Zahlensysteme und ihre Anwendung

- ASCII, ANSI und Unicode belegen übereinstimmend die Positionen 48 bis 57 mit den Ziffern '0' ... '9'.
- 48 … 57 sind hierbei die Kennziffern von Schriftzeichen.
- Der Wechsel von arithmetischen Schriftzeichen zu mathematischen Größen erfordert die Auseinander-setzung mit Zahlensystemen.

Zahlensysteme und ihre Anwendung

- Nicht systematische Zahlendarstellungen, z.B.:
  - Strichliste: 𝍸 𝍸 𝍸 𝍸 |||
  - römische Zahlen:

    I=1; V=5; X=10; L=50; C=100; D=500; M=1000
  - griechische Zahlen:

    α‘=1; β‘=2; γ‘=3; δ‘=4; ε‘=5; ς‘=6; ζ‘=7; η‘=8; θ‘=9; ι‘=10;
    κ‘=20; ... π‘=80; ϟ‘=90; ρ‘=100; … ,α=1.000; …; ,θ=9.000

- Systematische Zahlendarstellungen / Stellenwertsysteme:

  Allgemein (s.o.): B unterscheidbare Zeichen erlauben bei einer Wortlänge (=Zeichenanzahl) L die Codierung von $N = B^L$ Zuständen (hier: Größen, Zahlen)

  - Jede Zahl Z läßt sich als Sequenz von Zeichen a_i darstellen
  - Die Anzahl der unterscheidbaren Zeichen ist B (Basis)
  - $Z = \sum_{L} a_i \cdot B^i$

- Dezimalsystem:
  - B=10 unterscheidbare Zeichen: 0,1,2,3,4,5,6,7,8,9
  - $2015_{10} = 2 \cdot 10^3 + 0 \cdot 10^2 + 1 \cdot 10^1 + 5 \cdot 10^0$

- Dualsystem:

  B=2 unterscheidbare Zeichen: 0, 1

  $$\begin{aligned} 2015_{10} = \ & 1 * 2^{10} + 1 * 2^9 + 1 * 2^8 \\ + \ & 1 * 2^7 + 1 * 2^6 + 0 * 2^5 + 1 * 2^4 \\ + \ & 1 * 2^3 + 1 * 2^2 + 0 * 2^1 + 1 * 2^0 \\ = \ & 111\ 1101\ 1111_2 \end{aligned}$$

  Anmerkungen:
  - **‚Dual‘**: Dyadisch, zur Basis 2 (vgl. dezimal, dekadisch)
  - **‚Binär‘**: Zweiwertig (z.B.: Morse ohne Leerzeichen)

    Das Dualsystem ist das einzige breit verwendete Binärsystem, deshalb werden die Begriffe oft als Synonyme verwendet.

Zahlensysteme und ihre Anwendung

- Hexadezimalsystem (Sedezimalsystem)
  - Zeichen: 0,1,2,3,4,5,6,7,8,9,A,B,C,D,E,F
  - $2015_{10} = 7*16^2 + 13*16^1 + 15*16^0$
    $= 7DF_{16} = 0x07DF = H‘07DF$
  - 4 Zeichen einer Dualzahl lassen sich durch eine Hexadezimalziffer darstellen (4 Dualziffern nennt man auch ein **NIBBLE**)

- Oktalsystem
  - Zeichen: 0,1,2,3,4,5,6,7
  - $2015_{10} = 3*8^3 + 7*8^2 + 3*8^1 + 7*8^0 = 3737_8$
  - 3 Zeichen einer Dualzahl lassen sich durch eine Oktalziffer darstellen

Übung zur Darstellung ganzer Zahlen:

Kreuzen Sie in der Tabelle alle Zahlensysteme an, die für die entsprechenden Zahlen zutreffen können.

Zahlenwert	Dual	Oktal	Dezimal	Hexadezimal
1992577			x	x
15034820			x	x
102436		x	x	x
120A0110				x
12305		x	x	x
1001101	x	x	x	x

Wechsel zwischen Zahlensystemen

Beispiel: Ergänzung einer Umwandlungstabelle.

Dezimal	Dual	Oktal
?	**100010**	?
68	?	?
?	?	**153**

Tabelle zeilenweise betrachtet:

- Dual ⇨ Dezimal: Berechnung d. Potenzen: $100010_2=2^1+2^5=34_{10}$
- Dual ⇨ Oktal: je 3 Stellen zusammenfassen: $100\ 010_2 = 42_8$

Beispiel (Forts.)

- Dezimal ⇨ Dual: fortlaufende Divisionen:

  68:2=34 R **0**
  34:2=17 R **0**
  17:2= 8 R **1**
  8:2= 4 R **0**
  4:2= 2 R **0**
  2:2= 1 R **0**
  1:2= 0 R **1**, d.h.: $68_{10} = 1000100_2$

- Dezimal ⇨ Oktal: fortlaufende Divisionen:

  68:8= 8 R **4**
  8:8= 1 R **0**
  1:8= 0 R **1** , d.h.: $68_{10} = 104_8$

- Dual ⇨ Oktal: je 3 Stellen zu einer zusammenfassen: $1\ 0\ 4_8$

- Oktal ⇨ Dual: zifferweise übertragen: $153_8 = 1\ 101\ 011_2$
- Oktal ⇨ Dezimal: Potenzen: $153_8 = 3 + 5*8 + 1*64 = 107_{10}$

Beispiel (Forts.)

Ergebnis:

Dezimal	Dual	Oktal
34	100010	42
68	1000100	104
107	1101011	153

(Hexadezimalsystem ganz entsprechend)

Technisch ebenfalls relevantes Zahlencodierungsverfahren: (*)

Die BCD (Binary Coded Decimal) - Darstellung von Zahlen als Mischform aus Dezimal- und Binär-Darstellung:

- Jede Ziffer der Dezimalzahl wird binär dargestellt.
- Die Reihenfolge der Ziffern bleibt erhalten.
- Die Darstellung jeder Ziffer erfolgt mit 4 Bit.

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

Pseudotetraden (10–15)

Beispiele:

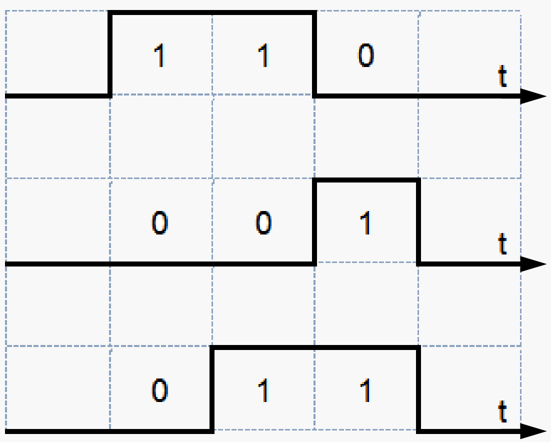
- 7 0111
- 53 0101 0011
- 1234 0001 0010 0011 0100
- 2015 0010 0000 0001 0101

(*) z.B. bei Digitalanzeigen (LED-Ansteuerung etc.)

Zahlensysteme und ihre Anwendung

Anforderungen bei Code-Übertragung über Entfernungen

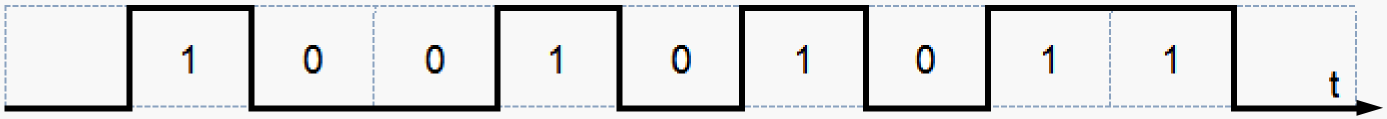
Übertragung z.B. von 100; 101; 011 im Parallelbetrieb:



Synchronisation der Leitungen:
Bei Parallelübertragung durch
- Sync-Signal (vgl. Taktgenerator)

Bei Serienübertragung durch
- alphabetfremdes Zeichen (Morse: Leerzeichen) – oder
- Einsatz gleich langer Wörter (z.B. ASCII)

Korrektheit der Übertragung im kostengünstigeren, aber sensibleren Serienbetrieb abhängig von der Synchronisation:



Bedingung von Robert Mario Fano (Ital./US-Prof., geb.1917):

> In Codes unterschiedlicher Länge darf kein Codewort dem Anfang eines längeren Codewortes gleich sein.

Die Fano-Bedingung ermöglicht fehlerrobuste Code-Übertragung (ohne alphabetfremde Zeichentrennung).

Zur eindeutigen Codierung von Signalen ist ihre Einhaltung

- hinreichend
  - denn sie führt immer zu eindeutigem Code
- nicht notwendig
  - denn auch ohne sie kann Codierung eindeutig sein (vgl. Morse)

Problem der Fehlerminimierung bei Übertragung ganzer Zahlen konstanter Wortlänge in üblichen (Dezimal-, Dual-, BCD-) Stellenwertsystemen:

Darstellungen v. Basispotenzen (B^i) unterscheiden sich von ihren unmittelbaren Vorgängern ($B^i - 1$) in allen Stellen, während sich die übrigen Zahlenfolgen in nur einer Ziffer unterscheiden – z.B.:

Dezimal:	Dual:
(:)	(:)
0998	0110
0999	0111
1000	1000
1001	1001
(:)	(:)

Übertragungs-/Ablesefehler schwer erkennbar u. behandelbar.

Zahlensysteme und ihre Anwendung

Gray-Code (F.Gray, 1887-1969, AT&T-Forscher):

Binär/denär “reflektierender” Code bildet Zahlen durch abwechselndes Vorwärts- / Rückwärts-Zählen zwischen Basispotenzen. (Auch für zyklische Skalen geeignet.)

Bildungsgesetz:

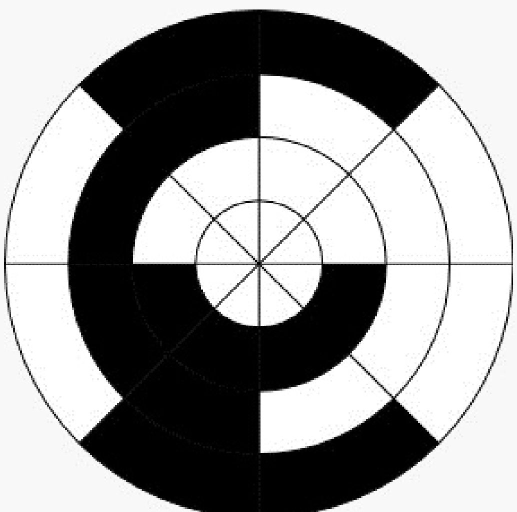


Bild: Wikipedia

Dual:	Gray:
000	000
001	001
010	011
011	010
100	110
101	111
110	101
111	100
(:)	(:)

Gray-Dezimal:

(:)
08
09
19
18
(:)
11
10
20
21
(:)
29
39
(:)

Schwäche: Decodierung erst nach Übertragung höchstwertiger Dualstelle (Zwischenspeicherung eines Wortes) – Zeitnachteil im Serienbetrieb!

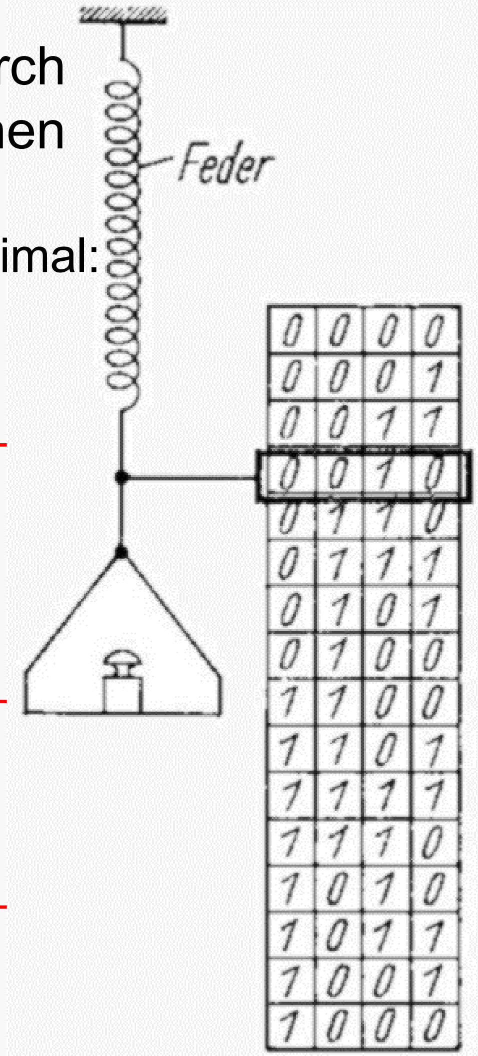


Bild: K.Steinbuch „Nachrichtentechnik“, Springer 1973

Zusammenstellung wichtiger Begriffe der Codierung:

- **Gewicht** eines Codewortes nennt man die Anzahl der Binär-Einsen in ihm.
- Codes mit *m* binären Stellen, bei denen alle 2^m Codewörter zulässig sind, heißen **vollständige Codes**.
- Codes mit Unterscheidung zwischen solchen Bitstellen, die informationstragend sind und solchen, die zur Fehlerprüfung dienen, heißen **systematische Codes** (*).
- Die Anzahl der unterschiedlichen Binärstellen zweier beliebiger Codewörter bezeichnet man als **Hamming-Distanz** der zwei Codewörter (*).
- **Gleichgewichtige** Codes haben stets die gleiche Anzahl von Binär-Einsen in jedem Codewort.
- Die Prüfung eines Codewortes auf zulässiges (ungerades / gerades) Gewicht heißt **parity check** (auch: Paritätskontrolle).

(*) Richard Wesley Hamming, US-Mathematiker, 1915–1998

- BCD läßt je Dezimalziffer 6 von 16 der möglichen Codewörter (37,5%) ungenutzt; Empfang einer Pseudotetrade kann eine erneute Übertragung veranlassen.
- Bessere Prüfbarkeit bei Verwendung „gleichgewichtiger“ Codes – z.B. (2-aus-5)-Codes: Alle 5stelligen Codierungen der Dezimalziffern haben „Gewicht 2“ (2 Binär-Einsen): Von 2^5=32 möglichen Codewörtern sind genau $\binom{5}{2}$=10 zulässig (31,25%).

Dezimal:	Rechencode:	Walking-Code:
0	0 0 0 1 1	0 0 0 1 1
1	1 1 0 0 0	0 0 1 0 1
2	1 0 1 0 0	0 0 1 1 0
3	0 1 1 0 0	0 1 0 1 0
4	1 0 0 1 0	0 1 1 0 0
5	0 1 0 1 0	1 0 1 0 0
6	0 0 1 1 0	1 1 0 0 0
7	1 0 0 0 1	0 1 0 0 1
8	0 1 0 0 1	1 0 0 0 1
9	0 0 1 0 1	1 0 0 1 0

Redundanz [bit]:

$r = l - h = 5 - ld10$
$= 5 - lg10/lg2$
$\approx 1{,}68$ bit

Vorteilhaft vor allem bei sog. **Einseiten-störungen** (wenn Fehlerwahrschein-lichkeiten 0 ⇨ 1 und 1 ⇨ 0 ungleich).

- Allgemein: „w-aus-m“- bzw. „m-über-w“-Codes, wenn genau w von m Zeichen (Elemente) jedes Codeworts Einsen sind – z.B. „3-aus-7“-Code (“Van Duuren-Code“)
- Verwendung gleichgewichtiger Codes zur Ein-Fehler- („1-F“-) Prüfbarkeit ist hinreichend, aber nicht notwendig:

  Auch andere Abweichungen vom vollständigen Code sind nützlich – z.B.: Beschränkung auf gerades Gewicht:

  Erhöhung der Binärstellen um eine Prüfstelle auf m+1; Anhängen einer 1 bei ungeradem Gewicht.

  Hinzunahme von 1 Bit macht 50% d. Codewörter nutzbar.

  (Redundanz: r = l – h = 1 bit, unabhängig von der Anzahl der Bitstellen)
- Parity check in systematischen Codes: Jedes Zeichen hat zu allen anderen eine Mindest-Hamming-Distanz d=2

Zahlensysteme und ihre Anwendung

- Interessanter als 1-F-Prüfbarkeit: Korrigierbarkeit

  Erläuterung anhand der Übertragung der Zahl 541.376, dezimalstellenweise im (2-aus-5)-Rechencode (oder einem anderen 1-F-prüfbaren Code).

  Anfügen eines Prüfwortes (PW) zur Gewährleistung einer geraden Quersumme über alle Bitebenen. Nach Überprüfung der Quersummen (parity check) kann ein Fehler erkannt und korrigiert werden:

Nachricht	Sendesignal	Empfang
5	0 1 0 1 0	0 1 0 1 0
4	1 0 0 1 0	1 0 1 1 0
1	1 1 0 0 0	1 1 0 0 0
3	0 1 1 0 0	0 1 1 0 0
7	1 0 0 0 1	1 0 0 0 1
6	0 0 1 1 0	0 0 1 1 0
PW	1 1 0 1 1	1 1 0 1 1

Zeile und Spalte mit jeweils unzulässiger (≠2, ungerader) Quersumme: Fehlererkennung und -korrektur möglich!

Darstellung ganzer (positiver u. negativer) Zahlen: *

Darstellung des Vorzeichens im ersten Bit, z.B.

000	= 0	100	= -0
001	= 1	101	= -1
010	= 2	110	= -2
011	= 3	111	= -3

Nachteile:

- Redundanz in der Darstellung der 0
- Probleme beim formalen Addieren:

```
  110    - 2
+ 001    +1
-------- ----
  111    - 3
```

Zweierkomplementdarstellung (n Bit):

000	= 0	100	= -4
001	= 1	101	= -3
010	= 2	110	= -2
011	= 3	111	= -1

- Darstellung des Vorzeichens im ersten Bit
- Abdeckung der 2^n Zahlen $-2^{n-1} \dots +(2^{n-1}-1)$: keine Redundanz!
- Keine Probleme beim formalen Addieren:

```
   110     - 2
 + 001     +1
---------  ----
   111     - 1
```

- Betragshohe positive und negative Zahlen hintereinander:

  $(2^{n-1}-1)+1$ ergibt evtl. -2^{n-1}!

Grundsätzliches zum Komplement:

- Das (B-1)-Komplement einer Zahl im Zahlensystem mit der Basis B wird gebildet, indem jede Ziffer z_i dieser Zahl durch die Ziffer $B-1-z_i$ (*) ersetzt wird.

  (*) Das ist die Differenz zur höchsten Ziffer des Zahlensystems.

- Das B-Komplement wird gebildet, indem zum (B-1)-Komplement die 1 addiert wird.

  Beispiel:

  - Das 9er-Komplement der Zahl 02_{10} ist 97_{10},
    das 10er-Komplement der Zahl 02_{10} ist 98_{10} .
  - Das Einerkomplement der Zahl 0111_2 ist 1000_2,
    das Zweierkomplement der Zahl 0111_2 ist 1001_2 .

Der Darstellung negativer Zahlen durch das Komplement liegt die Vorstellung eines Zahlenrads zugrunde (vgl. km-Zähler):

(:)		(:)
0003	[3]	0011
$\boxed{0002}$	[2]	$\boxed{0010}$
0001	[1]	0001
0000	[0]	0000
9999	[-1]	1111
$\boxed{9998}$	[-2]	$\boxed{1110}$
9997	[-3]	1101
(:)		(:)

- Das 10er-Komplement von 0002 ist (9997+1=) 9998; gleichzeitig entspricht die Position '9998‘ einer ‘-2‘.
- Das Zweierkomplement von 0010 ist (1101+1=)1110; gleichzeitig entspricht die Position ‘1110‘ einer ‘-2‘.
- Negative Zahl durch bitweise Komplementierung u. Addition von 1 (Bsp: 3 = 0011, Zweierkomplement = 1100 + 1 = 1101 = -3)