

Principles of Quantum information and Quantum Computing:

Computation basically is digital data processing. This requires the use of a computer which processes the data following certain set of instructions called a programme. Examples are, numerical data being processed by the executable version of a FORTRAN or C programme, text being edited by a word processor and a visual image being rendered by a graphics application. The input, intermediate and output data are internally expressed in terms of certain basic units called bits. Each bit has two possible values 0 and 1 and a string of such values corresponds to the binary representation of a number just like the more familiar decimal representation. The advantage of using the binary representation in a computer is that, it is relatively easy to construct devices that possess two clearly distinguishable states that may be used to represent the bit values. Examples are high and low voltage states of a capacitor and the two stable states of a flip-flop circuit. The devices used to represent bits behave essentially as classical systems, even though they may be inherently dependent on quantum phenomena for their operation. For example, a transistor used in a flip-flop circuit works on the

basis of the semiconducting properties of certain materials. This stems from the quantum mechanical energy band structure of electrons in those materials. However, because of the large number of electrons involved, quantum effects due to them add up incoherently to produce say, a current or voltage that behaves classically. The flip-flop or the capacitor, consequently exists in one of the two possible stable states and not in any arbitrary mixture of them. Thus, at any time, a processor using such devices to represent and store bits, can only process a particular set of data. In order to process several sets of data concurrently, one has to use several such processors and run parallel data channels through them. This is ordinary parallel processing.

Thus, to summarize, the laws of classical physics and mathematical logic underpin classical computing. Serial computation is the primary purpose of traditional computer software. It denotes that the logic flow occurs from one point to another in time. It means that one process is completed before beginning another. In parallel computation one splits the overall job among lots of classical computers, such that the processors can execute the job at the same time, and the result should be integrated. This no doubt leads to speed up as long as one

understands clearly which portions of the program can be parallelized and how the integration of the outputs can be done.

There are on the other hand intrinsically quantum systems which have two orthogonal basis states that may be used to represent the two values of a bit. Examples are the up and down states of a spin half object, the two orthogonal polarization states of a photon and two non-degenerate energy

eigenstates of an atom. Let us represent the two states corresponding to bit values 0 and 1 by $|0\rangle$ and $|1\rangle$ respectively. However, these are not the only possible states for a quantum mechanical bit or qubit as they are called. Because of the superposition principle in quantum mechanics, an arbitrary linear combination $\alpha |0\rangle + \beta |1\rangle$ with complex coefficients α and β is also a possible state. A qubit in such a state, in a sense, carries the two possible bit values simultaneously. A device based on such qubits possesses the potential for massive parallelism that may be harnessed to construct quantum computers which are immensely more powerful than their classical counterparts. We expect such a computer to be particularly useful in simulating efficiently quantum systems such as an atom, which is a task for which

classical computers are generally extremely inadequate. However, the catch is that a qubit, when measured at the end of a computation always collapses to one or the other basic state, yielding a value which is either 0 or 1. Thus, even though it may be possible for a quantum computer to carry out a large number of computations on different sets of data parallelly, at the end of the day we obtain the result for just one of the sets. Notwithstanding this difficulty, it is possible with clever design of quantum algorithms and quantum devices to implement them, to use quantum computers to solve certain problems that are very hard to solve otherwise.

Classical Computing

Computers are getting smaller and faster day by day because electronic components are getting smaller and smaller. But this process is about to meet its physical limit.

Electricity is the flow of electrons. Since the size of transistors is shrinking to the size of a few atoms, transistors cannot be used as switches because electrons may transfer themselves to the other side of a blocked passage by the process called quantum tunnelling.

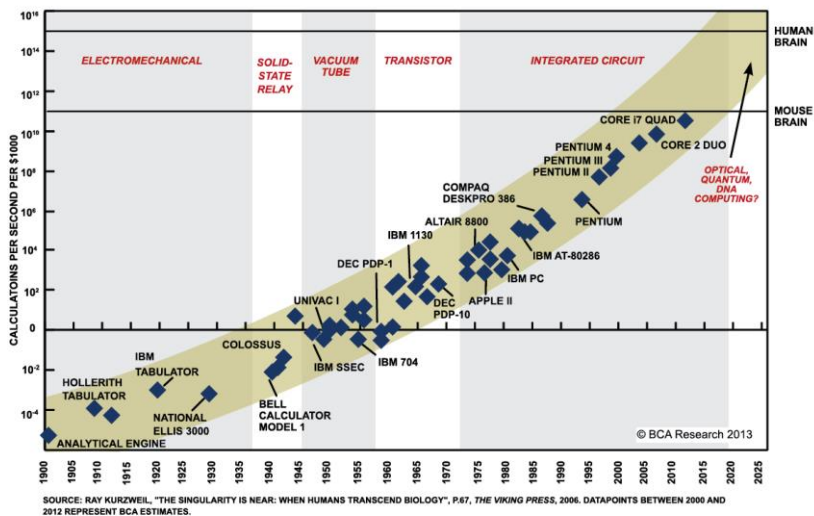
Moore's Law

Moore's law is not a natural law. It is an observation by the founder of INTEL, Gordon G. Moore in 1965, while he was working at Fairchild Semiconductor. It states that the number of transistors on a computer chip and, thus its computing power doubles roughly every 2 years, while its cost halves. This has been achieved due to extreme miniaturization that has been incorporated into chip fabrication technology, also called VLSI (Very Large Scale Integration). This has ensured that Computer hardware is getting smaller, cheaper and faster. Is Moore's law still holding up after 50 plus years? It would seem that Moore's law has slowed down, since the transistor count started doubling after 3 years, not 24 months.

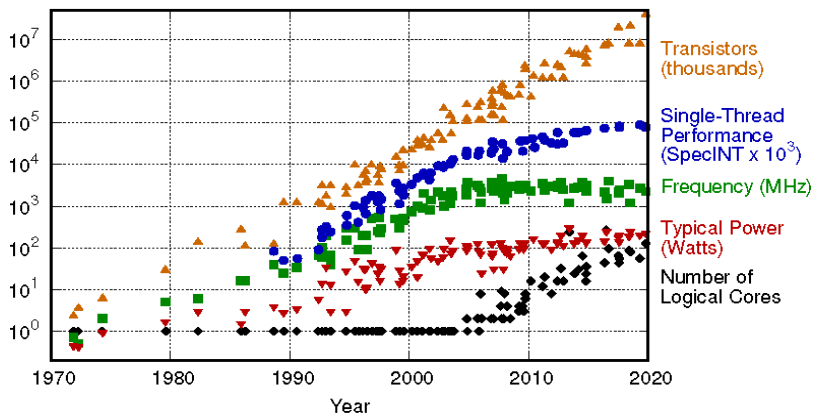
UNIT II – PRINCIPLES OF QUANTUM COMPUTATION

QUANTUM PHYSICS FOR ENGINEERS

Computer Science Stream- (CS, AI, CY, CD, IS, and BT)



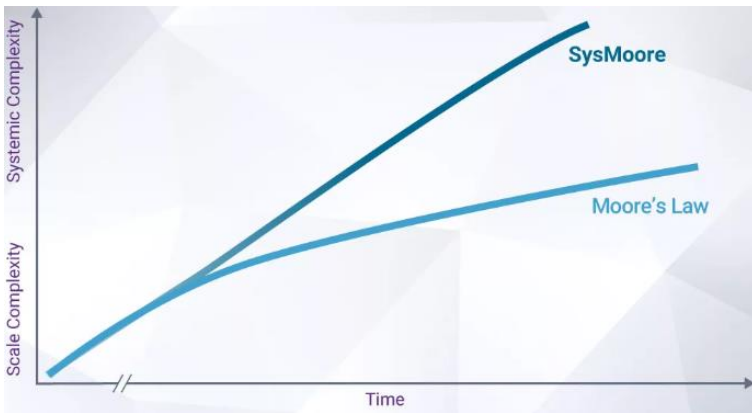
48 Years of Microprocessor Trend Data



Original data up to the year 2010 collected and plotted by M. Horowitz, F. Labonte, O. Shacham, K. Olukotun, L. Hammond, and C. Batten
New plot and data collected for 2010-2019 by K. Rupp

The slowing of Moore's law has prompted many to ask, "Is Moore's law finally ending? This, in fact, is not occurring. While Moore's law is still delivering exponential improvements, the results are being delivered at a slower pace. The pace of technology innovation is NOT slowing down, however. The explosion of hyperconnectivity, big data, artificial intelligence applications has increased the pace of innovation and the need for "Moore's law-style" improvements in delivered technology.

For many years, scale complexity drove Moore's law and the semiconductor industry's exponential technology growth. As the ability to scale a single chip slows, the industry is finding other methods of innovation to maintain exponential growth. This new design trend is driven by systemic complexity. Some aspects of this new approach to design have been dubbed "more than Moore." This term refers primarily to 2.5D and 3D integration techniques.



Key concepts of Quantum Computing :

- a) **Quantum Superposition Principle:** Qubits can exist as a linear superposition of multiple states simultaneously, allowing quantum computers to explore many potential solutions to a problem at once.
- b) **Quantum Entanglement:** Qubits can be entangled, which means that the measurement of the state of one particle in an entangled pair influences the measurement on the other particle in the entangled pair. It is a crucial quantum resource that finds application in quantum algorithms like Quantum Teleportation and others.

c) **Quantum Interference:** Quantum computers exploit interference effects to enhance the probability of obtaining the correct solution to a problem, while suppressing the incorrect solutions. This interference phenomenon is crucial for Quantum algorithms to outperform classical ones.

d) **Quantum Parallelism :**

The single-particle interference experiment is a powerful demonstration of the fundamental principles of quantum mechanics, particularly the wave-particle duality and the concept of superposition. One of the most famous versions of this experiment is the double-slit experiment performed with individual particles, such as electrons, photons, or atoms. Here we will study about the single photon interference experiment that is performed not with double slits but with mirrors and beam splitters.

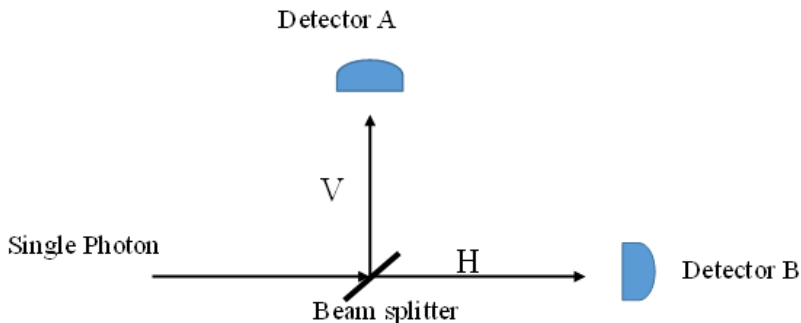
Single photon quantum interference experimental setup:

A laser that emits one photon at a time is used in both the experimental setups (a) and (b). This can be accomplished by employing a set of attenuators, which can filter out laser light and ensure that only one photon emerges at a time. This photon is then

split by a beam splitter. The beam splitter reflects half of the light that strikes it and allows the other half to pass through.

A beam splitter operates based on the principle of partial reflection and transmission. When light encounters the beam splitter's surface, a portion of the light is reflected, and the remainder is transmitted through the surface. The exact proportions of reflection and transmission depend on the coating and the design of the beam splitter. In the experiment that we are going to discuss, the beam splitter splits the incident light into reflected and transmitted light in the ratio of 1:1.

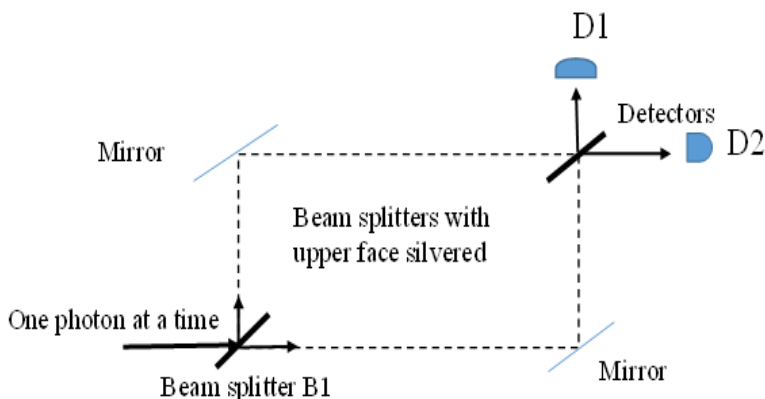
In the experimental setup (a), photon detectors A and B detect the photon with equal probabilities. It might be wrongly thought that a single photon cannot be split into two. Hence, the wrong inference would be that the single photon on arriving at the must have traveled any one of the paths, that is it is either completely transmitted in 50% of the cases or completely reflected in the other 50%. However, as we will see from the next experimental setup (b) that the correct understanding is that a single photon splits into two waves on arriving at the first beam splitter B1 and after travelling through two separate paths combine again at beam splitter 2 (B2) either constructively or destructively which is then detected at D1 or D2.



Experimental Setup (a)

In the experimental set up (b), a single photon may travel horizontally, gets deflected by a mirror, and reaches the detector. Another possibility is that it passes vertically, gets deflected by a mirror, and reaches the detector. Hence if the photon really takes a single path through the apparatus, both detectors would detect it with equal probabilities. However, in reality, we find that the photon only strikes detector D1 and never detector D2. It means that the photon was in a superposition state, and it travelled through both paths simultaneously. At the second beam splitter, the 2 components interfered constructively or destructively, and get detected by one of the detectors. Further, if we change the path length by introducing a half-wave glass plate in one of the paths

(say vertical one), the photon is now detected by detector D2 only and never by detector D1.



Experimental Setup (b)

To understand the above intriguing results, we need to understand the following facts:

- Light reflecting from the silvered face of a mirror or a beam splitter suffers phase change of π due to air-glass interface, that is change from small to large refractive index.
- Light reflecting from the back-side of the mirrored surface, there is no phase change.

- c) Similarly, there is no phase change when light is transmitted through a beam splitter.

The light going up vertically and reaching D1 suffers a phase change of 3π , while light going horizontally at B1 and reaching D1, has a phase change of π . The path difference is 2π , which means there will be constructive interference at D1, thus recording a photon at D1. The light going up vertically and reaching D2 suffers a phase change of 2π , while light going horizontally at B1 and reaching D2, has a phase change of π (there is no phase change for reflection from the back side of the beam splitter). The phase difference between the two beams which reach D2 is π , which means there will be destructive interference at the beam splitter B2. Thus no photon will be recorded at D2.

On introducing a half wave glass plate, the light going vertically upwards from B1 suffers an additional phase change of π , while the light going horizontally at B1 suffers no change. This means that now there will be destructive interference at B2 for the light reaching D1 and constructive interference at B2 for the light reaching D2. Thus after introduction of half-wave glass plate, no photons will be detected at D1 and all the photons released from the source will be detected at D2.

Differences between classical computing and quantum computing

Classical Computing	Quantum Computing
Conventional computing is based on the classical phenomenon of electrical circuits being in a single state at a given time, either on or off.	Quantum computing is based on the phenomenon of Quantum Mechanics, such as superposition and entanglement, the phenomenon where it is possible to be in more than one state at a time.
Information storage and manipulation are based on “bit”, which is based on voltage or charge; low is 0 and high is 1.	Information storage and manipulation are based on Quantum Bit or “qubit”, which is based on the spin of an electron or polarization of a single photon.
The circuit behaviour is governed by classical physics.	The circuit behaviour is governed by quantum physics or quantum mechanics.

Conventional computing use binary codes i.e. bits 0 or 1 to represent information.	Quantum computing use Qubits i.e. $ 0\rangle$, $ 1\rangle$ and the superposition state of both $ 0\rangle$ and $ 1\rangle$ to represent information.
CMOS transistors are the basic building blocks of conventional computers.	Superconducting Quantum Interference Devices or SQUID or Quantum Transistors are the basic building blocks of quantum computers.
In conventional computers, data processing is done in the Central Processing Unit or CPU, which consists of an Arithmetic and Logic Unit (ALU), processor registers and a control unit.	In quantum computers, data processing is done in a Quantum Processing Unit or QPU, which consists of several interconnected qubits.

Comparison of classical and quantum information

- Information, either classical or quantum, is physical.
- It is transmitted by physical means.

- It is stored in physical system

Quantum information	Classical information
Encoded to some property of a quantum system like photon polarization or spin of an electron.	Encoded to some property of a physical system obeying the laws of classical physics.
It is processed using quantum gates.	Processed using classical gates.
Fundamental unit of information is a qubit.	Fundamental unit of information is a bit.
It is difficult to store, transmit and process.	It is easy to store, transmit and process.
There is no way to copy unknown information.	It Easy to make copies of classical information.
Measurement of information destroys it.	Can be measured without disturbing it.

Quantum Superposition

Quantum superposition is a phenomenon associated with quantum systems such as nuclei, electrons and photons, for which wave-particle duality and other non-classical effects are observed. A quantum system can exist in more than one state at the same time. The result of the measurement is the observation of some definite state with a given probability. Quantum superposition is

easily demonstrated using a coin. A coin has a 50/50 probability of landing as either heads or tails.

What state is the coin in while it is in the air? Is it heads or tails?

A classical bit can be compared to a coin toss, where it can be in one of the two states : heads (representing 1) or tails (representing 0). Once the coin lands, its state is fixed.

A quantum bit, or qubit, is like a coin spinning in the air.

While it is spinning, it is in a superposition of both head and tail simultaneously. When measured, it collapses to either head or tail. This superposition allows qubits to perform complex computations more efficiently than classical bits.

The word “state” means any particular way that a system can possibly be described. For example, the coin can be either heads, or tails, or a combination of heads or tails while flipped in the air. All of these cases are called states of the coin system. The act of measurement destroys the superposition.

Qubit:

A qubit, like a bit, also makes use of two states $|0\rangle$ and $|1\rangle$ to hold information.

Mathematically, qubits $|0\rangle$ and $|1\rangle$ can be represented as column matrices:

$$|0\rangle = \begin{bmatrix} 1 \\ 0 \end{bmatrix}; \quad |1\rangle = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$$

However, unlike classical bits, a qubit, $|\Psi\rangle$ can also be **in a superposition state of $|0\rangle$ and $|1\rangle$ states.**

It can be written as $|\Psi\rangle = \alpha |0\rangle + \beta |1\rangle$

where α and β are generally complex numbers which represent the probability amplitudes of the states. When a qubit is measured, it only results in either $|0\rangle$ or $|1\rangle$.

Summation of probabilities

The probability of measuring the qubit in state $|0\rangle$ is $|\alpha|^2$, and the probability of measuring the qubit in state $|1\rangle$ is $|\beta|^2$. Since the total probability of observing all the states of the quantum system must add up to 100%, the modulus squared of the amplitudes must add up to 1. Thus, we have the following constraint:

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

This is called Normalization rule.

DIFFERENCE BETWEEN CLASSICAL “BITS” AND “QUBITS”:

A fundamental difference between classical bits and qubits is the way they operate. The classical bits can be deterministically set in a “0” or a “1” state.

It is read (or measured) any number of times as long as it is powered.

Reading a classical bit does not destroy its state. A bit retains its state as long as it is powered. The qubits are probabilistic.

They are in a superposition state of $|0\rangle$ and $|1\rangle$ with different probabilities.

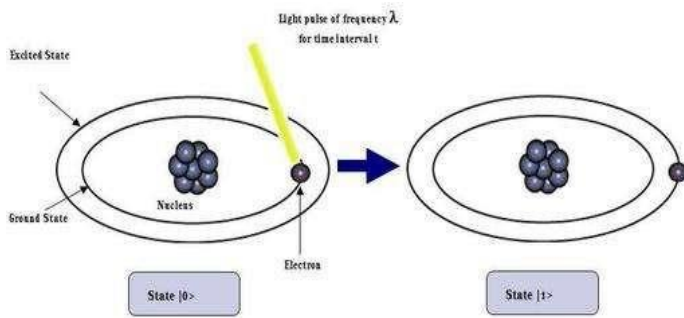
They possess characteristics of both states simultaneously, at all times, until measured. The qubits lose their internal state when they are measured.

PHYSICAL REALIZATION OF QUBITS:

In a classical computer, the 0 and 1 bit mathematically represent the two allowed voltages across a wire in a classical circuit. Semiconductor devices called transistors are used to control what happens to these voltages.

“WHAT IS A QUBIT MADE OUT OF?”

- **Energy levels of an atom:** Consider the electron in a hydrogen atom. It can be in its ground state (i.e. an s orbital) or in an excited state. So we can also store a qubit of information in the quantum state of the electron, i.e., in the superposition of the Ground state $|0\rangle$ and the Excited state $|1\rangle$



- **Spin:** Elementary particles like electrons and protons carry an intrinsic angular momentum called spin. Their spins can be used as qubits with $|0\rangle = |\uparrow\rangle$, $|1\rangle = |\downarrow\rangle$
- **Polarization of Photon:** A linearly polarized photon can be either horizontally or vertically polarized with respect to some direction in which the photon is moving. Quantum researchers can create photons one at a time and encode qubits of information into their polarization.

Wave function in Dirac notation:

A wave function (say ψ) represents the physical state of a system.

According to Paul Dirac, the state of a system is described by a vector, called a state vector, in Hilbert space \mathcal{H} . Depending on the degree of freedom (i.e. the type of state) of the system being considered, \mathcal{H} may have infinite-dimensional.

[Hilbert space \mathcal{H} : It is a complex vector space. It has all the properties of linear vector space like vector addition and scalar multiplication. In addition, it satisfies inner product operation.

An inner product is a generalization of the dot product. It is a method of multiplying vectors together in a vector space, with the result being a scalar.]

If ψ is a wavefunction, then in Dirac notation ψ is represented as $|\psi\rangle$, which is called a **ket vector**.

Example: Suppose $\psi = A e^{-i k x}$

Dirac notation $|\psi\rangle = A e^{-i k x}$

Note: Only the notation of ψ is changed. The form of the wave function remains unchanged.

If ψ^* is the complex conjugate of ψ , then ψ^* is represented as $\langle \psi |$, which is called a **bra vector**

Hence,

$$\langle \psi | = A^* e^{i k x}$$

PROPERTIES OF KET AND BRA VECTORS:

- To every ket vector $|\psi\rangle$, there corresponds a bra vector $\langle \psi |$ and vice versa.

$$|\psi\rangle \leftrightarrow \langle \psi |$$

- There is a one-to-one correspondence between ket vectors and bra vectors.

$$a |\psi\rangle + b |\phi\rangle \leftrightarrow a^* \langle \psi | + b^* \langle \phi |$$

Where a and b are complex numbers.

Basis: In quantum mechanics, the “basis vectors” can be thought of as a set of mutually perpendicular vectors, one for each “dimension” of the space in which the state vector is expressed. The magnitude of a basis vector is one. There is a one-to-one correspondence between basis vectors and dimensions of the space.

MATRIX FORM OF A WAVEFUNCTION:

Consider a discrete and complete basis that is made up of an infinite set of kets $|\phi_1\rangle$

, $|\phi_2\rangle$, $|\phi_3\rangle \dots$ etc.

The state vector $|\psi\rangle$ can be written as a linear combination of kets $|\phi_1\rangle$, $|\phi_2\rangle$, $|\phi_3\rangle \dots$ etc as follows:

$$|\psi\rangle = a_1 |\phi_1\rangle + a_2 |\phi_2\rangle + a_3 |\phi_3\rangle + \dots + a_n |\phi_n\rangle \\ = \sum_{n=1}^{\infty} a_n |\phi_n\rangle$$

Where the coefficients $a_1, a_2, a_3 \dots a_n$ represent the projection of $|\psi\rangle$ onto $|\phi_n\rangle$. Thus, a_n is the component of $|\psi\rangle$ along the vector $|\phi_n\rangle$.

Hence, $|\psi\rangle$ can be represented as a **column vector (column matrix)** given by

$$|\psi\rangle \rightarrow \begin{bmatrix} \langle \phi_1 | \psi \rangle \\ \langle \phi_2 | \psi \rangle \\ \langle \phi_3 | \psi \rangle \\ \vdots \\ \langle \phi_n | \psi \rangle \end{bmatrix} = \begin{bmatrix} a_1 \\ a_2 \\ a_3 \\ \vdots \\ a_n \end{bmatrix}$$

[A **column matrix** is a matrix having all its elements in a single column. The elements are arranged in a vertical manner. The order of a column matrix having n elements is $n \times 1$]

The bra vector $\langle \psi |$ can be represented by a **row vector (row matrix)**:

$$\begin{aligned} \langle \psi | &\rightarrow [\langle \psi | \phi_1 \rangle \quad \langle \psi | \phi_2 \rangle \quad \langle \psi | \phi_3 \rangle \quad \dots \dots \quad \langle \psi | \phi_n \rangle] \\ &= [a_1^* \quad a_2^* \quad a_3^* \quad \dots \dots \dots a_n^*] \end{aligned}$$

[A **row matrix** is a matrix having all its elements in a single row. The elements are arranged in a horizontal manner. The order of a row matrix having n elements is 1 x n]

Remark:

A ket $|\psi\rangle$ is normalized if $\langle \psi | \psi \rangle = \sum_n |a_n|^2 = 1$

If $|\psi\rangle$ is not normalized, we can multiply it by a constant ‘a’ so that

$\langle a \psi | a \psi \rangle = |a|^2 \langle \psi | \psi \rangle = 1$. This is the normalization equation.

The normalization constant ‘a’ = $1 / \sqrt{\langle \psi | \psi \rangle}$

INNER PRODUCT:

If $\psi = \psi(x)$ and $\phi = \phi(x)$ are two wavefunctions, then their inner product can be defined as

$$(\psi, \phi) = \int \psi^*(x) \phi(x) dx$$

In Dirac notation (ψ, ϕ) is written as $\langle \psi | \phi \rangle$.

Since the inner product (scalar product) is a complex number in quantum mechanics,

$$\langle \psi | \phi \rangle = \langle \phi | \psi \rangle^*$$

This property can be demonstrated as follows:

$$\langle \phi | \psi \rangle^* = \left(\int \phi^*(x) \psi(x) dx \right)^* = \int \psi^*(x) \phi(x) dx = \langle \psi | \phi \rangle$$

For any state vector $|\psi\rangle$, $\langle \psi | \psi \rangle$ is real and positive.

If the state $|\psi\rangle$ is normalized, $\langle \psi | \psi \rangle = 1$

Therefore, $\langle \psi | \psi \rangle = 0$ only if $|\psi\rangle = 0$.

CONDITION FOR ORTHOGONALITY:

Two ket vectors $|\psi\rangle$ and $|\phi\rangle$, are said to be orthogonal if their inner product is zero.

$$\text{i.e. } \langle \psi | \phi \rangle = 0$$

CONDITION FOR ORTHONORMALITY:

Two ket vectors $|\psi\rangle$ and $|\phi\rangle$, are said to be orthonormal if they are orthogonal and if each of them is normalized.

$$\text{i.e. } \langle \psi | \phi \rangle = 0, \langle \psi | \psi \rangle = 1, \langle \phi | \phi \rangle = 1$$

OPERATOR

An operator \hat{A} is a Mathematical “Transformation” that when applied to a ket vector $|\psi\rangle$ transforms it to another ket vector $|\phi\rangle$ in the same Hilbert space and when it acts on a bra vector

$\langle \psi |$ transforms it to another bra vector $\langle \phi |$ in the same Hilbert space.

Thus, we get: $\hat{A} |\psi\rangle = |\phi\rangle$ and $\hat{A} \langle\psi| = \langle\phi|$

If $\hat{A} |\psi\rangle = a |\psi\rangle$, with 'a' real, then $|\psi\rangle$ is an eigenfunction of \hat{A} with eigenvalue 'a'.

Linear operators can be represented as square matrices in quantum mechanics.

Unity operator (\hat{I}) : It leaves any ket vector unchanged.

$$\text{i.e. } \hat{I} |\psi\rangle = |\psi\rangle$$

MATRIX FORM OF INNER PRODUCT:

$$\text{Let } |\psi\rangle = a_1 |\phi_1\rangle + a_2 |\phi_2\rangle + a_3 |\phi_3\rangle + \dots + a_n |\phi_n\rangle$$

$$|\psi\rangle = \begin{bmatrix} a_1 \\ a_2 \\ a_3 \\ \vdots \\ a_n \end{bmatrix}$$

$$|\Phi\rangle = b_1 |\phi_1\rangle + b_2 |\phi_2\rangle + b_3 |\phi_3\rangle + \dots + b_n |\phi_n\rangle$$

$$|\Phi\rangle = \begin{bmatrix} b_1 \\ b_2 \\ b_3 \\ \vdots \\ b_n \end{bmatrix}$$

$$\langle \psi | \phi \rangle = [a_1^* \ a_2^* \ a_3^* \ \dots \dots \dots a_n^*] \begin{bmatrix} b_1 \\ b_2 \\ b_3 \\ \vdots \\ \vdots \\ b_n \end{bmatrix}$$

$$= a_1^* b_1 + a_2^* b_2 + \dots + a_n^* b_n$$

Identity matrix(I):

An identity matrix is a square matrix in which all the elements of principal diagonals are one, and all other elements are zeros. If any matrix is multiplied by the identity matrix, the result will be given a matrix.

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

Example: The matrix form of ket vectors $|0\rangle$ and $|1\rangle$ can be written as:

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

$$I |0\rangle = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \end{bmatrix} = \begin{bmatrix} 1 \times 1 + 0 \times 0 \\ 0 \times 1 + 1 \times 0 \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$$

$$I |1\rangle = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 0 \\ 1 \end{bmatrix} = \begin{bmatrix} 1 \times 0 + 0 \times 1 \\ 0 \times 0 + 1 \times 1 \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$$

Transpose of a Matrix:

A^T denotes the transpose of a matrix A and it is obtained by swapping the rows with columns.

$$\text{Suppose } A = \begin{bmatrix} A_{11} & A_{12} & A_{13} \\ A_{21} & A_{22} & A_{23} \\ A_{31} & A_{32} & A_{33} \end{bmatrix}$$
$$A^T = \begin{bmatrix} A_{11} & A_{21} & A_{31} \\ A_{12} & A_{22} & A_{32} \\ A_{13} & A_{23} & A_{33} \end{bmatrix}$$

Conjugate of a matrix:

A conjugate matrix is \bar{A} a matrix obtained from a given matrix A by taking the complex conjugate of each element.

Hence, the conjugate of matrix A defined above is

$$\bar{A} = \begin{bmatrix} A_{11}^* & A_{12}^* & A_{13}^* \\ A_{21}^* & A_{22}^* & A_{23}^* \\ A_{31}^* & A_{32}^* & A_{33}^* \end{bmatrix}$$

Identity matrix(I):

An identity matrix is a square matrix in which all the elements of principal diagonals are one, and all other elements are zeros. If any matrix is multiplied by the identity matrix, the result will be given a matrix.

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

Example: The matrix form of ket vectors $|0\rangle$ and $|1\rangle$ can be written as:

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

$$I |0\rangle = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \end{bmatrix} = \begin{bmatrix} 1 \times 1 + 0 \times 0 \\ 0 \times 1 + 1 \times 0 \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$$

$$I |1\rangle = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 0 \\ 1 \end{bmatrix} = \begin{bmatrix} 1 \times 0 + 0 \times 1 \\ 0 \times 0 + 1 \times 1 \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$$

Transpose of a Matrix:

A^T denotes the transpose of a matrix A and it is obtained by swapping the rows with columns.

$$\text{Suppose } A = \begin{bmatrix} A_{11} & A_{12} & A_{13} \\ A_{21} & A_{22} & A_{23} \\ A_{31} & A_{32} & A_{33} \end{bmatrix}$$

$$A^T = \begin{bmatrix} A_{11} & A_{21} & A_{31} \\ A_{12} & A_{22} & A_{32} \\ A_{13} & A_{23} & A_{33} \end{bmatrix}$$

Conjugate of a matrix:

A conjugate matrix is \bar{A} a matrix obtained from a given matrix A by taking the complex conjugate of each element.

Hence, the conjugate of matrix A defined above is

$$\bar{A} = \begin{bmatrix} A_{11}^* & A_{12}^* & A_{13}^* \\ A_{21}^* & A_{22}^* & A_{23}^* \\ A_{31}^* & A_{32}^* & A_{33}^* \end{bmatrix}$$

Hermitian matrix:

A Hermitian matrix is a square matrix composed of complex numbers, and it is equal to its conjugate transpose.

Example: $M = \begin{bmatrix} 1 & i \\ -i & 1 \end{bmatrix}$

The conjugate transpose of the matrix is M^H or $M^\dagger = \begin{bmatrix} 1 & i \\ -i & 1 \end{bmatrix}$

Here, $M = M^\dagger$

Hence M is a Hermitian matrix.

Unitary matrix:

Unitary Matrix is a square matrix of complex numbers. The product of the conjugate transpose of a unitary matrix, with the unitary matrix itself, gives an identity matrix.

Example:

$$U = \frac{1}{3} \begin{bmatrix} 2 & -2+i \\ 2+i & 2 \end{bmatrix}$$

Taking conjugate $\bar{U} = \frac{1}{3} \begin{bmatrix} 2 & -2-i \\ 2-i & 2 \end{bmatrix}$

If we take the transpose of the above matrix, it is called a Hermitian matrix. $U^\dagger = \frac{1}{3} \begin{bmatrix} 2 & 2-i \\ -2-i & 2 \end{bmatrix}$

$$\therefore U^\dagger \cdot U = \frac{1}{3} \begin{bmatrix} 2 & 2-i \\ -2-i & 2 \end{bmatrix} \frac{1}{3} \begin{bmatrix} 2 & -2+i \\ 2+i & 2 \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} = I$$

Similarly, $U \cdot U^\dagger = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} = I$

Hence, U is a unitary matrix.

Pauli matrices:

The Pauli matrices are a set of four 2x2 complex matrices. They are used to represent spin angular momentum. These matrices are Hermitian and Unitary.

These matrices are very powerful in quantum computing as they can be used to represent quantum logic gates. They can set the rotational parameters for qubits. These matrices go by a variety of notations.

$$\sigma_1 \equiv \sigma_x \equiv X \equiv \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \quad \sigma_2 \equiv \sigma_y \equiv Y \equiv \begin{bmatrix} 0 & -i \\ i & 0 \end{bmatrix}$$

$$\sigma_3 \equiv \sigma_z \equiv Z \equiv \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}$$

PROBABILITY IN BRA KET NOTATION:

Inner product: As we have seen above, the inner product, also called the dot product signifies the projection of a wavefunction onto a particular state. This state could be some abstract spin state, a particular momentum state or even a specific value of position(x).

We have seen that if $\psi = \psi(x)$ and $\phi = \phi(x)$ are two wavefunctions, then their inner product can be defined as

$$(\psi, \phi) = \int \psi^*(x) \phi(x) dx = \langle \psi | \phi \rangle, \\ \text{where } \langle \psi | \phi \rangle = \langle \phi | \psi \rangle^*$$

$$\langle \psi | \psi \rangle = 1 \text{ if the state } |\psi\rangle \text{ is normalized}$$

The wave-function $\psi(x)$ can be interpreted as the projection of $|\psi\rangle$ onto the x basis.

Hence, we can write $\psi(x) = \langle x | \psi \rangle$, $\psi(p) = \langle p | \psi \rangle$, $\psi(s) = \langle s | \psi \rangle$

The above projections project an abstract wavefunction ψ onto real space $|x\rangle$, momentum $|p\rangle$ and spin $|s\rangle$ respectively. This projected wave function is the probability amplitude of finding the system in the state (x,p,s), To get the probability we need to calculate the modulus squared of the amplitude. Thus

$$P(x) = |\psi(x)|^2 = |\langle x|\psi \rangle|^2$$

$$\text{Similarly } P(k) = |\psi(p)|^2 = |\langle p|\psi \rangle|^2$$

$$\text{and } P(s) = |\psi(s)|^2 = |\langle s|\psi \rangle|^2$$

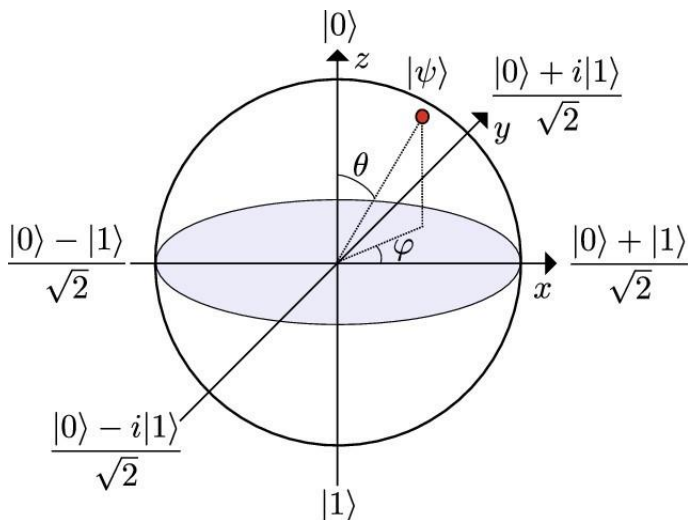
Bloch Sphere Representation

Bloch sphere is a physical representation of all possible qubit states. It is a sphere of unit radius and the state of a qubit can be represented by a vector in this sphere. $|0\rangle$ is at the north pole, $|1\rangle$ is at the south pole, as shown in Figure below.

Using the spherical coordinate system, an arbitrary position of the state vector of a qubit can be written in terms of the angles θ (elevation, the state vector makes from z-axis) and ϕ (azimuth, the angle of projection of the state vector in the x-y plane from the x-axis) it makes in the

$$|\psi\rangle = e^{i\gamma} \left[\cos\left(\frac{\theta}{2}\right) |0\rangle + e^{i\phi} \sin\left(\frac{\theta}{2}\right) |1\rangle \right] = \begin{bmatrix} \cos\left(\frac{\theta}{2}\right) \\ e^{i\phi} \sin\left(\frac{\theta}{2}\right) \end{bmatrix}$$

Where γ is a Global phase and ϕ is the Local phase. The Global phase cancels out with its own complex conjugate and can therefore be set to zero, which is done, yielding the column matrix.



NOTE:

- For $\varphi = 0$ and $\theta = 0$, the state $|\psi\rangle$ corresponds to $|0\rangle$ and is along z-axis.
- For $\varphi = 0$ and $\theta = 180^\circ$ the state $|\psi\rangle$ corresponds to $|1\rangle$ and is along -z-axis.
- When $\theta = 90^\circ$, $|\psi\rangle$ is in the x-y plane.
 - For $\varphi = 90^\circ$, $|\psi\rangle = \frac{1}{\sqrt{2}} (|0\rangle + i|1\rangle)$, is a superposition state along +y-axis.
 - For $\varphi = -90^\circ$, $|\psi\rangle = \frac{1}{\sqrt{2}} (|0\rangle - i|1\rangle)$, is a superposition state along -y-axis.

- For $\varphi = 0^\circ$, $|\psi\rangle = \frac{1}{\sqrt{2}} (|0\rangle + |1\rangle)$, is a superposition state along the +x-axis.
- For $\varphi = 180^\circ$, $|\psi\rangle = \frac{1}{\sqrt{2}} (|0\rangle - |1\rangle)$, is a superposition state along the -x-axis.

Explanation of the Bloch sphere representation of a Qubit:

To understand the Bloch representation, we have to make sense of the projection of a spin operator onto any generalized axis. As we know that the projection of spin $\frac{1}{2}$ onto any of the 3 axes, x,y,z gives us the components $\hbar/2$ and $-\hbar/2$. This result is true for projection about any generalized axis in 3 dimensions as well.

For a classical computer, the two logical states 0 and 1 are represented by the poles of a sphere.

In contrast, the state of a qubit can be represented by any point on the sphere. Since there are infinite points on the sphere, a qubit in principle has more capacity to store information compared to a classical bit.

Note: Bloch sphere represents the state of only one qubit. There is no generalization of Bloch sphere for multiple qubits.

QUANTUM GATES: IT'S ACTION ON QUBITS

Classical computer circuits consist of wires and logic gates. The wires carry information around the circuit, while the logic gates perform calculations and manipulate information, converting it from one to another. Computers manipulate bits using classical logic gates, such as OR, AND, NOT, NAND, etc.

Similarly, quantum computers manipulate qubits using quantum gates which are usually represented as matrices. A gate which acts on k qubits is represented by a $2^k \times 2^k$ unitary matrix. The number of qubits in the input and output of the gate has to be equal. The action of

the quantum gate is found by multiplying the matrix representing the gate with the vector which represents the quantum state. These gates, which correspond to rotations about the x -, y -, and z - axes of the Bloch sphere, are defined in terms of the Pauli gates. Below, we give the definitions of the Pauli gates.

SINGLE QUBIT GATES:

Pauli-X gate

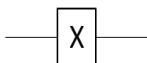
In classical computers, the NOT gate takes one input and reverses its value. For example, it changes the 0 bit to a 1 bit or changes a 1 bit to a 0 bit. It is like a light switch flipping a light from ON to OFF, or from OFF to ON.

Pauli-X gate is a quantum analogue of the classical NOT gate.

- The application of this gate rotates the qubit by 180° along the x-axis. It transforms $|0\rangle$ to $|1\rangle$ and vice versa.
- The matrix form of X-gate is obtained as follows

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

- Circuit representation



- Dirac notation

$$X |0\rangle = |1\rangle \quad \text{and} \quad X |1\rangle = |0\rangle$$

- When the qubit is in a superposition state $|\psi\rangle = \alpha |0\rangle + \beta |1\rangle$, then

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

- In matrix form $|\psi\rangle = \begin{bmatrix} \alpha \\ \beta \end{bmatrix}$

Thus the action of X gate is :

$$X \begin{bmatrix} \alpha \\ \beta \end{bmatrix} = \begin{bmatrix} \beta \\ \alpha \end{bmatrix}$$

Pauli -Y gate

- The application of this gate rotates the qubit by 180° along the y-axis.
- It transforms $|0\rangle$ to $i|1\rangle$ and $|1\rangle$ to $-i|0\rangle$.
- Matrix for of Y-gate

$$Y = |0\rangle \langle 0| (i) |1\rangle \langle 1| + |1\rangle \langle 1| (-i) |0\rangle \langle 0|$$

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

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

$$\text{Dirac notation } Y |0\rangle = i |1\rangle \quad Y |1\rangle = -i |0\rangle$$

- Circuit representation



Pauli - Z gate

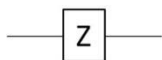
- The application of this gate rotates the qubit by 180° along the z-axis.
- It leaves $|0\rangle$ unchanged and flips the sign of $|1\rangle$ to $-|1\rangle$.
- Matrix for of Z-gate

$$Z = |0\rangle\langle 0| - |1\rangle\langle 1|$$

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

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

- Circuit representation



- Dirac notation $Z|0\rangle = |0\rangle$ $Z|1\rangle = -|1\rangle$

Phase gate can also be written as

$$\begin{bmatrix} 1 & 0 \\ 0 & e^{i\theta} \end{bmatrix}$$

For $\theta = \pi/2$ we get the S Gate and for $\theta = \pi/4$, we get the T Gate.

S Gate :

- It rotates the qubit by $\frac{\pi}{2}$ radian along the z-axis.
- The effect of this gate is to modify the phase of the quantum state.
- It maps $|0\rangle \leftrightarrow |0\rangle$ and $|1\rangle \leftrightarrow e^{\frac{i\pi}{2}}|1\rangle$.
- Matrix representation $S \equiv \begin{bmatrix} 1 & 0 \\ 0 & e^{\frac{i\pi}{2}} \end{bmatrix}$
- Circuit representation



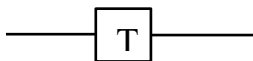
- Dirac notation

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

$$S |1\rangle = e^{\frac{i\pi}{2}} |1\rangle = i |1\rangle$$

T-gate

- Like S-gate, it modifies the phase of the quantum state.
- It maps $|0\rangle \leftrightarrow |0\rangle$ and $|1\rangle \leftrightarrow e^{\frac{i\pi}{4}}|1\rangle$.
- Matrix representation $T \equiv \begin{bmatrix} 1 & 0 \\ 0 & e^{\frac{i\pi}{4}} \end{bmatrix}$
- Circuit representation



- Dirac notation

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

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

HADAMARD GATE

- It is one of the most important gates for quantum computing. If the qubit starts in a definite $|0\rangle$ or $|1\rangle$ state, the Hadamard gate puts each into a superposition of $|0\rangle$ and $|1\rangle$ states.
- Matrix representation $H \equiv \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$
- Circuit representation



- Dirac notation

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

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

Multiple qubits gates:

CNOT gate : An Example of a 2 Qubit Gate

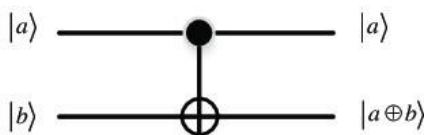
- It is a Controlled NOT (CNOT) gate.
- It acts on two qubits.
- It performs the NOT operation on the second qubit only when the first qubit is $|1\rangle$ otherwise leaves it unchanged.

Truth Table of the C NOT Gate:

Input		Output	
a	b	a	$a \oplus b$
0	0	0	0
0	1	0	1
1	0	1	1
1	1	1	0

MATRIX REPRESENTATION AND SYMBOLIC REPRESENTATION OF THE CNOT GATE:

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



- Dirac notation of CNOT Gate :

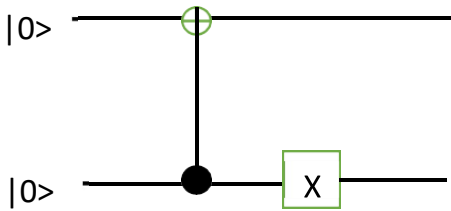
$$CNOT |00\rangle = |00\rangle ; CNOT |01\rangle = |01\rangle$$

$$CNOT |10\rangle = |11\rangle ; CNOT |11\rangle = |10\rangle$$

QUANTUM CIRCUITS:

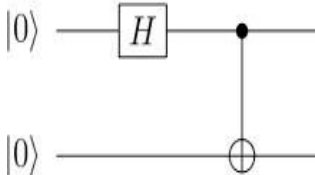
A quantum circuit is required to carry out computations on a quantum computer. It consists of a series of operations referred to as quantum gates. These quantum gates, which are assigned to certain qubits, change the quantum states of some of the qubits, causing those qubits to perform the calculations required to solve a problem.

(1)



CNOT gate leaves it unchanged as the first qubit is $|0\rangle$. In the second step, X-gate flips the second qubit to $|1\rangle$. Hence the output is $|01\rangle$.

(2)



The states change from the start to the end after every gate:

The Hadamard gate changes $|0\rangle$ to $(|0\rangle + |1\rangle) / \sqrt{2}$

Therefore,

$$|00\rangle \rightarrow [(|0\rangle + |1\rangle) / \sqrt{2}] |0\rangle = (|00\rangle + |10\rangle) / \sqrt{2}$$

In the last step, the second qubit is the control qubit, and because it is $|0\rangle$ there is no change in the first qubit.

Hence CNOT does not actually change anything.

The output is $(|00\rangle + |10\rangle) / \sqrt{2}$

Accounting for the extra-ordinary capability of quantum computing

The main advantage that a quantum computer has over a classical computer is **parallelism**. A quantum computer can perform operations on all of the states simultaneously because qubits can be in a superposition of states.

Let us consider two systems.

System 1: With 2 bits

This can represent **4 different values**.

Possible states are [00, 01, 10, 11]

Particular state-value = 2^0 bit 0 + 2^1 bit 1

i.e. particular state-value $\in \{0, 1, 2, 3\}$, one of the 4 possible values

System 2: With 2 qubits

This can represent infinite different values (vector space) formed from 4 different basis state,

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

Possible states: Infinite













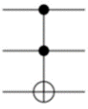
$$\text{particular state-value: } \alpha |00\rangle + \beta |01\rangle + \gamma |10\rangle + \delta |11\rangle$$

$$\text{such that } |\alpha|^2 + |\beta|^2 + |\gamma|^2 + |\delta|^2 = 1$$

Hence, an n-bit classical system can be in **one of the 2^n possible states at a time**, and all it needs is the value of these **n-bits to be fully recognized.**

Whereas an n-qubit system can be in a **superposition of all of those states 2^n states at a given time** and it needs the value of coefficients of all of the **$2^n - 1$ states (considering that summation is 1) basis to be fully recognized.**

Consequently, one can compute with 2^n values in a single step on a quantum computer. This enormous parallelism is one reason why quantum computers are so powerful.

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