--
Newsgroups: comp.sys.cbm
From: slaur@utu.fi (Sam Laur)
Subject: VIC-20 memory map (long)
Date: Wed, 2 Mar 1994 12:54:14 GMT

Okay, here's what all you VIC-20 freaks out there have been waiting for... Sorry for the delay, but I guess I'm just too lazy to correct the errors that OCR scanners make :-) Also I found a couple of typos in the VIC-20 Programmer's Reference Guide itself. I tried to correct everything, but who knows, there may still be quite many typos lurking in this. Here it comes:

MEMORY MAPS

The following memory maps provide a guide which shows which special locations are set aside for use by the VlC's operating system . . . and what those locations are used for.

Memory Map

HEX	DECIMAL	DESCRIPTION
0000	0	Jump for USR
0001-0002	1-2	Vector for USR
0003-0004	3-4	Float-Fixed vector
0005-0006	5-6	Fixed-Float vector
0007	7	Search character
0008	8	Scan-quotes flag
0009	9	TAB column save
000A	10	0=LOAD, 1=VERIFY
000B	11	Input buffer pointer/# subscript
000C	12	Default DIM flag
000D	13	Type: FF=string, 00=numeric
000E	14	Type: 80=integer, 00=floating point
000F	15	DATA scan/LlST quote/memory flag
0010	16	Subscript/FNx flag
0011	17	0 = INPUT;$40 = GET;$98 = READ
0012	18	ATN sign/Comparison eval flag
0013	19	Current l/O prompt flag
*0014-0015	20-21	Integer value
0016	22	Pointer: temporary string stack
0017-0018	23-24	Last temp string vector
0019-0021	25-33	Stack for temporary strings
0022-0025	34-37	Utility pointer area
0026-002A	38-42	Product area for multiplication
*002B-002C	43-44	Pointer: Start of Basic
*002D-002E	45-46	Pointer: Start of Variables
*002F-0030	47-48	Pointer: Start of Arrays
*0031-0032	49-50	Pointer: End of Arrays
*0033-0034	51-52	Pointer: String storage (moving down)
0035-0036	53-54	Utility string pointer
*0037-0038	55-56	Pointer: Limit of memory
0039-003A	57-58	Current Basic line number
003B-003C	59-60	Previous Basic line number
003D-003E	61-62	Pointer: Basic statement for CONT
003F-0040	63-64	Current DATA line number
0041-0042	65-66	Current DATA address
*0043-0044	67-68	Input vector

* Useful memory location

HEX	DECIMAL	DESCRIPTION
0045-0046	69-70	Current variable name
0047-0048	71-72	Current variable address
0049-004A	73-74	Variable pointer for FOR/NEXT
004B-004C	75-76	Y-save; op-save; Basic pointer save
004D	77	Comparison symbol accumulator
004E-0053	78-83	Misc work area, pointers, etc
0054-0056	84-86	Jump vector for functions
0057-0060	87-96	Misc numeric work area
*0061	97	Accum#1: Exponent
*0062-0065	98-101	Accum#1: Mantissa
*0066	102	Accum#1: Sign
0067	103	Series evaluation constant pointer
0068	104	Accum#1 hi-order (overflow)
*0069-006E	105-110	Accum#2: Exponent, etc.
006F	111	Sign comparison, Acc#1 vs #2
0070	112	Accum#1 lo-order (rounding)
0071-0072	113-114	Cassette buffer length/Series pointer
*0073-008A	115-138	CHRGET subroutine (get BASIC char)
007A-007B	122-123	Basic pointer (within subroutine)
008B-008F	139-143	RND seed value
*0090	144	Status word ST
0091	145	Keyswitch PIA: STOP and RVS flags
0092	146	Timing constant for tape
0093	147	Load=0, Verify=1
0094	148	Serial output: deferred char flag
0095	149	Serial deferred character
0096	150	Tape EOT received
0097	151	Register save
*0098	152	How many open files
*0099	153	Input device (normally 0)
*009A	154	Output (CMD) device, normally 3
009B	155	Tape character parity
009C	156	Byte-received flag
009D	157	Direct=$80/RUN=0 output control
009E	158	Tape Pass 1 error log/char buffer
009F	159	Tape Pass 2 error log corrected
*00A0-00A2	160-162	Jiffy Clock (HML)
00A3	163	Serial bit count/EOI flag
00A4	164	Cycle count
00A5	165	Countdown, tape write/bit count
00A6	166	Pointer: tape buffer
00A7	167	Tape Write ldr count/Read pass/inbit
00A8	168	Tape Write new byte/Read error/inbit cnt
00A9	169	Write start bit/Read bit err/stbit

* Useful memory location

HEX	DECIMAL	DESCRIPTION
00AA	170	Tape Scan;Cnt;Ld;End/byte assy
00AB	171	Write lead length/Rd checksum/parity
00AC-00AD	172-173	Pointer: tape buffer, scrolling
00AE-00AF	174-175	Tape end addresses/End of program
00B0-00B1	176-177	Tape timing constants
*00B2-00B3	178-179	Pointer: start of tape buffer
00B4	180	Tape timer (1 =enable); bit cnt
00B5	181	Tape EOT/RS-232 next bit to send
00B6	182	Read character error/outbyte buffer

```
  *00B7     183      # characters in file name
  *00B8     184      Current logical file
  *00B9     185      Current secondary address
  *00BA     186      Current device
*00BB-00BC  187-188  Pointer: to file name
   00BD     189      Write shift word/Read input char
   00BE     190      # blocks remaining to Write/Read
   00BF     191      Serial word buffer
   00C0     192      Tape motor interlock
00C1-00C2   193-194  I/O start addresses
00C3-00C4   195-196  KERNAL setup pointer
  *00C5     197      Current key pressed
  *00C6     198      # chars in keyboard buffer
  *00C7     199      Screen reverse flag
   00C8     200      Pointer: End-of-line for input
00C9-00CA   201-202  Input cursor log (row, column)
  *00CB     203      Which key: 64 if no key
   00CC     204      cursor enable (0=flash cursor)
   00CD     205      Cursor timing countdown
   00CE     206      Character under cursor
   00CF     207      Cursor in blink phase
   00D0     208      Input from screen/from keyboard
*00D1-00D2  209-210  Pointer to screen line
  *00D3     211      Position of cursor on above line
   00D4     212      0=direct cursor, else programmed
  *00D5     213      Current screen line length
  *00D6     214      Row where cursor lives
   00D7     215      Last inkey/checksum/buffer
  *00D8     216      # of INSERTs outstanding
*00D9-00F0  217-240  Screen line link table
   00F1     241      Dummy screen link
   00F2     242      Screen row marker
*00F3-00F4  243-244  Screen color pointer
00F5-00F6   245-246  Keyboard pointer
00F7-00F8   247-248  RS-232 Rcv pointer
00F9-00FA   249-250  RS-232 Tx pointer

* Useful memory location

                        172


HEX      DECIMAL  DESCRIPTION

*00FB-00FE  251-254  Operating system free zero page space
00FF        255      Basic storage

0100-010A   256-266  Floating to ASCII work area
0100-013E   256-318  Tape error log
0100-01FF   256-511  Processor stack area

*0200-0258  512-600  Basic input buffer
*0259-0262  601-610  Logical file table
*0263-026C  611-620  Device # table
*026D-0276  621-630  Secondary Address table
*0277-0280  631-640  Keyboard buffer
*0281-0282  641-642  Start of memory for op system
*0283-0284  643-644  Top of memory for op system
0285        645      Serial bus timeout flag
*0286       646      Current color code
0287        647      Color under cursor
*0288       648      Screen memory page
*0289       649      Max size of keyboard buffer
*028A       650      Key repeat (128=repeat all keys)
*028B       651      Repeat speed counter
```

028C	652	Repeat delay counter
*028D	653	Keyboard Shift/Control flag
028E	654	Last keyboard shift pattern
028F-0290	655-656	Pointer: decode logic
*0291	657	Shift mode switch (0 = enabled, 128 = locked)
0292	658	Autoscrolldownflag (0=on, <>0=off)
0293	659	RS-232 control register
0294	660	RS-232 command register
0295-0296	661-662	Nonstandard (Bit time/2-100)
0297	663	RS-232 status register
0298	664	Number of bits to send
0299-029A	665-666	Baud rate (full) bit time
029B	667	RS-232 receive pointer
029C	668	RS-232 input pointer
029D	669	RS-232 transmit pointer
029E	670	RS-232 output pointer
029F-02A0	671-672	Holds IRQ during tape operations
02A1-02FF	673-767	Program indirects
*0300-0301	768-769	Error message link
0302-0303	770-771	Basic warm start link
0304-0305	772-773	Crunch Basic tokens link
0306-0307	774-775	Print tokens link
0308-0309	776-777	Start new Basic code link

* Useful memory location

HEX	DECIMAL	DESCRIPTION	
030A-030B	778-779	Get arithmetic element link	
030C	780	Storage for 6502 .A register	
030D	781	Storage for 6502 .X register	
030E	782	Storage for 6502 .Y register	
030F	783	Storage for 6502 .P register	
0310-0313	784-787	??	
0314-0315	788-789	Hardware (IRQ) interrupt vector	[EABF]
0316-0317	790-791	Break interrupt vector	[FED2]
0318-0319	792-793	NMI interrupt vector	[FEAD]
031A-031B	794-795	OPEN vector	[F40A]
031C-031D	796-797	CLOSE vector	[F34A]
031E-031F	798-799	Set-input vector	[F2C7]
0320-0321	800-801	Set-output vector	[F309]
0322-0323	802-803	Restore l/O vector	[F3F3]
0324-0325	804-805	INPUT vector	[F20E]
0326-0327	806-807	Output vector	[F27A]
0328-0329	808-809	Test-STOP vector	[F770]
032A-032B	810-811	GET vector	[F1F5]
032C-032D	812-813	Abort l/O vector	[F3EF]
032E-032F	814-815	User vector (default BRK)	[FED2]
0330-0331	816-817	Link to load RAM	[F549]
0332-0333	818-819	Link to save RAM	[F685]
0334-033B	820-827	??	
*003C-03FB	828-1019	Cassette buffer	
0400-0FFF	1024-4095	3K expansion RAM area	
1000-1DFF	4096-7679	User Basic area	
1E00-1FFF	7680-8191	Screen memory	
2000-3FFF	8192-16383	8K expansion RAM/ROM block 1	
4000-5FFF	16384-24575	8K expansion RAM/ROM block 2	
6000-7FFF	24576-32767	8K expansion RAM/ROM block 3	

NOTE: When additional memory is added to block 1 (and 2 and 3), the KERNAL relocates the following things for BASIC:

```
1000-11FF    4096-4607     Screen memory

1200-?       4608-?     User Basic area

9400-95FF    37888-38399   Color RAM

8000-8FFF    32768-36863   4K Character generator ROM
8000-83FF    32768-33791   Upper case and graphics
8400-87FF    33792-33815   Reversed upper case and graphics
8800-8BFF    33816-35839   Upper and lower case
8C00-8FFF    35840-36863   Reversed upper and lower case
9000-93FF    36864-37887   I/O BLOCK 0

* Useful memory location

                        174


HEX       DECIMAL       DESCRIPTION

9000-900F    36864-36879   Address of VIC chip registers
9000     36864   bits 0-6 horizontal centering
                           bit 7 sets interlace scan
9001     36865   vertical centering
9002     36866   bits 0-6 set # of columns
                           bit 7 is part of video matrix address
9003     36867   bits 1-6 set # of rows
                           bit 0 sets 8x8 or 16x8 chars
9004     36868   TV raster beam line
9005     36869   bits 0-3 start of character memory
                           (default = 0)
                           bits 4-7 is rest of video address
                           (default= F)
                           BITS 3,2,1,0 CM startinq address
                                        HEX   DEC
                           0000   ROM   8000  32768
                           0001         8400  33792
                           0010         8800  34816
                           0011         8C00  35840
                           1000   RAM   0000  0000
                           1001 xxxx
                           1010 xxxx  unavail.
                           1011 xxxx
                           1100         1000  4096
                           1101         1400  5120
                           1110         1800  6144
                           1111         1C00  7168

9006     36870    horizontal position of light pen
9007     36871    vertical position of light pen
9008     36872    Digitized value of paddle X
9009     36873    Digitized value of paddle Y
900A     36874    Frequency for oscillator 1 (low)
                  (on: 128-255)
900B     36875    Frequency for oscillator 2 (medium)
                  (on: 128-255)
900C     36876    Frequency for oscillator 3 (high)
                  (on: 128-255)
900D     36877    Frequency of noise source
900E     36878    bit 0-3 sets volume of all sound
                  bits 4-7 are auxiliary color information
900F     36879    Screen and border color register
                   bits 4-7 select background color
                   bits 0-2 select border color
```

```
                    bit 3 selects inverted or normal mode


                         175


  HEX     DECIMAL DESCRIPTION

  9110-911F  37136-37151 6522 VIA#1
     9110    37136     Port B output register
                       (user port and RS-232 lines)
             PIN    6522 DESCRIPTION         EIA   ABV
             ID     ID

             C      PB0 Received data       (BB)  Sin
             D      PB1 Request to Send     (CA)  RTS
             E      PB2 Data terminal ready (CD)  DTR
             F      PB3 Ring indicator      (CE)  RI
             H      PB4 Received line signal (CF)  DCD
             J      PB5 Unassigned          ( )   XXX
             K      PB6 Clear to send       (CB)  CTS
             L      PB7 Data set ready      (CC)  DSR
             B      CB1 Interrupt for Sin   (BB)  Sin
             M      CB2 Transmitted data    (BA)  Sout

             A      GND Protective ground   (M)   GND
             N      GND Signal ground       (AB)  GND
    9111     37137     Port A output register
                      (PA0) Bit 0=Serial CLK IN
                      (PA1) Bit 1=Serial DATA IN
                      (PA2) Bit 2=Joy 0
                      (PA3) Bit 3=Joy 1
                      (PA4) Bit 4=Joy 2
                      (PA5) Bit 5 = Lightpen/Fire button
                      (PA6) Bit 6=Cassette switch sense
                      (PA7) Bit 7=Serial ATN out
  9112       37138  Data direction register B
  9113       37139  Data direction register A
  9114       37140  Timer 1 low byte
  9115       37141  Timer 1 high byte & counter
  9116       37142  Timer 1 low byte
  9117       37143  Timer 1 high byte
  9118       37144  Timer 2 low byte
  9119       37145  Timer 2 high byte
  911A       37146  Shift register
  911B       37147  Auxiliary control register
  911C       37148  Peripheral control register
                    (CA1, CA2, CB1, CB2)
                    CA1 = restore key (Bit 0)
                    CA2 = cassette motor control (Bits 1-3)
                    CB1 = interrupt signal for received
                        RS-232 data (Bit 4)
                    CB2=transmitted RS-232 data (Bits
                        5-7)
    911D     37149  Interrupt flag register

                         176


      HEX      DECIMAL        DESCRIPTION

     911E      37150            Interrupt enable register
     911F      37151            Port A (Sense cassette switch)
   9120-912F   37152-37167        6522 VIA#2
     9120      37152            Port B output register
```

```
                                keyboard column scan
                                (PB3) Bit 3 =cassette write line
                                (PB7) Bit 7 =Joy 3
   9121      37153            Port A output register
                                keyboard row scan
   9122      37154            Data direction register B
   9123      37155            Data direction register A
   9124      37156            Timer 1, low byte latch
   9125      37157            Timer 1, high byte latch
   9126      37158            Timer 1, low byte counter
   9127      37159            Timer 1, high byte counter
                                timer 1 is used for the
                                60 time/second interrupt
   9128      37160            Timer 2, low byte latch
   9129      37161            Timer 2, high byte latch
   912A      37162            Shift register
   912B      37163            Auxiliary control register
   912C      37164            Peripheral control register
                                CA1 Cassette read line (Bit 0)
                                CA2 Serial clock out (Bits 1-3)
                                CB1 Serial SRQ IN (Bit 4)
                                CB2 Serial data out (Bits 5-7)
   912D      37165            Interrupt flag register
   912E      37166            Interrupt enable register
   912F      37167            Port A output register

 9400-95FF   37888-38399        location of COLOR RAM with
                                additional RAM at blk 1
 9600-97FF   38400-38911        Normal location of COLOR RAM
 9800-9BFF   38912-39935        I/O block 2
 9C00-9FFF   39936-40959        I/O block 3
 A000-BFFF   40960-49152        8K decoded block for expansion ROM
 C000-DFFF   49152-57343        8K Basic ROM
 E000-FFFF   57344-65535        8K KERNAL ROM

                        177


USEFUL MEMORY LOCATIONS

This is a more in-depth guide to some of the memory locations
you can use.

HEX      DECIMAL  DESCRIPTION

0014-0015   20-21    Where BASIC stores integer variables
                     used in calculations. The fixed-float and
                     float-fixed routines (vectors at 1-2 and
                     3-4) use the value in this area.

002B-002C   43-44    The start of the BASIC program in
                     memory. Location 43 contains the low
                     byte, and location 44 has the high byte.
                     To compute the start of BASIC in
                     decimal, use the formula: PEEK(43) +
                     256 * PEEK(44)

002D-002E   45-46    The start of the numeric variables,
                     which is usually immediately after the
                     end of the BASIC program.

002F-0030   47-48    The start of arrays in memory, usually
                     immediately following the numeric vari-
                     ables.
```

0031-0032	49-50	The end of the arrays in memory.
0033-0034	51-52	Bottom of string storage, moving from the top of available memory down to the top of arrays.
0037-0038	55-56	The top of free RAM. By lowering this value, some RAM can be "protected" against BASIC putting values here.
0043-0044	67-68	Jump vector for INPUT statement.
0061-0066	97-102	Floating point accumulator #1 for calculations.
0069-006E	105-110	Floating point accumulator #2.
0073-008A	115-138	The CHRGET subroutine resides here. This routine gets the next BASIC character from machine language.
0090	144	Status word ST.
0098	152	Number of open files.
0099	153	Device number for input, normally 0 (keyboard).
009A	154	Output (CMD) device, normally 3 (screen).

HEX	DECIMAL	DESCRIPTION
00A0-00A2	160-162	3 byte jiffy clock. The Tl and Tl$ variables are translations of these locations.
00B2-00B3	178-179	Points to the start of the tape buffer. Can be used as an indirect zero-page jump to a routine in the buffer.
00B7	183	Number of characters in filename.
00B9	185	Which secondary address is currently being used.
00BA	186	Current device number being accessed.
00BB-00BC	187-188	Points to location of filename in memo-
00C5	197	Current key being held down. There will be a 64 here if nothing is held down. If more than 1 key is down, the key with the highest number on the chart is what shows up here.

#	key	#	key	#	key	#	key
0	1	16	none	32	space	48	Q
1	3	17	A	33	Z	49	E
2	5	18	D	34	C	50	T
3	7	19	G	35	B	51	U
4	9	20	J	36	M	52	O
5	+	21	L	37	.	53	@
6	� (pound)	22	;	38	none	54	^ (up arrow)
7	DEL	23	crsr lt/rt	39	f1	55	f5

```
8  <-           24 STOP         40 none         56 2
9  W            25 none         41 S            57 4
10 R            26 X            42 F            58 6
11 Y            27 V            43 H            59 8
12 I            28 N            44 K            60 0
13 P            29 ,            45 :            61 -
14 *            30 /            46 =            62 HOME
15 RETURN       31 crsr up/dn   47 f3           63 f7

00C6       198      Number of characters currently in key-
                    board buffer.
00C7       199      Flag for reverse on/off. A 1 here is on, a
                    0 is off.
00CB       203      Same as 197.
00D1-00D2  209-210  Address of start of line where cursor is.
00D3       211      Position of cursor on line.
00D5       213      Current screen line length--either 21,
                    43, 65, or 87.

                        179


HEX     DECIMAL  DESCRIPTION

00D6       214      Screen row where cursor is. To change
                    the cursor position, locations 209, 210,
                    211, and 214 must be changed.

00D8       216      Number of spaces left in INSERT mode.
                    POKEing this to a zero will turn off insert
                    mode.

00D9-00F0  217-240  Screen line link table. A 158 means that
                    the line is finished at the end of that line,
                    and a 30 means that the line continues
                    on the next line.

00F3-00F4  243-244  Pointer to the current space in color memory.

00FB-00FE  251-254  Available locations in zero page.

0200-0258  512-600  BASIC input buffer--where the charac-
                    ters being INPUT will go.

0259-0262  601-610  Logic 1 file table for OPEN files.

0263-026C  611-620  Device # table for OPEN files.

026D-0276  621-630  Secondary address table

0277-0280  631-640  Keyboard buffer. If characters are POKEd in
                    here and location 198 (# of characters in
                    buffer) is changed, it will be as if the
                    characters were typed from the keyboard.

0281-0282  641-642  Start of memory pointer.

0283-0284  643-644  Top of memory pointer.

0286       646      Current color code. This holds the color
                    number that goes into color memory
                    during PRINT operations.

0288       648      Screen memory page. If you want the
                    operating system to know where screen
                    memory is, this must be changed as well
                    as the VIC chip.

 0289       649      Maximum size of keyboard buffer. If this
                    is set greater than 10, vital pointers will
                    be destroyed.

 028A       650      Keyboard repeat flag. If this is a 0, only
                    cursor controls repeat; if 128, all keys
                    repeat.

 028B       651      This determines how long the VIC waits
                    before repeating key.

 028D       653      Keyboard SHIFT, CTRL, Commodore flag. The SHIFT
                    sets the 1 bit, Commodore sets the 2 bit, and
                    the CTRL sets the 4 bit.
```

```
 HEX        DECIMAL  DESCRIPTION
 0291        657    Setting this location to 128 will disable
                   switching case, and a 0 here enables
                   the ability to switch.

 0300-0301   768-769  This is the jump vector for errors. By
                     changing this vector, a routine can
                     intercept any error condition.

 033C-03FB   828-1019 Cassette buffer. This is where data files
                     are held before they are INPUT. When
                     not using files, this is available for
                     POKEing or machine language programs.
```

```
 --
 /*  Sam Laur slaur@utu.fi  */
 /*   Go finger yourself!   */
 -
```