

Grundlagen der Informatik

Prof. Dr. J. Schmidt
Fakultät für Informatik

GDI – WS 2018/19
Information und Quellencodierung
Grundbegriffe

gki-buch.de
Errata

- Einschub: Grundbegriffe der Wahrscheinlichkeitsrechnung
- Wie wird in der Informatik der Begriff Information statistisch gedeutet?
- Was versteht man unter der mittleren Wortlänge eines Codes und der Code-Redundanz?
- Welche Beispiele für Codierungen gibt es in der Informatik und welche charakteristischen Merkmale weisen diese auf?



Grundbegriffe der Wahrscheinlichkeitsrechnung



● Relative Häufigkeit h

- Quotient aus Anzahl von Dingen (Ereignissen), die ein bestimmtes Merkmal aufweisen und der Gesamtzahl der auf dieses Merkmal hin untersuchten Dinge

$$h = \frac{\text{Anzahl der Ereignisse, die ein gewünschtes Merkmal aufweisen}}{\text{Anzahl der betrachteten Ereignisse}}$$

- Bedingung: $0 \leq h \leq 1$ gilt immer
- Vorgehensweise zur Bestimmung der relativen Häufigkeit wird auch **Abzählregel** genannt



● Zufallsexperiment

- Vorgang oder Versuch, der dem Zufall unterliegt oder bei dem man aus anderen Gründen den Ausgang nicht vorhersagen kann
- Quantitative Aussagen sind unter Anwendung von mathematischen Methoden der Statistik möglich
 - Wiederholung der Versuche unter **gleichbleibenden Bedingungen**
- Gesamtheit aller möglichen Versuchsergebnisse wird als Menge der **Elementarereignisse** bezeichnet



- Beispiele Elementarereignisse
 - Versuch „einmaliges Werfen einer Münze“
 - Elementarereignisse: { Kopf, Zahl }
 - Versuch „einmaliges Werfen eines Würfels“
 - Elementarereignisse: { 1,2,3,4,5,6 }
 - Versuch „Messung der Lebensdauer einer Glühbirne“
 - Unendliche Menge von Elementarereignissen



- Ermittlung der relativen Häufigkeit durch eine **große Anzahl an Wiederholungen** des Zufallsexperiments

Beispiele

- Wurf einer Münze

Je öfter man eine Münze wirft, desto weniger werden sich h_{Kopf} und h_{Zahl} von dem Wert $\frac{1}{2}$ unterscheiden.

- Würfelspiel

Je öfter man einen Würfel wirft, desto weniger werden sich h_1, h_2, h_3, h_4, h_5 und h_6 von dem Wert $\frac{1}{6}$ unterscheiden.



- Beziehung zwischen mathematischer Wahrscheinlichkeit und relativer Häufigkeit:
Gesetz der großen Zahl

$$p(A) = \lim_{n \rightarrow \infty} h(A)$$

Wahrscheinlichkeit, dass
das Ereignis A eintritt

Relative Häufigkeit des
Ereignisses A

Limes / Grenzwert der Folge

A : betrachtetes Ereignis
 n : Anzahl der Versuche



- Exakte mathematische Definition der Wahrscheinlichkeit durch die

Kolmogorow'schen Axiome

- **Axiom 1:** Die Wahrscheinlichkeit $p(A)$ für das Eintreffen eines bestimmten Ereignisses A ist eine reelle Funktion, die alle Werte zwischen Null und Eins annehmen kann:

$$0 \leq p(A) \leq 1$$

- **Axiom 2:** Die Wahrscheinlichkeit für das Auftreten eines Ereignisses A , das mit Sicherheit eintritt, hat den Wert 1:
$$p(A) = 1$$



- **Axiom 3:** Für sich gegenseitig ausschließende Ereignisse A und B gilt:

$$\begin{aligned} p(A \text{ oder } B) &= p(A) + p(B) \\ p(A \cup B) &= p(A) + p(B) \end{aligned}$$

(**Additionsgesetz**)



- Folgerungen aus den Axiomen:
 - Wahrscheinlichkeit für ein mit Sicherheit nicht eintretendes Ereignis A:
 $p(A) = 0$
 - Wahrscheinlichkeit, dass das Ereignis A nicht eintritt:
 $p(\bar{A}) = 1 - p(A)$ $\bar{A} = \text{„nicht A“}$
 - Wahrscheinlichkeit, dass zwei Ereignisse A und B gemeinsam eintreten:
 $p(A \text{ und } B) = p(A) \cdot p(B)$
 $p(A \cap B) = p(A) \cdot p(B)$
Bedingung:
Ereignisse A und B schließen sich gegenseitig nicht aus und sind voneinander unabhängig



● Beispiele

- Würfeln mit zwei **unterscheidbaren Würfeln** (rot und grün) gleichzeitig
 - Wahrscheinlichkeit grüner Würfel zeigt 1 und roter Würfel zeigt 2

$$p(A \text{ und } B) = 1/6 \cdot 1/6 = 1/36$$

- Würfeln mit **einem Würfel** hintereinander
 - Wahrscheinlichkeit, dass erst eine 1 und dann eine 2 gewürfelt wird

$$p(A \text{ und } B) = 1/6 \cdot 1/6 = 1/36$$

- Würfeln mit zwei **nicht unterscheidbaren Würfeln**
 - Wahrscheinlichkeit, dass 1 und 2 gewürfelt wird

$$p(E) = (1/6 + 1/6) \cdot 1/6 = 1/18$$



- Die vorherigen Folien umfassen nur das absolut Notwendige
- Viele wichtige Begriffe wurden nicht erläutert (z.B. bedingte Wahrscheinlichkeiten)
- Dies folgt in einer separaten Lehrveranstaltung



Statistischer Informationsgehalt



- Betrachtung des Begriffs Information unter einem spezifischen Blickwinkel
 - Mathematisch fassbare Entscheidungsinformation
 - Also
 - nicht semantische Bedeutung einer Information
 - nicht orientiert an dem mit der Nachricht verfolgten Zweck
- Das heißt
 - Zwei Nachrichten (eine mit besonderem Inhalt – eine mit „Unsinn“) können genau die gleiche Menge an Information enthalten.



- Shannonsche Informationstheorie
 - Wurde maßgeblich von Claude Shannon bis 1950 entwickelt
- Zielsetzung
 - Mathematische Beschreibung des statistischen Informationsgehalts $I(x)$
 - eines Zeichens oder Wortes x ,
 - welches mit einer Auftrittswahrscheinlichkeit $p(x)$ vorkommt



- Anforderungen an die mathematische Beschreibung
 1. Je seltener ein bestimmtes Zeichen x auftritt, d.h. je kleiner $p(x)$, desto größer soll der Informationsgehalt dieses Zeichens sein

$$I(x) \sim \frac{1}{p(x)}$$



- Anforderungen an die mathematische Beschreibung

2. **Gesamtinformation** einer Zeichenkette, z.B. $x_1x_2x_3$ soll sich aus der **Summe der Einzelinformationen** ergeben

$$I(x_1x_2x_3) = I(x_1) + I(x_2) + I(x_3)$$

3. Für den **Informationsgehalt** eines mit **Sicherheit auftretenden Zeichens** x , also für den Fall $p(x) = 1$, soll gelten

$$I(x) = 0$$

➔ Logarithmusfunktion erfüllt die formulierten Anforderungen



- Mathematische Beschreibung des Zusammenhangs von Informationsgehalt und Auftrittswahrscheinlichkeit eines Zeichens x

$$I(x) = \log_b \frac{1}{p(x)}$$



b = Maßstab zur Informationsmessung

Festlegung: Zwei Zustände (0 und 1)

$\rightarrow b = 2$



● Statistischer Informationsgehalt

$$I(x) = \text{ld} \frac{1}{p(x)} = -\text{ld} p(x) \quad [\text{Bit}]$$

↗
Zweierlogarithmus

- Anzahl der Elementarentscheidungen, die nötig sind, um eine Nachricht Zeichen für Zeichen eindeutig identifizieren zu können
- Maßeinheit: Bit



- Der Informationsgehalt eines Zeichens

=

Anzahl der Stellen des Binärworts,
das man für eine eindeutige binäre Darstellung des
Zeichens verwenden muss.



- Berechnung Informationsgehalt bei nicht-binären Nachrichten
 - Gegeben: Buchstabe b tritt in einem deutschsprachigen Text mit einer Wahrscheinlichkeit von 0,016 auf
 - Gesucht: Informationsgehalt dieses Zeichens
 - Lösung:

$$I(b) = \lg \frac{1}{0.016} = \frac{\log(\frac{1}{0.016})}{\log(2)} \approx \frac{1.79588}{0.30103} \approx 5.97 \text{ [Bit]}$$



● Umwandlungsgleichung

$$\log_b(x) = \frac{\log_{10}(x)}{\log_{10}(b)} \quad \text{also} \quad \log_2(x) = \text{ld}(x) = \frac{\log_{10}(x)}{\log_{10}(2)}$$

● mit $\log_{10}(2) = \log(2) = 0.30103$

● Schreibweisen:

- $\log_{10}(x)$ wird zu $\log(x)$
- $\log_2(x)$ wird zu $\text{ld}(x)$
- $\log_e(x)$ wird zu $\ln(x)$ mit $e \approx 2.71828...$



Entropie einer Nachricht (1)

- Nachricht setzt sich i.A.
 - aus Zeichen bzw. aus zu Worten verbundenen Zeichen zusammen,
 - die jeweils unterschiedlichen Informationsgehalt tragen, da sie mit unterschiedlicher Häufigkeit auftreten
- Einführung des Begriffs
 - des mittleren Informationsgehalts
 - bzw. Entropie H einer Nachricht,die aus den Zeichen x_1, x_2, \dots, x_n eines Alphabets A besteht



Entropie einer Nachricht (2)

- Summe der mit den Auftrittswahrscheinlichkeiten gewichteten Informationsgehalte der Zeichen

$$H = \sum_{i=1}^n p(x_i) \cdot \lg \frac{1}{p(x_i)} = - \sum_{i=1}^n p(x_i) \cdot \lg p(x_i) = \sum_{i=1}^n p(x_i) \cdot I(x_i)$$

- Der höchste mittlere Informationsgehalt ergibt sich, wenn alle Zeichen mit der gleichen Wahrscheinlichkeit auftreten
- Einheit: Bit / Zeichen



- Andere Interpretation des Entropie-Begriffs...

... zum Vergleich von Nachrichtenquellen

- Je **kleiner** die **Entropie** umso größer die **Sicherheit** mit der man das Auftreten eines bestimmten Zeichens vorhersagen kann.
- Je **höher** die **Entropie** einer Nachrichtenquelle, desto größer ihre **Unsicherheit** (Surprisal).



Ungewissheit einer Nachrichtenquelle (2)

● Beispiel: Zwei Nachrichtenquellen

- $A_1 = \{a, b, c, d\}$ mit den Auftretswahrscheinlichkeiten

$$p(a) = 11/16, \quad p(b) = p(c) = 1/8, \quad p(d) = 1/16$$

- $A_2 = \{+, -, *\}$ mit den Auftretswahrscheinlichkeiten

$$p(+)=1/6, \quad p(-)=1/2, \quad p(*)=1/3$$

$$H_1 = \frac{11}{16} \cdot \lg \frac{16}{11} + \frac{1}{8} \cdot \lg 8 + \frac{1}{8} \cdot \lg 8 + \frac{1}{16} \cdot \lg 16 \approx 1.372 \left[\frac{\text{Bit}}{\text{Zeichen}} \right]$$

$$H_2 = \frac{1}{6} \cdot \lg 6 + \frac{1}{2} \cdot \lg 2 + \frac{1}{3} \cdot \lg 3 \approx 1.460 \left[\frac{\text{Bit}}{\text{Zeichen}} \right]$$

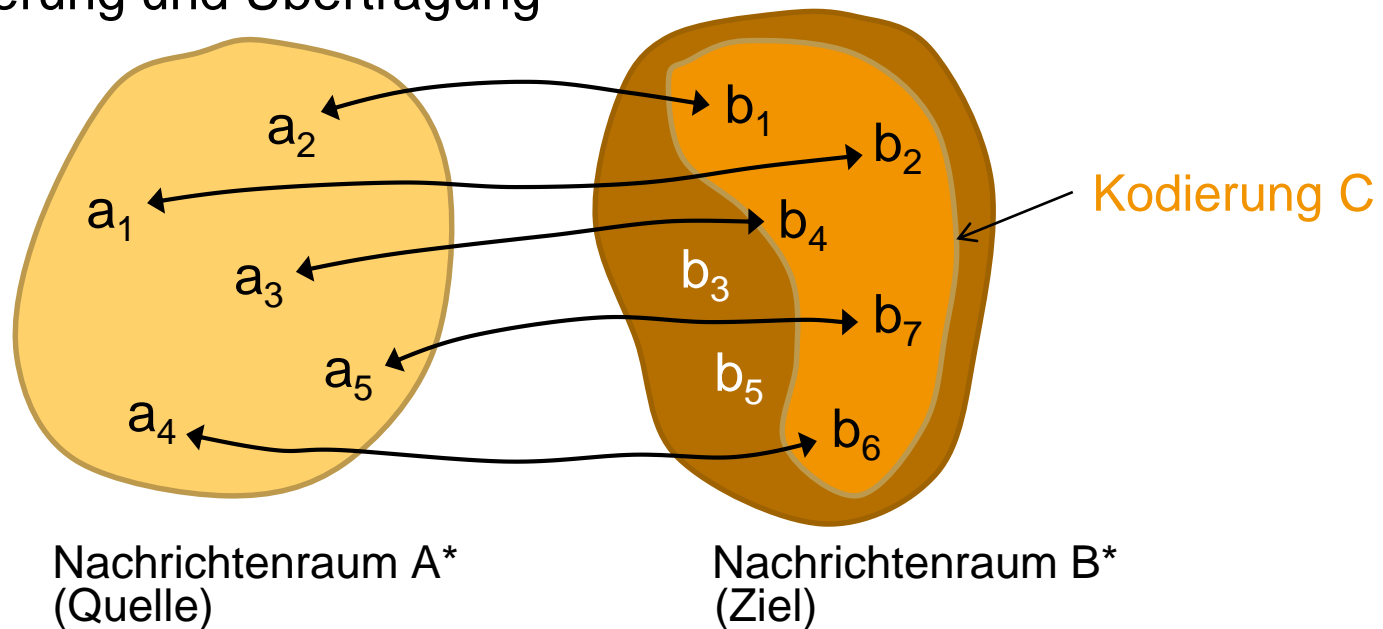
- Ungewissheit für A_2 ist größer als für A_1 , da H_2 größer als H_1



Codierung – Begriffsdefinitionen (1)

Kapitel 3: Information und Quellencodierung – Grundbegriffe

- Zielsetzung: **problemspezifische Darstellung** einer Nachricht bei Speicherung und Übertragung



- **Codierung C**
 - eine **umkehrbar eindeutige Abbildung** von A^* in B^*
 - Beachte: $C \subseteq B^*$ gilt, d.h. C ist eine Teilmenge von B^*
- **Binärcodierung C**
 - Zielmenge ist ein Nachrichtenraum B^* über dem Alphabet $\{0, 1\}$

- Schematische Darstellung der Codierung und Übertragung von Nachrichten



- Erwartungen an „gute“ Codierung
 - Darstellung der zu sendenden Daten mit möglichst wenig Zeichen
 - Möglichst unempfindlich gegen Störungen
 - Code sollte in DV-Anlage leicht zu verarbeiten sein

● Mittlere Wortlänge L

- wesentliches Charakteristikum eines Codes
- definiert als

$$L = \sum_{i=1}^n p_i \cdot l_i$$

mit

- p_i zugehörige Auftrittswahrscheinlichkeit
 - l_i Wortlänge des i -ten Zeichens bzw. Wortes im Zielcode
- Summenbildung über alle n codierten Zeichen



- Shannonsches Codierungstheorem

- Für jede Codierung einer Nachrichtenquelle ist

$$H \leq L$$

- H (Entropie) ist Untergrenze für eine optimale Codierung
 - Fokus: Wortlängenreduktion
- Wenn alle Wahrscheinlichkeiten gleich sind, gilt

$$H = L$$



- Code-Redundanz R

- Differenz aus L und H

$$R_C = L - H \quad (\text{Einheit: Bit/Zeichen})$$

- Gibt an wie groß der Anteil einer Nachricht ist, der im statistischen Sinne keine Information trägt
- Wünschenswert: Codes mit geringer Redundanz
 - geringerer Speicherbedarf und schnellere Nachrichtenübertragung
- Redundanz kann jedoch zur Störsicherheit beitragen
 - Rekonstruktion/Sicherheit bei Datenübertragung/Speicherung



- Quellen-Redundanz R_Q

- Differenz aus maximal möglicher Entropie der Quelle H_0 und tatsächlicher Entropie H

$$R_Q = H_0 - H \quad (\text{Einheit: Bit/Zeichen})$$

- maximale Entropie H_0

- erhält man, wenn alle Zeichen des Alphabets A gleich wahrscheinlich sind. Mit $|A| = n$:

$$H_0 = \sum_{i=1}^n p(x_i) \cdot \lg \frac{1}{p(x_i)} = \sum_{i=1}^n \frac{1}{n} \cdot \lg n = \frac{1}{n} \sum_{i=1}^n \lg n = \frac{1}{n} n \lg n = \lg n$$

- unabhängig vom tatsächlich verwendeten Code



- Unterscheidung Kodierung fest/variabel
 - Kodierung mit **fester** Wortlänge (Block-Codes)
 - alle kodierten Zeichen weisen eine konstante Wortlänge auf
 - Kodierung mit **variabler** Wortlänge
 - Häufig auftretende Zeichen erhalten kurzen Code
 - Selten auftretende Zeichen erhalten langen Code
 - Erster technischer Code mit variabler Länge
 - Morse-Code
(keine Binärkodierung, da drei Zeichen: Punkt, Strich und Pause)



● ASCII-Code

- ASCII = American Standard Code for Information Interchange
- ist eine festgelegte Abbildungsvorschrift (Norm) zur binären Kodierung von Zeichen
 - umfasst Klein-/Großbuchstaben des lateinischen Alphabets
 - (arabische) Ziffern
 - und viele Sonderzeichen
- Kodierung erfolgt in einem Byte
 - => 256 verschiedene Zeichen darstellbar



● ASCII-Code

- Erstes Bit wird vom Standard-ASCII-Code nicht genutzt
 - => 128 Zeichen darstellbar
- Unterschiedliche, speziell normierte, ASCII-Code-Erweiterungen
 - nutzen das erste Bit, um weitere 128 Zeichen darstellen zu können



Beispiele für Codes – Ausschnitt ASCII-Code

Kapitel 3: Information und Quellencodierung – Grundbegriffe

Dezimal	Oktal	Hexa	Binär	Zeichen
032	040	020	00100000	(leer)
033	041	021	00100001	!
034	042	022	00100010	"
035	043	023	00100011	#
036	044	024	00100100	\$
037	045	025	00100101	%
038	046	026	00100110	&
039	047	027	00100111	'
040	050	028	00101000	(
041	051	029	00101001)
042	052	02A	00101010	*
043	053	02B	00101011	+
044	054	02C	00101100	,
045	055	02D	00101101	-
046	056	02E	00101110	.
047	057	02F	00101111	/
048	060	030	00110000	0
049	061	031	00110001	1
050	062	032	00110010	2
051	063	033	00110011	3
052	064	034	00110100	4
053	065	035	00110101	5
054	066	036	00110110	6
055	067	037	00110111	7
056	070	038	00111000	8
057	071	039	00111001	9
058	072	03A	00111010	:
059	073	03B	00111011	;
060	074	03C	00111100	<
061	075	03D	00111101	=
062	076	03E	00111110	>
063	077	03F	00111111	?

Dezimal	Oktal	Hexa	Binär	Zeichen
064	100	040	01000000	@
065	101	041	01000001	A
066	102	042	01000010	B
067	103	043	01000011	C
068	104	044	01000100	D
069	105	045	01000101	E
070	106	046	01000110	F
071	107	047	01000111	G
072	110	048	01001000	H
073	111	049	01001001	I
074	112	04A	01001010	J
075	113	04B	01001011	K
076	114	04C	01001100	L
077	115	04D	01001101	M
078	116	04E	01001110	N
079	117	04F	01001111	O
080	120	050	01010000	P
081	121	051	01010001	Q
082	122	052	01010010	R
083	123	053	01010011	S
084	124	054	01010100	T
085	125	055	01010101	U
086	126	056	01010110	V
087	127	057	01010111	W
088	130	058	01011000	X
089	131	059	01011001	Y
090	132	05A	01011010	Z
091	133	05B	01011011	[
092	134	05C	01011100	\
093	135	05D	01011101]
094	136	05E	01011110	^
095	137	05F	01011111	_

Dezimal	Oktal	Hexa	Binär	Zeichen
096	140	060	01100000	`
097	141	061	01100001	a
098	142	062	01100010	b
099	143	063	01100011	c
100	144	064	01100100	d
101	145	065	01100101	e
102	146	066	01100110	f
103	147	067	01100111	g
104	150	068	01101000	h
105	151	069	01101001	i
106	152	06A	01101010	j
107	153	06B	01101011	k
108	154	06C	01101100	l
109	155	06D	01101101	m
110	156	06E	01101110	n
111	157	06F	01101111	o
112	160	070	01110000	p
113	161	071	01110001	q
114	162	072	01110010	r
115	163	073	01110011	s
116	164	074	01110100	t
117	165	075	01110101	u
118	166	076	01110110	v
119	167	077	01110111	w
120	170	078	01111000	x
121	171	079	01111001	y
122	172	07A	01111010	z
123	173	07B	01111011	{
124	174	07C	01111100	
125	175	07D	01111101	}
126	176	07E	01111110	~
127	177	07F	01111111	(entf.)



- Unterscheidung zwischen Ziffern und Zeichen im ASCII-Code

- Ziffern als ASCII-Codes

- Angabe des **Zeichens** (Ziffer) in Hochkomma

'0'	→	00110000	(dezimal 48)
'4'	→	00110100	(dezimal 52)
'5'	→	00110101	(dezimal 53)
'8'	→	00111000	(dezimal 56)

- Ziffern als numerischer Wert

- Angabe einer **Ziffer** (ohne Hochkomma)

0	→	00000000	(dezimal 0)
4	→	00000100	(dezimal 4)
5	→	00000101	(dezimal 5)
8	→	00001000	(dezimal 8)



● ASCII-Code

● Speicherung von Texten

- Einzelne Bytes kodieren jeweils immer ein Zeichen
- Werden hintereinander abgespeichert (Zeichenkette – String)

● Ende der Zeichenkette

- Unterschiedliche Verfahren zur Identifizierung (in den Programmiersprachen)
- (Pascal)
Länge der Zeichenkette wird im ersten bzw. in den ersten Bytes vor der eigentlichen Zeichenkette gespeichert
- (C/C++)
Ende der Zeichenkette wird durch ein besonderes, nicht darzustellendes Zeichen gekennzeichnet
 - 0-Byte (Byte, in dem alle Bits 0 sind)



● Uni-Code

- ASCII-Code mit seinen 256 Zeichen ist sehr begrenzt
- Unicode
 - Code, in dem die **Zeichen oder Elemente** praktisch **aller bekannten Schriftkulturen und Zeichensysteme** festgehalten werden können
 - Zeichen werden nach Klassen katalogisiert und erhalten einen Zeichenwert
- Einen eigenen Unicode erhalten auch
 - Steuerzeichen
(Silbentrennung, Leerzeichen oder Tabulatorzeichen)
 - oder Zeichen mathematischer Formeln
- Zusätzlich ist zu jedem Zeichen bzw. Element eine Menge von Eigenschaften definiert
 - z.B. Schreibrichtung



● Uni-Code

- 1991: Gründung des Unicode-Konsortium
 - Ermittelt die aufzunehmenden Zeichen
 - Vergebenen Zeichenwerte haben verbindlichen Charakter
- Zeichenwerte der von Unicode erfassten Zeichen wurden **bis vor kurzem** noch ausschließlich durch
 - eine **zwei Byte lange Zahl** ausgedrückt
 - bis zu 65536 verschiedene Zeichen darstellbar
 - **BMP** (Basic Multilingual Plane) = 2-Byte-System
- Unicode-Version 3.0 (1999)
 - bereits 49194 Zeichen enthalten



● Uni-Code

- Unicode-Version 3.1 (2001)
 - 94140 Zeichen enthalten
- 4-Byte-System wird verwendet
 - Codes von Unicode-Zeichen werden hexadezimal mit vorangestelltem **U+** dargestellt
- Neue Unicode-Version
 - Neuauflage des Buchs „The Unicode Standard“
 - Darstellung aller Zeichenklassen, Zeichen, Zeichenwerte, usw.
 - www.unicode.org



● BCD-Code

- Weitere Art der binären Kodierung von Zahlen bzw. Ziffern sind **BCD-Werte** (Binary Coded Decimals)
- Für jede Dezimalziffer werden **vier** oder manchmal auch acht **Bits** verwendet
- Jeweiligen Ziffern werden nacheinander durch ihren Dualwert angegeben

Dezimalzahl	Dualzahl	Duale BCD Darstellung
294	100100110	0010.1001.0100 2 9 4
16289	11111110100001	0001.0110.0010.1000.1001 1 6 2 8 9



● BCD-Code

- Bitmuster 1010, 1011, ..., 1111 werden im BCD-Code nicht verwendet, da nur 10 Ziffern existieren
- Oft anderweitige Nutzung
 - z.B. 1010 für das Vorzeichen +
 - und 1011 für das Vorzeichen –
- Ineffektive (Speicherplatz verschwendende) Art der Speicherung von Dezimalzahlen
- Spezielle Anwendungsbereiche
 - Ansteuerung von LCD-Anzeigen
 - Speicherung von Dezimalzahlen (Telefonnummern, o.ä.)
 - exakte Darstellung von Brüchen möglich (z.B. 0.1_{10})



● Gray-Codes

- sind Ziffern-Codes,
- die nach folgendem Prinzip erzeugt werden:
 - Benachbarte Zahlen werden so kodiert,
 - dass sie sich in möglichst wenigen Bits unterscheiden (Idealfall: nur 1 Bit)

● Folge

- 1-Bit-Fehler führen zwar zu fehlerhaften Code-Wörtern
- Aber:
 - Bei technischer Interpretation werden keine schwerwiegenden Fehler verursacht
 - (da man eine benachbarte Zahl erhält)



● Gray-Code

- Code, der zur Kodierung von Binärzahlen verwendet wird
- Zwei aufeinanderfolgende Codewörter **unterscheiden** sich immer nur um **ein Bit**

Dezimal	Gray (Binär)	Dezimal	Gray (Binär)	Dezimal	Gray (Binär)
1	0001	6	0101	11	1110
2	0011	7	0100	12	1010
3	0010	8	1100	13	1011
4	0110	9	1101	14	1001
5	0111	10	1111	15	1000



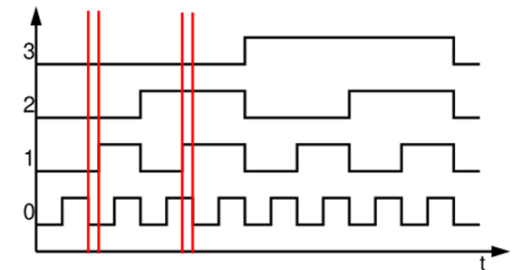
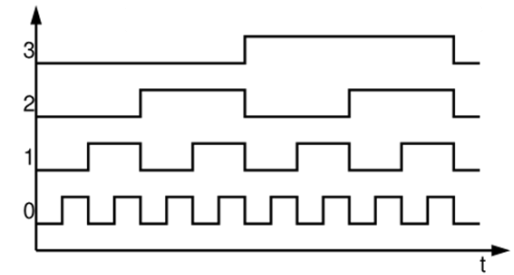
● Gray-Code

● Anwendungsbereich

- Binäre Ausgabe von Werten von A/D-Wandlern (A/D = Analog/Digital) zur Vermeidung unsinniger Zwischenwerte beim Auslesen
- → Verwendung zur Übertragung digitaler Signale über analoge Kanäle

● Sollen Werte in Gray-Zahlen arithmetisch weiterverarbeitet werden,

- müssen diese zuerst in Dualzahlen umgewandelt werden



- andere Zielsetzung:

- nutze nicht alle möglichen Codewörter
- nicht genutzte Codes: Fehlerwörter

- Beispiel:

- 10 Codewörter, 6 Fehlerwörter
- 1-Bit-Fehler
 - erzeugt mit hoher Wahrscheinlichkeit Codewort eines benachbarten Werts oder ein Fehlerwort
 - ergibt nur mit geringer Wahrscheinlichkeit Codewort eines wesentlich verschiedenen Werts
- Fehlerbehandlung (bei Entstehung eines Fehlerworts)
 - Günstigste Strategie:
Korrektur auf das nächstliegende Nutzwort

	00	01	11	10
00	0 0000	1 0001	2 0011	3 0010
01				4 0110
11	8 1100	7 1101	6 1111	5 1110
10	9 1000			

