Introduction to Quantum Computing

Kathrin Spendier

Wednesday 3rd May, 2023

Objective: The lecture aims to cover the fundamentals of computation by circuits, logic gates, and the differences between classical bits and qubits. It will also introduce the postulates of quantum computing, providing a foundation for understanding the principles that differentiate quantum computing from classical computing. Furthermore, the lecture will cover the Bloch sphere, superposition, complex numbers, measurement, linear algebra, and the quantum circuit model. By the end of the lecture, students should have a solid grasp of these principles and be better equipped to explore more advanced topics and applications in quantum computing.

1 Introduction

We just discussed Turing machines, which serve as a theoretical model for computing and laid the foundation for understanding computability. We explored the concept of decidability and the limits of what can be computed. We then delved into classes of complexity, particularly P, NP, and NP-Complete problems, to understand the relationship between problem difficulty and the computational resources required to solve them. Furthermore, we introduced probabilistic computation to solve complex problems with inherent uncertainty. We also covered the concepts of evolution and measurement in the context of probabilistic computing. We highlighted the importance of understanding computational complexity and probabilistic methods to tackle real-world problems.

During the next hour, we will dive into the fascinating world of quantum computing. Quantum computing requires a fundamentally different approach to information processing, involving encoding problems into qubits and creating algorithms that take advantage of quantum mechanical concepts like superposition and entanglement for faster computation.

One can imagine quantum computing as a powerful tool for encoding statistical assumptions about the world into a quantum circuit, allowing for analysis and inference of relationships between variables and making statistical predictions. While this process differs from classical programming, it shares similarities with the data science process of applying statistical analyses to datasets.

Using quantum computing, we can gain insights into complex systems and relationships beyond the capabilities of classical computing.

We will start with the fundamentals of computation by circuits, specifically logic gates and circuits. We will then compare classical bits and quantum bits, or qubits, to understand the unique properties that give quantum computers their immense potential.

Furthermore, we will introduce the postulates of quantum computing, providing a foundation for understanding the principles that differentiate quantum computing from classical computing. These postulates will help establish a solid theoretical background for exploring quantum algorithms and computation.

Next, we will explore the Bloch sphere, which visually represents qubits and their states. We'll discuss the concepts of superposition and measurement and how they are represented on the sphere. Additionally, we will cover the importance of complex numbers and linear algebra in the context of quantum computing.

Finally, we will introduce the quantum circuit model, which serves as the foundation for quantum algorithms and computation. Throughout this lecture, we will engage in active exercises and discussions to reinforce these concepts, providing you with a solid understanding of the principles underlying quantum computing.

2 Computation by Circuits: Logic Gates and Circuits (10 minutes)

2.1 Discussion

Prompt two students to share real-life examples of logic gates and their applications

1. Arithmetic operations: Logic gates are utilized in circuits like adders and subtractors to perform basic arithmetic operations, which are essential components of calculators and computers.

2. Traffic light controllers: The control system for traffic lights often relies on logic gates to manage the timing and sequencing of the lights, ensuring smooth and safe traffic flow.

3. Elevator control systems: Logic gates are used in elevator control systems to determine the elevator's movement, based on the floor selection and other relevant signals (e.g., door open/close).

4. Alarm systems: Home security systems and burglar alarms use logic gates to process input from various sensors (e.g., door/window contacts, motion detectors) and determine whether to trigger the alarm.

5. Washing machines: Logic gates control the various functions of washing machines, such as water levels, temperature settings, and spin cycles, based on user input and sensor data.

6. Digital clocks: Logic gates are used in digital clocks to manage the display of time and execute functions like setting alarms and timers.

2.2 Theory

Logic gates are the fundamental building blocks of digital circuits. These gates perform basic Boolean operations, such as AND, OR, and NOT, on input signals (0s and 1s) to produce an output.

Introduce three primary gates:

- AND Gate: This gate has two inputs and one output. The output is true (1) only when both inputs are true (1).

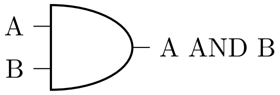


Figure 1: AND gate symbol and truth table

- OR Gate: This gate also has two inputs and one output. The output is true (1) when either or both inputs are true (1).

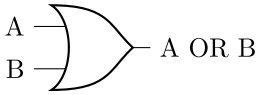


Figure 2: OR gate symbol and truth table

- NOT Gate: This gate has one input and one output. The output is the negation of the input, meaning it inverts the input (0 becomes 1, and 1 becomes 0).

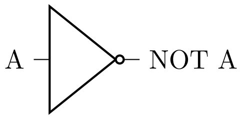


Figure 3: NOT gate symbol and truth table

Adding logic gates together creates a circuit that can implement an algorithm.

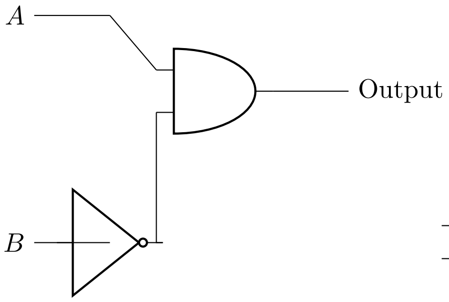
2.3 Exercise: Design a Simple Logical Circuit

Objective: Design a circuit using basic logic gates that control a garage door opener with two input signals – a remote control and an obstacle detector.

Solution: The garage door should open when the remote control is activated, and it should not open if there's an obstacle detected, even if the remote control signal is on.

Assign the input signals: Let A represent the remote control signal (1 for active, 0 for inactive) and B represent the obstacle detector signal (1 for obstacle detected, 0 for no obstacle). Determine the desired output: The garage door should open (output 1) only when the remote control signal is active (A=1) and there is no obstacle detected (B=0).

Design the circuit: To create the circuit, use an AND gate and a NOT gate. First, connect the B input (obstacle detector) to the input of the NOT gate. Then, connect the output of the NOT gate and the A input (remote control signal) to the inputs of the AND gate. The output of the AND gate will represent the garage door opener signal.



Inputs		Output
A	B	F
0	0	0
0	1	0
1	0	1
1	0	0

Figure 4: Logic circuit and truth table for a garage door opener.

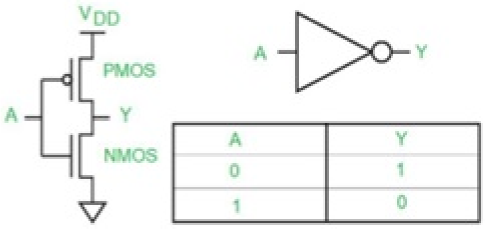
2.4 ASIDE: How do logic gates work on an electrical level?

In digital logic gates, the current flow is based on the underlying technology used to implement the gates. The most common technology for implementing digital logic gates is based on semiconductor devices, such as transistors. The two primary types of transistors are Bipolar Junction Transistors (BJTs) and Metal-Oxide-Semiconductor Field-Effect Transistors (MOSFETs).

For the purpose of understanding, let's consider CMOS (Complementary Metal-Oxide-Semiconductor) technology, which is widely used in modern digital circuits due to its low power consumption and high noise immunity. In CMOS technology, logic gates are built using pairs of complementary MOSFETs - PMOS (p-type MOSFETs) and NMOS (n-type MOSFETs).

For each type of gate (AND, OR, NOT), the arrangement and connections of transistors determine the current flow and the output of the gate based on the input signals.

For example, a NOT gate in CMOS consists of a PMOS transistor connected between the output and the positive voltage supply (Vdd), and an NMOS transistor connected between the output and the ground (GND). Both transistors have their gate terminals connected to the input. When the input is high (logic 1), the NMOS transistor conducts, and the PMOS is off, causing the output to be connected to GND (logic 0). Conversely, when the input is low (logic 0), the PMOS conducts, and the NMOS is off, connecting the output to Vdd (logic 1).



A	Y
0	1
1	0

Figure 5: CMOS NOT gate circuit.

CMOS logic gates can operate at various supply voltages, depending on the specific fabrication process. Early CMOS circuits used supply voltages of 5 volts. However, modern CMOS circuits have continually reduced their supply voltages to minimize power consumption. Nowadays, CMOS circuits can operate at supply voltages as low as 1.0 to 1.8 volts (or even lower in some cases) for advanced fabrication processes. The specific voltage levels for logic 0 (low) and logic 1 (high) in CMOS depend on the supply voltage, but generally, anything close to GND (0 volts) is considered logic 0, and anything close to the supply voltage (Vdd) is considered logic 1.

3 Bit versus Qubit (10 minutes)

3.1 Classical Computing

In classical computing, a bit is the basic unit of information and can only take on two values: 0 or 1.

You start from the rightmost digit to convert a binary number to its decimal equivalent. The value of each digit is determined by its position in the binary number, where each position has a value of 2 raised to the power of its position from the rightmost digit. Hence, the decimal equivalent of a binary number

$$b_{n-1}b_{n-2}\cdots b_1b_0$$

can be calculated using the following equation:

$$D = \sum_{i=0}^{n-1} b_i \cdot 2^i$$

In this equation, D is the decimal equivalent, b_i is the binary digit at position i, and n is the total number of binary digits.

Exercise: What is the binary number 1011 in the decimal number?

$$1 \times 2^0 + 1 \times 2^1 + 0 \times 2^2 + 1 \times 2^3 = 1 + 2 + 0 + 8 = 11$$

3.2 Quantum Computing

Qubits (pronounced “cue-bit”) or quantum bits are basic building blocks that encompass all fundamental quantum phenomena. They provide a mathematically simple framework in which to introduce the basic concepts of quantum physics. Qubits are 2-state quantum systems. For example, Consider a system with two distinguishable (classical) states, the electron in the Hydrogen atom can be in the ground state or the first excited state, or any superposition of the two. Hence, a qubit can take on both $|0\rangle$ and $|1\rangle$ simultaneously, a property known as superposition. This means that a qubit can represent multiple values at the same time, in contrast to classical bits that can only be in one 0 or 1 at a time.

The states $|0\rangle$ and $|1\rangle$ are commonly written in Dirac or bra-ket notation in quantum computing, which is a standard notation for representing quantum states and operators.

Example: Imagine that you have two coins, one red and one blue. In classical computing, you can represent the state of each coin using one bit, where 0 represents heads and 1 represents tails. So, for example, the state 01 would represent the red coin being heads and the blue coin being tails.

In quantum computing, you can represent the state of each coin using one qubit, where the state $|0\rangle$ represents heads and the state $|1\rangle$ represents tails. However, unlike classical bits, qubits can be in a superposition of both states simultaneously. So, for example, qubit one is a superpostion state $\frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$ and qubit two also in a superposition state of $\frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$.

Back to our coins, this means that the red coin is simultaneously in the state of heads and tails, and the blue coin is also in the state of heads and tails at the same time.

Furthermore, the two qubits can also be entangled, meaning that the state of one qubit can affect the state of the other (even if they are far apart). For example, if you measure the state of the red coin and obtain the result heads, then the state of the blue coin will also be heads with probability 1. But more on this later.

3.3 Types of Qubits and their Implementations

- For photons, the basis state $|0\rangle$ corresponds to a horizontally polarized photon, while the basis state $|1\rangle$ corresponds to a vertically polarized photon. Photonic qubits are typically manipulated and gated using various optical components and techniques.
- For trapped ions, for example, in a two-level system, the ground state $|0\rangle$ can represent the ion's lowest energy level (e.g., the S1/2 state for a calcium ion), while the excited state $|1\rangle$ can represent the ion's highest energy level (e.g., the D5/2 state for a calcium ion). Note that the specific energy levels and transitions used to implement qubits in trapped ion systems can vary depending on the choice of ion species, trap architecture, and experimental setup. Laser light is typically used to manipulate the internal energy levels of the ions and to induce transitions between different quantum states.
- For superconducting qubits, typically, the basis states are associated with the two lowest energy levels of a superconducting circuit, which correspond to the ground state $|0\rangle$ and the excited state $|1\rangle$. These states are typically represented by the two lowest energy levels of a superconducting circuit, which can be controlled using microwave pulses or other techniques.

3.4 ASIDE: Advantages and limitations of qubits compared to classical bits

Advantages:

1. **Superposition**: Unlike classical bits that can only have one of two states (0 or 1), qubits can exist in a superposition of both states simultaneously. This enables quantum computers to perform certain calculations exponentially faster than classical computers.
2. **Entanglement**: Qubits can be entangled, meaning that the state of one qubit can affect the state of another qubit, even if they are far apart. This enables quantum computers to perform certain tasks that are impossible with classical computers.

3. **Massive parallelism**: Qubits can represent many possible states at once, which enables quantum computers to perform multiple calculations simultaneously.

Limitations:

1. **Fragility**: Qubits are fragile and can be easily disrupted by external factors such as heat and electromagnetic radiation. This can cause errors in calculations and reduce the accuracy of quantum computers.

2. **Limited coherence time**: Qubits have a limited coherence time, meaning that they can maintain their superposition state for only a short period before collapsing to either 0 or 1. This can make it challenging to perform certain calculations that require a long coherence time.

3. **Difficult to control and read**: Qubits are challenging to control and read, requiring specialized equipment and techniques. This makes it difficult to scale up quantum computers and perform complex computations.

4 Main Postulates and Principles of Quantum Mechanics (10 minutes)

Postulates and principles form the basis of quantum mechanics and are essential for understanding the behavior of quantum systems. There are many consequences of the postulates and principles of quantum mechanics. Quantum computing is one of them.

”Postulates” usually refer to the foundational assumptions that form the basis of quantum mechanics and which cannot be derived from other principles or laws of physics.

1. State postulate: The state of a quantum system is described by a state vector, which contains all the information about the system that can be known.

2. Measurement postulate: When a quantum system is measured, the state vector ”collapses” to one of the possible eigenstates of the measurement operator, and the outcome of the measurement is one of the eigenvalues of the operator.

3. Superposition postulate: A quantum system can exist in a superposition of states, where it simultaneously has some probability of being in each state.

4. Evolution postulate: The state vector of a quantum system evolves over time according to the Schrödinger equation, which describes how the system’s state changes in response to its interactions with the environment.

On the other hand, ”principles” typically refer to the general concepts and overarching ideas that guide the interpretation and application of quantum mechanics. These principles are often derived from the postulates of quantum mechanics, but they may also be informed by experimental observations and theoretical considerations. Some of the key principles of quantum mechanics include:

1. Wave-particle duality: Particles can exhibit both wave-like and particle-like behavior, and their behavior depends on the experiment's nature.

2. Uncertainty principle: It is impossible to precisely measure certain pairs of properties of a quantum system (such as position and momentum) simultaneously. The more precisely one property is measured, the more uncertain the other property becomes.

3. Entanglement principle: States of quantum systems can become correlated in a way that classical systems cannot, even if large distances separate them.

There are many consequences of the postulates and principles of quantum mechanics, some of which include:

1. Quantization: Certain physical quantities, such as energy, can only take on discrete values in a quantum system, rather than continuous values as in classical mechanics.

2. Tunneling: Quantum particles can pass through energy barriers that would be impossible to cross in classical mechanics, due to the wave-like nature of particles.

3. Interference: Waves associated with quantum particles can interfere with each other constructively or destructively, leading to phenomena such as diffraction or interference patterns.

4. Scattering: When a quantum particle interacts with another system, it can scatter in a probabilistic manner, leading to a wide range of phenomena such as the Compton effect or Raman scattering.

5. Bell's theorem: Bell's theorem proves that the predictions of quantum mechanics are incompatible with local realism, and that there are certain correlations between entangled particles that cannot be explained by classical physics.

6. Quantum computing: Quantum mechanics has provided a theoretical basis for the development of quantum computers, which use the principles of superposition and entanglement to perform certain calculations more efficiently than classical computers.

4.1 The superposition principle

Consider a system with k distinguishable (classical) states. For example, the electron in a hydrogen atom is only allowed to be in one of a discrete set of energy levels, starting with the ground state, the first excited state, the second excited state, and so on. If we assume a suitable upper bound on the total energy, then the electron is restricted to being in one of k different energy levels – the ground state or one of $k-1$ excited states. As a classical system, we might use the state of this system to store a number between 0 and $k-1$.

The superposition principle says that if a quantum system can be in one of two states then it can also be placed in a linear superposition of these states with complex coefficients.

Let us introduce some notation. We denote the ground state of our qubit by $|0\rangle$, and the successive excited states by $|1\rangle, \ldots, |k-1\rangle$. These are the k possible classical states of the electron. The superposition principle tells us that, in general, the (quantum) state of the electron is $\alpha_0|0\rangle + \alpha_1|1\rangle + \cdots + \alpha_{k-1}|k-1\rangle$, where $\alpha_0, \alpha_1, \ldots, \alpha_{k-1}$ are complex numbers normalized so that $\sum_j |\alpha_j|^2 = 1$. α_j is called the amplitude of the state $|j\rangle$.

For instance, if $k = 3$, the state of the electron could be:

- $1/\sqrt{2}|0\rangle + 1/2|1\rangle + 1/2|2\rangle$
- $1/\sqrt{2}|0\rangle - 1/2|1\rangle + i/2|2\rangle$
- $(1+i)/3|0\rangle - (1-i)/3|1\rangle + (1+2i)/3|2\rangle$, where $i = \sqrt{-1}$.

The superposition principle is one of the most mysterious aspects about quantum physics – it flies in the face of our intuitions about the physical world. One way to think about a superposition is that the electron does not make up its mind about whether it is in the ground state or each of the $k-1$ excited states, and the amplitude α_0 is a measure of its inclination towards the ground state. Of course we cannot think of α_0 as the probability that an electron is in the ground state – remember that α_0 can be negative or imaginary. The measurement principle, which we will see shortly, will make this interpretation of α_0 more precise.

4.2 The Geometry of Hilbert Space

The quantum state of a k-state system can be described by a k-dimensional complex vector with complex amplitudes $\alpha_0, \ldots, \alpha_{k-1}$, normalized such that $\sum_{j=0}^{k-1} |\alpha_j|^2 = 1$. We can express this state as a k-dimensional column vector:

$$\begin{pmatrix} \alpha_0 \\ \alpha_1 \\ \vdots \\ \alpha_{k-1} \end{pmatrix}$$

The normalization condition on the complex amplitudes means that the state of the system is a unit vector in a k-dimensional complex vector space, known as a Hilbert space.

However, we can also express the quantum state using Dirac's ket notation:

$$\alpha_0|0\rangle + \alpha_1|1\rangle + \cdots + \alpha_{k-1}|k-1\rangle$$

where $|0\rangle, |1\rangle, \ldots, |k-1\rangle$ are mutually orthogonal unit vectors in a k-dimensional complex vector space, and they form an orthonormal basis for that space (called the standard basis). Thus, unit vectors can represent any two states in this space and the inner product of two states $|\psi\rangle$ and $|\phi\rangle$ is given by

$$\langle\psi|\phi\rangle = \sum_{j=0}^{k-1} \alpha_j^* \beta_j$$

,

where α_j^* is the complex conjugate of α_j. The absolute value of the inner product is the cosine of the angle between these two vectors. The advantage of using ket notation is that it explicitly labels the basis vectors and expresses the physical quantity of interest.

4.3 The Measurement Principle

This linear superposition $\psi = \sum_{j=0}^{k-1} \alpha_j|j\rangle$ is part of the private world of the electron. Access to the information describing this state is severely limited—in particular, we cannot actually measure the complex amplitudes α_j. This is not just a practical limitation; it is enshrined in the measurement postulate of quantum physics.

A measurement on this k-state system yields one of at most k possible outcomes, i.e., an integer between 0 and k-1. Measuring ψ in the standard basis yields j with probability $|\alpha_j|^2$.

One important aspect of the measurement process is that it alters the state of the quantum system: the effect of the measurement is that the new state is exactly the outcome of the measurement. I.e., if the outcome of the measurement is j, then following the measurement, the qubit is in state $|j\rangle$. This implies that you cannot collect any additional information about the amplitudes α_j by repeating the measurement.

Intuitively, a measurement provides the only way of reaching into the Hilbert space to probe the quantum state vector. In general, this is done by selecting an orthonormal basis $|e_0\rangle, \ldots, |e_{k-1}\rangle$. The outcome of the measurement is $|e_j\rangle$ with probability equal to the square of the length of the projection of the state vector ψ on $|e_j\rangle$. A consequence of performing the measurement is that the new state vector is $|e_j\rangle$. Thus, measurement may be regarded as a probabilistic rule for projecting the state vector onto one of the vectors of the orthonormal measurement basis.

Some of you might be puzzled about how a measurement is carried out physically? We will get to that soon when we give more explicit examples of quantum systems.

α_j, $|j\rangle$, $|\cdot\rangle$, and $|e_0\rangle, \ldots, |e_{k-1}\rangle$ are written in the Dirac notation, which is a standard notation in quantum mechanics.

4.4 Bra-ket notation

The notation $\langle v|$ (pronounced "bra v") denotes the row vector $|v\rangle^\dagger$, the conjugate-transpose of $|v\rangle$. For example, $\langle 0| = \begin{pmatrix} 1 & 0 \end{pmatrix}$ and $\langle 1| = \begin{pmatrix} 0 & 1 \end{pmatrix}$. More generally, if $|\psi\rangle = \alpha_0|0\rangle + \beta_1|1\rangle$, then $\langle\psi| = (\alpha_0^*, \beta_1^*)$.

In bra-ket notation, the inner product of two vectors $|\psi\rangle$ and $|\phi\rangle$ is denoted by $\langle\psi|\phi\rangle$, and is defined as the product of the conjugate transpose of the first vector with the second vector: $\langle\psi|\phi\rangle = |\langle\phi|\psi\rangle|^2$.

The outer product of two vectors $|\psi\rangle$ and $|\phi\rangle$ is denoted by $|\psi\rangle\langle\phi|$ and is defined as the product of the first vector with the conjugate transpose of the second vector: $|\psi\rangle\langle\phi| = \begin{pmatrix} \alpha_0 \\ \beta_1 \end{pmatrix} \begin{pmatrix} \alpha_0^* & \beta_1^* \end{pmatrix} = \begin{pmatrix} \alpha_0\alpha_0^* & \alpha_0\beta_1^* \\ \beta_1\alpha_0^* & \beta_1\beta_1^* \end{pmatrix}$.

The use of bra-ket notation greatly simplifies many quantum mechanical calculations, as it allows us to easily manipulate complex vector and matrix expressions using simple algebraic rules.

4.5 Unitary Operators

A postulate of quantum physics states that the evolution of a quantum system is necessarily unitary. Intuitively, a unitary transformation is a rigid body rotation (or reflection) of the Hilbert space, thus resulting in a transformation of the state vector that doesn't change its length.

Let us consider what this means for the evolution of a qubit. A unitary transformation on the Hilbert space C^2 is specified by mapping the basis states $|0\rangle$ and $|1\rangle$ to orthonormal states $|v_0\rangle = a|0\rangle + b|1\rangle$ and $|v_1\rangle = c|0\rangle + d|1\rangle$. It is specified by the linear transformation on C^2:

$$U = \begin{pmatrix} a & c \\ b & d \end{pmatrix}$$

A unitary operator U has the property that its conjugate transpose $U^\dagger$ is equal to its inverse U^{-1}:

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

where I is the identity matrix, we also say that U is a Hermitian matrix.

4.5.1 Example: Pauli Matrices

Pauli matrices are a set of three 2×2 unitary and Hermitian matrices, which are widely used in quantum mechanics. They are defined as follows:

$$\sigma_x = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, \quad \sigma_y = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix}, \quad \sigma_z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$$

For example, let's consider the unitary matrix σ_x. We can check its unitarity by computing its conjugate transpose and verifying that the product of the matrix and its conjugate transpose is equal to the identity matrix:

$$\sigma_x^\dagger = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, \quad \sigma_x \sigma_x^\dagger = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} = I$$

Therefore, σ_x is a unitary operator.

5 Bloch Sphere, Superposition, Complex Numbers, Measurement (15 minutes)

5.1 Bloch Sphere, Superposition, Complex Numbers

The Bloch sphere is named after the Swiss physicist Felix Bloch, who introduced the concept in his seminal paper "On the theory of the interaction of nuclear spins in a crystal" in 1946. Bloch was awarded the Nobel Prize in Physics in 1952 for his work on nuclear magnetic resonance, which laid the foundation for the development of magnetic resonance imaging (MRI). The Bloch sphere is now a standard tool for visualizing quantum states in quantum information science and has applications in fields such as quantum computing, quantum cryptography, and quantum metrology.

We are now interested in the Bloch sphere coordinates (θ, ϕ) of the qubit state $|\psi\rangle$. These coordinates correspond to the polar and azimuthal angles, respectively, of the point on the surface of the sphere that represents the state. On the Bloch sphere, the $|0\rangle$ state corresponds to the north pole (i.e., the point with spherical coordinates $\theta = 0$ and $\phi = 0$) and the $|1\rangle$ state corresponds to the south pole (i.e., the point with spherical coordinates $\theta = \pi$ and $\phi = 0$).

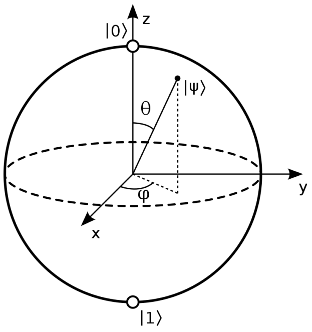


Figure 6: Bloch sphere with angles θ and ϕ.

The state vector $|+\rangle$, which is a superposition of $|0\rangle$ and $|1\rangle$ with equal amplitudes, corresponds to the equator of the Bloch sphere (i.e., the points with spherical coordinate $\theta = \pi/2$ and $\phi = 0$). Similarly, the state vector $|-\rangle$, which is a superposition of $|0\rangle$ and $|1\rangle$ with opposite phases, also corresponds to the equator of the Bloch sphere but with a different orientation.

Note that while the north and south poles of the Bloch sphere correspond to the $|0\rangle$ and $|1\rangle$ basis states, respectively, any point on the surface of the sphere can represent a valid quantum state.

5.1.1 Superposition and Probability Amplitudes

On the Bloch sphere, a superposition state is a state in which a qubit is in a linear combination of the $|0\rangle$ and $|1\rangle$ basis states. Mathematically, a superposition state can be written in Dirac notation as:

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

where α and β are complex probability amplitudes. $|0\rangle = \begin{pmatrix} 1 \\ 0 \end{pmatrix}$ and $|1\rangle = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$ are column vectors. To calculate the probability amplitudes, we can take the inner product of the state vector with the corresponding basis vector. For example, to calculate the probability amplitude α, we can take the inner product of the state vector $|\psi\rangle$ with the basis vector $|0\rangle$:

$$\alpha = \langle 0|\psi\rangle$$

Similarly, to calculate the probability amplitude β, we can take the inner product of the state vector $|\psi\rangle$ with the basis vector $|1\rangle$:

$$\beta = \langle 1|\psi\rangle$$

The coefficient α in front of $|0\rangle$ represents the probability amplitude of the qubit being in the state $|0\rangle$, and the coefficient β in front of $|1\rangle$ represents the probability amplitude of the qubit being in the state $|1\rangle$.

The probability of measuring the qubit in the state $|0\rangle$ is given by the squared magnitude of the probability amplitude α:

$$P(|0\rangle) = |\alpha|^2$$

Similarly, the probability of measuring the qubit in the state $|1\rangle$ is given by the squared magnitude of the probability amplitude β:

$$P(|1\rangle) = |\beta|^2$$

Note that the probability amplitudes α and β are complex numbers that satisfy the normalization condition:

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

This condition ensures that the total probability of the qubit being in one of the basis states is 1, as required by the laws of quantum mechanics.

Complex numbers have a real part and an imaginary part, and can be written in the form $z = x + iy$, where x and y are real numbers and i is the imaginary unit, defined as $i^2 = -1$. The magnitude of a complex number z is given by $|z| = \sqrt{x^2 + y^2}$, and the phase of z is given by $\arg(z) = \text{atan2}(y, x)$, where atan2 is the two-argument arctangent function that takes into account the signs of x and y.

In Euler's notation, a complex number $z = x + iy$ can be written as:

$$z = |z|e^{i\arg(z)}$$

where $|z|$ is the magnitude of the complex number and $\arg(z)$ is its phase angle.

Hence, we can write the complex numbers α and β in terms of their magnitudes and phases as well:

$$\alpha = |\alpha|e^{i\theta_1} \qquad \text{and} \qquad \beta = |\beta|e^{i\theta_2}$$

where θ_1 and θ_2 are the phases of α and β, respectively. The phase of each probability amplitude is encoded in the azimuthal angle φ of the corresponding point on the Bloch sphere.

Substituting these expressions into the state vector equation and simplifying, we get:

$$|\psi\rangle = |\alpha|e^{i\theta_1}|0\rangle + |\beta|e^{i\theta_2}|1\rangle = e^{i\theta_1}(|\alpha||0\rangle + |\beta|e^{i(\theta_2-\theta_1)}|1\rangle)$$

Note that the overall phase factor $e^{i\theta_1}$ does not affect the probabilities of measuring the qubit in the $|0\rangle$ or $|1\rangle$ states, so we can choose $\theta_1 = 0$ without loss of generality. This gives:

$$|\psi\rangle = |\alpha||0\rangle + e^{i\theta_2}|\beta||1\rangle$$

Now, let's represent the complex numbers $|\alpha|$ and $|\beta|$ in terms of their polar coordinates on the Bloch sphere, with θ and φ denoting the polar and azimuthal angles, respectively. Specifically:

$$|\alpha| = \cos\left(\frac{\theta}{2}\right) \qquad \text{and} \qquad |\beta| = \sin\left(\frac{\theta}{2}\right)$$

Substituting these expressions into the state vector equation and simplifying, we get:

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

This is the desired expression for a qubit state vector on the Bloch sphere.

In the Bloch sphere representation, the state of a qubit is given by a point on the surface of a unit sphere. The position of the point is characterized by two angles, θ and φ, which are related to the probability amplitudes α and β associated with the $|0\rangle$ and $|1\rangle$ basis states, respectively, as:

$$\alpha = \cos\left(\frac{\theta}{2}\right) \qquad \text{and} \qquad \beta = e^{i\varphi}\sin\left(\frac{\theta}{2}\right)$$

Here, α is a real number and β is generally a complex number. So, in the Bloch sphere representation, α is not a complex number, but β can be.

This representation of the qubit state vector in terms of complex exponentials provides a powerful tool for analyzing and manipulating quantum states using mathematical methods such as linear algebra and calculus.

But before we talk about how to manipulate quantum states, we need to discuss how measurement is represented on the sphere.

5.1.2 Two Qubits

Now let us examine a system of two qubits. Consider the two electrons in two hydrogen atoms, each regarded as a 2-state quantum system. Since each electron can be in either of the ground or excited state, classically the two electrons are in one of four states – 00, 01, 10, or 11 – and represent 2 bits of classical information. By the superposition principle, the quantum state of the two electrons can be any linear combination of these four classical states:

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

where $\alpha_{ij} \in C$, $\sum_{ij} |\alpha_{ij}|^2 = 1$. Again, this is just Dirac notation for the unit vector in C^4:

$$\begin{pmatrix} \alpha_{00} \\ \alpha_{01} \\ \alpha_{10} \\ \alpha_{11} \end{pmatrix}$$

5.1.3 Multi-Qubits

For a system consisting of n qubits, there are 2^n possible classical states, represented by all binary strings of length n. Just as in the case of two qubits, the superposition principle allows the quantum state of the n-qubit system to be any linear combination of these classical states:

$$|\psi\rangle = \sum_{i_1,i_2,\ldots,i_n=0}^{1} \alpha_{i_1 i_2 \ldots i_n} |i_1 i_2 \ldots i_n\rangle,$$

where $\alpha_{i_1 i_2 \ldots i_n} \in C$ and $\sum_{i_1,i_2,\ldots,i_n} |\alpha_{i_1 i_2 \ldots i_n}|^2 = 1$. This state vector is an element of the Hilbert space C^{2^n} and can be represented as:

$$\begin{pmatrix} \alpha_{00\ldots0} \\ \alpha_{00\ldots1} \\ \vdots \\ \alpha_{11\ldots1} \end{pmatrix}$$

In the case of multi-qubit systems, unitary transformations acting on the entire system must also be elements of the corresponding Hilbert space. For example, if we have a 3-qubit system, the unitary transformation acting on the system must be an element of $C^{8\times 8}$:

$$U = \begin{pmatrix} a_{00} & a_{01} & \cdots & a_{07} \\ a_{10} & a_{11} & \cdots & a_{17} \\ \vdots & \vdots & \ddots & \vdots \\ a_{70} & a_{71} & \cdots & a_{77} \end{pmatrix}$$

These unitary transformations can be used to manipulate and process the information stored in the quantum system, forming the basis of quantum computation and information processing.

5.2 Measurement

First, recall that the projection onto the z-axis represents the probability of measuring the state in the $|0\rangle$ or $|1\rangle$ basis. When a measurement is made, the state does collapse onto one of these basis states, but this is not directly related to the projection on the z-axis.

However, the collapse of a quantum state can be visualized on the Bloch sphere. Before measurement, the quantum state is represented by a point on the surface of the sphere, which can be any point corresponding to a superposition of the $|0\rangle$ and $|1\rangle$ states. When the state collapses as a result of a measurement, the point "jumps" to one of the poles of the sphere, representing the state having collapsed into a pure state. The specific pole to which the state collapses depends on the measurement outcome and the initial state of the system.

Using the Dirac notation, the collapse of a quantum state as a result of measurement can be written as:

$$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle \xrightarrow{\text{measurement}} |0\rangle \text{ or } |1\rangle,$$

where α and β are probability amplitudes, and the measurement outcome determines whether the state collapses to $|0\rangle$ or $|1\rangle$. Note that after measurement, the state is no longer in a superposition, but is in a definite state.

The collapse of a quantum state due to a measurement can be represented in Dirac notation using the projection operators. Let's say we have a quantum state $|\psi\rangle$ that we want to measure in the $|0\rangle$ and $|1\rangle$ basis. We can represent these basis states using the projection operators:

$$|0\rangle\langle 0| = \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix} \qquad |1\rangle\langle 1| = \begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix}$$

To measure the quantum state $|\psi\rangle$ in the $|0\rangle$ basis, we take the inner product of $|\psi\rangle$ with the projection operator $|0\rangle\langle 0|$:

$$P(|0\rangle) = |\langle 0|\psi\rangle|^2$$

The probability of measuring the state in the $|1\rangle$ basis can be similarly calculated:

$$P(|1\rangle) = |\langle 1|\psi\rangle|^2$$

If the measurement outcome is $|0\rangle$, then the quantum state collapses to the projection of $|\psi\rangle$ onto the $|0\rangle$ basis:

$$|\psi\rangle = \frac{|0\rangle\langle 0|\psi\rangle}{||0\rangle\langle 0|\psi\rangle|}$$

Similarly, if the measurement outcome is $|1\rangle$, then the quantum state collapses to the projection of $|\psi\rangle$ onto the $|1\rangle$ basis:

$$|\psi\rangle = \frac{|1\rangle\langle 1|\psi\rangle}{||1\rangle\langle 1|\psi\rangle|}$$

Here, $||$ denotes the normalization factor to ensure that the resulting state is a unit vector.

6 The Quantum Circuit Model (10 minutes)

Now we are ready to discuss how to manipulate quantum states. In this section, we will explore quantum circuits as a means to represent and manipulate qubits using quantum gates. Quantum gates are the fundamental building blocks of quantum circuits and can be represented as matrices that transform qubit states.

6.1 Linear Algebra

Quantum states can be manipulated using unitary operators, which are represented by matrices. These matrices must satisfy certain properties, such as being Hermitian (equal to their own conjugate transpose) and having a determinant of 1. Unitary operators can be used to transform a qubit or a quantum register from one state to another, or to create entanglement between qubits.

Tensor products can also be used to manipulate quantum states. When two quantum systems are combined, their states are represented by the tensor product of their individual states. This allows for the creation of more complex quantum states, such as entangled states.

Overall, linear algebra provides a powerful tool for describing and manipulating quantum states in quantum computing.

6.1.1 Matrix multiplication:

Let's consider the Hadamard gate, which is a commonly used gate in quantum computing that maps the state $|0\rangle$ to the superposition state $(|0\rangle + |1\rangle)/\sqrt{2}$, and maps the state $|1\rangle$ to the superposition state $(|0\rangle - |1\rangle)/\sqrt{2}$. The Hadamard gate can be represented by the following 2×2 matrix:

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

If we apply this gate to the state $|0\rangle$, which is represented by the column vector $\begin{pmatrix} 1 \\ 0 \end{pmatrix}$, we get:

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

which is the superposition state $(|0\rangle + |1\rangle)/\sqrt{2}$.

6.1.2 Tensor product:

Let's consider two qubits, represented by the states $|0\rangle$ and $|1\rangle$. We can combine these two qubits into a single system using the tensor product, which results in a 4×1 column vector representing the joint state of both qubits:

$$|0\rangle \otimes |1\rangle = \begin{pmatrix} 1 \\ 0 \end{pmatrix} \otimes \begin{pmatrix} 0 \\ 1 \end{pmatrix} = \begin{pmatrix} 1 \cdot \begin{pmatrix} 0 \\ 1 \end{pmatrix} & 0 \cdot \begin{pmatrix} 0 \\ 1 \end{pmatrix} \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \\ 0 \\ 0 \end{pmatrix}.$$

So the joint state of the two qubits is $(|01\rangle)$, which represents the qubit where the first qubit is in state $|0\rangle$ and the second qubit is in state $|1\rangle$.

A single quantum bit is a unit vector in the Hilbert space C^2. Now suppose we have two quantum bits. How do we write them together? We need a new Hilbert space which captures the interaction of the two bits. If V, W are vector spaces with bases $\{v_1, \ldots, v_n\}$, $\{w_1, \ldots, w_m\}$, the tensor product $V \otimes W$ of V and W is an nm-dimensional vector space which is spanned by elements of the form $v \otimes w$ - called elementary tensors. These elementary tensors behave bilinearly, that is, we have the relations:

$$\begin{aligned} \alpha(v \otimes w) &= \alpha v \otimes w = v \otimes \alpha w \\ u \otimes v + w \otimes v &= (u+w) \otimes v \\ u \otimes v + u \otimes w &= u \otimes (v+w). \end{aligned}$$

A basis for the tensor product space consists of the vectors $\{v_i \otimes w_j : 1 \leq i \leq n, 1 \leq j \leq m\}$, and thus a general element of $V \otimes W$ is of the form:

$$\sum_{i,j} \alpha_{ij} v_i \otimes w_j$$

This definition extends analogously to tensor products with more than two terms. The tensor product space is also a Hilbert space with the inherited inner product:

$$\langle v \otimes w, v' \otimes w' \rangle = \langle v, v' \rangle \langle w, w' \rangle$$

As it turns out, a two-bit system is conveniently represented by a unit vector in the Hilbert space $C^2 \otimes C^2$. $C^2 \otimes C^2$ is necessarily isomorphic to C^4 since there is only one complex four-dimensional Hilbert space, but as we will see, in the world of quantum mechanics it is convenient to be able to ”construct” the larger space from the smaller ones.

Using Dirac ”ket” notation, we write the basis of $C^2 \otimes C^2$ as:

$$\{|0\rangle \otimes |0\rangle, |0\rangle \otimes |1\rangle, |1\rangle \otimes |0\rangle, |1\rangle \otimes |1\rangle\}$$

We will often write $|0\rangle \otimes |0\rangle$ as $|0\rangle|0\rangle$ or $|00\rangle$.

In general, we represent an n-particle system by n copies of C^2 tensored together. We will often write $(C^2)^{\otimes n} = C^{2^n}$. So the state of an n-qubit system can be written as

6.1.3 Entangled states:

Let’s consider the Bell state, which is an entangled state of two qubits given by:

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

This state cannot be written as a tensor product of individual states, and it has interesting properties that are used in quantum communication and computation. For example, if we measure one qubit in the Bell state, the state of the other qubit collapses to a definite state that is correlated with the measurement outcome, regardless of how far apart the two qubits are. This property is known as quantum entanglement, and it is a key resource for certain types of quantum algorithms and protocols.

It turns out that we use the CNOT gate, short for Controlled-NOT gate, to create entangled states. It operates on two qubits, a control qubit, and a target qubit.

The CNOT gate is represented by the following 4 × 4 unitary matrix:

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

Suppose we have two qubits, qubit 1 and qubit 2. If qubit 1 is in the state $|0\rangle$, then the CNOT gate acts as the identity gate on qubit 2. If qubit 1 is in the state $|1\rangle$, then the CNOT gate applies a NOT gate to qubit 2.

We can write this transformation mathematically using Dirac notation as:

$$\text{CNOT}(|x\rangle \otimes |y\rangle) = |x\rangle \otimes (|x\rangle \oplus |y\rangle),$$

where $\oplus$ denotes addition modulo 2. In other words, if the control qubit (qubit 1) is in state $|1\rangle$, then the target qubit (qubit 2) is flipped. Otherwise, the target qubit.

For more single-qubit and two-qubit gates, see the section 7.

6.1.4 ASIDE: Show that the Bell state is entangled

Let's consider one of the Bell states, specifically the Φ^+ Bell state:

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

To show that this state is entangled, we need to prove that it cannot be represented as a tensor product of two separate qubit states. In other words, we need to show that there are no single qubit states $|\psi\rangle$ and $|\phi\rangle$ such that:

$$|\Phi^+\rangle = |\psi\rangle \otimes |\phi\rangle$$

Let's assume that the Bell state $|\Phi^+\rangle$ can be represented as a tensor product of two separate qubit states:

$$|\Phi^+\rangle = (a|0\rangle + b|1\rangle) \otimes (c|0\rangle + d|1\rangle)$$

Expanding this expression, we get:

$$|\Phi^+\rangle = ac|00\rangle + ad|01\rangle + bc|10\rangle + bd|11\rangle$$

Now, let's compare the coefficients of the expanded expression with those of the Bell state:

$$ac = \frac{1}{\sqrt{2}}, \quad ad = 0, \quad bc = 0, \quad bd = \frac{1}{\sqrt{2}}$$

From the equations $ad = 0$ and $bc = 0$, we can infer that either $a = 0$ or $d = 0$, and either $b = 0$ or $c = 0$. However, no combination of these conditions can satisfy both $ac = \frac{1}{\sqrt{2}}$ and $bd = \frac{1}{\sqrt{2}}$ simultaneously. Therefore, the Bell state $|\Phi^+\rangle$ cannot be represented as a tensor product of two separate qubit states, and it is an entangled state.

6.2 Quantum Circuit

A quantum circuit is a sequence of quantum gates and measurements that can be used to manipulate and process quantum information. The process of creating a quantum circuit involves initializing qubits, applying quantum gates to manipulate their quantum state, and measuring the results.

In a quantum circuit, the qubits are represented by wires, and the gates are applied to these wires to manipulate the qubits. After the gates are applied, the qubits are measured, which collapses their superposition into a definite classical state. The measurement outcomes are then recorded in classical registers, which are used to store the results of the computation.

Quantum circuits can be expressed in a variety of ways, including circuit diagrams, code, and text. After lunch, we will use TKET with its Python wrapper called PyTKET to build and run quantum circuits. TKET is a software development kit (SDK) for quantum computing developed by Quantinuum, and it will be used throughout the week and during the hackathon for building and optimizing quantum circuits.

6.3 The Bell State Circuit

Design a quantum circuit that generates the Bell state

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

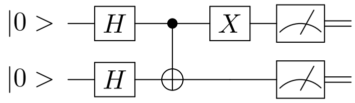


Figure 7: Sample quantum circuit with two qubits initialized in the state |0 >, Hadamard gates to create a superposition, a CNOT gate to entangle the qubits, an X gate to flip the state of one qubit, and measurements.

using two qubits. The circuit should include a single entangling gate and as few other gates as possible.

Solution: The initial state of the two qubits is $|00\rangle$, which we can express the initial state $|00\rangle$ in tensor product notation as:

$$|00\rangle = |0\rangle \otimes |0\rangle = \begin{pmatrix}1\\0\end{pmatrix} \otimes \begin{pmatrix}1\\0\end{pmatrix} = \begin{pmatrix}1\\0\\0\\0\end{pmatrix}$$

Then, we can apply the Hadamard gate H to the first qubit by taking the tensor product of H with the identity matrix I on the second qubit:

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

We can then multiply $H \otimes I$ with $|00\rangle$ using matrix multiplication:

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

$$CNOT(H \otimes I)|00\rangle = \frac{1}{\sqrt{2}}\begin{bmatrix}1 & 0 & 0 & 0\\0 & 1 & 0 & 0\\0 & 0 & 0 & 1\\0 & 0 & 1 & 0\end{bmatrix}\begin{pmatrix}1\\0\\1\\0\end{pmatrix} = \frac{1}{\sqrt{2}}\begin{pmatrix}1\\0\\0\\1\end{pmatrix} = \frac{1}{\sqrt{2}}(|00\rangle + |10\rangle)$$

Thus, the state of the two qubits after applying the Hadamard gate to the first qubit is $\frac{1}{\sqrt{2}}(|00\rangle + |10\rangle)$.

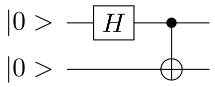


Figure 8: Bell state circuit.

6.3.1 ASIDE: Some examples of applications of the Bell state

– Quantum teleportation: transmitting an unknown quantum state from one location to another without physically sending the qubit that encodes the state.

– Quantum error correction: using the Bell state as a resource for implementing quantum error correction codes.

– Quantum secret sharing: distributing a secret among multiple parties in such a way that the secret can only be reconstructed if a majority of the parties collaborate.

– Quantum key distribution: establishing a shared secret key between two parties that can be used for secure communication.

– Quantum sensing: using the Bell state to enhance the sensitivity of measurements in quantum sensing applications, such as magnetometry.

7 Appendix

7.1 Common One-qubit Gates

Gate	Matrix Representation	Bloch Sphere Rotation
Pauli-X (X)	$\begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$	180° about x-axis
Pauli-Y (Y)	$\begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix}$	180° about y-axis
Pauli-Z (Z)	$\begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$	180° about z-axis
Hadamard (H)	$\frac{1}{\sqrt{2}}\begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}$	180° about x-axis, then 90° about y-axis
Phase (S)	$\begin{pmatrix} 1 & 0 \\ 0 & i \end{pmatrix}$	90° about z-axis
Phase Conjugate (Sdg)	$\begin{pmatrix} 1 & 0 \\ 0 & -i \end{pmatrix}$	−90° about z-axis
Phase (T)	$\begin{pmatrix} 1 & 0 \\ 0 & e^{i\pi/4} \end{pmatrix}$	45° about z-axis
Phase Conjugate (Tdg)	$\begin{pmatrix} 1 & 0 \\ 0 & e^{-i\pi/4} \end{pmatrix}$	−45° about z-axis
SX	$\frac{1}{\sqrt{2}}\begin{pmatrix} 1 & -i \\ -i & 1 \end{pmatrix}$	90° about x-axis
SXdg	$\frac{1}{\sqrt{2}}\begin{pmatrix} 1 & i \\ i & 1 \end{pmatrix}$	−90° about x-axis
V	$\frac{1}{\sqrt{2}}\begin{pmatrix} 1 & -i \\ 0 & 1 \end{pmatrix}$	45° about x-axis
V Conjugate (Vdg)	$\frac{1}{\sqrt{2}}\begin{pmatrix} 1 & i \\ 0 & 1 \end{pmatrix}$	−45° about x-axis

7.2 Common Two-qubit Gates

The following gates are 2-qubit gates and do not have corresponding Bloch sphere rotations for single qubits.

- **SWAP** - Exchange qubits
  Matrix representation: $\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$
- **Controlled-X or CNOT (CX)** - Flips the second qubit if the first qubit is $|1\rangle$

Matrix representation: $\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}$

- **Controlled-Y (CY)** - Applies a Y gate to the second qubit if the first qubit is $|1\rangle$

  Matrix representation: $\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & -i \\ 0 & 0 & i & 0 \end{pmatrix}$

- **Controlled-Z (CZ)** - Applies a Z gate to the second qubit if the first qubit is $|1\rangle$

  Matrix representation: $\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}$

- **Controlled-Hadamard (CH)** - Applies a Hadamard gate to the second qubit if the first qubit is $|1\rangle$

  Matrix representation: $\frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & -1 \\ 0 & 0 & \sqrt{2} & 0 \\ 0 & 0 & 0 & \sqrt{2} \end{pmatrix}$

- **Controlled-SX (CSX)** - Applies an SX gate to the second qubit if the first qubit is $|1\rangle$

  Matrix representation: $\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & \frac{1}{\sqrt{2}}(1+i) & \frac{1}{\sqrt{2}}(1-i) \\ 0 & 0 & \frac{1}{\sqrt{2}}(1-i) & \frac{1}{\sqrt{2}}(1+i) \end{pmatrix}$

- **Controlled-SXdg (CSXdg)** - Applies an SXdg gate to the second qubit if the first qubit is $|1\rangle$

  Matrix representation: $\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & \frac{1}{\sqrt{2}}(1-i) & \frac{1}{\sqrt{2}}(1+i) \\ 0 & 0 & \frac{1}{\sqrt{2}}(1+i) & \frac{1}{\sqrt{2}}(1-i) \end{pmatrix}$

- **Controlled-V (CV)** - Applies a V gate to the second qubit if the first qubit is $|1\rangle$

  Matrix representation: $\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & \frac{1}{\sqrt{2}}(1-i) & -\frac{1}{\sqrt{2}}(1+i) \\ 0 & 0 & \frac{1}{\sqrt{2}}(1+i) & \frac{1}{\sqrt{2}}(1+i) \end{pmatrix}$

- **Controlled-Vdg (CVdg)** - Applies a Vdg gate to the second qubit if the first qubit is $|1\rangle$

  Matrix representation: $\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & \frac{1}{\sqrt{2}}(1+i) & \frac{1}{\sqrt{2}}(1-i) \\ 0 & 0 & -\frac{1}{\sqrt{2}}(1+i) & \frac{1}{\sqrt{2}}(1+i) \end{pmatrix}$

References

[1] Nielsen, M. A., & Chuang, I. L. (2000). *Quantum Computation and Quantum Information.* Cambridge University Press, Cambridge, UK.

[2] Vazirani, U. (2009). *Lecture 1: Quantum Mechanics and Quantum Computation.* Retrieved from `https://people.eecs.berkeley.edu/~vazirani/s09quantum/notes/lecture1.pdf`