Integrating Quantum Noise into Quantum Gates for Enhanced Quantum Simulation

*T. de Laat, BSc, (tycho.delaat@student.fontys.nl)

**From the Master of Applied IT, Fontys University of Applied Sciences, Eindhoven, The Netherlands*

Keywords: *Quantum Noise Simulation | Noisy Gates | Quantum Computer | Quantum Simulator | QuantumSim | Qiskit | IBM | IBM Kyiv*

I. Abstract

In order to a acquire a good understanding of how quantum noise affects quantum circuits, a proper noise model is needed. This study investigated the possibility of modeling quantum noise within quantum gates, aiming to improve the realism of quantum simulators. The noise modeling method integrated into a quantum simulator was evaluated using a circuit which chained numerous Pauli X gates, Greenberger–Horne–Zeilinger (GHZ) state circuits, and inverse Quantum Fourier Transform (iQFT) circuits. Results demonstrate that the integrated noise model enhances the fidelity of simulations to a real quantum device, outperforming traditional Qiskit simulations. These findings suggest that incorporating noise at gate level can improve the accuracy of quantum simulations.

II. Introduction

Quantum computers have many possible future applications. One of these applications is their ability to perform accurate and efficient quantum chemical calculations to predict how candidate drugs interact with their targets [1], potentially speeding up and improving the drug design process. A second application that a quantum computer is expected to offer can be found in the cybersecurity field. One of the most widely used encryption tools is the Rivest–Shamir–Adleman (RSA) algorithm [2], which relies on the difficulty of factoring the product of two huge prime numbers [3]. In 1994, scientist Peter Shor presented a novel method of computing these prime factors using quantum computers, which could reduce the computation time from trillions of years to just eight hours [3].

Currently, quantum computers are in their noisy intermediate-scale quantum (NISQ) era [4]. This means that the current quantum computers are prone to high error rates and limited in size by the number of quantum bits (qubits) in the system. In short, this means they are not yet large enough to execute complex algorithms, such as Shor's algorithm [5], and unreliable to perform complex computations, such as the quantum chemical calculations needed to design new drugs [4].

These high error rates are a result of unwanted disturbances within a quantum system, and are often referred to as quantum noise. Quantum noise can arise from various sources, including thermal fluctuations, electromagnetic interferences, imperfections in quantum gates, and interactions with the environment [6].

Quantum noise is a significant barrier to the development of large-scale, fault-tolerant quantum computers. Even small amounts of noise can lead to decoherence, causing qubits to collapse and lose their superposition and entanglement properties [6]. Research into understanding the nature of quantum noise and developing methods to reduce or correct it, is a vital area of quantum computing [6].

In order to a acquire a good understanding of how quantum noise affects quantum circuits, a proper modeling of the noise is needed [7]. Currently, the simulation of noisy gate-based quantum computers is implemented by adding quantum operations before and after each ideal gate [7], [8]. This is visualized in Figure 1. If an ideal gate G is supposed to be executed, the noises affecting it are modeled by adding operations E_1 and E_2 to mimick the noise before and after the gate [7].

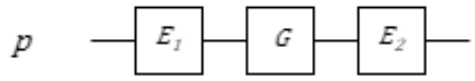

Figure 1 - Standard noise modelling in simulations.

This approach has been implemented by numerous available noise simulators of NISQ computers, some of which are shown in Table 1.

Table 1 - List of some of the quantum computing frameworks. Each of these frameworks supports noise simulation based on the approach described in Figure 1.

Company	Name
Fontys University of Applied Sciences	QuantumSim
IBM	Qiskit
Google	Cirq
Intel	Intel QS
Amazon	Braket

An example of such a simulator is QuantumSim [12], which is a quantum simulator developed by Fontys University of Applied Sciences. The goal of this simulator is to be a stepping stone for people entering the world of quantum computing by offering Python notebooks with in depth explanations and examples.

Another example is the AerSimulator, which is one of three simulators offered by Qiskit, the world's most popular software stack for quantum computing [9]. Qiskit provides a rich set of tools for creating, simulating, and executing quantum circuits, making it suitable for researchers, educators, and developers. Besides offering simulators, Qiskit allows users to execute circuits on a real quantum computers. The AerSimulator is capable of executing quantum circuits with or without noise.

This paper aims to extend QuantumSim with a noise model based on the model proposed by Di Bartolomeo et al. [7] and compare the results to other noise simulators, including the AerSimulator from Qiskit, as well as data from the IBM Kyiv quantum computer.

The model proposed by Di Bartolomeo et al. [7] aims to integrate the noise into the gates of a simulator. By doing this, the model represents a more faithful description of what happens inside a quantum computer, where, for example, imperfectly implemented gates and the environment act simultaneously and potentially affect one another [7].

The remainder of this paper systematically addresses the steps taken to enhance the noise simulation in QuantumSim.

III. Methodology

A literature study was conducted to identified relevant work that provided valuable insights into simulation methods [7], [8], [10]. As previously stated, for the purpose of this study, a noise model (hereinafter also referred to as 'noisy gates') inspired by the work of Di Bartolomeo et al.

[7] was implemented into the gates of the QuantumSim simulator.

The implemented noise model required a set of quantum noise parameters to be defined. These parameters represent the single-qubit depolarizing error probability (p), amplitude damping time of a qubit in ns ($T1$) and the dephasing time of a qubit in ns ($T2$). To guarantee the realism of these parameters, the values were extracted from a real quantum computer, specifically the Kyiv device from IBM.

As part of verifying the enhancements to the quantum noise simulation within QuantumSim, data was collected from multiple sources to assess alignment with real-world behaviour. These sources include QuantumSim, the AerSimulator from Qiskit, and the Kyiv quantum computer from IBM.

The simulators and the real quantum computer were provided with identical circuits for execution. These circuits include a circuit in which numerous Pauli X gates were chained after one another (see Appendix [A]), Greenberger–Horne–Zeilinger (GHZ) state circuits (see Appendix [B]), and inverse Quantum Fourier Transform (iQFT) circuits (see Appendix [C]). The first type of circuit was run with $n = 30, 60, 90, 120, 150, 250, 500$, where n is the number of gates in the circuit, the GHZ state circuits and iQFT circuits were executed for $n = 2, ..., 5$ and $n = 2, 3,$ respectively, where n denotes the number of qubits.

The GHZ state is a type of entangled quantum state, often represented as

$$|GHZ\rangle = \frac{1}{\sqrt{2}}\left(|0\rangle^{\otimes n} + |1\rangle^{\otimes n}\right)$$

[11]. Here, n represents the number of qubits in the system, and the symbol $\otimes$ denotes the tensor product, which combines the quantum states of multiple qubits. For example, the tensor product $|0\rangle^{\otimes n}$ represent all n qubits being in the $|0\rangle$ state. The GHZ state is a maximally entangled state, meaning that the qubits are correlated in such a way that the state of any individual qubit is completely dependent on the states of the others. This state extends the concept of the Bell state, another entangled state involving two qubits, to three or more qubits [11].

Quantum Fourier Transform (QFT) is a quantum implementation of the Discrete Fourier transform (DFT), which is an algorithm to determine the frequency components of a function [13]. The QFT is part of numerous quantum algorithms, including Shor's factoring algorithm [14]. Similarly to how the DFT acts on a vector $(x_0, \dots, x_{N-1})$ and maps it to a vector $(y_0, \dots, y_{N-1})$, the QFT acts on a quantum state $|X\rangle = \sum_{j=0}^{N-1} x_j |j\rangle$ and maps it to a quantum state $|Y\rangle = \sum_{k=0}^{N-1} y_k |k\rangle$ [14]. The inverse QFT, or iQFT, reverses the actions of a regular QFT and is often written as QFT†.

Each simulator was tasked with executing the circuits ten times, with each execution round being 1000 shots, while the real quantum computer was tasked with executing each circuit once for 10,000 shots. Each shot of the chained Pauli X and GHZ state circuits had initial state $|0\rangle^{\otimes n}$. The initial state of the QFT† circuits was set to be $|+\rangle^{\otimes n}$. The state $|+\rangle$ is a quantum state that represents a superposition of the basis states $|0\rangle$ and $|1\rangle$ and can also be defined as

$$|+\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle).$$

This means that, when measuring the qubit, it has an equal chance of being measured as $|0\rangle$ or $|1\rangle$. By using this input state, the result of the QFT† circuit is expected to be $|0\rangle^{\otimes n}$.

Following the execution of all circuits, the resulting data was saved and formatted to

ensure consistency across all platforms. For instance, both QuantumSim and the AerSimulator from Qiskit return the resulting state as a string; however, the output of the former is formatted with the most significant bit first (MSB first), and the latter with the least significant bit first (LSB first).

The formatted data was then used to facilitate a visual comparison between he different platforms. Histograms were plotted to show the percentage of occurrences of each classical state over 10,000 total shots, providing a visualization of the distribution of outcomes across the simulators and the IBM Kyiv quantum computer. To further quantify the differences between the platforms, the Hellinger distances were computed, which measures the similarity between the probability distributions produced by the simulators and the quantum computer. The Hellinger distance is defined by: $H^2(P,Q) = \frac{1}{2}\int_x \left(\sqrt{p(dx)} - \sqrt{q(dx)}\right)^2$, where $H^2(P,Q)$ is the squared Hellinger distance between two probability distributions P and Q, $p(dx)$ and $q(dx)$ are the probability density functions (PDFs) of distributions P and Q, respectively, and x is the random variable over which the distributions P and Q are defined.

IV. Results

Three types of circuits have been executed: a circuit containing only Pauli X gates chained after one another, the Greenberger–Horne–Zeilinger (GHZ) state circuit, and the inverse Quantum Fourier Transform (iQFT) circuit.

I. Chained Pauli X gates comparison

In Figure 2, the occurrence frequency of each classical state was recorded and represented as a percentage, with the QuantumSim noisy gates (light grey), the Qiskit AerSimulator (dark grey), and the IBM Kyiv quantum computer (black) shown for comparison.

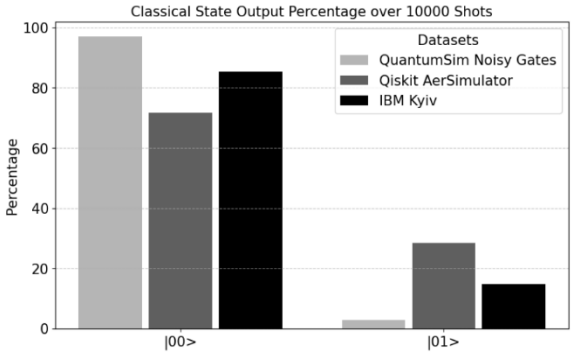

Figure 2 - Histogram for the percentage of occurrences per classical state over 10,000 shots for a circuit with 90 chained Pauli X gates.

An examination of Figure 2 shows that both QuantumSim noisy gates and Qiskit AerSimulator deviate from the IBM Kyiv results, which serves as the baseline for reality. The QuantumSim noisy gates implementation remains slightly more faithful to IBM Kyiv with a deviation of approximately 11.84%, compared to the Qiskit AerSimulator, which deviates approximately 13.67%. When comparing different circuit lengths through the calculation of the Hellinger distances, an interesting trend emerges, as shown in Figure 3. Specifically, the Hellinger distance of the QuantumSim noisy gates steadily increases to a maximum value of approximately 0.7 at a circuit length of 250, before dropping sharply to around 0.08 for a circuit with 500 gates. At the same circuit length where the QuantumSim noisy gates reach their peak distance, the Qiskit AerSimulator shows a significant jump, increasing from a Hellinger distance of approximately 0.06 to about 0.4.

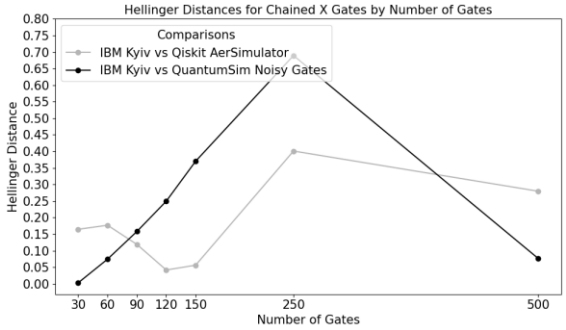

Figure 3 - Hellinger distances comparison between IBM Kyiv vs Qiskit AerSimulator (light grey) and IBM Kyiv vs QuantumSim noisy gates (black).

An analysis of the circuit with 250 gates, as shown in Figure 4, reveals that, the IBM Kyiv quantum computer no longer outputs the expected state $|0\rangle^{\otimes n}$.

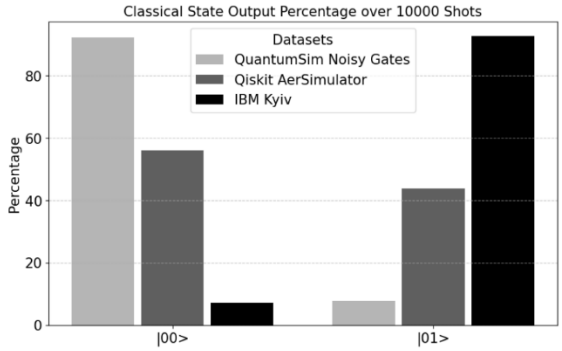

Figure 4 - Histogram for the percentage of occurrences per classical state over 10,000 shots for a circuit with 250 chained Pauli X gates.

II. GHZ state comparison

The number of occurrences of each classical state output was counted and converted into percentages. These percentages are shown in Figure 5 for: the QuantumSim noisy gates (light grey), the Qiskit AerSimulator (dark grey), and the IBM Kyiv quantum computer (black).

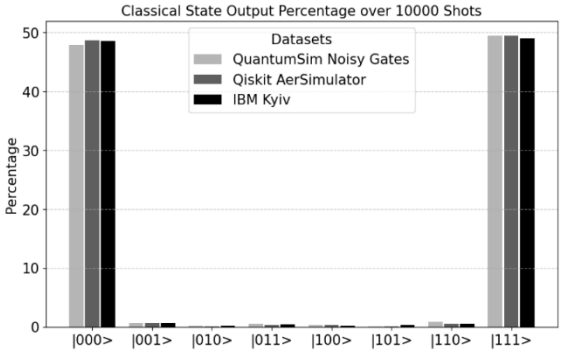

Figure 5 - Histogram for the percentage of occurrences per classical state over 10,000 shots for a three qubit GHZ state circuit.

When observing Figure 5, it becomes apparent that both simulations perform close to reality (IBM Kyiv), with a maximum deviation of approximately 0.74% which occurs at the classical state $|000\rangle$. To further compare the simulators to the real device, defined here as the IBM Kyiv results, the Hellinger distances were calculated. Figure 6 illustrates that when comparing the noisy gates implementation in QuantumSim to the Qiskit AerSimulator, the results obtained with QuantumSim are closer to the results obtained with the real device.

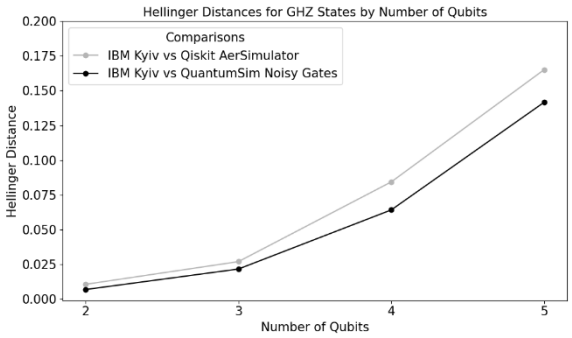

Figure 6 - Hellinger distances comparison between IBM Kyiv vs Qiskit AerSimulator (light grey) and IBM Kyiv vs QuantumSim noisy gates (black).

III. iQFT comparison

After execution, the frequency of each classical state output was measured and expressed as a percentage. Figure 7 depicts the classical state occurrence percentages for the QuantumSim noisy gates (light grey), the Qiskit AerSimulator (dark grey), and the IBM Kyiv quantum computer (black).

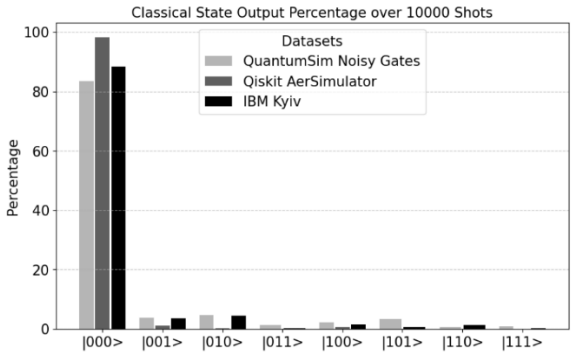

Figure 7 - Histogram for the percentage of occurrences per classical state over 10,000 shots for a three qubit QFT† circuit.

Figure 7 demonstrates that both simulators primarily produce the expected output of $|0\rangle^{\otimes n}$. However, at this point, their deviation from the actual results, as defined by the IBM Kyiv data, is the largest, approximately 9.91%. To further analyse the discrepancies between the simulators and a real quantum device (IBM Kyiv), the Hellinger distances for $n = 2$ and $n = 3$ qubits were computed. Figure 8 presents these results.

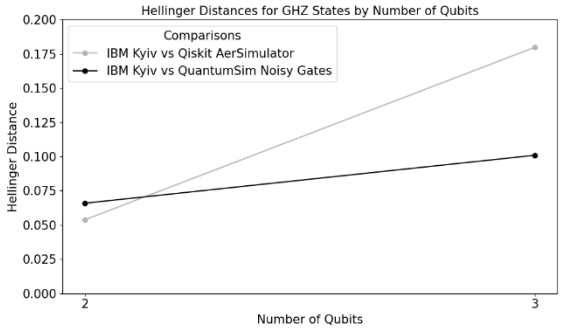

Figure 8 - Hellinger distances comparison between IBM Kyiv vs Qiskit AerSimulator (light grey) and IBM Kyiv vs QuantumSim noisy gates (black).

From Figure 8, it can be observed that while both simulators demonstrate close fidelity to reality, with a difference between the simulators of approximately 0.012 in Hellinger distance for the two-qubit QFT† circuit, the QuantumSim noisy gates implementation has a smaller Hellinger distance for the three-qubit QFT† circuit, with a difference of approximately 0.08 between the simulators.

V. Discussion

From the presented data, the increase in fidelity to real quantum devices, such as the Kyiv device from IBM, at relatively low circuit lengths, followed by a gradual decay in fidelity as the circuit size grows becomes apparant.

This decay in fidelity contradicts the predication made by Di Bartolomeo et al. [7], who suggest that their approach should provide a more accurate representation of a NISQ computer, particularly when the number of qubits and gates increases [7]. The observed discrepancy points to potential limitation in the approach, particularly at larger circuit sizes, suggesting that further investigation is necessary.

Additionally, the data revealed unexpected behaviour from the IBM Kyiv quantum computer, by highlighting that it failed to produce the correct output for a circuit with a length of 250 gates. A potential explanation for this behaviour is that, at this circuit length, the accumulated noise likely reduces the chance of the qubit being measured as $|0\rangle$, to an extent in which it became more likely to be measured as $|1\rangle$. This reasoning could, in addition, explain why, for a circuit length of 500 gates, the fidelity of the QuantumSim noisy gates increased significantly, as the accumulated noise reached a point where the probabilities flipped again, making it more likely to measure the qubit as $|0\rangle$.

A notable limitation of the present study can be found at the noise source level. Quantum devices, like the IBM Kyiv device, suffer from state preparation and measurement errors, the noise model implemented into the QuantumSim gates, does not yet simulate these types of noise. Future studies could implement these noise sources and reevaluate the fidelity of the noise model compared to a real quantum device. Additionally, the present study only utilized the IBM Kyiv quantum computer, this could be extended to, for example, IBM Brisbane or IBM Sherbrooke, to ensure the modularity of the model.

VI. Conclusion

This study investigated the integration of quantum noise into quantum gates, aiming to provide a more realistic representation of noise in quantum circuits. The results demonstrated that the integrated noise

simulation approach improved fidelity to real quantum devices, such as the Kyiv device from IBM, for the circuit that chained Pauli X gates with a length of less than 90 gates, and the GHZ and iQFT circuits. In particular, the approach outperformed the noise simulation provided by the AerSimulator from Qiskit.

These findings suggest that integrating noise at the gate level can enhance the accuracy of quantum simulators, offering a valuable tool for studying the performance of quantum devices. Study limitations include, not all noise sources were implemented in QuantumSim, and only one real quantum device was utilized.

VII. References

[1] R. Santagati et al., “Drug design on quantum computers,” Nature Physics, vol. 20, no. 4, pp. 549–557, Mar. 2024, doi: 10.1038/s41567-024-02411-5.

[2] “Security System analysis in combination method: RSA encryption and Digital signature algorithm,” *IEEE Conference Publication | IEEE Xplore*, Aug. 01, 2018. https://ieeexplore.ieee.org/abstract/document/8528584

[3] Akitra, “The Invisible Threat: How quantum computing could break today’s encryption?,” Medium, Nov. 22, 2024. [Online]. Available: https://medium.com/@akitrablog/the-invisible-threat-how-quantum-computing-could-break-todays-encryption-888e3ea99cf3#:~:text=Quantum%20computing%20poses%20a%20significant,problems%20fundamental%20to%20cryptographic%20security.

[4] J. Dargan, “What is NISQ Quantum Computing?,” The Quantum Insider, May 28, 2024. https://thequantuminsider.com/2023/03/13/what-is-nisq-quantum-computing/

[5] E. Parker, “When a quantum computer is able to break our encryption, it won’t be a secret,” RAND, Sep. 13, 2023. https://www.rand.org/pubs/commentary/2023/09/when-a-quantum-computer-is-able-to-break-our-encryption.html

[6] “What is Quantum Noise.” https://www.quera.com/glossary/noise

[7] G. Di Bartolomeo *et al.*, “Noisy gates for simulating quantum computers,” *Physical Review Research*, vol. 5, no. 4, Dec. 2023, doi: 10.1103/physrevresearch.5.043210.

[8] J. Sun, X. Yuan, T. Tsunoda, V. Vedral, S. C. Benjamin, and S. Endo, “Mitigating realistic noise in practical noisy Intermediate-Scale quantum devices,” *Physical Review Applied*, vol. 15, no. 3, Mar. 2021, doi: 10.1103/physrevapplied.15.034026.

[9] “Qiskit | IBM Quantum Computing.” https://www.ibm.com/quantum/qiskit

[10] E. Magesan, D. Puzzuoli, C. E. Granade, and D. G. Cory, “Modeling quantum noise for efficient testing of fault-tolerant circuits,” *Physical Review A*, vol. 87, no. 1, Jan. 2013, doi: 10.1103/physreva.87.012324.

[11] “What is GHZ State.” https://www.quera.com/glossary/ghz-state

[12] Nicokuijpers, “QuantumSim,” GitHub. https://github.com/nicokuijpers/QuantumSim

[13] “Quantum Fourier Transform,” in Chapter 5, [Online]. Available: https://d37djvu3ytnwxt.cloudfront.net/c4x/BerkeleyX/CS-191x/asset/chap5.pdf

[14] Qiskit, “textbook/notebooks/ch-algorithms/quantum-fourier-transform.ipynb at main · Qiskit/textbook,” GitHub. https://github.com/Qiskit/textbook/blob/main/notebooks/ch-algorithms/quantum-fourier-transform.ipynb

VIII. Appendix

[A] Chained X gate circuit

Below an example of a circuit which executes 20 X gates is depicted. While, during the process of gathering results for the present study circuits of lengths $n = 30, 60, 90, 120, 150, 250, 500$, where n is the number of gates in the circuit, were used, for readability reasons only a circuit of $n = 20$ is shown.

Figure 9 - 2 Qubit circuit with 20 chained X gates.

[B] GHZ state circuits

Below all different circuits used to gather data for the GHZ states are shown.

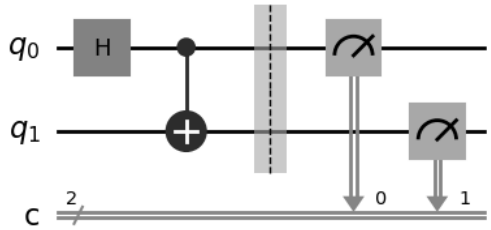

Figure 10 - 2 Qubit GHZ state circuit.

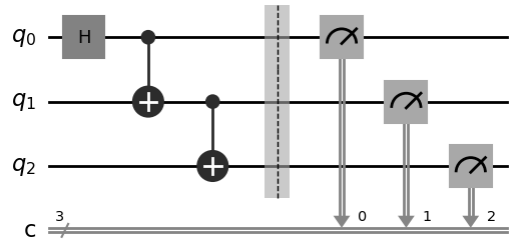

Figure 11 - 3 Qubit GHZ state circuit.

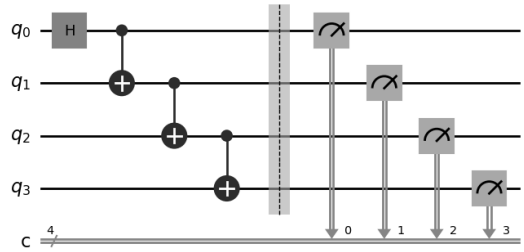

Figure 12 - 4 Qubit GHZ state circuit.

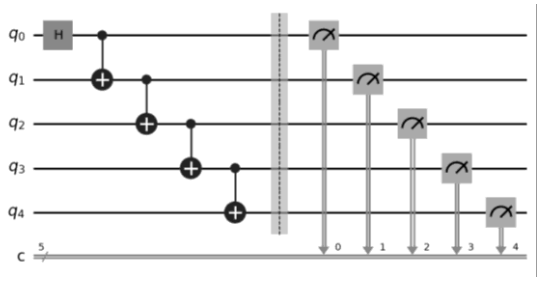

Figure 13 - 5 Qubit GHZ state circuit.

[C] iQFT circuits

Below all different circuits used to gather data for the iQFT algorithms are shown.

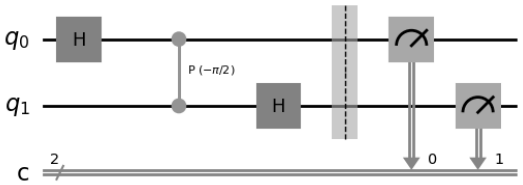

Figure 14 - 2 Qubit iQFT circuit.

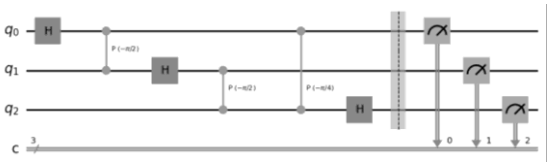

Figure 15 - 3 Qubit iQFT circuit.