Lab 2

THE QUANTUM FOURIER TRANSFORM

Brayden Hand | 2194051 | ELEN4022A | 27/04/2023

1. Introduction

To compare the performance of classical circuit simulation to quantum hardware, the implementation of Quantum Fourier Transform (QFT) and its applications (specifically in addition) are explored and analysed. In this report this includes, but is not limited to, the Quantum Full Adder (QFA), the Quantum Fourier Transform Adder (QFTA) and the Approximate Quantum Fourier Transform Adder (AQFTA). The remainder of this report explores validating and commenting on these results specifically using the already available Draper Adder for the QFTA, on both classical and quantum hardware. The Draper Adder is a quantum algorithm that utilises the QFT to perform the addition of two binary numbers with a speed faster that classical algorithms (Draper, 2000).

2. Background

Quantum computers excel in certain cases with high complexity such as Integer Factorization, which has its uses in cryptography because classical hardware struggles with factoring large numbers (Rieffel & Polak, 2011), and Optimization, which has its uses in finance and logistics because it is much faster (National Academies of Sciences, Engineering, and Medicine, 2019). However, quantum computers are not suited to low complexity problems where they can be outperformed by classical hardware. This is the case with very specific linear algebra problems where the algorithm complexity is increased by the use of quantum hardware (Press, 2018).

The QFT is an essential quantum algorithm used in a variety of applications such as:

- Error Correction: QFT helpful in methods of encoding and decoding information with minimum error (Gottesman, 2009).
- Cryptography: QFT utilised in quantum key distribution with the intent of preventing eavesdropping (Vedral, 2018).
- Phase Estimation: QFT utilised in estimating the phase of a quantum state for tasks such as chemistry and simulation (Rieffel & Polak, 2011).

3. The QFA

The implementation and description of the 'SUM' and 'CARRY' gates as well as the final ten qubit QFA, given in the lab brief, can be found in 'Lab_2_QFA_Simulated.ipynb' and 'Lab_2_QFA_Quantum.ipynb'. The QFA is capable of adding two, length two or three bit numbers and an example where five is added to seven to get twelve is demonstrated in both files. However, the classical simulation is the only one with results that are correct and then hence demonstrate the circuits successful simulation. I could not verify these results with quantum hardware as we do not have access to ten qubit quantum machines.

4. QFT Unitary Calculations

- 1 Wire Circuit: $QFT(x) = |y\rangle = \frac{1}{\sqrt{2^1}}(|0\rangle + e^{\frac{2\pi ix}{2^1}}|1\rangle)$
- 2 Wire Circuit: $QFT(x) = |y\rangle = \frac{1}{\sqrt{2^2}}(|0\rangle + e^{\frac{2\pi ix}{2^1}}|1\rangle) \otimes (|0\rangle + e^{\frac{2\pi ix}{2^2}}|1\rangle)$
- 3 Wire Circuit: $QFT(x) = |y\rangle = \frac{1}{\sqrt{2^3}}(|0\rangle + e^{\frac{2\pi ix}{2^1}}|1\rangle) \otimes (|0\rangle + e^{\frac{2\pi ix}{2^2}}|1\rangle) \otimes (|0\rangle + e^{\frac{2\pi ix}{2^3}}|1\rangle)$

(Asfaw, 2020)

5. The QFT and its Inverse (IQFT)

The implementation and description of the QFT and the IQFT can be found in 'Lab_2_QFT_and_IQFT_Simulated.ipynb' and 'Lab_2_QFT_and_IQFT_Quantum.ipynb'. Both files demonstrate the approach taken and adapted from the Qiskit textbook (Qiskit). An example in each file is done where the number three is encoded to the circuit, the QFT is applied, the IQFT is applied and then the result is determined classically and from quantum hardware to both produce '11' which is the binary equivalent of the number three. The quantum hardware does produce other results as expected due to the nature of quantum mechanics, but the highest count is '11', confirming our most expected outcome.

6. The QFTA

The implementation and description of the QFTA can be found in 'Lab_2_QFTA_Simulated.ipynb' and 'Lab_2_QFTA_Quantum.ipynb'. The QFTA is capable of adding two, length two or three bit numbers and examples where one is added to two to get three and where three is added to four to get seven are demonstrated in both files. As specified by the lab brief the QFTA can only add up to a maximum of $2^n - 1$ for an n-bit adder. For the classical simulation, both examples were successfully validated using the 'DraperQFTAdder'. However, for running the examples on quantum machines, issues arose. A two-bit adder, such as in the case of the first example, was successfully ran and validated (also using the 'DraperQFTAdder') on quantum hardware. In the case of the second example, a total of six qubits was needed. With the quantum hardware I have access to, I would need a seven-qubit machine. The seven-qubit machines are in high demand, and I would have to wait beyond the due date of this lab to view any results.

7. The AQFTA

The AQFTA was not implemented but was researched to a degree and then hence, still included in the analysis. It performs approximate addition by reducing the number of gates required while limiting its accuracy. It approximates the Fourier coefficients and then applies phase estimation to them to estimate the sum of the two encoded input numbers (Zhang, 2009).

8. Analysis

There are both advantages and disadvantages to classically simulating quantum circuits. In all cases (QFA, QFTA and AQFTA) classical simulation is not as restricted by qubit availability, error, or complexity. Quantum machines have limited qubit availability, high error rates (due to environmental noise) and physical complexities requiring a multitude of resources. However, quantum machines are capable of speeding up these algorithms for large inputs computationally impossible for classical machines and utilising quantum error correction techniques.

9. Conclusion

In conclusion, this report explored the implementation and analysis of the QFT and its applications, specifically in addition. The QFA, QFTA, and AQFTA were studied and validated using classical simulation. The report also attempted to validate the QFTA using quantum hardware, but issues arose due to qubit availability. It is evident that quantum computers excel in complex problems such as integer factorization and optimization. Still, they are not well-suited for low complexity problems, where classical hardware can outperform quantum hardware. In conclusion, both classical simulation and quantum hardware have their advantages and disadvantages, and it is essential to understand these factors to utilize them effectively.

References

1. "Addition on a Quantum Computer" by Robert D. Draper: https://arxiv.org/pdf/quant-ph/0008033.pdf. [Accessed 27 Apr.2023].
2. Rieffel, E. G., & Polak, W. H. (2011). Quantum computing: A gentle introduction. MIT Press. [Accessed 27 Apr.2023].

3. National Academies of Sciences, Engineering, and Medicine. (2019). Quantum computing: Opportunities and challenges. The National Academies Press. Accessed 27 Apr.2023].
4. Press, W. H. (2018). Quantum computing and its applications. Science, 362(6412), 1207-1212. [Accessed 27 Apr.2023].
5. "An Introduction to Quantum Error Correction and Fault-Tolerant Quantum Computation" by Daniel Gottesman, section 2.2: https://arxiv.org/pdf/0904.2557.pdf. [Accessed 27 Apr.2023].
6. "Introduction to Quantum Information Science" by Vlatko Vedral, section 7.3: https://books.google.com/books/about/Introduction_to_Quantum_Information_Sci.html?id=7HxkDwAAQBAJ&source=kp_book_description. [Accessed 27 Apr.2023].
7. Asfaw, A. Qiskit. (2020, September 1). 8.Shor's Algorithm I: Understanding Quantum Fourier Transform, Quantum Phase Estimation - Part 2 [Video file]. Retrieved from https://www.youtube.com/watch?v=pq2jkfJlLmY [Accessed 27 Apr.2023].
8. Qiskit (no date) Quantum Fourier transform, Qiskit Textbook. Available at: https://learn.qiskit.org/course/ch-algorithms/quantum-fourier-transform [Accessed 27 Apr.2023].
9. Zhang, J., Song, Z., Liu, L., & Long, G. L. (2009). Quantum approximate adder with fewer quantum resources. Physical Review A, 80(4), 042305. https://doi.org/10.1103/PhysRevA.80.042305 [Accessed 27 Apr.2023].