Rechner Architektur I (RAI)

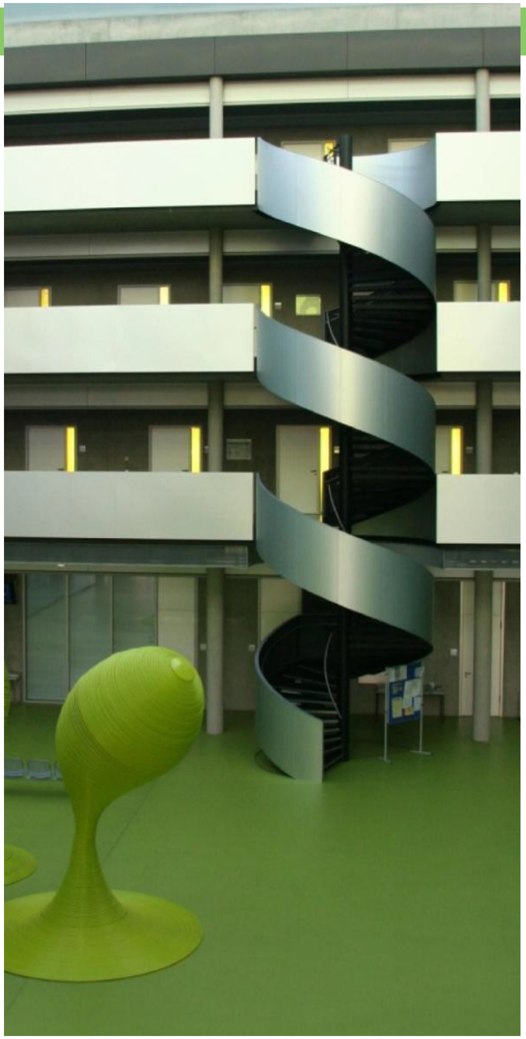
Informationskodierung

Prof. Dr. Akash Kumar

Chair for Processor Design

Gliederung

- Zielstellung
- Kodierung
- Binäre Kodierung
- Wichtige binäre Kodierungen
- Hamming-Abstand
- Fehlererkennung und –korrektur
- Hamming-Kode
- Zusammenfassung



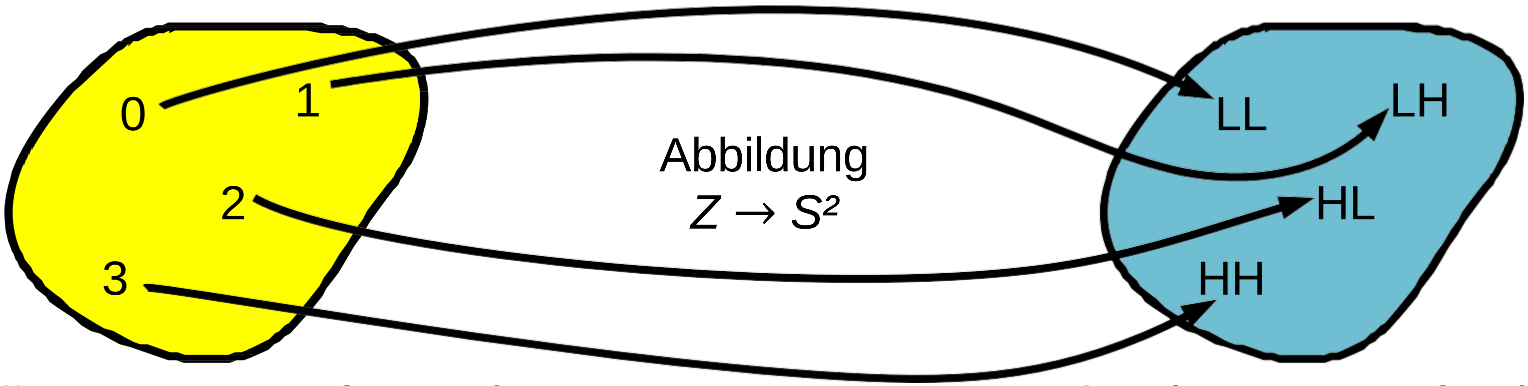
Zielstellung

- Erlangung eines Grundverständnisses für die Kodierungstheorie
- Auseinandersetzung mit den Begriffen Kodierung, Kodewort, Abbildung
- Kennenlernen des Zusammenhanges Information – Signal
- Erwerb von Grundlagen über binäre Kodierungen und binäre Blockkode
- Verständnis im Umgang mit üblichen binären Blockkode

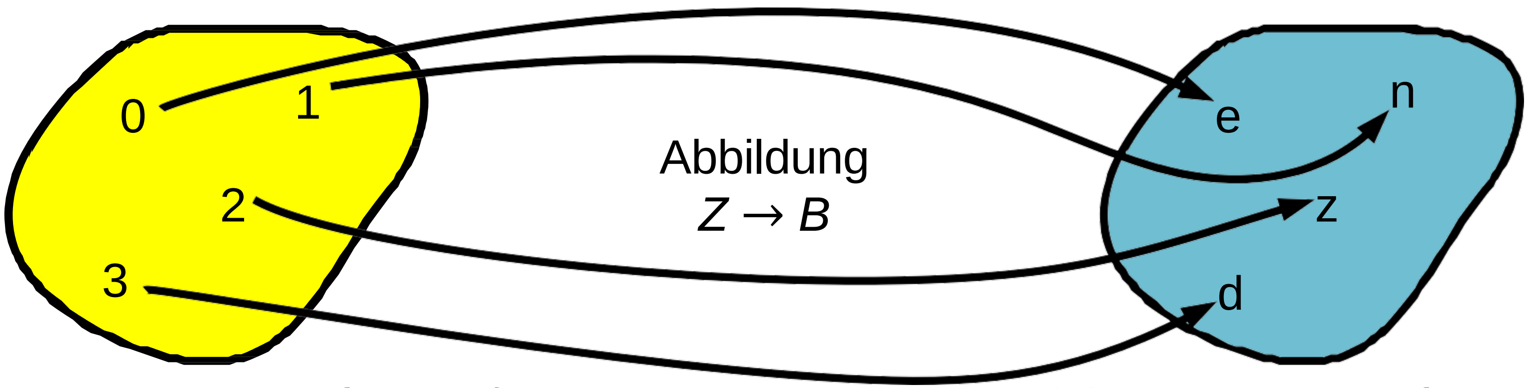
Kodierung

- Informationen können in kodierter Form als Zeichen bzw. Zeichenfolgen einer Zeichenmenge *Z* oder als Signale bzw. Signalfolgen einer Signalmenge *S* vorliegen.
- Kodierung ist eine eindeutige Abbildung einer endlichen Menge von Zeichen eines Alphabetes *A* in eine geeignete Folge über der unterliegenden
  - Signalmenge S^n ($S^n = S \times \ldots \times S$),
  - Zeichenmenge Z^n ($Z^n = Z \times \ldots \times Z$),
  - Zeichenmenge eines anderen Alphabetes *B.*
- Kodierung ist Alphabetwandlung: $\kappa : \mathrm{A} \rightarrow Z^{\mathrm{n}}$ bzw. $\kappa : \mathrm{A} \rightarrow S^{\mathrm{n}}$ oder $\kappa : \mathrm{A} \rightarrow \mathrm{B}$.
- Dabei bezeichnet S^n bzw Z^n das *n*-fache kartesische Produkt über der Menge
- *S* bzw. *Z*. Die Elemente von S^n bzw Z^n sind *n*-Tupel, Vektoren (Zeichenketten, Kodewörter) mit $(s_1 s_2 \ldots s_v \ldots s_n) \in S^n$, $s_v \in S$ bzw. $(z_1 z_2 \ldots z_v \ldots z_n) \in Z^n$, $z_v \in Z$.
- Kodes können in Kodetafeln, Kodetabellen dargestellt werden.

Kodierung als Abbildung



Ziffernmenge $Z = \{0,1,2,3\}$ Signalmenge $S = \{H,L\}$



Ziffernmenge $Z = \{0,1,2,3\}$ Zeichenmenge $B = \{n,e,z,d\}$

Beispiel Morse-Alphabet

- Signalmenge *M={●,—,○}* = *{kurzer Ton, langer Ton, Pause}*
- Kodierung: μ: *{A…Z, 0…9}* → *{●,—,○}*^{n}

A	● —	J	● — — —	S	● ● ●	1	● — — — —
B	— ● ● ●	K	— ● —	T	—	2	● ● — — —
C	— ● — ●	L	● — ● ●	U	● ● —	3	● ● ● — —
D	— ● ●	M	— —	V	● ● ● —	4	● ● ● ● —
E	●	N	— ●	W	● — —	5	● ● ● ● ●
F	● ● — ●	O	— — —	X	— ● ● —	6	— ● ● ● ●
G	— — ●	P	● — — ●	Y	— ● — —	7	— — ● ● ●
H	● ● ● ●	Q	— — ● —	Z	— — ● ●	8	— — — ● ●
I	● ●	R	● — ●	0	— — — — —	9	— — — — ●

(z.B. SOS: ● ● ● ○ — — — ○ ● ● ●)

Beispiel Morse-Alphabet

● ● ● ○ — — ○ ● ● ●

Beispiel Morse-Alphabet

- Eigenschaften der Morsekodierung
  - Kodewörter (ungleichmäßig) unterschiedlicher länge (n = 1...5),
  - nicht alle möglichen Kodewörter werden verwendet,
  - Ziffern haben Kodewörter fester Länge (n = 5) mit gewisser Systematik,
  - kürzeste Kodewörter für E und T,
  - die Pause (○) dient ausschließlich der Trennung der Kodewörter.
- Schlussfolgerungen
  - Kode mit Redundanz
  - Dekodierbarkeit durch Pausenzeichen (Trennzeichen) gesichert,
  - erleichterte Dekodierbarkeit von Ziffern, Zahlen durch feste Kodewortlänge und Systematik,
  - häufig vorkommende Zeichen haben eine kurze Kodewortlänge.

Dekodierbarkeit eines Kodes

- Eine Kodierung ist dekodierbar, wenn mindestens eine der folgenden Voraussetzungen gegeben ist:
  - alle Kodewörter sind gleich lang (Blockkode-Eigenschaft),
  - Verwendung eines gesonderten Trennzeichens,
  - kein (kurzes) Kodewort ist Anfang bzw. Ende eines anderen (langen) Kodewortes (Präfixeigenschaft).
- **Suffix-Kode**: Ungleichmäßiger Kode, bei dem kein (kurzes) Kodewort Ende (Suffix) eines anderen (langen) Kodewortes darstellt.
- **Präfix-Kode**: Ungleichmäßiger Kode, bei dem kein (kurzes) Kodewort Anfang (Präfix) eines anderen (langen) Kodewortes ist.
  - (Präfixeigenschaft: z.B. Huffmann-Kode).

- Suffix- und Präfix-Kode in der Computertechnik haben geringer Bedeutung.

Zielstellung der Kodierung

- **Beeinflussung der Informationsdarstellung durch gezielte Kodierung:**
  - Lesbarkeit
  - Verarbeitbarkeit
  - Übertragbarkeit
  - Fehlersicherheit
  - Speicherbarkeit
  - Vertraulichkeit
- **Anwendungsgebiete für die Kodierung:**
  - Informationsdarstellung allgemein (Signalfolgen)
  - Informationsverschlüsselung (Kryptographie)
  - Informationsübertragung (Kommunikationstechnik)
  - Informationsverarbeitung (Computertechnik)

Binäre Kodierung

- Binär bedeutet zweiwertig, dual, bivalent. Kodierungen für moderne elektronische Computer basieren praktisch ausschließlich auf der Menge
- *B* = *{0,1}* der binären Zeichen *0* und *1*. Die binären Zeichen $0{,}1 \in B$ werden physikalischen Signalen (zweiwertigen Zuständen) zugeordnet, z.B.:

Zeichen	0^{+} (1^{-})	1^{+} (0^{-})
Schalter	offen	geschlossen
Spannung	niedrig	hoch
Pegel	L (low)	H (high)
Kondensator	entladen	geladen
Magnetfeld	neg. Orientierung	pos. Orientierung

+ positive Logik, (- negative Logik)

➔ Im Weiteren soll nur noch positive Logik verwendet werden.

Binäre Kodierung mit Zeichenfolgen

- Mit $B = \{0,1\}$ können nur 2 Zeichen, *0* oder *1* dargestellt bzw. kodiert werden.
- ➔Übergang zu Zeichenfolgen, Zeichenketten der binären Zeichen $0,1 \in B$.
- Eine binäre Kodierung ist eine eindeutige Abbildung einer endlichen Menge von Zeichen eines Alphabetes *A* in geeignete Folgen von nur zwei verschiedenen (binären) Zeichen der unterliegenden binären Zeichenmenge *B.*

$\kappa : A \rightarrow B^n \quad ; \quad B^n = B \times \ldots \times B$

B^n bezeichnet das n-fache kartesische Produkt über der Menge *B.*

Die Elemente von B^n sind n-Tupel, Vektoren mit:

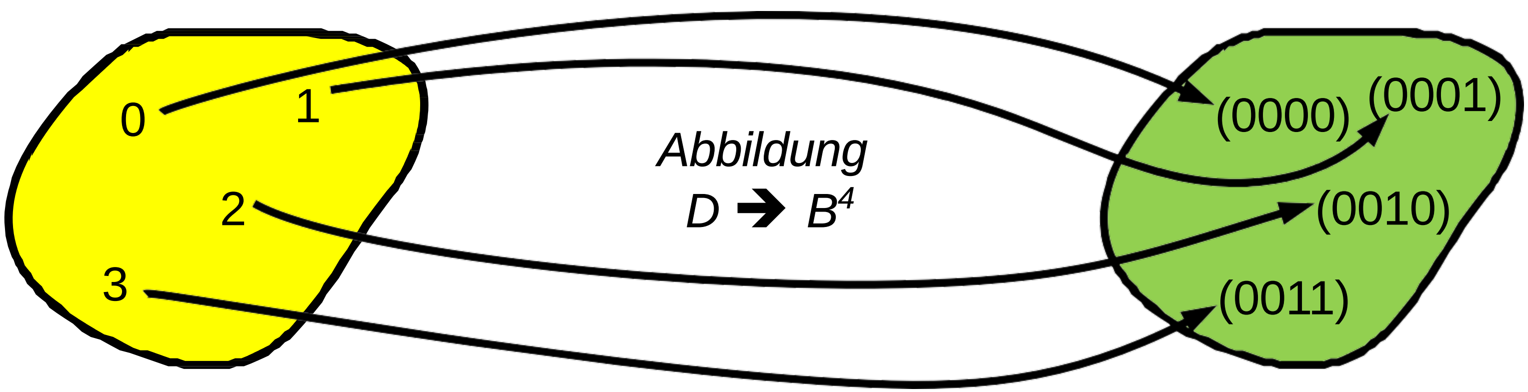
$(b_1 b_2 \ldots b_v \ldots b_n) \in B^n$ *und* $b_v \in B$

Sie werden als binäre Kodewörter der Länge *n* bezeichnet (*n*-Bit Kodewort).

Binäre Kodierung als Abbildung

- Beispiel: Kodierung von Dezimalziffern als binäre 4-Bit Kodewörtern

dezimale Ziffernmenge:	$D = \{0,1,2,3\}$
binäre Zeichenmenge:	$B = \{0,1\}$
Kodierung:	$\beta\text{: } D \rightarrow \{0,1\}^4$
Kodewortlänge:	4 Zeichen (4-Bit Zeichenfolge)
Zeichenvorrat:	$2^4 = 16$ 4-Bit Kodewörter



Ziffernmenge $D = \{0,1,2,3\}$

Teilmenge von B^4

Binäre Blockkodes

- In der Computertechnik dominieren aufgrund der leichteren Dekodierbarkeit binäre Kodes fester Länge *n* (*n*-Bit-Blockkode).

Kodierung:	$\beta: A \rightarrow \{0,1\}^n$
Kodewort:	$(b_1 b_2 \dots b_v \dots b_n) \in B^n$ und $b_v \in B$, $B = \{0,1\}$
Kodewortlänge:	*n*-Bit Kodewörter
Zeichenvorrat:	2^n verschieden Kodewörter

Die Kodewörter werden als Zeichenketten, Vektoren der Binärziffern *0* und*1* dargestellt. Eine Binärziffer wird Bit (**bi**nary digi**t**) genannt.

Beispiel $n = 4$ ➔ 16 verschiedene mögliche 4-Bit Kodewörter:

$\{0,1\}^4$ = {0000, 0001, 0010, 0011, 0100, 0101, 0110, 0111, 1000, 1001, 1010, 1011, 1100, 1101, 1110, 1111}

Binäre Blockkodes

- Ein binärer Blockkode mit n Stellen (n-Bit-Kodierung) mit $(b_1 b_2\, b_v \ldots\, b_n) \in B^n$ *und* $b_v \in B$, $B = \{0,1\}$ realisiert maximal 2^n verschiedene n-Bit lange Kodewörter (n-Bit Kodewörter).
- **Dichter Kode:** Ein dichter Blockkode liegt vor, wenn q verschiedene Kodewörter für die Abbildung benötigt werden und n die folgende Bedingung erfüllt:
  - $2^{n-1} < q \leq 2^n$.
- Eine Reduktion von n ist hier nicht mehr möglich.
- **Voller Kode:** Ein Blockkode wird dann voll genannt, wenn q verschiedene Kodewörter für die Abbildung benötigt werden und n genau die folgende Bedingung erfüllt:
  - $q = 2^n$.
- Damit ist kein weiteres Kodewort mehr darstellbar.

Übliche Formate binärer Blockkodes

- Folgende Kodewortlängen und Bezeichnungen sind für die binären Blockkode in der Computertechnik üblich:

n= 1				x	Bit
n= 4				xxxx	Halbbyte (Nibble)
n= 8			xxxx	xxxx	Byte
n= 16	xxxx	xxxx	xxxx	xxxx	16-Bit Wort
n= 32					32-Bit Wort
n= 64					64-Bit Wort

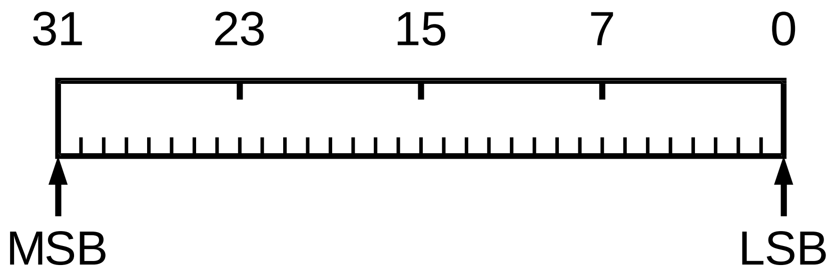
Orientierung der Wortbreite an der Verarbeitungsbreite des Computers:

- 16-Bit Mikroprozessoren → 16-Bit Wort,
- 32-Bit Mikroprozessoren → 32-Bit Wort,
- 64-Bit Mikroprozessoren → 64-Bit Wort.

Übliche Wortunterteilungen: Halbwort, Wort, Doppelwort, Quadwort.

Kodewortdarstellung binärer Blockkode

Beispiel: Darstellung von Kodewörtern (Bits im 32-Bit Wort)



LSB: Least Significant Bit
MSB: Most Signification Bit

Übliche Dimensionsangaben: 1 Byte = 2 Halbbyte = 8 Bit

KiB	(Kibibyte)	:	2^{10} Byte			=	1024 Byte
MiB	(Mebibyte)	:	2^{20} Byte	=	1024 KiB	=	1 048 576 Byte
Gi	(Gibibyte)	:	2^{30} Byte	=	1024 MiB	=	1 073 741 824 Byte
Ti	(Tebibyte)	:	2^{40} Byte	=	1024 GiB	=	1 099 511 627 776 Byte
PiB	(Pebibyte)	:	2^{50} Byte	=	1024 TiB	=	1 125 899 906 842 624 Byte

Wichtige binäre Kodes

- Binär-Kode
- BCD-Kode
- Hexadezimal-Kode
- Oktal-Kode
- M-aus-N-Kode (1-aus-N-Kode)
- Gray-Kode
- ASCII-Kode
- Zahlendarstellung

Binär-Kode (BIN)

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

Bildungsvorschrift:
Die Zuordnung erfolgt entsprechend dem Binäräquivalent (Dualzahl).

Anwendung:
Adressen in Computern, Dualzahlendarstellung .

Binäre Zahlendarstellung als Dualzahl

Jeder Dezimalzahl wird die zu ihr wertmäßig äquivalente binär kodierte Dualzahl zugeordnet (→ Stellenwertsystem zur Basis 2).

Das dezimale Zahlensystem wird direkt auf das Dualzahlensystem abgebildet:

Dezimalzahlen:	$g \in G$
Dezimalziffern:	$z_i \in Z = \{0,1,2,3,4,5,6,7,8,9\}$
Dualzahlen:	$d \in D$
Dualziffern:	$b_j \in B = \{0,1\}$
Kodierung:	$\delta : G \rightarrow D$

Wertgleiche Zuordnung:

$$g = \sum_{i=0}^{m-1} z_i \cdot 10^i = d = \sum_{j=0}^{n-1} b_j \cdot 2^j$$

Beispiel: 314_{10} = $0000\ 0001\ 0011\ 1010_2$

BCD-Kode (Binary Coded Decimals)

DEZ	BCD	Aiken	3XS
0	0000	0000	0011
1	0001	0001	0100
2	0010	0010	0101
3	0011	0011	0110
4	0100	0100	0111
5	0101	1011	1000
6	0110	1100	1001
7	0111	1101	1010
8	1000	1110	1011
9	1001	1111	1100

Bildungsvorschrift:
Die Zuordnung erfolgt entsprechend dem Binäräquivalent 0-9.

Anwendung:
Darstellung von Dezimalziffern.

Die nicht in der Abbildung berücksichtigten Tetraden (10-15) werden als Pseudotetraden bezeichnet (6 Pseudotetraden).

Binäre Zahlendarstellung im BCD-Kode

Jeder Dezimalziffer wird genau ein 4-Bit langes binäres Kodewort (Tetrade) entsprechend ihrem Binäräquivalent zugeordnet (auch 8241-Kode).

Die Codierung einer n-stelligen Dezimalzahl erfolgt ziffernweise durch Aneinanderreihung der BCD-kodierten Dezimalziffern.

Dezimalziffern: $z_i \in Z = \{0,1,2,3,4,5,6,7,8,9\}$
Kodierung: $\delta : Z \rightarrow \{0,1\}^4$

Dem BCD-Kode ähnliche Kodes sind der Aiken-Kode und der 3XS-Kode.

Bei der Rechnung mit BCD-Zahlen sind die Pseudotetraden unbedingt zu beachten.

Beispiel: $314_{10} = 0011\ 0001\ 0100_{BCD}$

Hexadezimale-Kode

DEZ	BIN	HEX
0	0000	0
1	0001	1
2	0010	2
3	0011	3
4	0100	4
5	0101	5
6	0110	6
7	0111	7
8	1000	8
9	1001	9
10	1010	A
11	1011	B
12	1100	C
13	1101	D
14	1110	E
15	1111	F

Bildungsvorschrift:
Die Zuordnung erfolgt zu je 4 Bit entsprechend Binär-äquivalent (Hexadezimalzeichen 0…9, A…F).
Für 10-15 wird A-F kodiert .

Anwendung:
Adressen in Computern,
Hexadezimalzahlendarstellung.

Hexadezimale Darstellung

Für die einfachere Darstellung binärer Kodierungen mit großer Binärstellenzahl werden oft hexadezimale Kodierungen verwendet.

Zahlendarstellung als Hexadezimalzahl: Stellenwertsystem zur Basis 16.

$$g = \sum_{i=0}^{l-1} h_i \cdot 16^i \quad \text{mit} \quad h_i \in H = \{0,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15\}$$

Konvertierung binär → hexadezimal

Unterteilung der binären Kodierung mit dem LSB beginnend in Vierergruppen. Ersetzen der Vierergruppen durch die entsprechenden Hexadezimalzeichen.

Konvertierung hexadezimal → binär

Ersetzen der Hexadezimalzeichen durch ihre Binärkodierung (Bitstellen).

Beispiel: $314_{10} = 013A_{16}$

Oktal-Kode

DEZ	BIN	OKT
0	000	0
1	001	1
2	010	2
3	011	3
4	100	4
5	101	5
6	110	6
7	111	7

Bildungsvorschrift
Die Zuordnung erfolgt zu je 3 Bit entsprechend dem Binäräquivalent (Oktalzeichen 0 – 7).

Anwendung
Adressen in Computern,
Oktalzahlendarstellung.

Oktale Darstellung

Für dein einfachere Darstellung binärer Kodierungen mit großer Binärstellenzahl wurden teilweise auch die oktale Kodierungen verwendet.

Zahlendarstellung als Oktalzahl: Stellenwertsystem zur Basis 8.

$$g = \sum_{i=0}^{k-1} o_i \cdot 8^i \quad \text{mit} \quad o_i \in O = \{0,1,2,3,4,5,6,7\}$$

- **Konvertierung binär → oktal**
- Unterteilung der binären Kodierung mit dem LSB beginnend in Dreiergruppen. Ersetzen der Dreiergruppen durch das entsprechende Oktalzeichen.
- **Konvertierung oktal → binär**
- Ersetzen der Oktalzeichen durch ihre Binärkodierung (Bitstellen).

- **Beispiel:** $314_{10} = 00472_8$

M-aus-N-Kode

Der M-aus-N-Kode hat genau $\binom{N}{M}$ Kodewörter der Länge *N*, die jeweils genau *M* 1-Bits enthalten, sonst alles 0-Bits (Sonderfall: 1-aus-N-Kode).

Kodewortlänge: *N*; Anzahl der 1-Bit-Stelle: M

Anzahl der Kodewörter: $\binom{N}{M} = \frac{N!}{M!(N-M)!}$

Beispiel: 2 aus 4 Kode

$\binom{4}{2} = \frac{4!}{2!(4-2)!} = 2 * 3 = 6$ Kodewörter: 1100, 1010, 1001, 0101, 0011, 0110.

Anwendung: Spezielle Kodierungen, Adressdekodierung, Zahlendarstellung.

Beispiel: 314_{10} = $0000001000\ 0000000010\ 0000010000_{\text{1-aus-10}}$

1-aus-N-Kode (One-Hot-Code)

DEZ	1-aus-10
0	00000 00001
1	00000 00010
2	00000 00100
3	00000 01000
4	00000 10000
5	00001 00000
6	00010 00000
7	00100 00000
8	01000 00000
9	10000 00000

Bildungsvorschrift 1-aus-10-Kode:

Die Dezimalzahl von rechts (LSB) entsprechende Bitstelle ist 1 alle anderen sind 0.

Anwendung:
Adressdekoder, einfache Zahlendarstellung

Der 1-aus-N-Kode (**One-Hot-Code**) hat genau *N* Kodewörter der Länge *N*, die jeweils nur ein 1-Bit enthalten, sonst alles 0-Bit. Die einzelnen Kodewörter unterscheiden sich immer nur in jeweils 2 Bitstellen.

Gray-Kode

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

Benachbarte Kodewörter unterscheiden sich jeweils nur in einer einzigen Bitstelle.

Bildungsvorschrift:

Generierung aus Binärkode

$X_{Gray} := (X_{BIN} \text{ XOR } (1/2 * X_{BIN}))$.

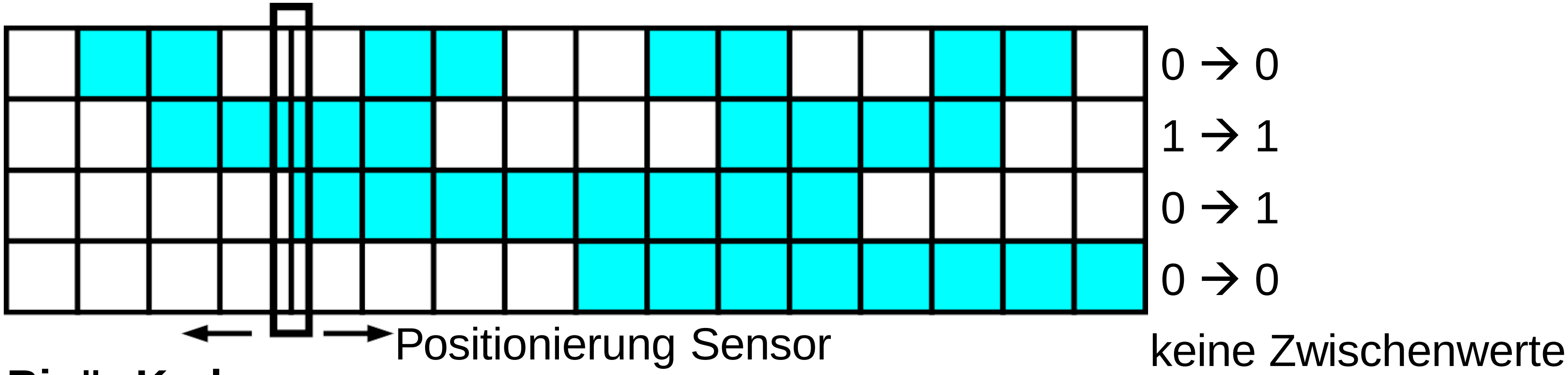
Anwendung:

Messtechnik, Zähler,

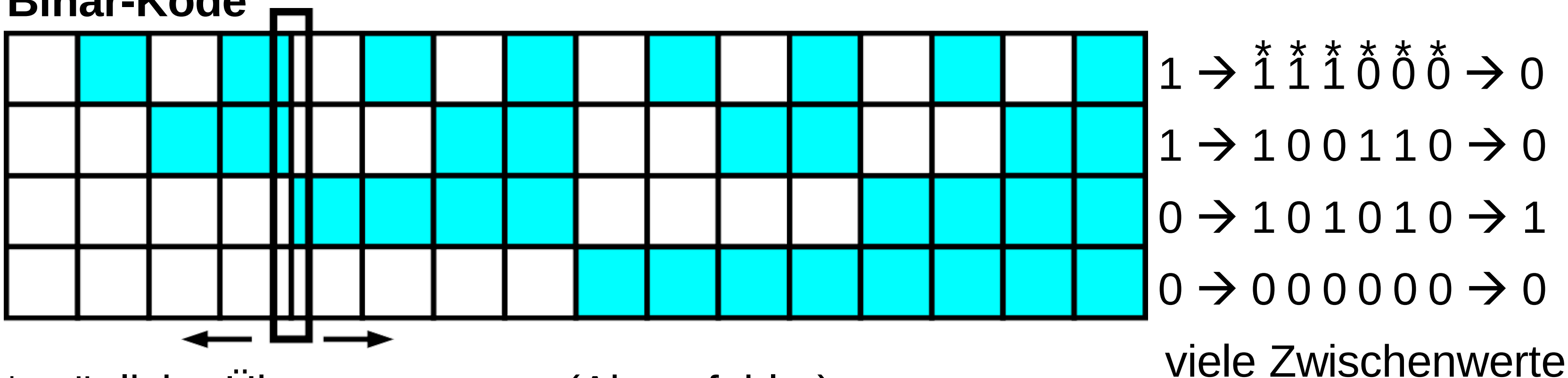
- Erhöhung der Störsicherheit,
- Reduzierung der Verlustleistung.

Positionssensor zur absoluten Lageposition

Gray-Kode



Binär-Kode



* mögliche Übergangswerte (Abtastfehler)

ASCII-Kode

American Standard Code for Information Interchange (ASCII, auch ANSI X3.4-1986) ist ein 7-Bit binärer Blockkode.

Die Zeichenkodierung des ASCII-Kode definiert 128 Zeichen, Davon 33 nicht- druckbare und 95 druckbare.

ASCII-Zeichen: $a \in A$

ASCII-Kodierung: $\alpha : A \rightarrow \{0,1\}^7$

Werden ASCII-Zeichen mit 7-Bit kodiert und im Byte-Format dargestellt, wird die Bitposition des MSB durchgängig mit 0 oder einem Paritätsbit aufgefüllt.

ASCII-Kode ist als gemeinsamer Subcode in fast allen 8-Bit-Zeichenkodierungen und auch im Unicode enthalten (Ausnahme **EBCDIC-Kode**, Extended Binary Coded Decimal Interchange Code, IBM).

UTF-8 ist eine 8-Bit-Kodierung von Unicode, die zum ASCII-Kode abwärtskompatibel ist. Ein Zeichen kann dabei ein bis vier 8-Bit-Wörter einnehmen. → Kodierung variabler Länge, kein reiner Blockkode.

Umschaltzeichen

Mit n-Bit Kodewortlänge können maximal 2^n Zeichen binär kodiert werden.

Erweiterungen bzw. Erhöhung der Anzahl der kodierbaren Zeichensätze durch:

- Erhöhung der Bitstellenanzahl n (Kodewortlänge),
- Einführung von Umschaltzeichen (Zeichensatzanzahl).

Mit u Umschaltzeichen können u verschiedene Zeichensätze mit jeweils 2^n-u Zeichen kodiert werden.

Zeichenvorrat:	$u(2^n-u)$
Maximaler Zeichenvorrat:	2^{2n-2} bei $u = 2^{n-1}$

Umschaltzeichen werden kaum noch verwendet.

ASCII-Kode (ISO-7-Bit-Kode)

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

(ASCII – American Standard Code for Information Interchange) (ISO 7-Bit-Code 646 oder DIN-Norm 66003)

ASCII-Control-Code

Dec	Code	HEX	Name	Dec	Code	Hex	Name
0	NUL	00	Null	17	DC1	11	Device control 1 (XON)
1	SOH	01	Start of heading	18	DC2	12	Device control 2
2	STX	02	Start of text	19	DC3	13	Device control 3 (XOFF)
3	ETX	03	End of text	20	DC4	14	Device control 4
4	EOT	04	End of transmission	21	NAK	15	Negative acknowledge
5	ENQ	05	Enquiry	22	SYN	16	Synchronous idle
6	ACK	06	Acknowledge	23	ETB	17	End transmission block
7	BEL	07	Bell	24	CAN	18	Cancel
8	BS	08	Back space	25	EM	19	End of medium
9	HT	09	Horizontal tab	26	SUB	1A	Substitute (EOF)
10	LF	0A	Linefeed	27	ESC	1B	Escape
11	VT	0B	Vertical tab	28	FS	1C	File separator
12	FF	0C	Formfeed	29	GS	1D	Group separator
13	CR	0D	Carriage return	30	RS	1E	Record separator
14	SO	0E	Shift out	31	US	1F	Unit separator
15	SI	0F	Shift in	...			
16	DLE	10	Data link escape	127	DEL	7F	Delete

ANSI (8-Bit) Code Table for Windows

HEX	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**A**	**B**	**C**	**D**	**E**	**F**	
0	NUL	DEL	SP	0	@	P	`	p				°	À	Đ	à	ð	**0000**
1	SOH	DC1	!	1	A	Q	a	q		`	¡	±	Á	Ñ	á	ñ	**0001**
2	STX	DC2	"	2	B	R	b	r	,	´	¢	^{2}	Â	Ò	â	ò	**0010**
3	ETX	DC3	#	3	C	S	c	s	f	“	£	^{3}	Ã	Ó	ã	ó	**0011**
4	EOT	DC4	$	4	D	T	d	t	“	“	¤	´	Ä	Ô	ä	ô	**0100**
5	ENQ	NAK	%	5	E	U	e	u	...	*	¥	µ	Å	Õ	å	õ	**0101**
6	ACK	SYN	&	6	F	V	f	v	?	-	¦	¶	Æ	Ö	æ	ö	**0110**
7	BEL	ETB	'	7	G	W	g	w	?	_	§	·	Ç	×	ç	÷	**0111**
8	BS	CAN	(	8	H	X	h	x		~	¨	¸	È	Ø	è	ø	**1000**
9	HT	EM	)	9	I	Y	i	y	?	[tm]	©	^{1}	É	Ù	é	ù	**1001**
A	LF	SUB	*	:	J	Z	j	z	S	s	ª	º	Ê	Ú	ê	ú	**1010**
B	VT	ESC	+	;	K	[	k	{	<	>	«	»	Ë	Û	ë	û	**1011**
C	FF	FS	,	<	L	\	l	|			¬	¼	Ì	Ü	ì	ü	**1100**
D	CR	GS	-	=	M	]	m	}			•	½	Í	Ý	í	ý	**1101**
E	SO	RS	.	>	N	^	n	~			®	¾	Î	Þ	î	þ	**1110**
F	SI	US	/	?	O	_	o	DEL		Y	¯	¿	Ï	ß	ï	ÿ	**1111**
	0000	**0001**	**0010**	**0011**	**0100**	**0101**	**0110**	**0111**	**1000**	**1001**	**1010**	**1011**	**1100**	**1101**	**1110**	**1111**	**Bin**

Kodierung zur Zahlendarstellung

OCT	3	0	5	4	1	7	0	2	6	3	1
BIN	11	000	101	100	001	111	000	010	110	011	001

BIN	1100	0101	1000	0111	1000	0101	1001	1001
HEX	C	5	8	7	8	5	9	9

$1100\ 0101\ 1000\ 0111\ 1000\ 0101\ 1001\ 1001_2$ Dualzahl

$= C5\ 87\ 85\ 99_{16}$ Hexadezimalzahl

$= 305\ 4170\ 2631_8$ Oktalzahl

$= 3\ 313\ 993\ 113_{10}$ Dezimalzahl

DEC	3	3	1	3	9	9	3	1	1	3
BCD	0011	0011	0001	0011	1001	1001	0011	0001	0001	0011

Zusammenfassung

- Kodierung ist zwingend notwendig, um Information durch Signale darzustellen, abzubilden
- Die Kodierung bestimmt die Darstellungsvorschrift von Informationseinheiten mit Hilfe der Zeichen eines Alphabetes
- Binäre Blockkode werden in der Computertechnik bevorzugt
- Eine Kodetabelle stellt als tabellarische Zuordnung die Abbildungsvorschrift dar
- Es gibt verschiedene Standardkodes für Zahlendarstellungen, Textdarstellungen, Adressen, Auswahlentscheidungen usw.