# Introduction to Quantum Computing & An Application in Power Systems

Parikshit Pareek
Assistant Professor

Department of Electrical Engineering
Indian Institute of Technology Roorkee

Faculty Development Program, NIT Warangal
December 31st, 2024

Parts of this talk are taken from Carleton Coffrin's talk on the same topic, with permission.
Reading: G. Nannicini, "An Introduction to Quantum Computing, without the Physics", 2020

# What this talk is about? And what not?

Basic Introduction of Quantum Computing without Requiring Physics Knowledge

A Lot of Linear Algebra & Notations!

Not about how to solve your favorite problems using Quantum Computers

Not an expert talk on all things Quantum Computing

# Two Part Talk:
# I. Basics of Quantum Computing
# II. Demystifying Quantum Power Flow

# Part I: Basics of Quantum Computing

# Quantum Computing

**What is Quantum Computing?**

- Computation and information processing based on `quantum mechanics` principles.
- Contrasts with traditional methods reliant on *classical physics*.

**Foundations of Quantum Mechanics**

- Describes nature at its most fundamental level.
- Developed in the early 20th century to explain subatomic particle behavior.

**Success and Impact**

- Underpins fields like Quantum Field Theory and Quantum Chromodynamics.
- Explains microscopic physical phenomena with exceptional accuracy.

# Origin of Quantum Computing: A Vision from Feynman

- **Feynman's Pivotal Question (1981):**

  *"Can probabilistic computers simulate quantum mechanics?"*
  His conclusion: **NO!**

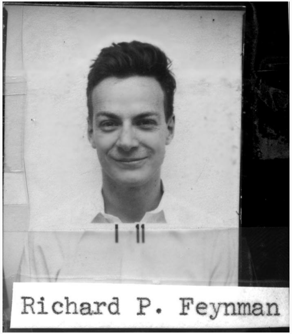



**Figure 1:** Richard Feynman's ID Card Photo at Los Alamos National Lab, 1943

# Origin of Quantum Computing: A Vision from Feynman

- **Feynman's Pivotal Question (1981):**

  *"Can probabilistic computers simulate quantum mechanics?"*
  His conclusion: **NO!**

- **Feynman's Insight:**

  *'". . . Nature isn't classical, damnit, and if you want to make a simulation of nature, you'd better make it quantum mechanical.
  By golly, it's a wonderful problem, because it doesn't look so easy."*

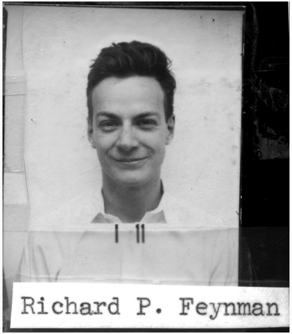



**Figure 1:** Richard Feynman's ID Card Photo at Los Alamos National Lab, 1943

# Origin of Quantum Computing: A Vision from Feynman

- **Feynman's Pivotal Question (1981):**

  *"Can probabilistic computers simulate quantum mechanics?"*
  His conclusion: **NO!**

- **Feynman's Insight:**

  *'". . . Nature isn't classical, damnit, and if you want to make a simulation of nature, you'd better make it quantum mechanical.*
  *By golly, it's a wonderful problem, because it doesn't look so easy."*

- **The Birth of Quantum Computing:**
  - Inspired by Feynman and contributions from David Deutsch, Paul Benioff, and Yuri Manin.
  - Introduced the concept of computation fundamentally based on *quantum mechanics*.

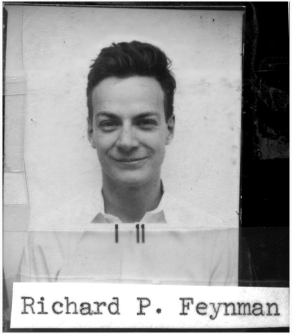



**Figure 1:** Richard Feynman's ID Card Photo at Los Alamos National Lab, 1943

Claim: Quantum computers solve things by trying every possibility at once.

Claim: Quantum computers solve things by trying every possibility at once.

MYTH

Claim: Quantum computers solve things by trying every possibility at once.

MYTH

Fact: Quantum computers can solve certain types of problems faster by carefully leveraging quantum effects such as entanglement, interference, superposition....

Claim: Quantum computers solve things by trying every possibility at once.

MYTH

Fact: Quantum computers can solve certain types of problems faster by carefully leveraging quantum effects such as entanglement, interference, superposition....

Also, There is The Measurement Issue!

# Quantum Computers are Here!

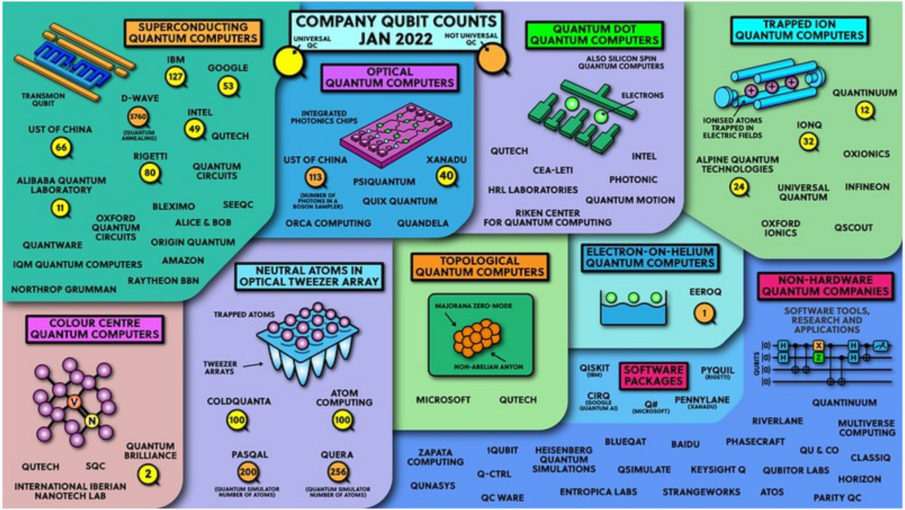



**Figure 2:** Domain of Science - Dominic Walliman

Most important point This introduction is mostly about Notation!

The gate model of quantum computing is just linear algebra over complex numbers with A LOT of special notations!

# Brief Refresher on Complex Numbers

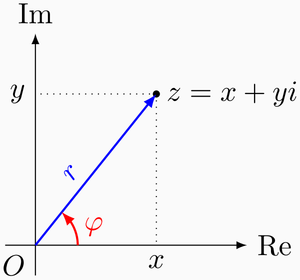
**A complex number has two parts:**

- "Real" part: Along the x-axis
- "Imaginary" part: Along the y-axis

**Can be represented as a magnitude $r$ and an angle $\phi$.**

**Two important operations:**

- Complex Conjugate: $Z^* = x - yi$
- Conjugate Transpose: $A^*$, $A^T$, $A^\dagger$, or $A^H$




$$Z = x + yi = r\angle\varphi$$
$$Z^* = x - yi = r\angle - \varphi$$

Quantum physics is made of waves $\implies$ Has amplitude & angle

$\implies$ Complex number are suitable for representation.

https://en.wikipedia.org/wiki/Complex_number https://en.wikipedia.org/wiki/Conjugate_transpose

# The Kronecker Product $\otimes$ (a.k.a. Tensor Product)

- A recursive matrix operation.
- Efficiently handles exponentially large structures.

$$\mathbf{u} \otimes \mathbf{v} = \begin{pmatrix} u_1 \\ u_2 \\ u_3 \end{pmatrix} \otimes \begin{pmatrix} v_1 \\ v_2 \end{pmatrix} = \begin{pmatrix} u_1 v_1 \\ u_1 v_2 \\ u_2 v_1 \\ u_2 v_2 \\ u_3 v_1 \\ u_3 v_2 \end{pmatrix}$$

$$\mathbf{A} \otimes \mathbf{B} = \begin{pmatrix} a_{11}\mathbf{B} & \cdots & a_{1n}\mathbf{B} \\ \vdots & \ddots & \vdots \\ a_{m1}\mathbf{B} & \cdots & a_{mn}\mathbf{B} \end{pmatrix}$$

**Example:**

$$\begin{pmatrix} 1 & 2 \\ 3 & 4 \end{pmatrix} \otimes \begin{pmatrix} 0 & 5 \\ 6 & 7 \end{pmatrix} = \begin{pmatrix} 1 \begin{pmatrix} 0 & 5 \\ 6 & 7 \end{pmatrix} & 2 \begin{pmatrix} 0 & 5 \\ 6 & 7 \end{pmatrix} \\ 3 \begin{pmatrix} 0 & 5 \\ 6 & 7 \end{pmatrix} & 4 \begin{pmatrix} 0 & 5 \\ 6 & 7 \end{pmatrix} \end{pmatrix} = \begin{pmatrix} 0 & 5 & 0 & 10 \\ 6 & 7 & 12 & 14 \\ 0 & 15 & 0 & 20 \\ 18 & 21 & 24 & 28 \end{pmatrix}$$

# Practice: Kronecker Product $\otimes$

Given the following matrices:

$$\mathbf{A} = \begin{pmatrix} 1 & 3 \\ 2 & 4 \end{pmatrix}, \quad \mathbf{B} = \begin{pmatrix} 0 & 5 \\ 6 & 7 \end{pmatrix},$$

calculate the Kronecker product $\mathbf{A} \otimes \mathbf{B}$.

**Class Question:** Can anyone tell what the first block $(a_{11}\mathbf{B})$ in the top-left corner will look like?

$$\begin{pmatrix} ? & ? \\ ? & ? \end{pmatrix}$$

- $a_{11} = 1$: $\quad 1 \cdot \mathbf{B} = \begin{pmatrix} 0 & 5 \\ 6 & 7 \end{pmatrix}$

$$\mathbf{A} = \begin{pmatrix} 1 & 3 \\ 2 & 4 \end{pmatrix}$$

$$\mathbf{B} = \begin{pmatrix} 0 & 5 \\ 6 & 7 \end{pmatrix}$$

$$\mathbf{A} = \begin{pmatrix} 1 & 3 \\ 2 & 4 \end{pmatrix}$$

$$\mathbf{B} = \begin{pmatrix} 0 & 5 \\ 6 & 7 \end{pmatrix}$$

- $a_{11} = 1$: $\quad 1 \cdot \mathbf{B} = \begin{pmatrix} 0 & 5 \\ 6 & 7 \end{pmatrix}$
- $a_{12} = 3$: $\quad 3 \cdot \mathbf{B} = \begin{pmatrix} 0 & 15 \\ 18 & 21 \end{pmatrix}$

$$\mathbf{A} = \begin{pmatrix} 1 & 3 \\ 2 & 4 \end{pmatrix}$$

$$\mathbf{B} = \begin{pmatrix} 0 & 5 \\ 6 & 7 \end{pmatrix}$$

- $a_{11} = 1$: $\quad 1 \cdot \mathbf{B} = \begin{pmatrix} 0 & 5 \\ 6 & 7 \end{pmatrix}$
- $a_{12} = 3$: $\quad 3 \cdot \mathbf{B} = \begin{pmatrix} 0 & 15 \\ 18 & 21 \end{pmatrix}$
- $a_{21} = 2$: $\quad 2 \cdot \mathbf{B} = \begin{pmatrix} 0 & 10 \\ 12 & 14 \end{pmatrix}$
- $a_{22} = 4$: $\quad 4 \cdot \mathbf{B} = \begin{pmatrix} 0 & 20 \\ 24 & 28 \end{pmatrix}$

$$\mathbf{A} = \begin{pmatrix} 1 & 3 \\ 2 & 4 \end{pmatrix}$$

$$\mathbf{B} = \begin{pmatrix} 0 & 5 \\ 6 & 7 \end{pmatrix}$$

- $a_{11} = 1$: $\quad 1 \cdot \mathbf{B} = \begin{pmatrix} 0 & 5 \\ 6 & 7 \end{pmatrix}$
- $a_{12} = 3$: $\quad 3 \cdot \mathbf{B} = \begin{pmatrix} 0 & 15 \\ 18 & 21 \end{pmatrix}$
- $a_{21} = 2$: $\quad 2 \cdot \mathbf{B} = \begin{pmatrix} 0 & 10 \\ 12 & 14 \end{pmatrix}$
- $a_{22} = 4$: $\quad 4 \cdot \mathbf{B} = \begin{pmatrix} 0 & 20 \\ 24 & 28 \end{pmatrix}$

**Final Answer: Combine into a 4x4 matrix**

$$\mathbf{A} \otimes \mathbf{B} = \begin{pmatrix} 0 & 5 & 0 & 15 \\ 6 & 7 & 18 & 21 \\ 0 & 10 & 0 & 20 \\ 12 & 14 & 24 & 28 \end{pmatrix}.$$

# Dirac Notation (BraKet) $\langle | \rangle$

- **Given a complex Euclidean space** $\mathbb{S} \equiv \mathbb{C}^n$:

  $|\psi\rangle \in \mathbb{S}$ denotes a column vector.

  $\langle\psi| \in \mathbb{S}$ denotes a row vector that is the conjugate transpose of $|\psi\rangle \in \mathbb{S}$:

$$\langle\psi| = |\psi\rangle^\dagger$$

- $\langle\psi|$ is called a **bra**.
- $|\psi\rangle$ is called a **ket**
- A sequence of matrix operations (applied **right to left**) looks like:

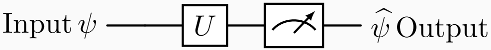
$$\langle\psi_2| \ldots CBA \, |\psi_1\rangle \, .$$

- **What does this have to do with Quantum Computing?**

  Quantum computers work with large complex Euclidean spaces ($\mathbb{C}^n$).

# What is a Gate-Based Quantum Computer?

- The quantum computer has a **state** ($\psi$) that is contained in a quantum register and is initialized in a predefined way.
- The state evolves by applying **operations** ($U$) specified in advance in the form of an algorithm.
- At the end of the computation, some information on the state of the quantum register is obtained by means of a special operation, called a **measurement**.




- There exists an alternative model of computation, called the **adiabatic model**. It is equivalent to the circuit model [Aharonov et al., 2008].

# State of A Quantum Computer

# The Standard Basis of One Qubit

- The Qubit's state is a *unit vector* of 2 complex numbers (i.e., $\mathbb{C}^2$).
- The standard basis is $|0\rangle$ for the first entry and $|1\rangle$ for the second entry.

**Standard Basis**

$$|0\rangle = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \quad |1\rangle = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$$

**Unit Vector Requirement**

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

**Two Qubit Example**

- The quantum state space grows as $(\mathbb{C}^2)^{\otimes q}$, where $q$ is the number of Qubits.
- For two Qubits, we have:

$$(\mathbb{C}^2)^{\otimes 2} = \mathbb{C}^2 \otimes \mathbb{C}^2 = \mathbb{C}^4$$

with the following 4 basis elements: $|00\rangle, |01\rangle, |10\rangle, |11\rangle$

`Quantum state vector grows exponentially:` $2^{\#\text{Qubits}}$

# A Trick

**How to write a $0, 1$ string for a Qubit?**

**Example:** $|010\rangle$

- $|010\rangle$ has 3-Qubit $\implies$ $(\mathbb{C}^2)^{\otimes 3} = \mathbb{C}^8$ 8-Dimensional Vector
- If $010$ is in binary, decimal equivalent is $2$
- Place '$1$' at $2^{nd}$ location of 8-Dimensional Vector with index starting from 0

$$|010\rangle = |0\rangle \otimes |1\rangle \otimes |0\rangle = \begin{pmatrix} 0 \\ 0 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}$$

# Secret Sauce of Quantum: Superposition

- `The quantum system is a linear combination of basis states`

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

- $\alpha, \beta$ are coefficient of basis states & are complex number
- $\alpha$ and $\beta$ represent probability of finding $|0\rangle$ and $|1\rangle$ respectively. Thus

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

- Square because these are complex number and sum is one as they are probabilities

**Very Important**

Superposition has `NO` classical equivalent
Bits are always in basis state $\implies$ String of $0s$ and $1s$

# Secret Sauce of Quantum: Entanglement

- A quantum system state that cannot be expressed as the Kronecker product of other states is an **entangled state**.
- Observed values of each Qubit (when measured) are correlated.
- Requires at least **2 Qubits**.

**"Bell Pair"**

(a.k.a. maximally entangled state):

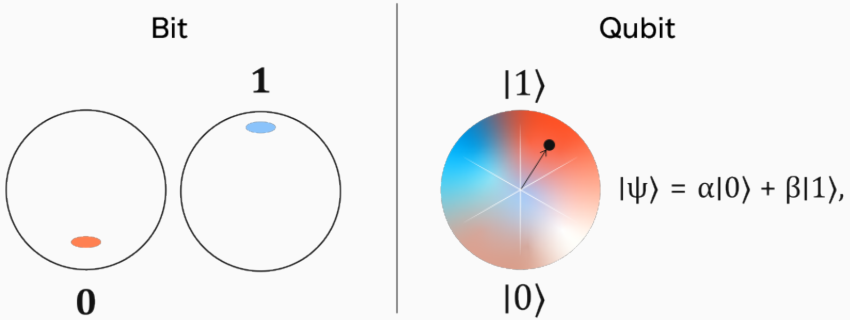
$$|\psi\rangle = \begin{array}{c} |00\rangle \\ |01\rangle \\ |10\rangle \\ |11\rangle \end{array} \begin{pmatrix} \sqrt{0.5} \\ 0.0 \\ 0.0 \\ \sqrt{0.5} \end{pmatrix}, \quad p_s = \begin{array}{c} |00\rangle \\ |01\rangle \\ |10\rangle \\ |11\rangle \end{array} \begin{pmatrix} 0.5 \\ 0.0 \\ 0.0 \\ 0.5 \end{pmatrix}$$

There is `No Setting` of constants $c_1$ to $c_8$ such that the Kronecker product produces the Bell state:

$$\begin{pmatrix} c_1 + c_2 i \\ c_3 + c_4 i \end{pmatrix} \otimes \begin{pmatrix} c_5 + c_6 i \\ c_7 + c_8 i \end{pmatrix} \neq \begin{pmatrix} \sqrt{0.5} + 0.0i \\ 0.0 + 0.0i \\ 0.0 + 0.0i \\ \sqrt{0.5} + 0.0i \end{pmatrix}$$

**Note:** If you read "0" on Qubit 1, then you will also read "0" on Qubit 2.

# Bit v/s Qubit




Source: https://www.qnulabs.com/blog/quantum-101-Qubit

# Reading a Quantum State

- When you "read" a Qubit, you get a basis state $0$ or $1$ (`not the` $\mathbb{C}^2$ `state vector!`).
- The amplitudes of the Quantum State ($\psi$) determine the probability distribution of the basis states that you will observe.

`Probability Distribution` $p_s = \text{diagonal}(|\psi\rangle\langle\psi|)$

$$|\psi\rangle = \begin{array}{c} |0\rangle \\ |1\rangle \end{array} \begin{pmatrix} \sqrt{0.5} + 0.0i \\ \sqrt{0.5} + 0.0i \end{pmatrix}, \quad p_s = \begin{array}{c} |0\rangle \\ |1\rangle \end{array} \begin{pmatrix} 0.5 \\ 0.5 \end{pmatrix}$$

# Reading a Quantum State

- When you "read" a Qubit, you get a basis state $0$ or $1$ (`not the` $\mathbb{C}^2$ `state vector!`).
- The amplitudes of the Quantum State ($\psi$) determine the probability distribution of the basis states that you will observe.

`Probability Distribution` $p_s = \text{diagonal}(|\psi\rangle\langle\psi|)$

$$|\psi\rangle = \begin{array}{c} |0\rangle \\ |1\rangle \end{array} \begin{pmatrix} \sqrt{0.5} + 0.0i \\ \sqrt{0.5} + 0.0i \end{pmatrix}, \quad p_s = \begin{array}{c} |0\rangle \\ |1\rangle \end{array} \begin{pmatrix} 0.5 \\ 0.5 \end{pmatrix} \qquad |\psi\rangle = \begin{array}{c} |0\rangle \\ |1\rangle \end{array} \begin{pmatrix} 0.5 + 0.25i \\ \sqrt{2}/4 + 0.75i \end{pmatrix}, \quad p_s = ?$$

# Step-by-Step: Calculating $p_s$ from $|\psi\rangle$

$$|\psi\rangle = \begin{matrix} |0\rangle \\ |1\rangle \end{matrix} \begin{pmatrix} 0.5 + 0.25i \\ \frac{\sqrt{2}}{4} + 0.75i \end{pmatrix}.$$

**Step 1: Find the Probability of Each Basis State ($p_s$)**

`Square of the magnitude of the corresponding amplitude`: $p_s = \text{diagonal}(|\psi\rangle \langle\psi|)$.

# Step-by-Step: Calculating $p_s$ from $|\psi\rangle$

$$|\psi\rangle = \begin{array}{c} |0\rangle \\ |1\rangle \end{array} \begin{pmatrix} 0.5 + 0.25i \\ \frac{\sqrt{2}}{4} + 0.75i \end{pmatrix}.$$

**Step 1: Find the Probability of Each Basis State ($p_s$)**

`Square of the magnitude of the corresponding amplitude`: $p_s = \text{diagonal}(|\psi\rangle \langle\psi|)$.

**Step 2: Compute the Magnitudes of Each Amplitude**

For $|0\rangle$: $|0.5 + 0.25i|^2 = (0.5)^2 + (0.25)^2 = 0.25 + 0.0625 = 0.3125 = \frac{5}{16}$.

For $|1\rangle$: $|\frac{\sqrt{2}}{4} + 0.75i|^2 = \left(\frac{\sqrt{2}}{4}\right)^2 + (0.75)^2 = \frac{2}{16} + 0.5625 = 0.625 = \frac{11}{16}$.

# Step-by-Step: Calculating $p_s$ from $|\psi\rangle$

$$|\psi\rangle = \begin{array}{c} |0\rangle \\ |1\rangle \end{array} \begin{pmatrix} 0.5 + 0.25i \\ \frac{\sqrt{2}}{4} + 0.75i \end{pmatrix}.$$

**Step 1: Find the Probability of Each Basis State ($p_s$)**

`Square of the magnitude of the corresponding amplitude`: $p_s = \text{diagonal}(|\psi\rangle \langle\psi|)$.

**Step 2: Compute the Magnitudes of Each Amplitude**

For $|0\rangle$: $|0.5 + 0.25i|^2 = (0.5)^2 + (0.25)^2 = 0.25 + 0.0625 = 0.3125 = \frac{5}{16}$.

For $|1\rangle$: $|\frac{\sqrt{2}}{4} + 0.75i|^2 = \left(\frac{\sqrt{2}}{4}\right)^2 + (0.75)^2 = \frac{2}{16} + 0.5625 = 0.625 = \frac{11}{16}$.

The probability distribution $p_s$ is:

$$p_s = \begin{array}{c} |0\rangle \\ |1\rangle \end{array} \begin{pmatrix} \frac{5}{16} \\ \frac{11}{16} \end{pmatrix}.$$

# Step-by-Step: Calculating $p_s$ from $|\psi\rangle$

$$|\psi\rangle = \begin{array}{c} |0\rangle \\ |1\rangle \end{array} \begin{pmatrix} 0.5 + 0.25i \\ \frac{\sqrt{2}}{4} + 0.75i \end{pmatrix}.$$

**Step 1: Find the Probability of Each Basis State ($p_s$)**

`Square of the magnitude of the corresponding amplitude`: $p_s = \text{diagonal}(|\psi\rangle\langle\psi|)$.

**Step 2: Compute the Magnitudes of Each Amplitude**

For $|0\rangle$: $|0.5 + 0.25i|^2 = (0.5)^2 + (0.25)^2 = 0.25 + 0.0625 = 0.3125 = \frac{5}{16}$.

For $|1\rangle$: $|\frac{\sqrt{2}}{4} + 0.75i|^2 = \left(\frac{\sqrt{2}}{4}\right)^2 + (0.75)^2 = \frac{2}{16} + 0.5625 = 0.625 = \frac{11}{16}$.

The probability distribution $p_s$ is:

$$p_s = \begin{array}{c} |0\rangle \\ |1\rangle \end{array} \begin{pmatrix} \frac{5}{16} \\ \frac{11}{16} \end{pmatrix}.$$

**The probabilities of measuring $|0\rangle$ and $|1\rangle$ are $\frac{5}{16}$ and $\frac{11}{16}$, respectively.**

# Reading a Quantum State

- When you "read" a Qubit, you get a basis state $0$ or $1$ (`not the` $\mathbb{C}^2$ `state vector!`).
- The amplitudes of the Quantum State ($\psi$) determine the probability distribution of the basis states that you will observe.

$$\text{Probability Distribution} \quad p_s = \mathsf{diagonal}(|\psi\rangle\langle\psi|)$$

$$|\psi\rangle = \begin{array}{c} |0\rangle \\ |1\rangle \end{array} \begin{pmatrix} \sqrt{0.5} + 0.0i \\ \sqrt{0.5} + 0.0i \end{pmatrix}, \quad p_s = \begin{array}{c} |0\rangle \\ |1\rangle \end{array} \begin{pmatrix} 0.5 \\ 0.5 \end{pmatrix} \quad |\psi\rangle = \begin{array}{c} |0\rangle \\ |1\rangle \end{array} \begin{pmatrix} 0.5 + 0.25i \\ \sqrt{2}/4 + 0.75i \end{pmatrix}, \quad p_s = \begin{array}{c} |0\rangle \\ |1\rangle \end{array} \begin{pmatrix} \frac{5}{16} \\ \frac{11}{16} \end{pmatrix}$$

# Reading a Quantum State

- When you "read" a Qubit, you get a basis state $0$ or $1$ (`not the` $\mathbb{C}^2$ `state vector!`).
- The amplitudes of the Quantum State ($\psi$) determine the probability distribution of the basis states that you will observe.

$$\text{Probability Distribution} \quad p_s = \mathsf{diagonal}(|\psi\rangle\langle\psi|)$$

$$|\psi\rangle = \begin{array}{c} |0\rangle \\ |1\rangle \end{array} \begin{pmatrix} \sqrt{0.5} + 0.0i \\ \sqrt{0.5} + 0.0i \end{pmatrix}, \quad p_s = \begin{array}{c} |0\rangle \\ |1\rangle \end{array} \begin{pmatrix} 0.5 \\ 0.5 \end{pmatrix} \quad |\psi\rangle = \begin{array}{c} |0\rangle \\ |1\rangle \end{array} \begin{pmatrix} 0.5 + 0.25i \\ \sqrt{2}/4 + 0.75i \end{pmatrix}, \quad p_s = \begin{array}{c} |0\rangle \\ |1\rangle \end{array} \begin{pmatrix} \frac{5}{16} \\ \frac{11}{16} \end{pmatrix}$$

**Goal of a Quantum Algorithm:**

- `Put most of the state probability into the "right" basis vector` (e.g., optimal solution to a combinatorial problem).
- `Then reading the solution out of the quantum computer is easy.`

# Quantum Gates

- **Quantum Operations** $(U)$ are called **Gates**.
  - A square matrix of complex numbers applied to the quantum state $(\psi)$.
  - Theory of quantum mechanics requires these operations to be **unitary**.
  - Given the state vector has size $\mathbb{C}^{2^q}$, the gates have size $\mathbb{C}^{2^q \times 2^q}$, where $q$ is the number of Qubits.
- **Unitary Property:**

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

- **Important Consequences:**

  Quantum operations are linear.

  Quantum operations are reversible.

## Quantum Gates

- **Quantum Operations** $(U)$ are called **Gates**.
  - A square matrix of complex numbers applied to the quantum state $(\psi)$.
  - Theory of quantum mechanics requires these operations to be **unitary**.
  - Given the state vector has size $\mathbb{C}^{2^q}$, the gates have size $\mathbb{C}^{2^q \times 2^q}$, where $q$ is the number of Qubits.
- **Unitary Property:**

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

- **Important Consequences:**

  Quantum operations are linear.

  Quantum operations are reversible.

**Note**

While these properties may seem to be extremely restrictive, a quantum computer is **Turing-complete!** [Deutsch1985]

$\Longrightarrow$ It can do whatever a classical computer can do, and more.

# Quantum Gates, One Qubit Example

$$U = \begin{pmatrix} 1 & 0 \\ 0 & i \end{pmatrix} \quad |\psi_0\rangle = \begin{pmatrix} 0 \\ -1 \end{pmatrix}$$

# Quantum Gates, One Qubit Example

$$U = \begin{pmatrix} 1 & 0 \\ 0 & i \end{pmatrix} \quad |\psi_0\rangle = \begin{pmatrix} 0 \\ -1 \end{pmatrix}$$

$$U|\psi_0\rangle = \begin{pmatrix} 0 \\ -i \end{pmatrix}$$

# Quantum Gates, One Qubit Example

$$U = \begin{pmatrix} 1 & 0 \\ 0 & i \end{pmatrix} \quad |\psi_0\rangle = \begin{pmatrix} 0 \\ -1 \end{pmatrix}$$

$$U|\psi_0\rangle = \begin{pmatrix} 0 \\ -i \end{pmatrix}$$

$$UU|\psi_0\rangle = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$$

# Quantum Gates, One Qubit Example

$$U = \begin{pmatrix} 1 & 0 \\ 0 & i \end{pmatrix} \quad |\psi_0\rangle = \begin{pmatrix} 0 \\ -1 \end{pmatrix}$$

$$U|\psi_0\rangle = \begin{pmatrix} 0 \\ -i \end{pmatrix}$$

$$UU|\psi_0\rangle = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$$

$$UUU|\psi_0\rangle = \begin{pmatrix} 0 \\ i \end{pmatrix}$$

# Quantum Gates, One Qubit Example

$$U = \begin{pmatrix} 1 & 0 \\ 0 & i \end{pmatrix} \quad |\psi_0\rangle = \begin{pmatrix} 0 \\ -1 \end{pmatrix}$$

$$U|\psi_0\rangle = \begin{pmatrix} 0 \\ -i \end{pmatrix}$$

$$UU|\psi_0\rangle = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$$

$$UUU|\psi_0\rangle = \begin{pmatrix} 0 \\ i \end{pmatrix}$$

$$UUUU|\psi_0\rangle = \begin{pmatrix} 0 \\ -1 \end{pmatrix}$$

# Quantum Gates, One Qubit Example

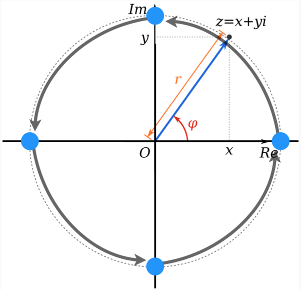
$$U = \begin{pmatrix} 1 & 0 \\ 0 & i \end{pmatrix} \quad |\psi_0\rangle = \begin{pmatrix} 0 \\ -1 \end{pmatrix}$$

$$U|\psi_0\rangle = \begin{pmatrix} 0 \\ -i \end{pmatrix}$$

$$UU|\psi_0\rangle = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$$

$$UUU|\psi_0\rangle = \begin{pmatrix} 0 \\ i \end{pmatrix}$$

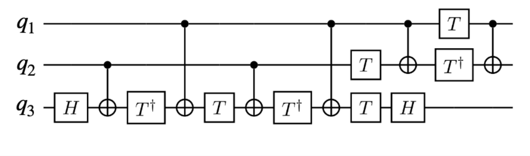
$$UUUU|\psi_0\rangle = \begin{pmatrix} 0 \\ -1 \end{pmatrix}$$




- Rotation shown in the complex plane ($z = x + yi$).
- $r$: Magnitude, $\phi$: Phase.

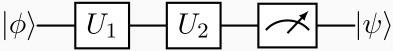
# Quantum Computer Program

- Most commonly shown as a sequence of one and two Qubit gates




- Scores are read from the left to right, But written right to left

$$|\psi\rangle = U_2|U_1\,|\phi\rangle$$




## Common Single Qubit Gates

| Operator | Gate(s) | Matrix |
|---|---|---|
| Pauli-X (X) | X | $\begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$ |
| Pauli-Y (Y) | Y | $\begin{bmatrix} 0 & -i \\ i & 0 \end{bmatrix}$ |
| Pauli-Z (Z) | Z | $\begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}$ |
| Hadamard (H) | H | $\frac{1}{\sqrt{2}}\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$ |
| Phase (S, P) | S | $\begin{bmatrix} 1 & 0 \\ 0 & i \end{bmatrix}$ |
| $\pi/8$ (T) | T | $\begin{bmatrix} 1 & 0 \\ 0 & e^{i\pi/4} \end{bmatrix}$ |

## Common Two-Qubit Gate

| | | |
|---|---|---|
| Controlled Not (CNOT, CX) | | $\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}$ |

# The Challenge of Quantum Computing

- What useful computations can you do with unitary transforms on exponentially large matrices?
- Only a few great answers... `So Far!`
  - Simulate Quantum Systems (solve Schrödinger Equation)
  - Factor Integers (Shor's Algorithm)
  - Unstructured Search (Grover's Algorithm)
  - Solving Linear Systems of Equations (HHL)
  - Simulating Large Collections of Oscillating Springs (arXiv:2303.13012)

An Excellent Resource: "`Quantum Algorithm Implementations for Beginners`"
ArXiv:1804.03719

# Questions on Part I?

# Part II: Demystifying Quantum Power Flow