

Digitaltechnik

Levin Baumann

12. Mai 2020

1 Begriffe

digital	analog
wertediskret	wertekontinuierlich
⇒ zwischen zwei benachbarten gibt es <u>keine</u> Zwischenwerte	⇒ zwischen zwei beliebigen gibt es unendlich viele Zwischenwerte
⇒ endlich oder auch unendlich (aber abzählbar viele)	⇒ Immer unendlich (und sogar überabzählbar) viele Werte
meist zeitdiskret („getaktet“)	zeitkontinuierlich
⇒ unsere heutigen Rechner arbeiten digital	⇒ unsere Welt „arbeitet“ analog

2 Codierung

Definition

Codierung ist die Darstellung von Informationen mit einem „Alphabet“.

Definition

Alphabet ist eine endliche Menge von Symbolen

Anmerkung: Codierung ist immer mit Digitalisierung verbunden.

2.1 Arten von Codierungen

- Zeichencodierung
- Zahlencodierung (Text-, Bildbearbeitung,...)
- Verschlüsselung (*Achtung:* Unterschied zu anderen Codierungen: Aus der codierten verschlüsselten Info sollen die meisten Menschen nicht auf die Ursprungsinfo schließen können)
- Signalcodierung

2.2 Codierungen

2.2.1 Zeichencodierung

- ASCII: Alphabet besteht aus (ganzen) Zahlen von 0 bis 127
- ISO8859-x: Alphabet besteht aus 0...255
 - x: verschiedene Sprachräume
 - 1: westeuropäisch
 - 5: kyrillisch
 - 7: griechisch
 - 15: westeuropäisch inkl. €
- Unicode: Anspruch, alle (derzeitigen, künftigen, ehemaligen) Schriftsprachen abzudecken, auch Phantasiesprachen wie Klingonisch oder Elbisch
 - UTF-8: 256 Zahlenwerte
 - UTF-16: 65536 Zahlenwerte
 - Zeichen werden als variabel lange Symbolfolgen dargestellt. Gesetztes MSB zeigt an, dass mindestens ein Folgesymbol folgt.

2.2.2 Zahlencodierung

I. Abzählssysteme

Fingerabzählssysteme

Alphabet = {Finger}

Symbolwert (Finger) = 1

- ⊖ stark beschränkter Wertebereich von 0 bis 10
- ⊖ bis auf weiteres nur nicht-negative ganze Zahlen
- ⊕ extrem einfach und verständlich
- ⊕ extrem einfache Addition und Subtraktion
- ⊖ Multiplikation und Division mit erhöhtem Aufwand
- ⊕ Immer verfügbar / „zur Hand“

Strichliste (einfache)

Alphabet = {I}

Wert (I) = 1

- ⊕ unbeschränkter Wertebereich
- ⊕ Addition weiter einfach, Subtraktion braucht man eine Entfernungsmöglichkeit/Entwertungsmöglichkeit
- ⊖ Hilfsmittel (Schreibwerkzeug) notwendig
- ⊖ unübersichtlich darstellbarer Wertebereich bis etwa 10 (10 Symbole notiert)

erweiterte Strichliste („Lattenzaunsystem“)

Alphabet = I, IIII

Wert (I) = 1

Wert (IIII) = 5

Regeln: Symbole müssen nach Wertigkeit sortiert notiert werden. Fünf einfache Striche müssen immer zu einem „Kombisymbol“ zusammengefasst werden.

- ⊕/⊖ Addition etwas erschwert durch Sortieren und Zusammenfassen; Subtraktion zusätzlich erschwert durch Auflösen des Kombisymbols
- ⊕/⊖ etwas komplexer durch zweites Symbol und Sortierregel
- ⊕/⊖ übersichtlich darstellbarer Wert bis etwa 50 erweitert

römisches Zahlensystem

Alphabet = {I, V, X, L, C, D, M}

Wert = {1, 5, 10, 50, 100, 500, 1000}

Zusatzregel: Symbol kleinerer Wertigkeit darf vor einem Symbol größerer Wertigkeit notiert werden. Sein Symbolwert wird dann subtrahiert, statt addiert. Mehr als drei gleiche Symbole sind nicht hintereinander erlaubt

- ⊕/⊖ übersichtlich darstellbarer Wertebereich bis etwa 10.000 (?)
- ⊖ beschränkter Wertebereich bis ; 4000
- ⊖ recht hohe Komplexität, relativ geringe Verständlichkeit
- ⊖ extrem komplizierte Addition und Subtraktion

II. Stellenwertsysteme

Dezimalsystem

Alphabet = $\{0, 1, 2, 3, 4, 5, 6, 7, 8, 9\}$

Symbole werden auch Ziffern genannt

Zahl ist notiert als Folge von Ziffern: z.B. 4711

Ziffernwert = Symbolwert

Ziffernwert (4) = 4

Zahlenwert = Summe (Ziffernwert * Stellenwert)

Stellenwert (rechte Ziffer) = 1

Stellenwert wächst mit jeder Stelle nach links um Faktor 10

$$\text{Wert}(Z_{n-1} \dots Z_0) = \sum_{i=0}^{n-1} |Z_i| \cdot 10^i$$

SWS zur Basis b

Alphabet enthält b Ziffern

Alphabet beginnt bei Ziffer „0“ und endet bei Ziffer „b-1“

$$\text{Wert}(Z_{n-1} \dots Z_0) = \sum_{i=0}^{n-1} |Z_i| \cdot b^i$$

b_{EIN} mit $b > 1$

Anmerkung: $b = 1$ nicht sinnvoll, da im SWS zur Basis nur die 0 dargestellt werden könnte.

⊕/⊖ gewisses „Erstverständnis“ Lernaufwand ist nötig

⊕/⊖ es gibt für jede Grundrechenart ein Verfahren mit etwas erhöhter Komplexität, welches aber nach erstmaligem Lernaufwand doch relativ problemfrei realisierbar ist

⊕/⊖ unbeschränkter Wertebereich

⊕/⊖ Wertebereich bis etwa b^10 ist übersichtlich darstellbar

Umrechnung

- Von Basis b nach Basis 10?
⇒ Wertformel
- Von Basis 10 nach Basis b ?
⇒ Wertformel umgekehrt
⇒ Ganzzahldivision

$$42_{10} = 101010$$

$$43 : 2 = 21 R0 = Z_0$$

$$21 : 2 = 10 R1 = Z_1$$

$$10 : 2 = 5 R0 = Z_2$$

$$5 : 2 = 2 R1 = Z_3$$

$$2 : 2 = 1 R0 = Z_4$$

$$1 : 2 = 0 R1 = Z_5$$

(1)

Hinweis: Führende „0“ können bei Zahlen im SWS beliebig hinzugefügt oder weggelassen werden. allg: Umrechnung von Basis b_1 nach Basis b_2

- meist: von Basis b_1 nach $b_Z = 10$ dann von b_Z nach b_2
- theoretisch: Ganzzahldivision durch Zielbasis b_2 , aber ausgeführt im SWS zur Ausgangsbasis ⇒ Nicht praktikabel

Direkte Umrechnung:

Falls $b_1 = b_2^n$, dann entsprechen n Ziffern zur Basis b_2 einer Ziffer zur Basis b_1 und es ist eine ziffern(block)weise Umrechnung.

Bsp.: $b_1 = 2$ und $b_2 = 16$

\Rightarrow 4 Ziffern zur Basis 2 entsprechen einer Ziffer zur Basis 16.

Gängige Basen im SWS

$b = 10$ „Dezimalsystem“ \Rightarrow Mensch

$b = 2$ „Binärsystem“ „Dualsystem“ \Rightarrow Digitalrechner

$b = 16$ „Hexadezimalsystem“ \Rightarrow Computernahe Darstellung von Zahlen

- weniger Stellen
- einfache Umrechnung

$b = 8$ „Oktalsystem“ \Rightarrow Computernahe Darstellung von Zahlen

$b = 16$ vs. $b = 8$ nur „normale“ Zahlen notwendig

Hexadezimalsystem

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

BSP: $ABEF_{16}$

1010 | 1011 | 1110 | 1111

0001 | 1001 | 0001 | 1100 | 0101₂ = 191C5₁₆
1 | 9 | 1 | C | 5

Falls $b_1^n = b_2^m$, dann entspricht ein Ziffernblock von n Ziffern zur Basis b_1 einem Ziffernblock von m Ziffern zur Basis b_2 .

$$\begin{aligned} 2^{3n} = 2^{4m} &\Rightarrow n = 4 \text{ \& } m = 3 \\ &\Rightarrow 4 \text{ Ziffern zur Basis 8 entsprechen} \\ &\quad 3 \text{ Ziffern zur Basis 16 entsprechen} \\ &\text{Problem: Tabelle hat } 2^{12} \text{ Zeilen} \end{aligned}$$

Einschränkungen aufheben

auch negative Zahlen!

Darstellung negativer Zahlen

3 Signalkodierung

Definition

Darstellung von abstrakten Informationen als Signalfolge.
Signal: physisch messbare Größe.

3.1 mögliche Signalformen

- elektrisch
Spannung, Stromstärke, elektromagnetische Wellen, Ladung
- optisch
Helligkeit, Farbe
- akustisch Lautstärke, Tonhöhe
- Druck hydraulisch, pneumatisch

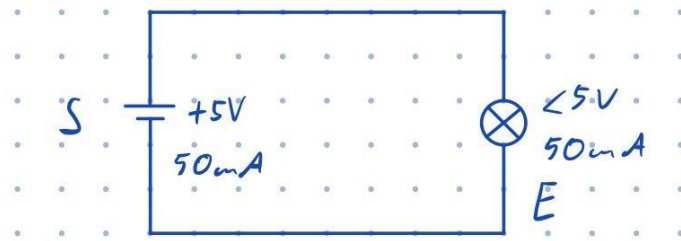


Abbildung 1: Schaltplan mit Spannungsquelle und Glühbirne

in der Computertechnik relevant:

optisch \Rightarrow für netzwerkschnittstellen

elektrisch \Rightarrow insbesondere in der Digitaltechnik

vor allem: Spannung (evtl. auch Stromstärke)

\Rightarrow eher kleinere Spannungspegel in der Digitaltechnik

Strom Vs. Spannung

Spannung: Verringert sich beim E durch Spannungsfall auf der Leitung, wird verändert durch Störungen von außen („Übersprechen“ elektromagnetische Einstrahlungen)

Strom: Kaum davon betroffen; Stromstärke verändert sich im geschlossenen Stromkreis nicht.

Nachteil Strom: hoher Energieaufwand, große Wärmeentwicklung

\Rightarrow deshalb Spannung statt Strom bei den meisten Computerschnittstellen verwandt.

typische Spannungspegel:

$$\left. \begin{array}{l} 0 \hat{=} 0V \\ 1 \hat{=} 5V \end{array} \right\} \text{TTL-Pegel}$$

„Transistor-Transistor-Logic“

\Rightarrow „erfunden“ um 1960 von TI (Texas Instruments)

große vs. kleine Spannungspegel

- \ominus größere sind stromanfällig bei kleineren Spannungen
- \oplus viel weniger Energieaufwand bei kleinen Spannungen, also auch weniger Wärmeentwicklung
- \oplus weniger Störauswirkungen bei kleineren Spannungen
- \oplus schnelleres Spannungswechseln bei kleinen Spannungshüben möglich

3.2 Umsetzung von Bitfolgen in Spannungspegelfolgen

meist getaktet, d.h. festes Zeitraster für die Bitfolge, d.h. jedes Bit braucht eine konstante, gleichlange Zeitdauer

4 Verfahren (1-4) und 4 Eigenschaften(a-d)

1. NRZ (Non-Return-to-zero)

Während der gesamten Schrittdauer wird der Pegel angelegt, welcher dem zu übertragenden Bitwert entspricht

2. RZ (Return-to-Zero) Jeder Schritt wird in zwei Schrittzeithälften eingeteilt. Während der ersten Hälfte wird der Pegel eingenommen, welcher den zu übertragenden Bitwert entspricht und während der zweiten Hälfte immer der „0“-Pegel.

TRG bei RZ: Bei jeder „1“ möglich, dann bei einer „1“ zu Beginn und in der Mitte der Schrittzeit ein Pegelwechsel stattfindet

3. AMI (Alternate Mark Inversion)

Ähnlich NRZ mit single-ended Pegeln, d.h. „0“ wird immer mit 0V (während der gesamten Schrittzeit) übertragen, aber „1“ abwechselnd mit z.B. +5V und -5V (während der gesamten Schrittzeit)

GFS bei AMI: nach jeder 2. „1“: In der Praxis ist der GS-Anteil nach der ungeraden „1“ vernachlässigbar (bei langer Übertragungszeit und vielen übertragenen „1“), also praktisch immer GSF.

TRG bei AMI: bei jeder „1“ nur bei langer Folge von „0“ keine TRG möglich

4. Manchester

Bitwert wird über Spannungswechsel in der Mitte der Schrittzeit dargestellt, z.B. steigende Flanke $\hat{=}$ „1“ und fallende Flanke $\hat{=}$ „0“.

ggf. ist ein weiterer Spannungswechsel zu Beginn der Schrittzeit notwendig, um den nachfolgenden (inhaltsführenden) Spannungswechsel durchführen zu können.

TRG bei Manchester: bei jedem Schritt möglich, da immer Regelwechsel in der Mitte der Schrittzeit.

GFS bei Manchester: immer, da sich die Pegel in erster und zweiter Schrittzeithälfte gegenseitig ausgleichen

SSH bei Manchester: optimal, da „nur“ 2 Pegel

a) TRG (Taktrückgewinnung)

Möglichkeit, nur aus dem übertragenen Datensignal eine Resynchronisierung beim Empfänger auf den Takt des Senders zu machen

b) GSF (Gleichspannungs-/stromfreiheit)

Motivation: Einsparung der Masseleitung. Im zeitlichen Mittel liegen auf der Signalleitung 0V an.

Ziel: Anhand des mittleren Signalpegels soll der Massepegel „errechnet“ werden.

-Vermeidung einer Potentialverschiebung beim E.

-Pseudoargument: keine Energieübertragung beim vom S zum E.

Grundvoraussetzung: symmetrische Pegel statt single-ended.

z.B. $1V \hat{=} +5V$ und $0 \hat{=} -5V$

dann: GSF bei NRZ: falls $\# „0“ = \# „1“$ (bzw. „1“ und „0“ im Datenstrom gleichverteilt)

Bei den meisten Anwendungs-, Zeichen- und Zahlencodierungen kann keine Gleichverteilung angenommen werden.
Ausnahme: Verschlüsselung und Kompression.

c) SSH (Störsicherheit)

(Un-)Anfälligkeit eines Verfahrens ggü. Störungen, welche durch Spannungsschwankungen auf der Leitung verursacht werden.

\Rightarrow direkt abhängig von der Anzahl der verwendeten Pegel, welche auf einen vorgegebenen Potentialbereich verteilt werden müssen (und so natürlich auch beim Empfänger voneinander unterschieden werden müssen)

SSH bei AMI: schlecht, da 3 Pegel

SSH bei NRZ und RZ: gut, da „nur“ 2 Pegel (und weniger Pegel geht nicht ☺)

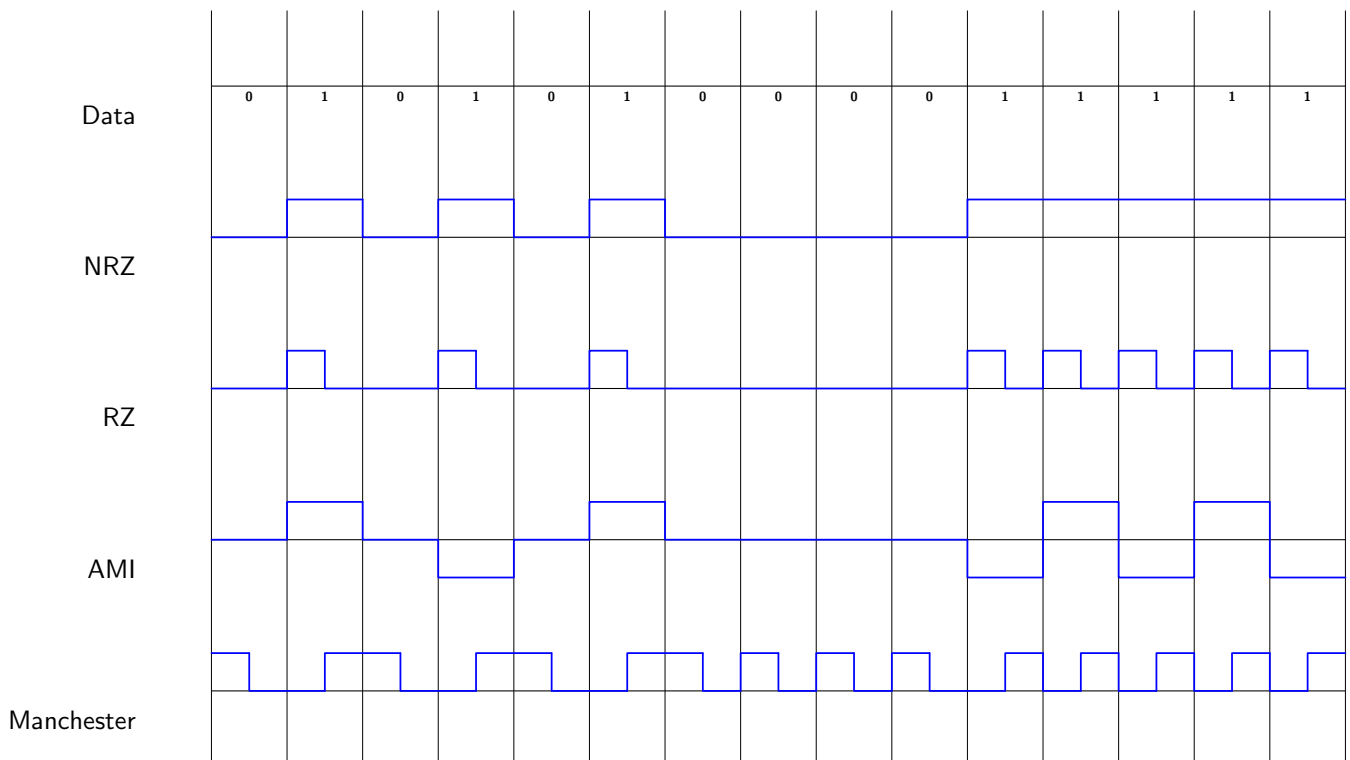


Abbildung 2: Pegelgraphen der einzelnen Verfahren

Bandbreitenbedarf, bandbreitenbegrenzte Übertragungskanäle

Nyquist-Theorem: Über einen Übertragungskanal mit beschränkter Bandbreite kann maximal mit der Schrittrate übertragen werden, welche der doppelten Bandbreite entspricht.