

# VIC 20 Memory Map

www.commodore.ca  
Free for personal use but you must have written permission to reproduce

Jim Butterfield, Toronto, Ont.

0000-0002	0-2	USR jump	009C	156	Byte-received flag	0287	647	Colour under cursor
0003-0004	3-4	Float-fixed vector	009D	157	Direct = \$80/RUN = 0 output control	0288	648	Screen memory page
0005-0006	5-6	Fixed-Float vector	009E	158	Tp Pass 1 error log/char buffer	0289	649	Max size of keyboard buffer
0007	7	Search character	009F	159	Tp Pass 2 err log corrected	028A	650	Repeat all keys
0008	8	Scan-quotes flag	00A0-00A2	160-162	Jiffy Clock HMTL	028B	651	Repeat speed cursor
0009	9	TAB column sav	00A3	163	Serial bit count/EOL flag	028C	652	Repeat delay counter
000A	10	0=LOAD, 1=VERIFY	00A4	164	Cycle count	028D	653	Keyboard Shift/Control flag
000B	11	Input buffer pointer/* subscript	00A5	165	Countdown.tape write/bit count	028E	654	Last shift pattern
000C	12	Default DIM flag	00A6	166	Tape buffer pointer	028F-0290	655-656	Keyboard table setup pointer
000D	13	Type: FF = string, 0 = numeric	00A7	167	Tp Wrt ldr count/Rd pass/inbit	0291	657	Keymode (Kattacanna)
000E	14	Type: 80 = integer, 00 = floating point	00A8	168	Tp Wrt new byte/Rd error/inbit cnt	0292	658	0 = scroll enable
000F	15	DATA scan/LIST quote/memry flag	00A9	169	Wrt start bit/Rd bit err/status	0293	659	VIC chip control
0010	16	Subscript/Fnx flag	00AA	170	Tp Scan;Cnt;Ld;End;byte assy	0294	660	VIC chip command
0011	17	0=INPUT;\$40 = GET;\$98 = READ	00AB	171	Wr lead length/Rd checksum/parity	0295-0296	661-662	Bit timing
0012	18	ATN sign/Comparison eval flag	00AC-00AD	172-173	Pointer: tape buf, scrolling	0297	663	RS-232 status
0013	19	Current I/O prompt flag	00AE-00AF	174-175	Tape end adds/End of program	0298	664	* bits to send
0014-0015	20-21	Integer value	00B0-00B1	176-177	Tape timing constants	0299-029A	665-666	RS-232 speed/code
0016	22	Pointer: temporary strg stack	00B2-00B3	178-179	Pointer: Start of Tape Buffer	029B	667	RS232 receive pointer
0017-0018	23-24	Last temp string vector	00B4	180	1 = Tp timer enabled; bit cnt	029C	668	RS232 input pointer
0019-0021	25-33	Stack for temporary strings	00B5	181	Tp EOT/RS232 next bit to send	029D	669	RS232 transmit pointer
0022-0025	34-37	Utility pointer area	00B6	182	Read character error/outbyte buf	029E	670	RS232 output pointer
0026-002A	38-44	Product area for multiplication	00B7	183	* characters in file name	029F-02A0	671-672	IRQ save during tape I/O
002B-002C	43-44	Pointer: Start of BASIC	00B8	184	Current logical file	0300-0301	768-769	Error message link
002D-002E	45-46	Pointer: Start of Variables	00B9	185	Current secndy address	0302-0303	770-771	BASIC warm start link
002F-0030	47-48	Pointer: Start of Arrays	00BA	186	Current device	0304-0305	772-773	Crunch BASIC tokens link
0031-0032	49-50	Pointer: String Storage (moving down)	00BB-00BC	187-188	Pointer to file name	0306-0307	774-775	Print tokens link
0033-0034	51-52	Pointer: Utility String	00BD	189	Wr shift word/Rd input char	0308-0309	776-777	Start new BASIC code link
0035-0036	53-54	Pointer: Utility String	00BE	190	* blocks remaining to Wr/Rd	030A-030B	778-779	Get arithmetic element link
0037-0038	55-56	Pointer: Limit of Memory	00BF	191	Serial word buffer	030C-0313	780-787	Unused
0039-003A	57-58	Current BASIC line number	00C0	192	Tape motor interlock	0314-0315	788-789	Hardware interrupt vector (EABF)
003B-003C	59-60	Previous BASIC line number	00C1-00C2	193-194	I/O start adds	0316-0317	790-791	Break interrupt vector (FED2)
003D-003E	61-62	Pointer: BASIC statement for CONT	00C3-00C4	195-196	Kernal setup pointer	0318-0319	792-793	NMI interrupt vector (FEAD)
003F-0040	63-64	Current DATA line number	00C5	197	Last key pressed	031A-031B	794-795	OPEN vector (F40A)
0041-0042	65-66	Current DATA address	00C6	198	* chars in keyboard buffer	031C-031D	796-797	CLOSE vector (F34A)
0043-0044	67-68	Input vector	00C7	199	Screen reverse flag	031E-031F	798-799	Set-input vector (F2C7)
0045-0046	69-70	Current variable name	00C8	200	End-of-line for input pointer	0320-0321	800-801	Set-output vector (F309)
0047-0048	71-72	Current variable address	00C9-00CA	201-202	Input cursor log (row, column)	0322-0323	802-803	Restore I/O vector (F3F3)
0049-004A	73-74	Variable pointer for FOR/NEXT	00CB	203	Which key: 64 if no key	0324-0325	804-805	INPUT vector (F20E)
004B-004C	75-76	Y-save; op-save; BASIC pointer save	00CC	204	0 = flash cursor	0326-0327	806-807	Output vector (F27A)
004D	77	Comparison symbol accumulator	00CD	205	Cursor timing countdown	0328-0329	808-809	Test-STOP vector (F770)
004E-0053	78-83	Misc work area, pointers, etc	00CE	206	Character under cursor	032A-032B	810-811	GET vector (F1F5)
0054-0056	84-86	Jump vector for functions	00CF	207	Cursor in blink phase	032C-032D	812-813	Abort I/O vector (F3EF)
0057-0060	87-96	Misc numeric work area	00D0	208	Input from screen/from keyboard	032E-032F	814-815	USR vector (FED2)
0061	97	Accum#1: Exponent	00D1-00D2	209-210	Pointer to screen line	0330-0331	816-817	LOCAL link (F549)
0062-0065	98-101	Accum#1: Mantissa	00D3	211	Position of cursor on above line	0332-0333	818-819	SAVE link (F685)
0066	102	Accum#1: Sign	00D4	212	0 = direct cursor, else programmed	033C-033B	828-1019	Cassette buffer
0067	103	Series evaluation constant pointer	00D5	213	Current screen line length	03FC-03FF	1020-1023	Unused
0068	104	Accum#1: hi-order (overflow)	00D6	214	Rw where cursor lives	0400-0FFF	1024-4095	3K RAM expansion area
0069	105	Accum#2: Exponent	00D7	215	Last inkey/checksum/buffer	1000-10FF	4096-7679	Normal BASIC memory
006A-006D	106-109	Accum#2: Mantissa	00D8	216	* of INSERTs outstanding	1E00-1FF9	7680-8191	Normal Screen memory
006E	110	Accum#2: Sign	00D9-00FD	217-240	Screen line link table	1000-11F9	4096-4601	Screen memory w/expansion
006F	111	Sign comparison, Acc#1 vs #2	00F1	241	Dummy screen link	1200 -	4608 -	BASIC memory w/expansion
0070	112	Accum#1: lo-order (rounding)	00F2	242	Screen row marker	2000-7FFF	8192-32767	Memory expansion area
0071-0072	113-114	Cassette buff len/Series pointer	00F3-00F4	243-244	Screen color pointer	8000-8FFF	32768-36863	Character bit maps
0073-007A	115-138	CHRGET subroutine; get BASIC char	00F5-00F6	245-246	Keyboard pointer	9000-900F	36864-36879	Video Interface Chip
007A-007B	122-123	BASIC pointer (within subtrn)	00F7-00F8	247-248	RS-232 Rcv ptr	9110-912F	37136-37151	VIA Interface - NMI
008B-008F	139-143	RND seed value	00F9-00FA	249-250	RS-232 Tx ptr	9120-912F	37152-37167	VIA Interface - IRQ
0090	144	Status word ST	00FF-010A	256-266	Floating to ASCII work area	9400-95FF	37888-38399	Alternate Colour Nybble area
0091	145	Keyswitch PIA: STOP and RVS flags	0100-013E	256-318	Tape error log	9600-97FF	38400-38911	Main Colour Nybble area
0092	146	Timing constant for tape	0100-01FF	256-511	Processor stack area	A000-BFFF	40960-49151	Plug-in ROM area
0093	147	Load = 0, Verify = 1	0200-0258	512-600	BASIC input buffer	C000-FFFF	49152-65535	ROM: BASIC and Operating System
0094	148	Serial output: deferred char flag	0259-0262	601-610	Logical file table	FF8A-FFFF	65418-65525	Jump Table, Including
0095	149	Serial deferred character	0263-026C	611-620	Device * table	FFC5		Set Input channel
0096	150	Tape EOT received	026D-0276	621-630	Sec Adds table	FFC9		Set Output channel
0097	151	Register save	0277-0280	631-640	Keyboard buffer	FFCC		Restore default I/O channels
0098	152	How many open files	0281-0282	641-642	Start of BASIC Memory	FFCF		INPUT
0099	153	Input device, normally 0	0283-0284	643-644	Top of BASIC Memory	FFD2		PRINT
009A	154	Output CMD device, normally 3	0285	645	Serial bus timeout flag	FFE1		Test Stop key
009B	155	Tape character parity	0286	646	Current colour code	FFE4		GET

## VIC 20 ROM Routines

C000	ROM control vectors	CD1E	Perform [NEXT]	D824	Perform [POKE]	E30B	Perform [ATN]	EDA3	Control key matrix	F675	SAVE program
C00C	Keyword action vectors	CD78	Type-match check	D82D	Perform [WAIT]	E378	Initialize	EDE4	VIC chip defaults	F728	SAVING
C050	Function vectors	CD9E	Evaluate expression	D849	Add 0.5	E387	CHRGET for zero page	EDFD	Screen line adds low	F734	Bump clock
C080	Operator vectors	CEA8	Constant - Pi	D850	Subtract-from	E3A4	Initialize BASIC	EE14	Send 'talk'	F760	Get time
C09E	Keywords	CEFF	Evaluate within brackets	D853	Perform [SUBTRACT]	E429	Power-up message	EE17	Send 'listen'	F767	Set time
C19E	Error messages	CEFF	Check for ?	D86A	Perform [ADD]	E44F	Vectors for \$300	EE1C	Send control char	F770	Action stop keys
C328	Error message vectors	CEFF	Check for comma	D947	Complement lac*1	E45B	Initialize vectors	EE19	Send to serial bus	F77E	File Error Messages
C365	Miscellaneous messages	CF08	Syntax error	D97E	OVERFLOW	E467	Warm restart	EEB7	Timeout on serial	F7AF	Find any tape location
C38A	Scan stack for FOR/GOSUB	CF14	Check range	D983	Multiply by zero byte	E476	Program patch area	EED0	Send listen SA	F7E7	Write tape header
C3B8	Move memory	CF28	Search for variable	D9EA	Perform [LOG]	E4A0	Serial output '1'	EED5	Clear ATN	F84D	Get buffer address
C3FB	Check stack depth	CFA7	Set up FN reference	DA2B	Perform [MULTIPLY]	E4A9	Serial output '0'	EEDC	Send talk SA	F854	Set buffer start and pointers
C408	Check memory space	CFE6	Perform [OR]	DA59	Multiply-a-bit	E4BC	Get serial input & clock	EED4	Send serial deferred	F867	Find specific header
C435	'OUT OF MEMORY'	CFE9	Perform [AND]	DA8C	Memory to FAC*2	E4BC	Program patch area	EED6	Send 'untalk'	F88A	Bump tape pointer
C437	Error routine	D016	Compare	DA87	Adjust FAC*1/*2	E500	Set 6522 address	EED4	Send 'unlisten'	F894	PRESS FLA
C469	Break entry	D081	Perform [DIM]	DAE4	Underflow/overflow	E505	Set screen limits	EED4	Receive from serial bus	F8A7	Check cassette status
C474	'READY'	D09B	Locate variable	DAE2	Multiply by 10	E518	Track cursor location	EF84	Check line on	F8B7	PRESS RECORD
C480	Ready for BASIC	D113	Check alphabetic	DAF9	-10 in floating pt	E518	Initialize I/O	EF8D	Close line off	F8C0	Initiate tape read
C49C	Handle new line	D11D	Create variable	DAFE	Divide by 10	E54C	Normalize screen	EF96	Delay 1 ms	F8E3	Initiate tape write
C533	Re-chain lines	D194	Array pointer subroutine	DB12	Perform [DIVIDE]	E55F	Clear screen	EFA3	RS232 send (NMI)	F8F4	Common tape read/write
C560	Receive input line	DA15	Value 32768	DBA2	Memory to fac*1	E581	Home cursor	EFE8	New RS232 bit send	F94B	Check tape stop
C579	Crunch tokens	DB12	Float-fixed conversion	DBC7	FAC*1 to memory	E587	Set screen pointers	F016	Error or quit	F95D	Set timing
C613	Find BASIC line	DD11	Set up array	DBFC	FAC*2 to fac*1	E5BB	Set I/O defaults	F027	Compute bit count	F98E	Read bits (IRQ)
C642	Perform [NEW]	DD25	'BAD SUBSCRIPT'	DDCC	FAC*1 to FAC*2	E5C3	Set VIC chip defaults	F036	RS232 receive (NMI)	FAAD	Store characters
C65E	Perform [CLR]	DD48	'ILLEGAL QUANTITY'	DD1B	Round FAC*1	E5CF	Input from keyboard	F05B	Setup to receive	FBD2	Reset pointer
C68E	Back up text pointer	DD4C	Compute array size	DD2B	Get sign	E648	Input from screen	F09D	Receive parity error	FBD8	New tape character setup
C69C	Perform [LIST]	DD7D	Perform [FRE]	DD39	Perform [SGN]	E668	Quote mark test	F0A2	Receive overrun error	FBEA	Toggle tape
C742	Perform [FOR]	DD91	Fixed-float conversion	DD58	Perform [ABS]	E6C5	Set up screen print	F0A5	Receive break error	FC06	Data write
CD7D	Execute statement	D39E	Perform [POS]	DD5B	Compare FAC*1 to mem	E6EA	Advance cursor	F0A8	Receive frame error	FC0B	Leader write (IRQ)
CD1D	Perform [RESTORE]	DDA6	Check direct	DD9B	Float-fixed	E715	Retreat cursor	F0B9	Bad device	F089	Bad device
C82C	Break	DD3B	Perform [DEF]	DDCC	Perform [INT]	E72D	Back into previous line	F0BC	File to RS232	FCFC	Restore vectors
C82F	Perform [STOP]	D3E1	Check FN syntax	DDCF	String to fac	E742	Output to screen	F0ED	Send to RS232 buffer	FCF6	Set vector
C831	Perform [END]	DD3F	Perform [FN]	DD7E	Get ASCII digit	E8C3	Go to next line	F116	Input from RS232 buffer	FD08	Kill motor
C857	Perform [CONT]	DD65	Perform [STR\$]	DDDD	Float to ASCII	E8D8	Do 'RETURN'	F14F	Get from RS232 buffer	FD11	Check read/write pointer
C871	Perform [RUN]	DD75	Calculate string vector	DD16	Decimal constants	E8E8	Check line decrement	F160	Check serial bus idle	FD1B	Bump read/write pointer
C883	Perform [GOSUB]	DD47	Set up string	DD3A	TI constants	E8FA	Check line increment	F174	Messages	F222	Powerup entry
C8A0	Perform [GOTO]	DD4F	Make room for string	DD71	Perform [SOR]	E912	Set colour code	F1E2	Print if direct	FD3F	Check A-ROM
C8D2	Perform [RETURN]	DD26	Garbage collection	DD7B	Perform [POWER]	E921	Colour code table	F1E2	Print if direct	FD52	Set Kernal
C8F8	Perform [DATA]	DD5D	Check salvageability	DD84	Perform [NEGATIVE]	E929	Code conversion	F205	Off on RS232	FD8D	Initialize system constants
C906	Scan for next statement	DD66	Collect string	DDFD	Perform [EXP]	E975	Scroll screen	F20E	Input	FD91	IRQ vectors
C928	Perform [IF]	DD3D	Concatenate	E040	Series evaluate 1	E9EE	Open space on screen	F250	Get .tape/serial/RS232	FD	