

Advanced Engineering Physics

Unit-4

Quantum Computing

1. Introduction to quantum computing:

The computer today is made by introducing several tiny transistors embedded on silicon chips. We call these computers as classical computers. Examples of classical computers are Desktop computer, laptop, Tab, mobile, smart watch, etc. Tiny transistors act as switches depicting the states 0 and 1. These discrete states are called bits which are readable by the classical computer. Physically, state '1' on a bit depicts a developed/high voltage (*a 'bit' refers here as an abstract physical segment on which 0 or 1 is assigned*) in a capacitor in combination with transistors on a chip (Example: RAM - random access memory). On the other hand, state '0' represents declined/low voltage in a capacitor in combination with transistors on a chip. All types of calculations are made easy for computers by converting information (numbers, characters etc.) into binary digit system. All arithmetic operations are carried out by the computers only when the numbers are represented in binary digits as 0 and 1. The simple addition operation on 3 and 4 is 7. However, this is an impossible task for a computer unless they are fed in the form of binary digits. Therefore, our job is to convert it into the required binary form and feed it. The binary forms of 3 and 4 are 11 and 100 respectively. The addition of them made by the computer is 111. The "111" binary digit is again reformed to decimal as "7" which is readable by a human. Increasing a greater number of such bits on a processor chip speeds up the computation.

As technology developed over the decades the transistors fabricated on an integrated chip within a certain area are being doubled, and price is halved every two years. This law was the extrapolation of the observation made on the development of chip technology predicted by Gordon Moore (one of the co-founders of Intel Company). This is popularly known as Moor's Law. Today, there can be 1.2 trillion transistors printed over 21 x 21 square cm. Such a big number enhanced the speed of a processor by a significant magnitude which in turn squeezed the individual transistor to nano dimension. This has made a single transistor to attain such a smaller size where roughly 100 electrons are participating in switching action. Such miniaturization of the size of an individual transistor makes switching of state 1 to 0 and vice-versa more difficult. This is because the quantum mechanical effects will start playing a major role by not allowing electrons and atoms to behave as they were expected to be before the size of a transistor was made smaller. This is a result of trapping electrons and atoms within a small, confined nano area. Because of the quantum principles like quantum superposition, energy discreteness, quantum tunneling, quantum interference etc., it is impossible for a transistor to create a resistive path permanently when the state of a bit must be written as 0 or conductive path for the state 1. As a result, resistive path may conduct electricity inevitably through quantum tunneling effect. Such restriction has significantly hindered the further development of microprocessor fabrication technology and showed the red flag. Physicists across the globe experienced the limitation of lowering the size of a transistor by such

quantum mechanical effects. However, this limitation turned the whole picture upside down and became a milestone for the inception of an entirely new computational method which is called “Quantum Computation”. They decided to exploit the same limitation as foundation to open an entirely new area of computation which is even greater than the earlier “Classical Computation.”

Moore's Law

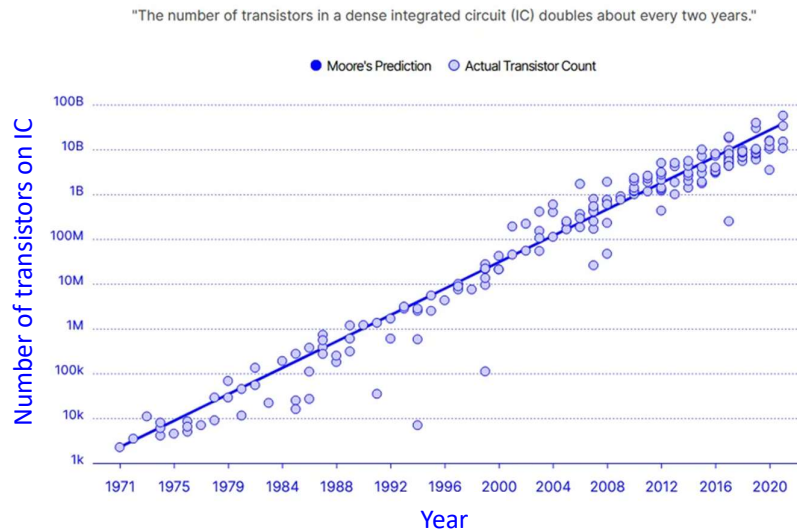


Fig. 1 Number of transistors in an integrated circuit developed over the years.

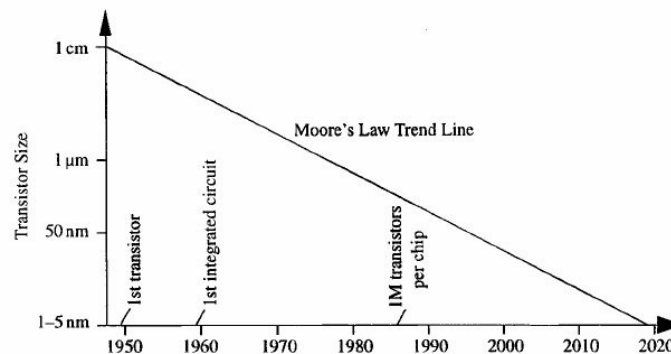


Fig. 2. Size of the transistor in an integrated circuit developed over the years

Quantum computers are not the replacement of classical computers with which we are working today. But they are fundamentally different in their basic principles of operations. It is known that classical and quantum laws behave similarly for a large-scale system. It is only in the very small systems (with dimension of nanometre) where peculiarities of quantum mechanics are visible. It is understood that quantum computers are useful (advantageous) only when these peculiarities can be exploited to do computation. So, we will call a system quantum computer, if it uses these distinguishing (weird) properties/ principles of quantum mechanics such as wave-particle duality, quantum superposition, quantum entanglement,

quantum Interference, quantum tunneling, Heisenberg's uncertainty principle, etc. These principles enable the quantum computer to drastically boost computational power among all other computers ever built.

Computers have become so polished that we can use and even program them without understanding how they work at a fundamental level. That's not a bad thing. It has allowed computers to become tools for more and more people. One day, quantum computing will get to the point of accessibility, where we can use and program them without worrying about their details. In their development, quantum computers are where classical computers were decades ago. Their inner working still matter, and to understand these inner workings, it is helpful to understand the inner workings of regular, classical computers. Without knowing classical computing, some aspect of quantum computing may seem arbitrary. A rudimentary understanding classical computing makes it easier to understand quantum computing.

Bits:

The term bits has become fairly commonplace, where many people will state that bits are zeros and ones. Consider a coin, Assuming that coins do not balance on their edges, a single coin lying on a flat surface either has heads facing up or tails facing up. Let us call these two possible conditions, or states, heads (H) and tails (T). The head side of the ₹20 coin is the side with the national emblem, while the tail side is the one that bears the numerical value.



Fig.3 Head and tail of ₹20 coin

If we have two coins, there are four possible states: ($2^2 = 4$ possible states)

HH, HT, TH, TT.

Adding a third coin, there are now eight possible states: ($2^3 = 8$)

HHH, HHT, HTH, HTT, THH, THT, TTH, TTT.

Generalizing this, with n coins, there are 2^n possible states.

Physical Bits:

Many physical systems only have two states. We already discussed coins, which can be heads or tails. Another example is a light switch, which can be "off" or "on." As another example, information is stored on optical discs (e.g., CDs and DVDs) using a laser that burns holes into the disc, called "pits." For example, here is a zoomed-in picture of a DVD which is taken using an atomic force microscope.

The picture shows microscopic holes and trenches burned into the disc, i.e., the pits. Where there is no hole is called a “land.” So, the two states are whether there is a pit or a land, and to read the disc, a laser shines on the disc and detects whether there is a pit or a land.

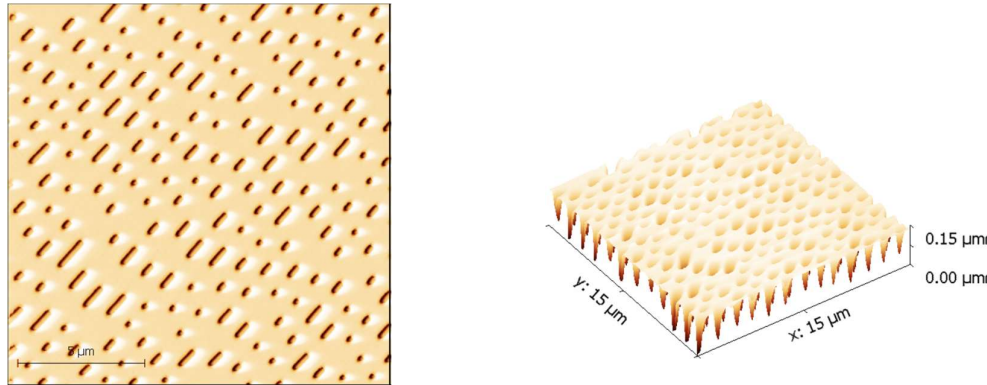


Fig 4 The Atomic force microscopic image of the surface of a DVD

In conventional computers, however, we typically only use two values, 0 volts and 5 volts, so we effectively only have two states. All of these examples have, or effectively have, just two states, called by different names: heads/tails, off/on, pits/lands, and 0 V / 5 V. Rather than using so many different names, it is often easier to use generic names for the two states so that we can describe the state without regard for the underlying physical system. Mathematically, it is convenient to use the binary digits 0 and 1. This is summarized in the table below Physical System States

Physical System	States	
Coin	Heads	Tails
Switch	Off	On
Disc	Pit	Land
Voltage	0 V	5 V
Binary Digit	0	1

With this convention, regardless of the physical system, we can just refer to the two possible states as 0 and 1, the binary digits. Since binary digits are used so much in computing, an abbreviation for **binary digit** was invented: the **bit**. So, a bit is just a binary digit, which can be 0 or 1, which represent the states of any physical system with two states.

Figure 6 illustrates the recording process using longitudinal recording technology. In this technology the polarities of the magnets are parallel to the surface of the hard disk. When two identical poles are next to each other (S–S or N–N), a strong magnetic field will emerge from the recording medium, but no field will emerge when opposite poles (S–N) are next to each other. Therefore, when a magnetic-field sensor (example: a giant magnetoresistive [GMR] sensor) moves across this surface, voltage will be produced only when the GMR sensor

goes over the transitions (regions where like poles meet). The GMR sensor produces a voltage, the voltage is represented as “1.” If no voltage is produced, the absence of voltage is represented by “0.” This is a simple illustration of how 1s and 0s are stored in hard disk media.

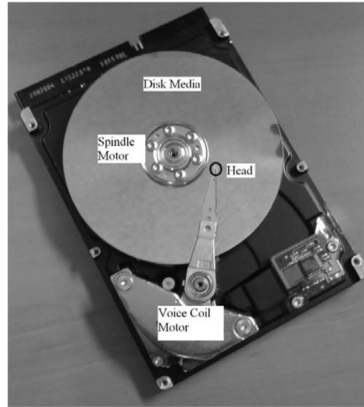


Fig. 6 Picture of a hard disk drive and various components

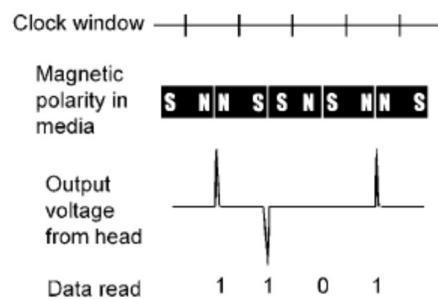


Fig 7. Illustration of the recorded pulses from magnetic transitions

The computation on these bits is performed using logic gates. The math of classical computing is based on Boolean algebra.

Quantum Bit (Qbit):

The *bit* is the fundamental concept of classical computation and classical information. Quantum computation and quantum information are built upon an analogous concept, the *quantum bit*, or *qubit* for short. Just as there are many ways to realize classical bits physically (two voltage levels, lights on or off in an array, positions of toggle switches), there are many ways to realize quantum bits physically. These qubits are realized from quantum particles like electrons, protons, molecules, quantum trapped systems etc.

Just as a classical bit has a *state* – either 0 or 1 – a qubit also has a state. Two possible states for a qubit are the states $|0\rangle$ and $|1\rangle$. Notation like $| \rangle$ is called the *Dirac notation*, and we will be seeing it often, as it is the standard notation for states in quantum mechanics. It was invented by Paul Dirac and is used often in quantum mechanics

The difference between bits and qubits is that a qubit can be in a state *other* than $|0\rangle$ and $|1\rangle$. It is also possible to form *linear combinations* of states, often called *superpositions*:

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

The numbers α and β are complex numbers. When we measure a qubit we get either the result 0, with probability $|\alpha|^2$, or the result 1, with probability $|\beta|^2$. Naturally,

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

since the probabilities must sum to one. The ability of a qubit to be in a superposition state runs counter to our 'common sense' understanding of the physical world around us. A classical bit is like a coin: either heads or tails up. By contrast, a qubit can exist in a *continuum* of states between $|0\rangle$ and $|1\rangle$ - until it is observed.



Fig: 8 A spinning coin

Let's imagine (for quantum analogy to understand the existence of continuum of states) a coin balanced on an edge and it is spinning as shown in the Fig 9. While spinning, the coin is not strictly heads or tails. It has some probability of being head and some probability of being tails. This spinning coin represents a qubit in superposition. Let us emphasize again that when a qubit is measured, it only ever gives '0' or '1' as the measurement result – probabilistically. For example, a qubit can be in the state

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

which, when measured, gives the result '0' fifty percent $\left(\frac{1}{\sqrt{2}}\right)^2$ of the time, and the result '1' fifty percent of the time. This is exactly what happens in quantum measurement. Despite this strangeness, qubits are decidedly real, their existence and behavior extensively validated by experiments and many different physical systems can be used to realize qubits.

Representation of qubits

Block sphere

Bloch sphere is an imaginary sphere which is used to represent pure single-qubit states as a point on its surface. It has unit radius. Its North Pole and South Pole are selected to represent the basis states namely $|0\rangle$ and $|1\rangle$. North Pole represents $|0\rangle$ (say spin up $\uparrow$) and South Pole represents $|1\rangle$ (say spin down $\downarrow$). All other points on the sphere represent superposed states. Bloch sphere allows the state of a qubit to be represented in spherical coordinates (ie, r , θ and φ) as shown in Fig.

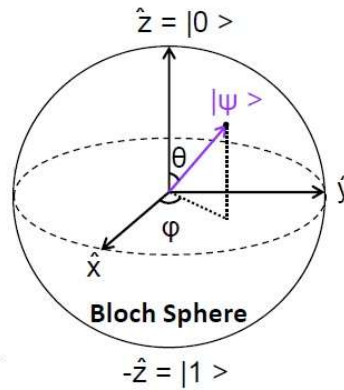


Fig 9. Bloch sphere representation of a qubit.

The state qubit $|\psi\rangle$ on the Bloch sphere makes an angle θ with z-axis and its projection (azimuth) makes angle φ with x-axis as shown. It is clear from the fig that $0 < \theta < \pi$ and $0 < \varphi < 2\pi$.

We already know that $|\psi\rangle$ is represented as

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

We can rewrite the $|\psi\rangle$ as

$$|\psi\rangle = r_1 e^{i\gamma_1} |0\rangle + r_2 e^{i\gamma_2} |1\rangle$$

$$|\psi\rangle = e^{i\gamma_1} (r_1 |0\rangle + r_2 e^{i\gamma_2 - \gamma_1} |1\rangle)$$

Remove global phase and consider $(\gamma_2 - \gamma_1) = \varphi$

$$|\psi\rangle = (r_1 |0\rangle + r_2 e^{i\varphi} |1\rangle)$$

Here we have a condition of $r_1^2 + r_2^2 = 1$

Which can be satisfied by the $r_1 = \cos \frac{\theta}{2}$ and $r_2 = \sin \frac{\theta}{2}$

We can rewrite the $|\psi\rangle$ as

$$|\psi\rangle = \left(\cos \frac{\theta}{2} |0\rangle + e^{i\varphi} \sin \frac{\theta}{2} |1\rangle \right) \text{-----(1)}$$

Because the value $|\alpha|^2 + |\beta|^2 = 1$. Even though, the given condition is satisfied by $\cos \theta$ and $\sin \theta$, still we are taking $\cos \theta/2$ and $\sin \theta/2$ in the formula. The reason for this is as follows:

In trigonometric functions, the sine function varies from 0 to 1 as the angle increases from **0°** to **90°**, and the cosine function also varies from 1 to 0 over the same interval. However, on the **Bloch sphere**, the polar angle **θ** is defined within the range:

$$0^\circ \leq \theta \leq 180^\circ$$

At the north pole of the Bloch sphere, $\theta = 0^\circ$, and at the south pole, $\theta = 180^\circ$.

So if we used $\sin(\theta)$ and $\cos(\theta)$ directly, then:

- At **$\theta = 90^\circ$** , $\sin(\theta)$ would already become 1 and $\cos(\theta)$ would become 0
- That means the transition from $|0\rangle$ to $|1\rangle$ would be finished halfway (at 90°), not at the full 180° of the Bloch sphere

But we want:

- $|0\rangle$ at $\theta = 0^\circ$
- **superposition states** at $0^\circ < \theta < 180^\circ$
- $|1\rangle$ at $\theta = 180^\circ$

So, we spread the variation of sine and cosine across 180° by using **$\theta/2$** instead of θ .

Now:

- When $\theta = 0^\circ \rightarrow \theta/2 = 0^\circ \rightarrow \cos(\theta/2) = 1, \sin(\theta/2) = 0$ (state $|0\rangle$)
- When $\theta = 180^\circ \rightarrow \theta/2 = 90^\circ \rightarrow \cos(\theta/2) = 0, \sin(\theta/2) = 1$ (state $|1\rangle$)

Thus, the correct state representation is:

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

Hence, We use **$\theta/2$** so that the sine and cosine amplitudes change correctly from $|0\rangle$ to $|1\rangle$ over the entire Bloch sphere, rather than finishing halfway at 90° .

Using this equation, we can represent $|\psi\rangle$ for different θ and φ as follows.

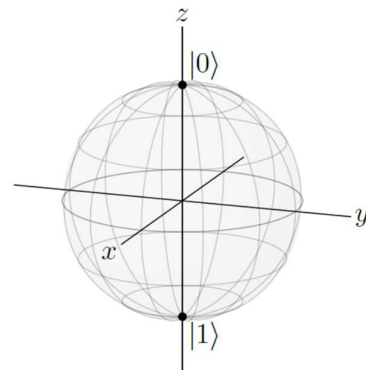
Case-1: let $\theta = 0$ and $\varphi = 0$, then eq. (1) becomes.

$$|\psi\rangle = \left(\cos 0 |0\rangle + e^{i0} \sin 0 |1\rangle \right) = |0\rangle + 0 |1\rangle = |0\rangle$$

$$|\psi\rangle = |0\rangle$$

Case-2: let $\theta = \pi$ and $\varphi = 0$, then eq. (1) becomes.

$$|\psi\rangle = \left(\cos \frac{\pi}{2} |0\rangle + e^{i0} \sin \frac{\pi}{2} |1\rangle \right) = 0 + |1\rangle$$



$$|\psi\rangle = |1\rangle$$

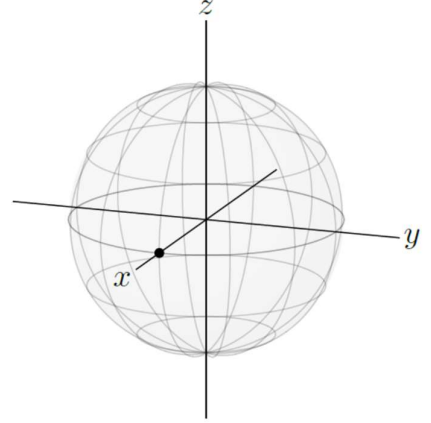
Following the standard physics convention, the x-axis comes out of the page, the y-axis points to the side, and z-axis is oriented up. Then, since the Bloch sphere has radius 1, $|0\rangle$ corresponds to the (x,y, z) point (0,0,1), and $|1\rangle$ corresponds to (0,0,-1).

Case-3: let $\theta = \frac{\pi}{2}$ and $\varphi = 0$, then eq. (1) becomes.

$$|\psi\rangle = \left(\cos \frac{\pi}{4} |0\rangle + e^{i0} \sin \frac{\pi}{4} |1\rangle \right) = \frac{1}{\sqrt{2}} |0\rangle + \frac{1}{\sqrt{2}} |1\rangle$$

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

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



In this state, the coefficient of $|0\rangle$ is $\frac{1}{\sqrt{2}}$ and the coefficient of $|1\rangle$ is also $\frac{1}{\sqrt{2}}$. So, it is equal parts $|0\rangle$ and $|1\rangle$. Given this, it should be on the equator of the Bloch sphere, which is halfway between the north and south poles. This state is at the (x,y, z) point (1,0,0), where the Bloch sphere intersects the x-axis.

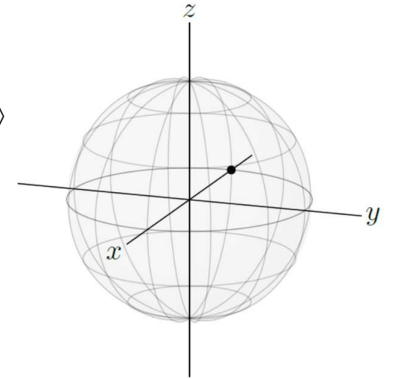
There are many other states on the equator of the Bloch sphere, all of which are equal parts $|0\rangle$ and $|1\rangle$. We can reach them by changing the relative phase of $|0\rangle$ and $|1\rangle$.

Case-4: let $\theta = \frac{\pi}{2}$ and $\varphi = \pi$, then eq. (1) becomes

$$|\psi\rangle = \left(\cos \frac{\pi}{4} |0\rangle + e^{i\pi} \sin \frac{\pi}{4} |1\rangle \right) = \frac{1}{\sqrt{2}} |0\rangle + (\cos \pi + i \sin \pi) \frac{1}{\sqrt{2}} |1\rangle = \frac{1}{\sqrt{2}} |0\rangle + (-1) \frac{1}{\sqrt{2}} |1\rangle$$

$$|\psi\rangle = \frac{1}{\sqrt{2}} (|0\rangle - |1\rangle)$$

This state is at the point (-1,0,0), on the -x-axis.

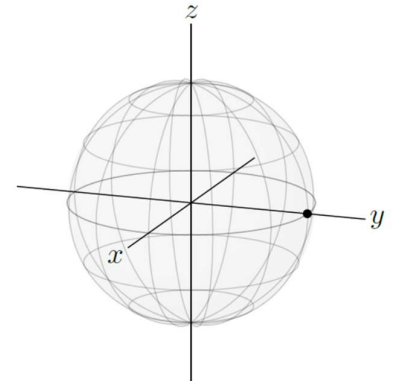


Case-5: : let $\theta = \frac{\pi}{2}$ and $\varphi = \frac{\pi}{2}$, then eq. (1) becomes

$$|\psi\rangle = \left(\cos \frac{\pi}{4} |0\rangle + e^{i\frac{\pi}{2}} \sin \frac{\pi}{4} |1\rangle \right) = \frac{1}{\sqrt{2}} |0\rangle + (\cos \frac{\pi}{2} + i \sin \frac{\pi}{2}) \frac{1}{\sqrt{2}} |1\rangle = \frac{1}{\sqrt{2}} |0\rangle + (i) \frac{1}{\sqrt{2}} |1\rangle$$

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

This state is at the point (0,1,0), on the y-axis.

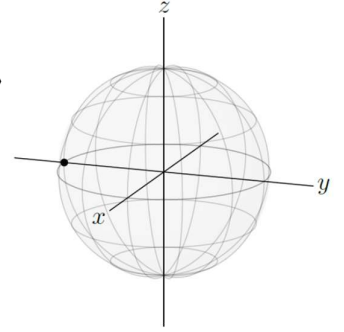


Case-6: : let $\theta = \frac{\pi}{2}$ and $\varphi = -\frac{\pi}{2}$, then eq. (1) becomes

$$|\psi\rangle = \left(\cos \frac{\pi}{4} |0\rangle + e^{-i\frac{\pi}{2}} \sin \frac{\pi}{4} |1\rangle \right) = \frac{1}{\sqrt{2}} |0\rangle + \left(\cos \frac{-\pi}{2} + i \sin \frac{-\pi}{2} \right) \frac{1}{\sqrt{2}} |1\rangle = \frac{1}{\sqrt{2}} |0\rangle + (-i) \frac{1}{\sqrt{2}} |1\rangle$$

$$|\psi\rangle = \frac{1}{\sqrt{2}} (|0\rangle - i|1\rangle)$$

This state is at the point (0,-1,0), on the -y-axis.



Dirac Notation:

Dirac notation also known as bra-ket notation, is a mathematical notation used in quantum mechanics to describe and manipulate quantum states, vectors, and operators. It was introduced by physicist Paul Dirac as a concise and powerful way to represent quantum concepts.

Consider a quantum state ψ in a vector space represented by $|\psi\rangle$. The notation $|\rangle$ indicates that the object is a vector and is called a ket vector. The examples of such ket vectors are like $|\psi\rangle$ and $|\varphi\rangle$ etc.

The wave function could be expressed in ket notation as $|\psi\rangle$ (ket Vector), is the wave function.

Hence, any arbitrary state can be represented as,

$$|\psi\rangle = \begin{bmatrix} \alpha \\ \beta \end{bmatrix} \quad \text{or} \quad |\psi\rangle = \alpha |0\rangle + \beta |1\rangle$$

The 'ket' vector typically represented as a column vector and 'bra' vector typically represented as a row vector. The matrix for of the states $|0\rangle$ and $|1\rangle$ as follows;

$$\text{Ket Notation:} \quad |0\rangle = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \quad |1\rangle = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$$

$$\text{Bra Notation:} \quad \langle 0| = [1 \quad 0] \quad \langle 1| = [0 \quad 1]$$

Please note that a bra is a transpose of the ket. Transpose means that the rows and columns of a ket switch to become the columns and rows of a bra, respectively.

$$\text{The state} \quad |\psi\rangle = \alpha \begin{bmatrix} 1 \\ 0 \end{bmatrix} + \beta \begin{bmatrix} 0 \\ 1 \end{bmatrix} = \begin{bmatrix} \alpha \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ \beta \end{bmatrix} = \begin{bmatrix} \alpha \\ \beta \end{bmatrix}$$

In the context of quantum computing, Dirac notation is used to represent quantum states, gates, and operations, making it a concise and elegant way to describe and manipulate quantum information. It simplifies calculations and allows for easy visualization of quantum concepts.

Advantages of Dirac Notation

(i) **Conciseness:** Dirac notation significantly simplifies the representation of quantum states and operations, making complex calculations easier to read and write. It is used to develop quantum algorithms and quantum error correction codes.

(ii) **Linear Algebra Focus:** It emphasizes the underlying linear algebra structure of quantum mechanics, making it easier to leverage mathematical tools for analyzing quantum systems.

(iii) **Operator Focus:** Dirac notation highlights the importance of operators in quantum mechanics, which represent observables like position, momentum, and energy.

Linear algebra for quantum computation:

Linear algebra is a branch of mathematics used to represent and manipulate quantum states (vectors), quantum operations (matrices), and predict the behaviour of quantum systems. In simple words quantum computation is about performing linear algebra on quantum states."

(a) Inner product to combine a bra and a ket:

An inner (dot product) product measures the overlap between two states.

Let us consider two quantum states or ket $|\phi\rangle$ and $|\psi\rangle$.

$$|\phi\rangle = \begin{bmatrix} \phi_1 \\ \phi_2 \end{bmatrix}, |\psi\rangle = \begin{bmatrix} \psi_1 \\ \psi_2 \end{bmatrix}$$

First calculate the bra of $|\phi\rangle$:

$$\langle\phi| = [\phi_1^* \phi_2^*]$$

Then, do matrix multiplication of $\langle\phi|$ and $|\psi\rangle$:

$$\langle\phi|\psi\rangle = [\phi_1^* \phi_2^*] \begin{bmatrix} \psi_1 \\ \psi_2 \end{bmatrix}$$

Which gives a single complex number:

$$\langle\phi|\psi\rangle = \phi_1^* \psi_1 + \phi_2^* \psi_2$$

- The above inner product looks like a bracket, hence the terms bra and ket.
- The bra forms the left part and the ket the right part of the inner product in Dirac notation.

(a) An Example for calculating Inner Product:

Let us take two kets

$$|\phi\rangle = \begin{bmatrix} 1 \\ i \end{bmatrix}, |\psi\rangle = \begin{bmatrix} 2 \\ 3i \end{bmatrix}$$

For calculating their inner product, let us first find the bra of $|\phi\rangle$, i.e. $\langle\phi|$:

$$\langle\phi| = [1^*(i)^*] = [1 - i]$$

On multiplication:

$$\begin{aligned} \langle\phi|\psi\rangle &= [1 - i] \begin{bmatrix} 2 \\ 3i \end{bmatrix} \\ \langle\phi|\psi\rangle &= (1)(2) + (-i)(3i) \\ \langle\phi|\psi\rangle &= 2 + (-i)(3i) = 2 + 3 = 5 \end{aligned}$$

Note:

(i) Orthogonality:

Whenever the inner product of two vectors equals 0, the vectors are orthogonal (or perpendicular).

The inner product of a vector with itself is equal to one $\langle\psi|\psi\rangle = 1$ and is known as normalization condition.

The physical significance of normalization is that the probability amplitude of the quantum system.

Consider

$$\begin{aligned} |0\rangle &= \begin{bmatrix} 1 \\ 0 \end{bmatrix} \text{ and } |1\rangle = \begin{bmatrix} 0 \\ 1 \end{bmatrix} \\ \langle 0|1\rangle &= [1 \ 0] \begin{bmatrix} 0 \\ 1 \end{bmatrix} = 0 \end{aligned}$$

Hence $|0\rangle$ is perpendicular to $|1\rangle$.

(ii) Orthonormality:

If each element of a set of vectors is normalized and the elements are orthogonal with respect to each other, we say the set is orthonormal.

Consider the set

$$\begin{aligned} |0\rangle &= \begin{bmatrix} 1 \\ 0 \end{bmatrix}, \quad |1\rangle = \begin{bmatrix} 0 \\ 1 \end{bmatrix} \\ \langle 0|0\rangle &= [1 \ 0] \begin{bmatrix} 1 \\ 0 \end{bmatrix} = 1 && \text{normalized} \\ \langle 0|1\rangle &= [1 \ 0] \begin{bmatrix} 0 \\ 1 \end{bmatrix} = 0 && \text{orthogonal} \end{aligned}$$

$$\langle 1 | 1 \rangle = [0 \ 1] \begin{bmatrix} 0 \\ 1 \end{bmatrix} = 1 \quad \text{normalized}$$

$$\langle 1 | 0 \rangle = [0 \ 1] \begin{bmatrix} 1 \\ 0 \end{bmatrix} = 0 \quad \text{orthogonal}$$

Hence set of $|0\rangle$ and $|1\rangle$ is orthonormal.

From the above relations, if states $|0\rangle$ and $|1\rangle$ are said to be orthonormal if

1. $|0\rangle$ and $|1\rangle$ are normalized.
2. $|0\rangle$ and $|1\rangle$ are orthogonal to each other.

(iii) Normalization of probability:

Let us consider a quantum state

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

$$|\psi\rangle = \alpha \begin{bmatrix} 1 \\ 0 \end{bmatrix} + \beta \begin{bmatrix} 0 \\ 1 \end{bmatrix} = \begin{bmatrix} \alpha \\ \beta \end{bmatrix}$$

The inner product $\langle\psi | \psi\rangle$:

$$\langle\psi | \psi\rangle = [\alpha^* \ \beta^*] \begin{bmatrix} \alpha \\ \beta \end{bmatrix} = \alpha^* \alpha + \beta^* \beta$$

$$\alpha^* \alpha + \beta^* \beta = |\alpha|^2 + |\beta|^2$$

This could also be written as $|\psi|^2 = \psi\psi^*$.

The above equation represents probability density.

As per the principle of normalization.

$$|\psi|^2 = \psi\psi^* \langle\psi | \psi\rangle = 1 = |\alpha|^2 + |\beta|^2$$

So, probability is normalized.

(c) Outer bracket to combine a Bra and Ket:

Consider two quantum states $|\phi\rangle$ and $|\psi\rangle$

$$|\phi\rangle = \begin{bmatrix} \phi_1 \\ \phi_2 \end{bmatrix}, |\psi\rangle = \begin{bmatrix} \psi_1 \\ \psi_2 \end{bmatrix}$$

Let us first calculate the bra of $|\psi\rangle$

$$\langle\psi| = [\psi_1^* \ \psi_2^*]$$

$$|\phi\rangle\langle\psi| = \begin{bmatrix} \phi_1 \\ \phi_2 \end{bmatrix} [\psi_1^* \ \psi_2^*]$$

The outer product is a matrix:

$$|\phi\rangle\langle\psi| = \begin{bmatrix} \phi_1\psi_1^* & \phi_1\psi_2^* \\ \phi_2\psi_1^* & \phi_2\psi_2^* \end{bmatrix}$$

(d) An example for calculating an outer product:

Consider two kets as

$$|\phi\rangle = \begin{bmatrix} 1 \\ i \end{bmatrix}, |\psi\rangle = \begin{bmatrix} 2 \\ 3i \end{bmatrix}$$

For calculating outer product let us find the bra of $|\psi\rangle$

$$\langle\psi| = [2^*(3i)^*] = [2 - 3i]$$

On multiplication of two kets:

$$|\phi\rangle\langle\psi| = \begin{bmatrix} 1 \\ i \end{bmatrix} [2 - 3i]$$

The product of two kets is a matrix:

$$|\phi\rangle\langle\psi| = \begin{bmatrix} (1)(2) & (1)(-3i) \\ (i)(2) & (i)(-3i) \end{bmatrix}$$

$$|\phi\rangle\langle\psi| = \begin{bmatrix} 2 & -3i \\ 2i & 3 \end{bmatrix}$$

Operators

Operators are mathematical entities that represent physical processes which alter the state vector of a system. These observables may also represent physical properties of a system which can be measured experimentally for example, position, momentum, or energy.

Def:

An operator is a mathematical rule (actions on quantum states) that converts a given function into another function.

Consider an operator '**A**' that transforms the vector $|a\rangle$ to another vector $|b\rangle$. It can be written as

$$\hat{A} |a\rangle = |b\rangle$$

There are different types of operators like **Linear operator, Identity operator, Null operator, Inverse operator, Singular & non-singular operator**, etc.

Identity operator I :

The identity operator is an operator which, operating on a function, leaves the function unchanged.

i.e.,

$$I | a \rangle = | a \rangle$$

It is given in matrix form by

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

This is also called the **identity matrix**.

There will be no change when I operates on either the $| 0 \rangle$ state or $| 1 \rangle$ state.

It is explained as follows:

$$\begin{aligned} I | 0 \rangle &= \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \\ \therefore I | 0 \rangle &= | 0 \rangle \end{aligned}$$

Similarly,

$$\begin{aligned} I | 1 \rangle &= \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 0 \\ 1 \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \end{bmatrix} \\ \therefore I | 1 \rangle &= | 1 \rangle \end{aligned}$$

Identity matrix acts as number 1.

It is always a square matrix.

Pauli Matrices:

These are the 2×2 complex matrices introduced by Pauli in order to account for the interaction of the spin with an external electromagnetic field. They are given by

$$\begin{aligned} \sigma_1 = \sigma_x = X &= \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \\ \sigma_2 = \sigma_y = Y &= \begin{bmatrix} 0 & -i \\ i & 0 \end{bmatrix} \\ \sigma_3 = \sigma_z = Z &= \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} \end{aligned}$$

Operation of Pauli Matrices on $| 0 \rangle$ and $| 1 \rangle$ states

Three Pauli matrices X , Y , and Z operate on states $| 0 \rangle$ and $| 1 \rangle$ as follows:

- **X operating on $| 0 \rangle$ and $| 1 \rangle$**

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

Since X inverts each input (i.e., $| 0 \rangle$ becomes $| 1 \rangle$ and $| 1 \rangle$ becomes $| 0 \rangle$).

It is also called as bit-flip gate.

If a superposed qubit goes through X gate, the result will be

$$\begin{aligned}
X | \psi \rangle &= \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} \alpha \\ \beta \end{bmatrix} = \begin{bmatrix} \beta \\ \alpha \end{bmatrix} = \alpha | 1 \rangle + \beta | 0 \rangle \\
X | \psi \rangle &= \alpha | 1 \rangle + \beta | 0 \rangle
\end{aligned}$$

• Y operating on $| 0 \rangle$ and $| 1 \rangle$

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

If a superposed qubit goes through Y gate, the result will be

$$\begin{aligned}
Y | \psi \rangle &= \begin{bmatrix} 0 & -i \\ i & 0 \end{bmatrix} \begin{bmatrix} \alpha \\ \beta \end{bmatrix} = \begin{bmatrix} -i\beta \\ i\alpha \end{bmatrix} = -i\beta | 0 \rangle + i\alpha | 1 \rangle \\
Y | \psi \rangle &= -i\beta | 0 \rangle + i\alpha | 1 \rangle
\end{aligned}$$

• Z operating on $| 0 \rangle$ and $| 1 \rangle$

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

If a superposed qubit goes through **Z gate**, the result will be

$$\begin{aligned}
Z | \psi \rangle &= \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} \begin{bmatrix} \alpha \\ \beta \end{bmatrix} = \begin{bmatrix} \alpha \\ -\beta \end{bmatrix} = \alpha | 0 \rangle - \beta | 1 \rangle \\
Z | \psi \rangle &= \alpha | 0 \rangle - \beta | 1 \rangle
\end{aligned}$$

Hilbert space:

Hilbert space is a mathematical concept used to describe the state space of quantum mechanical systems. It is a vector space equipped with an inner product, allowing for the definition of concepts like and orthogonality, which are crucial for understanding quantum phenomena.

Hilbert space is used to understand and study non-relativistic quantum mechanics (of objects travelling at real-world speeds). The various wave functions in quantum mechanics describe the states in which quantum particles live in Hilbert space. It is a fusion of algebra, topology and geometry.

Hilbert space is a mathematical framework suitable for describing the concepts, principles, processes and laws of quantum mechanics. Pure states of quantum systems are considered to be vectors of a Hilbert space. One can say that to each isolated quantum system corresponds a Hilbert space. Some even go farther by claiming that there is no reality on the quantum level; such a reality emerges only in the case of a measurement, and what we know about the quantum level are only computational procedures, expressed in terms of Hilbert space concepts, to compute evolutions of quantum systems and probabilities of the measurement outcomes.

The vector spaces we consider will be over the complex numbers, and are finite dimensional, which significantly simplifies the mathematics we need. Such vector spaces are members of a class of vector spaces called Hilbert spaces.

All the quantum computing literature refers to a finite-dimensional complex vector space by the name 'Hilbert space. Since H is finite-dimensional, we can choose a basis and alternatively represent vectors (kets) in this basis as finite column vectors, and represent operators with finite matrices.

Advantages of using Hilbert space

- Hilbert spaces are the fundamental building blocks for quantum information theory
- Physical systems can be abstracted, allowing the development of quantum algorithms.
- Offers powerful mathematical tools for analyzing and manipulating quantum states.
- For analyzing and manipulating quantum states.

Multiple Qbits (Two qubits and Extension to N qubits):

Two-Qubit States

Quantum states combine through the tensor product. Two-qubit states can be written as $|\psi_1\rangle \otimes |\psi_2\rangle$, where " $\otimes$ " indicates a **tensor product**, or more compactly $|\psi_1\rangle |\psi_2\rangle$.

- The **tensor product** combines two vector spaces into a new vector space. It is a way to create a larger, more complex space from smaller and simpler ones.

Example 1:

$$|\psi_1\rangle = a|0\rangle + b|1\rangle$$

$$|\psi_1\rangle = \begin{bmatrix} a \\ b \end{bmatrix}$$

$$|\psi_2\rangle = c|0\rangle + d|1\rangle$$

$$|\psi_2\rangle = \begin{bmatrix} c \\ d \end{bmatrix}$$

$$|\psi_1\rangle \otimes |\psi_2\rangle = \begin{bmatrix} a \\ b \end{bmatrix} \otimes \begin{bmatrix} c \\ d \end{bmatrix} = \begin{bmatrix} ac \\ ad \\ bc \\ bd \end{bmatrix}$$

Example 2:

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

$$01 \equiv \begin{bmatrix} 1 \\ 0 \end{bmatrix} \otimes \begin{bmatrix} 0 \\ 1 \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \end{bmatrix}$$

$$10 \equiv \begin{bmatrix} 0 \\ 1 \end{bmatrix} \otimes \begin{bmatrix} 1 \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \end{bmatrix}$$

$$11 \equiv \begin{bmatrix} 0 \\ 1 \end{bmatrix} \otimes \begin{bmatrix} 0 \\ 1 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix}$$

The state qubit is (i.e., linear combination of these four)

$$|\Psi\rangle = \alpha_{00}|00\rangle + \alpha_{01}|01\rangle + \alpha_{10}|10\rangle + \alpha_{11}|11\rangle$$

For a 2-qubit system we have 4 complex amplitudes, namely α_{00} , α_{01} , α_{10} and α_{11} . According to the normalization condition

$$|\alpha_{00}|^2 + |\alpha_{01}|^2 + |\alpha_{10}|^2 + |\alpha_{11}|^2 = 1$$

Similarly, if there are 3 qubits there will be 8 complex amplitudes.

In general, for N qubits, there will be 2^N complex amplitudes.

Thus, an n -qubit state can be represented by a column vector with 2^n elements.

Note:

(i) A two-qubit system can represent four possible states (00, 01, 10, 11) in superposition, leading to a more complex computational space than a single qubit.

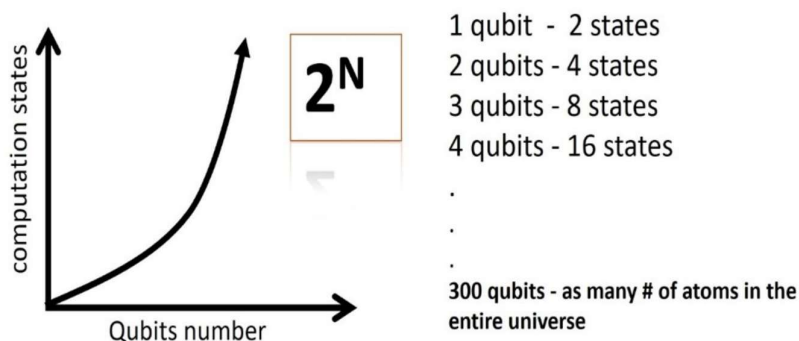
(ii) A two-qubit system can be represented by a four-dimensional vector, with basis states like $|00\rangle, |01\rangle, |10\rangle, |11\rangle$.

(iii) Two-qubit systems can be **entangled** (the state of one qubit is linked to the state of the other, regardless of the distance separating them).

Multi-Qubit System

It is a collection of multiple qubits, treated as a single system. In classical computing, a system of N bits can be in 2^N states. For example, for a 2-bit system, there are 4 possible states 00, 01, 10, and 11.

In a quantum system of N qubits, there are 2^N basis states and a qubit can be in any of these states or even their superposition. In multi-qubit systems, as the number of qubits increases, the number of possible states in superposition grows exponentially (2^n for n qubits). This exponential growth in computational space gives more computational power to tackle complex problems that are intractable for classical computers.



Consider a qubit with $|0\rangle$ and $|1\rangle$ as its computational states. For qubits these two states are available for computation **at the same time**. If there are two such qubits there, then there would be 4 states available **at the same time**. The vital aspect is the growth of the number of states available for computation after the addition of every single qubit. Because, after adding a qubit, the total number of states doubles. This would follow the

exponential trend as shown in the above figure. Because, if there are N qubits, then there would be 2^N states available **at the same time**. Such exponential power of growth of number of computation states is unique for quantum computer in making it more powerful than classical computers.

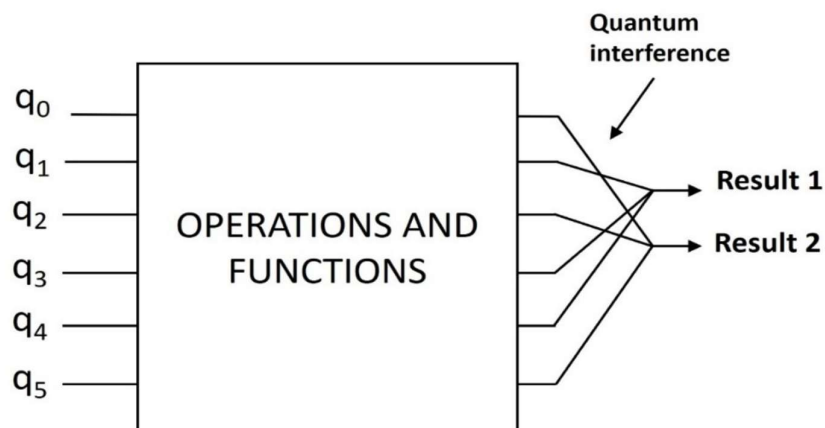
Note: Even though the same trend is observed in the classical computers case, not all the states are available for computation at the same time.

The concept of quantum computer:

Quantum computers would work under the principle of quantum mechanical rules such as

- Quantum superposition,
- Quantum interference
- Quantum entanglement

Because of such powerful rules, quantum computers can perform parallel computation. Particularly the quantum interference would vanish few of the unwanted states and arrive at multiple results obtained from multiple processing at the same time using a single quantum processor as can be seen in the below figure. Such parallel computation is impossible in classical computers to perform using a single processor.



Therefore, parallel computation feature of quantum computer has made it more powerful than any other classical computers. Because the process of quantum computers would play multi processors role during computation.

Quantum superposition

Quantum superposition is a fundamental principle of quantum mechanics that states a quantum particle can exist in multiple states at the same time until it is measured. When a measurement is performed, the particle collapses into one of the possible states.

Example

A qubit in quantum computing can be in state $|0\rangle$, state $|1\rangle$, or in a combination of both states at the same time:

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

where α and β are complex probability amplitudes.

Quantum interference:

Quantum interference arises from the wave-like nature of quantum particles such as electrons or photons. When a particle is in a superposition of multiple states, these states can interfere with each other leading to constructive or destructive interference. In this case, the probability amplitudes of quantum states combine, resulting in a probabilistic outcome. Qubit states are manipulated through quantum gates by manipulating their probability amplitudes. Quantum interference has been experimentally verified through the double-slit experiment.

Quantum entanglement

We have discussed the concepts of quantum superposition between different states of qubits. This superposition is the cause of the interference of the states. Now, we will discuss quantum entanglement. Another important and unique characteristic of quantum bit is entanglement, a phenomenon where the state of one qubit is dependent on the state of another, regardless of the distance between them. The act of calculating one decides the result of calculating the other even if they are far from each other.

In simple terms, an entangled system refers to a group of particles whose quantum states are linked together in a way that they cannot be described independently of each other. Even when separated by a large distance, measuring the property of one entangled particle instantaneously influences the state of the other. This interconnectedness is a fundamental property of quantum mechanics and has no classical analog.

Definition: Quantum entanglement is a quantum phenomenon where two or more particles become linked together in such a way that they share a quantum state, regardless of the distance separating them.

Quantum computing system for information processing:

a) A SQUID based qbit

SQUID refers to Superconducting quantum interferometer device. This device is sensitive to tiny amount of magnetic field. With such sensitivity people have used it to process quantum data. This consists of a ring with two Josephson junctions. Each Josephson junction is made by introducing a thin insulating material in between two superconducting segments as shown in the figure which acts as an impossible barrier for Cooper pair to be hopping across.

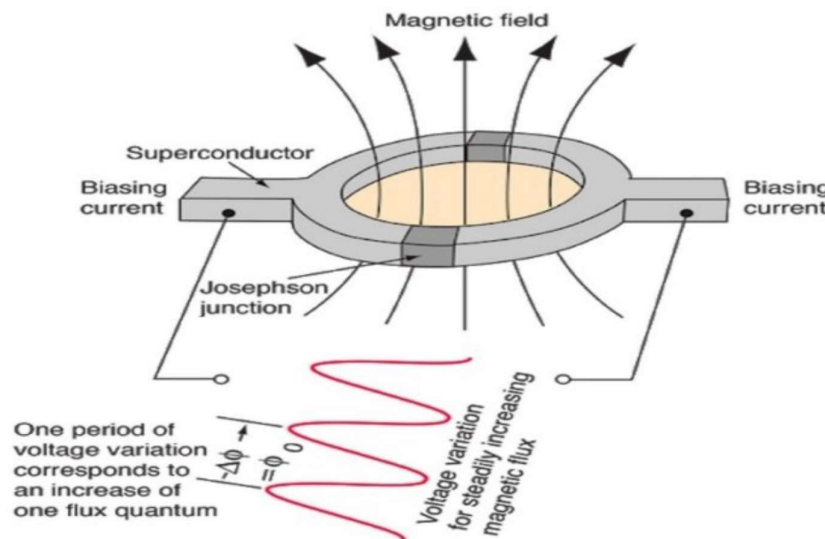


Fig. A SQUID based qbit for quantum computing

Cooper pairs can tunnel through it from one superconductor to the other one if a suitable magnetic field is applied to the ring perpendicular to its plan. Because of the same, there will be change in current density in the coil at the two Josephsen junctions when the ring is carrying supercurrent in it. Direction of presence of current will be considered as the paramter for representing two distintive states $|0\rangle$ and $|1\rangle$ in SQUID based quantum

computers as an example. If there is current in clockwise direction it is considered as $|0\rangle$ state and otherway around for the $|1\rangle$ state. Current in both the direction at the same time is nothing but superposition state for the qubit. In the same way all other features required for computation will be achieved.

(b) Photon qubits

Encoding of quantum information in photons-based qubits is through their polarization properties. Polarization refers to the orientation of the electric field vector associated with a photon's electromagnetic wave.

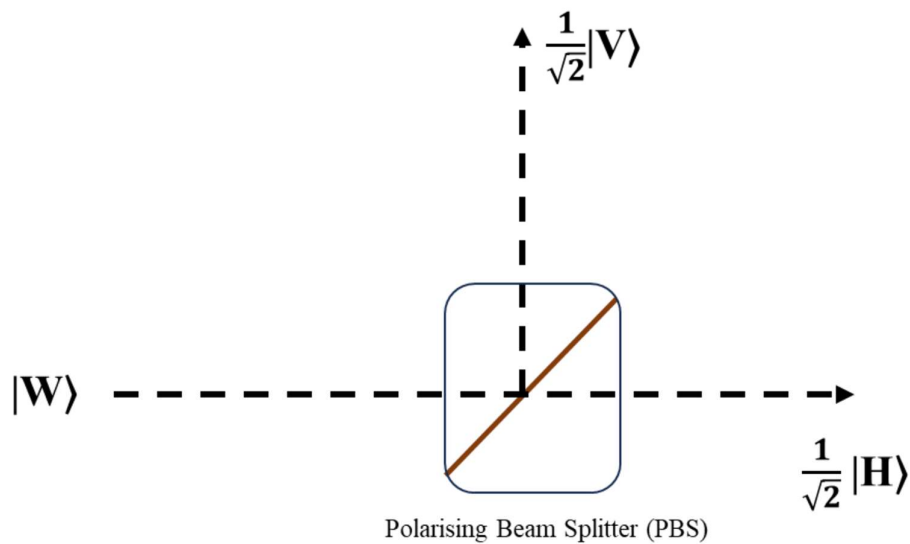


Fig: A photon based qbit for quantum computing

Polarization can be represented using various bases, such as the horizontal $|H\rangle$ and vertical $|V\rangle$ basis to represent $|0\rangle$ and $|1\rangle$ respectively, the diagonal basis for superposition states. Here, the quantum superposition is defined by the ability of a photon to exist in horizontal and vertical polarization states simultaneously as shown in the figure. Quantum entanglement will be achieved when two or more photons become correlated in such a way that the state of one photon is dependent on the state of another, even when they are separated by large distances. When such photonics-based qubits are measured their superposition state collapses into either $|0\rangle$ or $|1\rangle$ with a certain probability. This measurement process is a crucial step in extracting information from quantum systems for computation.

(c) Nuclear magnetic resonance (NMR) based qbit

In nuclear magnetic resonance (NMR) based quantum computing, superposition and entanglement are achieved using the properties of nuclear spins within molecules placed in a strong external magnetic field. Superposition: Manipulation of nuclear spins of certain atoms within a molecule is used for computation. Such nucleus with spins can act like tiny magnets aligning themselves parallel or antiparallel to the direction of the applied external magnetic field. These two directions will be used as two distinctive states to represent $|0\rangle$ and $|1\rangle$.

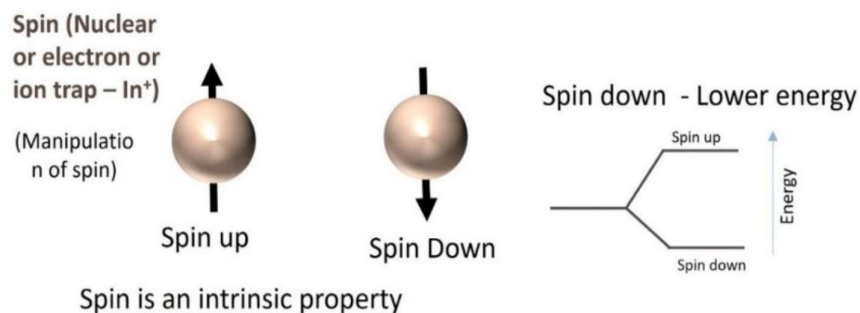


Fig: NMR based qbit for quantum computing

A Suitable radiofrequency (RF) pulse with the specific frequency and pulse duration, nuclear spins will be manipulated from their initial state into a superposition state. This is simply aligning them parallel and antiparallel at the same time. Quantum entangled state will be achieved when multiple qubits are aligned to the same magnetic

(d) Ion trapped qubits

In ion trap quantum computing, superposition is achieved by manipulating the internal energy levels of trapped ions using laser pulses and electromagnetic fields. Ion trap qubits are typically based on the electronic states of ions, such as trapped ions in suitable elements. The first step is to trap individual ions using electromagnetic fields in a vacuum chamber. The ions are typically held in place using a combination of radiofrequency (RF) and static electric fields. Cooling techniques, such as laser cooling, are then applied to bring the ions' motion to the ground state, reducing their thermal motion and interaction with the environment. The ground state and an excited state of the ion can often be used as the two basis states of your qubit. The two states could be represented as $|0\rangle$ (ground state) and $|1\rangle$ (excited state). If the ion is at ground state, the qubit is at $|0\rangle$ state. Once the trapped ion is taken to higher

energy, it gives electron for conduction which is considered as $|1\rangle$ state. The ion is at ground and excited states at the same state will make the quantum superposition of the qubit.

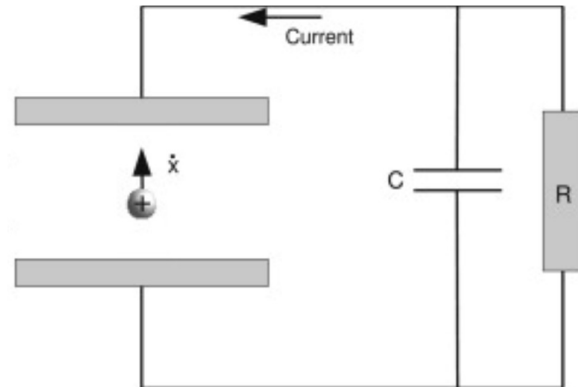


Fig: An ion trapped qbit for quantum computing

Evolution of quantum systems

Physical system changes in time, and so the state vector $|\psi\rangle$ of a system will actually be a function of time, $|\psi(t)\rangle$. If we consider a state

$$|\psi(0)\rangle = |0\rangle$$

then the value of $\alpha = 1$ and value of $\beta = 0$.

If we consider a state

$$|\psi(t)\rangle = \frac{1}{\sqrt{2}}(|0\rangle + i|1\rangle)$$

then the value of $\alpha = \frac{1}{\sqrt{2}}$ and value of $\beta = \frac{i}{\sqrt{2}}$.

Here from these two examples, we can observe that the state has evolved from

$|\psi\rangle = |0\rangle$ to $|\psi\rangle = \frac{1}{\sqrt{2}}(|0\rangle + i|1\rangle)$ in some time, t . When a system evolves, there is an indication of change in probability and relative phase. However, the sum of the probabilities must be 1. This type of evolution can be induced by an operator called unitary operator.

Hence, the time-evolution of the state of quantum system is described by a unitary operator.

That is, for any evolution of the system there exists a unitary operator U such that if the initial state of the system is $|\psi_1\rangle$, then after the evolution the state of the system will be

$$|\psi_2\rangle = U |\psi_1\rangle$$

Evolution in a quantum system means how a quantum state changes with time by changing the phases and sometimes the probabilities of its superposed components. This change is produced by a unitary operator, which ensures that the total probability of the system always remains equal to 1.

A unitary operator 'U' satisfies:

$$U^\dagger U = I$$

$U^\dagger$ is the conjugate transpose.

Unitary operations preserve total probability, i.e., $|\alpha|^2 + |\beta|^2 = 1$.

For Example:

Apply U to a qubit

Take an initial qubit:

$$|\psi\rangle = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \text{ (pure } |0\rangle \text{ state)}$$

Apply U :

$$|\psi'\rangle = U |\psi\rangle = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & i \\ i & 1 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \end{bmatrix} = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 \\ i \end{bmatrix}$$

The new state has **both amplitudes changed** (from $[1, 0]$ to $[\frac{1}{\sqrt{2}}, \frac{i}{\sqrt{2}}]$)

The relative **phase** is now $\frac{\pi}{2}$ (from 0 to $i = e^{i\pi/2}$)

Probability and phase change

Probability of measuring $|0\rangle$: $P_0 = |\alpha'|^2 = \left|\frac{1}{\sqrt{2}}\right|^2 = 0.5$

Probability of measuring $|1\rangle$: $P_1 = |\beta'|^2 = \left|\frac{i}{\sqrt{2}}\right|^2 = 0.5$

Phase change $\frac{\pi}{2}$ which is visible in the complex number ($i/\sqrt{2}$) (originally it was 0).

Quantum measurement:

The evolution of systems which don't interact with the rest of the world is all very well and it is closed, but there must also be times when the experimentalist and their experimental equipment – an external physical system in other words – observes the system to find out what is going on inside the system, an interaction which makes the system no longer closed, and thus not necessarily subject to unitary evolution. To explain what happens when this

is done, we will discuss here, which provides a means for describing the effects of measurements on quantum systems.

Quantum measurement is different from classical measurement. When we measure a quantum state, the result is probabilistic, and the state of the system changes after measurement (collapses). To describe measurements mathematically, we use a set of measurement operators $\{M_m\}$. Each operator corresponds to a possible measurement outcome m .

If the system is in a state:

$$|\psi\rangle$$

Then the probability of obtaining outcome m when we measure is given by:

$$p(m) = \langle \psi | M_m^\dagger M_m | \psi \rangle$$

If outcome m is observed, the state collapses to:

$$\frac{M_m |\psi\rangle}{\sqrt{\langle \psi | M_m^\dagger M_m | \psi \rangle}}$$

The denominator ensures the new state is **normalized** (total probability = 1).

The measurement operators must follow:

$$\sum_m M_m^\dagger M_m = I$$

This ensures that the sum of all possible outcome probabilities is 1:

$$\sum_m p(m) = 1$$

Example:

Some important measurement operators are

$$M_0 = |0\rangle\langle 0| \text{ and } M_1 = |1\rangle\langle 1|$$

$$M_0 = \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix} M_1 = \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix}$$

Observe that:

$$M_0^\dagger M_0 + M_1^\dagger M_1 = I$$

and are thus complete.

$$|\psi\rangle = a|0\rangle + b|1\rangle$$

$$p(0) = \langle\psi|M_0^\dagger M_0|\psi\rangle$$

Note that $M_0^\dagger M_0 = M_0$, hence

$$p(0) = \langle\psi|M_0|\psi\rangle = [a^*, b^*] \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} a \\ b \end{bmatrix} = [a^*, b^*] \begin{bmatrix} a \\ 0 \end{bmatrix} = |a|^2$$

Hence the probability of measuring $|0\rangle$ is related to its probability amplitude a by way of $|a|^2$.

It is important to note that the state after measurement is related to the outcome of the measurement. For example, suppose $|0\rangle$ was measured, then the state of the system after this measurement is re-normalized as:

$$\frac{M_0|\psi\rangle}{|a|} = \frac{a}{|a|}|0\rangle$$

After measurement, we are no longer dealing with a *probabilistic mixture of two states*. We are left with one definite state (either $|0\rangle$ or $|1\rangle$). And a definite physical state must always have total probability = 1. So, renormalization makes the remaining state a valid quantum state.

Let's understand the reason for renormalization.

Before measurement, we have superposition:

$$|\psi\rangle = a|0\rangle + b|1\rangle, \text{ with } |a|^2 + |b|^2 = 1$$

There is a probability spread across two basis states.

After measurement, suppose result = 0:

State collapses to:

$$|\phi\rangle = a|0\rangle$$

But check its norm:

$$\|a|0\rangle\| = |a| \text{ (usually } < 1)$$

After choosing one outcome (say 0), the remaining probability is $|a|^2$, which is usually less than 1. So we rescale it back to total probability = 1.

So $|\phi\rangle$ is **not** a valid quantum state now after measurement as it is not normalized. Quantum mechanics demands:

$$\| \text{state} \| = 1$$

Therefore, we must divide by $|a|$:

$$|\psi'\rangle = \frac{a}{|a|} |0\rangle$$

Now norm:

$$\| \frac{a}{|a|} |0\rangle \| = 1$$

This means, we have now obtained the system definitely in state $|0\rangle$ with probability 1.

Quantum Gates:

A quantum gate is a very simple computing device that performs quantum operation on qubits. Quantum gates are one of the essential parts of a quantum computer and are the building blocks of all quantum algorithms.

Quantum gates are mathematically represented as transformation matrices which operate on inputs to give outputs.

There are different types of quantum gates. Single-qubit gates and multiple qubit gates. These gates can flip a qubit from 0 to 1 as well as allowing superposition states to be created.

Single-Qubit Gates:

Single qubit inputs are $|0\rangle$ to $|1\rangle$ and can be represented by matrix forms as

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

Single Qubit Gates are X- gate, Y- gate, Z- gate, H –gate, S- gate, T- gate.

1. X – Gate or Quantum Not Gate

X- gate is the single qubit input gate and is also called as Pauli X – gate or quantum NOT gate.

The Matrix form of X is given by

The matrix form of the X gate is given by

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

Action of the X-gate on inputs:

When the X gate operates on inputs $|0\rangle$, $|1\rangle$, $|\psi\rangle$:

When X operates on $|0\rangle$ and $|1\rangle$, the output will be inverted (i.e., $|0\rangle$ becomes $|1\rangle$ and $|1\rangle$ becomes $|0\rangle$).

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

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

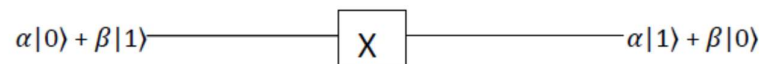
Since X inverts each input, it is also called the **bit-flip gate**.

Action on a superposed state

If a superposed qubit goes through an X gate, the result will be:

$$X|\psi\rangle = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} \alpha \\ \beta \end{bmatrix} = \begin{bmatrix} \beta \\ \alpha \end{bmatrix} = \alpha|1\rangle + \beta|0\rangle$$

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



Truth table is

X -gate	
Input	Output
$ 0\rangle$	$ 1\rangle$
$ 1\rangle$	$ 0\rangle$
$\alpha 0\rangle + \beta 1\rangle$	$\alpha 1\rangle + \beta 0\rangle$

2. Y-Gate

The Y-gate is a single-qubit gate. It is also called the Pauli-Y gate.

The matrix form of the Y-gate is

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

Action of Y-gate on inputs

When Y operates on $|0\rangle$ and $|1\rangle$:

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

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

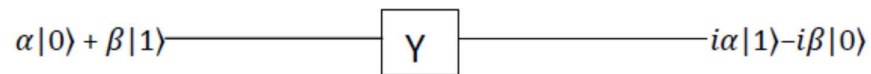
Action on a superposed state

If a superposed qubit goes through the Y-gate, the result will be:

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

$$Y|\psi\rangle = -i\beta|0\rangle + i\alpha|1\rangle$$

Gate representation



Truth table is

Y -gate	
Input	Output
$ 0\rangle$	$i 1\rangle$
$ 1\rangle$	$-i 0\rangle$
$\alpha 0\rangle + \beta 1\rangle$	$i\alpha 1\rangle - i\beta 0\rangle$

3. Z-Gate

Z-gate is the single-qubit input gate. This is also called the **Pauli-Z gate**.

The matrix form of Z-gate is

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

Action of Z-gate on inputs

When Z operates on $|0\rangle$ and $|1\rangle$, the phase will change.

Hence, this is also called the **phase-flip gate**.

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

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

Action on a superposed state

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

$$Z|\psi\rangle = \alpha|0\rangle - \beta|1\rangle$$

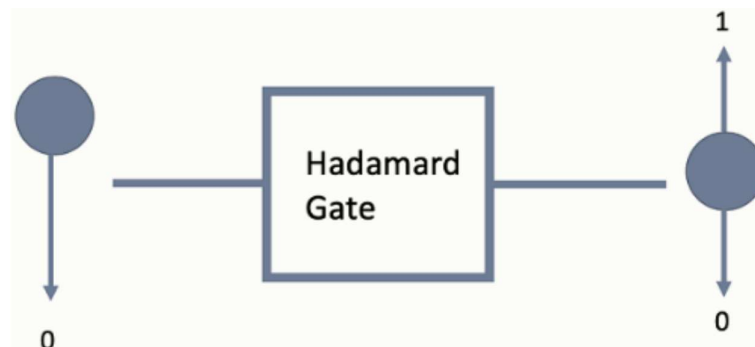
Gate representation

$$\alpha|0\rangle + \beta|1\rangle \rightarrow \boxed{Z} \rightarrow \alpha|0\rangle - \beta|1\rangle$$

Z-gate	
Input	Output
$ 0\rangle$	$ 0\rangle$
$ 1\rangle$	$- 1\rangle$
$\alpha 0\rangle + \beta 1\rangle$	$\alpha 0\rangle - \beta 1\rangle$

4. Hadamard Gate (H-gate) – It is a single qubit gate and is also gate to superposition. H gate acts on single qubit input and produce superposition state output. Matrix form of H gate and its symbol is

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



Let us find out what happens when Hadamard gate operates on a qubit that is in the $|0\rangle$ state

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

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

Let us find out what happens when Hadamard gate operates on a qubit that is in the $|1\rangle$ state

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

$$H|1\rangle = \frac{|0\rangle - |1\rangle}{\sqrt{2}}$$

If a superposed qubit goes through H gate, the result will be

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

$$H|\psi\rangle = \alpha \left(\frac{|0\rangle + |1\rangle}{\sqrt{2}} \right) + \beta \left(\frac{|0\rangle - |1\rangle}{\sqrt{2}} \right)$$

Gate representation

$$\alpha|0\rangle + \beta|1\rangle \xrightarrow{H} \alpha \left(\frac{|0\rangle + |1\rangle}{\sqrt{2}} \right) + \beta \left(\frac{|0\rangle - |1\rangle}{\sqrt{2}} \right)$$

The truth table is as follows

Input	Output
$ 0\rangle$	$\frac{ 0\rangle + 1\rangle}{\sqrt{2}}$
$ 1\rangle$	$\frac{ 0\rangle - 1\rangle}{\sqrt{2}}$
$\alpha 0\rangle + \beta 1\rangle$	$\alpha \left(\frac{ 0\rangle + 1\rangle}{\sqrt{2}} \right) + \beta \left(\frac{ 0\rangle - 1\rangle}{\sqrt{2}} \right)$

Note: Difference between X, Y, Z and H gates is that in X, Y and Z gates, output is in single state whereas in H gate output is superposed state.

5. Phase Gate (S Gate):

It is a single qubit gate. The Phase gate or S gate is a gate that transfers $|0\rangle$ into $|0\rangle$ and $|1\rangle$ into $i|1\rangle$. The matrix form of S gate is

$$S = \begin{bmatrix} 1 & 0 \\ 0 & i \end{bmatrix}$$

Action of the S gate on inputs

1. Applying S gate to state $|0\rangle$

Consider S gate apply to a state $|0\rangle$, it will remain the same:

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

So,

$$S|0\rangle = |0\rangle$$

Applying S gate to state $|1\rangle$

If S gate apply to a state $|1\rangle$, it will be transformed into $i|1\rangle$:

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

Thus,

$$S|1\rangle = i|1\rangle$$

3. Applying S gate to a general state $\alpha|0\rangle + \beta|1\rangle$

$$S|\psi\rangle = \begin{bmatrix} 1 & 0 \\ 0 & i \end{bmatrix} \begin{bmatrix} \alpha \\ \beta \end{bmatrix} = \begin{bmatrix} \alpha \\ i\beta \end{bmatrix}$$

So,

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

S Gate Representation is as follows

$$\alpha|0\rangle + \beta|1\rangle \rightarrow \boxed{S} \rightarrow \alpha|0\rangle + i\beta|1\rangle$$

Input → S Gate → Output

The truth table is as follows

Input	Output
$ 0\rangle$	$ 0\rangle$
$ 1\rangle$	$i 1\rangle$
$\alpha 0\rangle + \beta 1\rangle$	$\alpha 0\rangle + i\beta 1\rangle$

6. T- Gate

T- Gate: It is a single qubit gate and it is also called $\pi/8$ gate.

Its matrix form is $T = \begin{bmatrix} 1 & 0 \\ 0 & e^{i\pi/4} \end{bmatrix}$

Action of T gate on inputs

1. If T gate operates on input $|0\rangle$, then the output is also $|0\rangle$

$$T|0\rangle = \begin{bmatrix} 1 & 0 \\ 0 & e^{i\pi/4} \end{bmatrix} \begin{bmatrix} 1 \\ 0 \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \end{bmatrix} = |0\rangle$$

So,

$$T|0\rangle = |0\rangle$$

2. If T gate operates on input $|1\rangle$, then the output is $e^{i\pi/4}|1\rangle$

$$T|1\rangle = \begin{bmatrix} 1 & 0 \\ 0 & e^{i\pi/4} \end{bmatrix} \begin{bmatrix} 0 \\ 1 \end{bmatrix} = \begin{bmatrix} 0 \\ e^{i\pi/4} \end{bmatrix} = e^{i\pi/4}|1\rangle$$

Thus,

$$T|1\rangle = e^{i\pi/4}|1\rangle$$

3. If T gate operates on a superposed state $\alpha|0\rangle + \beta|1\rangle$

It transforms to:

$$T|\psi\rangle = \begin{bmatrix} 1 & 0 \\ 0 & e^{i\pi/4} \end{bmatrix} \begin{bmatrix} \alpha \\ \beta \end{bmatrix} = \begin{bmatrix} \alpha \\ e^{i\pi/4}\beta \end{bmatrix}$$

So,

$$T | \psi \rangle = \alpha | 0 \rangle + e^{i\pi/4} \beta | 1 \rangle$$

T Gate Representation

$$\alpha | 0 \rangle + \beta | 1 \rangle \rightarrow \boxed{T} \rightarrow \alpha | 0 \rangle + e^{i\pi/4} \beta | 1 \rangle$$

The truth table is as follows

Input	Output
$ 0\rangle$	$ 0\rangle$
$ 1\rangle$	$e^{i\pi/4} 1\rangle$
$\alpha 0\rangle + \beta 1\rangle$	$\alpha 0\rangle + e^{i\pi/4}\beta 1\rangle$

Multiple Qubit gates: Quantum gates operating on multiple qubits are called as multiple qubit gates. Multiple Qubit Gates operate on Two or More input Qubits. Multiple qubit consists of control gate and target gate. The action of gate as follows;

- i) The Target qubit is altered only when the control qubit is $|1\rangle$, and
- ii) The control qubit remains unaltered during the transformations.

For two qubits, inputs qubits are $|00\rangle, |01\rangle, |10\rangle$ and $|11\rangle$

For three qubits, inputs qubits are $|000\rangle, |001\rangle, |010\rangle, |011\rangle, |100\rangle, |101\rangle, |110\rangle$ and $|111\rangle$

In multiple qubit gate, the input qubit applied in the form like $|AB\rangle$, first term $|A\rangle$ goes to control qubit and the second term $|B\rangle$ goes to target qubit.

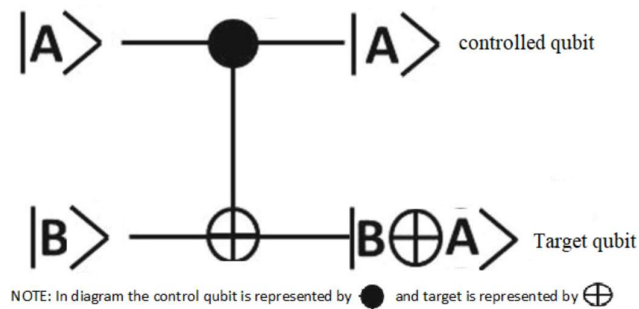
Some of the multiple qubit gate are as follows;

- Controlled gate (CNOT gate),
- Swap gate,
- Controlled Z gate and
-

1. Controlled Gate (CNOT)

The CNOT gate is a two-qubit operation, where the first qubit is referred as the control qubit $|A\rangle$ and the second qubit as the target qubit $|B\rangle$. If the control qubit is $|1\rangle$ then it will flip the target qubit state from $|0\rangle$ to $|1\rangle$ or from $|1\rangle$ to $|0\rangle$. When the control qubit is in state $|0\rangle$ then the target qubit remains unchanged.

The symbolic representation is as follows. The upper line represents control qubit and bottom line represents target qubit.



CNOT Gate

In the combined qubit, the first term is the **control qubit** and the second term is the **target qubit**.

For example, in $|AB\rangle$, A is the control qubit and B is the target qubit.

Matrix form of the CNOT Gate

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

Transformation Rule

$$|A, B\rangle \rightarrow |A, B \oplus A\rangle$$

(where $\oplus$ represents XOR)

Action of the gate

Consider the operations of CNOT gate on the four basis state inputs:

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

i) Operation of CNOT gate for input $|00\rangle$

$$|00\rangle \rightarrow |00\rangle$$

Here, control qubit is $|0\rangle$. Hence, no change in the state of the target qubit $|0\rangle$.

Operation of CNOT Gate for input $|01\rangle$: $|01\rangle \rightarrow |01\rangle$

Here, control qubit is $|0\rangle$. Hence no change in the state of Target qubit $|1\rangle$

iii) Operation of CNOT Gate for input $|10\rangle$: $|10\rangle \rightarrow |11\rangle$

Here, control qubit is $|1\rangle$. Hence the state of Target qubit flips from $|0\rangle$ to $|1\rangle$.

iv) Operation of CNOT Gate for input $|11\rangle$: $|11\rangle \rightarrow |10\rangle$

Here control qubit is $|1\rangle$. Hence the state of Target qubit flips from $|1\rangle$ to $|0\rangle$.

Input		Output	
$ A\rangle$	$ B\rangle$	$ A\rangle$	$ B \oplus A\rangle$
$ 0\rangle$	$ 0\rangle$	$ 0\rangle$	$ 0\rangle$
$ 0\rangle$	$ 1\rangle$	$ 0\rangle$	$ 1\rangle$
$ 1\rangle$	$ 0\rangle$	$ 1\rangle$	$ 1\rangle$
$ 1\rangle$	$ 1\rangle$	$ 1\rangle$	$ 0\rangle$

The Truth Table of operation of CNOT gate is as follows.

Input	Output
$ 00\rangle$	$ 00\rangle$
$ 01\rangle$	$ 01\rangle$
$ 10\rangle$	$ 11\rangle$
$ 11\rangle$	$ 10\rangle$

2. SWAP Gate

SWAP gate is a two-qubit operation gate and swaps the state of the two qubits involved in the operation.

It contains **3 CNOT gates**.

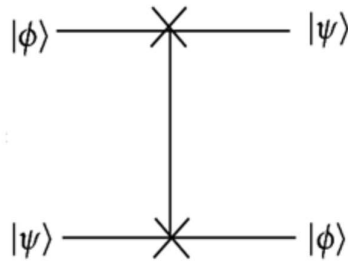
Matrix representation of the SWAP Gate

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

The schematic symbol of the SWAP gate circuit and its equivalent is shown in the figure below.

The swap gate is a combined circuit of 3 CNOT gates (1st CNOT gate, 2nd CNOT gate and 3rd CNOT gate) and the overall effect is that two input qubits are swapped at the output.

Gate representation is



Truth table of the swap gate

Input	Output
$ 00\rangle$	$ 00\rangle$
$ 01\rangle$	$ 10\rangle$
$ 10\rangle$	$ 01\rangle$
$ 11\rangle$	$ 11\rangle$

3. Controlled Z Gate

In Controlled Z Gate, the operation of Z Gate is controlled by a Control Qubit. If the control Qubit is $|A\rangle = |1\rangle$ then only the Z gate transforms the Target Qubit $|B\rangle$ as per the Pauli-Z operation. The action of Controlled Z-Gate could be specified by a matrix as follows.

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

The controlled Z gate representation is as follows:

(figure showing CZ gate with control on top qubit and Z on bottom qubit)

Action of controlled Z gate is

a) Operation of the Gate for input $|00\rangle$:

$$|00\rangle \rightarrow |00\rangle$$

Control qubit is $|0\rangle$, Z gate remains unchanged.

b) Operation of the Gate for input $|01\rangle$:

$$|01\rangle \rightarrow |01\rangle$$

Control qubit is $|0\rangle$, Z gate remains unchanged.

c) Operation of the Gate for input $|10\rangle$:

$$|10\rangle \rightarrow |10\rangle$$

Control qubit is $|1\rangle$, Z gate remains unchanged.

d) Operation of the Gate for input $|11\rangle$:

$$|11\rangle \rightarrow -|11\rangle$$

Control qubit is $|1\rangle$, Z gate flips the state from $|1\rangle$ to $-|1\rangle$.

Truth table of the controlled Z gate

Input	Output
$ 00\rangle$	$ 00\rangle$
$ 01\rangle$	$ 01\rangle$
$ 10\rangle$	$ 10\rangle$
$ 11\rangle$	$- 11\rangle$

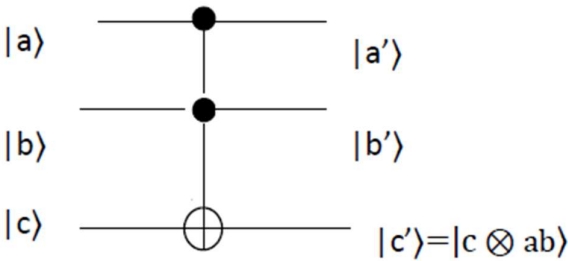
4. Toffoli Gate (CCNOT Gate):

The Toffoli Gate is also known as CCNOT Gate (Controlled-Controlled-Not). It has three inputs out of which two are Control Qubits and one is the Target Qubit. The Target Qubit flips only when both the Control Qubits are $|1\rangle$. The two Control Qubits are not altered during the operation.

The matrix representation of the Gate is:

$$\text{Toffoli Gate} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{pmatrix}$$

Gate representation:



Truth Table of Toffoli Gate.

Input to the gate $ abc\rangle$	Output of the gate $ a'b'c'\rangle$
$ 100\rangle$	$ 100\rangle$
$ 001\rangle$	$ 001\rangle$
$ 010\rangle$	$ 010\rangle$
$ 011\rangle$	$ 011\rangle$
$ 100\rangle$	$ 100\rangle$
$ 101\rangle$	$ 101\rangle$
$ 110\rangle$	$ 111\rangle$
$ 111\rangle$	$ 110\rangle$

Challenges and advantages of quantum computing over classical computation:

Advantages and Challenges of Quantum Computing

(a) Faster Computations

Quantum computers can evaluate multiple possibilities at the same time due to **superposition**. This makes them extremely fast for certain tasks. For example:

- Searching unsorted databases
- Simulating molecular structures
- Solving mathematical optimization problems

Thus, problems that might take **years** on classical computers could be solved in **minutes or seconds** by quantum computers.

(b) Enhanced Optimization

Optimization problems occur in many industries such as transportation, supply chain management, banking, and telecommunications. Quantum algorithms such as the **Quantum Approximate Optimization Algorithm (QAOA)** can find the best possible solution among millions of combinations efficiently.

For example:

- Finding the most efficient delivery routes (logistics)
- Portfolio optimization in finance
- Resource scheduling

c. Improved Security and Cryptography

Quantum computers can create highly secure cryptographic systems using **Quantum Key Distribution (QKD)**. It is almost impossible to intercept or crack such keys without disturbing the quantum state, which makes communication more secure.

d. Artificial Intelligence and Machine Learning

Quantum computing can speed up the training of machine learning models and improve data analysis. This could have big benefits in healthcare, robotics, and big-data processing.

e) Efficient Simulations:

Quantum computers are built to handle complex simulations, such as molecular interactions and quantum systems. By mimicking quantum phenomena, quantum computers can give accurate insights into chemical reactions, drug discovery, and material design. (a) For example, in industries like pharmaceuticals, energy, and climate science, quantum computers can simulate molecules at the atomic level. (b) Quantum computers with 50-100 qubits could precisely simulate complex enzyme mechanisms and chemical processes for the industrial production of fertilizers, polymers and biomass.

(f) Parallel Processing:

Quantum computers do parallel processing and are helpful for tasks like data analysis, optimization, and simulations. Parallelism greatly reduces computation time.

g) Healthcare advancement:

Quantum computing can transform healthcare and the life sciences through precise modelling of protein folding mechanisms, gene sequencing, disease pathology simulations, and personalised medicine.

(h) Climate change science:

By assimilating large datasets using quantum neural networks, NASA plans to model climate at high resolution at regional and global scales.

Challenges of Quantum Computing

Challenges of Quantum Computing

Although quantum computing is a promising and revolutionary technology, it is still in the early stages of development and faces many scientific, technological, and practical challenges. These challenges must be addressed before quantum computers can be widely usable and commercially available. The major challenges of quantum computing are discussed below:

1. Qubit Stability and Decoherence

Quantum computers use **qubits**, which can exist in superposition and entanglement states. However, qubits are extremely sensitive to environmental disturbances such as temperature changes, electromagnetic interference, and vibrations.

These disturbances cause **decoherence**, meaning the quantum state collapses and stored information is lost. Keeping qubits stable for sufficient time to perform calculations is one of the biggest challenges.

2. Error Rates and Error Correction

Quantum operations currently have a high probability of errors due to instability.

To correct errors, quantum error correction code is used, but it requires a large number of physical qubits to form a single logical qubit.

For example, **hundreds or even thousands of qubits may be needed to create one reliable qubit**, which makes the system extremely complex and difficult to scale.

3. Hardware and Engineering Challenges

Quantum computers need extremely controlled environments. They must be operated at:

- **Ultra-low temperatures** close to absolute zero (-273°C)
- **Vacuum conditions**
- **Noise-free electromagnetic shielding**

Creating and maintaining such conditions is scientifically complex and highly expensive. Only a few research laboratories can maintain such systems.

4. Scalability Issues

Current quantum computers have a limited number of qubits (tens or hundreds of qubits).

To solve large real-world problems, we need **millions of qubits**, which is far beyond present capability. Scaling up qubit systems while maintaining coherence is a major challenge.

5. Limited Algorithms and Software

Quantum programming languages, tools, and algorithms are still evolving. Only a few well-known algorithms exist, such as:

- Shor's algorithm (for factorization)
- Grover's search algorithm
- Quantum Approximate Optimization Algorithm (QAOA)

More research is required to design effective quantum algorithms that outperform classical methods.

6. High Cost and Limited Accessibility

Quantum computers are extremely expensive to build and maintain.

Currently, only large companies such as **IBM, Google, Microsoft, and government research labs** have access to real quantum machines.

This limits educational access and slows development progress.

7. Lack of Skilled Professionals

Quantum computing requires deep knowledge of:

- Quantum physics
- Mathematics
- Computer science
- Engineering

There are very few experts trained in these interdisciplinary areas, which slows down research and development.

Quantum algorithms:

(a) Deutsch algorithm

We are still in the early days of quantum computing, and we are yet to see real-world applications of it with real impact. But we are already seeing the first signs of quantum advantage, where quantum computers can outperform classical computers on certain tasks.

We will demonstrate quantum advantage by discussing Deutsch's algorithm in detail.

Note that this is a simple, contrived example without any real-world applications. However, it is a good starting point to understand the potential of quantum computing.

Deutsch's Algorithm:

Deutsch's algorithm (invented by [David Deutsch](#) in 1985) is an example of a quantum algorithm that can solve a problem **faster**, by some measure, than any classical algorithm.

Faster in this context means that the algorithm solves a very specific problem in **one** go (if we had a noise-free quantum computer). This is in contrast to a classical algorithm that we would have to run **twice** to solve the same problem.

The Problem Deutsch's Algorithm Solves is the following:

Deutsch's algorithm solves a simple problem:

Given a function $f(x)$ that takes a single bit (i.e. 0 or 1) as input and returns a single bit (i.e. 0 or 1) as output, determine if the function is constant or balanced.

- A constant function is one where the output is always 0 or always 1, regardless of the input.
- A balanced function is one where the output is 0 for one input and 1 for the other input.

In other words, we are given some function $f(x)$, which is unknown to us, but which we can call and observe the output, and **the problem is to determine if $f(x)$ is constant or balanced.**

As the function takes a single bit as input, there are only four possible functions.

1. The first 2 possible functions are constant:

1. $f(0) = 0$ and $f(1) = 0 \rightarrow f$ is constant (always 0)
2. $f(0) = 1$ and $f(1) = 1 \rightarrow f$ is constant (always 1)

Note: Here $f(0) = f(1)$

2. The other 2 possible functions are balanced:

1. $f(0) = 0$ and $f(1) = 1 \rightarrow f$ is balanced
2. $f(0) = 1$ and $f(1) = 0 \rightarrow f$ is balanced

Note: Here $f(0) \neq f(1)$

The problem is to determine if the function we are given is constant or balanced.

The Classical Solution involves 2 queries

It's easy to see that if we approach this problem classically, we would have to call the function twice to determine if it's constant or balanced.

For example, we can call the function with input 0 and learn that the output is $f(0) = 1$, we can tell that we are either in case 1.2 (constant) or case 2.2 (balanced).

But we would still have to call the function a second time with input 1 to distinguish between the two cases.

The Quantum Solution involves just 1 query

The Quantum solution (Deutsch's algorithm) can solve this problem in **one** go. As you might expect, the solution is a lot more involved than the classical one. But in a nutshell we will rely on the following principle:

- **Superposition:** We're putting a qubit in superposition, and then apply $f(x)$ on it. This allows us to evaluate $f(x)$ for both inputs at the same time. On a high level the algorithm works as follows:
- **Phase Kickback:** The effect where the phase of a qubit is "kicked back" onto another qubit during a controlled quantum operation like the CNOT gate.

The main steps involved in the algorithm are as follows:

- Prepare the qubits x and y in the state $|0\rangle|1\rangle$.
- Apply a Hadamard gate to both qubits to put them in superposition.
- Apply the gate U_f

Here U_f maps the state

$$U_f |x\rangle |y\rangle \text{ to } |x\rangle |y \oplus f(x)\rangle.$$

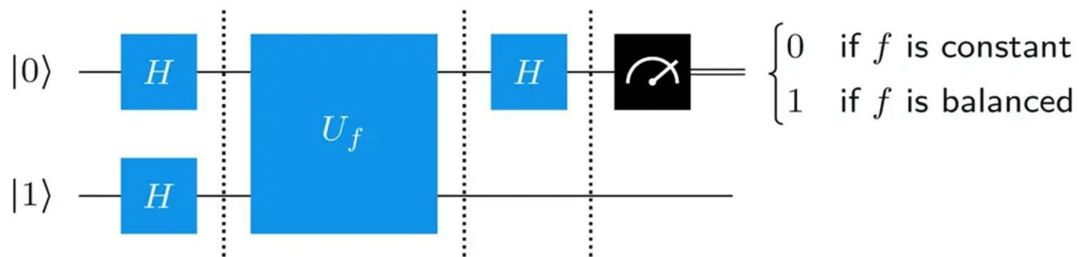
Remember that $\oplus$ is the XOR operator and $f(x)$ is one of the 4 cases defined above. The XOR operator, also known as exclusive OR, or addition modulo 2, is a binary operation **that takes two bits and returns 1 if the bits are different, and 0 if they are the same**. We sometimes use the symbol $\oplus$ to denote the XOR operator.

We can relate the effect of XOR operator as follows:

If $f(0) \oplus f(1) = 0$ we know that the function is constant

If $f(0) \oplus f(1) = 1$ we know that the function is balanced

- Apply a Hadamard gate to qubit x
- Measure qubit x



Let us analyze the algorithm to determine that it works correctly. Rather than taking the four cases separately, let us try to be more sophisticated and deal with the four cases all at the same time. The initial state is $|0\rangle |1\rangle$. Apply Hadamard gate on $|0\rangle$ and $|1\rangle$. Then the state $|0\rangle |1\rangle$ becomes after the first two Hadamard transforms is

$$\left(\frac{1}{\sqrt{2}} |0\rangle + \frac{1}{\sqrt{2}} |1\rangle\right) \left(\frac{1}{\sqrt{2}} |0\rangle - \frac{1}{\sqrt{2}} |1\rangle\right).$$

We can partially expand this state as

$$\frac{1}{2} |0\rangle(|0\rangle - |1\rangle) + \frac{1}{2} |1\rangle(|0\rangle - |1\rangle).$$

Performing the U_f operation transforms this state to

$$\begin{aligned} & \frac{1}{2} |0\rangle(|0 \oplus f(0)\rangle - |1 \oplus f(0)\rangle) + \frac{1}{2} |1\rangle(|0 \oplus f(1)\rangle - |1 \oplus f(1)\rangle) \\ &= \frac{1}{2} (-1)^{f(0)} |0\rangle(|0\rangle - |1\rangle) + \frac{1}{2} (-1)^{f(1)} |1\rangle(|0\rangle - |1\rangle) \\ &= \left(\frac{1}{\sqrt{2}} (-1)^{f(0)} |0\rangle + \frac{1}{\sqrt{2}} (-1)^{f(1)} |1\rangle\right) \left(\frac{1}{\sqrt{2}} |0\rangle - \frac{1}{\sqrt{2}} |1\rangle\right) \end{aligned}$$

Here we have used the fact that

$$|0 \oplus a\rangle - |1 \oplus a\rangle = (-1)^a(|0\rangle - |1\rangle)$$

for $a \in \{0,1\}$.

Notice that something seemingly strange has happened. The U_f transformation has apparently not changed the state of the second qubit; it has remained in the state

$$\frac{1}{\sqrt{2}}|0\rangle - \frac{1}{\sqrt{2}}|1\rangle.$$

There has been an important effect, however, which is that the factors $(-1)^{f(0)}$ and $(-1)^{f(1)}$ have appeared in the state of the first qubit. This phenomenon is sometimes known as “phase kick-back”. It is a commonly used trick in quantum algorithms.

At this point the second qubit is thrown in the trash, which should not bother us because we know that its state is completely independent from the state of the first qubit; we know that its state is $\frac{1}{\sqrt{2}}|0\rangle - \frac{1}{\sqrt{2}}|1\rangle$, regardless of the choice of f . This leaves the first qubit in the state

$$\frac{1}{\sqrt{2}}(-1)^{f(0)}|0\rangle + \frac{1}{\sqrt{2}}(-1)^{f(1)}|1\rangle,$$

Now, we can pull out $(-1)^{f(0)}$ and we can write it as

$$\left((-1)^{f(0)} \left(\frac{1}{\sqrt{2}}|0\rangle + \frac{1}{\sqrt{2}}(-1)^{f(0) \oplus f(1)}|1\rangle \right) \right).$$

where we used the fact that $(-1)^{f(1)-f(0)} = (-1)^{f(0) \oplus f(1)}$.

Apply H to the bracketed state (we can ignore the global $(-1)^{f(0)}$):

$$H\left(\frac{|0\rangle + (-1)^a|1\rangle}{\sqrt{2}}\right) = \frac{1}{\sqrt{2}}\left(\frac{|0\rangle + |1\rangle}{\sqrt{2}} + (-1)^a\frac{|0\rangle - |1\rangle}{\sqrt{2}}\right).$$

Combine the prefactors (overall factor $1/2$) and collect $|0\rangle, |1\rangle$ terms:

So the state after H is

$$\frac{1}{2}((1 + (-1)^a)|0\rangle + (1 - (-1)^a)|1\rangle).$$

Evaluate for $a = 0$ and $a = 1$

- If $a = 0$ (i.e. $f(0) = f(1)$, function **constant**):

$$1 + (-1)^0 = 2, \quad 1 - (-1)^0 = 0 \Rightarrow |0\rangle.$$

- If $a = 1$ (i.e. $f(0) \neq f(1)$, function **balanced**):

$$1 + (-1)^1 = 0, \quad 1 - (-1)^1 = 2 \Rightarrow |1\rangle.$$

Thus after the final Hadamard the first qubit is exactly $|a\rangle = |f(0) \oplus f(1)\rangle$

(Note: global phase ignored. Global phase has no physical effect. Hence we can neglect it. A **global phase** is a multiplication of a quantum state by a constant complex number of magnitude 1, usually $e^{i\theta}$, or simply ± 1 in our example.

Example global phase:

$$|\psi\rangle = |0\rangle \text{ and } |\psi'\rangle = -|0\rangle$$

These two states are physically **indistinguishable**, because **they produce the same measurement probabilities**

The probability of measuring each basis state is computed from the **square of amplitudes**, and

$$|-1|^2 = |1|^2 = 1$$

).

The final Hadamard transform takes this state to

$$|a\rangle = |f(0) \oplus f(1)\rangle$$

for $a \in \{0,1\}$, which is again easily verified by considering the cases $a = 0$ and $a = 1$. The measurement therefore results in the value $f(0) \oplus f(1)$ with certainty. This value is 0 if f is constant and 1 if f is balanced.

(b) Deutsch-Jozsa algorithm

The Deutsch-Jozsa Algorithm

Our next algorithm is a generalization of Deutsch's Algorithm called the Deutsch-Jozsa Algorithm. Although the computational problem being solved is still artificial and perhaps not particularly impressive, we are moving toward more interesting problems.

This time we assume that we are given a function $f: \{0,1\}^n \rightarrow \{0,1\}$, where n is some arbitrary positive integer.

We are given a function:

$$f: \{0,1\}^n \rightarrow \{0,1\}$$

Meaning of $\{0,1\}^n$

This represents all possible n -bit binary strings. For example:

1. If $n = 2$, $\{0,1\}^n = \{00,01,10,11\}$ (4 inputs).
2. If $n = 3$, $\{0,1\}^n = \{000,001, \dots, 111\}$ (8 inputs).

We are promised that one of two possibilities holds:

1. **f is constant.** In other words, either $f(x) = 0$ for all $x \in \{0,1\}^n$ or $f(x) = 1$ for all $x \in \{0,1\}^n$.
2. **f is balanced.** This means that the number of inputs $x \in \{0,1\}^n$ for which the function takes values 0 and 1 are the same:

$$|\{x \in \{0,1\}^n : f(x) = 0\}| = |\{x \in \{0,1\}^n : f(x) = 1\}| = 2^{n-1}.$$

The goal is to determine which of the two possibilities holds.

The transformation U_f acts like this:

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

where:

- $x \in \{0,1\}^n$ (n-bit input)
- $b \in \{0,1\}$ (1-bit auxiliary register)
- $\oplus$ is XOR addition mod 2

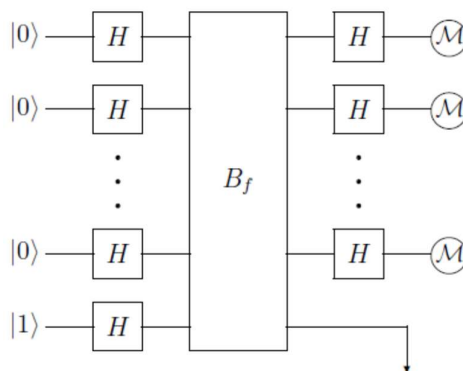
It turns out that classically this problem is pretty easy given a small number of queries if we allow randomness and accept that there may be a small probability of error.

Specifically, we can randomly choose say k inputs $x_1, \dots, x_k \in \{0,1\}^n$,

evaluate $f(x_i)$ for $i = 1, \dots, k$, and answer “constant” if $f(x_1) = \dots = f(x_k)$ and “balanced” otherwise.

If the function really was constant this method will be correct every time, and if the function was balanced, the algorithm will be wrong (and answer “constant”) with probability $2^{-(k-1)}$. Taking $k = 11$, say, we get that the probability of error is smaller than 1/1000. However, if you demand that the algorithm is correct every time, then $2^{n-1} + 1$ queries are needed in the worst case.

In the quantum case, 1 query will be sufficient to determine with certainty whether the function is constant or balanced. Here is the algorithm, which is called the **Deutsch-Jozsa Algorithm**:



There are n bits resulting from the measurements. If **all n measurement results are 0**, we conclude that the function was constant. Otherwise, if at least one of the measurement outcomes is 1, we conclude that the function was balanced.

Before we analyze the algorithm, it will be helpful to think more about Hadamard transforms. We have already observed that for $a \in \{0,1\}$ we have

$$H | a \rangle = \frac{1}{\sqrt{2}} | 0 \rangle + \frac{1}{\sqrt{2}} (-1)^a | 1 \rangle,$$

which we can also write as

$$H | a \rangle = \frac{1}{\sqrt{2}} \sum_{b \in \{0,1\}} (-1)^{ab} | b \rangle.$$

If we instead had two qubits, starting in state $| x \rangle$ for $x = x_1 x_2 \in \{0,1\}^2$, and applied Hadamard transforms to both, we would obtain

$$\begin{aligned} (H \otimes H) | x \rangle &= \left(\frac{1}{\sqrt{2}} \sum_{y_1 \in \{0,1\}} (-1)^{x_1 y_1} | y_1 \rangle \right) \left(\frac{1}{\sqrt{2}} \sum_{y_2 \in \{0,1\}} (-1)^{x_2 y_2} | y_2 \rangle \right) \\ &= \frac{1}{2} \sum_{y \in \{0,1\}^2} (-1)^{x_1 y_1 + x_2 y_2} | y \rangle. \end{aligned}$$

The pattern generalizes to any number of qubits. Writing $H^{\otimes n}$ to mean $H \otimes \cdots \otimes H$ (n times), we have

$$H^{\otimes n} | x \rangle = \frac{1}{\sqrt{2^n}} \sum_{y \in \{0,1\}^n} (-1)^{x_1 y_1 + \cdots + x_n y_n} | y \rangle$$

for every $x \in \{0,1\}^n$. We also use the shorthand

$$x \cdot y = \sum_{i=1}^n x_i y_i \pmod{2}$$

so that we may write

$$H^{\otimes n} | x \rangle = \frac{1}{\sqrt{2^n}} \sum_{y \in \{0,1\}^n} (-1)^{x \cdot y} | y \rangle.$$

The fact that $x \cdot y$ is defined modulo 2 is irrelevant for the previous expression (because $x \cdot y$ appears as an exponent of -1), but it will be convenient later to use this shorthand again and have the quantity defined modulo 2.

Using these formulas, the Deutsch–Jozsa Algorithm becomes easier to analyze. The state after the first layer of Hadamard transforms is performed is

$$\frac{1}{\sqrt{2^n}} \sum_{x \in \{0,1\}^n} |x\rangle \left(\frac{1}{\sqrt{2}} |0\rangle - \frac{1}{\sqrt{2}} |1\rangle \right).$$

When the B_f transformation is performed, the state will change to

$$\frac{1}{\sqrt{2^n}} \sum_{x \in \{0,1\}^n} (-1)^{f(x)} |x\rangle \left(\frac{1}{\sqrt{2}} |0\rangle - \frac{1}{\sqrt{2}} |1\rangle \right).$$

Once again we are seeing the **phase kick-back** effect. Now, the last qubit is discarded and n Hadamard transforms are applied. The resulting state is

$$\frac{1}{\sqrt{2^n}} \sum_{x \in \{0,1\}^n} (-1)^{f(x)} \left(\frac{1}{\sqrt{2^n}} \sum_{y \in \{0,1\}^n} (-1)^{x \cdot y} |y\rangle \right).$$

which simplifies to

$$\sum_{y \in \{0,1\}^n} \left(\frac{1}{2^n} \sum_{x \in \{0,1\}^n} (-1)^{f(x) + x \cdot y} \right) |y\rangle.$$

Now, all that we really care about is the probability that the measurements all give outcome 0.

The amplitude associated with the classical state $|0^n\rangle$ is

$$\frac{1}{2^n} \sum_{x \in \{0,1\}^n} (-1)^{f(x)}.$$

Thus, the probability that the measurements all give outcome 0 is

$$\left| \frac{1}{2^n} \sum_{x \in \{0,1\}^n} (-1)^{f(x)} \right|^2 = \begin{cases} 1, & \text{if } f \text{ is constant} \\ 0, & \text{if } f \text{ is balanced} \end{cases}$$

so the algorithm works as claimed.

Appendix:

For understanding some steps in the Duetsch-Josza algorithm

1. To understand the state after the first layer of Hadamard transforms is performed is:

The Deutsch-Josza initial state is

$$|0\rangle^{\otimes n} |1\rangle = |00 \dots 0\rangle |1\rangle.$$

Apply Hadamard to **each of the first n qubits** and also to the last (ancilla) qubit. Use two facts:

- For the first n qubits (all zeros):

$$H^{\otimes n} |0\rangle^{\otimes n} = \frac{1}{\sqrt{2^n}} \sum_{x \in \{0,1\}^n} |x\rangle.$$

This is a special case of the equation

$$H^{\otimes n} |x\rangle = \frac{1}{\sqrt{2^n}} \sum_{y \in \{0,1\}^n} (-1)^{x \cdot y} |y\rangle.$$

with $x = 0^n$. Indeed, $x \cdot y = 0$ so $(-1)^0 = 1$ for every y .)

- For the ancilla:

$$H |1\rangle = \frac{|0\rangle - |1\rangle}{\sqrt{2}}.$$

Tensor these together (product state $\rightarrow$ tensor product):

$$\begin{aligned} (H^{\otimes n} \otimes H)(|0\rangle^{\otimes n} \otimes |1\rangle) &= \left(\frac{1}{\sqrt{2^n}} \sum_{x \in \{0,1\}^n} |x\rangle \right) \otimes \left(\frac{|0\rangle - |1\rangle}{\sqrt{2}} \right) \\ &= \boxed{\frac{1}{\sqrt{2^n}} \sum_{x \in \{0,1\}^n} |x\rangle \left(\frac{|0\rangle - |1\rangle}{\sqrt{2}} \right)}. \end{aligned}$$

This is the expression: the first register is the uniform superposition over all $|x\rangle$, and the last qubit is $(|0\rangle - |1\rangle)/\sqrt{2}$.

2. Example for Deutsch-Josza Algorithm

to the case of **4 basis states**, i.e. $n = 2$ so $x \in \{00,01,10,11\}$ and $2^n = 4$.

Write the amplitude for $|00\rangle$ as

$$A_{00} = \frac{1}{4} ((-1)^{f(00)} + (-1)^{f(01)} + (-1)^{f(10)} + (-1)^{f(11)}).$$

The probability to measure $|00\rangle$ is $|A_{00}|^2$.

Now compute this for the important cases.

1) Constant function $f(x) = 0$ for all x

$$A_{00} = \frac{1}{4} (1 + 1 + 1 + 1) = \frac{4}{4} = 1 \Rightarrow P(|00\rangle) = |1|^2 = 1.$$

2) Constant function $f(x) = 1$ for all x

$$A_{00} = \frac{1}{4} ((-1) + (-1) + (-1) + (-1)) = \frac{-4}{4} = -1 \Rightarrow P(|00\rangle) = |-1|^2 = 1.$$

(Overall global phase -1 is irrelevant — measurement outcome still $|00\rangle$ with certainty.)

3) Any balanced function (two zeros and two ones)

If f is balanced, exactly two terms are $+1$ and two terms are -1 . So the sum is

$$(+1) + (+1) + (-1) + (-1) = 0,$$

therefore

$$A_{00} = 0 \Rightarrow P(|00\rangle) = 0.$$

Examples of balanced assignments (all give the same result $A_{00} = 0$):

- $f(00) = 0, f(01) = 0, f(10) = 1, f(11) = 1$:

$$A_{00} = \frac{1}{4}(1 + 1 - 1 - 1) = 0.$$

- $f(00) = 0, f(01) = 1, f(10) = 0, f(11) = 1$:

$$A_{00} = \frac{1}{4}(1 - 1 + 1 - 1) = 0.$$

- $f(00) = 1, f(01) = 0, f(10) = 1, f(11) = 0$:

$$A_{00} = \frac{1}{4}(-1 + 1 - 1 + 1) = 0.$$

(any pattern with two 0s and two 1s gives two $+1$ and two -1 in the sum $\rightarrow$ cancellation)

4) A mixed (non-constant, non-balanced) assignment — for completeness

If there were a function with three 0s and one 1 (not allowed by the Deutsch–Jozsa promise, but instructive):

Example: $f(00) = 0, f(01) = 0, f(10) = 0, f(11) = 1$:

$$A_{00} = \frac{1}{4}(1 + 1 + 1 - 1) = \frac{2}{4} = \frac{1}{2} \Rightarrow P(|00\rangle) = \left(\frac{1}{2}\right)^2 = \frac{1}{4}.$$

(Again: such functions are outside the DJ promise, but this shows intermediate amplitudes are possible.)

Conclusion (for the 4-state / $n = 2$ case)

- For **constant** f : $A_{00} = \pm 1 \rightarrow$ measure $|00\rangle$ with probability 1.
- For **balanced** f : $A_{00} = 0 \rightarrow$ probability of $|00\rangle$ is 0 (you will always get a non-zero string).