

# Quantencomputer

## Werkzeuge für die Algorithmenimplementierung

Timo Grautstück

Fachhochschule Dortmund  
FB 10: Informationstechnik

31. Mai 2022

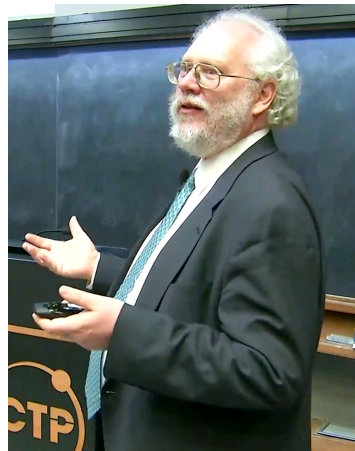
Grundlagen Quantencomputer

Werkzeuge zur Implementierung

Quantenalgorithmen

# Warum Quantencomputer?

- Shors Algorithmus:  $\mathcal{O}((\log n)^3)$
- Grover Algorithmus:  $\mathcal{O}(\sqrt{N})$
- Quantenkryptographie: key exchange problem
- **Quantenüberlegenheit (*Quantum Supremacy*)**



Peter Shor 2017

Bildquelle (wikimedia): [https://commons.wikimedia.org/wiki/File:Peter\\_Shor\\_2017\\_Dirac\\_Medal\\_Award\\_Ceremony.png](https://commons.wikimedia.org/wiki/File:Peter_Shor_2017_Dirac_Medal_Award_Ceremony.png)

## Article

## Quantum supremacy using a programmable superconducting processor

<https://doi.org/10.1038/s41586-019-1666-5>

Received: 22 July 2019

Accepted: 20 September 2019

Published online: 23 October 2019

Frank Arute<sup>1</sup>, Kunal Arya<sup>1</sup>, Ryan Babbush<sup>1</sup>, Dave Bacon<sup>1</sup>, Joseph C. Bardin<sup>1,2</sup>, Rami Barends<sup>1</sup>, Rupak Biswas<sup>1</sup>, Sergio Boixo<sup>1</sup>, Fernando G. S. L. Brandao<sup>1,3</sup>, David A. Buell<sup>1</sup>, Brian Burkett<sup>1</sup>, Yu Chen<sup>1</sup>, Zijun Chen<sup>1</sup>, Ben Chiaro<sup>1</sup>, Roberto Collins<sup>1</sup>, William Courtney<sup>1</sup>, Andrew Dunsworth<sup>1</sup>, Edward Farhi<sup>1</sup>, Brooks Foxen<sup>1,4</sup>, Austin Fowler<sup>1</sup>, Craig Gidney<sup>1</sup>, Marissa Giustina<sup>1</sup>, Rob Graff<sup>1</sup>, Keith Guerin<sup>1</sup>, Steve Habegger<sup>1</sup>, Matthew P. Harrigan<sup>1</sup>, Michael J. Hartmann<sup>1,5</sup>, Alan Ho<sup>1</sup>, Markus Hoffmann<sup>1</sup>, Trent Huang<sup>1</sup>, Travis S. Humble<sup>1</sup>, Sergei V. Isakov<sup>1</sup>, Evan Jeffrey<sup>1</sup>, Zhang Jiang<sup>1</sup>, Dvir Kafri<sup>1</sup>, Kostyantyn Kechedzhiev<sup>1</sup>, Julian Kelly<sup>1</sup>, Paul V. Klimov<sup>1</sup>, Sergey Knysh<sup>1</sup>, Alexander Korotkov<sup>1,6</sup>, Fedor Kostritsa<sup>1</sup>, David Landhuis<sup>1</sup>, Mike Lindmark<sup>1</sup>, Erik Lucero<sup>1</sup>, Dmitry Lyakh<sup>1</sup>, Salvatore Mandrà<sup>1,7,8</sup>, Jarrod R. McClean<sup>1</sup>, Matthew McEwen<sup>1</sup>, Anthony Megrant<sup>1</sup>, Xiao Mi<sup>1</sup>, Kristel Michielsen<sup>1,9</sup>, Masoud Mohseni<sup>1</sup>, Josh Mutus<sup>1</sup>, Ofer Naaman<sup>1</sup>, Matthew Neeley<sup>1</sup>, Charles Neill<sup>1</sup>, Murphy Yuezhen Niu<sup>1</sup>, Eric Ostby<sup>1</sup>, Andre Petukhov<sup>1</sup>, John C. Platt<sup>1</sup>, Chris Quintana<sup>1</sup>, Eleanor G. Rieffel<sup>1</sup>, Pedram Roushan<sup>1</sup>, Nicholas C. Rubin<sup>1</sup>, Daniel Sank<sup>1</sup>, Kevin J. Satzinger<sup>1</sup>, Vadim Smelyanskiy<sup>1</sup>, Kevin J. Sung<sup>1,10</sup>, Matthew D. Trevithick<sup>1</sup>, Amit Vainsencher<sup>1</sup>, Benjamin Villalonga<sup>1,4</sup>, Theodore White<sup>1</sup>, Z. Jamie Yao<sup>1</sup>, Ping Yeh<sup>1</sup>, Adam Zalcman<sup>1</sup>, Hartmut Neven<sup>1</sup> & John M. Martinis<sup>1,5\*</sup>

The promise of quantum computers is that certain computational tasks might be executed exponentially faster on a quantum processor than on a classical processor<sup>1</sup>. A fundamental challenge is to build a high-fidelity processor capable of running quantum algorithms in an exponentially large computational space. Here we report the use of a processor with programmable superconducting qubits<sup>2–7</sup> to create quantum states on 53 qubits, corresponding to a computational state-space of dimension  $2^{53}$  (about  $10^{16}$ ). Measurements from repeated experiments sample the resulting probability distribution, which we verify using classical simulations. Our Sycamore processor takes about 200 seconds to sample one instance of a quantum circuit a million times – our benchmarks currently indicate that the equivalent task for a state-of-the-art classical supercomputer would take approximately 10,000 years. This dramatic increase in speed compared to all known classical algorithms is an experimental realization of quantum supremacy<sup>8–14</sup> for this specific computational task, heralding a much-anticipated computing paradigm.

## The future

Quantum processors based on superconducting qubits can now perform computations in a Hilbert space of dimension  $2^{53} = 9 \times 10^{15}$ , beyond the reach of the fastest classical supercomputers available today. To our knowledge, this experiment marks the first computation that can be performed only on a quantum processor. Quantum processors have thus reached the regime of quantum supremacy. We expect that their computational power will continue to grow at a double-exponential rate: the classical cost of simulating a quantum circuit increases exponentially with computational volume, and hardware improvements will probably follow a quantum-processor equivalent of Moore's law<sup>32,33</sup>, doubling this computational volume every few years. To sustain the double-exponential growth rate and to eventually offer the computational volume needed to run well known quantum algorithms, such as the Shor or Grover algorithms<sup>25,34</sup>, the engineering of quantum error correction will need to become a focus of attention.

The extended Church–Turing thesis formulated by Bernstein and Vazirani<sup>35</sup> asserts that any ‘reasonable’ model of computation can be efficiently simulated by a Turing machine. Our experiment suggests that a model of computation may now be available that violates this

Nature | Vol 574 | 24 OCTOBER 2019 | 509

# Quantenbits I

## Basiszustände

$$|0\rangle = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \quad |1\rangle = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$$

## Zweizustandssystem

Kann sich in einer Superposition der Basiszustände befinden:

$$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle = \begin{bmatrix} \alpha \\ \beta \end{bmatrix} \quad \alpha, \beta \in \mathbf{C}$$

$$|\psi\rangle = \cos\left(\frac{\theta}{2}\right) |0\rangle + e^{i\phi} \sin\left(\frac{\theta}{2}\right) |1\rangle \quad \theta, \phi \in \mathbf{R}$$

→  $\phi$  und  $\theta$  sind Kugelkoordinaten, der Radius  $r = 1$

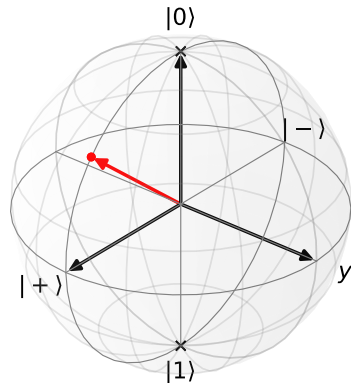
# Quantenbits II

## Normalisierung

$$\begin{aligned}\langle\psi|\psi\rangle &= 1 \\ \Rightarrow |\alpha|^2 + |\beta|^2 &= 1\end{aligned}$$

## Beispiel: $\phi=0$ und $\theta=\pi/2$

$$\begin{aligned}|\psi\rangle &= \cos\left(\frac{\pi}{4}\right)|0\rangle + e^{i0}\sin\left(\frac{\pi}{4}\right)|1\rangle \\ \Rightarrow |\psi\rangle &= |+\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)\end{aligned}$$



Bloch-Kugel (Blochsphere)

# Quantenbits III

## Quantenregister

→  $n$  Qubits besitzen  $2^n$  Wahrscheinlichkeitsamplituden

$$|\psi\rangle = \alpha_{00}|00\rangle + \alpha_{01}|01\rangle + \alpha_{10}|10\rangle + \alpha_{11}|11\rangle = \begin{bmatrix} \alpha_{00} \\ \alpha_{01} \\ \alpha_{10} \\ \alpha_{11} \end{bmatrix}$$

→ Können durch Produkte der Basiszustände beschreiben werden

$$|0\rangle \otimes |0\rangle = |00\rangle = \begin{bmatrix} 1 \times \begin{bmatrix} 1 \\ 0 \end{bmatrix} \\ 0 \times \begin{bmatrix} 1 \\ 0 \end{bmatrix} \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

# Quantengatter I

- Manipulation von Qubits
- Quantengatter die auf  $n$  Bits operieren sind unitäre  $2^n \times 2^n$ -Matrizen
- Für diese Matrizen existieren Eigenvektoren

## Unitär

$$I = A^\dagger A \quad A^\dagger = A^{*T}$$

## Eigenvektor & Eigenwert

$$A|\psi\rangle = \lambda|\psi\rangle = e^{2\pi i\theta}|\psi\rangle$$

## Beispiel: X- & H-Gatter

$$X = |0\rangle\langle 1| + |1\rangle\langle 0| = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$$

$$H = |0\rangle\langle +| + |1\rangle\langle -| = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$$

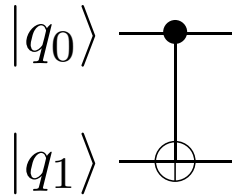


# Quantengatter II - (Pauligatter)

Matrix	Schaltungssymbol	Wahrheitstabelle									
$X = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$	$ q\rangle \text{ --- } \boxed{X} \text{ ---}$	<table> <tr> <th>Fall</th><th><math> q\rangle</math></th><th><math>X q\rangle</math></th></tr> <tr> <td>1</td><td><math> 0\rangle</math></td><td><math> 1\rangle</math></td></tr> <tr> <td>2</td><td><math> 1\rangle</math></td><td><math> 0\rangle</math></td></tr> </table>	Fall	$ q\rangle$	$X q\rangle$	1	$ 0\rangle$	$ 1\rangle$	2	$ 1\rangle$	$ 0\rangle$
Fall	$ q\rangle$	$X q\rangle$									
1	$ 0\rangle$	$ 1\rangle$									
2	$ 1\rangle$	$ 0\rangle$									
$Y = \begin{bmatrix} 0 & -i \\ i & 0 \end{bmatrix}$	$ q\rangle \text{ --- } \boxed{Y} \text{ ---}$	<table> <tr> <th>Fall</th><th><math> q\rangle</math></th><th><math>Y q\rangle</math></th></tr> <tr> <td>1</td><td><math> 0\rangle</math></td><td><math>i 1\rangle</math></td></tr> <tr> <td>2</td><td><math> 1\rangle</math></td><td><math>-i 0\rangle</math></td></tr> </table>	Fall	$ q\rangle$	$Y q\rangle$	1	$ 0\rangle$	$i 1\rangle$	2	$ 1\rangle$	$-i 0\rangle$
Fall	$ q\rangle$	$Y q\rangle$									
1	$ 0\rangle$	$i 1\rangle$									
2	$ 1\rangle$	$-i 0\rangle$									
$Z = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}$	$ q\rangle \text{ --- } \boxed{Z} \text{ ---}$	<table> <tr> <th>Fall</th><th><math> q\rangle</math></th><th><math>Z q\rangle</math></th></tr> <tr> <td>1</td><td><math> 0\rangle</math></td><td><math> 0\rangle</math></td></tr> <tr> <td>2</td><td><math> 1\rangle</math></td><td><math>- 1\rangle</math></td></tr> </table>	Fall	$ q\rangle$	$Z q\rangle$	1	$ 0\rangle$	$ 0\rangle$	2	$ 1\rangle$	$- 1\rangle$
Fall	$ q\rangle$	$Z q\rangle$									
1	$ 0\rangle$	$ 0\rangle$									
2	$ 1\rangle$	$- 1\rangle$									

# Quantengatter III - (kontrollierte Gatter)

- Alle Gatter können kontrolliert auf  $n$  Qubits angewandt werden
- Ein Zielqubit und  $n - 1$  kontrollierende Qubits
- Um eine Transformation auf einem Zielbit auszuführen, müssen sich alle kontrollierenden Bits im Zustand  $|1\rangle$  befinden

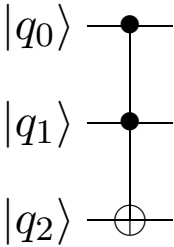


Kontrolliertes-Nicht-Gatter

## Beispiel: CNOT

$$CX_{01} = |0\rangle\langle 0| \otimes I + |1\rangle\langle 1| \otimes X = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

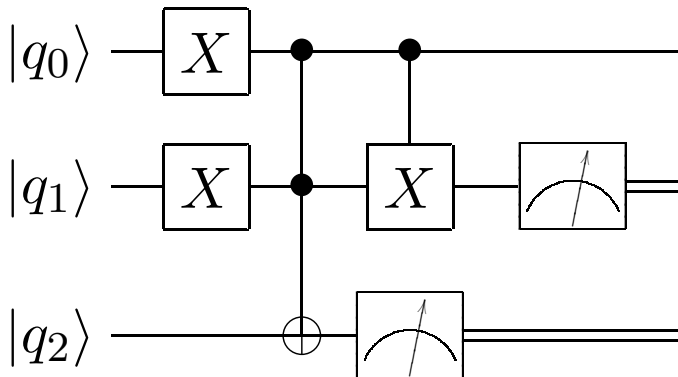
# Quantengatter IV - (Toffoli-Gate)

Matrix	Schaltungssymbol	Wahrheitstabelle																											
$CCX_{012} = \begin{bmatrix} I_2 & 0_2 & 0_2 & 0_2 \\ 0_2 & I_2 & 0_2 & 0_2 \\ 0_2 & 0_2 & I_2 & 0_2 \\ 0_2 & 0_2 & 0_2 & X \end{bmatrix}$		<table> <tr> <th>Fall</th><th><math> q_0q_1q_2\rangle</math></th><th><math>CCX_{012} q_0q_1q_2\rangle</math></th></tr> <tr><td>1</td><td><math> 000\rangle</math></td><td><math> 000\rangle</math></td></tr> <tr><td>2</td><td><math> 001\rangle</math></td><td><math> 001\rangle</math></td></tr> <tr><td>3</td><td><math> 010\rangle</math></td><td><math> 010\rangle</math></td></tr> <tr><td>4</td><td><math> 011\rangle</math></td><td><math> 011\rangle</math></td></tr> <tr><td>5</td><td><math> 100\rangle</math></td><td><math> 100\rangle</math></td></tr> <tr><td>6</td><td><math> 101\rangle</math></td><td><math> 101\rangle</math></td></tr> <tr><td>7</td><td><math> 110\rangle</math></td><td><math> 111\rangle</math></td></tr> <tr><td>8</td><td><math> 111\rangle</math></td><td><math> 110\rangle</math></td></tr> </table>	Fall	$ q_0q_1q_2\rangle$	$CCX_{012} q_0q_1q_2\rangle$	1	$ 000\rangle$	$ 000\rangle$	2	$ 001\rangle$	$ 001\rangle$	3	$ 010\rangle$	$ 010\rangle$	4	$ 011\rangle$	$ 011\rangle$	5	$ 100\rangle$	$ 100\rangle$	6	$ 101\rangle$	$ 101\rangle$	7	$ 110\rangle$	$ 111\rangle$	8	$ 111\rangle$	$ 110\rangle$
Fall	$ q_0q_1q_2\rangle$	$CCX_{012} q_0q_1q_2\rangle$																											
1	$ 000\rangle$	$ 000\rangle$																											
2	$ 001\rangle$	$ 001\rangle$																											
3	$ 010\rangle$	$ 010\rangle$																											
4	$ 011\rangle$	$ 011\rangle$																											
5	$ 100\rangle$	$ 100\rangle$																											
6	$ 101\rangle$	$ 101\rangle$																											
7	$ 110\rangle$	$ 111\rangle$																											
8	$ 111\rangle$	$ 110\rangle$																											

Toffoli-Gatter

# Quantenschaltungen

- Grundlage für Quantenalgorithmen
- Keine Rückführungen (azyklisch)
- Kopieren und Zusammenführen von Qubits ist nicht erlaubt



Quantenschaltung für einen Halbaddierer

- IBM Quantum Composer
  - Halbaddierer bauen
  - Hardware
  - Quantum Lab
- Qiskit

# Quanten Fourier-Transformation I

- Wird in vielen anderen Algorithmen genutzt z.B. Shor's
- Transformiert einen Basiszustand  $|x\rangle$  zu einem Fourierzustand  $|\tilde{x}\rangle$
- $N = 2^n$ ;  $n$  ist die Anzahl der genutzten Qubits

## QFT

$$QFT \Rightarrow |\tilde{x}\rangle = \frac{1}{\sqrt{N}} \sum_{j=0}^{N-1} e^{i\frac{2\pi}{N}xj} \cdot |j\rangle$$

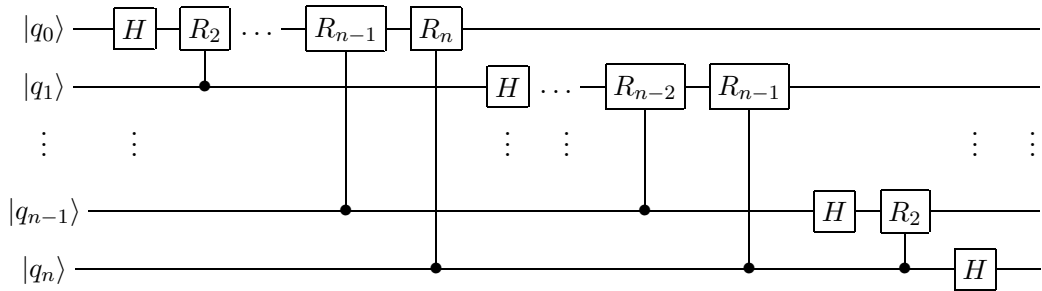
## Als Produkt - (siehe [2])

$$|\tilde{x}\rangle = \frac{1}{\sqrt{N}} \left( |0\rangle + e^{i\frac{2\pi x}{2^1}} |1\rangle \right) \otimes \left( |0\rangle + e^{i\frac{2\pi x}{2^2}} |1\rangle \right) \otimes \dots \otimes \left( |0\rangle + e^{i\frac{2\pi x}{2^n}} |1\rangle \right)$$

# Quanten Fourier-Transformation II

## kontrollierte Rotation [2]

$$R_k = \begin{bmatrix} 1 & 0 \\ 0 & e^{i\frac{2\pi}{2^k}} \end{bmatrix}$$



Schaltung der Quantum Fourier-Transformation vgl. [3]

# Quanten-Phasenschätzung I

- Quantumteil des Shors Algorithmus
- Operiert auf zwei Quantumregister
- Nutzt Phasenrückstoß (*phase kickback*), um die Phase eines unitären Operators  $U$  (im Fourierzustand) in ein Quantenregister zu schreiben [2].

## Eigenvektor

$$U|\psi\rangle = e^{2\pi i\theta}|\psi\rangle$$

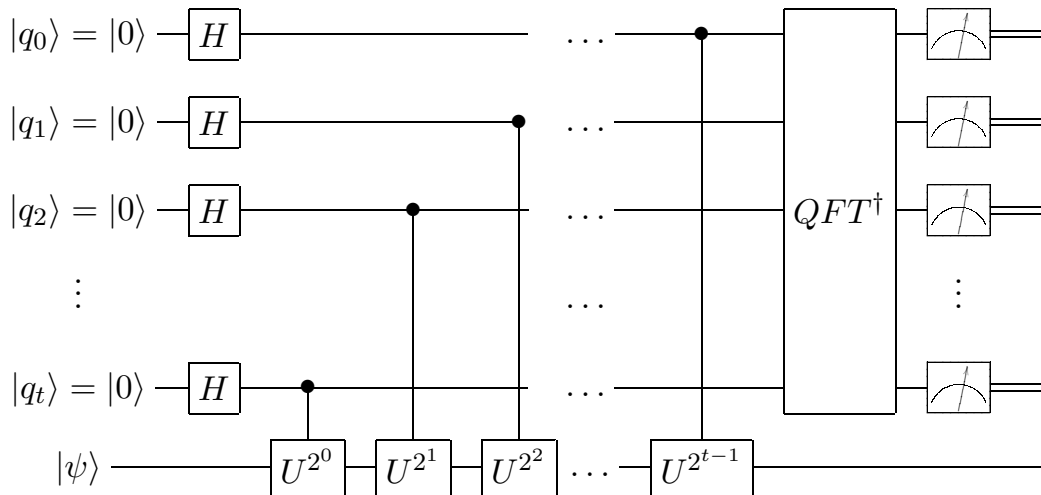
→ QPE bestimmt  $\theta$

## QPE - (siehe [2])

$$\frac{1}{\sqrt{2^t}} \sum_{k=0}^{2^t-1} e^{2\pi i\theta k} |k\rangle = \frac{1}{\sqrt{2^t}} \left( |0\rangle + e^{2\pi i 2^{t-1}\theta} |1\rangle \right) \otimes \left( |0\rangle + e^{2\pi i 2^{t-2}\theta} |1\rangle \right) \otimes \dots \otimes \left( |0\rangle + e^{2\pi i 2^0\theta} |1\rangle \right)$$



# Quanten-Phasenschätzung II



Schaltung der Quanten-Phasenschätzung vgl. [3]

# Quanten-Phasenschätzung III - (Beispiel)

→  $t = 3$  Qubits

## Gatter

$$U = P\left(\phi = \frac{\pi}{2}\right) = \begin{bmatrix} 1 & 0 \\ 0 & e^{\frac{i\pi}{2}} \end{bmatrix}$$

$$P|1\rangle = \begin{bmatrix} 1 & 0 \\ 0 & e^{\frac{i\pi}{2}} \end{bmatrix} \begin{bmatrix} 0 \\ 1 \end{bmatrix} = e^{\frac{i\pi}{2}} |1\rangle$$

## Erwartung

$$U|\psi\rangle = e^{2\pi i\theta} |\psi\rangle$$

→  $\theta = \frac{1}{4}$

## Warum Quantencomputer?

- Shor Algorithmus:  $\mathcal{O}(\log^3 n)$
- Grover Algorithmus:  $\mathcal{O}(\sqrt{N})$
- Quanteninterferometrie: key exchange problem
- Quantenüberlegenheit (Quantum Supremacy)



Peter Shor 2007

Microsoft Research: Why Quantum Computing Matters, Peter Shor, MIT, 2009, arXiv:0906.4798v2 [quant-ph]

Timo Grautstück Quantencomputer – Grundlagen Quantencomputer 0403

## Quanten supremacy

### Quantum supremacy using programmable superconducting qubits

Quantum supremacy is the point at which a quantum device can perform a task that is beyond the capabilities of a classical computer. This is a milestone in the development of quantum computing, as it demonstrates that quantum devices can solve problems that are intractable for classical computers.

Quantum supremacy is the point at which a quantum device can perform a task that is beyond the capabilities of a classical computer. This is a milestone in the development of quantum computing, as it demonstrates that quantum devices can solve problems that are intractable for classical computers.

Science, 549, 761–763, 2020

Timo Grautstück Quantencomputer – Grundlagen Quantencomputer 0403

## Quantenbits I

### Basiszustände

$$|0\rangle = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \quad |1\rangle = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$$

### Zweizustandensystem

Kann sich in einer Superposition der Basiszustände befinden

$$|\psi\rangle = \cos(\frac{\theta}{2})|0\rangle + e^{i\phi}\sin(\frac{\theta}{2})|1\rangle \quad \theta, \phi \in \mathbb{C}$$

$$|\psi\rangle = \cos(\frac{\theta}{2})|0\rangle + e^{i\phi}\sin(\frac{\theta}{2})|1\rangle \quad \theta, \phi \in \mathbb{R}$$

→  $\theta$  und  $\phi$  sind Eignerkoordinaten, der Radius  $r = 1$

Timo Grautstück Quantencomputer – Grundlagen Quantencomputer 0403

## Quantenbits II

### Normalisierung

$$\langle\psi|\psi\rangle = 1 \Rightarrow |\cos(\frac{\theta}{2})|^2 + |\sin(\frac{\theta}{2})|^2 = 1$$

### Beispiel: $\theta = \pi/2$ und $\phi = \pi/2$

$$|\psi\rangle = \cos(\frac{\pi}{4})|0\rangle + e^{i\pi/2}\sin(\frac{\pi}{4})|1\rangle$$

$$\Rightarrow |\psi\rangle = \frac{1}{\sqrt{2}}(|0\rangle + i|1\rangle)$$



Bloch-Kugel (Blochsphere)

Timo Grautstück Quantencomputer – Grundlagen Quantencomputer 0403

## Quantenbits III

### Quantenregister

- $n$  Qubits besitzen  $2^n$  Wahrscheinlichkeitsamplituden

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

$$\begin{bmatrix} a_{00} \\ a_{01} \\ a_{10} \\ a_{11} \end{bmatrix}$$

- Können durch Produkt der Basiszustände beschrieben werden

$$|0\rangle \otimes |0\rangle = |00\rangle = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

Timo Grautstück Quantencomputer – Grundlagen Quantencomputer 0403

## Quantengatter I

- Manipulation von Qubits
- Quantengatter die auf 2 Bits operieren sind unitäre  $2^n \times 2^n$ -Matrizen
- Für diese Matrizen existieren Eigenvektoren

### Unitar

$$U^\dagger = U^{-1}, \quad U^\dagger U = I = U U^\dagger$$

### Eigenvektor & Eigenwert

$$A|\psi\rangle = \lambda|\psi\rangle \Rightarrow e^{i\theta}|\psi\rangle$$

### Beispiel: X- & H-Gatter

$$X = |0\rangle\langle 1| + |1\rangle\langle 0| = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$$

$$H = |0\rangle\langle 0| + |1\rangle\langle 1| - \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$$


Timo Grautstück Quantencomputer – Grundlagen Quantencomputer 0403

## Quantengatter II – (Pauligatter)

Matrix	Schaltungssymbol	Wahrheitstabelle						
$X = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$	$ q\rangle \text{---} \boxed{X}$	<table><tr><th>q</th><th>X q</th></tr><tr><td>0</td><td>1</td></tr><tr><td>1</td><td>0</td></tr></table>	q	X q	0	1	1	0
q	X q							
0	1							
1	0							
$Y = \begin{bmatrix} 0 & -i \\ i & 0 \end{bmatrix}$	$ q\rangle \text{---} \boxed{Y}$	<table><tr><th>q</th><th>Y q</th></tr><tr><td>0</td><td>-i</td></tr><tr><td>1</td><td>i</td></tr></table>	q	Y q	0	-i	1	i
q	Y q							
0	-i							
1	i							
$Z = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}$	$ q\rangle \text{---} \boxed{Z}$	<table><tr><th>q</th><th>Z q</th></tr><tr><td>0</td><td>1</td></tr><tr><td>1</td><td>-1</td></tr></table>	q	Z q	0	1	1	-1
q	Z q							
0	1							
1	-1							

Timo Grautstück Quantencomputer – Grundlagen Quantencomputer 0403

## Quantengatter IV – (Toffoli-Gate)

Matrix	Schaltungsymbols	Wahrheitstabelle																												
$CCX_{230} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}$		<table><tr><th>q0</th><th>q1</th><th>q2</th><th>CCX<sub>230</sub>(q0,q1,q2)</th></tr><tr><td>0</td><td>000</td><td>000</td></tr><tr><td>0</td><td>001</td><td>001</td></tr><tr><td>0</td><td>010</td><td>010</td></tr><tr><td>0</td><td>011</td><td>011</td></tr><tr><td>1</td><td>100</td><td>100</td></tr><tr><td>1</td><td>101</td><td>101</td></tr><tr><td>1</td><td>110</td><td>111</td></tr><tr><td>1</td><td>111</td><td>110</td></tr></table>	q0	q1	q2	CCX <sub>230</sub> (q0,q1,q2)	0	000	000	0	001	001	0	010	010	0	011	011	1	100	100	1	101	101	1	110	111	1	111	110
		q0	q1	q2	CCX <sub>230</sub> (q0,q1,q2)																									
		0	000	000																										
		0	001	001																										
		0	010	010																										
		0	011	011																										
		1	100	100																										
		1	101	101																										
1	110	111																												
1	111	110																												

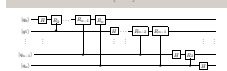
Toffoli-Gatter

Timo Grautstück Quantencomputer – Grundlagen Quantencomputer 0403

## Quanten Fourier-Transformation II

### kontrollierte Rotation [2]

$$R_\theta = \begin{bmatrix} 1 & 0 \\ 0 & e^{i\theta} \end{bmatrix}$$



Schaltung der Quanten Fourier-Transformation vgl. [2]

Timo Grautstück Quantencomputer – Quantenalgorithmen 0403

## Quanten-Phasenschätzung I

- Quantenteil des Shor Algorithmus
- Operiert auf zwei Quantenregister
- Nutzt Phasenschätz (phase kickback) um die Phase eines unitären Operators  $U$  (im Fourierzustand) in ein Quantenregister zu schreiben [2]

### Eigenvektor

$$U|\psi\rangle = e^{i\theta}|\psi\rangle$$

- CPT bestmst  $\theta$

$$\frac{1}{\sqrt{2}} \sum_{j=0}^{2^m-1} e^{i\theta j} |j\rangle = \frac{1}{\sqrt{2}} \left( |0\rangle + e^{i\theta} |1\rangle \right) \otimes \left( |0\rangle + e^{i\theta} |1\rangle \right) \otimes \dots \otimes \left( |0\rangle + e^{i\theta} |1\rangle \right)$$

Timo Grautstück Quantencomputer – Quantenalgorithmen 0403

## Werkzeuge

- IBM Quantum Composer
- Halblabeller bauen
- Hardware
- Quantum Lab
- Qiskit

Timo Grautstück Quantencomputer – Werkzeuge zur Implementierung 0403

## Quanten Fourier-Transformation I

- Wird in vielen anderen Algorithmen genutzt z.B. Shor's
- Transformiert einen Basiszustand  $|x\rangle$  zu einem Fourierzustand  $|X\rangle$
- $N = 2^n$ ,  $n$  ist die Anzahl der genutzten Qubits

### QFT

$$QFT \mapsto |X\rangle = \frac{1}{\sqrt{N}} \sum_{j=0}^{N-1} e^{i2\pi jx/N} |j\rangle$$

### Als Produkt - Gatte [23]

$$|X\rangle = \frac{1}{\sqrt{N}} \left( |0\rangle + e^{i2\pi x/N} |1\rangle \right) \otimes \left( |0\rangle + e^{i2\pi x/N} |1\rangle \right) \otimes \dots \otimes \left( |0\rangle + e^{i2\pi x/N} |1\rangle \right)$$

Timo Grautstück Quantencomputer – Quantenalgorithmen 0403

## Quanten-Phasenschätzung III – (Beispiel)

- $t = 3$  Qubits

### Gatter

$$U = P \left( \theta = \frac{\pi}{8} \right) = \begin{bmatrix} 1 & 0 \\ 0 & e^{i\theta} \end{bmatrix}$$

$$P(\theta) = \begin{bmatrix} 1 & 0 \\ 0 & e^{i\theta} \end{bmatrix} \begin{bmatrix} 1 \\ 1 \end{bmatrix} = e^{i\theta} |1\rangle$$

### Erwartung

$$U|\psi\rangle = e^{i\theta}|\psi\rangle$$

- $\theta = \frac{\pi}{8}$

Timo Grautstück Quantencomputer – Quantenalgorithmen 0403

Danke für die Aufmerksamkeit, gibt es Fragen?

# Literaturverzeichnis I



F. Arute, K. Arya, R. Babbush, D. Bacon, J. C. Bardin, R. Barends, and et al., “Quantum supremacy using a programmable superconducting processor,” *Nature*, vol. 574, pp. 505–510, Oct 2019.



A. Abbas, S. Andersson, A. Asfaw, A. Corcoles, L. Bello, Y. Ben-Haim, M. Bozzo-Rey, and et al., *Learn Quantum Computation Using Qiskit*. IBM, 2020.



M. A. Nielsen and I. L. Chuang, *Quantum Computation and Quantum Information: 10th Anniversary Edition*. Cambridge University Press, 2010.