VZ SOFTWARE MODIFICATIONS

Fast Graphics on a VZ200/300? It can be done! Here is the good oil!

Chris Griffin

I BOUGHT A VZ200 soon after they were released as an 'upgrade' from my old 6800-based CHIP-8 machine. But it soon became obvious something was missing. It seemed I could get speed or high resolution, but not both. I wanted something that was fast *and* took full advantage of the 128 x 64 dot colour graphics; so, 'VZChip-8' was born.

VZChip-8 is a 'low memory' interpreter (about 1.5K all up), designed for VZ200s/300s with only 8K of memory. Figure 1 shows a memory map of a typical VZ computer running my Chip-8 'system'. Notice the presence of an editor. This is used to write your Chip-8 program and can also be used to write machine code programs. It is a separate program in its own right — a stand-alone component in the CHIP-8 system, so I have decided to discuss it first.

The Chip-8/machine code editor

This program is about 1K long and allows you to work entirely independently of BASIC. In fact, it allows you to talk directly to the central processor. Programs are written in hexadecimal — or base 16, and consist of a string of op-codes and arguments. If you don't understand you should get hold of a book on machine code programming for the Z80.

The basic requirements of an editor are that it be able to write, run and modify programs, print listings and save to tape or disk. I have included a few extras because I find them helpful, but otherwise, the editor consists only of these things.

Editor commands consist of a single letter. Its features revolve around the memory pointer. This is just like an arrow, pointing to a particular place in the VZ's memory. The editor uses the arrow to indicate where it is to store or retrieve the information it needs. For example, if you want to list a program beginning at memory location 8260, you first set the memory pointer to 8260, then instruct the editor to list. How do you do all of these things? Easy; using the following commands:

A prints out the ASCII value of the next character typed.

B returns to BASIC; this is used for saving to disk and loading from tape or disk.

D converts a hexadecimal number to its decimal equivalent.

G is used to run machine code program.

H help, prints out a message to remind you of something.

L lists memory to the screen, beginning at the memory pointer.

M sets the memory pointer to a particular place.

O outputs (saves) a program to tape; produces **B** programs which run automatically when you CLOAD them.

P puts data to memory, beginning at the memory pointer position. This command is used for writing and modifying programs.

S searches for a particular byte (or two), and points the memory pointer to the place where a match occurs.

T type; the same as list, except to the printer.

V vector; places the pointer at the memory location which is stored at the present pointer position.

X eXtension; allows for user defined commands, and others; an extension is used to activate Chip-8 programs.

Command extensions: X

Commands beginning with X are two characters long: the second character is a

Figure 1. Memory map of an operational VZChip-8 programming environment.

(Memory map labels:)
```
                        — 8F30
EDITOR
                        — 8AFD
CHIP-8
PROGRAMMING AREA

CHIP-8 PROGRAM START    — 8200
CHIP-8 DATA/PROGRAM
AREA
                        — 8000
CHIP-8 INTERPRETER
                        — 7AE9
```



```
8000 ??M
ADDRESS = 3450
3450 ??M
ADDRESS = 8628
8628 ??S
VALUE = 5622
FINISH =
ØCE5 ??L
ØCE5    56 72 7A 23 ØD 2Ø F9 78
ØCED = D6 Ø8 FE CØ 2Ø E6 C3 78
ØCF5 = Ø7 Ø5 21 1C 79 CD 97 ØD
ØCFD = B7 F2 F6 ØC 78 B7 28 Ø9
ØDØ5 = 21 24 79 86 77 D2 78 Ø7
ØDØD = C8 3A 1C 79 B7 FC 2Ø ØD
ØD15 = 21 25 79 7E E6 8Ø 2B 2B
ØD1D = AE 77 C9 21 1D 79 Ø6 Ø7
ØCE5 ??M
ADDRESS = ØØ97ØØØ
7ØØØ ??P
7ØØØ =  48 45 4C 4C 4F 2Ø 29 2Ø
7ØØ8 =
7ØØØ ??X
EXTENSION #D
```

Figure 2. Some of the editor commands in operation.

number (between 0 and F). Some X commands are already defined:

XO prints out a message beginning at the memory pointer position; (all messages use the byte 00 to signify the end).

XD directs all output to the video screen.

XE directs all output to the printer; for instance, Figure 2 was generated in this fashion.

XC We shall use the XC command to activate the Chip-8 interpreter but since it hasn't yet been installed XO just clears the screen. The process of adding your own X commands will become obvious when we discuss connection of the Chip-8 interpreter.

LISTING 1. USING THE EDITOR.

```
Ø ' CHIP-8 INTERPRETOR PART I
1 ' EDITOR PROGRAM
2 ' DON'T BREAK THIS PROGRAM ONCEIT
3 ' BEGINS RUNNING...
4 '
5 CLS:PRINT@200,"PLEASE WAIT!!"
10 GOSUB50:IFA$="XX"THENGOSUB50:D=X:GOSU
B50:D=D*256+X:GOTO10
15 IFA$="ZZ"THENPOKE30863,112:POKE30862,
Ø:GOTO70
20 POKED,X:T=T+X:D=D+1:GOTO10
50 READA$:IFA$="XX"ORA$="ZZ"THENRETURN
51 X=ASC(LEFT$(A$,1))-48:B=ASC(RIGHT$(A$
,1))-48
60 X=(X+(X>9)*2)*16+(B+(B>9)*7)
65 RETURN
70 IFT=118309,PRINTUSR(1)
75 CLS:PRINT"AN ERROR HAS BEEN MADE, CHE
CK "
80 PRINT"THE LISTING CAREFULLY"
99 'MAIN PROGRAM LISTING
100 DATAXX,70,00,01,30,04,21,00,72,11,FD
,8A,ED,BØ,C3,FD,8A
110 DATAXX,72,00,C3,E5,8B,7C,CD,05,8B,7D
,F5,1F,1F,1F,1F,CD,ØE,8B
120 DATAF1,E6,ØF,C6,30,FE,3A,38,Ø2,C6,Ø7
,18,18,E5,C5,CD
130 DATAF4,2E,B7,2Ø,FA,CD,F4,2E,B7,28,FA
,ØE,3Ø,1Ø,FE,ØD
140 DATA2Ø,FB,C1,E1,C9,E5,C5,CD,E4,8E,36
,2Ø,CD,2A,Ø3,2A
150 DATA2Ø,78,36,AF,C1,E1,C9,E5,C5,F5,CD
,5Ø,34,F1,18,E7
160 DATAE5,C5,CD,1A,8B,47,FE,ØD,28,ØB,FE
,3Ø,38,F4,FE,3A
170 DATA3Ø,1Ø,E6,ØF,21,3E,8Ø,F5,78,CD,44
,8B,F1,FE,8Ø,C1
180 DATAE1,C9,FE,41,38,DC,FE,47,3Ø,D8,D6
,Ø7,18,E4,1A,B7
190 DATAC8,CD,32,8B,13,18,F7,CD,ØØ,8B,11
,AØ,8B,CD,2A,8B
200 DATAØ6,Ø8,3E,2Ø,CD,32,8B,7E,23,CD,Ø5
,8B,1Ø,F4,3E,ØD
210 DATAC3,32,8B,2Ø,3D,ØØ,CD,7B,8B,3E,2Ø
,CD,32,8B,21,ØØ
220 DATAØØ,Ø6,ØØ,CD,4D,8B,C8,29,29,29,29
,85,6F,Ø4,18,F3
230 DATA21,E9,7A,22,F9,78,21,Ø7,8F,22,8E
,78,2D,CD,F6,8E
240 DATAAF,32,9C,78,3E,11,32,3B,78,32,ØØ
,68,3E,Ø3,32,39
250 DATA78,21,ØØ,8Ø,22,1Ø,78,C9,F3,31,FF
,8F,CD,BD,8B,11
260 DATADD,8D,CD,7B,8B,2A,1Ø,78,CD,ØØ,8B
,11,2D,8C,CD,7B
270 DATA8B,CD,1A,8B,FE,41,38,F9,FE,5B,3Ø
,F5,47,CD,44,8B
280 DATA3E,Ø8,CD,ØØ,32,8B,21,31,8C,7E,FE,FF
,28,D8,23,B8,28
290 DATAØ4,23,23,18,F3,5E,23,56,D5,E1,CD
,2C,8C,18,C6,E9
300 DATA2Ø,3F,3F,ØØ,4C,59,8C,4D,65,8C,47
,6F,6E,53,79,8C
310 DATA5Ø,BD,8C,56,27,8D,41,32,8D,44,53
,8D,4F,64,8D,48
320 DATA46,8E,42,4C,8E,54,57,8E,58,98,8E
,FF,2A,1Ø,78,8E
330 DATAØ8,CD,84,8B,ØD,2Ø,FA,C9,11,Ø4,8E
,CD,A3,8B,22,1Ø
340 DATA78,C9,11,ØE,8E,CD,A3,8B,78,B7,C8
,E9,11,16,8E,CD
350 DATAA3,8B,78,B7,C8,FE,Ø3,F5,3Ø,Ø1,65
,E5,11,1E,8E,CD
360 DATAA3,8B,ED,5B,1Ø,78,13,78,B7,2Ø,Ø3
,2A,1Ø,78,C1,1A
370 DATA13,B8,2Ø,ØF,F1,38,Ø6,F5,1A,B9,2Ø
,Ø7,F1,1B,ED,53
380 DATA1Ø,78,C9,DF,2Ø,E9,11,27,8E,C3,7B
,8E,ØØ,ØØ,ØØ,ØØ
390 DATA2A,1Ø,78,Ø6,ØØ,CD,ØØ,8B,11,1D,8D
,CD,7B,8B,3E,Ø8
400 DATAF5,3E,2Ø,CD,32,8B,CB,78,2Ø,2D,CD
,1A,8B,FE,22,28
410 DATA1D,ØØ,ØØ,CD,18,8D,28,14,87,82,87
,87,F5,CD,4D,8B
420 DATAD1,28,Ø9,82,27,23,F1,3D,2Ø,D6,18
,27,F1,C9,CB,F8
430 DATA3E,41,CD,FF,8E,18,CF,CD,1A,8B,FE
,22,2Ø,Ø6,CB,B8
440 DATA3E,AF,18,EE,F5,CD,44,8B,F1,18,D9
,E5,C5,C3,52,8B
450 DATA2Ø,3D,ØØ,3E,ØD,CD,32,8B,18,9B,2A
,1Ø,78,7E,23,66
460 DATA6F,22,1Ø,78,C9,11,32,8E,CD,7B,8B
,CD,1A,8B,F5,CD
470 DATA44,8B,3E,ØD,CD,32,8B,11,16,8E,CD
,7B,8B,F1,CD,Ø5
480 DATA8B,3E,ØD,C3,32,8B,11,4Ø,8E,CD,A3
,8B,11,16,8E,CD
490 DATA7B,8B,CD,AF,ØF,18,EA,11,39,8E,CD
,7B,8B,21,9D,7A
500 DATAØ6,1Ø,CD,1A,8B,F5,CD,44,8B,F1,FE
,Ø1,C8,FE,ØD,28
510 DATAØ4,77,23,1Ø,ED,36,ØØ,3E,11,9Ø,32
,D6,7A,11,ØE,8E
520 DATACD,A3,8B,E5,11,1E,8E,CD,A3,8B,F3
,ØE,F1,CD,5B,35
530 DATAD1,CD,A3,8D,F3,C9,D8,Ø1,9A,Ø1,ØB
,79,BØ,2Ø,FB,DD
540 DATA21,23,78,7B,CD,11,35,DD,77,ØØ,AF
,DD,77,Ø1,7A,CD
550 DATAD7,8D,7D,CD,D7,8D,7C,CD,D7,8D,CD
,E8,3A,D8,1A,13
560 DATACD,D7,8D,DF,2Ø,F4,E5,C3,FA,34,CD
,11,35,C3,8E,38
570 DATA1F,56,5A,2D,32,3Ø,3Ø,2Ø,48,45,58
,2Ø,45,44,49,54
580 DATA4F,52,ØD,56,45,52,2Ø,32,2E,31,ØD
,28,43,29,2Ø,43
590 DATA47,27,38,35,ØD,ØD,ØØ,41,44,44,52
,45,53,53,2Ø,3D
600 DATAØØ,53,54,41,52,54,2Ø,3D,ØØ,56,41
,4C,55,45,2Ø,3D
610 DATAØØ,46,49,4E,49,53,48,2Ø,3D,ØØ,4E
,4F,54,2Ø,46,4F
620 DATA55,4E,44,ØD,ØØ,43,48,41,52,2Ø,3D
,ØØ,4E,41,4D,45
630 DATA2Ø,3D,ØØ,48,45,58,2Ø,3D,ØØ,11,64
,8E,C3,7B,8B,FB
640 DATACD,7A,1E,ED,7B,E8,78,C3,19,1A,21
,9C,78,36,Ø1,E5
650 DATACD,59,8C,E1,36,ØØ,C9,43,4F,4D,4D
,41,4E,44,53,2Ø
660 DATA41,52,45,ØD,41,2C,42,2C,44,2C,47
,2C,48,2C,4C,2C
680 DATA4D,2C,4F,2C,5Ø,2C,53,2C,54,2C,56
,2C,58,ØD,ØØ,45
690 DATA58,54,45,4E,53,49,4F,4E,2Ø,23,ØØ
,11,8C,8E,CD,7B
700 DATA8B,CD,4D,8B,C8,87,C6,AF,6F,26,8E
,F1,CD,4E,8D,C3
710 DATA22,8C,DA,8E,E4,8B,E4,8B,E4,8B,E4
,8B,E4,8B,E4,8B
720 DATAE4,8B,E4,8B,E4,8B,E4,8B,E4,8B,C9
,Ø1,Ø5,8E,CF,8E
730 DATAE4,8B,3E,Ø1,32,9C,78,C9,AF,32,9C
,78,C9,ED,5B,1Ø
740 DATA78,CD,7B,8B,C3,4E,8D,2A,2Ø,78,47
,3A,9C,78,B7,78
750 DATAC8,FE,8Ø,D8,C6,2Ø,E6,7F,C9,21,FC
,8A,22,B1,78,2D
760 DATA18,12,32,4Ø,8B,3E,Ø1,C3,44,8B,F3
,31,FF,8F,CD,CD
```
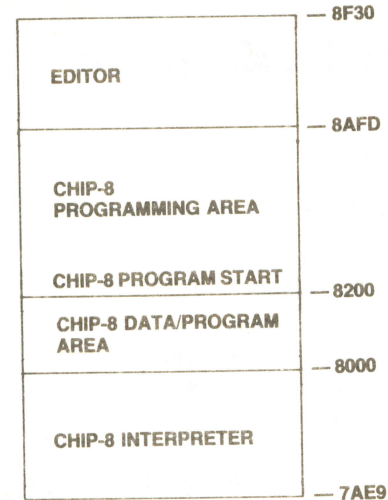
Using the editor

Key in the listing given (Listing 1), save a copy of it, then run the program. You will have to wait a while, until everything is set up. If an error results, check the listing carefully. An introductory message will be printed when the editor is installed. Save a copy in this form to tape or disk. To do this tape users should type: OVZEDITOR (cr) 8AFD (cr) 8F30 (cr), where (cr) means the RETURN key. The last (cr) is not typed until the tape recorder is on and in record mode. Alternatively, type BBSAVE "VZEDITOR",8AFD,8F30('cr'). Both Bs are essential. The first is needed to exit the editor. This step eliminates the delay from occurring every time the editor program is run. It saves the machine code part, produced by Listing 1, to the relevant medium.

Commands

Now, try out some commands: particularly M,L,H and T (if you have a printer). It is a good idea not to use the G or K commands just yet.

You will find that many commands prompt for ADDRESSes, START locations, STOP locations, etc. The answer accepted by the computer consists of the *last* four digits of whatever is typed in. If you meant to type 8BD8, and instead, entered 8BE, just type in the right response and the problem is fixed, so that 8BE8BD8 is interpreted as 8BD8. This is important because the edi-tor is *not* equipped with a backspace facility.

The P command, as I said before, allows you to put data in memory. To test it out, set the memory pointer to 7080 (use M7080 (cr)) and type P. Now, type in the following data: 48454C4C4F (cr). Notice that the word HELLO appears on the screen as you type. You have stored the ASCII values for HELLO at location 7080-7084, which is in screen memory.

How did I know to use 4845 . . .? I looked it up; but that's a laborious task if you want to enter lots of words into memo-ry. Instead, you can use an easier form: type M70C0 (cr) P", the " (shift 2) allows for character data entry — the computer does all of the conversions for you! (Notice that while in this mode, the normally blue cursor turns into an 'A'.) After typing in the re-quired word, pressing another " returns the cursor to blue again, so you can enter hexa-decimal data as usual.

S is used to search for one or two bytes, depending on what you type in, from the memory pointer to the end position (which you also type in). If a two-byte search is re-quired, make sure the search string is more than two digits long. For example, to search for 6A00 in the region of memory 8200 to 8500, type M8200 (cr) S6A00 (cr) 8500 (cr). The message NOT FOUND means that 6A00 could not be found anywhere between loca-tions 8200 and 8500. ●

IMPORTANT EDITOR MEMORY LOCATIONS

The editor has a small collection of useful subroutines. These can be used when prototyping a Chip-8 program or when writing machine code programs. Care should be taken to ensure that calls to these subroutines are not present in the final program, unless the editor is to be included in the final program.

Location	Description	Registers altered
8AFD	Jump location, COLD START.	HL,BC,DE,AF
8B00	Show HL register pair as a hexadecimal value.	AF
8B05	Show A register as a hexadecimal value.	AF
8B1A	Wait for a key press, A contains the ASCII value of the key that was pressed.	AF
8B32	Show the character stored in A.	none
8B44	Show character in A, and beep.	none
8B4D	Get a hexadecimal key (0-F, or (cr)) and put the value in A, A equals 80 if (cr) is pressed.	AF
8B7B	Show a string using DE as the pointer, up to the character stored as 00.	DE,AF
8BA3	Shows a message off DE, and gets a two-byte number from the keyboard; the number is stored in HL, while B contains the number of keys pressed.	HL,B,DE,AF

The following locations contain prompt messages used by the editor. Each message consists of a string of ASCII characters ending with the byte 00. These messages can be changed to suit your own personal requirements.

Location	Length	Description
8DDD	38	Introductory message; this is the heading displayed when the editor first begins.
8E64	39	Help message; the 39 characters here are reserved for a simple memo which is called up by pressing H.
8C2D	3	Prompt string, normally consists of a space and two question marks.

Example: to change the help message, type:
M8E64 (cr) P"this is the new message (cr) "00 (cr)
Make sure that whatever you type as the message is less than the maximum size of 39 characters.
Next month: the CHIP-8 interpreter.