

Digitaltechnik & Rechnersysteme

Schaltnetze

Martin Kumm



WiSe 2025/2026

Was bisher geschah...

- Zahlencodierung im Stellenwertsystem
- Dezimal ($R = 10$), Binär ($R = 2$), Hexadezimal ($R = 16$)
- Vorzeichenbehaftete Zahlen
 - Vorzeichen-Betrag: Vorzeichenbit + Betrag separat codiert
 - Zweierkomplement: negative Zahlen bitweise invertiert + 1

Wenn man Wertebereiche missachtet... I

Die Explosion der Ariane 5 (V88) 1996

- Die unbemannte Ariane 5 Rakete explodierte 40 Sekunden nach lift-off (Kosten: \$500 Millionen)
- Der Fehler entstand aufgrund einer Konvertierung (cast) einer 64 Bit Gleitkommazahl in eine 16 Bit (vorzeichenbehaftete) ganze Zahl.
- Zahl repräsentierte die horizontale Ausrichtung
- Abzüglich Vorzeichenbit hat diese den max. Wert $2^{15} - 1 = 32.767$.
- Leider war die horizontale Ausrichtung größer...

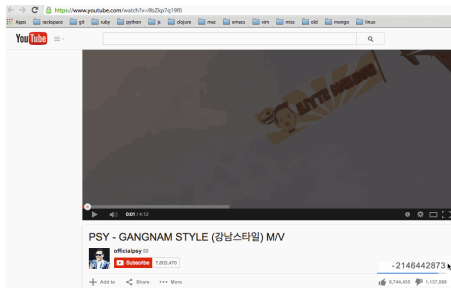


Quelle: <http://www.math.umn.edu/~arnold/disasters/>

Wenn man Wertebereiche missachtet... II

Gangnam Style Video

- Als die Anzahl der Youtube views des »Gangnam Style« Video die 2.147.483.647 überstieg wurde diese negativ
- Warum?
 $2.147.483.647 = 2^{31} - 1$ ist die größte positive Zahl die sich mit 32 Bit vorzeichenbehaftet darstellen lässt.
- Das nächste Codewort repräsentiert die Zahl $-2^{31} = -2.147.483.648$



Quelle: <https://www.wired.com/2014/12/gangnam-style-youtube-math>

Festkommazahlen

Bei Festkommazahlen wird die ganze Zahl um eine feste Anzahl Nachkommastellen erweitert:

$$X = (x_{n-1}x_{n-2} \dots x_1x_0, x_{-1}x_{-2} \dots x_{-f})$$

Der Wert der Zahl ist dann gegeben durch

$$X = \sum_{i=-f}^{n-1} x_i 2^i$$

Beispiel: $101,1101_2 = 2^2 + 2^0 + 2^{-1} + 2^{-2} + 2^{-4} = 5,8125_{10}$

Vorlesungsaufgabe



Bestimmen Sie den Dezimalwert der Festkommazahl $110,11_2$

Konvertierung in Festkommazahlen

Konvertierung einer (Dezimal-)Kommazahl in eine Festkommazahl:

- Multipliziere Zahl mit 2^f
- Runde das Ergebnis zur nächsten ganzen Zahl
- Konvertiere ganze Zahl ins Binärsystem
- Verschiebe die Zahl um f Bit nach rechts

Beispiel: Darstellung von $\pi = 3.14159265358979\dots$ mit 8 Nachkommastellen:

- $\pi \cdot 2^8 = 804.2477\dots$
- $\text{round}(804, 2477) = 804_{10} = 1100100100_2$
- $\pi_{10} \approx 11,00100100_2 = 3,140625_{10}$

Gleitkommazahlen Idee

Der Dynamikbereich von Festkommazahlen ist stark eingeschränkt.

Sehr große oder sehr kleine Zahlen benötigen viele Bits, die Genauigkeit ist dabei unnötig hoch.

Lösung: Anpassung der Position des Kommas durch Gleitkomma- bzw. Fließkommazahlen

Das tun wir bereits im Umgang mit großen oder kleinen Zahlen.

AI | Angewandte Informatik

Beispiel: Das Elektronenvolt (eV) hat eine Energie von $1,602176634 \cdot 10^{-19}$ Joule

➡ Statt 0,00000000000000000001602176634 Joule schreiben wir kompakt $1,602176634 \cdot 10^{-19}$ Joule

Genauso können wir statt der Festkommandarstellung

[illegible]

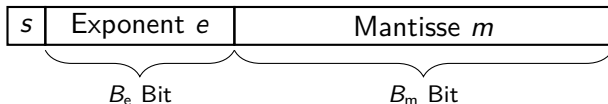
die Gleitkommadarstellung $\underbrace{1,011110100100110_2}_{\text{Mantisse}} \cdot \underbrace{2^{-63}}_{\text{Exponent}}$ verwenden.

Zur Kodierung dieser Zahl muss...

- dank Normalisierung der sog. Mantisse zu 1,011... die führende 1 nicht gespeichert werden
- der Exponent (-63) extra gespeichert werden

Gleitkommazahl nach IEEE 754

Eine Gleitkommazahl nach IEEE 754 Standard ist folgendermaßen definiert:



- Das Sign-Bit s bestimmt das Vorzeichen ($0 \hat{=}$ +, $1 \hat{=}$ -)
- Der Exponent e definiert die Position des Kommas
- Mantisse m definiert normalisierte Zahl (ohne führende Eins)

Die Zahl berechnet sich hieraus zu:

$$X = (-1)^s \left(1 + \sum_{i=0}^{B_m-1} m_i 2^{-i-1} \right) 2^{e-\text{bias}} \quad \text{mit} \quad \text{bias} = 2^{B_e-1} - 1$$

Gleitkomma-Konvertierung

Beispiel: Reele Zahl → Gleitkommadarstellung

$$X = (-1)^s \, 1, m \, 2^{e-\text{bias}} \quad \text{mit} \quad \text{bias} = 2^{B_e-1} - 1$$

Beispiel: Darstellung der Zahl $x = 9,25_{10}$ mit einer Exponentenwortbreite von $B_e = 6$ Bit und einer Mantissenwortbreite von $B_m = 5$ Bit – (1,6,5)-Format:

- 1 Ermittlung der Festkommadarstellung: $x = 9,25_{10} = 1001,01_2$
- 2 Normalisierung: $1001,01_2 = 1,\underbrace{00101}_m_2 \cdot 2^3$
- 3 Ermittlung des Exponenten: $e - \text{bias} = 3 \rightarrow$
 $e = 3 + \text{bias} = 3 + 2^5 - 1 = 34 = 100010_2$

$$\Rightarrow \text{Codewort: } \underbrace{0}_s \underbrace{100010}_e \underbrace{00101}_m$$

Gleitkomma-Konvertierung

Beispiel: Fließkommazahl → reelle Darstellung:

$$A = (-1)^s 1, m 2^{e-\text{bias}} \quad \text{mit} \quad \text{bias} = 2^{B_e-1} - 1$$

- 1 Bestimmung der Komponenten: $\underbrace{0}_s \underbrace{100010}_e \underbrace{00101}_m$
- 2 Erweiterung der Mantisse: $1, m = 1,00101_2$
- 3 Ermittlung Exponent $e = 100010_2 = 34_{10}$
- 4 Verschiebung und Konvertierung:
 $x = +1,00101_2 \cdot 2^{34-31} = +1001,01_2 = 9,25_{10}$

IEEE Standard



Der IEEE 754r Standard definiert folgende Gleitkommaformate:

	Half	Single	Double
Wortbreite	16 Bit	32 Bit	64 Bit
Mantisse (B_m)	10 Bit	23 Bit	52 Bit
Exponent (B_e)	5 Bit	8 Bit	11 Bit
Bias	15	127	1023
Wertebereich	$\pm 2^{16} \approx 6 \cdot 10^4$	$\pm 2^{128} \approx 10^{38}$	$\pm 2^{1024} \approx 10^{308}$

Einheitliches Format wichtig zum Austausch von Daten!

Zusätzlich spezielle Werte für

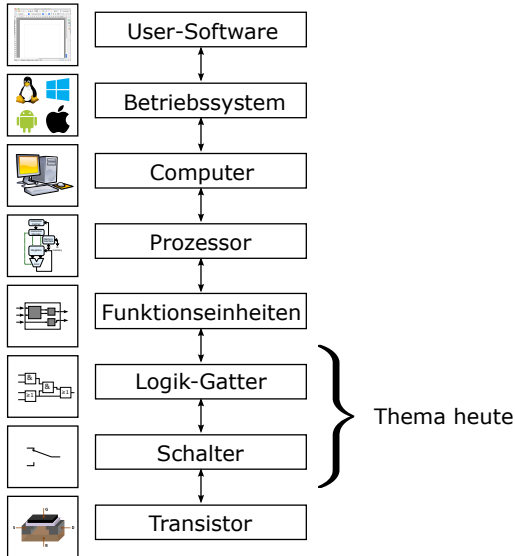
- 0 (lässt sich durch Normalisierung sonst nicht darstellen!)
- $\pm\infty$ (für Überläufe)
- NaN: Not-a-Number (keine Zahl), entsteht z.B. bei $\frac{0}{0}$, $0 \times \infty$

Inhalte



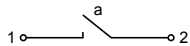
- 2 Festkommazahlen
- 3 Gleitkommazahlen
- 4 Informationsverarbeitung mit Schaltern
- 5 Digitale Logik-Gatter
- 6 Kombinatorische Schaltungen

Die Macht der Abstraktion



Informationsverarbeitung mit Schaltern

Symbolische Darstellung eines Schalters:



Eingang a besagt ob Schalter offen oder geschlossen:

Schalter a geschlossen	Verbindung $1 \leftrightarrow 2$?
nein	nein
ja	ja

🤔 Aber wie lassen sich damit Informationen verarbeiten?

Informationsverarbeitung mit Schaltern



Serienschaltung zweier Schalter:



Schalter <i>a</i> geschlossen	Schalter <i>b</i> geschlossen	Verbindung 1↔2?
nein	nein	nein
nein	ja	nein
ja	nein	nein
ja	ja	ja

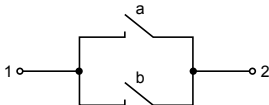
Logische Abstraktion über binäre Variablen:

- Schalter offen/geschlossen: 0/1
- Verbindung vorhanden nein/ja: 0/1

<i>a</i>	<i>b</i>	1↔2
0	0	0
0	1	0
1	0	0
1	1	1

Informationsverarbeitung mit Schaltern

Parallelschaltung zweier Schalter:



Vorlesungsaufgabe: Wie lautet die Wahrheitstabelle?

<i>a</i>	<i>b</i>	$1 \leftrightarrow 2$
0	0	
0	1	
1	0	
1	1	

Komplexe Informationsverarbeitung

🤔 Aber wie lassen sich damit komplexe Informationen verarbeiten?

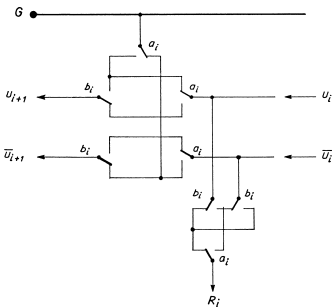
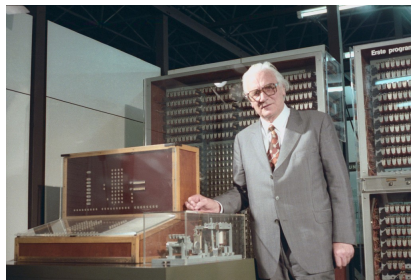


Bild 16. Einschrittige Addierschaltung für eine Binärstelle. Sie enthält nur Kontakte a , und b , der beiden Summanden. Es gibt zwei Übertragungsketten u und \bar{u} . („Es wird übertragen“ und „es wird nicht übertragen“)

Schalter-Realisierung eines 1-Bit
 Addiers im Z1 / Z3 Computer
 (Quelle: Konrad Zuse, »Der Computer, mein
 Lebenswerk«)

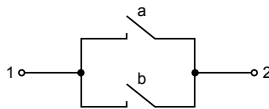


Konrad Zuse mit Z3
 (Foto: Deutsches Museum)

Schaltalgebra



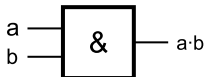
Reihenschaltung



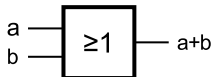
Parallelschaltung

Abstraktion:

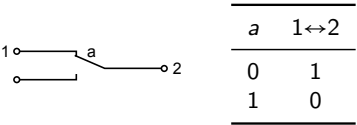
UND-Verknüpfung



ODER-Verknüpfung

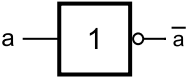


NICHT-Verknüpfung/Negation



Abstraktion:

NICHT-Verknüpfung/Negation



NICHT-Verknüpfung / Negation

Funktionsweise: Das Ergebnis der Negation ist 1 wenn der Eingang 0 ist und 0 wenn der Eingang 1 ist.

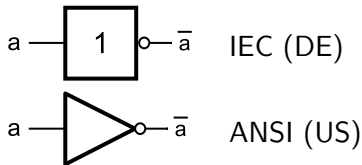
Die Schaltungsrealisierung wird als NICHT-Gatter, NOT-Gate oder als **Inverter** bezeichnet.

Boolescher Ausdruck: $y = \bar{a}$, (alt. $y = \neg a$)

Wahrheitstabelle:

a	\bar{a}
0	1
1	0

Schaltsymbole:



UND-Verknüpfung / Konjunktion

Funktionsweise: Das Ergebnis der UND-Verknüpfung ist genau dann 1 wenn **alle** Eingänge 1 sind

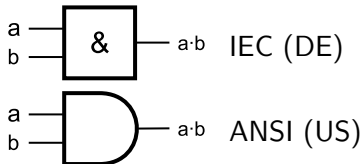
Die Schaltungsrealisierung wird als UND-Gatter oder AND-Gate bezeichnet.

Boolescher Ausdruck: $y = a \cdot b$, kurz $y = ab$ (alt. $y = a \wedge b$)

Wahrheitstabelle:

a	b	$a \cdot b$
0	0	0
0	1	0
1	0	0
1	1	1

Schaltsymbole:



ODER-Verknüpfung / Disjunktion

Funktionsweise: Das Ergebnis der ODER-Verknüpfung ist genau dann 1 wenn **mindestens ein** Eingang 1 ist

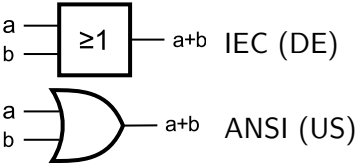
Die Schaltungsrealisierung wird als ODER-Gatter oder OR-Gate bezeichnet.

Boolescher Ausdruck: $y = a + b$ (alt. $y = a \vee b$)

Wahrheitstabelle:

a	b	a + b
0	0	0
0	1	1
1	0	1
1	1	1

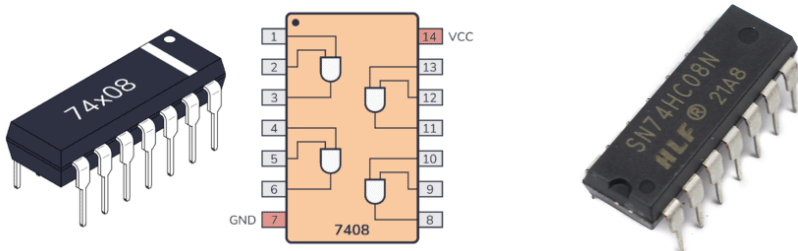
Schaltsymbole:



Technische Realisierung von Gatter

Technisch werden die Schalter durch Transistoren in integrierten Schaltkreisen (*Integrated Circuit*, IC) realisiert

Entweder integriert in komplexen Chips (heute üblich) oder als eigene Bausteine (heute eher unüblich)



<https://www.build-electronic-circuits.com/7400-series-integrated-circuits/74hc08-74ls08/>

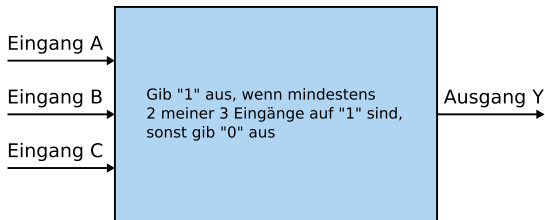
Ein digitales Verarbeitungselement



Ein **Schaltnetz** oder auch **kombinatorische Schaltung** ist eine Zusammenschaltung von Logik-Gattern mit

- einem oder mehreren digitalen **Eingängen**
- einem oder mehreren digitalen **Ausgängen**
- einer **funktionalen Spezifikation**, die für jede mögliche Kombination von Eingaben die Ausgabe angibt

Beispiel:

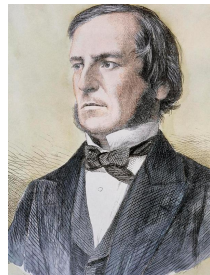


Wie konstruiert man so ein Schaltnetz?

Boolesche Algebra



- Der Entwurf und die Analyse von Systemen, die binäre Signale verarbeiten erfolgt mit Hilfe der **Booleschen Algebra**
- Die Boolesche Algebra ist ein Teilgebiet der Algebra und behandelt mengentheoretische und logische Verknüpfungen zwischen den Elementen einer Menge
- Uns interessiert hier nur die **zweielementige** Boolesche Algebra auf der Menge $\{0, 1\}$
- Die Boolesche Algebra findet u.A. Anwendung in der **Aussagenlogik** und der **Schaltalgebra**



George Boole
(um 1860)

Schaltalgebra



Eine Schaltalgebra ist eine Boolesche Algebra mit

- einer binären Trägermenge $B = \{0, 1\}$
- der Konjunktion/UND-Verknüpfung $\gg \cdot \ll$ (auch: \wedge ; $\&$)
- der Disjunktion/ODER-Verknüpfung $\gg + \ll$ (auch: \vee)
- der Negation/NICHT-Verknüpfung $\gg - \ll$ (auch \neg)
- den beiden neutralen Elementen 0 und 1.

Wir verwenden hier die $\gg \cdot \ll$, $\gg + \ll$ und $\gg - \ll$ für die UND-, ODER und NICHT-Verknüpfung, da viele Regeln (z.B. Punkt vor Strichrechnung) ähnlich sind.

Wie bei der gewöhnlichen Multiplikation kann der $\gg \cdot \ll$ weggelassen werden, also $a \cdot b \cdot c = abc$

Boolesche Funktionen

Zur Beschreibung kombinatorischer Schaltungen sind Boolesche Funktionen (auch Schaltfunktionen oder logische Funktionen genannt) geeignet.

Das sind Ausdrücke in denen Variablen und Konstanten durch Boolesche Operatoren verknüpft werden.

Durch Auswertung der Ausdrücke ergibt sich der Funktionswert.

Beispiel: $f(a, b, c) = (a + \overline{b}) \cdot \overline{c} + \overline{b + a}$

Reihenfolge der Operatoren bei der Auswertung:

- 1 Klammersausdrücke
- 2 Negation (ist äquivalent zu Klammer)
- 3 Konjunktion (UND)
- 4 Disjunktion (ODER)

Darstellung von Booleschen Funktionen



Darstellung durch Wertetabellen

Index	x_{n-1}	x_{n-2}	...	x_2	x_1	x_0	$y = f(x_{n-1}, \dots, x_1, x_0)$
0	0	0	...	0	0	0	$f(0, 0, \dots, 0, 0, 0)$
1	0	0	...	0	0	1	$f(0, 0, \dots, 0, 0, 1)$
2	0	0	...	0	1	0	$f(0, 0, \dots, 0, 1, 0)$
3	0	0	...	0	1	1	$f(0, 0, \dots, 0, 1, 1)$
\vdots	\vdots	\vdots		\vdots	\vdots	\vdots	\vdots
$2^n - 2$	1	1	...	1	1	0	$f(1, 1, \dots, 1, 1, 0)$
$2^n - 1$	1	1	...	1	1	1	$f(1, 1, \dots, 1, 1, 1)$

Wertetabelle einer Funktion

Beispiel:

$$f(x_0, x_1) = \overline{x_0} x_1 + x_0 \overline{x_1}$$

x_0	x_1	$f(x_0, x_1)$
0	0	0
0	1	1
1	0	1
1	1	0

➡ Exklusiv-Oder (*Exclusive Or*, XOR): $f(x_0, x_1) = x_0 \oplus x_1$
 (Genau dann "1", wenn eine Variable exklusiv "1" ist.)

Vorlesungsaufgabe

Bestimmen Sie die Wertetabelle der Funktion

$$g(x_0, x_1) = (x_0 + x_1) \cdot (\overline{x_0} + \overline{x_1})$$

x_0	x_1	$g(x_0, x_1)$
0	0	
0	1	
1	0	
1	1	