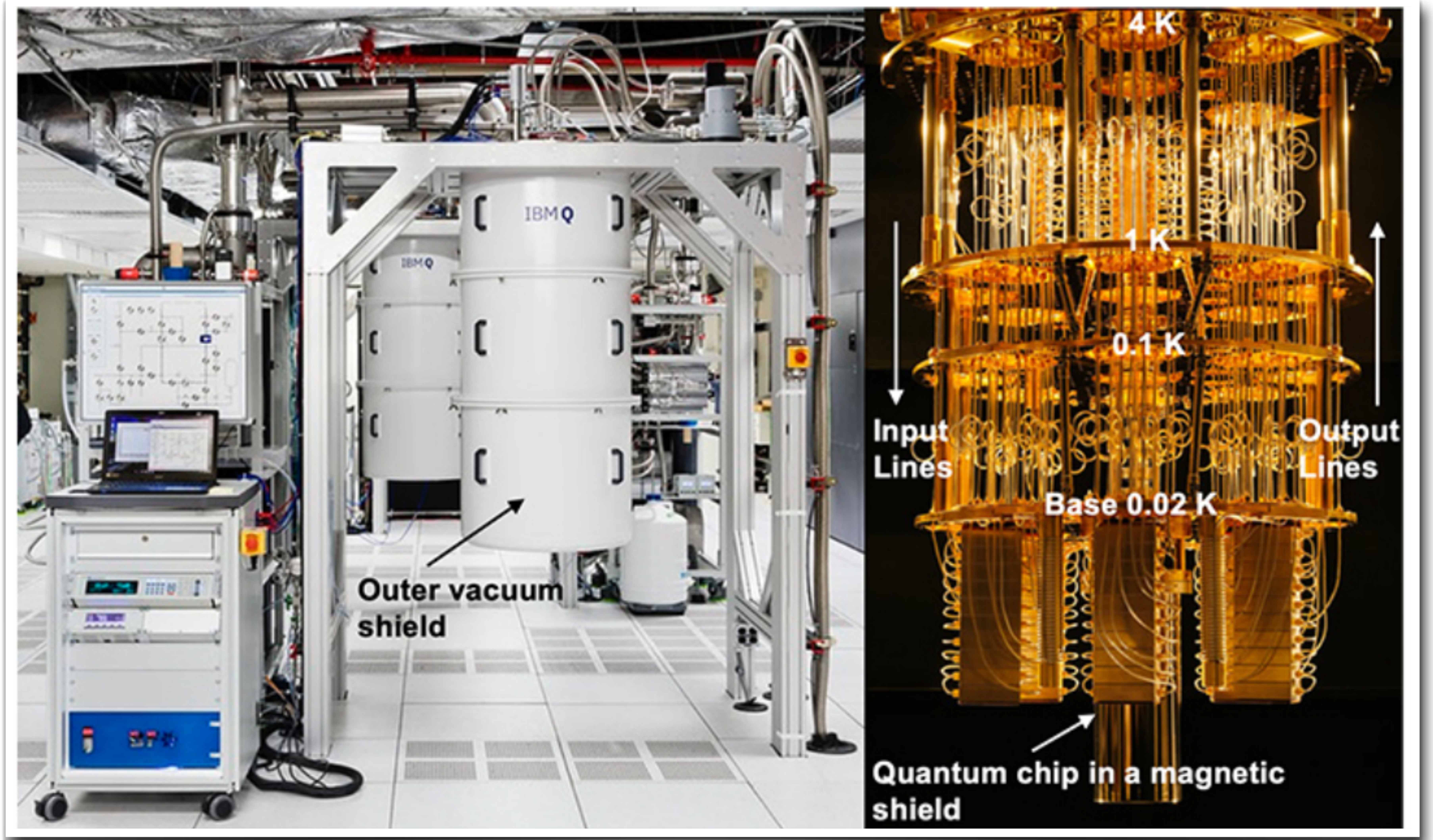
Developing Quantum Computing Circuits

Optics and Spectroscopy
Sofia University

Vasil Yordanov and Lachezar Georgiev - 31 July 2021

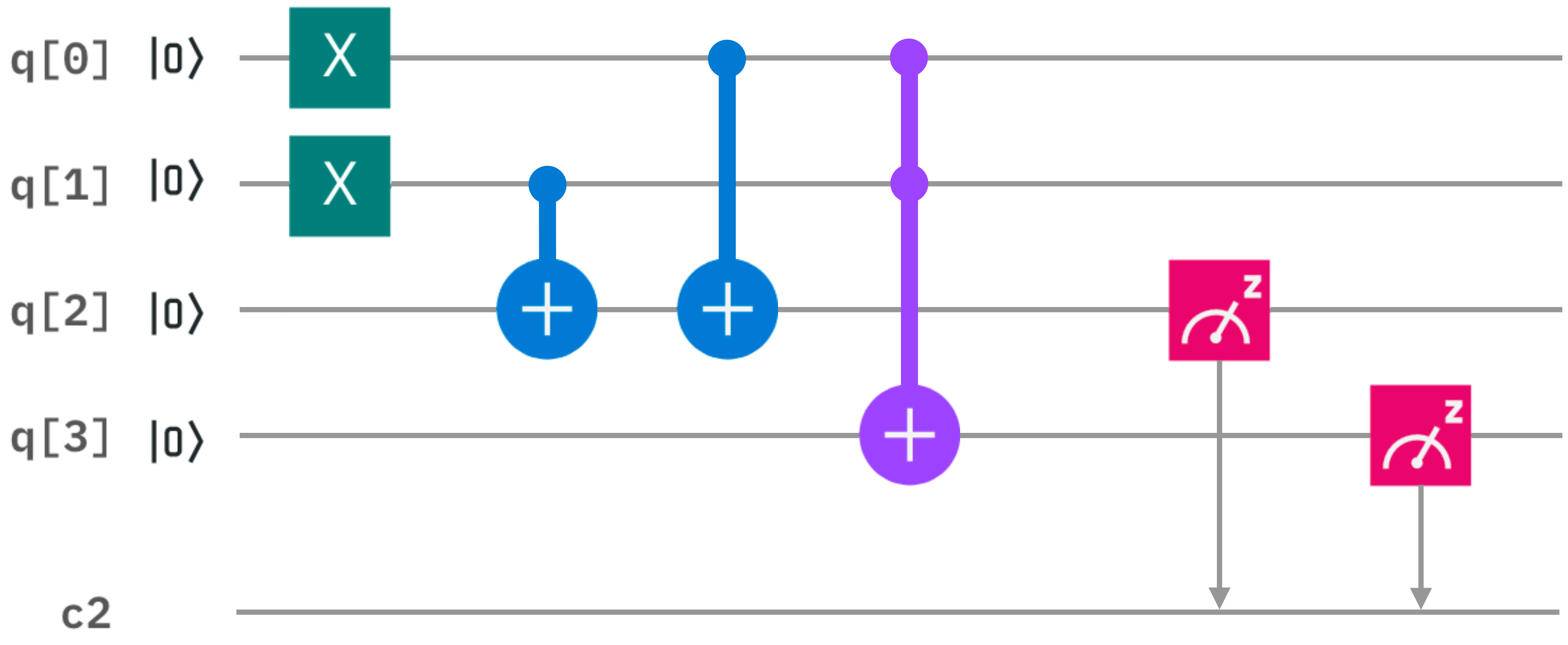
IBM's Quantum Computer



DiVincenzo's criteria

Constructing a quantum computer requires that the experimental setup meet he following conditions

1. A scalable physical system with well characterised qubit
2. The ability to initialise the state of the qubits to a simple fiducial state, such as $|000...\rangle$
3. Long relevant decoherence times, much longer than the gate operation time
4. A "universal" set of quantum gates
5. A qubit-specific measurement capability



Qiskit

- Qiskit is an open-source SDK for working with quantum computers at the level of pulses, circuits and application modules
- Designed to work with IBM's quantum processors
- Qiskit webpage - https://qiskit.org

Setup Qiskit

1. Install Anaconda Python - https://www.anaconda.com/products/individual

2. Start Anaconda Prompt

3. Install qiskit by typing "pip install qiskit”

4. Start Jupyter notebook by typing “jupyter notebook”

Demo

Get Access to Quantum Computers

1. Get access token from IBM Quantum Experience - https://quantum-computing.ibm.com
2. Setup Access token in Qiskit

Demo Source

Single Qubit Quantum state

Can be written as: $\left|\psi\right\rangle = \alpha\left|0\right\rangle + \beta\left|1\right\rangle \quad \alpha, \beta \in \mathbb{C}$

Probability of the bit being in $\left|0\right\rangle$ is $|\alpha|^2$ and $\left|1\right\rangle$ is $|\beta|^2 \Rightarrow |\alpha|^2 + |\beta|^2 = 1$

Vector notation of $\left|\psi\right\rangle = \binom{\alpha}{\beta} \Rightarrow \left|0\right\rangle = \begin{pmatrix}1\\0\end{pmatrix}, \quad \left|1\right\rangle = \begin{pmatrix}0\\1\end{pmatrix}$

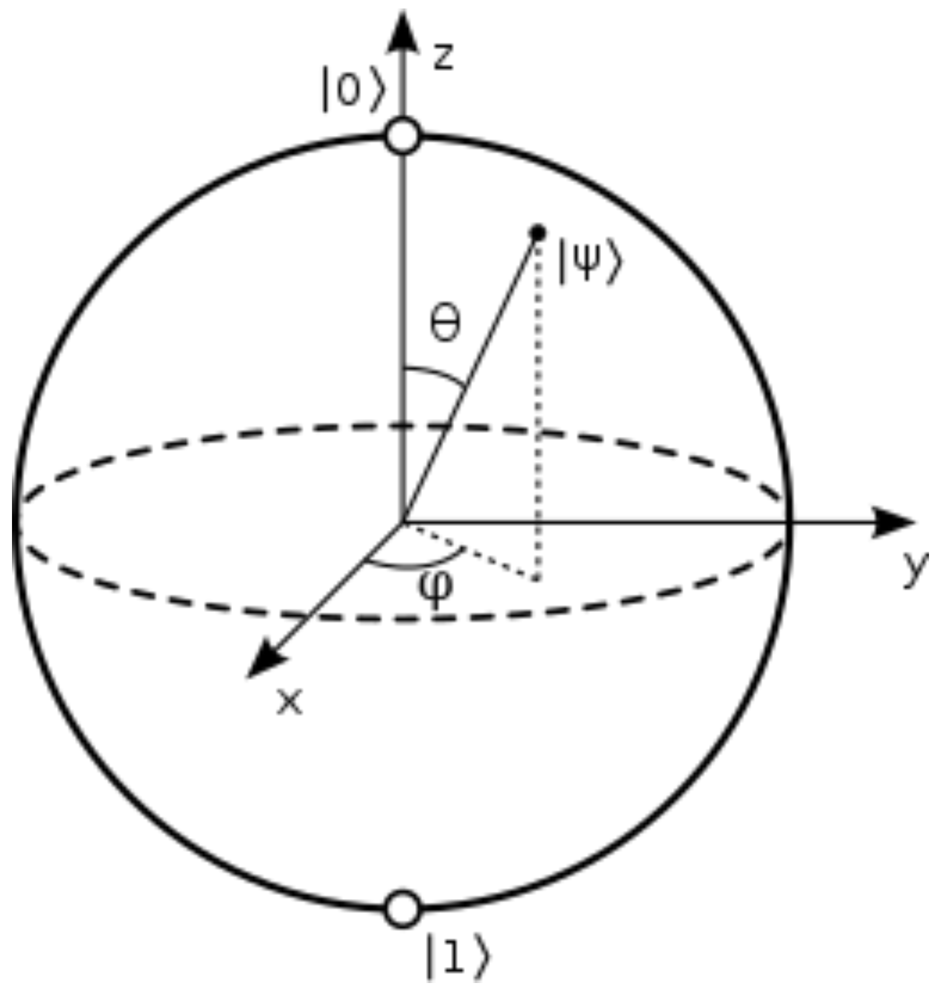
From $|\alpha|^2 + |\beta|^2 = 1$ and since global phase is undetectable $\left|\psi\right\rangle := e^{i\delta}\left|\psi\right\rangle$

$$\left|\psi\right\rangle = e^{i\delta}\left(\cos(\theta/2)\left|0\right\rangle + \sin(\theta/2)e^{i(\phi-\delta)}\left|1\right\rangle\right)$$

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

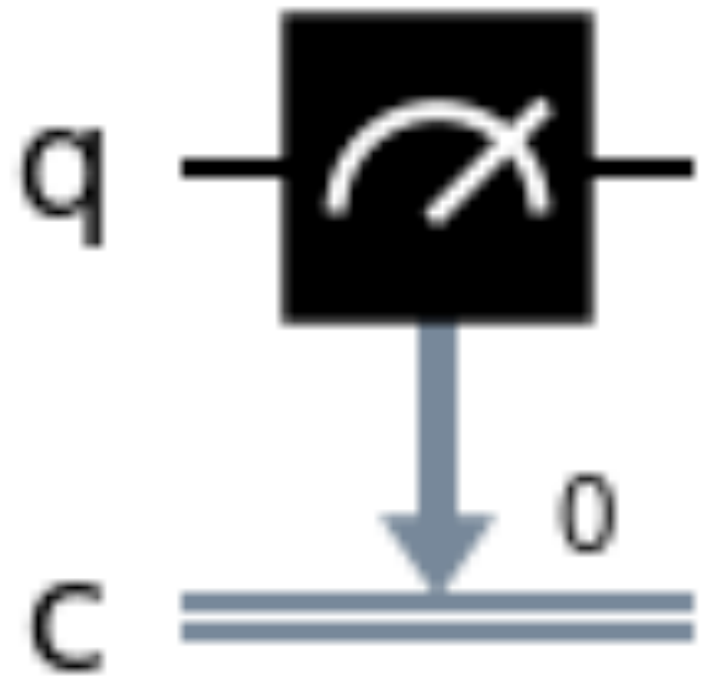
$$0 \leq \phi < 2\pi, \quad 0 \leq \theta \leq \pi$$

- Bloch sphere representation of qubit state



Measurement operator

- Make once time measurement on a qubit
- Records the result in classical bit

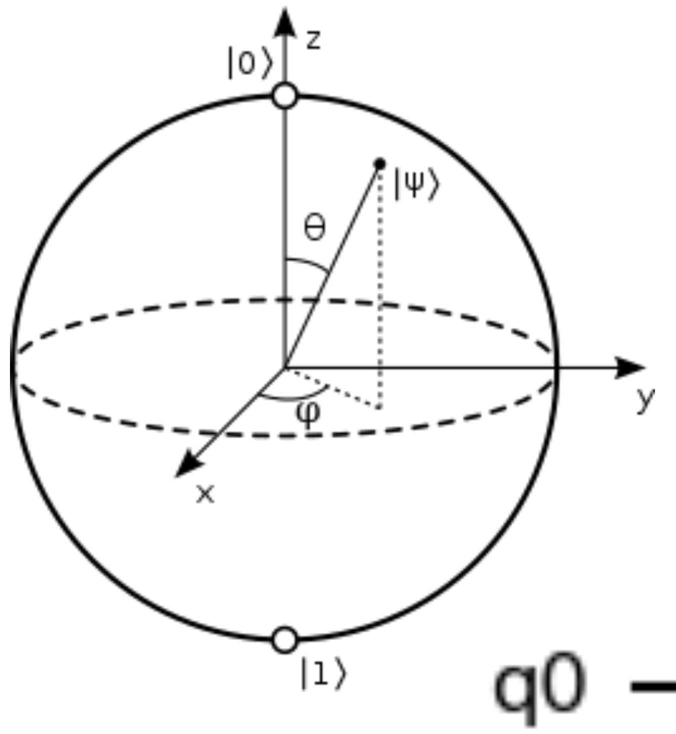


Running Hello World

1. Writing ‘Hello World’ program
2. Visualising the program
3. Running ‘Hello World’ program on Simulator
4. Running ‘Hello World’ program on Quantum Computer

Demo Source

Initialise qubit state

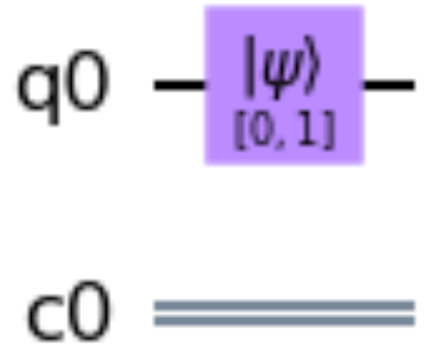


- In Qiskit we can specify the initial state of a qubit
- Let's remind the Bloch representation of a qubit that will be used in the examples

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

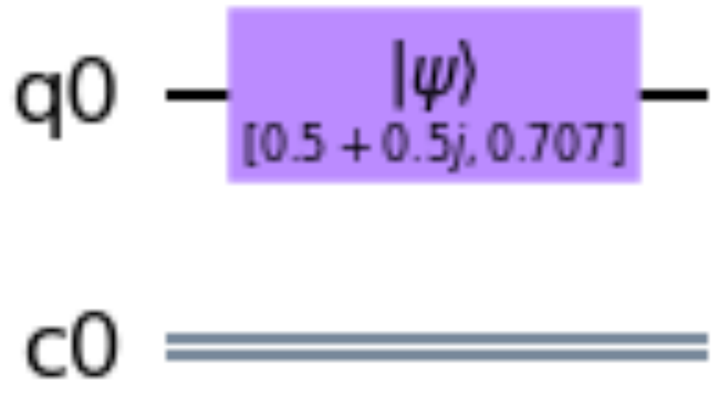
- **Example 1**

Initialise qubit in state $\left|1\right\rangle = 0\left|0\right\rangle + 1\left|1\right\rangle \Rightarrow \alpha = 0, \quad \beta = 1 \Rightarrow \quad \theta = \pi, \quad 0 \leq \phi < 2\pi$



- **Example 2**

Initialise qubit in state $\left|\psi\right\rangle = \left(\frac{1}{2} + \frac{1}{2}i\right)\left|0\right\rangle + \frac{\sqrt{2}}{2}\left|1\right\rangle = e^{i\frac{\pi}{4}}\left(\frac{\sqrt{2}}{2}\left|0\right\rangle + \frac{\sqrt{2}}{2}e^{-i\frac{\pi}{4}}\left|1\right\rangle\right) \Rightarrow \theta = \frac{\pi}{2}, \quad \phi = -\frac{\pi}{4}$



Demo Source

Quantum gates/operations

- A gate which acts on a qubit is represented by a 2×2 unitary matrix U: $|\psi'\rangle = U|\psi\rangle$
- A general unitary must be able to take the $|0\rangle$ to the state
  $|\psi\rangle = \cos(\theta/2)|0\rangle + \sin(\theta/2)e^{i\phi}|1\rangle = U|0\rangle$
  $\Rightarrow U = \begin{pmatrix} \cos(\theta/2) & a \\ e^{i\phi}\sin(\theta/2) & b \end{pmatrix}$, where a and b are complex numbers such that $U^\dagger U = I$
- The general expression of 2×2 unitary matrix, which depends on 4 real parameters, is:

$$U = \begin{pmatrix} \alpha & -e^{-i\varphi}\beta* \\ \beta & e^{-i\varphi}\alpha* \end{pmatrix}, \qquad |\alpha|^2 + |\beta|^2 = 1$$

- This is the most general form of single quit unitary, if we substitute above $\varphi = -\lambda - \phi$

$$U = \begin{pmatrix} \cos(\theta/2) & -e^{i\lambda}\sin(\theta/2) \\ e^{i\phi}\sin(\theta/2) & e^{i\lambda+i\phi}\cos(\theta/2) \end{pmatrix}, \quad 0 \le \phi < 2\pi, \quad 0 \le \theta \le \pi, \quad 0 \le \lambda < 2\pi$$

Why quantum operations are represented as unitary matrix?

- Schrödinger equation

$$\frac{d}{dt}\left|\psi(t)\right\rangle = -\frac{i}{\hbar}\hat{H}(t)\left|\psi(t)\right\rangle,$$

where Hamiltonian $\hat{H}(t)$ is Hermitian operator $\hat{H}(t) = \hat{H}^{\dagger}(t)$

- Suppose $\left|\psi(t)\right\rangle = \hat{U}(t)\left|\psi(0)\right\rangle$ for some matrix $\hat{U}(t)$ which is not unitary yet

$$\frac{d\hat{U}(t)}{dt} = -\frac{i}{\hbar}\hat{H}(t)\hat{U}(t), \quad \frac{d\hat{U}^{\dagger}(t)}{dt} = -\frac{i}{\hbar}\hat{H}(t)\hat{U}(t)$$

At $t = 0, \quad \hat{U}(0) = \hat{I}$, so $\hat{U}^{\dagger}(0)\hat{U}(0) = \hat{I}$

$$\frac{d}{dt}\left(\hat{U}^{\dagger}(t)\hat{U}(t)\right) = \frac{1}{\hbar}\hat{U}^{\dagger}(t)\left(i\hat{H}(t) - i\hat{H}(t)\right)\hat{U}(t) = 0$$

- So $\hat{U}^{\dagger}(t)\hat{U}(t) = \hat{I}$ at all times t, and $\hat{U}(t)$ must always be unitary.

Single-Qubit Gates

- u gates
  - U gate $u3(\theta, \phi, \lambda) = u(\theta, \phi, \lambda)$
  - P gate $p(\lambda) = \begin{pmatrix} 1 & 0 \\ 0 & e^{i\lambda} \end{pmatrix} = u3(0,0,\lambda)$
- Identity gate $Id = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} = u3(0,0,0)$
- Pauli gates
  - $\boxed{X : \text{bit-flip gate } X = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} = u3(\pi,0,\pi)}$
  - Y : bit- and phase-flip gate $Y = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix} = u3(\pi, \pi/2,\pi/2)$
  - Z : phase-flip gate $Z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} = u3(0,0,\pi)$
- Clifford gates
  - Hadamard gate $\boxed{H = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} = u3(0,0,\pi)}$
  - S gate $S = \begin{pmatrix} 1 & 0 \\ 0 & i \end{pmatrix} = u3(0,0,\pi/2)$
  - $S^\dagger$ gate $S^\dagger = \begin{pmatrix} 1 & 0 \\ 0 & -i \end{pmatrix} = u3(0,0,-\pi/2)$
- $C3$ gate
  - T gate $T = \begin{pmatrix} 1 & 0 \\ 0 & e^{i\pi/4} \end{pmatrix} = u3(0,0,\pi/4)$
  - $T^\dagger$ gate $T^\dagger = \begin{pmatrix} 1 & 0 \\ 0 & e^{-i\pi/4} \end{pmatrix} = u3(0,0,-\pi/4)$
- Standart Rotations
  - Rotation around X-axis

$$R_x(\theta) = \begin{pmatrix} \cos(\theta/2) & -i\sin(\theta/2) \\ -i\sin(\theta/2) & \cos(\theta/2) \end{pmatrix} = u3(\theta, -\pi/2,\pi/2)$$

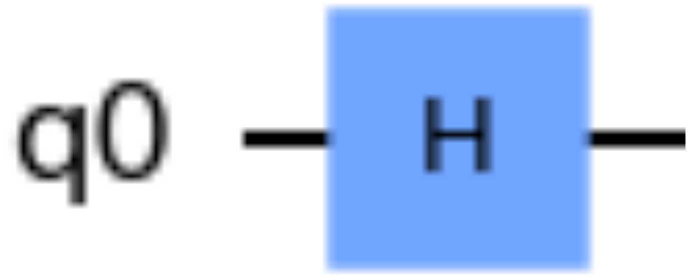
  - Rotation around Y-axis

$$R_y(\theta) = \begin{pmatrix} \cos(\theta/2) & -\sin(\theta/2) \\ \sin(\theta/2) & \cos(\theta/2) \end{pmatrix} = u3(\theta,0,0)$$

  - Rotation around Z-axis

$$R_z(\phi) = \begin{pmatrix} e^{-i\phi/2} & 0 \\ 0 & e^{i\phi/2} \end{pmatrix} \equiv u3(0,\phi,0)$$

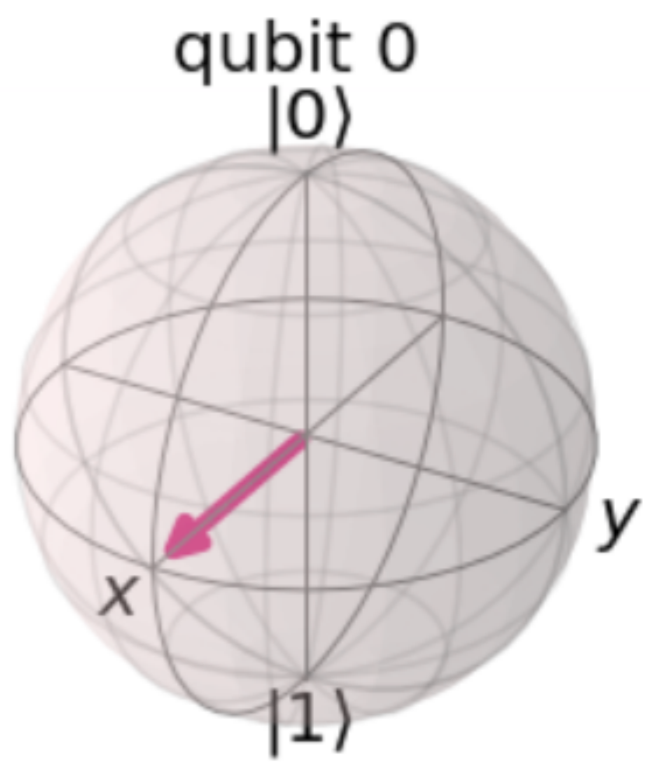
Hadamard Gate

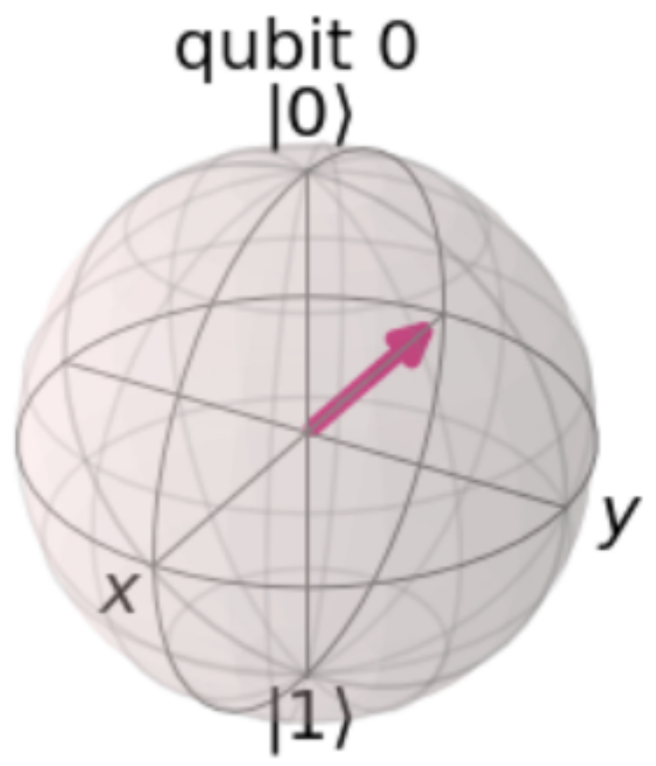


- Considering $|0\rangle = \begin{pmatrix} 1 \\ 0 \end{pmatrix}$, $|1\rangle = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$, and $H = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}$
- Let's apply H-gate on state vector $|0\rangle$:

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

$$\Rightarrow \alpha = \frac{1}{\sqrt{2}}, \quad \beta = \frac{1}{\sqrt{2}} \quad \Rightarrow |\alpha|^2 = 0.5, \quad |\beta|^2 = 0.5$$

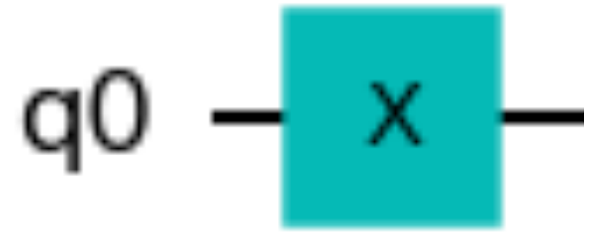




Demo Source

X-Gate

- Considering $|0\rangle = \begin{pmatrix} 1 \\ 0 \end{pmatrix}$, $|1\rangle = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$, and $X = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$

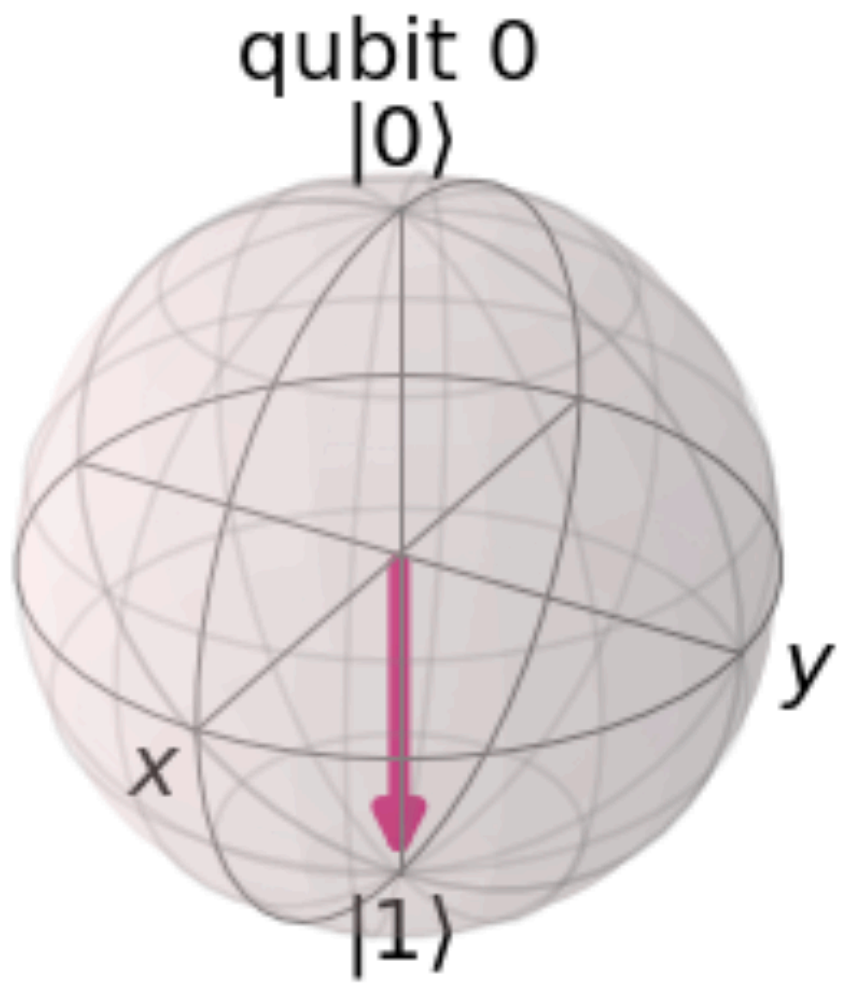


- Let's apply X-gate on state vector $|0\rangle$:

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

$$\Rightarrow \alpha = 0, \quad \beta = 1$$

$$\Rightarrow |\alpha|^2 = 0, \quad |\beta|^2 = 1$$



Demo Source

Multi and Two-qubit system

- For n qubits the complex vector space has dimensions $d = 2^n$ - grows exponentially with the number of qubits
- The basis vectors for the 2-qubit system are formed using the tensor product of basis vectors for a single qubit:

$$\left|0\right\rangle \otimes \left|0\right\rangle = \left|00\right\rangle = \begin{pmatrix} 1\begin{pmatrix}1\\0\end{pmatrix} \\ 0\begin{pmatrix}1\\0\end{pmatrix} \end{pmatrix} = \begin{pmatrix}1\\0\\0\\0\end{pmatrix} \quad \left|0\right\rangle \otimes \left|1\right\rangle = \left|01\right\rangle = \begin{pmatrix} 1\begin{pmatrix}0\\1\end{pmatrix} \\ 0\begin{pmatrix}0\\1\end{pmatrix} \end{pmatrix} = \begin{pmatrix}0\\1\\0\\0\end{pmatrix}$$

$$\left|1\right\rangle \otimes \left|0\right\rangle = \left|10\right\rangle = \begin{pmatrix} 0\begin{pmatrix}1\\0\end{pmatrix} \\ 1\begin{pmatrix}1\\0\end{pmatrix} \end{pmatrix} = \begin{pmatrix}0\\0\\1\\0\end{pmatrix} \quad \left|1\right\rangle \otimes \left|1\right\rangle = \left|11\right\rangle = \begin{pmatrix} 0\begin{pmatrix}0\\1\end{pmatrix} \\ 1\begin{pmatrix}0\\1\end{pmatrix} \end{pmatrix} = \begin{pmatrix}0\\0\\0\\1\end{pmatrix}$$

$$\left|\psi\right\rangle = a_{00}\left|00\right\rangle + a_{01}\left|01\right\rangle + a_{10}\left|10\right\rangle + a_{11}\left|11\right\rangle = \begin{bmatrix}a_{00}\\a_{01}\\a_{10}\\a_{11}\end{bmatrix}, \ |a_{00}|^2 + |a_{01}|^2 + |a_{10}|^2 + |a_{11}|^2 = 1$$

Demo Source

Two-qubit Gates - General Form

- Most of the two-qubit gates are of the controlled type (except for the SWAP gate)
- Controlled type two-qubit gates applies U to one qubit, conditioned on the state of another qubit
- Lets apply on one qubit 1 the gate U on condition qubit 0 is equal to $|1\rangle$, where $U = \begin{pmatrix} u_{00} & u_{01} \\ u_{10} & u_{11} \end{pmatrix}$

$$C_U: \underbrace{|0\rangle}_{\text{qubit } 1} \otimes \underbrace{|0\rangle}_{\text{qubit } 0} \rightarrow \underbrace{|0\rangle}_{\text{qubit } 1} \otimes \underbrace{|0\rangle}_{\text{qubit } 0}$$

$$C_U: \underbrace{|0\rangle}_{\text{qubit } 1} \otimes \underbrace{|1\rangle}_{\text{qubit } 0} \rightarrow \underbrace{U|0\rangle}_{\text{qubit } 1} \otimes \underbrace{|1\rangle}_{\text{qubit } 0}$$

$$C_U: \underbrace{|1\rangle}_{\text{qubit } 1} \otimes \underbrace{|0\rangle}_{\text{qubit } 0} \rightarrow \underbrace{|1\rangle}_{\text{qubit } 1} \otimes \underbrace{|0\rangle}_{\text{qubit } 0}$$

$$C_U: \underbrace{|1\rangle}_{\text{qubit } 1} \otimes \underbrace{|1\rangle}_{\text{qubit } 0} \rightarrow \underbrace{U|1\rangle}_{\text{qubit } 1} \otimes \underbrace{|1\rangle}_{\text{qubit } 0}$$

LSB MSB

when Controlled bit is MSB: $C_U = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & u_{00} & 0 & u_{01} \\ 0 & 0 & 1 & 0 \\ 0 & u_{10} & 0 & u_{11} \end{pmatrix}$

when Controlled bit is LSB: $C_U = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & u_{00} & u_{01} \\ 0 & 0 & u_{10} & u_{11} \end{pmatrix}$

Two-Qubit Gates

- ***Controlled Pauli gates***
  - ***Controlled-X*** - flips the target qubit if the control qubit is in the state $|1\rangle$.

    Controlled-bit MSB: $\boxed{C_X = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}, \text{ LSB: } C_X = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix}}$

  - ***Controlled-Y*** gate

    Controlled-bit MSB: $C_Y = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & -i \\ 0 & 0 & i & 0 \end{pmatrix}$, LSB: $C_Y = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & -i \\ 0 & 0 & 1 & 0 \\ 0 & i & 0 & 0 \end{pmatrix}$

  - ***Controlled-Z*** - flips the phase of the target qubit if the control qubit is in the state $|1\rangle$. Controlled-bit MSB or LSB: $C_Z = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & -1 \end{pmatrix}$

- ***Hadamard gate*** - LSB: $C_H = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & \frac{1}{\sqrt{2}} & 0 & \frac{1}{\sqrt{2}} \\ 0 & 0 & 1 & 0 \\ 0 & \frac{1}{\sqrt{2}} & 0 & -\frac{1}{\sqrt{2}} \end{pmatrix}$

- ***Controlled rotation gates***
  - ***Controlled rotation around Z-axis*** $C_{R_Z}(\lambda) = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & e^{-i\lambda/2} & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & e^{i\lambda/2} \end{pmatrix}$
  - ***Controlled Phase rotation*** - performs a phase rotation if both qubits are in the state $|11\rangle$

    $$\boxed{C_p(\lambda) = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & e^{i\lambda} \end{pmatrix}}$$

  - ***Controlled-U rotation***

    $$C_u(\theta, \phi, \lambda) \equiv \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & e^{-i(\phi+\lambda)/2}\cos(\theta/2) & 0 & -e^{-i(\phi-\lambda)/2}\sin(\theta/2) \\ 0 & 0 & 1 & 0 \\ 0 & e^{i(\phi-\lambda)/2}\sin(\theta/2) & 0 & e^{i(\phi+\lambda)/2}\cos(\theta/2) \end{pmatrix}$$

- ***SWAP gate -*** It transforms the basis vectors as

  $|00\rangle \rightarrow |00\rangle$, $|01\rangle \rightarrow |10\rangle$, $|10\rangle \rightarrow |01\rangle$, $|11\rangle \rightarrow |11\rangle$,

  $$\mathrm{SWAP} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

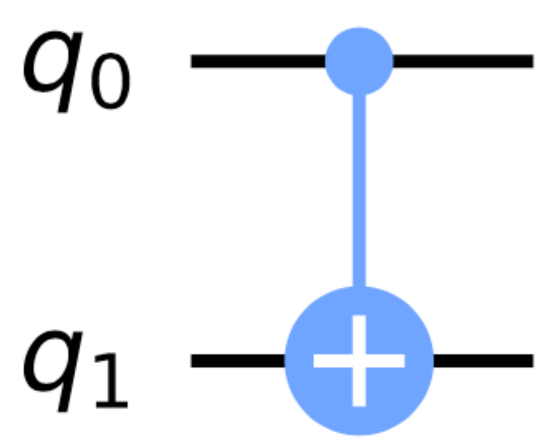
CNOT Gate

- CNOT (Controlled-X) Gate
- When the qubits are not in superposition of $|0\rangle$ and $|1\rangle$, they behave like classical bits and for the CNOT gate the following truth table is valid:
- CNOT gate matrices:

$$\text{CNOT}_{\text{MSB}} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix} \quad \text{CNOT}_{\text{LSB}} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}$$

- The matrix swaps the amplitudes of $|01\rangle$ and $|11\rangle$ vectors

$$|a\rangle = \begin{bmatrix} a_{00} \\ a_{01} \\ a_{10} \\ a_{11} \end{bmatrix}, \quad \text{CNOT}\,|a\rangle = \begin{bmatrix} a_{00} \\ a_{11} \\ a_{10} \\ a_{01} \end{bmatrix} \begin{matrix} \\ \leftarrow \\ \\ \leftarrow \end{matrix}$$

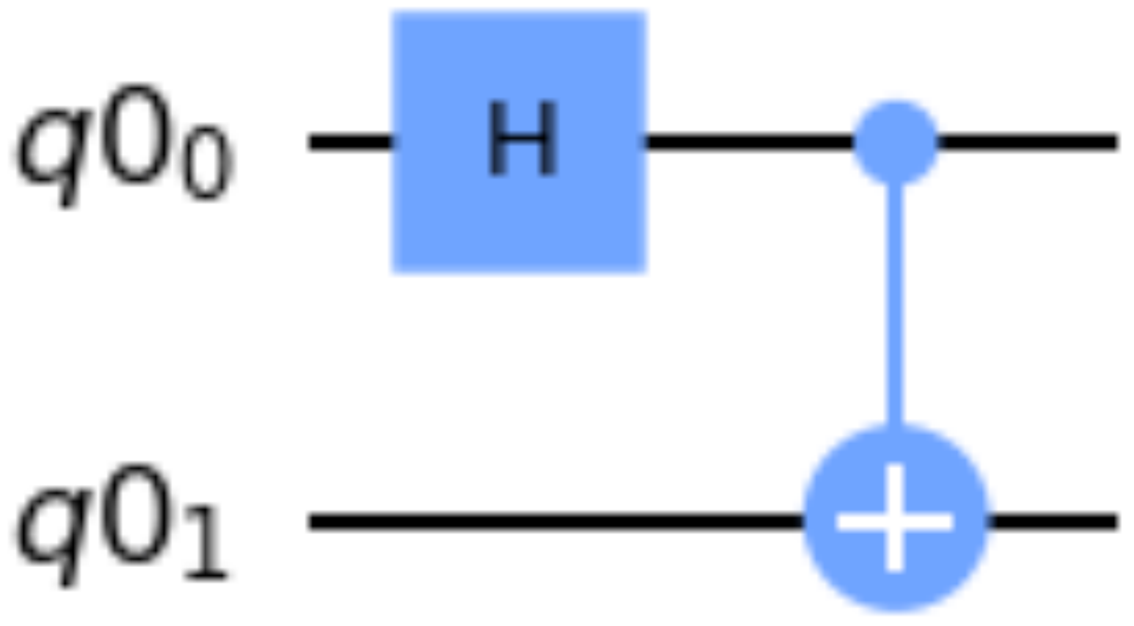


Input	Output
00	00
01	11
10	10
11	01

Demo Source

Entanglement

- Entanglement implements with the following circuit



[Demo]() [Source]()

Quantum Algorithms

- Quantum Fourier Tranformation
- Quantum Phase Estimation
- Shor’s Algorithm
- Variational Quantum Eigensolver

Quantum Fourier Tranform (QFT)

- Quantum Fourier transform acts on a quantum state $|X\rangle = \sum_{j=0}^{N-1} x_j |j\rangle$ and maps it to the quantum state $|Y\rangle = \sum_{k=0}^{N-1} y_k |k\rangle$, according to the formula:

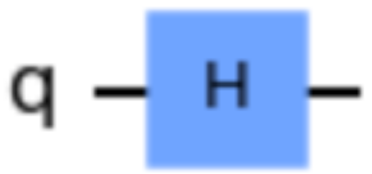
$$y_k = \frac{1}{\sqrt{N}} \sum_{j=0}^{N-1} x_j \omega_N^{jk}, \text{ where } \omega_N^{jk} = e^{2\pi i \frac{jk}{N}}$$

- This can also be expressed as unitary matrix:

$$U_{QFT} = \frac{1}{\sqrt{N}} \sum_{j=0}^{N-1} \sum_{k=0}^{N-1} \omega_N^{jk} |k\rangle\langle j|$$

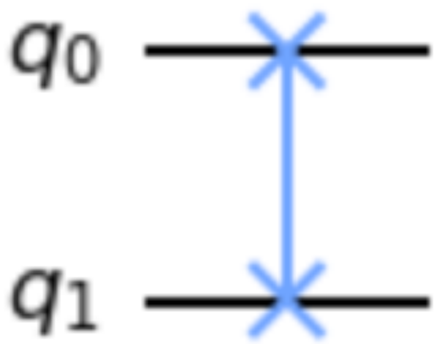
QFT implementation

- Needed operators:
  - H-Gate



  - Controlled Phase Gate

  - SWAP Gate



Controlled Phase Gate in two-qubit system

- Performs a phase rotation if both qubits are in the state $|11\rangle$

$$C_p(\lambda) = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & e^{i\lambda} \end{pmatrix}$$

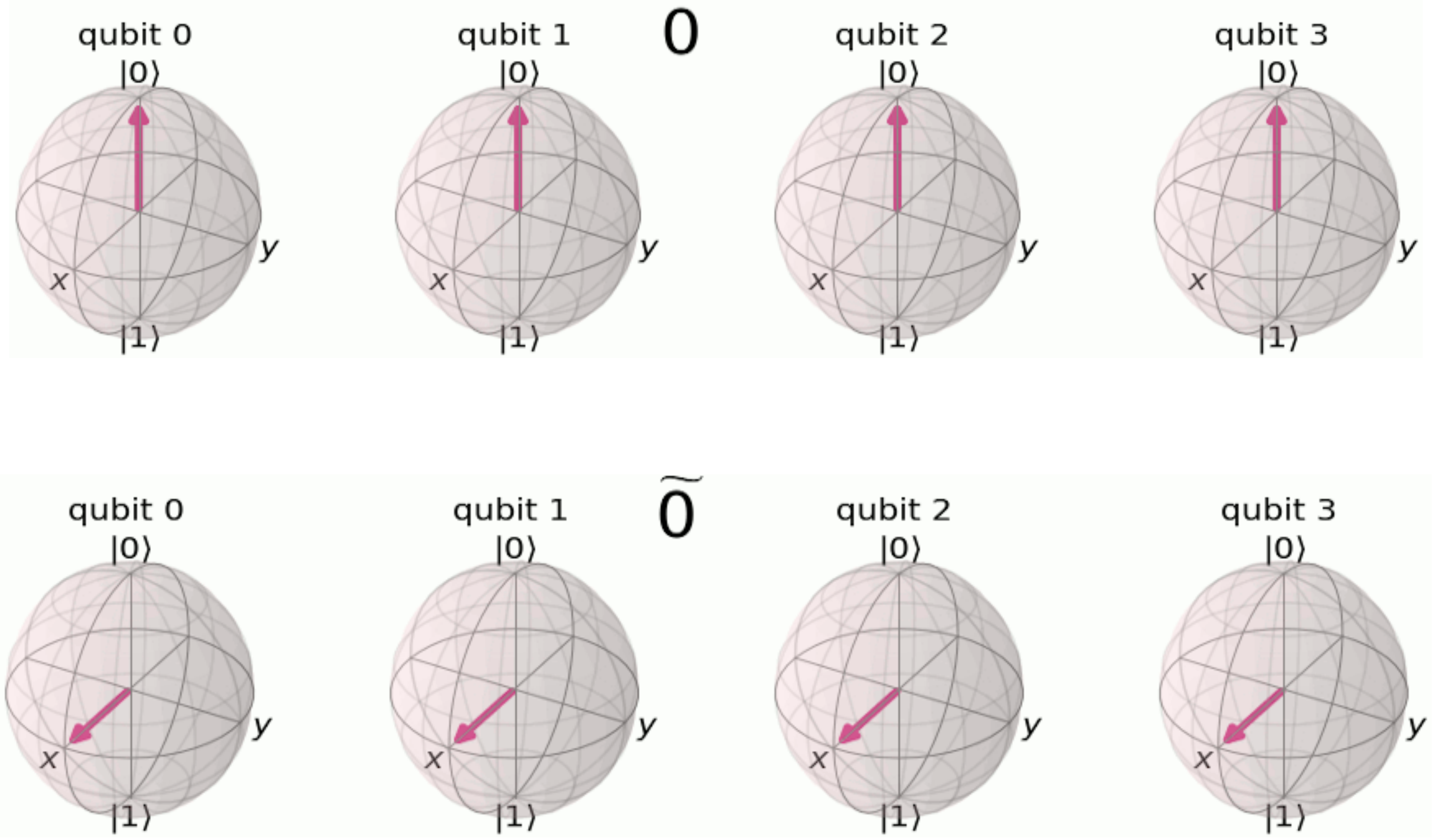
Demo Source

QFT Demos

- QFT Example for 1-qubit system [Source]
- QFT Example for 2-qubit system [Source]

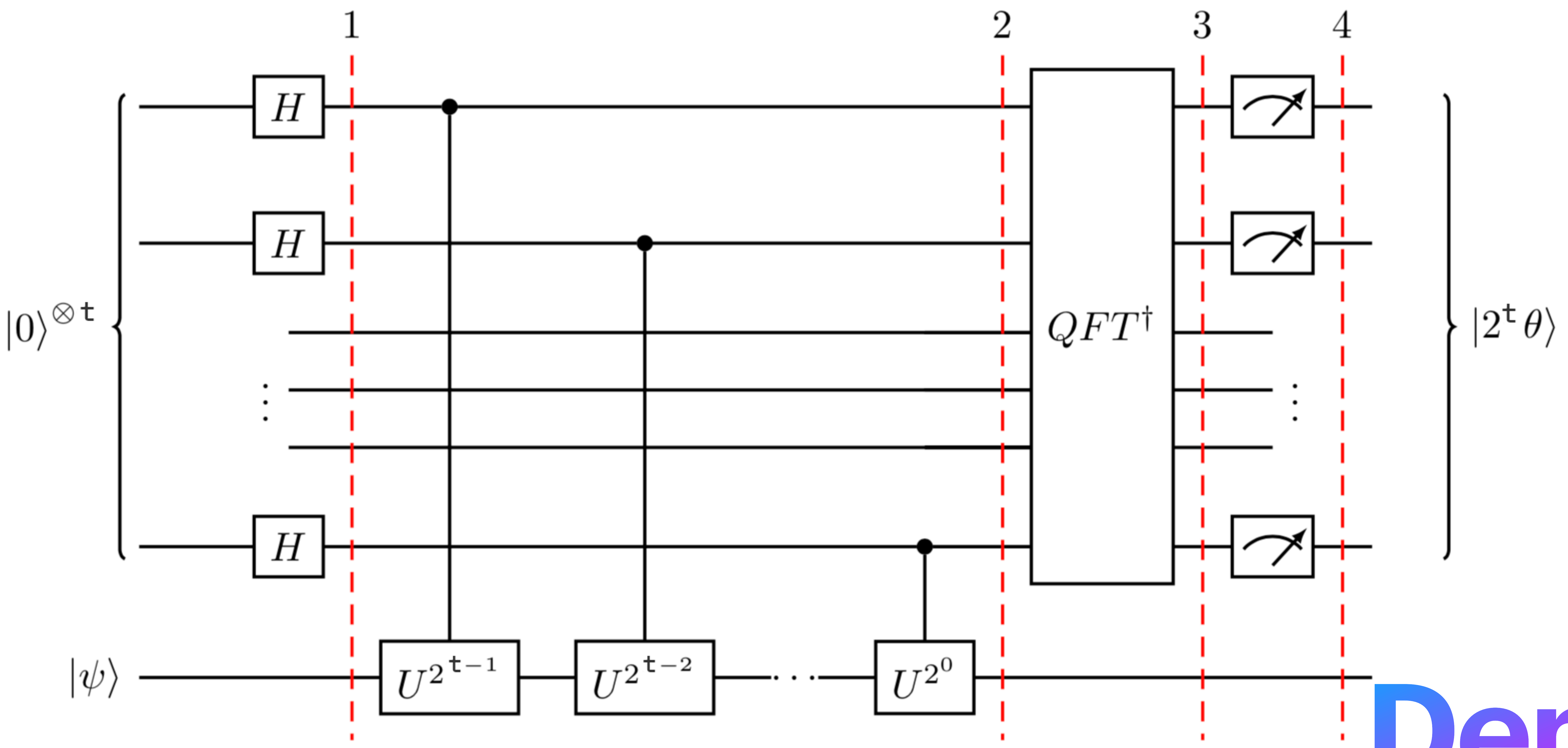
[Demo]

QFT with 4 qubits



Quantum Phase Estimation

- Given a unitary operator U, the algorithm estimates θ in $U|\psi\rangle = e^{2\pi i\theta}|\psi\rangle$, where $|\psi\rangle$ is an eigenvector and $e^{2\pi i\theta}$ is the corresponding eigenvalue



Demo Source

Simulating Molecules using VQE

- In the future Quantum Phase Estimation may be useful to find Eigen values of Hermitian matrix
- In 2014, Peruzzo et al. proposed Variational Quantum Eigensolver (VQE) method to estimate the ground state energy of molecule
- Given a Hermitian matrix H with unknown minimum eigenvalue λ_{min} and eigenstate $|\psi_{min}\rangle$, VQE provides an estimate λ_θ bounding λ_{min}:

  $$\lambda_{min} \leq \lambda_\theta \equiv \langle \psi(\theta) | H | \psi(\theta) \rangle$$

  where $|\psi(\theta)\rangle$ is eigenstate associated with λ_θ.

  - If we apply $U(\theta)$ to some arbitrary starting state $|\psi\rangle$, we can obtain an estimate $U(\theta)|\psi\rangle \equiv |\psi(\theta)\rangle$ on $|\psi_{min}\rangle$
  - Using classical controller the algorithm iteratively minimise the expectation value of $\langle \psi(\theta) | H | \psi(\theta) \rangle$

Demo Source

Resources

- The first part of the presentation is mostly based on:
  - https://qiskit.org/documentation/tutorials/circuits/3_summary_of_quantum_operations.html
- Second part is based on:
  - https://qiskit.org/textbook/ch-algorithms/index.html
- Useful Tutorial:
  - https://www.youtube.com/watch?v=a1NZC5rqQD8&list=PLOFEBzvs-Vvp2xg9-POLJhQwtVktlYGbY
- Find the presentation and examples on:
  - https://drive.google.com/drive/folders/1Dwh8zK2YD2cu3BWt-2HO0DQGvrYv2PfH?usp=sharing

Thank you!