Quantum Computing

Intensive Computation

Annalisa Massini
2022-2023

Lecture 19

References

- *Dancing with Qubits*
  Robert S. Sutor – Packt> – 2019

- *Quantum Computation and Quantum Information*
  Michael A. Nielsen & Isaac L. Chuang – Cambridge Press – 2010

- *Principles of quantum computation and Information*
  G. Benenti, G. Casati, G. Strini – World Scientific Pub – 2004
  - *Ch. 3* Quantum Computation

- *https://qiskit.org/textbook/what-is-quantum.html*

- *https://en.wikipedia.org/wiki/Quantum_logic_gate*

QUANTUM LOGIC GATES

(Continued)

One-Qubit gates

Exercises

- Verify that all gates introduced so far are their **own inverse**

- Verify that you can **create an X-gate** by sandwiching a Z-gate between two H-gates, that is $X = HZH$
  - Starting in the Z-basis, the H-gate switches our qubit to the X-basis, the Z-gate performs a NOT in the X-basis, and the final H-gate returns our qubit to the Z-basis

One-Qubit gates: the Pauli gates

Exercises - solutions

- Verifying $\boldsymbol{X, Y, Z, H}$ are their own inverse
  - $XX = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}\begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$
  - $YY = \begin{bmatrix} 0 & -i \\ i & 0 \end{bmatrix}\begin{bmatrix} 0 & -i \\ i & 0 \end{bmatrix} = \begin{bmatrix} -i^2 & 0 \\ 0 & -i^2 \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$
  - $ZZ = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}\begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$
  - $HH = \frac{1}{\sqrt{2}}\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}\frac{1}{\sqrt{2}}\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} = \frac{1}{2}\begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$

- Verifying $\boldsymbol{HZH}$ behaves like an X-gate
  - $HZH = \frac{1}{\sqrt{2}}\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}\begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}\frac{1}{\sqrt{2}}\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} = \frac{1}{2}\begin{bmatrix} 0 & 2 \\ 2 & 0 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} = X$

One-Qubit gates: Arbitrary rotations

- There are three gates that allow to do an **arbitrary rotation** around the x, y and z axis, respectively
- These operators are R_x, R_y and R_z, and are defined as:

$$R_x(\theta) = \begin{bmatrix} \cos\frac{\theta}{2} & -i\sin\frac{\theta}{2} \\ i\sin\frac{\theta}{2} & \cos\frac{\theta}{2} \end{bmatrix} \quad R_y(\theta) = \begin{bmatrix} \cos\frac{\theta}{2} & -\sin\frac{\theta}{2} \\ \sin\frac{\theta}{2} & \cos\frac{\theta}{2} \end{bmatrix} \quad R_z(\varphi) = \begin{bmatrix} 1 & 0 \\ 0 & e^{i\varphi} \end{bmatrix}$$

- Notice that while R_x and R_y change the probabilities of the system states, R_z does not (i.e. the probability of measuring $|0\rangle$ rather than $|1\rangle$ remains the same)
- What R_z changes is the relative phase of the qubit

One-Qubit gates: Arbitrary rotations

- R_z performs a rotation of φ around the Z-axis direction and changes the relative phase of the qubit

- R_z is a **parametrized** gate and is also called **P-gate**
- It needs a real number φ to tell it exactly what to do

- Notice that the Z-gate, that is $Z = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}$, is a special case of the P-gate $P = \begin{bmatrix} 1 & 0 \\ 0 & e^{i\varphi} \end{bmatrix}$, with $\varphi = \pi$:

$$Z = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & e^{i\pi} \end{bmatrix}$$

(Remember that the action of the Z-gate is a rotation around the z-axis of π radians, that is 180°)

One-Qubit gates: the S-gate

- The **S-gate**, also known as $\sqrt{Z}$-gate, is a P-gate with $\varphi = \pi/2$ around the Z-axis direction
- The **S-gate** does a quarter-turn around the Bloch sphere
- The matrice is: $S = \begin{bmatrix} 1 & 0 \\ 0 & e^{i\frac{\pi}{2}} \end{bmatrix}$
- The name $\sqrt{Z}$-gate is due to the fact that two successively applied S-gates has the same effect as one Z-gate:

$$SS|q\rangle = \begin{bmatrix} 1 & 0 \\ 0 & e^{i\frac{\pi}{2}} \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & e^{i\frac{\pi}{2}} \end{bmatrix} |q\rangle = \begin{bmatrix} 1 & 0 \\ 0 & e^{i\pi} \end{bmatrix} |q\rangle = Z|q\rangle$$

One-Qubit gates: the S-gate

- Unlike other gates introduced so far, the **S-gate is not its own inverse**
- Hence, we can have $S^\dagger$-gate (or $\sqrt{Z}^\dagger$-gate)
- The $S^\dagger$-gate is clearly a P-gate with $\varphi = -\pi/2$
- The matrix is: $$S^\dagger = \begin{bmatrix} 1 & 0 \\ 0 & e^{-i\frac{\pi}{2}} \end{bmatrix}$$
- It holds

$$SS^\dagger = \begin{bmatrix} 1 & 0 \\ 0 & e^{i\frac{\pi}{2}} \end{bmatrix}\begin{bmatrix} 1 & 0 \\ 0 & e^{-i\frac{\pi}{2}} \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & e^{i(\frac{\pi}{2}-\frac{\pi}{2})} \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

This confirms that is S is a unitary matrix

One-Qubit gates: the T-gate

- The **T-gate** is a P-gate with $\phi = \pi/4$
- The matrices are: $T = \begin{bmatrix} 1 & 0 \\ 0 & e^{i\frac{\pi}{4}} \end{bmatrix}$ and $T^\dagger = \begin{bmatrix} 1 & 0 \\ 0 & e^{-i\frac{\pi}{4}} \end{bmatrix}$
- As with the S-gate, the T-gate is sometimes also known as the $\sqrt[4]{Z}$-gate

One-Qubit gates: the U-gate

- The **U-gate** is the most general of all single-qubit quantum gates and is a parametrised gate of the form:

$$U(\theta,\phi,\lambda) = \begin{bmatrix} \cos\left(\frac{\theta}{2}\right) & -e^{i\lambda}\sin\left(\frac{\theta}{2}\right) \\ e^{i\phi}\sin\left(\frac{\theta}{2}\right) & e^{i(\phi+\lambda)}\cos\left(\frac{\theta}{2}\right) \end{bmatrix}$$

- Every gate could be specified as $U(\theta,\phi,\lambda)$, but it is unusual to see this in a circuit diagram
- As an example, we see the **U-gate** for representing the **H-gate** and **P-gate** respectively

$$H = U(\tfrac{\pi}{2},0,\pi) = \frac{1}{\sqrt{2}}\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$$ and $$\mathrm{P} = U(0,0,\lambda) = \begin{bmatrix} 1 & 0 \\ 0 & e^{i\lambda} \end{bmatrix}$$

MULTI-QUBIT GATES

Multi-Qubit gates

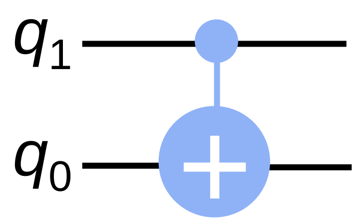
- Among the **multiple-qubit gates**, there is a wide range of gates which is based on the same principle: **controlled gates**
- A given number of **control qubits** decide if a given operation must be performed on another set of qubits or not
- In the case of a two-qubit, there is one **control** qubit and one **target** qubit

Multi-Qubit gates: CNOT gate

- An important two-qubit gate is the **CNOT-gate**

- It is a conditional gate that performs an X-gate on the second qubit, **target** bit, if the state of the first qubit, **control** bit is $|1\rangle$

- In the picture q1 is the control and q0 is the target

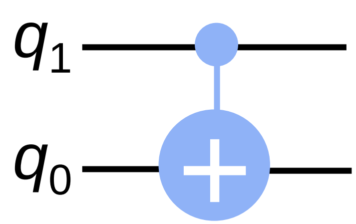


Multi-Qubit gates: CNOT gate

- The matrix of the CNOT gate is $$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

- This matrix **swaps the amplitudes of $|10\rangle$ and $|11\rangle$** in the statevector:

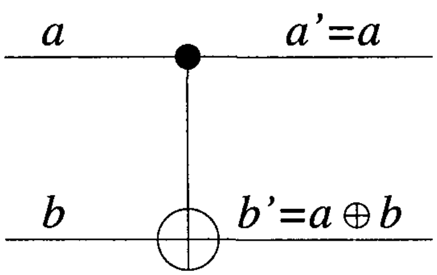
$$|a\rangle = \begin{bmatrix} a_{00} \\ a_{01} \\ a_{10} \\ a_{11} \end{bmatrix} \quad \mathrm{CNOT}|a\rangle = \begin{bmatrix} a_{00} \\ a_{01} \\ a_{11} \\ a_{10} \end{bmatrix}$$



Multi-Qubit gates: CNOT gate

- The **controlled-NOT**, or CNOT, is a **reversible** gate and perform **the XOR**, as shown in the true table below

a	b	a'	b'
0	0	0	0
0	1	0	1
1	0	1	1
1	1	1	0

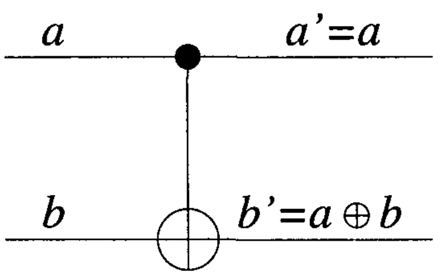


- The **second bit**, or target bit, is flipped if and only if the first bit is set to one and therefore $b' = a \oplus b$

Multi-Qubit gates: CNOT gate

- Note that, if we set the **target bit to 0**, the CNOT gates becomes the **FANOUT gate**: $(a, 0) \rightarrow (a, a)$

a	b	a'	b'
0	0	0	0
0	1	0	1
1	0	1	1
1	1	1	0



- It is easy to check that **CNOT is self-inverse**:
  - Indeed, the application of two CNOT gates, leads to

$$(a, b) \rightarrow (a, a \oplus b) \rightarrow \big(a, a \oplus (a \oplus b)\big) = (a, b)$$

  - Therefore, $(\text{CNOT})^2 = I$, that is $\text{CNOT}^{-1} = \text{CNOT}$

Multi-Qubit gates: Controlled gates

Generic controlled gates

- It is possible to define the operation performed by the generic single-qubit gate U by using the generic matrix

$$U = \begin{bmatrix} u_{00} & u_{01} \\ u_{10} & u_{11} \end{bmatrix}$$

- Assuming that the action of U on the target qubit must be taken only if the first qubit is equal to $|1\rangle$, for the controlled-U gate it holds that:

$$\text{CNOT } U = cU = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & u_{00} & u_{01} \\ 0 & 0 & u_{10} & u_{11} \end{bmatrix}$$

- All the single qubit gates previously presented can be theoretically implemented in the ***controlled version***

Multi-Qubit gates: Controlled gates

- We can write the action for all the four possible input patterns

$$\boldsymbol{cU|00\rangle} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & u_{00} & u_{01} \\ 0 & 0 & u_{10} & u_{11} \end{bmatrix} \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{bmatrix} = |00\rangle \qquad \boldsymbol{cU|01\rangle} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & u_{00} & u_{01} \\ 0 & 0 & u_{10} & u_{11} \end{bmatrix} \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \end{bmatrix} = |01\rangle$$

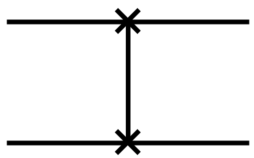
$$\boldsymbol{cU|10\rangle} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & u_{00} & u_{01} \\ 0 & 0 & u_{10} & u_{11} \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ u_{00} \\ u_{10} \end{bmatrix} = |1\rangle \otimes U|0\rangle$$

$$\boldsymbol{cU|11\rangle} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & u_{00} & u_{01} \\ 0 & 0 & u_{10} & u_{11} \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ u_{01} \\ u_{11} \end{bmatrix} = |1\rangle \otimes U|1\rangle$$

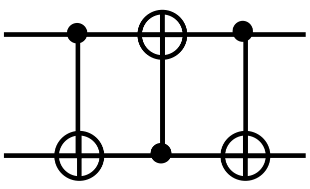
Multi-Qubit gates: Swap gate

- The **Swap gate** allows to swap two qubits
- It is defined as follows:

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



- In general, the action is: $|\psi'\rangle = SWAP|\psi\rangle = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} a \\ b \\ c \\ d \end{bmatrix} = \begin{bmatrix} a \\ c \\ b \\ d \end{bmatrix}$
- The **SWAP gate** is that it can be implemented, for example, using **three CNOT gates**



Multi-Qubit gates: CCNOT gate

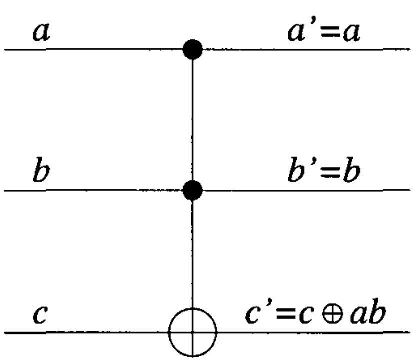
- It is possible to show that two-bit reversible gates are not enough for **universal computation**
- Instead, a universal gate is the **controlled-controlled-NOT** (**CCNOT**) or **Toffoli gate**, which is a three-bit gate
- The Toffoli gate has **two** control qubits and **one** target qubit
- The X operation is applied to the target qubit if and only if both control qubits are set to $|1\rangle$

Multi-Qubit gates: CCNOT gate

- The **CCNOT** gate acts as follows:
  - the **two control bits are unchanged**, that is $a' = a$ and $b' = b$
  - the **target bit is flipped** if and only if the two control bits are set to 1, that is $c' = c$ xor ab

Table and circuit of the CCNOT

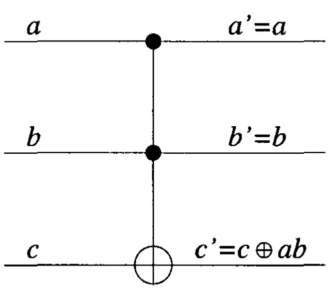
a	b	c	a'	b'	c'
0	0	0	0	0	0
0	0	1	0	0	1
0	1	0	0	1	0
0	1	1	0	1	1
1	0	0	1	0	0
1	0	1	1	0	1
1	1	0	1	1	1
1	1	1	1	1	0



Multi-Qubit gates: CCNOT gate

- The **CCNOT** gate (Toffoli gate) is **universal**
- To prove the CCNOT **universality**, we show how to use it to construct **both NAND and FANOUT gates**
  - If we set a = 1, the Toffoli gate acts on the other two bits as a CNOT and we have seen that the FANOUT gate can be constructed from the CNOT
  - Since $c' = c \oplus ab = \bar{c}\, ab + c\, \overline{ab}$, if we set $c = 1$, then $c' = 1 \oplus ab = 0\, ab + 1\, \overline{ab} = \overline{ab}$
- It is possible to **construct the NOT, AND, OR gates from the Toffoli gate**

a	b	c	a'	b'	c'
0	0	0	0	0	0
0	0	1	0	0	1
0	1	0	0	1	0
0	1	1	0	1	1
1	0	0	1	0	0
1	0	1	1	0	1
1	1	0	1	1	1
1	1	1	1	1	0

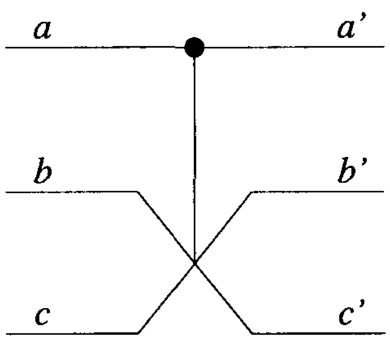


Multi-Qubit gates: CSWAP gate

- Another universal reversible gate is the **controlled-EXCHANGE gate** or **CSWAP gate** or **Fredkin gate**
- The SWAP operation is performed if and only if the control bit ***a*** **is set to 1** and the two target qubits b and c e are **swapped**

Table and circuit of the controlled-EXCHANGE

a	*b*	*c*	*a'*	*b'*	*c'*
0	0	0	0	0	0
0	0	1	0	0	1
0	1	0	0	1	0
0	1	1	0	1	1
1	0	0	1	0	0
1	0	1	1	1	0
1	1	0	1	0	1
1	1	1	1	1	1



Multi-Qubit gates

- Both the Toffoli and Fredkin gates are **self-inverse**
- The price to pay to have **reversible gates** is the introduction of additional qubits and on output this produces **garbage** qubits
- **Garbage bits**
  - are **not reused** during the computation
  - are needed to **store the information** that would allow us **to reverse the operations**
  - For instance, if we set c = 1 at the input of the Toffoli gate, we obtain $c' = \overline{ab}$ plus two garbage bits $a' = a$ and $b' = b$

HOW TO ANALYZE QUANTUM CIRCUITS

Analyzing a quantum circuit

- **Quantum operators** are described by means of **unitary matrices**
- A **quantum circuit** can be seen as **set of gates** connected to each other, where each gate is represented by a unitary matrix
- There can be two kinds of connections between gates belonging to the same circuit: **series** and **parallel** connections

- To understand the **behavior of a given circuit**, it is necessary to understand how to compute the **overall unitary matrix** describing the action of gates placed in parallel or in series

Analyzing a quantum circuit

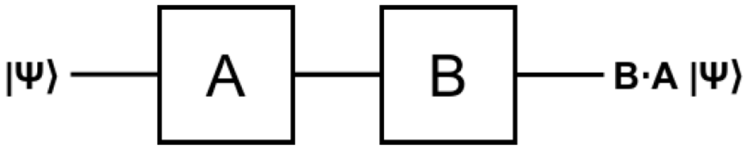
- The **time-flow in a circuit** is represented **from left to right**
- This means that the **evolution of the state of a qubit** has a physical meaning if considered from left to right

- However, when the matrix transfer function of the whole (or a part of the) circuit has to be computed, **unitary matrices must be written from right to left**
  - The **leftmost gate** in the circuit is described by the **rightmost unitary matrix**

Analyzing a quantum circuit

Gates Connected in Series

- The overall transfer function of two generic one-qubit quantum gates connected in series can be computed as shown in the figure below



- The output after the input passed through gate A and B is:

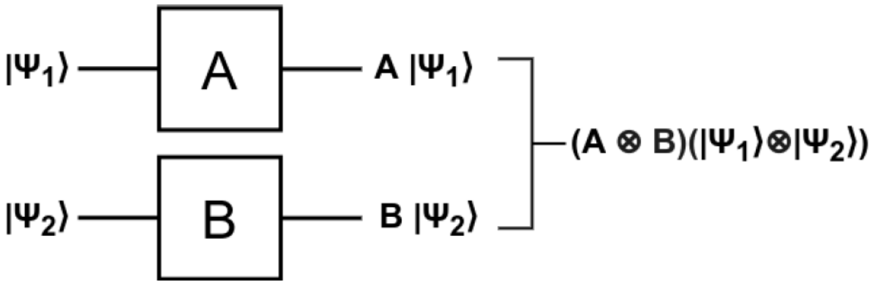
$$|\psi\rangle = BA|\psi\rangle$$

- The method can be extended to an arbitrary number of gates

Analyzing a quantum circuit

Gates Connected in Parallel

- When two gates are placed in parallel, the **overall unitary matrix** acting on the **two qubits** is obtained using the **Kronecker product**, as shown in the figure below



- The output after the inputs passed through gates A and B is:

$$A|\psi_1\rangle \otimes B|\psi_2\rangle = (A \otimes B)(|\psi_1\rangle \otimes |\psi_2\rangle) = (A \otimes B)|\psi_1\psi_2\rangle$$

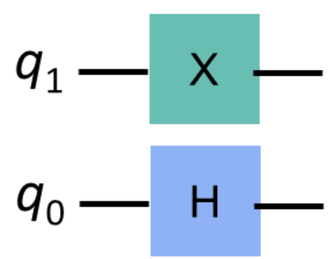
- The method can be extended to an arbitrary number of gates

One-Qubit gates on multi-Qubit

Example

- We have that a single bit gate acts on a qubit in a multi-qubit vector using the **tensor product** to calculate matrices that act on multi-qubit statevectors
- For example, if on q_1 acts the **X-gate** (NOT) and on q_0 acts the **H-gate** we can represent the simultaneous operations *X and H* using their **Kronecker product**:

$$X|q_1\rangle \otimes H|q_0\rangle = (X \otimes H)|q_1 q_0\rangle$$

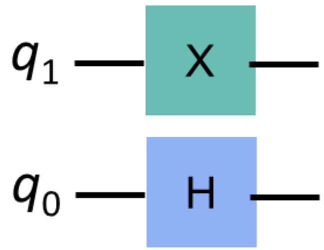


One-Qubit gates on multi-Qubit

- The **operation** $X|q_1\rangle \otimes H|q_0\rangle = (X \otimes H)|q_1 q_0\rangle$ is given by:

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

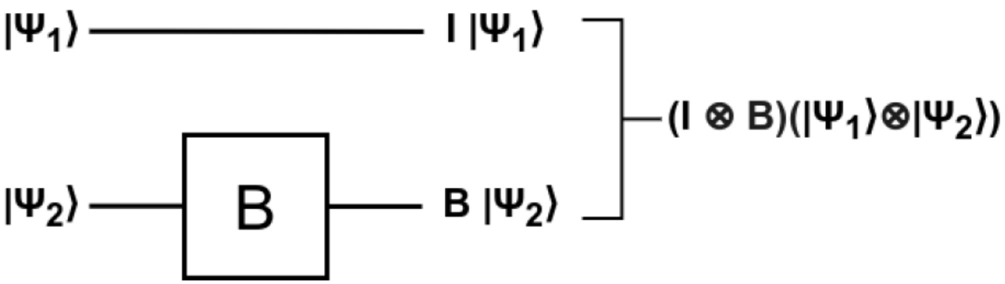
$$= \frac{1}{\sqrt{2}} \begin{bmatrix} 0 \times \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} & 1 \times \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \\ 1 \times \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} & 0 \times \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \end{bmatrix} = \frac{1}{\sqrt{2}}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



Analyzing a quantum circuit

Gates Connected in Parallel

- If gates are applied only to a subset of the inputs, qubits where no gates are acting can be treated as operated by an identity, as shown in the figure below

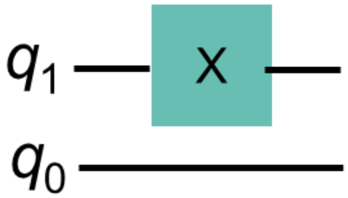


- The output after the inputs passed through gate B is:
$|\psi_1\rangle \otimes B|\psi_2\rangle = (I \otimes B)(|\psi_1\rangle \otimes |\psi_2\rangle) = (I \otimes B)|\psi_1\psi_2\rangle$

One-Qubit gates on multi-Qubit

Example

- We need to **apply a gate to only one qubit** at a time, such as in the circuit below where on q_1 acts the **X-gate** (NOT)

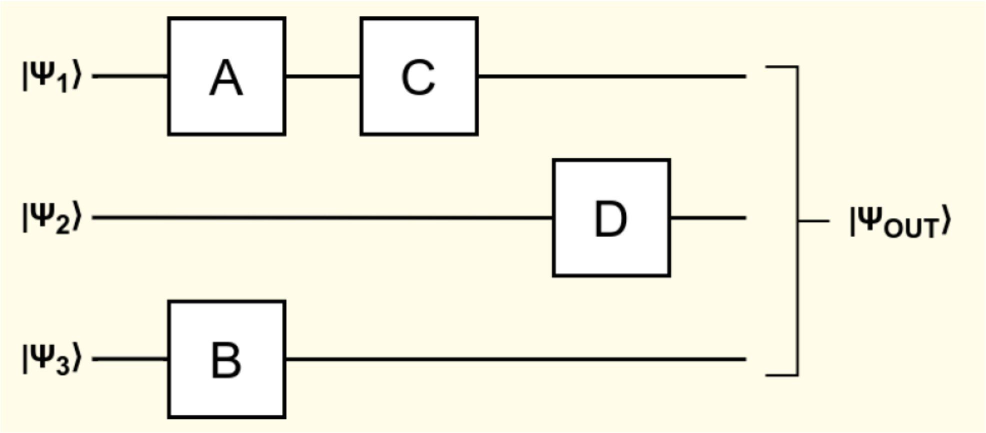


- In such a case, we describe the operation using **Kronecker product with the identity matrix**, e.g.: $X \otimes I$, giving

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

Example of a circuit

- Let us consider the following circuit, where A, B, C and D represent generic gates



- To analyze this circuit, two steps have to be followed

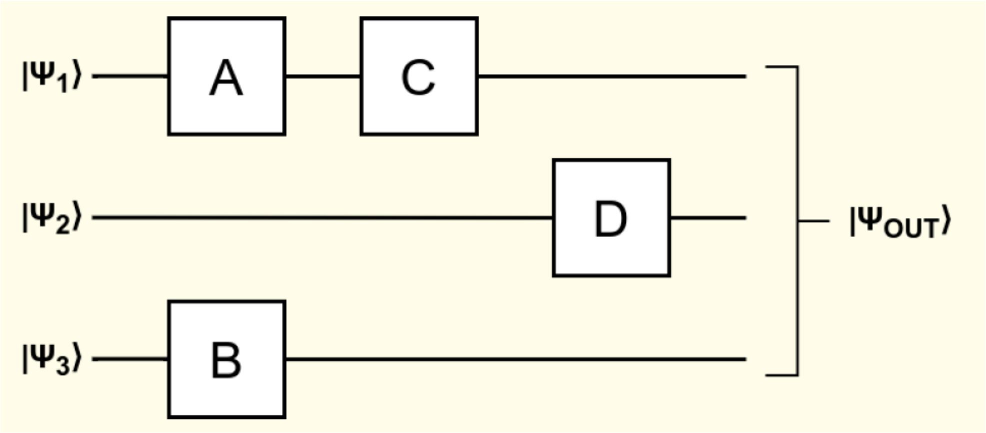
Example of a circuit

1) Write a unique expression for the three input qubits by performing the tensor product among them:

$$|\psi_1\rangle \otimes |\psi_2\rangle \otimes |\psi_3\rangle = |\psi_1\psi_2\psi_3\rangle$$

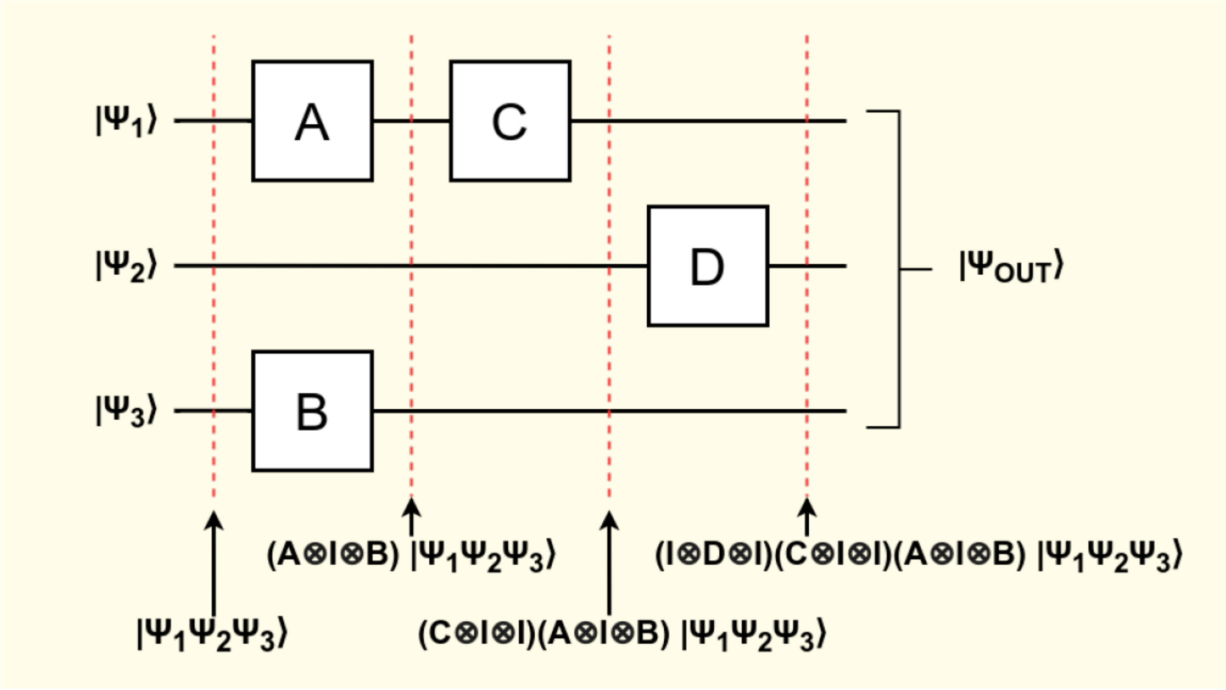
2) Compute the overall matrix function considering the gates from right to left (where I_k is the identity matrix of order k):

$$|\psi_{out}\rangle = (I_2 \otimes D \otimes I_2) \cdot (C \otimes I_4) \cdot (A \otimes I_2 \otimes B)\ |\psi_1\psi_2\psi_3\rangle$$



Example of a circuit

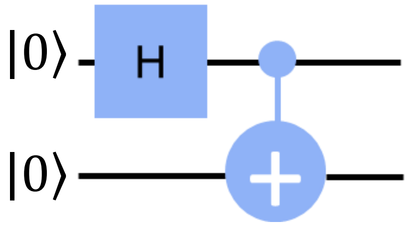
- The step-by-step analysis is shown here below



Example with H and CNOT gates

- In real quantum circuit analysis, we can follow two different strategies:
  - Exploiting the **matrix calculation**, as done before
  - Adopting a method based on **truth tables** of different gates, that can be faster

- Let us consider the circuit below



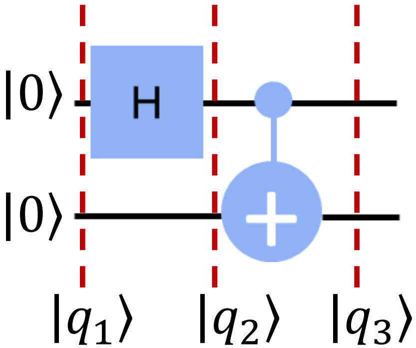
Example with H and CNOT gates

Matrix multiplication

- In this circuit we have two operators: the Hadamard gate and the CNOT gate, represented by the two unitary matrices

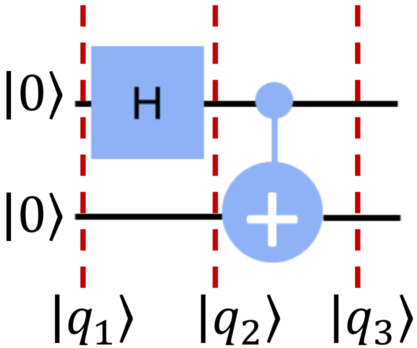
$$\mathrm{H} = \frac{1}{\sqrt{2}}\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \qquad \mathrm{CNOT} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

- We compute $|q_1\rangle$, $|q_2\rangle$ and $|q_3\rangle$ corresponding to the values shown in the figure



Example with H and CNOT gates

Matrix multiplication



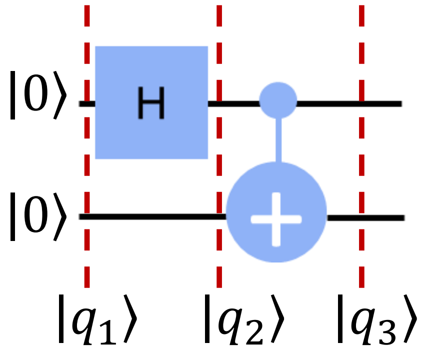
- $|q_1\rangle = |0\rangle \otimes |0\rangle = |00\rangle = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{bmatrix}$

- $|q_2\rangle = (H \otimes I)|00\rangle = \left(\frac{1}{\sqrt{2}}\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \otimes \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}\right)\begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{bmatrix} =$

$$\frac{1}{\sqrt{2}}\begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 1 & 0 & -1 & 0 \\ 0 & 1 & 0 & -1 \end{bmatrix}\begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{bmatrix} = \frac{1}{\sqrt{2}}\begin{bmatrix} 1 \\ 0 \\ 1 \\ 0 \end{bmatrix} = \frac{1}{\sqrt{2}}(|00\rangle + |10\rangle)$$

Example with H and CNOT gates

Matrix multiplication



- $|q_3\rangle = CNOT \cdot \frac{1}{\sqrt{2}}(|00\rangle + |10\rangle) =$

$$= \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix} \frac{1}{\sqrt{2}} \begin{bmatrix} 1 \\ 0 \\ 1 \\ 0 \end{bmatrix} = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 \\ 0 \\ 0 \\ 1 \end{bmatrix} =$$

$$= \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle)$$

Example with H and CNOT gates

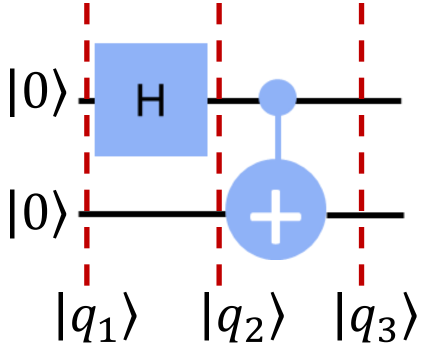
Truth tables

- This approach exploits the **truth tables** as shown here below for the involved operators

Hadamard	CNOT
$H\|0\rangle = \frac{1}{\sqrt{2}}(\|0\rangle + \|1\rangle) = \|+\rangle$	$\text{CNOT}\|0x\rangle = \|0x\rangle$
$H\|1\rangle = \frac{1}{\sqrt{2}}(\|0\rangle - \|1\rangle) = \|-\rangle$	$\text{CNOT}\|1x\rangle = \|1\bar{x}\rangle$

- It is typically much quicker to apply than the matrix method

Example with H and CNOT gates



Truth tables

- $|q_1\rangle = |0\rangle \otimes |0\rangle = |00\rangle$

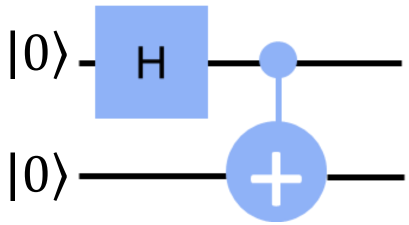
- $|q_2\rangle = H|0\rangle \otimes |0\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle) \otimes |0\rangle = \frac{1}{\sqrt{2}}(|00\rangle + |10\rangle)$

- $|q_3\rangle = CNOT \cdot \frac{1}{\sqrt{2}}(|00\rangle + |10\rangle) = \frac{1}{\sqrt{2}}(CNOT|00\rangle + CNOT|10\rangle) =$

$= \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle)$

Example with H and CNOT gates

- We can look at this circuit also in a different way

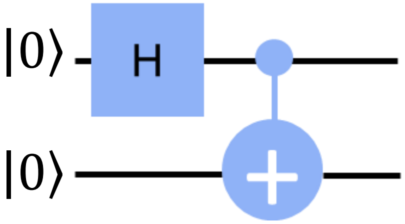


- Applying the H gate to $|0\rangle$ we obtain state $|+\rangle$

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

Example with H and CNOT gates

- So, we can see how CNOT gate acts on a qubit in superposition given by the state $|+\rangle$



- Before we apply the CNOT we have

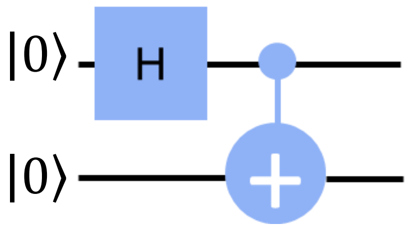
$$|{+}0\rangle = |+\rangle \otimes |0\rangle = \mathrm{H}|0\rangle \otimes |0\rangle = \frac{1}{\sqrt{2}}\begin{bmatrix}1\\1\end{bmatrix} \otimes \begin{bmatrix}1\\0\end{bmatrix} = \frac{1}{\sqrt{2}}\begin{bmatrix}1 \times \begin{bmatrix}1\\0\end{bmatrix}\\ 1 \times \begin{bmatrix}1\\0\end{bmatrix}\end{bmatrix}$$

$$= \frac{1}{\sqrt{2}}\begin{bmatrix}1\\0\\1\\0\end{bmatrix} = \frac{1}{\sqrt{2}}\left(\begin{bmatrix}1\\0\\0\\0\end{bmatrix} + \begin{bmatrix}0\\0\\1\\0\end{bmatrix}\right) = \frac{1}{\sqrt{2}}(|00\rangle + |10\rangle)$$

Example with H and CNOT gates

- When we **apply the CNOT gate**, we have the state

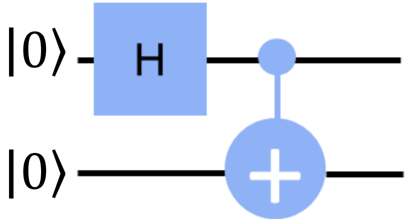
$$\mathrm{CNOT}|{+0}\rangle = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix} \frac{1}{\sqrt{2}} \begin{bmatrix} 1 \\ 0 \\ 1 \\ 0 \end{bmatrix} = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 \\ 0 \\ 0 \\ 1 \end{bmatrix} = \frac{1}{\sqrt{2}} \left(\begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} \right)$$
$$= \frac{1}{\sqrt{2}} (|00\rangle + |11\rangle)$$



Example with H and CNOT gates

- $\mathrm{CNOT}|{+0}\rangle = \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle)$ is one of the ***Bell states***

- As we said, this state is interesting because it is ***entangled*** and it has:
  - 50% probability of being measured in the state $|00\rangle$
  - 50% chance of being measured in the state $|11\rangle$
  - And, most interestingly, **0%** chance of being measured in the states $|01\rangle$ or $|10\rangle$
  - This state cannot be written as two separate qubit states

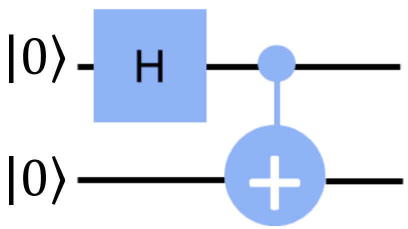


Example with H and CNOT gates

- Although our qubits are in superposition, measuring one will tell us the state of the other and **collapse its superposition**

- For example, if we measured the top qubit and got the state $|1\rangle$ the **collective state of our qubits** changes like

$$\frac{1}{\sqrt{2}}(|00\rangle + |11\rangle) \xrightarrow{\text{measure}} |11\rangle$$



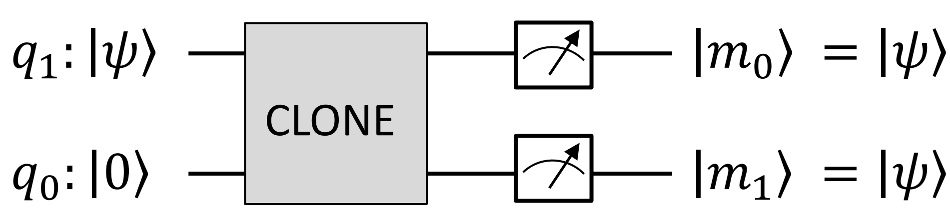
- Even ***if we separated these qubits*** light-years away, measuring one qubit collapses the superposition and appears to have an *immediate effect on the other*

NO CLONING THEOREM

No-Cloning theorem

- **Cloning** – so easy to accomplish with classical information – turns out not to be possible in general in quantum mechanics
- ***No-cloning theorem***, discovered in the early 1980s, is one of the earliest results of quantum computation and quantum information

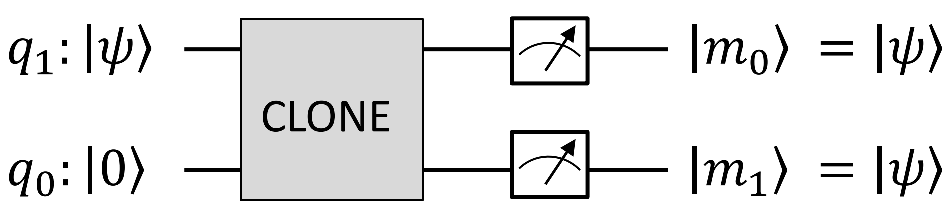
- Let’s try to build a circuit that makes a copy of a qubit’s state
- We’re looking for something like:



No-Cloning theorem

- The initial state of q_0 does not matter since it is a placeholder we want to replace with the state of q_1
- We are not looking for a gate that clones one particular qubit state but rather one that makes a copy of any arbitrary state
- If the **CLONE** gate exists, let *C* be its unitary matrix in the standard ket basis

- As usual, we take: $|\psi\rangle = a|0\rangle + b|1\rangle$



No-Cloning theorem

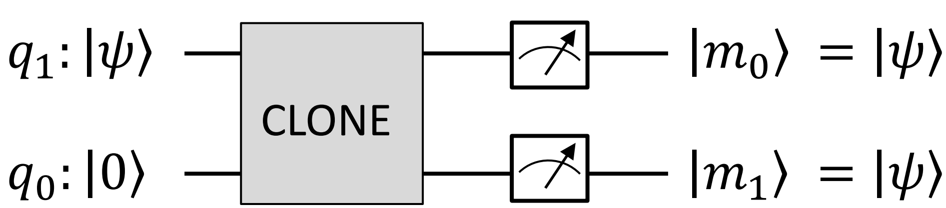
- The result after cloning is: $|\psi\rangle \otimes |\psi\rangle$
- That is: $C(|\psi\rangle \otimes |0\rangle) = |\psi\rangle \otimes |\psi\rangle$
- **But, are these really equal?**
- On the **left** we have:

$$C(|\psi\rangle \otimes |0\rangle) = C((a|0\rangle + b|1\rangle) \otimes |0\rangle) =$$

$$= C(a|0\rangle \otimes |0\rangle + b|1\rangle \otimes |0\rangle) =$$

$$= aC(|0\rangle \otimes |0\rangle) + bC(|1\rangle \otimes |0\rangle) =$$ by linearity

$$= a|00\rangle + b|11\rangle$$ by definition of Clone and C



No-Cloning theorem

- On the **right** we have:

$$|\psi\rangle \otimes |\psi\rangle = (a|0\rangle + b|1\rangle) \otimes (a|0\rangle + b|1\rangle) =$$
$$= a^2|00\rangle + ab|01\rangle + ba|10\rangle + b^2|11\rangle$$

- For arbitrary a and b in $\mathbb{C}$ with $|a|^2 + |b|^2 = 1$

$$a|00\rangle + b|11\rangle \neq a^2|00\rangle + ab|01\rangle + ba|10\rangle + b^2|11\rangle$$

- **Hence,** there is no **CLONE** gate that can duplicate the quantum state of a qubit
- This is called the *No-Cloning Theorem*