BASIC 4.0 / 2.0 Kernal Routines

CBM Label	Address Hex	Address Dec	Operation	Registers In .A	Registers In .X	Registers In .Y	Registers Out .A	Registers Out .X	Registers Out .Y
CHKIN	FFC6	65478	Open channel for input		LF#		alt.		
CHKOUT	FFC9	65481	Open channel for output		LF#		alt.		
CHRIN	FFCF	65487	Input character from channel				data	alt.	
CHROUT	FFD2	65490	Output character to channel	data					
CLALL	FFE7	65511	Close all channels and files				alt.	alt.	
CLOSE	FFC3	65475	Close a specified logical file	LF#			alt.	alt.	alt.
CLRCHN	FFCC	65484	Restore default I/O devices				alt.	alt.	
CSYS	FFDE	65502	SYS vector		addr lo	addr hi	alt.	alt.	alt.
CVERF	FFDB	65499	Verify ram from a device		start lo	start hi		end lo + 1	end hi
GETIN	FFE4	65508	Get character from current input device				data	alt.	alt.
LOAD	FFD5	65493	Load ram from a device		start lo	start hi		end lo + 1	end hi
OPEN	FFC0	65472	Open a logical file				alt.	alt.	alt.
SAVE	FFD8	65496	Save 'ram' to device	txttab lo	start lo	start hi		end lo + 1	end hi
STOP	FFE1	65505	Scan stop key depressed	yes: .Z = 1, no .A = last row kybd scan					
UDTIM	FFEA	65514	Increment real time clock				alt.	alt.	

alt. = altered

VIC 20 And Commodore 64 Kernal Routines

CBM Label	Address Hex	Address Dec	Operation	Registers In .A	Registers In .X	Registers In .Y	Registers Out .A	Registers Out .X	Registers Out .Y
ACPTR	FFA5	65445	Input byte from Serial Port				data	alt.	
CHKIN	FFC6	65478	Open channel for input		LF#		alt.		
CHKOUT	FFC9	65481	Open channel for output		LF#		alt.		
CHRIN	FFCF	65487	Input character from channel				data	alt.	
CHROUT	FFD2	65490	Output character to channel	data					
CIOUT	FFA8	65448	Output byte to serial port	data					
CINT	FF81	65409	Initialize screen editor				alt.	alt.	alt.
CLALL	FFE7	65511	Close all channels and files				alt.	alt.	
CLOSE	FFC3	65475	Close a specified logical file	LF#			alt.	alt.	alt.
CLRCHN	FFCC	65484	Restore default I/O devices				alt.	alt.	
GETIN	FFE4	65508	Get character from current input device				data	alt.	alt.
IOBASE	FFF3	65523	Returns base address of I/O devices					addr lo	addr hi
IOINIT	FF84	65412	Initialize Input/Output				alt.	alt.	alt.
LISTEN	FFB1	65457	Command devices on the serial bus to listen	DEV#					
LOAD	FFD5	65493	Load (.A = 0) or Verify (.A = 1) 'ram' from a device		start lo	start hi		end lo + 1	end hi
MEMBOT	FF9C	65436	Read (.C = 1) or Set (.C = 0) the bottom of memory	.C = 0:	bot lo	bot hi	.C = 1:	bot lo	bot hi
MEMTOP	FF99	65433	Read (.C = 1) or Set (.C = 0) the top of memory	.C = 0:	top lo	top hi	.C = 1:	top lo	top hi
OPEN	FFC0	65472	Open a logical file				alt.	alt.	alt.
PLOT	FFF0	65520	Read (C = 1) or Set (C = 0) x, y cursor position		row	col		row	col
RAMTAS	FF87	65415	Init. ram, allocate tape buff, set screen $0400				alt.	alt.	alt.
RDTIM	FFDE	65502	Read real time clock				msb	msb2	lsb
READST	FFB7	65463	Read I/O status word				ST		
RESTOR	FF8A	65418	Restore default I/O vectors				alt.	alt.	alt.
SAVE	FFD8	65496	Save 'ram' to device	txttab lo	start lo	start hi		end lo + 1	end hi
SCNKEY	FF9F	65439	Scan keyboard				alt.	alt.	alt.
SCREEN	FFED	65517	Return screen size in rows & columns					#rows	#cols
SECOND	FF93	65427	Send secondary address after 'listen'	SA OR $60					
SETLFS	FFBA	65466	Set logical, first, and second addresses	LF#	DEV#	SA			
SETMSG	FF90	65424	Enable/Disable 'Kernal' messages	.A val: $40 control msgs on. $80 error msgs on. $00 off					
SETNAM	FFBD	65469	Set file name	len	addr lo	addr hi			
SETTIM	FFDB	65499	Set real time clock	msb	msb2	lsb			
SETTMO	FFA2	65442	Set (.A<#128) Reset (.A>#127) IEEE timeout						
STOP	FFE1	65505	Scan stop key depressed	yes: .Z = 1, no .A = last row kybd scan					
TALK	FFB4	65460	Command serial bus device to 'talk'	DEV#					
TKSA	FF96	65430	Send secondary address after 'talk'	SA					
UDTIM	FFEA	65514	Increment real time clock				alt.	alt.	
UNLSN	FFAE	65454	Command serial bus to 'unlisten'				alt.		
UNTLK	FFAB	65451	Command serial bus to 'untalk'				alt.		
VECTOR	FF8D	65421	Store (.C = 1) or Restore (.C = 0) ram vectors	.C = 1:	tabl lo	tabl hi	.C = 0:	tabl lo	tabl hi

alt. = altered

User Callable ROM Subroutines

Some I/O routines require extra memory set up. See the appropriate Memory Map. Data within parenthesis are for Basic 2.0/4.0 users. (Direct call) indicates no required set up.

#	Entry Point For: 2.0		4.0		VIC 20		C64		Operation	Registers In .A	.X	.Y	Registers Out .A	.X	.Y
1	C2D8	49880	B350	45904	C3BB	50107	A3BB	41915	Open Up Space In BASIC Text New:	AryTop Lo		AryTop Hi	Unaltered		
2	C328	49960	B3A0	45984	C408	50184	A408	41992	Check Available Memory (called by 1)	(same as above) Start address of move in $5F, 60 (5C, 5D)					
3	C355	50005	B3CD	46029	C435	50229	A435	42037	?OUT OF MEMORY	(direct call)					
4	C357	50007	BC3F	48191	C437	50231	A437	42039	Send BASIC Error Message	Error #					
5	C389	50057	B3FF	46079	C474	50292	A474	42100	Warm start, BASIC	(direct call)					
6	C399	49960	B40D	46093	C48A	50314	A48A	42122	Main CHRGET entry	(direct call) $7A = #$FF, $7B = #$01 ($77, 78);01FF = Basic Inbuf-1					
7	C3AB	50091	B41F	46111	C49C	50220	A49C	42028	Crunch tokens, insert line		Inbuf len.				
8	C439	50233	B4AD	46253	C52A	50474	A52A	42282	Fix chaining, CLR, & READY.	(direct call)					
9	C442	50242	B4B6	46262	C533	50483	A533	42291	Fix chaining	(direct call)					
10	C46F	50287	B4E2	46306	C560	50528	A560	42336	Receive line from keyboard	(direct call) $7A = #$FF, $7B = #$01					
11	C495	50213	B4FB	46331	C579	50553	A579	42361	Crunch tokens (called by 7)	.X = Inbuf Len. ($0200,X) = #$00					
12	C52C	50476	B5A3	46499	C613	50707	A613	42515	Find line in BASIC	StrtBAS Lo	StrtBAS Hi				
13	C55D	50525	B5D4	46548	C642	50754	A642	42562	Do NEW	(direct call)					
14	C572	50546	B5E9	46569	C659	50777	A659	42585	Reset BASIC and do CLR	(direct call)					
15	C575	50549	B5EC	46572	C65E	50782	A65E	42590	Do CLR	(direct call)					
16	C5A7	50599	B622	46626	C68E	50830	A68E	42638	Reset Chrget to Start of BASIC	(direct call)			StrtBAS Hi		
17	C6C4	50884	B74A	46922	C857	51287	A857	43095	Continue BASIC execution [CONT]	CurLin Lo		CurLin Hi			
18	C873	51315	B8F6	47350	C96B	49771	A96B	41579	Get fixed-pt number from BASIC text	Address of text in Chrget ptr; $7A, 7B ($77, 78)					
19	C9DE	49886	BADB	47835	CAD3	51923	AAD3	43731	Send RETURN, LF if in screen mode	(direct call)			LF (#$0A)		
20	C9E2	49890	BADF	47839	CAD7	51927	AAD7	43735	Send RETURN, LINEFEED	(direct call)			LF (#$0A)		
21	CA1C	51740	BB1D	47901	CB1E	51998	AB1E	43806	Print string from A, Y	Addr Lo		Addr Hi			
22	CA22	51746	BB23	47907	CB24	52004	AB24	43812	Print pre-computed string	Length	Addr in $22,23 ($1F,20)				
23	CA43	51779	BB44	47940	CB45	52037	AB45	43845	Print '?'	(direct call)					
24	CA45	51781	BB46	47942	CB47	52039	AB47	43847	Print char (output .A to device)	Char			Char		
25	CC9F	52383	BD98	48536	CD9E	52638	AD9E	44446	Evaluate Result string $0D = #$FF Expression ($07) numeric $0D = #$00	Address of Expression In Chrget Pointer			Addr Lo result in Acc#1		Addr Hi
26	CDF8	52728	BEF5	48885	CEFF	52991	AEFD	44797	Check for comma	(direct call)			Char		
27	CDF7	52727	BEF2	48882	CEFA	52986	AEFA	44794	Check for '('	(direct call)			Char		
28	CDF4	52724	BEEF	48879	CEF7	52983	AEF7	44791	Check for ')'	(direct call)			Char		
29	CE03	52739	BF00	48896	CF08	53000	AF08	44808	Send 'SYNTAX ERROR'	(direct call)					
30	CFC9	53193	C187	49543	D0E7	53479	B0E7	45287	Find fl-pt variable, given name				VarAddr Lo		VarAddr Hi
31	D069	53353	C2B9	49849	D185	53637	B185	45445	Bump Variable Addr by 2 (called by 30)	Name in $45, 46 ($42, 43)			VarAddr Lo		VarAddr Hi
32	D09A	53290	C2EA	49898	D1BF	53695	B1BF	45503	Float to Fixed conversion in Acc#1	(direct call)					
33	D26D	53869	C4BC	50364	D391	54049	B391	45857	Fixed to Float conversion in Acc#1	(direct call)					
34	D67B	54907	C8D7	51415	D79E	55086	B79E	46894	Get Acc#1 least significant byte to X register					Data	
35	D68F	54927	C8EB	51435	D7B5	55221	B7B5	47029	Evaluate string [VAL]	Address = (Chrget Ptr.)			Fl. Pt. result in Acc#1		
36	D69D	54931	C8EF	51439	D7B9	55225	B7B9	47033	Evaluate string from X, Y (above + 4)		Addr Lo	Addr Hi	Fl. Pt. result in Acc#1		
37	D6C6	54982	C921	49697	D7EB	55275	B7EB	47083	Get two params for POKE, WAIT	Address = (Chrget Ptr.)			.X = Pram2, Pram1 in Acc#1 (sig pt)		
38	D773	55155	C99D	49709	D867	55399	B867	47207	Add (from memory)	Addr Lo		Addr Hi	Fl. Pt. result in Acc#1		
39	D934	53812	CB5E	52062	DA28	55848	BA28	47656	Multiply from memory location	Addr Lo		Addr Hi	Fl. Pt. result in Acc#1		
40	D9EE	53998	CC18	52248	DAE2	56034	BAE2	47842	Multiply Acc#1 by ten				(result in Acc#1)		
41	DAAE	55982	CCD8	52440	DBA2	56226	BBA2	48034	Unpack memory variable to Acc#1	Addr Lo		Addr Hi			
42	DAE3	56035	CD0D	52493	DBD7	56279	BBD7	48087	Copy Acc#1 to (X,Y) Location	Addr Lo	Addr Hi				
43	DB08	56072	CD32	52530	DBFC	56316	BBFC	48124	Move Acc#2 to Acc#1	(direct call)					
44	DB18	56088	CD42	52546	DC0C	56332	BC0C	48140	Move Rounded Acc#1 to Acc#2	(direct call)					
45	DB1D	56093	CD45	52549	DC0F	56335	BC0F	48143	Move Un-Rounded Acc#1 to Acc#2	(direct call)					
46	DB27	56103	CD51	52561	DC1B	56347	BC1B	48155	Round Acc.#1	(direct call)					
47	DCD9	56537	CF83	53123	DDCD	56781	BDCD	48589	Print fixed-point value	Value Hi	Value Lo				
48	DCE3	56547	CF8D	53133	DDD7	56791	BDD7	48599	Print floating-point value in Acc#1	(direct call)					
49	DCE9	56553	CF93	53027	DDDD	56797	BDDD	48605	Cnvert num to strng at $0100 (calld by 47)	#$00		#$01			
50	FD11	64785	D472	54386	n/a	n/a	n/a	n/a	Entry to M.L.M.	(direct call)					
51	E3D8	58328	E202	57858	E742	59202	E716	59158	Print a character	Char					
52	F156	61782	F185	61829	F1E6	61926	F12F	61743	Print system message			Offset			
53	F0B6	61622	F0D2	61650	EE14	60948	ED09	60681	Send 'talk' to IEEE/Serial	Dev #					
54	F0BA	61626	F0D5	61653	EE17	60951	ED0C	60684	Send 'listen' to IEEE/Serial	Dev #					
55	F128	61736	F143	61763	FF93	65427	FF93	65427	Send secondary address	SA OR $60					
56	F16F	61807	F19E	61742	EEE4	61156	ED40	60736	Send char to IEEE/Serial	Char					
57	F17F	61823	F1AE	61870	EEF6	61174	EDEF	60911	Send 'untalk'	(direct call)					
58	F183	61827	F1B9	61881	EF04	61188	EDFE	60926	Send 'unlisten'	(direct call)					