

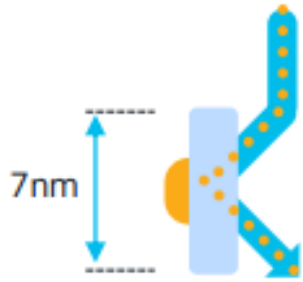
บทที่ 4 INTRO TO QUANTUM COMPUTING



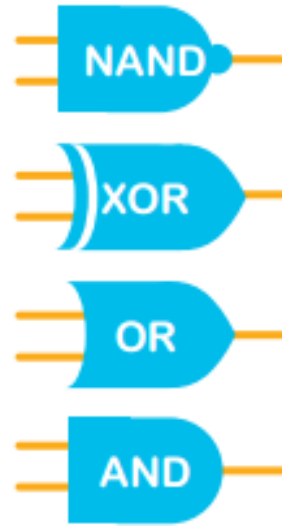
ผู้ช่วยศาสตราจารย์จุฑาวุฒิ จันทรมานี

หลักสูตรวิทยาศาสตรบัณฑิต สาขาวิชาวิทยาการคอมพิวเตอร์
คณะวิทยาศาสตร์และเทคโนโลยี มหาวิทยาลัยสวนดุสิต

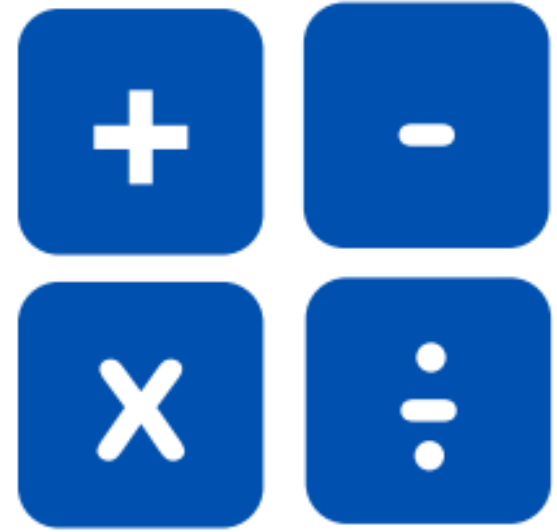
Digital Circuits Quick Recap



Transistors



Logic Gates



Circuits

Input

0
1

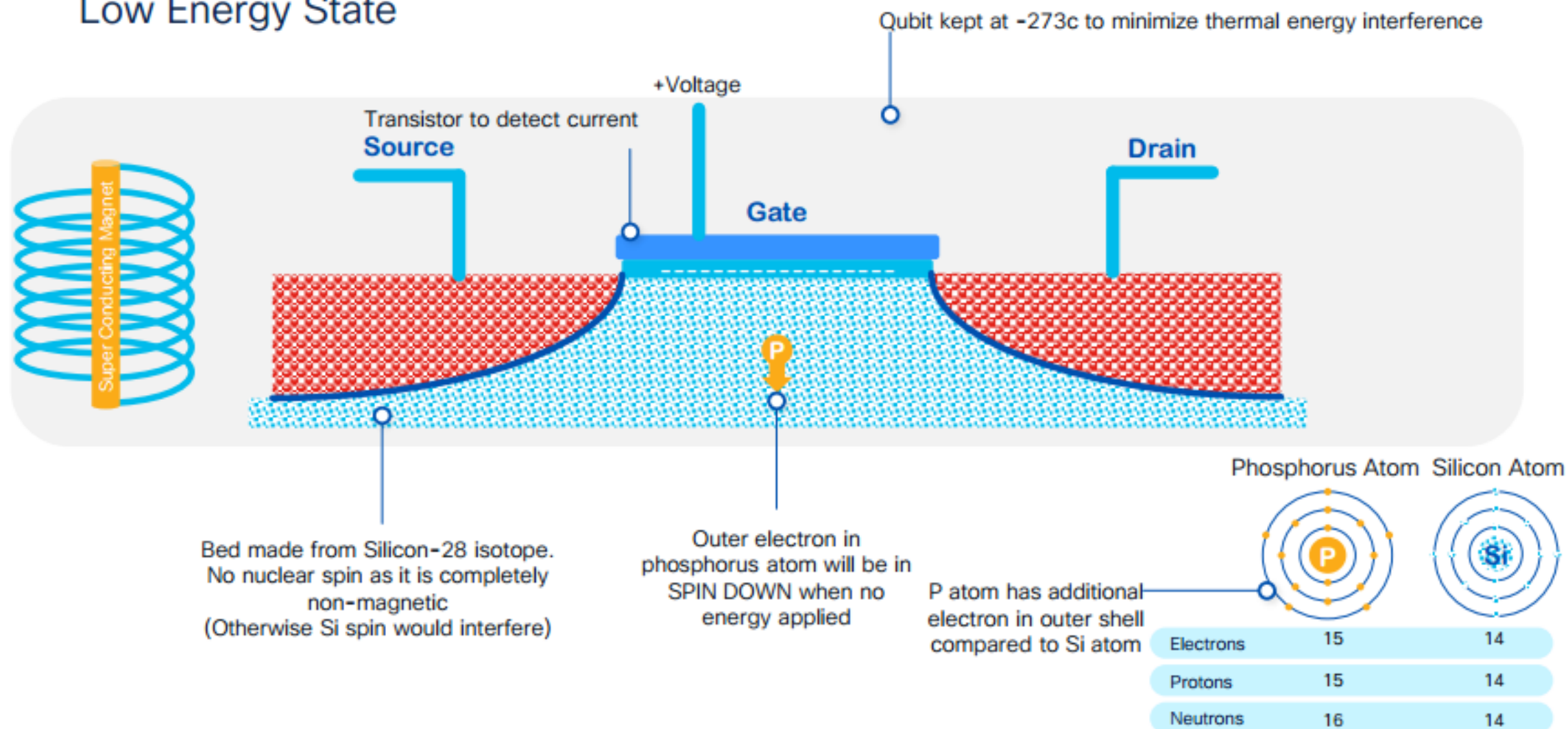


0

Output

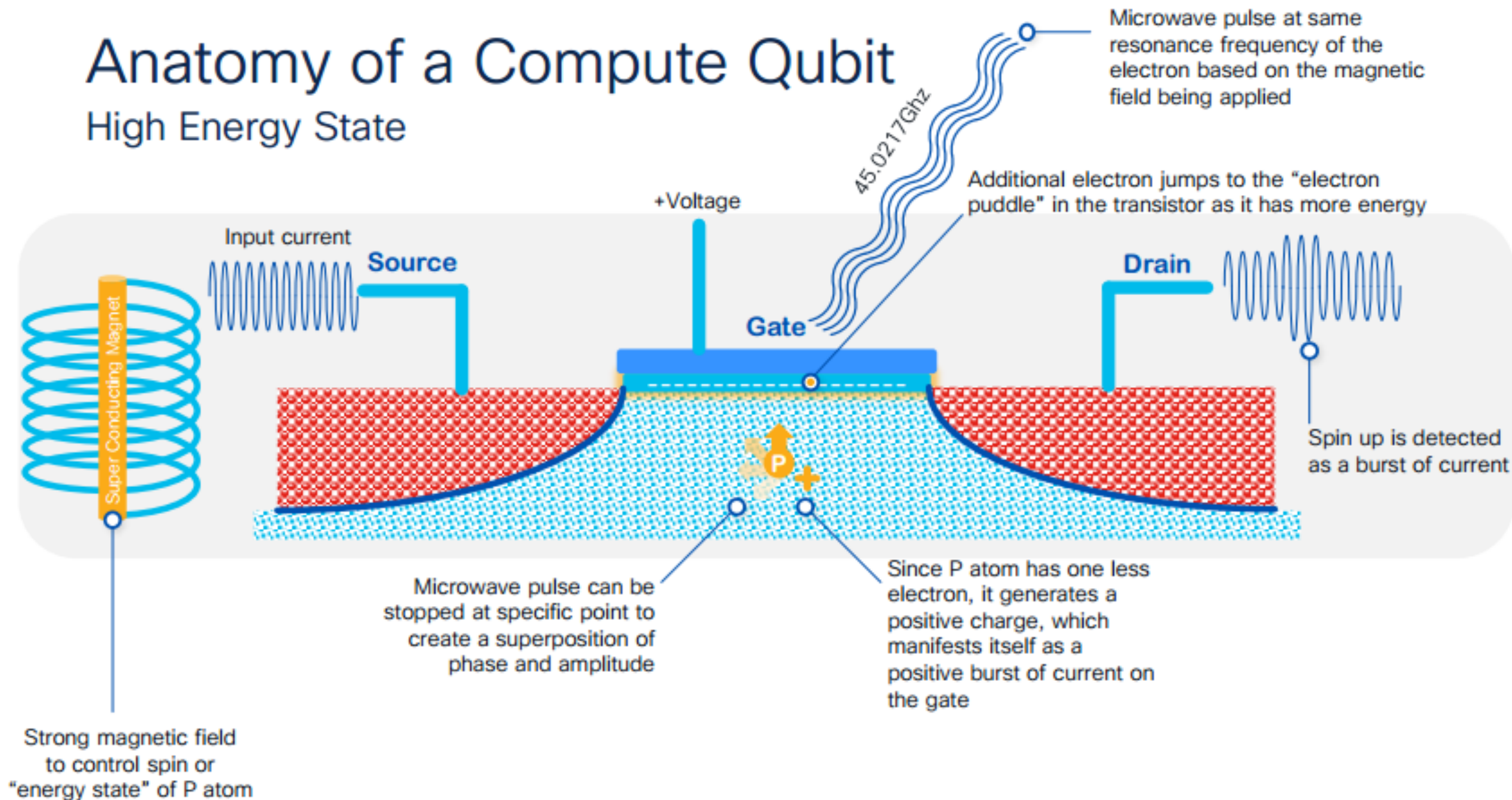
Anatomy of a Compute Qubit

Low Energy State

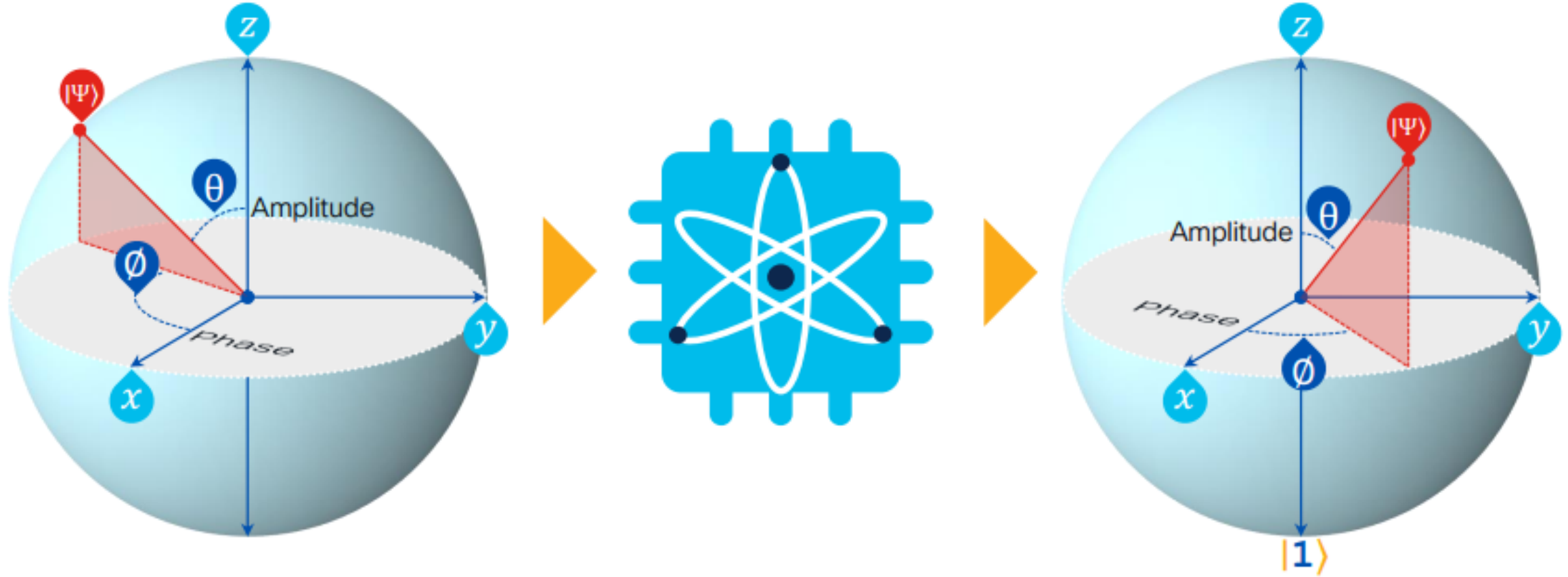


Anatomy of a Compute Qubit

High Energy State



Quantum Gate Operation



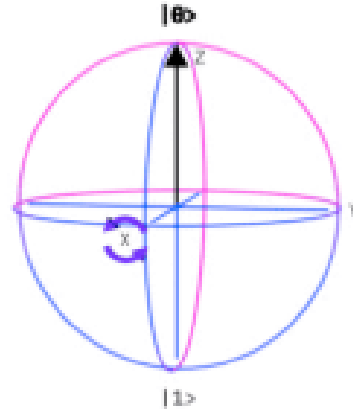
Quantum Gates :

- manipulate Amplitude θ and Phase ϕ of the state vector
- take superpositions as inputs, rotate their probabilities, and produce *another* superposition as outputs

Quantum Gate Examples

X

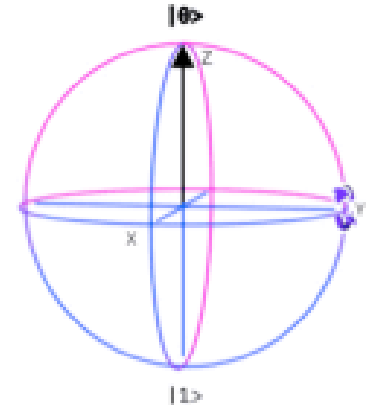
Pauli-X Gate is a NOT operation. It will turn a spin-up state to a spin-down and visa versa.



$|0\rangle \rightarrow |1\rangle$

Y

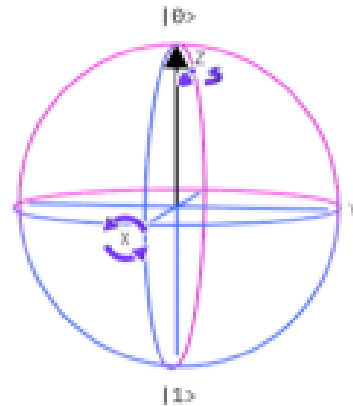
Y-gate rotates around the Y-axis. It is similar to the X-gate but different in phase.



$|0\rangle \rightarrow |1\rangle$

H

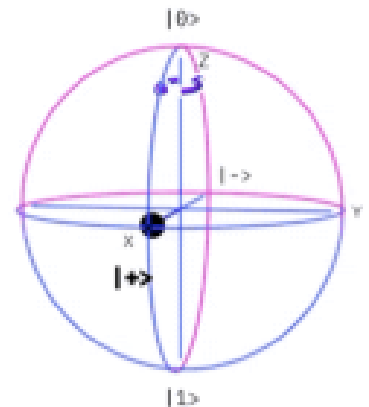
Hadamard Gate sets the qubit into a superposition state of a 50/50 chance that it will end up as $|0\rangle$ or $|1\rangle$.



$|0\rangle \rightarrow |+\rangle$

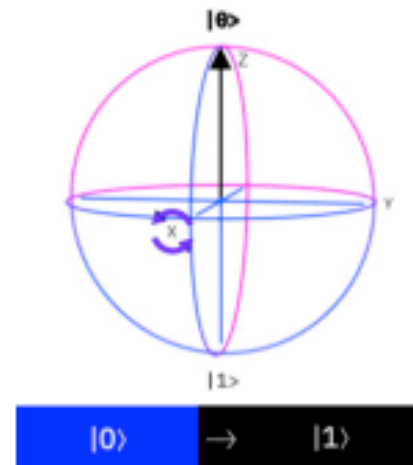
T

T gate rotates a qubit $\pi/4$ around the z-axis.



$|+\rangle \rightarrow (|0\rangle + j\pi/4|1\rangle)/\sqrt{2}$

Quantum NOT Gate Example (Pauli-X Gate)



MATRIX

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

NOT Gate

VECTOR

$$\begin{bmatrix} 1 \\ 0 \end{bmatrix}$$

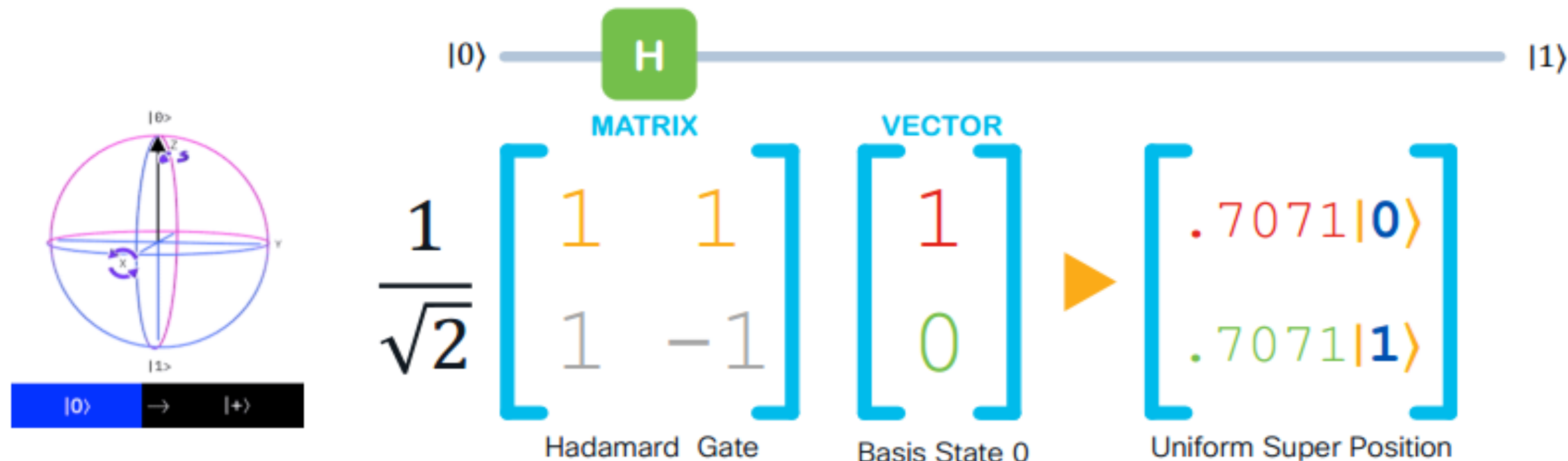
Basis State $|0\rangle$

Row x Column

$$\begin{aligned} &0 \times 1 + 1 \times 0 = 0 \\ &1 \times 1 + 0 \times 0 = 1 \end{aligned}$$

Basis State $|1\rangle$

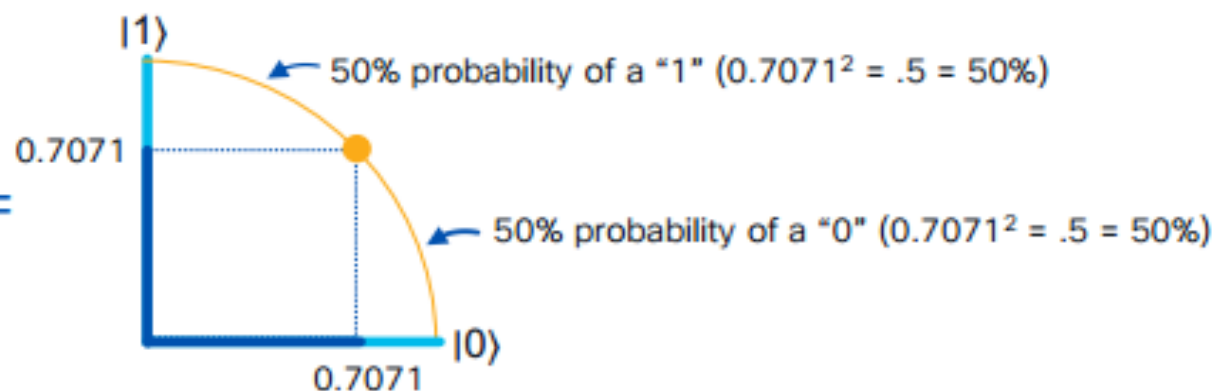
Hadamard Gate Example (Set to 50/50 State)



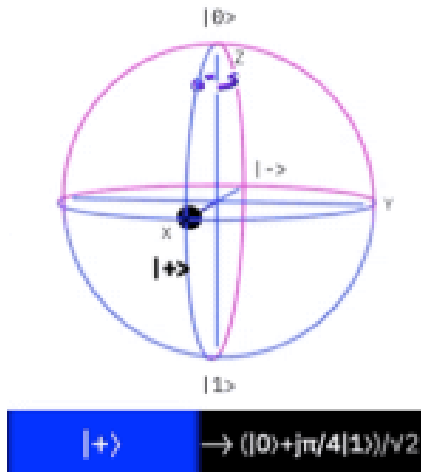
$$|\psi\rangle = \frac{1}{\sqrt{2}} |0\rangle + \frac{1}{\sqrt{2}} |1\rangle$$

$$|\psi\rangle = 0.7071|0\rangle + 0.7071|1\rangle =$$

Sums of the squares of probabilities must equal 1



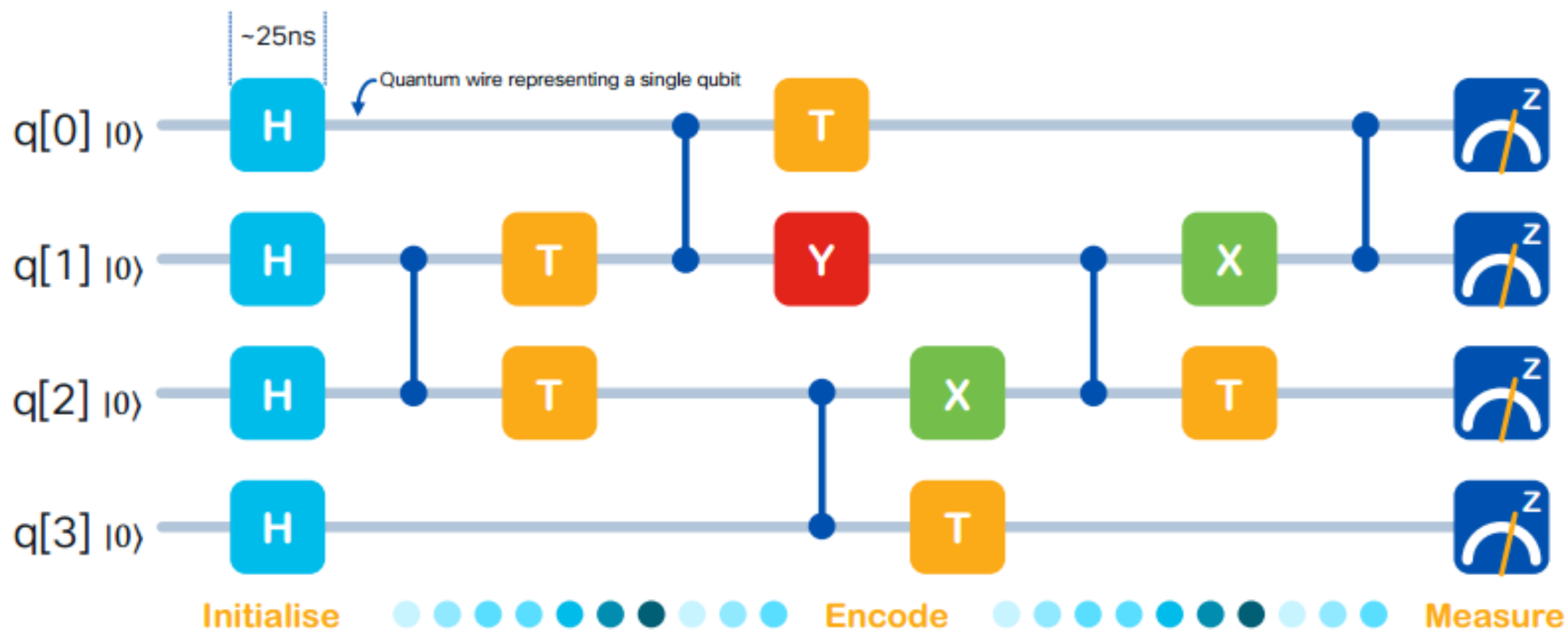
T Gate Example (Rotate $\pi/4$ around the Z-axis)



$$\begin{array}{ccc}
 \text{MATRIX} & \text{VECTOR} & \\
 \left[\begin{array}{cc} 1 & 0 \\ 0 & e^{i\pi/4} \end{array} \right] & \left[\begin{array}{c} \alpha \\ \beta \end{array} \right] & \rightarrow \left[\begin{array}{c} \alpha|0\rangle \\ e^{i\pi/4} \beta|1\rangle \end{array} \right] \\
 \text{T Gate} & \text{Basis State} & \text{Rotated position}
 \end{array}$$

Quantum Circuits

Include Both Quantum Operators + Classical Computing



Number of input Qubits must match number of output Qubits

IBM Quantum Composer

The screenshot displays the IBM Quantum Composer interface. The top bar includes the title "IBM Quantum Composer" and navigation icons. The left sidebar shows a "Files" panel with a list of files. The main workspace contains a quantum circuit diagram with gates and qubits. The right sidebar shows the "Setup and run" panel with a "Simulator seed" and a "Code" panel displaying OpenQASM 2.0 code. The bottom section features three visualization panels: "Measurement Probabilities", "Q-sphere", and "Statevector".

Files Panel:

- 2 files
- New file +
- Untitled circuit (a minute ago)
- Untitled circuit (2 hours ago)

Menu Bar: File, Edit, Inspect, View, Sha

Quantum Circuit:

- Qubits: q0, q1, q2, c3
- Gates: H, CNOT, CNOT, CNOT, T, S, Z, T†, S†, P, RZ, |0⟩, if, √X, √X†, Y, RX, RY, U, RXX, RZ
- Measurement: Measurement gate on q0

Right Panel: Setup and run

- Simulator seed: 4393
- OpenQASM 2.0
- Open in Quantum Lab
- Code:

```
1 OPENQASM 2.0;
2 include "qelib1.inc";
3
4 qreg q[3];
5 creg c[3];
6
7 h q[0];
8 cx q[0],q[1];
9 measure q[0] -> c[0];
```

Bottom Panels:

- Measurement Probabilities:** Bar chart showing probabilities for computational basis states. The state 000 has a probability of 1.0.
- Q-sphere:** Bloch sphere visualization showing the state of the qubit. The state is at the top pole (0,0,1).
- Statevector:** Bar chart showing the amplitudes for computational basis states. The state 000 has an amplitude of 1.0.

<https://learning.quantum.ibm.com/tutorial/explore-gates-and-circuits-with-the-quantum-composer>

Quantum Parallelism



Holds &
operates on
values of
0 and 1
simultaneously



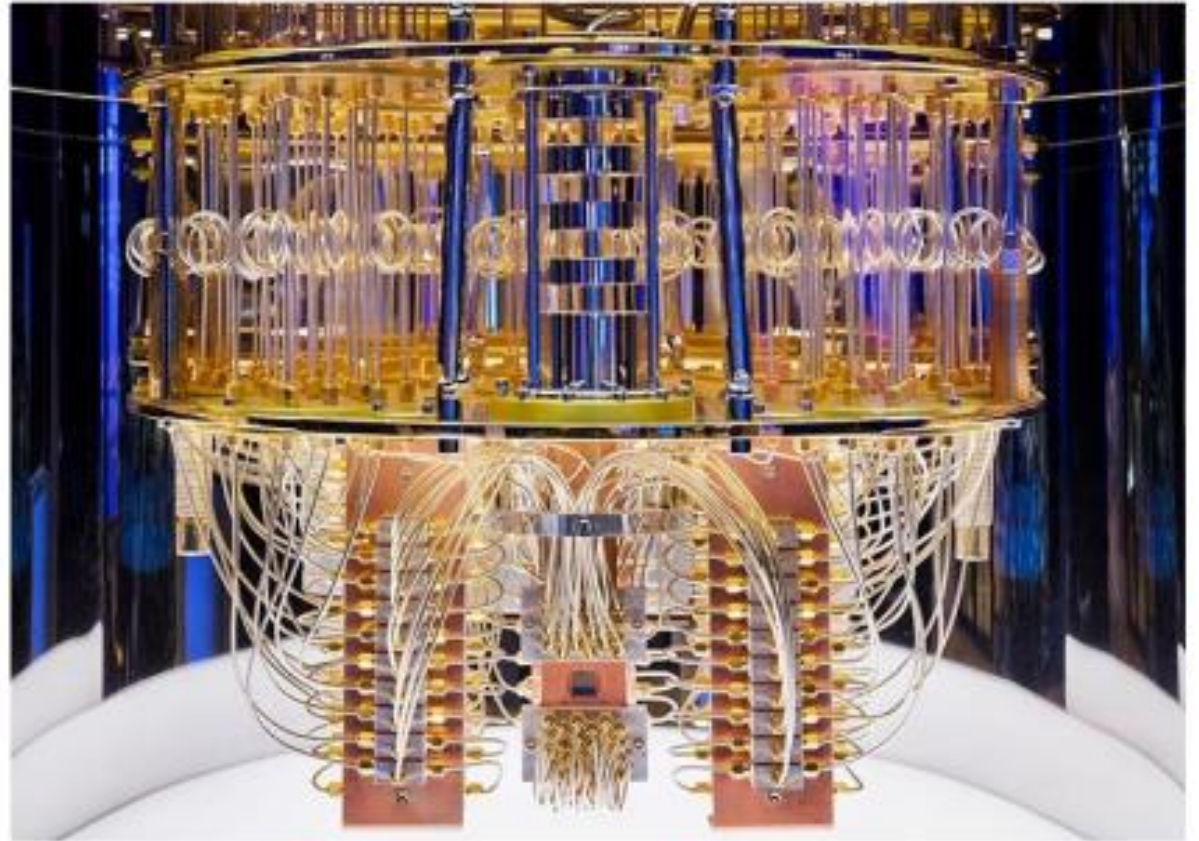
Holds &
operates on
values of
00, 01, 10, 11
simultaneously



Holds & operates
on **values** of
000, 001, 010,
011, 100, 101,
110, 111
simultaneously

How Much Faster is a Quantum Computer?

- In 2021, the world's largest quantum computer had 127 Qubits
- This machine was **158M** times faster than its classic counterpart
- Example Task Execution Times:
 - Classic Computer: 2,500 years
 - Quantum Computer: **1 minute**

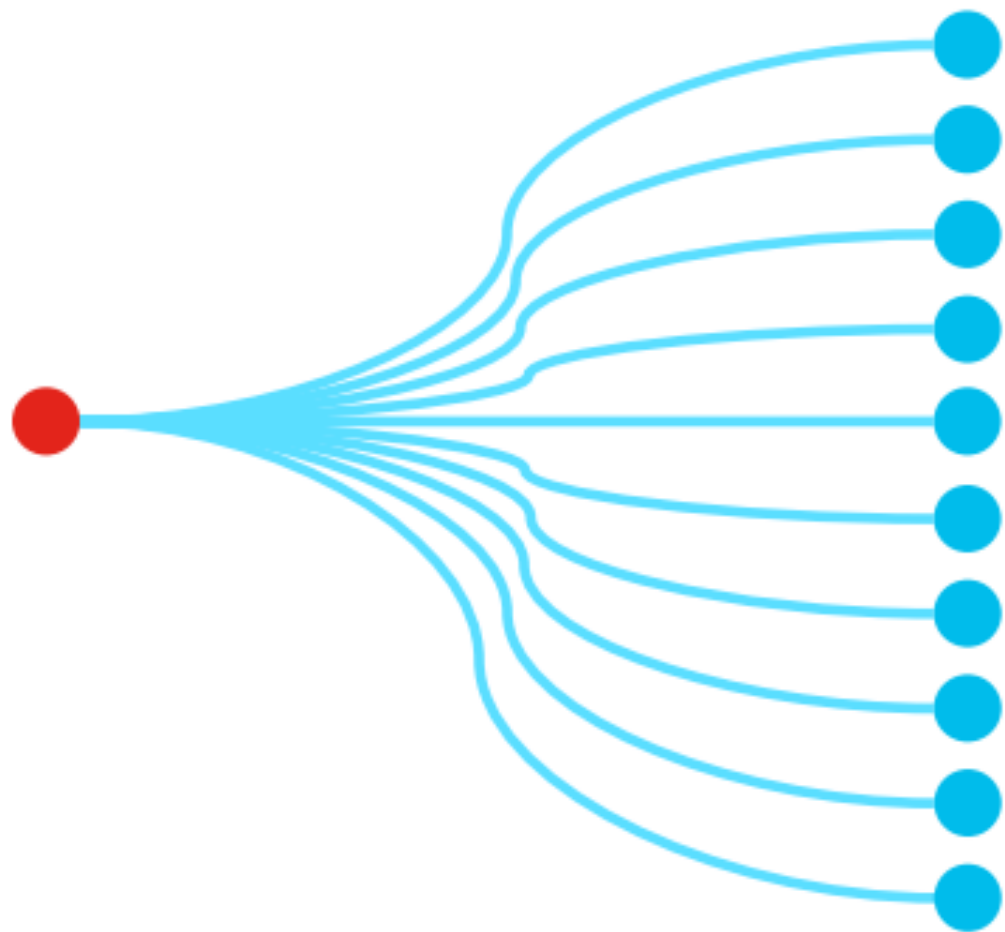


$$2^{127} = 1.7 \times 10^{38} \text{ values}$$

170,000,000,000,000,000,000,000,000,000,000,000

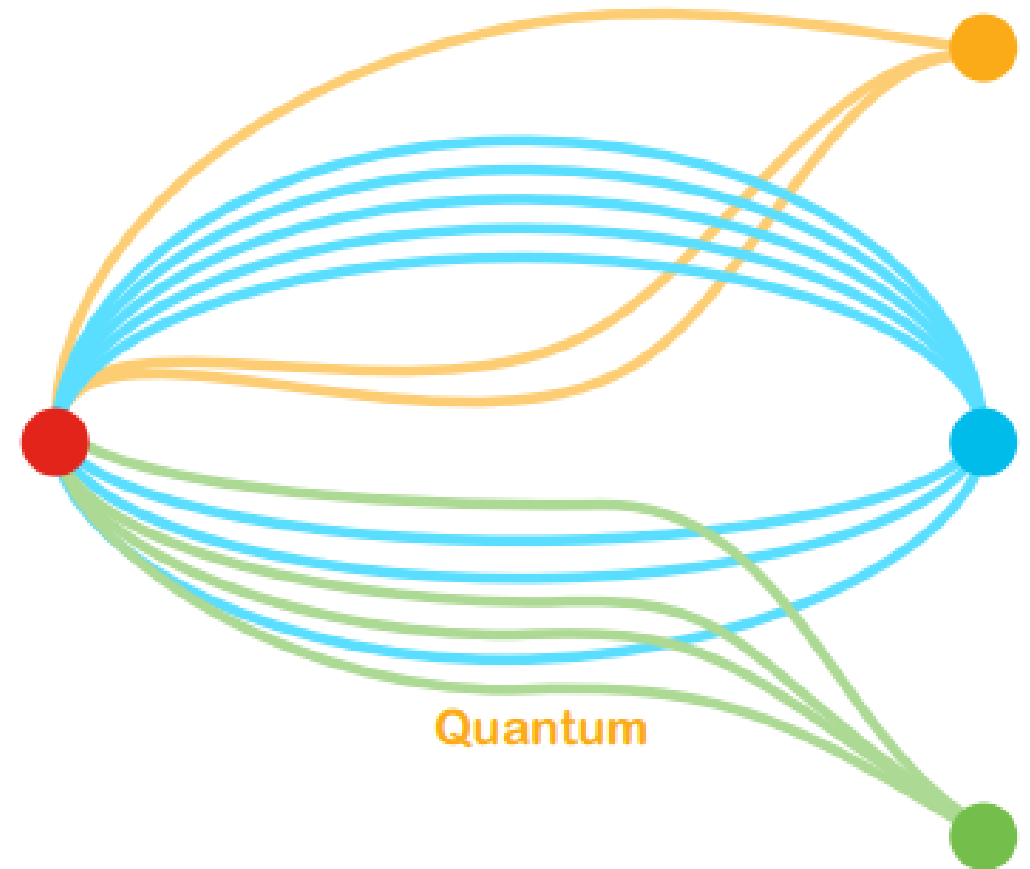
Classical Computing Problem Solving

- A classic computer needs to sequentially iterate through a problem until the correct result is found



Quantum Computing Problem Solving

- Quantum computing can provide a single or small number of answers with the highest probability of being correct, which narrows down the search for the correct solution



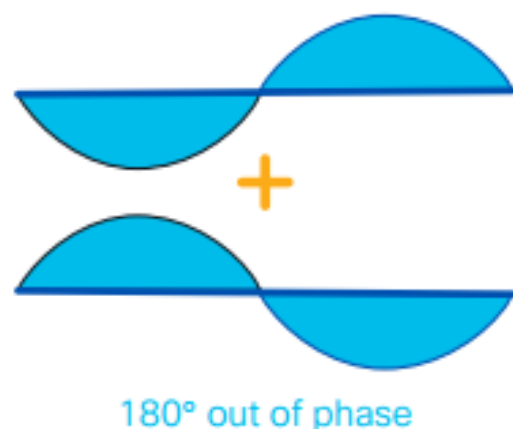
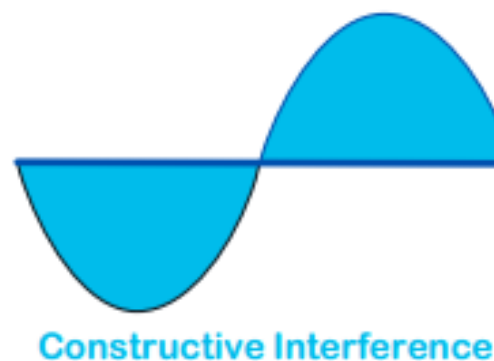
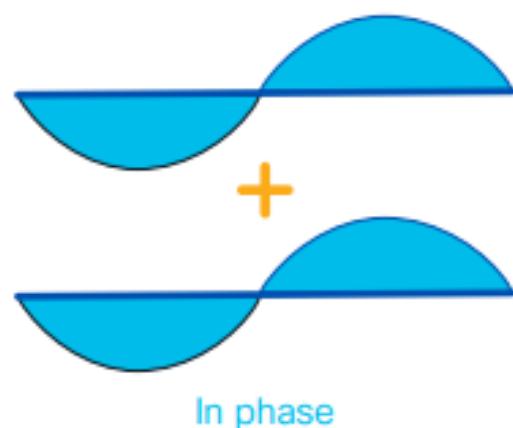
Interference Manipulation

- Another benefit that can be realized by quantum computing comes from manipulating interference
- Interference may be
 - constructive or
 - destructive
- Programmers of quantum algorithms (like Grover's and Shor's algorithms) endeavor to arrange qubits so that :
 - **correct** answers generate **constructive interference**
 - **incorrect** answers generate **destructive interference**
- Remember: Probability = Amplitude²

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

Sums of the squares
of probabilities must equal 1

$$\alpha^2 + \beta^2 = 1$$



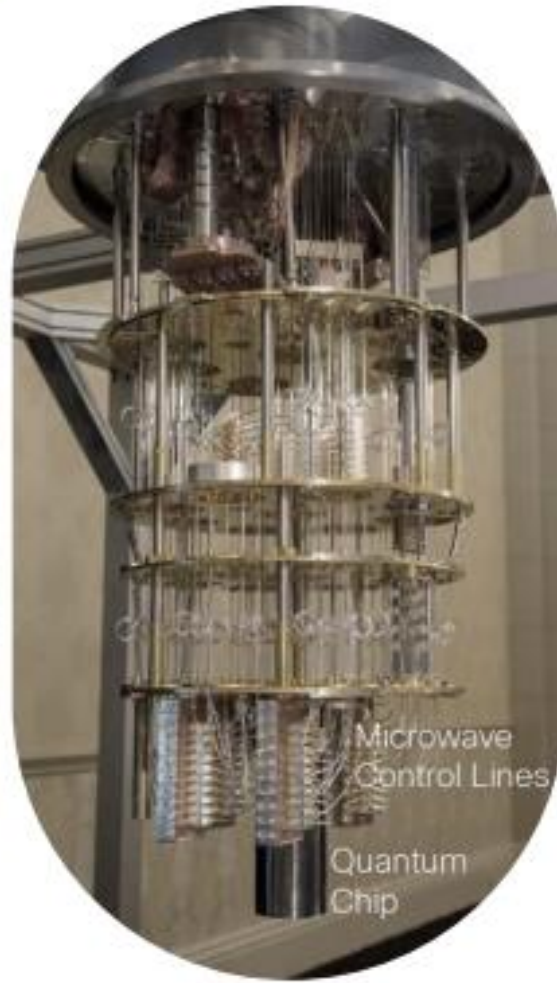
What Do Quantum Computers Look Like?



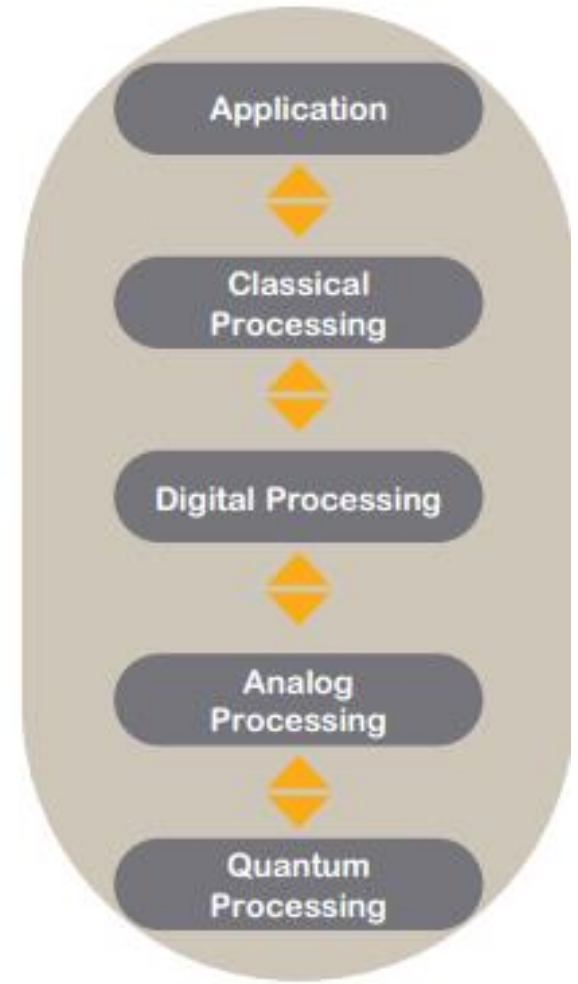
External (IBM)



External (IBM)



Internal (IBM)



Functions

