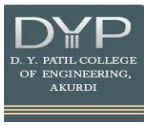


Dr. D. Y. Patil Pratishthan's
D. Y. PATIL COLLEGE OF ENGINEERING

Department of Artificial Intelligence and Data Science

LAB MANUAL
Computer Laboratory-II
(BE)
Semester I

Prepared by:
Dr. Bhagyashree A. Tingare

Computer Laboratory -II

Course Code	Course Name	Teaching Scheme (Hrs./ Week)	Credits
417526	Computer Laboratory-II: Quantum AI	4	2

Course Objectives:

- To develop real-world problem-solving ability
- To enable the student to apply AI techniques in applications that involve perception, reasoning, and planning
- To work in a team to build industry-compliant Quantum AI applications

Course Outcomes:

On completion of the course, learner will be able to–

- CO1: Evaluate and apply core knowledge of Quantum AI to various real-world problems.
- CO2: Illustrate and demonstrate Quantum AI tools for different dynamic applications.

Operating System recommended: Practical can be performed on suitable development platform

Table of Contents

Lab Assignment No.	1
Title	Implementations of 16 Qubit Random Number Generator
Roll No.	
Class	BE
Date of Completion	
Subject	Computer Laboratory-II :Quantum AI
Assessment Marks	
Assessor's Sign	

ASSIGNMENT No: 01

Title: Implementations of 16 Qubit Random Number Generator

Problem Statement: Implementations of 16 Qubit Random Number Generator

Prerequisite:

Basics of Python

Software Requirements: Jupyter

Hardware Requirements:

PIV, 2GB RAM, 500 GB HDD

Learning Objectives:

Learn to build 16 Qubit Random Number Generator

Outcomes:

After completion of this assignment students are able to understand how to build 16 Qubit Random Number Generator.

Theory: A 16-qubit random number generator is a device or system that utilizes a quantum computer's 16 qubits (quantum bits) to generate random numbers. Unlike classical computers that rely on deterministic algorithms to generate pseudo-random numbers, quantum computers leverage the inherent uncertainty and superposition properties of quantum states to create genuinely unpredictable outcomes.

In a 16-qubit random number generator, the qubits are prepared in specific quantum states that undergo controlled operations, leading to an entangled quantum state. The final measurement of these qubits in their entangled state produces a sequence of random bits that are the result of quantum effects. Due to the probabilistic nature of quantum measurements, the outcome cannot be predicted with certainty, ensuring the generated numbers are truly random.

These quantum random number generators have applications in various fields such as cryptography, secure communications, simulations, and scientific research, where high-quality random numbers are crucial for ensuring security and enhancing the performance of certain algorithms.

A 16-qubit random number generator uses quantum bits to create truly unpredictable numbers. Its applications include:

Super Secure Codes: Generates keys for ultra-safe encryption.

Unhackable Messages: Makes sure messages stay private in communication.

Better Guesses: Improves computer predictions and simulations.

Fair Selection: Helps pick nodes fairly in blockchain networks.

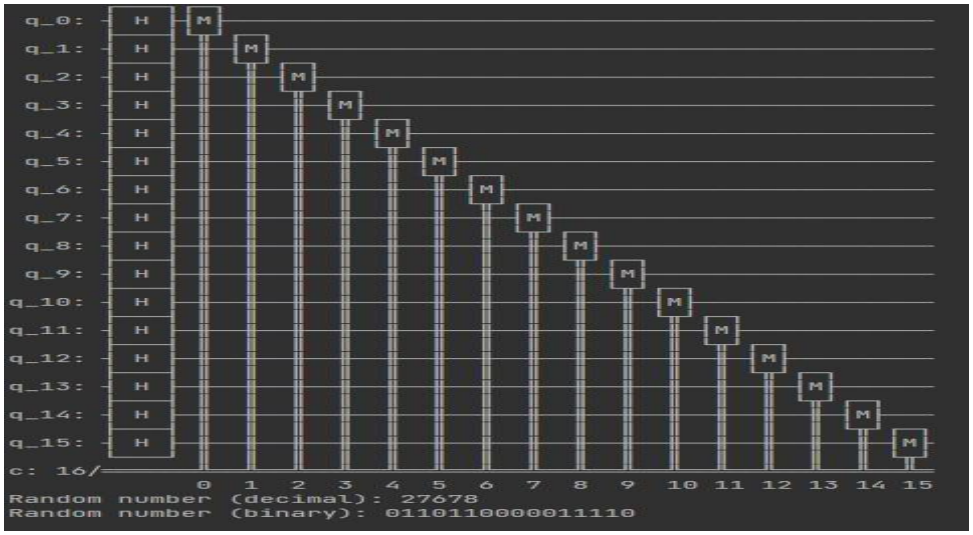
Smarter AI Learning: Adds randomness to help AI learn better.

Cool Creative Stuff: Creates unique art and surprises for entertainment.

Steps:

- Step 1: Initialize the quantum and classical registers
- Step 2: Create the circuit
- Step 3: Apply a Hadamard gate to all qubits
- Step 4: Measure the qubits

Output:



Conclusion: I have understood how to generate 16 Qubit Random Number Generator.

Lab Assignment No.	02
Title	Tackle Noise with Error Correction
Roll No.	
Class	BE
Date of Completion	
Subject	Computer Laboratory-II: Quantum AI
Assessment Marks	
Assessor's Sign	

ASSIGNMENT No: 02

Title: Tackle Noise with Error Correction

Problem Statement: Tackle Noise with Error Correction

Prerequisite:

Basics of Python

Software Requirements: Jupyter

Hardware Requirements:

PIV, 2GB RAM, 500 GB HDD

Learning Objectives:

Learn to Tackle Noise with Error Correction

Outcomes:

After completion of this assignment students are able to understand how to Tackle Noise with Error Correction

Theory: Tackling noise with error correction is a crucial concept in quantum computing to ensure the reliability and accuracy of quantum computations. Here's a brief description:

In a quantum computer, quantum bits (qubits) are susceptible to errors due to external factors like temperature fluctuations or electromagnetic interference. These errors can disrupt the delicate quantum states required for computation. Error correction involves using additional qubits and quantum operations to detect and rectify errors, maintaining the integrity of quantum information.

Quantum error correction codes encode the logical qubits across multiple physical qubits in a way that allows errors to be detected and corrected without directly measuring the quantum state. This is achieved through carefully designed quantum gates that manipulate the qubits and enable error detection. If an error is detected, the information can be recovered through quantum operations, ensuring the correct outcome of the computation.

Error correction techniques, such as the surface code or the stabilizer codes, help extend the lifespan of quantum information and enable quantum computers to perform complex computations with high accuracy. However, error correction comes at the cost of requiring additional qubits and more intricate quantum operations, which poses a challenge in terms of hardware and computational resources.

A brief description of how error correction helps tackle noise in quantum computing, presented in bullet points:

Error Vulnerability: Quantum computers use delicate quantum states (qubits) that are sensitive to external factors, leading to errors in computations.

Quantum Error Correction: Error correction is a technique to mitigate errors by encoding qubits across multiple physical qubits in a way that allows errors to be detected and corrected.

Error Detection: Quantum error correction codes involve measuring specific properties of the encoded qubits without directly measuring the fragile quantum state.

Tackling noise with error correction in quantum computing offers several key advantages that are pivotal for the reliable and practical implementation of quantum technologies. **Here are its advantages:**

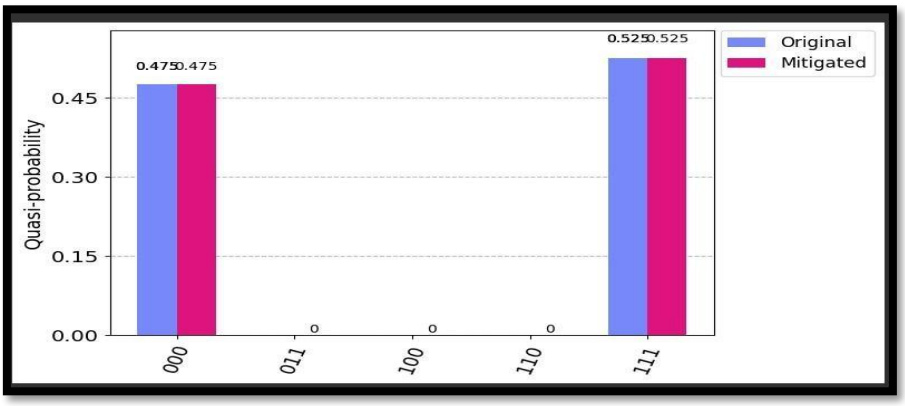
Enhanced Reliability: Error correction enables quantum computations to remain accurate and reliable even in the presence of noisy and error-prone quantum hardware.

Extended Qubit Lifespan: By actively identifying and correcting errors, error correction helps maintain the coherence and stability of quantum states, prolonging the effective lifespan of qubits.

Higher-Quality Results: The use of error correction ensures that the outcomes of quantum computations are closer to the desired results, minimizing the impact of errors on the final output.

Steps-

1. Identify the noisy channel.
2. Choose an error correction technique.
3. Implement error detection and correction.
4. Integrate into your system.
5. Test and optimize.
6. Monitor and maintain.
7. **OUTPUT:**



Conclusion: I have understood how to Tackle Noise with Error Correction

Lab Assignment No.	03
Title	Implement Quantum Teleportation algorithm in Python.
Roll No.	
Class	BE
Date of Completion	
Subject	Computer Laboratory-II: : Quantum AI
Assessment Marks	
Assessor's Sign	

ASSIGNMENT No: 03

Title: Implement Quantum Teleportation algorithm in Python.

Problem Statement: Implement Quantum Teleportation algorithm in Python.

Prerequisite:

Basics of Python

Software Requirements: Jupyter

Hardware Requirements:

PIV, 2GB RAM, 500 GB HDD

Learning Objectives:

Learn to Implement Quantum Teleportation algorithm in Python.

Outcomes:

After completion of this assignment students are able to understand Implement Quantum Teleportation algorithm in Python.

Theory: Quantum teleportation is a fundamental concept in quantum mechanics that allows the transfer of quantum information from one location to another without the physical transfer of particles. It's important to note that this process doesn't involve "teleporting" matter in the way it's often depicted in science fiction; rather, it's a transfer of quantum states between particles. Here's a detailed explanation of quantum teleportation:

Entanglement and Quantum States:

Entanglement is a phenomenon in quantum mechanics where two or more particles become correlated in such a way that the state of one particle is dependent on the state of another, even when they are separated by large distances.

Quantum states, such as the spin or polarization of particles, can be in superposition, meaning they exist in acombination of multiple states simultaneously.

Principle of Quantum Teleportation:

Quantum teleportation involves transferring the complete quantum state of one particle (the "sender" or "Alice's" qubit) to another distant particle (the "receiver" or "Bob's" qubit) through entanglement and classical communication.

The sender and receiver particles are entangled beforehand, usually using a process like the Bell state measurement.

Teleportation Process:

Assume Alice has a qubit in an unknown state she wants to teleport to Bob.

Alice and Bob share an entangled pair of qubits. This shared entanglement serves as the "quantum channel" for teleportation.

Alice performs a joint measurement (Bell measurement) on her qubit and the qubit she wants to teleport. This measurement collapses both qubits into one of four Bell states.

Classical Communication:

Alice sends the result of her Bell measurement to Bob using classical communication. This result consists of two classical bits of information.

Conditional Operations by Bob:

Based on the information received from Alice, Bob applies specific quantum gates to his qubit to transform it into the desired state.

Bob's qubit now holds the quantum state that was initiallyon Alice's qubit. The state has effectively "teleported" from Alice to Bob.

Properties and Implications:

Quantum teleportation ensures the transfer of the exact quantum state, including its superposition and entanglement properties.

It's important to note that the process involves destroying the original state on Alice's qubit. The no-cloning theorem of quantum mechanics prevents exact copying of an arbitrary quantum state.

Applications:

Quantum teleportation is a crucial building block for various quantum communication and quantum networking protocols.

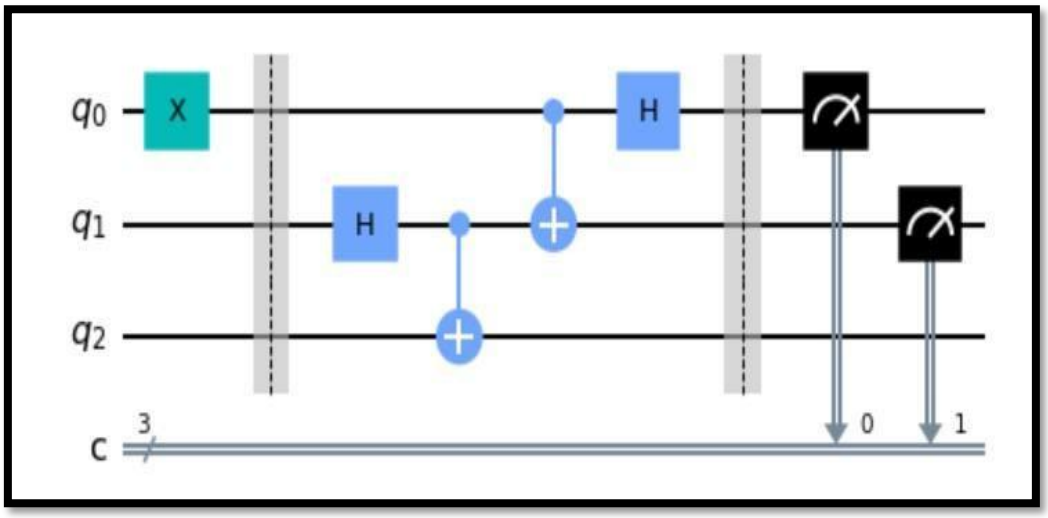
It's used in quantum cryptography for secure key distribution, quantum teleportation-based quantum repeaters for long-distance quantum communication, and potentially in future quantum computing architectures.

Quantum teleportation showcases the non-intuitive and unique aspects of quantum mechanics, demonstrating the entanglement and superposition properties that distinguish quantum systems from classical ones.

Steps-

1. Import the necessary libraries (Quantum Circuit, Aer, execute, plot_bloch_multivector, and plot_histogram).
2. Create a quantum circuit with three qubits and three classical bits.
3. Prepare the initial state by applying gates to create entanglement.
4. Perform the teleportation protocol by applying gates and measurements.

Output:



Conclusion: I have understood the implementation of Quantum Teleportation algorithm in Python.

Lab Assignment No.	04
Title	The Randomized Benchmarking Protocol
Roll No.	
Class	BE
Date of Completion	
Subject	Computer Laboratory-II: : Quantum AI
Assessment Marks	
Assessor's Sign	

ASSIGNMENT No: 04

Title: The Randomized Benchmarking Protocol.

Problem Statement: The Randomized Benchmarking Protocol

Prerequisite:

Basics of Python

Software Requirements: Jupyter

Hardware Requirements:

PIV, 2GB RAM, 500 GB HDD

Learning Objectives:

Learn to Implement the Randomized Benchmarking Protocol

Outcomes:

After completion of this assignment students are able to understand the Randomized Benchmarking Protocol

Theory: Randomized Benchmarking Protocol (RB) is a technique used in quantum information science to assess the quality of quantum gates and operations in a quantum computing system. Quantum gates are the fundamental building blocks of quantum circuits, and their accurate implementation is crucial for the reliable execution of quantum algorithms. RB provides a standardized way to quantify the error rates of these gates and to characterize the overall performance of a quantum processor.

Basic Idea: RB involves applying a sequence of random gate operations (also known as "cliffords") to a quantum state, followed by an inverse sequence of gates to return the state to its original form. The error accumulation during this process provides insight into the overall gate quality.

Randomized Operations (Clifford Gates):

RB employs sequences of random Clifford gates, which are a well-defined set of quantum gates with known mathematical properties. Clifford gates are chosen because they are easily implementable and form a universal set, meaning they can be combined to approximate any quantum operation.

Procedure:

The RB procedure can be broken down into the following steps:

Choose a set of Clifford gates to use in the protocol.

Create a random sequence of Clifford gates, also known as a "randomized sequence."

Apply the randomized sequence to a specific initial quantum state, often the logical zero state (|0⟩).

Apply the inverse of the randomized sequence to the resulting quantum state to attempt to return it to the initial state.

Advantages:

RB has several advantages, including:

It is relatively robust to certain types of errors, making it suitable for characterizing gate quality in noisy quantum systems.

It provides a standardized metric for comparing the performance of gates across different quantum computing platforms.

It is a practical tool for identifying areas of improvement in gate operations and guiding error correction strategies.

12. Limitations:

RB has some limitations:

It assumes that errors are largely independent and do not exhibit strong correlations over the gate sequence.

It only provides information about the average gate fidelity and doesn't capture the entire error spectrum.

13. Applications:

RB is widely used by researchers, engineers, and practitioners in the quantum computing field for:

Benchmarking and validating quantum hardware by quantifying gate performance.

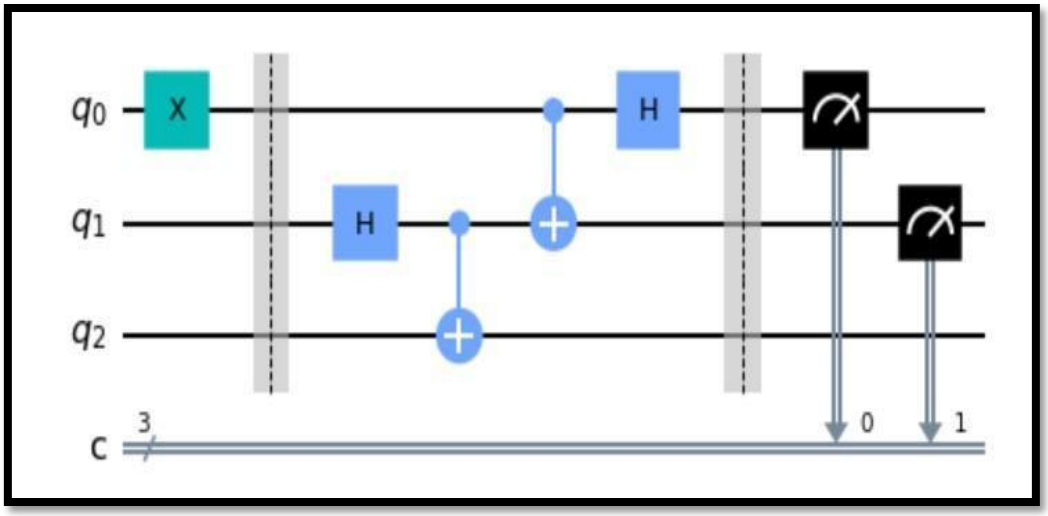
Identifying areas for gate operation enhancement and optimization.

Guiding error mitigation and error correction efforts.

In summary, the Randomized Benchmarking Protocol is a powerful technique for assessing the quality of quantum gates by subjecting them to randomized gate sequences and analyzing the decay in fidelity. It offers insights into the average error rates of quantum operations and plays a crucial role in characterizing and improving the performance of quantum computing systems.

- Steps-
- Import the necessary libraries (qiskit, numpy, matplotlib, etc.).
- Define a benchmarking sequence of random Clifford gates.
- Create benchmarking circuits by applying the sequence to qubits.
- Execute the circuits on a quantum simulator or device.
- Analyze the measurement data to calculate fidelity and error rates.
- Visualize the results by plotting the fidelity decay curve.

Output:



Conclusion: I have understood working of the Randomized Benchmarking Protocol.

Lab Assignment No.	05
Title	Implementing a 5 qubit Quantum Fourier Transform
Roll No.	
Class	BE
Date of Completion	
Subject	Computer Laboratory-II: : Quantum AI
Assessment Marks	
Assessor's Sign	

ASSIGNMENT No: 05

Title: Implementing a 5 qubit Quantum Fourier Transform

Problem Statement: Implementing a 5 qubit Quantum Fourier Transform

Prerequisite:

Basics of Python

Software Requirements: Jupyter

Hardware Requirements:

PIV, 2GB RAM, 500 GB HDD

Learning Objectives:

Learn to Implement 5 qubit Quantum Fourier Transform

Outcomes:

After completion of this assignment students are able to understand the Randomized Benchmarking Protocol

Theory: The Quantum Fourier Transform (QFT) is a fundamental operation in quantum computing that plays a crucial role in many quantum algorithms, including Shor's algorithm for integer factorization and quantum phase estimation. The QFT essentially performs a discrete Fourier transform on the amplitudes of a quantum state, leading to a change in the basis of the state.

Steps of performing a 5-qubit Quantum Fourier Transform:

1. Initial State:

Start with a 5-qubit quantum register initialized in the computational basis state $|0\rangle$. This means that all qubits are in the state $|0\rangle$.

$|00000\rangle$

2. Apply Hadamard Gates:

Apply a Hadamard gate to each qubit. The Hadamard gate creates a superposition of the basis states.

$(H \otimes H \otimes H \otimes H \otimes H) |00000\rangle = (1/\sqrt{32}) \sum x=0^{31} |x\rangle$

Here, $\sum x=0^{31}$ indicates a sum over all possible 5-bit binary numbers.

3. Apply Controlled Rotations:

Apply controlled-phase rotations to create the quantum interference necessary for the Fourier transformation. This involves applying controlled Rk gates, where k is determined by the qubit index and rotation angle. For a 5-qubit QFT, the rotations are defined as follows:

C-R1: Rotate qubit 1 by $\pi/2$

C-R2: Rotate qubit 2 by $\pi/4$

C-R3: Rotate qubit 3 by $\pi/8$

C-R4: Rotate qubit 4 by $\pi/16$

4. Swap Qubits:

Perform a series of controlled-swap operations (also known as swap gates) to rearrange the qubits. Swapping qubits is necessary to achieve the desired QFT ordering of amplitudes.

5. Measurement:

Perform measurements on each qubit to collapse the quantum state into classical bit strings. The measurement outcomes provide the results of the Quantum Fourier Transform in terms of the probabilities of different bit strings.

The final measurement outcomes will represent the amplitudes in the new basis, which corresponds to the Fourier transformed values of the input state. The probability distribution of the measurement outcomes should ideally match the QFT of the original state.

It's important to note that implementing the Quantum Fourier Transform on real quantum hardware can be challenging due to the sensitivity of quantum states to errors and noise. As a result, practical implementations may require error correction techniques and careful calibration of quantum gates.

The 5-qubit Quantum Fourier Transform serves as a building block for more complex quantum algorithms and demonstrates the power of quantum computing in efficiently performing certain mathematical operations that are classically hard to compute.

- **Steps**
- Import the necessary libraries: Import the required libraries for quantum circuit creation and execution
- Create a quantum circuit with 5 qubits.
- Apply Hadamard gates to each qubit.
- Apply controlled phase shift gates to implement the QFT.
- Measure the qubits to obtain the final result.

Key Properties:

1. The Quantum Fourier Transform takes advantage of quantum parallelism to perform computations on multiple states simultaneously.
2. The QFT is a critical component of Shor's algorithm for integer factorization, where it enables efficient period finding.
3. The QFT is used in quantum phase estimation to estimate eigenvalues of unitary operators, which is crucial for various quantum algorithms.
4. The QFT plays a role in quantum algorithms for solving problems in number theory, cryptography, and optimization.

Conclusion: I have understood the implementation of a 5 qubit Quantum Fourier Transform.