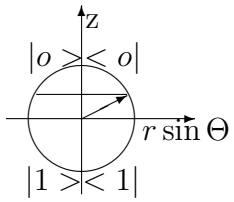
Quantum Computation:
Zusammenfassung der 2. Vorlesung (29.94.2011)

1.1 *Klassische und Quanten- Information (Fortsetzung)*

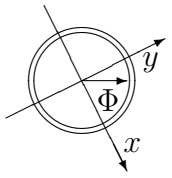
Der im letzten Paragraphen geschilderte Sachverhalt soll noch etwas genauer betrachtet werden. Zum Zustand

$$\rho = \frac{1}{2}(\mathbf{1} + \vec{r} \cdot \vec{\sigma})$$

gehört der Blochvektor $\vec{r}$, dessen Norm für reine Zustände gleich 1 und für



Schnitt in der (z,r)-Ebene



Draufsicht

Blochkugel mit Blochvektor der Norm 1

statistische Mischungen kleiner als 1 ist.
Wir berechnen die Wahrscheinlichkeit für das Messergebnis 1 bei Messung von

$$A = |1><1| \cong \begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix} = \frac{1}{2}(\mathbf{1} - \sigma_3)$$

im Zustand ρ. Es ist

$$\rho A = \frac{1}{4}(\mathbf{1} - \sigma_3 + \vec{r} \cdot \vec{\sigma} - (\vec{r} \cdot \vec{\sigma})\sigma_3)$$

und mit $\sigma_1\sigma_2 = i\sigma_3$, $\sigma_2\sigma_3 = i\sigma_1$ und $\sigma_3\sigma_3 = \mathbf{1}$ ist

$$\rho A = \frac{1}{4}(\mathbf{1} - \sigma_3 + \vec{r} \cdot \vec{\sigma} - i(r_1\sigma_3 + r_2\sigma_1) - r_3)\mathbf{1}.$$

Die Wahrscheinlichkei für das Messergebnis 1 ist also auch für statistische Mischungen

$$p = \mathrm{tr}(\rho A) = \frac{1}{4}(1 - r_3)\mathrm{tr}\mathbf{1} = \frac{1}{2}(1 - r_3) = \frac{1}{2}(1 - \cos\Theta).$$

Alle Zustände ρ, deren Blochvektoren die gleiche 3-Komponente haben, also auf eine zur z-Achse senkrechte Ebene weisen, ergeben die gleiche Wahrscheinlichkeit p. Dass der analoge geometrische Sachverhalt auch bei Messung von $|\varphi><\varphi| = \frac{1}{2}(\mathbf{1} + \vec{e}(\Theta, \Phi) \cdot \vec{\sigma})$ vorliegt, kann man auch direkt nachrechnen, wenn man die Formel

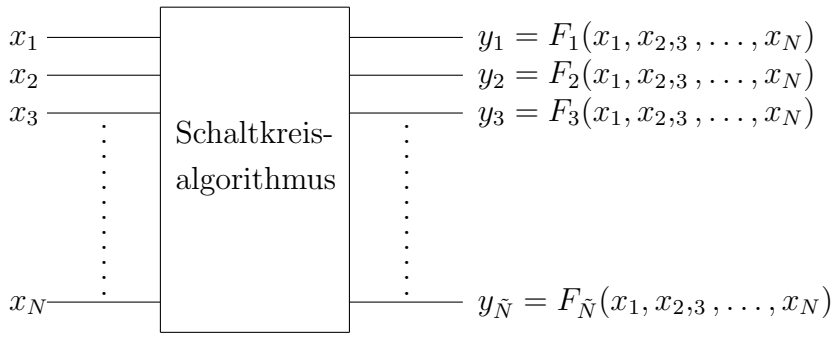
$$(\vec{x} \cdot \vec{\sigma})(\vec{y} \cdot \vec{\sigma}) = (\vec{x} \cdot \vec{y})\mathbf{1} + i(\vec{x} \times \vec{y}) \cdot \vec{\sigma}$$

verwendet.

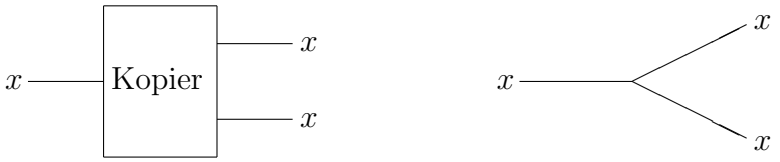
1.2 *Das Schaltkreismodell, logische und unitäre Gatter*

1.2.1 *Klassische Rechner, logische Gatter*

Ziel eines Rechners ist es, aus einem Wort, $x_1 x_2 \ldots x_N \in \{0,1\}^N$, nach einem Algorithmus ein anderes Wort, $y_1 y_2 \ldots y_{\tilde{N}} \in \{0,1\}^{\tilde{N}}$, zu erschließen. Dies kann, wie im Folgenden erläutert werden wird, mit einem Schaltkreis aus logischen Gattern erreicht werden.

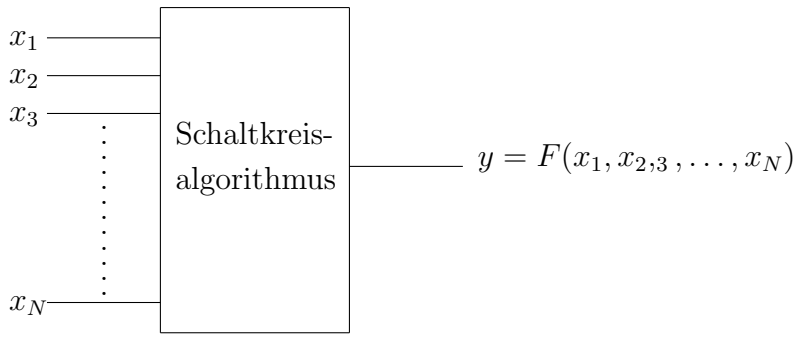


Nun ist es möglich, auch unbekannte klassische Information zu kopieren, denn es gibt also Algorithnen der Art:



Auf der rechten Seite der Skizze befindet sich das Gattersymbol für die

Verwendung in Schaltkreisen. Unbekannte Quantenzustände können nicht vervielfältigt werden, deshalb fehlt dieses Element in Quantenschaltkreisen. Im klassischen Fall kann man aufgrund der Kopiermöglichkeit den Rechner aus den einfacheren Schaltkreisen



zusammensetzen. Die Funktionen $F : \{0,1\}^N \to \{0,1\}$ sind als N-adische Wahrheitswertfunktoren in der klassischen Aussagenlogik bekannt und werden dort eingehend untersucht. Ihre Darstellung durch elementare, d.h. monadische und dyadische, Funktoren sind die Gesetze der Aussagenlogik, die unserem Denken angepasst sind. Stellt man in einer "Wahrheitswerttabelle" den 2^N Wörtern, die man mit N Bits bilden kann, die Werte eines Funktors gegenüber, dann sieht man. dass es soviêle N-adische Wahrheitswertfunktoren gibt, wie man Wörter aus 2^N Bits bilden kann, nämlich 2^{2^N}. Von den 4 monadischen Funktoren

Antilogie

x	y
0	0
1	0

Identität

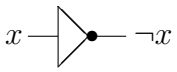
x	y
0	0
1	1

Negation

x	y
0	1
1	0

Tautologie

x	y
0	1
1	1

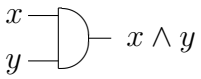


NON

benötigen wir nur die Negation, deren Gattersymbol eingezeichnet ist. Von den 16 dyadischen Wahrheitswertfunktoren betrachten wir zunächsst nur die Konjunktion und die Disjunktion:

Konjunktion

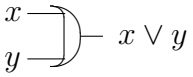
x	y	z
0	0	0
0	1	0
1	0	0
1	1	1



AND

Disjunktion

x	y	z
0	0	0
0	1	1
1	0	1
1	1	1

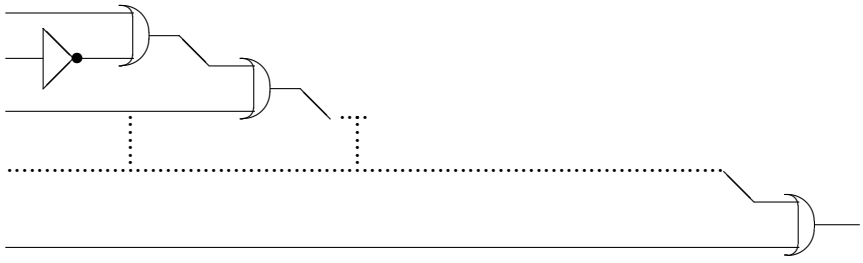


OR

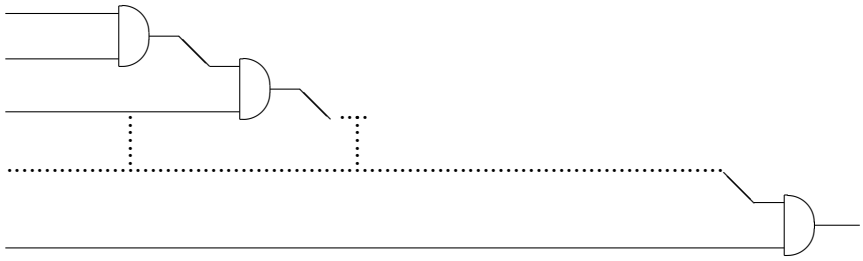
In der Aussagenlogik beweist man, dass jeder Wahrheitswertfunktor in der konjunktiven Nornalform

$$F(x_1, x_{2,3}, \ldots, x_N) = \cdots \wedge (x_k \vee \neg x_l \vee \ldots) \wedge (x_{k'} \vee x_{l'} \ldots) \wedge \ldots$$

geschrieben werden kann. Dies bedeutet, dass jede Boolesche Funktion in einem Schaltkreis implement weden kann, in der nur die Gatter *NON, AND* und *OR* verschaltet sind. Der Schaltkreis für jede Klammer ist von der Gestalt:



Diese Schaltkreise werden schließlich mit



verbunden und damit in Konjunktion gestellt. Ebenso leicht kann man die disjunktive Normalform

$$F(x_1, x_{2,3}, \ldots, x_N) = \cdots \vee (x_k \wedge \neg x_l \wedge \ldots) \vee (x_{k'} \wedge x_{l'} \ldots) \vee \ldots$$

eines N-adischen Wahrheitswertfunktors in einem Schaltkreis implementieren.

Da die Normalformrnen für die Implementierung von Algorithmrn in Schalkreisen zentrale Bedeutung haben, sei zur Illustration ein induktiver Beweis für deren Existenz gegeben: Aus der folgenden Tabelle kann man die Wahrheutswerte der 16 binären Funktoren entnehmen. Etwa in folgender Reihenfolge.

8 binäre Funktoren:

$x\,y$	$\top$	$\vee$	$\Leftarrow$	$\Rightarrow$	$\mid$	$\Leftrightarrow$	$\barwedge$	$\barvee$
0 0	1	0	1	1	1	1	1	1
0 1	1	1	0	1	1	0	1	0
1 0	1	1	1	0	1	0	0	1
1 1	1	1	1	1	0	1	0	0

und deren 8 Negationen:

$x\,y$	$\bot$	$\dagger$	$\prec$	$\succ$	$\wedge$	$\succcurlyeq$	$\barwedge$	$\barvee$
0 0	0	1	0	0	0	0	0	0
0 1	0	0	1	0	0	1	0	1
1 0	0	0	0	1	0	1	1	0
1 1	0	0	0	0	1	0	1	1

Anhand dieser Tabellen und der der Negation prüft man leicht die folgenden Gesetze nach:

$$\begin{aligned}
x \top y &\;.\equiv.\; (x \vee \neg x) \wedge (y \vee \neg y), \\
&\qquad\quad x \vee y \\
x \Leftarrow y &\;.\equiv.\; x \vee \neg y, \\
x \Rightarrow y &\;.\equiv.\; \neg x \vee y, \\
x \mid y &\;.\equiv.\; \neg x \vee \neg y, \\
x \Leftrightarrow y &\;.\equiv.\; (x \wedge y) \vee (\neg x \wedge \neg y), \\
x \barwedge y &\;.\equiv.\; \neg x \wedge (y \vee \neg y), \\
x \barvee y &\;.\equiv.\; (x \vee \neg x) \wedge \neg y
\end{aligned}$$

Die rechten Seiten der Gesetze stellen die sogenannten “reduzierten” Darstellungen der Funktoren dar, in der nur Konjuntionen und Disjunktionen von “Elementarsussagen” wie x, y und deren Negationen wie $\neg x$, $\neg y$ vorkommen.

In der ersten, siebten und achten Zeile ist die reduzierte Darstellung die konjunktive Normalform, in der sechsten Zeile die disjunktive Normalform und in den übrigen Zeilen kann die Darstellung sowohl als die disjunktive als auch die konjunktive (in der keine Konjunktion auftritt) angesehen werden. Konjunktive Normalformen lassen sich mit den Distributivegesetzen,

$$\begin{aligned} x \wedge (y \vee z) \quad &. \equiv . \quad (x \wedge y) \vee (x \wedge z), \\ x \vee (y \wedge z) \quad &. \equiv . \quad (x \vee y) \wedge (x \vee z), \end{aligned}$$

stets in Distributive Normalformen umwandeln und umgekehrt. Die Negationen der genannten 8 binären Wahrheitswertfunktoren, deren Wahrheiswerte in der rechten Tabelle stehen, lassen sich mit Hilfe der de Morganschen Gesetze,

$$\begin{aligned} \neg(x \vee y). &\equiv .\neg x \wedge \neg y, & \neg(x \wedge y). &\equiv .\neg x \vee \neg y, \\ x \vee y. &\equiv .\neg(\neg x \wedge \neg y), & x \wedge y. &\equiv .\neg(\neg x \vee \neg y), \end{aligned}$$

und ggf. der Distributivgesetze in eine reduzierte Darstellung und eine der Normalformen umwandeln. Dies gilt für alle binären Wahrheitswertfunktoren $F(x, y)$ und stellt den Induktionsanfang dar. Wir nehme nun an , dass dies auch für N-adische Wahrheitswertunktoren $F(x_1, x_2, x_3, \ldots, x_N)$ gilt. Ist nun ein $(N+1)$-adische Wahrheitswertfunktor $G(x_1, x_2, x_3, \ldots, x_N, x_{N+1})$ gegeben, dann sind $G(x_1, x_2, x_3, \ldots, x_N, 1)$ und $G(x_1, x_2, x_3, \ldots, x_N, 0)$ N-adische Wahrheitswertfuntoren, die aufgrund der Induktionsannahme in reduzierter Form dargestellt werden können. Offenbar ist

$$\begin{aligned} &G(x_1, x_2, x_3, \ldots, x_N, x_{N+1}) \quad . \equiv . \\ &(G(x_1, x_2, x_3, \ldots, x_N, 1) \wedge x_{N+1}) \succ\!\prec (G(x_1, x_2, x_3, \ldots, x_N, 0) \wedge \neg x_{N+1}), \end{aligned}$$

denn für $x_{N+1} = 1$ hat die zweite der in Kontavalenz ($\succ\!\prec$) stehenden Aussagen den Wahrheitswert 0 und die erste den Wahrheitsert 1, falls $G(x_1, x_2, x_3, \ldots, x_N, 1) = 1$ ist, und den Wert 0, falls $G(x_1, x_2, x_3, \ldots, x_N, 1) = 0$ ist. Für $x_{N+1} = 0$ ist der Wert ser ersten in Konmtravalenz stehenden Aussagen 0 und die zweite hat den Wert 1, falls $G(x_1, x_2, x_3, \ldots, x_N, 0) = 1$ ist, und den Wert 0, falls $G(x_1, x_2, x_3, \ldots, x_N, 0) = 0$ ist. Damit stimmen die Wahrheitswerte der in Äquivalenz gesetzten Funktoren überein. Man muss nur noch eine reduzierte Darstellung für die Kontravalenz, $\succ\!\prec$, verwenden, um eine reduzierte Darstellung für den $(N+1)$-adische Wahrheitswertfunktor

$G(x_1, x_2, x_3, \ldots, x_N, x_{N+1})$ zu finden. Damit ist durch vollstäbdige Induktion gezeigt:

Satz: Jeder Wahrheitswertfunktor kann in einer reduzierten Darstellung ausgedrückt werden und mithin auch in konjunktiver und disjunktiver Normalform.

Für das Rechnen mit Wahrheitswertfunktoren ist es besonders bequem, wenn man diese mit den Kompositionen des Booleschen Ringes mit zwei Elementen darstellt. Die Kompositionstafeln dieses kommutativen Ringes sind:

$\oplus$	0	1
0	0	1
1	1	0

$\cdot$	0	1
0	0	0
1	0	1

Die Negation und die Konjunktion sind durch

$$\neg x \cong 1 \oplus x \qquad \text{und} \qquad x \wedge y \cong x \cdot y$$

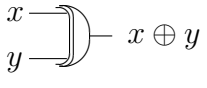
dargestellt. Die Darstellung der ´Disjunktion liegt damit bach dem dritten de Morganschen Gesetz fest:

$$\begin{aligned} x \vee y = \neg(\neg x \wedge \neg y) &\cong 1 \oplus (1 \oplus x) \cdot (1 \oplus y) \\ &= 1 \oplus 1 \oplus x \oplus y \oplus x \cdot y = x \oplus y \oplus x \cdot y \end{aligned}$$

Die Boolesche Addition ist Kongruent zur Kontravalenz $x \succ\!\prec y$ der Aussagenlogik:

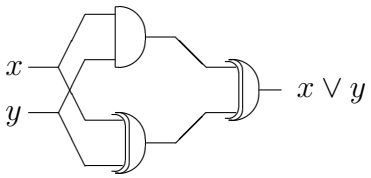
Kontravalenz

x	y	z
0	0	0
0	1	1
1	0	1
1	1	0



XOR

Disjunktion und Kontravalenz



OR

Eine Menge von Wahrheitswertfunktoren oder Gattern heißt universell, wenn ihre Elemente ausreichen, um alle Funktoren daraus zu erzeugen. Mit den Normalformen haben wir schon gesehen, dass $\{\neg, \wedge, \vee\}$ eine universelle Menge ist. Das dritte de Morgansche Gesetz zeigt jedoch, dass schon $\{\neg, \wedge\}$ universell ist. Das vierte de Morgansche Gesetz, $x \wedge y = \neg(\neg x \vee \neg y)$, das man im Booleschen Ring leicht beweist,

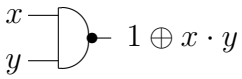
$$\begin{aligned} 1 \oplus (1 \oplus x) \oplus (1 \oplus y) \oplus (1 \oplus x) \cdot (1 \oplus y) &= 1 \oplus x \oplus y \oplus 1 \oplus x \oplus y \oplus x \cdot y \\ &= x \cdot y, \end{aligned}$$

zeigt, dass such $\{\neg, \vee\}$ universell ist. Dagegen ist $\{\neg, \nsucc\}$ nicht universell, weil man die Boolesche Multiplikation, $\wedge$, nicht erzeugen kann.

Gebräuchliche Wahrheitswertfunktoren sind auch

Exklusion

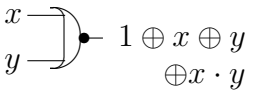
x	y	z
0	0	1
0	1	1
1	0	1
1	1	0



NAND

Rejektion

x	y	z
0	0	1
0	1	0
1	0	0
1	1	0



NOR

Scheffer publizierte 1913, dass die Exklusion eine universelle Menge darstellt, $x|y$ wird deeshalb auch der Scheffersche Strich genannt. Dass $\{|\}$ universell ist, folgt aus den Gleichungen

$$\begin{aligned} \neg x = x|x \qquad &\text{bzw.} \qquad 1 \oplus x = 1 \oplus x \cdot x \\ x \wedge y = (x|y)|(x|y) \qquad &\text{bzw.} \qquad x \cdot y = 1 \oplus (1 \oplus x \cdot y). \end{aligned}$$

Man kann die erste Zeile der Gleichungen auch direkt aus der Wahrheitswerttabelle der Exklusion ablesen, wenn man $x = y$ bedenkt. Auch mit der Rejektion, $x \dagger y$, ist $\{\dagger\}$ universell, denn es gilt

$$\begin{aligned} \neg x = x \dagger x \qquad &\text{bzw.} \qquad 1 \oplus x = 1 \oplus x \oplus x \oplus x \cdot x \\ x \vee y = (x \dagger y) \dagger (x \dagger y) \qquad &\text{bzw.} \qquad x \oplus y \oplus x \cdot y = 1 \oplus (1 \oplus x \oplus y \oplus x \cdot y). \end{aligned}$$

Als Beispiel für die Darstellung eines arithmetischen Algorithmus durch einen Schaltkreis sei die Addition von Dualzahlen betrachtet: Die Aufgabe

$$0x_1x_2\ldots x_N + 0y_1y_2\ldots y_N = s_0z_1z_2\ldots z_N$$

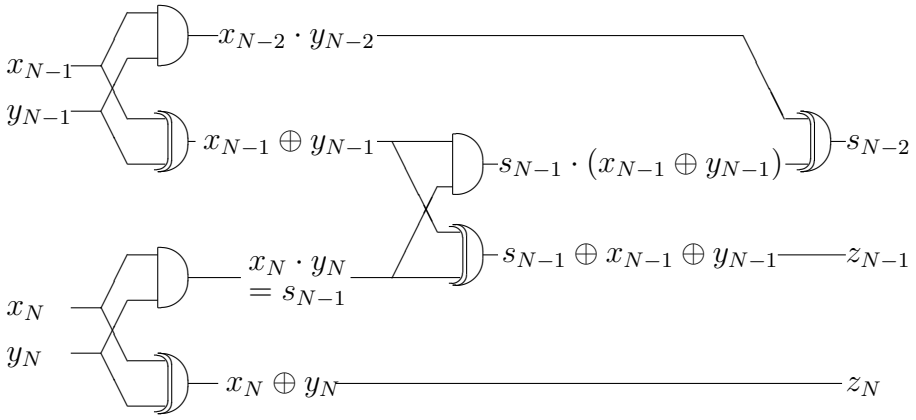
wird durch den folgenden Algorithnus gelöst, der sich aus einem ´´Halbaddierer"

$$z_N = x_N \oplus y_N, \qquad s_{N-1} = x_N \cdot y_N, \qquad \textit{(Halbaddierer)}$$

und $N-1$ "Volladdierern"

$$\begin{aligned} z_{N-n} &= s_{N-n} \oplus x_{N-n} \oplus y_{N-n}, \\ s_{N-n-1} &= s_{N-n} \cdot x_{N-n} \oplus s_{N-n} \cdot y_{N-n} \oplus x_{N-n} \cdot y_{N-n} \\ &\quad (n = 1, 2, 3, \ldots, (N-1)) \qquad \textit{(Volladdierer)} \end{aligned}$$

zusammensetzt. Der folgende Schaltkreis zeigt einen Haladdierer mit einem Volladdierer, der für $N > 2$ um weitere Volladdierer zu ergänzen ist.



Die Anzahl der verschalteten elementaren Gatter ist ein Maß für die "Komplexität". Hier werden $2 + (N-1)5$ Gatter verwendet. Der Algorithmus gehört damit zur Klasse der polynomialen Komplexität. Oft wird auch die "Dauer" des Rechenvorgangs angegeben. Diese ist im vorliegenden Fall unabhängig von N gleich drei, weil übereinander angeordnete Gatter parallel arbeiten.