IJCSE International Journal of Computer Sciences and Engineering Open Access

Research Paper Vol.7, Special Issue.1, Jan 2019 E-ISSN: 2347-2693

Quantum Computers

Megha Roy[1*], Soumen Santra[2], Sarasij Majumdar[3]

[1,2,3]Bachelor of Computer Application, Techno International New Town, Kolkata, India

Corresponding Author: megroy1997@gmail.com,

Available online at: www.ijcseonline.org

Abstract- Computers have become an indispensible part of our life. Advancements in the field of technology have led to smaller and more compact computers, while their processing power has increased. However, with parts getting smaller and more compact, quantum physics are making things a bit problematic. Transistors act like switches, either allowing electrons to pass through or blocking their passage. Transistors are already almost as small as atoms. Electrons will eventually be able to transfer themselves to the other side of the blocked passage via a process called *quantum tunneling,* making transistors useless. Classical computers and the way it works stop making sense in the quantum realm. A possible solution to this problem could be quantum computers, which takes advantage of quantum principles to work. In this paper, an overview of quantum computers, their working principles and the basics of the math behind such computers have been discussed. The future scope of such a computer, problems associated with implementing it as well as the problems to be dealt with in case a scalable quantum computer is built is also provided.

Keywords- Quantum Computer, Qubit, Quantum Gates, Decoherence, CNOT, Hadamard Gate, Pauli Gates, Phase Gates, SWAP Gate, Entanglement, Quantum Algorithm, Quantum Fourier Transform

I. INTRODUCTION

For the sake of abstraction and simplification of advanced computing systems, we started leaving behind many laws of physics a long time ago. However, scientists are now realizing that classical computers are just not efficient enough to support and solve many complex problems. They are realizing the need to reinvent the way computers work to solve such problems. This is where quantum computers come in handy. In the early 1980s, Richard Feynman and Yuri Manin were two scientists among many others to propose the idea of a quantum computer [1, 8].

Quantum computer is a kind of computer that uses the principles of quantum mechanics to its advantage to solve complex problems. Instead of using traditional bits, as done by classical computers, quantum computers use *qubits*. In a traditional computer, a bit may have a value of either ‘0’ or ‘1’ at a time. If zero and one are to be stored, two individual bits are required- one to store 0 and the other to store 1. A qubit, on the other hand, to put it simply, may have both the value of ‘0’ and ‘1’ at the same time, until it is measured, at which point it chooses to have the value of either ‘0’ or ‘1’. This basic working principle of quantum computers takes advantage of the principle of ‘Superposition’ in Quantum Mechanics [4-6, 7, 9-10].

SUPERPOSITION

To understand what quantum superposition is, we may look at the famous thought experiment known as “Schrödinger’s cat”. The experiment suggests that a cat is put in a steel box with a tiny amount of radioactive sample that has 50% probability of killing the cat. While the box is closed, observers cannot definitively say if the cat is dead or alive. However, when the box is opened and the observers look inside, the cat will be recorded as either dead or alive. Thus, while the box is covered, the cat is both dead and alive for it to be in all states it could possibly be in. This is called superposition. Superposition collapses into one of the two definitive states- dead or alive- when the cat is observed.

Similarly, a qubit can be in a superposition of either ‘0’ or ‘1’ when it is unobserved. When it is measured, it has to choose to be in either the ‘0’ configuration or ‘1’ configuration. The quantum ‘0’ and ‘1’ are represented by vectors which are shown in the following Dirac notation:

$$|1\rangle = \begin{pmatrix}0\\1\end{pmatrix} \qquad |0\rangle = \begin{pmatrix}1\\0\end{pmatrix}$$

The linear combination of $|0\rangle$ and $|1\rangle$ represents the state of the qubit which is:

$$|\Psi\rangle = x|0\rangle + y|1\rangle\text{, where } |\Psi\rangle \text{ is a qubit state} \qquad (1)$$

This is revolutionary in the field of computing. In a classical computer, n bits could have 2^n configurations out of which only one value could be stored at a time. In quantum computers, we can store 2^n configurations of bits using n qubits at the same time, providing exponential growth with each added qubit. Thus, if we have 20 qubits in superposition, we can store 2^{20} configurations at the same time, which is 1,048,576 numbers of different configurations. Using only 20 qubits, we can store more than a million values in parallel.

Measuring a qubit in superposition takes it out of the superposed state and gives an output where the qubit has either the value of '1' or '0'. This output is not provided randomly. The probability of getting either '0' or '1' depends on the original configuration of the qubit in superposition. The qubit state depends on the *probability amplitude* which is determined by the coefficient of the corresponding vector. The probability amplitude is the quantity which, when squared, gives the probability of obtaining a certain quantum state. Thus, from equation (1), we have:

$$|x|^2 + |y|^2 = 1 \qquad (2)$$

As a visual example, a tossed coin is considered- the coin could either be in heads or tails when it is spinning. When it is measured, it gives us heads if it was more inclined towards heads or it gives us tails if the coin was inclined towards tails. The measurement of qubits made by quantum computers is probabilistic.

ENTANGLEMENT

Another quantum principle that quantum computers take advantage of is Entanglement. Entanglement suggests that two qubits can have a connection that makes one qubit store a certain value depending on the value of the other qubit no matter how far apart they may be. Thus, if one qubit is measured, we will automatically know the value of the entangled qubit without measuring it. Entangled qubits can be used to execute quantum teleportation. Entanglement is used to effectively execute the ability to communicate 'faster than light'. Let us consider Alice and Bob having two entangled qubits in the state $(|00\rangle + |11\rangle)/\sqrt{2}$. If Alice measures her qubit, it would either have a 50% chance of collapsing to $|00\rangle$ and 50% chance of collapsing to $|11\rangle$. No matter how far Bob's qubit may be, it would give the value of either $|00\rangle$ with a 50% chance or $|11\rangle$ with a 50% chance. This provides instantaneous measurement of two qubits by the measurement of just one entangled qubit [10].

To understand the working principle of quantum computers as a whole, we look at a seating optimization problem. To seat 10 people at a table, there are 10! ways to do so. In a quantum computer, some qubits are taken in superposition to take into consideration all the different seating configurations at the same time. A phase is applied to each of the configurations. Using the concept of interference, microwave pulses are used to amplify the phase of some answers, as waves that are in phase result in 'constructive interference' and cancelling out the phase of some answers, as waves that are out of phase result in 'destructive interference'. Eventually, we arrive at the required optimum answer.

II. BITS AND QUBITS

As discussed before, traditional bits are only capable of representing either value 0 or 1 at a time. Quantum bits- qubits- may hold any linear combination of $|0\rangle$ or $|1\rangle$ to represent its superposition state [2]. However, at the time of measurement, qubits collapse into the fundamental states of either 0 or 1. Thus, as long as a qubit is not measured, it holds its probability amplitudes of superposition state. As soon as it is measured, the qubit state collapses to the fundamental states of 0 or 1, resembling a traditional bit.

III. QUBITS- A DEEPER LOOK

We know that a qubit is a two-level quantum mechanical system, i.e., it can exist in the quantum superposition of either one or another independent and differentiable quantum state. Now, we look at how a qubit is made and how information is written and read to and from a qubit. Scientists are using many different ways to make a qubit out of which two ways of its physical implementation are listed below:

Table 1: Qubit implementation

Physical implementation	Information Support	$\|0\rangle$	$\|1\rangle$
Electrons	Spin	Up	Down
Nucleus	Spin	Up	Down

Electron as a qubit

There are countless different ways in which qubits can be physically implemented and newer implementations are always emerging. Some researchers are using the outermost electron of a phosphorous atom as a qubit. This electron is embedded in a silicon crystal, next to a transistor. The electron has a magnetic dipole which is called its *spin*. Its two configurations of *spin up* and *spin down* correspond to the two classical states of '1' and '0'. To differentiate between the energy states of the electron, we would require a superconducting magnet to apply a strong magnetic field. Thus, when the electron is in spin down state, it is its lowest energy state. To put it into the spin up state, it would take a certain amount of energy. But an electron, at room temperature, has enough thermal energy to bounce between its spin up and spin down states. Thus the whole apparatus is cooled down to a temperature which is only a few hundredths of a degree above absolute zero. Thus, there would not be enough thermal energy in the surrounding to flip the electron the other way. Information is written onto the electron by using a pulse of microwaves. If the microwaves are applied at a certain frequency, we can create a superposition of spin up and spin down states.

There are countless other ways in which qubit can be implemented. The vertical and horizontal polarization of light, Josephson-junction used by IBM, nucleus spin of a phosphorous atom etc. can be used to implement qubits.

IV. DECOHERENCE

Qubit retains the information written onto it by keeping itself isolated from the rest of the universe. Qubits are often in superposition states and measuring a qubit forces its superposition to collapse into either state '0' or '1'. If the qubit is not isolated enough, outside influences can cause the qubit to lose the information it holds, leading to decoherence. Thus, it is important to shield the qubit form the rest of the universe and make sure it is isolated. To deal with the threat of quantum decoherence, scientists are formulating error-correcting algorithms to withstand a certain amount of loss of information. As the number of qubits in a quantum system is increased, the error rate due to decoherence also increases. The goal, however, is to move towards creating a system with increased number of qubit along with a decreased error rate.

V. MATH BEHIND QUANTUM COMPUTERS

For any device, it isn't sufficient to have the units of data by themselves. The device must allow the user to manipulate the units of data to provide some meaningful information. In a classical computer, we use logic gates like AND, OR, NOT, XOR gates to operate on bits of information and produce some meaningful output. Similarly in Quantum Computing, there are many *quantum gates* at our disposal which can be used to manipulate qubit states and get some output [6, 10]. Before we jump into the math behind quantum computers, we need to understand what kind of computations can be done on such computers [6].

In a classical computer, there are four basic operations that can be done on one bit-

- Constant 0- No matter what the input, the output is always 0
- Constant 1- No matter what the input, the output is always 1
- Negation- The output is the flipped value of the input, i.e., the output is 0 if the input is 1 and vice versa
- Identity- The output is the same value as the input, i.e., if the input is 1, the output is 1.

Two of the above operations- constant-0 and constant-1- are irreversible operations, while negation and identity are reversible operations. Reversible operations refer to those operations which when applied twice to the same input simultaneously, gives us the value of the input. It can be seen that identity and negation are clearly reversible; meanwhile the other two operations are not. Quantum computers can only use reversible operations to manipulate the qubits. Thus, all quantum logic gates are their own inverses, i.e., if the operators are applied twice in succession, they provide us the original input values.

Representing multiple bits

A qubit can be represented in the format:

$|\Psi\rangle = \binom{a}{b}$ where $|a|^2$ is the probability of the qubit to collapse to value 0 and $|b|^2$ is the probability of the qubit value to collapse to 1, and $|a|^2 + |b|^2 = 1$.

Multiple qubits can be represented by their tensor product. To provide a brief overview of tensor products, we see the following example:

We take two individual qubits. One has probability amplitude $x|0\rangle + y|1\rangle$ and the other qubit has probability amplitude $p|0\rangle + q|1\rangle$. The tensor product of the qubits would be:

$$\binom{x}{y} \otimes \binom{p}{q} = \begin{pmatrix} xp \\ xq \\ yp \\ yq \end{pmatrix}$$

where $|xp|^2 + |xq|^2 + |yp|^2 + |yq|^2 = 1$, that is, the qubits have $|xp|^2$ probability to collapse to $|00\rangle$, $|xq|^2$ probability of collapsing to $|01\rangle$, $|yp|^2$ probability of collapsing to $|10\rangle$, $|yq|^2$ probability of collapsing to $|11\rangle$.

Thus, n bits can be represented by the tensor product of the probability vector of n qubits which will result in 2^n values in the resultant column matrix.

CNOT gate

CNOT gate or Controlled NOT gate is an important logic gate in quantum computing. Its basic use is that in a two bit system, it uses one bit (the most significant bit) as the control bit and the other bit (the least significant bit) as the target bit. If the control bit is 0, it leaves the target bit unchanged. If the control bit is 1, it flips the value of target bit. Similarly, using two qubits, the CNOT gate does the following:

Table 2: Working of CNOT Gate

Input		Output	
Control	Target	Control	Target
$\|0\rangle$	$\|0\rangle$	$\|0\rangle$	$\|0\rangle$
$\|0\rangle$	$\|1\rangle$	$\|0\rangle$	$\|1\rangle$
$\|1\rangle$	$\|0\rangle$	$\|1\rangle$	$\|1\rangle$
$\|1\rangle$	$\|1\rangle$	$\|1\rangle$	$\|0\rangle$

The transformation matrix of the CNOT gate looks like this:

$$C = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}$$

As an example, we take $|10\rangle$ and apply the CNOT gate:

$$C|10\rangle = C\left(\begin{pmatrix} 0 \\ 1 \end{pmatrix} \otimes \begin{pmatrix} 1 \\ 0 \end{pmatrix}\right)$$

$$= \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} 0 \\ 0 \\ 1 \\ 0 \end{pmatrix}$$

$$= \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix} = |11\rangle$$

Hadamard Gate

As mentioned before, we can use the principle of superposition to get 2^n values using n qubits at the same time. But how can qubits be put into a superposed state? Such a state is achieved by putting qubits through a gate called Hadamard gate. The Hadamard gate takes input $|0\rangle$ or $|1\rangle$ and puts them in exactly equal superposition. The Hadamard gate is represented by the following transformation matrix:

$$H = 1/\sqrt{2} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}$$

Using this gate, we can put a qubit into superposition. If we use two Hadamard gates in succession, they provide us with the original input.

Pauli Gates

Pauli gates, named after Wolfgang Pauli, are based on the Pauli spin matrices which calculate the spin of an individual electron. Pauli-X gate is the corresponding quantum gate to the classical NOT gate. There are 3 types of Pauli gates, Pauli-X gate, Pauli-Y gate and Pauli-Z gate. The following are the transformation matrices for each gate:

$$\text{Pauli-X} = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$$

$$\text{Pauli-Y} = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix}$$

$$\text{Pauli-Z} = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$$

Phase Shift Gates

Phase Gate leaves the basis state of $|0\rangle$ as it is, and changes the state of $|1\rangle$ to $e^{i\phi}|1\rangle$. This gate basically shifts the quantum state of the qubit in a horizontal circle on the Bloch sphere by ϕ radians.

$$R_\phi = \begin{pmatrix} 1 & 0 \\ 0 & e^{i\phi} \end{pmatrix} \quad \text{where } \phi \text{ is the phase shift.}$$

There are other gates such as $\pi/8$ or T gate where $\phi = \pi/4$.

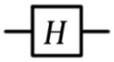
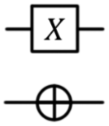
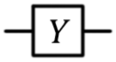
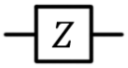
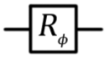
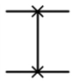
SWAP Gate

As the name suggests, the swap gate swaps two qubits. With respect to the qubits $|00\rangle$, $|01\rangle$, $|10\rangle$, $|11\rangle$, we have the swap gate matrix as follows:

$$\text{SWAP} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

The circuit symbols of the given gates are given below[12]:

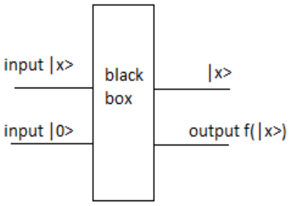
Table 3: Quantum Gates

Gate	Circuit Symbol
Hadamard	
Pauli-X	
Pauli-Y	
Pauli-Z	
Phase Shift	
SWAP	

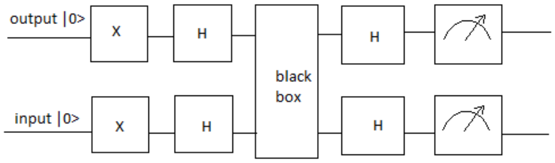
There are many other quantum gates that are used to execute operations on a quantum computer such as the Toffoli gate etc. The Toffoli gate is one of the universal gates which can be used to create bigger circuits. However, to execute certain operations such as quantum entanglement of bits, it is quite enough to have gates like CNOT, Hadamard, Pauli gates, Phase shift gates and SWAP gate.

VI. QUANTUM ALGORITHM

There are countless quantum algorithms but the algorithm that provides the basis for all other quantum computing algorithms is the Deutsch oracle or Deutsch algorithm. But to understand the Deutsch oracle, we have to see how to write irreversible operations in a reversible manner. To do so, we use the following process, where the black box could be any of the four basic operations:



On a quantum computer, we can use one single query to infer whether or not the black box contains a constant operation such as constant 0 or 1 or it contains a variable function such as identity and negation. We do so by using the Deutsch oracle. The Deutsch oracle says that if the black box contains a constant function, the qubits will be in state $|11\rangle$ after measurement and if the black box contains a variable function, the qubits will be in state $|01\rangle$ after measurement. The circuit that represents the Deutsch oracle is given below:



Such an algorithm is important to the study of quantum computing algorithms because to find such properties of a black box function, it takes classical computer 2 queries whereas it takes the quantum computer only 1. The generalized form of this Deutsch Oracle is the Deutsch-Josza problem which leads into the study of other algorithms like Shor's algorithm [10].

VII. QUANTUM FOURIER TRANSFORM

Quantum computers provide an efficient way to implement the solutions to the problems that require enormous amount of resources in a classical computer. The quantum algorithms that abide by this promise are divided into two broad classes, one of which is based on Shor's quantum Fourier transform that provide exceptional quantum speedup compared to classical algorithms. These algorithms can be used to solve discrete logarithm and factoring problems. QFT (short for Quantum Fourier Transform) is a linear transformation on qubits. The Fourier transform of n-qubits can be represented as a tensor product of n qubits. In discrete Fourier transform, an input of n vectors x_0, x_1,……,x_{n-1} provides an output of n vectors y_0, y_1,……,y_{n-1}. The QFT is similar as well with slight differences in the notation:

$$|j\rangle \longrightarrow \frac{1}{\sqrt{2^n}} \sum_{k=0}^{2^n-1} e^{2\pi ijk/2^n} |k\rangle$$

where transformation is done on an orthonormal basis of $|0\rangle$,…..|N-1>. QFT is used in Shor's algorithm to find the periodicity p which is then used to find the GCD of $x^{(p/2)}+1$ and N and the GCD of $x^{(p/2)}-1$ and N where x<N, x and N are co-prime and N is the number to be factorized. This gives us the factors of N, thus, completing Shor's algorithm. Therefore, we can see that QFT is an essential step of Shor's algorithm.

VIII. FUTURE SCOPE AND POWER OF QUANTUM COMPUTERS

Small scale quantum computers do already exist. IBM has a 5-qubit quantum computer available to be used for free via the cloud [1]. The problem with building a system with more qubits is that error rates increase exponentially with each new qubit as the system is a probabilistic one. However, scientists are looking into algorithms with may reduce quantum error rates and increase efficiency of quantum computers.

Entanglement, or as Einstein called it, "spooky action at a distance", thins the line between science and science fiction as Teleportation becomes a reality by transporting the information state from one entangled qubit to another no matter how far apart they may be. This may also lead to the implementation of quantum communication.

Quantum supremacy or quantum advantage refers to the ability of quantum computers to solve problems that classical computers are not capable of solving. This means that quantum computing algorithms can provide incredible speedup compared to the best known classical algorithm. Although the term was popularized by John Preskill, the concept of quantum supremacy was proposed by Yuri Manin and Richard Feynman [11]. Quantum algorithms such as Shor's and Grover's algorithms can finally be implemented efficiently. Shor's algorithm provides superpolynomial speedup over the best known classical algorithm, thus, providing quantum supremacy. However, if Shor's algorithm is implemented, it would render RSA cryptography useless. Thus, scientists are looking into post-quantum cryptography to deal with such a problem and keep data safe [6].

Quantum computing also makes its efficient to simulate nature effectively. If we want to properly simulate a molecule in a traditional computer, we have to consider each electron in the molecule, its repulsion from every other electron and its attraction toward the nucleus. Supercomputers can simulate small molecules properly. The problem arises when we want to simulate bigger molecules. As the molecule size increases, the number of electrons increases. This increases the complexity of the simulation exponentially as we have to calculate the attraction of each electron to the nuclei and the electrostatic repulsion of each electron with every other electron in the molecule. Even the best supercomputer in existence cannot handle that much data before running out of resources. However, such complex data can be easily stored in qubits and molecules can be simulated effortlessly. This may open up various avenues of discoveries in the fields of chemistry and bio-technology.

Quantum computing has influenced physicists and computer scientists to think about computation as we know it, through the various aspects of physics [7-9, 10].

IX. CONCLUSION

The field of quantum computing is still in its infancy; many scientists who are skeptical of quantum computers believe that due to decoherence, large scale quantum computers may never end up being a reality. However, if quantum computers do come into being, quantum computing could change the way many problems are solved. If and when scalable quantum computers become a reality, it still would not replace classical computers. Quantum computers are efficient only in certain specialized problems which day to day computer users are hardly faced with. Problems such as optimization or simulating a large molecule cannot be done even on the best supercomputer, whereas such tasks are done easily by quantum computers. Quantum computers can solve certain problems exponentially faster than a classical computer, but there are still many problems for which there is no quantum algorithm that is faster than classical algorithms. However, as mentioned earlier, the field of quantum computing is still in its early phases and holds enormous potential for the future. The extent of this new field of technology is yet to be explored.

REFERENCES

[1] *What is quantum computing?*- IBM Q
[2] *Quantum Computers Explained- Limits of Human Technology*- Kurzgesagt- In a Nutshell
[3] *A Beginner's Guide to Quantum Computing*- IBM Research
[4] *Quantum Computing for Computer Scientists*- Microsoft Research
[5] *Building the Bits and Qubits*- Frame of Essence
[6] *Quantum Logic Gate*- Wikipedia
[7] *Demystifying Quantum Gates- One Qubit At A Time*- Towards Data Science
[8] *How Does a Quantum Computer Work?*- Veritasium
[9] Siddhartha Kasivajhula- *Quantum Computing- A Survey*
[10] Michael A. Nielsen & Isaac L. Chuang-*Quantum Computation and Quantum Information*
[11] *Quantum Supremacy*- Wikipedia
[12] Microsoft- *Quantum Gates and Circuits: The Crash Course*