

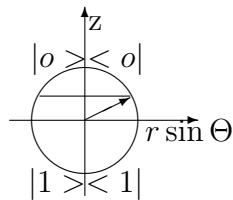
Quantum Computation: Zusammenfassung der 2. Vorlesung (24.10.08)

1.1 Klassische und Quanten- Information (Fortsetzung)

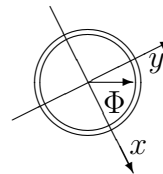
Der im letzten Paragraphen geschilderte Sachverhalt soll noch etwas genauer betrachtet werden. Zum Zustand

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

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



Schnitt in der (z,r)-Ebene



Draufsicht

Blochkugel mit Blochvektor der Norm 1

tistische Mischungen kleiner als kleiner als 1 ist.

Wir berechnen die Wahrscheinlichkeit dafür, dass bei Messung von

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

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 Wahrscheinlichkeit für das Messergebnis 1 ist also auch für statistische Mischungen

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

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

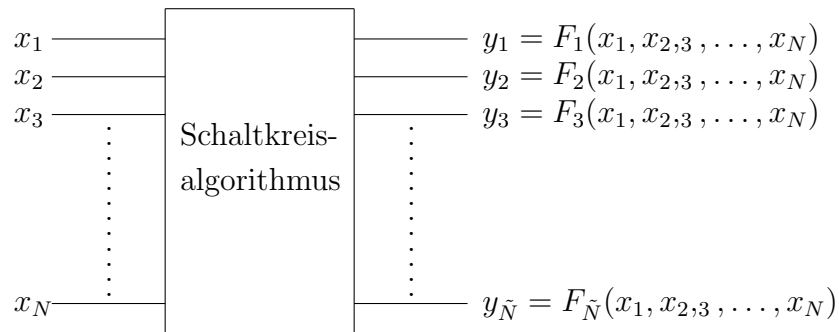
$$(\vec{x} \cdot \vec{\sigma})(\vec{y} \cdot \vec{\sigma}) = (\vec{x} \cdot \vec{y})\mathbf{1} + i(\vec{x} \times \vec{y}) \cdot \vec{\sigma}$$

verwendet.

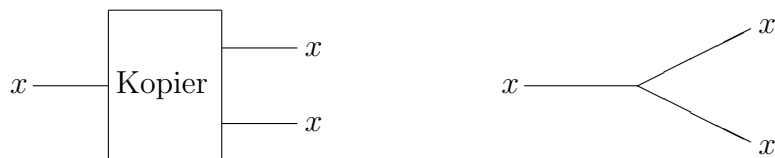
1 Das Schaltkreismodell, logische und unitäre Gatter

1.1 Klassische Rechner, logische Gatter

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

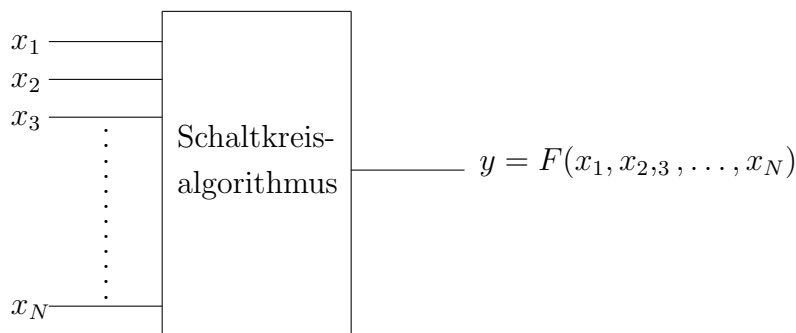


Nun ist es möglich, auch unbekannte klassische Information zu kopieren, denn es gibt also Algorithmen der Art:



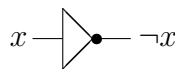
Auf der rechten Seite der Skizze befindet sich das Gattersymbol für die

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



zusammensetzen. Die Funktionen $F : \{0, 1\}^N \rightarrow \{0, 1\}$ 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 "Wahrheitstabelle" den 2^N Wörtern, die man mit N Bits bilden kann, die Werte eines Funktors gegenüber, dann sieht man, dass es sovièle N -adische Wahrheitswertfunktoren gibt, wie man Wörter aus 2^N Bits bilden kann, nämlich 2^{2^N} . Von den 4 monadischen Funktoren

<i>Antilogie</i>	<i>Identität</i>	<i>Negation</i>	<i>Tautologie</i>
$\begin{array}{c c} x & y \\ \hline 0 & 0 \\ 1 & 0 \end{array}$	$\begin{array}{c c} x & y \\ \hline 0 & 0 \\ 1 & 1 \end{array}$	$\begin{array}{c c} x & y \\ \hline 0 & 1 \\ 1 & 0 \end{array}$	$\begin{array}{c c} x & y \\ \hline 0 & 1 \\ 1 & 1 \end{array}$

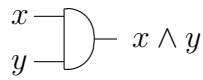


NON

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

Konjunktion

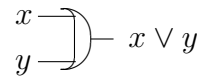
x	y	z
0	0	0
0	1	0
1	0	0
1	1	1



AND

Disjunktion

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

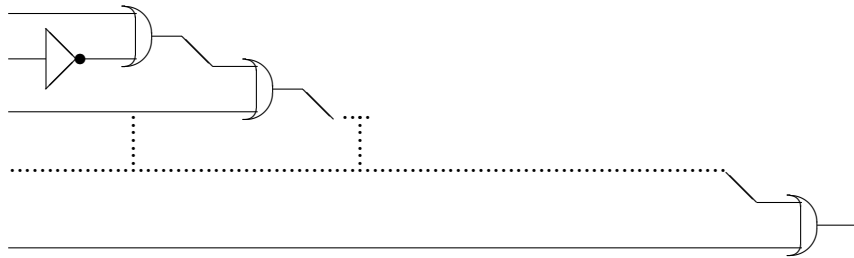


OR

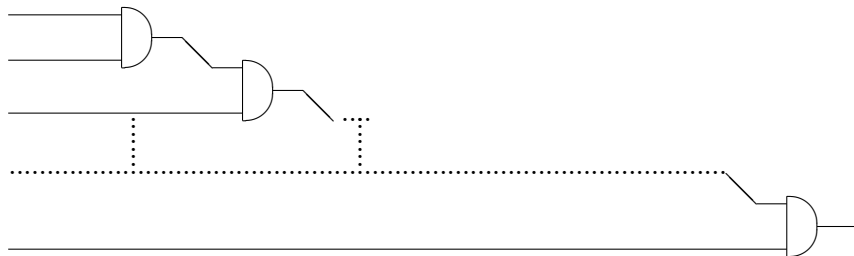
In der Aussagenlogik beweist man, dass jeder Wahrheitswertfunktore in der konjunktiven Normalform

$$F(x_1, x_2, x_3, \dots, x_N) = \dots \wedge (x_k \vee \neg x_l \vee \dots) \wedge (x_{k'} \vee x_{l'} \dots) \wedge \dots$$

geschrieben werden kann. Dies bedeutet, dass jede Boolesche Funktion in einem Schaltkreis implementiert werden 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

$$F(x_1, x_2, x_3, \dots, x_N) = \dots \vee (x_k \wedge \neg x_l \wedge \dots) \vee (x_{k'} \wedge x_{l'} \dots) \vee \dots$$

eines N -adischen Wahrheitswertfunktors in einem Schaltkreis implementieren.

Für das Rechnen mit Wahrheitswertfunktoren ist es besonders bequem, wenn man diese mit den Kompositionen des Booleschen Ringes mit zwei Elementen darstellt. Die Kompositionstabellen 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

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

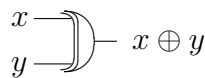
dargestellt. Die Darstellung der Disjunktion liegt damit nach dem dritten de Morganschen Gesetz fest:

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

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

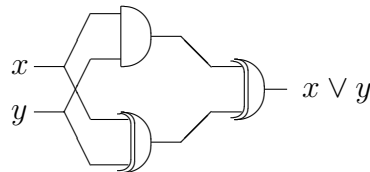
Kontravalenz

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



XOR

Disjunktion und Kontravalenz



OR

Eine Menge von Wahrheitswertfunktionen 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,

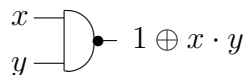
$$\begin{aligned} 1 \oplus (1 \oplus x) \oplus (1 \oplus y) \oplus (1 \oplus x) \cdot (1 \oplus y) &= 1 \oplus x \oplus y \oplus 1 \oplus x \oplus y \oplus x \cdot y \\ &= x \cdot y, \end{aligned}$$

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

Gebräuchliche Wahrheitswertfunktionen sind auch

Exklusion

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

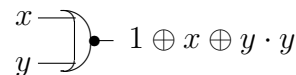


$$1 \oplus x \cdot y$$

NAND

Rejektion

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



$$1 \oplus x \oplus y \cdot y$$

NOR

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

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

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

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

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

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

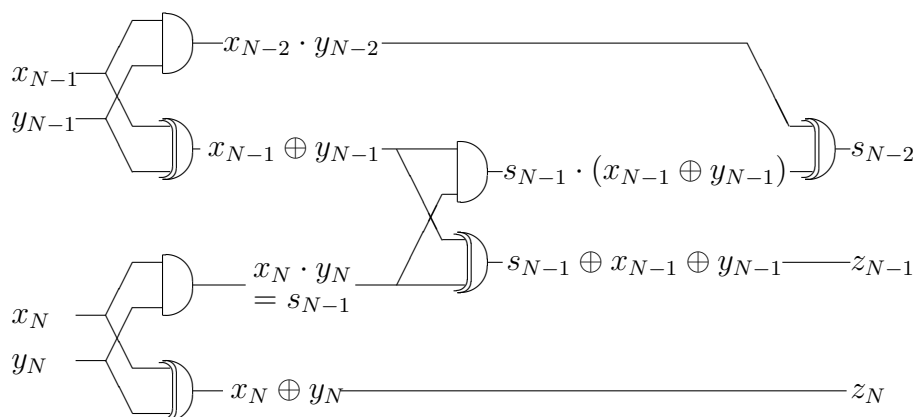
wird durch den folgenden Algorithmus gelöst, der sich aus einem ‘‘Halbaddierer’’

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

und $N - 1$ ‘‘Volladdierern’’

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

zusammensetzt. Der folgende Schaltkreis zeigt einen Halbaddierer 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 Rechengangs angegeben. Diese ist im vorliegenden Fall unabhängig von N gleich drei, weil übereinander angeordnete Gatter parallel arbeiten.