

Die Codes des C 64

Die kompletten ASCII-, CHR\$- und Bildschirmcodes und dazu noch die vollständigen Tastencodes des C 64 finden Sie in der folgenden Tabelle anwenderfreundlich, also so, daß man sie auch benutzen kann, gegenübergestellt.

Programmieren in Maschinensprache beinhaltet oft Ein- und Ausgaben, zum Beispiel für eine Benutzerführung, oder um Eingaben vom Bildschirm einzulesen. Häufig steht man vor dem Problem, daß man nur einzelne Zeichen eines Textes in einem Maschinenprogramm verändern will. In allen Fällen benötigt man dann die CHR\$- oder Bildschirmcodes, von denen hier die Rede ist. Das gleiche gilt natürlich auch für die Tastatur. Das Betriebssystem stellt uns die momentan gedrückte Taste in zwei Speicherstellen zur Verfügung.

Warum verschiedene Codes?

Vergleicht man in der Tabelle 1 die Spalten, so wird man feststellen, daß es einige nicht druckende Zeichen gibt, Codes, die auf dem Bildschirm nichts schreiben. Es sind dies die ASCII- beziehungsweise die CHR\$-Codes von 0 bis 31 und von 128 bis 159. Auch findet man auf dem Bildschirm Zeichen, die in der ASCII-Tabelle nicht auftauchen – die reversen Zeichen mit den Codes von 128 bis 255. Um alle druckenden ASCII-Zeichen auch auf dem Bildschirm sichtbar zu machen, und dabei keine sinnlos leeren Felder mitzuführen, mußte man die Tabelle im Computer umstellen.

Ganz anders verhält es sich mit der Tastatur. Hier sieht die Codierung zunächst sehr zufällig aus. Natürlich ist dies mit der Anordnung der Tasten und der Methode, wie das Betriebssystem die Tasten abfragt, zu erklären. Uns soll es hier genügen, zu wissen, wie man von einem (Maschinen-) Programm aus einen Tastendruck registrieren kann. Man findet die zuletzt gedrückte Taste in den Speicherstellen 197 (\$C5) und 203 (\$CB), die mit PEEK beziehungsweise auf der Maschinensprache-Ebene mit LDA ausgelesen werden kann.

Die nachfolgende Tabelle 1 bietet auf einen Blick alle Codes, die im C 64 enthalten sind. Die Tabelle ist nach den CHR\$-Zeichen sortiert, also eine aufsteigende Reihe von 0 bis 255. Um das Programmieren in Maschinensprache zu erleichtern, sind neben den dezimalen Zahlen auch die hexadezimalen Werte aufgeführt.

Die Spalten drei und vier enthalten die CHR\$-Zeichen des C 64. Spalte drei zeigt dabei die Zeichen im Groß-/Grafikmodus, Spalte vier die entsprechenden Zeichen in Klein-/Großschriftmodus.

Die Zeichen von 0 bis 31 und von 128 bis 159 sind diejenigen CHR\$-Codes, deren Aufruf auf dem Bildschirm kein sichtbares Zeichen hervorbringt, so zum Beispiel das Umstellen der Cursorfarbe, oder die Codes der Funktionstasten, die man nicht schreiben kann (?CHR\$(133) ergibt nicht »F1«), aber für eine Abfrage (IF A\$ = CHR\$(133) THEN ...) nutzen kann. Hier wird nicht zwischen den Modi unterschieden, da das Ergebnis in beiden Fällen das gleiche ist. Damit der Text in die

Tabelle paßt, ließen wir ihn sogar bis in die sechste Spalte übergreifen. Über den Bildschirmcode kann man nicht über diese Funktionen verfügen!

Die Bildschirmcodes der fünften und sechsten Spalte sind die Werte, die man erhält, wenn man das CHR\$-Zeichen mit PRINT ausgibt und die entsprechende Bildschirmspeicherstelle mit PEEK ausliest. Für Maschinensprache ist auch hier wieder der hexadezimale Wert angegeben. Die reversen Zeichen erhält man durch Addition von 128, in Maschinensprache durch den Befehl »ORA # \$80«.

Obwohl der C 64 keinen echten ASCII-Zeichensatz hat, ist dieser in Spalte sieben aufgeführt. Man benötigt den ASCII-Code vor allem für die Datenfernübertragung (DFÜ) und zur Ansteuerung von nicht MPS-kompatiblen Druckern, die nicht mit einem »intelligenten« Interface betrieben werden. Die Erklärungen der verwendeten Kurzzeichen stehen in der Tabelle 2.

In den letzten beiden Spalten sind die Tastaturcodes enthalten, wie man sie beim Auslesen der Speicherstellen 197 und 203 erhält. Bitte beachten Sie, daß der Code 63 (\$3F) der RUN/STOP-Taste entspricht, also dem CHR\$-Code 3. Da aber PRINT CHR\$(3) ein Programm nicht anhält und sich nicht abfragen läßt, ist es in der Tabelle nicht aufgeführt.

(og)

NUL	= Null
SOH	= Start of Heading
STX	= Start of Text (Textbeginn)
ETX	= End of Text (Textende)
EOT	= End of Transmission (Übertragungsende)
ENQ	= Enquiry (Testanfrage)
ACK	= Acknowledge (Quittierung)
BEL	= Bell (Klingelzeichen)
BS	= Backspace (Zeichen zurück)
HT	= Horizontal Tabulation
LF	= Line Feed (Zeilenvorschub)
VT	= Vertical Tabulation
FF	= Form Feed (Formatanpassung)
CR	= Carriage Return (Wagenrücklauf)
SO	= Shift out (Shift aus)
SI	= Shift in (Shift ein)
DLE	= Data Link Escape (Datenverbindung abhalten)
DC	= Device Control (Gerätesteuerung)
NAK	= Negative Acknowledge (Gegenquittierung)
SYN	= Synchronous Idle (synchronlos)
ETB	= End of Transmission Block (Block Übertragungsende)
CAN	= Cancel (Stornieren, zum Beispiel Puffer leeren)
EM	= End of Medium (z. B. Papierende)
SUB	= Substitute (Austausch)
ESC	= Escape (übergehen, Druckercode)
FS	= File Separator (Filetrennung)
GS	= Group Separator (Trennung von Gruppen)
RS	= Record Separator (Aufzeichnung trennen)
US	= Unit Separator (Einheitentrenner)
SP	= Space (Leerzeichen)
DEL	= Delete (Löschen)

Tabelle 2. Erklärung der in Tabelle 1 abgekürzten ASCII-Zeichen

Wert		CHR\$-Zeichen		Bildschirmcode		ASCII-Code	Tastencode	
dez	hex	Groß/Grafik	Klein/Groß	dez	hex		dez	hex
0	00					NUL		
1	01					SOH		
2	02					STX		
3	03					ETX	63	3F
4	04					EOT		
5	05	weiß				ENQ		
6	06					ACK		
7	07					BEL		
8	08	blockiert SHIFT + C entriegelt SHIFT + C				BS		
9	09					HT		
10	0A					LF		
11	0B					VT		
12	0C					FF		
13	0D	Carriage RETURN Umschaltung auf Kleinschrift				CR	1	01
14	0E					SO		
15	0F					SI		
16	10					DLE		
17	11	Cursor down Revers ein				DC1	7	07
18	12					DC2		
19	13	HOME (Cursor nach oben links) DEL (ein Zeichen löschen)				DC3	51	33
20	14					DC4	0	00
21	15					NAK		
22	16					SYN		
23	17					ETB		
24	18					CAN		
25	19					EM		
26	1A					SUB		
27	1B					ESC		
28	1C	rot Cursor right				FS		
29	1D					GS	2	02
30	1E	grün				RS		
31	1F	blau				US		
32	20			32	20	SP	60	3C
33	21	!	!	33	21	!		
34	22	"	"	34	22	"		
35	23	#	#	35	23	#		
36	24	\$	\$	36	24	\$		
37	25	%	%	37	25	%		
38	26	&	&	38	26	&		
39	27	'	'	39	27	'		
40	28	((40	28	(
41	29))	41	29)		
42	2A	*	*	42	2A	*	49	31
43	2B	+	+	43	2B	+	40	28
44	2C	,	,	44	2C	,	47	2F
45	2D	-	-	45	2D	-	43	2B
46	2E	.	.	46	2E	.	44	2C
47	2F	/	/	47	2F	/	55	37
48	30	0	0	48	30	0	35	23
49	31	1	1	49	31	1	56	38
50	32	2	2	50	32	2	59	3B
51	33	3	3	51	33	3	8	08
52	34	4	4	52	34	4	11	0B
53	35	5	5	53	35	5	16	10
54	36	6	6	54	36	6	19	13
55	37	7	7	55	37	7	24	18
56	38	8	8	56	38	8	27	1B
57	39	9	9	57	39	9	32	20
58	3A	:	:	58	3A	:	45	2D
59	3B	;	;	59	3B	;	50	32
60	3C	<	<	60	3C	<		
61	3D	=	=	61	3D	=	53	35
62	3E	>	>	62	3E	>		
63	3F	?	?	63	3F	?		
64	40	@	@	0	00	@	46	2E
65	41	A	a	1	01	A	10	0A
66	42	B	b	2	02	B	28	1C
67	43	C	c	3	03	C	20	14
68	44	D	d	4	04	D	18	12
69	45	E	e	5	05	E	14	0E
70	46	F	f	6	06	F	21	15
71	47	G	g	7	07	G	26	1A
72	48	H	h	8	08	H	29	1D
73	49	I	i	9	09	I	33	21
74	4A	J	j	10	0A	J	34	22
75	4B	K	k	11	0B	K	37	25
76	4C	L	l	12	0C	L	42	2A
77	4D	M	m	13	0D	M	36	24
78	4E	N	n	14	0E	N	39	27
79	4F	O	o	15	0F	O	38	26
80	50	P	p	16	10	P	41	29
81	51	Q	q	17	11	Q	62	3E
82	52	R	r	18	12	R	17	11
83	53	S	s	19	13	S	13	0D
84	54	T	t	20	14	T	22	16
85	55	U	u	21	15	U	30	1E
86	56	V	v	22	16	V	31	1F
87	57	W	w	23	17	W	9	09
88	58	X	x	24	18	X	23	17
89	59	Y	y	25	19	Y	25	19
90	5A	Z	z	26	1A	Z	12	0C
91	5B	[[27	1B	[
92	5C	\	\	28	1C	\	48	30
93	5D]]	29	1D]		
94	5E	^	^	30	1E	^	54	36
95	5F	_	_	31	1F	_	57	39
96	60	~	~	64	40	~		
97	61	A	A	65	41	a		
98	62	B	B	66	42	b		
99	63	C	C	67	43	c		

Wert		CHR\$-Zeichen		Bildschirmcode		ASCII-Code	Tastencode	
dez	hex	Groß/Grafik	Klein/Groß	dez	hex		dez	hex
100	64	—	D	68	44	d		
101	65	—	E	69	45	e		
102	66	—	F	70	46	f		
103	67	—	G	71	47	g		
104	68	—	H	72	48	h		
105	69	—	I	73	49	i		
106	6A	—	J	74	4A	j		
107	6B	—	K	75	4B	k		
108	6C	—	L	76	4C	l		
109	6D	—	M	77	4D	m		
110	6E	—	N	78	4E	n		
111	6F	—	O	79	4F	o		
112	70	—	P	80	50	p		
113	71	—	Q	81	51	q		
114	72	—	R	82	52	r		
115	73	—	S	83	53	s		
116	74	—	T	84	54	t		
117	75	—	U	85	55	u		
118	76	—	V	86	56	v		
119	77	—	W	87	57	w		
120	78	—	X	88	58	x		
121	79	—	Y	89	59	y		
122	7A	—	Z	90	5A	z		
123	7B	—	+	91	5B	[
124	7C	—	=	92	5C	\		
125	7D	—	—	93	5D]		
126	7E	—	—	94	5E	↑		
127	7F	—	—	95	5F	DEL		
128	80	orange						
129	81							
130	82							
131	83							
132	84							
133	85	f1					4	04
134	86	f3					5	05
135	87	f5					6	06
136	88	f7					3	03
137	89	f2						
138	8A	f4						
139	8B	f6						
140	8C	f8						
141	8D	SHIFT RETURN Umschaltung auf Groß/Grafik						
142	8E							
143	8F							
144	90	schwarz						
145	91	Cursor up						
146	92	Revers aus						
147	93	CLR (Bildschirm löschen)						
148	94	INSERT (Zeichen einfügen)						
149	95	braun						
150	96	hellrot						
151	97	grau 1						
152	98	grau 2						
153	99	hellgrün						
154	9A	hellblau						
155	9B	grau 3						
156	9C	lila						
157	9D	Cursor left						
158	9E	gelb						
159	9F	cyan						
160	A0			96	60			
161	A1	—	—	97	61			
162	A2	—	—	98	62			
163	A3	—	—	99	63			
164	A4	—	—	100	64			
165	A5	—	—	101	65			
166	A6	—	—	102	66			
167	A7	—	—	103	67			
168	A8	—	—	104	68			
169	A9	—	—	105	69			
170	AA	—	—	106	6A			
171	AB	—	—	107	6B			
172	AC	—	—	108	6C			
173	AD	—	—	109	6D			
174	AE	—	—	110	6E			
175	AF	—	—	111	6F			
176	B0	—	—	112	70			
177	B1	—	—	113	71			
178	B2	—	—	114	72			
179	B3	—	—	115	73			
180	B4	—	—	116	74			
181	B5	—	—	117	75			
182	B6	—	—	118	76			
183	B7	—	—	119	77			
184	B8	—	—	120	78			
185	B9	—	—	121	79			
186	BA	—	—	122	7A			
187	BB	—	—	123	7B			
188	BC	—	—	124	7C			
189	BD	—	—	125	7D			
190	BE	—	—	126	7E			
191	BF	—	—	127	7F			
192	C0	—	—	64	40			
193	C1	—	—	65	41			
194	C2	—	—	66	42			
195	C3	—	—	67	43			
196	C4	—	—	68	44			

Tabelle 1. Die Codes des C64

Wert		CHR\$-Zeichen		Bildschirmcode		ASCII-Code	Tastencode	
dez	hex	Groß/Grafik	Klein/Groß	dez	hex		dez	hex
197	C5	—	E	69	45			
198	C6	—	F	70	46			
199	C7	—	G	71	47			
200	C8	—	H	72	48			
201	C9	—	I	73	49			
202	CA	—	J	74	4A			
203	CB	—	K	75	4B			
204	CC	—	L	76	4C			
205	CD	—	M	77	4D			
206	CE	—	N	78	4E			
207	CF	—	O	79	4F			
208	D0	—	P	80	50			
209	D1	—	Q	81	51			
210	D2	—	R	82	52			
211	D3	—	S	83	53			
212	D4	—	T	84	54			
213	D5	—	U	85	55			
214	D6	—	V	86	56			
215	D7	—	W	87	57			
216	D8	—	X	88	58			
217	D9	—	Y	89	59			
218	DA	—	Z	90	5A			
219	DB	—	+	91	5B			
220	DC	—	*	92	5C			
221	DD	—	—	93	5D			
222	DE	—	—	94	5E			
223	DF	—	—	95	5F			
224	E0	—	—	96	60			
225	E1	—	—	97	61			
226	E2	—	—	98	62			
227	E3	—	—	99	63			
228	E4	—	—	100	64			
229	E5	—	—	101	65			
230	E6	—	—	102	66			
231	E7	—	—	103	67			
232	E8	—	—	104	68			
233	E9	—	—	105	69			
234	EA	—	—	106	6A			
235	EB	—	—	107	6B			
236	EC	—	—	108	6C			
237	ED	—	—	109	6D			
238	EE	—	—	110	6E			
239	EF	—	—	111	6F			
240	F0	—	—	112	70			
241	F1	—	—	113	71			
242	F2	—	—	114	72			
243	F3	—	—	115	73			
244	F4	—	—	116	74			
245	F5	—	—	117	75			
246	F6	—	—	118	76			
247	F7	—	—	119	77			
248	F8	—	—	120	78			
249	F9	—	—	121	79			
250	FA	—	—	122	7A			
251	FB	—	—	123	7B			
252	FC	—	—	124	7C			
253	FD	—	—	125	7D			
254	FE	—	—	126	7E			
255	FF	—	—	94	5E			

Tabelle 1. Die Codes des C 64 (Schluß)

Befehlsübersicht - Zusammenfassung

Kennen Sie die Programme Hypra-Ass, Reassembler und SMON? Wenn ja, dann wird Ihnen die Befehlsübersicht bei der Arbeit mit diesen Programmen sehr hilfreich sein.

Selbst für den eingefleischten Profi ist es fast unmöglich alle Befehle von Hypra-Ass, Reassembler und SMON zu jeder Zeit parat zu haben. Um aber diese Programme optimal zu nutzen, ist die Kenntnis aller Befehle und deren Wirkungsweise sehr wichtig. Die nun folgende Befehlsübersicht soll Ihnen das ewige Nachschlagen ersparen. Natürlich kann es sich bei einer Übersicht nicht um eine ausführliche Anleitung handeln. Bei Verständnisschwierigkeiten finden Sie nähere Informationen an den entsprechenden Stellen in den Artikeln. (ak)

Quickreferenz Reassembler

P adresse	Einsprungspunkt durch Label markieren.
T adresse, adresse	Tabelle definieren.
E (byte)	Startet den Reassembler. Die einzelnen Bits des Bytes haben folgende Bedeutung:
Bit 0 gesetzt	Alle Zeropage-Adressen durch ein Label mit drei Buchstaben markieren.
Bit 1 gesetzt	Nach RTS, RTI, BRK, JMP Kommentarzeile einfügen.
Bit 2 gesetzt	Bei unmittelbarer Adressierung ASCII-Zeichen ausgeben.
Bit 3 gesetzt	Zwischen jede zweite Tabellenzeile Kommentarzeile einfügen.
Bit 4 gesetzt	Der ASCII-Ausdruck wird bei Tabellen unterdrückt.
Bit 5 gesetzt	Externe- und Tabellenlabel kennzeichnen.
Bit 6 gesetzt	Nach Tabellen suchen.
Bit 7 gesetzt	Speicherbereiche unter dem RAM reassemblieren.

Quickreferenz Hypra-Ass

Editorbefehle von Hypra-Ass

/A 100, 10	Automatische Zeilennummerierung. (Startzeile, Schrittweite)
/O	RENEW eines Quelltextes.
/D 100-200	Löschen von Zeilen und Zeilenbereichen.
/E 100-200	Listen von Zeilen und Zeilenbereichen.
TO,13;T1,24;T2,0;T3,10	Setzen von Tabulatoren.
	T0 = Tabulator für Assemblerbefehle.
	T1 = Tabulator für den Kommentar.
	T2 = Anzahl Blanks am Anfang einer Ausgabezeile.
	T3 = Tabulator für Symboltabelle.
/X	Verlassen des Assemblers.
/P1,100,200	Setzen eines Arbeitsbereichs (Page).
/ziffer (n)	Formatiertes Listen der Page.
/N1,100,10	Neu durchnummerieren einer Page mit Startnummer und Schrittweite.