFFF FFFF;

F400  C4F3 F4B0 3029 803A 7ED3 0A30 7EFF FFFF;
F410  F3EF ACF3 F4BC DC12 46BF 3221 D4F5 A430;
F420  18D5 A3D3 D320 303C FF90 B5B3 F830 A5D5;
F430  E571 5564 00F8 FEA3 D3F8 9CA3 D38D D30A;
F440  D32A F800 ADBD F83B A3D3 FB24 32D6 FB05;
F450  A1CE FB1E 3A42 D3FB 4D3A CAD3 3B5B D333;
F460  5EFB 203A CA9D B08D A081 32B4 F800 ADBD;
F470  D333 70FB 0D3A CAF8 9CA3 8DA1 9DB1 D30A;
F480  900F F8AE A3D3 800F F8AE 3022 D320 400F;
F490  F8AE A3D3 2181 3A9B 9132 3980 FA0F 3AA6;
F4A0  D33B D33D 3006 F633 8E30 8CD3 3BAB D33B;
F4B0  CA3D 5010 D333 AEFB 0D32 39FB 2132 ABFB;
F4C0  173A B4D3 FB0D 3AC3 305B F89C A3D3 8DC0;
F4D0  F5F3 FFFF FFFF D3FB 503A CAD3 33DB FB0D;
F4E0  3ACA 9DB0 8DA0 F89C A3D3 0AE5 7000 D39E;
F4F0  F6AE 2E43 FF01 3AF4 8E32 EE23 30F2 93BC
*
```

6.01.15

```
?MF500 30J
F500  F800 AEAF F8EF AC37 0738 09F8 03FF 013A;
F510  0D8F 3A17 3719 1F37 1E1E F807 300D 2E2E;
F520  8EF9 01BE DC0C 3F2C 9EFA FEBE DC26 D5FC;
F530  0733 37FC 0A33 87FC 009F D5F8 0038 83C3;
F540  F801 AFF8 80BF E38F F63B 4D67 803F 4D37;
F550  4FDC 0237 4F8F F63B 5B67 40E2 C49E F633;
F560  6837 667B 3068 7AC4 DC07 C4C4 9FF6 BF33;
F570  73F9 303F 5BBF 305D 7A32 438F 3A39 9FFF;
F580  413B 2FFF 0633 37FE FEFE FEFC 08FE AE3D;
F590  7EAD 9D7E BD8E FE3A 8E30 39FF DC17 33D5;
F5A0  4538 4638 9FAE FB0A 30BF F88B 30C1 9FF6;
F5B0  F6F6 F6FC F63B B9FC 07FF C6AE F81B C8F8;
F5C0  0BAF 7B8E ADDC 072F F58D 76AD 33D1 7B30;
F5D0  D37A C43F FA0F C4C4 3AC5 8FFC FBAF 3B9F;
F5E0  FF1B 329F 3BEA F800 30F5 9FFA 0FFC F63B;
F5F0  F3FC 07FF C6AE 30C2 D30A D33F C0F4 39FF;

F600  9083 B4B5 F899 A3D3 B683 A646 B346 A3D3;
F610  9673 8673 9330 0883 86A3 6072 A6F0 B6D3;
F620  9630 17F3 0BAF D4F5 EAD5 F330 ADAB 3F2E;
F630  6B37 31FA 7FBF FB0D 3263 D4F5 A483 ADD4;
F640  F57E 3DAB 332E D4F4 103F 0030 2EFC 00C3;
F650  FF00 09AA C7FC 0159 89F9 10A9 09BA 7C00;
F660  5990A BFD5 FC00 C8FF 0009 AACF FF01 5989;
F670  F910 A909 BA7F 0059 D583 FE33 8D98 FA0F;
F680  A9D4 F650 88FA 4083 8CD4 F5AE D5D4 F62A;
F690  D4F4 102F 0083 BFD5 E291 B9B2 8182 F810;
F6A0  A4F8 20A5 F300 B8A8 A959 F910 A988 5988;
F6B0  01A9 8159 F811 A991 FF01 5993 F4BC F8EF;
F6C0  ACF8 62BE D4F4 100D 0A00 F800 AAD4 F410;
F6D0  5223 0083 BFD4 F623 D4F4 1029 3D00 8AF9;
F6E0  10A9 09BF D4F5 AE39 FA0F A909 BFD4 F5AE;
F6F0  1A3A FA03 32FF D4F4 1020 2020 0030 CDD4;

F700  F410 0D0A 0083 FA0F CAF6 CDD4 F410 503D;
F710  0093 FA0F BFA9 D4F6 23D4 F410 2020 204D;
F720  2352 2350 2929 3D03 D4F6 4DD4 F5AE D4F4;
F730  1020 2020 533D 0093 F6F6 F6F6 BFA9 D4F6;
F740  23D4 F410 2020 204D 2352 2358 2929 3D00;
F750  D4F6 4DD4 F5AE D4F4 1020 2020 443D 0097;
F760  BFD4 F5AE D4F4 1020 2023 543D 009B BFD4;
F770  F5AE D4F4 100D 0A44 463D 0083 F6BF D4F6;
F780  23D4 F410 2020 2051 3D00 88FA 01BF D4F6;
F790  23D4 F410 2020 2049 453D 0083 FEFE FE7E;
F7A0  FB01 BFD4 F623 D4F4 100D 0A00 88FA 3FA3;
F7B0  3FB0 6B37 B3FB 8632 C2FB 40CA F805 88F9;
F7C0  40A3 D4F4 1046 4948 4420 00D4 F63D 9FBD;
F7D0  88FA 403A DBD4 F410 0D0A 00D4 F834 8752;
F7E0  9DF3 3AF2 D4F4 100D 0A46 4F55 4E44 00C0;
F7F0  F6C4 6BFB C23A D0D4 F410 0D0A 4252 4541
*
```

6.01.16

```
?MF800 300
F800  4B00 C0F6 C4FB 033A 1BD4 F410 4558 4543;
F810  5554 4520 0088 F980 A830 2EFB 11C2 F6CA;
F820  FB40 3A29 D4F8 3430 02FB 55CA F7B0 D4F8;
F830  34C0 F7A6 D4F6 799F A73A 66D4 F410 0D0A;
F840  0088 FA20 324F D4F4 104E 4F20 0030 5898;
F850  BBF8 21B8 88F9 20A8 D4F4 1049 4E54 4552;
F860  5255 5054 00D5 FBCC C2F9 5287 FA0F A987;
F870  FAF0 3A7A D4F6 4D9F B7D5 FB10 3A82 D4F6;
F880  50D5 FB30 3A8A D4F6 64D5 FB60 3A93 D4F6;
F890  5030 77FB 103A 9DD4 F667 975A D5FB D03A;
F8A0  A409 B7D5 FB10 3AAE 899F 10A9 30A1 FB30;
F8B0  3AB5 9759 D5FB 103A BF89 F910 A930 B2FB;
F8C0  603A CB89 5298 FAF0 F1B8 D5FB 303A DA89;
F8D0  FEFE FEFE 5298 FA0F 30C8 FB20 32EC FBF0;
F8E0  CAF9 63D4 F679 87FA 04CA F912 87F6 F6F6;
F8F0  F687 FA03 C2F9 09FB 01CA F904 88FA 01E2;

F900  3330 3033 FB03 3A0D 9733 3330 3088 FA02;
F910  3000 F8D3 73F8 D573 82FC 0273 8752 D222;
F920  98FA 0FA9 88FE D4F6 6998 FA0F A99F 59D5;
F930  3A57 C832 5787 FA30 3A20 87FA 043A 2FD4;
F940  F679 9F73 D4F6 7998 FA0F F910 A960 F059;
F950  3029 88FA 203A 2F87 FA30 3A2F D4F6 79D4;
F960  F679 D552 98F6 F6F6 F6A9 02FB 5032 FB87;
F970  FB78 3A7A D4F6 679B 5AD5 FB01 3A91 F802;
F980  A9D4 F664 985A BBFA 0F52 FEFE FEFE F1B8;
F990  D5FB 033A 9A88 FAFE A8D5 FB01 3AA3 88F9;
F9A0  01A8 D5FB 09C2 F88E FB01 3AB2 D4F6 6797;
F9B0  5AD5 FB03 3ABC 88FA DFA8 30C4 FB01 3ACA;
F9C0  88F9 20A8 D4F6 509F B3D5 22F8 D373 87FA;
F9D0  07FB 063A D887 30EA 87FA 0832 E2D4 F679;
F9E0  30E5 D4F6 4D9F 7387 F908 5288 F6F6 97D2;
F9F0  B788 33F7 FAFD C8F9 02A8 D522 F8D3 7387;

FA00  FA08 3A0D D4F6 509F 7387 52D2 D5D4 F667;
FA10  EA87 52D2 B7E2 D5FF FFFF FFFF FFFF FFFF;
FA20  FFFF FFFF FFFF FFFF FFFF FFFF FFFF FFFF;
FA30  FFFF FFFF FFFF FFFF FFFF FFFF FFFF FFFF;
FA40  FFFF FFFF FFFF FFFF FFFF FFFF FFFF FFFF;
FA50  FFFF FFFF FFFF FFFF FFFF FFFF FFFF FFFF;
FA60  FFFF FFFF FFFF FFFF FFFF FFFF FFFF FFFF;
FA70  FFFF FFFF FFFF FFFF FFFF FFFF FFFF FFFF;
FA80  FFFF FFFF FFFF FFFF FFFF FFFF FFFF FFFF;
FA90  FFFF FFFF FFFF FFFF FFFF FFFF FFFF FFFF;
FAA0  FFFF FFFF FFFF FFFF FFFF FFFF FFFF FFFF;
FAB0  FFFF FFFF FFFF FFFF FFFF FFFF FFFF FFFF;
FAC0  FFFF FFFF FFFF FFFF FFFF FFFF FFFF FFFF;
FAD0  FFFF FFFF FFFF FFFF FFFF FFFF FFFF FFFF;
FAE0  FFFF FFFF FFFF FFFF FFFF FFFF FFFF FFFF;
FAF0  FFFF FFFF FFFF FFFF FFFF FFFF FFFF FFFF
*
```

6.01.17

Debug by Michael Weigert

The Debug program executes machine language programs instruction by instruction and gives you an update on the status of all of the 1802 microprocessor memories. This will be a great tool for any machine language programer. Not only can it be used to find bugs but also can be used to analize unfamiliar programs. The debug program manipulates pseudo registers, Q, DF, D, IE, P, X, and T. These are either in memory (pseudo registers) or stored in real registers (pseudo everything else). The real status of the external flags is checked when the debug program executes an external flag instruction and the debug program executes all input and output instructions. The debug program should not execute any 69 instructions which turn on the video. This will cause the program to freek out because register 1 is never initilized to anything or used by the program. Registers 1 and 0 are not used by my program so that somebody that wishes to revise my program to work on the display could do it. When I get time to add on to my program I will add some functions to allow this program to debug Chip 8 programs. Here is a list of functions (typed in on a parallel keyboard -- sometimes a return is required).

T	Type out all of the microprocessor memories
(Ctrl)T	Execute next instruction and type out all ...
A	Advance in the program one instruction
E	Execute an instruction you enter from the keyboard
F	Find an instruction and execute all the instructions in the process
(Ctrl)F	Type all of the instructions executed until the instruction is found
B	Break the Find routine

Note that when the find routine ends it types out all ...

The debug program is located at F600-FA16 in our operating system please look at the operating system listing for the debug program. Here is an example of the program analizing the beginning of Chip 8.

```
R(0)=0000   R(1)=EEFF   R(2)=0000   R(3)=0000
R(4)=0000   R(5)=0000   R(6)=0000   R(7)=0000
R(8)=0000   R(9)=0000   R(A)=0000   R(B)=0000
R(C)=0000   R(D)=0000   R(E)=0000   R(F)=0000
P=0   M(R(P))=91   X=0   M(R(X))=91   D=43   T=EF
DF=0   Q=0   IE=1
91
R(0)=0001   R(1)=EEFF   R(2)=0000   R(3)=0000
R(4)=0000   R(5)=0000   R(6)=0000   R(7)=0000
R(8)=0000   R(9)=0000   R(A)=0000   R(B)=0000
R(C)=0000   R(D)=0000   R(E)=0000   R(F)=0000
P=0   M(R(P))=BB   X=0   M(R(X))=BB   D=EE   T=EF
DF=0   Q=0   IE=1
```

6.01.18

Globe II by Michael Weigert

Globe II started as a science fair project at school and I worked on it off and on for about 10 months. This program simulates the rotation of the earth on the VIP. The program makes use of the 64x64 pixel display which was choosen for its symmetry (we also have a video monitor which can manipulate the picture to get square pixels with a 64x64 display). Sorry if you and your readers do not have this capability but it still should be a Pretty Impressive Program. The program occupies 8K of memory and it is possibl to fit the program in 4K but I have not been able to take the time to debug my 4K version. Here is a listing of my Globe II program.

```
*?M0 200
0000  90B3 F306 A3D3 F302 B1A1 B2F3 7FA2 E2F3;
0010  00B3 6993 A4AD F310 B4F3 02AE F31C BF93;
0020  AF98 5F1F 3F3A 212E 3E3A 2193 A3A5 F313;
0030  B5F3 0EB7 F3E0 A7F3 1CB6 F304 A607 F6F6;
0040  F652 07FA 07AC 36F7 A626 F332 B993 A9AA;
0050  AB93 BC3C 3A6D 36FA 07FB 073A 6202 FF30;
0060  3B64 9C56 983C 1602 32C1 FF01 5222 3CFF;
0070  01FA 07AC 3DFA 073A 70AD 0414 BD1D 33FA;
0080  013A 9B05 F6F6 F6F6 5299 FF01 32FE B933;
0090  FA02 3295 2533 F901 A330 A905 FA0F 5283;
00A0  FA02 3AA5 1533 F905 A33B F4FF 0F33 C93B;
00B0  F4AB 9DFE 3D33 3C39 F4A9 3074 3AF4 AA30;
00C0  7433 F903 A325 0230 6AAF F552 9DFE BD33;
00D0  E083 F452 3AF7 9C7E BC3F A9AB F300 AA30;
00E0  FB00 0030 3D00 3013 0000 0000 006A F452;
00F0  8995 9C7E BC3F AAAB F300 A912 3053 983E;

0100  3CAF 1F3F 320C 9C7E BC2F 3003 9C56 1226;
0110  36FF F33B 2796 FB1D 325B 35FC 20A5 957C;
0120  00B5 1733 F904 A333 FA04 A332 3327 85FF;
0130  20A5 957F 0035 3041 1735 FC20 A595 7C00;
0140  B526 36FA F8FC 0CA6 967C 00B6 F300 AD34;
0150  FAE0 FC20 A494 7C00 B430 E093 3363 F300;
0160  B830 66F3 FFB3 F31C B6F3 1EBF F300 A6AF;
0170  0006 5F16 1F36 3A71 96FB 1E3A 71F3 10B4;
0180  F300 A434 14F6 0476 5434 FA1F FB13 3293;
0190  1430 3634 FAE0 A404 7654 34FC 20A4 947C;
01A0  00B4 343A 3394 FB13 3A33 7B00 7A30 E300;
01B0  0000 0000 0000 0000 0000 0000 0000 0000;
01C0  0000 0000 0000 0000 0000 0000 0000 0000;
01D0  0000 0000 0000 0000 0000 0000 0000 0000;
01E0  F300 B3F3 00B3 0000 0000 0000 0000 0000;
01F0  0000 0000 0000 0000 0000 0000 0000 0000
*
```

```
*?M200 20
0200  4270 2276 2252 C4C4 C4F6 1EB0 F600 A030;
0210  E2E2 20A0 E260 A0E2 20A0 3C0F 3000 0000
*?MEE0 20
0EE0  060A 0C0F 1012 1315 1617 1819 191A 1B1B;
0EF0  1C1D 1D1D 1E1E 1F1F 1F1F 2020 2020 2020
*?M1000 300
1000  0000 0000 0000 0000 0FBF FE00 0000 0000;
1010  0000 0000 0000 0000 0000 0000 0000 0000;
1020  0001 8000 EF0E EFE0 07FF FE00 0000 0000;
1030  0000 01FF F000 0000 0000 0000 0000 0000;
1040  007F FF3F FF01 FC1F E1FF F3F0 0003 FFF6;
1050  0036 3FFF FFFF FFFF C000 0000 0000 0000;
1060  F87F FFFF FFFF F003 81FF 0000 001F 3F07;
1070  FFFF FFFF FFFF FFFF FF00 0000 0000 0000;
1080  00FF FFFF FFFF 867C 003E 0000 20FC 7FFF;
1090  FFFF FFFF FFFF FFFE 0000 0000 0000 0000;
10A0  0076 00FF FFFF E07F F800 0000 B006 FFFF;
10B0  FFFF FFFF FFFF E006 0030 0000 0000 0000;
10C0  0080 000F FFFF FEFF F800 0001 9367 FFFF;
10D0  FFFF FFFF FFFF C00C 0000 0000 0000 0000;
10E0  0000 0003 FFFF FFFF E700 0000 1BFF FFFF;
10F0  FFFF FFFF FFFF E30C 0000 0000 0000 0000;

1100  0000 0001 FFFF FFFF E000 0000 1FFF FFFF;
1110  FFFF FFFF FFFF E300 0000 0000 0000 0000;
1120  0000 0001 FFFF FFFF 4000 0000 1FFF F93F;
1130  FFFF FFFF FFFF C180 0000 0000 0000 0000;
1140  0000 0001 FFFF FFFC 0000 0000 FC19 F81F;
1150  FFFF FFFF FFF9 9C00 0000 0000 0000 0000;
1160  0000 0000 FFFF FFF0 0000 0000 F804 CFFF;
1170  FFFF FFFF FFF8 0000 0000 0000 0000 0000;
1180  0000 0000 FFFF FFE0 0000 0000 F000 4FFF;
1190  FFFF FFFF FE30 3000 0000 0000 0000 0000;
11A0  0000 0000 7FFF FFC0 0000 0000 07E0 00FF;
11B0  FFFF FFFF FF90 6000 0000 0000 0000 0000;
11C0  0000 0000 1FFF FF00 0000 0000 7FF0 C2FF;
11D0  FFFF FFFF FF01 E000 0000 0000 0000 0000;
11E0  0000 0000 0FFF FF00 0000 0000 FFFF FFFF;
11F0  FFFF FFFF FF05 0000 0000 0000 0000 0000;

1200  0000 0000 05FF A300 0000 0003 FFFF FFFF;
1210  3FFF FFFF FF80 0000 0000 0000 0000 0000;
1220  0000 0000 02FE 0100 0000 0007 FFFF FFFF;
1230  BFFF FFFF FF00 0000 0000 0000 0000 0000;
1240  0000 0000 017E 0000 0000 0007 FFFF FF7F;
1250  83FF FFFF FE10 0000 0000 0000 0000 0000;
1260  0000 0000 003E 0380 0000 000F FFFF FF3F;
1270  F80F FEFF FC80 0000 0000 0000 0000 0000;
1280  0000 0000 001F 1060 0000 001F FFFF FF9F;
1290  FC0B F87F F000 0000 0000 0000 0000 0000;
12A0  0000 0000 000F 700E 8000 001F FFFF FFDF;
12B0  F803 F03F 2000 0000 0000 0000 0000 0000;
12C0  0000 0000 0007 F004 0000 001F FFFF FFCF;
12D0  F003 E03F 0080 0000 0000 0000 0000 0000;
12E0  0000 0000 0001 FE00 0000 001F FFFF FFE7;
12F0  C003 C01F 0080 0000 0000 0000 0000 0000
*
```

```
?M1300 300
1300  0000 0000 0000 1E00 0000 001F FFFF FFF3;
1310  0003 C01F 30C0 0000 0000 0000 0000 0000;
1320  0000 0000 0000 0600 0000 000F FFFF FFF8;
1330  0001 C01F 8020 0000 0000 0000 0000 0000;
1340  0000 0000 0000 031B E000 000F FFFF FFFF;
1350  8001 801B 80C0 0000 0000 0000 0000 0000;
1360  0000 0000 0000 00FF F000 0003 FFFF FFFF;
1370  8001 8019 0020 0000 0000 0000 0000 0000;
1380  0000 0000 0000 007F F800 0001 FFFF FFFF;
1390  8000 4003 0060 0000 0000 0000 0000 0000;
13A0  0000 0000 0000 007F FE00 0000 7FFF FFFF;
13B0  0000 400C 0600 0000 0000 0000 0000 0000;
13C0  0000 0000 0000 007F FF80 0000 007F FFFE;
13D0  0000 0036 1C00 0000 0000 0000 0000 0000;
13E0  0000 0000 0000 007F FF80 0000 003F FFFE;
13F0  0000 001A 7E00 0000 0000 0000 0000 0000;

1400  0000 0000 0000 00FF FF80 0000 003F FFFC;
1410  0000 000C 7CE3 0000 0000 0000 0000 0000;
1420  0000 0000 0000 00FF FFB8 0000 003F FFF8;
1430  0000 0006 3DC3 0000 0000 0000 0000 0000;
1440  0000 0000 0000 00FF FFFF 0000 001F FFF0;
1450  0000 0003 0149 F080 0000 0000 0000 0000;
1460  0000 0000 0000 00FF FFFF 8000 000F FFF0;
1470  0000 0000 0000 FC00 0000 0000 0000 0000;
1480  0000 0000 0000 00FF FFFF 8000 000F FFF0;
1490  0000 0000 FA00 7E00 0000 0000 0000 0000;
14A0  0000 0000 0000 007F FFFF 8000 0007 FFE0;
14B0  0000 0000 01B0 3A00 0000 0000 0000 0000;
14C0  0000 0000 0000 007F FFFF 0000 0007 FFF0;
14D0  0000 0000 0000 0002 0000 0000 0000 0000;
14E0  0000 0000 0000 003F FFFF 0000 0007 FFF0;
14F0  0000 0000 0000 0300 0000 0000 0000 0000;

1500  0000 0000 0000 003F FFFF 0000 0007 FFF0;
1510  8000 0000 0001 0300 8000 0000 0000 0000;
1520  0000 0000 0000 001F FFFF 0000 000F FFE1;
1530  8000 0000 0007 8C00 0000 0000 0000 0000;
1540  0000 0000 0000 000F FFFE 0000 000F FFC3;
1550  8000 0000 003F DC00 0000 0000 0000 0000;
1560  0000 0000 0000 0007 FFFE 0000 0007 FF83;
1570  0000 0000 007F FE00 0000 0000 0000 0000;
1580  0000 0000 0000 0007 FFFE 0000 0007 FF87;
1590  0002 0000 01FF FE00 0000 0000 0000 0000;
15A0  0000 0000 0000 0007 FFF0 0000 0003 FF36;
15B0  0000 0000 07FF FF00 4000 0000 0000 0000;
15C0  0000 0000 0000 0007 FFE0 0000 0003 FF02;
15D0  0000 0000 07FF FF80 0000 0000 0000 0000;
15E0  0000 0000 0000 000F FFC0 0000 0003 FF00;
15F0  0000 0000 07FF FFC0 0000 0000 0000 0000
*
```

6.01.21

```
?M1600 300
1600  0000 0000 0000 000F FFC0 0000 0001 FE00;
1610  0000 0000 07FF FFC0 0000 0000 0000 0000;
1620  0000 0000 0000 000F FF80 0000 0001 FC00;
1630  0000 0000 03FF FFC0 0000 0000 0000 0000;
1640  0000 0000 0000 000F FE00 0000 0000 0000;
1650  0000 0000 03E1 FF80 0000 0000 0000 0000;
1660  0000 0000 0000 000F F800 0000 0000 0000;
1670  0000 0000 0000 7F00 0600 0000 0000 0000;
1680  0000 0000 0000 000F E000 0000 0000 0000;
1690  0000 0000 0000 1E00 0700 0000 0000 0000;
16A0  0000 0000 0000 000F C000 0000 0000 0000;
16B0  0000 0000 0000 0100 0200 0000 0000 0000;
16C0  0000 0000 0000 000F 0000 0000 0000 0000;
16D0  0000 0000 0000 0300 1800 0000 0000 0000;
16E0  0000 0000 0000 000E 0000 0000 0000 0000;
16F0  0000 0000 0000 0000 3000 0000 0000 0000;

1700  0000 0000 0000 000E 0000 0000 0000 0000;
1710  0000 0000 0000 0000 0000 0000 0000 0000;
1720  0000 0000 0000 000C 0000 0000 0000 0000;
1730  0000 0000 0000 0000 0000 0000 0000 0000;
1740  0000 0000 0000 0006 0000 0000 0000 0000;
1750  0000 0000 0000 0000 0000 0000 0000 0000;
1760  0000 0000 0000 0000 0000 0000 0000 0000;
1770  0000 0000 0000 0000 0000 0000 0000 0000;
1780  0000 0000 0000 0000 0000 0000 0000 0000;
1790  0000 0000 0000 0000 0000 0000 0000 0000;
17A0  0000 0000 0000 0007 E000 0001 F37F 39FF;
17B0  FFFF FFFF FFFF FFF0 7000 0000 0000 0000;
17C0  0000 01FF FFFF FFFF FFE0 FFFF FFFF FFFF;
17D0  FFFF FFFF FFFF FFF0 7000 0000 0000 0000;
17E0  07FF FFFF FFFF FFFF FFE1 FFFF FFFF FFFF;
17F0  FFFF FFFF FFFF FFFF F000 0000 0000 0000;

1800  0000 0011 1111 1111 1122 2222 2222 2222;
1810  2222 3333 3333 3333 3300 0000 0000 0000;
1820  0001 1111 1122 2222 2233 3333 3334 4444;
1830  4444 4444 4455 5555 5500 0000 0000 0000;
1840  0001 1111 1222 2223 3333 3444 4444 4555;
1850  5555 5555 5556 6666 6600 0000 0000 0000;
1860  0011 1112 2222 3333 3444 4455 5555 6666;
1870  6666 6777 7777 7777 7700 0000 0000 0000;
1880  0011 1122 2223 3334 4444 5555 5566 6666;
1890  6777 7777 7777 7778 8800 0000 0000 0000;
18A0  0011 1122 2333 3444 4455 5566 6666 7777;
18B0  7788 8888 8888 8888 9900 0000 0000 0000;
18C0  0011 1222 2333 3444 4555 5666 6677 7777;
18D0  8888 8888 9999 9999 9900 0000 0000 0000;
18E0  0011 1222 3333 4445 5556 6667 7778 8888;
18F0  8999 9999 9AAA AAAA AA00 0000 0000 0000
*
```

6.01.22

```
?M1900 300
1900  0011 1222 3334 4455 5566 6777 7888 8999;
1910  9999 AAAA AAAA AAAA AA00 0000 0000 0000;
1920  0111 2223 3334 4455 5666 6777 8888 8999;
1930  99AA AAAA AAAB BBBB BB00 0000 0000 0000;
1940  0111 2223 3344 4555 6667 7778 8889 9999;
1950  AAAA AABB BBBB BBBB BB00 0000 0000 0000;
1960  0111 2223 3344 4556 6667 7788 8899 99AA;
1970  AAAB BBBB BBBC CCCC CC00 0000 0000 0000;
1980  0111 2223 3344 4556 6667 7788 8899 99AA;
1990  AAAB BBBB BBBC CCCC CC00 0000 0000 0000;
19A0  0111 2223 3444 5556 6677 7888 9999 AAAA;
19B0  ABBB BBCC CCCC CCCC CC00 0000 0000 0000;
19C0  0111 2233 3444 5566 6777 8889 999A AAAB;
19D0  BBBB CCCC CCCC CDDD DD00 0000 0000 0000;
19E0  0111 2233 3444 5566 6777 8889 999A AAAB;
19F0  BBBB CCCC CCCC CDDD DD00 0000 0000 0000;

1A00  0111 2233 3445 5566 7778 8899 99AA ABBB;
1A10  BBCC CCCD DDDD DDDD DD00 0000 0000 0000;
1A20  0111 2233 4445 5666 7778 8999 AAAA BBBB;
1A30  CCCC CDDD DDDD DEEE EE00 0000 0000 0000;
1A40  0111 2233 4445 5666 7778 8999 AAAA BBBB;
1A50  CCCC CDDD DDDD DEEE EE00 0000 0000 0000;
1A60  0111 2233 4445 5666 7778 8999 AAAA BBBB;
1A70  CCCC CDDD DDDD DEEE EE00 0000 0000 0000;
1A80  0112 2233 4455 5667 7788 899A AAAB BBCC;
1A90  CCCD DDDD EEEE EEEE EE00 0000 0000 0000;
1AA0  0112 2233 4455 5667 7788 899A AAAB BBCC;
1AB0  CCCD DDDD EEEE EEEE EE00 0000 0000 0000;
1AC0  0112 2333 4455 6667 7888 999A AABB BCCC;
1AD0  CDDD DEEE EEEE EEFF FF00 0000 0000 0000;
1AE0  0112 2333 4455 6667 7888 999A AABB BCCC;
1AF0  CDDD DEEE EEEE EEFF FF00 0000 0000 0000;

1B00  0112 2333 4455 6667 7888 999A AABB BCCC;
1B10  CDDD DEEE EEEE EEFF FF00 0000 0000 0000;
1B20  0112 2333 4455 6667 7888 999A AABB BCCC;
1B30  CDDD DEEE EEEE EEFF FF00 0000 0000 0000;
1B40  0112 2334 4455 6677 7889 99AA ABBB CCCD;
1B50  DDDE EEEE EFFF FFFF FF00 0000 0000 0000;
1B60  0112 2334 4455 6677 7889 99AA ABBB CCCD;
1B70  DDDE EEEE EFFF FFFF FF00 0000 0000 0000;
1B80  0112 2334 4455 6677 7889 99AA ABBB CCCD;
1B90  DDDE EEEE EFFF FFFF FF00 0000 0000 0000;
1BA0  0112 2334 4455 6677 7889 99AA ABBB CCCD;
1BB0  DDDE EEEE EFFF FFFF FF00 0000 0000 0000;
1BC0  0112 2334 4455 6677 7889 99AA ABBB CCCD;
1BD0  DDDE EEEE EFFF FFFF FF00 0000 0000 0000;
1BE0  0112 2334 4455 6677 7889 99AA ABBB CCCD;
1BF0  DDDE EEEE EFFF FFFF FF00 0000 0000 0000
*
```

6.01.23

Some final thoughts...

Since the first part of the editorial was written, there have been a few more renewals, but none at all in the past two weeks. I admit I've been dragging my feet about getting this issue out, but I wanted to see if there would be any more renewals. Also, it's a little tough to get the juices going when the weather is warm and there a lot of fun things to do. However, here is the issue.

During this interval of time I've been pondering what direction we should go with VIPHCA. The survey results of the questionaire, I think, indicate that there is a general willingness to support some other computer or computers, perhaps in conjunction with any VIP or 1802 projects that might come along. And that's nice to see. There are some interesting computers available at very modest cost.

My personal recomendations for a good hobbyist computer are these: 1. Commodore 64, because of the HUGE market support in software, magazines, and hardware; 2. Radio Shack CoCo II, because of it's clearly superior processor, speed, and new compact case and even newer keyboard, and magazine support; 3. Apple IIe, even though it is still expensive compared to the others, (the price has dropped somewhat lately) because of the large software and hardware support and because it is easy to add extra boards, languages, etc. Each of these machines has 64K memory, which is quite enough to do some serious computing and support the use of a disk system for programs and data. The Commodore disk is very slow, compared to the others, but still much faster than a cassette system. They also have seemed to solve some early problems with overheating and mechanical reliability. I also like the VIC-20, but it has a smaller memory and suffers from sluggish sales these days, and is in risk of being discontinued. Most VIC-20 hardware and accessories (printers, disk drive, etc.) will work with the C-64 which is a feature in its favor.

So, although my heart says to continue with VIPER, my head says that it isn't practical. There are just too many good magazines and clubs already in existance supporting those computers mentioned above. Therefore, I think the time has come for us to set VIPER to rest. It had a good run, but its time has now past. I owe a lot to the little VIP for helping me learn about computers. I've had lots of fun and don't regret for one minute my choice of a VIP for a first computer. I'm sure many of you feel the same way.

Since I've deposited the checks that you sent in for this years's dues, what I'll do is deduct a modest amount, most likely $1 per member, to cover expenses involved with this final issue, and refund the balance of the dues to those of you who renewed for '84.

I've made many friends through VIPHCA over the years, and I'll miss our little "chats", even though most of the time it's a one-way conversation. And I hope, should you feel so inclined, that you won't hesitate to drop me a card or letter just to keep in touch or if you think that I might be able to help solve a computer problem. I can't guarantee that I'll have the answer for you, but I'll try to find out who might have the answer. And if you have a VIP or 1802 related problem, I will again do my best to help you.

73, Ray