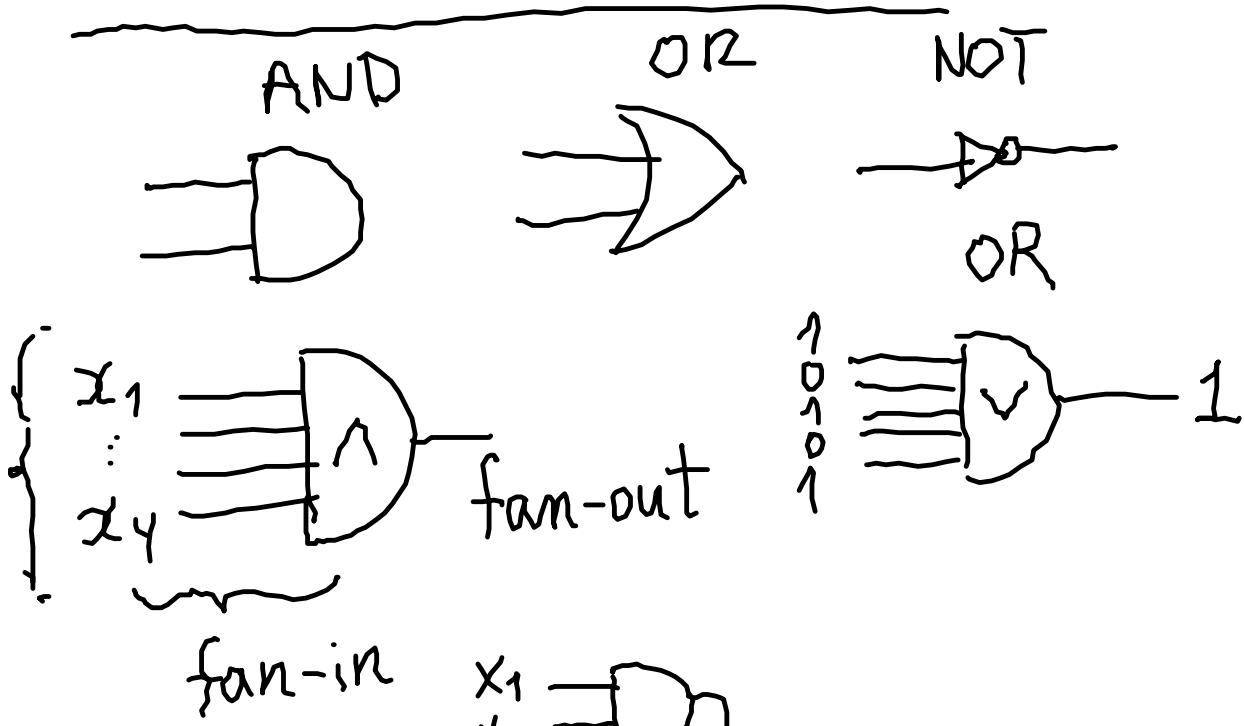
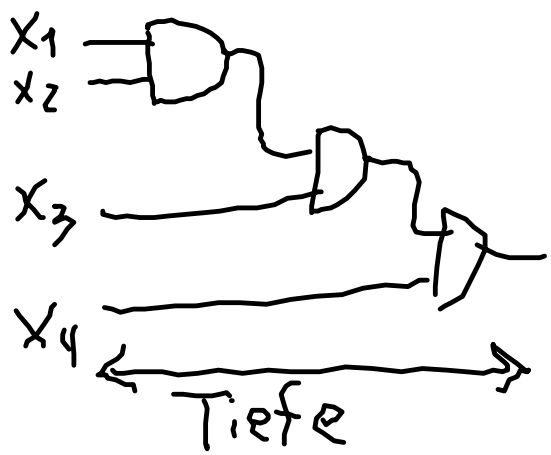
Neuronale Netze

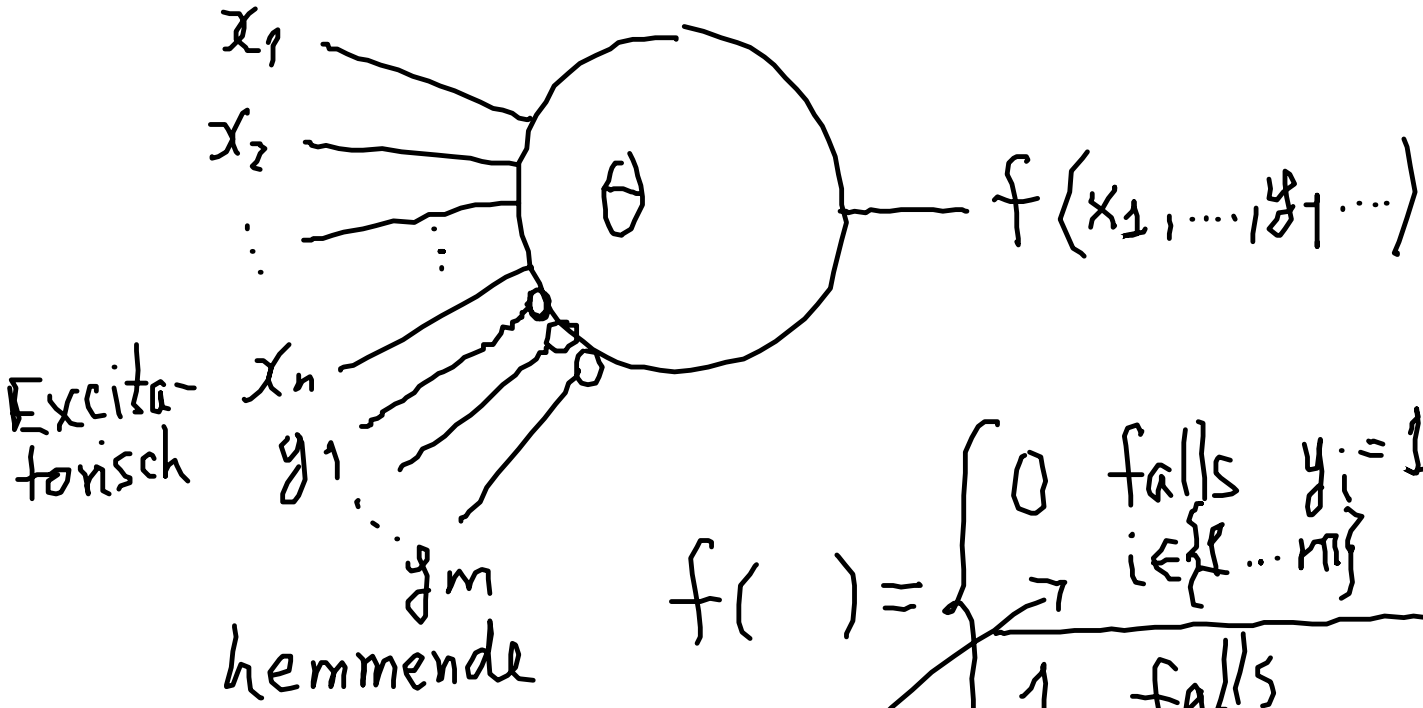
Schwellenwertlogik
Threshold logic





Domino-Logik

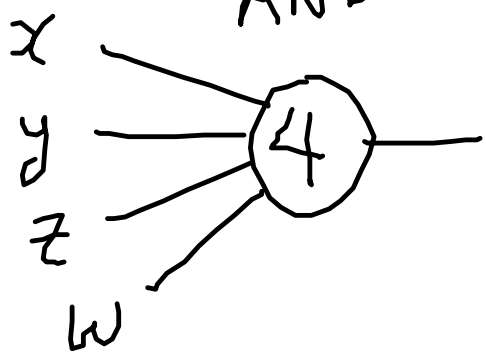
McCulloch/Pitts - Elemente



$$f(\ ) = \begin{cases} 0 & \text{falls } y_i = 1 \quad i \in \{1 \dots m\} \\ 1 & \text{falls } y_1 = y_2 = \dots = 0 \text{ und } \sum_{i=1}^{n} x_i \geq \theta \end{cases}$$

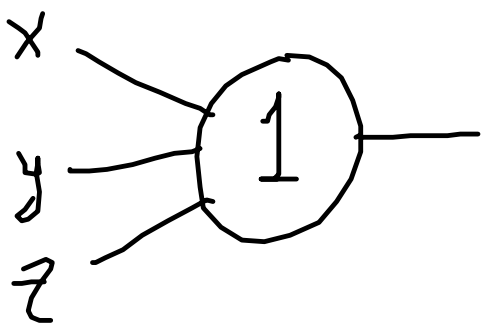
$*$ oder $\sum_{i=1}^{n} x_i < \theta$

Bsp: AND



$$\begin{cases} 1 & x+y+z+w \geq 4 \\ 0 & x+y+z+w < 4 \end{cases}$$

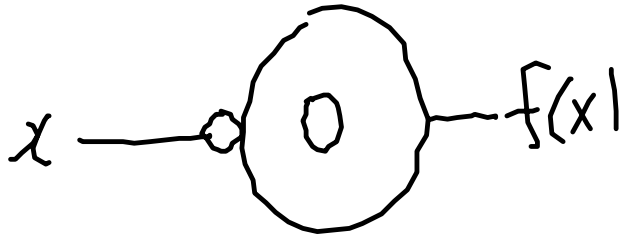
OR



$$\begin{cases} 1 & x+y+z \geq 1 \\ 0 & x+y+z < 1 \end{cases}$$

- Threshold Gatter sind allgemeiner als die klassische Gatter

NOT



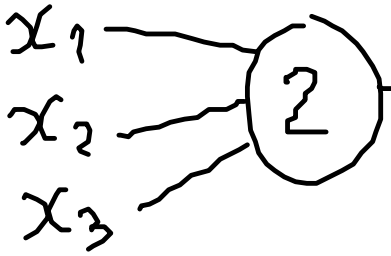
$$f(x) = \begin{cases} 0 & x = 1 \\ 1 & \text{Erregung} = 0 \\ & \geq 0 \end{cases}$$

Kybernetik

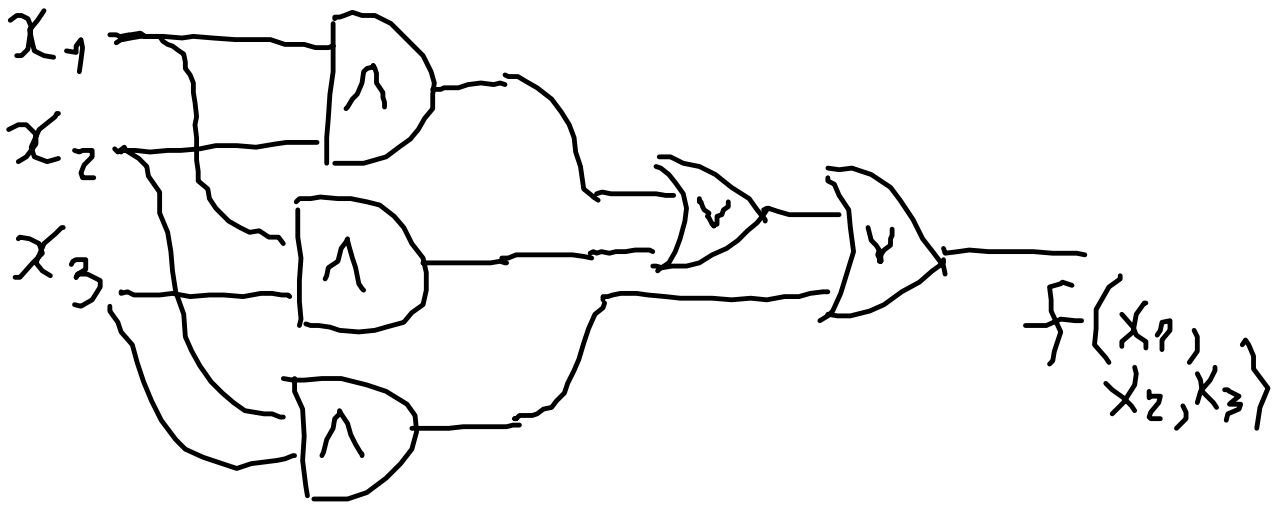
← Wiener

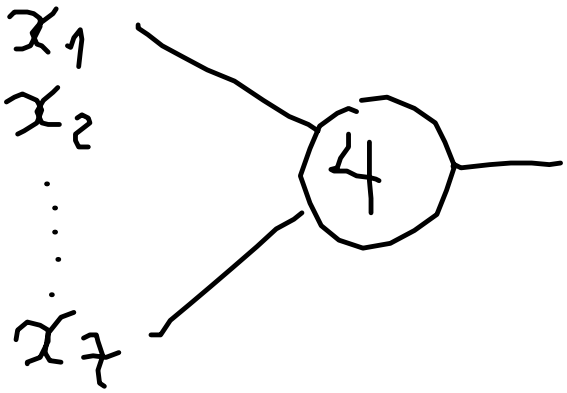
≈ 1945

Mehrheit-Berechung



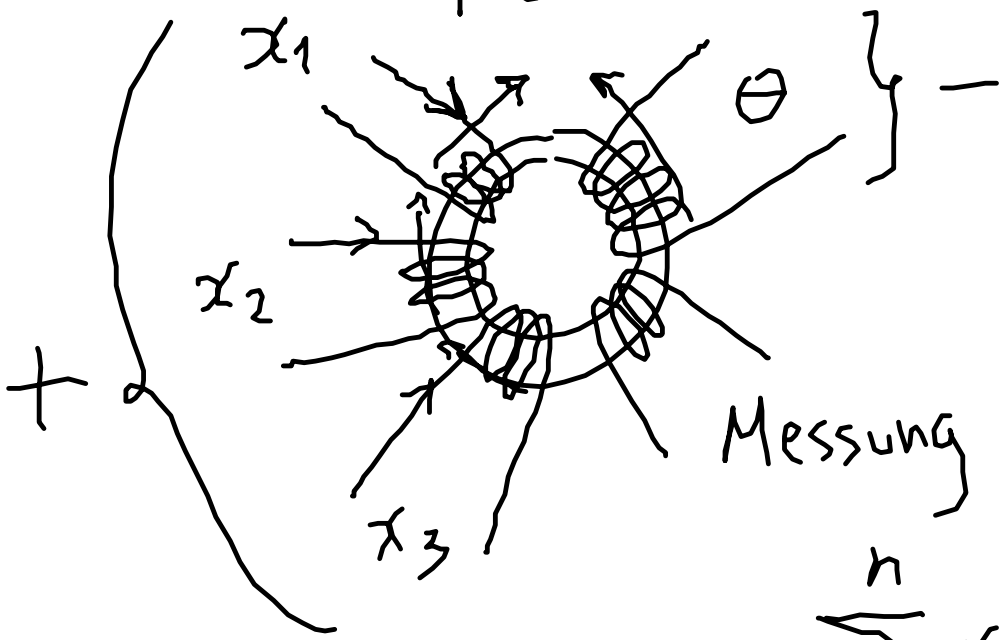
$$f(x_1, x_2, x_3) = \begin{cases} 1 & \text{wenn Mehrheit für 1} \\ 0 & \text{sonst} \end{cases}$$


x_1
x_2
x_3
$\wedge$
$\wedge$
$\wedge$
$\vee$
$\vee$
$f(x_1, x_2, x_3)$

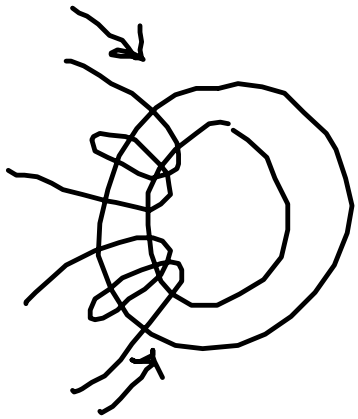

x_1
x_2
x_7
4

1950er

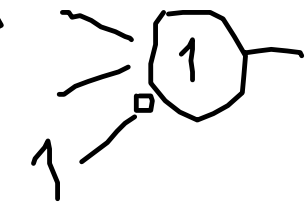
Parameter



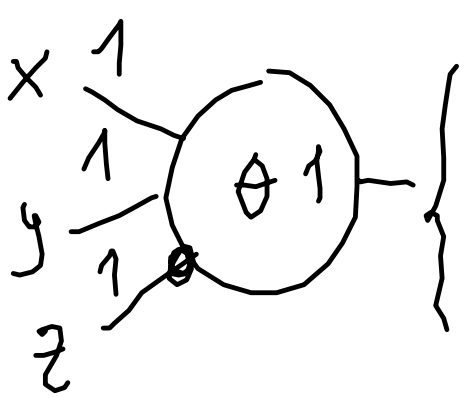
$$\sum^{n} x_i - \theta \geq 0$$



Absolute Hemmung



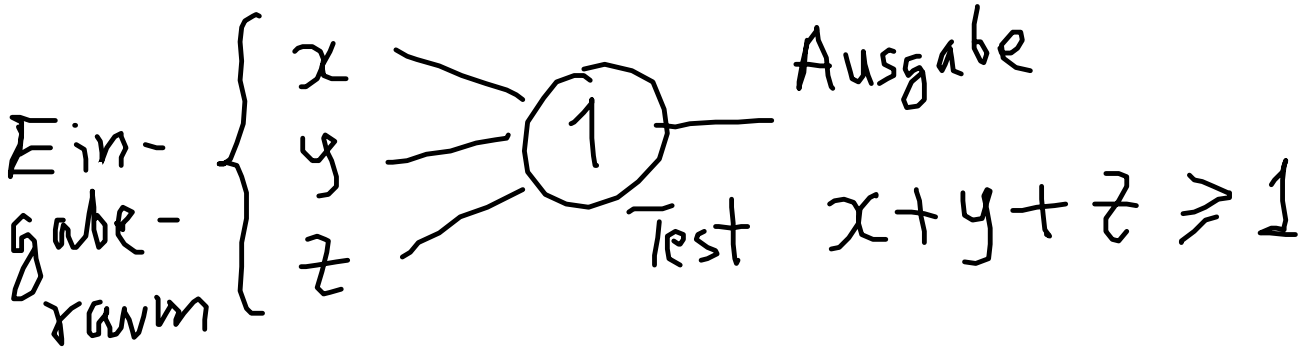
Relativ

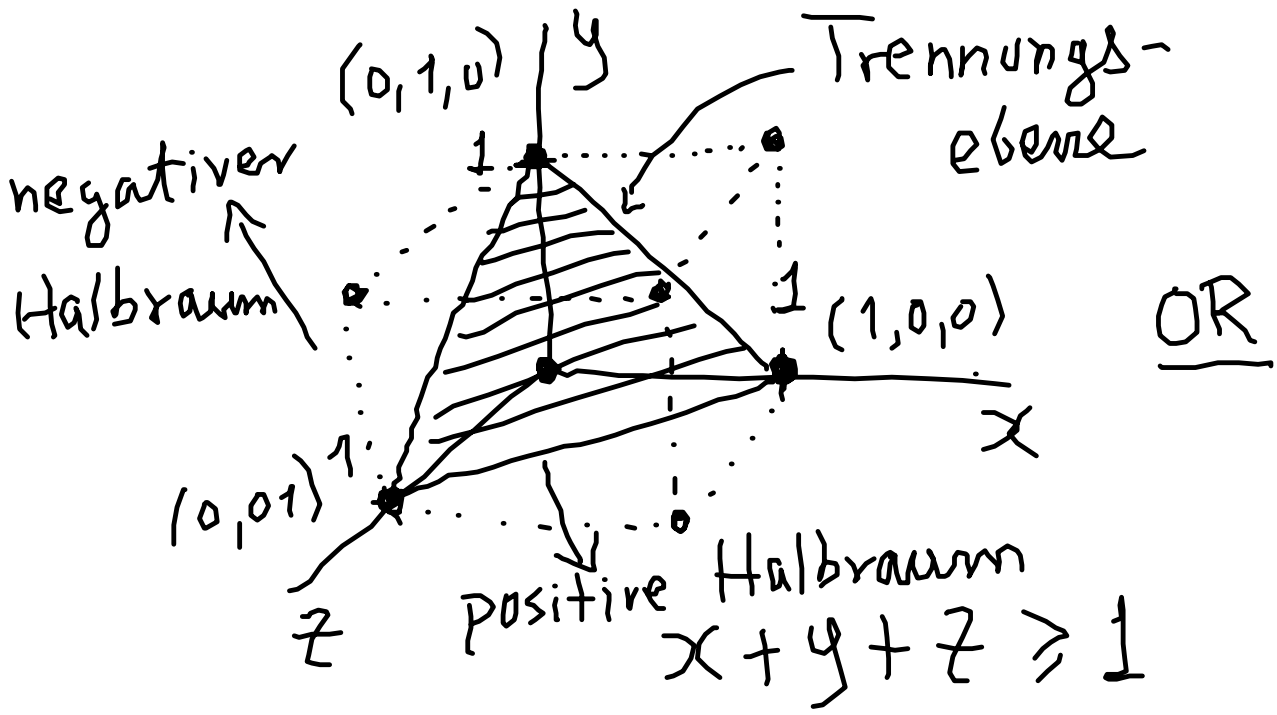


Test

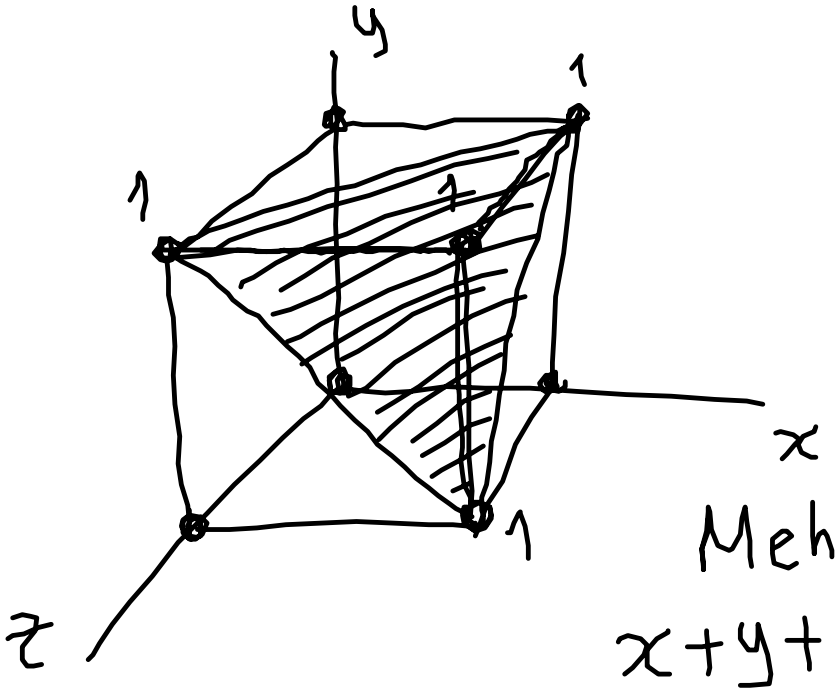
$x + y - z \geq 1$

Visualisierung





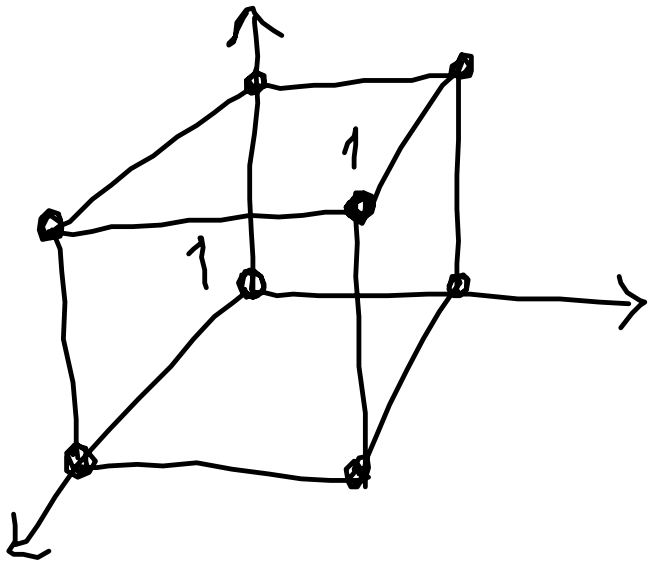
Ebene $x+y+z=1$



Mehrheitsfunktion

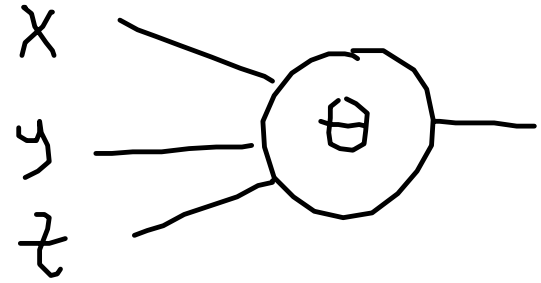
$x+y+z \geq 2$

Ebene $x+y+z=2$



f			
x	y	z	
0	0	0	1
⋮			0
			0
			⋮
			0
1	1	1	1

Ist $f \in$ T.L.



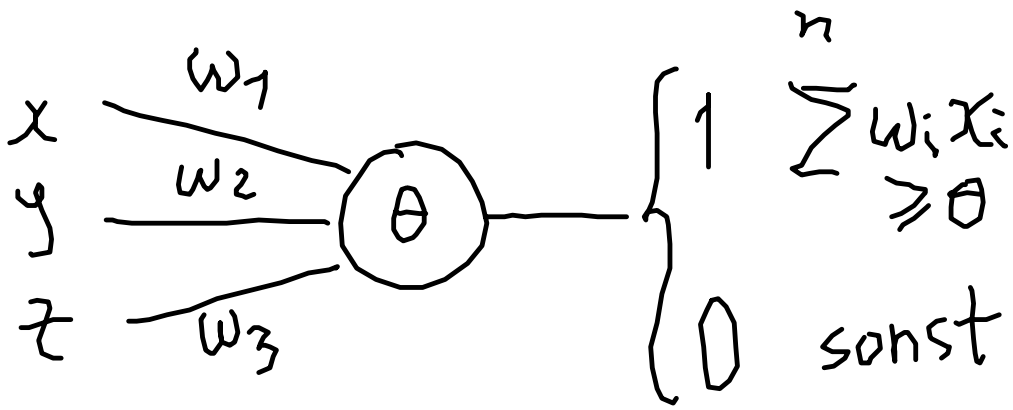
- Nicht alle log. Funktionen sind linear trennbar
- "<u>Fast alle</u>" log. Funktionen sind nicht linear trennbar

$$n \to \infty$$

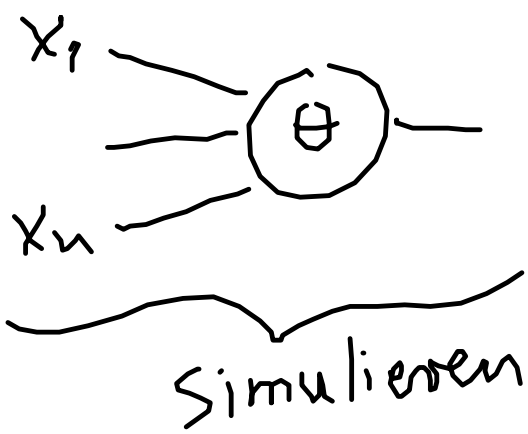
$$\frac{\#\ \text{Linear Trennbar}}{\text{Alle log. Funktionen}} \to 0$$

1. Verallgemeinerung

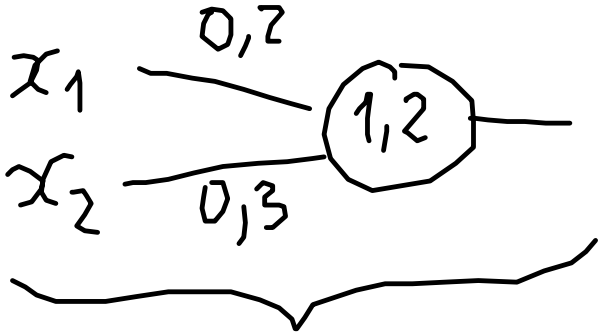
Gewichte



Keine Gewichte



mit Gewichte

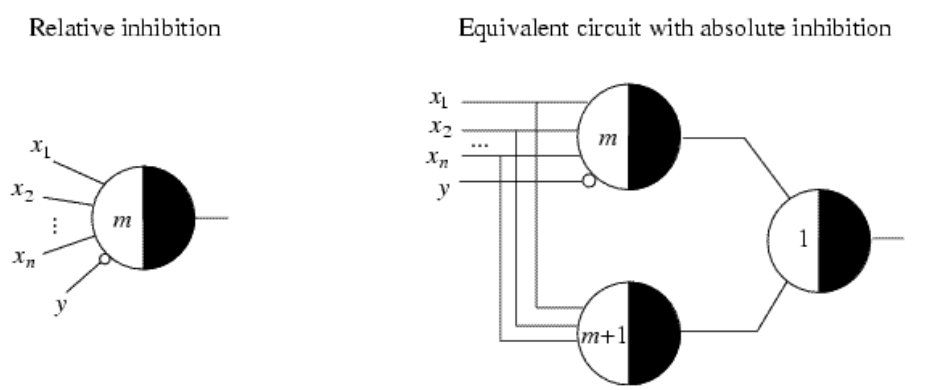


$0{,}2\,x_1 + 0{,}3\,x_2 \geq 1{,}2$

$2x_1 + 3x_2 \geq 12$

Äquivalent

$x_1 = 0{,}3$

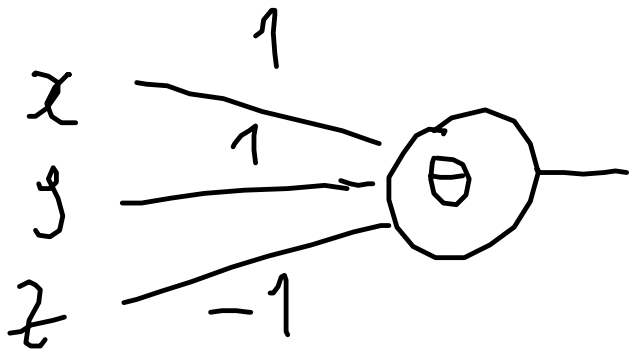


relativ

Absolut

1. Fall $y=1$	
$\underbrace{\sum^{n} x_i - 1 \geq m}$	$\underbrace{\sum^{n} x_i \geq m+1}$
2. Fall $y=0$	
$\underbrace{\sum^{n} x_i \geq m}_{\text{wahr}}$ falsch	$\sum^{n} x_i \geq m$ oder $\underbrace{\sum^{n} x_i \geq m+1}_{\text{wahr}}$ falsch

$\Rightarrow$ ich brauche kein Vorzeichen



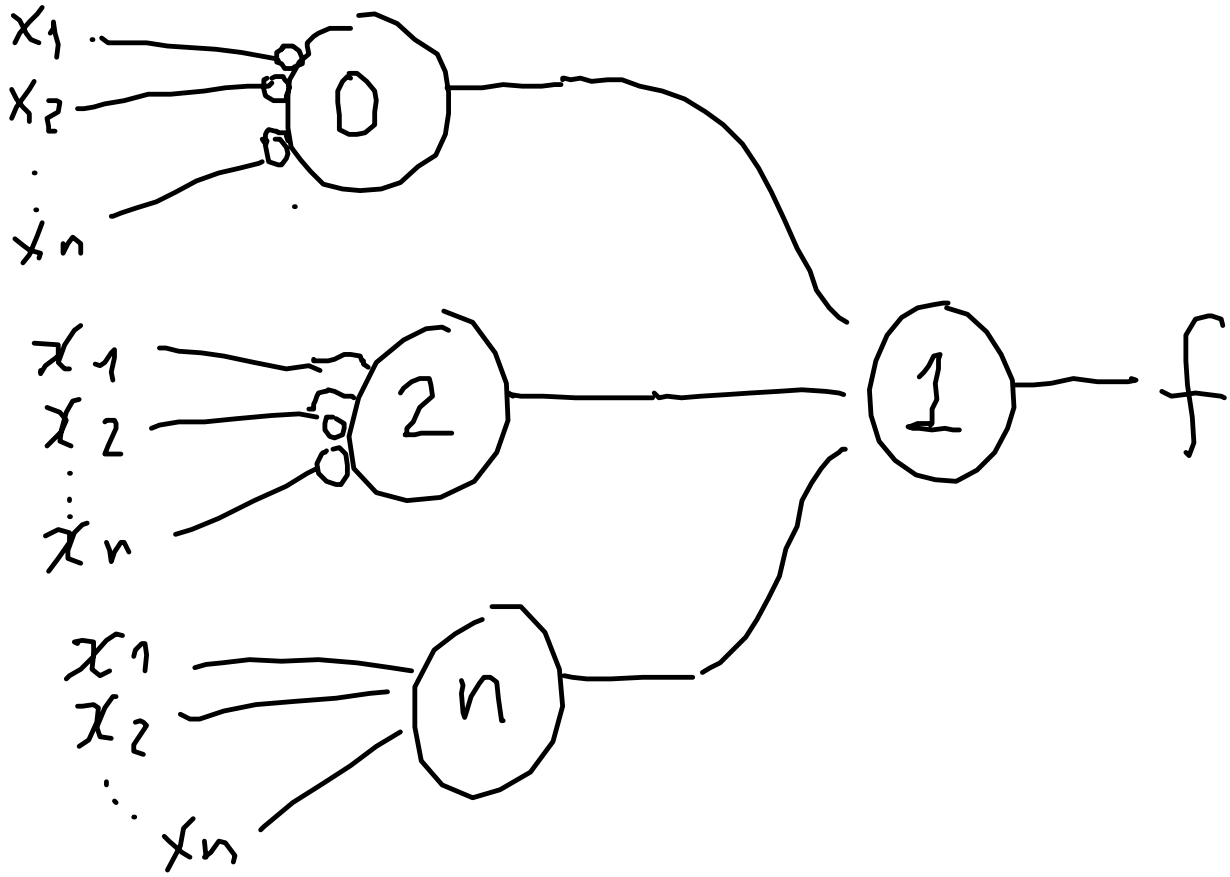
Wir können beliebige

- Funktionen implementieren
- Automaten "
  (Speicher)

Bsp.

x_1	x_2	....	x_n	f
0	0	.....	0	1
1	1	0....	0	1
0	....			0
				0
⋮				0
				⋮
0	.....			0
1	1	1....	1	1

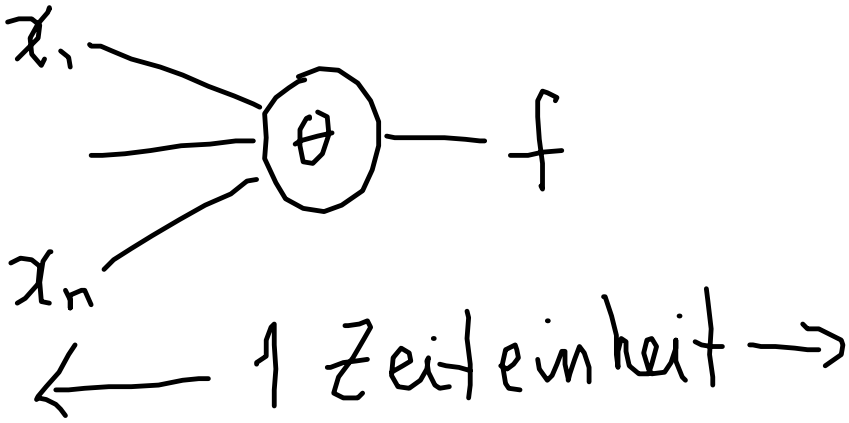
Decoder



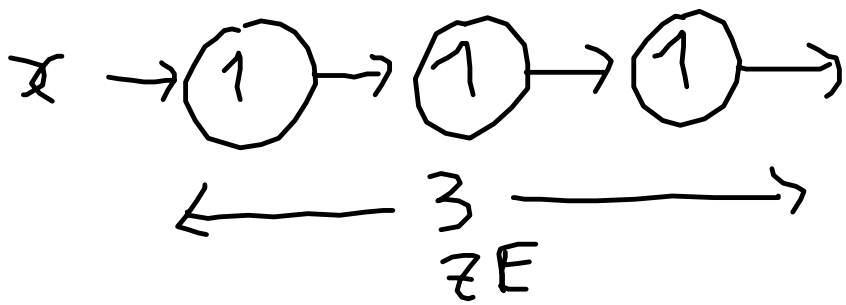
Logik-Synthese

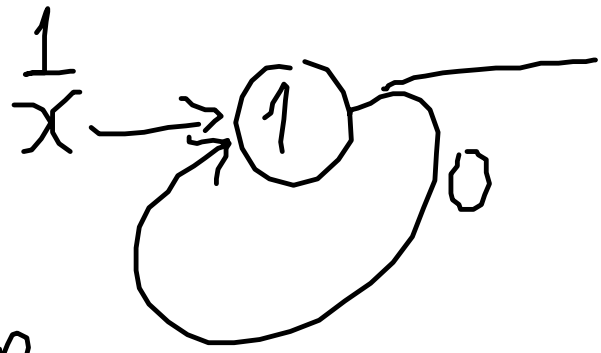
<u>Automaten</u>

Zeit

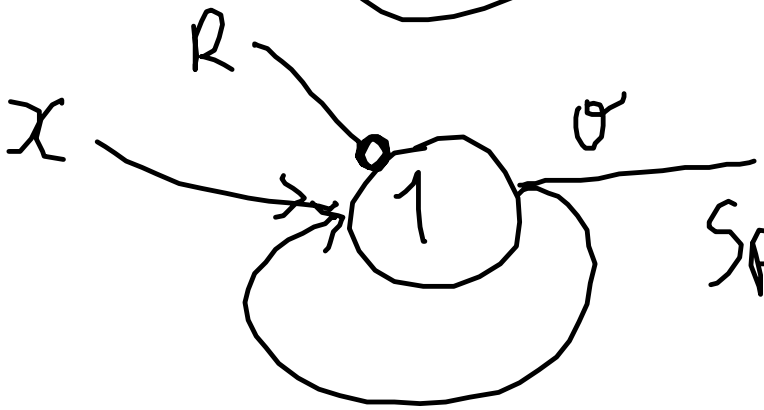


Rekursive Netze

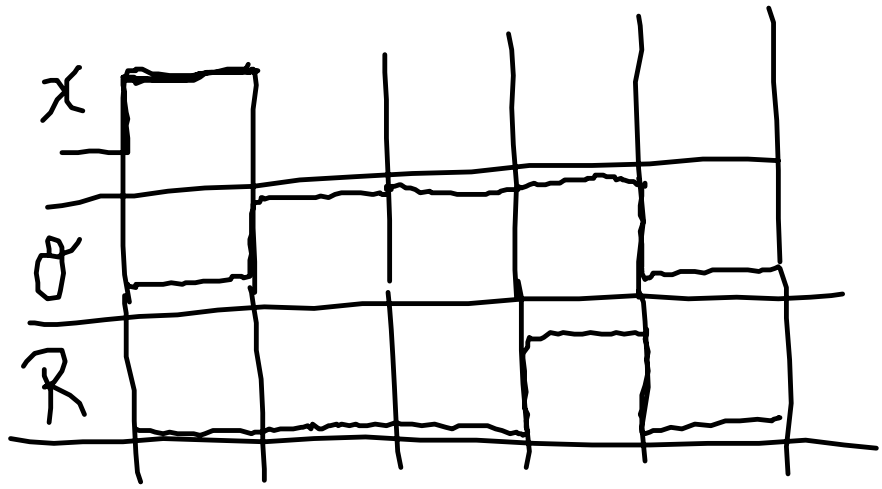


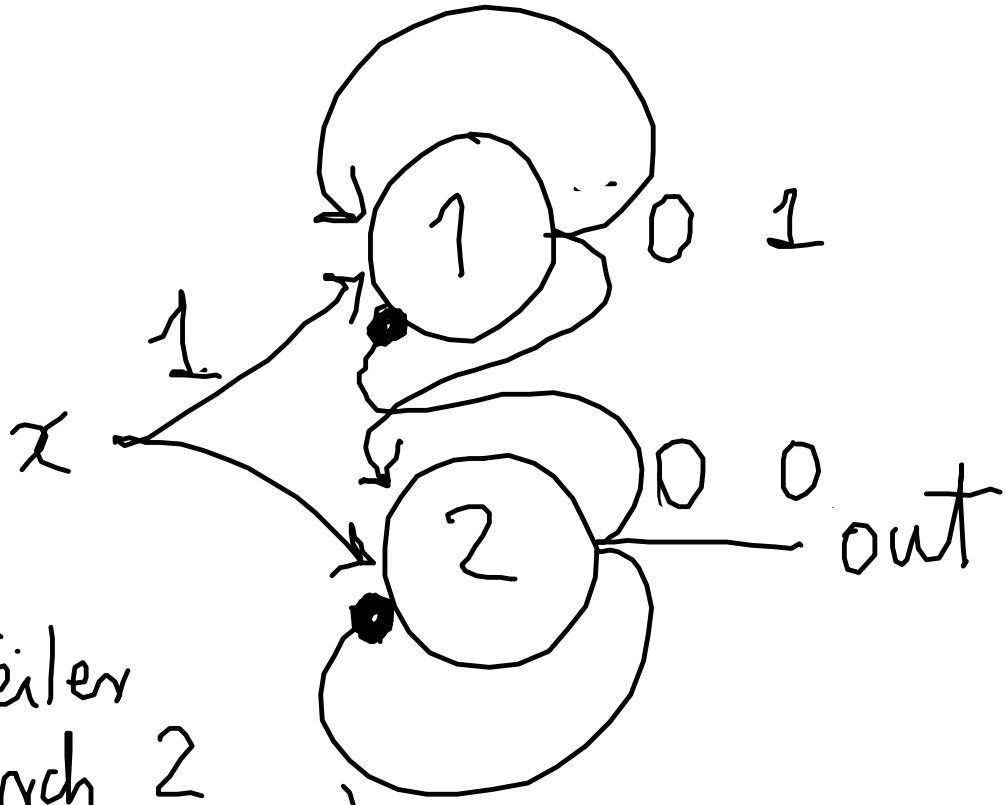


1-Bit Speicher



Speicher = Rekursion





Teiler
durch 2
(mit Fehler)

1 1 0 1 1 0 1 1 0
0 1 0 0 1 0 0 1

0 1 0 0
1 1 1 1 1 0 1 0 0 0 0 1

Automat

State transitions

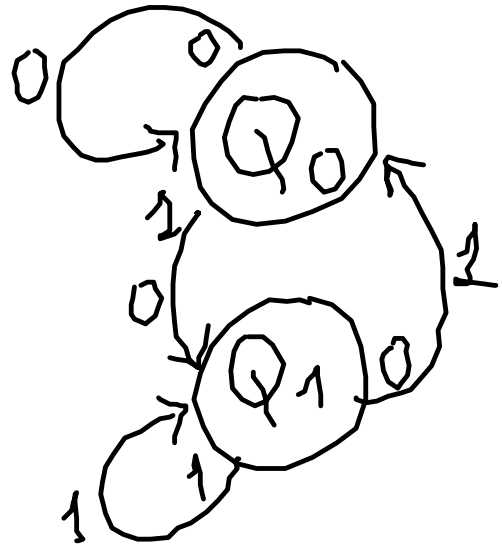
input	State Q_0	State Q_1
0	Q_0	Q_0
1	Q_1	Q_1

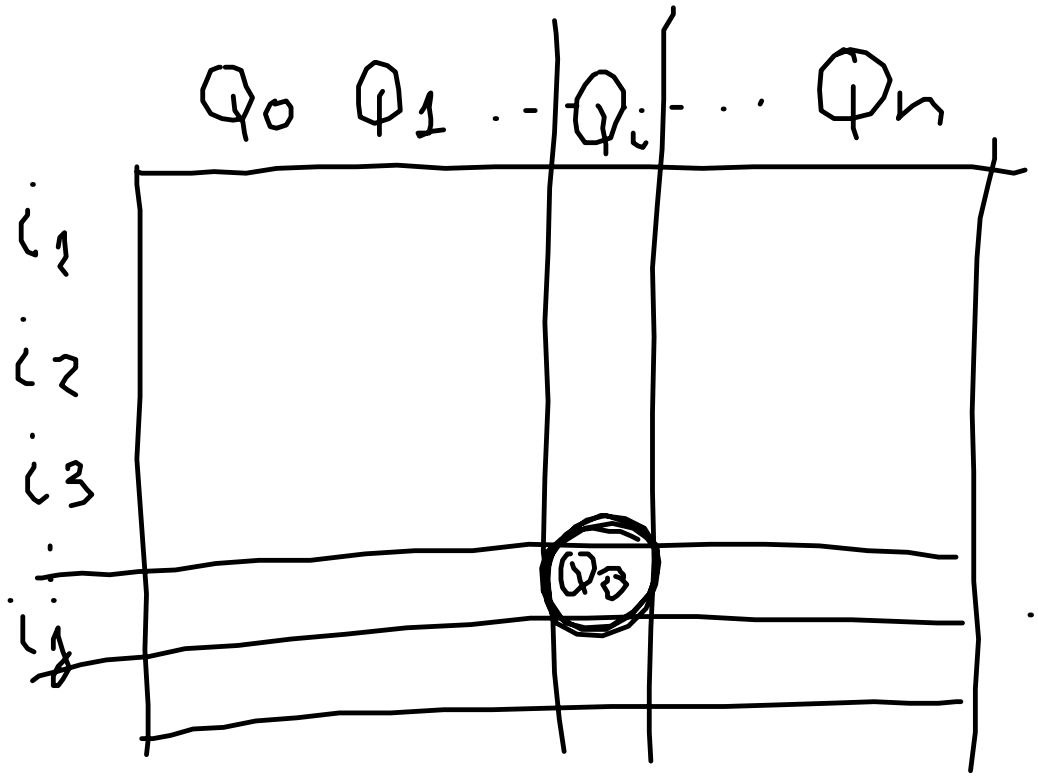
Output table

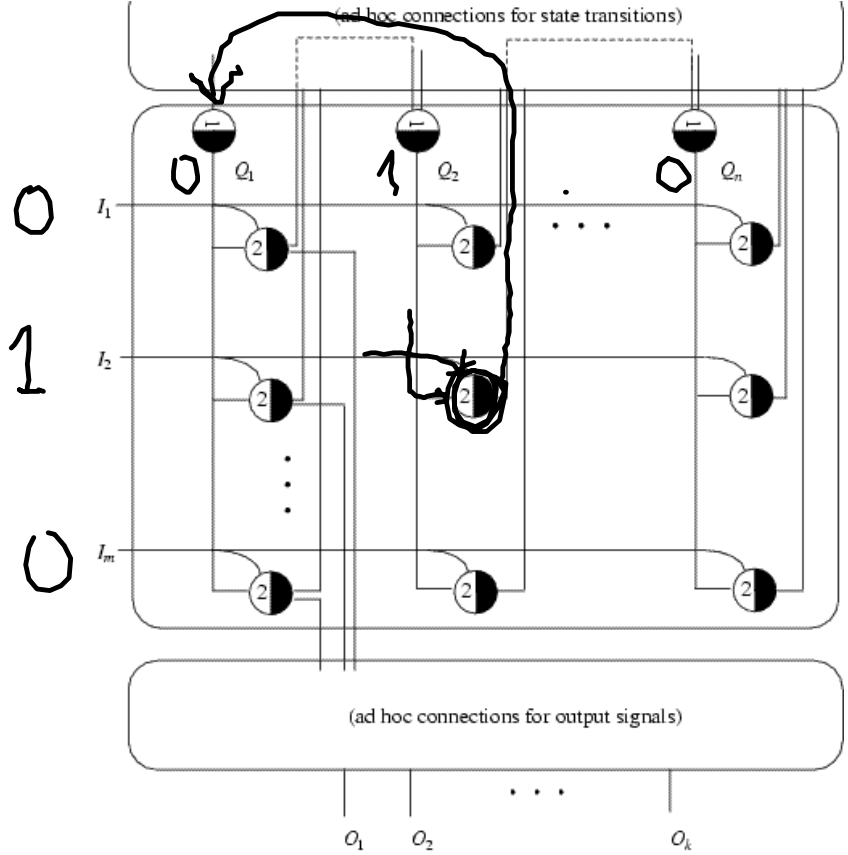
input	State Q_0	State Q_1
0	0	1
1	0	1

Q_0

Q_1






(ad hoc connections for state transitions)
Q_1
Q_2
Q_n
I_1
I_2
I_m
(ad hoc connections for output signals)
O_1
O_2
O_k