Sorting Circuit for Quantum Computer

by Hamza, Yousaf Izhar

Submission date: 01-Mar-2021 10:01AM (UTC+0500)
Submission ID: 1520956161
File name: thesis_2_merged.pdf (2.37M)
Word count: 4313
Character count: 19764

Sorting Circuit for Quantum Computer

Project Team

Muhammad Hamza	P16-6068
Muhammad Yousaf Khan	P16-6059
Syed Mir Izhar Ali Shah	P16-6125

Session 2016-2021

Supervised by

Dr. Omer Usman Khan

10

Department of Computer Science

National University of Computer and Emerging Sciences
Peshawar, Pakistan

February, 2021

Abstract

Sorting Circuits are almost non-existent for Quantum Computer(QC) However, they are curial to a number of Algorithms, in Computer Science. Our work is to implement such a circuit which runs on both simulator as well as actual Quantum machines. Quantum computation has gained a lot of attention especially in these days because quantum computers are faster and more accurate than classical computers. So, for searching in unsorted database usually requires O(N) operations but in quantum computer this can be done in O(NlogN) operations. This circuit works on quantum entanglment technique that is a theory of quantum physics.

Chapter 1

Preliminaries and Introduction

1.1 Introduction

Sorting Circuit for Quantum Computer(QC) is basically a circuit designed to sort elements in quantum computers(QC). This is a new experince in the field of computer science because quantum computer(QC) are very rare. There are very few computers in the world only 5 companies Google, IBM, D-Wave Systems, Microsoft and Intel are producing these systems. Some small companies are also investing in this study. Many other companies of different countries are patenting the families assosiated with QC. Some companies are patenting the hardware designs and some softwares. Almost every technical firm (Nokia, Sony, Hitachi, Toshiba) are investing in QC and patenting famillies assosiative with QC.

1.2 Project Purposes

The concept of Quantum Computer(QC) has been around since 1980's atleast in theoretical perspective. Richard Feynman first spoke about this technology, intially it was thought an impossible technology because of unstable nature of quantum particals now with the advancment of technology it become possible to design a QC. Still it is not common and software are not wriiten for QCs. The project aims to provide a circuit that will sort elements for QC using followig techniques:

- Instant Target: Converting Digital Number to binary and breaking them to half.
- Comparing: Comparing the numbers using if operation in classical bit.
- Sorting: Sorting is done by using quantum entanglment technique.

1.3 Project Domain

The domain of this project is to design a circuit that will do sorting in quantum computers as there is very small work done in this field.

1.4 Idea of Project

The idea of this project is to provide quantum computers a sorting circuit using Qiskit.

1.5 Scope of Project

This circuit will open a different door for sorting data using quantum computer. Data that a classical computer can sort in N time this will sort in only NlogN time that is a major breakdown in the field of technology.

Chapter 2

Literature Study

2.1 Quantum Physics

Quantum physics is a branch of physics which explain how atom works inside the nature of the particles that makeup matter and how different forces of nature interact with it as we know that electron is the smallest particle that interact with a computer chip, Photons of light (Electron) get turned to electrical current signal that moves inside the whole computer system and make it perform unbelievable operations.

2.2 Quantum Mechanics

Quantum mechanics is the study of the dynamics of particles at its most fundamental level. The state of a particle, such as its position or momentum, is described by a statistical distribution given by its wave-function. As the name says, this formalism gives matter many properties that are classically associated with waves.And how we can manipulate the position or momentum of this particle and obtain the desire results. Using the laws of Quantum physics.

2.3 Quantum Computing

Using Quantum phenomenons such as superposition and entanglement to manipulate the quantum particle and than perform computation on the basis of it. Computer used to perform such computation are known as Quantum Computers. Using Quantum Computers to perform computation is called Quantum Computing.

2.4 Bra–Ket Notation / Dirac notation

$$|00\rangle, |01\rangle, |10\rangle, |11\rangle.$$

It denotes a vector,In an abstract (complex) vector space, Physically it represents a state of some Quantum System and Quantum particles, Moreover it denotes a linear form, i.e. A linear map that maps each vector in to a number in the complex plane

2.5 Introduction of Quantum Computer

We studied different previous research works and Quantum Computer(QC) techniques. How it works and what is it made of for better view of quantum computer(QC) to design a circuit. We focused on the working of Qbits because that is main difference between classical computer and quantum computer. Quantum computing is the use of quantum-mechanical phenomena such as superposition and entanglment to perform computation. The bits use in quantum computer are qubits. Qubits needs very low temperature that is -273 degrees to perform operations. They kept in super-conductors. Qubits are composed of molecules and atoms containing electrons.

2.6 Phenomena of Quantum Computer

There are two phenomenas of Quantum Mechanics

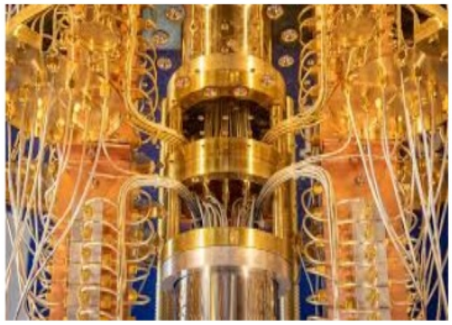


Figure 2.1: Quantum Computer

- Superposition

- Entanglment

2.6.1 Superposition

A classical bit is in 0 or 1 state at a time but qubit can be in both states at a time and this phenomena is superposition.

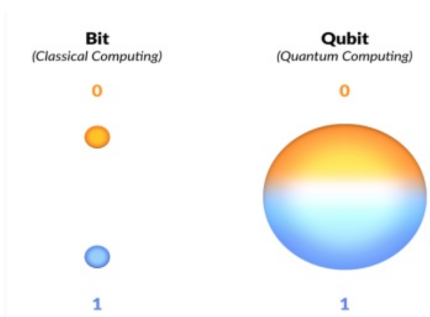


Figure 2.2: Superposition

2.6.2 Quantum Entanglment

This phenomena occuerrs when two particles become inextricably linked, and whatever happened to one immediately affects the other, regardless how far they are. Quantum entanglment is a technique of quantum computers(QC) which we are using in designing the sorting circuit. this technique is explained in most of the research papers. According to this technique we can sort elements on the basis of one qubit we can sort the other one thats how entanglment works.

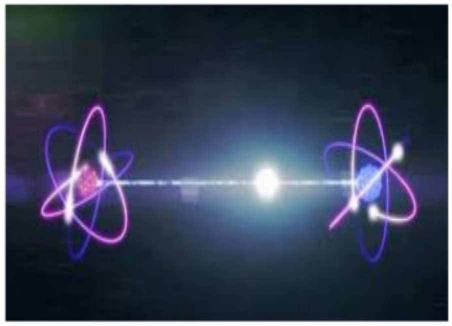


Figure 2.3: Quantum Entanglment

Chapter 3

Quantum Gates and Operations

3.1 Introduction

As we know quantum computers are in early age so what can we achieve through quantum Computers, with classical Computers we achieve impossible in the last few decades just by using simple Arthimatic operation like plus minus division multiplication and many long tired calculation just in few seconds to perfom these operation some logical gates are used like AND, OR, NOT NAND and using these gates in computer circuits we achieve unbelievable. But there are some problems which are still unsolvable by Modern days super computer that's where Quantum Physics came in explaining the Dream to solve those impossible problems. Following are the gates that are used in Quantum Computers which will help us to achieve the impossible.

- Hadamard Gate
- Pauli – X / NOT
- Pauli - Y
- Pauli - Z
- CX Gate
- CCX Gate

- SWAP Gate

- CSWAP Gate

3.2 Hadamard Gate

Hadamard Gate / H – Gate is the wonder of Quantum Mechanics which helped us to achieve the impossible and resolve a problem which even the modern days super Classical Computers can't solve was "To generate a number on the basis of total Randomness". What Hadamard / H – gate do to a Qubit (Quantum particle) is it puts it into the Superposition (Quantum Mechanics Phenomena) A state in which the qubit have exactly 50% probability of falling into state 1 and 50% probability of falling into state 1. which is totally Random it's like a fair coin toss can be either Head or tale similarly in case of a Qubit Either 0 or 1.

Figure 3.1: Hadamard Gate

$$H = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$$

$$H|0\rangle = |+\rangle \quad H|1\rangle = |-\rangle$$

3.3 Pauli – X / NOT

Pauli – X gate is the Quantum gate in Quantum Computers which is equivalent to the NOT Gate in Classical Computers. As in Classical computers we can change a bit from 1 to 0 or 0 to 1 using simple NOT gate, Similarly in Quantum Computers we can flip the Qubit (Quantum Bit) from 0 to 1 and 1 to 0. than what's the difference in Classical NOT and Quantum NOT / Pauli X gate. To understand the difference between these two think of the Qubit as a Bloch Sphere and think it contains 1 at one edge and 0 and the other end. Now what pauli - X Gate / NOT gate do is it rotates the Qubit towards X- axis and help us to flip the result. If qubit contains 1 and we apply Pauli - X Gate / NOT gate. It moves the momentum of the Quantum particle towards the opposite site and flip it towards 0. Now as the Qubit have more probability of falling in 0 state has we flip the result.

Figure 3.2: X Gate

$$|a\rangle = v_0|0\rangle + v_1|1\rangle \rightarrow \begin{bmatrix} v_0 \\ v_1 \end{bmatrix}$$

$$|0\rangle = \begin{bmatrix} 1 \\ 0 \end{bmatrix}, |1\rangle = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$$

$$X = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} = |0\rangle\langle 1| + |1\rangle\langle 0|$$

$$X|0\rangle = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \end{bmatrix} = |1\rangle$$

3.4 Pauli - Y and Pauli – Z

As explained above for the case of Pauli – X gate Similarly Pauli – Y and Pauli – Z gates are used to manipulate the momentum of qubit and move it towards the Y, Z axis and help us to achieve the desired result.

Figure 3.3: Y and Z Gate

$$|+\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle) = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 \\ 1 \end{bmatrix}$$

$$|-\rangle = \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle) = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 \\ -1 \end{bmatrix}$$

3.5 CX Gate

Controlled X gate / CX gate as mentioned in the name it controls another qubit with X / Not gate, Basically CX is applied on two Qubits Think of it as One Qubit is Controlling the other Qubit which is the target Qubit. So there's Control Qubit and a Target Qubit. We can use it as a if statement as we do in Classical Computers, remember all Quantum operations are reversible and so does the Quantum Gates so on real Quantum Computers we can't use if statement , else if statement, and other operations like loops , greater than , smaller than because these operations are based on logical operations which are irreversible so in-order to do these things on Quantum Computers. we have to use Quantum Gates and build these operations from scratch. In CX, X gate is applied on target Qubit if the Control Qubit is in state 1 otherwise X / NOT gate isn't applied on target Qubit.

Figure 3.4: CX Gate

$$|00\rangle, |01\rangle, |10\rangle, |11\rangle.$$

$$CNOT = CX = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

3.6 CCX Gate

CCX / Toffoli gate is a 3- Qubit gate. There is 1 target Qubit and 2 controlled Qubits, X – gate is applied to the target Qubit if the controlled Qubits receive the input of 1 on both otherwise it won't apply it to the target Qubit. CCX is not a Universal gate for Quantum Computers because it's irreversible and if we place Hadamard gate at the input end of the Controlled Qubits than it's universal.

Figure 3.5: CCX Gate

$$CCX = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}$$

3.7 SWAP Gate

As we can guess from the name it SWAP the states of two Qubits let's say you have two Qubits One with the state 0, and other with the state 1. Applied SWAP gate the states of the two Qubits SWAP with each other.

Figure 3.6: SWAP Gate

$$|00\rangle, |01\rangle, |10\rangle, |11\rangle.$$

$$SWAP = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

3.8 CSWAP

As explained above in swap gate section this gate also swap two Qubits on the basis of controlled Qubit. If the controlled Qubit is in state 1 then, Target Qubits will be swapped.

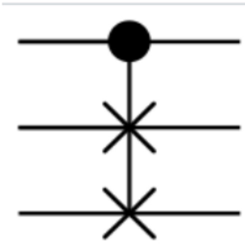


Figure 3.7: CSWAP Gate

$$CSWAP = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

There are more Gates than these but to understand our project these are the base.

3.9 Quantum Operations

There are some operations that are useful in desiging the circuit.

3.10 IF Operation

IF operation is use to check the equality of two Qubits weather they are equal or not.

Figure 3.8: IF Operation

3.11 Barrier

Barrier operation is use to seperate two gates in complex circuit to avoid mixing.

Figure 3.9: Barrier Opertaion

3.12 Z-Measurment

Z operation is use to measure the value of qubit after applying the gates.

Figure 3.10: Z-Measurment Operation

3.13 Reset Operation

This opertaion is use to set the qubit to zero no matter what its perivous state.

Figure 3.11: Reset Opertaion

Their are more operations but for our projects they are more than enough to understand.

Chapter 4

Quantum Circuits

4.1 Qubit Representation

To understand the working of Quantum Computers we need to understand what are Qubits. So let's Think remember a bit in Classical Computer the tiny electron converted into electric signal wave carrying information of either 0 or 1 at a time running through the Computer circuits. From one Component of it to another and being stored on Hard-drive carrying all your huge Data and Personal Information in the form of 0's and 1's in binary string. Similarly a Qubit Carries 0 and 1 in it and at a time if not being observed/ measured. Just imagine the information it carries along it, it's magical Using this power of Qubits we can achieve unimaginable. Qubits and Bits aren't very different just think of it as a Bit Carrying information but you can manipulate that information using a Quantum Computers Working on the Laws of Quantum Mechanics. To represent a Qubit we can think of it as a Bloch Sphere as explained in the picture.

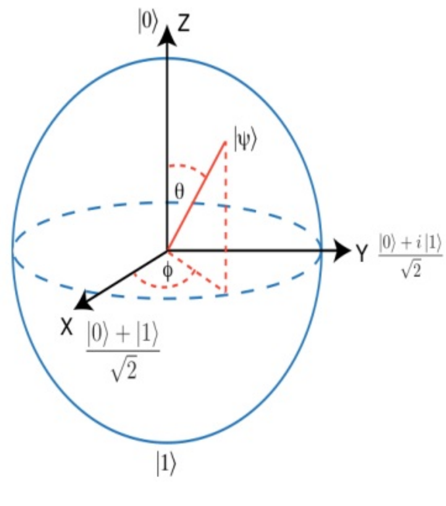


Figure 4.1: Qubit Representation

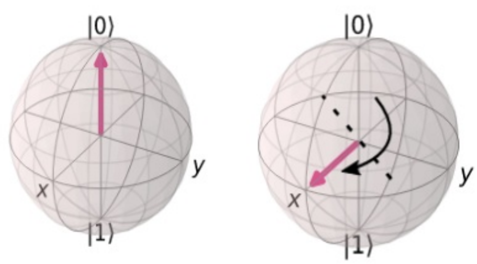


Figure 4.2: Start Qubit at 0 and Hadamard Gate Applied

These are the graphical representation of a Qubit in first figure Qubit is in state 0 and after applying Hadamard Gate there is a probabilty of 50% falling into state 0 or 1.

4.2 Quantum Circuit

Quantum Circuits are composed of Quantum gates which are applied on Qubits to create a whole Circuit . For now in the world there are Quantum Computers consisting on 73 Qubits created by Google, and another Computer Consisting on 53 Qubits at IBM also some other Computers are available to IBM with less number of Qubits. As to make them work is very expensive and have constraints like it was for Classical Computer in it's early age. We will show you the circuits we created and some other interesting examples consisting on quantum Algorithms which are being created so far by the researchers.

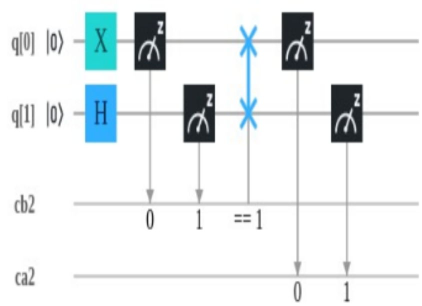


Figure 4.3: Example of a Circuit

In this circuit 2 Qubits have been used. We have placed an X-Gate on 1st Qubit and Hadamard Gate to 2nd Qubit. After that we are measuring both Qubits and storing them in 2 classical registers.Then we use a Swap Gate to swaps these Qubits and measuring again to confirm the swap.

Chapter 5

Quantum Algorithms

5.1 Circuits

Quantum Algorithms are very rare for now because they are based Quantum Mechanics Phenomenas and to run these Algorithms we need Quantum Computers as Quantum Computers are in early age and will get stable soon in near future but thanks to classical Computers we can still run them on Simulators.We designed the following Quantum Circuits to compare Qubits either Qubit is greater than other, Or smaller, Or equal,So we can sort them by using CSWAP Gate.Other than that we will show you some Quantum Algorithms that we studied and helped us to build this Sorting Circuit for Quantum Computers.

- Random Byte Generator
- Bernstein-Vazirani Algorithm
- 2-Bit Simple Swapping Circuit

5.2 Random Byte Generator

Random Byte Generator, So far being highly succesful Classical Computers were stil dealing with the problem of generating a number based on total randomness, Quantum Computers resloved it easily with it's Quantum magic as we explained above the phenomena of Quantum Mechanics "Superposition", which we use in Hadamard gate to assign it randomly with either 0 or 1. So using Hadamard gate on 8 Qubits we are generating a random number in the algorithm. Running it on a simulator will always gives u a total random number.

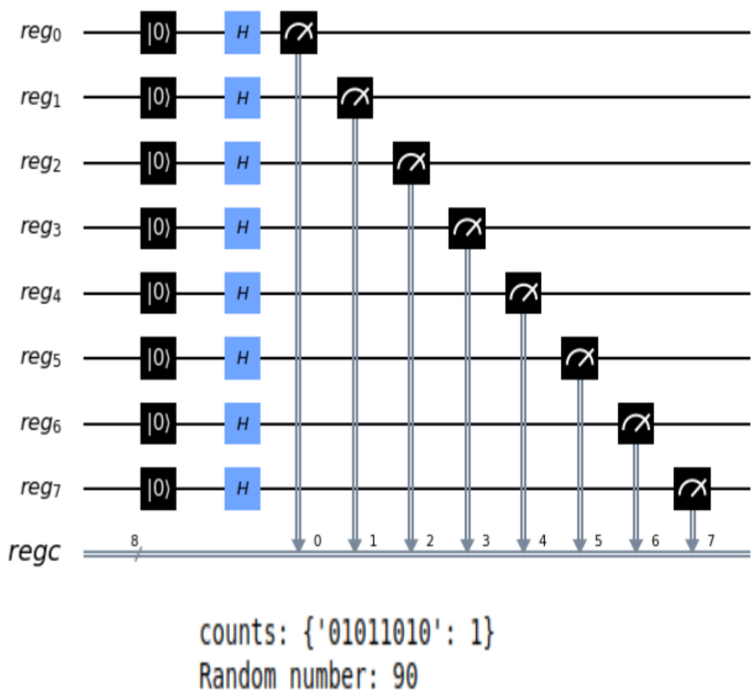


Figure 5.1: Random Byte Generator

5.3 Bernstein-Vazirani Algorithm

Bernstein-Vazirani Algorithm, named after the scientist propsed that it can guess the secret number placed inside a black box in one Shot. In the circuit below you can see 9 Qubits on which we have placed Hadamard gate, and an X gate before applying the hadamard gate on the 9th Qubit after that there's a barrier inside those barrier is our black box onto which CX Gate is Applied on all those Qubits receving 1 as an input and controlled by the last Qubit and after the 1st barrier we have placed again Hadamard gate on Qubits except for last one and after that we are measuring all those 8 Qubits and store them in classical register showing us our hidden number.

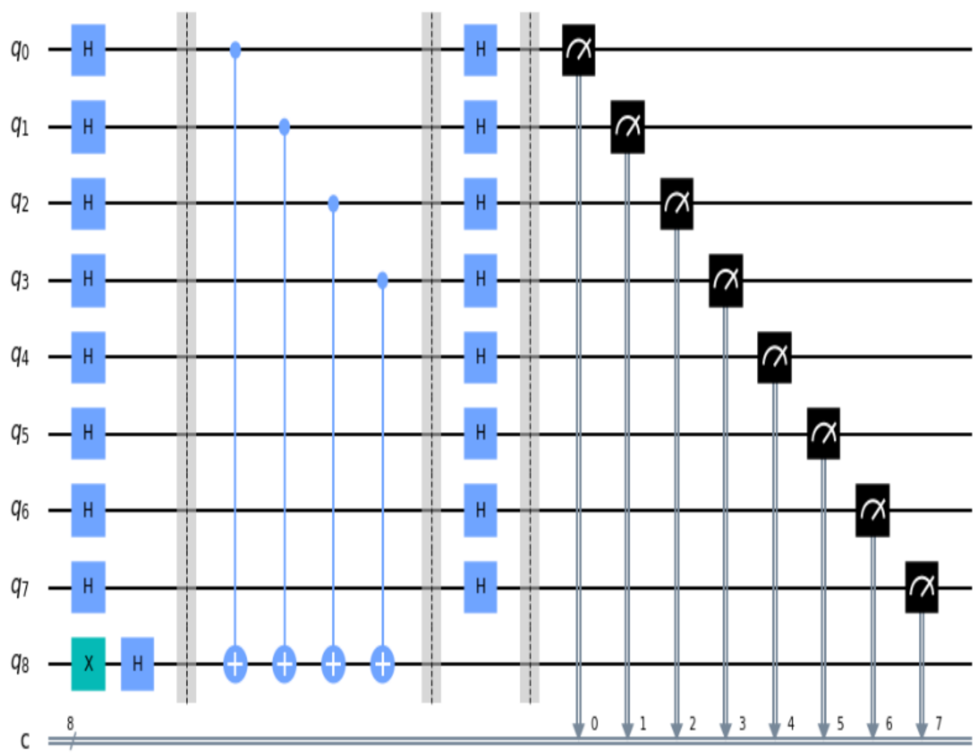


Figure 5.2: Bernstein-Vazirani Algorithm Circuit

5.4 2-Bit Simple Swapping Circuit

In this circuit 2 Qubits have been used. We have applied Hadamard Gates to both Qubits. After that we are measuring both Qubits and storing them in 2 classical registers. Then we use If condition based on, Measuring of 2nd Qubit that is stored on classical register. If candition show whether both Qubits are equal or not. Then we use Swap Gate to swap the Qubits on Condition if they are not equal. In the end we measured both Qubits again and stored on same classical registers to confirm that they are swapped or not.

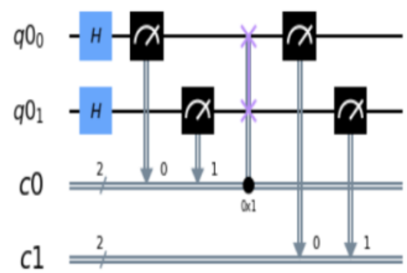


Figure 5.3: 2-Bit Simple Swapping Circuit

Chapter 6

IBM Quantum Experience

6.1 IBMQ

IBM has provided the world a global online platform to run Quantum Algorithms on Actual Quantum Computers they named it IBMQ experience. It's an online platform that provide access to IBM's prototype quantum processors via the Cloud, an online internet forum for discussing quantum computing relevant topics, a set of tutorials on how to program the IBM Q devices, and other educational material about quantum computing. It is an example of cloud-based quantum computing.This service can be used to run algorithms and experiments, and explore tutorials and simulations around what might be possible with quantum computing.

We can interact with Quantum Computer using this Cloud based server here you can interact with Quantum Processors by building Quantum Circuits using Quantum gates applied on Qubits in order to compose Quantum Alogrithms and Run it on Actual Quantum Computers. User can use the following methods to interact with Quantum Computers.

- Circuit Composer
- Quantum Assembly language
- Qiskit Python

6.2 Quantum Circuit Composer

IBM provided online circuit Composer in which all the Quantum Gates can be found along with Qubits. You can assign Qubits to the circuit and than apply Quantum Gates to compose the circuit than you can save the circuit and send it to any available server of Quantum Computer consisting on different number of Qubits.Applying Quantum gates to Qubit is very easy all you have to do is just drag and drop that Gate to the desired Qubit. Once the Circuit is send to the server it's placed inside a queue of Jobs as it's a Universal server, wait until your job is being run on the Quantum Computer, Once the Job is Done. You can see your circuits result.

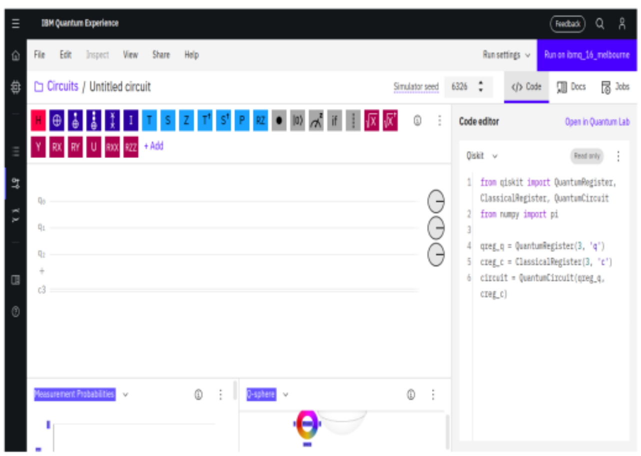


Figure 6.1: Circuit Composer

6.3 Quantum Assembly language

In case you love coding you can use also Quantum assembly language instead of drag and drop you can code Quantum algorithms. IBMQ also provide you with different Quantum Algorithms that you can study and implement on IBMQ.

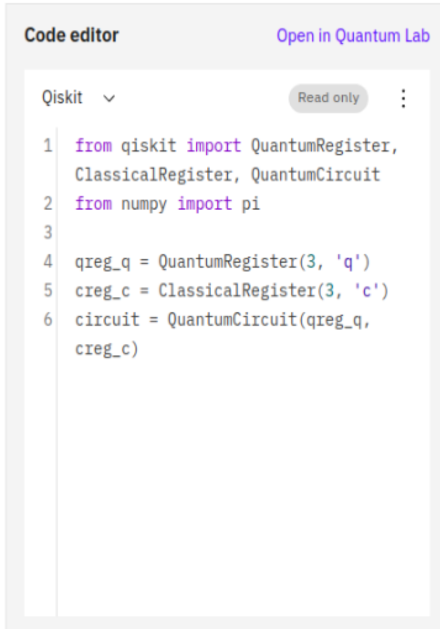


Figure 6.2: Quantum Assembly language

6.4 Qiskit Python

You can also use Qiskit Python to code these Quantum Circuits using Jupyter Notebook Just by using the command of "pip install Qiskit", moreover you can send these cirucits to Run over IBMQ by using Your Token ID provided By IBMQ.

Figure 6.3: Qiskit Python

Chapter 7

Results

7.1 1-Bit Comparator

1-Bit Comparator is the core part of our project. In the circuit below we have used 4 Qubits, Applied X-Gate and CX-Gate, Given input on Qubit 1 and 2, The circuit will return us values of 0,1,2. 0 incase both inputs are equal, 1 incase input-1 is greater than input-2, 2 incase input-2 is greater than input-1. The results are measured from Qubit 3 and 4 applied Z-Measurment Gate.

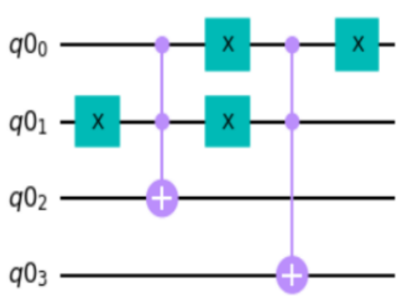


Figure 7.1: 1-Bit Comparator

		Input 2: 0	Input 2: 1
Input 1	0	0	2
	1	1	0

7.2 2-Bit Comparator

In 2-Bit Comparator we are using 9 Qubits. In which (0 and 5) Qubits are used for input of 1st number (i.e. 00 = 0, 11 = 3). For 2nd number we use (1 and 6) Qubits. Then we converted 1-Bit Comparator to a Gate named circuit7(in the fig.) and apllied that gate to Qubits (0.1,2,3) and (5,6,7,8). Then X-Gate is applied to Qubits (2 and 3) and a CX-Gate is applied to Qubits(2 and 3) controlled by Qubit (4), And then two more CX-Gates to the Qubits(4 and 8) and (4 and 7) controlled by Qubits(3 and 2) respectively. Then we applied again X-Gate to the Qubits(2 and 3), After that we measure these 2 Qubits to get the result of camparision. This circuit Showes the result in the form of (0,1,2) For equal numbers it show 0 and If 1st Number is Greater then 2nd then it show 1 and than 2 if 2nd is greater.

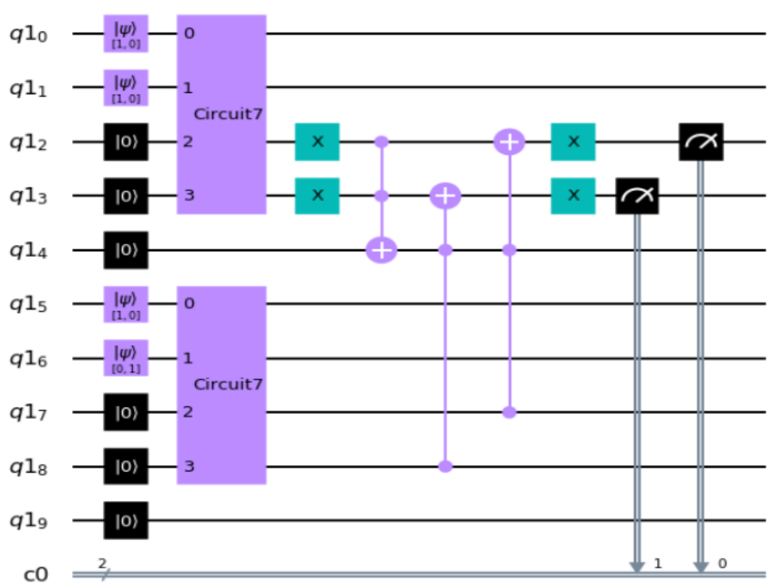


Figure 7.2: 2-Bit Comparator

		Input 2			
		0	1	2	3
Input 1	0	0	2	2	2
	1	1	0	2	2
	2	1	1	0	2
	3	1	1	1	0

Table 7.1: 2-Bit Comparator Matrix Output

7.3 3-Bit Comparator

In 3-Bit Comparator the input Qubits for 1st number are (0,5 and 10) and for 2nd number (1,6 and 11) in the form of (101 = 5 or 111 = 7). Same 1-bit comparator circuit converted to Gate named as circuit10(in the fig.) applied to the Qubits to compare these numbers Qubit by Qubit and some X-Gates, CX-Gates and Z-measurment Operations are used to get the comparison results as you can see in the circuit below. This circuit shows the result in the same way that is explaind above.

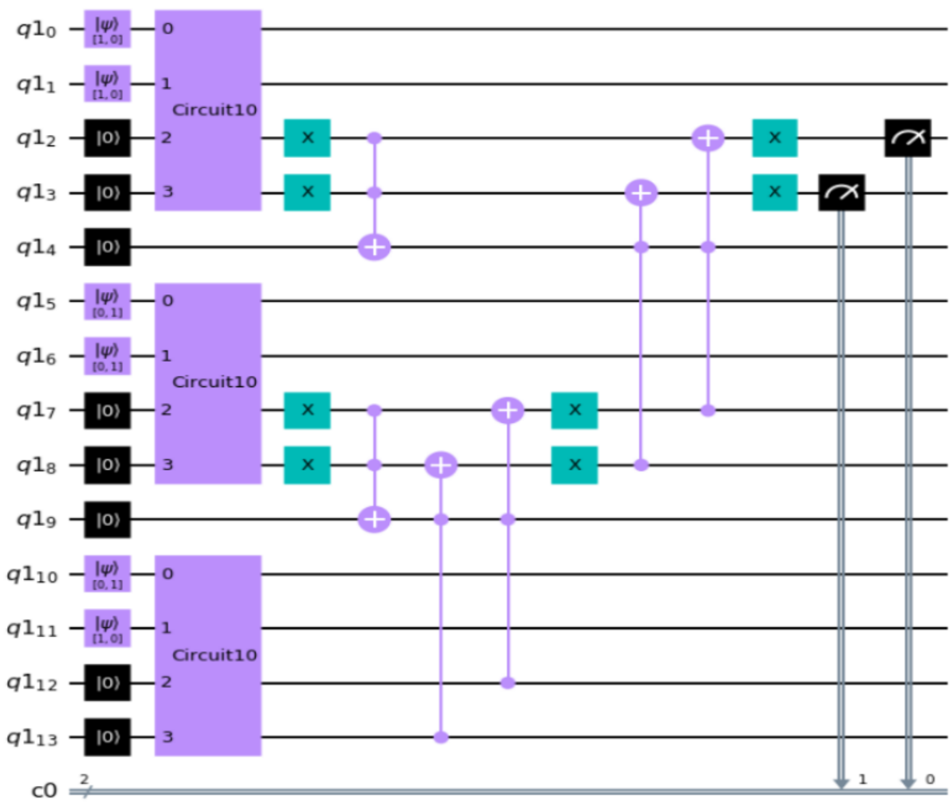


Figure 7.3: 3-Bit Comparator

		Input 2							
		0	1	2	3	4	5	6	7
Input 1	0	0	2	2	2	2	2	2	2
	1	1	0	2	2	2	2	2	2
	2	1	1	0	2	2	2	2	2
	3	1	1	1	0	2	2	2	2
	4	1	1	1	1	0	2	2	2
	5	1	1	1	1	1	0	2	2
	6	1	1	1	1	1	1	0	2
	7	1	1	1	1	1	1	1	0

Table 7.2: 3-Bit Comparator Matrix Output

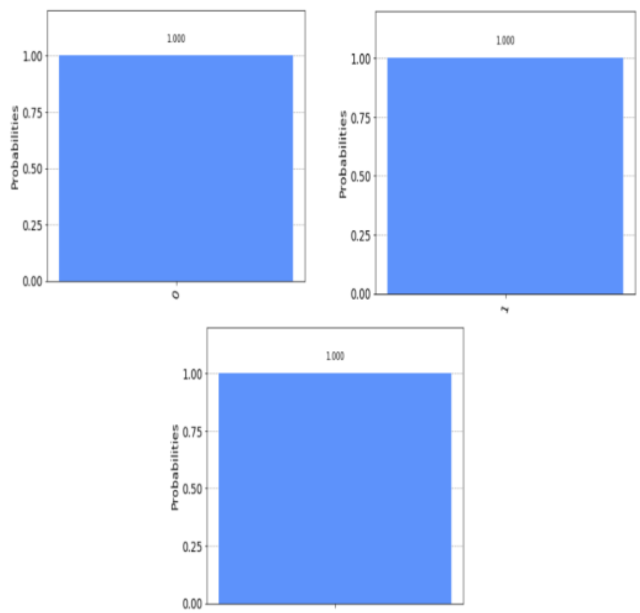


Figure 7.4: Probablity of getting (00,01,10) 0,1,2

7.4 Extended CSwap Circuit

In the previous chapters we studied some Quantum Algorithms and understand there working. We studied 1-Bit Comparator to Compare two Qubits using that 1-bit comparator by extended it to 3-bit Comparator and than using its result to swap the Numbers using CSwap gate to swap the Qubits in betweeen,In order to sort them.After swapping we have placed another 3-bit Comparator just to check if the qubits are being swapped or not and the results we get proves that Qubits are being swapped as before swapping we were getting 1 and after swapping we got 2. Hence Numbers are being swapped. The results we obatin from these comparator circuits are in the form of (00,01,10) in binary 00 if compared Qubits are equal, 01 in case Input-1 is greater than Input-2, 10 in case Input-2 is greater than input 1. As in Extended CSwap we use two bit comparator so it compares 4 qubits at a time, In order to sort more than two numbers we need more Extended CSwap Gates.

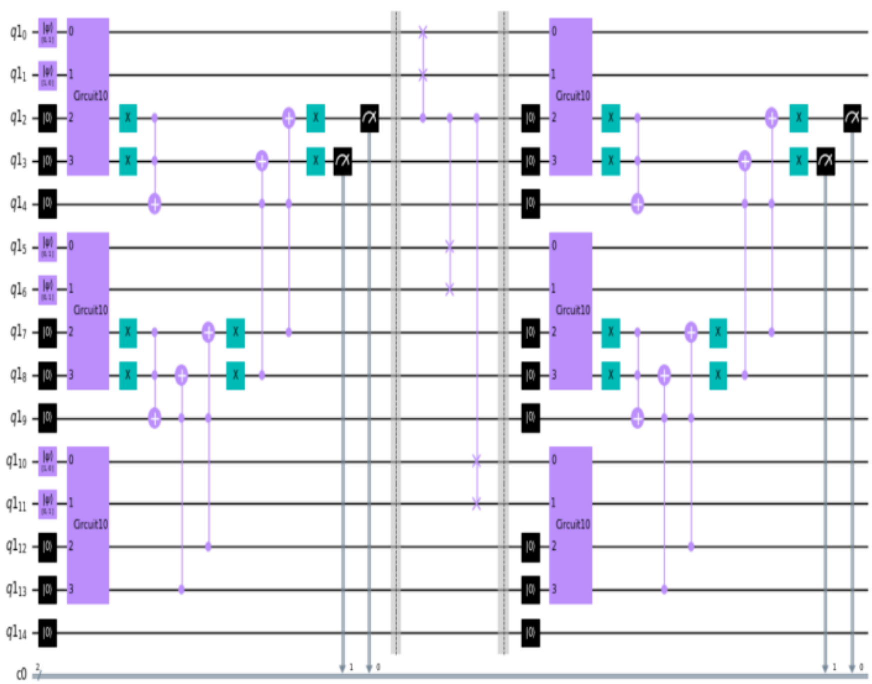
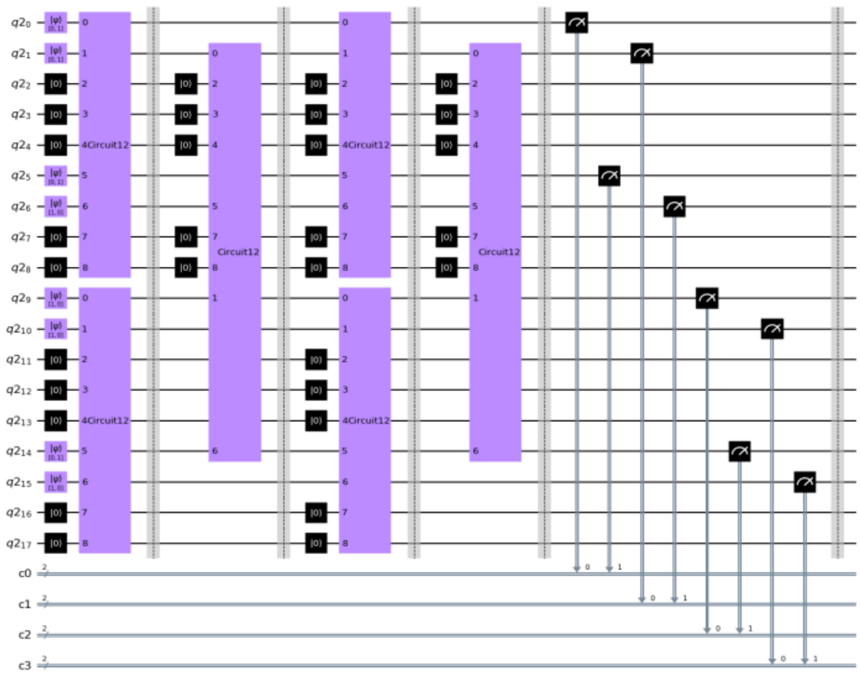


Figure 7.5: Extended CSwap

7.5 Sorting Circuit

Converting the above circuit into 2 Qubit circuit and to the gate named circuit12(in the fig.). We can sort four numbers (00,01,10,11). And in the result we get sorted numbers.



{'11 01 10 00': 1}

Figure 7.6: Sorting Circuit

7.6 Conclusion

Quantum Computers are rare for now but they carry huge Computation power along them based on Quantum Physic, Quantum Mechnaics laws But currently they are in their early age just like Classical Computers were Hopefully with the passage of time we will be able to witness phenomenas like teleportation, Quantum computers are believed to be able to solve certain computational problems, such as integer factorization (which underlies RSA encryption), substantially faster than classical computers.We know that they will be faster for many computational tasks, from modeling nature to searching large amounts of data. I think there are many more applications and, perhaps, the most important ones are still waiting to be discovered.

Sorting Circuit for Quantum Computer

ORIGINALITY REPORT

8%	7%	2%	4%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	en.wikipedia.org Internet Source	2%
2	www.nature.com Internet Source	1%
3	wikimili.com Internet Source	1%
4	www.ias.edu Internet Source	1%
5	engineers-channel.blogspot.com Internet Source	1%
6	Submitted to Melbourne Institute of Business and Technology Student Paper	1%
7	V. P. Gerdt, A. N. Prokopenya. "Some algorithms for calculating unitary matrices for quantum circuits", Programming and Computer Software, 2010 Publication	<1%
	gkready.com	

8	Internet Source	<1%
9	Casey R. Myers, Marcus Silva, Kae Nemoto, William J. Munro. "Stabilizer quantum error correction with quantum bus computation", Physical Review A, 2007 Publication	<1%
10	genomebiology.biomedcentral.com Internet Source	<1%
11	researcher.watson.ibm.com Internet Source	<1%
12	Vladimir P. Gerdt. "A Mathematica Package for Simulation of Quantum Computation", Lecture Notes in Computer Science, 2009 Publication	<1%
13	Igarashi, . "Quantum Computing", Computing, 2014. Publication	<1%
14	en.unionpedia.org Internet Source	<1%
15	arxiv.org Internet Source	<1%

Exclude quotes	On	Exclude matches	Off

Exclude bibliography On