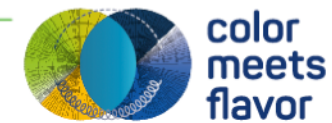


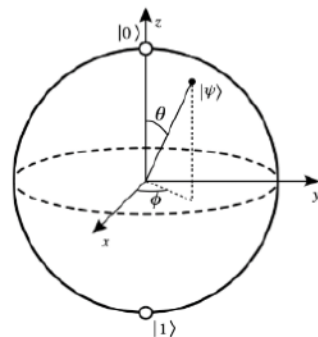
2025 Quantum2025



Ankündigung für das Wintersemester 2025/26

Quanten Computing

Prof. Dr. Alexander Lenz

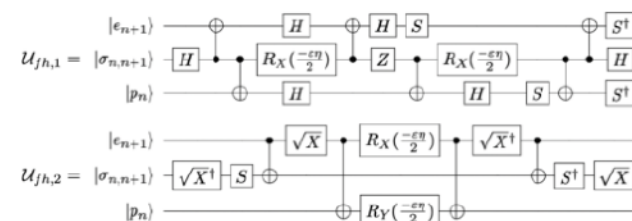


$$|\psi\rangle = \cos \frac{\theta}{2} |0\rangle + e^{i\phi} \sin \frac{\theta}{2} |1\rangle$$

$$\begin{aligned} |00\rangle &\rightarrow |00\rangle, \\ |01\rangle &\rightarrow |01\rangle, \\ |10\rangle &\rightarrow |11\rangle, \\ |11\rangle &\rightarrow |10\rangle. \end{aligned}$$

Diese Vorlesungsreihe gibt eine Einführung in die Grundlagen des Quanten Computing,

2025 feiern wir das 100-jährige Jubiläum der Entdeckung der Quantentheorie. Lange Zeit war Quantentheorie reinste Grundlagenforschung, welche ausschliesslich zum fundamentalen Verständnis unserer Welt diente, aber keinerlei praktische Anwendung hatte. Sie einigen Jahren zeichnet sich nun ein immenses Potential von Quanten Computing ab, welches bei manchen Anwendungen, herkömmliche Supercomputer bei Weitem übertreffen kann. An der Universität Siegen wurde 2010 der erste deutsche Quantencomputer in Betrieb genommen.



In dieser Vorlesung wird eine elementare Einführung in Quanten Computing gegeben und es werden auch praktische Programmierübungen an Quantensimulatoren durchgeführt. Die Vorlesung ist für Schülerinnen und Schüler ab der 10. Klasse geeignet, ebenso für Studierende und Lehrkräfte, sowie mathematisch interessierte Laien, die über Mathematik Kenntnisse auf dem Oberstufen-Niveau verfügen.

Termine:

19.11:



26.11:

3.12: Vertretung

10.12.:



17.12:

7.1.:

14.1:



21.1:

28.1:

4.2.:



10 Termine im Wintersemester 25/26, mittwochs 16-18:
19.11., 26.11., 3.12., 10.12., 17.12., 7.1., 14.1., 21.1., 28.1., 4.2.
Emmy Noether Campus ENC-D-114, 57072 Siegen
Kontakt: alexander.lenz@uni-siegen.de
Weitere Informationen unter:
<https://tp1.physik.uni-siegen.de/mittwochsakademie/>

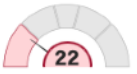


Einführung

Quanten Computing

Was denkt die Welt darüber?

Einführung

 Business Markets Tech Media Calculators Videos			 Watch  Listen 		Sign in
Markets →			Fear & Greed Index →		Latest Market News →
DOW	47,147.48	0.65% ▼	 Extreme Fear is driving the US market		Disney content will return to YouTube TV under a new deal
S&P 500	6,734.11	0.05% ▼			Trump lowers tariffs on coffee, beef and fruits, as Americans' c...
NASDAQ	22,900.59	0.13% ▲			Judge says he'll approve opioid settlement with OxyContin ma...

BUSINESS > TECH • 5 MIN READ

A seismic shift in computing is on the horizon (and it's not AI)

UPDATED NOV 12, 2025 ▾

By [Lisa Eadicicco](#)



The Albany NanoTech Complex where IBM's quantum wafers are fabricated.
(IBM)

Creating revolutionary pharmaceutical drugs, testing new materials for cars and simulating how market scenarios can affect banks — these are just some of the tasks that could take months or years to develop, even with the most advanced computers.

But what if that timeframe could be cut down to minutes or hours?

Einführung

Markets →		
DOW	47,147.48	0.65% ▼
S&P 500	6,734.11	0.05% ▼
NASDAQ	22,900.59	0.13% ▲



BUSINESS > TECH • 5 MIN READ

A seismic shift in comp (and it's not AI)

UPDATED NOV 12, 2025 ▾

By [Lisa Eadicicco](#)



The Albany NanoTech Complex where IBM's quantum wafers are produced (IBM)

Creating revolutionary pharmaceuticals and simulating how market scenarios will unfold are just some of the tasks that could take months or years on today's advanced computers.

But what if that timeframe could be

New Chinese optical quantum chip allegedly 1,000x faster than Nvidia GPUs for processing AI workloads - firm reportedly producing 12,000 wafers per year

News

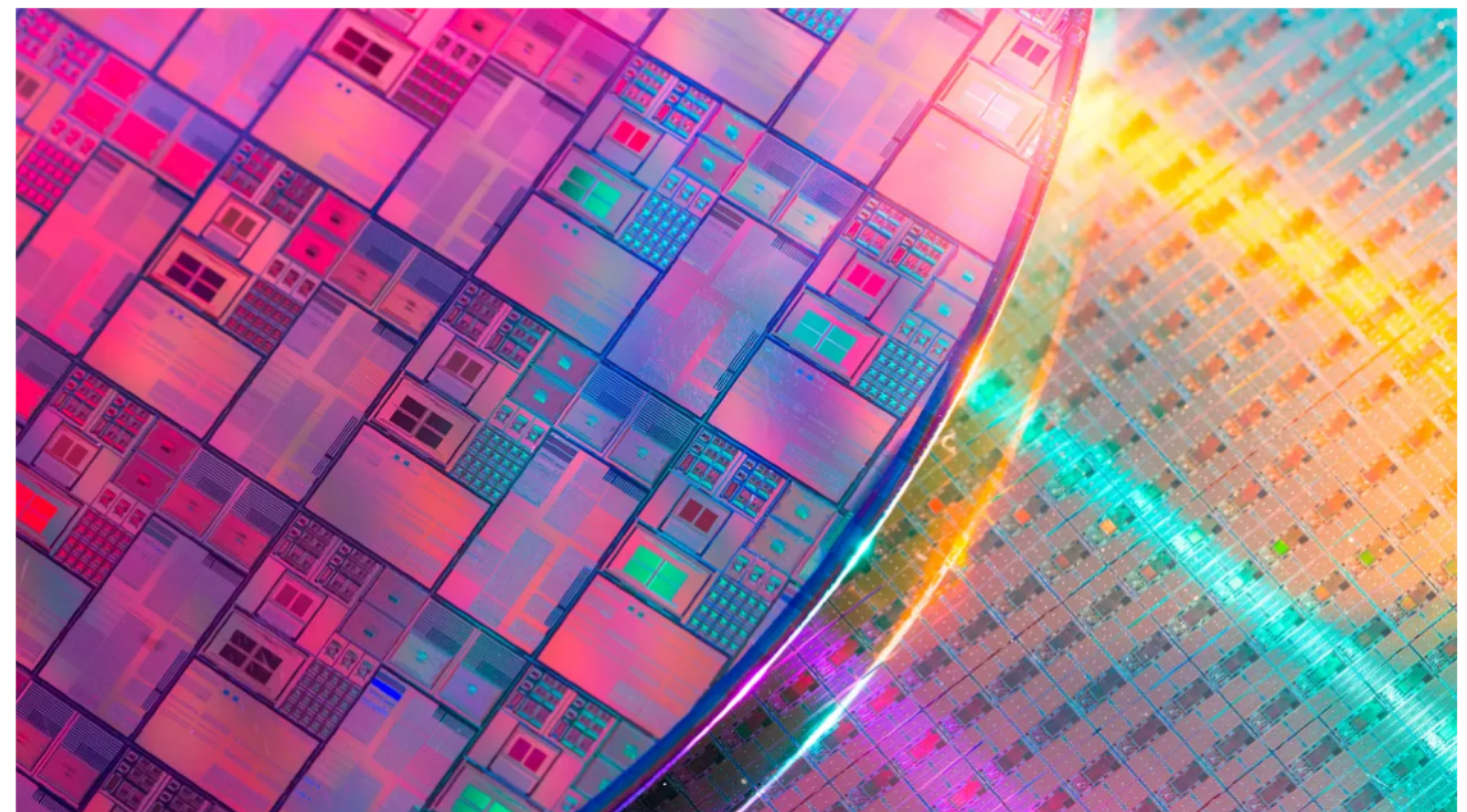
By [Aaron Klotz](#) published 20 hours ago

Nvidia is also investing in similar optical quantum technology.



Comments (5)

When you purchase through links on our site, we may earn an affiliate commission. [Here's how it works.](#)



(Image credit: Shutterstock)

Einführung

New Chinese optical quantum chip allegedly 1,000x faster than Nvidia GPUs for processing AI workloads, firm reportedly

CNN Business Markets Tech Media Calculators Videos
Markets →
DOW 47,147.48 0.65% ▼
Fear & Greed Index →

MARKETS

IBM +0.27%



IBM Pushes Ahead in the Quantum Computing Race

November 13, 2025 — 05:40 am EST

Written by Timothy Green for The Motley Fool →

ntum technology.

iliate commission. [Here's how it](#)

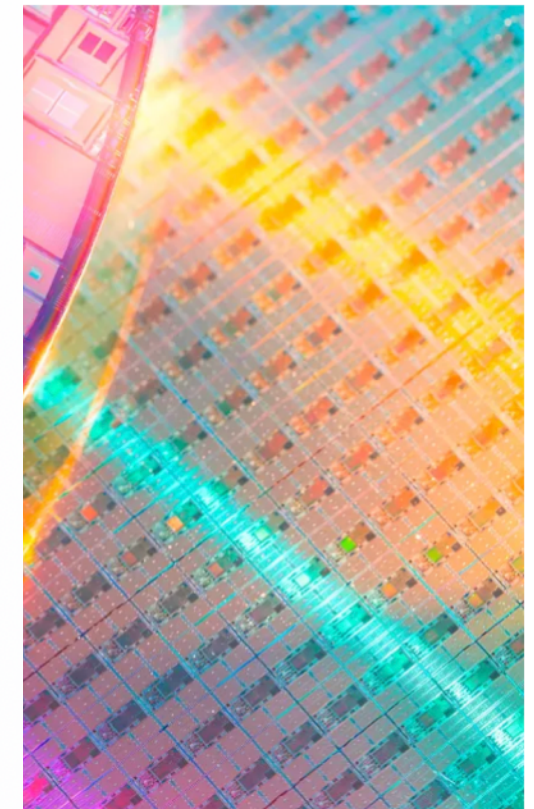
in

f

X

Key Points

- IBM disclosed details on its Nighthawk and Loon quantum processors.
- The company expects quantum advantage next year and fault-tolerance by 2029.



Einführung

Quanten Computing

Warum Siegen?

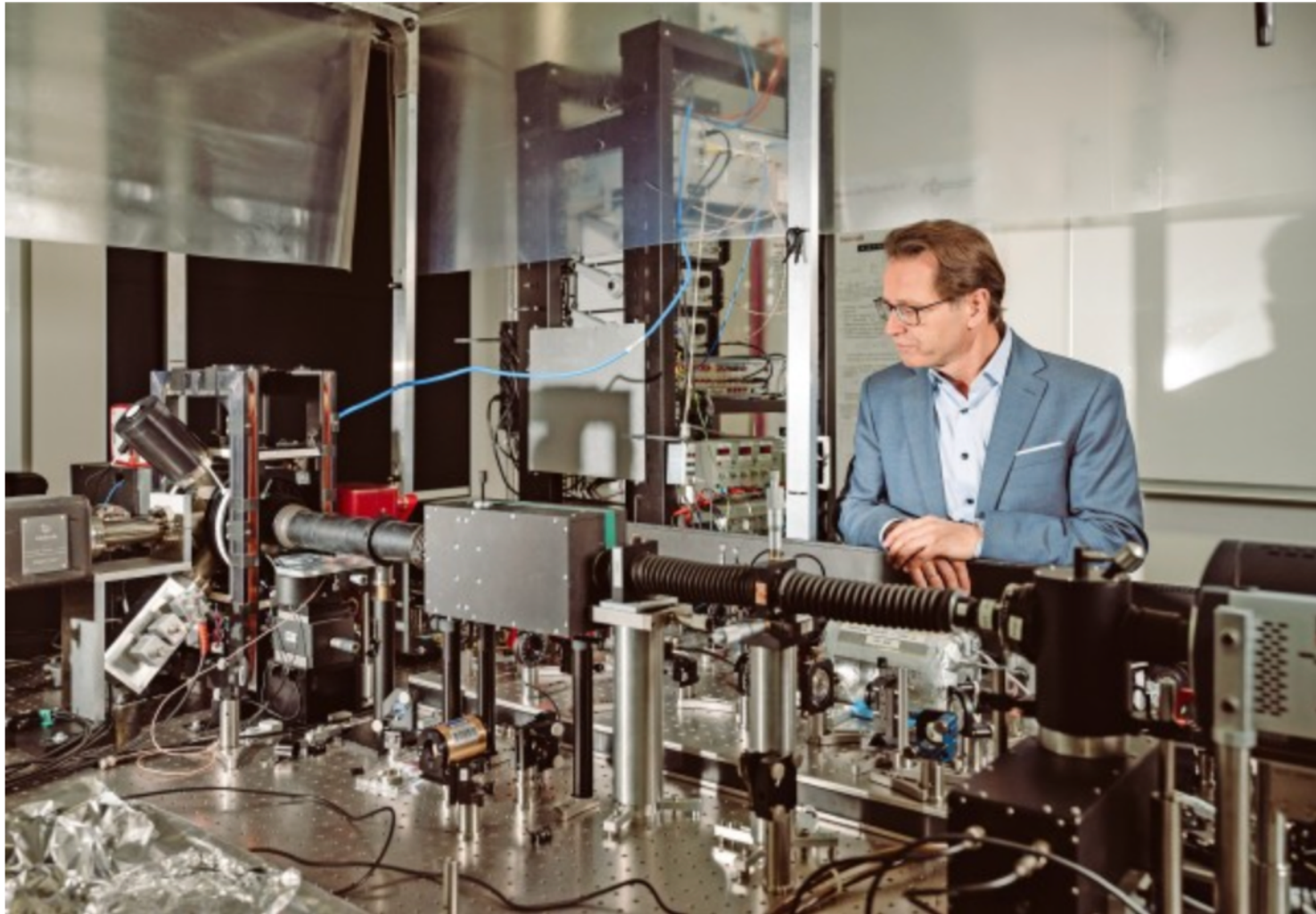
Einführung

F+ START-UP

Ionen in der Falle

VON STEPHAN FINSTERBUSCH - AKTUALISIERT AM 30.11.2023 - 19:54

[Zurück zum Artikel](#)



1/4



Der Professor und sein Werk: Christof Wunderlich neben dem ersten Quantencomputer Deutschlands auf dem Emmy-Noether-Campus der Universität Siegen

Bild: AARON LEITHÄUSER



Einführung



Es gibt eine Quantencomputing Firma in Siegen

Deutscher Quantencomputer

eleQtron erhält ersten »Quantum Effects Award«

12. Oktober 2023, 6:38 Uhr | [Heinz Arnold](#)



Matchmaker+

ANBIETER ZUM THEMA:

START-UP

Ionen in der Falle

VON STEPHAN FINSTERBUSCH · AKTUALISIERT AM 30.11.2023 · 19:54

Zurück zum Artikel

1/4

Der Professor und sein Werk: Christof Wunderlich neben dem ersten Quantencomputer Deutschlands auf dem Emmy-Noether-Campus der Universität Siegen

Bild: AARON LEITHÄUSEN

Einführung

Deutscher Quantencomputer

eleQtron erhält ersten »Quantum Effects Award«

12. Oktober 2023, 6:38 Uhr | Heinz Arnold

In Siegen gibt es riesige Forschungsprojekte in denen Quanten Computing eine wichtige Rolle spielt

Historischer Moment für die Universität Siegen: Exzellenz bescheinigt

22.05.2025 Tanja Hoffmann

Die Universität Siegen erhält den Zuschlag für einen Exzellenzcluster im Bereich der Elementarteilchenphysik. Das Vorhaben mit den Partnern Bonn, Dortmund und Jülich überzeugte in der Exzellenzstrategie des Bundes und der Länder – in den kommenden sieben Jahren wird es mit insgesamt rund 60 Millionen Euro gefördert.

Siegener Physik noch „exzellenter“

27.05.2025 Tanja Hoffmann

Neben dem Exzellenzcluster „Color meets Flavor“ sind Physiker*innen der Universität Siegen ab 2026 noch an einem weiteren Exzellenzcluster beteiligt. Die Arbeitsgruppe „Experimentelle Quantenoptik“ von Prof. Dr. Christof Wunderlich ist Kooperationspartner im Cluster „ML4Q“ zur Quantenforschung.



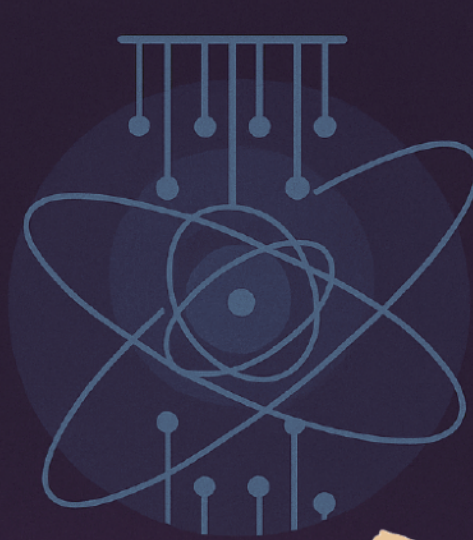
© Sascha Hüttenhain

Einführung



PHYSIK IM APOLLO

QUANTENCOMPUTING
TRIFFT AKROBATIK



23.11.25 18:00
24.11.25 10:00
25.11.25 19:30

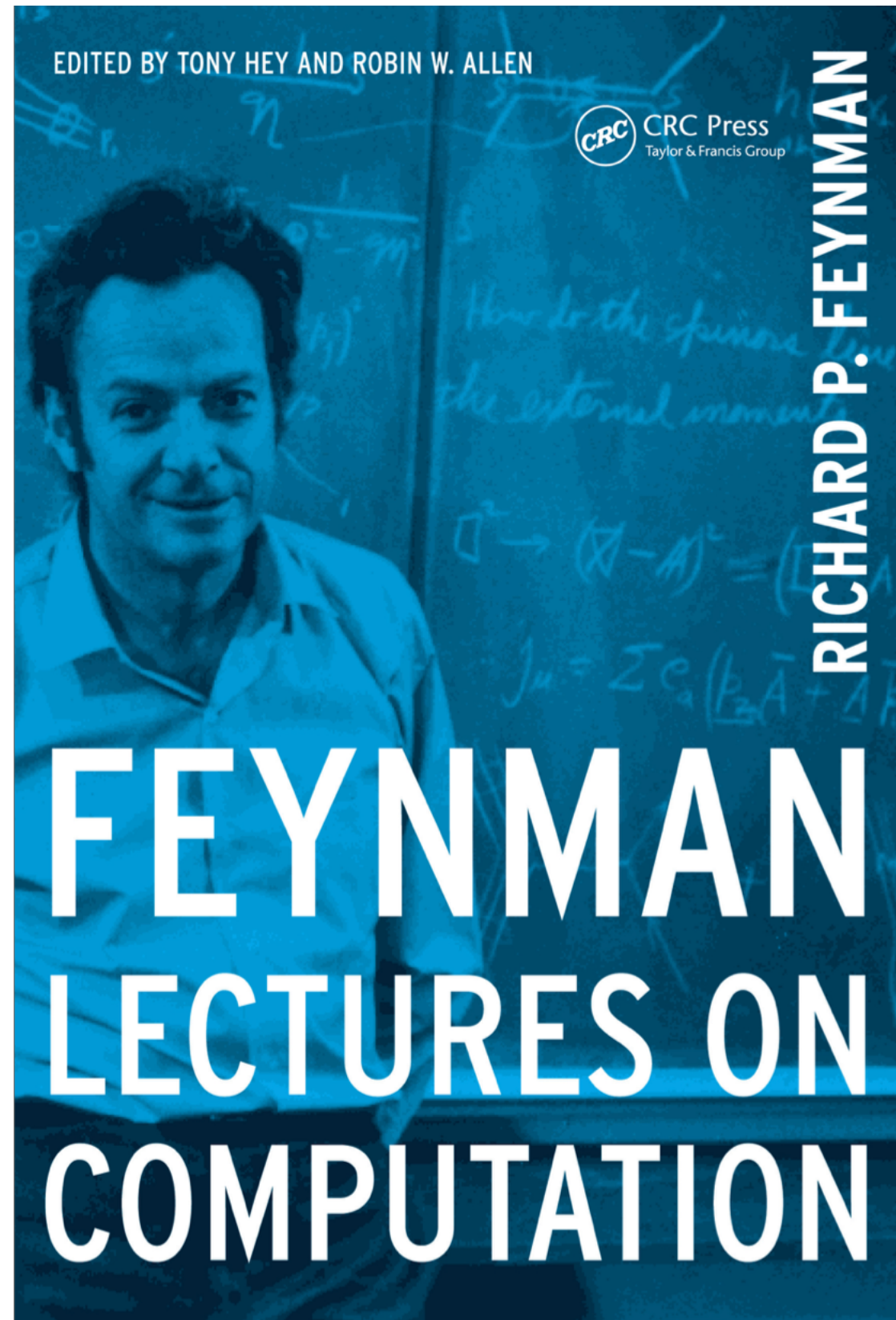


Einführung

Quanten Computing

Fangen wir ganz vorne an

Einführung



Einführung

1. Was sind klassische Bits

2. Klassische Operatoren/Gates

3. Reversible Operatoren

Klassische Bits

Klassische Computer rechnen mit “Bits”

Klassische Bits

Klassische Computer rechnen mit “Bits”

**Ein Bit ist ein Objekt, das nur zwei Zustände haben kann
Null oder Eins**

Klassische Bits

Klassische Computer rechnen mit “Bits”

**Ein Bit ist ein Objekt, das nur zwei Zustände haben kann
Null oder Eins**

Beispiele:

- **Münze: bei Wurf endet man mit Kopf oder mit Zahl, z.B. Kopf = 1, Zahl = 0**

Klassische Bits

Klassische Computer rechnen mit “Bits”

**Ein Bit ist ein Objekt, das nur zwei Zustände haben kann
Null oder Eins**

Beispiele:

- **Münze: bei Wurf endet man mit Kopf oder mit Zahl, z.B. Kopf = 1, Zahl = 0**
- **Stromkreis: Strom fließt = 1, kein Strom = 0**

Klassische Bits

Klassische Computer rechnen mit “Bits”

**Ein Bit ist ein Objekt, das nur zwei Zustände haben kann
Null oder Eins**

Beispiele:

- **Münze:** bei Wurf endet man mit Kopf oder mit Zahl, z.B. Kopf = 1, Zahl = 0
- **Stromkreis:** Strom fließt = 1, kein Strom = 0



Klassische Bits

Klassische Computer rechnen mit “Bits”

**Ein Bit ist ein Objekt, das nur zwei Zustände haben kann
Null oder Eins**

Beispiele:

- **Münze: bei Wurf endet man mit Kopf oder mit Zahl, z.B. Kopf = 1, Zahl = 0**
- **Stromkreis: Strom fließt = 1, kein Strom = 0**



Klassische Bits

Mit Bits kann man zählen

Klassische Bits

Mit Bits kann man zählen

Dezimalsystem: Basis 10

Klassische Bits

Mit Bits kann man zählen

Dezimalsystem: Basis 10

10 Ziffern: 0,1,2,3,4,5,6,7,8,9

Klassische Bits

Mit Bits kann man zählen

Dezimalsystem: Basis 10

10 Ziffern: 0,1,2,3,4,5,6,7,8,9

2025 =

Klassische Bits

Mit Bits kann man zählen

Dezimalsystem: Basis 10

10 Ziffern: 0,1,2,3,4,5,6,7,8,9

$$2025 = 2 * 1000 + 0 * 100 + 2 * 10 + 5 * 1$$

Klassische Bits

Mit Bits kann man zählen

Dezimalsystem: Basis 10

10 Ziffern: 0,1,2,3,4,5,6,7,8,9

$$\begin{aligned} 2025 &= 2 * 1000 + 0 * 100 + 2 * 10 + 5 * 1 \\ &= 2 * 10^3 + 0 * 10^2 + 2 * 10^1 + 5 * 10^0 \end{aligned}$$

Klassische Bits

Mit Bits kann man zählen

Dezimalsystem: Basis 10

10 Ziffern: 0,1,2,3,4,5,6,7,8,9

$$\begin{aligned} 2025 &= 2 * 1000 + 0 * 100 + 2 * 10 + 5 * 1 \\ &= 2 * 10^3 + 0 * 10^2 + 2 * 10^1 + 5 * 10^0 \end{aligned}$$

- 1. Stelle von rechts: $10^0 = 1$**
- 2. Stelle von rechts: $10^1 = 10$**
- 3. Stelle von rechts: $10^2 = 100$**
- 4. Stelle von rechts: $10^3 = 1000$**
- 5.**

Klassische Bits

Mit Bits kann man zählen

Dezimalsystem: Basis 10

Binärsystem: Basis 2

10 Ziffern: 0,1,2,3,4,5,6,7,8,9

$$\begin{aligned} 2025 &= 2 * 1000 + 0 * 100 + 2 * 10 + 5 * 1 \\ &= 2 * 10^3 + 0 * 10^2 + 2 * 10^1 + 5 * 10^0 \end{aligned}$$

- 1. Stelle von rechts: $10^0 = 1$**
- 2. Stelle von rechts: $10^1 = 10$**
- 3. Stelle von rechts: $10^2 = 100$**
- 4. Stelle von rechts: $10^3 = 1000$**
- 5.**

Klassische Bits

Mit Bits kann man zählen

Dezimalsystem: Basis 10

10 Ziffern: 0,1,2,3,4,5,6,7,8,9

$$\begin{aligned} 2025 &= 2 * 1000 + 0 * 100 + 2 * 10 + 5 * 1 \\ &= 2 * 10^3 + 0 * 10^2 + 2 * 10^1 + 5 * 10^0 \end{aligned}$$

- 1. Stelle von rechts: $10^0 = 1$**
- 2. Stelle von rechts: $10^1 = 10$**
- 3. Stelle von rechts: $10^2 = 100$**
- 4. Stelle von rechts: $10^3 = 1000$**
- 5.**

Binärsystem: Basis 2

2 Ziffern: 0,1

Klassische Bits

Mit Bits kann man zählen

Dezimalsystem: Basis 10

10 Ziffern: 0,1,2,3,4,5,6,7,8,9

$$\begin{aligned} 2025 &= 2 * 1000 + 0 * 100 + 2 * 10 + 5 * 1 \\ &= 2 * 10^3 + 0 * 10^2 + 2 * 10^1 + 5 * 10^0 \end{aligned}$$

- 1. Stelle von rechts: $10^0 = 1$**
- 2. Stelle von rechts: $10^1 = 10$**
- 3. Stelle von rechts: $10^2 = 100$**
- 4. Stelle von rechts: $10^3 = 1000$**
- 5.**

Binärsystem: Basis 2

2 Ziffern: 0,1

$$1001_2 =$$

Klassische Bits

Mit Bits kann man zählen

Dezimalsystem: Basis 10

10 Ziffern: 0,1,2,3,4,5,6,7,8,9

$$\begin{aligned} 2025 &= 2 * 1000 + 0 * 100 + 2 * 10 + 5 * 1 \\ &= 2 * 10^3 + 0 * 10^2 + 2 * 10^1 + 5 * 10^0 \end{aligned}$$

- 1. Stelle von rechts: $10^0 = 1$**
- 2. Stelle von rechts: $10^1 = 10$**
- 3. Stelle von rechts: $10^2 = 100$**
- 4. Stelle von rechts: $10^3 = 1000$**
- 5.**

Binärsystem: Basis 2

2 Ziffern: 0,1

$$1001_2 = 1 * 2^3 + 0 * 2^2 + 0 * 2^1 + 1 * 2^0$$

Klassische Bits

Mit Bits kann man zählen

Dezimalsystem: Basis 10

10 Ziffern: 0,1,2,3,4,5,6,7,8,9

$$\begin{aligned} 2025 &= 2 * 1000 + 0 * 100 + 2 * 10 + 5 * 1 \\ &= 2 * 10^3 + 0 * 10^2 + 2 * 10^1 + 5 * 10^0 \end{aligned}$$

- 1. Stelle von rechts: $10^0 = 1$**
- 2. Stelle von rechts: $10^1 = 10$**
- 3. Stelle von rechts: $10^2 = 100$**
- 4. Stelle von rechts: $10^3 = 1000$**
- 5.**

Binärsystem: Basis 2

2 Ziffern: 0,1

$$\begin{aligned} 1001_2 &= 1 * 2^3 + 0 * 2^2 + 0 * 2^1 + 1 * 2^0 \\ &= 1 * 8 + 0 * 4 + 0 * 2 + 1 * 1 = 9 \end{aligned}$$

Klassische Bits

Mit Bits kann man zählen

Dezimalsystem: Basis 10

10 Ziffern: 0,1,2,3,4,5,6,7,8,9

$$\begin{aligned} 2025 &= 2 * 1000 + 0 * 100 + 2 * 10 + 5 * 1 \\ &= 2 * 10^3 + 0 * 10^2 + 2 * 10^1 + 5 * 10^0 \end{aligned}$$

- 1. Stelle von rechts: $10^0 = 1$**
- 2. Stelle von rechts: $10^1 = 10$**
- 3. Stelle von rechts: $10^2 = 100$**
- 4. Stelle von rechts: $10^3 = 1000$**
- 5.**

Binärsystem: Basis 2

2 Ziffern: 0,1

$$\begin{aligned} 1001_2 &= 1 * 2^3 + 0 * 2^2 + 0 * 2^1 + 1 * 2^0 \\ &= 1 * 8 + 0 * 4 + 0 * 2 + 1 * 1 = 9 \end{aligned}$$

- 1. Stelle von rechts: $2^0 = 1$**
- 2. Stelle von rechts: $2^1 = 2$**
- 3. Stelle von rechts: $2^2 = 4$**
- 4. Stelle von rechts: $2^3 = 8$**
- 5.**

Klassische Bits

1. Wie lautet die Dezimalzahl 49 im System mit der Basis 7

Klassische Bits

1. Wie lautet die Dezimalzahl 49 im System mit der Basis 7

$$100_7 \equiv 1 * 7^2 + 0 * 7^1 + 0 * 7^0 = 49$$

Klassische Bits

1. Wie lautet die Dezimalzahl 49 im System mit der Basis 7

$$100_7 \equiv 1 * 7^2 + 0 * 7^1 + 0 * 7^0 = 49$$

2. In welchen System lautet die Dezimalzahl 91, 111

Klassische Bits

1. Wie lautet die Dezimalzahl 49 im System mit der Basis 7

$$100_7 \equiv 1 * 7^2 + 0 * 7^1 + 0 * 7^0 = 49$$

2. In welchen System lautet die Dezimalzahl 91, 111

$$111_9 \equiv 1 * 9^2 + 1 * 9^1 + 1 * 9^0 = 81 + 9 + 1 = 91$$

Klassische Bits

1. Wie lautet die Dezimalzahl 49 im System mit der Basis 7

$$100_7 \equiv 1 * 7^2 + 0 * 7^1 + 0 * 7^0 = 49$$

2. In welchen System lautet die Dezimalzahl 91, 111

$$111_9 \equiv 1 * 9^2 + 1 * 9^1 + 1 * 9^0 = 81 + 9 + 1 = 91$$

3. Die Ausserirdischen vom Exo-Planeten Kepler-452b, der 1800 Lichtjahre von uns entfernt ist, sehen menschenähnlich aus, allerdings besitzen sie 12 Finger. Wie wird vermutlich auf Kepler-452 die Zahl 160 dargestellt werden?

Klassische Bits

1. **Wie lautet die Dezimalzahl 49 im System mit der Basis 7**

$$100_7 \equiv 1 * 7^2 + 0 * 7^1 + 0 * 7^0 = 49$$

2. **In welchen System lautet die Dezimalzahl 91, 111**

$$111_9 \equiv 1 * 9^2 + 1 * 9^1 + 1 * 9^0 = 81 + 9 + 1 = 91$$

3. **Die Ausserirdischen vom Exo-Planeten Kepler-452b, der 1800 Lichtjahre von uns entfernt ist, sehen menschenähnlich aus, allerdings besitzen sie 12 Finger. Wie wird vermutlich auf Kepler-452 die Zahl 160 dargestellt werden?**

$$114_{12} \equiv 1 * 12^2 + 1 * 12^1 + 4 * 12^0 = 144 + 12 + 4 = 160$$

Klassische Bits

800
+950

Addition von Dezimalzahlen

Klassische Bits

800
+950

Addition von Dezimalzahlen

$0 + 0 = 0$ -> 1 Stelle => kein Übertrag

Klassische Bits

800
+950

Addition von Dezimalzahlen

$0 + 0 = 0$ -> 1 Stelle => kein Übertrag

$0 + 5 = 5$ -> 1 Stelle => kein Übertrag

Klassische Bits

800
+950
— — —

Addition von Dezimalzahlen

$0 + 0 = 0 \rightarrow 1 \text{ Stelle} \Rightarrow \text{kein Übertrag}$

$0 + 5 = 5 \rightarrow 1 \text{ Stelle} \Rightarrow \text{kein Übertrag}$

$8 + 9 = 17 \rightarrow 2 \text{ Stellen} \Rightarrow \text{Übertrag}$

Klassische Bits

800
+950

1750

Addition von Dezimalzahlen

$0 + 0 = 0 \rightarrow 1 \text{ Stelle} \Rightarrow \text{kein Übertrag}$

$0 + 5 = 5 \rightarrow 1 \text{ Stelle} \Rightarrow \text{kein Übertrag}$

$8 + 9 = 17 \rightarrow 2 \text{ Stellen} \Rightarrow \text{Übertrag}$

Klassische Bits

800
+950
— — —
1750

Addition von Dezimalzahlen

$0 + 0 = 0 \rightarrow 1 \text{ Stelle} \Rightarrow \text{kein Übertrag}$

$0 + 5 = 5 \rightarrow 1 \text{ Stelle} \Rightarrow \text{kein Übertrag}$

$8 + 9 = 17 \rightarrow 2 \text{ Stellen} \Rightarrow \text{Übertrag}$

Addition von Binärzahlen

0111001
+0111001
— — — — —

Klassische Bits

800
+950

1750

Addition von Dezimalzahlen

$0 + 0 = 0 \rightarrow 1 \text{ Stelle} \Rightarrow \text{kein Übertrag}$

$0 + 5 = 5 \rightarrow 1 \text{ Stelle} \Rightarrow \text{kein Übertrag}$

$8 + 9 = 17 \rightarrow 2 \text{ Stellen} \Rightarrow \text{Übertrag}$

Addition von Binärzahlen

$0 + 0 = 0 = 2^0 \equiv 0_2 \rightarrow 1 \text{ Stelle} \Rightarrow \text{kein Übertrag}$

0111001
+0111001

Klassische Bits

800
+950

1750

Addition von Dezimalzahlen

$0 + 0 = 0 \rightarrow 1 \text{ Stelle} \Rightarrow \text{kein Übertrag}$

$0 + 5 = 5 \rightarrow 1 \text{ Stelle} \Rightarrow \text{kein Übertrag}$

$8 + 9 = 17 \rightarrow 2 \text{ Stellen} \Rightarrow \text{Übertrag}$

0111001
+0111001

Addition von Binärzahlen

$0 + 0 = 0 = 2^0 \equiv 0_2 \rightarrow 1 \text{ Stelle} \Rightarrow \text{kein Übertrag}$

$0 + 1 = 1 = 2^1 \equiv 1_2 \rightarrow 1 \text{ Stelle} \Rightarrow \text{kein Übertrag}$

Klassische Bits

800
+950
— — —
1750

Addition von Dezimalzahlen

$0 + 0 = 0 \rightarrow 1 \text{ Stelle} \Rightarrow \text{kein Übertrag}$

$0 + 5 = 5 \rightarrow 1 \text{ Stelle} \Rightarrow \text{kein Übertrag}$

$8 + 9 = 17 \rightarrow 2 \text{ Stellen} \Rightarrow \text{Übertrag}$

0111001
+0111001
— — — — —

Addition von Binärzahlen

$0 + 0 = 0 = 2^0 \equiv 0_2 \rightarrow 1 \text{ Stelle} \Rightarrow \text{kein Übertrag}$

$0 + 1 = 1 = 2^1 \equiv 1_2 \rightarrow 1 \text{ Stelle} \Rightarrow \text{kein Übertrag}$

$1 + 1 = 2 = 2^1 \equiv 10_2 \rightarrow 2 \text{ Stellen} \Rightarrow \text{Übertrag, Rest } 0$

Klassische Bits

800
+950

1750

Addition von Dezimalzahlen

$0 + 0 = 0 \rightarrow 1 \text{ Stelle} \Rightarrow \text{kein Übertrag}$

$0 + 5 = 5 \rightarrow 1 \text{ Stelle} \Rightarrow \text{kein Übertrag}$

$8 + 9 = 17 \rightarrow 2 \text{ Stellen} \Rightarrow \text{Übertrag}$

0111001
+0111001

Addition von Binärzahlen

$0 + 0 = 0 = 2^0 \equiv 0_2 \rightarrow 1 \text{ Stelle} \Rightarrow \text{kein Übertrag}$

$0 + 1 = 1 = 2^1 \equiv 1_2 \rightarrow 1 \text{ Stelle} \Rightarrow \text{kein Übertrag}$

$1 + 1 = 2 = 2^1 \equiv 10_2 \rightarrow 2 \text{ Stellen} \Rightarrow \text{Übertrag, Rest 0}$

$1 + 1 + 1 = 3 \equiv 11_2 \rightarrow 2 \text{ Stellen} \Rightarrow \text{Übertrag, Rest 1}$

Klassische Bits

800
+950

1750

Addition von Dezimalzahlen

$0 + 0 = 0 \rightarrow 1 \text{ Stelle} \Rightarrow \text{kein Übertrag}$

$0 + 5 = 5 \rightarrow 1 \text{ Stelle} \Rightarrow \text{kein Übertrag}$

$8 + 9 = 17 \rightarrow 2 \text{ Stellen} \Rightarrow \text{Übertrag}$

0111001
+0111001

1110010

Addition von Binärzahlen

$0 + 0 = 0 = 2^0 \equiv 0_2 \rightarrow 1 \text{ Stelle} \Rightarrow \text{kein Übertrag}$

$0 + 1 = 1 = 2^1 \equiv 1_2 \rightarrow 1 \text{ Stelle} \Rightarrow \text{kein Übertrag}$

$1 + 1 = 2 = 2^1 \equiv 10_2 \rightarrow 2 \text{ Stellen} \Rightarrow \text{Übertrag, Rest 0}$

$1 + 1 + 1 = 3 \equiv 11_2 \rightarrow 2 \text{ Stellen} \Rightarrow \text{Übertrag, Rest 1}$

Klassische Bits

800
+950

1750

Addition von Dezimalzahlen

$0 + 0 = 0 \rightarrow 1 \text{ Stelle} \Rightarrow \text{kein Übertrag}$

$0 + 5 = 5 \rightarrow 1 \text{ Stelle} \Rightarrow \text{kein Übertrag}$

$8 + 9 = 17 \rightarrow 2 \text{ Stellen} \Rightarrow \text{Übertrag}$

0111001
+0111001

1110010

Addition von Binärzahlen

$0 + 0 = 0 = 2^0 \equiv 0_2 \rightarrow 1 \text{ Stelle} \Rightarrow \text{kein Übertrag}$

$0 + 1 = 1 = 2^1 \equiv 1_2 \rightarrow 1 \text{ Stelle} \Rightarrow \text{kein Übertrag}$

$1 + 1 = 2 = 2^1 \equiv 10_2 \rightarrow 2 \text{ Stellen} \Rightarrow \text{Übertrag, Rest 0}$

$1 + 1 + 1 = 3 \equiv 11_2 \rightarrow 2 \text{ Stellen} \Rightarrow \text{Übertrag, Rest 1}$

Test: 0111001=
 1110010=

Klassische Bits

800
+950

1750

Addition von Dezimalzahlen

$0 + 0 = 0 \rightarrow 1 \text{ Stelle} \Rightarrow \text{kein Übertrag}$

$0 + 5 = 5 \rightarrow 1 \text{ Stelle} \Rightarrow \text{kein Übertrag}$

$8 + 9 = 17 \rightarrow 2 \text{ Stellen} \Rightarrow \text{Übertrag}$

0111001
+0111001

1110010

Addition von Binärzahlen

$0 + 0 = 0 = 2^0 \equiv 0_2 \rightarrow 1 \text{ Stelle} \Rightarrow \text{kein Übertrag}$

$0 + 1 = 1 = 2^1 \equiv 1_2 \rightarrow 1 \text{ Stelle} \Rightarrow \text{kein Übertrag}$

$1 + 1 = 2 = 2^1 \equiv 10_2 \rightarrow 2 \text{ Stellen} \Rightarrow \text{Übertrag, Rest 0}$

$1 + 1 + 1 = 3 \equiv 11_2 \rightarrow 2 \text{ Stellen} \Rightarrow \text{Übertrag, Rest 1}$

Test: 0111001= 1+8+16+32
 1110010=

Klassische Bits

800
+950

1750

Addition von Dezimalzahlen

$0 + 0 = 0 \rightarrow 1 \text{ Stelle} \Rightarrow \text{kein Übertrag}$

$0 + 5 = 5 \rightarrow 1 \text{ Stelle} \Rightarrow \text{kein Übertrag}$

$8 + 9 = 17 \rightarrow 2 \text{ Stellen} \Rightarrow \text{Übertrag}$

0111001
+0111001

1110010

Addition von Binärzahlen

$0 + 0 = 0 = 2^0 \equiv 0_2 \rightarrow 1 \text{ Stelle} \Rightarrow \text{kein Übertrag}$

$0 + 1 = 1 = 2^1 \equiv 1_2 \rightarrow 1 \text{ Stelle} \Rightarrow \text{kein Übertrag}$

$1 + 1 = 2 = 2^1 \equiv 10_2 \rightarrow 2 \text{ Stellen} \Rightarrow \text{Übertrag, Rest 0}$

$1 + 1 + 1 = 3 \equiv 11_2 \rightarrow 2 \text{ Stellen} \Rightarrow \text{Übertrag, Rest 1}$

Test: 0111001 = $1+8+16+32=57$;
1110010 =

Klassische Bits

800
+950

1750

Addition von Dezimalzahlen

$0 + 0 = 0 \rightarrow 1 \text{ Stelle} \Rightarrow \text{kein Übertrag}$

$0 + 5 = 5 \rightarrow 1 \text{ Stelle} \Rightarrow \text{kein Übertrag}$

$8 + 9 = 17 \rightarrow 2 \text{ Stellen} \Rightarrow \text{Übertrag}$

0111001
+0111001

1110010

Addition von Binärzahlen

$0 + 0 = 0 = 2^0 \equiv 0_2 \rightarrow 1 \text{ Stelle} \Rightarrow \text{kein Übertrag}$

$0 + 1 = 1 = 2^1 \equiv 1_2 \rightarrow 1 \text{ Stelle} \Rightarrow \text{kein Übertrag}$

$1 + 1 = 2 = 2^1 \equiv 10_2 \rightarrow 2 \text{ Stellen} \Rightarrow \text{Übertrag, Rest 0}$

$1 + 1 + 1 = 3 \equiv 11_2 \rightarrow 2 \text{ Stellen} \Rightarrow \text{Übertrag, Rest 1}$

Test: **0111001 = $1+8+16+32=57$; $57*2=114$**
 1110010 =

Klassische Bits

800
+950

1750

Addition von Dezimalzahlen

$0 + 0 = 0 \rightarrow 1 \text{ Stelle} \Rightarrow \text{kein Übertrag}$

$0 + 5 = 5 \rightarrow 1 \text{ Stelle} \Rightarrow \text{kein Übertrag}$

$8 + 9 = 17 \rightarrow 2 \text{ Stellen} \Rightarrow \text{Übertrag}$

0111001
+0111001

1110010

Addition von Binärzahlen

$0 + 0 = 0 = 2^0 \equiv 0_2 \rightarrow 1 \text{ Stelle} \Rightarrow \text{kein Übertrag}$

$0 + 1 = 1 = 2^1 \equiv 1_2 \rightarrow 1 \text{ Stelle} \Rightarrow \text{kein Übertrag}$

$1 + 1 = 2 = 2^1 \equiv 10_2 \rightarrow 2 \text{ Stellen} \Rightarrow \text{Übertrag, Rest 0}$

$1 + 1 + 1 = 3 \equiv 11_2 \rightarrow 2 \text{ Stellen} \Rightarrow \text{Übertrag, Rest 1}$

Test: **0111001 = 1+8+16+32 = 57; 57*2 = 114**
 1110010 = 2+16+32+64

Klassische Bits

800
+950

1750

Addition von Dezimalzahlen

$0 + 0 = 0 \rightarrow 1 \text{ Stelle} \Rightarrow \text{kein Übertrag}$

$0 + 5 = 5 \rightarrow 1 \text{ Stelle} \Rightarrow \text{kein Übertrag}$

$8 + 9 = 17 \rightarrow 2 \text{ Stellen} \Rightarrow \text{Übertrag}$

0111001
+0111001

1110010

Addition von Binärzahlen

$0 + 0 = 0 = 2^0 \equiv 0_2 \rightarrow 1 \text{ Stelle} \Rightarrow \text{kein Übertrag}$

$0 + 1 = 1 = 2^1 \equiv 1_2 \rightarrow 1 \text{ Stelle} \Rightarrow \text{kein Übertrag}$

$1 + 1 = 2 = 2^1 \equiv 10_2 \rightarrow 2 \text{ Stellen} \Rightarrow \text{Übertrag, Rest 0}$

$1 + 1 + 1 = 3 \equiv 11_2 \rightarrow 2 \text{ Stellen} \Rightarrow \text{Übertrag, Rest 1}$

Test: $0111001 = 1+8+16+32 = 57; 57*2 = 114$
 $1110010 = 2+16+32+64 = 114$

Einführung

1. Was sind klassische Bits

2. Klassische Operatoren/Gates

3. Reversible Operatoren

Operationen mit 1 klassischen Bit

Operationen mit 1 klassischen Bit

Eingang

Ausgang

Bit A



Bit C

Operationen mit 1 klassischen Bit

Eingang

Ausgang

Bit A



Bit C

Identität

A	C
0	0
1	1

Operationen mit 1 klassischen Bit

Eingang

Ausgang

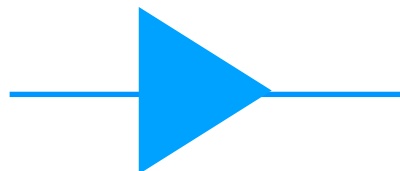
Bit A

Bit C



Identität

A	C
0	0
1	1



Operationen mit 1 klassischen Bit

Eingang

Ausgang

Bit A

Bit C

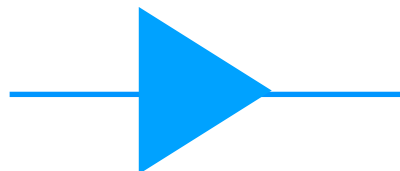


Identität

A	C
0	0
1	1

Not-Operation

A	C
0	1
1	0



Operationen mit 1 klassischen Bit

Eingang

Ausgang

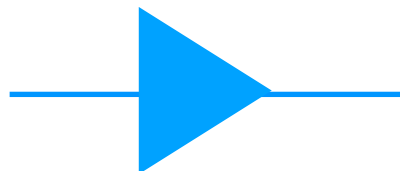
Bit A

Bit C



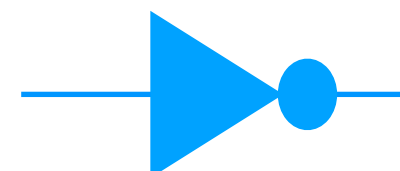
Identität

A	C
0	0
1	1



Not-Operation

A	C
0	1
1	0



Operationen mit 2 klassischen Bits

Eingang

Ausgang

Bit A

Bit B

Bit C



Operationen mit 2 klassischen Bits

Eingang

Ausgang

Bit A

Bit B



Bit C

AND

A	B	C
0	0	0
0	1	0
1	0	0
1	1	1

Operationen mit 2 klassischen Bits

Eingang

Ausgang

Bit A

Bit B

Bit C



AND

A	B	C
0	0	0
0	1	0
1	0	0
1	1	1

OR

A	B	C
0	0	0
0	1	1
1	0	1
1	1	1

Operationen mit 2 klassischen Bits

Eingang

Ausgang

Bit A

Bit B

Bit C



AND

A	B	C
0	0	0
0	1	0
1	0	0
1	1	1

OR

A	B	C
0	0	0
0	1	1
1	0	1
1	1	1

(E)XOR

A	B	C
0	0	0
0	1	1
1	0	1
1	1	0

Operationen mit 2 klassischen Bits

Eingang

Ausgang

Bit A

Bit B

Bit C



AND

A	B	C
0	0	0
0	1	0
1	0	0
1	1	1

OR

A	B	C
0	0	0
0	1	1
1	0	1
1	1	1

(E)XOR

A	B	C
0	0	0
0	1	1
1	0	1
1	1	0

NAND

A	B	C
0	0	1
0	1	1
1	0	1
1	1	0

Operationen mit 2 klassischen Bits

Eingang

Ausgang

Bit A

Bit B

Bit C



AND

A	B	C
0	0	0
0	1	0
1	0	0
1	1	1

OR

A	B	C
0	0	0
0	1	1
1	0	1
1	1	1

(E)XOR

A	B	C
0	0	0
0	1	1
1	0	1
1	1	0

NAND

A	B	C
0	0	1
0	1	1
1	0	1
1	1	0

NOR

A	B	C
0	0	1
0	1	0
1	0	0
1	1	0

Operationen mit 2 klassischen Bits

Eingang

Ausgang

Bit A

Bit B

Bit C



AND

A	B	C
0	0	0
0	1	0
1	0	0
1	1	1

OR

A	B	C
0	0	0
0	1	1
1	0	1
1	1	1

(E)XOR

A	B	C
0	0	0
0	1	1
1	0	1
1	1	0

NAND

A	B	C
0	0	1
0	1	1
1	0	1
1	1	0

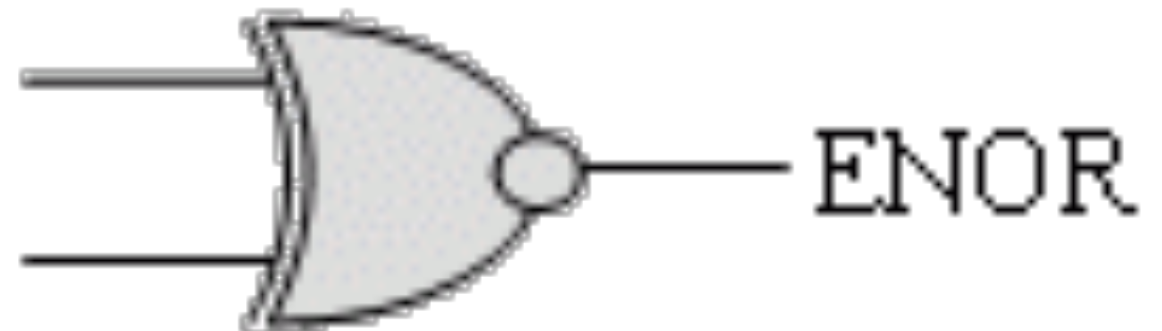
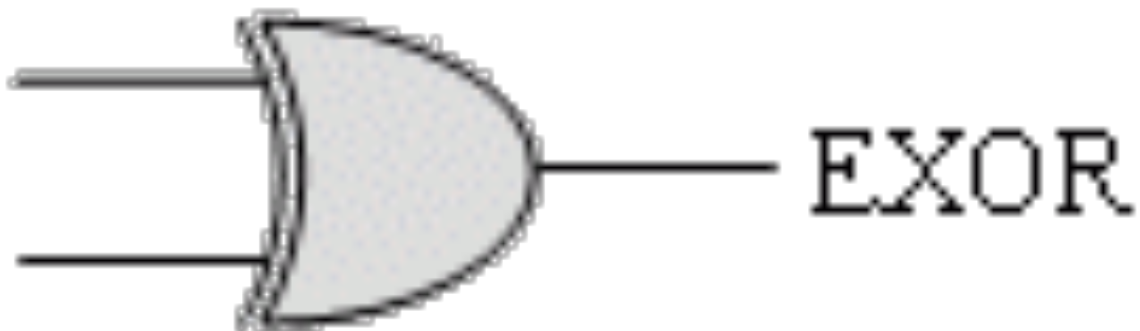
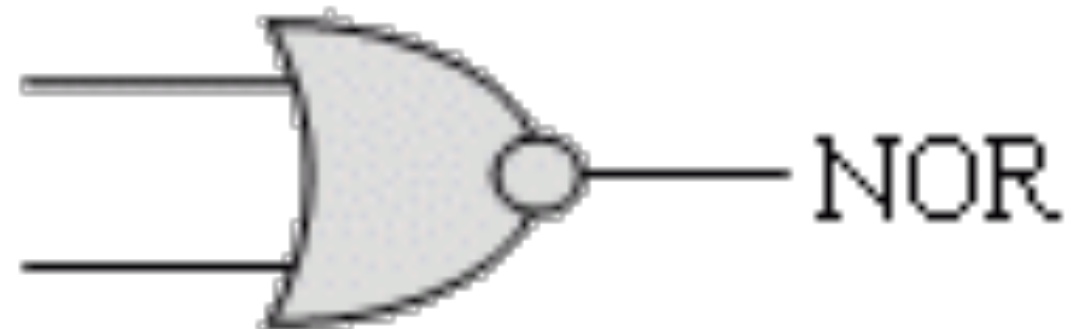
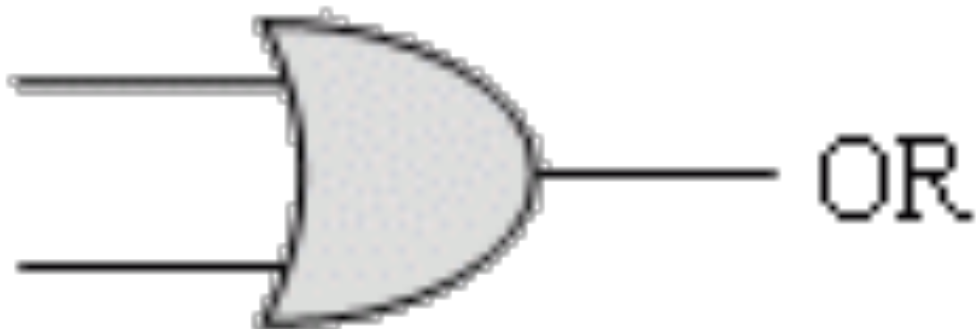
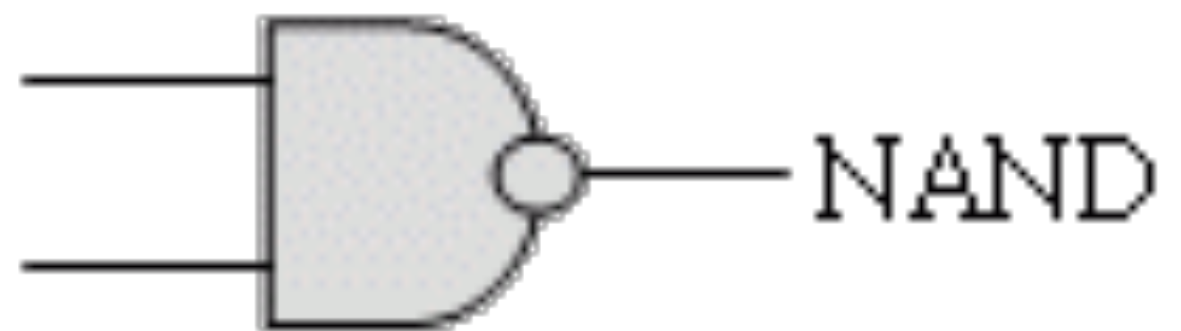
NOR

A	B	C
0	0	1
0	1	0
1	0	0
1	1	0

N(E)XOR

A	B	C
0	0	1
0	1	0
1	0	0
1	1	1

Operationen mit 2 klassischen Bits



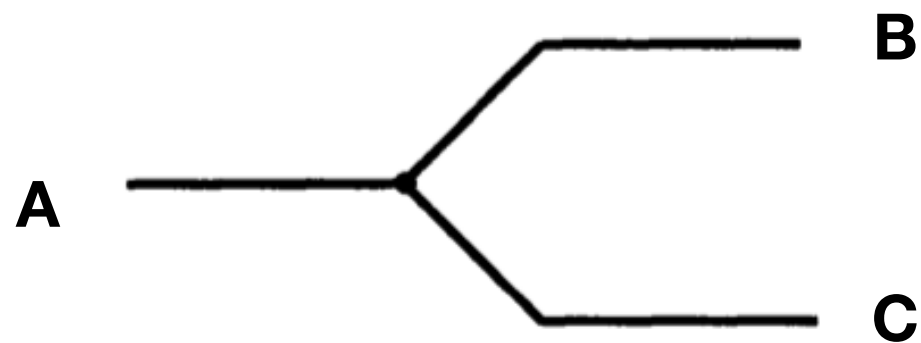
Diese Elemente werden auch
Gatter (engl. **gate**) genannt.

Operationen mit 2 klassischen Bits

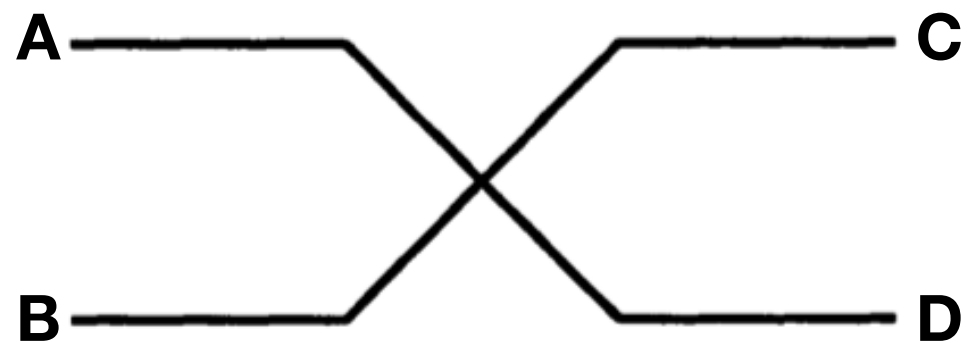
**Beim Bauen von Schaltkreisen mit unseren Gates ergeben sich noch weitere Elemente,
die wir auch als Gates auffassen können**

Operationen mit 2 klassischen Bits

Beim Bauen von Schaltkreisen mit unseren Gates ergeben sich noch weitere Elemente, die wir auch als Gates auffassen können



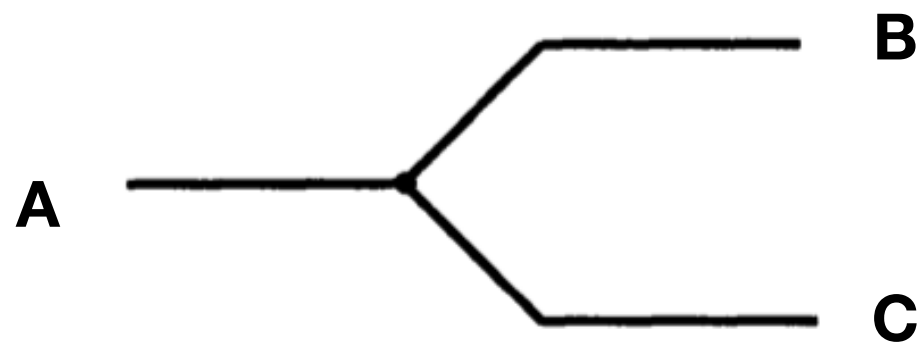
FANOUT



EXCHANGE

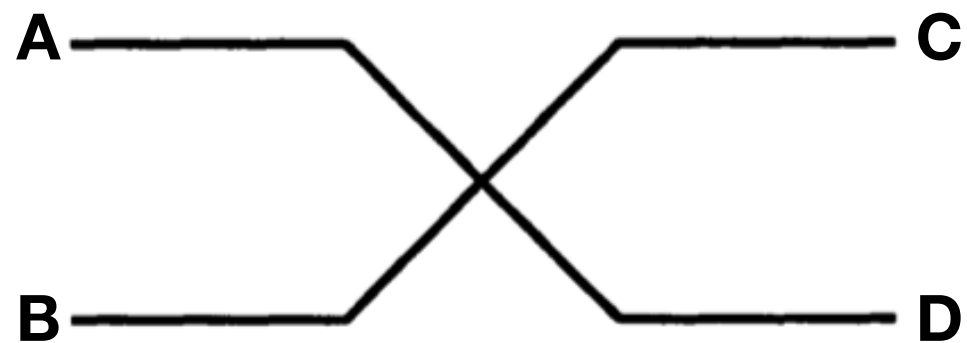
Operationen mit 2 klassischen Bits

Beim Bauen von Schaltkreisen mit unseren Gates ergeben sich noch weitere Elemente, die wir auch als Gates auffassen können



FANOUT

A	B	C
0	0	0
1	1	1

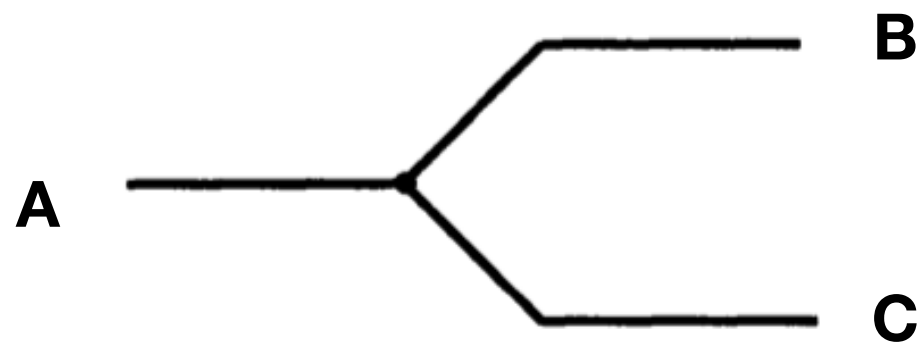


EXCHANGE

A	B	C	D
0	0	0	0
0	1	1	0
1	0	0	1
1	1	1	1

Operationen mit 2 klassischen Bits

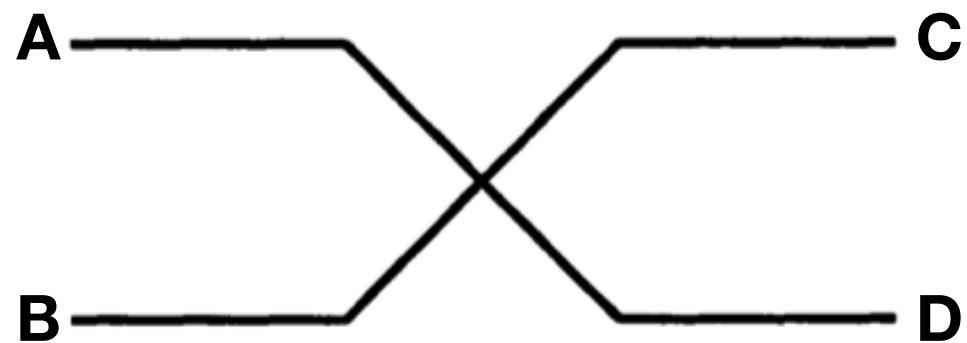
Beim Bauen von Schaltkreisen mit unseren Gates ergeben sich noch weitere Elemente, die wir auch als Gates auffassen können



FANOUT

A	B	C
0	0	0
1	1	1

Verbindung von Drähten



EXCHANGE

A	B	C	D
0	0	0	0
0	1	1	0
1	0	0	1
1	1	1	1

Überkreuzung von Drähten

Operationen mit 2 klassischen Bits

Addierer

Eingang

Ausgang

Bit A

Bit B



Zähler

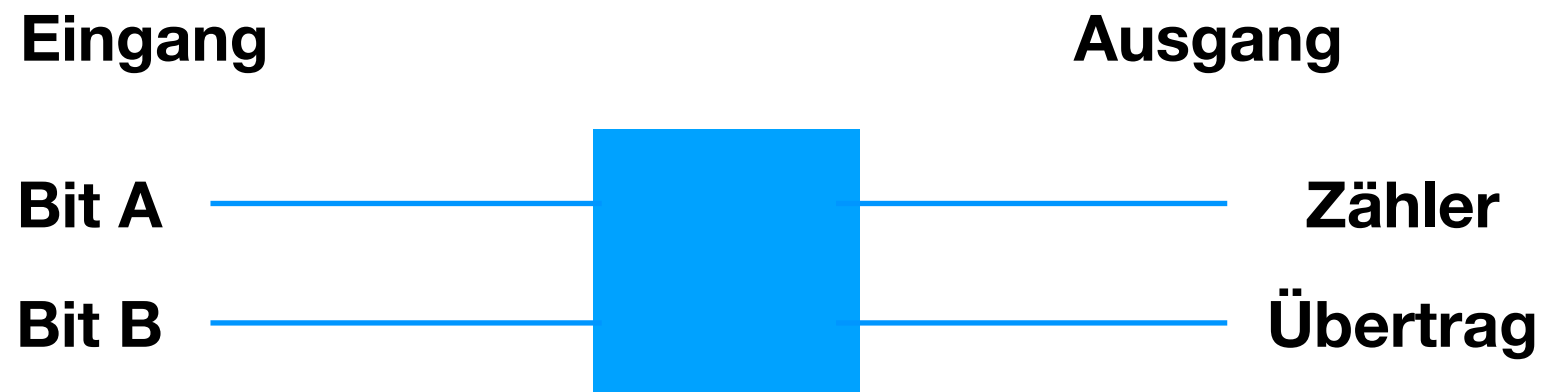
Übertrag

A	B	Z	Ü
0	0		
0	1		
1	0		
1	1		

Wie kann man das mit unseren Gates aufbauen?

Operationen mit 2 klassischen Bits

Addierer



A	B	Z	Ü
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

Wie kann man das mit unseren Gates aufbauen?

Operationen mit 2 klassischen Bits

Addierer

Eingang

Ausgang

Bit A

Bit B



Zähler

Übertrag

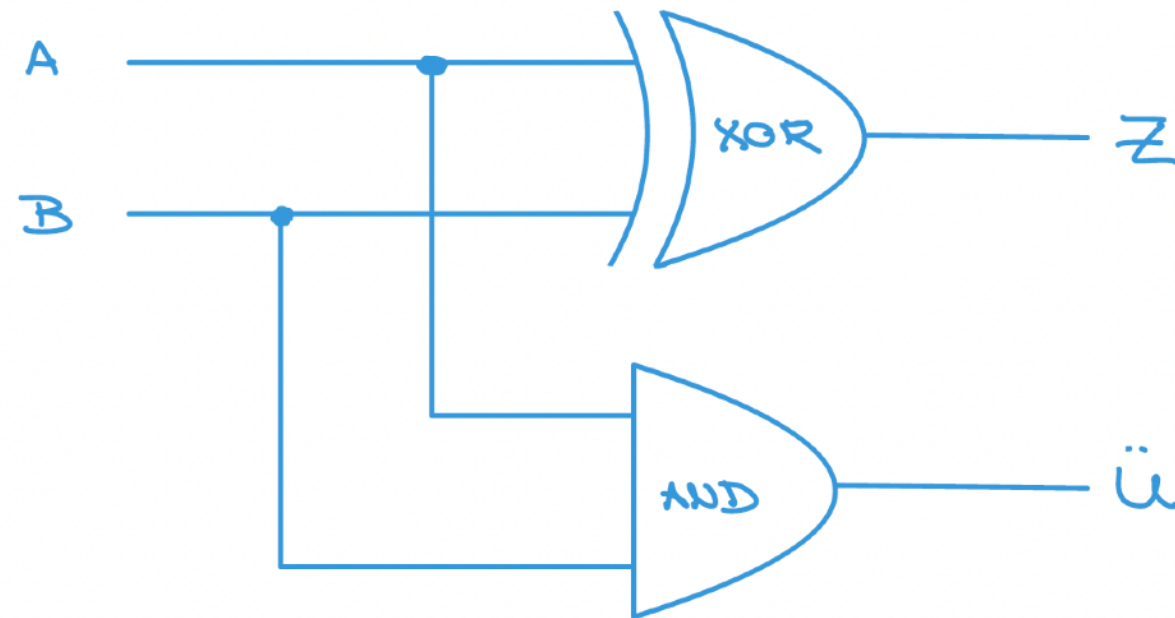
A	B	Z	Ü
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

XOR

AND

Operationen mit 2 klassischen Bits

Addierer



A	B	Z	Ü
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1



XOR

AND

Operationen mit 2 klassischen Bits

Sind alle Gates unabhängig oder gibt es ein universelles Gate?

Operationen mit 2 klassischen Bits

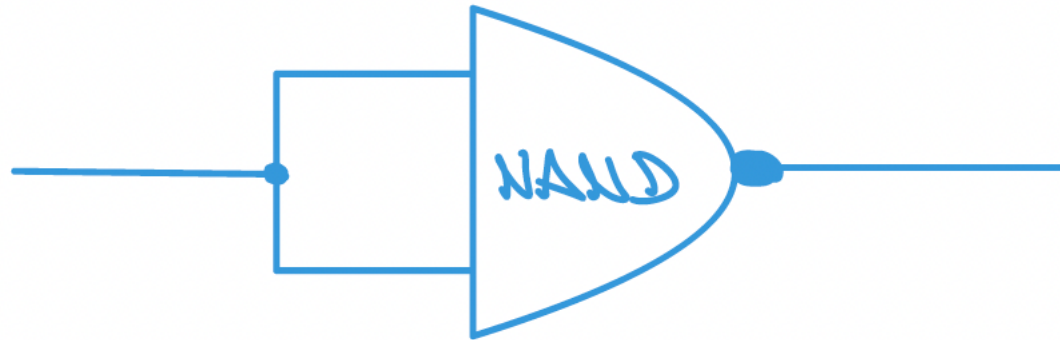
Sind alle Gates unabhängig oder gibt es ein universelles Gate?

Kann man ein NOT gate aus einem NAND gate bauen?

Operationen mit 2 klassischen Bits

Sind alle Gates unabhängig oder gibt es ein universelles Gate?

Kann man ein NOT gate aus einem NAND gate bauen?



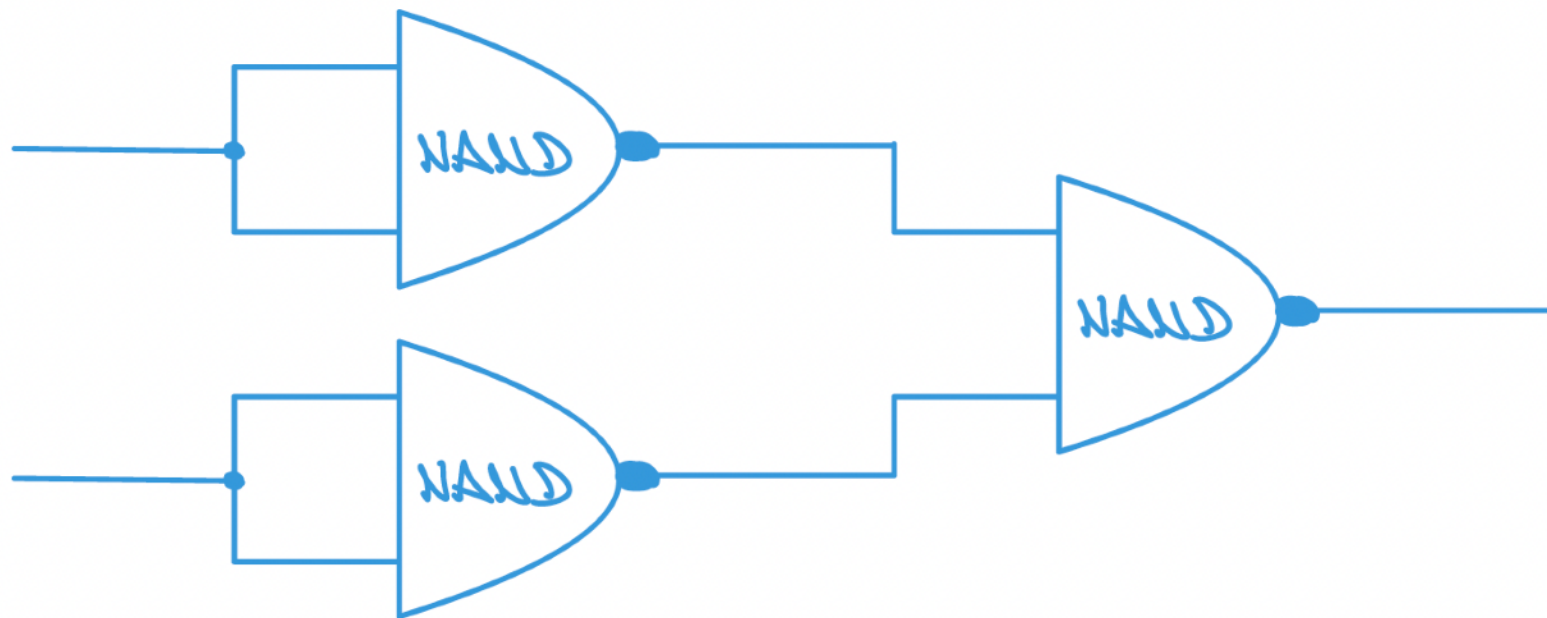
$$\begin{array}{lcl} 0 & \left\{ \begin{array}{l} 0 \\ 0 \end{array} \right. & \text{NAND} \quad 1 \\ & & = \text{NOT} \\ 1 & \left\{ \begin{array}{l} 1 \\ 1 \end{array} \right. & \text{NAND} \quad 0 \end{array}$$

Operationen mit 2 klassischen Bits

Kann man ein OR gate aus NAND gates bauen?

Operationen mit 2 klassischen Bits

Kann man ein OR gate aus NAND gates bauen?



1 0 0 0

1 1 1 0 0 0

0 1 1 1

$\equiv$ OR

Operationen mit 2 klassischen Bits

Man kann zeigen: das NAND gate ist ein universelles Gate!

Operationen mit 2 klassischen Bits

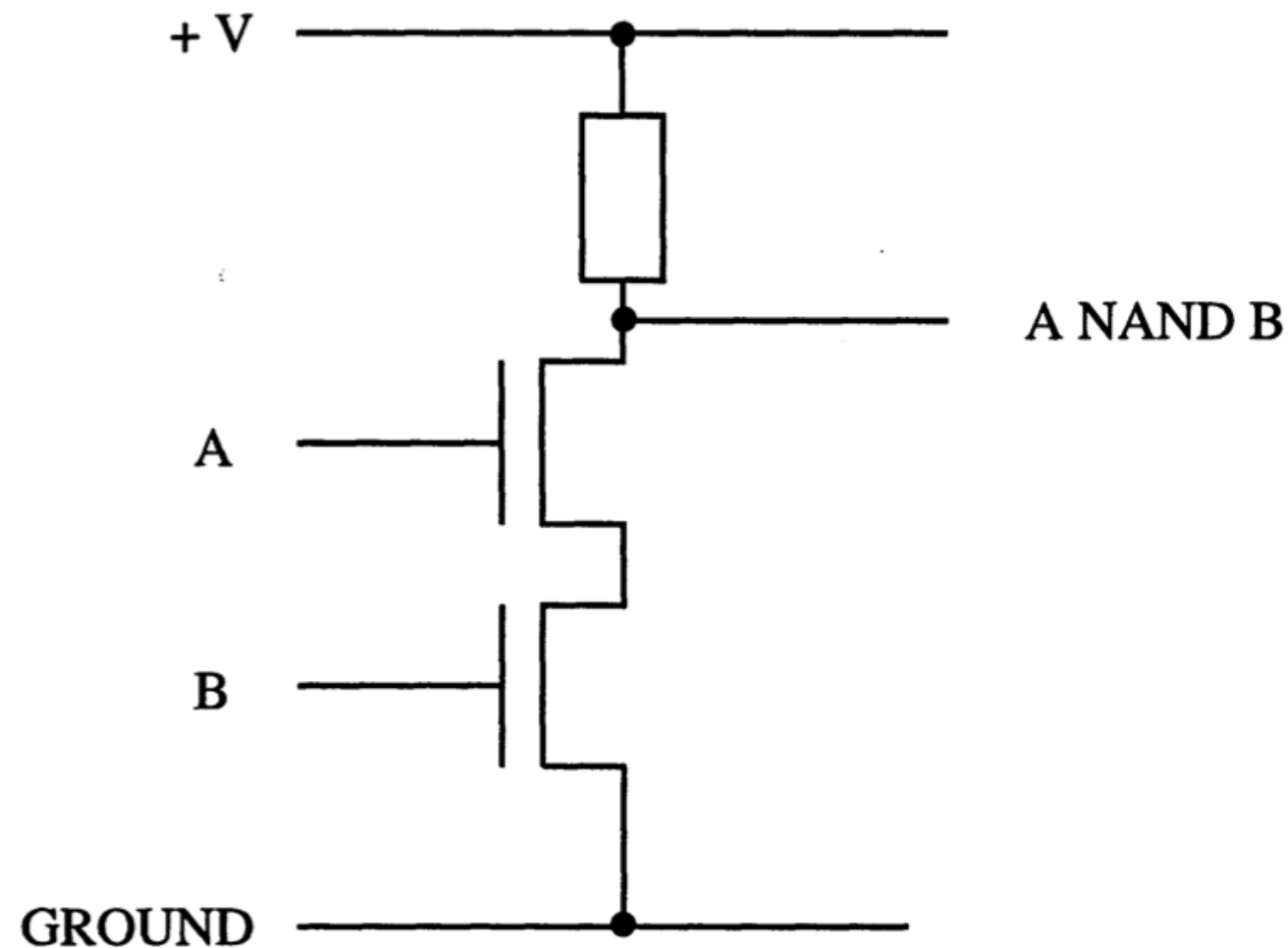
Man kann zeigen: das NAND gate ist ein universelles Gate!

Wie baut man so ein Gate?

Operationen mit 2 klassischen Bits

Man kann zeigen: das NAND gate ist ein universelles Gate!

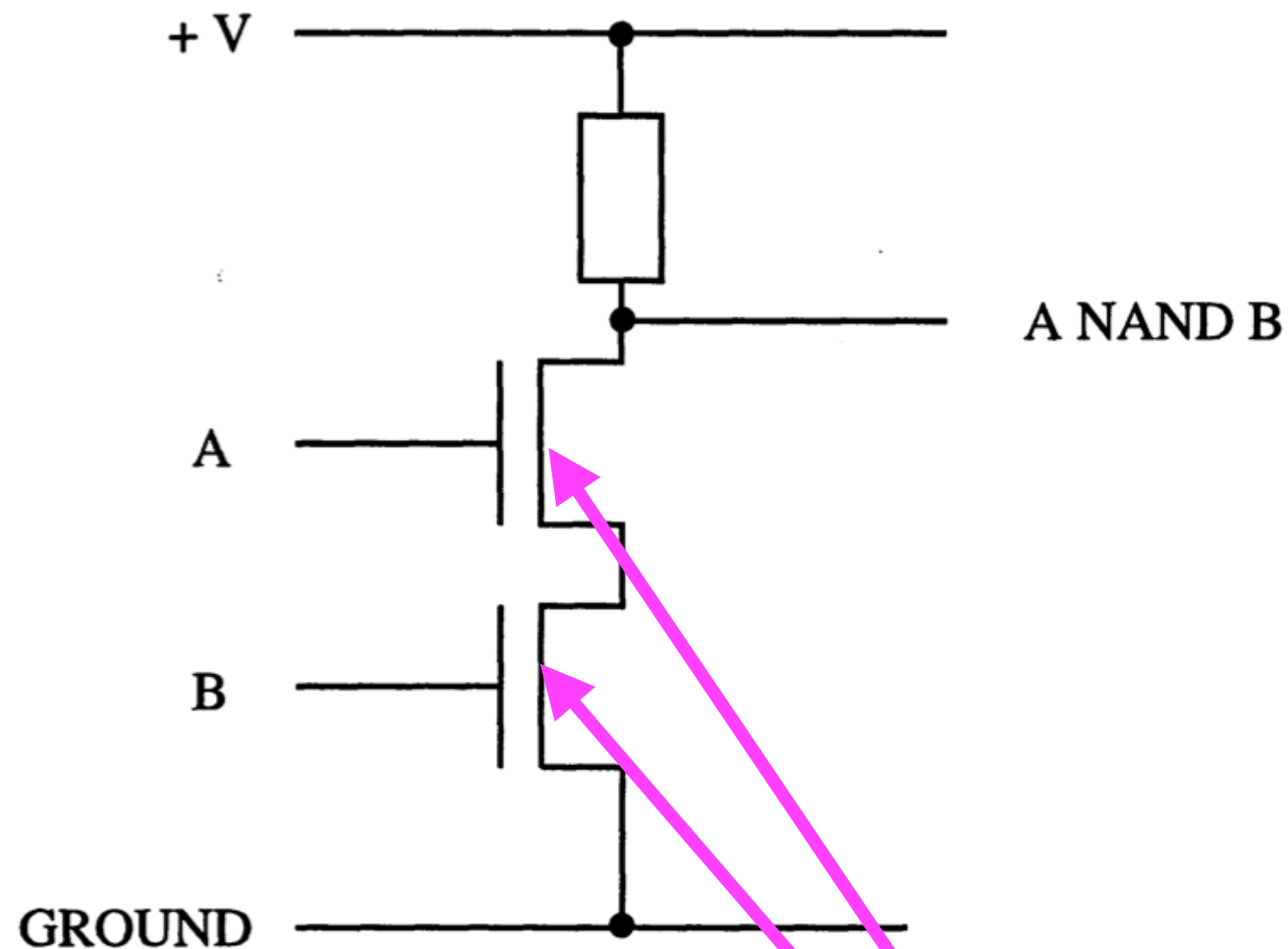
Wie baut man so ein Gate?



Operationen mit 2 klassischen Bits

Man kann zeigen: das NAND gate ist ein universelles Gate!

Wie baut man so ein Gate?



Man braucht 2 Transistoren

Operationen mit 2 klassischen Bits

Damit sind wir mit dem klassischen Computing fertig und wir können im Prinzip die schnellsten Computer bauen...

In der Praxis gibt es da schon einige Probleme



Operationen mit 2 klassischen Bits

Reversibilität: kann man aus dem Ergebnis eines Gates auf die Ausgangsbits zurückschliessen?

Operationen mit 2 klassischen Bits

Reversibilität: kann man aus dem Ergebnis eines Gates auf die Ausgangsbits zurückschliessen?

AND

A	B	C
0	0	0
0	1	0
1	0	0
1	1	1

OR

A	B	C
0	0	0
0	1	1
1	0	1
1	1	1

(E)XOR

A	B	C
0	0	0
0	1	1
1	0	1
1	1	0

NAND

A	B	C
0	0	1
0	1	1
1	0	1
1	1	0

NOR

A	B	C
0	0	1
0	1	0
1	0	0
1	1	0

N(E)XOR

A	B	C
0	0	1
0	1	0
1	0	0
1	1	1

Operationen mit 2 klassischen Bits

Reversibilität: kann man aus dem Ergebnis eines Gates auf die Ausgangsbits zurückschliessen?

AND

A	B	C
0	0	0
0	1	0
1	0	0
1	1	1

OR

A	B	C
0	0	0
0	1	1
1	0	1
1	1	1

(E)XOR

A	B	C
0	0	0
0	1	1
1	0	1
1	1	0

NAND

A	B	C
0	0	1
0	1	1
1	0	1
1	1	0

NOR

A	B	C
0	0	1
0	1	0
1	0	0
1	1	0

N(E)XOR

A	B	C
0	0	1
0	1	0
1	0	0
1	1	1

NEIN!

Einführung

1. Was sind klassische Bits

2. Klassische Operatoren/Gates

3. Reversible Operatoren

Reversible Operationen

Reversibilität: kann man aus dem Ergebnis eines Gates auf die Ausgangsbits zurückschliessen?

AND, OR, XOR, NAND, NOR, NXOR sind irreversible gates

Reversible Operationen

Reversibilität: kann man aus dem Ergebnis eines Gates auf die Ausgangsbits zurückschliessen?

**AND, OR, XOR, NAND, NOR, NXOR sind irreversible gates
Ist der 2 Bit Addierer reversibel?**

Reversible Operationen

Reversibilität: kann man aus dem Ergebnis eines Gates auf die Ausgangsbits zurückschliessen?

**AND, OR, XOR, NAND, NOR, NXOR sind irreversible gates
Ist der 2 Bit Addierer reversibel?**

Gibt es reversible Gates?

Reversible Operationen

Reversibilität: kann man aus dem Ergebnis eines Gates auf die Ausgangsbits zurückschliessen?

**AND, OR, XOR, NAND, NOR, NXOR sind irreversible gates
Ist der 2 Bit Addierer reversibel?**

**Gibt es reversible Gates?
Identität, NOT**

Reversible Operationen

Reversibilität: kann man aus dem Ergebnis eines Gates auf die Ausgangsbits zurückschliessen?

**AND, OR, XOR, NAND, NOR, NXOR sind irreversible gates
Ist der 2 Bit Addierer reversibel?**

Gibt es reversible Gates?

Identität, NOT

Gibt es reversible 2 Bit gates?

Reversible Operationen

Reversibilität: kann man aus dem Ergebnis eines Gates auf die Ausgangsbits zurückschliessen?

**AND, OR, XOR, NAND, NOR, NXOR sind irreversible gates
Ist der 2 Bit Addierer reversibel?**

Gibt es reversible Gates?

Identität, NOT

Gibt es reversible 2 Bit gates?

Ja! Bennett, Fredkin

Reversible Operationen

Reversibilität: kann man aus dem Ergebnis eines Gates auf die Ausgangsbits zurückschliessen?

**AND, OR, XOR, NAND, NOR, NXOR sind irreversible gates
Ist der 2 Bit Addierer reversibel?**

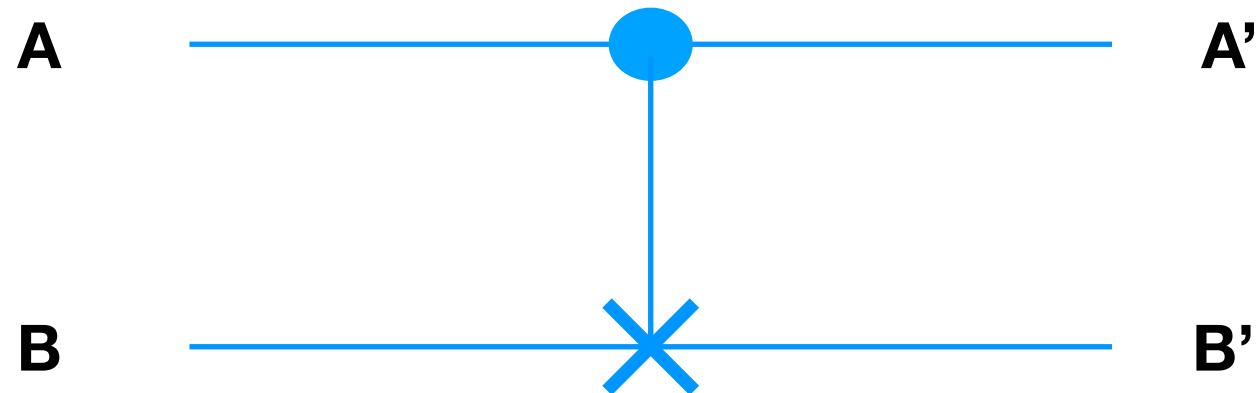
**Gibt es reversible Gates?
Identität, NOT**

**Gibt es reversible 2 Bit gates?
Ja! Bennett, Fredkin**

N	NOT
CN	Controlled Not
CCN	Controlled Controlled Not

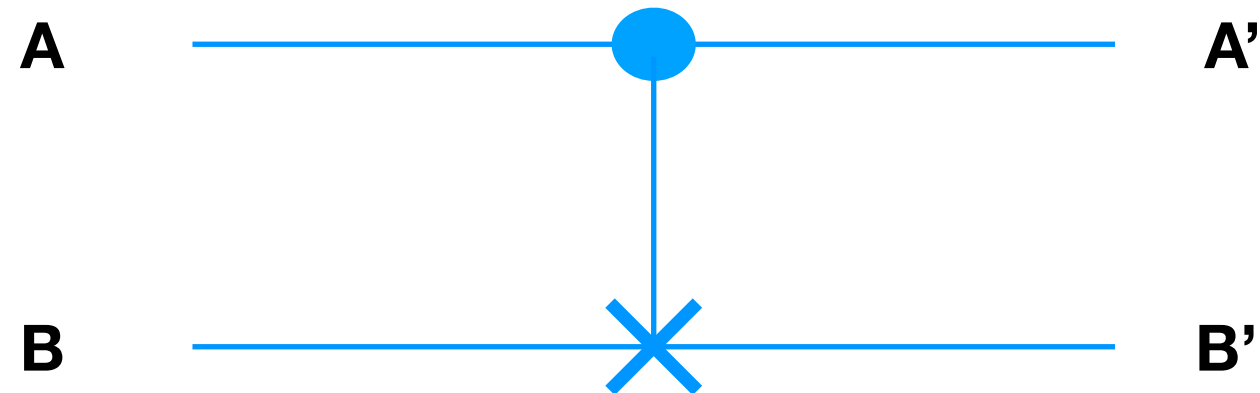
Reversible Operationen

Reversible 2 Bit gates: Controlled NOT



Reversible Operationen

Reversible 2 Bit gates: Controlled NOT

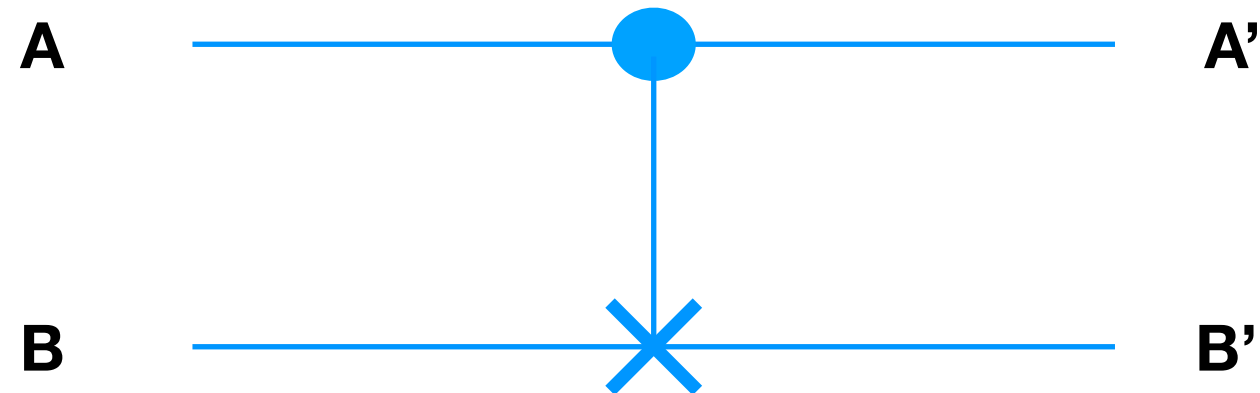


Die A-Linie ist immer die Identität

A	B	A'	B'
0	0	0	
0	1	0	
1	0	1	
1	1	1	

Reversible Operationen

Reversible 2 Bit gates: Controlled NOT



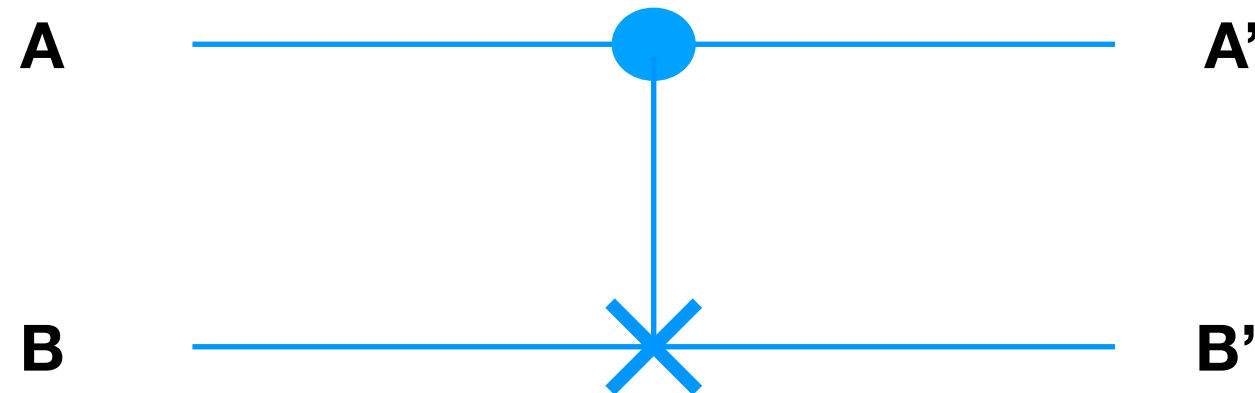
Die A-Linie ist immer die Identität

Wenn $A = 0$, dann ist die untere Linie die Identität

A	B	A'	B'
0	0	0	0
0	1	0	1
1	0	1	
1	1	1	

Reversible Operationen

Reversible 2 Bit gates: Controlled NOT



Die A-Linie ist immer die Identität

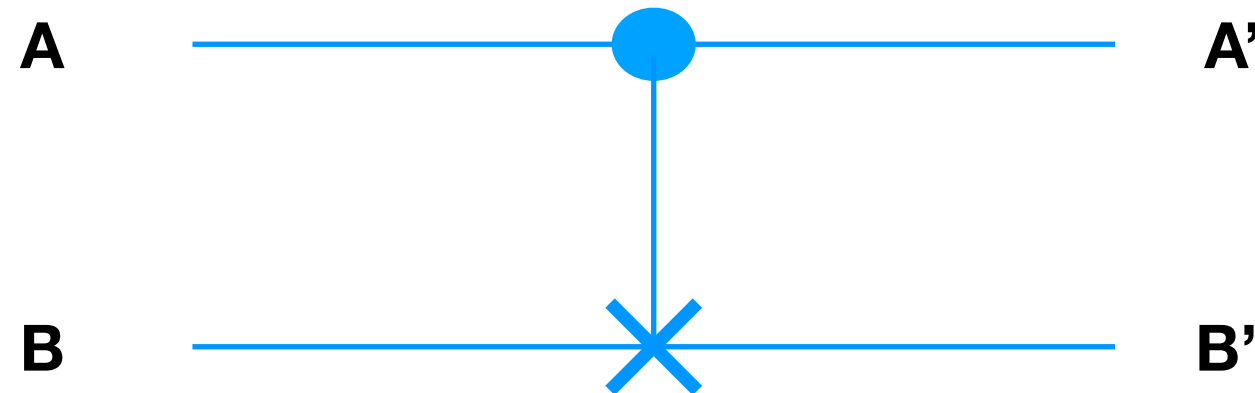
Wenn $A = 0$, dann ist die untere Linie die Identität

Wenn $A = 1$, dann ist die untere Linie die NOT Operation

A	B	A'	B'
0	0	0	0
0	1	0	1
1	0	1	1
1	1	1	0

Reversible Operationen

Reversible 2 Bit gates: Controlled NOT



Die A-Linie ist immer die Identität

Wenn $A = 0$, dann ist die untere Linie die Identität

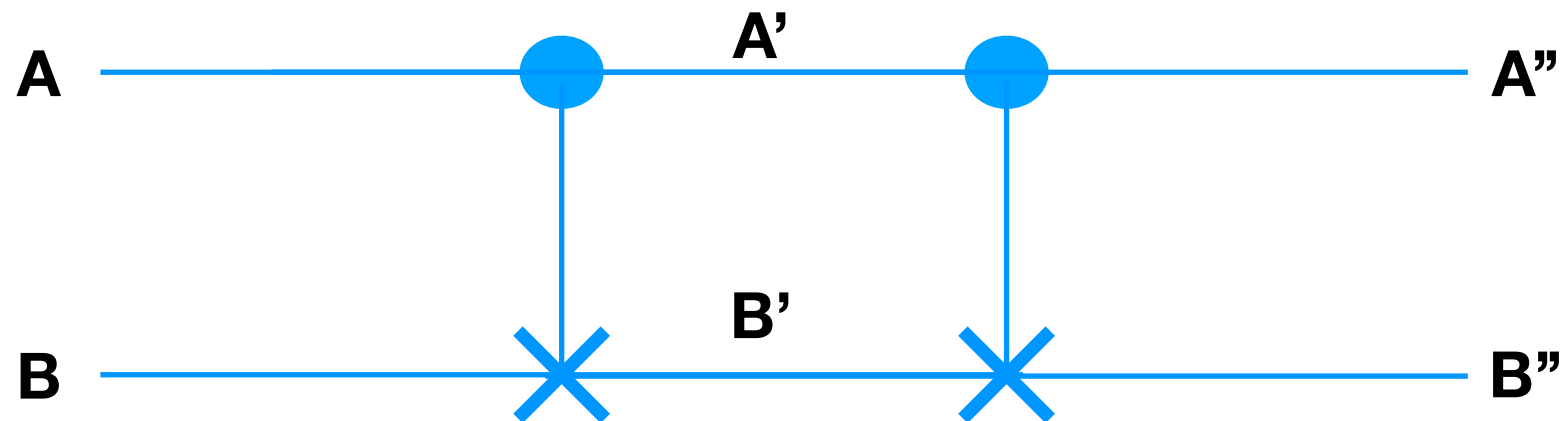
Wenn $A = 1$, dann ist die untere Linie die NOT Operation

A	B	A'	B'
0	0	0	0
0	1	0	1
1	0	1	1
1	1	1	0

Ist das wirklich reversibel?

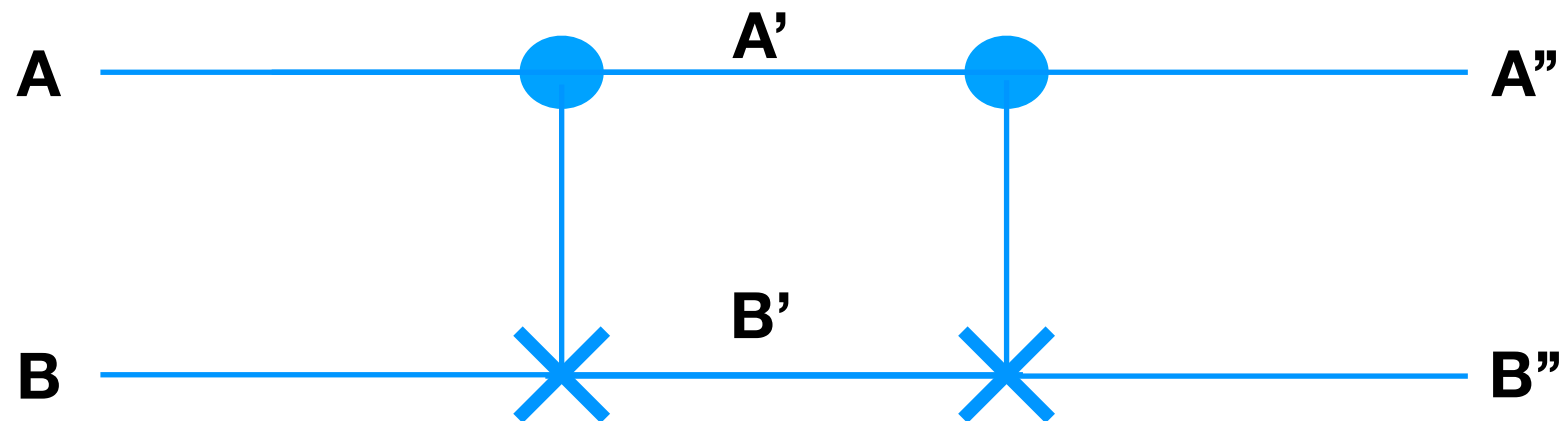
Reversible Operationen

Was macht dieses Gate?



Reversible Operationen

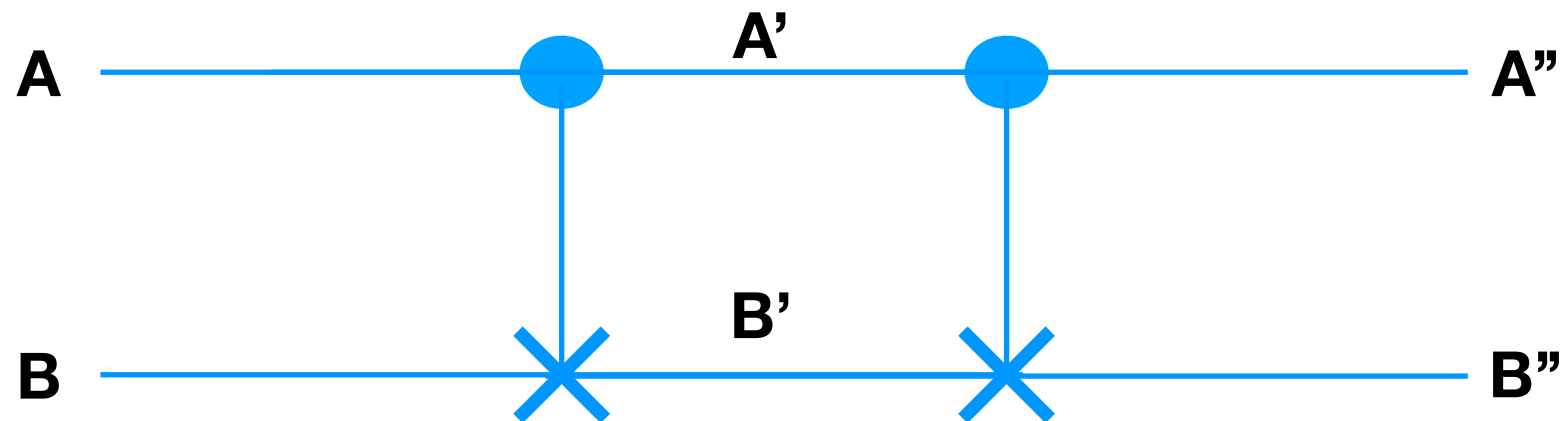
Was macht dieses Gate?



A	B	A'	B'	A''	B''
0	0	0	0		
0	1	0	1		
1	0	1	1		
1	1	1	0		

Reversible Operationen

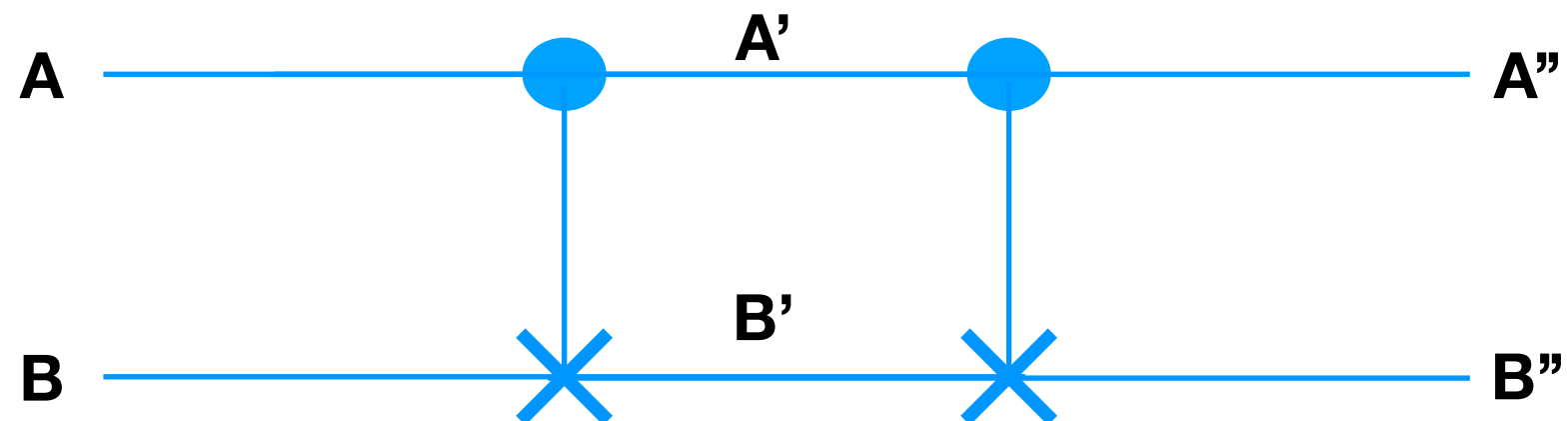
Was macht dieses Gate?



A	B	A'	B'	A''	B''
0	0	0	0	0	
0	1	0	1	0	
1	0	1	1	1	
1	1	1	0	1	

Reversible Operationen

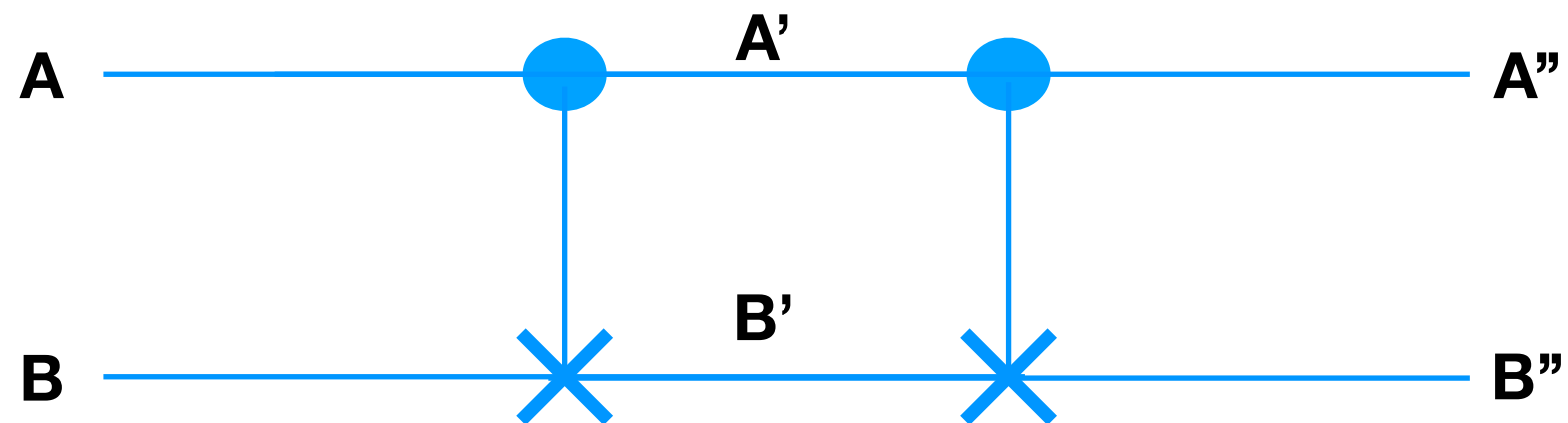
Was macht dieses Gate?



A	B	A'	B'	A''	B''
0	0	0	0	0	0
0	1	0	1	0	1
1	0	1	1	1	
1	1	1	0	1	

Reversible Operationen

Was macht dieses Gate?

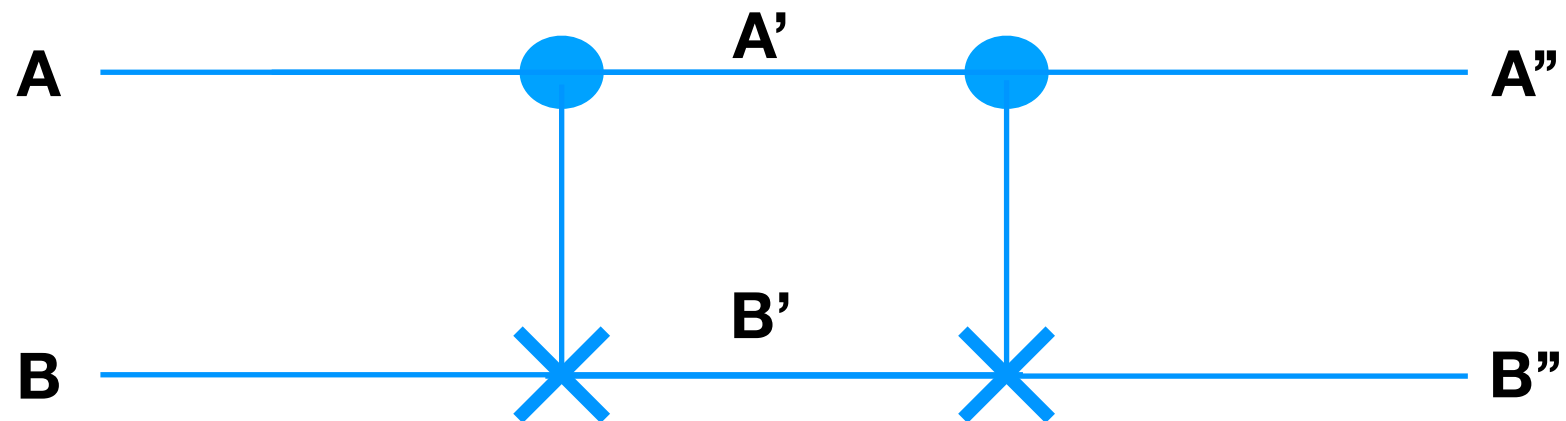


A	B	A'	B'	A''	B''
0	0	0	0	0	0
0	1	0	1	0	1
1	0	1	1	1	0
1	1	1	0	1	1

Dies ist die Identität

Reversible Operationen

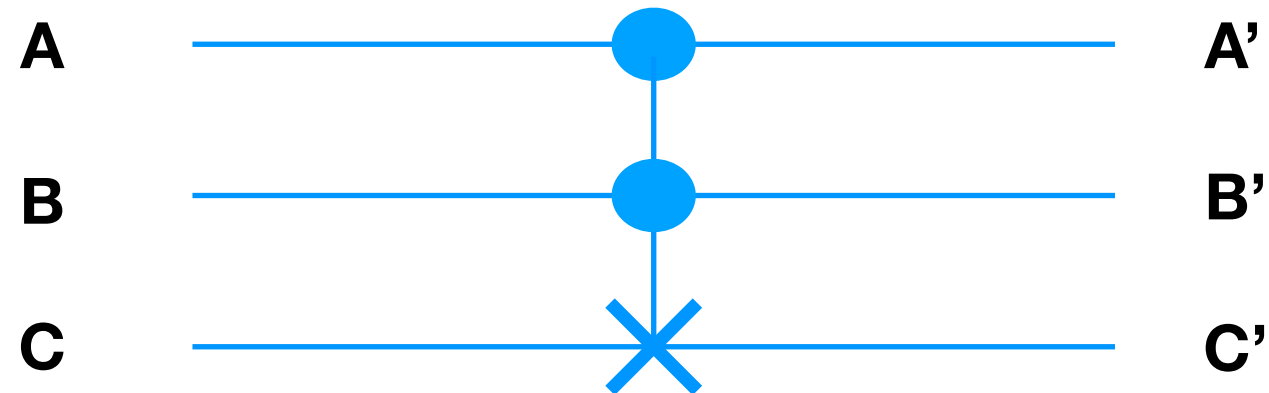
Was macht dieses Gate?



A	B	A'	B'	A''	B''
0	0	0	0	0	0
0	1	0	1	0	1
1	0	1	1	1	0
1	1	1	0	1	1

Reversible Operationen

Reversibles 3 Bit gate: Controlled controlled NOT

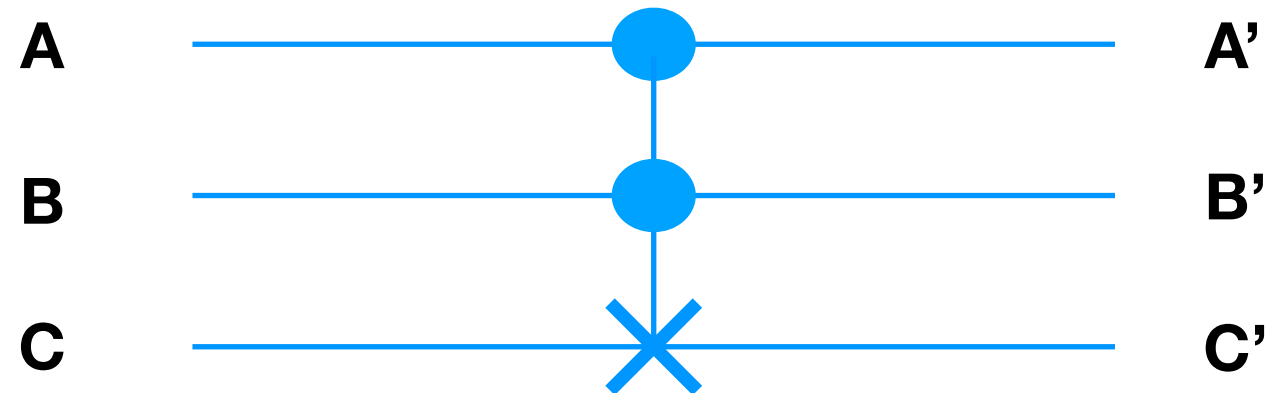


A	B	C	A'	B'	C'
0	0	0			
0	1	0			
1	0	0			
1	1	0			

A	B	C	A'	B'	C'
0	0	1			
0	1	1			
1	0	1			
1	1	1			

Reversible Operationen

Reversibles 3 Bit gate: Controlled controlled NOT



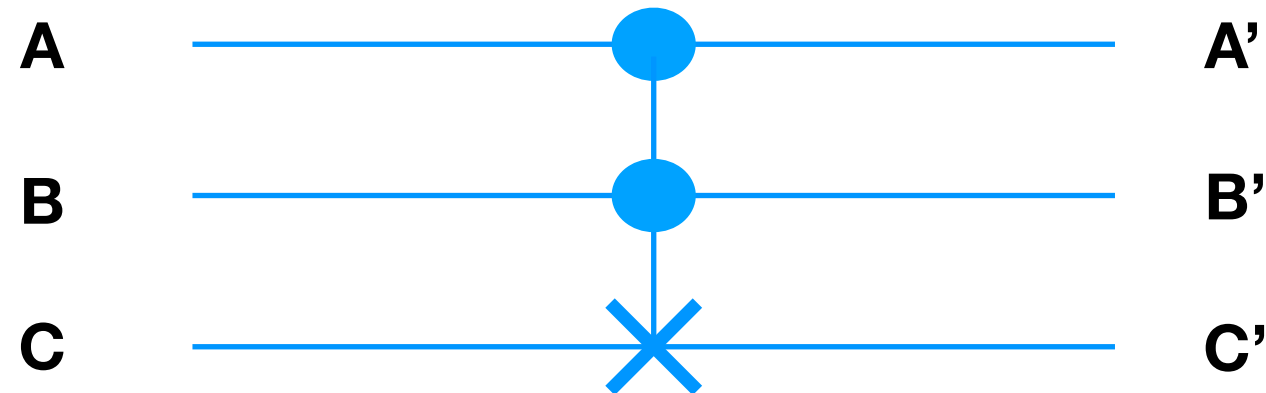
Die A- und B-Linie ist immer die Identität

A	B	C	A'	B'	C'
0	0	0	0	0	
0	1	0	0	1	
1	0	0	1	0	
1	1	0	1	1	

A	B	C	A'	B'	C'
0	0	1	0	0	
0	1	1	0	1	
1	0	1	1	0	
1	1	1	1	1	

Reversible Operationen

Reversibles 3 Bit gate: Controlled controlled NOT



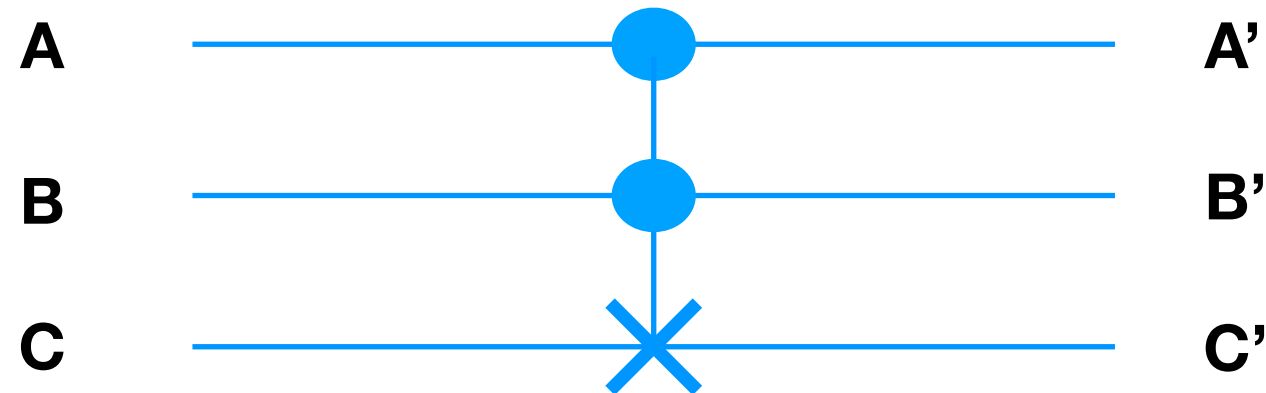
Die A- und B-Linie ist immer die Identität
Wenn $A = 1 = B$, dann ist die untere Linie die NOT Operation,

A	B	C	A'	B'	C'
0	0	0	0	0	
0	1	0	0	1	
1	0	0	1	0	
1	1	0	1	1	1

A	B	C	A'	B'	C'
0	0	1	0	0	
0	1	1	0	1	
1	0	1	1	0	
1	1	1	1	1	0

Reversible Operationen

Reversibles 3 Bit gate: Controlled controlled NOT



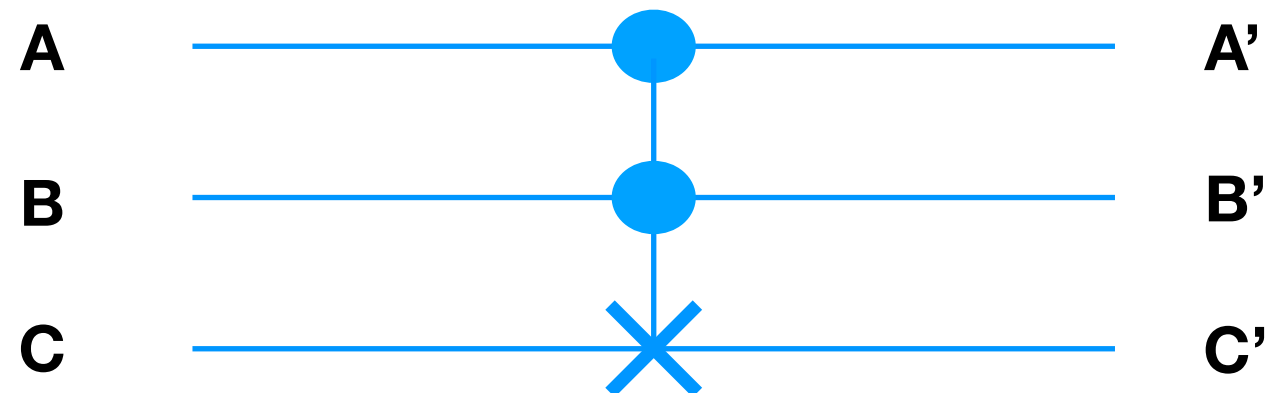
Die A- und B-Linie ist immer die Identität
Wenn $A = 1 = B$, dann ist die untere Linie die NOT Operation,
Sonst ist die untere Linie die Identität

A	B	C	A'	B'	C'
0	0	0	0	0	0
0	1	0	0	1	0
1	0	0	1	0	0
1	1	0	1	1	1

A	B	C	A'	B'	C'
0	0	1	0	0	1
0	1	1	0	1	1
1	0	1	1	0	1
1	1	1	1	1	0

Reversible Operationen

Reversibles 3 Bit gate: Controlled controlled NOT
Toffoli gate



Die A- und B-Linie ist immer die Identität
Wenn $A = 1 = B$, dann ist die untere Linie die NOT Operation,
Sonst ist die untere Linie die Identität

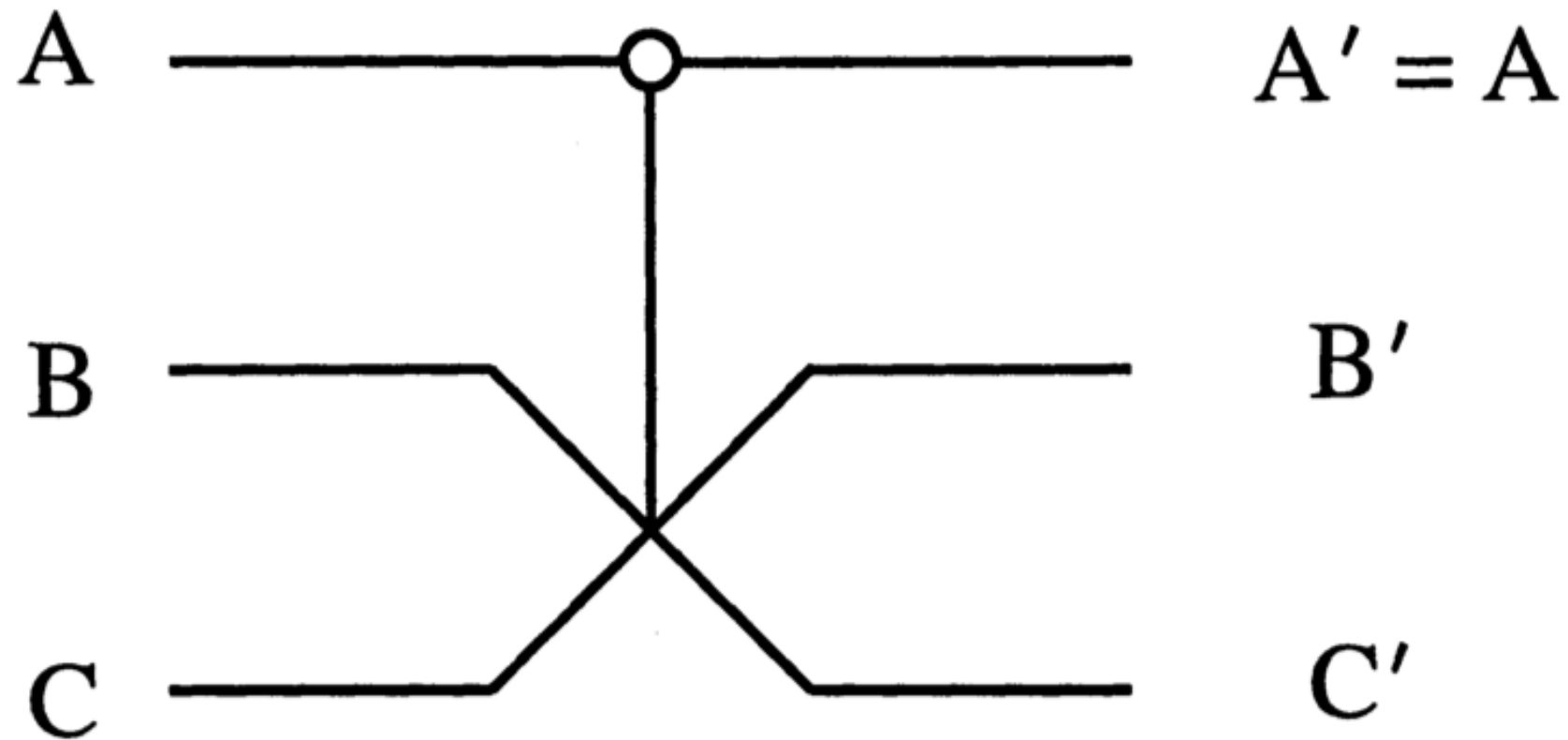
A	B	C	A'	B'	C'
0	0	0	0	0	0
0	1	0	0	1	0
1	0	0	1	0	0
1	1	0	1	1	1

A	B	C	A'	B'	C'
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 das wirklich reversibel?

Reversible Operationen

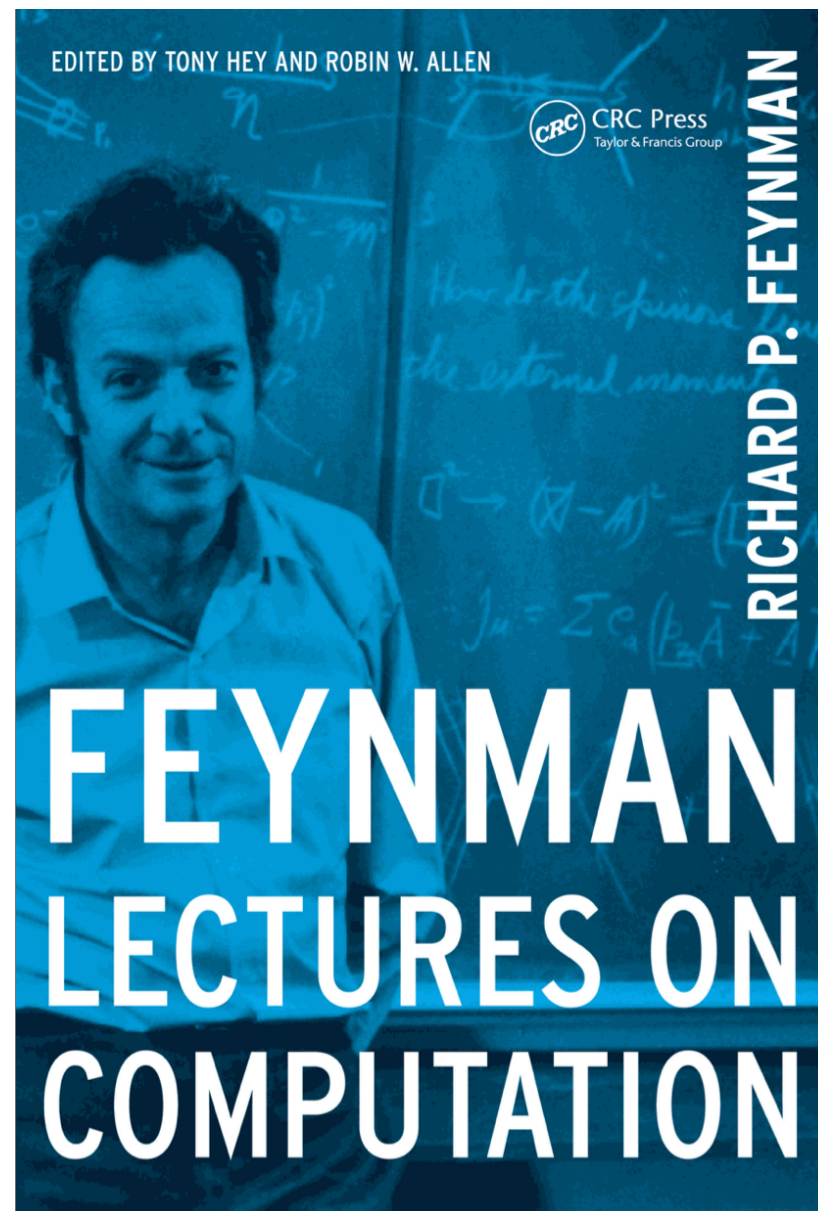
Reversible 3 Bit gate: Controlled exchange
Fredkin gate



Reversible Operationen

Mit N, CN und CNN kann man alle Gates aufbauen

Für mehr Informationen in dieser Richtung



Jetzt kennen wir die Grundelemente eines klassischen Computer (Irreversible Gates) aber auch eines Quantencomputers (reversible Gates)

Quantencomputing

Für die Hintergründe/Grundlagen von
Quantencomputing
braucht man eigentlich
Quantenmechanik

Wer sich dafür interessiert, kann sich die
Folien vom letzten Semester unter
[https://tp1.physik.uni-siegen.de/
mittwochsakademie/](https://tp1.physik.uni-siegen.de/mittwochsakademie/)
ansehen

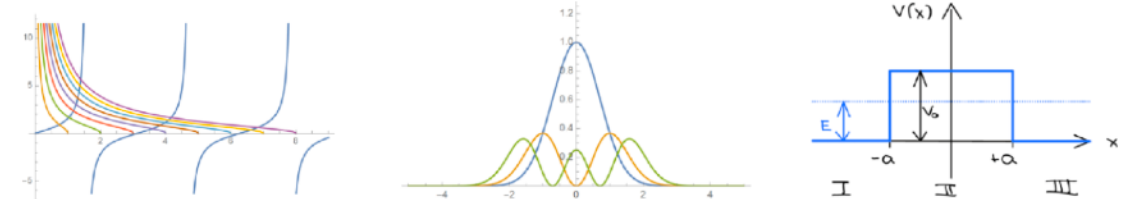
Wir gehen hier einen etwas anderen Weg



Schrödinger's Katze am Emmy Noether Campus: sie lebt!



Ankündigung für das Sommersemester 2025
Das theoretische Minimum II
Quantenmechanik - Prof. Dr. Alexander Lenz, 4PHY00011V



Diese Vorlesungsreihe gibt eine Einführung in die Grundprinzipien der theoretischen Physik.

2025 wird weltweit das 100 jährige Jubiläum der Entdeckung der Quantenmechanik gefeiert. Ursprünglich war dies über viele Jahrzehnte lang reinste Grundlagenforschung ohne jegliche Hinweise auf potentielle Anwendungen. 100 Jahre später finden wir, dass ein Großteil der technologischen Errungenschaften der Menschheit im letzten Jahrhundert auf der Quantenmechanik basiert - zuletzt gipfelte dies in den ersten Quantencomputern.

Im Sommersemester 2025 beschäftigen wir uns daher mit einer Einführung in die Grundprinzipien der Quantenmechanik:

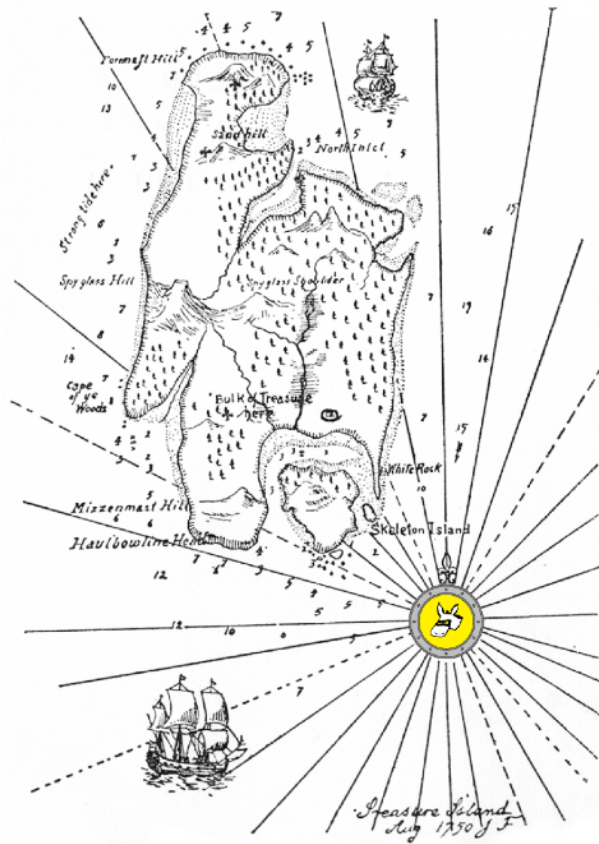
$$i\hbar \frac{\partial}{\partial t} \Psi(\vec{x}, t) = \left[\frac{\hbar^2}{2m} \Delta + V(\vec{x}) \right] \Psi(\vec{x}, t)$$

Die Vorlesung richtet sich an Mittwochsakademiker, Oberstufenschülerinnen und -schüler, Lehrkräfte und Physikenthusiasten mit einem großen Interesse an aktuellen Themen der Physik. Es werden mathematische Konzepte (auf dem Niveau der gymnasialen Oberstufe) eingeführt und benutzt. Die Vorlesung ist an die erfolgreiche Vorlesungs- und Buchreihe "The theoretical Minimum" von Leonard Susskind angelehnt, welche auf dieselbe Zielgruppe ausgerichtet war. Vom Niveau her wird sich die Veranstaltung auf dem schmalen Grat zwischen einer rein populärwissenschaftlichen Bildershow und einer theoretischen Physikvorlesung im Bachelorstudium bewegen.



11 Termine im Sommersemester 25:
7.5., 14.5., 21.5., 28.5., 4.6., 11.6., 18.6., 25.6., 2.7., 9.7., 16.7.
Mittwochs 16-18
Emmy Noether Campus ENC-D-114
Infos unter: alexander.lenz@uni-siegen.de
<https://tp1.physik.uni-siegen.de/mittwochsakademie/>





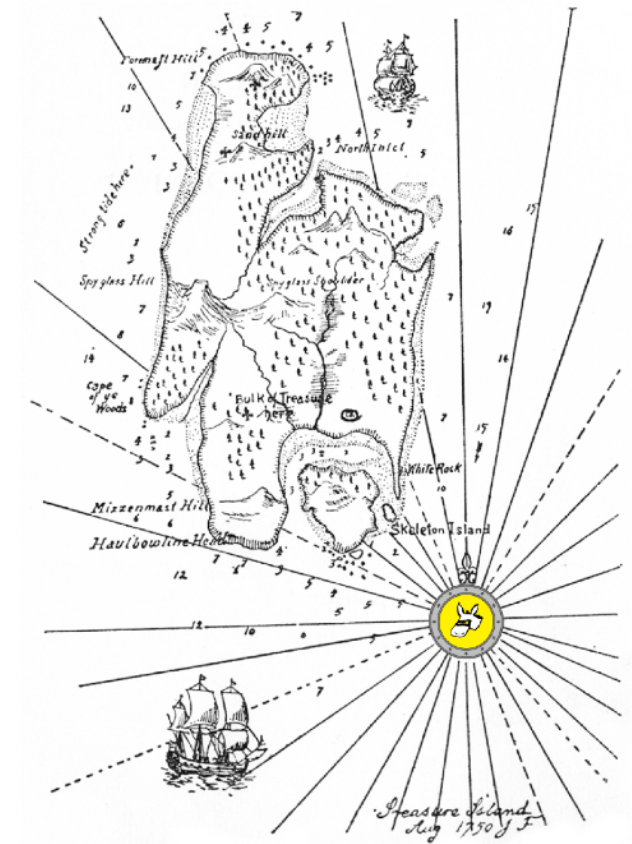
Ab nächste Woche: Quantencomputing

Mittwochsakademie/ Begabte Siegen

angelehnt an

“The Quantum Quest“ von Maris Ozols & Michael Walter

<https://qi-rub.github.io/quantum-quest/2023/de/>



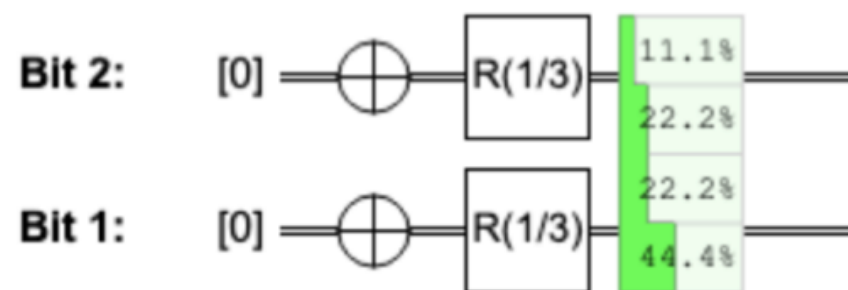
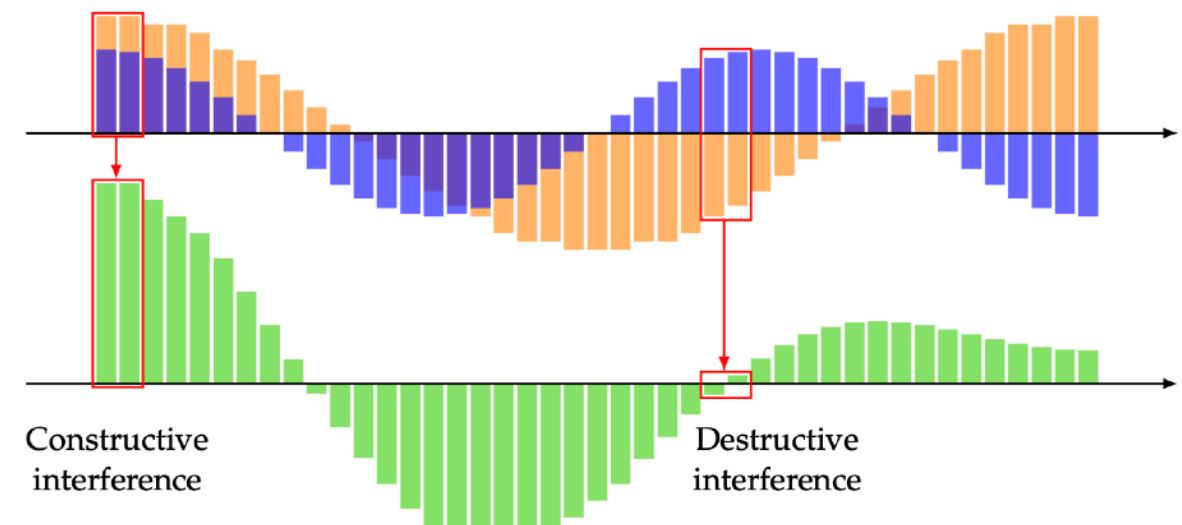
Ziel dieser Vorlesung ist es
nicht-triviale Einblicke in die Grundprinzipien des
Quantencomputings zu geben,
sowie erste QC-Programme zu erstellen

Vorlesung: Das theoretische Minimum

Mittwochsakademie

Ablauf

- 19.11.: Einführung
- 26.11.: Q1 Maestro der Wahrscheinlichkeit
 - 3.12.: Q2a Das Qubit bezwingen 1- Vertretung
- 10.12.: Q2b Das Qubit bezwingen 2
- 17.12.: Q3a Verzaubernde Verschränkungen 1
 - 7. 1.: Q3b Verzaubernde Verschränkungen 2
- 14. 1.: Q4a Quantenkompositionen 1
- 21. 1.: Q4b Quantenkompositionen 2
- 28. 1.: Q5a Virtuose Algorithmen 1
 - 4. 2.: Q5b Virtuose Algorithmen 2



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