

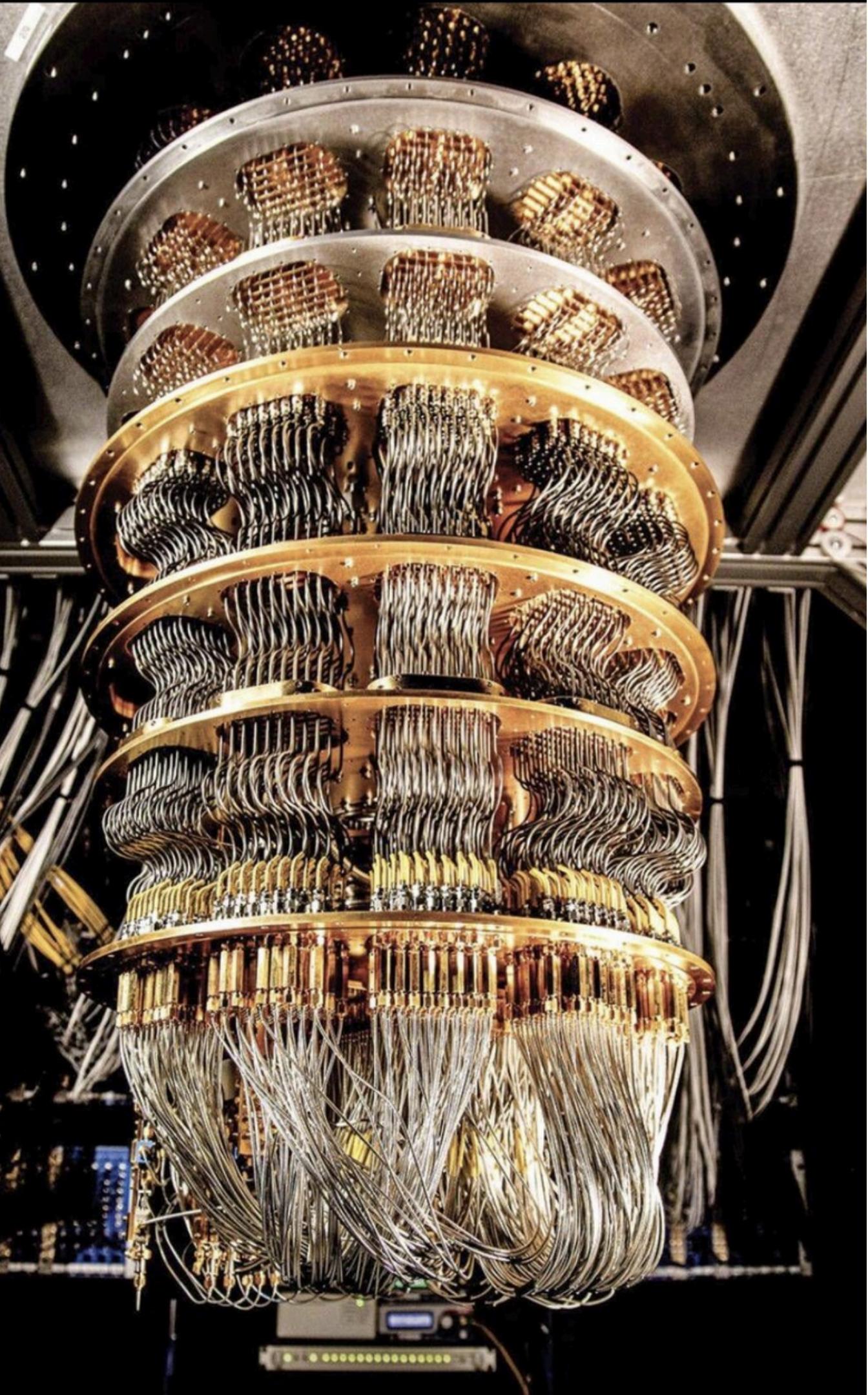
Quantum Networks

The future of the internet ...

Jan Krzyszkowski

Spis treści

- Wstęp
- Quantum Computing
- Quantum Network
- Teleportacja



Wstęp

Biblioteki do pobrania:

- qiskit
- qiskit_aer
- pandas

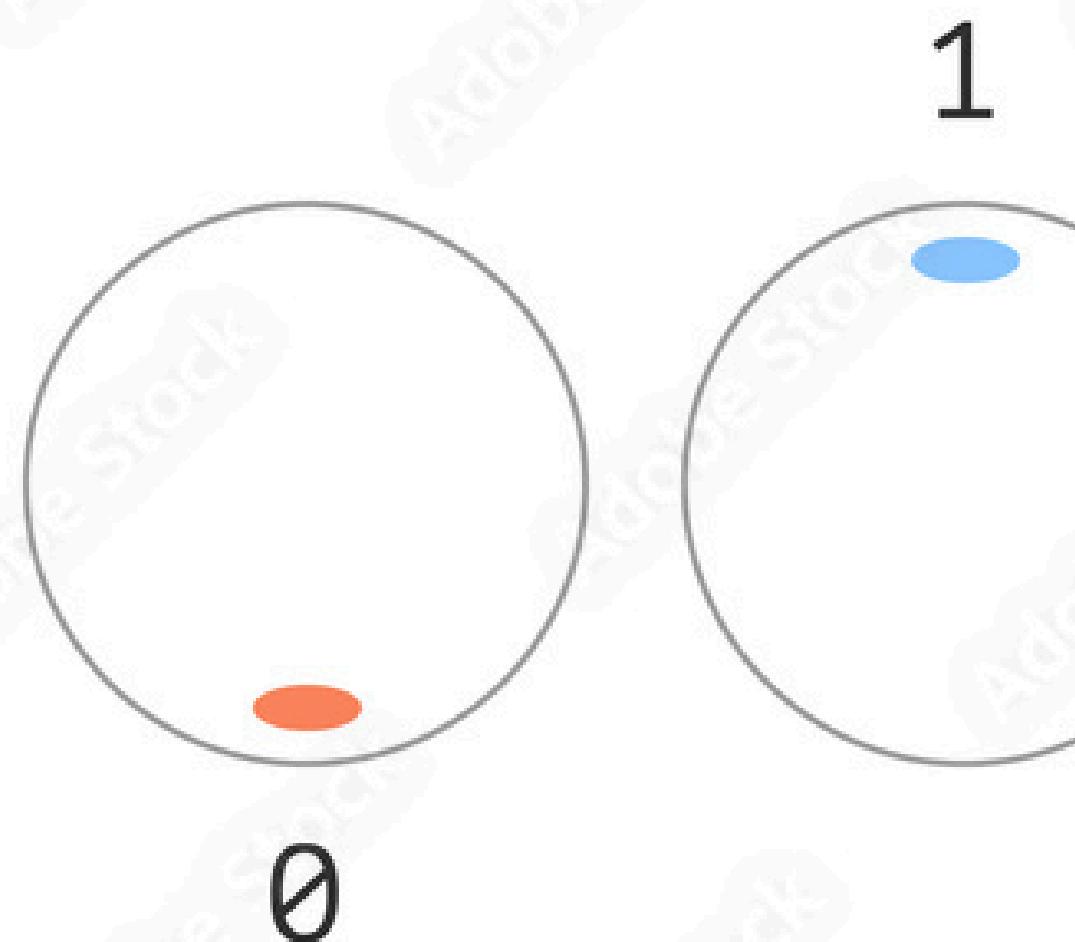
Używamy pythona

Quantum Computing

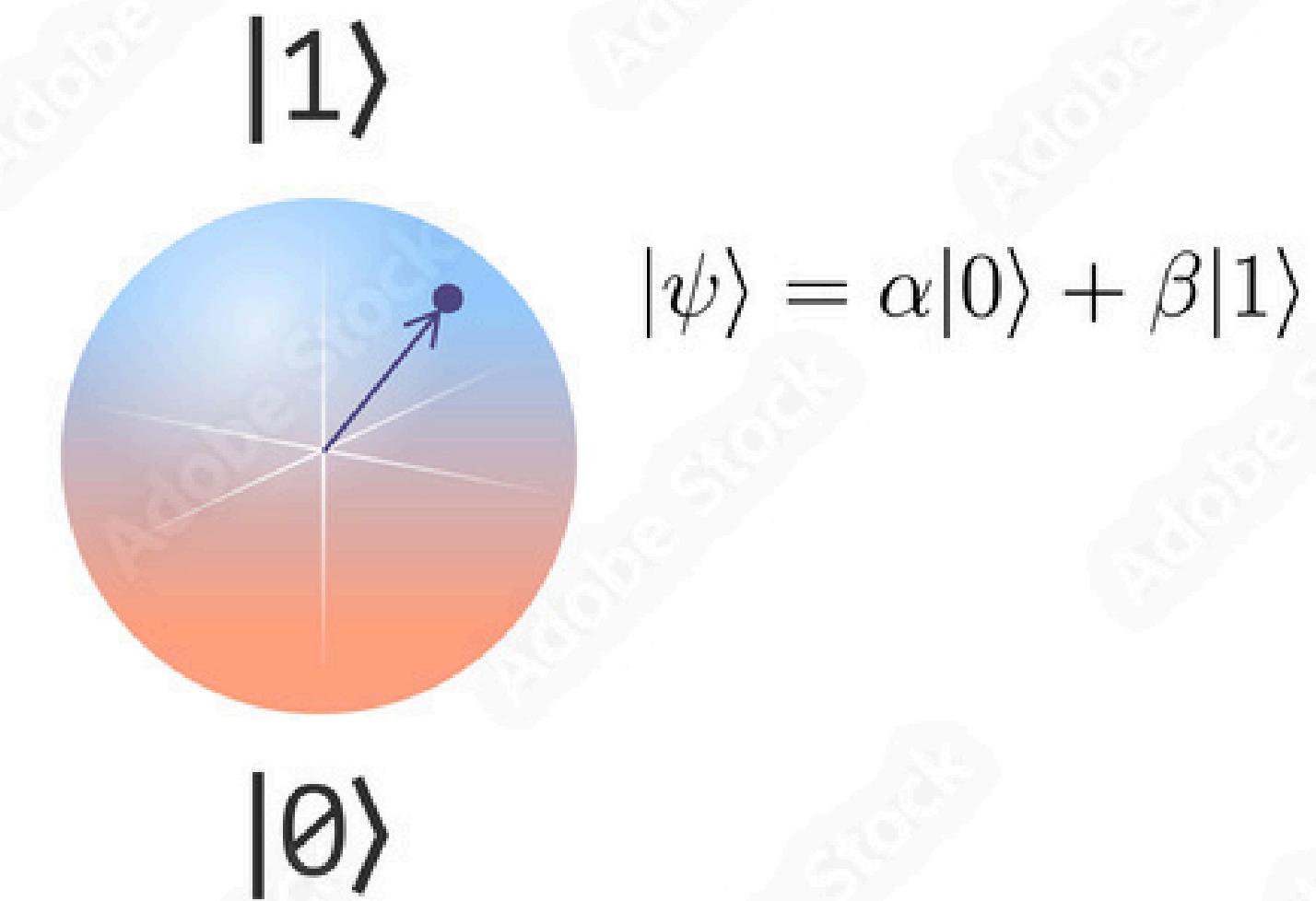
Tl;dr

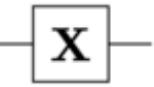
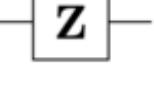
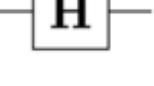
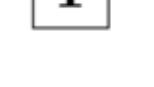
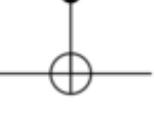
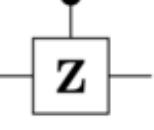
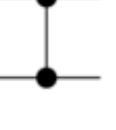
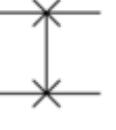
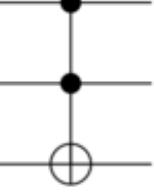
- Zamiast bitów pracujemy na qubitach.
- Operacje wykonywane na specjalnych komputerach przy użyciu specjalnych algorytmów
- Informatyka kwantowa daje nam NIEOGRANICZONE możliwości

Bit



Qubit

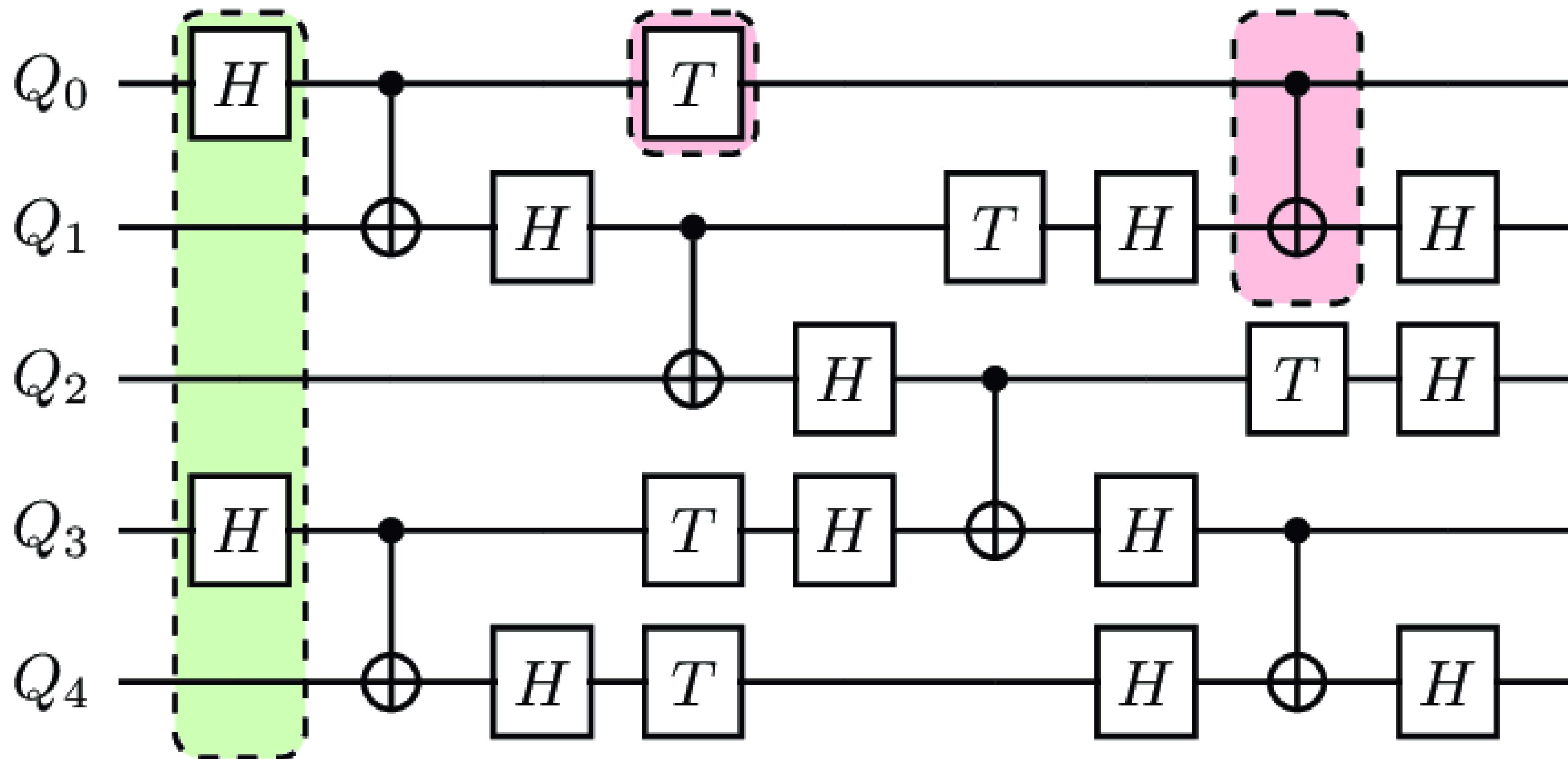


Operator	Gate(s)	Matrix
Pauli-X (X)	 	$\begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$
Pauli-Y (Y)		$\begin{bmatrix} 0 & -i \\ i & 0 \end{bmatrix}$
Pauli-Z (Z)		$\begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}$
Hadamard (H)		$\frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$
Phase (S, P)		$\begin{bmatrix} 1 & 0 \\ 0 & i \end{bmatrix}$
$\pi/8$ (T)		$\begin{bmatrix} 1 & 0 \\ 0 & e^{i\pi/4} \end{bmatrix}$
Controlled Not (CNOT, CX)		$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}$
Controlled Z (CZ)	 	$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & -1 \end{bmatrix}$
SWAP	 	$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
Toffoli (CCNOT, CCX, TOFF)		$\begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}$

LAYER

1-QUBIT GATE

2-QUBIT GATE



Qubits

Memory

Machine

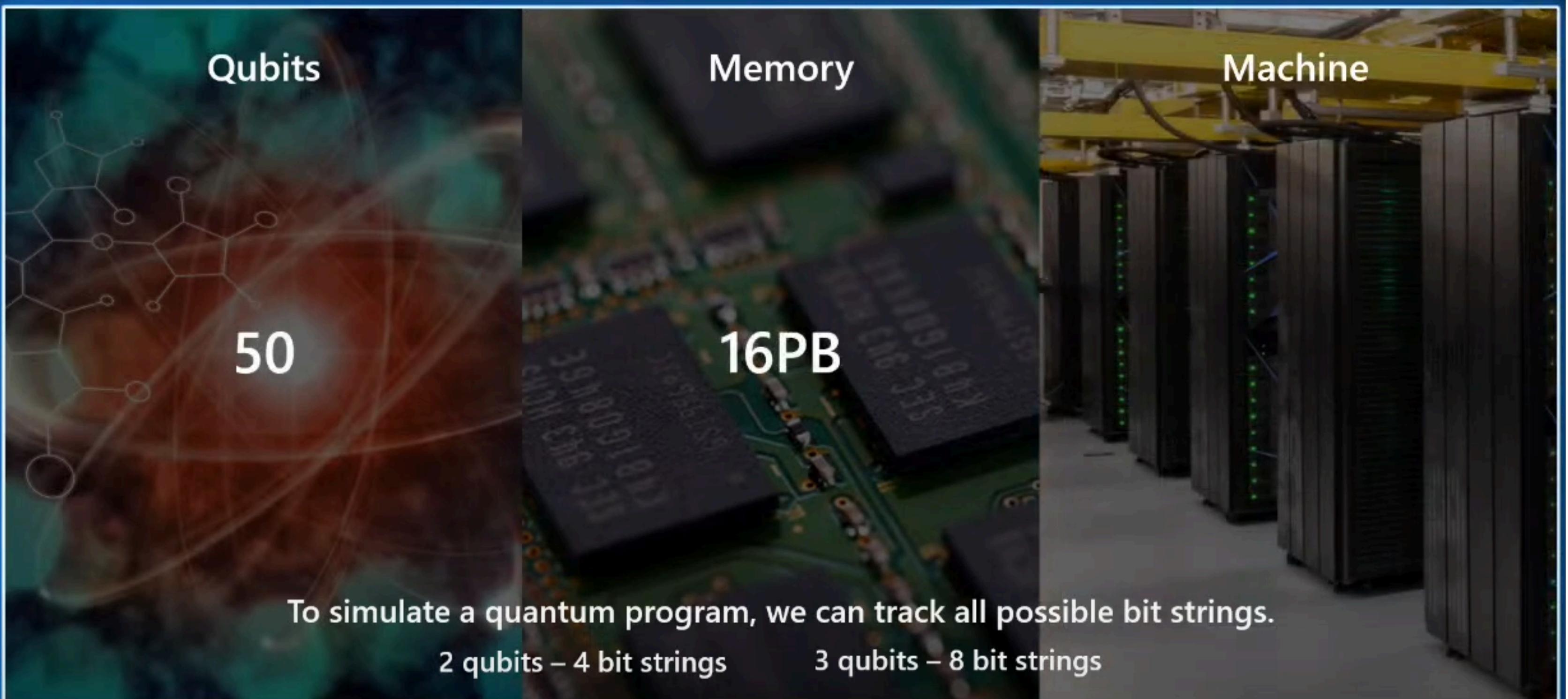
30

16GB

To simulate a quantum program, we can track all possible bit strings.

2 qubits – 4 bit strings 3 qubits – 8 bit strings

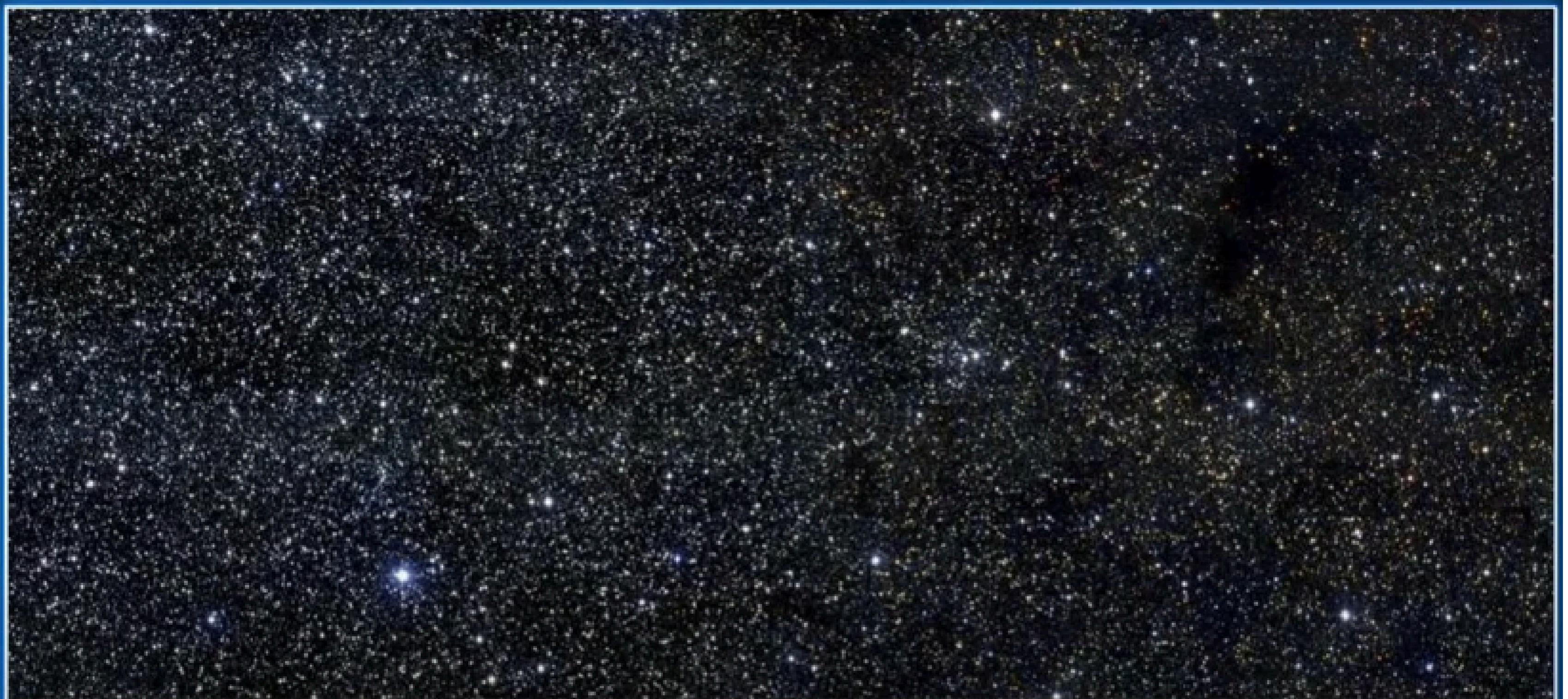
Simulating quantum computers on classical computers (2^N)



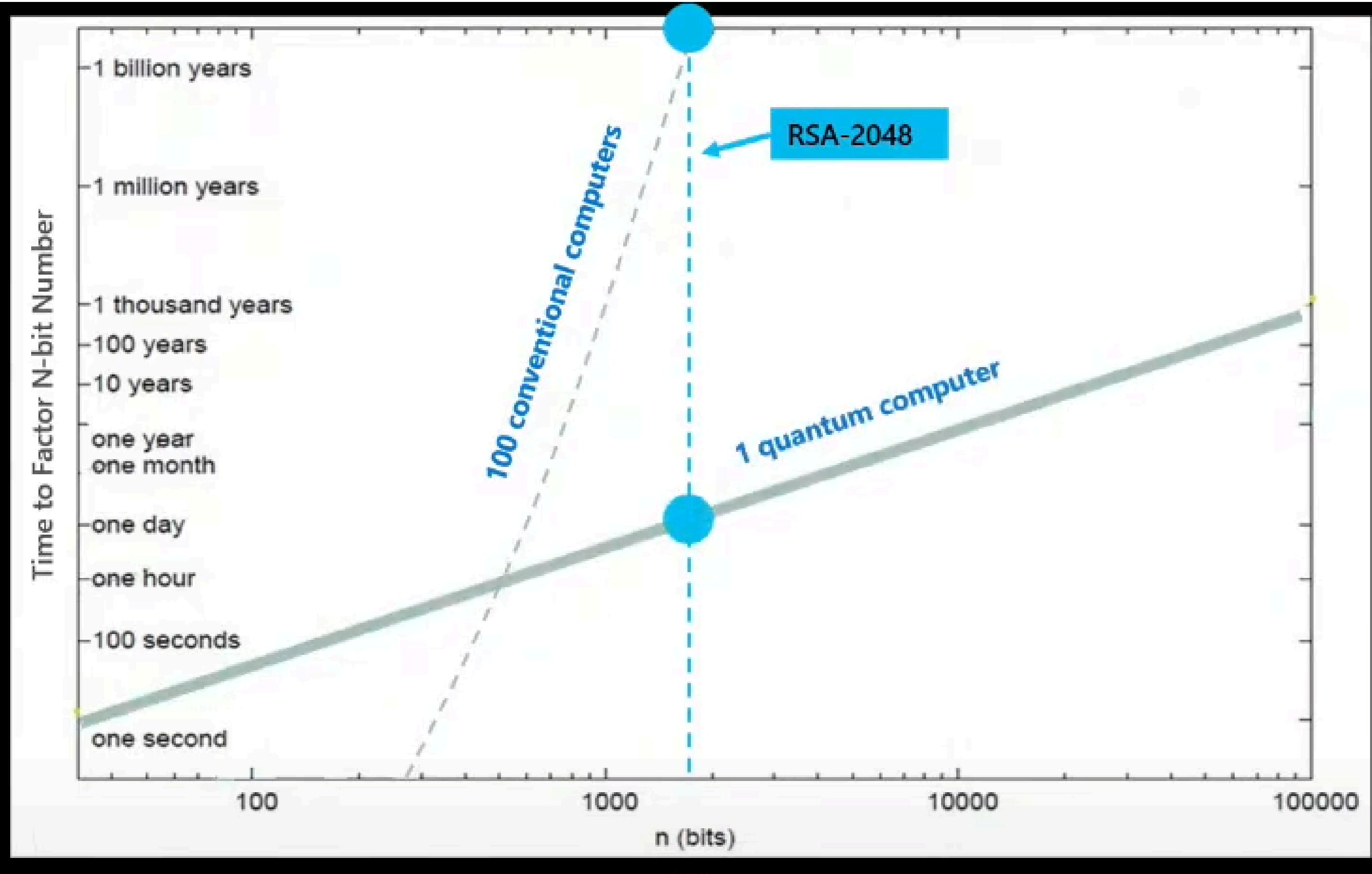
N

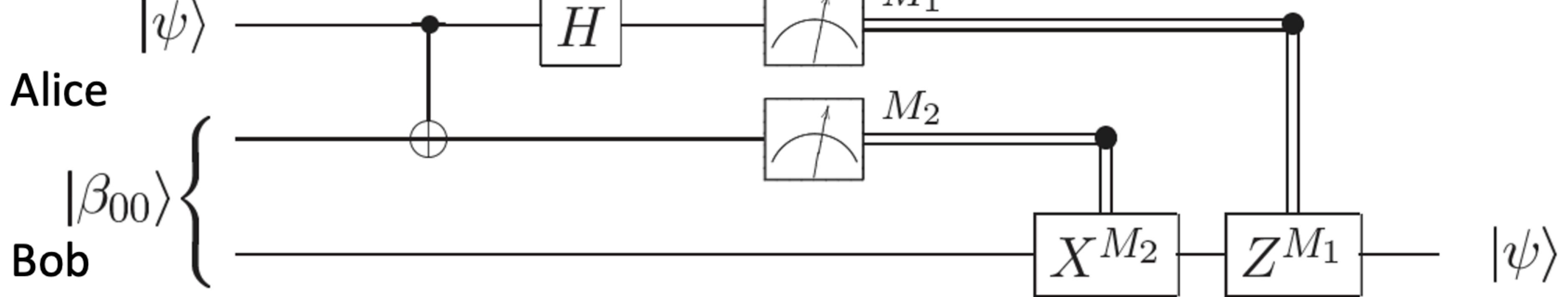
Simulating quantum computers on classical computers (2^N)

250 qubits...



250 qubits ... more states than atoms in the visible universe





Quantum Networks

Systemy komunikacyjne wykorzystujące zasady mechaniki kwantowej do przesyłania i przetwarzania informacji. Umożliwiają one przesyłanie danych za pomocą kubitów, które mogą znajdować się w stanie superpozycji, co zwiększa pojemność i szybkość przetwarzania informacji. Kluczową cechą tych sieci jest splątanie kwantowe, który pozwala na natychmiastowe przekazywanie informacji między splątanymi kubitami, niezależnie od odległości.

Protokoły w QN



BB84 Protocol

To jeden z pierwszych i najważniejszych protokołów do Quantum Key Distribution (QKD).



E91 Protocol

Używany do QKD. Wykorzystuje splątanie kwantowe do zapewnienia bezpiecznej wymiany kluczy.

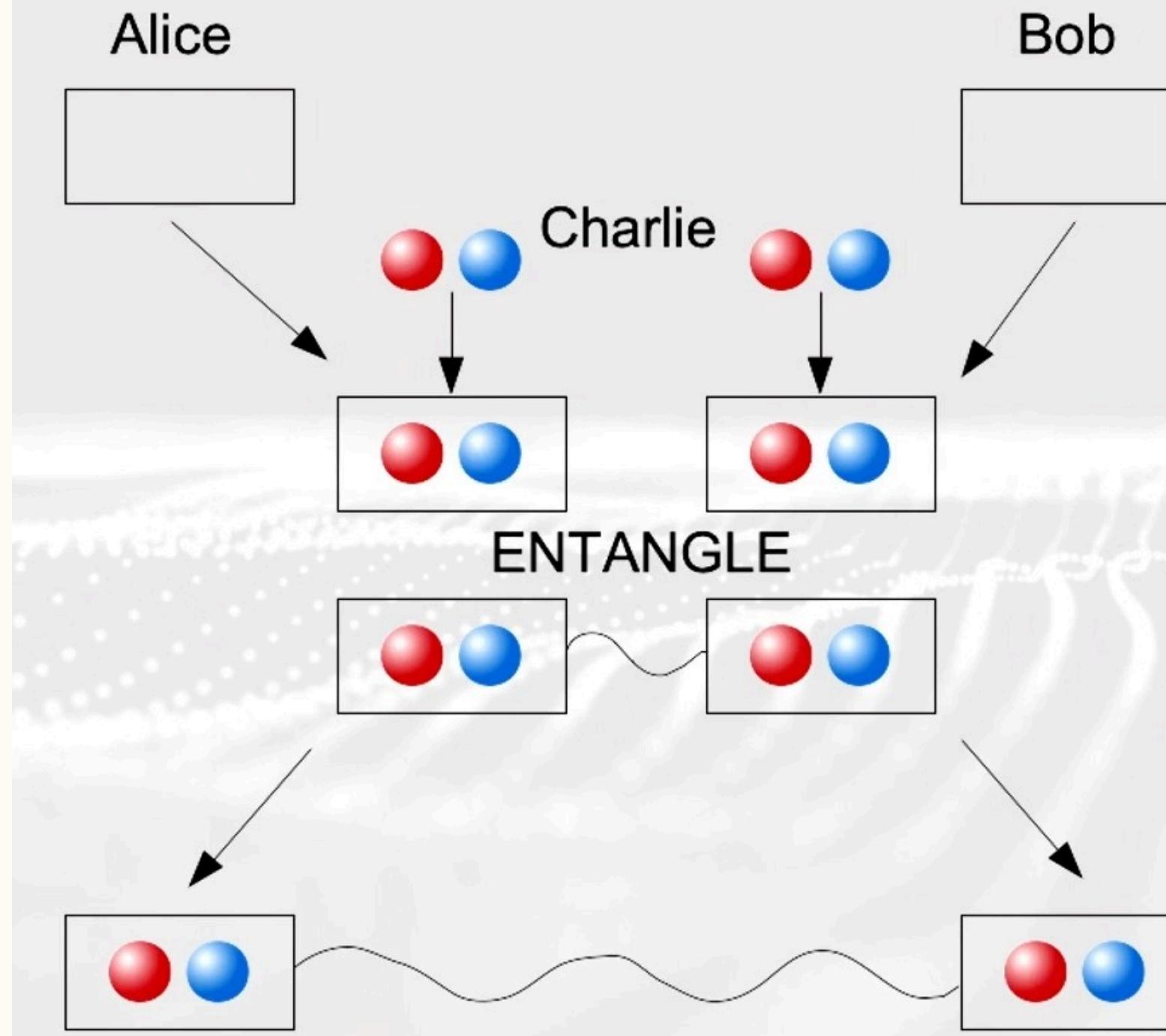


Quantum Teleportation Protocol

Pozwala na przesyłanie stanu kwantowego z jednego kubitu na drugi, wykorzystując splątanie i klasyczną komunikację.

What about more qubits than one??

Let me introduce you to entanglement ... it's getting spooky now ...



Alice, Bob and Charlie meet in Berlin:

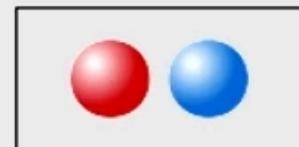
1. Alice comes from Madrid and brings a backpack
2. Bob comes from Vienna brings a backpack
3. They give their backpacks to Charlie
4. Charlie puts each a blue and red ball
5. Charlie entangles the backpacks
6. Alice and Bob go back home

Note, there is ***NO*** physical connection between Alice and Bob anymore

What about more qubits than one??

Let me introduce you to entanglement ... it's getting spooky now ...

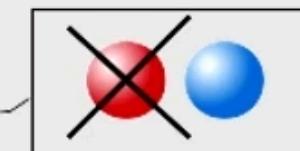
Alice



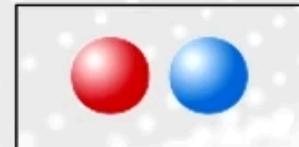
„I've got a red ball“



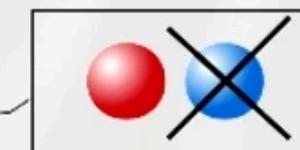
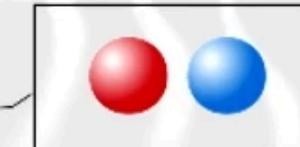
Bob



The red ball disappears!
It's like magic ...



„I've got a blue ball“



The blue ball disappears!
It's like magic ...

Entanglement is about correlation between systems!

Einstein: "Spooky action at a distance ..."

```
 1 from qiskit import *
 2 from qiskit_aer import *
 3
 4 # Creating registers
 5
 6 q = QuantumRegister( size: 3, name: 'q' )
 7 c0 = ClassicalRegister( size: 1, name: 'c0' ) #Alice
 8 c1 = ClassicalRegister( size: 1, name: 'c1' )
 9 c2 = ClassicalRegister( size: 1, name: 'c2' ) #Bob
10
11 # Creates the quantum circuit
12 teleport = QuantumCircuit( *regs: q, c0,c1,c2)
13 print(teleport)
14
15 # Make the shared entangled state
16 teleport.h(q[1])
17 # Bob checks the state of the teleported qubit
18 teleport.measure(q[2], c2[0])
19 print(teleport)
20 # Shows gates of the circuit
21 simulator = Aer.get_backend('qasm_simulator')
22
23
24 # Transpile and assemble the circuit
25 transpiled_circuit = transpile(teleport, simulator)
26 qobj = assemble(transpiled_circuit, shots=1024)
27
28 # Run the algorithm
29 result = simulator.run(qobj).result()
30
31
32 counts = result.get_counts(teleport)
33 print('\nThe measured outcomes of the circuits are:', counts)
34
```

Materiały źródłowe

- Podaj materiały źródłowe
- Podaj materiały źródłowe
- Podaj materiały źródłowe
- Podaj materiały źródłowe

Sesja pytań i odpowiedzi

Dziękuję za uwagę!