QOSF Mentorship Program

1 Task 4

A common challenge in the design and implementation of quantum circuits is that they become too extensive and complex due to the large number of qubits required. The number of quantum operations and the interdependence between qubits can further complicate the problem, making some traditional optimization methods ineffective or insufficient for reducing the complexity. Additionally, as the number of qubits increases, the fidelity of the circuit may be affected, which implies a greater need for advanced techniques to maintain the precision of operations.
On the other hand, some quantum computing frameworks offer predefined optimization methods that may seem like a convenient option. However, they are not always the best alternative if one has a deep understanding of the quantum hardware structure. Customizing the circuits by taking advantage of the specific features of the quantum device in use can result in a more efficient design, better adapted to the physical limitations of the system. Therefore, having a solid understanding of the hardware can make the difference between a generic quantum circuit and one that is highly optimized for performance.
Find a quantum circuit that represents the state vector with a depth less than 50, using the basis gates=[x,h,rz,cx] and the following architecture

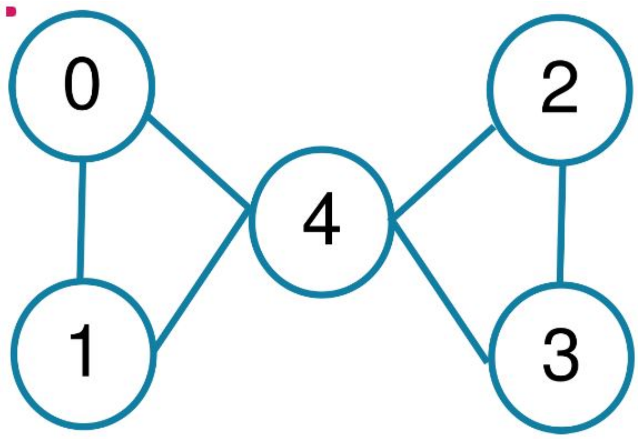

Figure 1: Qubit architecture

2 Solution

The aim of the above task was to represent the following state using a quantum circuit,

$$|\psi\rangle = \frac{1}{2}(|10110\rangle + |10001\rangle + |11011\rangle + |01100\rangle) \tag{1}$$

where the notation used is $|q_4q_3q_2q_1q_0\rangle$ with q_4 being the most significant bit and q_0 being the least significant bit. Below is my thought process for going about designing the quantum circuit.

1. The first aspect I took note of was the number of states in the superposition. Four states with equal amplitudes indicated that I will need two Hadamard gates at the start.

2. Also, upon inspection I realised that q_3, q_2, q_1 and q_0 all have two 1s and two 0s across the four states. And therefore, I can look for two of those to build the state $|\varphi\rangle = \frac{1}{2}(|00\rangle + |01\rangle + |10\rangle + |11\rangle)$. I went with q_3 and q_2.

3. The next step was coming up with a relationship that can connect another qubit with q_3 and q_2. First was, $q_1 = q_3 \oplus q_2$, where $\oplus$ is the XOR operation, which can be implemented using two CNOT gates(q_3 and q_2 are control and q_1 is the target).

4. Next was $q_0 = q_3 \odot q_1$, where $\odot$ is the XNOR operation, which can be implemented using a X gate on q_0, followed by two CNOT gates(q_3 and q_1 are control and q_0 is the target).

5. Lastly, $q_4 = \neg(q_3 \land q_2)$, which is a NAND operation and can be implemented using a X gate on q_4 followed by a Toffoli gate (double control of q_3 and q_2 with q_4 being the target).

Upon implementing the above using IBM's Qiskit software, I obtained the following diagram.

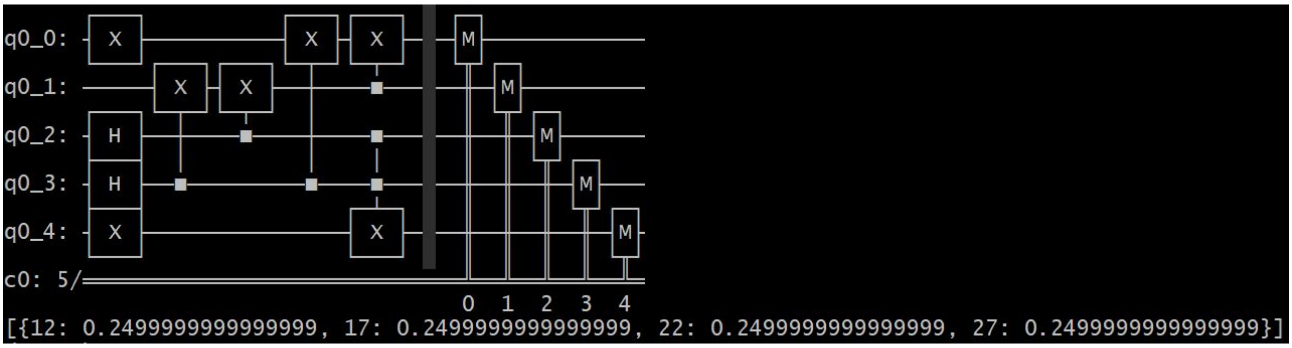

Figure 2: Quantum Circuit assuming all-to-all connectivity and using non-native gates (Depth = 5).

The next task was qubit mapping. Looking at Figure 1, qubit labelled 4 is connected to all the other qubits. Hence, I made a choice to have q_3 interact with all other qubits in my quantum circuit. And since q_4 does not interact with q_1 or q_0, the mapping is as follows: $0 \to q_0$, $1 \to q_1$, $2 \to q_2$, $3 \to q_4$ and $4 \to q_3$. This leaves one pair that interact with each other (q_1 and q_2) but are not next to each other. I used a SWAP gate (implemented using 3 CNOTs) to swap the position of q_2 and q_3. Then I after applying the required CNOT gate, I used another SWAP gate to return them to their original positions. This can be seen in Figure 3.

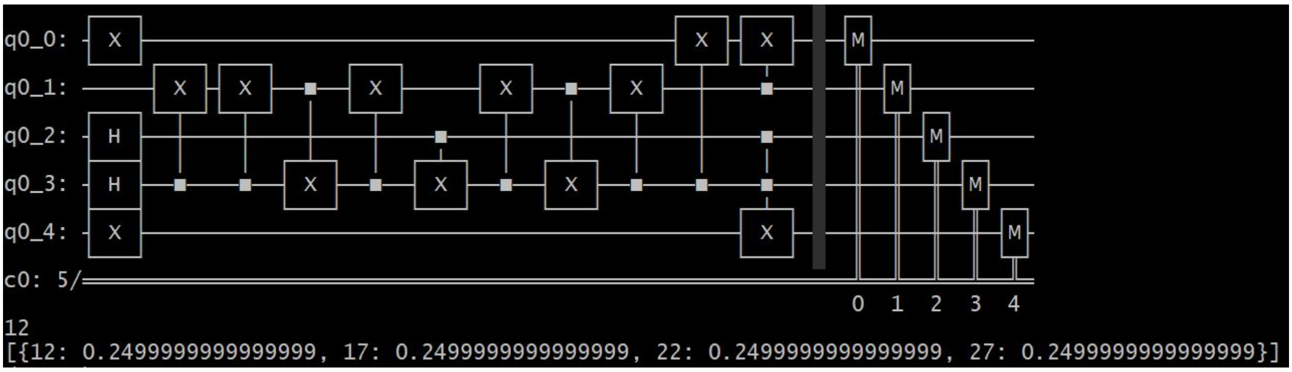

Figure 3: Quantum Circuit with SWAP gates (Depth 10).

One thing to note is that the CNOT(q_3, q_1) and CNOT(q_2, q_1) commute. But since there is a CNOT gate being applied as part of the SWAP gate between q_3 and q_2, we can make use of some CNOT cancellation to reduce the gate depth. This is seen by comparing Figure 3 and Figure 4.

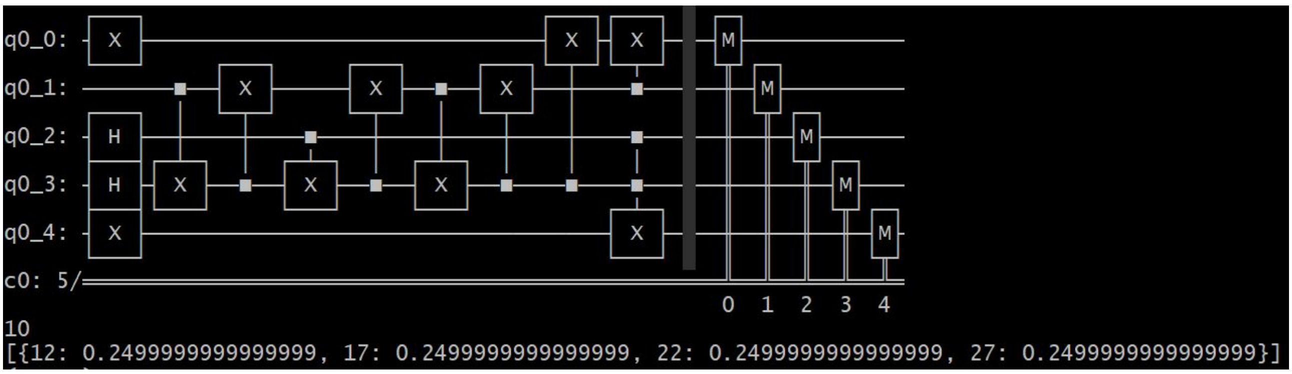

Figure 4: Quantum Circuit with CNOT cancellation (Depth 10).

The last step in the process was implementing the Toffoli gate in terms of the basis gates=[x,h,rz,cx]. There is a standard decomposition that can be found in a paper by Shende and Markov [SM08]. The final quantum circuit is presented in Figure 5.

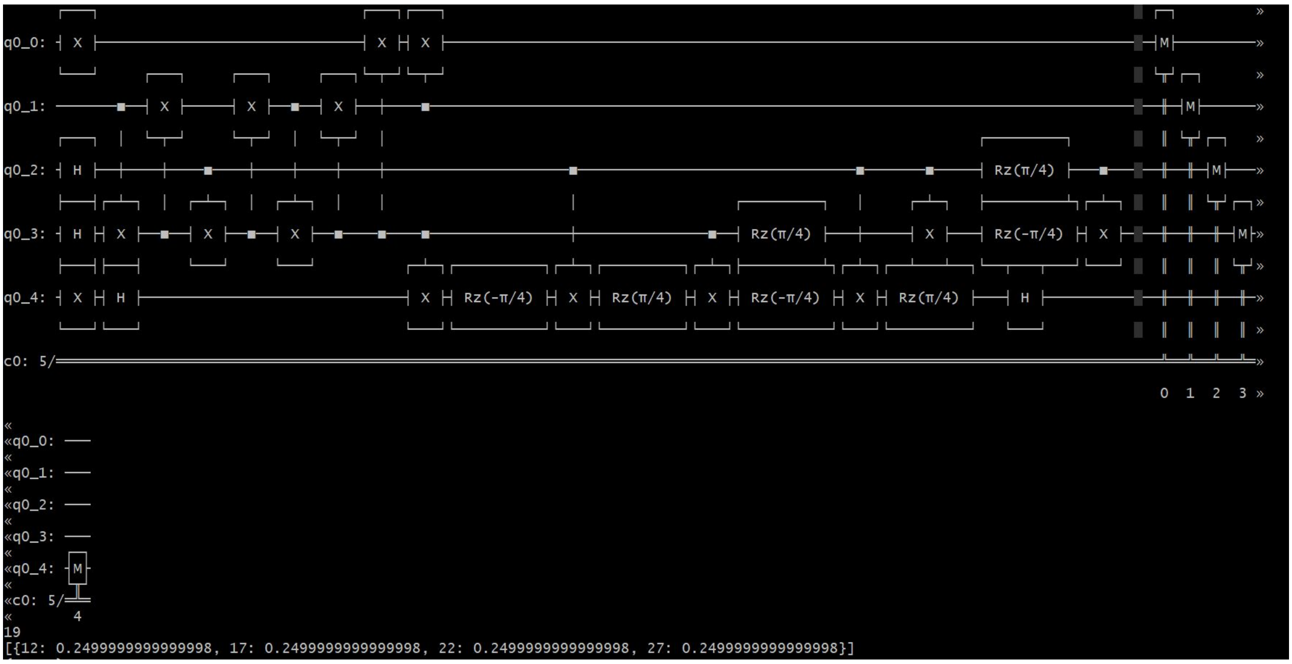

Figure 5: Final quantum circuit (Depth 19).

References

[SM08] Vivek V. Shende and Igor L. Markov. *On the CNOT-cost of TOFFOLI gates*. 2008. arXiv: `0803.2316 [quant-ph]`. URL: `https://arxiv.org/abs/0803.2316`.