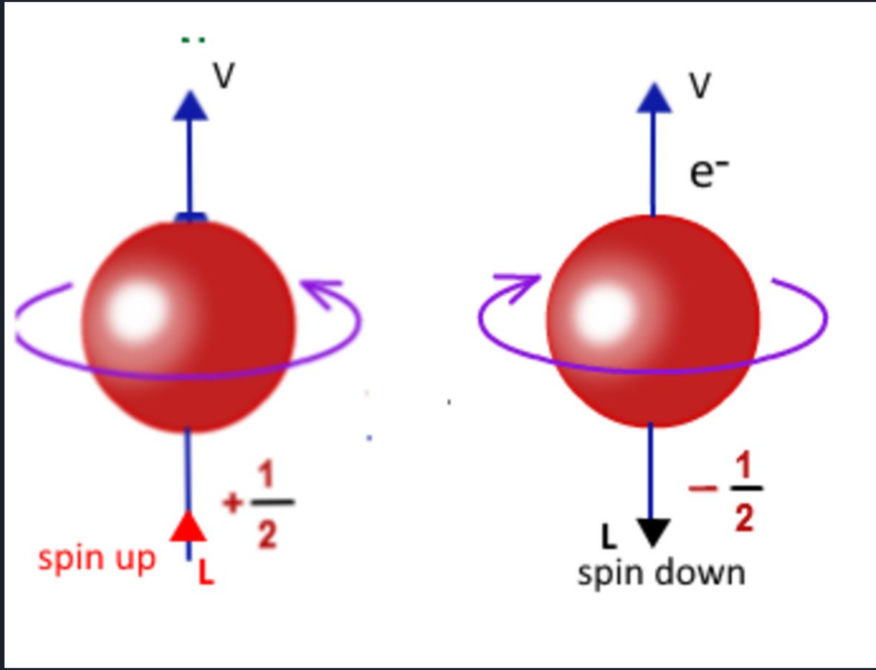
Introduction to Quantum Computing

Austin Shouli | CSCI 490

What is Quantum Computing?

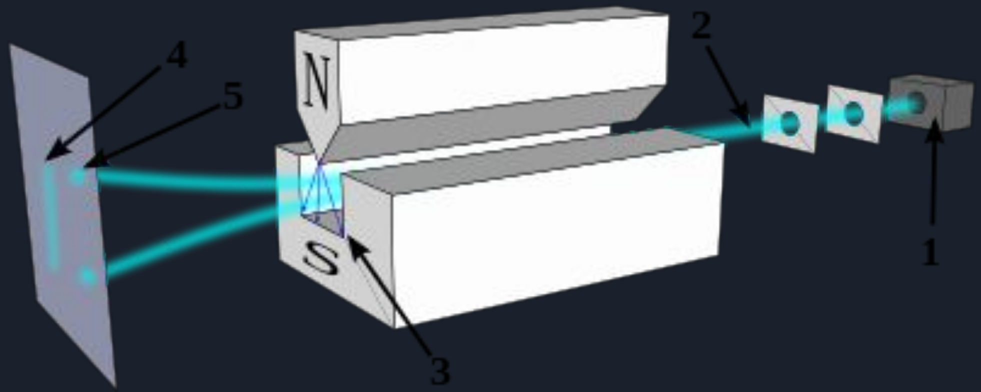
- Functioning quantum computers exist in the world today
- Can provide increased algorithmic efficiency over classical computers
- Make use of quantum phenomena to perform computation
- Built from qubits instead of bits

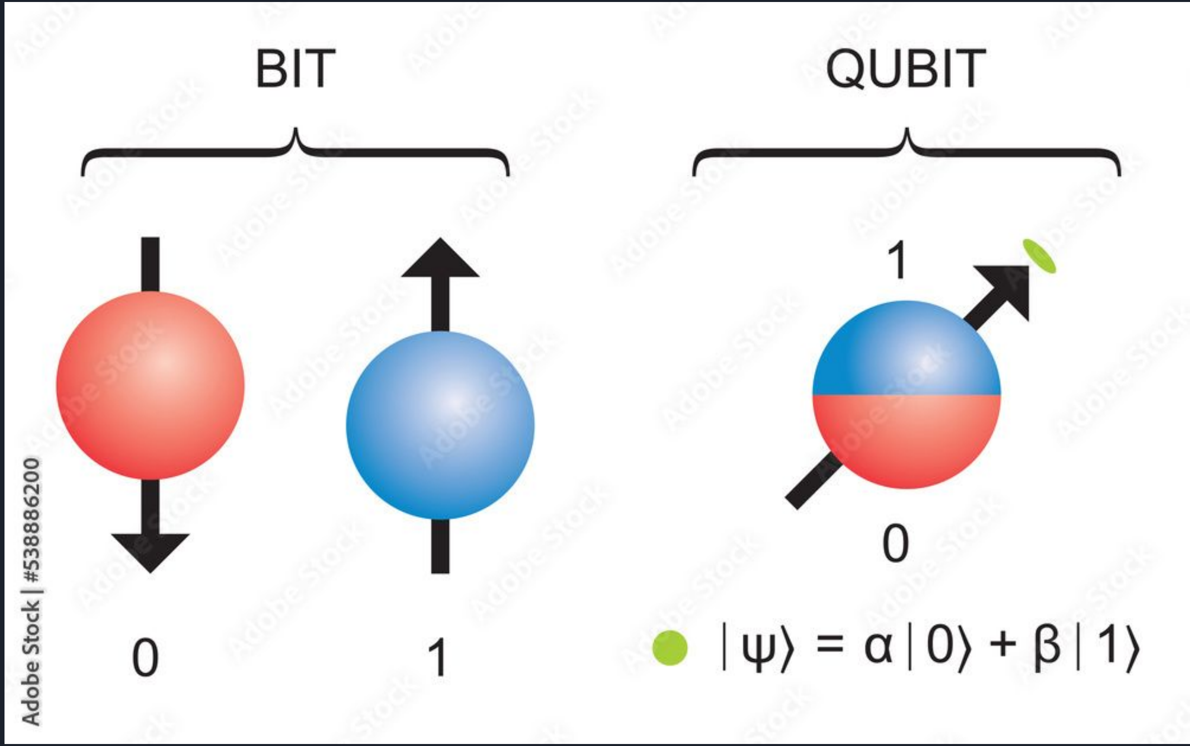
Natural Qubits



https://discover.hubpages.com/education/Why-Do-Quantum-Particles-Have-Spin

Stern-Gerlach Experiment




BIT
QUBIT
1
0
0
1
|ψ⟩ = α|0⟩ + β|1⟩
Adobe Stock | #538886200

Qubits

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

Classical State Vector Example

$$\begin{bmatrix} \frac{1}{2} \\ \frac{1}{2} \end{bmatrix} = \frac{1}{2}|0\rangle + \frac{1}{2}|1\rangle$$

$$= \frac{1}{2} + \frac{1}{2}$$

$$= 1$$

Superposition

$$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$$

Superposition

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

Superposition

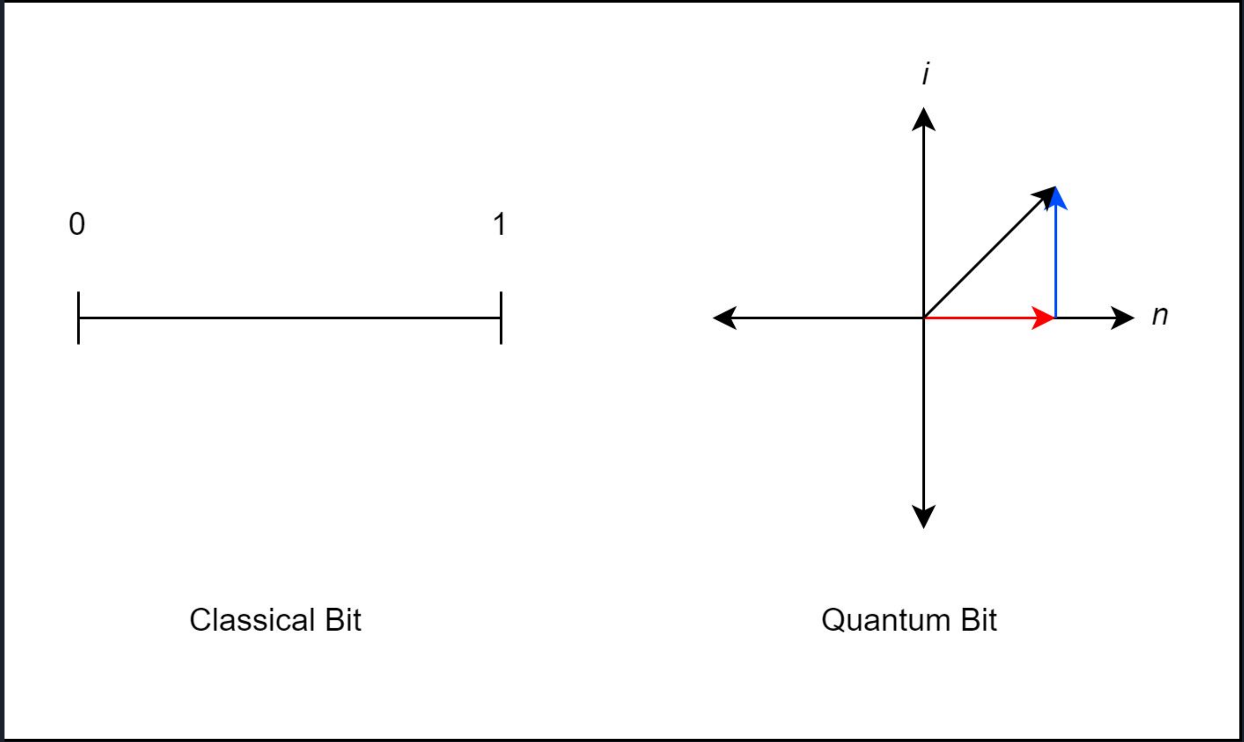
$$|+\rangle = \begin{bmatrix} \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} \end{bmatrix} \equiv \left|\left(\frac{1}{\sqrt{2}}\right)\right|^2 |0\rangle + \left|\left(\frac{1}{\sqrt{2}}\right)\right|^2 |1\rangle$$

$$= \frac{1}{2}|0\rangle + \frac{1}{2}|1\rangle$$

$$\equiv \frac{2}{2}$$

$$= 1$$

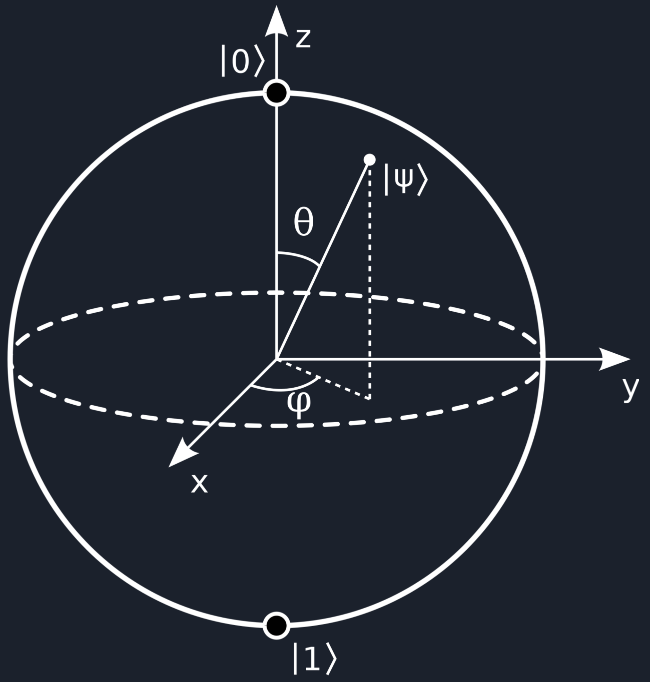
State Space



The Bloch Sphere

$$\alpha = \cos\left(\frac{\theta}{2}\right),\ \beta = e^{i\phi}\sin\left(\frac{\theta}{2}\right), \text{ such that}$$

$$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle = \cos\left(\frac{\theta}{2}\right)|0\rangle = e^{i\phi}\sin\left(\frac{\theta}{2}\right)|1\rangle$$



Multi-Qubit Systems

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

Multi-Qubit Systems

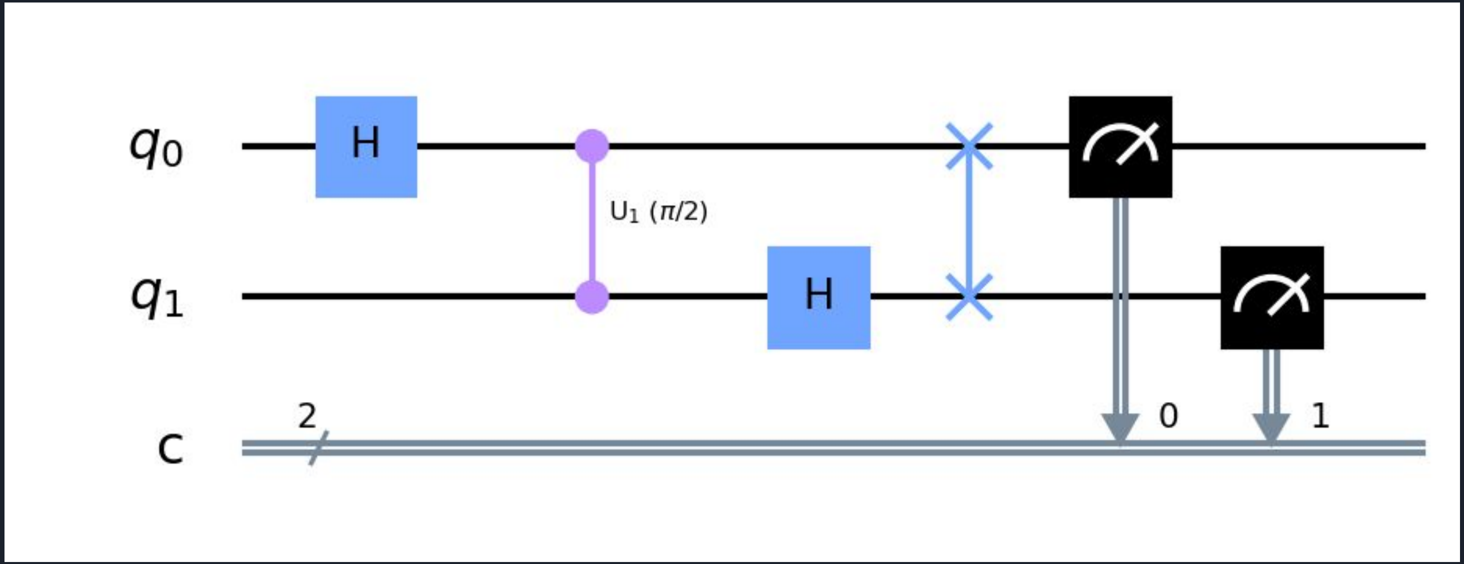
$$|010\rangle = \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}, \; states = \begin{matrix} 000 \\ 001 \\ 010 \\ 011 \\ 100 \\ 101 \\ 110 \\ 111 \end{matrix}$$

Quantum Operators

Quantum Operators

- To perform quantum computing, we need to perform operations on bits
- These operations are called ‘operators’ or ‘gates’
- Quantum computers have a different set of operations compared to classical computers
- Mathematically, quantum operators are represented as matrices
- A series of operators can be represented by a circuit

Quantum Circuit

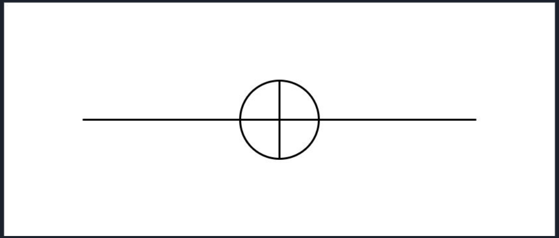


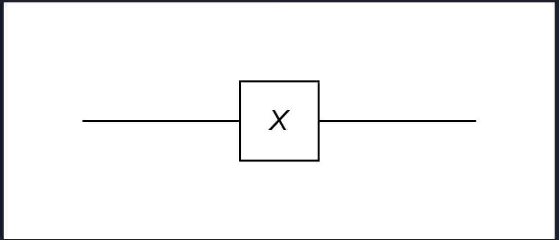
Pauli X Gate

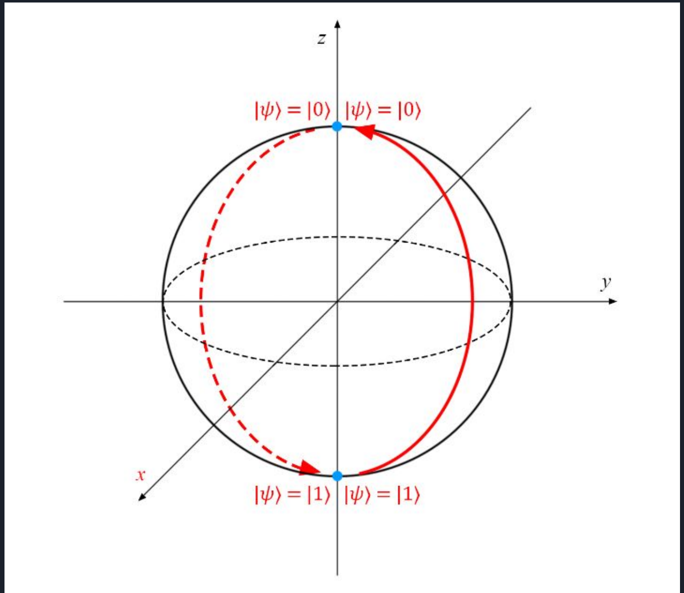
$$X = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$$

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

Pauli X Gate

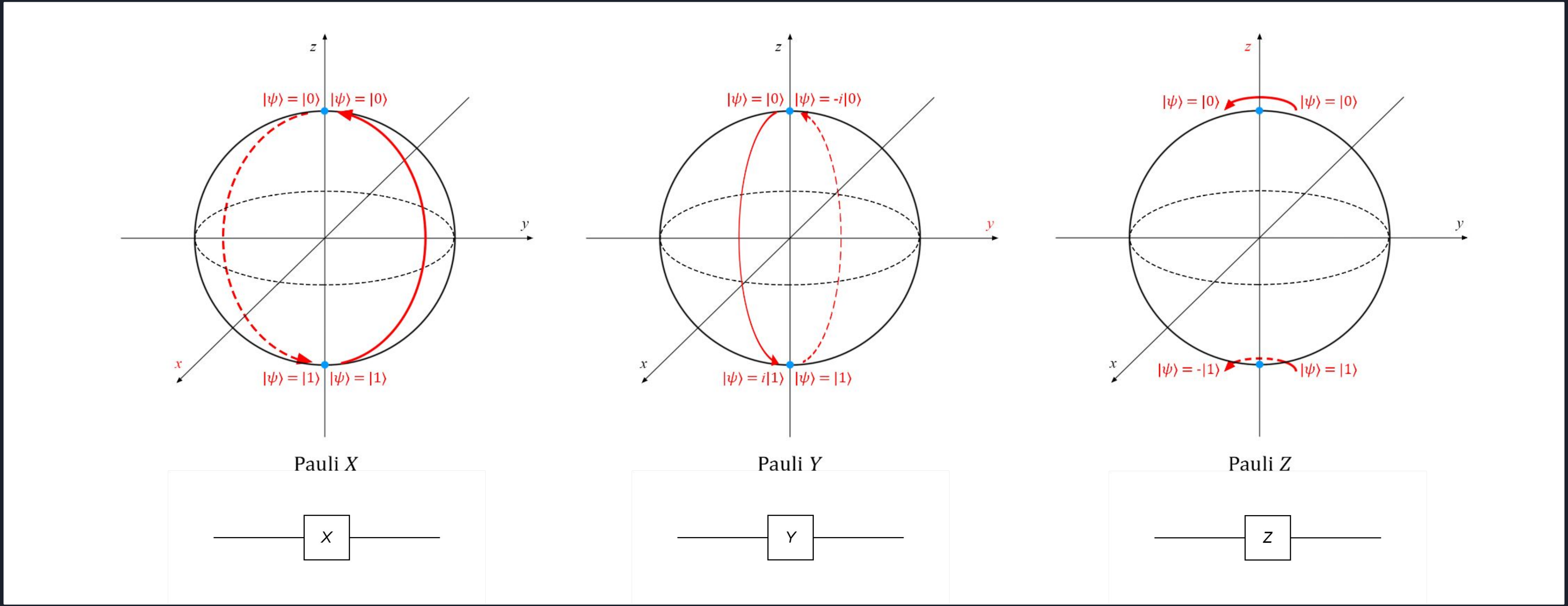






Pauli *X*

Pauli Gate Set

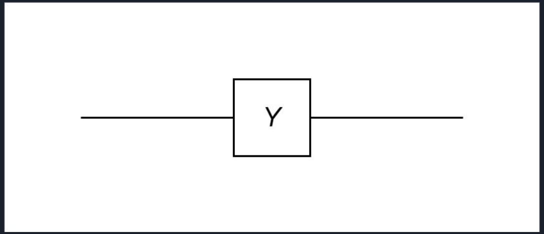


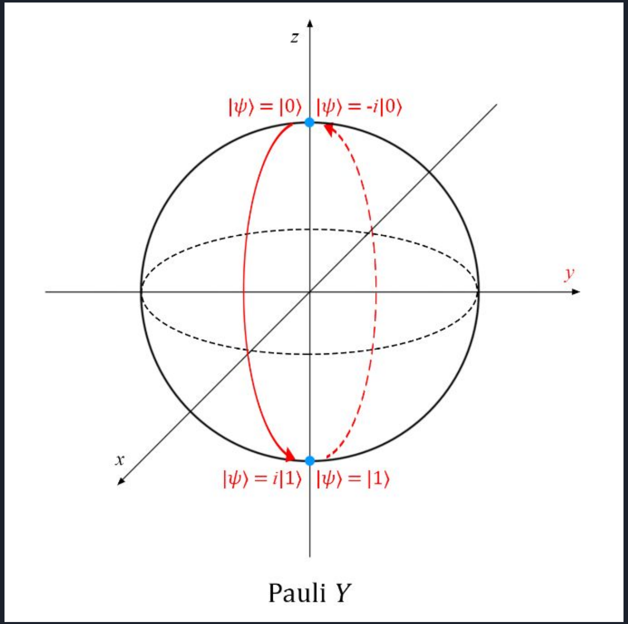
Pauli Y Gate

$$Y = \begin{bmatrix} 0 & -i \\ i & 0 \end{bmatrix}$$

$$Y|0\rangle = \begin{bmatrix} 0 & -i \\ i & 0 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ i \end{bmatrix} = i|1\rangle$$

Pauli Y Gate


Y

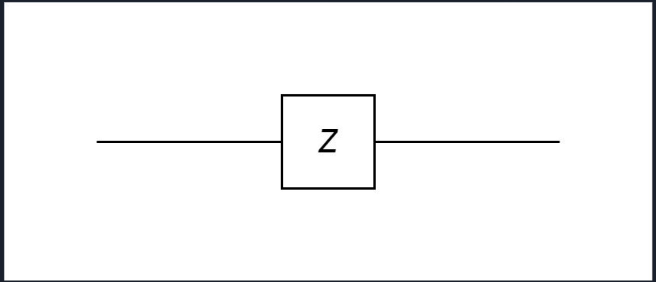

z
$|\psi\rangle = |0\rangle$
$|\psi\rangle = -i|0\rangle$
y
x
$|\psi\rangle = i|1\rangle$
$|\psi\rangle = |1\rangle$
Pauli Y

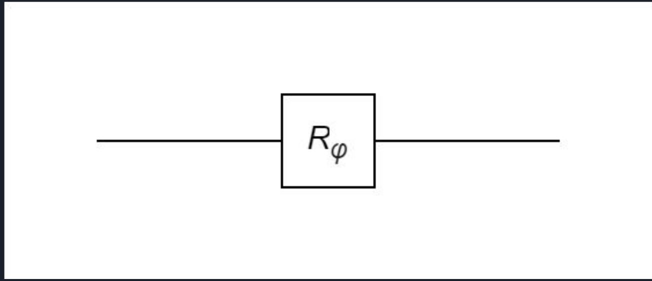
Pauli Z Gate

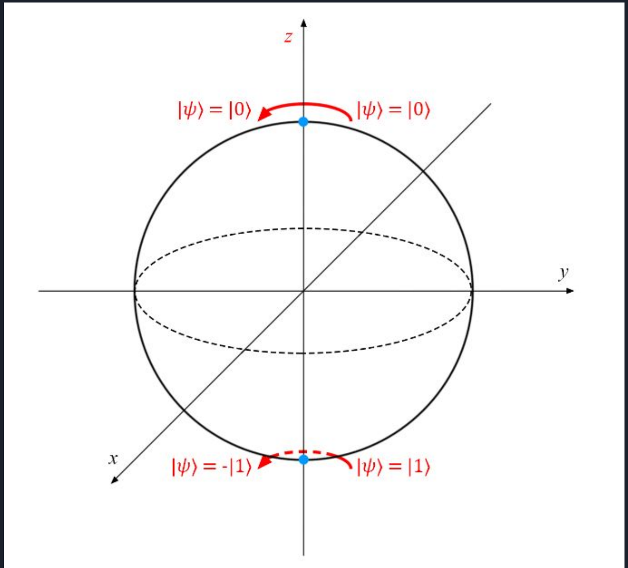
$$Z = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}$$

$$Z|0\rangle = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \end{bmatrix} = |0\rangle$$

Pauli Z Gate



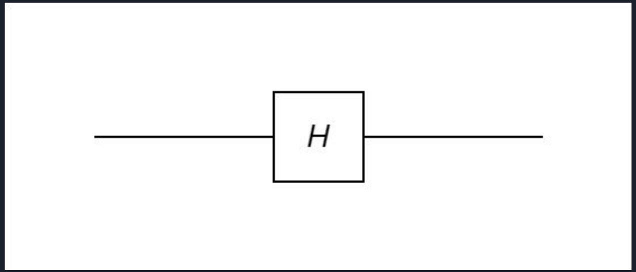




Pauli Z

Hadamard Gate

$$H = \frac{1}{\sqrt{2}}\begin{bmatrix}1 & 1\\ 1 & -1\end{bmatrix} = \begin{bmatrix}\frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}}\\ \frac{1}{\sqrt{2}} & \frac{-1}{\sqrt{2}}\end{bmatrix}$$

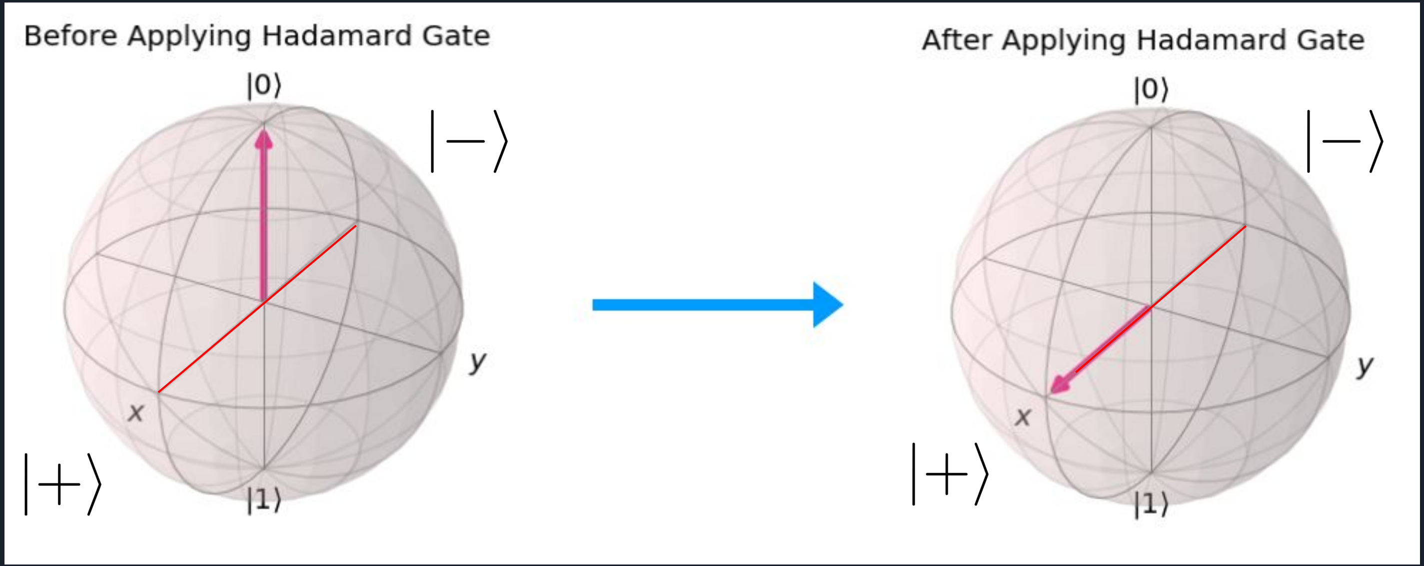


Hadamard Gate

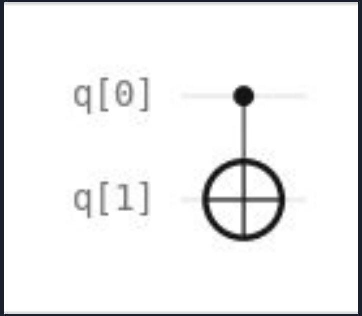
$$H|0\rangle = \frac{1}{\sqrt{2}}\begin{bmatrix}1 & 1\\ 1 & -1\end{bmatrix}\begin{bmatrix}1\\ 0\end{bmatrix} = \frac{1}{\sqrt{2}}|0\rangle + \frac{1}{\sqrt{2}}|1\rangle = |+\rangle$$

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

Hadamard Gate



CNOT Gate



$$CNOT = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

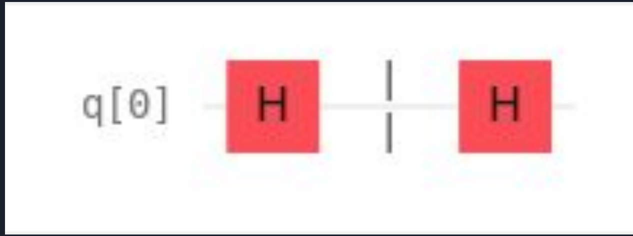
Quantum Phenomena

Quantum Phenomena

There are two main quantum phenomena used in quantum computing:

1. Interference
2. Entanglement

Interference

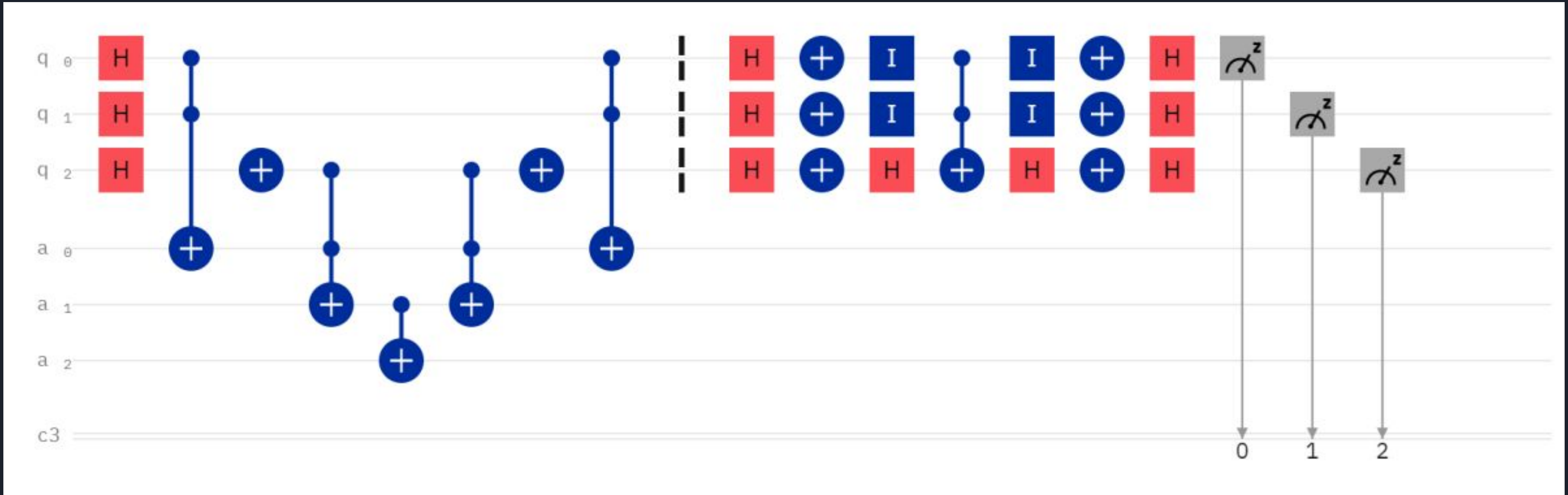


$$H|0\rangle = \frac{1}{\sqrt{2}}|0\rangle + \frac{1}{\sqrt{2}}|1\rangle = \begin{bmatrix} \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} \end{bmatrix} = |+\rangle$$

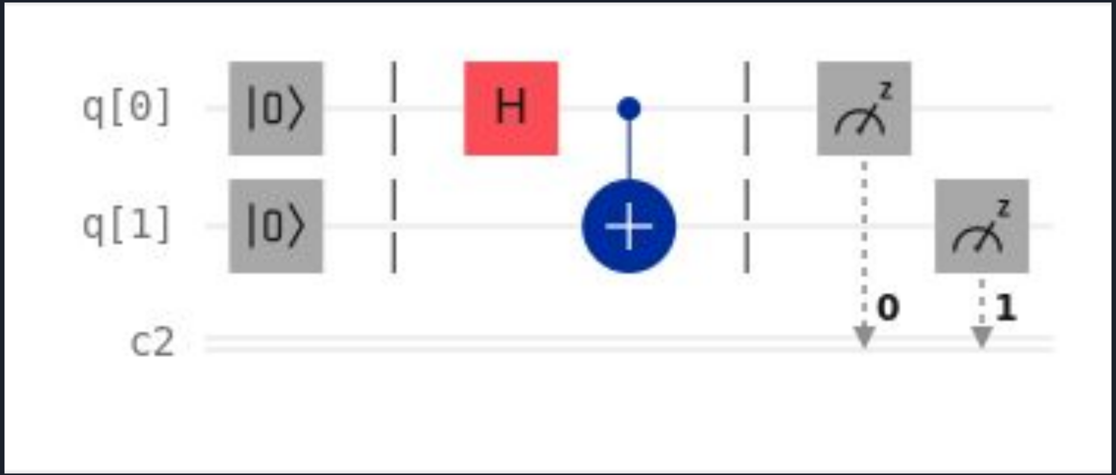
Interference

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

Interference Example



Entanglement



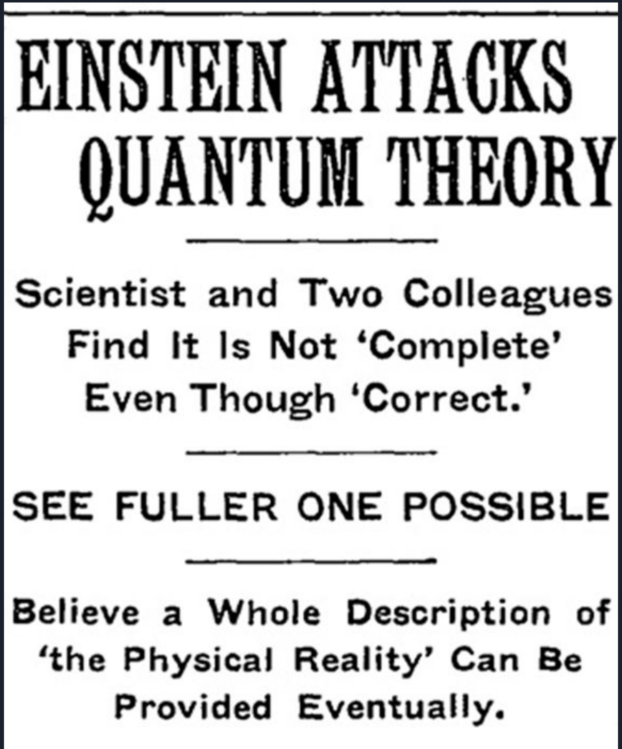
$$|\phi^+\rangle = \frac{1}{\sqrt{2}}|00\rangle + \frac{1}{\sqrt{2}}|11\rangle$$

Entanglement

$$\left|\phi^{+}\right\rangle = \frac{1}{\sqrt{2}}\left|00\right\rangle + 0\cdot\left|01\right\rangle + 0\cdot\left|10\right\rangle + \frac{1}{\sqrt{2}}\left|11\right\rangle$$

$$\left|\phi^{+}\right\rangle = \frac{1}{\sqrt{2}}\left|00\right\rangle + \frac{1}{\sqrt{2}}\left|11\right\rangle$$

Entanglement

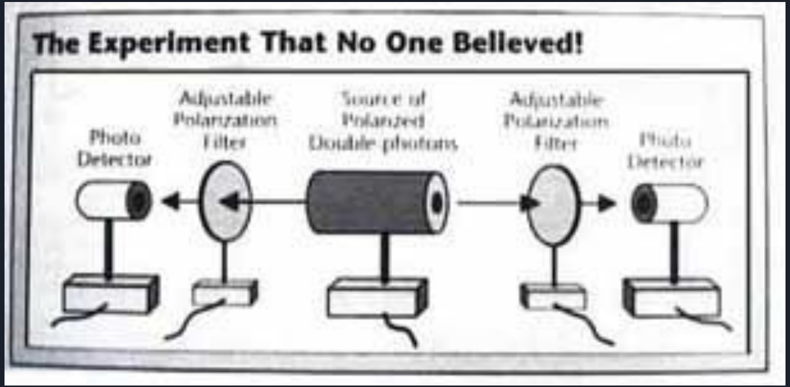


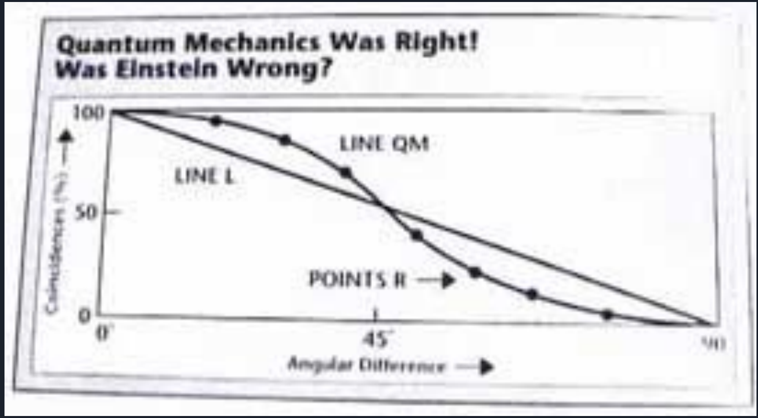
EINSTEIN ATTACKS QUANTUM THEORY

Scientist and Two Colleagues Find It Is Not 'Complete' Even Though 'Correct.'

SEE FULLER ONE POSSIBLE

Believe a Whole Description of 'the Physical Reality' Can Be Provided Eventually.



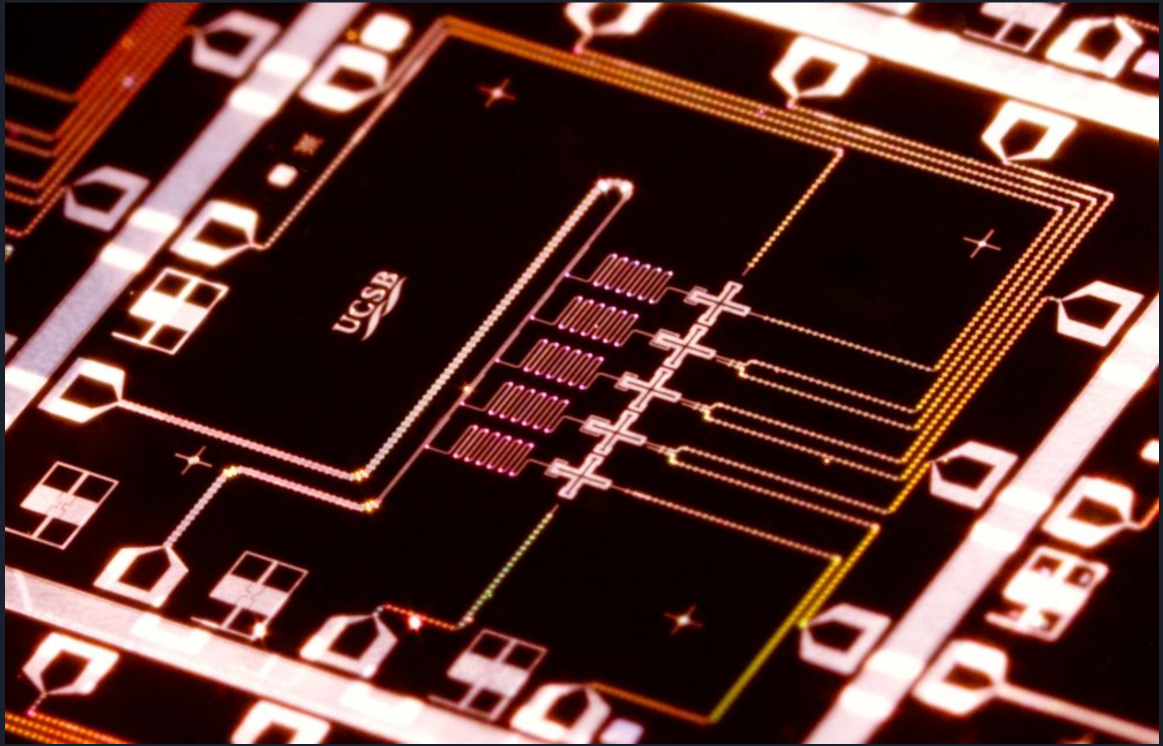


Entanglement

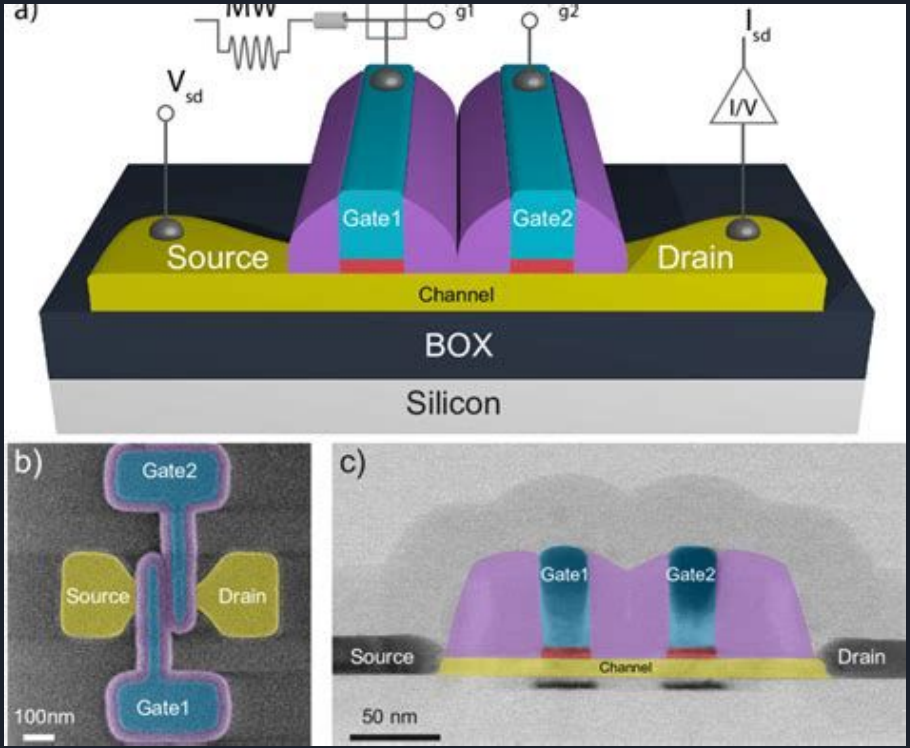
Bell States Example

Quantum Hardware

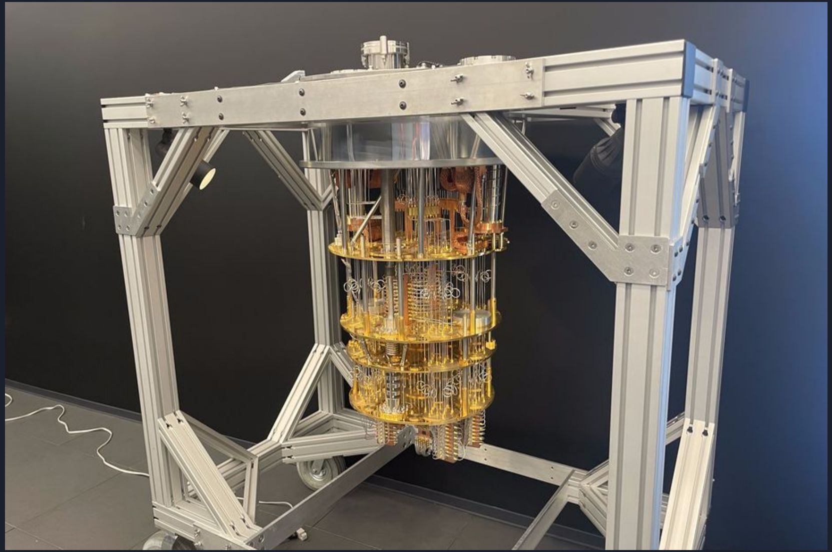
Artificial Qubits



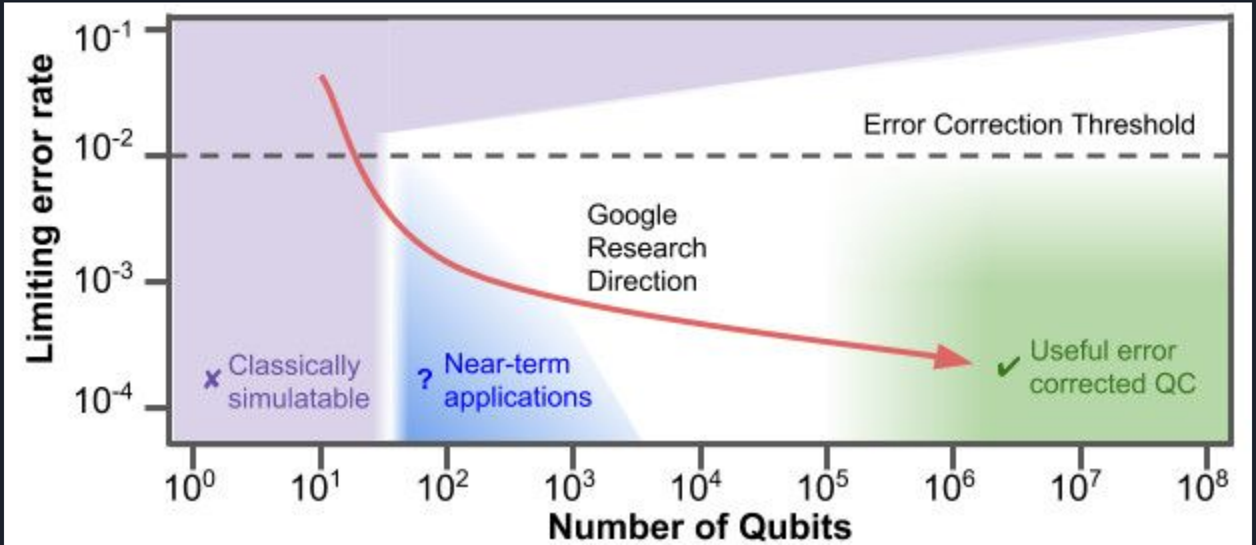
Artificial Qubits



Quantum Computers



Quantum Computers



Development Roadmap | Executed by IBM ✓ On target

IBM Quantum

2019 ✓	2020 ✓	2021 ✓	2022	2023	2024	2025	Beyond 2026
Run quantum circuits on the IBM cloud	Demonstrate and prototype quantum algorithms and applications	Run quantum programs 100x faster with Qiskit Runtime	Bring dynamic circuits to Qiskit Runtime to unlock more computations	Enhancing applications with elastic computing and parallelization of Qiskit Runtime	Improve accuracy of Qiskit Runtime with scalable error mitigation	Scale quantum applica-tions with circuit knitting toolbox controlling Qiskit Runtime	Increase accuracy and speed of quantum workflows with integration of error correction into Qiskit Runtime

Model Developers
- Prototype quantum software applications → Quantum software applications
- Machine learning | Natural science | Optimization

Algorithm Developers
- Quantum algorithm and application modules ✓
  - Machine learning | Natural science | Optimization
- Quantum Serverless
  - Intelligent orchestration
  - Circuit Knitting Toolbox
  - Circuit libraries

Kernel Developers
- Circuits ✓
- Qiskit Runtime ✓
  - Dynamic circuits (on target)
  - Threaded primitives
  - Error suppression and mitigation
  - Error correction

System Modularity
- Falcon, 27 qubits ✓
- Hummingbird, 65 qubits ✓
- Eagle, 127 qubits ✓
- Osprey, 433 qubits (on target)
- Condor, 1,121 qubits
- Flamingo, 1,386+ qubits
- Kookaburra, 4,158+ qubits
- Scaling to 10K-100K qubits with classical and quantum communication
- Heron, 133 qubits x p
- Crossbill, 408 qubits

Quantum Computers

Quantum Software

Quantum Software

- Many different frameworks exist for quantum programming
- Quantum assembly languages are used to interact directly with the hardware
- Higher-level languages / libraries are compiled to quantum assembly language

Quantum Assembly Language

OpenQASM

- Most popular quantum assembly language
- Open Source

Quil

- Alternative quantum assembly language

Quantum Programming Languages

Qiskit

- Supported by IBM
- Implemented as a Python Library

Cirq

- Developed by Google
- Implemented as a Python Library

Q#

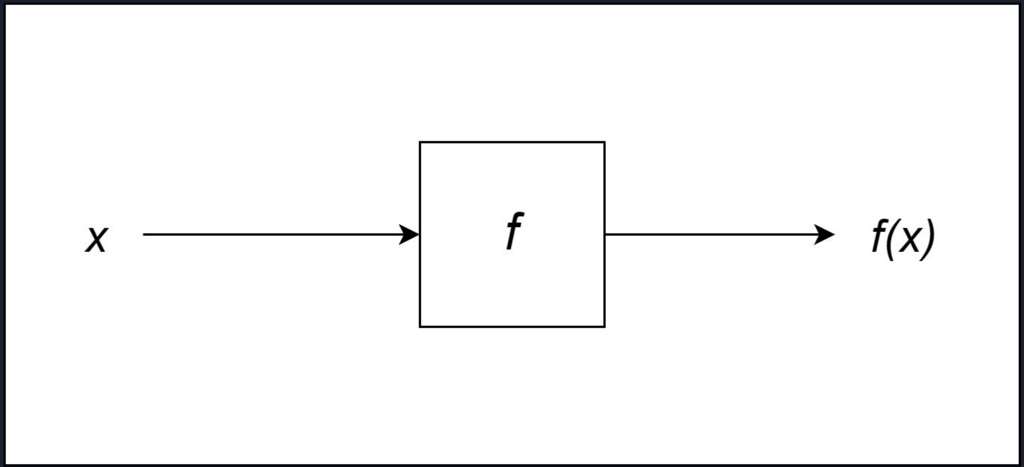
- Developed by Microsoft
- Based on C family of languages

Quantum Algorithms

Quantum Supremacy

- Quantum computers can perform specific tasks more efficiently than classical computers
- This is known as Quantum Supremacy
- Quantum algorithms may be more efficient in terms of:
  - Computational Speed
  - Memory Required
- Since classical memory is abundant and cheap, we are more interested in *speed*

Deutsch's Problem

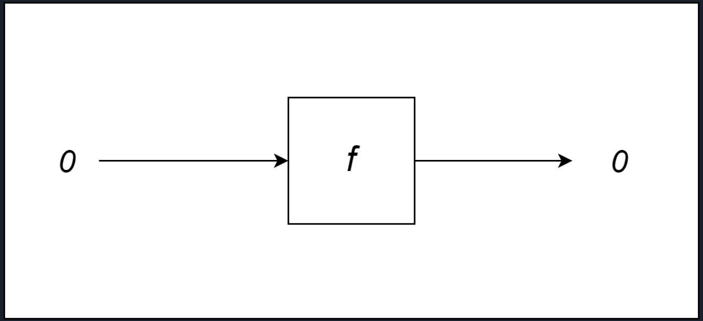


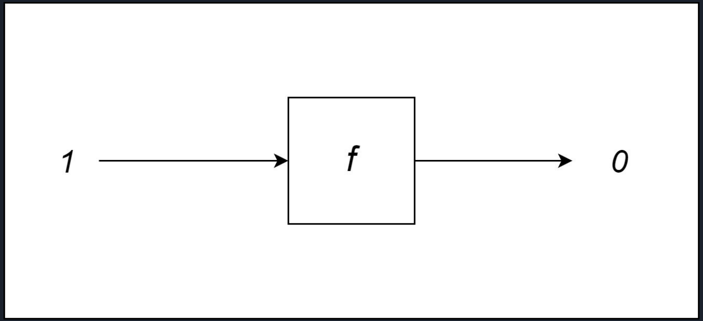
$$f : \{0, 1\} \to \{0, 1\}$$

Deutsch's Problem

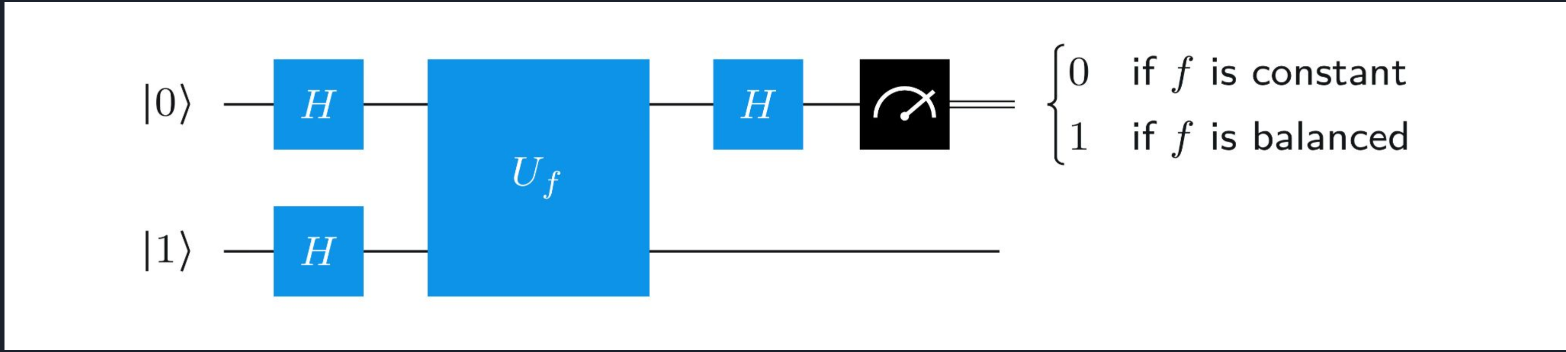
Input	Constant Functions		Balanced Functions	
x	$f_0(x)$	$f_1(x)$	$f_2(x)$	$f_3(x)$
0	0	1	0	1
1	0	1	1	0

Deutsch’s Problem

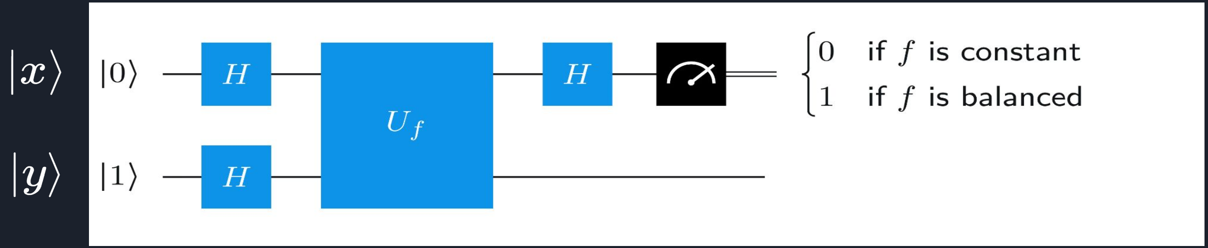




Deutsch's Algorithm



Query Gate



$$U_f|x\rangle|y\rangle = |x\rangle|y \oplus f(x)\rangle$$

$$\text{if } |y\rangle = |-\rangle\text{: } U_f|x\rangle|-\rangle = (-1)^{f(x)}|x\rangle|-\rangle$$

Phase Query Gate

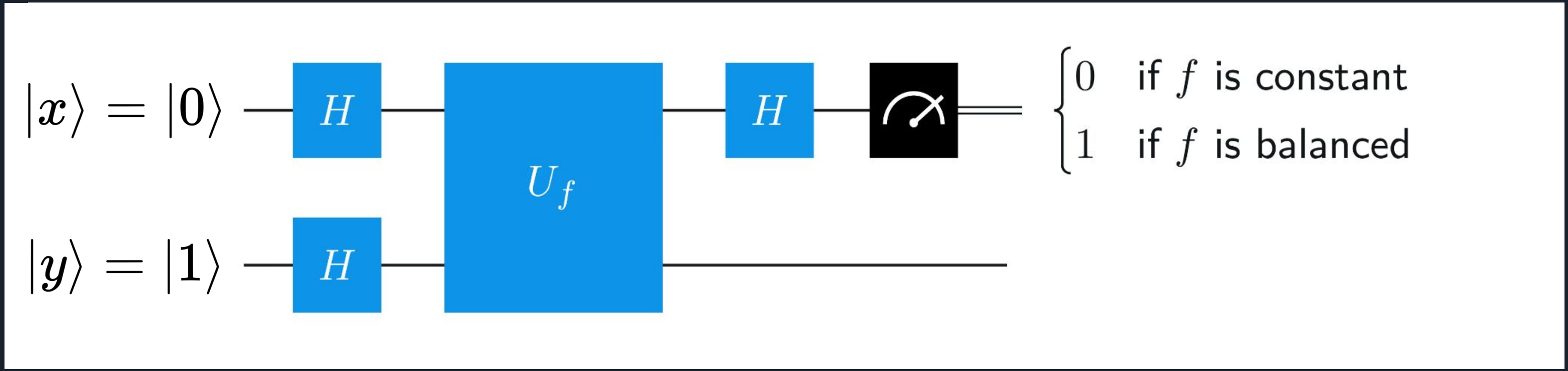
$$\text{if } |y\rangle = |-\rangle : \ U_f|x\rangle|-\rangle$$

$$\text{if f(x)} = 0 : \ U_f|x\rangle|-\rangle \ = \ |x\rangle|-\rangle$$

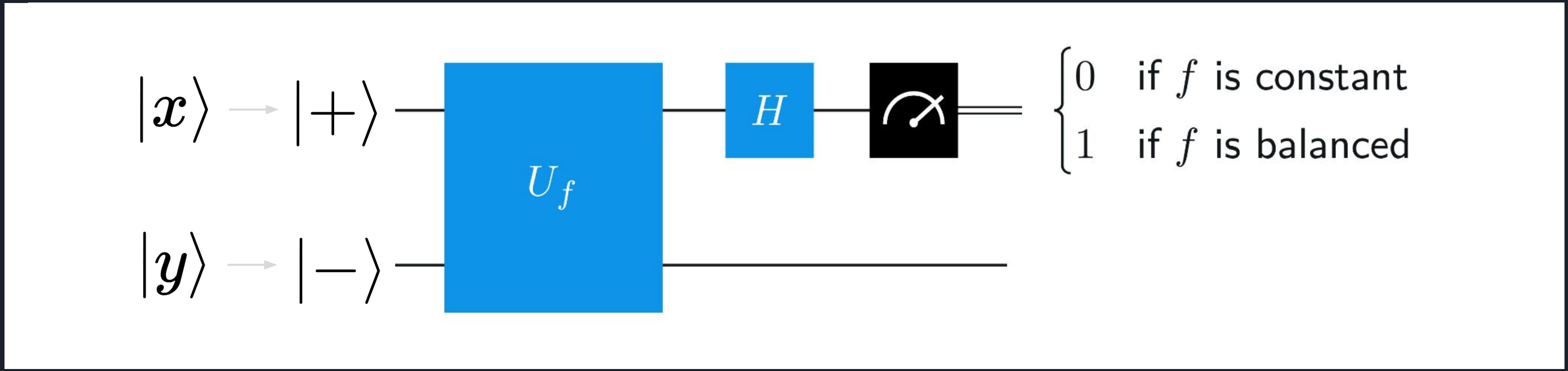
$$\text{if f(x)} = 1 : \ U_f|x\rangle|-\rangle \ = \ -|x\rangle|-\rangle$$

$$U_f|x\rangle|-\rangle \ = \ (-1)^{f(x)}|x\rangle|-\rangle$$

Deutsch's Algorithm

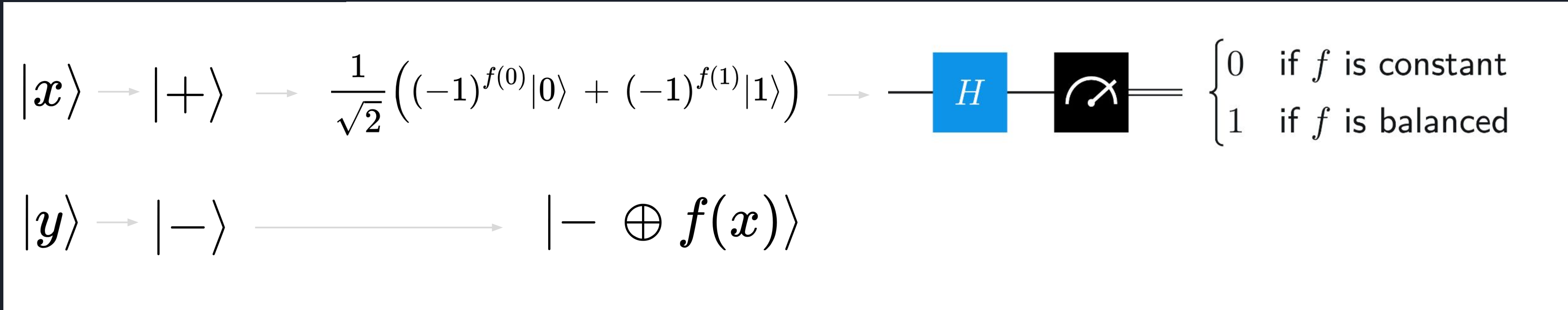


Deutsch's Algorithm



Deutsch’s Algorithm

$$U_f|+\rangle|-\rangle = (-1)^{f(x)}|+\rangle|-\rangle$$



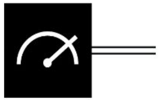
Deutsch's Algorithm

$$\frac{1}{\sqrt{2}}\Big((-1)^{f(0)}|0\rangle + (-1)^{f(1)}|1\rangle\Big) \begin{cases} \text{if } f(0) = f(1): & \pm|+\rangle \\ \text{if } f(0) \neq f(1): & \pm|-\rangle \end{cases} \longrightarrow$$

$$\begin{cases} 0 & \text{if } f \text{ is constant} \\ 1 & \text{if } f \text{ is balanced} \end{cases}$$

Deutsch's Algorithm

$$\begin{cases} \text{if } f(0) = f(1): & \pm|+\rangle \longrightarrow \pm H|+\rangle = |0\rangle \\ \text{if } f(0) \neq f(1): & \pm|-\rangle \longrightarrow \pm H|-\rangle = |1\rangle \end{cases} \longrightarrow$$



$$\begin{cases} 0 & \text{if } f \text{ is constant} \\ 1 & \text{if } f \text{ is balanced} \end{cases}$$

Demonstration

Other Quantum Algorithms

- Deutsch-Jozsa Algorithm
- Grover's Algorithm
- Shor's Algorithm
- Quantum Fourier Transform
- Many other quantum algorithms

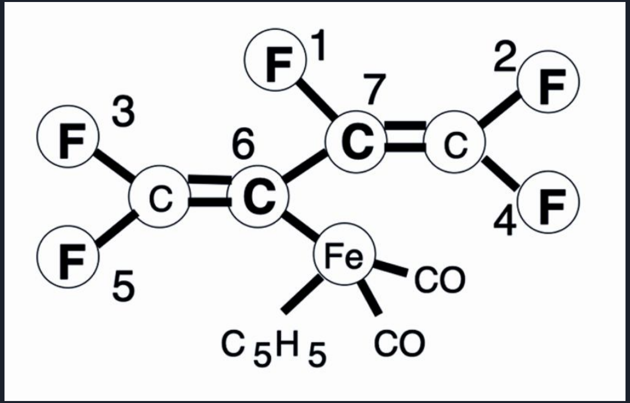
Conclusion

Conclusion

- We are rapidly moving towards the useful, error-corrected era of quantum computing
- Quantum computers work in tandem with classical devices, to solve some problems more efficiently than classical computers
- For the foreseeable future, quantum computers will likely exist as cloud-based, timesharing infrastructure
- Quantum computing is a rapidly evolving field, with the potential to solve problems that could not be realistically computed on a classical computer

Questions

Bonus Material



The perfluorobutadienyl iron complex molecule

https://www.ibm.com/quantum/blog/factor-15-shors-algorithm