

## Syllabus

**1. Principles of Quantum Information and Quantum Computing**

Introduction to Quantum Computing, Moore's law and its end, Differences between Classical and Quantum computing. Concept of qubit and its properties. Representation of qubit by Bloch sphere. Single and Two qubits. Extension to N qubits.

**2. Dirac Representation and Matrix Operations**

Matrix representation of  $|0\rangle$  and  $|1\rangle$  states, Identity Operator I, Applying I to  $|0\rangle$   $|1\rangle$  states, Pauli Matrices and its operations on  $|0\rangle$  and  $|1\rangle$  states. Explanation of (i) Conjugate of a matrix and (ii) Transpose of a matrix. Unitary matrix U, Examples: Row and Column Matrices and their multiplication (Inner Product), Probability, and Quantum Superposition, normalization rule. Orthogonality, Ortho normality. Numerical Problems

**3. Quantum Gates**

**Single Qubit Gates:** Quantum Not Gate, Pauli X, Y and Z Gates, Hadamard Gate, Phase Gate (or S Gate), T Gate **Multiple Qubit Gates:** Controlled gate, CNOT Gate (Discussion for 4 different input states), Representation of Swap gate, Controlled-Z gate, Toffoli gate

## Contents

|                                                                  |          |
|------------------------------------------------------------------|----------|
| <b>1 Principles of Quantum Information and Quantum Computing</b> | <b>3</b> |
| 1. Introduction.....                                             | 3        |
| 1.1 Classical vs. Quantum computing.....                         | 3        |
| 1.2 Moore's Law and its end.....                                 | 4        |
| 1.3 Qubits.....                                                  | 5        |
| 1.3.1. Concept of qubit.....                                     | 5        |
| 1.3.2 Properties of qubit.....                                   | 6        |
| 1.4 Representation of a Qubit.....                               | 7        |
| 1.4.1 Bloch Sphere.....                                          | 8        |
| 1.5 Single qubit system and multiple qubit system.....           | 9        |
| 1.5.1 One-qubit system.....                                      | 10       |
| 1.5.2 Two-qubit system.....                                      | 10       |
| 1.5.3 n-qubit system.....                                        | 10       |
| 1.6 Model and Previous Year Questions.....                       | 11       |

|                                                           |           |
|-----------------------------------------------------------|-----------|
| <b>2 Dirac Representation and Matrix Operations</b>       | <b>11</b> |
| 2.0 Introduction.....                                     | 11        |
| 2.1 Matrix representation of 0 and 1 states.....          | 11        |
| 2.2 Identity Operator.....                                | 11        |
| 2.3 Pauli Matrices.....                                   | 12        |
| 2.4 Action of the Pauli matrices on the basis states..... | 13        |
| 2.5 Conjugate of a matrix.....                            | 14        |
| 2.6 Transpose of a matrix.....                            | 14        |
| 2.7 Hermitian adjoint of a matrix.....                    | 14        |
| 2.8 Hermitian Matrix.....                                 | 15        |
| 2.9 Unitary Matrix.....                                   | 15        |
| 2.10 Column and Row Matrices.....                         | 15        |
| 2.11 Inner Product.....                                   | 16        |
| 2.12 Probability.....                                     | 17        |
| 2.13 Orthogonality and Orthonormality.....                | 17        |
| 2.13.1 Orthogonality.....                                 | 17        |
| 2.13.2 Orthonormality.....                                | 18        |
| <b>3 Quantum Gates</b>                                    |           |
| 3.1. Single Qubit Gates.....                              | 19        |
| 3.1.1 Quantum NOT gate(Pauli-X gate).....                 | 20        |
| 3.1.2 Pauli-Z gate.....                                   | 20        |
| 3.1.3 Pauli-Y gate .....                                  | 21        |
| 3.1.4 Hadamard gate.....                                  | 21        |
| 3.1.5 Phase gate (S gate) .....                           | 23        |
| 3.1.6 T gate .....                                        | 23        |
| <b>3.2 Multiple Qubit Gates</b>                           |           |
| 3.2.1 Controlled-NOT(CNOT)gate.....                       | 24        |
| 3.2.2 SWAP gate.....                                      | 24        |
| 3.2.3 Controlled-Z gate.....                              | 27        |
| 3.2.4 Toffoli gate.....                                   | 27        |
| 3.3 Model and Previous Year Questions.....                | 28        |
| 3.4 Numerical.....                                        | 29        |

## 1. Principles of Quantum Information and Quantum Computing

### 1. Introduction

Quantum information revolutionizes problem-solving and data processing by harnessing the power of quantum systems as information carriers, going beyond the binary framework of conventional computing. Just as classical information is handled by digital computers, quantum information can be processed, transmitted, and analyzed using sophisticated algorithms, computer science, and mathematics. At the heart of quantum information lies the qubit, the quantum counterpart of the classical bit. Quantum information systems offer the potential for fundamentally secured transmission and the ability to tackle problems that exceed the capabilities of modern computers. This interdisciplinary field combines quantum mechanics, computer science, information theory, philosophy, and cryptography to explore the profound implications of quantum information.

In quantum information science, information resides within the physical state of a quantum system. While quantum mechanics investigates the microscopic properties of matter, quantum information science extracts valuable information from these properties. Quantum computation takes center stage, manipulating and processing information through logical operations implemented via cutting-edge quantum information processing techniques.

A key highlight of quantum information sciences is the development of quantum algorithms like Shor's algorithm and Grover's algorithm, which exhibit exponential speedup compared to their classical counterparts. This extraordinary feature empowers quantum computers with remarkable potential in cryptography, simulation, and optimization domains. However, it is important to note that quantum computing is still in its nascent stage, and numerous challenges must be addressed before its widespread adoption becomes feasible.

### 1.1 Classical vs. Quantum computing

| Classical Computing                                                                 | Quantum Computing                                                                  |
|-------------------------------------------------------------------------------------|------------------------------------------------------------------------------------|
| Calculates with transistors, which can represent <i>either</i> 0 or 1.              | Calculates with qubits, which can represent <i>both</i> 0 and 1 at the same time.  |
| Computational power increases in a 1:1 relationship with the number of transistors. | Computational power increases exponentially in proportion to the number of qubits. |

|                                                                               |                                                                                        |
|-------------------------------------------------------------------------------|----------------------------------------------------------------------------------------|
| Classical computers have low error rates and can operate at room temperature. | Quantum computers have high error rates and need to be kept at ultra cold temperature. |
| Most of every day processing is best handled by classical computers.          | Well suited for tasks like optimization problems, data analysis, and simulations.      |
| Calculations are deterministic.                                               | Calculations are probabilistic.                                                        |
| Data processing is carried out by logic and in sequential order.              | Data processing is carried out by quantum logic at parallel instances.                 |

### 1.2 Moore's Law and its end

Moore's law, along-standing observation in computer technology, states that the number of transistors on a chip doubles approximately every two years (see figure 1.1), resulting in increased processing power. This prediction has held true for over five decades, with the number of transistors soaring from 3,500 to nearly 50 billion on a similarly sized chip. However, there is a limit to how far this trend can continue.

As transistors become increasingly tiny, reaching sizes on the nanometer scale, quantum effects like *quantum tunneling* come into play and hinder their performance. Transistors, acting as switches, control the flow of electrons by representing the "OFF" state as 0 and the "ON" state as 1. Yet, when transistors shrink to atom-sized dimensions, quantum tunnelling allows electrons to traverse the barrier even in the "OFF" state, resulting in unpredictable behavior and erroneous calculations. This challenge poses a significant obstacle as we confront the impending end of Moore's law, potentially leading to a notable slowdown in the advancement of computer hardware.

### 1.3 Qubits:

#### 1.3.1 Concept of qubit

A qubit, the fundamental unit of information in quantum computing, distinguishes itself from classical bits by its ability to exist in a superposition of states. While classical bits can only represent a single value of either 0 or 1, qubits can simultaneously represent both 0 and 1 due to their unique properties.

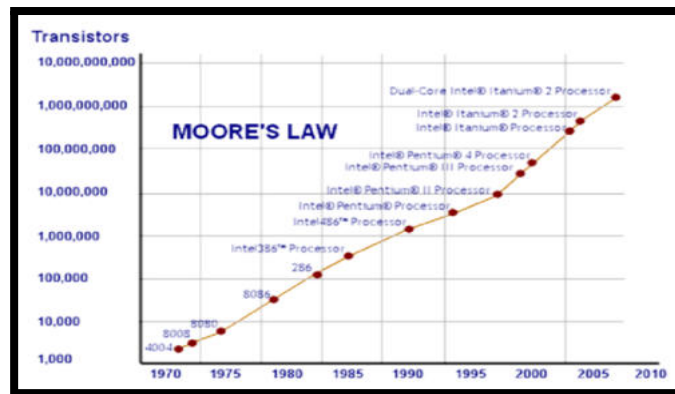


Figure 1.1: Moore's Law

Physically, a qubit can be realized using different systems, such as the two spin states of an electron (spin up and spin down) or the two polarizations of a photon (horizontal and vertical polarization). These physical implementations provide a basis for encoding and manipulating qubits in quantum computing systems.

The remarkable capabilities of qubits enable quantum computers to outperform classical computers in certain tasks. For instance, Grover's algorithm empowers quantum computers to search unsorted databases quadratically faster than any classical algorithm, while Shor's algorithm exponentially accelerates the factorization of large numbers.

Nevertheless, realizing qubits in practical quantum computing faces significant challenges. Qubits are exceedingly sensitive to their surroundings and can be easily disturbed by external factors like temperature fluctuations and electromagnetic interference. These disturbances introduce errors in quantum computations, posing a substantial obstacle that must be mounted to achieve practical and reliable quantum computers. Addressing these challenges in qubit implementation remains a critical focus in the field of quantum computing.

### 1.3.2 Properties of qubit

Qubits possess distinctive properties that set them apart from classical bits, shaping the power and challenges of quantum computing:

1. **Superposition:** Unlike classical bits that represent a single value, qubits can exist in a superposition of states, simultaneously representing both 0 and 1. This superposition enables quantum computers to perform multiple calculations in parallel, vastly expanding computational possibilities.
2. **Entanglement:** Qubits can be entangled with one another, establishing correlations between their states. This entanglement allows quantum computers to perform certain computations

more efficiently than classical computers, providing a foundation for advanced algorithms and parallel processing.

3. Probabilistic Nature: Qubits are described probabilistically, meaning that their states can only be determined with certain probabilities. This probabilistic nature introduces an element of uncertainty into quantum computations, making it challenging to predict the precise output of a quantum algorithm.
4. Unitarity: Qubits must maintain unitarity, which ensures that the sum of probabilities for all possible states of a qubit equals 1. This requirement preserves the reversibility of quantum computations and ensures the conservation of total probability throughout quantum operations.
5. No-cloning theorem: Due to the quantum nature of qubits, information cannot be copied from one qubit to another in the same way as classical bits. The no-cloning theorem states that it is impossible to create an identical copy of an arbitrary unknown qubit, making quantum information distinct and preventing direct replication.

These inherent properties of qubits render quantum computing both powerful and challenging. Understanding and harnessing these properties are essential for the development of practical and scalable quantum computers, as they hold the key to unlocking the full potential of quantum information processing.

#### 1.4. Representation of a Qubit

Mathematically, a qubit is represented in the following way  $|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$ , Where  $|\psi\rangle$  represents the qubit,  $|0\rangle$  and  $|1\rangle$  represent the 0 and the 1 bits.  $|\rangle$  is the notation given to something called a ket vector. A ket vector  $\psi$  is analogous to the Euclidian vector, say  $\vec{a} = a_1\vec{i} + a_2\vec{j}$ . (In mathematics, the unit vectors  $\vec{i}$  and  $\vec{j}$  are also called as basis vectors. Similarly, in our case,  $|0\rangle$  and  $|1\rangle$  are the basis vectors. In particular, these two are called the computational basis.

We see that the above expression for the qubit is a linear combination of the pure bits  $|0\rangle$  and  $|1\rangle$ . This is also called a superposition state (the qubit is in both the  $|0\rangle$  and  $|1\rangle$  states).

The coefficients  $\alpha$  and  $\beta$  are in general complex numbers, and they have a probabilistic interpretation.  $|\alpha|^2$  is the probability that you would get a  $|0\rangle$  if you measured the qubit  $|\psi\rangle$ . Similarly,  $|\beta|^2$  is the probability that you would get a  $|1\rangle$  if you measured the qubit  $|\psi\rangle$ . And from the rules of probability, the sum of probabilities of these two independent events must be equal to 1. Therefore, we get

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

This normalization condition guarantees that the total probability remains conserved throughout quantum computations.

### 1.4.1 Bloch Sphere

There is another way to represent a qubit: a graphical representation called the Bloch sphere. The Bloch sphere is a sphere of unit radius with a point on its surface that represents the state of the qubit.

In two-dimensional Euclidian space, a vector can be written as, say,  $\vec{a} = a_1 \hat{i} + a_2 \hat{j}$ . This means that, to uniquely describe this vector (state), we need exactly two numbers,  $a_1$  and  $a_2$ .

For different values of  $a_1$  and  $a_2$ , we get vectors pointing in different directions and having different lengths. We can say a similar thing for our ket vector  $|\psi\rangle$ . But remember that  $\alpha$  and  $\beta$  are both complex numbers and that each complex number is made of two real numbers (for example,  $\alpha = r_1 e^{i\phi_1}$  and  $\beta = r_2 e^{i\phi_2}$ ). So to uniquely describe the ket vector  $|\psi\rangle$  mentioned

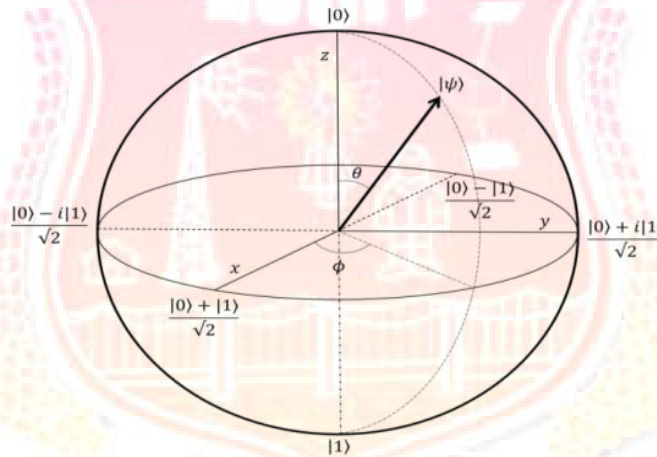


Figure1.2: Bloch Sphere

above, we need 4 real numbers, which also makes it impossible for us to visualize such a ket vector in our restricted 3-dimensional space. But there is a way for us to reduce the 4 numbers to 2. We can omit an over all phase factor to reduce the total required numbers to 3. We can also use the condition that the sum of probabilities must be one; this reduces the Total required numbers to 2. After following these steps, the mathematical form for the

qubit changes to the following

$$|\psi\rangle = \cos \frac{\theta}{2} |0\rangle + e^{i\phi} \sin \frac{\theta}{2} |1\rangle \quad (1.1)$$

We see here that we need only two numbers  $\theta$  and  $\phi$  to uniquely describe a ket vector. We call these numbers as  $\theta$  and  $\phi$  because they represent respectively the polar angle and azimuthal angle in the 3-dimensional spherical polar coordinates.

Remember that to uniquely represent a Euclidian vector in 3D space, we need three numbers. These 3 numbers depend on the coordinate system we use. If we use the Cartesian coordinate system, the three numbers are x, y, z. If we use the spherical polar coordinates, the three numbers are r,  $\theta$ ,  $\phi$ . In our qubit representation, we have these  $\theta$  and  $\phi$  and we consider  $r=1$ . This means that depending on the values of  $\theta$  and  $\phi$ , we have an arrow in 3D space pointing in a certain direction and has length of 1.

The definitions for  $\theta, \phi$  are as follows:

$\theta$ : This is the angle made by the vector with the positive z-axis such that  $0 \leq \theta \leq \pi$ .

$\phi$ : This is the angle made by the vector with the positive x-axis along the xy-plane such that  $0 \leq \phi \leq 2\pi$ .

So given a value of  $\theta$  and  $\phi$ , we shall start with a unit vector pointing in the positive z direction. We shall rotate the vector by an angle  $\theta$  from the positive z-axis along the xz-plane. Then we will rotate the vector by an angle  $\phi$  from the positive x-axis along the xy-plane.

Note: You will need to remember this formula (called Euler's formula) for the following section:

$$e^{ix} = \cos(x) + i \sin(x) \quad (1.2)$$

Where  $i$  is the imaginary number such that  $i^2 = -1$ .

Now, let's see where some common qubit states are presented on the Bloch sphere. **For  $\theta=0, \phi=0$**

Since both angles are zero, our unit vector stays on the positive z-axis (we call this the north pole of the sphere). Let's see which state these values correspond to. Substitute these angle values in (1.1). We get

$$|\psi\rangle = \cos \frac{\theta}{2} |0\rangle + e^{i\phi} \sin \frac{\theta}{2} |1\rangle \quad |\psi\rangle = |0\rangle$$

Since  $\cos(0)=1$  and  $\sin(0)=0$ . So at this point (the north pole), we have the state  $|0\rangle$ .

**For  $\theta=\pi, \phi=0$**  For these angles, the unit vector rotates by  $180^\circ$  with the positive z-axis and now points in the negative z-axis (we call this the south pole of the sphere). Let's see which state these values correspond to. Substitute these angle values in (1.1). We get

$$|\psi\rangle = \cos(\pi/2)|0\rangle + e^{i*0} \sin(\pi/2)|1\rangle, |\psi\rangle = |1\rangle \text{ since } \cos \pi/2 = 0, e^0 = 1 \text{ and } \sin(\pi/2) = 1. \text{ So at this}$$

Point (the south pole), we have the state  $|1\rangle$ .

**For  $\theta=\pi/2, \phi=0$**

For these angles, the unit vector rotates by  $90^\circ$  with the positive z-axis and now points in the positive x-axis. Let's see which state these values correspond to. Substitute these angle values in (1.1). We get

$$|\psi\rangle = \cos(\pi/4)|0\rangle + e^{i*0} \sin(\pi/4)|1\rangle$$

$|\psi\rangle = \frac{1}{\sqrt{2}}|0\rangle + \frac{1}{\sqrt{2}}|1\rangle$  since  $\cos \pi/4 = \sin \pi/4 = 1/\sqrt{2}$  and  $e^0 = 1$ . So at this point, we have the state  $\frac{1}{\sqrt{2}}|0\rangle + \frac{1}{\sqrt{2}}|1\rangle$ . is so common in quantum computing that it is given a name:  $|+\rangle$ , i.e.,  $\frac{1}{\sqrt{2}}|0\rangle + \frac{1}{\sqrt{2}}|1\rangle$ .

## 1.5 Single qubit system and multiple qubit system

### 1.5.1 One-qubit system

A one-qubit or a single qubit system has two computational basis states  $|0\rangle$  and  $|1\rangle$ . The mathematical representation of a single qubit system  $|\psi\rangle$ , as described before, is as follows:  $|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$  Where  $\alpha$  and  $\beta$  are complex numbers.

Pictorially, we can represent a single qubit as shown in figure 1.3

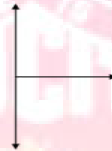


Figure 1.3: Single qubit system

### 1.5.2 Two-qubit system

A two-qubit system has four computational basis states denoted as  $|00\rangle$ ,  $|01\rangle$ ,  $|10\rangle$  and  $|11\rangle$  corresponding to the different possible values for the two qubits. Note that in this notation, the first number corresponds to the first qubit of the system and the second number corresponds to the second qubit of the system (for example,  $|01\rangle$  means that the first qubit is in state  $|0\rangle$  and the second qubit is in state  $|1\rangle$ ).

The mathematical representation of a two-qubit system, say,  $|\chi\rangle$  is as follows:

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

Where  $\alpha_{00}$ ,  $\alpha_{01}$ ,  $\alpha_{10}$ , and  $\alpha_{11}$  are complex numbers. These have a similar probabilistic interpretation as before, i.e.,  $|\alpha_{00}|^2$  is the probability that you would measure  $a_0$  for both the Qubits when you measured  $|\chi\rangle$ ,  $|\alpha_{01}|^2$  is the probability that you would measure  $a_0$  for the First qubit and  $a_1$  for these qubit when you measured the system  $|\chi\rangle$ , and soon. Pictorially, we can represent a two-qubit system as shown in figure 1.4.

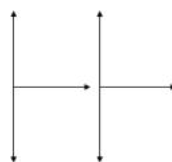


Figure 1.4: Two-qubit system

### 1.5.3 Extended to N qubits system

Multi-qubit system of N qubits has  $2^N$  computational basis states for example a state with 3 qubits has  $2^3$  thus computational basis states are denoted as  $|00...00\rangle$ ,  $|00...01\rangle$ ,  $|00...10\rangle$ ,  $|00...11\rangle$ , .....  $|11...11\rangle$ . Where each basis ket has n numbers. Pictorially, we can represent a n qubits system as shown in figure 1.5.

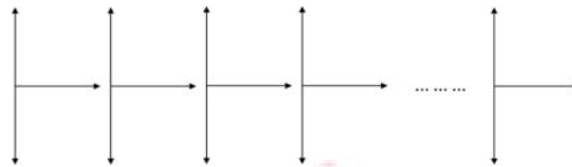


Figure 1.5: N-qubit system

### 1.6 . Model and Previous Year (Frequently asked) Questions

1. Define a bit and qubit and explain the properties of qubit.
  2. Define single and two qubits. Explain the Bloch sphere representation of qubit.
  3. Differentiate between classical and quantum computing.
  4. Explain the representation of qubit using Bloch sphere
  5. Explain in your own words how Moore's Law influenced the growth of computing power.
- Summarize the main reasons why Moore's Law is slowing down or ending.

## Chapter-2

### Dirac Representation and Matrix Operations

#### 2. Introduction

Linear algebra is crucial in quantum mechanics, offering a mathematical foundation for handling **quantum states using** state or ket vectors in complex vector spaces. **Linear operators** facilitate transformations on these states. Matrix representation allows efficient calculations by representing operators as matrices based on selected basis states. Matrix multiplication of operator and state matrices predicts quantum system evolution in quantum mechanics and is vital in quantum computing for algorithm design and computations.

In classical physics, Bra vectors represent forces and velocities and are denoted by, say,  $\mathbf{a}$  with specified length and direction. In quantum mechanics, ket vectors, say,  $|\psi\rangle$ , represent quantum states with complex amplitudes signifying state probabilities. While  $\hat{\mathbf{i}}$  and  $\hat{\mathbf{j}}$  provide a simple representation for two-dimensional space in classical physics,  $|0\rangle$  and  $|1\rangle$  states Enable powerful and efficient quantum computation.

In quantum computing, the qubit is the basic unit of information, similar to bits in classical computing. While classical bits exist in states of either 0 or 1, qubits can exist in state of 0, 1, or both simultaneously - a phenomenon called quantum superposition. The use of binary states (0 and 1) in qubits aligns with the binary nature of digital computing, enabling efficient representation and implementation.

#### 2.1. Matrix representation

We can write  $|0\rangle$  and  $|1\rangle$  as column vectors,  $|0\rangle = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$  and  $|1\rangle = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$  these are called column vectors as they have single column.

It is now convenient to represent the vectors notation of quantum states

$$|v\rangle = \sum_{i=1}^n \alpha_i |v_i\rangle$$

for  $n=2$   $|v\rangle = \alpha_1 |v_1\rangle + \alpha_2 |v_2\rangle$  where  $|v\rangle, |v_1\rangle, |v_2\rangle$  are ket vectors which can be represented as column matrices as  $|v_1\rangle = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$   $|v_2\rangle = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$

$$|v\rangle = \alpha_1 \begin{bmatrix} 1 \\ 0 \end{bmatrix} + \alpha_2 \begin{bmatrix} 0 \\ 1 \end{bmatrix}$$

Using the conventional method of adding matrix  $|\psi\rangle = [\alpha_1 \ \alpha_2]$

## 2.2. Identity Operator

The identity operator is an operator which when acts on a quantum state keeps it intact. So the action of the identity operator  $I$  on a general quantum state  $|\psi\rangle$  is given as  $I|\psi\rangle = |\psi\rangle$ . Now, let us find the matrix representation of the identity operator. To do that, we need to know the action of the operator on the basis states  $|0\rangle$  and  $|1\rangle$ .

But we already know that from the definition of the identity operator:

$$I|0\rangle = |0\rangle \text{ and } I|1\rangle = |1\rangle$$

Replacing the  $|0\rangle$  and  $|1\rangle$  states with their matrix representations, we have

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

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

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

**\*Show that  $\begin{pmatrix} 1 & -1 \end{pmatrix}$ ,  $\begin{pmatrix} 1 & 2 \end{pmatrix}$  and  $\begin{pmatrix} 2 & 1 \end{pmatrix}$  are linearly independent**

$$\alpha_1 \begin{bmatrix} 1 & -1 \end{bmatrix} + \alpha_2 \begin{bmatrix} 1 & 2 \end{bmatrix} + \alpha_3 \begin{bmatrix} 2 & 1 \end{bmatrix} = 0 \text{-----(1)}$$

$$\alpha_1 + \alpha_2 + 2\alpha_3 = 0$$

$$-\alpha_1 + 2\alpha_2 + \alpha_3 = 0$$

$$3\alpha_2 + 3\alpha_3 = 0 \quad \text{-----} \square \quad \alpha_2 = -\alpha_3$$

$$\alpha_1 - \alpha_2 + 2\alpha_3 = 0 \text{-----} \square \quad \alpha_1 + \alpha_3 = 0 \quad \text{-----} \square \quad \alpha_1 = -\alpha_3$$

put  $\alpha_3=1$ ,  $\alpha_2=-1$ ,  $\alpha_1=-1$  equation 1 we get

$$-1 \begin{bmatrix} 1 & -1 \end{bmatrix} + (-1) \begin{bmatrix} 1 & 2 \end{bmatrix} + 1 \begin{bmatrix} 2 & 1 \end{bmatrix} = 0$$

$$-1-1+2=0, \quad -1-2+1=0, \text{ they are linearly dependent.}$$

## 2.3. Pauli Matrices

There are three extremely useful operators called the **Pauli operators** which are used in quantum computing and quantum mechanics. These operators have different notations in different fields:  $\sigma_x$ ,  $\sigma_y$ ,  $\sigma_z$  or  $\sigma_1$ ,  $\sigma_2$ ,  $\sigma_3$  or  $X$ ,  $Y$ ,  $Z$ . We will use the latter notation.

The matrix representations of the Pauli operators are called the Pauli matrices. These are a set of three  $2 \times 2$  matrices with special properties (which we won't discuss here). Sometimes the identity matrix  $I$  is also clubbed with these matrices and thus is also written as  $\sigma_0$ . Here, we give you the Pauli matrices in the  $|0\rangle$  and  $|1\rangle$  basis.

$$\sigma_x = \sigma_1 = X = \begin{bmatrix} 0 & 1 & 1 & 0 \end{bmatrix}$$

$$\sigma_y = \sigma_2 = Y = \begin{bmatrix} 0 & -i & i & 0 \end{bmatrix}$$

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

#### 2.4. Action of the Pauli matrices on the basis states:

##### i) Pauli-I matrix operating on $|0\rangle$ and $|1\rangle$ basis states

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

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

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

##### ii) Pauli-X matrix operating on $|0\rangle$ and $|1\rangle$ basis states

$$X |0\rangle = |1\rangle$$

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

$$X |0\rangle = |1\rangle$$

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

$$X |1\rangle = |0\rangle$$

##### iii) Pauli-Y matrix operating on $|0\rangle$ and $|1\rangle$ basis states

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

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

##### iv) Pauli-Z matrix operating on $|0\rangle$ and $|1\rangle$ basis states

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

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

| Matrix | $ 0\rangle$   | $ 1\rangle$  |
|--------|---------------|--------------|
| I      | $ 0\rangle$   | $ 1\rangle$  |
| X      | $ 1\rangle$   | $ 0\rangle$  |
| Y      | $-i 1\rangle$ | $i 0\rangle$ |
| Z      | $ 0\rangle$   | $- 1\rangle$ |

## 2.5 Conjugate of a matrix

The conjugate of a matrix is obtained by replacing each element of the matrix with its complex conjugate. For example, consider the following matrix

$$A = [3 \ 1 \ - \ 2i \ 0 \ - \ 4i]$$

The conjugate of A, denoted by  $A^*$ , is given by

$$A^* = [3 \ 1 \ + \ 2i \ 0 \ 4i]$$

## 2.6 Transpose of a matrix

The transpose of a matrix is obtained by interchanging the rows into columns or vice versa of the matrix. For example, consider the following matrix  $A = [3 \ 1 \ - \ 2i \ 0 \ - \ 4i]$

The transpose of A, denoted by  $A^T$ , is given by  $A^T = [3 \ 0 \ 1 \ - \ 2i \ - \ 4i]$

## 2.7 Hermitian adjoint of a matrix

The Hermitian adjoint of a matrix is obtained by taking the conjugate **and** the transpose of the matrix. Hence the Hermitian adjoint is also called the conjugate-transpose of a matrix. For example, consider the following matrix

$$B = [3 \ 1 \ - \ 2i \ 0 \ - \ 4i]$$

The Hermitian adjoint of B, denoted by  $B^\dagger$  ( $\dagger$  is called “dagger”), is written as

$$\begin{aligned} B^\dagger &= (B^*)^T = [3 \ 1 \ + \ 2i \ 0 \ 4i]^T \\ &= [3 \ 0 \ 1 \ - \ 2i \ - \ 4i] \end{aligned}$$

## 2.8 Hermitian Matrix

If the Hermitian adjoint of a matrix is equal to the matrix itself, that matrix is called a Hermitian matrix. A Hermitian matrix is also called a self-adjoint matrix. So A is said to be a Hermitian matrix if it satisfies the following condition

$$A^\dagger = A$$

For example, consider the matrix

$$C = [1 \ 1 \ - \ 2i \ 1 \ + \ 2i \ 3]$$

Consider the Hermitian adjoint of C

$$C^\dagger = (C^*)^T$$

$$= [1 \ 1 + 2i \ 1 - 2i \ 3]$$

$$= [1 \ 1 - 2i \ 1 + 2i \ 3]$$

$$C^\dagger = C \quad \text{So } C \text{ is a Hermitian matrix.}$$

## 2.9 Unitary Matrix

A matrix U is said to be unitary if it satisfies the following condition  $UU^\dagger = U^\dagger U = I$  where I is the identity matrix.

$$UU^\dagger = U^\dagger U = I = 1$$

$$\text{Let } U = [0 \ 1 \ 1 \ 0]$$

$$\text{Conjugate of U is } U^* = [0 \ 1 \ 1 \ 0]$$

$$\text{Transpose of U is } U^T = [0 \ 1 \ 1 \ 0]$$

$$UU^\dagger = [0 \ 1 \ 1 \ 0][0 \ 1 \ 1 \ 0] = [0 \ 1 \ 1 \ 0] = I$$

$$U^\dagger U = [0 \ 1 \ 1 \ 0][0 \ 1 \ 1 \ 0] = [0 \ 1 \ 1 \ 0] = I$$

$$UU^\dagger = I$$

## 2.10 Column and Row Matrices

As we have discussed earlier, ket vectors can be described by column matrices. For example, the ket vector  $|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$  can be written as a column matrix as follows  $[\alpha \ \beta]$  where  $\alpha$  and  $\beta$  are generally complex numbers.

In the theory of linear vector spaces, there is a concept called the dual vector space, which says that for every vector in a vector space, there exists a dual vector (or simply called the dual) in the dual vector space (this is a very simplified statement). So in the context of ket vectors, there exists dual of ket vectors which are called bra vectors. These bra vectors are represented by row matrices. We can convert a ket vector to a bra vector by taking its Hermitian adjoint.

For example, consider the same ket vector as above,  $|\psi\rangle$ . The dual or the bra form of this vector is given as

$$|\psi\rangle^\dagger = \alpha^* + \beta^*$$

In fact, there is a better convention to denote bra vectors:  $\langle\psi| = |\psi\rangle^\dagger$ . So if  $|\psi\rangle = [\alpha \ \beta]$  then  $\langle\psi| = [\alpha^* \ \beta^*]$  the Dirac's bra-ket notation.

For example, consider the following ket vector  $|\phi\rangle = [1/\sqrt{2} \ i/\sqrt{2}]$  Its bra form is then written as

$$\langle\phi| = [1/\sqrt{2} \ -i/\sqrt{2}]$$

Now, just as  $\langle\psi| = |\psi\rangle^\dagger$ , we can also write  $|\psi\rangle = \langle\psi|^\dagger$  since  $|\psi\rangle^\dagger^\dagger = |\psi\rangle$ .

### 2.11 Inner Product

One way of multiplying two arrow vectors is taking their dot product. If  $\vec{u} = u_1\hat{i} + u_2\hat{j}$  and  $\vec{v} = v_1\hat{i} + v_2\hat{j}$  are two arrow vectors, then their dot product is denoted by  $\vec{u} \cdot \vec{v}$  and given by  $\vec{u} \cdot \vec{v} = u_1v_1 + u_2v_2$ . So if we use the matrix representation of vectors, the dot product of two vectors can be written as the matrix multiplication of a row matrix with a column matrix. In a similar manner, we have the inner product of two ket vectors. Consider two ket vectors

$$|\phi\rangle = \alpha_1 |0\rangle + \beta_1 |1\rangle \text{ and } |\psi\rangle = \alpha_2 |0\rangle + \beta_2 |1\rangle \text{ Or, in matrix representation we have } \alpha_1, \alpha_2$$

$$|\phi\rangle = [\alpha_1 \ \beta_1] \text{ and } |\psi\rangle = [\alpha_2 \ \beta_2] \text{ The inner product of } |\phi\rangle \text{ with } |\psi\rangle \text{ is denoted as } \langle\phi|\psi\rangle \text{ (notice the single vertical line in the center) and given by } \langle\phi|\psi\rangle = \alpha_1^* \alpha_2 + \beta_1^* \beta_2.$$

Note that the left side of the inner product is the bra form of the given vector  $|\phi\rangle$ . Unlike the dot product of arrow vectors, the inner product of ket vectors is not commutative, i.e.,  $\langle\phi|\psi\rangle \neq \langle\psi|\phi\rangle$  in general. For example, consider the same two ket vectors,  $|\phi\rangle$  and  $|\psi\rangle$ . The inner product of  $|\phi\rangle$  with  $|\psi\rangle$  is as shown above. But there is a relation between the two inner products. In general we can actually write  $\langle\phi|\psi\rangle = \langle\psi|\phi\rangle^*$  where  $\langle\psi|\phi\rangle^*$  is the complex conjugate of  $\langle\psi|\phi\rangle$ .

### 2.12 Probability

Consider a 2-dimensional arrow vector in physical space  $\vec{a} = 5\hat{i} + 2\hat{j}$ . What is the meaning of the coefficients of the unit vectors, in this case 5 and 2? Referring to figure 2.1, we see that the vector has 5 units of contribution from the  $\hat{i}$  direction whereas only 2 units of contribution from the  $\hat{j}$  direction. In other words, we could say that the vector is tilted more toward the  $\hat{i}$  direction than the  $\hat{j}$  direction. This idea is important for the following discussion. Now let us consider a ket vector, say,  $|\psi\rangle = \alpha |0\rangle + \beta |1\rangle$ , where  $\alpha$  and  $\beta$  are in general complex numbers. What is the meaning of these coefficients? In the context of quantum mechanics, these are called probability amplitudes. The ket vector  $|\psi\rangle$  represents a quantum state that is in a superposition of the 0 and 1 classical states. By itself, the quantum state will be in this superposition state indefinitely, but if the state is measured, it will collapse to (or become) any one of the two classical states (this is one of the fundamental postulates of quantum mechanics).

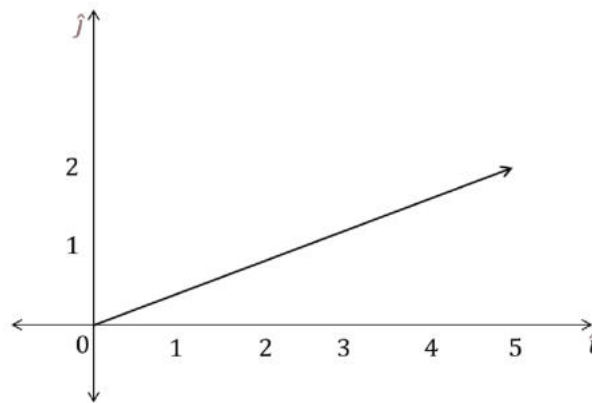


Figure 2.1: Example arrow vector

$\alpha$  and  $\beta$  tell us what is the probability of getting a 0 or a 1 respectively as the outcome of the measurement of  $|\psi\rangle$ . Actually, since these coefficients are complex numbers, they are not actual probabilities, but their modulus squares are. In other words,  $|\alpha|^2$  is the probability of getting a 0 state after measuring  $|\psi\rangle$ , whereas  $|\beta|^2$  is the probability of getting a 1 state after measuring  $|\psi\rangle$ . As the 0 and 1 states are mutually exclusive measurement results, we can say that the sum of  $|\alpha|^2$  and  $|\beta|^2$  is equal to 1 (since the sum of probabilities of different events is always equal to 1), i.e.,  $|\alpha|^2 + |\beta|^2 = 1$ . So all ket vectors in quantum mechanics and quantum computing have to satisfy this condition. There is also another way to represent this condition, in the bra-ket notation. Consider the inner product of  $|\psi\rangle$  with itself:  $\langle\psi|\psi\rangle = \alpha^*\alpha + \beta^*\beta$ ,  $\langle\psi|\psi\rangle = |\alpha|^2 + |\beta|^2$

## 2.13 Orthogonality and Orthonormality

### 2.13.1 Orthogonality

Two ket vectors  $|\psi\rangle$  and  $|\phi\rangle$  are said to be orthogonal to each other if their inner product is zero. This is similar to (but not the same as) saying that two arrow vectors are orthogonal (or perpendicular) if their dot product is zero. (Note that you cannot use the word 'perpendicular' when referring to ket vectors.) Mathematically, the orthogonality condition is written as  $\langle\psi|\phi\rangle = 0$ .

### 2.13.2 Orthonormality

Two ket vectors are said to be orthonormal if they are orthogonal to each other *and* are individually normalized. For example,  $|0\rangle$  and  $|1\rangle$  are orthonormal since they are orthogonal to each other and are both normalized.

The orthonormality of  $|0\rangle$  and  $|1\rangle$  is described by four equations:

$$\langle 0|0\rangle = 1, \langle 1|1\rangle = 1$$

$$\langle 0|1\rangle = 0, \langle 1|0\rangle = 0$$

There is another concise way of writing these four equations, as a single equation

$\langle m|n \rangle = \delta_{mn}$  where  $m, n \in \{0, 1\}$  and  $\delta_{mn} = 1$  if  $m = n$  0 if  $m \neq n$  is called the Kronecker Delta symbol. One can easily verify that for different values of  $m$  and  $n$  we get the four equations stated.

**\*\*Given**  $|\psi\rangle = [\alpha_1 \ \alpha_2]$ ,  $|\phi\rangle = [\beta_1 \ \beta_2]$ . **Show that**  $\langle \phi|\psi \rangle = \langle \psi|\phi \rangle^*$

Solution:

$$\langle \psi| = [\alpha_1^* \ \alpha_2^*]$$

$$\langle \phi| = [\beta_1^* \ \beta_2^*]$$

$$\langle \psi|\phi \rangle = [\alpha_1^* \ \alpha_2^*] [\beta_1 \ \beta_2]$$

$$\langle \psi|\phi \rangle = (\alpha_1^* \beta_1 + \alpha_2^* \beta_2)$$

$$\langle \phi|\psi \rangle = [\beta_1^* \ \beta_2^*] [\alpha_1 \ \alpha_2]$$

$$\langle \phi|\psi \rangle = (\beta_1^* \alpha_1 + \beta_2^* \alpha_2) \text{ -----(1)}$$

$$\langle \psi|\phi \rangle^* = [(\alpha_1^* \beta_1 + \alpha_2^* \beta_2)]^*$$

$$\langle \psi|\phi \rangle^* = (\beta_1^* \alpha_1 + \beta_2^* \alpha_2) \text{ ) -----(2)}$$

Comparing equations (1) and (2)

$$\text{We get } \langle \phi|\psi \rangle = \langle \psi|\phi \rangle^*$$

### 3. Quantum Gates

#### 3.Introduction

Quantum circuits are an essential tool in quantum computing, allowing us to perform computations using qubits. Similar to classical circuits in conventional computers, quantum circuits are made up of a series of operations or gates that manipulate the state of qubits.

A quantum gate can be thought of as an operation that transforms the state of a qubit or multiple qubits. Quantum gates come in various types, each performing a specific quantum operation, such as changing the phase of a qubit, rotating the qubit's state in the Bloch sphere, or entangling multiple qubits. The ability to perform operations on qubits using quantum gates is a key feature of quantum computing, as it allows for the development of quantum algorithms that can solve certain problems exponentially faster than classical algorithms.

Let's take a look at the elements of a quantum circuit. A simple quantum circuit containing two quantum gates (represented by boxes) and three inputs is shown in figure 3.1.

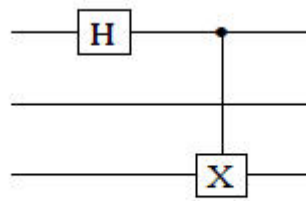


Figure 3.1: A quantum circuit

The circuit is to be read from left to right. Each horizontal line in the circuit represents a wire in the quantum circuit. This wire does not necessarily correspond to a physical wire; it may correspond instead to the passage of time, or perhaps to a physical particle moving from one location to another through space. It is conventional to assume that the state input to the circuit is a computational basis state, usually the state consisting of all  $|0\rangle$ s unless otherwise specified. The label inside the box denotes the kind of gate; for example, H denotes the Hadamard gate and X denotes the Pauli-X gate. The vertical line starting from the solid dot and ending on the X gate represents a connection between the top line and the bottom line without affecting the middle line. If the middle line also were to be affected by the vertical line, then their intersection would have a solid dot. In other words, if there is no solid dot at the intersection of two lines, then those lines do not interact with each other. To get the result of the action of a quantum gate on an input state, we multiply the matrix representation of the gate with the matrix (column vector) representation of the input state. On the other hand, if we don't already know the matrix representation of the gate but know the result of the action of the gate on an input state, we can find out the matrix representation of the gate in some suitable basis.

### 3.1 Single Qubit Gates

Single qubit gates are a type of quantum gate that operates on a single qubit. These gates are fundamental building blocks in quantum computing and are used to manipulate and transform the state of individual qubits.

#### 3.1.1 Quantum NOT gate (Pauli-X gate)

The Pauli-X gate has the action of flipping  $|0\rangle$  to  $|1\rangle$  and  $|1\rangle$  to  $|0\rangle$ . Since this is similar to the action of the classical NOT gate, the Pauli-X gate is also called the Quantum NOT gate (or sometimes also called the bit-flip gate). To see how we get the definition, consider the Pauli-X matrix acting on the  $|0\rangle$  state:  $X|0\rangle = |1\rangle$ . Similarly for the  $|1\rangle$  state:  $X|1\rangle = |0\rangle$ . But what if we have a general superposition state such as  $|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$  or in matrix form  $|\psi\rangle = [\alpha \ \beta]$ . We can write The action of the Pauli-X gate on a general qubit state is to switch the coefficients. This can be seen explicitly if we consider the bra-ket notation:  $X|\psi\rangle = X(\alpha|0\rangle + \beta|1\rangle) = \alpha X|0\rangle + \beta X|1\rangle = \alpha|1\rangle + \beta|0\rangle$ .  $X|\psi\rangle = \beta|0\rangle + \alpha|1\rangle$

The action of the quantum NOT gate is summarized as shown in table 3.2.

| Input                                | Output                               |
|--------------------------------------|--------------------------------------|
| $ 0\rangle$                          | $ 1\rangle$                          |
| $ 1\rangle$                          | $ 0\rangle$                          |
| $\alpha  0\rangle + \beta  1\rangle$ | $\beta  0\rangle + \alpha  1\rangle$ |

Figure 3.2: Quantum NOT gate

### 3.1.2 Pauli-Z gate

The Pauli-Z gate has an action similar to that of the identity operator except that when acting on  $|1\rangle$ , it introduces a phase of -1. Hence it is also called the phase-flip gate. So mathematically, the definition of the Z gate is

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

To see how we get the definition, consider the Pauli-Z matrix acting on the  $|0\rangle$  state:  $Z |0\rangle = |0\rangle$

Similarly for the  $|1\rangle$  state:  $Z |1\rangle = -|1\rangle$  But what if we have a general superposition state such as

$$|\psi\rangle = \alpha |0\rangle + \beta |1\rangle \text{ or in matrix form } |\psi\rangle = [\alpha \ \beta]$$

We can write The action of the Pauli-Z gate on a general qubit state is to flip the sign of the second coefficient. This can be seen explicitly if we consider the bra-ket notation:

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

The action of the Z gate is summarized as shown in table 3.3b.

| Input                                | Output                               |
|--------------------------------------|--------------------------------------|
| $ 0\rangle$                          | $ 0\rangle$                          |
| $ 1\rangle$                          | $- 1\rangle$                         |
| $\alpha  0\rangle + \beta  1\rangle$ | $\alpha  0\rangle - \beta  1\rangle$ |

Table 3.3b Truth table

### 3.1.3 Pauli-Y gate

The Pauli-Y gate has a somewhat combined action of both the X and Z gates. It flips  $|0\rangle$  to  $|1\rangle$  and vice versa along with introducing phase factors in both cases. Hence it is also called the bit-and-phase-flip gate. Mathematically, the definition of the Y gate is

$$Y |0\rangle = i |1\rangle, \quad Y |1\rangle = -i |0\rangle$$

To see how we get the definition, consider the Pauli-Y matrix acting on the  $|0\rangle$  and  $|1\rangle$  states:

$$Y |0\rangle = i |1\rangle$$

$$Y |1\rangle = -i |0\rangle$$

But what if we have a general superposition state such as  $|\psi\rangle = \alpha |0\rangle + \beta |1\rangle$  or in matrix

Form  $|\psi\rangle = [\alpha \ \beta]$  We can write  $Y|\psi\rangle = -i[\alpha \ \beta]$

The action of the Pauli-Y gate on a general qubit state is to switch the coefficients and flip the sign of the second coefficient, upto a global phase factor. This can be seen explicitly if we consider the bra-ket notation:

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

The action of the Y gate is summarized as shown in table 3.4.

| Input                              | Output                                 |
|------------------------------------|----------------------------------------|
| $ 0\rangle$                        | $i 1\rangle$                           |
| $ 1\rangle$                        | $-i 0\rangle$                          |
| $\alpha 0\rangle + \beta 1\rangle$ | $-i(\beta 0\rangle - \alpha 1\rangle)$ |

Table 3.4 Truth table Pauli-Y gate

### 3.1.4 Hadamard gate

The Hadamard gate is one of the most important single-qubit gates. In simple terms, the action of the Hadamard gate on the  $|0\rangle$  and  $|1\rangle$  states is to produce superposition states. The definition of the Hadamard gate (represented as H) is as follows

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

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

These special superposition states are encountered so frequently in quantum computing that they are given their own names:

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

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

They are called the "plus" state and the "minus" state respectively, for obvious reasons.

Note: Just as the  $|0\rangle$  and  $|1\rangle$  states form something called the computational basis in our single-qubit space, the  $|+\rangle$  and  $|-\rangle$  states form an alternate set of basis in our single-qubit space and are called the Hadamard basis. This distinction is important to note because the matrix representations of states and gates are always written with respect to a particular basis. In our case, we use the computational basis. If we used the Hadamard basis instead, the matrix elements

of all the entities would change. Another thing to be noted is that we can always express one set of basis in terms of the other set of basis and vice versa. For example, in the above expressions, the Hadamard basis is written in terms of the computational basis. One can also rearrange the terms and also write the computational basis in terms of the Hadamard basis.

### 3.1.5 Phase gate (S gate)

The S gate can also be called the square-root-of-Z gate, because we get the matrix representation of the S gate by taking the square root of the elements of the Z gate matrix. Therefore we can write:

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

Let's look at the action of the S gate on the  $|0\rangle$  and  $|1\rangle$  states

The action of the S gate on the  $|0\rangle$  is

$$S|0\rangle = \begin{bmatrix} 1 & 0 & 0 & i \end{bmatrix} \begin{bmatrix} 1 & 0 \end{bmatrix} = \begin{bmatrix} 1 & 0 \end{bmatrix} \text{ that is output is } |0\rangle,$$

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

The action of the S gate on the  $|1\rangle$  is

$$S|1\rangle = \begin{bmatrix} 1 & 0 & 0 & i \end{bmatrix} \begin{bmatrix} 0 & 1 \end{bmatrix} = \begin{bmatrix} 1 & i \end{bmatrix} \text{ that is output is } i|1\rangle$$

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

It transforms the state  $\alpha|0\rangle + \beta|1\rangle$  as

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

To the state  $\alpha|0\rangle + i\beta|1\rangle$

The action of the S gate is summarized as shown in table 3.6

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

### 3.6.S gate Truth table

### 3.1.6 T gate

T gate rotates the qubit state by  $\pi/4$  radian ( $45^\circ$ ) about the z axis.

$$T = \begin{bmatrix} 1 & 0 & 0 & \exp(\frac{i\pi}{4}) \end{bmatrix}$$

The T gate (or  $\pi/8$  gate) can also be called the square-root-of-S gate, because we get the matrix representation of the T gate by taking the square root of the elements of the S gate matrix.

Therefore we can write:

$$T = \exp(\frac{i\pi}{8}) \begin{bmatrix} \exp(\frac{i\pi}{8}) & 0 & 0 & \exp(\frac{i\pi}{8}) \end{bmatrix}$$

If the input is  $|0\rangle$ , then the out put is

$$\begin{bmatrix} 1 & 0 & 0 & \exp(\frac{i\pi}{4}) \end{bmatrix} \begin{bmatrix} 1 & 0 \end{bmatrix} = \begin{bmatrix} 1 & 0 \end{bmatrix}$$

That is state  $T|0\rangle = |0\rangle$

If the input is  $|1\rangle$ , then the out put is

$$\begin{bmatrix} 1 & 0 & 0 & \exp(\frac{i\pi}{4}) \end{bmatrix} \begin{bmatrix} 0 & 1 \end{bmatrix} = \begin{bmatrix} 0 & \exp(\frac{i\pi}{4}) \end{bmatrix} = \exp(\frac{i\pi}{4}) \begin{bmatrix} 0 & 1 \end{bmatrix}$$

That is state  $T|1\rangle = \exp(\frac{i\pi}{4})|1\rangle$

It transforms the state  $\alpha|0\rangle + \beta|1\rangle$  as

$$\begin{bmatrix} 1 & 0 & 0 & \exp(\frac{i\pi}{4}) \end{bmatrix} \begin{bmatrix} \alpha & \beta \end{bmatrix} = \begin{bmatrix} \alpha & \beta \exp(\frac{i\pi}{4}) \end{bmatrix}$$

to  $\alpha|0\rangle + \beta \exp(\frac{i\pi}{4})|1\rangle$

| Input                              | Output                                                 |
|------------------------------------|--------------------------------------------------------|
| $ 0\rangle$                        | $ 0\rangle$                                            |
| $ 1\rangle$                        | $\exp(\frac{i\pi}{4}) 1\rangle$                        |
| $\alpha 0\rangle + \beta 1\rangle$ | $\alpha 0\rangle + \exp(\frac{i\pi}{4})\beta 1\rangle$ |

Figure 3.6: S gate truth Table

### 3.2 Multiple Qubit Gates

Multiple qubit gates are a crucial component of quantum computing, providing the means to manipulate and entangle multiple qubits simultaneously. Unlike single-qubit gates that act on individual qubits, multiple qubit gates operate on two or more qubits, enabling more complex operations and interactions. These gates are designed to exploit the unique properties of entangled quantum states, allowing for parallel computations and intricate quantum information processing. Multiple qubit gates, such as the controlled-NOT (CNOT) gate and the Toffoli gate, serve as the

building blocks for constructing quantum circuits that can perform powerful calculations and solve problems that are beyond the reach of classical computing.

### Controlled gates

These gates act on something called a target qubit and one or more control qubits. The action of these gates depends on the state of the control qubit. The target qubit will be affected by the gate only when the control qubit(s) is(are) in the  $|1\rangle$  state(s), otherwise the target qubit remains as is. In any case, the control qubit(s) always remain the same after the action of such gates.

#### 3.2.1 Controlled-NOT (CNOT) gate

The CNOT gate, short for Controlled-NOT gate, (or also called the CX gate) is a fundamental two-qubit gate that plays a central role in quantum circuits. The CNOT gate acts on two qubits, known as the control qubit and the target qubit. If the control qubit is in the  $|1\rangle$  state, the CNOT gate flips the state of the target qubit (action of the X gate). However, if the control qubit is in the  $|0\rangle$  state, the target qubit remains unchanged. And in either way, the control qubit remains unchanged. This gate's ability to flip the target qubit based on the state of the control qubit makes it an essential tool for creating entangled states and performing controlled operations in quantum circuits. The CNOT gate's versatility and its role in various quantum algorithms highlight its significance in quantum computing and quantum information processing. The circuit for the CNOT gate is as shown in figure 3.8.

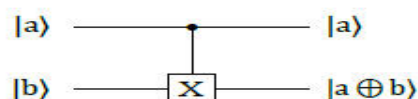


Figure 3.8: Circuit symbol of CNOT gate

The above circuit is equivalently and more frequently written as shown in figure 3.9.

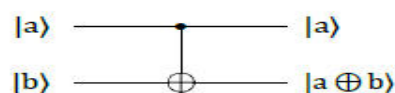


Figure 3.9: Circuit symbol of CNOT gate

(The  $a$ ,  $b$ , and  $a \oplus b$  symbols will be explained later.)

So according to the above description of the action of the CNOT gate, we can write the outputs for the four different input states as follows:

- Input  $|00\rangle$ : Since the first qubit (control) is 0, the second qubit (target) which is also 0 remains as is. So the output is  $|00\rangle$ .

- Input  $|01\rangle$ : Since the first qubit (control) is 0, the second qubit (target) which is 1 remains as is. So the output is  $|01\rangle$ .
- Input  $|10\rangle$ : Since the first qubit (control) is 1, the second qubit (target) which is 0 flips to 1. So the output is  $|11\rangle$ .
- Input  $|11\rangle$ : Since the first qubit (control) is 1, the second qubit (target) which is also 1 flips to 0. So the output is  $|10\rangle$ .

The matrix representation of the CNOT gate is as follows:

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

All the above cases are summarized in the truth table as follows:

| Input |   | Output |
|-------|---|--------|
| 0     | 0 | 0      |
| 0     | 1 | 1      |
| 1     | 0 | 1      |
| 1     | 1 | 0      |

Figure 3.11: Truth table of XOR gate

So we can write a general expression for the action of the CNOT gate on a two-qubit system in terms of the XOR operation. If the input state is  $|a, b\rangle$  where  $a, b \in \{0, 1\}$ , then the result is  $U_{\text{CNOT}} |a, b\rangle = |a, a \oplus b\rangle$  where  $\oplus$  represents the XOR operation.

Note: The XOR operation is sometimes also called bit-wise addition or modulo-2 addition.

Here are all the rules for the XOR operation (this can also be seen from the truth table of XOR):  $0 \oplus 0 = 0$ ,  $0 \oplus 1 = 1$ ,  $1 \oplus 0 = 1$ ,  $1 \oplus 1 = 0$

All these rules can be generalized into a simple mathematical expression:

$x \oplus y = (0 \text{ if } x = y, 1 \text{ if } x \neq y)$  where  $x, y \in \{0, 1\}$ . The following general rules are also helpful  $0 \oplus x = x$ ,  $x \oplus 0 = x$  and  $1 \oplus x = x \oplus 1 = x^{-}$  where  $x \in \{0, 1\}$  and  $x^{-}$  is the inverse of  $x$ .

### 3.2.2 SWAP gate

The SWAP gate is a two-qubit gate. Its action is to swap the qubits of the input states, i.e., if the input state is  $|a, b\rangle$ , then the output of is  $|b, a\rangle$ . We can write this mathematically as

$$U_{\text{SWAP}} |a, b\rangle = |b, a\rangle$$

The circuit representation of the SWAP gate is as shown in figure 3.12.

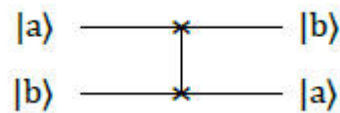


Figure 3.12: Circuit symbol of SWAP gate

The matrix representation of the SWAP gate is as follows:

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

According to the definition of the SWAP gate, the truth table is written as shown in table 3.13.

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

Figure 3.13: Truth table of SWAP gate

But how is it that the two qubits are swapped due to this gate? The SWAP gate actually consists of three consecutive CNOT gates aligned in an alternating manner as shown in figure 3.14. The circuit shown in figure 3.12 is actually an equivalent/conventional way of representing the circuit shown in figure 3.14.

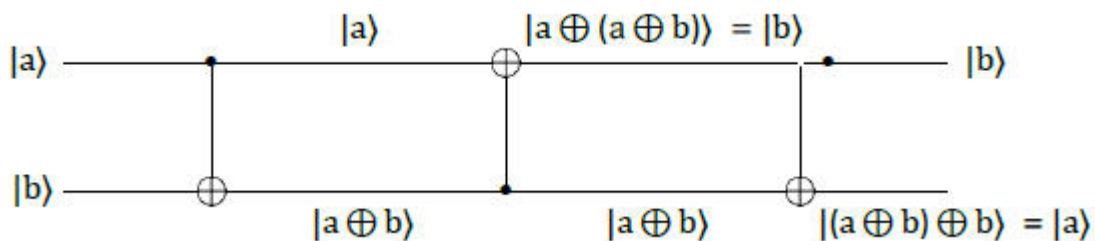


Figure 3.14: Working circuit symbol of SWAP gate

Let's look at the actions step by step. The input state is  $|a, b\rangle$ .

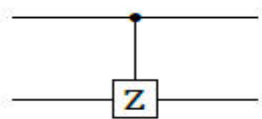
For the first CNOT gate,  $|a\rangle$  is the control qubit,  $|b\rangle$  is the target qubit. So the result of the action of the first CNOT gate is  $|a, a \oplus b\rangle$ .

The output of the first CNOT gate will be the input to the second CNOT gate, but this time the bottom qubit  $|a \oplus b\rangle$  is the control and the top qubit  $|a\rangle$  is the target. So the result of the action of the second CNOT gate is  $|a \oplus (a \oplus b), a \oplus b\rangle$ . But from the rules of the XOR operation, we can write  $a \oplus (a \oplus b) = (a \oplus a) \oplus b = 0 \oplus b = b$ . So the output of the second CNOT gate is  $|b, a \oplus b\rangle$ . The output of the second CNOT gate will be the input to the third CNOT gate where

$|b\rangle$  is the control qubit and  $|a \oplus b\rangle$  is the target qubit. So the result of the action of the third CNOT gate is  $|b, (a \oplus b) \oplus b\rangle$ . Again, we can write  $(a \oplus b) \oplus b = a \oplus (b \oplus b) = a \oplus 0 = a$ . So finally, the output of the third CNOT gate is  $|b, a\rangle$ .

### 3.2.3 Controlled-Z gate

The controlled-Z gate (also called the CZ gate) is again a two-qubit gate, acting on a control qubit and a target qubit. Only when the control qubit is  $|1\rangle$ , the Z gate will act on the target qubit. On the other hand, if the control qubit is  $|0\rangle$ , then the target qubit remains unchanged. In both cases, the control qubit stays the same after the action of the gate. The circuit for the CZ gate and its truth table are shown in figure 3.15.



(a) Circuit symbol

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

(b) Truth table

Figure 3.15: Controlled-Z gate

Remember the action of the Z gate:  $Z|0\rangle = |0\rangle$  and  $Z|1\rangle = -|1\rangle$ .

The matrix representation of the CZ gate is as follows:

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

### 3.2.4 Toffoli gate

The Toffoli gate (also called the controlled-controlled-NOT gate or simply the CCNOT gate) is a vital three-qubit gate that allows for controlled operations on multiple qubits. The Toffoli gate acts on three qubits, with two control qubits and one target qubit. If *both* control qubits are in the state  $|1\rangle$ , the Toffoli gate flips the state of the target qubit; otherwise, the target qubit remains unchanged. And just as in the other controlled gates discussed above, the control qubits remain the same after the action of the gate. The circuit representation for the Toffoli gate is as shown in figure 3.16.



### 3.4 Numericals

1. Using Matrix multiplication show that on applying Hadamard gate twice to a  $|0\rangle$  results in its original state.
2. Using two X-gates in series show that two not gates in series are equivalent to a quantum wire.
3. Show that the Hadamard Gate is Unitary.
4. Two qubits are passed through a CNOT gate. If the first qubit is the control qubit then what is the output for the following initial states 1)  $|00\rangle$ , 2)  $|01\rangle$ , and 3)  $|11\rangle$ .
5. Show that S gate can be formed by connecting two T gates in series.

