PH-UY 3801 GUIDED STUDIES IN PHYSICS: Quantum Teleportation using Qiskit

Cesar Lema
Department of Physics
New York University
New York, NY 10003
cl4393@nyu.edu

Advisor
Vladimir Tsifrinovich, vtsifrin@nyu.edu

Spring 2020

Contents

1 Introduction

A qubits is the basic unit of information in a quantum computer. Physically, a qubit is represented by a quantum object. To carry out computations two states of the quantum object are used like states of a switch to represent a qubit. However contrast to classical bits and a fundamental distinction is that the state of a qubit can be in a superposition of its binary states. Its superposition cannot be explicitly output. Instead the information must be observed through measurement and stored onto a classical bit which can then output the result. The benefit of using a qubit over a classic bit comes from the ability to run quantum algorithms on systems of qubits which offer many benefits over classical computation [1].

2 Quantum Teleportation

Quantum teleportation is the protocol for the transmission of the state of a desired quantum system (possibly unknown) to a target quantum system. Entangled qubits are used to facilitate the transmission [2][3].

2.1 Teleportation of a Single Qubit State

Suppose a student wishes to send an arbitrary single qubit state $|\psi\rangle = \alpha|0\rangle + \beta|1\rangle = \sum_i \alpha_i|i\rangle : i = 1, 2$ from a source qubit to another student with a target qubit. To do so they would need to share an entangled Bells state $|\phi\rangle = \frac{1}{\sqrt{2}}[|00\rangle + |11\rangle] = \sum_j \gamma_j|jj\rangle : j = 1, 2$. The sender should have one of the entangled qubits and the receiver should have the other qubit as the target qubit. Then the sender can transmit the state of their source qubit to the target qubit by appropriate transformation of the combined quantum system and using classical communication channels. The steps for the teleportation protocol are given below in terms of the appropriate quantum gate transformations.

1. The combined three qubit state is setup.

$$\begin{aligned} |\psi_a\rangle &= |\psi\rangle \otimes |\phi\rangle \\ &= \sum_i \alpha_i|i\rangle \otimes \sum_i \gamma_j|jj\rangle \\ &= \sum_{ij} \nu_{ij}|ijj\rangle \end{aligned}$$

   The intermediate $|\psi_a\rangle$ state is represented by a vector in the 8-dimensional three qubit vector space in standard notation. The first and second qubit (from left to right) are held by the sender and the third qubit is held by a receiver. The coordinate vector of $|\psi_a\rangle$ for an arbitrary source qubit is

$$|\psi_a\rangle = \begin{bmatrix} \frac{\alpha}{\sqrt{2}} \\ 0 \\ 0 \\ \frac{\alpha}{\sqrt{2}} \\ \frac{\beta}{\sqrt{2}} \\ 0 \\ 0 \\ \frac{\beta}{\sqrt{2}} \end{bmatrix}$$

2. A controlled NOT gate **CNOT** is applied to the system with the first and second qubits as the control and target respectively.

$$\begin{aligned}|\psi_b\rangle &= \mathbf{CNOT}_{12}|\psi_a\rangle \\ &= \sum_{ij} \nu_{i(j\oplus i)j}|i\,(j\oplus i)\,j\rangle\end{aligned}$$

This operation in matrix notation is

$$|\psi_b\rangle = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} \frac{\alpha}{\sqrt{2}} \\ 0 \\ 0 \\ \frac{\alpha}{\sqrt{2}} \\ \frac{\beta}{\sqrt{2}} \\ 0 \\ 0 \\ \frac{\beta}{\sqrt{2}} \end{bmatrix}$$

$$= \begin{bmatrix} \frac{\alpha}{\sqrt{2}} \\ 0 \\ 0 \\ \frac{\alpha}{\sqrt{2}} \\ 0 \\ \frac{\beta}{\sqrt{2}} \\ \frac{\beta}{\sqrt{2}} \\ 0 \end{bmatrix}$$

3. A Hadamard gate **H** is applied to the first qubit.

$$\begin{aligned}|\psi_c\rangle &= \mathbf{H}_1|\psi_b\rangle \\ &= \sum_{ijk} \nu_{(i\oplus k)(j\oplus i)j}|\,(i\oplus k)\,(j\oplus i)\,j\rangle\end{aligned}$$

where $k = 1, 2$. This operation in matrix notation is

$$|\psi_c\rangle = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & -1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & -1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & -1 \end{bmatrix} \begin{bmatrix} \frac{\alpha}{\sqrt{2}} \\ 0 \\ 0 \\ \frac{\alpha}{\sqrt{2}} \\ 0 \\ \frac{\beta}{\sqrt{2}} \\ \frac{\beta}{\sqrt{2}} \\ 0 \end{bmatrix}$$

$$= \frac{1}{2} \begin{bmatrix} \alpha \\ \beta \\ \beta \\ \alpha \\ \alpha \\ -\beta \\ -\beta \\ \alpha \end{bmatrix}$$

4. The state of the first and second qubits are measured by the sender.

$$\begin{aligned}|\psi_d\rangle &= \mathbf{M}_1\mathbf{M}_2|\psi_c\rangle \\ &= \sum_{j} \nu_{abj}|j\rangle\end{aligned}$$

where a and b are the results of the measurement on the first and second qubits respectively. The measurements result in classical information and is transmitted by the sender to the receiver. It is used by the receiver to transform the intermediate state of the target qubit they hold so that it recovers the source state. The receivers single qubit operations can easily be derived by analyzing the results of measurement on the intermediate state $|\psi_c\rangle$ and omitting the global phase. They are summarized below.

(a) If $a = 0$ and $b = 0$ is measured the third qubit is already in the desired state so no operation is needed .

$$|\psi'\rangle = |\psi_d\rangle = \begin{bmatrix}\alpha \\ \beta\end{bmatrix}$$

(b) If $a = 0$ and $b = 1$ is measured an **X** gate applied to the intermediate state gives the original state of the source qubit.

$$|\psi'\rangle = \mathbf{X}|\psi_d\rangle = \begin{bmatrix}0 & 1 \\ 1 & 0\end{bmatrix}\begin{bmatrix}\beta \\ \alpha\end{bmatrix} = \begin{bmatrix}\alpha \\ \beta\end{bmatrix}$$

(c) If $a = 1$ and $b = 0$ is measured a **Z** gate applied to the intermediate state gives the original state of the source qubit.

$$|\psi'\rangle = \mathbf{Z}|\psi_d\rangle = \begin{bmatrix}1 & 0 \\ 0 & -1\end{bmatrix}\begin{bmatrix}\alpha \\ -\beta\end{bmatrix} = \begin{bmatrix}\alpha \\ \beta\end{bmatrix}$$

(d) If $a = 1$ and $b = 1$ is measured a **Z** and a **X** gate applied to the intermediate state and omitting the global phase gives the original state of the source qubit.

$$|\psi'\rangle = \mathbf{ZX}|\psi_d\rangle = \begin{bmatrix}1 & 0 \\ 0 & -1\end{bmatrix}\begin{bmatrix}0 & 1 \\ 1 & 0\end{bmatrix}\begin{bmatrix}-\beta \\ \alpha\end{bmatrix} = \begin{bmatrix}\alpha \\ \beta\end{bmatrix}$$

After the teleportation protocol the state of the source qubit is transmitted to the second qubit of the entangled Bell's state system. The source qubit and the first qubit of the entangled Bell's state system stochastically collapsed to one of the basis states.

2.2 Teleportation of a 2 Qubit State

The quantum teleportation protocol of an arbitrary single qubit state can be generalized to teleport an arbitrary N qubit state to a target quantum system [3][2]. The qubits of a source N qubit system are iterated over to teleport the source state. At each iteration the source system is combined with an entangled Bell's state system and the single qubit teleportation protocol is used to teleport the state of the current qubit to a target system. The resulting intermediate states consist of the source qubits that have not been iterated over and the second qubit of a Bell's states. The generalization for an arbitrary two qubit state $|\psi\rangle = \alpha|00\rangle + \beta|01\rangle + \gamma|10\rangle + \delta|11\rangle = \sum_{ij} \alpha_{ij}|ij\rangle$ is given.

An entangled Bells state $|\phi_a\rangle = \frac{1}{\sqrt{2}}[|00\rangle + |11\rangle] = \sum_k \gamma_k|kk\rangle$ is used at the first iteration and the single qubit teleportation protocol follows.

1. The combined four qubit state is setup.

$$\begin{aligned}|\psi_a\rangle &= |\psi\rangle \otimes |\phi_a\rangle \\ &= \sum_{ij} \alpha_{ij}|ij\rangle \otimes \sum_k \gamma_k|kk\rangle \\ &= \sum_{ijk} \nu_{ijk}|ijkk\rangle\end{aligned}$$

2. A controlled NOT gate **CNOT** is applied to the system with the first and third qubits as the control and target respectively.

$$
\begin{aligned}
|\psi_b\rangle &= \mathbf{CNOT}_{13}|\psi_a\rangle \\
&= \sum_{ijk} \nu_{ij(k\oplus i)k}|ij(k\oplus i)k\rangle
\end{aligned}
$$

3. A Hadamard gate **H** is applied to the first qubit.

$$
\begin{aligned}
|\psi_c\rangle &= \mathbf{H}_1|\psi_b\rangle \\
&= \sum_{ijkw} \nu_{(i\oplus w)j(k\oplus i)k}|(i\oplus w)j(k\oplus i)k\rangle
\end{aligned}
$$

   where $w = 1, 2$.

4. The state of the first and third qubits are measured by the sender.

$$
\begin{aligned}
|\psi_d\rangle &= \mathbf{M}_1\mathbf{M}_3|\psi_c\rangle \\
&= \sum_{jk} \nu_{ajbk}|jk\rangle
\end{aligned}
$$

   where a and b are the measured states of the first and third qubit respectively. The states observed indicate which operation should be applied to get the true intermediate state. The operations in this generalization are the same single qubit operations.

   (a) if $a = 0$ and $b = 0$ is measured then $|\psi_d\rangle$ is the desired result.
   (b) if $a = 0$ and $b = 1$ is measured then $\mathbf{X}_4|\psi_d\rangle$ gives the desired result.
   (c) if $a = 1$ and $b = 0$ is measured then $\mathbf{Z}_4|\psi_d\rangle$ gives the desired result.
   (d) if $a = 1$ and $b = 1$ is measured then $\mathbf{Z}_4\mathbf{X}_4|\psi_d\rangle$ gives the desired result.

After the first iteration the state of the first qubit in the source system is teleported to the target quantum system.

Similarly to teleport the state of the second qubit in the source system to the target quantum system an entangled Bells state $|\phi_b\rangle = \frac{1}{\sqrt{2}}[|00\rangle + |11\rangle] = \sum_l \alpha_l|ll\rangle$ is used. The qubits of this bells state are labeled as the fifth and sixth qubit of the protocol. After completing the measurement step of the single qubit protocol for the second qubit the state of the two qubits in the target system is

$$
|\psi_f\rangle = \sum_{kv} \alpha_{acbkdl}|kl\rangle
$$

where c and d are the measured states of the second and fifth qubits respectively. The original state of the source system is recovered by the appropriate gate operations on the sixth qubit (the second qubit of the target system).

1. if $a = 0$ and $b = 0$ is measured then $|\psi_f\rangle$ is the desired result.
2. if $a = 0$ and $b = 1$ is measured then $\mathbf{X}_6|\psi_f\rangle$ gives the desired result.
3. if $a = 1$ and $b = 0$ is measured then $\mathbf{Z}_6|\psi_f\rangle$ gives the desired result.
4. if $a = 1$ and $b = 1$ is measured then $\mathbf{Z}_6\mathbf{X}_6|\psi_f\rangle$ gives the desired result.

After the teleportation protocol for the N qubit state the state of the source quantum system is transmitted to the target quantum system. The qubits of the source system and the first qubit of the entangled Bell's states stochastically collapsed to one of their basis states. Note that the receiver can transform the individual states of the qubits in the target system after all the iterations have been done since the iterations only changes the current source and entangled qubit of the sender.

3 Qiskit

Qiskit was founded by IBM Research to allow software development for their cloud quantum computing service. It is an open source software development kit (SDK) for working with OpenQASM and the IBM Q quantum processors. Qiskit is composed of four foundational elements (python modules) which are developed for different aspects of quantum computing software. Additionally the Qiskit IBMQ module allows users to interact with IBM Quantum devices and simulators.

3.1 Qiskit Implementation of Two Qubit Teleportation

The teleportation protocol for a quantum system in an arbitrary N qubit states is implemented for systems of two qubits using the Qiskit SDK. Reference [4] gives general instruction for working with Qiskit.

3.1.1 Initialization

Qiskit is used to initialize the quantum systems needed for the teleportation protocol. For this implementation the initial state of the quantum system is a determinate state of the two qubit vector space. The decimal notation of the determinate states is used i.e. the number that represents a determinate state $|ab\rangle$ is the decimal represented by the binary number ab. This makes it easy to validate the outcome of the teleportation protocol. The classes that are needed include

- **QuantumRegister**: A generic quantum register.
- **QuantumCircuit**: A circuit is a list of instructions bound to some registers.

For the initial state of the source quantum system a **QuantumRegister** instance is used to reference two bits and a **QuantumCircuit** instance is initialized using this register. Given a number m for the desired initial state the n qubits in the circuit are iterated over and an **X** gate is applied if the corresponding symbol in the binary representation of m is nonzero. For this implentation $n = 2$ and $m = 3$. The code for general n and m values and the resulting circuit for this implementation are shown below.

```
qRegister = QuantumRegister(n,name = "bit%i_single%iState_q"%(n,m))
circuit = QuantumCircuit(qRegister)
binary = format(m,"b")
for i in range(n - len(binary) ):
    binary = "0"+binary

for qindex in range(n):
    if int(binary[qindex]):
        circuit.x([qindex])
```

Figure 1: Initialization of circuit to transform n qubits to a given determinate state m of the n qubit vector space.

The first for loop adds zeros (padding) to the left of the binary representation of m. The padding does not change the value of m and helps the second for loop work properly.

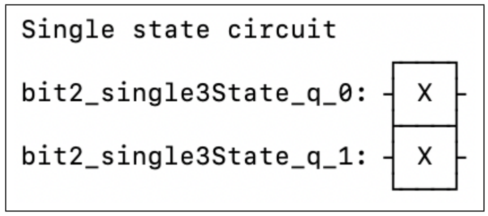

Figure 2: Quantum circuit transforming a two qubits system to the determinate state 3.

Similary to initialize a system in a Bells states a **QuantumRegister** instance is used to reference two bits and a **QuantumCircuit** instance is initialized using the register. The entangled state is achieved by applying a Hadamard gate **H** and CNOT gate **CX** to the qubits in the circuit. The gate operations are implemented as methods of the **QuantumCircuit** class and take the index of a qubit in the circuit as its argument. The code and resulting circuit are shown below.

```
qRegister = QuantumRegister(2,name = "bellsState%s_q"%(ind))
circuit = QuantumCircuit(qRegister)
circuit.h([0])
circuit.cx([0],[1])
```

Figure 3: Initialization of circuit to transform a two qubits system to an entangled Bells state

The string variable ind is a label for the bells state.

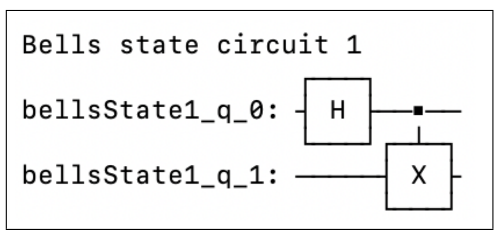

Figure 4: Quantum circuit transforming a two qubits system to an entangled Bells state.

Custom functions **singleState()** and **BellsState()** are defined to encapsulate the initialization of the quantum registers and circuits for the initial source state and the Bells state. For the initialization of the quantum system for the protocol the initial and Bells state circuits are required. They are stored in the variables **initStateCircuit**, **bState1Circuit**, **bState2Circuit** respectively. The **combine()** method of the **QuantumCircuit** class is used to combine the three quantum systems and circuits. The functions and methods used are summarized below

- **singleState()**: Given a number of qubits n and appropriate integer m returns initialized **QuantumRegister** and **QuantumCircuit** instances for initializing a n qubit system to a determinate state m.
- **BellsState()**: Given a label returns initialized **QuantumRegister** and **QuantumCircuit** instances for constructing a system in an entngled Bells state.
- **combine()**: Method of the **QuantumCircuit** class that appends an input **QuantumCircuit** instance to the caller.

The code for the initialization of the quantum system and the corresponding quantum circuit are shown below.

```
initStateQbits, initStateCircuit = singleState(2,3,provider)
bStateQbits1, bState1Circuit = BellsState(ind = "1")
bStateQbits2, bState2Circuit = BellsState(ind = "2")

currCir = initStateCircuit
currCir = initStateCircuit.combine(bState1Circuit)
currCir = currCir.combine(bState2Circuit)
```

Figure 5: Initialization of the combined circuit of the initial state and Bells state circuits

Note, a 6 qubit circuit can be initialized with the gates applied respectively to achieve the same result.

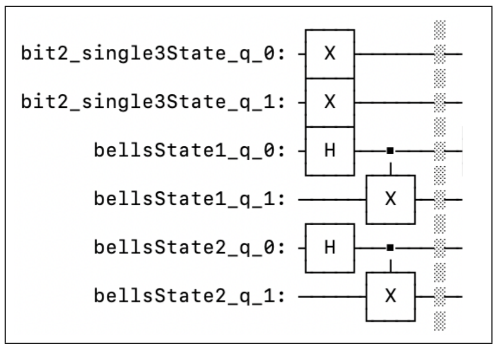

Figure 6: Quantum circuit for the combined initial source state and bells state

3.1.2 Teleportation Circuit

Following the two qubit teleportation protocol outlined in section 2.2 the qubits of the initial state are iterated over and the teleportation protocol for a single qubit state is applied. First six bits on classical registers are added to the initial quantum circuits. These will hold the measured values of the qubits in the protocol. The class and method used are

- **ClassicalRegister**: A generic classical register.
- **add_register()**: Method of the **QuantumCircuit** class that appends registers to the instance.

The code for adding the classical register to the initialized quantum circuit is shown below.

```
cBit1 = ClassicalRegister(1,name = "c1_iter1")
cBit2 = ClassicalRegister(1,name = "c2_iter1")
cBit3 = ClassicalRegister(1,name = "c1_iter2")
cBit4 = ClassicalRegister(1,name = "c2_iter2")
resultCB = ClassicalRegister(2, name = "results")

currCir.add_register(cBit1,cBit2,cBit3,cBit4)
currCir.add_register(resultCB)
```

Figure 7: Adding classical registers to the initialized quantum circuit.

The protocol iterations are applied to the initialized quantum circuit. The quantum gates required are implemented by Qiskit as methods of the **QuantumCircuit** class. The methods used are

- **barrier()**: An instruction to separate pieces of a circuit
- **cx()**: Apply a CNOT gate to the qubits in its argument.
- **h()**: Apply a Hadamard gate to the qubit in its argument.
- **z()**: Apply a Pauli Z gate to the qubit in its argument.
- **x()**: Apply a Pauli X gate to the qubit in its argument.
- **c_if()**: Adds a condition on a classical register for an operation.
- **measure()**: Measure quantum bit(s) into classical bit(s)

The code for the two iteration of the protocol and the corresponding quantum circuit are shown below.

```
# First Iteration
currCir.barrier()
currCir.cx([0],[2])
currCir.h([0])

currCir.measure([0],[1])
currCir.measure([2],[0])

currCir.barrier()
currCir.z([3]).c_if(cBit2, 1)
currCir.x([3]).c_if(cBit1, 1)

# Second Iteration
currCir.barrier()
currCir.cx([1],[4])
currCir.h([1])

currCir.measure([1],[3])
currCir.measure([4],[2])

currCir.barrier()
currCir.z([5]).c_if(cBit4, 1)
currCir.x([5]).c_if(cBit3, 1)
```

Figure 8: Two iterations of the two qubit teleportation protocol.

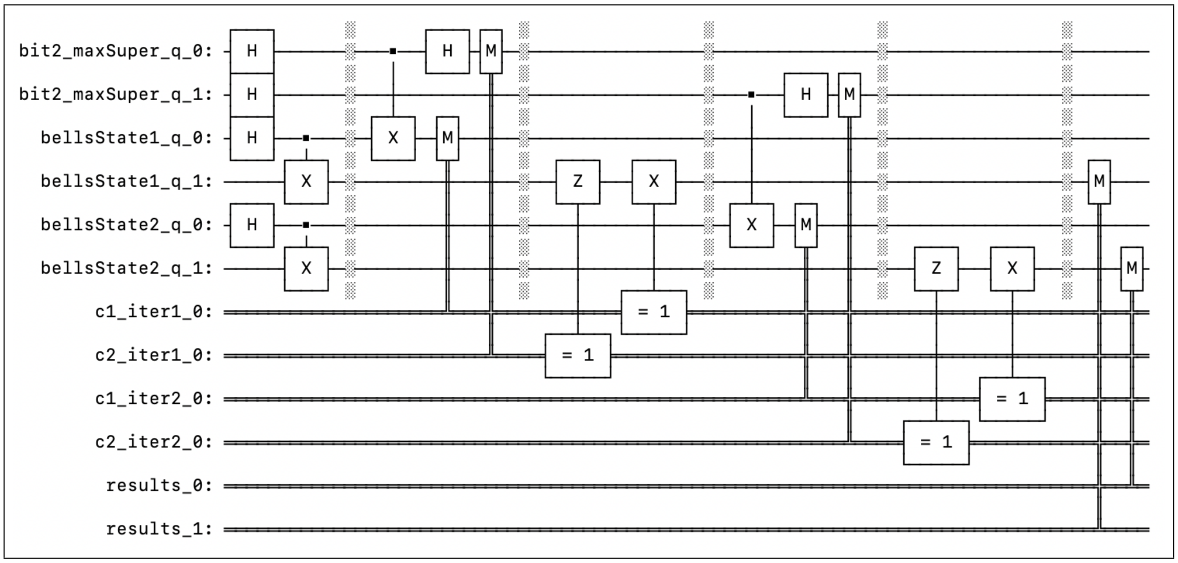

Figure 9: Quantum circuit that executes 2 qubit teleportation of the initial state of a two qubit system

3.1.3 Execution and Results

A custom function **executeAndPlot()** is defined to encapsulate the execution and output of a quantum circuit on a specified backend device. It takes in an instance of the **QuantumCircuit** class and executes it using the Qiskit function **execute()** on a given **backend**. The **backend** for this implementation was the Qiskit Aer quantum simulator labeled "qasm_simulator" but other backends including IBM Quantum devices can be used. IBM quantum computers do not currently support instruction after measurement so they cannot execute this implementation of the two qubit teleportation circuit. However the quantum simulator is capable of executing the circuit and sufficient. The functions and class used are

- **execute()**: Execute a list of qiskit.circuit.QuantumCircuit or qiskit.pulse.Schedule on a backend.
- **job_monitor()**: Monitor the status of a IBMQJob instance.
- **plot_histogram()**: Plot a histogram of its data argument.
- **draw()**: A method of the **QuantumCircuit** class that draws the quantum circuit. The argument specifies the output format.
- **backend**: Instance of a backend class responsible for running the quantum circuit.

The custom function is given below

```python
def executeAndPlot(cir, backend):
    print(cir.draw(output="text"))
    job = execute(cir, backend = backend,shots = 1024)
    job_monitor(job)
    results = job.result()
    plot_histogram(results.get_counts(cir))
    plt.show()
    plt.close()
```

Figure 10: Code for execution of quantum circuit on a specified backend.

Teleportation of Digits

The circuit that executes the two qubit teleportation protocol and the results of the measurements on the two target qubits after execution are shown below for the initial source states 0-3. In the probability against state plots the first two symbols (from the bottom) of the state labels represent the measured state of the target qubits. The other symbols represent the measurement of the qubits in the intermediate steps of the teleportation protocol. The probabilities are calculated using the default 1024 shots.

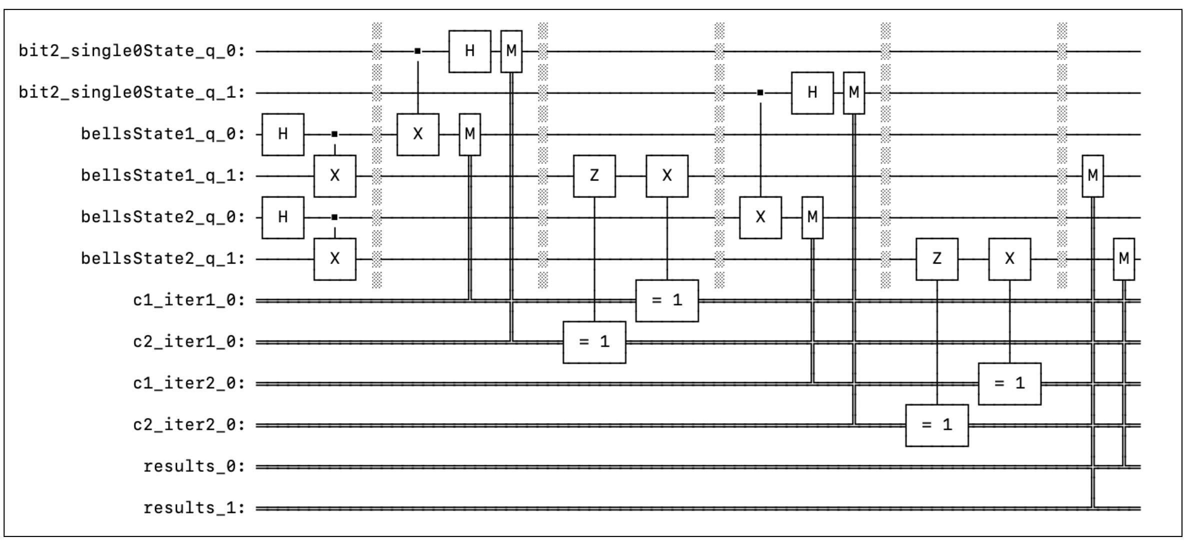

Figure 11: Quantum circuit for 2 qubit teleportation of the determinate 0 state.

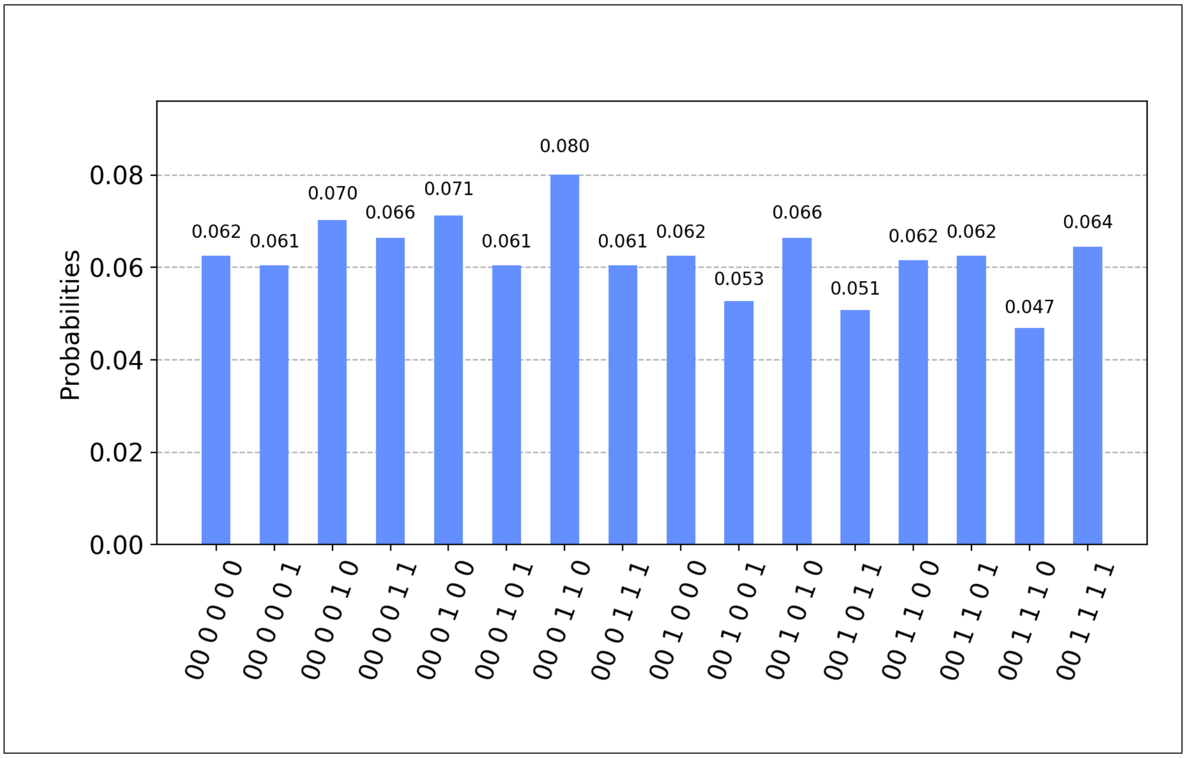

Figure 12: Measurement results of the state of the target qubits after execution of the circuit.

In figure 11. the probabilities sum to 1 for the for the state 0 of the target qubits. This shows that the initial state has indeed been teleported.

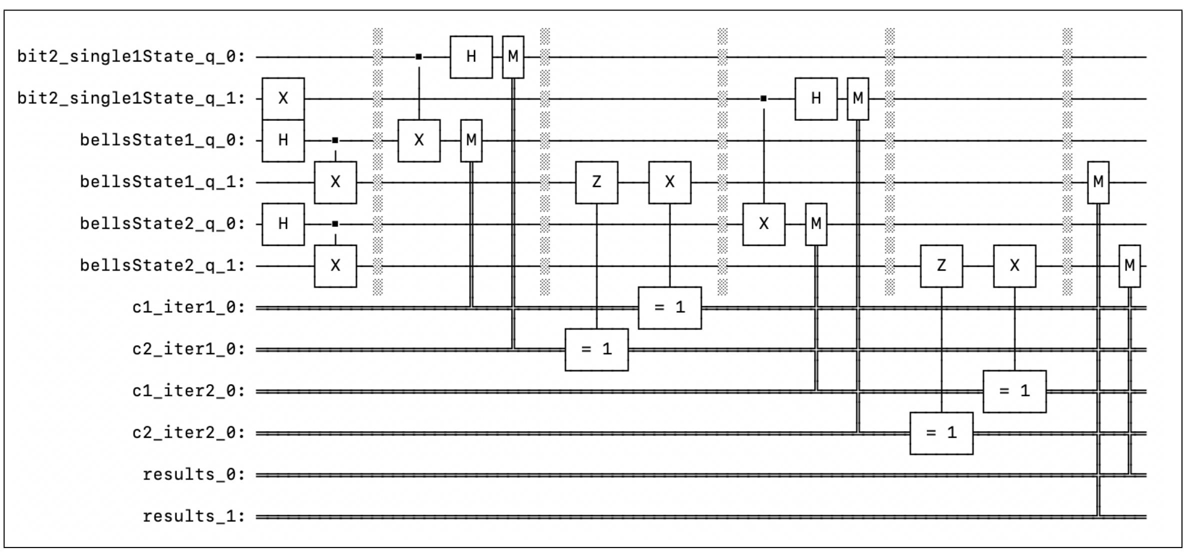

Figure 13: Quantum circuit for 2 qubit teleportation of the determinate 1 state.

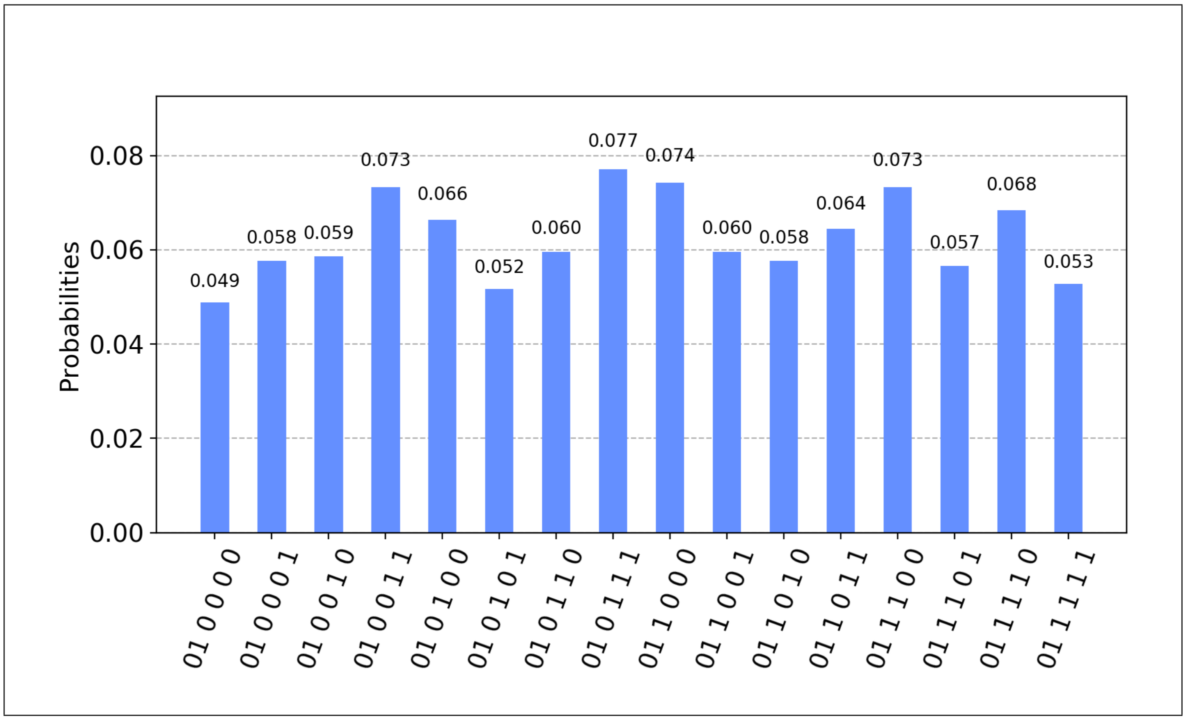

Figure 14: Measurement results of the state of the target qubits after execution of the circuit.

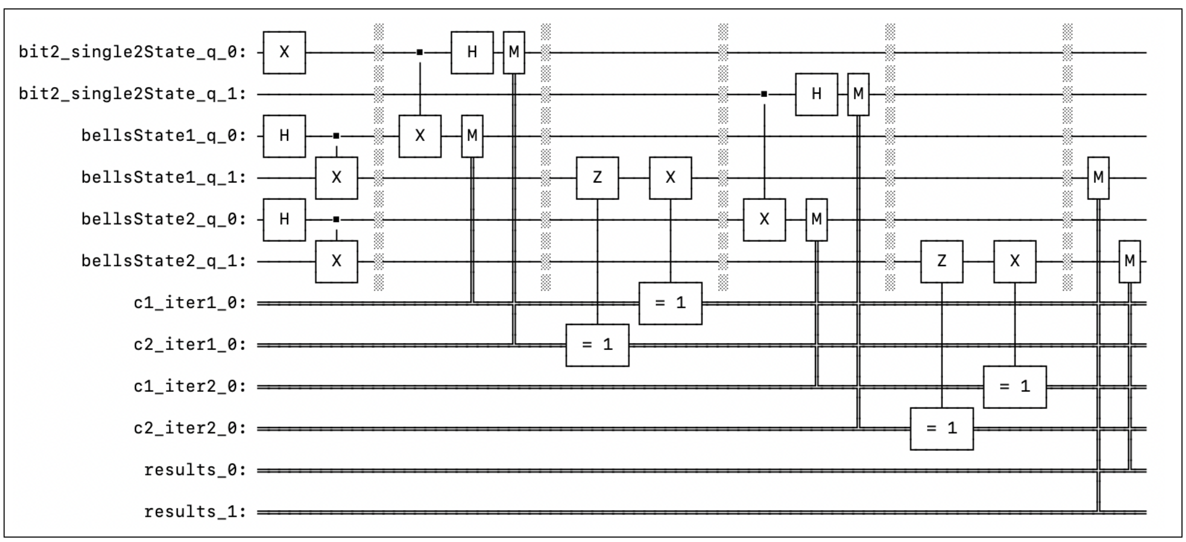

Figure 15: Quantum circuit for 2 qubit teleportation of the determinate 2 state.

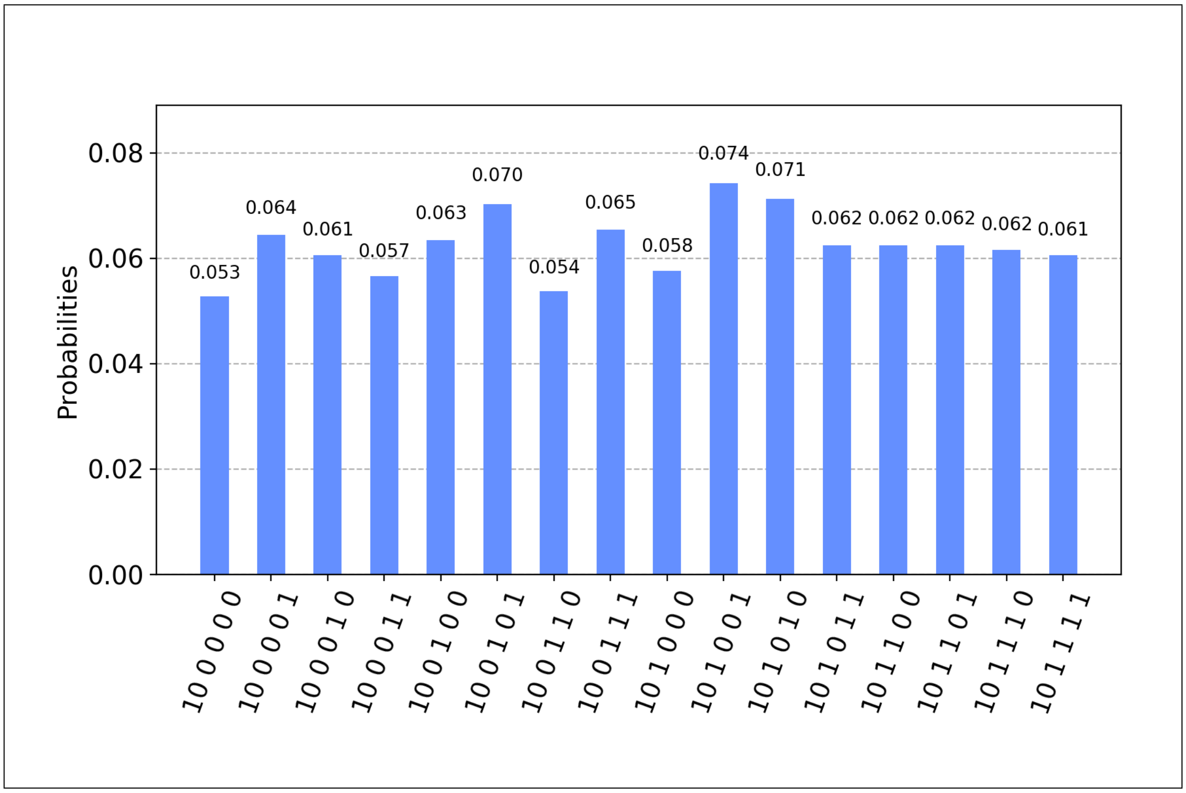

Figure 16: Measurement results of the state of the target qubits after execution of the circuit.

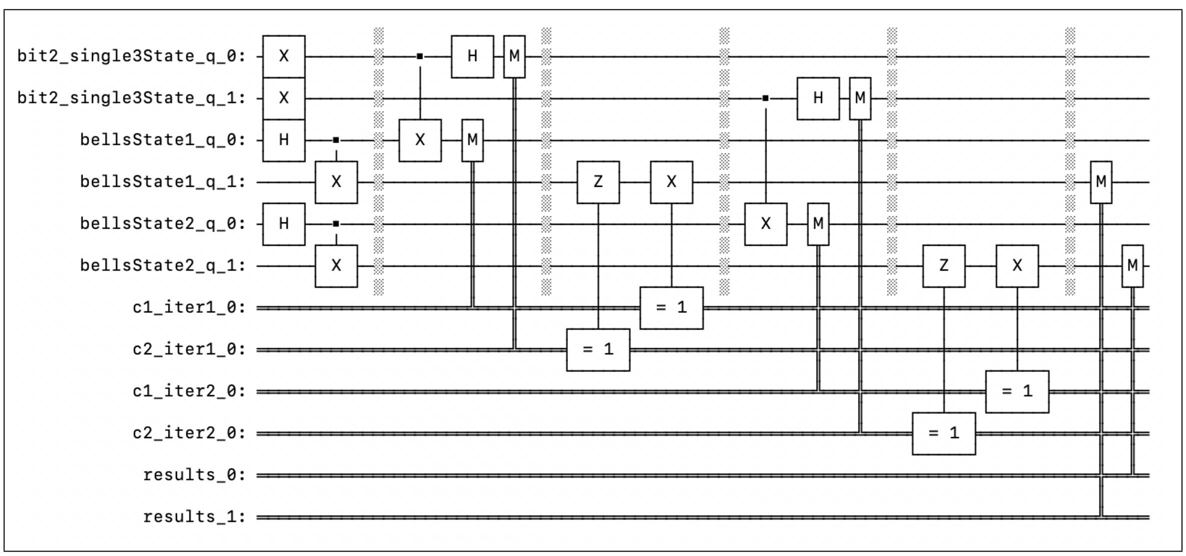

Figure 17: Quantum circuit for 2 qubit teleportation of the determinate 3 state.

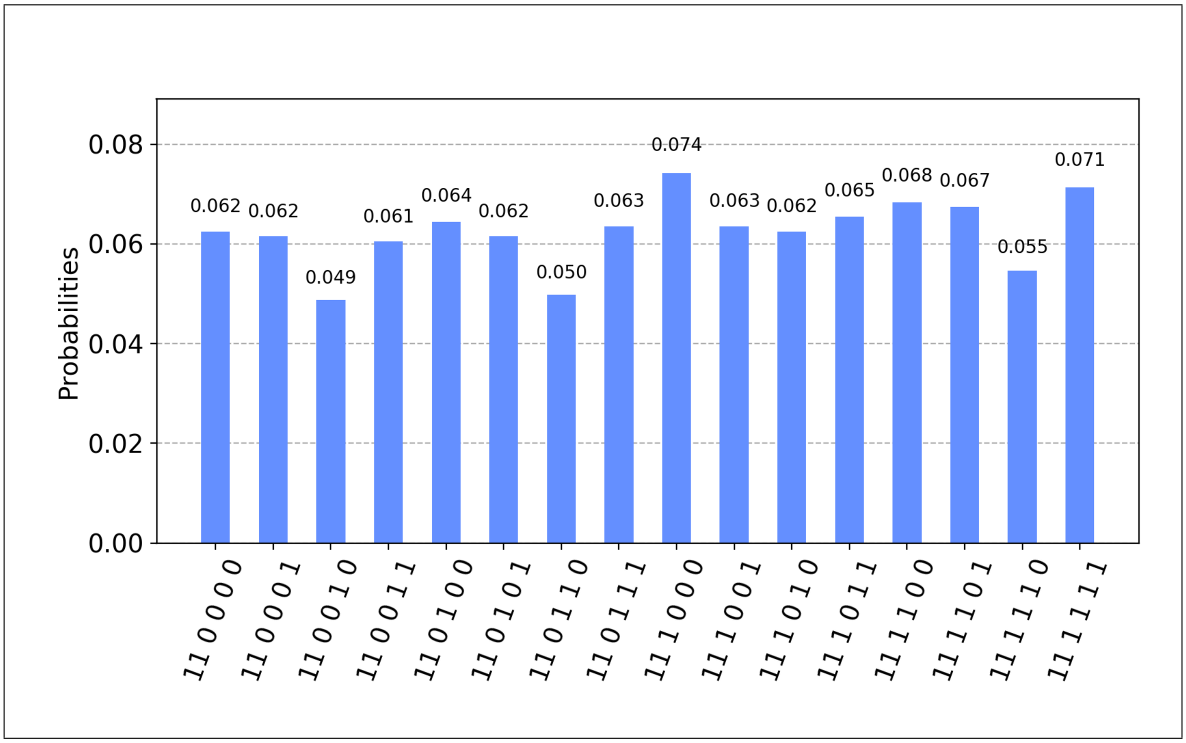

Figure 18: Measurement results of the state of the target qubits after execution of the circuit.

Teleportation of an equal superposition of all two qubit states

The circuit that executes the two qubit teleportation protocol and the results of the measurements on the two target qubits after execution are shown below for an initial source state that is a equal superposition of all the basis states.

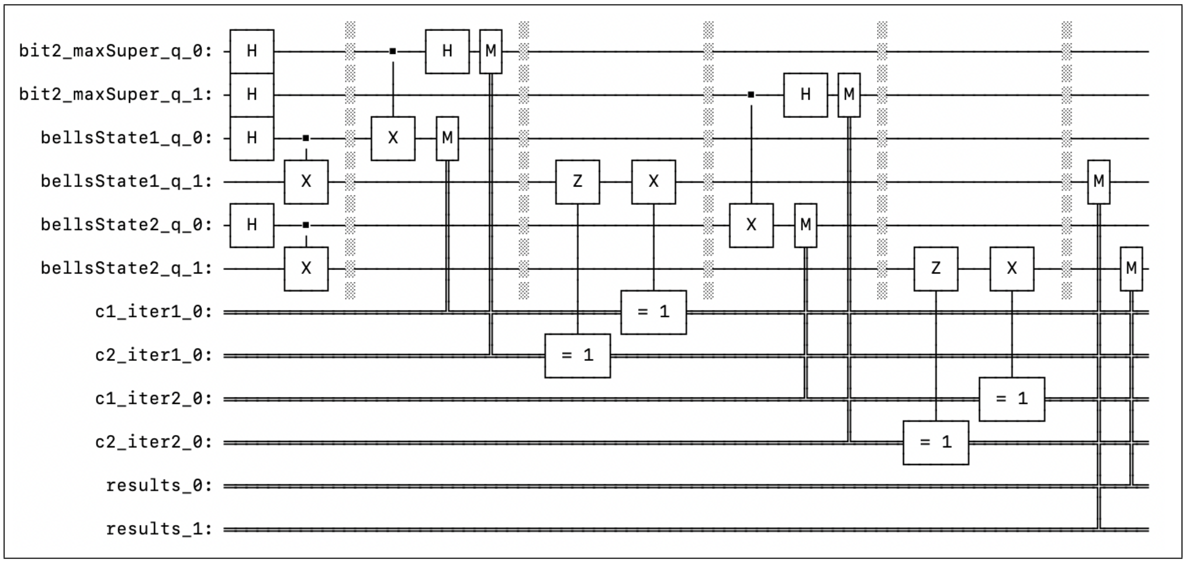

Figure 19: Quantum circuit for the 2 qubit teleportation of a source state in an equal superposition of all 2 qubit states

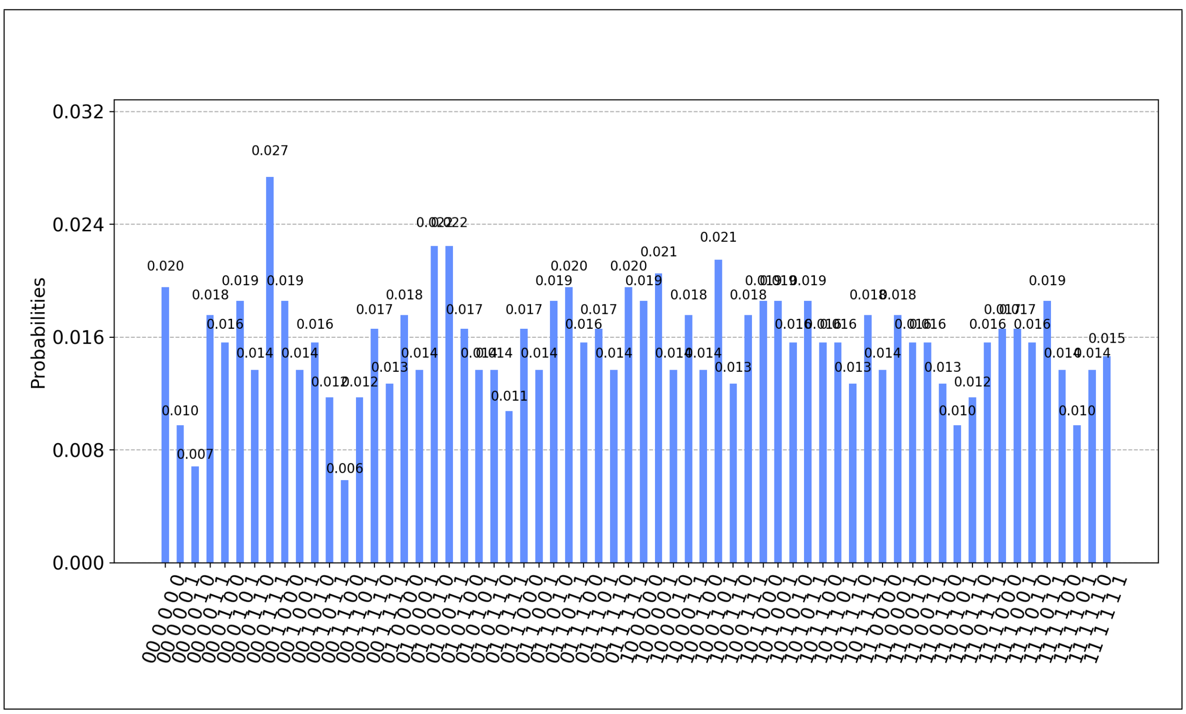

Figure 20: Measurement results of the state of the target qubits after execution of the circuit.

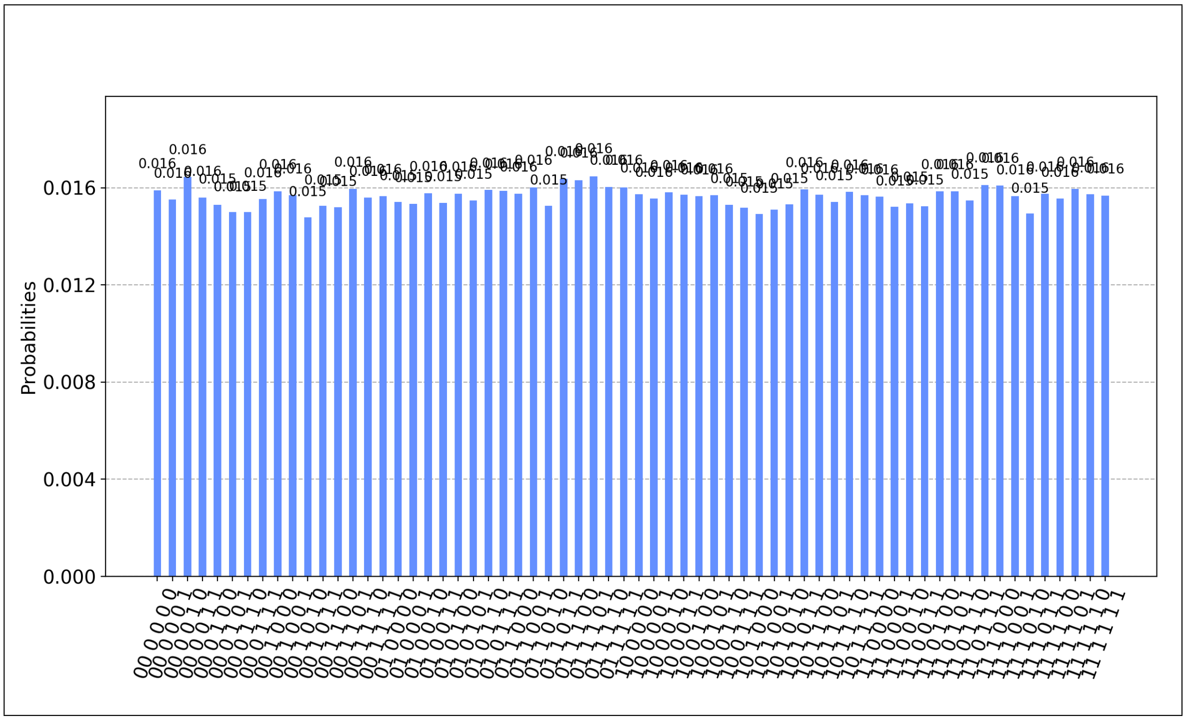

Figure 21: Measurement results of the state of the target qubits after execution of the circuit. The probability are calculated using results from 100000 shots.

The state of the target qubits after the teleportation protocol are in an equal superposition of all the basis states so the initial state of the source system has indeed been teleported. Note, the sum of the probabilities for each state are approximately equal and are not exactly equal due to the probabilistic nature of the classical measurements at the intermediate steps.

References

[1] Gennady P. Berman, Gary D. Doolen, Ronnie Mainieri, Vladimir I Tsifrinovich. *Introduction to Quantum Computers*. World Scientific, 1998.

[2] Charles H. Bennett, Gilles Brassard, Claude Crépeau, Richard Jozsa, Asher Peres, and William K. Wootters. Teleporting an unknown quantum state via dual classical and Einstein-Podolsky-Rosen channels. *Phys. Rev. Lett*, 70(13):678–682, 1993.

[3] Zhang, Q., Goebel, A., Wagenknecht, C. et al. Experimental quantum teleportation of a two-qubit composite system. *Nature Phys*, 2:678–682, 2006.

[4] Daniel Koch, Laura Wessing, Paul M Alsing. Introduction to Coding Quantum Algorithms: A Tutorial Series Using Qiskit. 2019.