Vorlesung: Quantencomputing

Mittwochsakademie

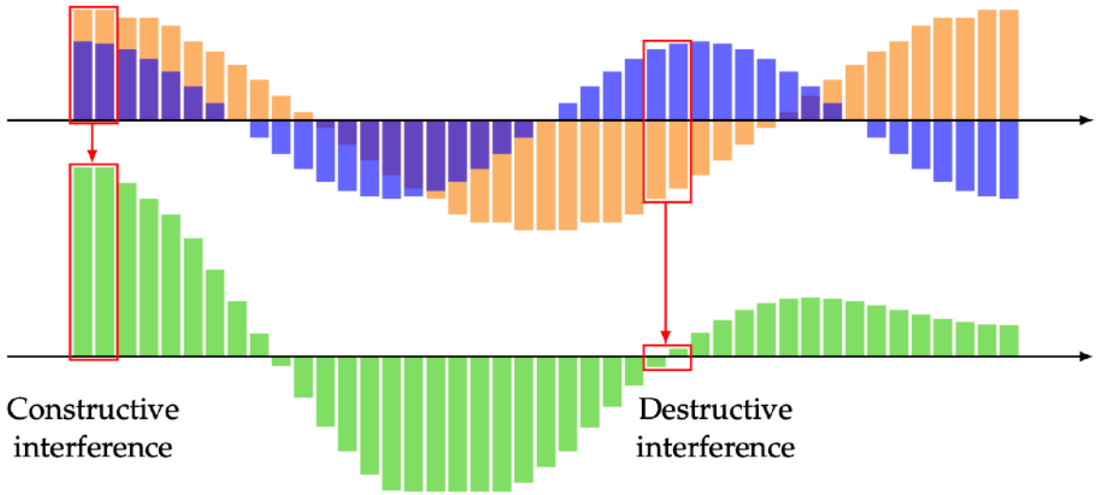
angelehnt an
“The Quantum Quest“ von Maris Ozols & Michael Walter
https://qi-rub.github.io/quantum-quest/2023/de/

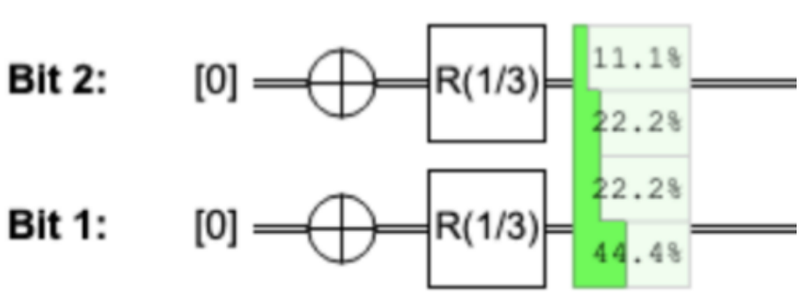
Ablauf

19.11.: Einführung
26.11.: Q1 Maestro der Wahrscheinlichkeit
3.12.: Q2a KEINE Vorlesung
10.12.: Q2b Das Qubit bezwingen
17.12.: Q3a Verzaubernde Verschränkungen 1

7. 1.: Q3b Verzaubernde Verschränkungen 2
14. 1.: Q3b Verzaubernde Verschränkungen 3
21. 1.: Q4b Quantenkompositionen 1
28. 1.: Q4b Quantenkompositionen 2
4. 2.: Q5 Virtuose Algorithmen

Verabschiedung von Prof. Claus Grupen





$$
\begin{aligned}
H\left(\frac{1}{\sqrt{2}}|0\rangle+\frac{1}{\sqrt{2}}|1\rangle\right) &= \frac{1}{\sqrt{2}}H|0\rangle+\frac{1}{\sqrt{2}}H|1\rangle \\
&= \frac{1}{\sqrt{2}}\left(\frac{1}{\sqrt{2}}|0\rangle+\frac{1}{\sqrt{2}}|1\rangle\right)+\frac{1}{\sqrt{2}}\left(\frac{1}{\sqrt{2}}|0\rangle-\frac{1}{\sqrt{2}}|1\rangle\right) \\
&= \left(\frac{1}{\sqrt{2}}\cdot\frac{1}{\sqrt{2}}+\frac{1}{\sqrt{2}}\cdot\frac{1}{\sqrt{2}}\right)|0\rangle+\left(\frac{1}{\sqrt{2}}\cdot\frac{1}{\sqrt{2}}-\frac{1}{\sqrt{2}}\cdot\frac{1}{\sqrt{2}}\right)|1 \\
&= |0\rangle\,.
\end{aligned}
$$

https://www.quantum-quest.org/quirky

Wdh.: Operationen auf einem Qubit

NOT-Operation: $\hat{\mathbf{NOT}}|0\rangle = |1\rangle \quad \hat{\mathbf{NOT}}|1\rangle = |0\rangle$

Z-Operation: $\hat{Z}|0\rangle = |0\rangle\ \hat{Z}|1\rangle = -|1\rangle$ **Spiegelung an der** $|0\rangle$ **Achse**

Rotationen: $\hat{U}(\theta)\begin{pmatrix}1\\0\end{pmatrix} = \begin{pmatrix}\cos\theta\\ \sin\theta\end{pmatrix} = |\psi(\theta)\rangle$; $\hat{U}(\theta)\begin{pmatrix}0\\1\end{pmatrix} = \begin{pmatrix}-\sin\theta\\ \cos\theta\end{pmatrix} = |\psi(\theta+\frac{\pi}{2})\rangle$

Allgemeinste Spiegelung $V(\theta)$ **hat die Form:** $V(\theta) = \mathbf{NOT}\ U(\theta) = U(-\theta)\ \mathbf{NOT}$

Hadamard Transformation $\hat{H}$ $\hat{H} = \hat{V}\left(\frac{\pi}{4}\right) = \mathbf{NOT}\ \hat{U}\left(\frac{\pi}{4}\right)$

$$|+\rangle := \frac{|0\rangle + |1\rangle}{\sqrt{2}} \qquad \hat{H}|0\rangle = \frac{1}{\sqrt{2}}\left(|0\rangle + |1\rangle\right) =: |+\rangle$$

$$|-\rangle := \frac{|0\rangle - |1\rangle}{\sqrt{2}} \qquad \hat{H}|1\rangle = \frac{1}{\sqrt{2}}\left(|0\rangle - |1\rangle\right) =: |-\rangle$$

Wdh.: Kontrollierte Operationen

Man kann für jede Ein QuBit Operation $\hat{U}$ verallgemeinerte kontrollierte Operationen $C\hat{U}_{1\to 2}$ einführen:

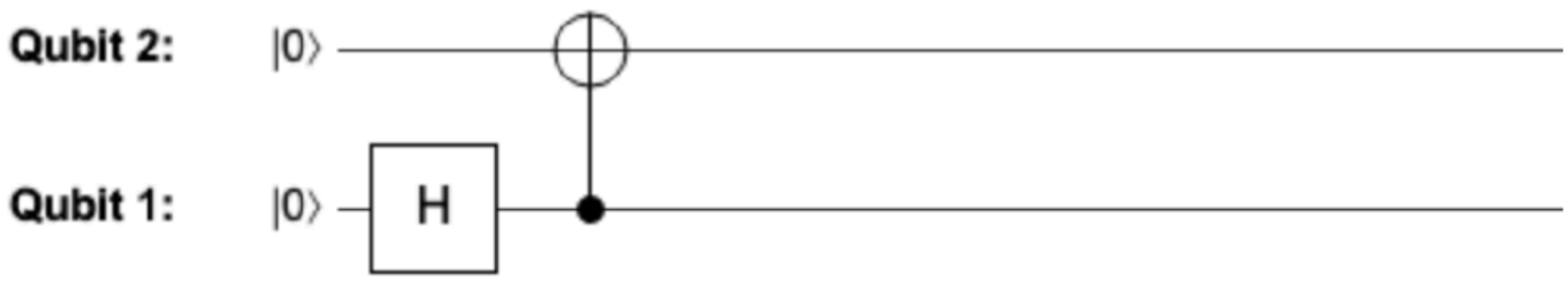
$$
\begin{aligned}
C\mathcal{U}_{1\to 2}\,|00\rangle &= |0\rangle \otimes |0\rangle\,, \\
C\mathcal{U}_{1\to 2}\,|01\rangle &= |0\rangle \otimes |1\rangle\,, \\
C\mathcal{U}_{1\to 2}\,|10\rangle &= |1\rangle \otimes \mathcal{U}\,|0\rangle\,, \\
C\mathcal{U}_{1\to 2}\,|11\rangle &= |1\rangle \otimes \mathcal{U}\,|1\rangle\,.
\end{aligned}
$$

Wdh.: Verschränkte Zustände

Beispiel: $|\Phi^+\rangle = \frac{1}{\sqrt{2}}|00\rangle + \frac{1}{\sqrt{2}}|11\rangle \Rightarrow \Delta(\Phi^+) = \frac{1}{2} \neq 0$

Dieser Zustand wir auch der maximal verschränkte Zustand genannt

Erzeugung via Quirky:



Beweis:

$$\mathrm{CNOT}_{1\to 2}\,(H \otimes I)\,|00\rangle = \mathrm{CNOT}_{1\to 2}\left(\frac{1}{\sqrt{2}}|00\rangle + \frac{1}{\sqrt{2}}|10\rangle\right) = \frac{1}{\sqrt{2}}|00\rangle + \frac{1}{\sqrt{2}}|11\rangle$$

Wdh.: Verschränkte Zustände

$|\Phi^+\rangle$ gehört zu einer Familie von 4 Zuständen die Bell-Zustände genannt werden

$$|\Phi^+\rangle = \frac{1}{\sqrt{2}}|00\rangle + \frac{1}{\sqrt{2}}|11\rangle,$$
$$|\Phi^-\rangle = \frac{1}{\sqrt{2}}|00\rangle - \frac{1}{\sqrt{2}}|11\rangle,$$
$$|\Psi^+\rangle = \frac{1}{\sqrt{2}}|01\rangle + \frac{1}{\sqrt{2}}|10\rangle,$$
$$|\Psi^-\rangle = \frac{1}{\sqrt{2}}|01\rangle - \frac{1}{\sqrt{2}}|10\rangle.$$

Wir definieren folgende Operation:
$\hat{U}_{\textbf{Bell}} := C\hat{\textbf{NOT}}_{1\to 2}(\hat{H}\otimes\hat{1})$ **und finden**

$$|\Phi^+\rangle = U_{\text{Bell}}|00\rangle,$$
$$|\Psi^+\rangle = U_{\text{Bell}}|01\rangle,$$
$$|\Phi^-\rangle = U_{\text{Bell}}|10\rangle,$$
$$|\Psi^-\rangle = U_{\text{Bell}}|11\rangle.$$

$$\text{CNOT}_{1\to 2}(H\otimes I)|00\rangle = \text{CNOT}_{1\to 2}\left(\frac{1}{\sqrt{2}}|00\rangle + \frac{1}{\sqrt{2}}|10\rangle\right) = \frac{1}{\sqrt{2}}|00\rangle + \frac{1}{\sqrt{2}}|11\rangle$$

$\hat{U}^{-1}_{\textbf{Bell}} := (\hat{H}\otimes\hat{1})C\hat{\textbf{NOT}}_{1\to 2}$

$\hat{H}|0\rangle = |+\rangle,\ \hat{H}|1\rangle = |-\rangle$

$\hat{H}^2|0\rangle = \hat{H}|+\rangle = \frac{1}{\sqrt{2}}(\hat{H}|0\rangle + \hat{H}|1\rangle) = \frac{1}{\sqrt{2}}(|+\rangle + |-\rangle) = \frac{1}{2}(2|0\rangle) = |0\rangle$

$\hat{H}^2|1\rangle = \hat{H}|-\rangle = \frac{1}{\sqrt{2}}(\hat{H}|0\rangle - \hat{H}|1\rangle) = \frac{1}{\sqrt{2}}(|+\rangle - |-\rangle) = \frac{1}{2}(2|1\rangle) = |0\rangle$

Wdh.: Die Macht von Verschränkung

Superdense Coding = übertrage 2 Bit Info mit einem Bit

Start: Bob und Alice teilen sich schon vorher den Zustand $|\Phi^+\rangle$,
d.h. Alice besitzt das erste Bit und Bob das zweite Bit dieses Zustandes

Übungsaufgabe 3.14: Einen Bell Zustand in einen Anderen überführen

Zeige, dass Alice den maximal verschränkten Zustand $|\Phi^+\rangle$ in jeden anderen Bell Zustand $|\Phi^-\rangle$, $|\Psi^+\rangle$, oder $|\Psi^-\rangle$ überführen kann mittels lokaler Operationen nur auf ihrem Qubit.

Alice ändert nur das 1. Bit und kann damit alle 2 QuBit Bell-zustände erzeugen!

$$|\Phi^+\rangle = \frac{1}{\sqrt{2}}|00\rangle + \frac{1}{\sqrt{2}}|11\rangle, \qquad \textbf{1. QuBit: } \hat{1}_1 \qquad \textbf{Dies wendet Alice im Fall } \{0,0\} \textbf{ an}$$

$$|\Phi^-\rangle = \frac{1}{\sqrt{2}}|00\rangle - \frac{1}{\sqrt{2}}|11\rangle, \qquad \textbf{1. QuBit: } \hat{Z}_1 \qquad \textbf{Dies wendet Alice im Fall } \{0,1\} \textbf{ an}$$

$$|\Psi^+\rangle = \frac{1}{\sqrt{2}}|01\rangle + \frac{1}{\sqrt{2}}|10\rangle, \qquad \textbf{1. QuBit: } \hat{\text{NOT}}_1 \qquad \textbf{Dies wendet Alice im Fall } \{1,0\} \textbf{ an}$$

$$|\Psi^-\rangle = \frac{1}{\sqrt{2}}|01\rangle - \frac{1}{\sqrt{2}}|10\rangle. \qquad \textbf{1. QuBit: } \hat{Z}_1\,\hat{\text{NOT}}_1 \qquad \textbf{Dies wendet Alice im Fall } \{1,1\} \textbf{ an}$$

Dann sendet Alice Ihr QBuit an Bob und der hat den gesamten Zustand
und kann des gesamten Zustand extrahieren (Ü 3.13)

Wdh.: Die Macht von Verschränkung

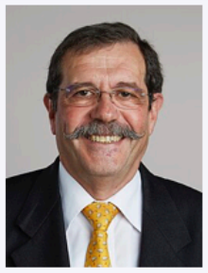
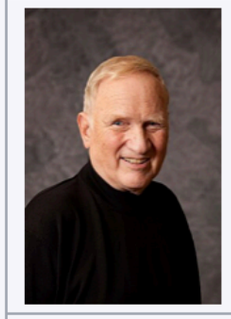
Klassisch: Maximal 75% Gewinnwahrscheinlichkeit - Beispiel für Bell-Ungleichung

John Stewart Bell; 1928-1990; 1964 Ungleichung

Obiges Spiel: John Clauser, Michael Horne, Abner Shimoney, Richard Holt

QM: verletzt Bell-Ungleichung, mehr als 75%

- dies ist experimentell bewiesen, z.B. Alan Aspect

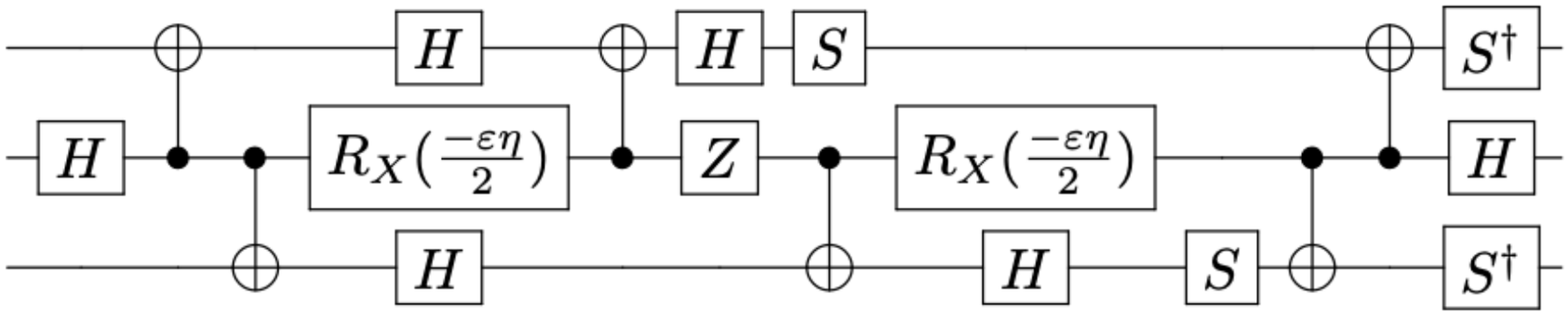
2022		Alain Aspect (b. 1947)	French	"for experiments with entangled photons, establishing the violation of Bell inequalities and pioneering quantum information science"
		John Clauser (b. 1942)	American	
		Anton Zeilinger (b. 1945)	Austrian	

Kann auch als Beweis genutzt werden, dass man einen echten Quantencomputer hat!

Man spielt das Spiel und bei > 75% war es ein QC :-)

Wdh.: Quantenschaltungen

Bildliche Darstellung: welche Operation wird an welchen Qubit durchgeführt



Beispiel: Simulation einer vereinfachten Theorie des Elektromagnetismus

Formal besteht eine Quantenschaltung aus 3 Teilen:

1. **Initialzustand: typischerweise $|0\rangle$**
2. **Quantenoperationen: meist 1 oder 2 QuBits gleichzeitig involviert**
3. **Messungen, um QuBits auszulesen**

(Siehe Quirky)

Die Operationen werden oft auch als Gatter oder Gates bezeichnet

z.B.: Hadamard-Operation ≡ Hadamard-Gatter ≡ Hadamard-Gate

Wdh.: Viele Quantenbits

Beliebiger Zustand mit n QuBits: 2^n Basis Elemente

$$|\psi\rangle = \psi_{00\ldots00}\,|00\ldots00\rangle + \psi_{00\ldots01}\,|00\ldots01\rangle + \ldots + \psi_{11\ldots11}\,|11\ldots11\rangle$$

Es muss gelten:

$$\psi^2_{00\ldots00} + \psi^2_{00\ldots01} + \ldots + \psi^2_{11\ldots11} = 1$$

Mögliche Darstellung als Vektor in einem 2^n dimensionalen Vektorraum.

Bei $n = 300$ gibt es $2^{300} \approx 2 \cdot 10^{91}$ Amplitude (mehr als Atome im Universum) d.h. sowas kann nicht klassisch gespeichert werden, aber als Quanten Computer gebaut!

Wdh.: Viele Quantenbits

Mit dem Tensorprodukt können Zustände beschrieben werden, die zu kombinierten QuBits gehören, allgemein:

$$|a_1, \ldots, a_n\rangle \otimes |b_1, \ldots, b_m\rangle = |a_1, \ldots, a_n, b_1, \ldots, b_m\rangle .$$

Beispiel 1: $|101\rangle \otimes |01\rangle = |10101\rangle$

1 QuBit-Operationen $\hat{U}$ wirken wie folgt:

$$\hat{U}_1 |a_1, \ldots, a_n\rangle = \hat{U} |a_1\rangle \otimes |a_2, \ldots, a_n\rangle$$

Analoge Definitionen für $\hat{U}_2$, $\hat{U}_3$,...

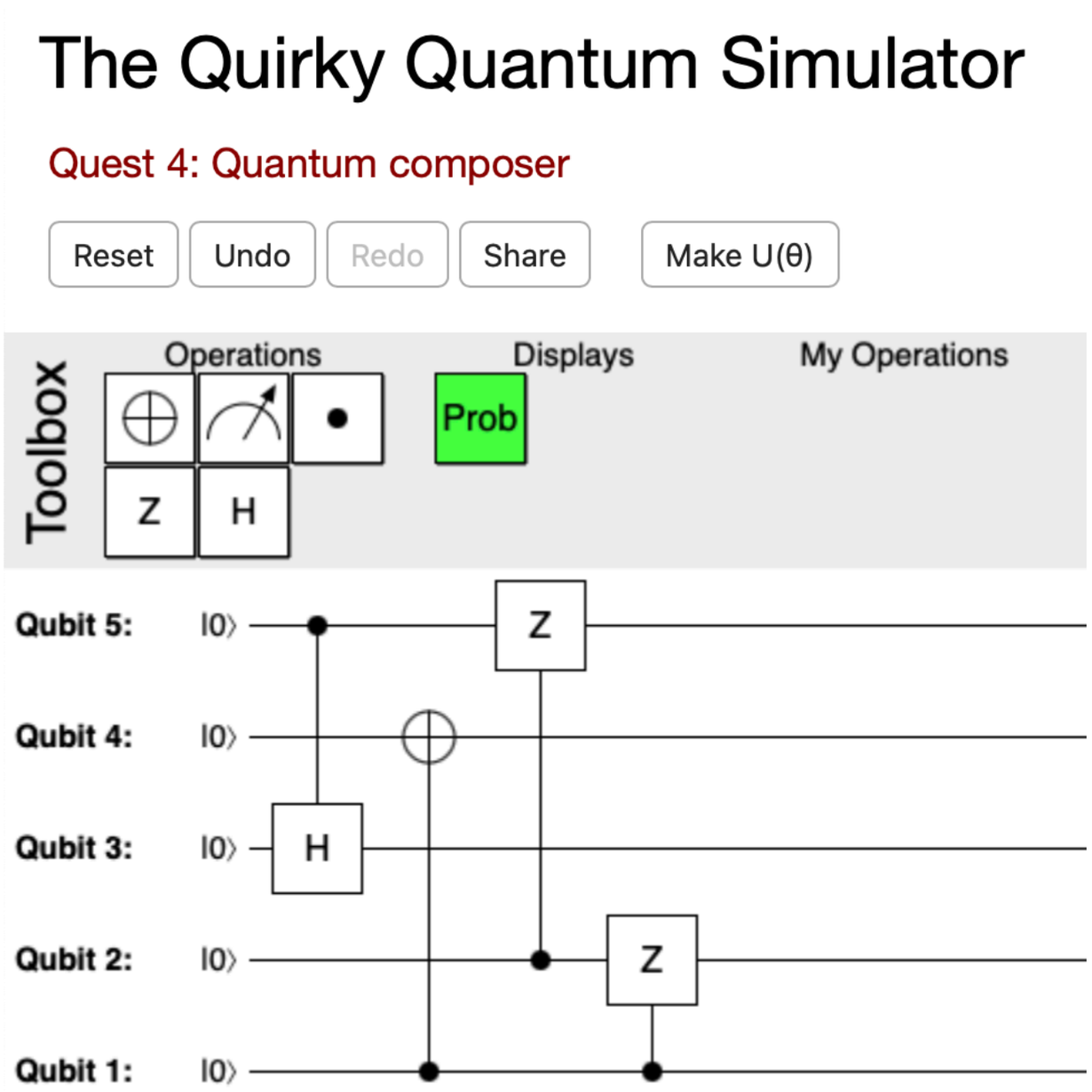
Beispiel 2: $|\Phi^+\rangle \otimes |1\rangle = \frac{1}{\sqrt{2}} \left(|001\rangle + |111\rangle \right)$

$$\hat{H}_2 |\Phi^+\rangle \otimes |1\rangle = \frac{1}{2} \left(|001\rangle + |011\rangle + |101\rangle - |111\rangle \right)$$

Wdh.: Operationen

2 QuBit-Operationen $\hat{\text{CNOT}}_{i\to k}$ wirken wie folgt:

$$\hat{\text{CNOT}}_{i\to k}\,|a_1,\ldots,a_k,\ldots,a_n\rangle = |a_1,\ldots,a_i\oplus a_k,\ldots,a_n\rangle$$



Wdh.: Die allgemeinsten Quantenoperationen

Die allgemeinste Quantenoperation hat folgende Eigenschaften:

1. **Sie ist linear**
2. **Sie bildet Quantenzustände auf Quantenzustände ab (Normierung)**
3. **Sie ist invertierter (reversibel)**

Wdh.: Die allgemeinsten Quantenoperationen

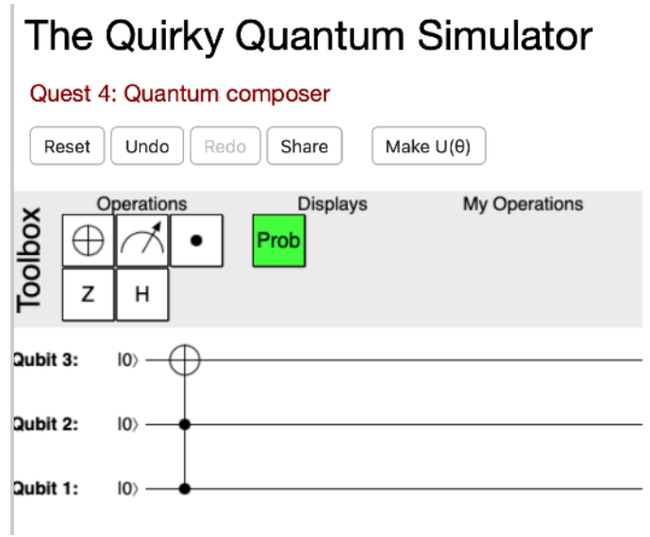
Übungsaufgabe 4.4: Toffoli

Definiere die Toffoli-Operation auf drei Qubits durch

$$T\,|a,b,c\rangle = |a,b,c \oplus ab\rangle$$

auf Basiszuständen (ab ist dabei das Produkt der zwei Bits $a, b \in \{0,1\}$, und $\oplus$ wurde in Gl. (3.20) definiert), und erweitere sie durch Linearität auf beliebige Drei-Qubit-Zustände. Zeige, dass T alle Quantenzustände auf Quantenzustände abbildet, und dass T invertierbar ist.

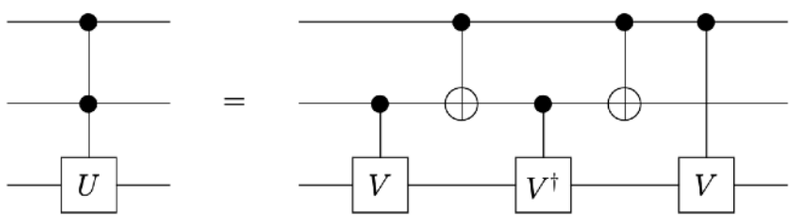
Bemerkung: T invertiert das dritte Bit genau dann, wenn beide ersten Bits beide eins sind – es ist also eine "zweifach-kontrollierte"-NOT-Operation.



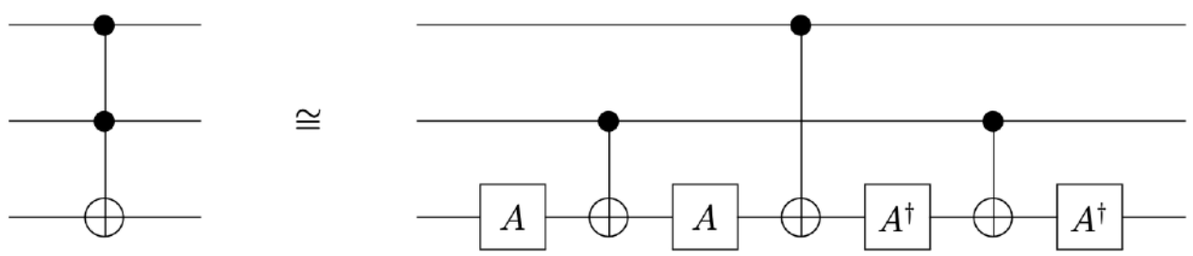
T kann auch als eine Reihe von 1- und 2 QuBit-Operationen geschrieben werden!

https://arxiv.org/pdf/quant-ph/9503016

Lemma 6.1: *For any unitary 2×2 matrix U, a $\wedge_2(U)$ gate can be simulated by a network of the form*



where V is unitary.



where $A = \mathrm{R_y}(\frac{\pi}{4})$. In the above, the "≅" indicates that the networks are not identical, but differ at most in the phases of their amplitudes, which are all ± 1 (the phase of the $|101\rangle$ state is reversed in this case).

Jede Quanten-Operation auf n Qu-Bits kann auch als eine Reihe von 1- und 2 QuBit-Operationen geschrieben werden!

Wdh.: Regeln für Schaltungen

Es gibt Tricks zur Vereinfachung von Quantenschaltungen
Einfachere Schaltungen sind meist auch schneller

Übungsaufgabe 4.5: Z und NOT

Aus Gl. (2.20) wissen wir, dass das Hadamard-Gate H sich wie folgt auswirkt:

$$H|0\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle) = |+\rangle\,, \qquad H|1\rangle = \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle) = |-\rangle\,.$$

1. Prüfe, dass ein erneutes anwenden von H wieder zu den Zuständen $|0\rangle$ und $|1\rangle$ führt.

$$H|+\rangle = |0\rangle\,, \qquad H|-\rangle = |1\rangle\,.$$

2. Zeige, dass $HZH = \mathrm{NOT}$, wobei Z in Gl. (2.12) definiert ist.
3. Zeige, dass $H\,\mathrm{NOT}\,H = Z$.

Übungsaufgabe 4.6: Spiegelungen und Drehungen (optional)

Zeige, dass das Produkt zweier Spiegelungen eine Rotation ist. Zeige also, dass

$$V(\theta_2)V(\theta_1) = U(\theta),$$

für einen Winkel θ. Kannst du θ relativ zu θ_1 und θ_2 bestimmen?

Hint: Nutze Gl. (2.19) und die Gleichung $U(\varphi_2)U(\varphi_1) = U(\varphi_1 + \varphi_2)$.

$$\hat{H}|+\rangle = \frac{1}{\sqrt{2}}\left(\hat{H}|0\rangle + \hat{H}|1\rangle\right) = \frac{1}{2}\left(|0\rangle + |1\rangle + |0\rangle - |1\rangle\right) = |0\rangle$$

$$\hat{H}|-\rangle = \frac{1}{\sqrt{2}}\left(\hat{H}|0\rangle - \hat{H}|1\rangle\right) = \frac{1}{2}\left(|0\rangle + |1\rangle - |0\rangle + |1\rangle\right) = |1\rangle$$

Definition:

$$\hat{V}(\theta) = \mathbf{N\hat{O}T}\,\hat{U}(\theta) = \hat{U}(-\theta)\,\mathbf{N\hat{O}T}$$

$$\hat{H}\hat{Z}\hat{H}|0\rangle = \hat{H}\hat{Z}|+\rangle = \hat{H}|-\rangle = |1\rangle = \mathbf{N\hat{O}T}|0\rangle$$
$$\hat{H}\hat{Z}\hat{H}|1\rangle = \hat{H}\hat{Z}|-\rangle = \hat{H}|+\rangle = |0\rangle = \mathbf{N\hat{O}T}|1\rangle$$

$$\hat{V}(\theta_2)\hat{V}(\theta_1) = \hat{U}(-\theta_2)\,\mathbf{N\hat{O}T}\,\mathbf{N\hat{O}T}\,\hat{U}(\theta_1) = \hat{U}(-\theta_2)\hat{U}(\theta_1) = \hat{U}(\theta_1 - \theta_2)$$

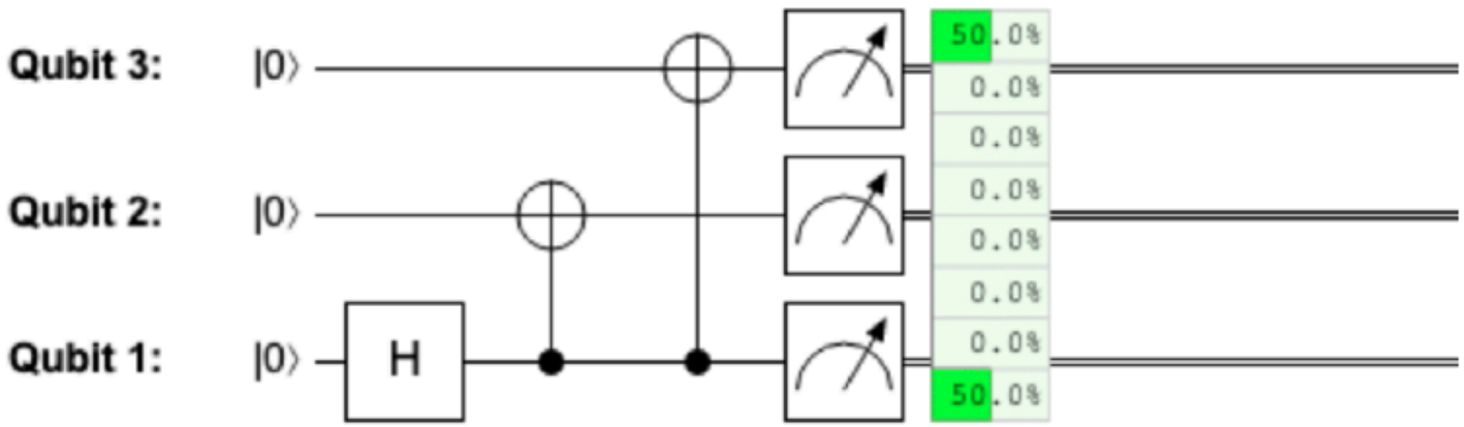
$$\hat{H}\mathbf{N\hat{O}T}\hat{H}|0\rangle = \hat{H}\mathbf{N\hat{O}T}|+\rangle = \hat{H}|+\rangle = |0\rangle = \hat{Z}|0\rangle$$
$$\hat{H}\mathbf{N\hat{O}T}\hat{H}|1\rangle = \hat{H}\mathbf{N\hat{O}T}|-\rangle = -\hat{H}|-\rangle = -|1\rangle = \hat{Z}|1\rangle$$

Wdh.: Alle Qubits messen

Wenn wir n QuBits messen dann erhalten wir mit der Wahrscheinlichkeit

$$p_{a_1....a_n} = \psi^2_{a_1....a_n}$$ **den Bit-String** $a_1 \ldots . a_n$

Quirky



$$\frac{1}{\sqrt{2}} |000\rangle + \frac{1}{\sqrt{2}} |111\rangle$$

Wdh.: Einzelne Qubits messen

Annahme: Wir haben einen 3-QuBit Zustand

$$|\psi\rangle = \psi_{000}|000\rangle + \psi_{001}|001\rangle + \psi_{010}|010\rangle + \psi_{100}|100\rangle + \psi_{011}|011\rangle + \psi_{101}|101\rangle + \psi_{110}|110\rangle + \psi_{111}|111\rangle$$

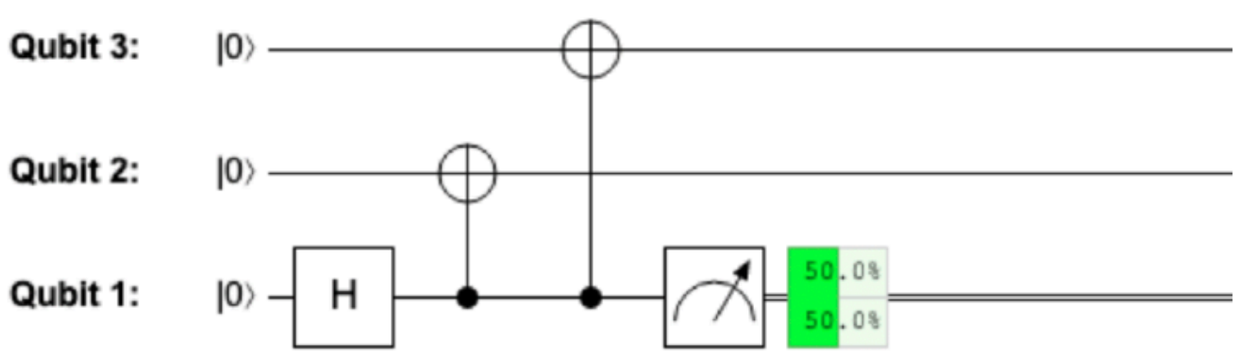
Wenn wir das erste Bit messen, dann finden wir mit der Wahrscheinlichkeit

$$\psi_{a00}^2 + \psi_{a01}^2 + \psi_{a10}^2 + \psi_{a11}^2$$

den Wert $a \in \{0,1\}$**.**

Beispiel: Messen wir das erste QuBit von $\frac{1}{\sqrt{8}}|000\rangle + \sqrt{\frac{2}{8}}|010\rangle + |\sqrt{\frac{5}{8}}|111\rangle$ **so finden wir den Wert Null mit der Wahrscheinlichkeit** $\frac{1}{8} + \frac{2}{8} = \frac{3}{8}$

Quirky:



$$\hat{\mathbf{CNOT}}_{1\to3}\hat{\mathbf{CNOT}}_{1\to2}\hat{H}_1|000\rangle = \frac{1}{\sqrt{2}}\left(\hat{\mathbf{CNOT}}_{1\to3}\hat{\mathbf{CNOT}}_{1\to2}|000\rangle + \hat{\mathbf{CNOT}}_{1\to3}\hat{\mathbf{CNOT}}_{1\to2}|100\rangle\right) = \frac{1}{\sqrt{2}}\left(|000\rangle + |111\rangle\right)$$

Wdh.: Einzelne Qubits messen

Zustand des 2. und 3. QuBit, nachdem das 1. gemessen wurde?

Zu der Messung tragen folgende Zustände bei:

$$|\psi\rangle = \psi_{a00}|a00\rangle + \psi_{a01}|a01\rangle + \psi_{a10}|a10\rangle + \psi_{a11}|a11\rangle$$

Nach der Messung des 1. QuBits können wir dies auch weglassen

$$|\psi\rangle = \psi_{a00}|00\rangle + \psi_{a01}|01\rangle + \psi_{a10}|10\rangle + \psi_{a11}|11\rangle$$

Jetzt müssen wir noch sicherstellen, dass dieser Zustand auch normiert ist

$$|\psi\rangle = \frac{\psi_{a00}}{c}|00\rangle + \frac{\psi_{a01}}{c}|01\rangle + \frac{\psi_{a10}}{c}|10\rangle + \frac{\psi_{a11}}{c}|11\rangle$$

mit $c = \sqrt{\psi_{a00}^2 + \psi_{a01}^2 + \psi_{a10}^2 + \psi_{a11}^2}$

Beispiel: **Messe 1. QuBit von** $\frac{1}{\sqrt{2}}|000\rangle + \frac{1}{\sqrt{2}}|111\rangle$ **und finde 0**

dann ist die verbleibende unnormierte Wellenfunktion $\frac{1}{\sqrt{2}}|00\rangle$

und die normierte Wellenfunktion lautet dann $|00\rangle$

Wdh.: Einzelne Qubits messen

Messe 1. QuBit von 3 QuBit Zustand

Dies kann man sich vorstellen als

$$|\psi\rangle = \sqrt{p_0}\,|0\rangle \otimes \frac{\psi_{000}\,|00\rangle + \psi_{001}\,|01\rangle + \psi_{010}\,|10\rangle + \psi_{011}\,|11\rangle}{\sqrt{p_0}} + \sqrt{p_1}\,|1\rangle \otimes \frac{\psi_{100}\,|00\rangle + \psi_{101}\,|01\rangle + \psi_{110}\,|10\rangle + \psi_{111}\,|11\rangle}{\sqrt{p_1}}.$$

Oder (beachte: $p_0 + p_1 = 1$!)

$$|\psi\rangle = \sqrt{p_0}\,|0\rangle \otimes |\psi_0\rangle + \sqrt{p_1}\,|1\rangle \otimes |\psi_1\rangle\,,$$

Beispiel:

$$\frac{1}{\sqrt{2}}\,|000\rangle + \frac{1}{\sqrt{2}}\,|111\rangle = \frac{1}{\sqrt{2}}\,|0\rangle \otimes |00\rangle + \frac{1}{\sqrt{2}}\,|1\rangle \otimes |11\rangle\,,$$

Hieraus kann man sofort die Wahrscheinlichkeiten ablesen

Wdh.: Einzelne Qubits messen

Wir können auch die ersten zwei QuBits eines allgemeinen Drei QuBit zustand messen.

Wir erhalten $|ab\rangle$ mit der Wahrscheinlichkeit $p_{a,b} = \psi_{ab0}^2 + \psi_{ab1}^2$

Nach der Messung haben wir dann den Zustand: $|\psi_{a,b}\rangle = \frac{\psi_{ab0}|0\rangle + \psi_{ab1}|1\rangle}{\sqrt{\psi_{ab0}^2 + \psi_{ab1}^2}}$

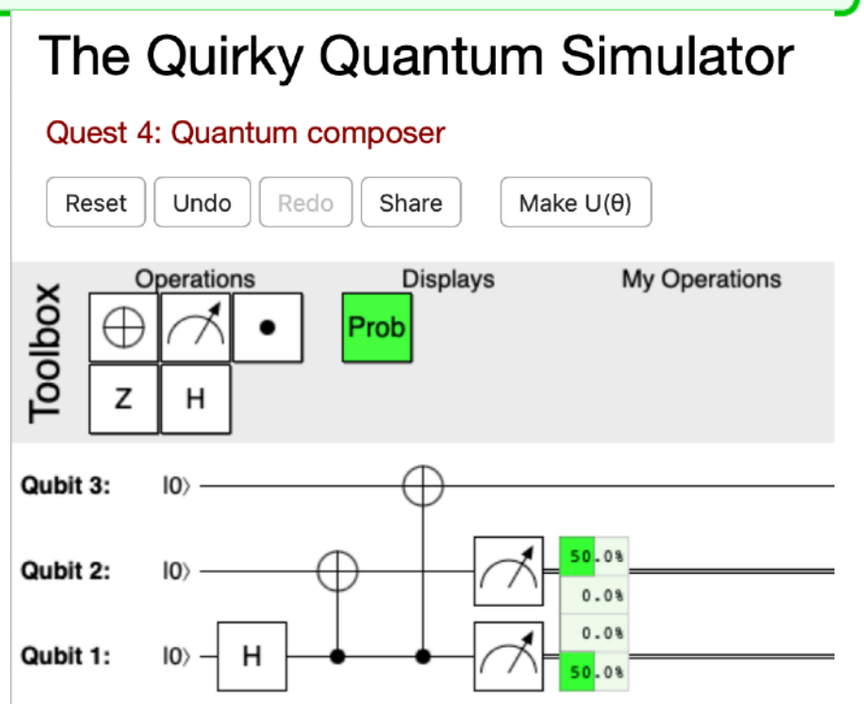
Übungsaufgabe 4.7: Zwei von Drei

Mit welchen Wahrscheinlichkeiten ergibt das Messen der ersten beiden Qubits des Drei-Qubit-Zustands aus Gl. (4.7) welche Messergebnisse? Überprüfe dein Ergebnis mit QUIRKY.

$$\frac{1}{\sqrt{2}}|000\rangle + \frac{1}{\sqrt{2}}|111\rangle$$

$$\frac{1}{2} : |00\rangle$$

$$\frac{1}{2} : |11\rangle$$

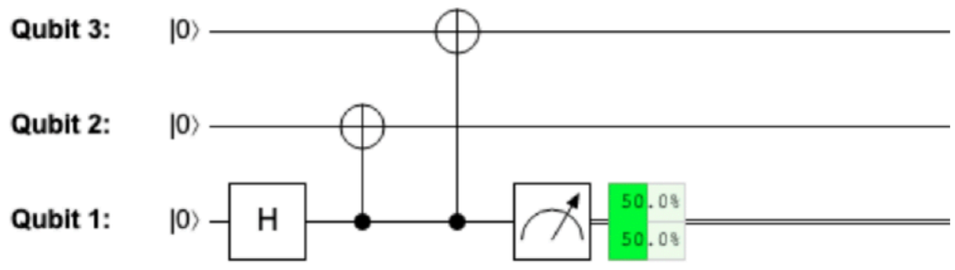


Wdh.: Einzelne Qubits messen

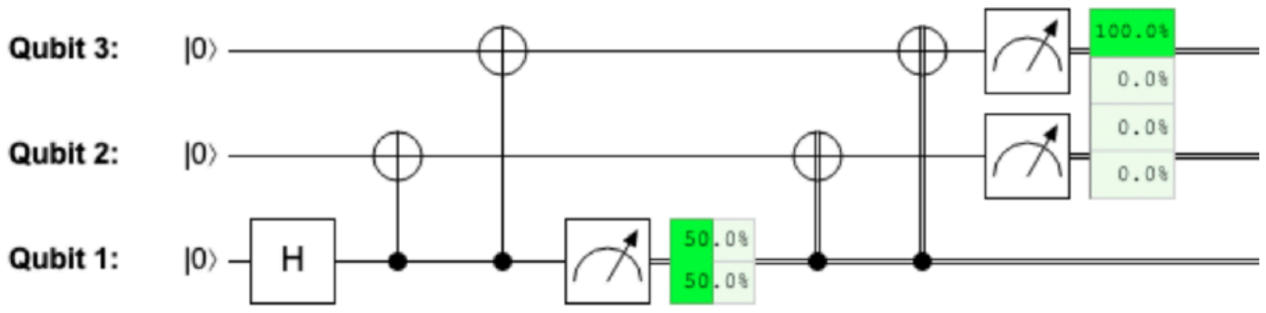
Wir können einzelne QuBits messen und abhängig vom Messergebnis die verbleibenden QuBits modifizieren

Beispiel

$$\frac{1}{\sqrt{2}}|000\rangle + \frac{1}{\sqrt{2}}|111\rangle$$



Messe 1. QuBit: beim Ergebnis $|1\rangle$ wollen wir die verbleibenden QuBits auf $|00\rangle$ zurücksetzen



Kontroll-Bit ist hier ein klassisches Bit

Formal:

$$\mathrm{CNOT}_{1\to 2}[a] \otimes |b, c\rangle = [a] \otimes |a \oplus b, c\rangle$$

Wdh.: Quanten-Überraschungen

**Verschiedene interessante Phänomene,
die beim Umgang mit QuBits auftreten**

Wdh.: Unklonbarkeit

Klassische Bits können einfach geklont werden:
wir schauen es an und das was wir sehen, das kopieren wir

$$[0] \mapsto [00],$$
$$[1] \mapsto [11].$$

Kann man Qu-Bits auch klonen?

Annahme: Klonen geht, d.h. bei gegeben Zustand $|\psi\rangle$ gibt es die Operation C,
die z.B. aus dem Zustand $|0\rangle$ den Zustand $|\psi\rangle$ macht .

$$C\left(|\psi\rangle \otimes |0\rangle\right) = |\psi\rangle \otimes |\psi\rangle$$

Diese Operation macht dann folgendes aus den Basis Operatoren

$$C\left(|0\rangle \otimes |0\rangle\right) = |0\rangle \otimes |0\rangle$$
$$C\left(|1\rangle \otimes |0\rangle\right) = |1\rangle \otimes |1\rangle$$

Dies ist möglich und wäre z.B. die $\hat{\text{CNOT}}_{1\to 2}$

Wdh.: Unklonbarkeit

Hausaufgabe 4.1: Unklonbarkeit

In dieser Hausaufgabe wollen wir beweisen, dass es keine Quantenoperation C geben kann, die Gl. (4.16) erfüllt. Wir nutzen dafür einen Trick, der sich Widerspruchsbeweis nennt. Das bedeutet, wir werden zeigen, dass die Existenz einer Klon-Operation C etwas impliziert, von dem wir wissen, dass es nicht stimmt (z.B. "$0 = 1$"). Daraus können wir dann schließen, dass kein solches C existieren kann.

Lass uns also zu Beginn annehmen, dass es eine Quantenoperation C gibt, die Gl. (4.16) erfüllt. Nun kannst du $C(|+\rangle \otimes |0\rangle)$ auf zwei verschiedene Arten berechnen:

1. Nutze zuerst Gl. (4.16) und schreibe das Ergebnis dann in der Form von Gl. (3.30).
2. Schreibe erst $|+\rangle \otimes |0\rangle$ in der Form von Gl. (3.30), nutze dann die Linearität von C und wende abschließend Gl. (4.16) an.

Erhältst du in beiden Fällen das gleiche Ergebnis? Wenn nicht, was kannst du daraus schließen?

$$\hat{C}\left(|+\rangle \otimes |0\rangle\right) \xrightarrow{C(|\psi\rangle \otimes |0\rangle) = |\psi\rangle \otimes |\psi\rangle} = |+\rangle \otimes |+\rangle = \frac{1}{2}|00\rangle + \frac{1}{2}|01\rangle + \frac{1}{2}|10\rangle + \frac{1}{2}|11\rangle$$

$$\hat{C}\left(|+\rangle \otimes |0\rangle\right) \rightarrow = \frac{1}{\sqrt{2}}\hat{C}\left(|00\rangle + |10\rangle\right) = \frac{1}{\sqrt{2}}|00\rangle + \frac{1}{\sqrt{2}}|10\rangle$$

$$C\left(|0\rangle \otimes |0\rangle\right) = |0\rangle \otimes |0\rangle$$
$$C\left(|1\rangle \otimes |0\rangle\right) = |1\rangle \otimes |1\rangle$$

Widerspruch zur Linearität

Wdh.: Unklonbarkeit

No-Cloning Theorem

Ein unbekannter Quantenzustand kann nicht kopiert werden.
Dieser beinhaltet im Allgemeinen unendlich viel Information!

Received 15 June; accepted 1 September 1982.

1. Kagi, J. H. R. & Nordberg, M. (eds) *Metallothionein* (Birkhauser, Basle, 1979).
2. Karin, M. & Herschman, H. R. *Science* **204,** 176–177 (1979).
3. Pulido, P., Kagi, J. H. R. & Vallee, B. L. *Biochemistry* **5,** 1768–1777 (1966).
4. Rudd, C. J. & Herschmann, H. R. *Tox. appl. Pharmac.* **47,** 273–278 (1979).
5. Karin, M. & Herschman, H. R. *Eur. J. Biochem.* **107,** 395–401 (1980).
6. Kissling, M. M. and Kagi, J. H. R. *FEBS Lett.* **82,** 247–250 (1977).
7. Karin, M. *et al. Nature* **286,** 295–297 (1980).
8. Karin, M., Slater, E. P. & Herschman, H. R. *J. cell. Physiol.* **106,** 63–74 (1981).
9. Durnam, D. M. & Palmiter, R. D. *J. biol. Chem.* **256,** 5712–2716 (1981).
10. Hager, L. J. & Palmiter, R. D. *Nature* **291,** 340–342 (1981).
11. Karin, M. & Richards, R. *Nucleic Acids Res.* **10,** 3165–3173 (1982).
12. Lawn, R. M. *et al. Cell* **15,** 1157–1174 (1978).
13. Southern, E. M. *J. molec. Biol.* **98,** 503–517 (1975).
14. Benton, W. D. & Davis, R. W. *Science* **196,** 180–182 (1977).
15. Glanville, N., Durnam, D. M. & Palmiter, R. D. *Nature* **292,** 267–269 (1981).
16. Breathnach, R. *et al. Proc. natn. Acad. Sci. U.S.A.* **75,** 4853–4857 (1978).
17. Weaver, R. F. & Weissman, C. *Nucleic Acids Res.* **5,** 1175–1193 (1979).
18. Kayb, K. E., Warren, R. & Palmiter, R. D. *Cell* **29,** 99–108 (1982).
19. Brinster, R. L. *et al. Nature* **296,** 39–42 (1982).
20. Kingsbury, R. & McKnight, S. L. *Science* **217,** 316–324 (1982).
21. Larsen, A. & Weintraub, H. *Cell* **29,** 609–672 (1982).
22. Proudfoot, N. J. & Brownlee, G. G. *Nature* **263,** 211–214 (1976).
23. Calos, M. P. & Miller, J. H. *Cell* **20,** 579–595 (1980).
24. Hollis, F. G. *et al. Nature* **296,** 321–325 (1982).
25. Leuders, K., Leder, A., Leder, P. & Kuff, E. *Nature* **295,** 426–428 (1982).
26. Van Arsdell, S. W. *et al. Cell* **26,** 11–17 (1981).
27. Jagadeeswaran, P., Forget, B. G. & Weissman, S. M. *Cell* **26,** 141–142 (1982).
28. Nishioka, Y., Leder, A. & Leder, P. *Proc. natn. Acad. Sci. U.S.A.* **77,** 2806–2809 (1980).
29. Wilde, C. D. *et al. Nature* **297,** 83–84 (1982).
30. Shaul, Y., Kaminichik, J. & Aviv, H. *Eur. J. Biochem.* **116,** 461–466 (1981).
31. Perry, R. P. *et al. Proc. natn. Acad. Sci. U.S.A.* **77,** 1937–1941 (1980).
32. Hofer, E. & Darnel, J. E. *Cell* **23,** 585–593 (1981).
33. Bell, G., Karam, J. H. & Rutter, W. J. *Proc. natn. Acad. Sci. U.S.A.* **78,** 5759–5763 (1981).
34. Rigby, P. W. J. *et al. J. molec. Biol.* **113,** 237–251 (1977).
35. Wahl, G. M., Stern, M. & Stark, G. R. *Proc. natn. Acad. Sci. U.S.A.* **76,** 3683–3687 (1979).
36. Maxam, A. & Gilbert, W. *Meth. Enzym.* **65,** 499–559 (1980).
37. Sanger, F., Nicklen, S. & Coulson, A. R. *Proc. natn. Acad. Sci. U.S.A.* **74,** 5463–5468 (1979).
38. Goodman, H. M. *Meth. Enzym.* **65,** 63–64 (1980).
39. Heidecker, G., Messing, J. & Gronenborn, B. *Gene* **10,** 69–73 (1980).
40. O'Farrel, P. *Focus* **3,** 1–3 (1981).

LETTERS TO NATURE

A single quantum cannot be cloned

W. K. Wootters*

Center for Theoretical Physics, The University of Texas at Austin, Austin, Texas 78712, USA

W. H. Zurek

Theoretical Astrophysics 130–33, California Institute of Technology, Pasadena, California 91125, USA

on an incoming photon with polarization state $|s\rangle$:

$$|A_0\rangle|s\rangle \rightarrow |A_s\rangle|ss\rangle \quad (1)$$

Here $|A_0\rangle$ is the 'ready' state of the apparatus, and $|A_s\rangle$ is its final state, which may or may not depend on the polarization of the original photon. The symbol $|ss\rangle$ refers to the state of the radiation field in which there are two photons each having the polarization $|s\rangle$. Let us suppose that such an amplification can in fact be accomplished for the vertical polarization $|\updownarrow\rangle$ and for the horizontal polarization $|\leftrightarrow\rangle$. That is,

$$|A_0\rangle|\updownarrow\rangle \rightarrow |A_{vert}\rangle|\updownarrow\updownarrow\rangle \quad (2)$$

Wdh.: One-Time-Pad = Teleportation von probabilistischen Zuständen

Problem: **Alice möchte eine Nachricht an Bob verschicken.**
Die Nachricht soll so verschlüsselt sein, dass nur Bob sie verstehen kann, insbesondere nicht Eve, auch wenn sie die Nachricht liest .

Start: **Alice und Bob treffen sich davor in einem Cafe**

2 Münzen im Zustand $r = \frac{1}{2}[00] + \frac{1}{2}[11]$ **- jeder nimmt eine Münze mit, die nun entweder Kopf oder Zahl zeigt und dies nicht mehr verändert.**

Alice will nun die Nachricht $m \in \{0,1\}$ **an Bob schicken**

Der Gesamt-zustand aller Bits lautet dann $[m] \otimes r = \frac{1}{2}[m00] + \frac{1}{2}[m11]$

Alice besitzt die ersten beiden Bits von diesem Zustand, Bob das dritte.

Protokoll: **1) Alice schaut sich ihr Bit von** r **an**

2a) $r = 0 \Rightarrow$ **sie schickt** m **an Bob.** **2b)** $r = 1 \Rightarrow$ **sie schickt** $\hat{\text{NOT}}(m)$ **an Bob**

3) Wenn Eve diese Nachricht abfängt, dann erhält sie immer mit 50% 0 oder 1

4) Bob schaut sich sein Bit von r **an, wenn er** m **empfängt**

5a) $r = 0 \Rightarrow$ **Bob nimmt** m **5b)** $r = 1 \Rightarrow$ **Bob nimmt** $\hat{\text{NOT}}(m)$

Formal werden folgende Operationen durchgeführt: **Bob** **Alice**

$$\text{CNOT}_{3\to 1}\ \text{CNOT}_{2\to 1}([m] \otimes r).$$

Wdh.: One-Time-Pad

Formal werden folgende Operationen durchgeführt:

$$\underset{\text{Bob}}{\mathrm{CNOT}_{3\to 1}}\ \underset{\text{Alice}}{\mathrm{CNOT}_{2\to 1}}([m]\otimes r).$$

$$\mathrm{CNOT}_{3\to 1}\,\mathrm{CNOT}_{2\to 1}\,\frac{1}{2}([m,0,0]+[m,1,1]) = \mathrm{CNOT}_{3\to 1}\,\frac{1}{2}([m,0,0]+[\mathrm{NOT}(m),1,1])$$
$$= \frac{1}{2}([m,0,0]+[\mathrm{NOT}(\mathrm{NOT}(m)),1,1]) = \frac{1}{2}([m,0,0]+[m,1,1]) = [m]\otimes r.$$

Am Ende hat Bob das Bit m

Man kann nicht nur ein Basis-Bit 0 oder 1 verschicken, wegen Linearität kann man auch ein ganzes probabilistisches Bit p verschicken
Das gesamte Bit ist dann $p \otimes r$

Wdh.: Quanten-Teleportation

QuBits können nicht geklont werden, aber sie können von einem Ort zu einem anderen gebracht werden!

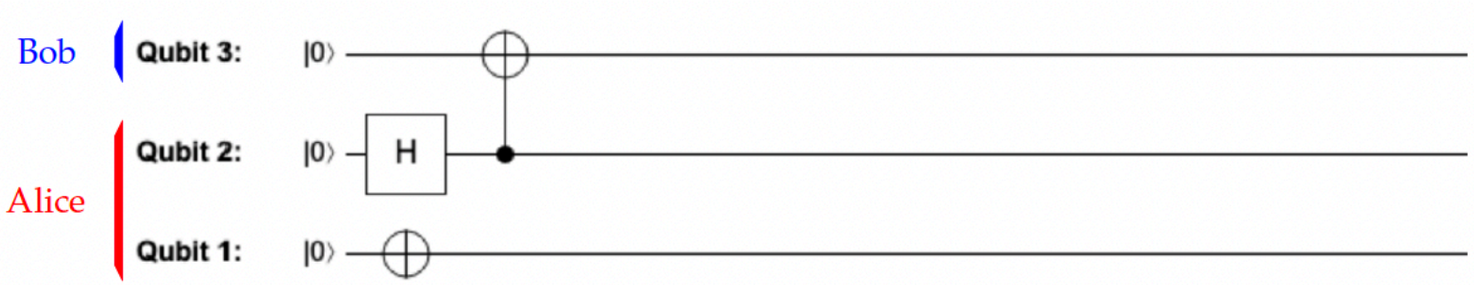
Ausgangszustand:

Alice und Bob teilen sich den maximal verschränkten Zustand $|\Phi^+\rangle$, Alice hat das erste QuBit davon und Bob das zweite.

Alice hat zusätzlich noch das Nachrichten-QuBit $|\psi\rangle$

Insgesamt beschreibt dies den 3 QuBit Zustand $|\psi\rangle \otimes |\Phi^+\rangle$

Falls Alice den Zustand $|\psi\rangle = |1\rangle$ senden will, sieht das so aus

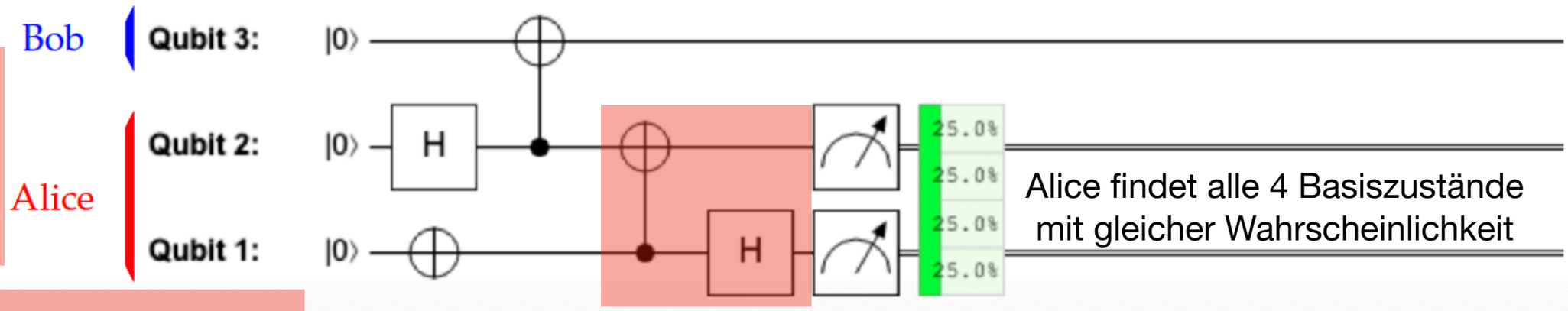


Wie machen wir weiter?

Wdh.: Quanten-Teleportation

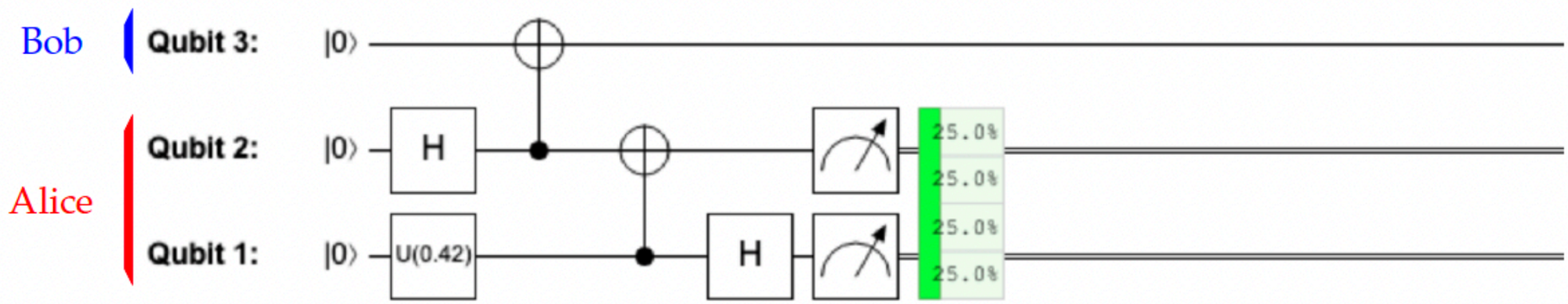
Wie machen wir weiter? Wir können $|\psi\rangle$ nicht klonen, vielleicht muss Alice's Zustand durch eine Messung zerstört werden? Einfache Messung reicht nicht, da $|\psi\rangle$ nicht aus einer einzigen Messung extrahiert werden kann.

Man wird finden: (Operation zur Unterscheidung der 4 Bell-zustände)

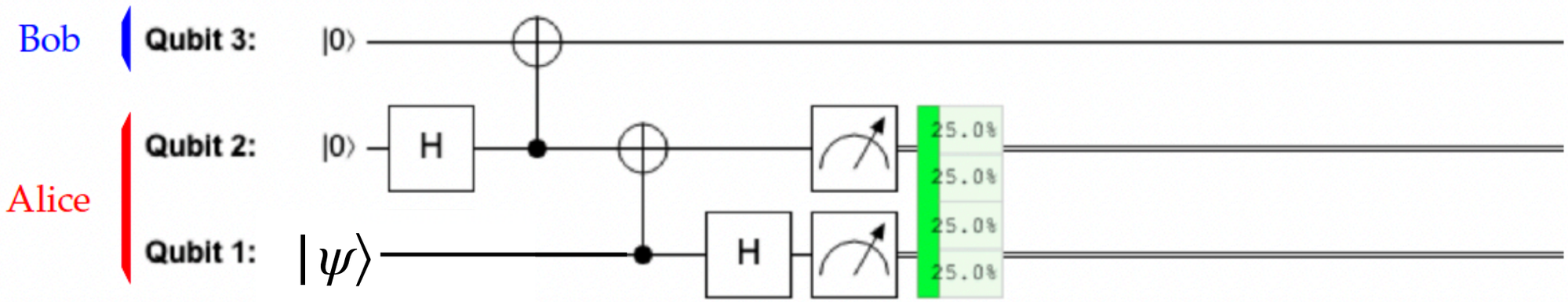


$$\hat{U}^{-1}_{\textbf{Bell}} := (\hat{H} \otimes \hat{1}) C\hat{\textbf{NOT}}_{1\to 2}$$

Will Alice einen anderen Zustand schicken, dann findet man dasselbe Messergebnis



Wdh.: Quanten-Teleportation



Der allgemeine Zustand vor Messung lautet $(\hat{H} \otimes \hat{1} \otimes \hat{1})(\hat{\text{CNOT}}_{1_2} \otimes \hat{1})\left(|\psi\rangle \otimes |\Phi^+\rangle\right)$

$$
\begin{aligned}
&(H \otimes I \otimes I)\,(\text{CNOT}_{1\to 2} \otimes I)\,(|\psi\rangle \otimes |\Phi^+\rangle)\\
&= \frac{1}{\sqrt{2}}\,(H \otimes I \otimes I)\,(\text{CNOT}_{1\to 2} \otimes I)\,(\psi_0\,|000\rangle + \psi_0\,|011\rangle + \psi_1\,|100\rangle + \psi_1\,|111\rangle)\\
&= \frac{1}{\sqrt{2}}\,(H \otimes I \otimes I)\,(\psi_0\,|000\rangle + \psi_0\,|011\rangle + \psi_1\,|110\rangle + \psi_1\,|101\rangle)\\
&= \frac{1}{2}\,(\psi_0\,|000\rangle + \psi_0\,|100\rangle + \psi_0\,|011\rangle + \psi_0\,|111\rangle + \psi_1\,|010\rangle - \psi_1\,|110\rangle + \psi_1\,|001\rangle - \psi_1\,|101\rangle)\\
&= |00\rangle \otimes \frac{\psi_0\,|0\rangle + \psi_1\,|1\rangle}{2} + |01\rangle \otimes \frac{\psi_1\,|0\rangle + \psi_0\,|1\rangle}{2}\\
&+ |10\rangle \otimes \frac{\psi_0\,|0\rangle - \psi_1\,|1\rangle}{2} + |11\rangle \otimes \frac{-\psi_1\,|0\rangle + \psi_0\,|1\rangle}{2}.
\end{aligned}
$$

Wdh.: Quanten-Teleportation

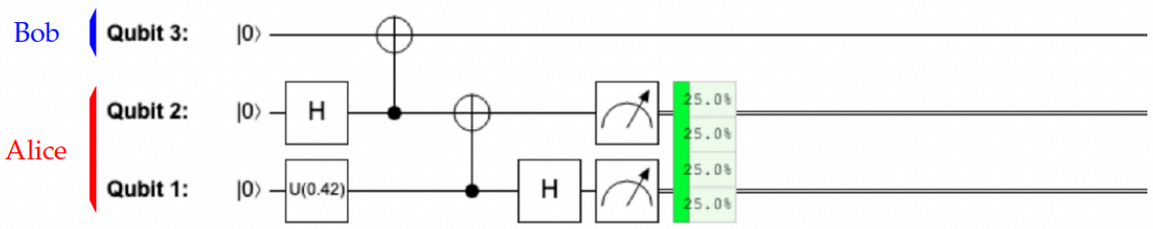
$$
\begin{aligned}
&(H \otimes I \otimes I)\,(\mathrm{CNOT}_{1\to 2} \otimes I)\,(|\psi\rangle \otimes |\Phi^+\rangle) \\
&= \frac{1}{\sqrt{2}}\,(H \otimes I \otimes I)\,(\mathrm{CNOT}_{1\to 2} \otimes I)\,(\psi_0\,|000\rangle + \psi_0\,|011\rangle + \psi_1\,|100\rangle + \psi_1\,|111\rangle) \\
&= \frac{1}{\sqrt{2}}\,(H \otimes I \otimes I)\,(\psi_0\,|000\rangle + \psi_0\,|011\rangle + \psi_1\,|110\rangle + \psi_1\,|101\rangle) \\
&= \frac{1}{2}\,(\psi_0\,|000\rangle + \psi_0\,|100\rangle + \psi_0\,|011\rangle + \psi_0\,|111\rangle + \psi_1\,|010\rangle - \psi_1\,|110\rangle + \psi_1\,|001\rangle - \psi_1\,|101\rangle) \\
&= |00\rangle \otimes \frac{\psi_0\,|0\rangle + \psi_1\,|1\rangle}{2} + |01\rangle \otimes \frac{\psi_1\,|0\rangle + \psi_0\,|1\rangle}{2} \\
&+ |10\rangle \otimes \frac{\psi_0\,|0\rangle - \psi_1\,|1\rangle}{2} + |11\rangle \otimes \frac{-\psi_1\,|0\rangle + \psi_0\,|1\rangle}{2}.
\end{aligned}
$$

Die Wahrscheinlichkeiten p_{ab}, das Ergebnis $|ab\rangle$ für die ersten beiden QuBits zu erhalten:

$$
\begin{aligned}
p_{00} &= \left(\frac{\psi_0}{2}\right)^2 + \left(\frac{\psi_1}{2}\right)^2 = \frac{\psi_0^2 + \psi_1^2}{4} = \frac{1}{4}, \\
p_{01} &= \left(\frac{\psi_1}{2}\right)^2 + \left(\frac{\psi_0}{2}\right)^2 = \frac{1}{4}, \\
p_{10} &= \left(\frac{\psi_0}{2}\right)^2 + \left(\frac{-\psi_1}{2}\right)^2 = \frac{1}{4}, \\
p_{11} &= \left(\frac{-\psi_1}{2}\right)^2 + \left(\frac{\psi_0}{2}\right)^2 = \frac{1}{4}.
\end{aligned}
$$

Wie von Quirky behauptet, tritt jedes Ergebnis mit $\frac{1}{4}$ auf

Wdh.: Quanten-Teleportation



$$
\begin{aligned}
&(H \otimes I \otimes I)\,(\mathrm{CNOT}_{1\to 2} \otimes I)\,\left(|\psi\rangle \otimes \left|\Phi^{+}\right\rangle\right) \\
&= \frac{1}{\sqrt{2}}\,(H \otimes I \otimes I)\,(\mathrm{CNOT}_{1\to 2} \otimes I)\,(\psi_0\,|000\rangle + \psi_0\,|011\rangle + \psi_1\,|100\rangle + \psi_1\,|111\rangle) \\
&= \frac{1}{\sqrt{2}}\,(H \otimes I \otimes I)\,(\psi_0\,|000\rangle + \psi_0\,|011\rangle + \psi_1\,|110\rangle + \psi_1\,|101\rangle) \\
&= \frac{1}{2}\,(\psi_0\,|000\rangle + \psi_0\,|100\rangle + \psi_0\,|011\rangle + \psi_0\,|111\rangle + \psi_1\,|010\rangle - \psi_1\,|110\rangle + \psi_1\,|001\rangle - \psi_1\,|101\rangle) \\
&= |00\rangle \otimes \frac{\psi_0\,|0\rangle + \psi_1\,|1\rangle}{2} + |01\rangle \otimes \frac{\psi_1\,|0\rangle + \psi_0\,|1\rangle}{2} \\
&+ |10\rangle \otimes \frac{\psi_0\,|0\rangle - \psi_1\,|1\rangle}{2} + |11\rangle \otimes \frac{-\psi_1\,|0\rangle + \psi_0\,|1\rangle}{2}.
\end{aligned}
$$

Bob's Zustand hängt nun vom Ergebnis der Messung von Alice ab:

Alice: 00 ⇒ Bob: $|\psi'_{00}\rangle = \psi_0\,|0\rangle + \psi_1\,|1\rangle$

Alice: 01 ⇒ Bob: $|\psi'_{01}\rangle = \psi_1\,|0\rangle + \psi_0\,|1\rangle$

Alice: 10 ⇒ Bob: $|\psi'_{10}\rangle = \psi_0\,|0\rangle - \psi_1\,|1\rangle$

Alice: 11 ⇒ Bob: $|\psi'_{11}\rangle = -\psi_1\,|0\rangle + \psi_0\,|1\rangle$

Alice sendet nun Bob ihr Ergebnis $[ab]$, damit weiss Bob was er mit seinem Zustand machen muss

Alice: 00 ⇒ Bob: $\hat{1}\,|\psi'_{00}\rangle = \psi_0\,|0\rangle + \psi_1\,|1\rangle$

Alice: 01 ⇒ Bob: $\mathrm{N\hat{O}T}\,|\psi'_{01}\rangle = \psi_0\,|0\rangle + \psi_1\,|1\rangle$

Alice: 10 ⇒ Bob: $\hat{Z}\,|\psi'_{10}\rangle = \psi_0\,|0\rangle + \psi_1\,|1\rangle$

Alice: 11 ⇒ Bob: $\hat{Z}\mathrm{N\hat{O}T}\,|\psi'_{11}\rangle = \psi_0\,|0\rangle + \psi_1\,|1\rangle$

Wdh.: Quanten-Teleportation

Alice sendet nun Bob ihr Ergebnis $[ab]$, damit weiss Bob was er mit seinem Zustand machen muss

Alice: 00 $\Rightarrow$ Bob: $\quad \hat{1}\,|\psi'_{00}\rangle = \psi_0|0\rangle + \psi_1|1\rangle$

Alice: 01 $\Rightarrow$ Bob: $\quad \text{NÔT}\,|\psi'_{01}\rangle = \psi_0|0\rangle + \psi_1|1\rangle$

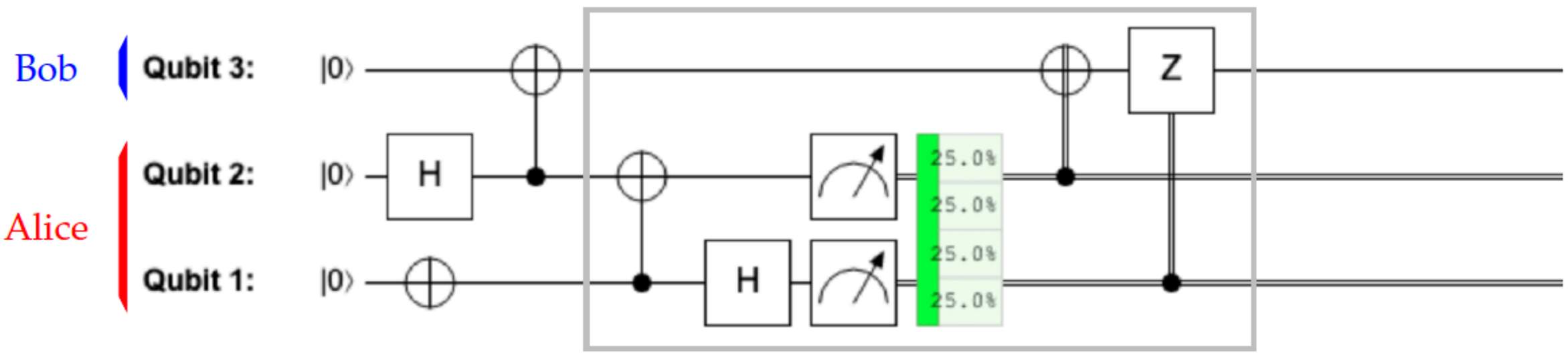
Alice: 10 $\Rightarrow$ Bob: $\quad \hat{Z}|\psi'_{10}\rangle = \psi_0|0\rangle + \psi_1|1\rangle$

Alice: 11 $\Rightarrow$ Bob: $\quad \hat{Z}\text{NÔT}\,|\psi'_{11}\rangle = \psi_0|0\rangle + \psi_1|1\rangle$

Diese vier Fälle können einfach zusammengefasst werden:

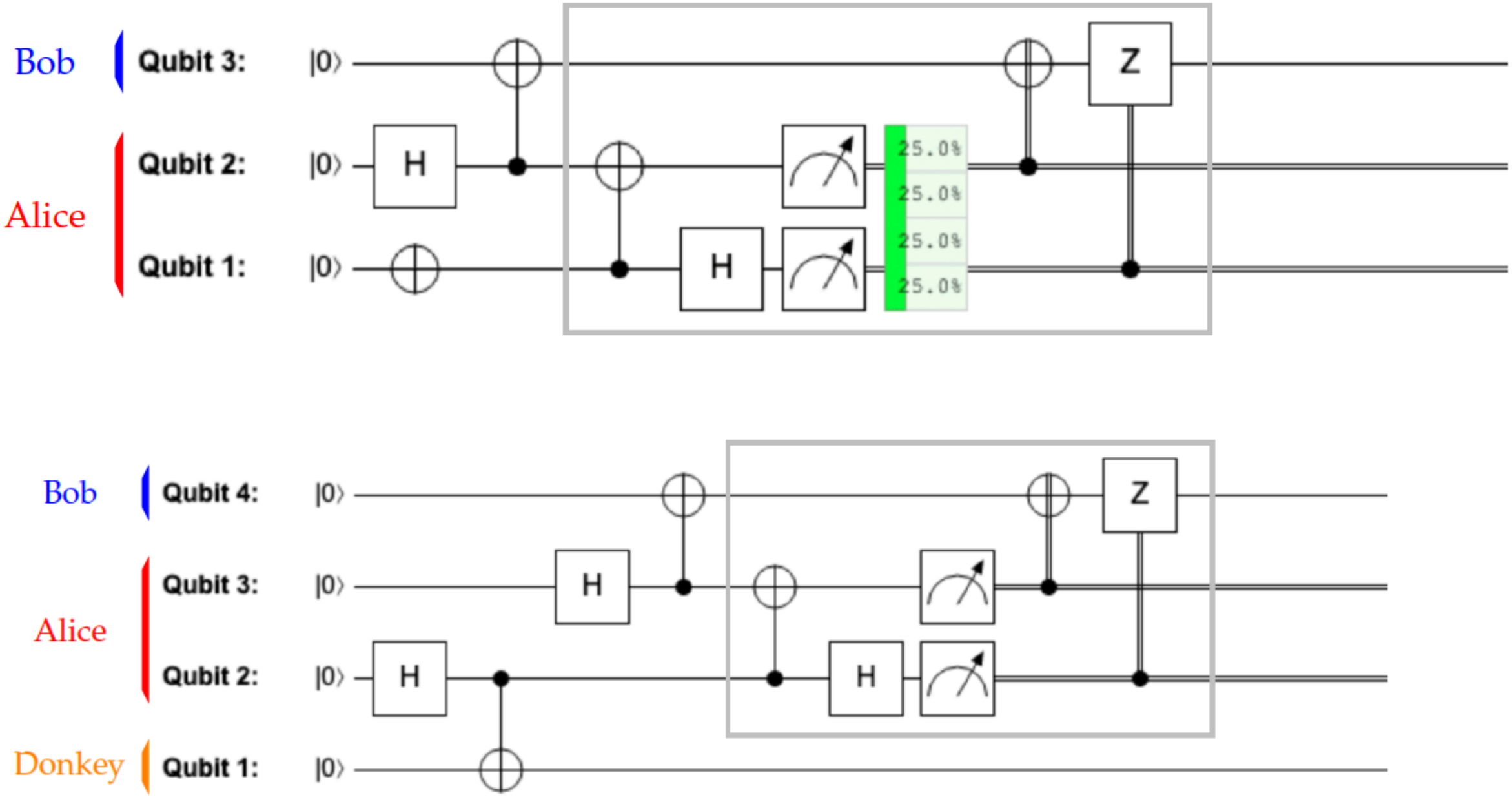
1. schaue das Bit b and und wenn $b = 1$, wende NOT an,
2. schaue das Bit a und wenn $a = 1$, wende Z an.

Teleportationsschaltkreis:



4.2.4 Ein Blick auf Quanten-Netzwerke

Wiederholte Quanten-Teleportation -> Quanteninternet…..



4.2.5 Die Unschärferelation

Heisenbergsche Unschärfe Relation….

Betrachten wir den Zustand $|\psi\rangle = \psi_0|0\rangle + \psi_1|1\rangle$

Hier sind nur die beiden Zustände $|0\rangle$ **und** $|1\rangle$ **mit keiner Ungewissheit (uncertainty) behaftet**

Führen wir vor der Messung z.B. eine Hadamard-Operation aus

Dann sind nur die Zustände $|+\rangle$ **und** $|-\rangle$ **mit keiner Ungewissheit behaftet, denn** $\hat{H}$ **macht daraus die beiden Zustände** $|0\rangle$ **und** $|1\rangle$

$$\hat{H}|\psi\rangle = \psi_0\hat{H}|0\rangle + \psi_1\hat{H}|1\rangle = \ldots = \frac{\psi_0+\psi_1}{\sqrt{2}}|0\rangle + \frac{\psi_0-\psi_1}{\sqrt{2}}|1\rangle$$

Jetzt finden wir mit $q_0 = \frac{(\psi_0+\psi_1)^2}{2}$ **den Zustand** $|0\rangle$ **und mit** $q_1 = \frac{(\psi_0-\psi_1)^2}{2}$ **den Zustand** $|1\rangle$

4.2.5 Die Unschärferelation

Heisenbergsche Unschärfe Relation….

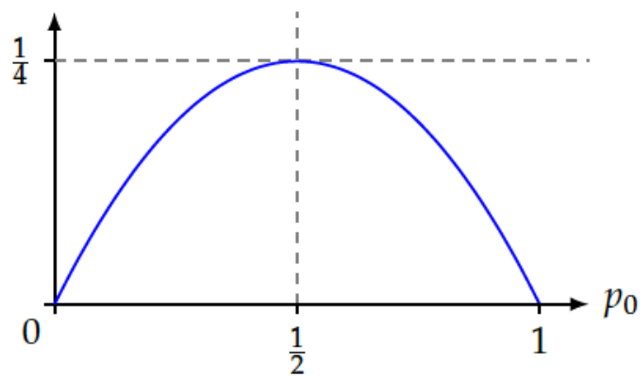
Messung

=> $|0\rangle$ **und** $|1\rangle$ **sind mit keiner Ungewissheit behaftet**
$|+\rangle$ **und** $|-\rangle$ **schon**

Messung

=> $|+\rangle$ **und** $|-\rangle$ **sind mit keiner Ungewissheit behaftet**
$|0\rangle$ **und** $|1\rangle$ **schon**

Mass für die Ungewissheit: $(p) = p_0(1 - p_0) = p_0 p_1$



Heisenbergsche Unschärfe Relation: $(p) + (q) = \frac{1}{4}$

Es gibt keinen Zustand für den beide Ungewissheiten gleich Null sind

5 Virtuose Algorithmen

Ziel: Quantencomputer, der (manche) Probleme (viel) schneller löst als jeder klassische Computer (benutzt klassische Physik, i.e. Elektrodynamik, und rechnet mit Bits)

5 Virtuose Algorithmen

Ziel: Quantencomputer, der (manche) Probleme (viel) schneller löst als jeder klassische Computer (benutzt klassische Physik, i.e. Elektrodynamik, und rechnet mit Bits)

Wie misst man die Geschwindigkeit eines Algorithmus?
Soll von expliziter Hardware unabhängig sein:

5 Virtuose Algorithmen

Ziel: Quantencomputer, der (manche) Probleme (viel) schneller löst als jeder klassische Computer (benutzt klassische Physik, i.e. Elektrodynamik, und rechnet mit Bits)

Wie misst man die Geschwindigkeit eines Algorithmus?
Soll von expliziter Hardware unabhängig sein:

Man zählt die Zahl der elementaren Operationen

5 Virtuose Algorithmen

Ziel: Quantencomputer, der (manche) Probleme (viel) schneller löst als jeder klassische Computer (benutzt klassische Physik, i.e. Elektrodynamik, und rechnet mit Bits)

Wie misst man die Geschwindigkeit eines Algorithmus?
Soll von expliziter Hardware unabhängig sein:

Man zählt die Zahl der elementaren Operationen

Genauer:
wie skaliert die Zahl der notwendigen elementaren Operationen mit der Größe des Problems (Komplexitätstheorie)

5 Virtuose Algorithmen

Ziel: Quantencomputer, der (manche) Probleme (viel) schneller löst als jeder klassische Computer (benutzt klassische Physik, i.e. Elektrodynamik, und rechnet mit Bits)

Wie misst man die Geschwindigkeit eines Algorithmus?
Soll von expliziter Hardware unabhängig sein:

Man zählt die Zahl der elementaren Operationen

Genauer:
wie skaliert die Zahl der notwendigen elementaren Operationen mit der Größe des Problems (Komplexitätstheorie)

Quantenalgorithmen:
- elementare Operation: Hadamard, CNOT, Messung…

5 Virtuose Algorithmen

Ziel: Quantencomputer, der (manche) Probleme (viel) schneller löst als jeder klassische Computer (benutzt klassische Physik, i.e. Elektrodynamik, und rechnet mit Bits)

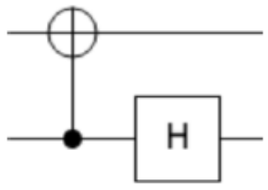
Wie misst man die Geschwindigkeit eines Algorithmus?
Soll von expliziter Hardware unabhängig sein:

Man zählt die Zahl der elementaren Operationen

Genauer:
wie skaliert die Zahl der notwendigen elementaren Operationen mit der Größe des Problems (Komplexitätstheorie)

Quantenalgorithmen:
- elementare Operation: Hadamard, CNOT, Messung…

Beispiel für
2 elementare Operationen




```
CNOT q1 -> q2;
H q1;
```

5.1 Mit Quantenorakeln sprechen

Es gibt verschiedene Arten von elementaren Operationen, die langsamsten sind mit einem Zugriff auf Daten (Speicher, Festplatte, Internet,..) verbunden

5.1 Mit Quantenorakeln sprechen

Es gibt verschiedene Arten von elementaren Operationen, die langsamsten sind mit einem Zugriff auf Daten (Speicher, Festplatte, Internet,..) verbunden

1. Abschätzung der Laufzeit: Zahl der Operationen mit Datenzugriff

5.1 Mit Quantenorakeln sprechen

Es gibt verschiedene Arten von elementaren Operationen, die langsamsten sind mit einem Zugriff auf Daten (Speicher, Festplatte, Internet,..) verbunden

1. Abschätzung der Laufzeit: Zahl der Operationen mit Datenzugriff

Orakel = Subroutine die Datenzugriff beschreibt (Daten auslesen, aus internet runterladen oder selber erzeugen…)

5.1 Mit Quantenorakeln sprechen

Es gibt verschiedene Arten von elementaren Operationen, die langsamsten sind mit einem Zugriff auf Daten (Speicher, Festplatte, Internet,..) verbunden

1. Abschätzung der Laufzeit: Zahl der Operationen mit Datenzugriff

Orakel = Subroutine die Datenzugriff beschreibt (Daten auslesen, aus internet runterladen oder selber erzeugen…)

Klassisches Orakel: Funktion $f : \{0,1\}^n \to \{0,1\}$

5.1 Mit Quantenorakeln sprechen

Es gibt verschiedene Arten von elementaren Operationen, die langsamsten sind mit einem Zugriff auf Daten (Speicher, Festplatte, Internet,..) verbunden

1. Abschätzung der Laufzeit: Zahl der Operationen mit Datenzugriff

Orakel = Subroutine die Datenzugriff beschreibt (Daten auslesen, aus internet runterladen oder selber erzeugen…)

Klassisches Orakel: Funktion $f : \{0,1\}^n \to \{0,1\}$

Eingabe $x \in \{0,1\}^n$ **= Menge aller Bitstrings der Länge** n **- entspricht der Frage**

5.1 Mit Quantenorakeln sprechen

Es gibt verschiedene Arten von elementaren Operationen, die langsamsten sind mit einem Zugriff auf Daten (Speicher, Festplatte, Internet,..) verbunden

1. Abschätzung der Laufzeit: Zahl der Operationen mit Datenzugriff

Orakel = Subroutine die Datenzugriff beschreibt (Daten auslesen, aus internet runterladen oder selber erzeugen…)

Klassisches Orakel: Funktion $f : \{0,1\}^n \to \{0,1\}$

Eingabe $x \in \{0,1\}^n$ **= Menge aller Bitstrings der Länge** n **- entspricht der Frage**

Ausgabe $f(x) \in \{0,1\}$ **- entspricht der Antwort**

5.1 Mit Quantenorakeln sprechen

Es gibt verschiedene Arten von elementaren Operationen, die langsamsten sind mit einem Zugriff auf Daten (Speicher, Festplatte, Internet,..) verbunden

1. Abschätzung der Laufzeit: Zahl der Operationen mit Datenzugriff

Orakel = Subroutine die Datenzugriff beschreibt (Daten auslesen, aus internet runterladen oder selber erzeugen…)

Klassisches Orakel: Funktion $f : \{0,1\}^n \to \{0,1\}$
Eingabe $x \in \{0,1\}^n$ **= Menge aller Bitstrings der Länge** n **- entspricht der Frage**
Ausgabe $f(x) \in \{0,1\}$ **- entspricht der Antwort**

Beispiel: 4 Bit Speicher kann durch $f : \{0,1\}^2 \to \{0,1\}$ **modelliert werden**

5.1 Mit Quantenorakeln sprechen

Es gibt verschiedene Arten von elementaren Operationen, die langsamsten sind mit einem Zugriff auf Daten (Speicher, Festplatte, Internet,..) verbunden

1. Abschätzung der Laufzeit: Zahl der Operationen mit Datenzugriff

Orakel = Subroutine die Datenzugriff beschreibt (Daten auslesen, aus internet runterladen oder selber erzeugen…)

Klassisches Orakel: Funktion $f : \{0,1\}^n \to \{0,1\}$

Eingabe $x \in \{0,1\}^n$ **= Menge aller Bitstrings der Länge** n **- entspricht der Frage**

Ausgabe $f(x) \in \{0,1\}$ **- entspricht der Antwort**

Beispiel: 4 Bit Speicher kann durch $f : \{0,1\}^2 \to \{0,1\}$ **modelliert werden**

Um alle vier Bits rauszukriegen, muss man f **viermal anwenden, und man erhält die vier Bitwerte** $f(0,0), f(0,1), f(1,0)$ **und** $f(1,1)$**.**

5.1 Mit Quantenorakeln sprechen

Es gibt verschiedene Arten von elementaren Operationen, die langsamsten sind mit einem Zugriff auf Daten (Speicher, Festplatte, Internet,..) verbunden

1. Abschätzung der Laufzeit: Zahl der Operationen mit Datenzugriff

Orakel = Subroutine die Datenzugriff beschreibt (Daten auslesen, aus internet runterladen oder selber erzeugen…)

Klassisches Orakel: Funktion $f : \{0,1\}^n \to \{0,1\}$

Eingabe $x \in \{0,1\}^n$ **= Menge aller Bitstrings der Länge** n **- entspricht der Frage**

Ausgabe $f(x) \in \{0,1\}$ **- entspricht der Antwort**

Beispiel: 4 Bit Speicher kann durch $f : \{0,1\}^2 \to \{0,1\}$ **modelliert werden**

Um alle vier Bits rauszukriegen, muss man f **viermal anwenden, und man erhält die vier Bitwerte** $f(0,0), f(0,1), f(1,0)$ **und** $f(1,1)$**.**

Wichtig: uns interessiert nicht der Wert von f **für eine bestimmte Eingabe**

5.1 Mit Quantenorakeln sprechen

Es gibt verschiedene Arten von elementaren Operationen, die langsamsten sind mit einem Zugriff auf Daten (Speicher, Festplatte, Internet,..) verbunden

1. Abschätzung der Laufzeit: Zahl der Operationen mit Datenzugriff

Orakel = Subroutine die Datenzugriff beschreibt (Daten auslesen, aus internet runterladen oder selber erzeugen…)

Klassisches Orakel: Funktion $f : \{0,1\}^n \to \{0,1\}$

Eingabe $x \in \{0,1\}^n$ **= Menge aller Bitstrings der Länge** n **- entspricht der Frage**

Ausgabe $f(x) \in \{0,1\}$ **- entspricht der Antwort**

Beispiel: 4 Bit Speicher kann durch $f : \{0,1\}^2 \to \{0,1\}$ **modelliert werden**

Um alle vier Bits rauszukriegen, muss man f **viermal anwenden, und man erhält die vier Bitwerte** $f(0,0), f(0,1), f(1,0)$ **und** $f(1,1)$**.**

Wichtig: uns interessiert nicht der Wert von f **für eine bestimmte Eingabe**

Wir sind an den Eigenschaften von f **interessiert, wobei wir** f **so selten wie möglich auswerten wollen**

5.1 Mit Quantenorakeln sprechen

Es gibt verschiedene Arten von elementaren Operationen, die langsamsten sind mit einem Zugriff auf Daten (Speicher, Festplatte, Internet,..) verbunden

1. Abschätzung der Laufzeit: Zahl der Operationen mit Datenzugriff

Orakel = Subroutine die Datenzugriff beschreibt (Daten auslesen, aus internet runterladen oder selber erzeugen…)

Klassisches Orakel: Funktion $f : \{0,1\}^n \to \{0,1\}$

Eingabe $x \in \{0,1\}^n$ **= Menge aller Bitstrings der Länge** n **- entspricht der Frage**

Ausgabe $f(x) \in \{0,1\}$ **- entspricht der Antwort**

Beispiel: 4 Bit Speicher kann durch $f : \{0,1\}^2 \to \{0,1\}$ **modelliert werden**

Um alle vier Bits rauszukriegen, muss man f **viermal anwenden, und man erhält die vier Bitwerte** $f(0,0), f(0,1), f(1,0)$ **und** $f(1,1)$**.**

Wichtig: uns interessiert nicht der Wert von f **für eine bestimmte Eingabe**

Wir sind an den Eigenschaften von f **interessiert, wobei wir** f **so selten wie möglich auswerten wollen**

Beispiel: wir wollen wissen, ob $f(x) = 0$ **für alle** $x \in \{0,1\}^n$

5.1 Mit Quantenorakeln sprechen

Es gibt verschiedene Arten von elementaren Operationen, die langsamsten sind mit einem Zugriff auf Daten (Speicher, Festplatte, Internet,..) verbunden

1. Abschätzung der Laufzeit: Zahl der Operationen mit Datenzugriff

Orakel = Subroutine die Datenzugriff beschreibt (Daten auslesen, aus internet runterladen oder selber erzeugen…)

Klassisches Orakel: Funktion $f : \{0,1\}^n \to \{0,1\}$
Eingabe $x \in \{0,1\}^n$ **= Menge aller Bitstrings der Länge** n **- entspricht der Frage**
Ausgabe $f(x) \in \{0,1\}$ **- entspricht der Antwort**

Beispiel: 4 Bit Speicher kann durch $f : \{0,1\}^2 \to \{0,1\}$ **modelliert werden**
Um alle vier Bits rauszukriegen, muss man f **viermal anwenden, und man erhält die vier Bitwerte** $f(0,0), f(0,1), f(1,0)$ **und** $f(1,1)$**.**

Wichtig: uns interessiert nicht der Wert von f **für eine bestimmte Eingabe**
Wir sind an den Eigenschaften von f **interessiert, wobei wir** f **so selten wie möglich auswerten wollen**

Beispiel: wir wollen wissen, ob $f(x) = 0$ **für alle** $x \in \{0,1\}^n$
Mögliche Lösung: Frage Orakel für zufällige Werte von x**, bis wir** $f(x) = 1$ **finden**

5.1.1 Umkehrbare Berechnungen

Können wir auf Quantencomputern alles machen, was man auch auf klassischen Computern machen kann?

5.1.1 Umkehrbare Berechnungen

Können wir auf Quantencomputern alles machen, was man auch auf klassischen Computern machen kann?

Beachte: Quantenoperationen müssen invertierter sein!
Das gilt nicht bei klassischen Operationen!

5.1.1 Umkehrbare Berechnungen

Können wir auf Quantencomputern alles machen, was man auch auf klassischen Computern machen kann?

Beachte: Quantenoperationen müssen invertierter sein!
Das gilt nicht bei klassischen Operationen!

Zeige: jede Berechnung kann umkehrbar gemacht werden

5.1.1 Umkehrbare Berechnungen

Können wir auf Quantencomputern alles machen, was man auch auf klassischen Computern machen kann?

Beachte: Quantenoperationen müssen invertierter sein!
Das gilt nicht bei klassischen Operationen!

Zeige: jede Berechnung kann umkehrbar gemacht werden

Beispiel für unumkehrbar klassische Operation:

x_1	x_2	$\mathrm{AND}(x_1, x_2)$
0	0	0
0	1	0
1	0	0
1	1	1

5.1.1 Umkehrbare Berechnungen

Können wir auf Quantencomputern alles machen, was man auch auf klassischen Computern machen kann?

Beachte: Quantenoperationen müssen invertierter sein!
Das gilt nicht bei klassischen Operationen!

Zeige: jede Berechnung kann umkehrbar gemacht werden

Beispiel für unumkehrbar klassische Operation:

x_1	x_2	$\mathrm{AND}(x_1, x_2)$
0	0	0
0	1	0
1	0	0
1	1	1

$$[x_1, x_2] \mapsto [\mathrm{AND}(x_1, x_2)].$$

5.1.1 Umkehrbare Berechnungen

Können wir auf Quantencomputern alles machen, was man auch auf klassischen Computern machen kann?

Beachte: Quantenoperationen müssen invertierter sein!
Das gilt nicht bei klassischen Operationen!

Zeige: jede Berechnung kann umkehrbar gemacht werden

Beispiel für unumkehrbar klassische Operation:

x_1	x_2	$\mathrm{AND}(x_1, x_2)$
0	0	0
0	1	0
1	0	0
1	1	1

$$[x_1, x_2] \mapsto [\mathrm{AND}(x_1, x_2)].$$

2 Eingangsbit und 1 Ausgangsbit: das kann niemals umkehrbar sein

5.1.1 Umkehrbare Berechnungen

Nächster Versuch: 2 Eingangsbits und 2 Ausgangsbits

5.1.1 Umkehrbare Berechnungen

Nächster Versuch: 2 Eingangsbits und 2 Ausgangsbits

$$[x_1, x_2] \mapsto [x_1, \mathrm{AND}(x_1, x_2)].$$

5.1.1 Umkehrbare Berechnungen

Nächster Versuch: 2 Eingangsbits und 2 Ausgangsbits

$$[x_1, x_2] \mapsto [x_1, \mathrm{AND}(x_1, x_2)].$$

Schon besser, aber 00 und 01 führen zum selben Ergebnis 00…

5.1.1 Umkehrbare Berechnungen

Nächster Versuch: 2 Eingangsbits und 2 Ausgangsbits

$$[x_1, x_2] \mapsto [x_1, \mathrm{AND}(x_1, x_2)].$$

Schon besser, aber 00 und 01 führen zum selben Ergebnis 00…

5.1.1 Umkehrbare Berechnungen

Nächster Versuch: 2 Eingangsbits und 2 Ausgangsbits

$$[x_1, x_2] \mapsto [x_1, \mathrm{AND}(x_1, x_2)].$$

Schon besser, aber 00 und 01 führen zum selben Ergebnis 00…

Erweiterung auf 3 Bits:

5.1.1 Umkehrbare Berechnungen

Nächster Versuch: 2 Eingangsbits und 2 Ausgangsbits

$$[x_1, x_2] \mapsto [x_1, \mathrm{AND}(x_1, x_2)].$$

Schon besser, aber 00 und 01 führen zum selben Ergebnis 00…

Erweiterung auf 3 Bits:

$$[x_1, x_2, y] \mapsto [x_1, x_2, y \oplus \mathrm{AND}(x_1, x_2)],$$

5.1.1 Umkehrbare Berechnungen

Nächster Versuch: 2 Eingangsbits und 2 Ausgangsbits

$$[x_1, x_2] \mapsto [x_1, \mathrm{AND}(x_1, x_2)].$$

Schon besser, aber 00 und 01 führen zum selben Ergebnis 00…

Erweiterung auf 3 Bits:

$$[x_1, x_2, y] \mapsto [x_1, x_2, y \oplus \mathrm{AND}(x_1, x_2)],$$

x_1	x_2	y	x_1	x_2	$y \oplus (x_1 \& x_2)$
0	0	0	0	0	0
0	1	0	0	1	0
1	0	0	1	0	0
1	1	0	1	1	1
0	0	1	0	0	1
0	1	1	0	1	1
1	0	1	1	0	1
1	1	1	1	1	0

5.1.1 Umkehrbare Berechnungen

Die Abbildung $[x_1, x_2, y] \mapsto [x_1, x_2, y \oplus \mathrm{AND}(x_1, x_2)],$

x_1	x_2	y	x_1	x_2	$y \oplus (x_1 \& x_2)$
0	0	0	0	0	0
0	1	0	0	1	0
1	0	0	1	0	0
1	1	0	1	1	1
0	0	1	0	0	1
0	1	1	0	1	1
1	0	1	1	0	1
1	1	1	1	1	0

ist gleichzeitig ihre eigene Inverse

5.1.1 Umkehrbare Berechnungen

Die Abbildung $[x_1, x_2, y] \mapsto [x_1, x_2, y \oplus \mathrm{AND}(x_1, x_2)]$,

x_1	x_2	y	x_1	x_2	$y \oplus (x_1 \& x_2)$
0	0	0	0	0	0
0	1	0	0	1	0
1	0	0	1	0	0
1	1	0	1	1	1
0	0	1	0	0	1
0	1	1	0	1	1
1	0	1	1	0	1
1	1	1	1	1	0

ist gleichzeitig ihre eigene Inverse

$$
\begin{aligned}
[x_1, x_2, y] &\mapsto [x_1, x_2, y \oplus \mathrm{AND}(x_1, x_2)] \\
&\mapsto [x_1, x_2, y \oplus \mathrm{AND}(x_1, x_2) \oplus \mathrm{AND}(x_1, x_2)] = [x_1, x_2, y].
\end{aligned}
$$

5.1.1 Umkehrbare Berechnungen

Die Abbildung $[x_1, x_2, y] \mapsto [x_1, x_2, y \oplus \mathrm{AND}(x_1, x_2)]$,

x_1	x_2	y	x_1	x_2	$y \oplus (x_1 \& x_2)$
0	0	0	0	0	0
0	1	0	0	1	0
1	0	0	1	0	0
1	1	0	1	1	1
0	0	1	0	0	1
0	1	1	0	1	1
1	0	1	1	0	1
1	1	1	1	1	0

ist gleichzeitig ihre eigene Inverse

$$
\begin{aligned}
[x_1, x_2, y] &\mapsto [x_1, x_2, y \oplus \mathrm{AND}(x_1, x_2)] \\
&\mapsto [x_1, x_2, y \oplus \mathrm{AND}(x_1, x_2) \oplus \mathrm{AND}(x_1, x_2)] = [x_1, x_2, y].
\end{aligned}
$$

Es gibt also umkehrbare Erweiterungen der AND-Operation

5.1.2 Bit-Orakel

Obiger Ansatz zur Erweiterung einer nicht-reversiblen Operation auf eine reversible funktioniert nicht nur für die AND Operation, sondern für jede beliebige Funktion

$$f : \{0,1\}^n \rightarrow \{0,1\}$$

5.1.2 Bit-Orakel

Obiger Ansatz zur Erweiterung einer nicht-reversiblen Operation auf eine reversible funktioniert nicht nur für die AND Operation, sondern für jede beliebige Funktion

$$f : \{0,1\}^n \rightarrow \{0,1\}$$

Wir erweitern die n Eingangs-Bits auf $n + 1$ und definieren

5.1.2 Bit-Orakel

Obiger Ansatz zur Erweiterung einer nicht-reversiblen Operation auf eine reversible funktioniert nicht nur für die AND Operation, sondern für jede beliebige Funktion

$$f : \{0,1\}^n \to \{0,1\}$$

Wir erweitern die n Eingangs-Bits auf $n+1$ und definieren

$$[x_1, \ldots, x_n, y] \mapsto [x_1, \ldots, x_n, y \oplus f(x_1, \ldots, x_n)]$$

5.1.2 Bit-Orakel

Obiger Ansatz zur Erweiterung einer nicht-reversiblen Operation auf eine reversible funktioniert nicht nur für die AND Operation, sondern für jede beliebige Funktion

$$f : \{0,1\}^n \to \{0,1\}$$

Wir erweitern die n Eingangs-Bits auf $n+1$ und definieren

$$[x_1, \ldots, x_n, y] \mapsto [x_1, \ldots, x_n, y \oplus f(x_1, \ldots, x_n)]$$

Diese Operation ist invertierbar (wieder die eigene Inverse)

5.1.2 Bit-Orakel

Obiger Ansatz zur Erweiterung einer nicht-reversiblen Operation auf eine reversible funktioniert nicht nur für die AND Operation, sondern für jede beliebige Funktion

$$f : \{0,1\}^n \rightarrow \{0,1\}$$

Wir erweitern die n Eingangs-Bits auf $n+1$ und definieren

$$[x_1, \ldots, x_n, y] \mapsto [x_1, \ldots, x_n, y \oplus f(x_1, \ldots, x_n)]$$

Diese Operation ist invertierbar (wieder die eigene Inverse)

D.h.: jede Berechnung, die auf einem klassischen Computer läuft, kann prinzipiell auch auf einem Quanten-Computer laufen

5.1.2 Bit-Orakel

Quantenversion von

$$[x_1, \ldots, x_n, y] \mapsto [x_1, \ldots, x_n, y \oplus f(x_1, \ldots, x_n)]$$

lautet:

5.1.2 Bit-Orakel

Quantenversion von

$$[x_1, \ldots, x_n, y] \mapsto [x_1, \ldots, x_n, y \oplus f(x_1, \ldots, x_n)]$$

lautet:

$$U_f \, |x_1, \ldots, x_n, y\rangle = |x_1, \ldots, x_n, y \oplus f(x_1, \ldots, x_n)\rangle$$

5.1.2 Bit-Orakel

Quantenversion von

$$[x_1, \ldots, x_n, y] \mapsto [x_1, \ldots, x_n, y \oplus f(x_1, \ldots, x_n)]$$

lautet:

$$U_f \, |x_1, \ldots, x_n, y\rangle = |x_1, \ldots, x_n, y \oplus f(x_1, \ldots, x_n)\rangle$$

Wegen der Linearität reicht es aus, zu betrachten was $\hat{\mathcal{U}}_f$ mit den Basis-Zuständen macht: Permutation der Basiszustände

5.1.2 Bit-Orakel

Quantenversion von

$$[x_1, \ldots, x_n, y] \mapsto [x_1, \ldots, x_n, y \oplus f(x_1, \ldots, x_n)]$$

lautet:

$$U_f \, |x_1, \ldots, x_n, y\rangle = |x_1, \ldots, x_n, y \oplus f(x_1, \ldots, x_n)\rangle$$

Wegen der Linearität reicht es aus, zu betrachten was $\hat{\mathcal{U}}_f$ mit den Basis-Zuständen macht: Permutation der Basiszustände

Übungsaufgabe 4.4: Toffoli

Definiere die Toffoli-Operation auf drei Qubits durch

$$T \, |a, b, c\rangle = |a, b, c \oplus ab\rangle$$

auf Basiszuständen (ab ist dabei das Produkt der zwei Bits $a, b \in \{0, 1\}$, und $\oplus$ wurde in Gl. (3.20) definiert), und erweitere sie durch Linearität auf beliebige Drei-Qubit-Zustände. Zeige, dass T alle Quantenzustände auf Quantenzustände abbildet, und dass T invertierbar ist.

Bemerkung: T invertiert das dritte Bit genau dann, wenn beide ersten Bits beide eins sind – es ist also eine "zweifach-kontrollierte"-NOT-Operation.

5.1.2 Bit-Orakel

Quantenversion von

$$[x_1, \ldots, x_n, y] \mapsto [x_1, \ldots, x_n, y \oplus f(x_1, \ldots, x_n)]$$

lautet:

$$U_f \, |x_1, \ldots, x_n, y\rangle = |x_1, \ldots, x_n, y \oplus f(x_1, \ldots, x_n)\rangle$$

Wegen der Linearität reicht es aus, zu betrachten was $\hat{\mathcal{U}}_f$ mit den Basis-Zuständen macht: Permutation der Basiszustände

Übungsaufgabe 4.4: Toffoli

Definiere die Toffoli-Operation auf drei Qubits durch

$$T \, |a, b, c\rangle = |a, b, c \oplus ab\rangle$$

auf Basiszuständen (ab ist dabei das Produkt der zwei Bits $a, b \in \{0, 1\}$, und $\oplus$ wurde in Gl. (3.20) definiert), und erweitere sie durch Linearität auf beliebige Drei-Qubit-Zustände. Zeige, dass T alle Quantenzustände auf Quantenzustände abbildet, und dass T invertierbar ist.

Bemerkung: T invertiert das dritte Bit genau dann, wenn beide ersten Bits beide eins sind – es ist also eine "zweifach-kontrollierte"-NOT-Operation.

Die Quantenoperation $\hat{\mathcal{U}}_f$ heisst das Bit-Orakel für f

Name: "allmächtiges" Orakel gibt uns den Wert der Funktion für beliebige Eingaben….

Uns interessiert: wie oft müssen wir das Orakel befragen, um Eigenschaften von f zu bestimmen?

5.1.2 Bit-Orakel

Vergleiche: ‘Rate meine Zahl’-Spiel

- **Freund/Freundin denkt sich Zahl X aus**
- **Frage: Ist Deine Zahl X?**
- **Antwort: Ja/Nein**

Wie kann man die Zahl der Fragen minimieren?

5.1.2 Bit-Orakel

Vergleiche: ‘Rate meine Zahl’-Spiel

- **Freund/Freundin denkt sich Zahl X aus**
- **Frage: Ist Deine Zahl X?**
- **Antwort: Ja/Nein**

Wie kann man die Zahl der Fragen minimieren?

Beispiel 1: $f(x_1, x_2) = AND(x_1, x_2) = x_1 x_2$

5.1.2 Bit-Orakel

Vergleiche: ‘Rate meine Zahl’-Spiel

- **Freund/Freundin denkt sich Zahl X aus**
- **Frage: Ist Deine Zahl X?**
- **Antwort: Ja/Nein**

Wie kann man die Zahl der Fragen minimieren?

Beispiel 1: $f(x_1, x_2) = AND(x_1, x_2) = x_1 x_2$

Bit-Orakel:

5.1.2 Bit-Orakel

Vergleiche: ‘Rate meine Zahl’-Spiel

- **Freund/Freundin denkt sich Zahl X aus**
- **Frage: Ist Deine Zahl X?**
- **Antwort: Ja/Nein**

Wie kann man die Zahl der Fragen minimieren?

Beispiel 1: $f(x_1, x_2) = AND(x_1, x_2) = x_1 x_2$

Bit-Orakel: $U_{\mathrm{AND}} |a, b, c\rangle = |a, b, c \oplus ab\rangle$

5.1.2 Bit-Orakel

Vergleiche: ‘Rate meine Zahl’-Spiel

- **Freund/Freundin denkt sich Zahl X aus**
- **Frage: Ist Deine Zahl X?**
- **Antwort: Ja/Nein**

Wie kann man die Zahl der Fragen minimieren?

Beispiel 1: $f(x_1, x_2) = AND(x_1, x_2) = x_1 x_2$

Bit-Orakel: $U_{\mathrm{AND}} |a, b, c\rangle = |a, b, c \oplus ab\rangle$ **Das ist das Toffoli-Gate!**

5.1.2 Bit-Orakel

Vergleiche: ‘Rate meine Zahl’-Spiel

- **Freund/Freundin denkt sich Zahl X aus**
- **Frage: Ist Deine Zahl X?**
- **Antwort: Ja/Nein**

Wie kann man die Zahl der Fragen minimieren?

Beispiel 1: $f(x_1, x_2) = AND(x_1, x_2) = x_1 x_2$

Bit-Orakel: $U_{\mathrm{AND}} |a, b, c\rangle = |a, b, c \oplus ab\rangle$ **Das ist das Toffoli-Gate!**

Beispiel 2 : $f(x) = x$

Bit-Orakel:

5.1.2 Bit-Orakel

Vergleiche: ‘Rate meine Zahl’-Spiel

- **Freund/Freundin denkt sich Zahl X aus**
- **Frage: Ist Deine Zahl X?**
- **Antwort: Ja/Nein**

Wie kann man die Zahl der Fragen minimieren?

Beispiel 1: $f(x_1, x_2) = AND(x_1, x_2) = x_1 x_2$

Bit-Orakel: $U_{\mathrm{AND}} |a, b, c\rangle = |a, b, c \oplus ab\rangle$ **Das ist das Toffoli-Gate!**

Beispiel 2 : $f(x) = x$

Bit-Orakel: $U_f |a, b\rangle = |a, b \oplus a\rangle .$

5.1.2 Bit-Orakel

Vergleiche: ‘Rate meine Zahl’-Spiel

- **Freund/Freundin denkt sich Zahl X aus**
- **Frage: Ist Deine Zahl X?**
- **Antwort: Ja/Nein**

Wie kann man die Zahl der Fragen minimieren?

Beispiel 1: $f(x_1, x_2) = AND(x_1, x_2) = x_1 x_2$

Bit-Orakel: $U_{\mathrm{AND}} |a, b, c\rangle = |a, b, c \oplus ab\rangle$ **Das ist das Toffoli-Gate!**

Beispiel 2 : $f(x) = x$

Bit-Orakel: $U_f |a, b\rangle = |a, b \oplus a\rangle .$ **Das ist das CNOT-Gate!**

5.1.2 Bit-Orakel

Das Konzept Bit-Orakel beinhaltet also mehrere Quanten-Operationen, die wir zuvor per Hand eingeführt hatten

Übungsaufgabe 5.1: Bit-Orakel für Ein-Bit-Funktionen

Sei $f : \{0,1\} \to \{0,1\}$ eine Funktion mit einem einzelnen Eingabe- und Ausgabe-Bit. Eine solche Funktion ist vollständig durch die Werte $f(0), f(1) \in \{0,1\}$ definiert. Das sind zwei Bits, also gibt es genau vier solcher Funktionen. Wir haben gerade besprochen, wie man die das Bit-Orakel U_f für die Funktion $f(x) = x$ implementiert. Kannst du die Bit-Orakel U_f für die anderen drei Funktionen in QUIRKY implementieren?

Lösung Übungsaufgabe 5.1

Die anderen drei Funktionen sind $f = \text{NOT}$ sowie die zwei konstanten Funktionen $f(0) = f(1) = 0$ und $f(0) = f(1) = 1$.

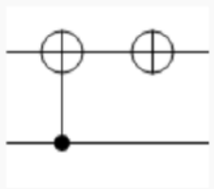
- Für die NOT-Funktion gilt:

$$U_{\text{NOT}} |a,b\rangle = |a, b \oplus \text{NOT}(a)\rangle = |a, b \oplus a \oplus 1\rangle = |a, \text{NOT}(b \oplus a)\rangle ,$$

was wie als Zusammensetzung einer kontrollierten-NOT-Operation und einer NOT-Operation auf dem zweiten Qubit geschrieben werden kann, also,

$$U_{\text{NOT}} = (I \otimes \text{NOT})\, \text{CNOT}_{1\to 2}.$$

In QUIRKY sieht das dann so aus:



- Für die Alles-Null-Funktion $f(0) = f(1) = 0$ gilt:

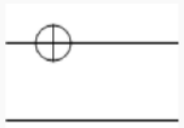
$$U_f |a,b\rangle = |a,b\rangle ,$$

also müssen wir gar nichts machen:

- Für die Alles-Eins-Funktion $f(0) = f(1) = 1$ gilt:

$$U_f |a,b\rangle = |a, b \oplus 1\rangle = |a, \text{NOT}(b)\rangle ,$$

also müssen wir nur das zweite Qubit invertieren:



5.1.3 Phasen-Orakel

Das Bit-Orakel kann nicht nur auf Basiszustände, sondern auch auf allgemeine Zustände angewandt werden.

5.1.3 Phasen-Orakel

Das Bit-Orakel kann nicht nur auf Basiszustände, sondern auch auf allgemeine Zustände angewandt werden.

Frage: Was passiert, wenn wir das letzte Register auf $|-\rangle$ statt 0 oder 1 setzen?

5.1.3 Phasen-Orakel

Das Bit-Orakel kann nicht nur auf Basiszustände, sondern auch auf allgemeine Zustände angewandt werden.

Frage: Was passiert, wenn wir das letzte Register auf $|-\rangle$ statt 0 oder 1 setzen?

Beachte: $\hat{\text{NOT}}|-\rangle = -|-\rangle$ statt 0 oder 1 setzen?

5.1.3 Phasen-Orakel

Das Bit-Orakel kann nicht nur auf Basiszustände, sondern auch auf allgemeine Zustände angewandt werden.

Frage: Was passiert, wenn wir das letzte Register auf $|-\rangle$ statt 0 oder 1 setzen?

Beachte: $\hat{\text{NOT}}|-\rangle = -|-\rangle$ statt 0 oder 1 setzen?

$$
\begin{aligned}
& U_f\left(|x_1, \ldots, x_n\rangle \otimes |-\rangle\right) \\
&= U_f\left(\frac{1}{\sqrt{2}}|x_1, \ldots, x_n, 0\rangle - \frac{1}{\sqrt{2}}|x_1, \ldots, x_n, 1\rangle\right) \\
&= \frac{1}{\sqrt{2}}|x_1, \ldots, x_n, f(x_1, \ldots, x_n)\rangle - \frac{1}{\sqrt{2}}|x_1, \ldots, x_n, f(x_1, \ldots, x_n) \oplus 1\rangle \\
&= |x_1, \ldots, x_n\rangle \otimes \frac{1}{\sqrt{2}}\left(|f(x_1, \ldots, x_n)\rangle - |f(x_1, \ldots, x_n) \oplus 1\rangle\right) \\
&= (-1)^{f(x_1, \ldots, x_n)}|x_1, \ldots, x_n\rangle \otimes |-\rangle .
\end{aligned}
$$

5.1.3 Phasen-Orakel

Das Bit-Orakel kann nicht nur auf Basiszustände, sondern auch auf allgemeine Zustände angewandt werden.

Frage: Was passiert, wenn wir das letzte Register auf $|-\rangle$ statt 0 oder 1 setzen?

Beachte: $\hat{\text{NOT}}|-\rangle = -|-\rangle$ statt 0 oder 1 setzen?

$$
\begin{aligned}
& U_f\left(|x_1, \ldots, x_n\rangle \otimes|-\rangle\right) \\
&= U_f\left(\frac{1}{\sqrt{2}}|x_1, \ldots, x_n, 0\rangle-\frac{1}{\sqrt{2}}|x_1, \ldots, x_n, 1\rangle\right) \\
&= \frac{1}{\sqrt{2}}|x_1, \ldots, x_n, f(x_1, \ldots, x_n)\rangle-\frac{1}{\sqrt{2}}|x_1, \ldots, x_n, f(x_1, \ldots, x_n) \oplus 1\rangle \\
&= |x_1, \ldots, x_n\rangle \otimes \frac{1}{\sqrt{2}}\left(|f(x_1, \ldots, x_n)\rangle-|f(x_1, \ldots, x_n) \oplus 1\rangle\right) \\
&= (-1)^{f(x_1, \ldots, x_n)}|x_1, \ldots, x_n\rangle \otimes|-\rangle .
\end{aligned}
$$

$f(x_1, \ldots, x_n) = 1 \Rightarrow -1$ $\qquad$ $f(x_1, \ldots, x_n) = 0 \Rightarrow +1$

5.1.3 Phasen-Orakel

Somit hat sich das $n+1$-te QuBit im Zustand $|-\rangle$ einfach reproduziert und die ersten n QuBits transformieren sich wie folgt:

$$O_f \left|x_1, \ldots, x_n\right\rangle = (-1)^{f(x_1,\ldots,x_n)} \left|x_1, \ldots, x_n\right\rangle ,$$

5.1.3 Phasen-Orakel

Somit hat sich das $n+1$-te QuBit im Zustand $|-\rangle$ einfach reproduziert und die ersten n QuBits transformieren sich wie folgt:

$$O_f \left|x_1, \ldots, x_n\right\rangle = (-1)^{f(x_1,\ldots,x_n)} \left|x_1, \ldots, x_n\right\rangle ,$$

$\hat{\mathcal{O}}_f$ heisst das Phasen-Orakel für f - allgemeines Vorzeichen hat keine Auswirkung :-(

5.1.3 Phasen-Orakel

Somit hat sich das $n+1$-te QuBit im Zustand $|-\rangle$ einfach reproduziert und die ersten n QuBits transformieren sich wie folgt:

$$O_f \left|x_1, \ldots, x_n\right\rangle = (-1)^{f(x_1,\ldots,x_n)} \left|x_1, \ldots, x_n\right\rangle ,$$

$\hat{\mathcal{O}}_f$ heisst das Phasen-Orakel für f - allgemeines Vorzeichen hat keine Auswirkung :-(Phasen-Orakel kann auch relative Vorzeichen erzeugen! :-)

5.1.3 Phasen-Orakel

Somit hat sich das $n+1$-te QuBit im Zustand $|-\rangle$ einfach reproduziert und die ersten n QuBits transformieren sich wie folgt:

$$O_f \left|x_1, \ldots, x_n\right\rangle = (-1)^{f(x_1,\ldots,x_n)} \left|x_1, \ldots, x_n\right\rangle ,$$

$\hat{\mathcal{O}}_f$ heisst das Phasen-Orakel für f - allgemeines Vorzeichen hat keine Auswirkung :-(Phasen-Orakel kann auch relative Vorzeichen erzeugen! :-)

Wirkung des Phasen-Orakels auf einen allgemeinen 2-QuBit zustand

$$|\psi\rangle = \psi_{00}\,|00\rangle + \psi_{01}\,|01\rangle + \psi_{10}\,|10\rangle + \psi_{11}\,|11\rangle$$

5.1.3 Phasen-Orakel

Somit hat sich das $n+1$-te QuBit im Zustand $|-\rangle$ einfach reproduziert und die ersten n QuBits transformieren sich wie folgt:

$$O_f |x_1, \ldots, x_n\rangle = (-1)^{f(x_1,\ldots,x_n)} |x_1, \ldots, x_n\rangle ,$$

$\hat{\mathcal{O}}_f$ heisst das Phasen-Orakel für f - allgemeines Vorzeichen hat keine Auswirkung :-(Phasen-Orakel kann auch relative Vorzeichen erzeugen! :-)

Wirkung des Phasen-Orakels auf einen allgemeinen 2-QuBit zustand

$$|\psi\rangle = \psi_{00}|00\rangle + \psi_{01}|01\rangle + \psi_{10}|10\rangle + \psi_{11}|11\rangle$$

$$O_f |\psi\rangle = (-1)^{f(0,0)}\psi_{00}|00\rangle + (-1)^{f(0,1)}\psi_{01}|01\rangle + (-1)^{f(1,0)}\psi_{10}|10\rangle + (-1)^{f(1,1)}\psi_{11}|11\rangle .$$

5.1.3 Phasen-Orakel

Somit hat sich das $n+1$-te QuBit im Zustand $|-\rangle$ einfach reproduziert und die ersten n QuBits transformieren sich wie folgt:

$$O_f \left|x_1, \ldots, x_n\right\rangle = (-1)^{f(x_1,\ldots,x_n)} \left|x_1, \ldots, x_n\right\rangle ,$$

$\hat{\mathcal{O}}_f$ heisst das Phasen-Orakel für f - allgemeines Vorzeichen hat keine Auswirkung :-(Phasen-Orakel kann auch relative Vorzeichen erzeugen! :-)

Wirkung des Phasen-Orakels auf einen allgemeinen 2-QuBit zustand

$$|\psi\rangle = \psi_{00}\,|00\rangle + \psi_{01}\,|01\rangle + \psi_{10}\,|10\rangle + \psi_{11}\,|11\rangle$$

$$O_f\,|\psi\rangle = (-1)^{f(0,0)}\psi_{00}\,|00\rangle + (-1)^{f(0,1)}\psi_{01}\,|01\rangle + (-1)^{f(1,0)}\psi_{10}\,|10\rangle + (-1)^{f(1,1)}\psi_{11}\,|11\rangle .$$

Es stellt sich heraus, dass das Phasen-Orakel O_f nützlicher und meist einfacher in Quantenalgorithmen anzuwenden ist als das Bit-Orakel U_f, weswegen wir das Bit-Orakel nicht mehr weiter benutzen.

5.1.3 Phasen-Orakel

Übungsaufgabe 5.2: Phasen-Orakel für eine Ein-Qubit-Funktion

Erinnere dich an Übungsaufgabe 5.1, wo die vier Funktionen $f : \{0,1\} \to \{0,1\}$ mit einem Eingabe- und Ausgabe-Bit vorgestellt wurden. Kannst du das Phasen-Orakel O_f für jede davon in QUIRKY implementieren?

Lösung Übungsaufgabe 5.2

Es gibt vier Funktionen: die 'Identitäts'-Funktion $f(x) = x$, die NOT-Funktion, die Alles-Null-Funktion und die Alles-Eins-Funktion.

- Für die Identitäts-Funktion $f(x) = x$ kann Gl. (5.6) gelesen werden als

$$O_f |x\rangle = (-1)^x |x\rangle ,$$

also ist das genau das Z-Gatter:

- Für die NOT-Funktion $f(x) = \text{NOT}(x)$ soll gelten, dass

$$O_f |x\rangle = (-1)^{\text{NOT}(x)} |x\rangle = \text{NOT}\, Z\, \text{NOT} |x\rangle ,$$

was folgender Reihe an Operationen entspricht:

- Für die Alles-Null-Funktion $f(0) = f(1) = 0$ gilt:

$$O_f |x\rangle = |x\rangle ,$$

daher müssen wir nichts tun:

- Für die Alles-Eins-Funktion $f(0) = f(1) = 1$ gilt:

$$O_f |x\rangle = - |x\rangle ,$$

was wir erreichen, indem wir die ersten beiden Orakel nacheinander schalten:

Tatsächlich fügt das erste Orakel ein Minuszeichen hinzu, wenn $x = 1$ ist, während das zweite Orakel ein Minuszeichen hinzuzufügt, wenn $x = 0$ ist, also erhalten wir immer eines:

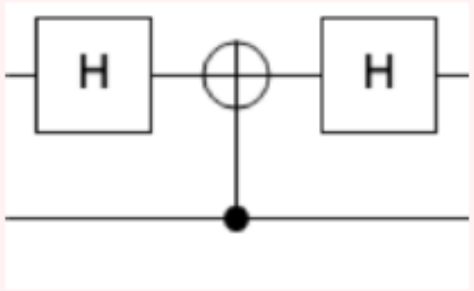
$$\begin{aligned} \text{NOT}\, Z\, \text{NOT}\, Z\, |0\rangle &= \text{NOT}\, Z\, \text{NOT} |0\rangle = - |0\rangle , \\ \text{NOT}\, Z\, \text{NOT}\, Z\, |1\rangle &= \text{NOT}\, Z\, \text{NOT}(- |1\rangle) = -\text{NOT}\, Z\, \text{NOT} |1\rangle = - |1\rangle . \end{aligned}$$

Im vorletzten Schritt haben wir dabei die Linearität ausgenutzt, um das Minuszeichen nach vorne zu bringen.

5.1.3 Phasen-Orakel

Hausaufgabe 5.1: Bestimme die Funktion anhand ihres Phasen-Orakels

Betrachte den folgenden Zwei-Qubit-Schaltkreis (wie üblich ist der untere Draht das erste Qubit):



Für welche Funktion $f : \{0,1\}^2 \to \{0,1\}$ stellt der Schaltkreis das Phasen-Orakel dar?

Hinweis: Benutze, dass $H\,\mathrm{NOT}\,H = Z$, was aus Übungsaufgabe 4.5 folgt.

Lass uns kurz zusammenfassen, was wir bisher erreicht haben: Mit Bit-Orakeln können wir Quantencomputer eine Funktion $f : \{0,1\}^n \to \{0,1\}$ genau so auswerten lassen, wie mit einem klassischen Computern, der umkehrbar agiert (vergleiche Gl. (5.4) und (5.5)). Das ist wichtig, da wir also nicht Äpfel mit Birnen vergleichen wenn wir zählen, wie viele Fragen man das Bit-Orakel U_f fragen muss, um eine Eigenschaft über f zu lernen, im Vergleich zu wie oft man f auf einem klassischen Computer auswerten muss, um die gleiche Eigenschaft zu lernen. Und da wir gerade gezeigt haben, dass man das Phasen-Orakel O_f immer mit einem Bit-Orakel U_f bauen kann, macht es keinen Unterschied wenn wir stattdessen das Phasen-Orakel O_f fragen

5.2 Quantenalgorithmen

Beispiel für Quantenalgorithmen, die ein Rechenproblem sehr viel schneller als ein klassischer Computer lösen

Basiert auf Interferenz…

Geschwindigkeitsmessung: wie viele Fragen müssen wir an das Orakel stellen?
Wie oft müssen wir $\hat{\mathcal{O}}_f$ auswerten, um die Eigenschaften von f zu bestimmen?

5.2.1 Der Algorithmus von Deutsch

Es ist ein später Sonntagabend. Alice und Bob haben gerade eben ihre Hausaufgaben für den Quantencomputerkurs gemacht und wollen jetzt einen 3D-Film schauen. Als sie ihren holografischen Fernseher anschalten, stellen sie fest, dass der Film wegen unerwarteter dramatischer Neuigkeiten von der Internationalen Transgalaktischen Station verschoben wurde. Es gab einen schrecklichen Unfall: ein Modul mit den zwei Crew-Mitgliedern Hila und Iman hat sich vom Hauptschiff getrennt. Die letzte empfangene Nachricht vom Modul war, dass Iman verletzt wurde und blutete – er braucht dringend eine Bluttransfusion. Leider stehen sowohl Hila als auch Iman unter Schock und haben ihren eigenen Blutgruppen vergessen – sie können sich nur daran erinnern, dass sie jeweils entweder Blutgruppe A oder B hatten. Die Moderation der Sendung appelliert an alle Zuschauende, Vorschläge zu machen, wie Hila und Iman herausfinden könnten, ob sie die selbe Blutgruppe haben, dann könnte Hila nämlich ihr Blut an Iman übertragen um sein Leben zu retten. An Bord befindet sich nämlich ein Lympho-Transcoder, der die beiden Blutgruppen in den jeweils anderen umwandeln kann. Selbst wenn sie also nicht die gleiche Blutgruppe haben, könnte Hila den Transcoder nutzen um ihr Blut zum richtigen Typ umzuwandeln.

5.2.1 Der Algorithmus von Deutsch

Nachdem sie diese Nachrichten gehört haben, entscheiden sich Alice und Bob zu überlegen, wie man Hila und Iman helfen könnte, anstelle den Film zu schauen. Die Nachrichtensendung setzt fort mit weiteren Informationen. Glücklicherweise hat das Modul einen Datenbank-Chip, auf dem Hila und Imans Blutgruppe gespeichert ist. Wir können diesen durch eine Funktion $f: \{0,1\} \to \{0,1\}$ modellieren, wobei

$$f(0) = \begin{cases} 0 & , \\ 1 & . \end{cases}$$

$$f(1) = \begin{cases} 0 & , \\ 1 & . \end{cases}$$

Nun muss herausgefunden werden, ob $f(0) = f(1)$ gilt oder nicht!

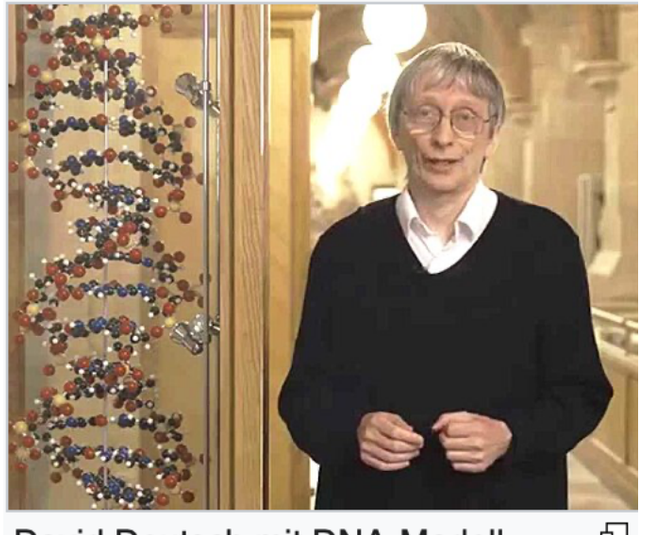
5.2.1 Der Algorithmus von Deutsch

Die Lösung scheint einfach: Hila und Iman fragen die Datenbank einfach zweimal ab um ihre jeweiligen Blutgruppen $f(0)$ und $f(1)$ herauszufinden und vergleichen anschließend die Werte. Unglücklicherweise wurde der Chip bei dem Unfall beschädigt und die Nachrichtensendung berichtet, dass die Datenbank höchstwahrscheinlich nach einer einzigen Abfrage völlig zerstört sein würde.

Unsere beiden Protagonisten befinden sich in einer Pattsituation. Offensichtlich muss jeder klassische Algorithmus f genau zweimal auswerten um herauszufinden, ob $f(0) = f(1)$. Wenn wir den Wert von $f(0)$ kennen hängt $f(0) = f(1)$ immer noch von $f(1)$ ab und lässt sich nicht bestimmen, außer man berechnet $f(1)$. Genauso lässt sich ein bekanntes $f(1)$ nicht mit einem unbekannten $f(0)$ vergleichen. Egal welchen Ansatz man verfolgt, man muss sowohl den Wert von $f(0)$ als auch von $f(1)$ kennen, um $f(0) = f(1)$ zu bestimmen. Gibt es da wirklich keinen Ausweg?

5.2.1 Der Algorithmus von Deutsch (1985)

Nachdem Bob ein paar Bedienungsanleitungen durchblättert hat, stellt Bob fest, dass der Datenbank-Chip auch einen *Quantenmodus* besitzt. Wenn dieser aktiviert ist, wertet die Datenbank die Funktion nicht mehr klassisch aus, sondern nutzt stattdessen das Phasen-Orakel O_f. Könnte das vielleicht dabei helfen, das Problem zu lösen? Alice überlegt eine Weile und stellt dann überrascht fest, dass es genau für dieses Problem den Algorithmus von Deutsch gibt! Die beiden prüfen mit ein paar Berechnungen noch schnell, dass alles funktioniert und senden dann eine intergalaktische E-Mail mit Anweisungen zum lösen des Dilemmas an Hila und Iman. Ihre Instruktionen sind die folgenden:



David Deutsch mit DNA-Modell

5.2.1 Der Algorithmus von Deutsch

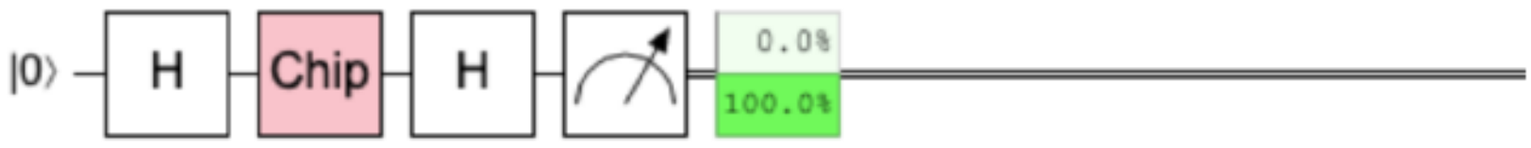
1. Bereitet ein Qubit im Zustand $|+\rangle = (|0\rangle + |1\rangle)/\sqrt{2}$ vor.

2. Nutzt den Datenbank-Chip im Quantenmodus um die Operation O_f auf das Qubit anzuwenden.

3. Wendet das Hadamard-Gatter H auf das Ausgabe-Qubit an und messt anschließend.

4. Wenn das Messergbnis 0 ist, haben Hila und Iman die gleiche Blutgruppe, ansonsten haben sie unterschiedliche.

Beachte, dass bei dieser Prozedur der Datenbank-Chip nur *einmal* abgefragt wird, um zu bestimmen, ob sie die gleiche Blutgruppe haben. Hier ist eine Implementierung des Algorithmus in QUIRKY:



Das Bild zeigt, dass das Ergebnis 1 ist, also haben Hila und Iman unterschiedliche Blutgruppen.

5.2.1 Der Algorithmus von Deutsch

Aber wieso funktioniert der Algorithmus von Deutsch? Lass uns den Algorithmus Schritt für Schritt analsysieren. Das erste Hadamard-Gatter erstellt den Zustand $|+\rangle = H|0\rangle$. Anschließend wenden wir das Phasen-Orakel O_f an, was zum folgenden Zustand führt

$$\begin{aligned} O_f|+\rangle &= \frac{1}{\sqrt{2}} O_f|0\rangle + \frac{1}{\sqrt{2}} O_f|1\rangle \\ &= \frac{1}{\sqrt{2}} (-1)^{f(0)}|0\rangle + \frac{1}{\sqrt{2}} (-1)^{f(1)}|1\rangle . \end{aligned}$$

Nach dem anwenden des zweiten Hadamard-Gatters, erhalten wir den Zustand:

$$\begin{aligned} HO_f|+\rangle &= \frac{1}{\sqrt{2}} (-1)^{f(0)} H|0\rangle + \frac{1}{\sqrt{2}} (-1)^{f(1)} H|1\rangle \\ &= \frac{1}{\sqrt{2}} (-1)^{f(0)}|+\rangle + \frac{1}{\sqrt{2}} (-1)^{f(1)}|-\rangle \\ &= \frac{1}{2} (-1)^{f(0)} (|0\rangle + |1\rangle) + \frac{1}{2} (-1)^{f(1)} (|0\rangle - |1\rangle) \\ &= \frac{(-1)^{f(0)} + (-1)^{f(1)}}{2}|0\rangle + \frac{(-1)^{f(0)} - (-1)^{f(1)}}{2}|1\rangle . \end{aligned} \tag{5.7}$$

5.2.1 Der Algorithmus von Deutsch

Beachte, dass die beiden Vorzeichen $(-1)^{f(0)}$ und $(-1)^{f(1)}$ in der ersten Amplitude *addiert*, in der zweiten aber *subtrahiert* werden. Je nach den Werten von $f(0)$ und $f(1)$ sehen wir für jede Amplitude entweder konstruktive oder destruktive Interferenz (siehe §2.6.1). Tatsächlich bestimmt sich nur daran, ob $f(0)$ und $f(1)$ gleich sind oder nicht, welche Amplitude übrig bleibt:

$$\begin{aligned} f(0) = f(1): &\qquad HO_f\,|+\rangle = \pm\,|0\rangle\,, \\ f(0) \neq f(1): &\qquad HO_f\,|+\rangle = \pm\,|1\rangle\,. \end{aligned} \tag{5.8}$$

Es ist eine gute Übung, dies explizit zu verifizieren:

Übungsaufgabe 5.3: Den Algorithmus von Deutsch verifizieren

Erinnere dich aus Übungsaufgabe 5.1 daran, dass es vier Funktionen $f : \{0,1\} \to \{0,1\}$ gibt. Berechne für jede Funktion den Zustand $HO_f\,|+\rangle$ mit Gl. (5.7).

5.2.1 Der Algorithmus von Deutsch

Gl. (5.8) zeigt, dass die abschließende Messung nur dann das Ergebnis 0 ergibt, wenn $f(0) = f(1)$ gilt. Also bestimmt der Algorithmus ob $f(0) = f(1)$. Dabei evaluiert der Algorithmus die Funktion $f : \{0,1\} \to \{0,1\}$ nur ein einziges Mal mit dem Phasen-Orakel. Im Gegensatz dazu hatten wir gesehen, dass ein klassischer Algorithmus notwendigerweise beide Funktionswerte $f(0)$ und $f(1)$ getrennt auswerten muss.

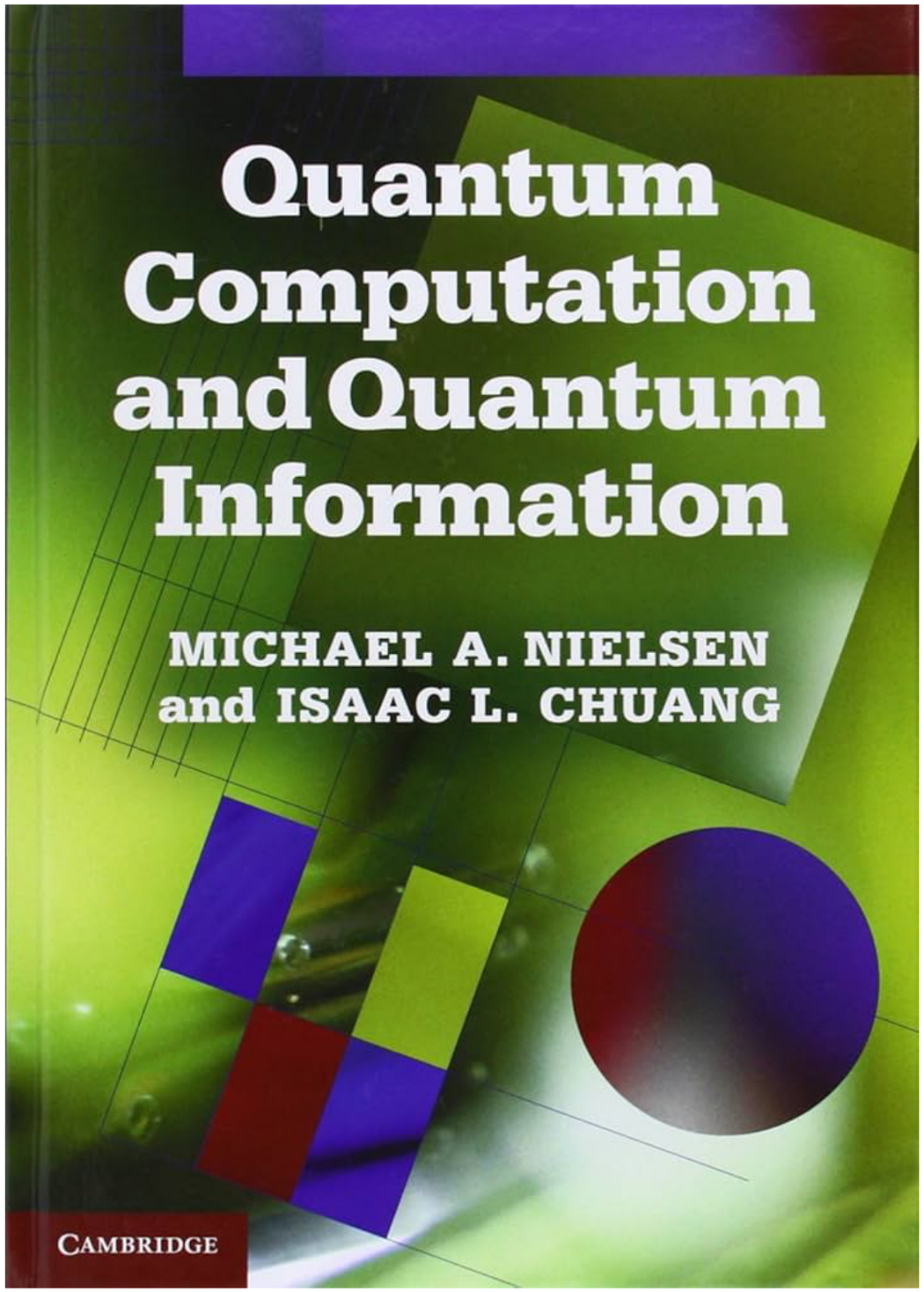
Eine weitere Interpretation des Algorithmus von Deutsch ist, dass er die Summe der beiden Bits $f(0)$ und $f(1)$ modulo zwei berechnet. Das liegt daran, dass $f(0) \oplus f(1) = 0$ nur genau dann gilt, wenn $f(0) = f(1)$. Aus Gl. (3.20) erinnern wir uns an die Definition von Addition modulo zwei:

x_1	x_2	$x_1 \oplus x_2$
0	0	0
0	1	1
1	0	1
1	1	0

(5.9)

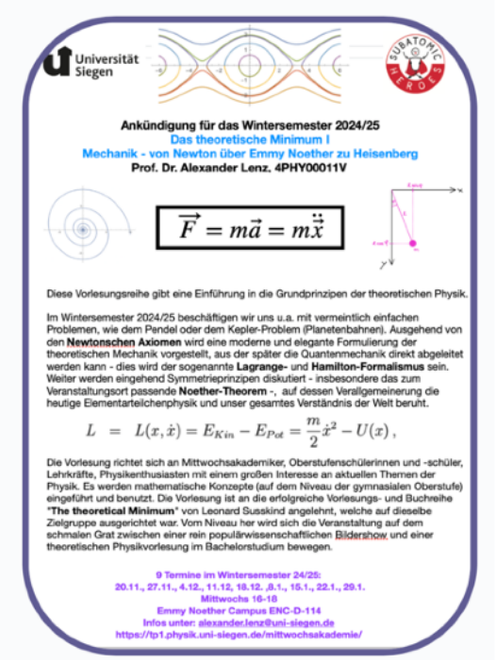
Aus diesem Grund ist die Summe modulo zwei auch als *XOR* (engl. Abkürzung für "exklusives Oder") der beiden Bits, da sie 1 genau dann ergibt, wenn nur eines der beiden Bits gesetzt ist.

5.4 Deine Quanten-Reise



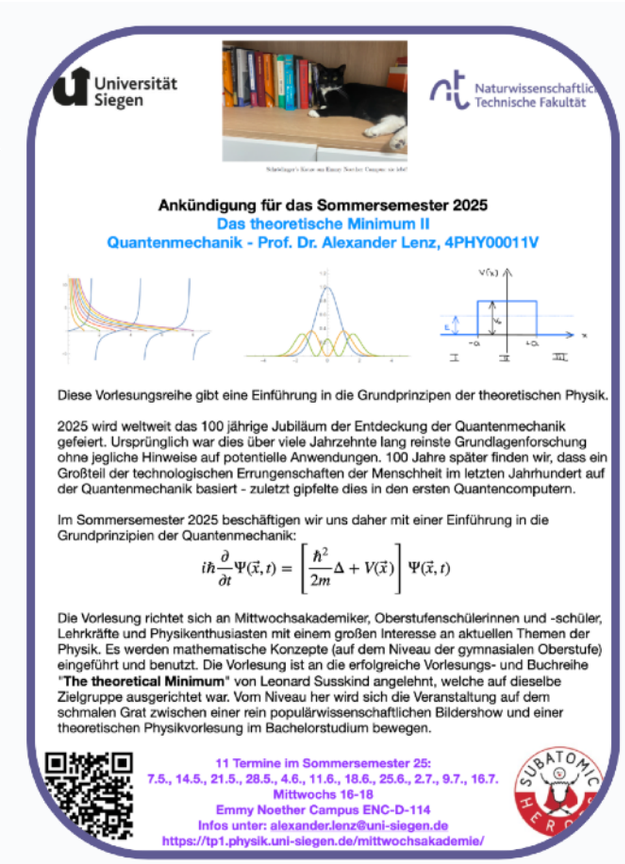
Die Zukunft der Physik-Mittwochsakademie

SS 26:
WS 26/27: A. Lenz - The theoretical Minimum - Klassische Mechanik

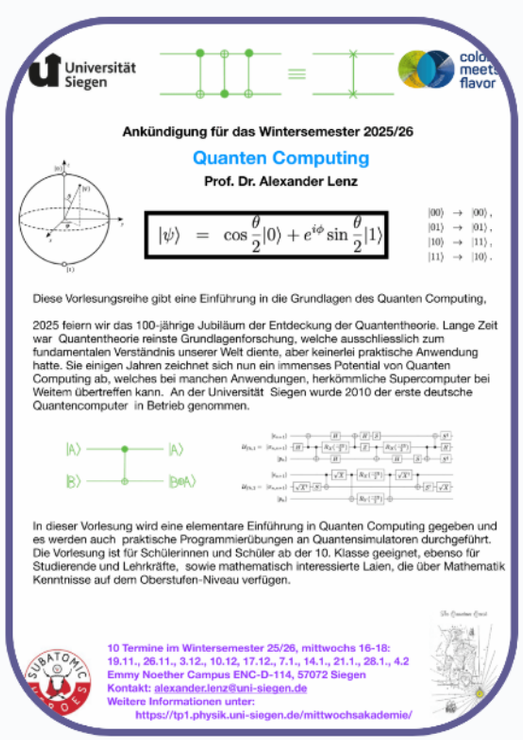


SS 27:
WS 27/28: A. Lenz - The theoretical Minimum - Klassische Feldtheorie Neu!!!

SS 28:
WS 28/29: A. Lenz - The theoretical Minimum - Quanten Mechanik



SS 29:
WS 29/30: A. Lenz - Quantencomuting



Die glorreiche Vergangenheit der Physik-Mittwochsakademie

Claus Grupen Farewell

Wednesday Feb 4, 2026, 5:00 PM → 8:00 PM Europe/Berlin

D-114 -> B-127 (ENC)

Description

Claus Grupen obtained his PhD in Physics in 1970 from Kiel University. After spending time as Visiting Fellow of the Royal Society at Durham University (UK) he became Professor at the University of Siegen in 1978 and during his career he made numerous important contributions to the field of experimental particle physics.

Besides his research he also put a significant effort in the popularisation of science, in particular with lecture courses for pupils and for the interested public. A lot of the material he created is collected on his webpage: https://www.hep.physik.uni-siegen.de/~grupen/ .

Finally Prof. Grupen authored several textbooks and he is a very talented cartoonist - his drawings were also regularly published..

To celebrate his achievements we cordially invite you to attend our farewell event on Wednesday, 4th February 2026, at 17:00 in the big physics lecture hall (ENC D 114) at the Emmy Noether Campus. Drinks will be served later on in the seminar room and in the coffee room of TP1 (ENC B 127&128)