

## **13 Zeichencodierung**

**ISO, ASCII, DIN, IBM-Zeichensatz, ANSI**

**Steuerzeichen**

**EBCDIC**

**Unicode**

**Druckerumschaltung Codepages 437 und 850**

### Gliederung

13.1	Zeichen, Bitmuster, Dezimaläquivalent .....	2
13.2	ISO 646, ASCII, DIN 66003, IBM-Zeichensatz, ANSI.....	3
13.3	Der IBM-Zeichensatz.....	4
13.4	Der ANSI-Zeichensatz.....	7
13.5	Die Steuerzeichen .....	8
13.5	Der EBCDIC-Zeichensatz .....	10
13.6	Unicode, der universelle 16-Bit-Zeichensatz .....	10
13.7	Druckerumschaltung Codepages 437 und 850.....	11

## 13.1 Zeichen, Bit-Muster und Dezimal-Äquivalent

### Kurze Wiederholung:

Alle Zeichen (druckbare Zeichen und Steuerzeichen) werden im Rechner und auf Datenträgern wie Magnetband, Diskette, Magnetplatte, Lochstreifen usw. durch Bit-Muster dargestellt, die nur aus den beiden Elementen »1« und »0« bestehen (Strom fließt/ Strom fließt nicht, magnetisiert/nicht magnetisiert, Loch gestanzt/Loch nicht gestanzt, usw.). Das Wort Bit ist ein Kunstwort, es entstand aus dem englischen Binary Digit = Binärziffer. Erst bei der Ausgabe auf dem Bildschirm oder auf den Drucker wird das Bit-Muster der Zeichen in das visuell lesbare Zeichen umgesetzt.

Das Zeichen "S" wird z.B. mit einem auf PCs weit verbreiteten Code durch folgendes Bit-Muster dargestellt: 01010011. Wegen der besseren Lesbarkeit schreibt man das Bit-Muster in Vierer-Gruppen an, für unseren Fall also: 0101 0011. Die Wertigkeit der einzelnen Bits richtet sich nach dem dualen Zahlensystem. Somit kann man für das Zeichen "S" folgende Darstellung angeben:

Bit-Nr (8-Bit-Code):	7	6	5	4	3	2	1	0
Bit-Muster für Zeichen S:	0	1	0	1	0	0	1	1
Bit-Wertigkeit, dezimal:	$2^7$	$2^6$	$2^5$	$2^4$	$2^3$	$2^2$	$2^1$	$2^0$
Bit-Wertigkeit, dezimal:	128	64	32	16	8	4	2	1

Das gezeigte Bit-Muster 0101 0011 hat das Dezimal-Äquivalent 83, da:

$$\begin{array}{r}
 0 \cdot 2^7 + 1 \cdot 2^6 + 0 \cdot 2^5 + 1 \cdot 2^4 + 0 \cdot 2^3 + 0 \cdot 2^2 + 1 \cdot 2^1 + 1 \cdot 2^0 = 83 \\
 0 \quad + 64 \quad + 0 \quad + 16 \quad + 0 \quad + 0 \quad + 2 \quad + 1 = 83
 \end{array}$$

Es ist leicht einsehbar, daß ein Datenaustausch zwischen der Zentraleinheit des Rechners und den Peripheriegeräten und auch anderen Rechnern nur dann möglich ist, wenn Vereinbarungen über die Zuordnung der Zeichen zu den Bit-Mustern bestehen. Dabei muß die Zuordnung beim Sender der Daten und beim Empfänger nicht unbedingt identisch sein, sie muß aber bekannt sein. Diese Vereinbarungen nennt man Zeichencodierung oder kurz Codierung. Die Zusammenstellung aller Zeichen mit den Codierungen nennt man Zeichensatz. Weitgreifende Normen sollen sicherstellen, daß ein einfacher Datenaustausch möglich ist.

In der Datenverarbeitung sind Zeichencodierungen mit 6 bit, 7 bit und 8 bit gebräuchlich; 8 bit vorrangig bei PCs. Neuere Entwicklungen gehen aber zu 16 bit, siehe Unicode Kap. 13.6.

Mit 6 bit kann man  $2^6 = 64$  verschiedene Bit-Muster darstellen,  
mit 7 bit kann man  $2^7 = 128$  verschiedene Bit-Muster darstellen,

mit 8 bit kann man  $2^8 = 256$  verschiedene Bit-Muster darstellen, usw.

## **13.2 Der internationale 7-Bit-Code ISO 646**

### **Der 7-Bit-Code ASCII**

### **Der 7-Bit-Code DIN 66003**

### **Der IBM-8-Bit-Zeichensatz (PC-8)**

### **Der ANSI-Zeichensatz**

Bei PCs dominiert der 8-Bit-Code. Dieser ging aus einem international genormten 7-Bit-Code hervor (ISO 646). Für die Erweiterung von 7 bit auf 8 bit hat IBM mit der Einführung des IBM-PCs einen Vorschlag gemacht, der sich zwischenzeitlich als Industriestandard durchgesetzt hat. Die Erweiterung ist aufwärtskompatibel, d.h. der 7-Bit-Code ist als Untermenge im 8-Bit-Code enthalten. Mit der Erweiterung um ein weiteres Bit können weitere nationale Sonderzeichen und vor allem Graphik-Zeichen dargestellt werden. Der Zeichensatz hat die Kurzbezeichnung *PC-8*.

Die internationale 7-Bit-Norm ist nur als eine Referenznorm zu betrachten, da sie nationale Abweichungen (deutsch, englisch, amerikanisch, französisch, italienisch, dänisch usw.) bei bestimmten Positionen zuläßt.

Wie bereits erwähnt, kann man mit 7 bit 128 verschiedene Bit-Muster und somit auch 128 verschiedene Zeichen darstellen. Die Zählung der Zeichen beginnt bei 0. Somit stehen die Zeichen 0..127 zur Verfügung. Als Zeichen-Nr (Code-Nr) dient das Dezimal-Äquivalent des betreffenden Bit-Musters.

In der internationalen 7-Bit-Norm sind die Zeichen 0 bis 31 und 127 für Steuerzwecke (z.B. Zeilenvorschub, Seitenvorschub, Rückwärtsschritt u.ä) reserviert. Somit stehen noch 95 druckbare Zeichen zur Verfügung. Damit kann man die üblichen Satzzeichen, die Ziffern 0..9, 26 Großbuchstaben und 26 Kleinbuchstaben abdecken, nicht aber die Summe der nationalen Sonderzeichen und auch keine Graphik-Zeichen.

Die internationale 7-Bit-Norm entstand aus dem US-Standard **ASCII** = **American Standard Code for Information Interchange**. Wegen der Herkunft spricht man häufig vom ASCII-Code, auch wenn eine nationale Abwandlung vorliegt. Damit besteht Verwechslungsgefahr. In der internationalen 7-Bit-Norm sind einige Bit-Muster nicht belegt, somit sind Anpassungen an nationale Zeichensätze in gewissen Grenzen möglich. Der deutsche Ableger ist die Norm DIN 66003. Diese Norm enthält, abweichend von ASCII, das Paragraphenzeichen, die Umlaute und das Scharf-S. Dafür mußten der "Klammeraffe", die eckigen und geschweiften Klammern, der Rückwärtsstrich (back slash), der senkrechte Strich und die Tilde geopfert werden. Im PC-Bereich wird der Zeichensatz nach DIN 66003 nicht benutzt.

Die Erweiterung von 7 auf 8 bit (IBM-Zeichensatz) verschafft zusätzliche 128 Zeichen. Probleme können auftreten, wenn z.B. Drucker nicht den IBM-Zeichensatz besitzen. Zu bedenken ist, daß die deutschen Sonderzeichen (Ä Ö Ü ä ö ü ß) beim IBM-Zeichensatz mit Salzstreuer-Systematik im Codebereich über dez 127 verteilt sind. Da beim Sortieren von Strings grundsätzlich nach den Codenummern der Zeichen vorgegangen wird, ergeben sich bei nationalen Sonderzeichen Probleme, die nur durch besondere programmtechnische Maßnahmen gelöst werden können.

Bei Druckern, Plottern usw. kann man in der Regel zwischen verschiedenen nationalen Zeichensätzen wählen, entweder durch entsprechendes Einstellen der dafür vorgesehen kleinen Schiebeschalter (Mäuseklaviere) oder softwaremäßig über Steuerbefehle. Oft kann noch innerhalb der nationalen Zeichensätze zwischen 2 Varianten gewählt werden. Die Unterschiede betreffen in der Regel die Nutzung des zusätzlichen Bits und somit die Code-Nummern ab 128. Häufig kann zusätzlich zwischen einem Zeichensatz 1 (die Code-Nummern 128 bis 159 sind mit Steuerzeichen belegt) und einem Zeichensatz 2 (= IBM-Zeichensatz) gewählt werden. Daneben gibt es aber auch Drucker, die das zusätzliche Bit zur Darstellung eines Kursiv-Zeichensatzes nutzen. Die notwendigen Informationen zur Einstellung der Drucker und Plotter sind den Handbüchern zu entnehmen. Siehe auch Kap. 25: Steuerbefehle für Drucker und Plotter.

### Zum ANSI-Zeichensatz

Die amerikanische Normungsbehörde ANSI (American National Standards Institute) hat ebenfalls eine Erweiterung des ASCII-Codes auf acht bit vorgenommen. Die zusätzlichen 128 Zeichen sind aber anders vergeben als beim IBM-Zeichensatz. Statt der Grafikzeichen enthält der ANSI-Zeichensatz weitere nationale Sonderzeichen. Die deutschen Sonderzeichen sind aber anders plaziert als beim IBM-Zeichensatz. Der ANSI-Zeichensatz wird von Microsoft Windows und vom Macintosh-Betriebssystem benutzt. Details siehe Kap. 13.4.

#### Beispiele:

Zeichen	IBM-Zeichensatz	ANSI-Zeichensatz
ä	dez 132	dez 228
ö	dez 148	dez 246
ü	dez 129	dez 252
Ä	dez 142	dez 196
Ö	dez 153	dez 214
Ü	dez 154	dez 220
ß	dez 225	dez 223

### 13.3 Der IBM-Zeichensatz

Mit dem folgenden Pascal-Instrumentarium kann man sich den verwendeten Zeichensatz anzeigen lassen:

Die Pascal-Standardfunktion  $\text{Chr}(n)$  liefert das Zeichen mit der Ordnungsnummer (Codenummer)  $n$ , wobei  $n$  ein Byteaudruck ist (0..255).

**Beispiel:** `Write(Chr(83));`

In Turbo-Pascal auch: `Write(#83);`

Die Ausgabe: `S`

Die Pascal-Standardfunktion  $\text{Ord}(Ch)$  liefert die Ordnungsnummer (Codenummer) eines Zeichens.

**Beispiel:** `Write(Ord('S'));`

Die Ausgabe: `83`

Mit dem folgenden kleinen Pascal-Programm können die druckbaren Zeichen mit ihren zugehörigen Codenummern in übersichtlicher Weise angezeigt werden:

```

program Pas13031;    { Zeichensatz }
uses
  CRT;
var
  i: Byte;
begin
  ClrScr;
  for i := 32 to 255 do
    Write(i:6, Chr(i):2);
  repeat
  until KeyPressed;
end.

```

Alle Zeichen können mit der *Alt*-Taste und der Code-Nr des Zeichens auf dem Bildschirm dargestellt werden. Die Taste *Alt* muß gedrückt bleiben, während man die Zifferntasten des Ziffernblocks betätigt.

**Beispiel:** `Alt+228` Auf dem Bildschirm erscheint beim IBM-Zeichensatz das griechische Zeichen  $\Sigma$  (Groß-Sigma).

Folgende Belegungen sollte man sich einprägen:

Leerzeichen	' ':	dez 032, hex \$20	(Blank, Space)
Zeichen	'A':	dez 065, hex \$41	
Zeichen	'a':	dez 097, hex \$61	
Ziffer	'0':	dez 048, hex \$30	

Zwischen den Großbuchstaben und den Kleinbuchstaben besteht eine Code-Verschiebung von dez 032 oder hex \$20. Die Hex-Codes der Ziffern sind leicht zu merken: Der Ziffer eine 3 voranstellen, ergibt den Hex-Code des Ziffernzeichens.

Bei den Umlauten, dem Scharf-S, dem Paragraphen-Zeichen und den Zeichen mit Code-Nr ab 128 können je nach Zeichensatz des Rechners bzw. des Druckers Abweichungen vorliegen. Bei Druckern können z.B. zwischen 128 und 159 nochmals Steuerzeichen liegen.

**IBM-Zeichensatz, Zeichensatz PC-8 (bis #127 identisch mit ASCII-Zeichensatz)**

Hier dargestellt für Codepage 437. Bei Codepage 850 sind die aus Einfach- und Doppellinien zusammengesetzten Semigrafikzeichen durch nationale Sonderzeichen ersetzt.

Dez	Hex	Dez	Hex	Dez	Hex	Dez	Hex	Dez	Hex	Dez	Hex	Dez	Hex
032	20	064	40	@	096	60	`	128	80	Ç	160	A0	á
033	21	065	41	A	097	61	a	129	81	ù	161	A1	í
034	22	066	42	B	098	62	b	130	82	é	162	A2	ó
035	23	067	43	C	099	63	c	131	83	â	163	A3	ú
036	24	068	44	D	100	64	d	132	84	ä	164	A4	ñ
037	25	069	45	E	101	65	e	133	85	à	165	A5	Ñ
038	26	070	46	F	102	66	f	134	86	â	166	A6	ª
039	27	071	47	G	103	67	g	135	87	ç	167	A7	º
040	28	072	48	H	104	68	h	136	88	ê	168	A8	¿
041	29	073	49	I	105	69	i	137	89	é	169	A9	¸
042	2A	074	4A	J	106	6A	j	138	8A	è	170	AA	¬
043	2B	075	4B	K	107	6B	k	139	8B	ï	171	AB	½
044	2C	076	4C	L	108	6C	l	140	8C	î	172	AC	¼
045	2D	077	4D	M	109	6D	m	141	8D	ì	173	AD	ı
046	2E	078	4E	N	110	6E	n	142	8E	Ä	174	AE	«
047	2F	079	4F	O	111	6F	o	143	8F	Å	175	AF	»
048	30	080	50	P	112	70	p	144	90	É	176	B0	⋈
049	31	081	51	Q	113	71	q	145	91	æ	177	B1	⋈
050	32	082	52	R	114	72	r	146	92	Æ	178	B2	⋈
051	33	083	53	S	115	73	s	147	93	ø	179	B3	⋈
052	34	084	54	T	116	74	t	148	94	ö	180	B4	⋈
053	35	085	55	U	117	75	u	149	95	ò	181	B5	⋈
054	36	086	56	V	118	76	v	150	96	û	182	B6	⋈
055	37	087	57	W	119	77	w	151	97	ù	183	B7	⋈
056	38	088	58	X	120	78	x	152	98	ÿ	184	B8	⋈
057	39	089	59	Y	121	79	y	153	99	Ó	185	B9	⋈
058	3A	090	5A	Z	122	7A	z	154	9A	Ü	186	BA	⋈
059	3B	091	5B	[	123	7B	{	155	9B	ç	187	BB	⋈
060	3C	092	5C	\	124	7C		156	9C	£	188	BC	⋈
061	3D	093	5D	]	125	7D	}	157	9D	¥	189	BD	⋈
062	3E	094	5E	^	126	7E	~	158	9E	€	190	BE	⋈
063	3F	095	5F	_	127	7F		159	9F	ƒ	191	BF	⋈

Die Steuerzeichen:

Dez	000	001	002	003	004	005	006	007	008	009	010	011	012	013	014	015
Hex	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F
Steuerz.	NUL	SOH	STX	ETX	EOT	ENQ	ACK	BEL	BS	HT	LF	VT	FF	CR	SO	SI
Control-	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O

  

Dez	016	017	018	019	020	021	022	023	024	025	026	027	028	029	030	031
Hex	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E	1F
Steuerz.	DLE	DC1	DC2	DC3	DC4	NAK	SYN	ETB	CAN	EM	SUB	ESC	FS	GS	RS	US

Ctrl-	P	Q	R	S	T	U	V	W	X	Y	Z	[	\	]	^
-------	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

Steuerzeichen DEL: Dez 127, Hex 7F

**Unterschiede zwischen ASCII und DIN 66003 (falls noch Bedarf besteht):**

Dez	Hex		ASCII	DIN
064	\$40	Klammeraffe (siehe unten)	@	Ä
091	\$5B	eckige Klammer auf	[	Ö
092	\$5C	Schrägstrich rückwärts (Backslash)	\	Ü
093	\$5D	eckige Klammer zu	]	ä
123	\$7B	geschweifte Klammer auf	{	ö
124	\$7C	Senkrechtstrich		ü
125	\$7D	geschweifte Klammer zu	}	ß
126	\$7E	Tilde	~	

Es wird darauf nochmals hingewiesen, daß beim IBM-Zeichensatz die nationalen Sonderzeichen und somit auch die oben aufgeführten deutschen Sonderzeichen im Code-Bereich ab #128 liegen. Durch diesen Umstand führen einfache String-Sortier-Algorithmen zu falschen Resultaten, wenn die Strings Umlaute oder das Scharf-S enthalten. Das Paragraphenzeichen »§« liegt bei diesem Zeichensatz auf dez 021 bzw. hex \$15.

**13.4 Der ANSI-Zeichensatz**

Der ANSI-Zeichensatz wird u.a. von Microsoft Windows und vom Macintosh-Betriebssystem benutzt. Die folgende Tabelle zeigt den Zeichensatz in der TrueType Schrift *Times New Roman*. Die Steuerzeichen 009 bis 014 wurden ausgeklammert.

000	□	032		064	@	096	`	128	□	160		192	À	224	à
001	□	033	!	065	A	097	a	129	□	161	;	193	Á	225	á
002	□	034	"	066	B	098	b	130	,	162	ç	194	Â	226	â
003	□	035	#	067	C	099	c	131	f	163	£	195	Ã	227	ã
004	□	036	\$	068	D	100	d	132	„	164	¤	196	Ä	228	ä
005	□	037	%	069	E	101	e	133	...	165	¥	197	Å	229	å
006	□	038	&	070	F	102	f	134	†	166		198	Æ	230	æ
007	□	039	'	071	G	103	g	135	‡	167	§	199	Ç	231	ç
008	□	040	(	072	H	104	h	136	^	168	¨	200	È	232	è
009		041	)	073	I	105	i	137	‰	169	©	201	É	233	é
010		042	*	074	J	106	j	138	Š	170	ª	202	Ê	234	ê
011		043	+	075	K	107	k	139	‹	171	«	203	Ë	235	ë
012		044	,	076	L	108	l	140	Œ	172	¬	204	Ì	236	ì
013		045	-	077	M	109	m	141	□	173	-	205	Í	237	í
014		046	.	078	N	110	n	142	Ž	174	®	206	Î	238	î
015	□	047	/	079	O	111	o	143	□	175	¯	207	Ï	239	ï
016	□	048	0	080	P	112	p	144	□	176	°	208	Ð	240	ð
017	□	049	1	081	Q	113	q	145	´	177	±	209	Ñ	241	ñ
018	□	050	2	082	R	114	r	146	’	178	²	210	Ò	242	ò
019	□	051	3	083	S	115	s	147	“	179	³	211	Ó	243	ó

020	□	052	4	084	T	116	t	148	”	180	´	212	Ô	244	ô
021	□	053	5	085	U	117	u	149	•	181	μ	213	Õ	245	õ
022	□	054	6	086	V	118	v	150	-	182	¶	214	Ö	246	ö
023	□	055	7	087	W	119	w	151	—	183	·	215	×	247	÷
024	□	056	8	088	X	120	x	152	˘	184	,	216	Ø	248	ø
025	□	057	9	089	Y	121	y	153	™	185	¡	217	Ù	249	ù
026	□	058	:	090	Z	122	z	154	š	186	°	218	Ú	250	ú
027	□	059	;	091	[	123	{	155	›	187	»	219	Û	251	û
028	□	060	<	092	\	124		156	œ	188	¼	220	Ü	252	ü
029	□	061	=	093	]	125	}	157	□	189	½	221	Ý	253	ý
030	-	062	>	094	^	126	~	158	ž	190	¾	222	Þ	254	þ
031		063	?	095	_	127	□	159	ÿ	191	¿	223	ß	255	ÿ

In Windows können Zeichen, die auf der Tastatur nicht enthalten sind, mit **Alt+0xxx** auf dem Bildschirm dargestellt werden. "xxx" ist die Codenummer nach vorstehender Tabelle. Die Taste "Num" muß für die Codeingabe auf dem Ziffernblock aktiviert sein.

## 13.5 Die Steuerzeichen

Die folgende Tabelle enthält alle Steuerzeichen nach ASCII, ANSI bzw. DIN 66003

Dez	Hex	Abk.	Ctrl	Bezeichnung	Wirkung
000	00	NUL	@	null	Null, keine Wirkung
001	01	SOH	A	start of header	Kopfblock-Beginn
002	02	STX	B	start of text	Textbeginn
003	03	ETX	C	end of text	Ende der Textes
004	04	EOT	D	end of transmission	Ende der Übertragung
005	05	ENQ	E	enquiry	Anfrage
006	06	ACK	F	acknowledge	Bestätigung
007	07	BEL	G	bell	Klingel, Piepser
008	08	BS	H	backspace	Cursor (Druckkopf) zurück
009	09	HT	I	horizontal tab	Horizontal-Tabulator
010	0A	LF	J	line feed	Cursor (Druckkopf) n. unten
011	0B	VT	K	vertical tab	Vertikal-Tabulator
012	0C	FF	L	form feed	Seitenvorschub
013	0D	CR	M	carriage return	Wagenrücklauf
014	0E	SO	N	shift out	hinausschieben
015	0F	SI	O	shift in	hineinschieben
016	10	DLE	P	data link escape	Verbindung unterbrechen
017	11	DC1	Q	direct control 1	Direktsteuerung 1
018	12	DC2	R	direct control 2	Direktsteuerung 2
019	13	DC3	S	direct control 3	Direktsteuerung 3
020	14	DC4	T	direct control 4	Direktsteuerung 4
021	15	NAK	U	not acknowledged	nicht bestätigt
022	16	SYN	V	synchronous idle	Synchronisierungszeichen
023	17	ETB	W	end of transm.block	Ende des Übertragungsblocks
024	18	CAN	X	cancel	löschen
025	19	EM	Y	end of medium	Ende des Speicher-Mediums



026	1A	SUB	Z	substitute	ersetzen
027	1B	ESC	[	escape	entfliehen, umschalten
028	1C	FS	\	form separator	Blatt-Trennung
029	1D	GS	]	group separator	Gruppen-Trennung
030	1E	RS	^	record separator	Record-Trennung
031	1F	US	_	unit separator	Einheiten-Trennung
127	7F	DEL		delete	letztes Zeichen löschen

Die meisten Steuerzeichen waren ursprünglich für die Datenübertragung vorgesehen. Die Anwendung seitens der Hersteller wird mitunter sehr unterschiedlich gehandhabt. Die meisten Steuerzeichen sind zudem mit Sonderzeichen belegt, die zum Teil auch gedruckt werden können, wie z.B. Chr(26). Drückt man dieses Zeichen, so erhält man den "Pfeil nach rechts". Schreibt man dieses Zeichen aber in eine Text-Datei, so wird es beim späteren Lesen als Kennung für Datei-Ende (Strg+Z, EoF, End of File) gewertet und nicht als Graphik-Zeichen, d.h. alle nachfolgenden Zeichen können nicht mehr von der Datei gelesen werden. Andere Steuerzeichen können mitunter auf dem Bildschirm als Sonderzeichen dargestellt werden, wogegen sie auf dem Drucker Steuerfunktionen auslösen.

#### Die Steuerzeichen bei PCs:

Dez	Hex	Abk.	Strg+	Wirkung des Steuerzeichens
007	\$07	BEL	G	Klingel, Piepser
008	\$08	BS	H	Backspace. Cursor (Druckkopf) ein Zeichen zurück
009	\$09	TAB	I	Tabulatorsprung
010	\$0A	LF	J	Line Feed. Cursor nach unten, Zeilenvorschub ohne CR
012	\$0C	FF	L	Form Feed. Bildschirm löschen, Seitenvorschub
013	\$0D	CR	M	Carriage Return. Wagenrücklauf. Cursor auf Anfang nächste Zeile, Wagenrücklauf. Bei Druckern auch meistens mit LF.
027	\$1B	ESC	[	Escape. Einleitung von Steuersequenzen, z.B. für Drucker
127	\$7F	DEL		Delete. Letztes Zeichen löschen

Die Steuerzeichen unter dez 032 können mit der Taste *Strg* (Ctrl) und einem Buchstaben (groß oder klein) nach vorstehender Tabelle erreicht werden. Dazu Taste *Strg* gedrückt halten und den betreffenden Buchstaben drücken. Beispiel: *Strg+L* löscht Bildschirm (Tasten *Strg* und *L*).

Mit *Escape* (entfliehe dem normalen Code, schalte um auf einen anderen Code) können in Verbindung mit weiteren Zeichen noch viele weitere Steuerzeichen gebildet werden.

Bei Druckern und Plottern werden diese sog. Escape-Sequenzen sehr häufig zur Gerätesteuerung genutzt. Die im ASCII selbst definierten 32 Steuerzeichen würden nicht ausreichen, die vielfältigen Möglichkeiten dieser "intelligenten" Geräte zu nutzen. Man denke nur an die Schriftenwahl (normal, fett, breit, schmal, doppelte Höhe, kursiv, hochgestellt, tiefgestellt usw.). Bei üblichen Matrixdruckern hat sich der Escape-Steuerzeichensatz von *Epson* durchgesetzt (Esc/P, relativ kurze Escape-Sequenzen); bei La-

serdruckern dominiert die von Hewlett-Packard (HP) stammende Seitenbeschreibungssprache PCL (Page Control Language) mit z.T. sehr langen Escape-Sequenzen.

**Beispiel:** Die Escape-Sequenz `Chr(27) + 'W' + Chr(1)` schaltet die Breitschrift eines Druckers mit Epson-Steuerzeichensatz ein. Siehe auch Kap. 25: Steueranweisungen für Drucker und Plotter.

Für die erweiterte Bildschirm- und Tastatursteuerung unter MS-DOS gibt es die ANSI-Escape-Sequenzen, die dann benutzt werden können, wenn in die Konfigurationsdatei *Config.SYS* der Treiber *ANSI.SYS* aufgeführt ist; siehe DV II, Kap.1, Betriebssystem MS-DOS.

## 13.6 Der EBCDIC-Zeichensatz

In der Groß-EDV wird mit abnehmender Bedeutung der 8-Bit-Code **EBCDIC** (**E**xtended **B**CD **I**nterchange **C**ode, **BCD** steht für **B**inary **C**oded **D**ecimals) benutzt. Dieser Code ist grundsätzlich anders aufgebaut als der bereits beschriebene ASCII-Code und dessen 7-Bit- oder 8-Bit-Abwandlungen. Bei Mikrocomputern wird EBCDIC praktisch nicht benutzt. Nachfolgend der EBCDIC-Zeichensatz (Bk = Blank, Leerzeichen):

Dez	Hex	Dez	Hex	Dez	Hex	Dez	Hex	Dez	Hex	Dez	Hex						
064	40	Bk	096	60	128	80	160	A0	192	C0	224	E0					
065	41		097	61	/	129	81	a	161	A1	193	C1	A	225	E1		
066	42		098	62		130	82	b	162	A2	s	194	C2	B	226	E2	S
067	43		099	63		131	83	c	163	A3	t	195	C3	C	227	E3	T
068	44		100	64		132	84	d	164	A4	u	196	C4	D	228	E4	U
069	45		101	65		133	85	e	165	A5	v	197	C5	E	229	E5	V
070	46		102	66		134	86	f	166	A6	w	198	C6	F	230	E6	W
071	47		103	67		135	87	g	167	A7	x	199	C7	G	231	E7	X
072	48		104	68		136	88	h	168	A8	y	200	C8	H	232	E8	Y
073	49		105	69		137	89	i	169	A9	z	201	C9	I	233	E9	Z
074	4A		106	6A	^	138	8A		170	AA		202	CA		234	EA	
075	4B	.	107	6B	,	139	8B		171	AB		203	CB		235	EB	
076	4C	<	108	6C	%	140	8C		172	AC		204	CC		236	EC	
077	4D	(	109	6D	-	141	8D		173	AD		205	CD		237	ED	
078	4E	+	110	6E	>	142	8E		174	AE		206	CE		238	EE	
079	4F		111	6F	?	143	8F		175	AF		207	CF		239	EF	
080	50	&	112	70		144	90		176	B0		208	CF		240	F0	0
081	51		113	71		145	91	j	177	B1		209	D1	J	241	F1	1
082	52		114	72		146	92	k	178	B2		210	D2	K	242	F2	2
083	53		115	73		147	93	l	179	B3		211	D3	L	243	F3	3
084	54		116	74		148	94	m	180	B4		212	D4	M	244	F4	4
085	55		117	75		149	95	n	181	B5		213	D5	N	245	F5	5
086	56		118	76		150	96	o	182	B6		214	D6	O	246	F6	6
087	57		119	77		151	97	p	183	B7		215	D7	P	247	F7	7
088	58		120	78		152	98	q	184	B8		216	D8	Q	248	F8	8
089	59		121	79		153	99	r	185	B9		217	D9	R	249	F9	9
090	5A	!	122	7A	:	154	9A		186	BA		218	DA		250	FA	
091	5B	\$	123	7B	#	155	9B		187	BB		219	DB		251	FB	

092 5C *	124 7C @	156 9C	188 BC	220 DC	252 FC
093 5D )	125 7D '	157 9D	189 BD	221 DD	253 FD
094 5E ;	126 7E =	158 9E	190 BE	222 DE	254 FE
095 5F 7	127 7F "	159 9F	191 BF	223 DF	255 FF

### 13.7 Unicode, der universelle 16-Bit-Zeichensatz

Bei allen bisher genannten Codes wird ein Zeichen mit einem Byte = 8 bit dargestellt. Dadurch kann ein Zeichensatz nur  $2^8 = 256$  verschiedene Zeichen umfassen. Mit zwei Byte = 16 bit könnte man  $2^{16} = 65536$  verschiedene Zeichen darstellen. Verschiedene Hard- und Software-Hersteller haben sich auf die Festlegung eines Zeichensatzes auf der 16-Bit-Basis geeinigt. Das Ergebnis hat die Bezeichnung »Unicode« und soll einen »*global computer code for storage and transmission of text around then world*« darstellen und nach diesem Anspruch alle größeren heute lebenden Sprachen abdecken, also auch z.B. chinesisch, japanisch, griechisch, kyrillisch, hebräisch, arabisch usw. Außerdem soll Unicode auch mathematische und technische Sonderzeichen sowie Steuerzeichen umfassen. Die Codes für über 27 000 Zeichen sind bereits vergeben. Das chinesische "Alphabet" umfaßt mehr als 30 000 verschiedene Zeichen. Durch Zerlegung in in kombinierbare Grundelemente ist es gelungen, diesen Zeichenumfang in Unicode auf ca. 19 000 zu reduzieren. Unicode kann die Basis für die internationale Kommunikation werden. Auch die Entwicklung von Programmen für den weltweiten Einsatz wird durch Unicode begünstigt werden. Ein Nachteil darf nicht übersehen werden: Verdoppelung des Speicherbedarfs!

Im Codebereich 0 bis 127 ist der Unicode identisch mit dem Ascii-Code. Das Euro-Zeichen liegt im Unicode auf #8365. Das griechische Zeichen liegen zwischen #913 (Groß-Alpha) und #937 (Groß-Omega) bzw. zwischen #945 (Klein-Alpha) und #969 (Klein-Omega).

### 13.8 Druckerumschaltung Codepages 437 und 850

Im Betriebssystem MS-DOS kann mit entsprechenden Eintragungen in den Startdateien "Config.SYS" bzw. "AutoExec.BAT" (siehe DV II, Kap. 1, Betriebssystem) beim IBM-Zeichensatz zwischen verschiedenen nationalen Ausgestaltungen, den Codepages, gewechselt werden.

Die **Codepage 437** entspricht dem ursprünglichen IBM-Zeichensatz und ist jetzt überwiegend in den USA und dem englischsprachigen Kanada gebräuchlich. In dieser Codepage stehen u.a. alle IBM-Semigrafikzeichen zur Verfügung.

Für den mehrsprachigen Einsatz in Europa ist die **Codepage 850** geeigneter, bei der über 40 Zeichen aus dem Codebereich > 128 durch weitere nationale Sonderzeichen ersetzt wurden. Betroffen sind u.a. fast alle Zeichen im Codebereich 224 bis 255 und auch die IBM-Semigrafikzeichen, die aus Kombinationen von Einfach- und Doppellinien bestehen.

Während man im DOS bei jeder Sitzung nur die eine oder die andere Codepage einstellen kann, ist es bei geeigneten Druckern möglich, bei Bedarf während des Ausdrucks die Codepage zu wechseln, natürlich auch vor dem Ausdrucken (initialisieren).

Zu diesen Druckern zählen u.a. auch die Drucker von Hewlett-Packard oder dazu kompatible Drucker mit der Seitenbeschreibungssprache PCL 5 (PCL = Page Control Language).

Das deutsche Windows arbeitet im Windows-Modus mit dem Ansi-Zeichensatz (siehe Pascal-Arbeitsblatt) und im DOS-Modus standardmäßig mit der Codepage 850. Ein Umstellen z.B. auf Codepage 437 ist mit Codechange-Hilfsprogramm<sup>1</sup> von Windows möglich.

Das folgende Programm demonstriert die Codepage-Umschaltung für den Drucker, damit werden in einer Zeichensatztablelle sowohl die Zeichen der Codepage 437 (erstes Zeichen) und 850 (zweites Zeichen) dargestellt. Diese Demo greift notgedrungen auf Kap. 18 (Dateien) und Kap. 25 (Druckersteuerung) vor.

```

program Pas13071; { Druckerumschaltung Codepages 437/850 }
                { "Pas13071.PAS", früher "Codepage.PAS" }
                { 76170397, Dr. K. Haller, FHM DR }
uses
  CRT, PRINTER;

const
  Start = 128;

var
  i, j,
  Ascii: Byte;
  Ch: Char;
  Drucken: Boolean;

function HP_Codepage(CP: Word): Char; { Für Hewlett-Packard-Drucker }
const { oder andere Drucker mit PCL 5 }
  Esc = #27; { PCL = Page Control Language }
var
  CPStr: string[4];
begin
  case CP of
    437: CPStr := '(10U'; { Bei HP auch "PC-8" genannt. }
        { 437: IBM-Layout vorzugsweise für USA }
    850: CPStr := '(12U'; { Bei HP auch "PC-850" genannt. }
        { 850: IBM-Layout vorzugsweise für Europa }
  end;
  if Drucken { Bei HP "4099T" Schrift "Courier" }

```

<sup>1</sup> Von der Windows-CD-ROM Codechange-Programm auf die Festplatte laden, z.B. in "C:\Temp" und dort ausführen. Im Dialog "437" oder "850" einstellen.

```

    then Write(Lst, Esc + CPStr + Esc + '(s0p12v0s0b4099T');
    HP_Codepage := ' '; { Dummy, damit Funktionsaufruf }
end;          { in Write-Prozedur möglich }

begin
  ClrScr;
  WriteLn; WriteLn; WriteLn;
  WriteLn('          Demo Codepage 437 und 850.   Dr. K. Haller');
  WriteLn('          -----');
  WriteLn('          1      HP-Druckerausgabe (mit  Umschaltung)');
  WriteLn('          2      Bildschirmausgabe (keine Umschaltung)');
  WriteLn('          Esc   Abbruch          ');
  WriteLn('          -----');
  Write(' ');
  repeat
    Ch := ReadKey;
    if Ch = #27 then Halt;
  until Ch in ['1', '2'];
  Write(Ch);

  if Ch = '1'
  then begin Drucken := True; Assign(Lst, 'LPT1'); end
  else begin Drucken := False; Assign(Lst, 'CON'); end;
  Rewrite(Lst);

  for i := 1 to 4 do
    WriteLn(Lst);

  WriteLn(Lst, HP_Codepage(437));
  WriteLn(Lst, '          Erweiterter Ascii-Zeichensatz, Codepages 437/850');
  WriteLn(Lst, '          Erstes Zeichen Codepage 437 (USA, engl. Kanada)');
  WriteLn(Lst, '          Zweites Zeichen Codepage 850 (überwiegend Europa)');
  WriteLn(Lst, '          i-----D-----D-----D-----D-----@');

  for i := Start to Start + 31 do
  begin
    Write(Lst, '          | ');
    for j := 0 to 3 do
    begin
      Ascii := i + j*32;
      Write(Lst, Ascii:3, ':',
             HP_Codepage(437), Chr(Ascii),
             HP_Codepage(850), Chr(Ascii), ' | ');
    end;
    if not Drucken
    then if ((i + 1) mod 16) = 0
    then repeat { Damit Bildschirm nicht scrollt }
    until ReadKey <> '';
    WriteLn(Lst);
  end;
  Write(Lst, HP_Codepage(437));
  WriteLn(Lst, '          +-----+');
  WriteLn(Lst, '          Anmerkung: Die gewünschte Codepage (437, 850 oder)');
  WriteLn(Lst, '          andere) ist einzutragen in:          ');
  WriteLn(Lst, '          | "Config.SYS", Eintrag "country = ..."          ');
  WriteLn(Lst, '          | "Autoexec.BAT", Einträge          ');
  WriteLn(Lst, '          "Mode CON Codepage Prepare = ..."          ');
  WriteLn(Lst, '          "Mode CON Codepage Select = ..."          ');
  WriteLn(Lst, '          Der Drucker ist gegebenenfalls ebenfalls auch auf');
  WriteLn(Lst, '          die gewünschte Codepage einzustellen.          ');
  WriteLn(Lst, '          Windows arbeitet im Codebereich ab 128 nicht mit          ');
  WriteLn(Lst, '          dem Ascii-Zeichensatz, sondern mit dem Ansi-          ');
  WriteLn(Lst, '          Zeichensatz. Zeichen siehe Pascal-Arbeitsblatt.          ');
  WriteLn(Lst, '          -----');
  WriteLn(Lst, '          Programm "Pas13071.PAS". 76170397, Dr. K. Haller');

```

```
if Drucken
  then Write(Lst, #12)
  else repeat until ReadKey <> '';
Close(Lst);
end.
```

Siehe Rückseite ... (Tabelle auf Rückseite fehlt in der PDF-Version des Skriptums)