

Codes (1)

Beispiele für die Bedeutung (Semantik) eines n-bit-Wortes:

- Befehl (instruction)
- Zahl
- Zeichen oder Symbol (character)
- Bildelement

ASCII (bzw, ISO 7-bit) - Zeichencode

	$b_6b_5b_4$	$b_3b_2b_1b_0$	0 000	1 001	2 010	3 011	4 100	5 101	6 110	7 111
0	0000		NUL	DLC	SP	0	@	P		D
1	0001		SOH	DC1	!	1	A	Q	a	q
2	0010		STX	DC2	!"	2	B	R	b	r
3	0011		ETX	DC3	#\$	3	C	S	c	s
4	0100		EOT	DC4	\$%	4	D	T	d	t
5	0101		ENQ	NAK	%&	5	E	U	e	u
6	0110		ACK	SYN	&'	6	F	V	f	v
7	0111		BEL	ETB	'(7	G	W	g	w
8	1000		BS	CAN	()	8	H	X	h	x
9	1001		HT	EM)*	9	I	Y	i	y
A	1010		LF	SUB	*+	:	J	Z	j	z
B	1011		VT	ESC	+.	;	K	[k	{
C	1100		FF	FS	.-	<	L	\	l	
D	1101		CR	GS	./	=	M]	m	}
E	1110		SO	RS	./	>	O	^	o	~
F	1111		SI	VS	/	?		_		DEL

← Control characters

 ← Printing characters

Codes (2)

Erweiterungen des ASCII-Codes:

- IS 646 (Latin-1) (8 bit - Code)
- IS 8859 (code pages)
 - IS 8859-1 (Latin-1)
 - IS 8859-2 (slawische Sprachen)
 - IS 8859-3 (z.B. türkisch, maltesisch, Esperanto)
- IS 10646 (Unicode)

Unicode:

- 65,536 code points (16 bit - Code)
- enthält Latin-1 als Untermenge (336 code points□)
- weitere Sprachenbeispiele: griechisch (144), kyrillisch (256), armenisch (96), jüdisch (112)
- code points für Sonderzeichen: z.B.: Indizes (48), Währungssymbole (48), math. Symbole (256)
- Symbole für chinesisch, japanisch, koreanisch
- 6,400 code points frei definierbar für weitere Spezialsymbole
- steigende Akzeptanz, wird schon unterstützt von einigen Programmiersprachen (Java) und Betriebssystemen (Windows NT)

Codes (4)

Bit-map:

- Transformation eines Bildes in ein rechteckiges Feld mit Bildpunkten (Pixel)
- Pixel: kleinste Bildinformationseinheit (Analogie zu bit)
- Pixel kann Attribute besitzen (z.B. Farbe)
- sehr kapazitätsintensiv (Speicher, Bandbreite)
- komplexe Komprimierungstechniken (z.B. Mpeg zur Bildübertragung)

Distanz zwischen 2 Binärwörtern

Table 4.4 The Hamming distance

Word 1	00101101	00101101	00101101	00101101
Word 2	00101100	11101100	11101101	00100101
Places different		✓✓	✓✓	✓
Hamming distance	1	3	2	1

The Hamming distance is simply the number of positions in which two binary sequences differ.

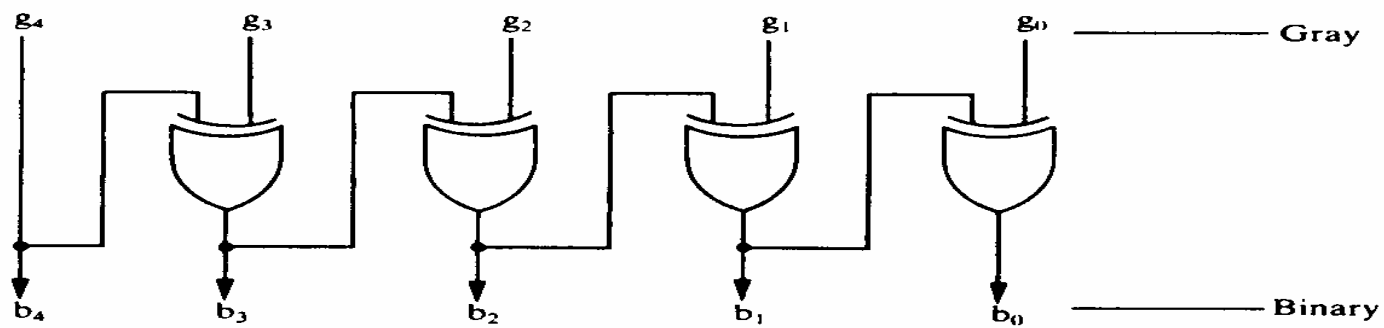
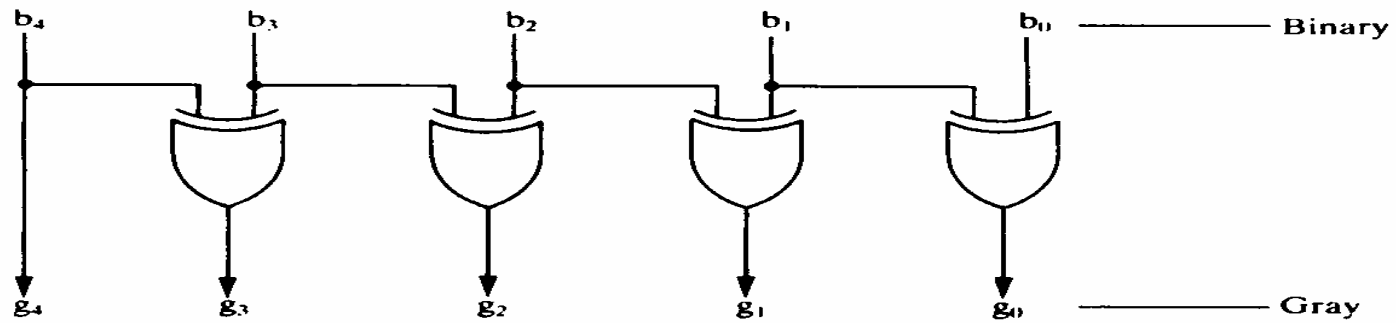
Codes (4a)

Der 4-bit GRAY CODE

Decimal value	Natural binary value	Gray code
0	0000	0000
1	0001	0001
2	0010	0011
3	0011	0010
4	0100	0110
5	0101	0111
6	0110	0101
7	0111	0100
8	1000	1100
9	1001	1101
10	1010	1111
11	1011	1110
12	1100	1010
13	1101	1011
14	1110	1001
15	1111	1000

Codes (5)

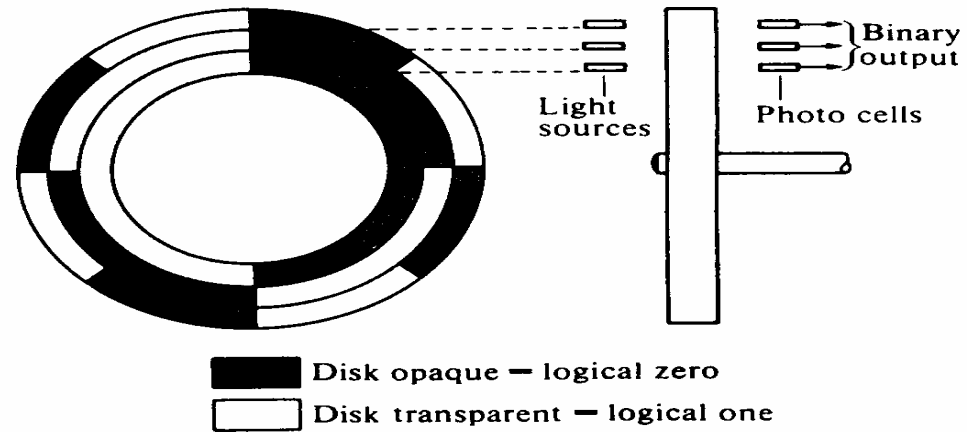
Umwandlung Binär Codes in Gray Codes



Codes (5a)

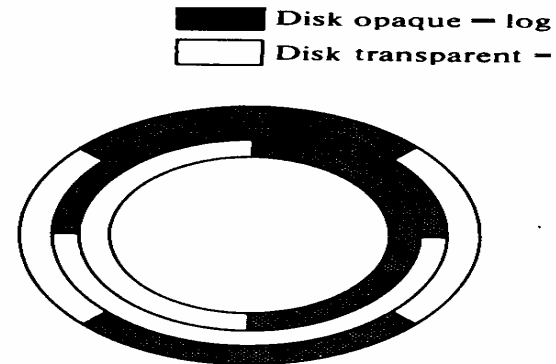
Optischer Codierer mit BINÄR CODES

Sector	Angle	Binary code
0	0-45	0 0 0
1	45-90	0 0 1
2	90-135	0 1 0
3	135-180	0 1 1
4	180-225	1 0 0
5	225-270	1 0 1
6	270-315	1 1 0
7	315-360	1 1 1



Optischer Codierer mit GRAY CODES

Sector	Angle	Gray code
0	0-45	0 0 0
1	45-90	0 0 1
2	90-135	0 1 1
3	135-180	0 1 0
4	180-225	1 1 0
5	225-270	1 1 1
6	270-315	1 0 1
7	315-360	1 0 0



Codes (14)

Huffman - Code:

Kodierung von 4 Lebensmittelartikeln mit einem 2 Bit - Binärcode:

Item	Code
Potatoes	00
Onions	01
Beans	10
Avocado pears	11

Der entsprechende Huffman - Code:

Item	Percentage of Code transactions	Code
Potatoes	75	0
Onions	12.5	10
Beans	6.25	110
Avocado pears	6.25	111

Beispiel für die Darstellung des Huffman - Codes als Binärbaum:

