Buch: ASCII-Code S.396 Speicher S.90

......Diverse Codierungen: S 395

Exzessdarstellung: Gegenüberstell 127 Dargesteilte Werte -128 -100 100 28 78 128 378 228 255 00000000 00011100 01001110 10110010 100000000 11100100 Werte sind jeweils um 128 gegeneinander verschoben

Einerkomplement: Nur Binär Zahl invertieren

speziell: 00000000 = -0, 11111111 = 0

UTF 8 16 32 -> von hinten nach vorne auffüllen und den Rest mit 0en füllen

Anz.	Unicode Zeichen	UTF-8 Codierung
2	0000 0000 - 0000 007F ₁₆	0xxxx xxxx ₂
2	0000 0067 ₁₆	0110 01112
	110 01112	
3	0000 0080 - 0000 07FF ₁₆	110x xxxx 10xx xxxx ₂
3	0000 03A9 ₁₆	1100 1110 1010 10012
	11 1010 10012	CEA9 ₁₆
4	0000 0800 - 0000 FFFF ₁₆	1110 xxxx 10xx xxxx 10xx xxxx ₂
4	0000 2623 ₁₆	1110 0010 1001 1000 1010 00112
	0010 0110 0010 00112	
5-6	0001 0000 – 0010 FFFF ₁₆	1111 0xxx 10xx xxxx 10xx xxxx 10xx xxxx ₂
5	0001 F640 ₁₆	1111 0000 1001 1111 1001 1001 1000 00002
	1 1111 0110 0100 0000	F09F9980 ₁₆
Anz.	Unicode Zeichen	UTF-16 Codierung (Bits) Ebene bei 0
2-4	0000 0000 - 0000 FFFF ₁₆	XXXX XXXX XXXX XXXX
2	0000 0067 ₁₆	0067 ₁₆
	0110 01112	0000 0000 0110 01112
	0110 01112	0000 0000 0110 01112
5-6		0000 0000 0110 0111 ₂
	0110 0111 ₂ 0001 0000 - 0010 FFFF ₁₆	0000 0000 0110 0111 ₂ 1101 10xx xxxx xxxx 1101 11xx xxxx xxx
5-6 6	0110 0111 ₂ 0001 0000 - 0010 FFFF ₁₆ Unicode: 0001 F36B ₁₆	0000 0000 0110 0111 ₂ 1101 10xx xxxx xxxx 1101 11xx xxxx xxx
	0110 0111 ₂ 0001 0000 - 0010 FFFF ₁₆ Unicode: 0001 F36B ₁₆ Berechnung: - 0001 0000 ₁₆	1101 10xx xxxx xxxx 1101 11xx xxxx xxxx 1101 1100 0011 1100 ₂ 1101 1111 0110 1011 ₂
	0110 0111 ₂ 0001 0000 - 0010 FFFF ₁₆ Unicode: 0001 F36B ₁₆ Berechnung: - 0001 0000 ₁₆ F36B ₁₆	1101 10xx xxxx xxxx 1101 11xx xxxx xxxx 1101 11xx xxxx xxxx 1101 1000 0011 1100 ₂ 1101 1111 0110 1011 ₂ entspricht: D83C DF6B _{16BE}
	0110 0111 ₂ 0001 0000 - 0010 FFFF ₁₆ Unicode: 0001 F36B ₁₆ Berechnung: - 0001 0000 ₁₆ F36B ₁₆ 0000 1111 0011 0110 1011 ₂	1101 10xx xxxx xxxx 1101 11xx xxxx xxxx 1101 11xx xxxx xxxx 1101 1000 0011 11002 1101 1111 0110 10112 entspricht: D83C DF6B _{16BE} entspricht: 3CD8 6BDF _{16LE}
6	0110 0111 ₂ 0001 0000 - 0010 FFFF ₁₆ Unicode: 0001 F36B ₁₆ Berechnung: - 0001 0000 ₁₆ F36B ₁₆ 0000 1111 0011 0110 1011 ₂ Unicode Zeichen	1101 10xx xxxx xxxx 1101 11xx xxxx xxxx 1101 11xx xxxx xxxx 1101 1000 0011 11002 1101 1111 0110 10112 entspricht: D83C DF6B _{16BE} entspricht: 3CD8 6BDF _{16LE} UTF-32 Codierung (Bits)
	0110 0111 ₂ 0001 0000 - 0010 FFFF ₁₆ Unicode: 0001 F36B ₁₆ Berechnung: - 0001 0000 ₁₆ F36B ₁₆ 0000 1111 0011 0110 1011 ₂	1101 10xx xxxx xxxx 1101 11xx xxxx xxxx 1101 11xx xxxx xxxx 1101 1000 0011 11002 1101 1111 0110 10112 entspricht: D83C DF6B _{16BE} entspricht: 3CD8 6BDF _{16LE}

Hammingabstand

1) 10101110

2) 10000001

Achtung muss zum Teil mit 0 gefüllt werden! Zweierkomplement

Exzessdarstellung

 $x = 2^{(AnzBit-1)}$

x meistens = 128

 $Dez \rightarrow Exz = + x$ $Exz \rightarrow Dez = -x$

2er Komplement Erklärung (umkehren + 1)

Zweierkomplement		
Ausgangsoahl (Immer positiv)	62	7
Zahl Binär schreiben	0011 1110	1
Zahl invertieren	1100 0001	<-Komplement
Zahl +1	1100 0010	<-Zweierkomplemen

Subtraktion 28-17 mit 5Bit
28-11100; 1/-10001
K17 - 01110->K17+1->01111
(28) 11100
(2K17) +01111
101011->stellenüberlaufignorient

Die Zahl als Binärzahl darstellen, dabei eine genügend grosse Länge auswählen	13 ₁₀ ± 00001101 ₂
Das Vorzeichen wird gewechselt indem alle Bits umgekehrt (invertiert) werden	13 ₁₀ ± 00001101 ₂ -13 ₁₀ ± 11110010 ₂
Es wird zur Zahl 1 dazugerechnet	-13 ₁₀ △ 11110011 ₂

5pszele Zafren im 2er Komplema -15 = 00001111 11110000 invertieren 11110001 i+1	-127 = 011111111 10000000 Anvertieren 10000001 (+1	0 = 00000000 11111111110ywtiaren 100000000 /+1
-128 + 10000000 01111111 /hy-entieres 10000000 /+1	-1 = 60000001 11111110 Anvertieren	-2 = 00000010 11111101 (myertleren 11111110./+1

Hammingcode

Hammingcode

Originalnachricht: 10110010

Nr.	12	11	10	9	- 8	7	6	.5	4	3	2	1
Data	1	.0	1	1	- 1	0	0	- 1	0	0	4	.0

Reine 10 = 1010 Reine 9 = 1001

Achtung kein Übertrag!

5 = 0101

Empfänger: Addiert alle "1" Positionen

Reine 12 = 1100 Reine 10 = 1010 Reihe 9 = 1001 Reihe 8 = 1000 Reihe 5 = 0101

Achtung kein Übertrag!

Total

= 0000 Bei 0 Fehlerfrei sonst entspricht die Zahl der Falschen Stelle

BCD

Ziffer	BCD Code					1	0	0	1	1	0	0	0	= 9 8
. 0	0000				-	+-	-	Ē	-	+-	Ē	F	-	
1	0001					0	0	1	0	0	0	1	1	= 2 3
2	0010					1	n	1	1	1	n	1	1	Pseudo tetrade
3	0011				_	1	v	-	1	-	v	-	-	rseudo tetrade
-4	0100									0	1	1	0	Addition von 6
-5	0101									٠.	_			
4 5 6 7	0110						1	1	1	1	1			
	0111					1	1	0	0	0	0	0	1	Pseudo tetrade
8	1000				-	+-			-	+-	Ē	-	-	
. 9	1001					0	1	1	0					Addition von 6
lear)	1010				1	1								
(eer)	1011				1	1								
(leer)	1100	0	0	0	1	0	0	1	0	0	0	0	1	
(leel)	1101		_	÷	_	÷	-	=	-	+	÷	÷	=	
(reen)	1110		1				2				1			
(1990)	1111									_				

Bildcodierung

kleinste Veränderung von 1 zu 2 z.B.:

= <u>5</u>

Allgemein: Bei BMP + zusätzlich 54 MB

8 Bit = 1 B (Byte)Basis 10

1 kB (Kilobyte) = 1'000 B

1 MB (Megabyte) = 1'000'000 B

1 GB (Gigabyte) = 1'000'000'000 B

1 TB (Terabyte) = 1'000'000'000'000 B

Basis 2

Parität

1 KiB (Kibibyte) = 1'024 B

1 MiB (Mebibyte) = 1'048'576 B

1 GiB (Gibibyte) = 1'073'741'824 B

Even Parity: gerade Anz 1 = 0 sonst 1

Odd Parity: ungerade Anz 1 = 0 sonst 1

1 TiB (Tebibyte) = 1'099'511'627'776 B

Formel Schwarz & Weiss: Länge*Breite:8(=Byte):Einheit z.B. 1920x1080:8:1024(2er) -> 253 KiB

1000(10er) -> 259kB

Für 32 Grau/Farbstufen = 5 bit = 31 mögliche + 0 =

Formel Graustufen: Länge*Breite*(Anz Graustufen(-1) in Binär -> Xbit):8:Einheit z.B. 1920x1080*5(18 -> 10010 -> 5bit):8:1024(2er) -> 1265 KiB

1000(10er) -> 1296 kB

Formel Farbstufen indexiert: ((Länge*Breite*(Anz Farben(-1) in Binär -> Xbit))+(Anz Farben*Bit von Tabelle))/8:Einheit

z.B. ((600x800*8(256-1 -> 111111111-> 8bit)+(256*24))/8/1024(2er) ->

/1000(10er) -> 480,77kB

Oit.	Name/verwendung	Farben
1 Bt	Monochrom	2'+2
4 tit	Verwendet bei EGA-Grafikkarten	2* = 16
6 Bit	Verwendet von den Amiga-Computern für HAM- und Halfbright-Modus	2 ⁶ = 64 (durch speziellen HAM- Mechaniemus aber bis zu 4096)
8 Bi	Verwendet von den MSX2-Computern	2 ⁸ = 256 (durch spezielen HAM5- Mechanismus beim Amiga aber bis zu ca. 2 Millionen)
12 Bit	Verwendet in mehreren NeXT-Workstations	2 ¹² = 4090
15 Bit	Real Color	2° + 32.768
16 Bit	High Color	2'6 = 65.536
24 Bit	True Color	224 = 16.777.216
24 Bit Farbe + 8 Bit Alpha	True Color mit 8-Bit-Alphakanal	2 ²² = 4.294.967.296
30 Bit	Deep Color, z. B. interne Fartitiete bei Flachbettscannern	2 ³⁰ = 1,073,741,624
36 Bit	Deep Color, beispielsweise hochwertige Fotografie	2 ⁸ = 68.719.476.736
42 BR	Deep Color, beispielsweise hochwertige Flachbildfernacher	2 ⁴⁰ + 4.398.045.511.104
48 Bit	Deep Color, beispielsweise hochwertige Flachbettscanner	2 ⁴⁸ = 291.474.976.710.656

SVG Zeichnen

<path>:

M = gehe zu

L= Linie

H = Horizontale Linie

V = Vertikale Linie

C = Kurve

S = Sanfte Kurfe

Q = quadratic Bézier curve

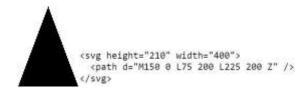
T = smooth quadratic Bézier curveto

A = elliptical Arc

Z = zurück zum Ausgangspunkt

Beispiel SVG

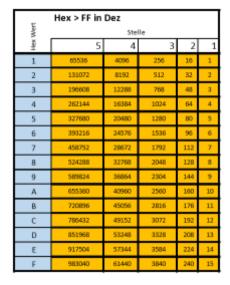
Path



Circle



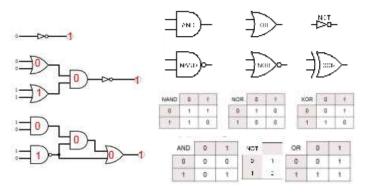
Zahlensysteme

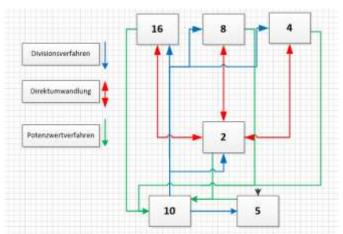




	17	16	15	14	13	12	11	10	9	- 8	7	- 6	5	4	3	2	1
X	X^16	X^15	X^14	X^13	X^12	X^11	X^10	X^9	X^8	XA7	X^6	X^5	X^4	X/3	X^2	X^1	XAG
2	65536	32768	16384	8192	4096	2048	1024	512	256	128	-64	32	16	8	4	-2	1

Logische Rechenoperationen





HEX Berechnung: ABCD

+DCBA D = 13

A = 10

S.396: 23dez=17hex ABCD

+DCBA 1111 18887

Additonsregeln
0+0=0;
0+1=1;1+0=1
1+1 =0 +1 Übertrag
1+1+1 =1 +1 Übertrag

Potenzverfahren

Bin > 8Bit in Dea	z															
Dezimal Wert	32768	16384	8192	4096	2048	1024	512	256	128	64	32	16	8	4	2	1
Binärwert	0	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0
Rechnung	0+	0+	8192	+0+	2048	+0+	512	+0+	128	+0+	32	+0+	8	+0+	2	

Direktumwandlung

decinal	RINV	deginal	RINA
0	0000		2000
5	0001	19	1001
2	0010	4	2010
3	1100		1011
4	0100	c	1100
\$	0101	D	1101
	0110	E	1110
7	0111		1111

0100

1001

Bindr	Okto
000	0
001	1
010	2
011	3
100	74
101	5
110	6
111	7

	Binär	4er
	00	0
	01	1
Ø.	10	2
70	11	3

11	01	00 106)= 11150	102(1)
1:	1	01	00	10
3		1	- 0	2

0100 1001 1010 0000 0010(z)-49A02(zs)

1010	0000	0010
Α	0	2

001	001	001	101	000	.000	01

Divisionsverfahren

Dez >255 in	HEX		ı
Alternative D	ez -> B	Sin->Hex	ı
5116 : 16 =	319	Rest 12 -C	ı
319 : 16 -	19	Rost 15 -F	ı
19:16=	1	Rest 3	ı
1:16=	0	Rest 1	ľ

ı	Rest 0	617	1234:2=
ı	Rest 1	306	617:2=
ı	Baset 0	154	301:2=
ı	Rest 0	- 33	154:2=
ı	Rest 1	31	27:2=
ı	Rest 0	19	38:2-
ı	Rest 1	- 9	19:2-
ı	Revi 1	- 4	9:2-
ı	Bast 0	- 2	4:2-
ı	Bast 0	1	2:2-
ı	Best 1	0	1:2-

Dez >255 in Bin

5116(10)=13FC(16)

