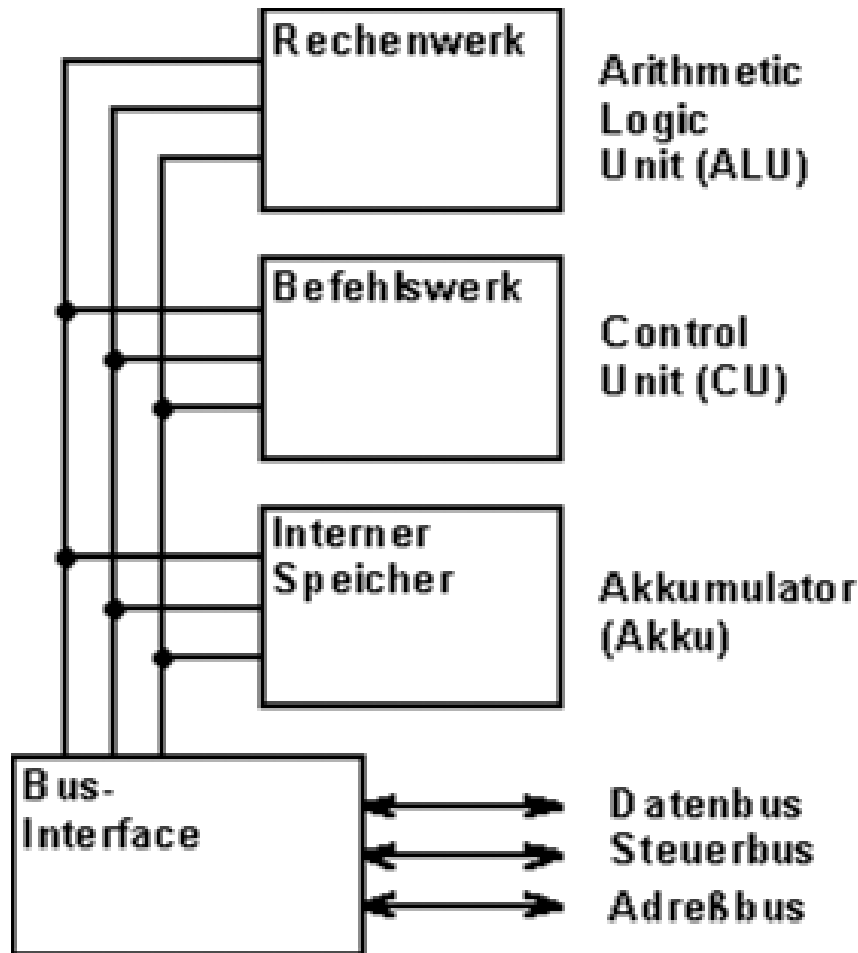


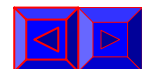
Mikroprozessor (CPU)



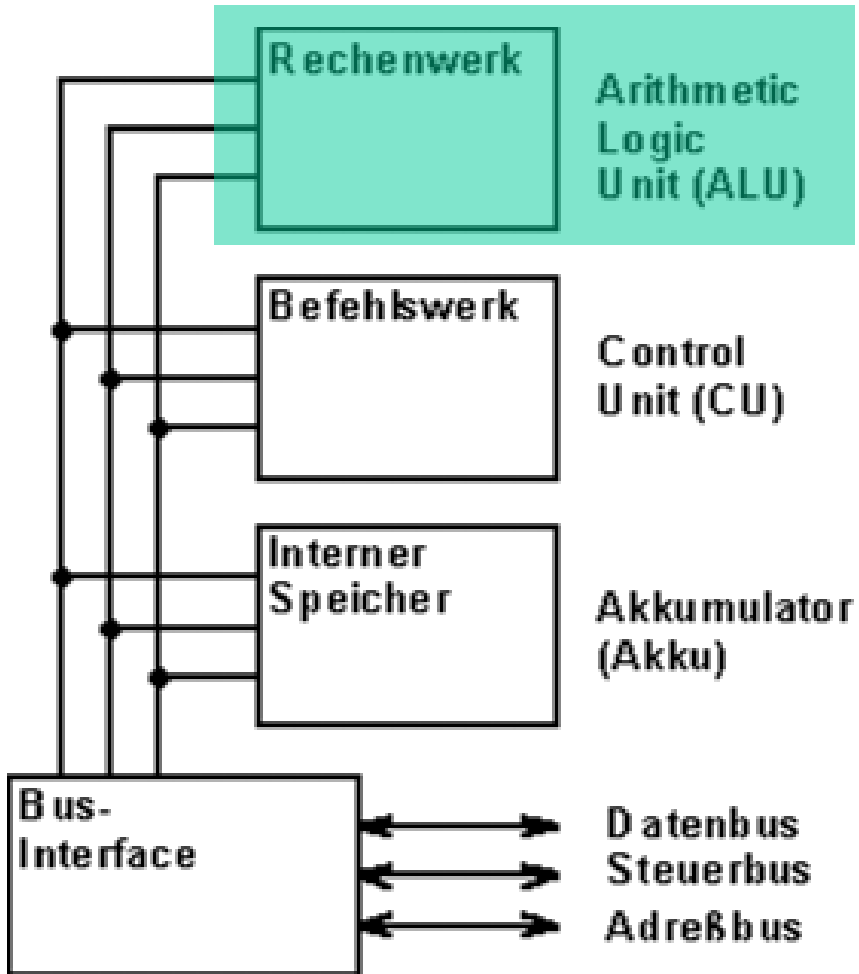
Der Mikroprozessor (μP) ist heutzutage das Herzstück eines jeden modernen Gerätes. Er wird in Handys, Taschenrechnern, HiFi-Geräten und in Computern, für die er eigentlich erfunden wurde, eingesetzt. Eine Welt ohne diese Rechengenieis ist nicht mehr vorzustellen.

Obwohl jeder Mikroprozessor für eine spezielle Anwendung entwickelt wird, sind sie vom Grundaufbau her alle gleich.

Die Central-Processing-Unit, kurz CPU, besteht grob gesehen aus dem Rechenwerk, dem Befehlswerk und dem internen Speicher. Über das Bus-Interface greift die CPU auf den Systembus zu.



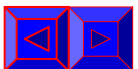
Mikroprozessor (CPU)



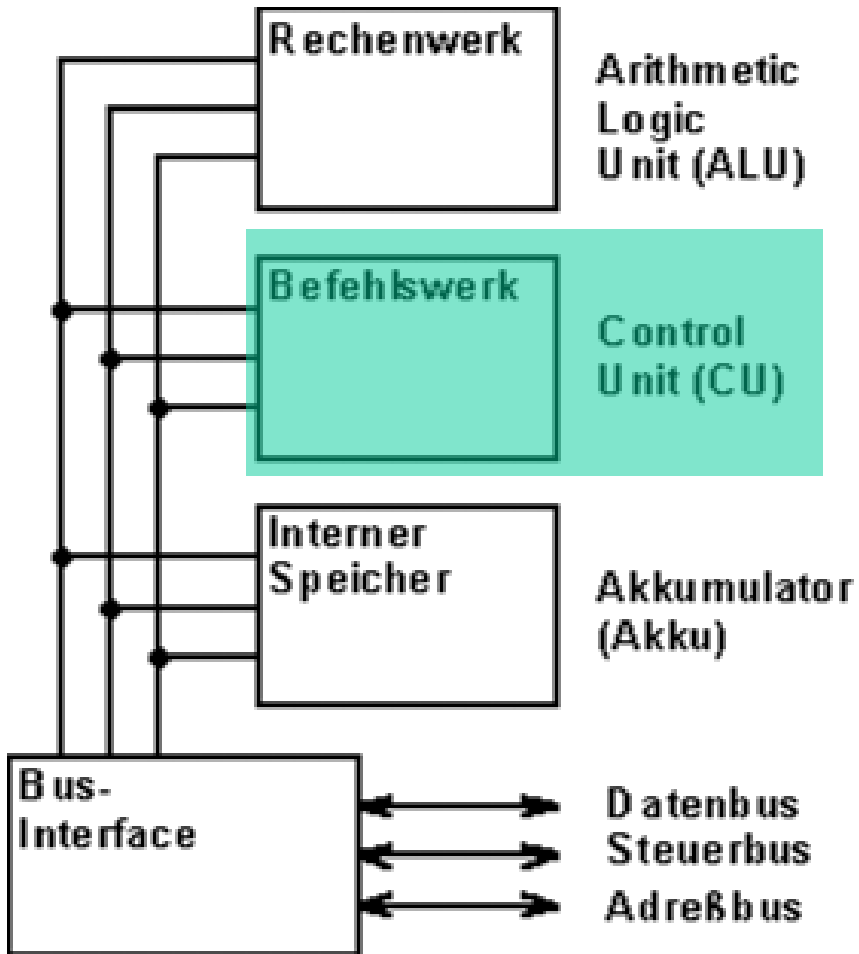
Arithmetic Logic Unit (ALU)

Die ALU ist der eigentliche Rechner. In ihr werden alle arithmetische und logische Funktionen und Berechnungen ausgeführt.

Zur ALU gehören auch der Akku (Speicher) und die Flags (Ereignispeicher).

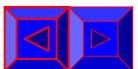


Mikroprozessor (CPU)

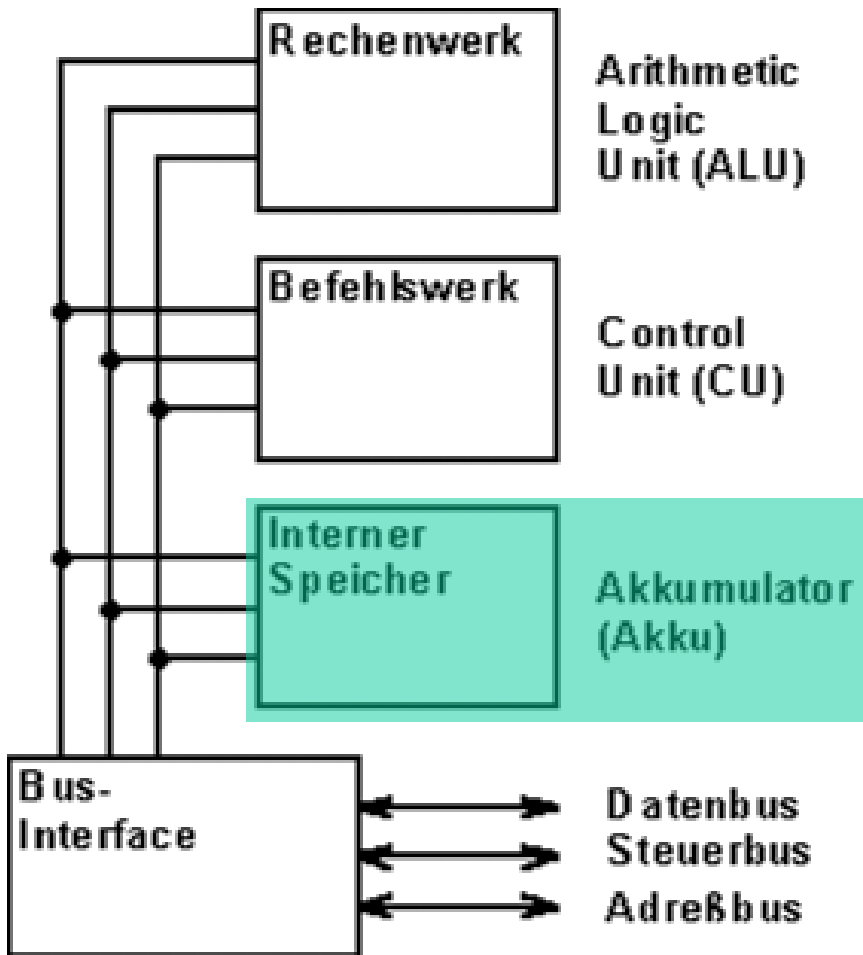


Control Unit (CU)

In der Control Unit befindet sich das Befehlsregister, das alle Befehle enthält, die der Mikroprozessor ausführen kann. Hier werden auch die Befehle dekodiert. Dann gibt es noch eine zeitliche und logische Steuerung, die auf die ALU bei Rechenoperationen zugreift. Von hier wird auch der Steuerbus, Interrupts und serielle Ein- und Ausgabe gesteuert.

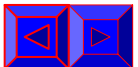


Mikroprozessor (CPU)

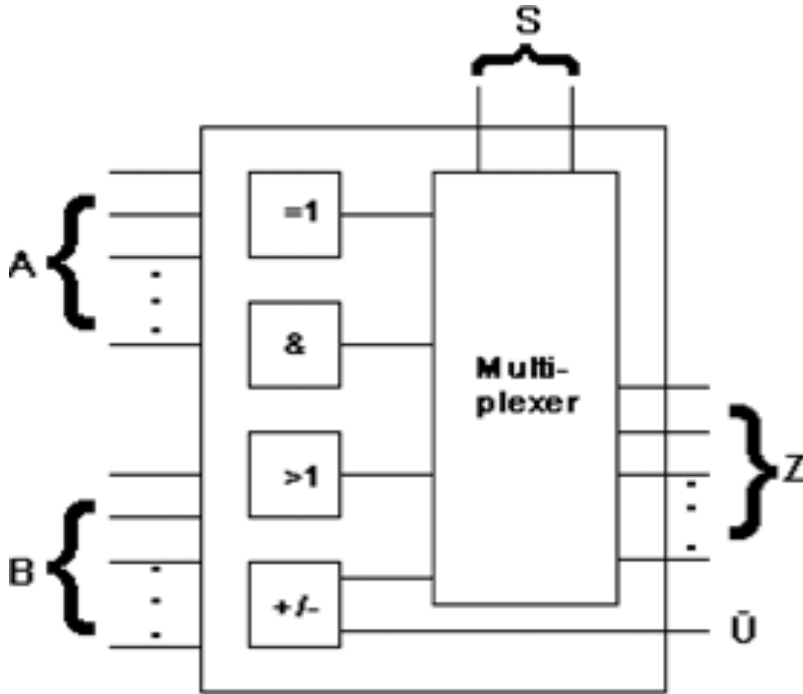


Interner Speicher (Akku)

Zum internen Speicher gehören wichtige Register, die als Zwischen-Speicher dienen und der Befehlszähler, in dem steht aus welcher Speicherzelle der nächste Befehl geladen wird.



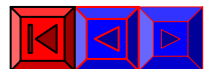
Arithmetisch-logische Einheit (ALU)



Die ALU ist eine Universal-Schaltung, die zwei n-Bit Datenworte wahlweise

- addiert
- subtrahiert
- vergleicht
- UND-verknüpft
- ODER-verknüpft
- XOR-verknüpft

Jede ALU hat die Dateneingänge A und B. Die Signalzustände an diesen Eingängen werden für die Operationen genutzt. Jede Operation wird in der ALU gleichzeitig ausgeführt. Welches Ergebnis an die Ausgänge gelangt, bestimmen die Codiereingänge S, an denen die jeweilige Operation eingestellt wird. Die Addier-/Subtrahier-Operationen können zusätzlich noch einen Übertrag erzeugen. Zur Signalisierung wird der Übertragsausgang Ü genutzt.

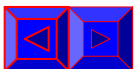


Integerarithmetik

$$\begin{array}{r} 1\ 0\ 0\ 1 \\ +\ 1\ 0\ 1\ 1 \\ \hline 1\ 0\ 1\ 0\ 0 \\ \hline \hline 9 + 11 = 20 \end{array}$$

$$\begin{array}{r} 1\ 0\ 1\ 1\ * \ 1\ 0\ 0\ 1 \\ \hline 1\ 0\ 1\ 1 \\ 0\ 0\ 0\ 0 \\ 0\ 0\ 0\ 0 \\ 1\ 0\ 1\ 1 \\ \hline 1\ 1\ 0\ 0\ 0\ 1\ 1 \\ \hline \hline 11 * 9 = 99 \end{array}$$

Wie lassen sich diese Rechenoperationen mit elektrischen bzw. elektronischen Bauelementen realisieren ?

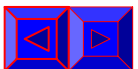


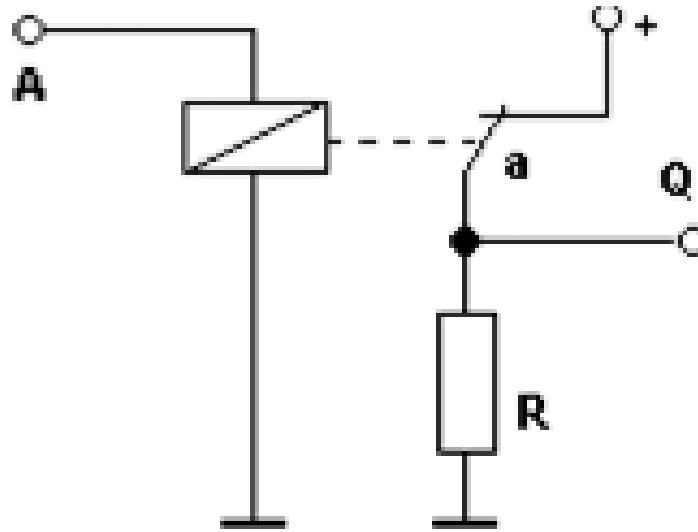
Für die Addition:	S	Ü
0 + 0	0	0
0 + 1	1	0
1 + 0	1	0
1 + 1	0	1

Wie lassen sich Summe und Übertrag durch logische Funktionen darstellen ?

B1	B2	B1 and B2	B1 or B2	B1 xor B2
0	0	0	0	0
0	1	0	1	1
1	0	0	1	1
1	1	1	1	0

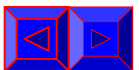
Lassen sich diese logischen Verknüpfungen mit elektronischen Bauelementen realisieren ?

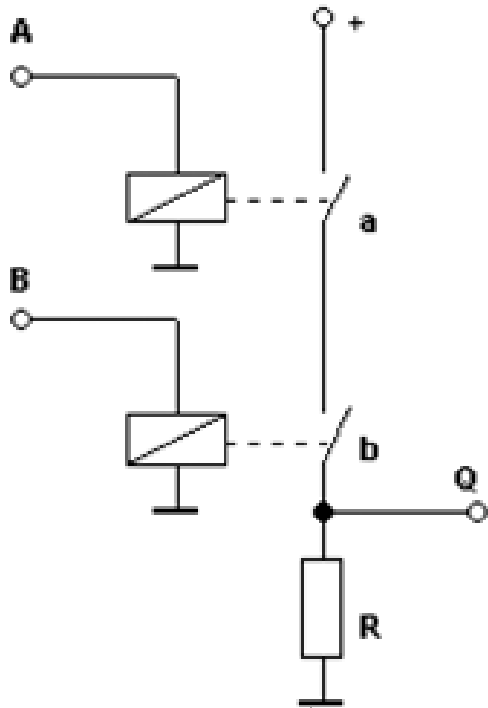




$$Q \equiv \text{not}(A) \equiv \neg A \equiv \bar{A}$$

An Q ist Spannung genau dann, wenn an A keine Spannung

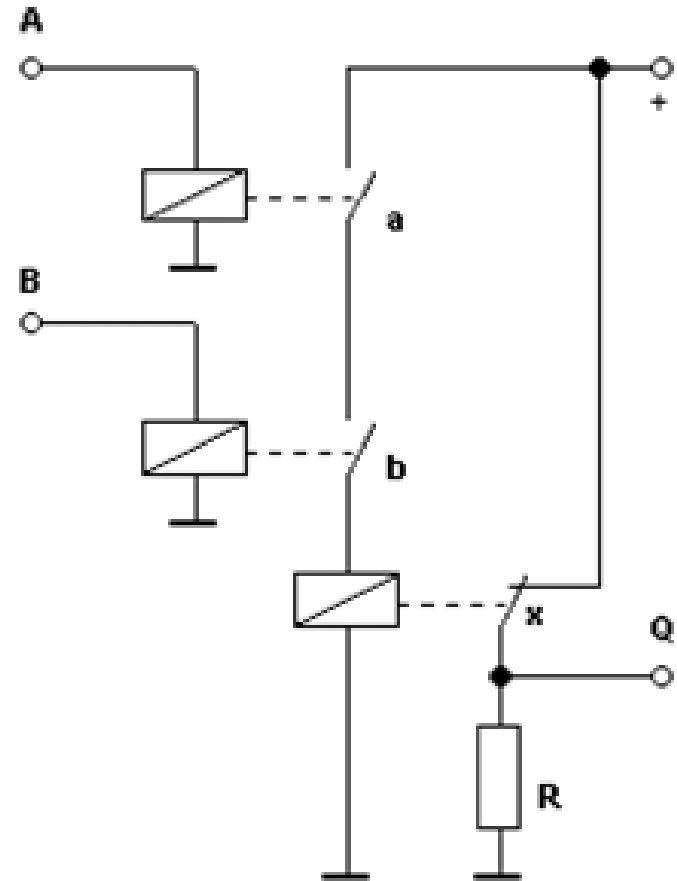




$$Q \equiv A \text{ and } B \equiv A \wedge B$$

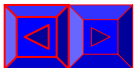
An Q ist Spannung genau dann,
wenn an A und an B Spannung

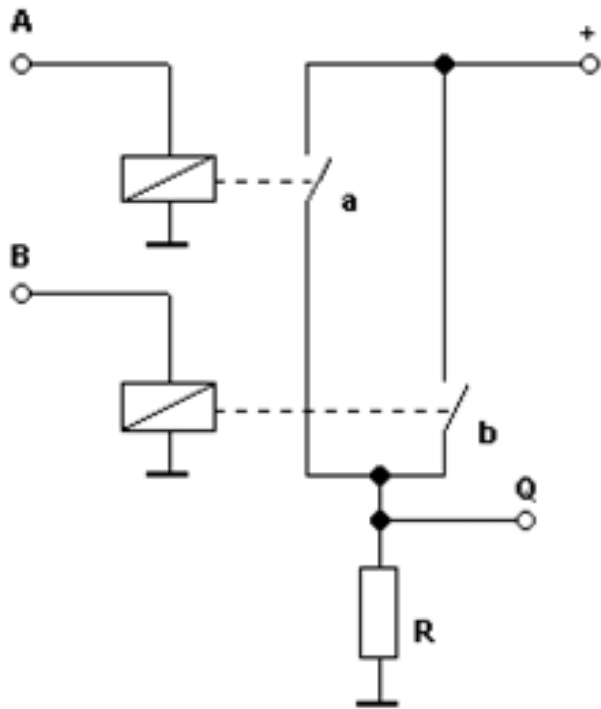
Wie sagt man das
auf Deutsch ???



$$Q \equiv A \text{ nand } B \equiv \neg(A \wedge B) \equiv \overline{A \wedge B}$$

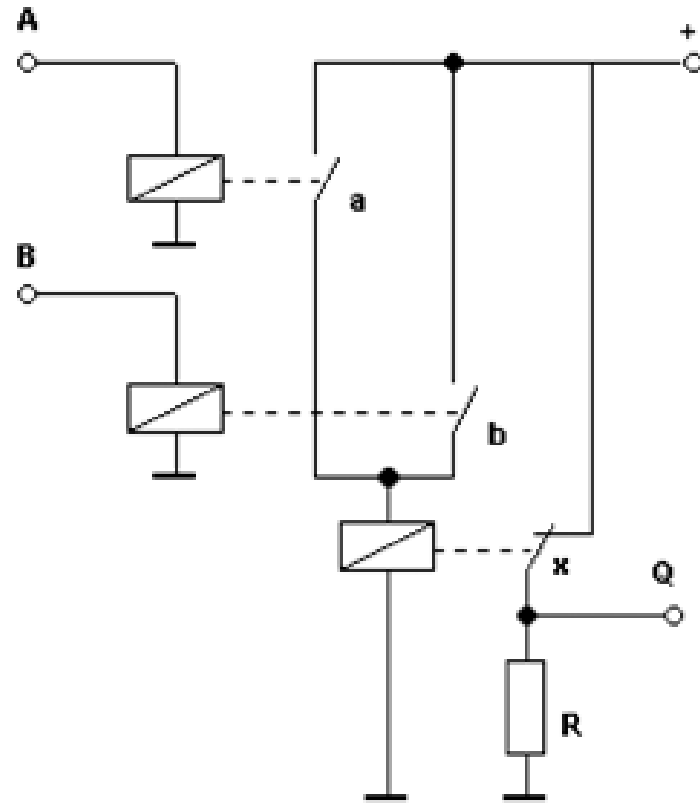
An Q ist Spannung genau dann, wenn
nicht (an A und an B Spannung)





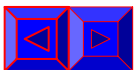
$$Q \equiv A \text{ or } B \equiv A \vee B$$

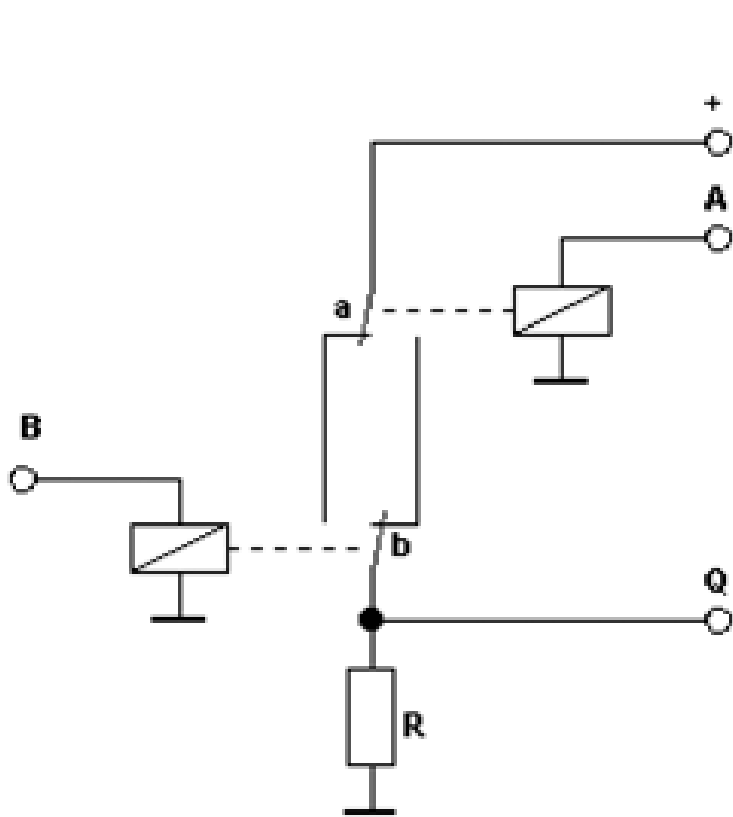
An Q ist Spannung genau dann, wenn an A oder an B Spannung



$$Q \equiv A \text{ nor } B \equiv \neg(A \vee B) \equiv \overline{A \vee B}$$

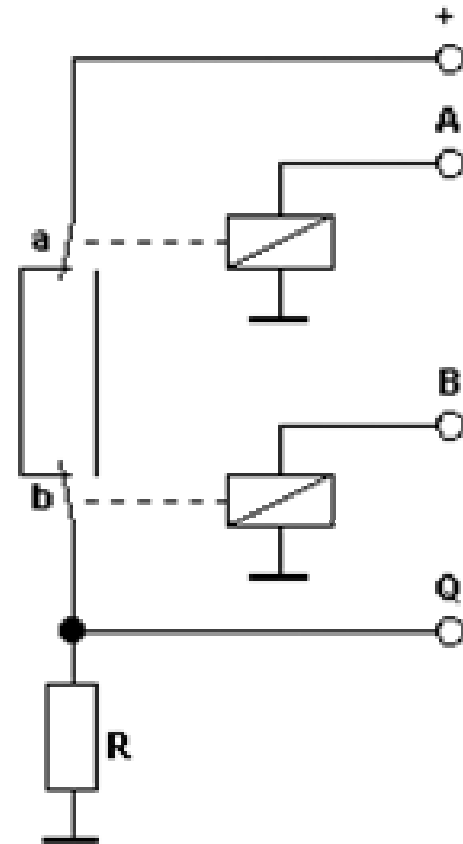
An Q ist Spannung genau dann, wenn nicht (an A oder an B Spannung)





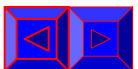
$$Q \equiv A \text{ xor } B \equiv A \oplus B$$

An Q ist Spannung genau dann, wenn entweder an A Spannung oder an B Spannung



$$Q \equiv A \text{ xnor } B \equiv \neg(A \oplus B) \equiv \overline{A \oplus B}$$

An Q ist Spannung genau dann, wenn nicht (entweder an A oder an B Spannung)



Inhalt des kommenden Unterrichts wird das Finden von Schaltungen für die einfachsten Aufgaben in der ALU sein (z.B. Addition/Multiplikation zweier Dualzahlen, Vergleich von Dualzahlen, ...). Die Schaltungen werden dabei mit **DIGITAL WORKS** erstellt. Für jede Schaltung wird ein Programm zu deren **Simulation** geschrieben.

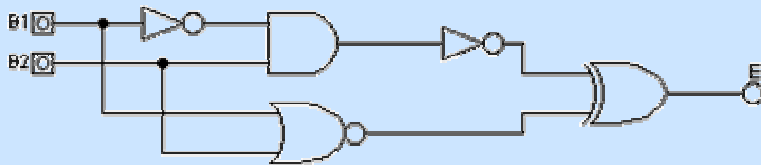
In Digital Works gibt es für die einzelnen logischen Verknüpfungen folgende Schaltsymbole:



Diese digitale Schaltung

realisiert

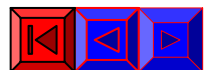
diese logische Äquivalenz.



$$E \equiv \overline{\overline{B1} \wedge B2} \text{ XOR } \overline{B1 \vee B2}$$

$$\equiv \neg(\neg B1 \wedge B2) \text{ XOR } \neg(B1 \vee B2)$$

DigitalWorks starten



Gleitkommazahlen nach dem IEEE-Standard 754 von 1985

Bis 1985 hatte quasi jeder Prozessorhersteller seinen eigenen Standard. Damit hatte man das Problem, dass dasselbe Programm auf verschiedenen Rechnern verschiedene Ergebnisse liefern konnte.

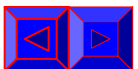
Der IEEE-Standard hat sich seit 1992/93 durchgesetzt und wird von den führenden Prozessorherstellern eingehalten.

Nicht der gesamte Standard kann hardwaremäßig implementiert werden, einige Rechnungen müssen per Software durchgeführt werden, damit nicht der gesamte Rechner ausgebremst wird

Vorzeichen s



Für den Typ SINGLE stehen zur Verfügung für : s : 1 Bit e : 8 Bit f : 23 Bit



Für die Umrechnung aus der Gleitkommadarstellung in die Dezimaldarstellung gilt: $x = (-1)^s \cdot 2^{e-127} \cdot 1.f$

Auch wenn eine normalisierte Darstellung von x nicht möglich ist (x ist zu klein), legt der Standard Regeln fest.

f ist der Nachkommateil der normalisierten Darstellung der Single-Zahl

Beispiel: $x=13,7$

- Lies das Vorzeichen ab s=0
- Normalisiere x (dividiere x durch die größte Zweierpotenz, die kleiner als x ist) $13,7:2^3=1,7125$
- es ergeben sich **e=3+127** und **f=0,7125**
- stelle f im Dualsystem dar f=0,1011011001100110011001100110011...

Somit ergibt sich für die Gleitkommadarstellung von 13,7:

P1. P2.

1	1	2	3	4	5	6	7	8	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23
0	1	0	0	0	0	0	1	0	1	0	1	1	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1

Um beim Weiterrechnen möglichst genaue Ergebnisse zu erhalten, gibt der IEEE-Standard verschiedene Modi zum Runden vor. Um den günstigsten zu finden, rechnet der Standard intern in der Mantisse mit einigen Bits mehr.