

Die Welt der Codes

Jobst-Hartmut Lüddecke

27. September 2011

Zusammenfassung

Um *Information* transportieren zu können, brauchen wir einen *Code*, der vom *Sender* der *Information* genauso verstanden werden muss, wie vom *Empfänger*. Nur dann kommt es zu einer *Übertragung* und einem *Austausch der Information*. Zwar zeigt uns selbst die *Langenscheidt*-Redaktion, dass dies oft nicht möglich ist und sieht da *geschlechtsspezifischen* Bedarf für Wörterbücher der Art *Deutsch - Frau*, *Deutsch - Mann*, *Deutsch - Chef* usw.

Weiterhin sind *Sprachen* und *Codes* abhängig von der Umgebung. Sie sind also *kontextsensitiv*. So ist ein *Seenotzeichen* ein anderes, als das *Alpine Notsignal*, oder ein *Notsignal der Jagd* im Wald. Der Kreis aus Zeigefinger und Daumen ist bei einem Koch ein *Bon* (soviel wie gut und lecker), bei einem Taucher ein *alles O.K.* und bei einem Autofahrer ein Schimpfwort.

Die *Informationstheorie*, die *Nachrichtentechnik* und das *Kommunikationsdesign* beschäftigt sich hauptberuflich mit diesem Problem. An dieser Stelle muss dieser Problembereich etwas *angerissen* werden, um ein Bewusstsein für *Codes*, *Informationsübermittlung* und *Datenstrukturen* aufzubauen und es damit selbstverständlich erscheinen lässt, dass es nicht nur *einen* Code gibt, sondern dass man ganze Buchreihen (z.B. in der Nachrichtentechnik) damit füllen kann.

In diesem Skript möchte ich zeigen, dass nicht nur *Sprache* für den *Informationsaustausch* verwendet wird und dass diese *Übertragung* auch recht unabhängig vom *Medium der Informationsübermittlung* ist. Auch soll das Bewusstsein für unterschiedlichste *Vokabularien* geschärft werden.

Inhaltsverzeichnis

1	Codes im täglichen Leben	3
1.1	Dress-Code	3
1.2	Verkehrsampel	5
1.3	Ideogramme	5
1.4	Silbenschrift	5
1.5	Buchstaben	5
1.6	Ziffern	5
1.7	Morse-Alphabet	6
2	Codes aus Bit-Kombinationen	7
2.1	Der 5-Bit Zeichensatz CCITT-2, Baudot-Code	7
2.2	ASCII und andere 7-Bit Zeichensätze	8
2.2.1	nationale Versionen vom ASCII	10
2.3	8-Bit Zeichensätze	10
2.3.1	ISO-8859-1, Latin-1 Westeuropäisch	11

2.3.2	ISO-8859-2, Latin-2 Mitteleuropäisch	11
2.3.3	ISO-8859-3, Latin-3 Südeuropäisch	11
2.3.4	ISO-8859-4, Latin-4 Baltisch	11
2.3.5	ISO-8859-5, Kyrillisch	11
2.3.6	ISO-8859-6, Arabisch	11
2.3.7	ISO-8859-7, Griechisch	11
2.3.8	ISO-8859-8, Hebräisch	11
2.3.9	ISO-8859-9, Latin-5 Türkisch	11
2.3.10	ISO-8859-10, Latin-6 Nordisch	11
2.3.11	ISO-8859-11, Thai	12
2.3.12	ISO-8859-12, wurde verworfen	12
2.3.13	ISO-8859-13, Latin-7 Baltisch	12
2.3.14	ISO-8859-14, Latin-8 Keltisch	12
2.3.15	ISO-8859-15, Latin-9 Westeuropäisch	12
2.3.16	ISO-8859-16, Latin-10, Südeuropäisch	12
2.4	16-Bit Zeichensätze	12
2.5	Zeichensätze aus mehreren 8-Bit Zeichen	12
2.5.1	UTF-8	12

3 Zahlen und Zahlensysteme 13

3.1	Dezimalzahlen	13
3.2	Binärzahlen	13
3.3	Oktalzahlen	13
3.4	Hexadezimalzahlen	13
3.5	BCD = binary coded decimals	14
3.6	Byte	14
3.7	Word	14
3.8	Double-word	14
3.9	Quod-word	14
3.10	Integer	14
3.11	Real	15

1 Codes im täglichen Leben

1.1 Dress-Code

Anhand der Kleidung soll ein sehr komplexer Code, der *Dress-Code* illustriert werden. Wahrscheinlich ist kaum jemandem bewusst, wie viele Signale und Informationen darin stecken, wenn man morgens seine Kleidung – oft gedankenlos – anzieht.

Mit einer bestimmten Kleidung ordne ich mich einer bestimmten Gruppe und einem bestimmten Event zu, kann aber auch bewusst Protest ausdrücken. Die einfachste Art der Zugehörigkeit ist die *Uniform*. Weiterhin zieht man sich zu bestimmten Gelegenheiten wie *Arbeit, Freizeit, Hochzeit, Oper, Rock-Konzert, Pferde-Rennen* u.v.a. in der Regel unterschiedlich an, wenn man zu der bestimmten Gruppe gehören will. Man kann sich natürlich auch bewusst *nicht* an diese Gelegenheiten halten und damit etwas ausdrücken.

Ein recht auffälliger Code ist der *Business Dress*. Ganz *hanseatisch* kann dieser nur die Farben *Grau* oder *Dunkelblau* haben. Mit *Anthrazit* als Anzug kann man kaum etwas verkehrt machen. Im *angelsächsischen Raum*¹ – der einigen Einfluss auf den *hanseatischen Raum* hat – ist ein englisch – nicht italienisch – geschnittener (dreiteiliger) *marineblauer* Anzug eine *Machtdemonstration*, auf die kein führender Politiker und Wirtschaftskapitän verzichten möchte. Ein *brauner* Anzug ist in diesem Fall ein absolutes *do-not*, da er nach wie vor in Hamburg als *nationalsozialistisch* belastet gilt. Ein *schwarzer* Anzug ist dann für Feierlichkeiten reserviert und wenn man nicht gerade einen Trauerfall in der Familie hat, sollte er zu einem geschäftlichen Meeting nicht angezogen werden.

Schwierig ist auch ein Nadelstreifenanzug. Entweder man gehört zu einer bestimmten Managerkaste, ist Mafioso, oder macht sich lächerlich.

Wenn man z.B. bei einer Bank arbeitet führt kein Weg daran vorbei sich diesem *Dress Code* zu fügen. Als Mann hat man nur noch den *Schlips*² und ggf. eine *Weste* als persönliches Ausdrucksmittel. Als Frau kann man sich dann nur über *Schmuck* und die *Tiefe des Ausschnitts*³ ausdrücken.

Hält man sich nicht an diesen Code und erscheint bei einem Geschäft z.B. in einem *Country-Jackett* aus kariert gewebtem *Harris-Tweed* – welches auf keinem Fall billiger ist, als der *Business-Dress* – dokumentiert man entweder *Ich habe es nicht nötig mich an eure Regeln zu halten* oder *Ich bin zu blöd um die Regeln zu kennen*⁴. Eine Ausnahme gilt für Mitglieder einer Hochschule, denen eine gewisse *Narrenfreiheit* im Geschäftsleben zugestanden wird.

Nicht weniger schwierig ist der Dress-Code *Business-Casual*. Hier wird ein Balanceakt zwischen *Business* und *Casual* (Freizeitkleidung) propagiert, der gar keiner ist, denn diese betont *lockere* Kleidung ist *keine Freizeitkleidung*, bzw. zwar auch tauglich für die Freizeit, aber nicht minder Regeln unterworfen wie die klassische *Geschäftskleidung*. *Business-Casual* gilt im Geschäftsleben, wenn der

¹England und die amerikanische Ostküste.

²Für Krawatten gibt es dann wieder einen eigenen Code. So müssen bei einer gestreiften Krawatte immer der Streifen von der linken Schulter zur rechten Hüfte gehen. Nicht – wie schon gelesen – damit ein Aufwärtstrend wie bei einem Aktienkurs symbolisiert werden soll, sondern weil dieses Design von den Farben britischer Regimenter abstammt. Laufen die Streifen anders herum, zeigt man, dass man entweder keine Ahnung hat, oder Amerikaner ist – was für Briten aufs gleiche hinausläuft. Weiterhin kann die Farbkombination gerade im *United Kingdom* ein wahres Minenfeld sein, denn jedes Internat, jede Hochschule, jedes Regiment und jeder Club hat seine eigene Farbkombination. Somit kann man mit seiner Krawatte eine entsprechende Zugehörigkeit kenntlich machen. Peinlich kann es dann werden, wenn man als Deutscher bei Verhandlungen in England z.B. die Krawatte des Regiments der *Black Watch* trägt und ganz bestimmt nie dort gedient hat. Dies ist nicht weit hergeholt, sondern solche Krawatten sind durchaus im Angebot einiger *exklusiver* Läden, ohne auf die Bedeutung hinzuweisen.

³Oft und gern verwendetes Mittel der Ablenkung und damit ein beabsichtigter Vorteil bei Vertragsverhandlungen. Auch schon oft in Kombination mit einer viel zeigenden Rocklänge bei mündlichen Prüfungen erlebt.

⁴Dies hängt dann vom weiteren Auftreten ab. Wenn man bei den Verhandlungen am wesentlich längerem Hebel sitzt und der Andere auf dieses Geschäft angewiesen ist, kann man dies so zeigen. Ist man selbst darauf angewiesen und sitzt am kürzeren Hebel, dann fällt man wohl in die Kategorie 2, es sei denn man ist wild am pokern und versucht Kategorie 1 glaubhaft zu machen.

normale *Business-Dress overdressed* wäre, wie bei einem *Informellen Essen*, einer *geschäftlichen Party*, oder am *Casual Friday*.

Die Hemden sind etwas bunter, habe einen kleineren Kragen, der offen oder mit Schlips getragen werden kann. Die Ärmel sind kurz oder geknöpft. Zu diesem Stil gehört kein Anzug, dafür ein Sakko oder Blazer mit einer anders farbigen Chino und passende Schuhe. Hier darf ein Sakko aus Baumwolle, Leinen oder Gemisch auch schon mal braun sein, wenn es z.B. mit einer blauen Chino kombiniert wird. Man kann schon mal zu einer blauen Hose braune Schuhe tragen. Bei einem *blauen Business-Anzug* ein großes *Faux pas*, denn es gehen dort nur schwarze Schuhe.

Also die Kleidung ist ein Mittel der *Selbst-Inszenierung* und damit ein sehr *komplexer Code* und Ausdrucksmittel. Es kann aber auch ein *Verstecken in der Masse* sein. Angefangen bei der *Uniform*. In Uniform sehen alle mehr oder weniger gleich aus und man fällt nicht auf. Möchte man sich aus der Masse positiv hervorheben, muss man sich um *Orden, Ehrenzeichen*⁵ und etwas mehr *Lametta auf der Schulter* bemühen.

Auch wenn man jedem Trend hinterherläuft, möchte man nicht auffallen und sich in der Menge verstecken. Dies ist aber auch ein Zeichen von geringem Selbstbewusstsein und besonders bei pubertierenden Mädchen anzutreffen.⁶

Will man so richtig in eine Sache einsteigen, wie z.B. bei einem *Reenactment* dann fängt es damit an, sich so authentisch wie möglich, entsprechend der *nachzuerlebenden* Epoche zu kleiden und sich nicht (billig) zu *verkleiden* wie beim Karneval. Mit dieser Inszenierung fängt das gewünschte Lebensgefühl erst an. Es kann leicht vorkommen, dass man sich für so eine Veranstaltung wesentlich sorgfältiger und teurer kleidet, als im Alltag. Was glauben Sie kostet ein Stoff aus handgewebter Wolle mit pflanzliche Farben per Hand gefärbt und danach von einer Schneiderin per Hand zu einer Tunika konfektioniert? Dafür bekommt man einen Anzug in der *Saville Row* und ein Hemd aus der *Jermyn Street* gleich dazu.⁷

Dann gibt es noch verschiedene Formen des Protestes, wie das *Outfit* eines *Punks* – das eigentlich auch schon wieder eine Uniform ist – und andere Haltungen. Aktuell kombiniert meine Tochter Kleidungsstücke, die maximal *nicht* zusammen passen.

Ist das Tragen von Kopfbedeckungen in geschlossenen Räumen⁸ nun Protest, Unwissen, oder *spät-pubertierende Coolness*? Bei nach hinten gedrehten Schirmen wird es ab einem gewissen Alter – eben nach der normalen Pubertät – absolut lächerlich und ich tendiere zur letzten Interpretation.

Weitere Beispiele gibt es viele, Hauptsache man versteht die Nachricht dieses Codes und Sie, worauf es mir jetzt ankommt.

Kleiner Tipp: Mit braunen Schuhen zu einem blauen Business-Anzug macht man sich genauso lächerlich, wie mit Bundeswehr-Kampfstiefeln zu einer mittelalterlichen Tunika. Genauso gut könnte man sich ein Schild mit der Aufschrift *Ich bin blöd* umhängen. Die Meinung von *Blöden* ist aber nicht gefragt und mit *Blöden* macht man auch keine Geschäfte. Es kann also sehr wichtig sein die richtigen Signale zu senden und diese auch zu verstehen.

⁵Gerade aus diesem Grund sind zulässig sichtbare Belege der eigenen Leistung in so einer Gemeinschaft so wertvoll und heiß begehrt. Außerhalb dieser Gemeinschaft werden sie uninteressant und durch andere Statussymbole ersetzt. Wen interessiert schon im Privatleben eine *Schützenschmuck*, oder ein *Einzelkämpfer-Abzeichen*?

⁶Jungs haben da andere Status-Symbole um möglichst *cool* zu erscheinen – die auch nicht besser sind und genauso die eigene pubertäre Unsicherheit übertünchen sollen.

⁷Die ersten Adressen in London für diese Kleidungsstücke

⁸Sofern man nicht gerade auf dem Weg in oder aus dem Gebäude ist, setzt man Kopfbedeckungen eigentlich ab. Die Ausnahme ist ein schwer zu installierendes Kunstwerk auf weiblichen Köpfen, welches ohne fremde Hilfe nicht auf- und abzusetzen ist.

1.2 Verkehrsampel

Signalgerät, genauer eine *Lichtzeichenanlage* für die Verkehrsregelung mit 3 Zuständen. Dieser Code umfasst also folgende Signale:

rot : Halt⁹

gelb : Achtung, bzw. bei blinkendem gelben Licht ein Ausfall der Ampel

grün : Fahren

1.3 Ideogramme

Symbole für *vollständige Begriffe*, wie z.B. das *Strichmännchen* an der WC-Tür.¹⁰

Dann gibt es Schriften wie *Kanji* (chinesisch, japanisch) usw. in denen jedes Zeichen für einen vollständigen Begriff steht. Dabei muss man ca. 20.000 Zeichen kennen um eine normale Zeitung lesen zu können, aber man muss nicht die Sprache sprechen müssen. Sehr praktisch in einem Land wie China mit mehreren Sprachen (Mandarin, Kantonesisch usw.) und über 90 Dialekten.

1.4 Silbenschrift

Schriftarten, die ganze Silben-Laute wiedergeben. Vertreter sind z.B. *Steno* und das japanische *Katakana*. Es gilt im Prinzip das Gleiche wie für die folgenden Buchstaben, nur hat man wesentlich mehr Zeichen für eine kürzere und schnellere Schreibweise.

1.5 Buchstaben

Buchstaben sind ein definierter Code für bestimmte *Laute*¹¹. Diese Laute werden zwar national etwas unterschiedlich ausgelegt, es ist aber ein umfangreicher Code der ab dem 1. Schuljahr intensiv in der Mustererkennung (Lesen) und der Produktion (Schreiben) trainiert wird. Neben unserem gebräuchlichen Code (römische Schriftzeichen) gibt es noch ähnliche Laut-Codes wie *Griechisch*, *Kyrillisch*, *Runen* usw.

Der Vorteil liegt in der relativ kleinen Anzahl der Symbole, dafür muss man die Laute zusammen setzen können und die Sprache sprechen um den Inhalt zu verstehen.

1.6 Ziffern

Zeichen für gewisse Mengen in unterschiedlichen Ausprägungen. Man denke nur an *Strichlisten*, *römische Ziffern* und *arabische Ziffern*¹²

Beispiel für gleiche Bedeutungen mit unterschiedlichen Darstellungsformen.

⁹Stopp

¹⁰Wie nennt man eigentlich das Ideogram an der Damentoilette? *Strichmädchen* ist es definitiv nicht. ☺

¹¹*Kannst Du Laute malen?* Zitat aus dem Film *der 13. Krieger*

¹²Die heute gebräuchlichen arabischen Ziffern wurden von Kaiser Friedrich II im Abendland eingeführt. Friedrich II wuchs im *normannischem* Königreich Sizilien auf und hatte christliche, moslemische und jüdische Lehrer und damit die zu der Zeit maximal mögliche Bildung. Die arabischen Ziffern mit dem Übertrag zur Basis 10 haben das Rechnen gegenüber den römischen Ziffern erheblich vereinfacht. Zusätzlich wurde dadurch auch die Zahl *Null* eingeführt, was die Sache noch mehr erleichtert hat.

1.7 Morse-Alphabet

Code aus 4 Elementen, aus einem Signal und einer Pause mit jeweils 2 verschieden langen Zeiten. Es gibt ein kurzes Zeichen – dem *Punkt* oder *di* – und ein langes Zeichen – dem *Strich* oder *da* –, sowie kurze Pausen zwischen den Elementen eines Zeichens und lange Pausen zwischen den Zeichen. Für diesen Code ist das **Trägermedium** der Informationsübermittlung beliebig. Es kann ein *Licht an* und *Licht aus* bei Blinkzeichen, ein *Piep* oder *Nichts* im Funkverkehr, oder ein *Strich* oder *Lücke* auf einem Papierstreifen eines Telegraphen sein.

Dieser Code wurde 1832 von *Samuel Finlay Breese Morse* erfunden und am 24. Mai 1844 zum internationalen Standard. Da dieser Code noch lebt, zeigt ein Update des Codes nach **160 Jahren**. Ab Mai 2004 gibt es für die 189 Mitgliedsstaaten ein neues Zeichen für @ und zwar `.--.-.`

Das Morse-Zeugnis ist auch weiterhin Voraussetzung für die Erteilung einer Amateurfunklizenz. Im *Seefunk* muss man es noch in einer recht hohen Geschwindigkeit beherrschen.

Beispiel dafür, dass es nicht nur e-Mails gibt und man sich auch ohne Handy über weite Strecken kommunizieren kann. Wir machen jetzt aber nicht den kleinen *Tastfunk-Schein*.

Wer es genauer wissen möchte, kann sich <http://de.wikipedia.org/wiki/Morsecode> anschauen.

2 Codes aus Bit-Kombinationen

2.1 Der 5-Bit Zeichensatz CCITT-2, Baudot-Code

5-Bit Fernschreib-Code CCITT Nr.2.

Beispiel dafür, dass nicht alle Codes aus einem Byte bestehen, sondern auch andere Codes im täglichen Leben vorkommen. Zu Kenntnis nehmen, verstehen, aber nicht auswendig lernen.

00000	NUL	NUL	10000	t	5
00001	e	3	10001	z	+
00010	LF	LF	10010	l	)
00011	a	-	10011	w	2
00100	SP	SP	10100	h	SZ
00101	s	'	10101	y	6
00110	i	8	10110	p	0
00111	u	7	10111	q	1
01000	CR	CR	11000	o	9
01001	d	wer	11001	b	?
01010	r	4	11010	g	SZ
01011	j	BEL	11011	ZI	ZI
01100	n	,	11100	m	*
01101	f	SZ	11101	x	/
01110	c	:	11110	v	=
01111	k	(	11111	BU	BU

Tabelle 1: CCITT-2

BEL *bell*, Klingel oder anderes akustisches Zeichen.

BU *Buchstaben*, auch **A...**, Umschaltung auf Buchstaben.

CR *carriage return*, Wagenrücklauf, Sprung zum Zeilenanfang. Je nach Fernschreibertyp wird dabei die Walze oder der Druckkopf bewegt.

LF *line feed*, Zeilenvorschub. Die Walze beim Fernschreiber wird – wie bei einer Schreibmaschine – einen Zeilenschritt weiter gedreht.

NUL *null*, Leerlauf, keine Operation

SP *space*, auch **ZWR** für Zwischenraum, gemeint ist die Leertaste.

SZ *Sonderzeichen*, frei für die Belegung von nationalen Sonderzeichen.

ZI *Ziffern*, auch **1...**, Umschaltung auf Ziffern und sonstige Zeichen.

2.2 ASCII und andere 7-Bit Zeichensätze

American Standard Code for Information Interchange, **7-Bit** Code amerikanischer Fernschreiber und Grundlage vieler Computer-Systeme.

Wichtig! *Aufwachen!* Hier steckt die Lösung, warum in e-Mail-Adressen, Dateinamen, Verzeichnisnamen usw. **keine** Umlaute und Ligaturen wie *ß* stehen dürfen. Dies ist der **kleinste gemeinsame Nenner** vieler Computer-Systeme. **Merken!** In Klarsichthülle abheften, wird noch oft gebraucht!

dez	bin	hex	ascii	dez	bin	hex	ascii	dez	bin	hex	ascii	dez	bin	hex	ascii
0	00000000	00	NUL (ctl @)	32	00100000	20	SP	64	01000000	40	@	96	01100000	60	'
1	00000001	01	SOH (ctl A)	33	00100001	21	!	65	01000001	41	A	97	01100001	61	a
2	00000010	02	STX (ctl B)	34	00100010	22	"	66	01000010	42	B	98	01100010	62	b
3	00000011	03	ETX (ctl C)	35	00100011	23	#	67	01000011	43	C	99	01100011	63	c
4	00000100	04	EOT (ctl D)	36	00100100	24	\$	68	01000100	44	D	100	01100100	64	d
5	00000101	05	ENQ (ctl E)	37	00100101	25	%	69	01000101	45	E	101	01100101	65	e
6	00000110	06	ACK (ctl F)	38	00100110	26	&	70	01000110	46	F	102	01100110	66	f
7	00000111	07	BEL (ctl G)	39	00100111	27	'	71	01000111	47	G	103	01100111	67	g
8	00001000	08	BS (ctl H)	40	00101000	28	(	72	01001000	48	H	104	01101000	68	h
9	00001001	09	HT (ctl I)	41	00101001	29	)	73	01001001	49	I	105	01101001	69	i
10	00001010	0A	LF (ctl J)	42	00101010	2A	*	74	01001010	4A	J	106	01101010	6A	j
11	00001011	0B	VT (ctl K)	43	00101011	2B	+	75	01001011	4B	K	107	01101011	6B	k
12	00001100	0C	FF (ctl L)	44	00101100	2C	,	76	01001100	4C	L	108	01101100	6C	l
13	00001101	0D	CR (ctl M)	45	00101101	2D	-	77	01001101	4D	M	109	01101101	6D	m
14	00001110	0E	SO (ctl N)	46	00101110	2E	.	78	01001110	4E	N	110	01101110	6E	n
15	00001111	0F	SI (ctl O)	47	00101111	2F	/	79	01001111	4F	O	111	01101111	6F	o
16	00010000	10	DLE (ctl P)	48	00110000	30	0	80	01010000	50	P	112	01110000	70	p
17	00010001	11	DC1 (ctl Q)	49	00110001	31	1	81	01010001	51	Q	113	01110001	71	q
18	00010010	12	DC2 (ctl R)	50	00110010	32	2	82	01010010	52	R	114	01110010	72	r
19	00010011	13	DC3 (ctl S)	51	00110011	33	3	83	01010011	53	S	115	01110011	73	s
20	00010100	14	DC4 (ctl T)	52	00110100	34	4	84	01010100	54	T	116	01110100	74	t
21	00010101	15	NAK (ctl U)	53	00110101	35	5	85	01010101	55	U	117	01110101	75	u
22	00010110	16	SYN (ctl V)	54	00110110	36	6	86	01010110	56	V	118	01110110	76	v
23	00010111	17	ETB (ctl W)	55	00110111	37	7	87	01010111	57	W	119	01110111	77	w
24	00011000	18	CAN (ctl X)	56	00111000	38	8	88	01011000	58	X	120	01111000	78	x
25	00011001	19	EM (ctl Y)	57	00111001	39	9	89	01011001	59	Y	121	01111001	79	y
26	00011010	1A	SUB (ctl Z)	58	00111010	3A	:	90	01011010	5A	Z	122	01111010	7A	z
27	00011011	1B	ESC (ctl [)	59	00111011	3B	;	91	01011011	5B	[	123	01111011	7B	{
28	00011100	1C	FS (ctl \)	60	00111100	3C	<	92	01011100	5C	\	124	01111100	7C	
29	00011101	1D	GS (ctl])	61	00111101	3D	=	93	01011101	5D	]	125	01111101	7D	}
30	00011110	1E	RS (ctl ^)	62	00111110	3E	>	94	01011110	5E	^	126	01111110	7E	~
31	00011111	1F	US (ctl _)	63	00111111	3F	?	95	01011111	5F	_	127	01111111	7F	DEL

Tabelle 2: ASCII - Tabelle

NUL *null*, Leerlauf, keine Operation

SOH *start of heading*, Beginn des Vorspannes

STX *start of text*, Beginn des Textes

ETX *end of text*, eigentlich das Ende-Zeichen von Datenübertragungen, kann aber auch meistens dazu gebraucht werden um laufende Programme abzubrechen.

EOT *end of transmission*, Ende der Übertragung.

ENQ *enquiry*, Anfrage, wird in unterschiedlichen Zusammenhängen unterschiedlich verwendet. Beispiele: Ein ENQ von einem Computer an z.B. ein Terminal kann das Terminal dazu veranlassen seinen Typ und damit seine Fähigkeiten mitzuteilen (so eine Art *plug and play*). Im Zusammenhang einer synchronen Datenübertragung wird das ENQ verwendet den Empfänger zu einer Bestätigung des empfangenen Datenpaketes zu veranlassen. Als Antwort kommt dann vom Empfänger ein ACK oder ein NAK (s.u.).

ACK *acknowledge*, positive Bestätigung, wird oft in einer synchronen, oder Paket-orientierten Datenübertragung dazu benutzt den Erhalt eines intakten Datenpaketes zu quittieren und damit den Sender aufzufordern das nächste Paket zu senden.

BEL *bell*, Signalton (um den Anwender zu wecken).

BS *backspace*, ein Zeichenschritt zurück, entspricht der ←-Taste auf der Tastatur.

HT *horizontal tabulation*, Sprung nach rechts zur nächsten horizontalen Tabulatormarke. Entspricht der TAB oder →|-Taste auf der Tastatur.

LF *line feed*, Zeilenvorschub. Eigentlich sind 2 Zeichen zu verwenden um an den Anfang einer neuen Zeile zu gelangen. Dies sind CR um an den Anfang der gleichen Zeile zu gelangen (beim alten Fernschreiber der Wagenrücklauf) und ein LF um eine Zeile vor zu schieben (die Walze, oder den Traktor bewegen). Bei MS-DOS oder Windows sind auch genau diese Zeichen (CR + LF) im nicht sichtbaren Zeilenende enthalten. Bei UNIX-Systemen hat man sich das CR gespart und das LF ist das vollständige Zeilenende. Apple-Macs haben genau das andere Zeichen gespart und CR macht den Job von CR + LF. Dies muss man wissen, wenn Dateien zwischen diesen Betriebssystemen ausgetauscht werden sollen. Trifft man keine Vorbereitungen, dann kann man z.B. eine ganze Datei in einer Zeile haben (die je nach Rechner/Betriebssystem nach 132 oder 255 Zeichen abgeschnitten wird). Wenn man Dateien von Windows nach Unix kopiert muss man mit spukenden CR bzw. *ctl.M* an den Zeilenenden rechnen, die evtl. einem verarbeitendem Programm Verdauungsprobleme bereiten können.

VT *vertical tabulation*, Vorschub bis zur nächsten vertikalen Tabulatormarke

FF *form feed*, Seitenvorschub. Bei Druckern wird ein neues Blatt Papier verwendet, Terminals machen einen größeren VT oder einen *clear screen*.

CR *carriage return* Sprung zum Zeilenanfang (siehe LF).

SO *shift out* Umschaltung in den alternativen Betrieb, wird z.B. benutzt um Drucker in eine andere, alternative Betriebsart (mit z.B. anderen Zeichensätzen) umzuschalten

SI *shift in* Umschaltung in den Normalbetrieb, also Rückschaltung aus der alternativen Betriebsart (Gegenteil von SO).

DLE *data link escape*

DC1 *device control 1* Gerätesteuerung, stark abhängig vom Gerät, ob damit z.B. unterschiedliche Papierschächte angesteuert werden oder was auch immer beim Gerät umgeschaltet wird. Da der ASCII-Satz nur 4 Umschaltungen zulässt, hat es sich durchgesetzt *Escape-Sequenzen* zu verwenden (siehe ESC).

DC2 *device control 2* siehe DC1

DC3 *device control 3* siehe DC1

DC4 *device control 4* siehe DC1

NAK *negative acknowledge* negative Bestätigung, wird oft in einer synchronen, oder Paket-orientierten Datenübertragung dazu benutzt den Erhalt eines defekten Datenpaketes zu quittieren und damit den Sender aufzufordern das gleiche Paket noch einmal zu senden.

SYN *synchronous idle* Synchronisierzeichen

ETB *end of transmission block*

CAN *cancel* Ungültigkeitsmeldung

EM *end of medium*

SUB *substitute* Ersetzungsbefehl

ESC *escape*, Umschaltzeichen, wird oft in Escape-Sequenzen (ESC, gefolgt von weiteren Zeichen) benutzt um spezielle Befehle an z.B. Ausgabegeräte zu geben (Attributswechsel wie z.B. Farbwechsel usw.)

FS *form separator* Hauptgruppen-Trennzeichen

GS *group separator* Gruppen-Trennzeichen

RS *record separator* Datensatz-Trennzeichen

US *unit separator*

SP *space*, Leerschritt

DEL *delete*, Löschzeichen

2.2.1 nationale Versionen vom ASCII

Es gibt nationale Versionen vom ASCII-Zeichensatz. Dabei sind einzelne Zeichen anders belegt. Betroffen sind z.B. nationale Währungssymbole, Umlaute usw. Bei der deutschen Variante liegen z.B. die Umlaute auf den eckigen und geschweiften Klammern.

Diese nationalen Versionen verlieren zunehmend an Bedeutung. Wenn Sie z.B. einen älteren Drucker an einen Computer anschließen und dieser konsequent Umlaute statt Klammern druckt, sollten Sie diesen Absatz im Hinterkopf behalten und einfach nur wissen, dass es so etwas gibt.

2.3 8-Bit Zeichensätze

Da ein ASCII-Zeichen **7 Bit** hat, aber als Platzbedarf ein Byte, also **8 Bit** verwendet, gibt es schon über 30 Jahre die Bestrebung diese **256 Zustände** auch mit entsprechend vielen Zeichen auszunutzen. Die Hersteller waren sich über die **ersten 128 Zeichen einig, dem ASCII-Zeichensatz**. In den oberen 128 Zeichen kann man dann noch jede Menge Sonderzeichen hineinlegen. So weit, so gut, aber jeder Hersteller hatte eine *andere* Belegung. Wo beim *IBM-PC* Blockgrafik-Zeichen für z.B. Rahmen lagen, lagen beim *Atari ST* hebräische Zeichen und beim *Apple Mac* europäische Umlaute usw. Diese Liste lässt sich beliebig fortführen und es war eine Zeit der Vergleichslisten und Zeichen zu konvertieren die man auf einen anderen Rechner übertragen wollte, oder z.B. auf einem Drucker eines anderen Herstellers ausdrucken wollte. Zu der Zeit habe ich als Studenten-Job bei einer Computerfirma sehr viele Druckertreiber geschrieben, die für *Burroughs*-Computer diverse Drucker anderer Hersteller in Steuerung und Zeichensätzen nutzbar – und damit verkaufbar – machten.

Was man daraus lernen sollte: Alles was man nicht im ASCII-Zeichensatz findet kann mit hoher Wahrscheinlichkeit *Ärger machen*.

In den letzten Jahren kam mit einigen genormten Definitionen etwas Systematik in die Sache. Nur es gibt *viele* genormte Definitionen, die nur funktionieren, wenn man gerade die richtige (für Quelle und Anzeige) erwischt hat.

Sie sollen hier die Problematik von Zeichensätzen (Umlauten und Sonderzeichen) erkennen und brauchen eigentlich nur *ISO-8859-2* z.B. für Ihre Einstellung am Browser merken.

Hier einige Exemplare:

2.3.1 ISO-8859-1, Latin-1 Westeuropäisch

Westeuropäischer Zeichensatz, wird z.B. in GB, Canada, USA verwendet. Die Umlaute und das ß sind mit drin, aber eigentlich sollen folgende Sprachen damit abgedeckt werden: Albanisch, Dänisch, Englisch, Färisch (Faröer Inseln), Finnisch, Galizisch, Irisch, Isländisch, Katalanisch und Schwedisch.

Dies ist auch die *Default*-Einstellung bei vielen *Browsern*, wie z.B. *Mozilla*. D.h. wenn man selbst nichts anderes einstellt, wird ISO-8859-1 verwendet.

2.3.2 ISO-8859-2, Latin-2 Mitteleuropäisch

Zentraleuropäischer Zeichensatz. Dies ist die richtige Wahl für Deutschland und drumherum. Abgedeckt werden auch Kroatisch, Polnisch, Rumänisch, Slowakisch, Slowenisch, Tschechisch und Ungarisch.

2.3.3 ISO-8859-3, Latin-3 Südeuropäisch

Südeuropäisch mit Eperanto, Galizisch, Maltesisch und Türkisch.

2.3.4 ISO-8859-4, Latin-4 Baltisch

Baltisch, mit Estnisch, Lettisch und Litauisch.

2.3.5 ISO-8859-5, Kyrillisch

Kyrillisch. Gilt für Bulgarisch, Mazedonisch, Russisch, Serbisch und Ukrainisch.

2.3.6 ISO-8859-6, Arabisch

Arabisch.

2.3.7 ISO-8859-7, Griechisch

Neu-Griechisch.

2.3.8 ISO-8859-8, Hebräisch

Hebräisch.

2.3.9 ISO-8859-9, Latin-5 Türkisch

Türkisch.

2.3.10 ISO-8859-10, Latin-6 Nordisch

Inuit (Grönland) und Sami (Lappland).

2.3.11 ISO-8859-11, Thai

2.3.12 ISO-8859-12, wurde verworfen

2.3.13 ISO-8859-13, Latin-7 Baltisch

2.3.14 ISO-8859-14, Latin-8 Keltisch

2.3.15 ISO-8859-15, Latin-9 Westeuropäisch

2.3.16 ISO-8859-16, Latin-10, Südeuropäisch

ISO-8859 wird nicht mehr weiterentwickelt und der aktuelle Trend geht zu *UTF-8*. Eine Tabelle mit der Belegung der einzelnen Zeichensätze findet man unter

http://de.wikipedia.org/wiki/ISO_8859

2.4 16-Bit Zeichensätze

Hier findet man den *Alt-Code*, der z.B. von *Word* verwendet wird.

2.5 Zeichensätze aus mehreren 8-Bit Zeichen

2.5.1 UTF-8

8-bit Unicode Transformation Format ist eine weit verbreitete Kodierung für Unicode-Zeichen. Jedes Zeichen wird einer Zeichenkette variabler Länge zwischen einem 8-bit Zeichen und vier 8-bit Zeichen zugeordnet. Die *Internet Engineering Task Force* verlangt von allen neuen Protokollen im Internet, UTF-8 zu verwenden. Dies gilt auch für neue e-Mail-Programme und Webbrowser und somit auch für HTML.

Allerdings ist der aktuelle Stand noch weit davon entfernt.

Mehr Informationen findet man unter:

<http://de.wikipedia.org/wiki/UTF-8>

3 Zahlen und Zahlensysteme

3.1 Dezimalzahlen

Allgemein bekannte Zahlen arabischen Ursprungs zur **Basis 10**. D.h. jede Ziffer (Stelle) kann 10 Zuständen mit den Zeichen 0_{dez} bis 9_{dez} annehmen. Dieses Zahlensystem sollte Ihnen seit der 1. Klasse geläufig sein. Allerdings gibt es noch weit mehr Zahlensysteme.

3.2 Binärzahlen

Zahlen zur **Basis 2**, also mit genau 2 möglichen Zuständen pro Ziffer und den Zeichen 0_{bin} und 1_{bin} , oder *high* und *low*, bzw. *Strom an* oder *Strom aus* auch *true* und *false* usw. Egal wie es bezeichnet oder dargestellt wird, es sind genau zwei Zustände, die man bündeln und unterschiedlich *belegen* kann.

Die Bedeutung und Wertigkeit ist jeweils abhängig vom verwendeten Code. Aus der **Kombination** können je nach **Definition** ASCII-Zeichen, Oktalzahlen, BCD, Hexadezimalzahlen, EBCDIC und vieles mehr werden. Die Nachrichtentechnik kennt noch viele andere Codes um die Störanfälligkeit bei Übertragungen zu minimieren und die Datenmenge zu komprimieren.

Der **Maschinencode** – mit dem Prozessoren arbeiten – besteht auch nur aus solchen Kombinationen von Binärzahlen, die dann als **direkte Arbeitsanweisungen** ausgeführt werden. Damit der Mensch damit programmieren kann, wird dieser **Maschinencode** in einem **Assembler** mit für den Menschen sinnvollen Befehlen abgebildet. Dies ist jetzt aber eine andere Vorlesung.

3.3 Oktalzahlen

Zahlen zur **Basis 8** aus einem *Bündel* von 3 Binaerziffern, also **3 bit**. Der rechten Binaerziffer wird dabei die Wertigkeit $2^0 = 1_{dez}$, der mittleren die Wertigkeit $2^1 = 2_{dez}$ und der linken die Wertigkeit $2^2 = 4_{dez}$ zugeordnet. Daraus ergeben sich für eine Oktalziffer 8 Zustände mit den Zeichen 0_{okt} bis 7_{okt} .

Beispiele:

$$101_{bin} = 5_{okt} = 2^2 + 2^0 = 4_{dez} + 1_{dez} = 5_{dez}$$
$$111110_{bin} = 76_{okt} = 2^5 + 2^4 + 2^3 + 2^2 + 2^1 = 32_{dez} + 16_{dez} + 8_{dez} + 4_{dez} + 2_{dez} = 62_{dez}$$

Das 1. Beispiel ist nur eine 3er-Gruppe und dort bleiben sich die Oktal- und Dezimalwerte gleich, da die 8 Zustände der Oktalgruppe im Dezimalbereich abgebildet werden können.

Im 2. Beispiel wird die Zahl 111110_{bin} in die Gruppen $111_{bin} = 7_{okt}$ und $110_{bin} = 6_{okt}$ aufgeteilt. Zur dezimalen Umrechnung werden die Wertigkeiten der 2er-Potenzen natürlich weitergeführt.

Oktalzahlen werden noch wichtig bei der Vergabe von Zugriffsrechten unter *UNIX*, also nicht völlig verdrängen.

3.4 Hexadezimalzahlen

Zahlen zur **Basis 16** aus einem Bündel von 4 Binaerziffern. Von rechts gesehen werden den Binaerziffern die Werte $2^0 = 1_{dez}$, $2^1 = 2_{dez}$, $2^2 = 4_{dez}$ und $2^3 = 8_{dez}$ zugeordnet. Daraus ergeben sich für eine Hexadezimalziffer 16 Zustände mit den Zeichen 0_{hex} bis 9_{hex} und $A_{hex} = 10_{dez}$ bis $F_{hex} = 15_{dez}$.

Beispiel:

$$111110_{bin} = 3E_{hex} = 62_{dez}$$

Das Prinzip ist das Gleiche wie bei den Oktalzahlen, nur bestehen die *Bündel* jetzt aus **4 Bit**. Also die Zahl 111110_{bin} wird von rechts ausgehend in Gruppen zu 4 Bit aufgeteilt. Geht die Anzahl nicht in Teilen von 4 auf, kann mit führenden Nullen aufgefüllt werden. Also in unserem Beispiel $0011_{bin} = 3_{hex}$ und $1110_{bin} = E_{hex} = 14_{dez}$.

Wichtig! 8, 16, 32, 64 Bit lassen sich prima durch 4 teilen und handlich in Hexadezimalzahlen darstellen.

3.5 BCD = binary coded decimals

Bündel aus 4 Bit, wie bei den Hexadezimalzahlen, nur werden die ersten 10 Zustände von 0 bis 9 verwendet und nach 9 erfolgt der Übertrag. Diesen Code findet man häufig in normalen Taschenrechnern.

3.6 Byte

Bündel von 8 Bit. Je nach verwendetem Code kann es z.B. ein Zeichen aus dem ASCII-Code, oder eine positive, ganze Zahl sein. Im Falle einer Zahl habe die einzelnen Bits die Wertigkeit von $2^0 = 1_{dez}$ bis $2^7 = 128_{dez}$. Dies ergibt also 256 Zustände mit den ganzen Zahlen von 0_{dez} bis 255_{dez} .

3.7 Word

Bündel von 16 Bit analog zum Byte. Das ergibt also 65536_{dez} Zustände mit den Zahlen 0_{dez} bis 65535_{dez} oder anders ausgedrückt $64k$.

In der normalen Technik ist ein *Kilo* gleich Tausend, also entspricht $1kg$ eben $1000g$ und $1kHz$ sind 1000 Schwingungen pro Sekunde. In der Informatik ist ein *Kilo* aber 1024 (eben zur Basis 2 und nicht zur Basis 10). Entsprechend gilt für *Kilo* = 2^{10} , *Mega* = 2^{20} , *Giga* = 2^{30} , *Tera* = 2^{40} , *Peta* = 2^{50} , *Exa* = 2^{60} , *Zetta* = 2^{70} ...

Dies ergibt dann auch regelmäßig Probleme z.B. bei Herstellerangaben von Plattengrößen und man wundert sich, dass die mit $320GByte$ gekaufte Platte nur $296GByte$ Speicherplatz verfügbar hat. Da ist nichts kaputt sondern ein *Kilo* oder *Giga* zur **Basis 10** ist eben eine größere Zahl als zur **Basis 2** – nicht falsch – und verkauft sich damit besser.

3.8 Double-word

Bündel von 32 Bit. Das ergibt $1.0000.0000_{hex} = 4.294.967.296_{dez}$ Zustände mit Zahlen von 0 bis $FFFF.FFFF_{hex} = 4.294.967.295_{dez} = 4G$.

3.9 Quod-word

Bündel von 64 Bit. Das ergibt $1.0000.0000.0000.0000_{hex} = 18.446.744.073.709.551.616_{dez}$ Zustände mit Zahlen von 0 bis $FFFF.FFFF.FFFF.FFFF_{hex} = 18.446.744.073.709.551.615_{dez} = 16ExaByte$

3.10 Integer

Bündel von 16 Bit, wobei 1 Bit als positives oder negatives Vorzeichen gilt und 15 Bit den Zahlenwert repräsentieren. Dies ergibt $64k$ Zustände mit den Zahlen $-32k$ bis $32k$. Entsprechendes gilt bei 32 Bit und 64 Bit.

3.11 Real

Gleitpunktzahl, die aus 2 Bündeln besteht. Das eine Bündel bildet die *Mantisse* und das andere Bündel den *Exponenten*. Wie viel Bits in jedem Bündel sind, hängt davon ab wie groß das Bündel für die Real-Zahl selbst gewählt wird. Gängige Größen sind 16, 32, 64 und 80 Bit.