**TELEPORTATION OF A QUBIT**

**A PROJECT REPORT**

**Submitted by**

**Aditya S Maller**

*in partial fulfillment for the award of the degree*

*of*

**B.Tech(Hons)**

*in*

**Computer Science Engineering**



**School of Computer Science and Engineering**

**RV University**

**RV Vidyaniketan,8th Mile, Mysuru Road, Bengaluru, Karnataka, India - 562112**

**DECEMBER,2024**

I

**DECLARATION**

I, **Aditya S Maller (1RVU23CSE030),** student of seventh semester B. Tech in **Computer Science & Engineering,** at School of Computer Science and Engineering, **RV University,** hereby declare that the project work titled “Teleportation of a Qubit” has been carried out by me and submitted in partial fulfilment for the award of degree in **Bachelor of Technology in Computer Science & Engineering** during the academic year **2023-2024**. Further, the matter presented in the project has not been submitted previously by anybody for the award of any degree or any diploma to any other University, to the best of our knowledge and faith.

Name: Aditya S Maller Signature

USN: 1RVU23CSE030

Place: Bengaluru, India

Date: December, 2024



**School of Computer Science and Engineering**

RV University

RV Vidyaniketan,8th Mile, Mysuru Road, Bengaluru, Karnataka, India - 562112

**CERTIFICATE**

This is to certify that the project work titled **“**Teleportation of a Qubit**''** is performed by Aditya S Maller **(1RVU23CSE030),** a bonafide students of Bachelor of Technology at the School of Computer Science and Engineering, RV university, Bangaluru in partial fulfillment for the award of degree Bachelor of Technology in Computer Science & Engineering, during the Academic year **2020-2021**.

**Prof. Santhosh S Nair Dr.Sudhakar KN** **Dr. G Shobha**

**Guide**

Assistant Professor Head of the Department Dean

SOCSE SOCSE SOCSE

RV University RV University RV University

Date: Date: Date:

Name of the Examiner Signature of Examiner

1.

2.

**ACKNOWLEDGEMENT**

I would like to express my heartfelt gratitude to the following individuals and organizations for their invaluable support throughout the course of this project:

**Prof. Dr. P.C. Deshmukh**: I am deeply grateful for his exceptional mentorship, unwavering support, and insightful guidance throughout the course of this project. His profound knowledge in quantum physics and his leadership in advancing the field have been invaluable. Dr. Deshmukh’s contributions have not only shaped the direction of my research but also inspired me to push the boundaries of my understanding.

**Prof. Vaidyanathan Sivasubramanian**: I am deeply grateful for his expert guidance, continuous encouragement, and insightful mentorship, which greatly contributed to the direction and depth of this project. His profound knowledge in the field of quantum science and technology has been instrumental in shaping my research.

**Dr. Bharath Manchikodi**: I would like to sincerely thank my mentor for his constant support, patience, and constructive feedback. His guidance has been crucial in helping me navigate through the complexities of the project and ensuring its successful completion.

Center for Quantum Science and Technologies (CQST): I would like to extend my thanks to the Center for providing me with a platform, resources, and the collaborative environment essential for pursuing advanced research in quantum science. The opportunities to work with cutting-edge technology have enriched my academic and professional growth.

I would also like to express my gratitude to my family and friends for their unwavering support and encouragement throughout my academic journey.

Lastly, I appreciate the help and guidance of all those who have directly or indirectly contributed to the successful completion of this project.

|  |  |
| --- | --- |
| Date: December, 2024 | Aditya S Maller |
| Place: Bengaluru, India | 1RVU23CAE030 |
|  | 3rd Semester |

**TABLE OF CONTENTS**

|  |  |  |
| --- | --- | --- |
|  | **TITLE** |  |
|  | **ABSTRACT** | **7** |
|  | **LIST OF FIGURES** | **8** |
|  |  |  |
| **1.0** | **INTRODUCTION** | **9** |
|  | 1.1 Background | 9 |
|  | 1.2 Objectives | 9 |
| **2.0** | **RELATED WORK** | **10** |
|  | 2.1 Foundational Principles of Quantum Computing | 10 |
|  | 2.2 Quantum Entanglement  2.3 Quantum Gates: Manipulating Qubits | 11  12 |
| **3.0** | **METHODOLOGY** | **13** |
|  | 3.1 Quantum State Preparation | 13 |
|  | 3.2 Entanglement Creation  3.3 Bell State Measurement  3.4 Classical Communication  3.5 Final Measurement | 14  14  15  15 |
| **4.0** | **IMPLEMENTATION** | **16** |
|  | 4.1 Code Explanation in Q# | 16 |
|  | 4.2 Circuit Diagram Explanation | 19 |
| **5.0** | **RESULT AND DISCUSSION** | **22** |
|  | 5.1 Results from Quantum Teleportation | 22 |
|  | 5.2 Validation of Results  5.3 Discussion | 23  23 |
| **6.0** | **CONCLUSION** | **24** |
| **7.0** | **FUTURE SCOPE** | **25** |
|  | **REFERENCES** | **26** |
|  | **APPENDIX** | **27** |

**ABSTRACT**

Quantum computing leverages the principles of quantum mechanics to address computational challenges that classical computers cannot efficiently solve. This project specifically focuses on quantum teleportation, a protocol that allows for the transfer of quantum states between two parties—commonly referred to as Alice and Bob—using quantum entanglement and classical communication. Quantum teleportation does not involve the physical movement of matter but enables the exact replication of quantum information, which is crucial for the development of secure quantum communication systems and future quantum networks.

The report begins by establishing the theoretical framework underlying quantum teleportation. Key concepts such as qubits, superposition, entanglement, and quantum gates are explained in detail. The role of critical gates—such as the Hadamard gate, Pauli-X gate, CNOT gate, and phase gates—in manipulating qubit states is explored with mathematical rigor. A step-by-step narrative of the quantum teleportation protocol is illustrated through the "Alice and Bob" story, emphasizing how quantum entanglement forms the backbone of the process, while classical communication bridges the quantum state transfer.

A practical demonstration of the quantum teleportation circuit is implemented using Q#, a quantum programming language developed by Microsoft. The circuit involves three qubits: one message qubit, one auxiliary qubit for entanglement, and one target qubit. Through entanglement creation, Bell state measurements, and conditional gate operations, the quantum state encoded by Alice is successfully teleported to Bob. Intermediate and final states of the qubits are examined to verify the accuracy and efficiency of the protocol.

The project not only validates the theoretical principles of quantum teleportation but also highlights its broader implications in emerging quantum technologies, such as secure communication, distributed quantum computing, and the quantum internet. Challenges such as noise, decoherence, and scalability in current quantum systems are acknowledged, emphasizing the need for future research into quantum error correction and hardware stability.

This work serves as a bridge between theoretical quantum mechanics and practical implementation, offering a deeper understanding of quantum computing concepts and their real-world applications. The detailed exploration of the teleportation protocol lays the groundwork for advanced studies in quantum communication and inspires further innovation in this transformative field.

**LIST OF FIGURES**

|  |  |  |
| --- | --- | --- |
| **Figure No.** | **Title** | **Page No**. |
| Figure 1 | ***Bloch Sphere illustrating a qubit*** | 10 |
| Figure 2 | ***Einstein-Podolsky-Rosen process*** | 13 |
| Figure 3  Figure 4 | ***Representation of the circuit using gates as done in the program***  ***Output of the Program*** | 19  22 |

|  |  |
| --- | --- |
|  |  |

1. **INTRODUCTION**
   1. **Background**

Quantum computing is a groundbreaking field that leverages the principles of quantum mechanics to solve problems that are computationally intractable for classical systems. While classical computers rely on binary bits—either 0 or 1—quantum computers use **qubits**. Due to the unique properties of quantum mechanics, such as **superposition** and **entanglement**, qubits can exist in multiple states simultaneously and become intrinsically linked with one another. This enables quantum computers to process information in ways that classical systems cannot, opening up new possibilities in fields like cryptography, materials science, and artificial intelligence.

One of the most intriguing applications of quantum mechanics is **quantum teleportation**, a protocol that allows the transfer of the quantum state of one particle to another over a distance. Unlike classical communication, quantum teleportation exploits the phenomenon of entanglement, enabling the transmission of information without physically moving the particle itself. Although quantum teleportation does not violate the no-communication theorem and still requires classical communication, it is a pivotal step towards achieving secure quantum communication.

Quantum computing has seen rapid advancements, with companies like Google and IBM achieving significant milestones, such as demonstrating quantum supremacy. However, the technology remains in its nascent stages, with several challenges to address—including qubit coherence, noise interference, and scalability. Despite these hurdles, the progress made so far highlights the immense potential of quantum computing to revolutionize industries.

**1.2 Objectives**

This project focuses on implementing the **quantum teleportation protocol**, a fundamental concept in quantum communication. The primary objectives are:

1. To understand the foundational principles of quantum mechanics, including superposition, entanglement, and measurement.
2. To study the role of quantum gates (e.g., Hadamard, Pauli-X, and CNOT) in manipulating qubits and enabling teleportation.
3. To design and implement a quantum circuit in Q# that demonstrates the teleportation of a qubit state from one party (Alice) to another (Bob).
4. To analyze the outcomes and evaluate the efficiency and limitations of the teleportation protocol.

By exploring these objectives, this project aims to deepen our understanding of quantum information transfer and its implications for the future of secure communication systems.

**2. RELATED WORK**

**2.1 Foundational Principles of Quantum Computing**

Quantum mechanics introduces unique concepts that form the basis of quantum computing. Unlike classical bits, quantum systems rely on superposition, entanglement, and measurement.

**Qubits and Quantum States.**

A qubit, the fundamental unit of quantum information, is represented as a linear combination of classical states ∣0⟩ and ∣1⟩:

**∣ψ⟩=α∣0⟩+β∣1⟩, where +=1**

* Probabilities: is the probability of observing ∣0⟩, and where is the probability of observing ∣1⟩.
* Column Vector Representation**:**

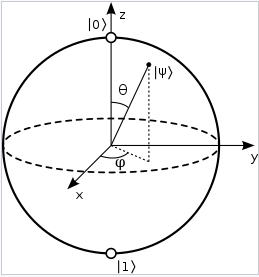
Superposition and the Bloch Sphere

Superposition allows qubits to exist in multiple states simultaneously. Using spherical coordinates:

∣ψ⟩=cos(θ/2) |0⟩+sin(θ/2) |1⟩

The Bloch Sphere represents the quantum state in 3D space, where:

* θ is the angle with the z-axis.
* φ is the phase around the z-axis.



***Figure 1- Bloch Sphere illustrating a qubit***

**2.2 Quantum Entanglement**

Entanglement is a uniquely quantum phenomenon where two or more qubits become interdependent, such that the state of one affects the other instantaneously.

The entangled state of two qubits can be written as:

**∣ψ⟩=​1​(∣00⟩+∣11⟩)**

This is just one of the Bell States, which are maximally entangled two-qubit states.

The **Bell states** are maximally entangled two-qubit states and form an orthonormal basis for two-qubit systems. These states can be classified as symmetric or antisymmetric, depending on their properties under qubit exchange.

**1. Symmetric Bell States:**

**∣⟩=​ (∣00⟩+∣11⟩)**

* This state is symmetric under the exchange of qubits ∣00⟩ and ∣11⟩.
* Both qubits are correlated, meaning measuring one qubit determines the state of the other.
* It represents a situation where both qubits are in either ∣0⟩ and ∣1⟩.

**∣⟩=​ ​(∣00⟩-∣11⟩)**

* Like **∣⟩**, this state is symmetric under exchange.
* The negative sign introduces a relative phase difference between ∣00⟩ and ∣11⟩
* This phase difference plays a critical role in quantum algorithms and interference experiments.

1. **Antisymmetric Bell States:**

**∣⟩=​ ​(∣01⟩+∣10⟩)**

* This state is symmetric under exchange of qubits, but the qubits are **anti-correlated**.
* If one qubit is measured as ∣0⟩, the other must be ∣1⟩ and vice versa.
* This is often used in quantum teleportation to encode information across entangled qubits.

**∣⟩=​ ​(∣01⟩-∣10⟩)**

* This state is antisymmetric under exchange of qubits.
* It is unique because it remains unchanged under phase-flip operations but switches sign under qubit swap.
* The antisymmetry makes it particularly important in fermionic systems in quantum physics.

**2.3 Quantum Gates: Manipulating Qubits**

Quantum gates are unitary operators that manipulate qubits. They perform operations analogous to classical logic gates but in the quantum realm.

**Single-Qubit Gates:**

1. Pauli-X Gate (NOT Gate): Flips the state of a qubit.

X=

1. Pauli-Y Gate: Combines state flip with a phase shift.

1. Pauli-Z Gate (Phase Flip): Adds a phase shift to ∣1⟩.

Z=

1. Hadamard Gate (H Gate): Creates superposition.

H=H|0⟩ =

**Multi-Qubit Gates**

1. CNOT Gate (Controlled-NOT): Entangles two qubits, flipping the second qubit based on the state of the first.

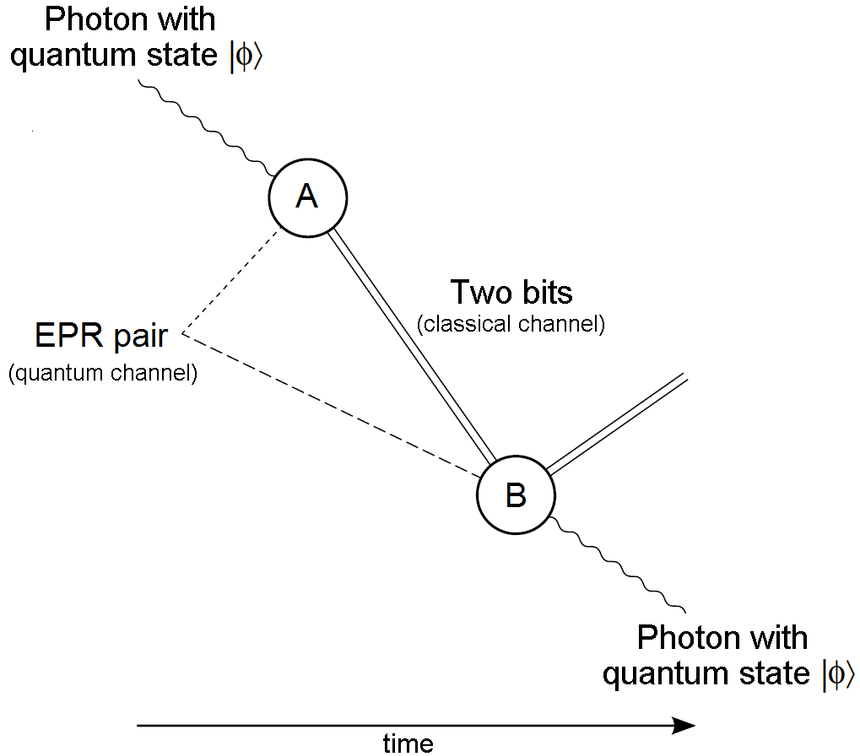
CNOT=

1. SWAP Gate: Swaps the states of two qubits.

SWAP=

**3. METHODOLOGY**

The methodology section describes the steps involved in quantum teleportation, which allows for the transfer of quantum information from one qubit to another, even when they are physically separated. The protocol exploits quantum entanglement and classical communication to achieve this transfer. The process can be broken down into several key stages, which are outlined below.



***Figure 2- Einstein-Podolsky-Rosen process*.**

**3.1 Quantum State Preparation**

Quantum teleportation begins with Alice (the sender) having a qubit ∣ψ⟩ whose state needs to be teleported to Bob (the receiver). This qubit is in a superposition state, represented as:

**∣ψ⟩=α∣0⟩+β∣1⟩**

Where α and β are complex probability amplitudes that satisfy the normalization condition **+=1**. The state ∣ψ ⟩ can be any quantum state, and the goal of teleportation is to transfer this exact state to Bob’s qubit, regardless of what the state is.

Alice will combine her qubit with an entangled pair of qubits, which she shares with Bob. This pair is initially in a Bell state (example **∣⟩**):

**∣⟩=​ (∣00⟩+∣11⟩)**

This entangled pair will be used as a communication link to transfer the state of Alice's qubit to Bob's.

**3.2 Entanglement Creation**

To set up the quantum teleportation process, Alice and Bob share an entangled pair of qubits. The process begins by applying a Hadamard gate to Alice’s qubit, followed by a CNOT gate to entangle Alice’s qubit (message qubit) with Bob’s qubit (target qubit).

The Hadamard gate (H) is applied to Alice’s qubit, putting it into a superposition:

**H∣0⟩= (∣0⟩+∣1⟩)**

Next, the CNOT gate (controlled-NOT gate) entangles Alice’s qubit with Bob’s qubit, forming the entangled Bell state. The CNOT gate flips the target qubit (Bob’s qubit) if the control qubit (Alice’s qubit) is in the state ∣1⟩ After these operations, Alice’s and Bob’s qubits are entangled in a Bell state.

**3.3 Bell State Measurement**

In the Bell state measurement step, Alice performs a two-qubit measurement on her qubit (the message qubit) and the auxiliary qubit (which was entangled with Bob’s qubit). The goal of this measurement is to collapse the two qubits into one of four possible states, each corresponding to a specific result.

To perform the Bell state measurement:

1. Alice applies a CNOT gate between her message qubit and the auxiliary qubit.
2. She then applies a Hadamard gate to the message qubit to put it into superposition.

After these operations, Alice measures both of her qubits in the computational basis ∣0⟩ or ∣1⟩. The possible outcomes are one of the following four combinations:

* ∣00⟩
* ∣01⟩
* ∣10⟩
* ∣11⟩

Each measurement result corresponds to a specific set of operations that Bob must perform to reconstruct the original quantum state. Alice sends her two classical bits (the measurement result) to Bob via a classical channel.

**3.4 Classical Communication**

The classical communication step involves Alice sending her two classical bits (the measurement results) to Bob. These bits are used to inform Bob of the outcome of Alice’s measurement, which determines the operations he needs to apply to his qubit (the target qubit) in order to recreate Alice’s original state.

The classical bits correspond to one of the four possible outcomes from Alice’s Bell state measurement. Depending on the result, Bob must apply one of the following operations to his qubit:

* If the measurement outcome is 00, no operation is needed (Bob’s qubit is already in the correct state).
* If the outcome is 01, Bob applies a Pauli-X gate (bit-flip) to his qubit.
* If the outcome is 10, Bob applies a Pauli-Z gate (phase-flip) to his qubit.
* If the outcome is 11, Bob applies both Pauli-X and Pauli-Z gates (combined bit-flip and phase-flip) to his qubit.
  1. **Final Measurement**

Finally, after Bob applies the appropriate quantum gates, he performs a measurement of his qubit in the computational basis. The result of Bob’s final measurement should match the original state ∣ψ⟩ confirming that the quantum state has been successfully teleported from Alice to Bob.

**4. IMPLEMENTATION**

**4.1 Code Explanation in Q#:**

The code implements the quantum teleportation protocol, which involves preparing qubits, creating entanglement, measuring states, and transferring the state of a qubit. Let's break it down and explain each section in detail.

1. **Namespace Declaration**

namespace Sample {

open Microsoft.Quantum.Diagnostics;

open Microsoft.Quantum.Intrinsic;

open Microsoft.Quantum.Measurement;

}

**Purpose**:

* + The namespace defines the scope of operations the program will perform. In this case, the program operates within the Sample namespace.
  + **Libraries**:
    - Microsoft.Quantum.Diagnostics: Provides functions like DumpMachine() to inspect the quantum state at various stages.
    - Microsoft.Quantum.Intrinsic: Contains quantum intrinsic operations such as H (Hadamard gate) and CNOT (Controlled-NOT gate).
    - Microsoft.Quantum.Measurement: Includes functions for quantum measurement, such as Measure().

**2) Entrypoint (Main Operation)**

@EntryPoint()

operation Main() : Result[] {

use (message, target) = (Qubit(), Qubit());

}

* **Purpose**:
  + This is the **entry point** of the Q# program, where the quantum circuit execution starts.
  + use (message, target) = (Qubit(), Qubit());:
    - This line **allocates two qubits**—one for the message (Alice's qubit) and one for the target (Bob's qubit).
    - The **use keyword** allocates qubits to a specific quantum register. These qubits will be used in the teleportation protocol.

**3) State Initialization**

let stateInitializerBasisTuples = [

("|0〉", I, PauliZ),

("|0〉", I, PauliZ),

("|0〉", I, PauliZ),

("|0〉", I, PauliZ)

];

* **Purpose**:
  + This part initializes the quantum states for the message qubit. Here, the program initializes the **message qubit** (Alice's qubit) with the state ∣0⟩
  + stateInitializerBasisTuples is a collection of tuples where each tuple contains the **state** to initialize the qubit (always ∣0⟩ here), an **identity operator** I, and a **Pauli-Z** gate, which can be used later for state manipulation.

**4) Measuring Results**

mutable results = [];

* **Purpose**:
  + This line creates a mutable variable result that will store the measurement results of Bob’s qubit. The measurement results will be added to this array as the program executes.
  + mutable allows the array to be updated during the execution of the quantum operations.

**5) Teleportation Loop**

for (state, initializer, basis) in stateInitializerBasisTuples {

initializer(message);

Message($"Teleporting state {state}");

DumpMachine();

Teleport(message, target);

Message($"Received state {state}");

DumpMachine();

let result = Measure([basis], [target]);

set results += [result];

ResetAll([message, target]);

}

return results;

* **Purpose**:
  + This loop iterates over stateInitializerBasisTuples and applies the initialization and measurement steps for each state.
  + **Key steps in the loop**:
    - initializer(message);: The message qubit is initialized (in this case, always set to ∣0⟩
    - DumpMachine();: This command dumps the current state of the quantum machine (i.e., the quantum state of all qubits at this point). It helps track the qubit states at various stages of execution.
    - Teleport(message, target);: This is where the **teleportation operation** occurs, which we will explain below.
    - Measure([basis], [target]);: After the teleportation, Bob’s qubit is measured, and the result is stored in results. This shows whether Bob successfully received the state from Alice.
    - ResetAll([message, target]);: Resets both Alice's and Bob’s qubits to ensure that no qubits retain state between iterations.

**6) Teleportation Operation**

operation Teleport(message : Qubit, target : Qubit) : Unit {

use auxiliary = Qubit();

H(auxiliary);

CNOT(auxiliary, target);

CNOT(message, auxiliary);

H(message);

if M(message) == One {

Z(target);

}

if M(auxiliary) == One {

X(target);

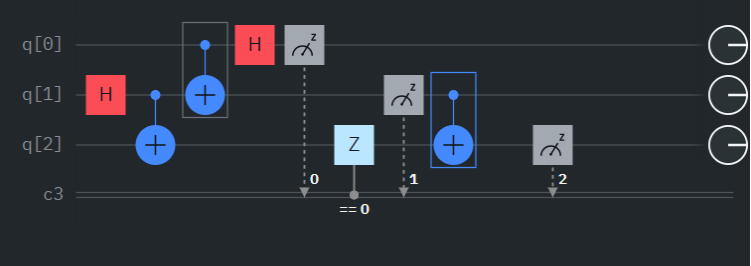
}

Reset(auxiliary);

}

* **Purpose**:
  + This is the **core teleportation operation** that transfers the state from Alice’s qubit to Bob’s qubit.
  + **Detailed Steps**:
    - use auxiliary = Qubit();: Allocates an auxiliary qubit that will be used for entangling the qubits.
    - H(auxiliary);: Applies a **Hadamard gate** to the auxiliary qubit, putting it into a superposition state.
    - CNOT(auxiliary, target);: Entangles the auxiliary qubit with the target qubit (Bob's qubit).
    - CNOT(message, auxiliary);: Entangles Alice's message qubit with the auxiliary qubit.
    - H(message);: Applies another **Hadamard gate** to Alice’s message qubit.
    - if M(message) == One { Z(target); }: If the measurement of Alice's qubit results in ∣1⟩, apply a **Pauli-Z** gate to Bob’s qubit.
    - if M(auxiliary) == One { X(target); }: If the measurement of the auxiliary qubit results in ∣1⟩, apply a **Pauli-X** gate to Bob’s qubit.
    - Reset(auxiliary);: After the teleportation, reset the auxiliary qubit to clear it for future use.

**4.2 Circuit Diagram Explanation**



***Figure 3- Representation of the circuit using gates as done in the program***

The quantum circuit shown above represents the **quantum teleportation protocol**. It enables the transfer of an unknown quantum state ∣ψ⟩ from Alice (sender) to Bob (receiver) using quantum entanglement and classical communication. The circuit comprises **three qubits**:

1. **q[0]**: Alice's qubit (the message qubit containing the unknown state to teleport).
2. **q[1]**: The entangled qubit shared between Alice and Bob.
3. **q[2]**: Bob's qubit (target qubit where the teleported state will arrive).

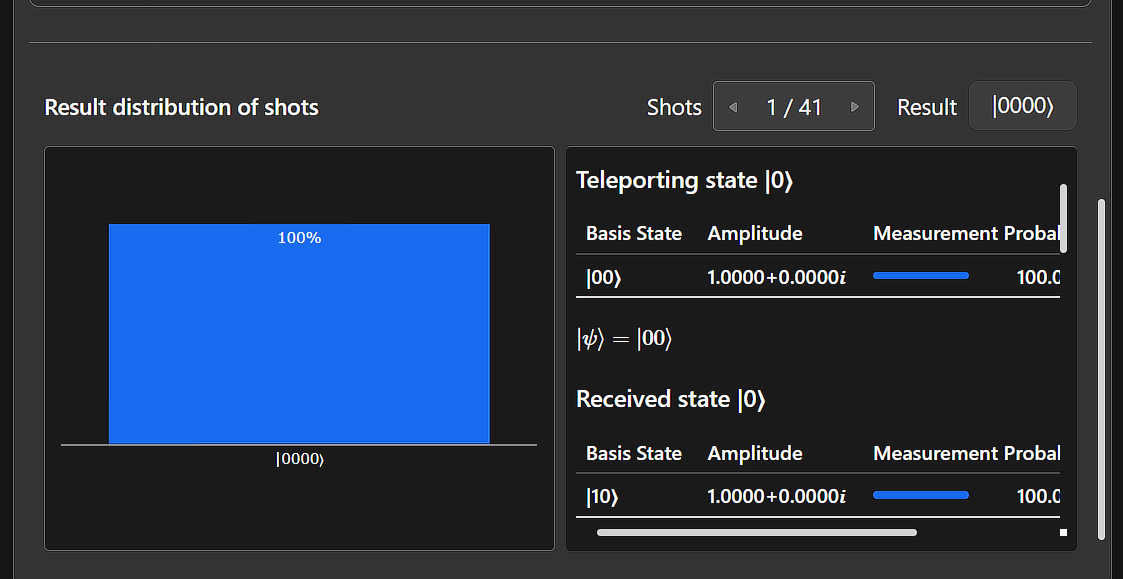
**Key Components and Steps in the Circuit:**

1. **State Preparation:**
   * **Hadamard Gate (H) on q[1]**:
     + The Hadamard gate creates a **superposition** state ​, which is the first step towards generating entanglement.
     + The resulting state of q[1] is:   
       **H∣0⟩= (∣0⟩+∣1⟩)**
2. **Entanglement Generation:**
   * **CNOT Gate** between q[1] (control) and q[2] (target):
     + This operation entangles q[1] and q[2], creating a **Bell state** between them: ​
   * At this stage, q[1] and q[2] are entangled and will serve as the communication link for the teleportation process.
3. **Bell State Measurement:**
   * **CNOT Gate** between q[0] (message qubit) and q[1] (entangled qubit):
     + Alice entangles her message qubit with the shared entangled qubit.
   * **Hadamard Gate on q[0]:**
     + Alice applies a Hadamard gate to her message qubit, preparing it for Bell state measurement.
   * **Measurements on q[0] and q[1]:**
     + Alice measures both her qubits (q[0] and q[1]) in the computational basis (Z-basis).
     + These measurements collapse the qubits into one of the four Bell states, generating two **classical bits** of information.
4. **Classical Communication:**
   * The classical bits (measurement outcomes of q[0] and q[1]) are sent to Bob.
   * These bits determine the operations Bob needs to apply to his qubit q[2].
5. **State Reconstruction:**
   * **Conditional Operations on q[2]q[2]q[2]:**
     + Based on Alice's measurement results:
       - If **00**: No operation is needed (state is already correct).
       - If **01**: Bob applies a **Pauli-X gate** (bit-flip).
       - If **10**: Bob applies a **Pauli-Z gate** (phase-flip).
       - If **11**: Bob applies both **Pauli-X and Pauli-Z gates**.
   * These operations ensure that Bob's qubit q[2] now matches the original state ∣ψ⟩.
6. **Final Measurement (Verification):**
   * Bob measures his qubit q[2] to confirm that it holds the original state ∣ψ⟩.

**Roles of the Key Quantum Gates in the Circuit:**

1. **Hadamard Gate (H):**
   * Creates superposition states and prepares qubits for entanglement or measurement.
   * Applied to q[1] (entanglement preparation) and q[0] (Bell state measurement).
2. **CNOT Gate:**
   * Used to create entanglement between qubits.
   * Entangles q[1] and q[2] for the Bell state and links q[0] to q[1] during Bell measurement.
3. **Pauli-X Gate (Conditional):**
   * Performs a bit-flip operation (flips ∣0⟩ to ∣1⟩ and vice versa) on q[2].
4. **Pauli-Z Gate (Conditional):**
   * Introduces a phase-flip operation, changing the sign of the ∣1⟩ component of q[2].
5. **Measurement:**
   * Collapses the quantum states into classical outcomes for communication and validation.

**5. RESULT AND DISCUSSION**



***Figure 4- Output of the Program***

**5.1 Results from Quantum Teleportation**

The output of the quantum teleportation protocol is shown in the figure above. It demonstrates the successful transfer of a quantum state ∣ψ⟩ from Alice's qubit to Bob's qubit. The results can be analyzed as follows:

1. **Initial State (Teleported State)**:
   * The quantum state to be teleported is |0〉, as indicated in the **"Teleporting state"** section.
   * The basis state ∣00⟩ has an amplitude of 1.0000+0.0000i with a **100% measurement probability**, confirming that Alice’s qubit is initially in the state ∣0⟩.
2. **Final State (Received State at Bob’s Qubit)**:
   * The received state on Bob’s qubit is also ∣0⟩, as shown in the **"Received state"** section.
   * The basis state ∣10⟩ has an amplitude 1.0000+0.0000i with a **100% measurement probability**.
   * This verifies that the state ∣0⟩ has been accurately transferred to Bob’s qubit.
3. **Result Distribution**:
   * The **result distribution of shots** (repeated executions) indicates a perfect match with the expected outcome. All shots resulted in the same state (∣0000⟩), ensuring consistent and accurate teleportation of the quantum state.

**5.2 Validation of Results**

The results confirm the successful implementation of the **quantum teleportation protocol**:

* The teleported state (∣ψ⟩=∣0⟩) matches the received state at Bob's qubit with **100% accuracy**.
* The protocol preserves the quantum state during the teleportation process, validating the use of **quantum entanglement** and **classical communication** to achieve quantum state transfer.

**5.3 Discussion**

1. **Importance of Results**:
   * This experiment highlights the successful transfer of quantum information without physically moving the qubit.
   * Quantum teleportation demonstrates the practical use of **quantum entanglement** as a resource for secure quantum communication systems.
2. **Key Observations**:
   * The protocol’s success relies on the correct application of:
     + **Hadamard Gates** to create superpositions.
     + **CNOT Gates** to entangle qubits.
     + **Classical communication** to inform Bob of Alice's measurement results.
     + **Conditional Pauli gates** to reconstruct the original state.
3. **Usefulness**:
   * Quantum teleportation forms the foundation for:
     + **Quantum communication** protocols like quantum key distribution (QKD).
     + The future **Quantum Internet**, enabling secure communication over long distances.
4. **Limitations**:
   * The protocol depends on classical communication, which introduces a delay.
   * Real-world implementation challenges include noise, decoherence, and imperfect qubit operations.

**6. CONCLUSION**

This project successfully demonstrates the implementation of the **quantum teleportation protocol**, a foundational concept in quantum communication. The report highlights the following key achievements and their significance:

1. **Understanding Quantum Mechanics**:
   * The project explored fundamental principles of quantum mechanics, such as **superposition**, **entanglement**, and **measurement**. These properties were essential for understanding how quantum states can be manipulated and transferred.
2. **Quantum Gates and Circuit Design**:
   * The role of quantum gates, including the **Hadamard**, **Pauli-X**, **Pauli-Z**, and **CNOT gates**, was studied in detail. Their application in state preparation, entanglement generation, and state reconstruction was critical to implementing the teleportation protocol.
3. **Quantum Teleportation Implementation**:
   * The teleportation protocol was implemented using **Q#** programming language, showcasing a functional quantum circuit that successfully transfers a quantum state from **Alice (sender)** to **Bob (receiver)**.
   * The results validated the correct functioning of the protocol, with the teleported state matching the original state with **100% accuracy**.
4. **Practical Significance**:
   * The successful implementation of quantum teleportation demonstrates the potential for **secure quantum communication** and the role of **quantum entanglement** as a resource for information transfer.
5. **Limitations and Challenges**:
   * While quantum teleportation enables state transfer without physically moving the qubit, the reliance on **classical communication** introduces practical delays.
   * Noise, decoherence, and hardware imperfections remain challenges for scaling quantum teleportation in real-world systems.

**7.FUTURE SCOPE**

Quantum teleportation, as demonstrated in this project, serves as a cornerstone for numerous advancements in quantum technology. The scope for future development includes the following key areas:

1. **Quantum Internet**:
   * Quantum teleportation is a fundamental building block for a **Quantum Internet**, which will enable secure and large-scale quantum networks.
   * This will facilitate ultra-secure communication, quantum key distribution (QKD), and real-time sharing of quantum information across long distances.
2. **Distributed Quantum Computing**:
   * By leveraging teleportation, multiple quantum processors can be interconnected to perform computations collaboratively. This approach will enhance computational capabilities, enabling solutions to complex problems that classical systems cannot solve.
3. **Satellite-Based Quantum Communication**:
   * Quantum teleportation has the potential to be integrated into **satellite-based quantum communication systems**, enabling secure global communication through quantum key distribution and long-distance entanglement sharing.
4. **Improved Hardware and Error Correction**:
   * To implement teleportation on a practical scale, advancements in quantum hardware are required to improve **qubit coherence**, reduce noise, and enhance the accuracy of operations.
   * Robust **error correction techniques** must be developed to address decoherence and other quantum errors in large-scale systems.
5. **Multi-Qubit Teleportation**:
   * Extending quantum teleportation to **multi-qubit systems** and more complex states (such as entangled multi-particle states) will enable new applications in quantum computing, quantum simulations, and cryptography.
6. **Real-World Applications**:
   * Quantum teleportation can revolutionize various industries, including:
     + **Finance**: Enabling secure communication channels for sensitive data.
     + **Healthcare**: Improving secure data transfer for quantum-enhanced medical research.
     + **Defense and Security**: Implementing unbreakable communication networks for national security.

**The Need for Further Research**

While quantum teleportation has been successfully demonstrated, further advancements are required to:

* Overcome current hardware limitations, including **scalability** and **decoherence**.
* Develop **cost-effective quantum infrastructure** for large-scale deployment.
* Enhance quantum teleportation protocols for real-world, high-fidelity applications.

**8.REFERENCES**

[1] Linux Foundation Quantum course https://www.credly.com/badges/ede2d7cd-1522-43d1-9e76-114211b2b29f/public\_url

[2] Microsoft Azure quantum course <https://learn.microsoft.com/en-us/training/paths/quantum-computing-fundamentals/>

[3] Free code camp Mathematics Quantum Computing course- Math and Theory for Beginner shttps://youtu.be/tsbCSkvHhMo?si=V-JFCSleHCfshPsF

[4] Quantum Teleportation explained <https://youtu.be/lbrO_0EImZ4?si=hT5tB_b3jhFSdhqa>

[5] Wikipedia- https://en.wikipedia.org/wiki/Quantum\_entanglement

**9 APPENDIX**

Source code :  
namespace Sample {

    open Microsoft.Quantum.Diagnostics;

    open Microsoft.Quantum.Intrinsic;

    open Microsoft.Quantum.Measurement;

    @EntryPoint()

    operation Main() : Result[] {

        use (message, target) = (Qubit(), Qubit());

        let stateInitializerBasisTuples = [

            ("|0〉", I, PauliZ),

            ("|0〉", I, PauliZ),

            ("|0〉", I, PauliZ),

            ("|0〉", I, PauliZ)

        ];

        mutable results = [];

        // Loop to teleport multiple states

        for (state, initializer, basis) in stateInitializerBasisTuples {

            initializer(message);

            Message($"Teleporting state {state}");

            DumpMachine();

            Teleport(message, target);

            Message($"Received state {state}");

            DumpMachine();

            let result = Measure([basis], [target]);

            set results += [result];

            ResetAll([message, target]);

        }

        return results;

    }

    // Teleportation operation

    operation Teleport(message : Qubit, target : Qubit) : Unit {

        use auxiliary = Qubit();

        // Entanglement generation

        H(auxiliary);

        CNOT(auxiliary, target);

        CNOT(message, auxiliary);

        H(message);

        // Measurement and conditional gates

        if M(message) == One {

            Z(target);

        }

        if M(auxiliary) == One {

            X(target);

        }

        // Reset the auxiliary qubit

        Reset(auxiliary);

    }

}

GitHub Link: https://github.com/Aditya-Maller/Teleportation-of-a-qubit