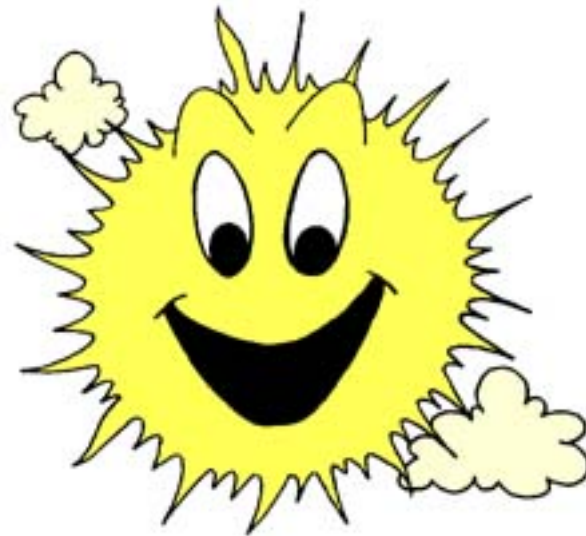


Wiederholung der 8. Vorlesung



5.8 Quine und Mc Cluskey – Verfahren

Minimierungsverfahren von Quine und Mc Cluskey eignet sich für die Implementierung auf Rechnern. Es wird im Folgenden für die DNF vorgestellt.

Für die KNF würde es prinzipiell in ähnlicher Weise erfolgen.

1. Ermittlung aller Minterme der Funktion.
2. Unterteilung der Minterme in Gruppen. Eine Gruppe fasst Minterme mit gleicher Anzahl negierter Eingangsvariablen zusammen.
3. Paarweises Zusammenfassen von Termen benachbarter Gruppen zu Termen geringerer Komplexität durch Anwendung der 5. Kürzungsregel. Kennzeichnung der zusammengefassten Terme.

...

5.8 Quine und Mc Cluskey – Verfahren (2)

...

- Wiederholung von Schritt 3 für die zusammengefassten Terme, bis keine Vereinfachung mehr durchgeführt werden kann.
Ist keine Vereinfachung mehr möglich, bilden alle nicht gekennzeichneten Minterme und Terme die Primimplikanten der Funktion.
- Ermittlung der Kern-Primimplikanten aus den gefundenen Primimplikanten. Die Kern-Primimplikanten gehören auf jeden Fall zu den Termen der gesuchten Funktion.
- Hinzufügen von Primimplikanten zur Funktion, bis alle Minterme der Funktion berücksichtigt sind.

5.8 2. Beispiel Quine und Mc Cluskey

x4	x3	x2	x1	f	DNF enthält folgende Minterme	Kombination ergibt	
0	0	0	0	1	m	Vektor	
0	0	0	1	0			
0	0	1	0	1	0	0000	
0	0	1	1	0	2	0010	
0	1	0	0	1	4	0100	
0	1	0	1	1	5	0101	
0	1	1	0	1	6	0110	
0	1	1	1	1	7	0111	
1	0	0	0	0			
1	0	0	1	0	10	1010	
1	0	1	1	1	11	1011	
1	1	0	0	0			
1	1	0	1	0			
1	1	1	0	0			
1	1	1	1	0			
						m	Vektor
						0,2	00-0
						0,4	0-00
						2,0	wie oben
						2,6	0-10
						2,10	-010
						4,0	wie oben
						4,5	010-
						4,6	01-0
						5,4	wie oben
						5,7	01-1
						6,2	wie oben
						6,4	wie oben
						6,7	011-
						7,5	wie oben
						7,6	wie oben
						10,2	wie oben
						10,11	101-
						11,10	wie oben

5.8 2. Beispiel Quine und Mc Cluskey (3)

m	Vektor	ursprünglich enthalten
0,2;4,6	0-0	0000 0010 0100 0110
2,10	-010	0010 1010
4,5;6,7	01--	0100 0101 0110 0111
10,11	101-	1010 1011

Wenn man nun gezielt die Doppel so streicht, dass eine ganze Zeile überflüssig wird, findet man das tatsächliche Optimum. In diesem Fall würde die zweite Zeile entfallen.

	x1	x1		
	1	0	1	1
x2	1	0	1	1
x2	1	1	0	0
	0	0	0	0
		x3	x3	

Die **optimierte** Funktion lautet dann:

$$y = \neg x_4 \cdot \neg x_1 + \neg x_4 \cdot x_3 + x_4 \cdot \neg x_3 \cdot x_2$$

5.8 Binäre Entscheidungsdiagramme

Boolesche Ausdrücke können auch durch sogenannte geordnete binäre Entscheidungsdiagramme (**OBDD** **Ordered Boolean Decision Diagram**) dargestellt werden. Diese sind oft sehr kompakt und lassen sich algorithmisch effizient behandeln.

OBDDs finden in vielen Bereichen der Informatik eine Anwendung.

5.8 Binäre Entscheidungsdiagramme (2)

Ein OBDD ist ein gerichteter azyklischer Graph mit Wurzel, Zwischenknoten und zwei Endknoten (0-Knoten und 1-Knoten) ohne Ausgangskanten.

Die Zwischenknoten sind mit Booleschen Variablen beschriftet.

Von jedem Zwischenknoten gehen genau zwei Kanten aus (0-Kante und 1-Kante).

Die Variablen sind linear geordnet, z.B. $a < b < c < \dots$, und tauchen in dieser Reihenfolge von der Wurzel beginnend im Graphen auf, d.h., jedem Zwischenknoten ist ein Variablenname zugeordnet, wobei gilt, daß für alle Pfade von der Wurzel des Graphen zu einem der beiden Endknoten die Variablen in der gleichen Reihenfolge auftreten.

Es müssen aber nicht in jedem Pfad alle Variablennamen auftreten! (Ordnung).

5.8 Beispiel OBDD

Der Boolescher Ausdruck $Z = (\overline{a}bc + de) \cdot f \cdot (\overline{g} + hi)$ lässt sich wie folgt zerlegen:

$$Z1 = \overline{a}bc$$

$$Z2 = de$$

$$Z3 = \overline{g}$$

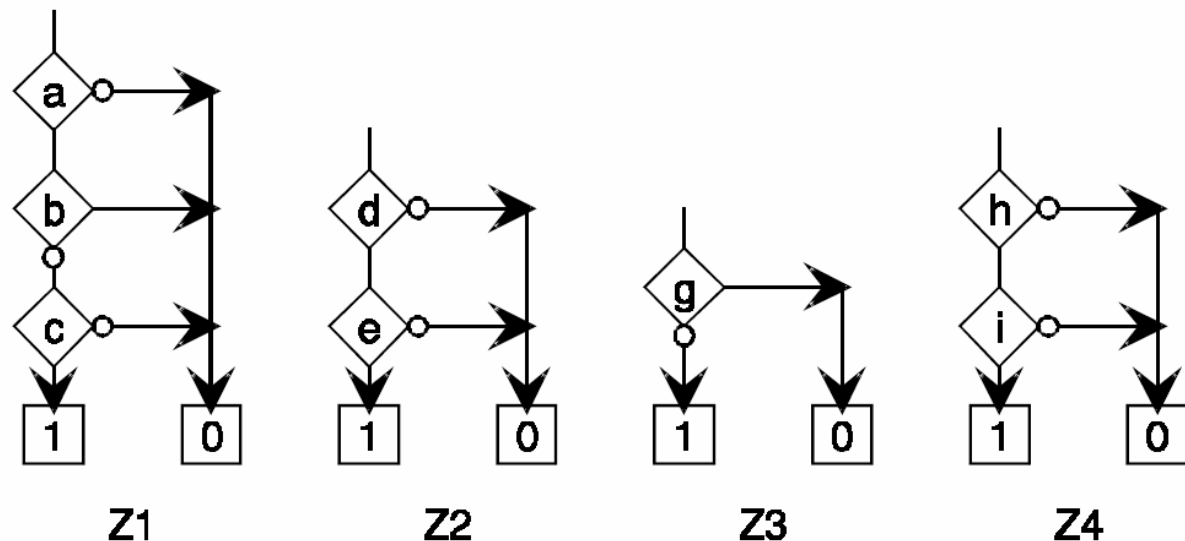
$$Z4 = hi$$

$$Z5 = Z1 + Z2$$

$$Z6 = Z3 + Z4$$

$$Z = Z5 \cdot f \cdot Z6$$

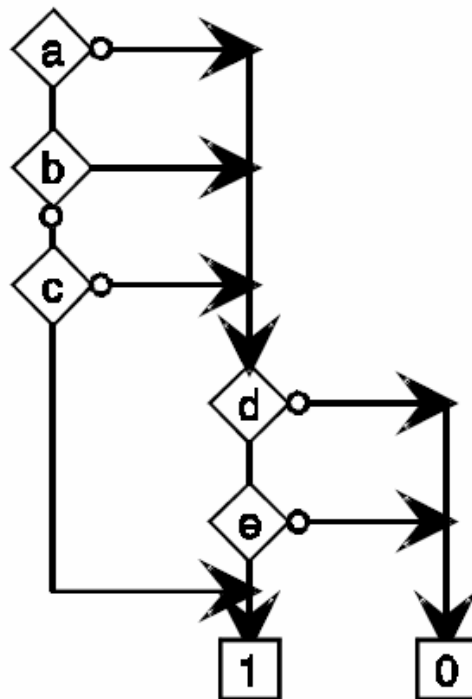
Für Z1 bis Z4 lassen sich sofort OBDDs angeben. Eine Raute steht für den Test einer binären Variablen auf Wert 0 oder 1. Wenn dieser Test (die Entscheidung) ein 1 ergibt, wird längs des Pfeiles (1-Kante); wenn sich eine 0 ergibt, wird längs der Kante mit Kreis (0-Kante) zum nächsten Test gegangen.



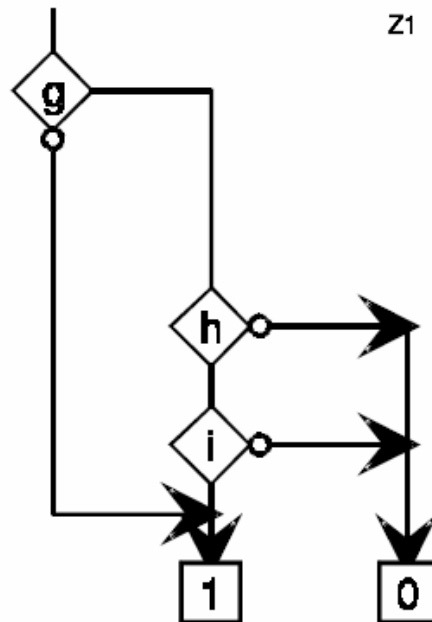
5.8 Beispiel OBDD (2)

Teildiagramme werden wie folgt zusammengesetzt:

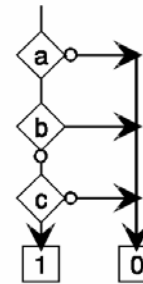
$$Z = (a\bar{b}c + de) \cdot f \cdot (\bar{g} + hi)$$



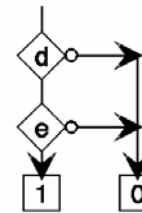
$$Z5 = Z1 + Z2$$



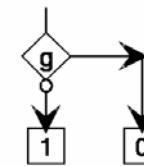
$$Z6 = Z3 + Z4$$



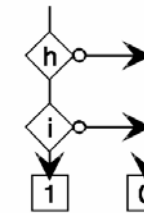
Z1



Z2



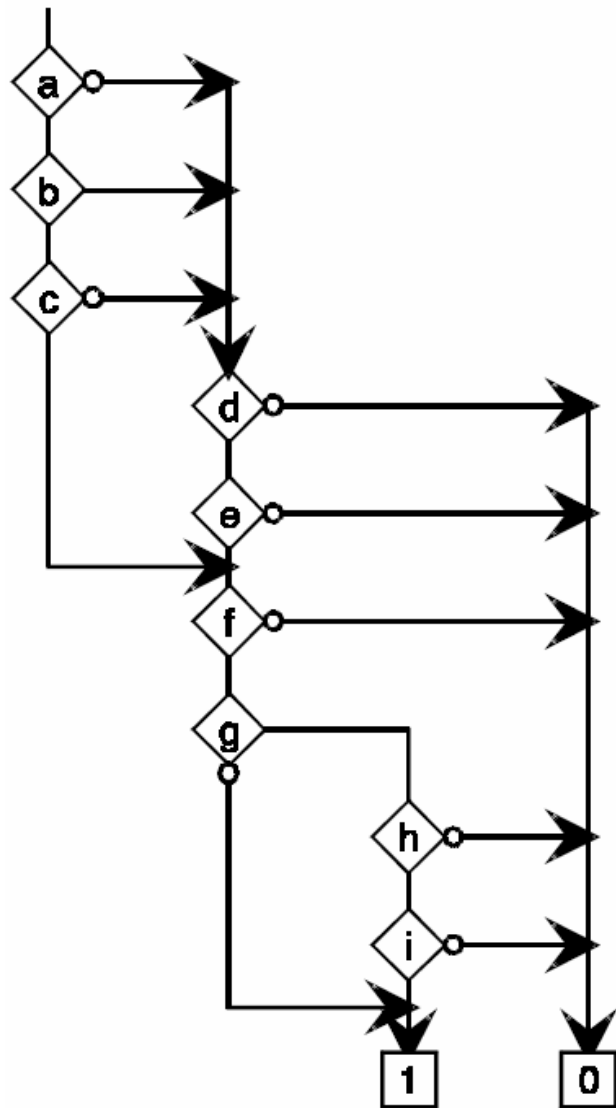
Z3



Z4

- ODER-Verknüpfung bedeutet:
- anhängen am 0-Knoten;
- UND-Verknüpfung bedeutet:
- anhängen am 1-Knoten.

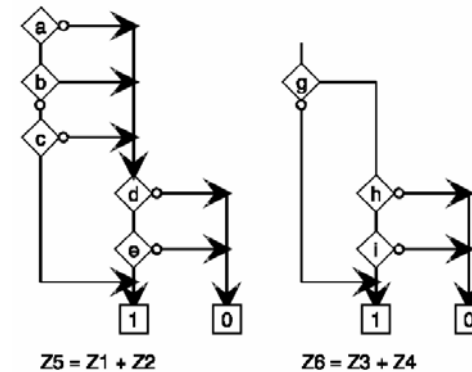
5.8 Beispiel OBDD (3)



Z

Der komplette OBDD für Funktion und Teildiagramme (s.u.) sieht dann wie folgt aus:

$$Z = (a\bar{b}c + de) \cdot f \cdot (\bar{g} + hi)$$



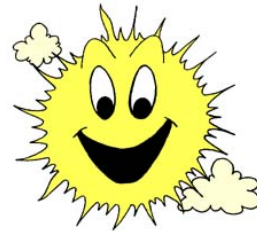
ODER-Verknüpfung bedeutet:

- anhängen am 0-Knoten;

UND-Verknüpfung bedeutet:

- anhängen am 1-Knoten.

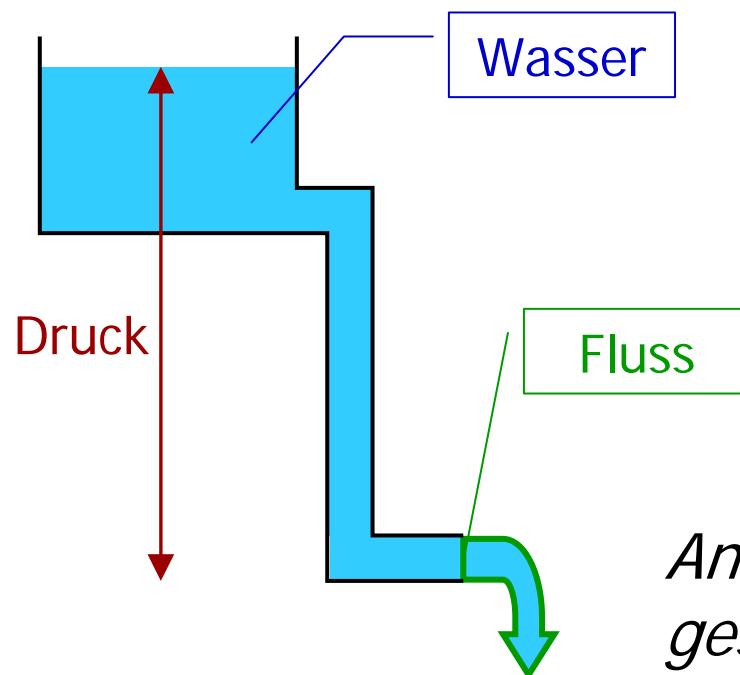
Ende der Wiederholung



6 Wiederholung Grundbegriffe E-Lehre

Elektrizität ist die Grundlage der meisten technischen Errungenschaften der Gegenwart.

*Bis zu gewissen Grenzen und mit der notwendigen Vorsicht kann man für **Elektrizität Analogien zu Flüssigkeiten** ziehen, z.B. Wasser.*

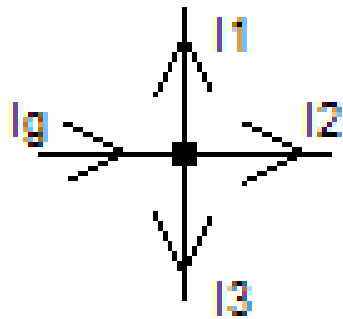


Analogie-Elektrizität:

- Druck – **Spannung U**
- Wasser – **Ladung Q**
- Fluss = Wasser / Zeit
- Fluss – **Strom I**
= Ladung / Zeit = **dQ/dt**

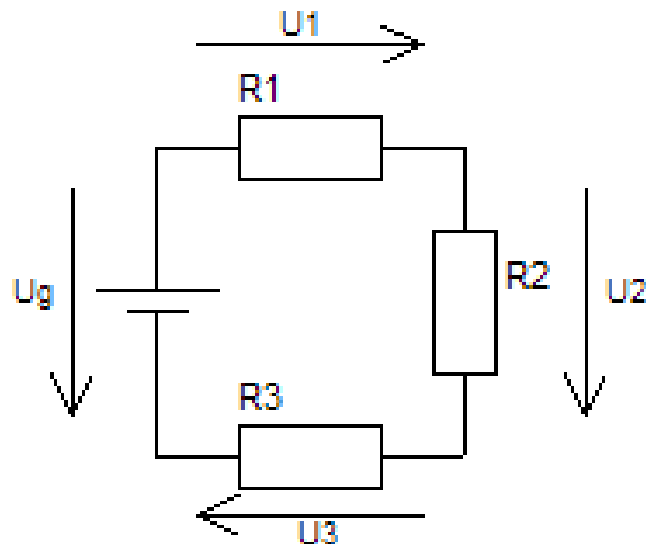
Analogie nur vernünftig bei einem geschlossenem Wasser-Kreislauf

6 Grundbegriffe Knoten- und Maschenregel



Die **Knotenregel** sagt aus, dass sämtliche Ströme eines Knoten in einer Schaltung in der Summe immer 0 ergeben müssen. D.h. die Summe aller zufließenden Ströme entspricht der Summe der abfließenden Ströme.

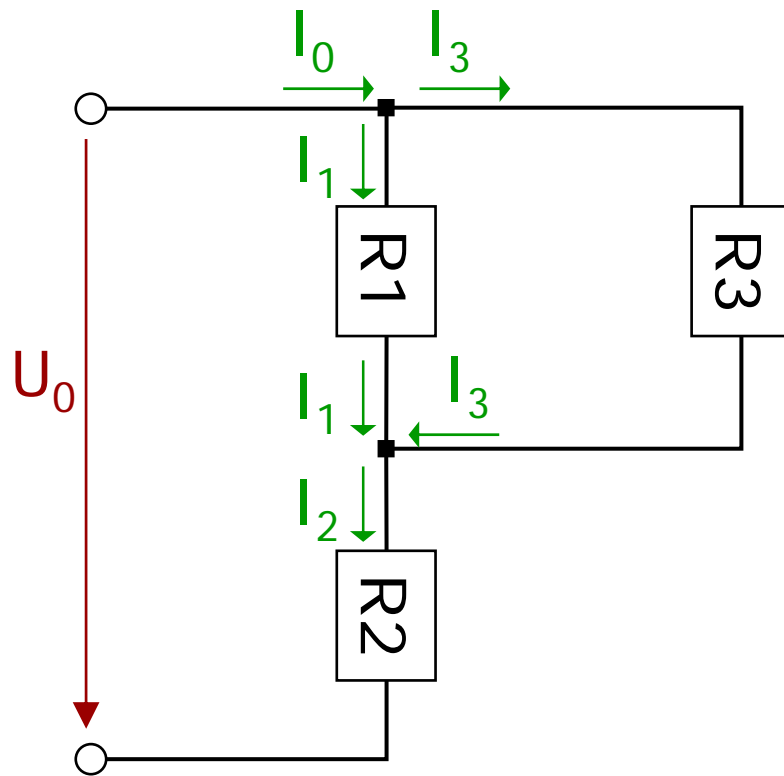
$$\sum_i I_i = 0$$



Die **Maschenregel** besagt, dass die Summe aller Spannungen innerhalb einer Masche 0 ergeben muss. D.h. die Werte der Spannungsquellen muss gleich der Summe der einzelnen Verbraucherspannungen sein.

$$\sum_i U_i = 0$$

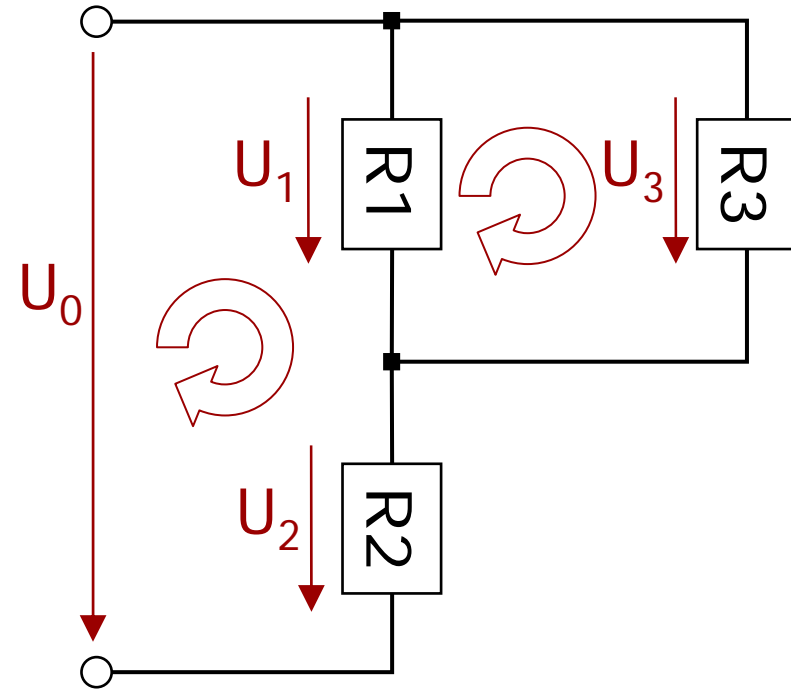
6 Grundbegriffe Knoten- und Maschenregel



Zwei **Knoten**:

1. Knoten: $I_0 = I_1 + I_3$
2. Knoten: $I_2 = I_1 + I_3$

$$\rightarrow I_2 = I_0$$



Zwei **Maschen**:

1. Masche: $U_0 + U_1 + U_2 = 0$
2. Masche: $U_1 + U_3 = 0$

$$\rightarrow U_0 = U_1 + U_2 \text{ und } U_1 = U_3$$

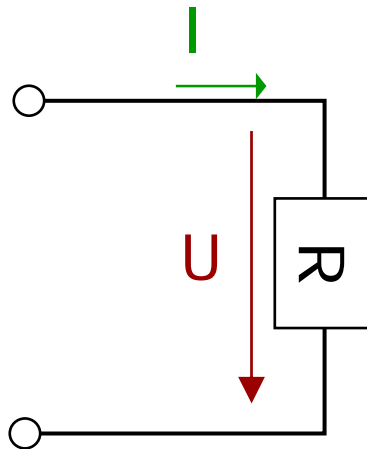
6 Spannung, Strom und Widerstand

Spannung U :

Einheit: Volt, V

Strom I :

Einheit: Ampère, A



Elektrischer **Widerstand**:

$$\mathbf{R = U / I} \quad \text{Einheit: Ohm, } \Omega \\ = \text{V/A}$$

$$\leftrightarrow \mathbf{U = R \times I}$$

$$\leftrightarrow \mathbf{I = U / R}$$

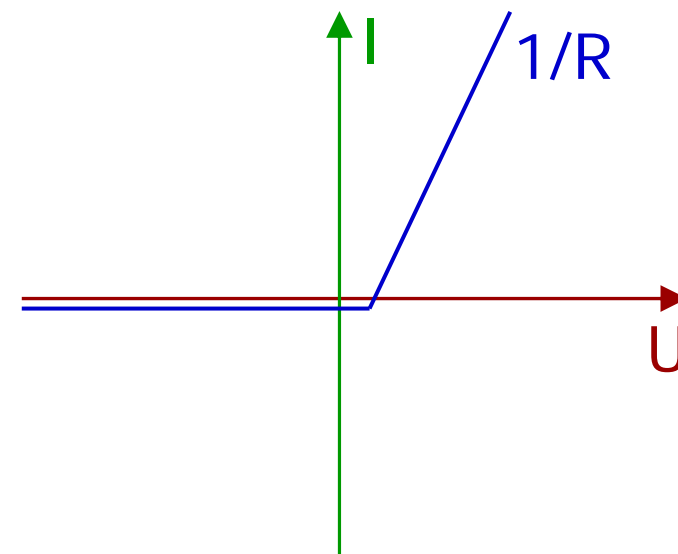
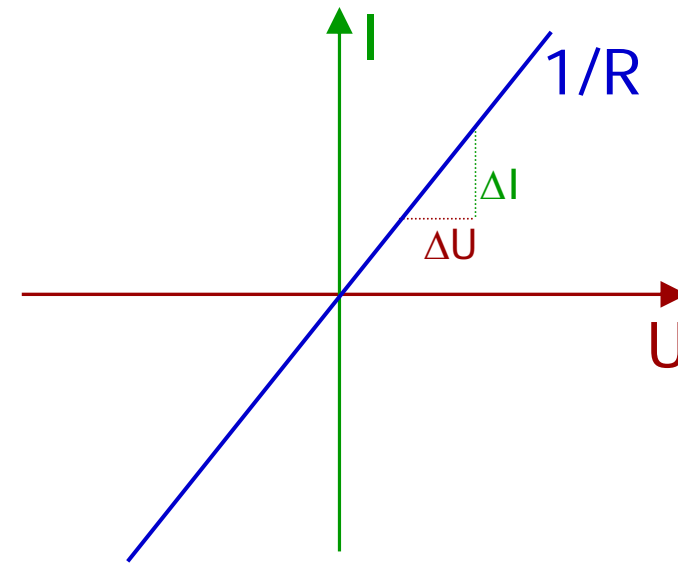
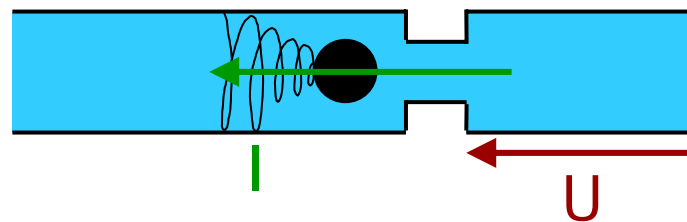
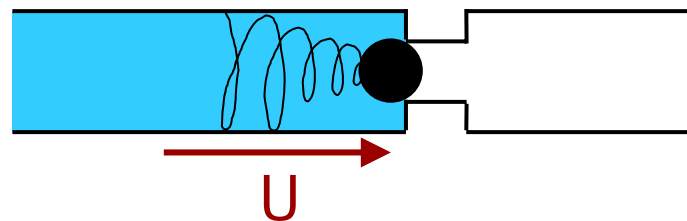
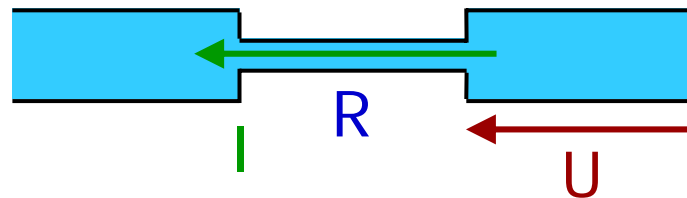
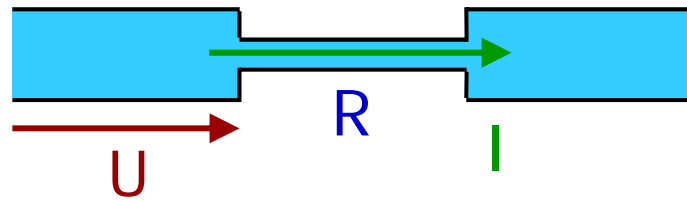
Leistung P : Einheit: Watt, VA

$$\mathbf{P = U \times I}$$

$$\leftrightarrow \mathbf{P = R \times I^2}$$

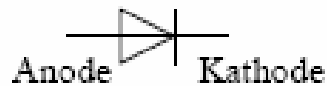
$$\leftrightarrow \mathbf{P = U^2 / R}$$

6 Widerstand – Ventil

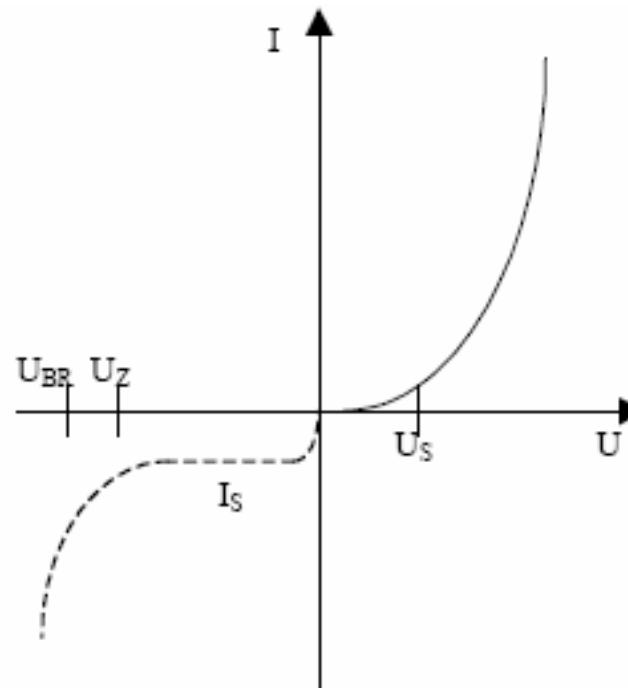
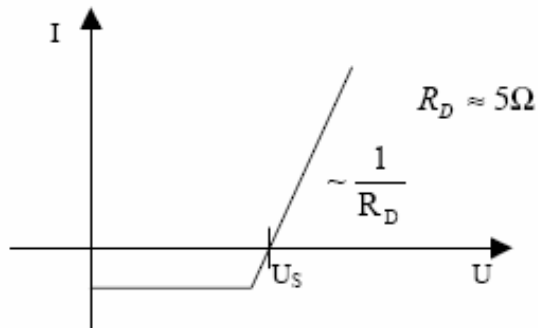


6. Diode

Halbleiterdioden besitzen einen Ventil- oder Gleichrichtereffekt ihres pn-Überganges, d.h. ein Stromfluss von **Anode** zu **Kathode** ist näherungsweise ideal möglich, während ein Stromfluß umgekehrt näherungsweise versperrt bleibt. Symbol und Kennlinie siehe unten.



Aufbau (Prinzip):



$$I = I_0 \cdot \left(e^{\frac{U}{U_s}} - 1 \right)$$

$$I = I_F - I_S$$

$$U_{S\text{Ge}} \approx 0,3\text{V}$$

$$U_{S\text{Si}} \approx 0,6\text{V}$$

Kenndaten:

U_F : Durchlaßspannung

U_R : Sperrspannung

U_S : Schwellenspannung

U_Z : Zenerspannung

I_F : Durchlaßstrom

I_S : Sperrstrom

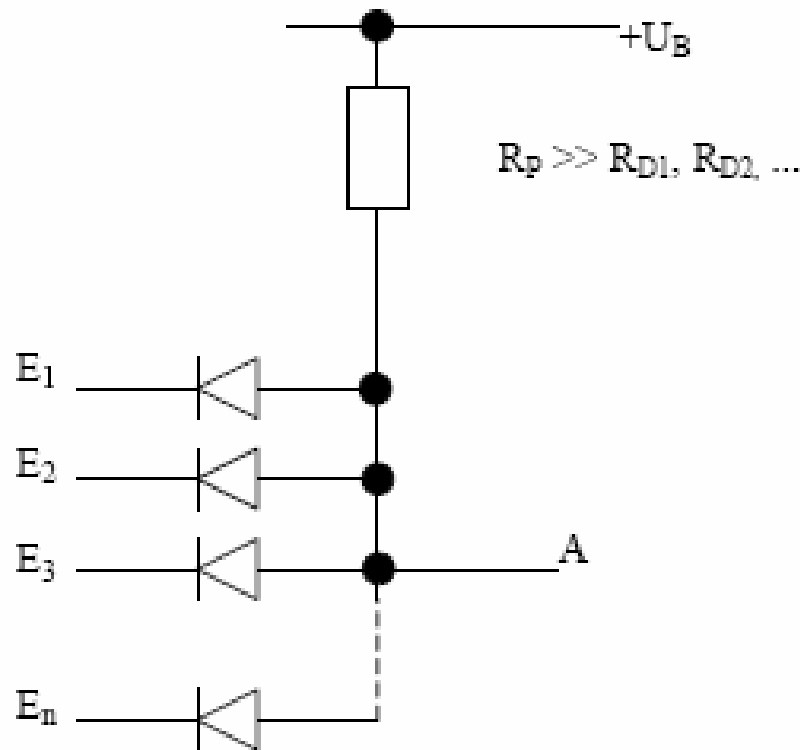
6 Dioden als logisches Element

- In einem Bereich $U > U_s$ ist die Diode durchlässig mit einem Widerstand R_D , **Durchlaßbereich**.
- In einem Bereich $U < U_s$ sperrt die Diode mit einem annähernd unendlichen Widerstand, **Sperrbereich**
- Diese Eigenschaften von Dioden erlauben den Aufbau einfacher logischer Schaltungen

Die folgenden Beispiele gehen von **positiver Logik** aus:

- ‚High‘ (hohe positive Spannung) = 1
- ‚Low‘ (niedrige Spannung) = 0

6 Dioden als logisches Element: UND



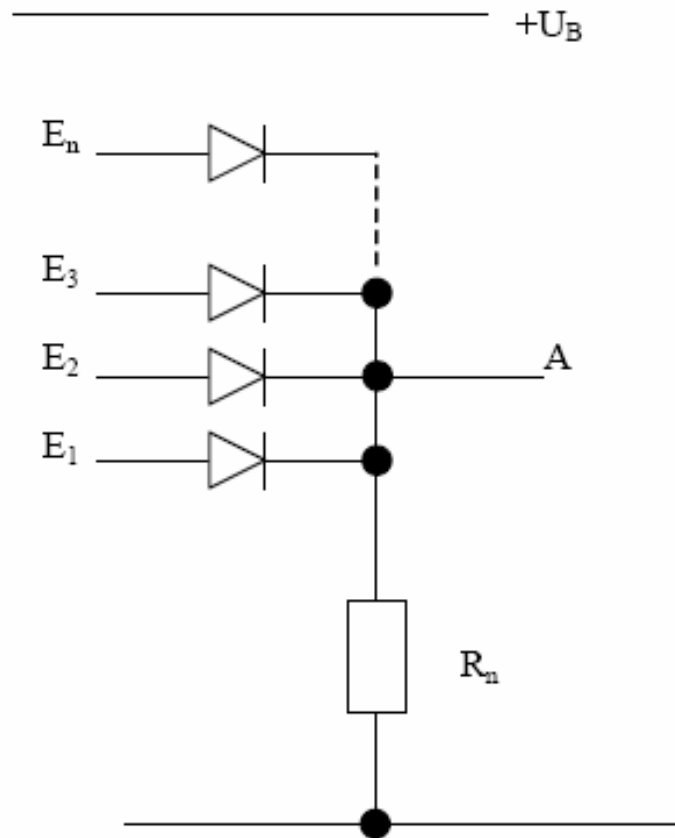
Nur wenn an allen Eingängen E_i die Betriebsspannung anliegt oder die Eingänge offen sind, ist auch am Ausgang die Spannung hoch, 'High'=1 bei positiver Logik

→ logische **UND-Schaltung**.

Liegt auch nur an einem Eingang eine niedrige Spannung ($\sim 0V$) an, so beträgt die unbelastete Ausgangsspannung $U_s = 0,6V$, also näherungsweise 'Low'=0 bei positiver Logik.

(Bei negativer Logik erhielten wir so eine ODER-Schaltung)

6 Dioden als logisches Element: ODER



Voraussetzung:
 $R_p \gg R_{D1}, R_{D2}, \dots$

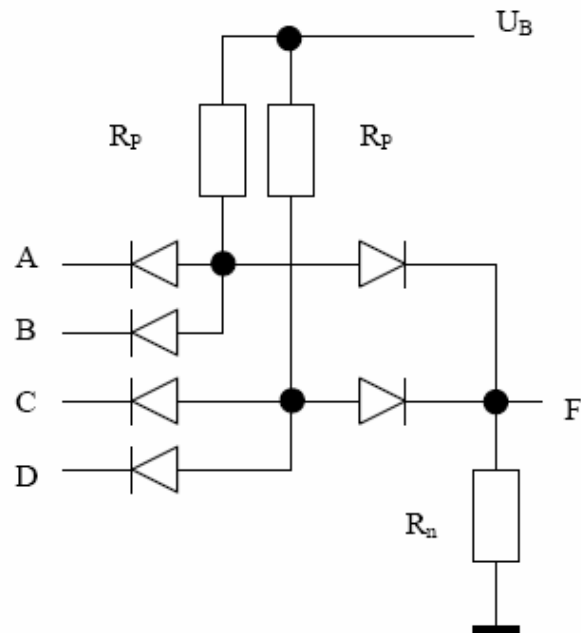
Auch für positive Logik lässt sich mit Dioden eine logische **ODER-Schaltung** realisieren.

Bereits ein Eingangssignal auf ‚H‘ reicht aus, um auch die Ausgangsspannung auf ‚H‘ zu bekommen.

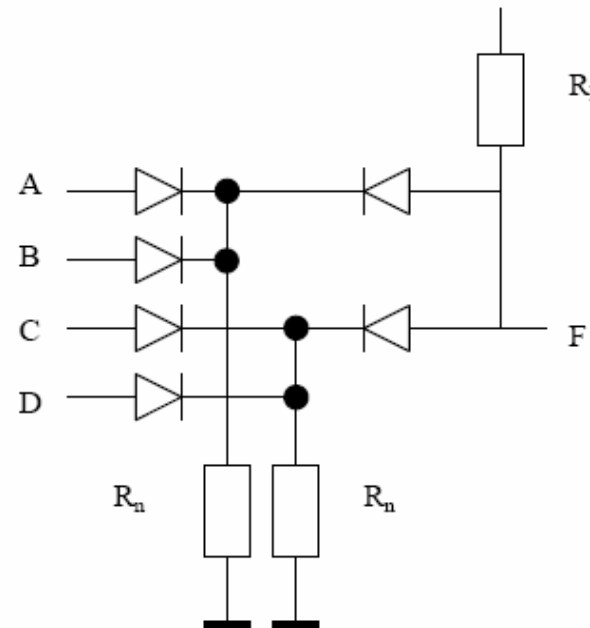
(Bei negativer Logik erhielten wir eine UND-Schaltung)

6 Logische Schaltnetze mit Dioden

Durch Kaskadierung solcher Schaltungen lassen sich logische Schaltnetze realisieren



$$F = (AB) + (CD)$$



$$F = (A+B)(C+D)$$

Für die Negation wird ein aktives Element benötigt:

→ **Transistor**