THESIS TITLE

DESIGN AND ANALYSIS OF QUANTUM COMPUTING BASED KEY DISTRIBUTION ALGORITHMS

STUDENT NAME: Zara Shahid

ENROLLMENT NO. 01-245182-010

A thesis submitted in fulfilment of the

requirements for the award of degree of

Masters of Science (Telecom and Networking)

Department of Computer Science

BAHRIA UNIVERSITY ISLAMABAD

NOVEMBER 2020

THESIS COMPLETION CERTIFICATE

Scholar's Name: Zara Shahid Registration Number: 59408

Enrollment: 01-245182-010 Program of Study: MS Telecom and Networking

Thesis Title: Design and analysis of quantum computing based key distribution algorithms.

It is to certify that the above scholar's thesis has been completed to my satisfaction and, to my belief, its standard is appropriate for submission for examination. I have also conducted plagiarism test of this thesis using HEC prescribed software and found similarity index __13__%. that is within the permissible limit set by the HEC for the MS/M.Phil. degree thesis. I have also found the thesis in a format recognized by the BU for the MS/M.Phil. thesis.

Principal Supervisor Signature: _____

Date: _____30.11.2020_____

Name: ___Dr. Najam-ul-Islam_____

AUTHOR'S DECLARATION

I, Zara Shahid, hereby state that my Master thesis titled "Design and analysis of quantum computing based key distribution algorithm" is my own work and has not been submitted previously by me for taking any degree from this university (Bahria University Islamabad) or anywhere else in the country/world.

At any time if my statement is found to be incorrect even after my Graduation the university has the right to withdraw/cancel my PhD degree.

Name of student: Zara Shahid

Date: 4th September, 2020

UNDERTAKING

I, solemnly declare that research work presented in the thesis titled" Design and analysis of quantum computing based key distribution algorithms" is solely my research work with no significant contribution from any other person. Small contribution/help wherever taken has been duly acknowledged and that complete thesis has been written by me. I understand the zero tolerance policy of the HEC and Bahria University towards plagiarism. Therefore, I as an Author of the above titled thesis declare that no portion of my thesis has been plagiarized and any material used as reference is properly referred / cited.

I undertake that if I am found guilty of any formal plagiarism in the above titled thesis even after award of MS/M.Phil degree, the university reserves the right to withdraw / revoke my MS/M.Phil degree and that HEC and the University has the right to publish my name on the HEC / University website on which names of scholars are placed who submitted plagiarized thesis.

Scholar/Author's Sign:

Name of Scholar: Zara Shahid

DEDICATION

I dedicate this thesis to my beloved father, mother and husband.

ACKNOWLEDGEMENTS

As my Master degree will come to an end after submitting the thesis work, I would like to show gratitude to those people who became my great support while working on my thesis work.

First of all, I want to say thanks to supervisor Professor Dr. Najam-ul-Islam who provided me the guidance and right direction to complete my research on thesis topic efficiently. I would like to express my appreciation to co-supervisor Engr. Madiha Zoheb who was always there to provide assistance whenever I faced a problem in my thesis work. She is the one who makes me enable to reach to the conclusion. My appreciation extends to my family and friends who gave me courage to get through all the difficulties and cooperated with me. My special thanks goes to my father who was my great support. He also supported me financially to achieve my goal. I am also very thankful to the librarian of Bahria University for providing relevant literature. I am grateful to all my class fellows for providing relevant knowledge and support. As it is not possible to list down all the people in this limited space, I want to thank all those people who cooperated with me directly or indirectly to solve the problems encountered in my thesis work.

Table of Contents

LIST OF TABLES

LIST OF FIGURES

LIST OF SYMBOL

Φ – Longitude

Θ – Latitude

+ – Rectilinear bases

X – Diagonal Bases

→ – Zero-degree polarization state

↑ – 90-degree polarization state

↗ – 45-degree polarization state

↖ – 135-degree polarization degree

$\not{\nearrow}$ – Non orthogonal bases

° – Degree

% – Percentage

ABSTRACT

We are living in the new era of computing where internet of things is increasing the connectivity of machine to machine or machine to environment which leads to the massive amount of data over the internet. Hence, securing the data is one of the most important areas of interest these days. The process of cryptography is used to secure the transmission of data from being stolen or intercepted by third party. Up till now classical cryptography techniques based on digital computers were enough to provide integrity, confidentiality and authenticity to the transmitted data but the processing power of transistors used in digital computers is growing fast with the exponential decrease in the transistor size. As stated in Moore's law that the size of transistor will become too small in the coming years that the moving electrons in the transistor will make it very hot and it will not function properly. This will lead to the era of digital computers to come to an end and there will be need of quantum computers and hence quantum cryptography algorithms for securing data. As quantum cryptography is device specific and can only be done by using quantum computers. These quantum computers are large, expensive and complex to use and we do not have access to quantum computers. Considering these limitations, we need some simulation and emulation models that can check the working of key distribution algorithms. The main focus of the proposed research work is to design a resource efficient standalone system that will help to test the performance of quantum cryptography protocols and benefits the future researchers to implement already tested quantum key distribution protocols for making the IoT (Internet of Things) communication reliable and secure.

Chapter 1: Quantum Computing

1.1. Introduction:

Physicist Richard Feynman about three decades ago gave the concept of quantum mechanics that works on laws different than classical physics such as superposition, entanglement, no-cloning theorem etc. He also used these quantum principles to introduce new area of computing called Quantum computing that works in different way as compared to classical computers[1]. . Instead of storing information in bit, quantum computers use the quantum bits known as qubits to store information. Quantum computers use the laws of quantum computing e.g. entanglement, superposition and probabilistic measurement theorem. It offers parallelism to perform operations significantly faster than any classical computer. The technology of quantum computing offers more efficient way of problem solving than what is possible with classical computers.

1.2. Classical computing to quantum computing:

The researchers have worked a lot on decreasing the size of transistor with increasing computational power to make the systems compact. They are inventing new technologies for shrinking the size of transistor so quickly that they have already attained the limit of transistor's scaling that was predicted by Moore. The statement of Moore's law says that the number of transistors will double every 18 years. He predicted that in the next few years, the transistor will shrink to a size of few atoms and become unable to control charge in classical sense due to involvement of quantum mechanical characteristics[2]. According to the law described, transistors are shrinking their size with the emergence of new technologies such as MOSFET (Metal oxide semiconductor field effect transistor), PETs (piezo-electric transistors), TFETs (tunneling field-effect transistors), NTV (near-threshold voltage) and photolithography. This decrease in the

size of transistor leads to the decrease in size of microchips placed in digital computers and it resulted in the conversion of the first generation computers into sophisticated machines containing microchips with more and more transistors making them more powerful and efficient[3]. To improve the performance of microprocessor, developers are scaling down the elements present in circuit so more elements can be packed into the chips and electrons move between them more quickly. This increases the performance but at the same time heating up the system due to movement of electrons. The microprocessor that are used currently have circuit features that are as small as nanometers. This exponential improvement predicts that in some years, transistor size will shrink to the size of 2-3 nanometer limit which will make the chip too small that the electrons had to move faster and chips will begin to get too hot. At that time, the electron's behavior will be controlled by quantum uncertainties that will make the working of transistor unpredictable[4]. This limitation forced the researchers to introduce some new techniques like quantum computing that can replace the transistors in the coming years. This technique promises to speed up the calculations and provide more efficient problem solving techniques to enhance the performance.[5]

Quantum computers perform multiple operations at a time and makes it faster and efficient than the digital computers. The technology of quantum computing offers more efficient way of problem solving than what is possible with classical computers[6]. In a digital computer, the bits are used to transmit information that can

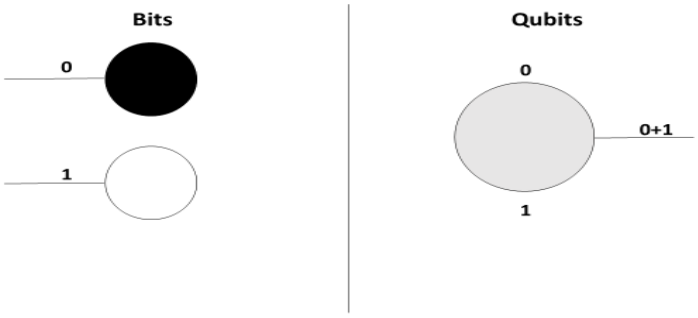

Figure 1. Representation of qubits[7]

be in 0 or 1 state. Whereas in quantum computing, the fundamental blocks used for transmitting data are called $qubits$. $Qubits$ are transmitted through a special quantum channel that can be optical fiber or free space. When transmitted, $qubit$ has a probability of being 1 or 0 but when measured, it collapses to the bit 0 or 1[8]. This behavior of $qubit$ is based on the principle of law of superposition which states that the microscopic objects can be at more than one place simultaneously before measurement. The state in superposition can be described as:

$$[\Psi] = \alpha\,|0\rangle + \beta|1\rangle$$

Where α and β are complex numbers, $|0\rangle$ state represents qubit $\begin{bmatrix}1\\0\end{bmatrix}$ and $|1\rangle$ state represents qubit $\begin{bmatrix}0\\1\end{bmatrix}$. A state of superposition Ψ is the linear combination of state $|0\rangle$ and$|1\rangle$. A qubit cannot be predicted in this state but when it is measured, it will collapse to 0 or 1.

1.3. Applications of Quantum Computers:

Based on the properties of fast handling of large data sets, security and efficiency of quantum computers, they are used for a number of different scientific and business applications. Quantum computers are highly used for big data analysis and error correction in machine learning. Quantum assisted optimization techniques are empowering new machine-learning and artificial-intelligence systems[9]. These systems are improving the control and management of remote sensing systems, renewable power generators and early warning systems. These techniques are also providing aid to dynamic pricing for online services, automation of warehouse and self-driving cars[10]. Modelling chemical reactions and materials is also an example of application of quantum computing. Quantum computing is also improving the quality of products and services in many industries. These quantum technologies are finding its applications

in health care by improving the patient diagnostics[11]. Some applications of quantum computing are defined in table 1.

Table 1. Applications of quantum computing

S. No.	DOMAIN	APPLICATION
1.	Machine Learning	Big data analysis and error correction. Improves remote sensing, renewable power generators and early warning systems. [9]
2.	Industries	Improving quality of products and services. [11]
3.	Financial Modelling	Efficient analysis of behavior of assets to lower the risk. [9]
4.	Medical	Provide faster and efficient diagnostics and delivery of optimal treatment.[11]
5.	Meteorology	Improved pattern recognition to predict weather events.[12]
6.	Security	Quantum cryptography techniques ensures the security of data.[12]

All these domains are using quantum techniques for improving their processes and requires the security of data used in all domains which will be ensured by quantum computing algorithms. Quantum computing has a vast scope in the field of security which leads to the whole new and exciting field of research called “quantum cryptography”. Cryptography has a vital role in today’s communication system because most of the information is communicated over the internet which leads to the expansion of digital data containing different passwords, financial transactions, emails etc. for

which security is required. Cryptography ensures the security of data shared between two parties[13].

1.4. Cryptography:

Cryptography is the process of converting plain text into the scrambled text by using some protocols in order to make it meaningless for the unauthorized person but meaningful for the person who has the right “key” to read it. The original message that needs to be sent is the plain text and the scrambled one is known as cipher text. This process of converting plain text into cipher text is known as encryption and the method

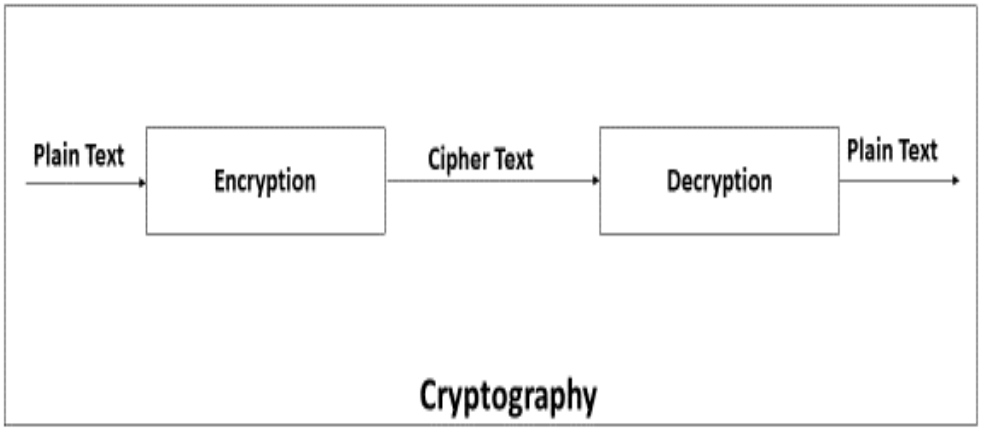

Figure 2. Cryptography

of encryption is called encryption protocol. Whereas, the opposite action i.e. restoring the original message by using key is called decryption[7]. This process makes sure that the encrypted data can be read by only those parties who have shared secret key between them. Hence, securing the key is the fundamental element of cryptography.

The two main categories of cryptography are symmetric cryptography and asymmetric cryptography. Symmetric cryptography is also called single key cryptography. Symmetric cryptography uses a single key for the process of encryption and decryption. Hence, the key needs to be distributed among the authorized sender and receiver. The most popular symmetric cryptography systems are AES (Advanced

Encryption Standard) and DES (Data Encryption Standard). Asymmetric cryptography uses a pair of key: one is public key that is available for everyone and the other is private key that is only known to the user of that key. Public key of receiver is used by the sender to encrypt the data and private key of receiver is used by the receiver to decrypt the data [14]. Most popular asymmetric cryptosystems are RSA and Elliptic Curve Cryptography (ECC) [15].

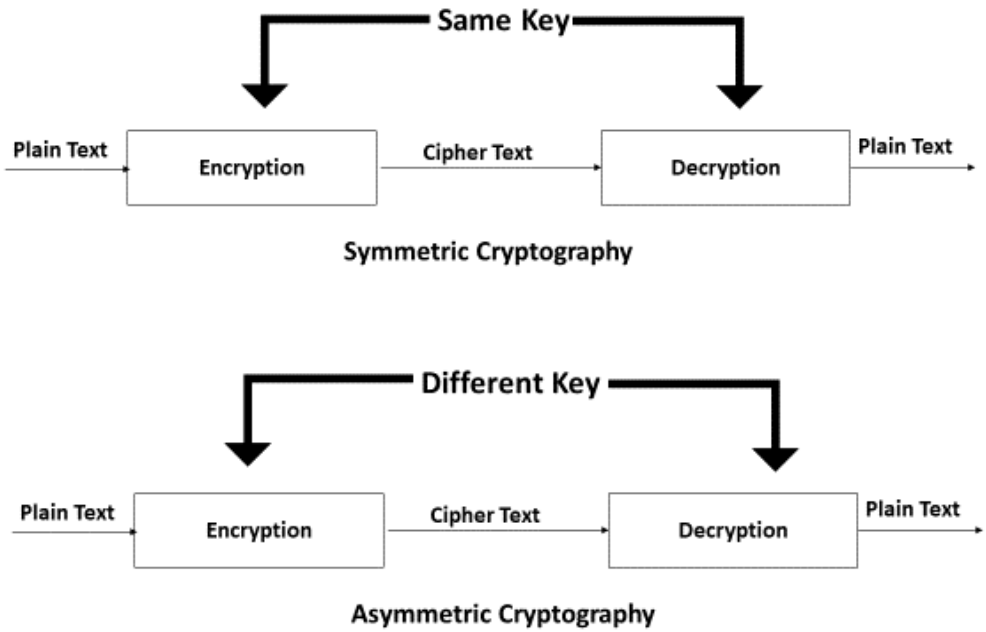

Figure 3. Types of cryptography

1.5. Classical Cryptosystems vulnerable to Quantum Algorithm.

The security of classical cryptography was based on the formation of longer keys that cannot be factorized. One of the main classical cryptography technique was RSA cryptography invented by Ronald Rivest, Adi Shamir, and Leonard Adleman in 1977. It was based on difficulty of factorizing two prime numbers which were used to generate a secret key. Other asymmetric cryptography systems such Elliptic Curve Cryptography (ECC) were based on discrete logarithm problem (DLP). The security of these

cryptosystems were based on the difficulty of determining the integer[15]. These cryptosystems were considered the most efficient before the development of such quantum algorithms that make it possible to solve factorization problem that was the basis of classical cryptography systems.

In 1995, Peter Shor, professor at MIT concluded that the quantum computer can factorize large prime numbers and compute discrete logarithmic problems which results in the collapse of asymmetric cryptography systems[16]. In 1996, Lov Grover introduced Grover's algorithm that uses quantum computer to do unsorted database searches[17]. The algorithm needs only 185 searches to find 56-bit key[15]. Hence, this algorithm defeated the strength of symmetric cryptography systems such as AES and DES[18].

After the introduction of quantum algorithms, it became difficult to secure data by using this technique of cryptography so researchers started focusing on post quantum cryptosystems whose security cannot be harmed by the quantum algorithms. Researcher moved towards the new form of communicating the key securely between sender and receiver. This type of quantum cryptography is known as Quantum Key Distribution (QKD). Conveniently, such form of cryptography already existed as it was introduced in 1984, when Charles Bennett and Gilles Brassard used $qubits$ to employ Wieners coding scheme for distribution of cryptographic keys[15, 19].

1.6. Quantum key distribution:

Quantum cryptography takes the advantage of unusual and unique behavior of small microscopic elements which enables the user to develop the secret keys securely and making the user able to identify the presence of eavesdropper who is trying to steal message that was not possible in the previous cryptography techniques.

QKD algorithms practice some properties of quantum physics to securely exchange the secret key between two users. These properties include:

1. Heisenberg uncertainty principle:

   Heisenberg, A German physicist in 1927 states that we cannot accurately measure the position and momentum of an atomic particle at the same time. It is not possible to know both the properties with the certainty at the same time. This property is not only true for momentum and velocity. It affects all the inter related properties. According to this property, measuring the state of photons will disturb its value and this interference can be detected.

2. No-cloning theorem:

   No-cloning theorem was defined by Zurek, Wooters and Dieks in 1982. This theorem states that we cannot make identical copies of an arbitrary unknown quantum state. This theorem makes it impossible for the eavesdropper to create identical copy of the key that he is trying to steal[19].

3. Entanglement:

   This property states that there is a possibility that two particles get entangled in such a way that if specific property is observed in one particle, an opposite state will be measured on entangled particle simultaneously. The action performed on one particles have an effect on other particle even if both are millions light years away. [20]

4. Superposition:

   Classical physics states that every large object exists in a unique and well-defined place whereas the quantum mechanics tells that the microscopic objects can be at more than one place simultaneously before measurement. This property of being in more than one place at a time is called superposition.[20]

Every protocol has its own working based on the properties of quantum physics on which they are based on but workflow of all the quantum key distribution protocol is same.

1.7. Workflow of Quantum Key Distribution protocols:

The generic model of QKD is based on two parties usually named as Alice and Bob. Alice is considered as the sender and Bob as a receiver. These two parties wish to exchange the secret key securely in such a way that eavesdropper cannot steal the key. For this purpose, they are exchanging the key in the form of $qubits$ so eavesdropper cannot make an identical copy of key exchanged according to no-cloning theorem. Key is exchanged between two parties through quantum channel while other communication takes place through classical channel[21]. The general quantum key exchanged protocol is defined in figure 4.

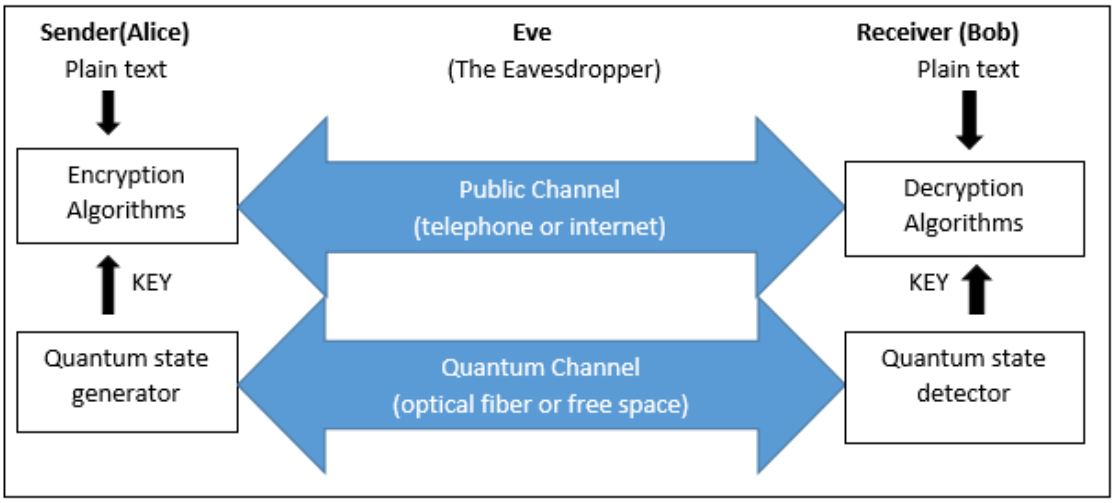

Figure 4. Fundamental Quantum Key Exchange Algorithm[22]

First Alice encodes the random bits into quantum states and starts the communication by transmitting these quantum states through quantum channel. When Bob receives the quantum states, he applies quantum measurement to the received quantum states and gives a string of bits as an output according to his measurement. This bit string that Bob measures is called raw key. Then he sends the chosen bases to

Alice. Alice compares her bases with Bob's bases and tells Bob which bases were same. Both throw out the bits for which they used different bases, and keeps only those for which they have used the same basis. These bits are called sifted key[23].

QKD Algorithms uses the law of quantum mechanics to securely exchange key between the users. Some of the quantum key distribution (QKD) algorithms are based on no-cloning theorem and probabilistic measurement e.g. BB84 protocol and some are based on entanglement e.g. ERP protocol. The first QKD algorithm was BB84 based on probabilistic measurement and no-cloning theorem. Other variants of BB84 protocol are BB92, COW (coherent one-way protocol), SARG04, Eckert's E91 Entanglement protocol, KMB09 (High Error-rate QKD protocol by Khan et al) etc. The distribution of QKD algorithms can be shown by the hierarchy defined in figure 5.

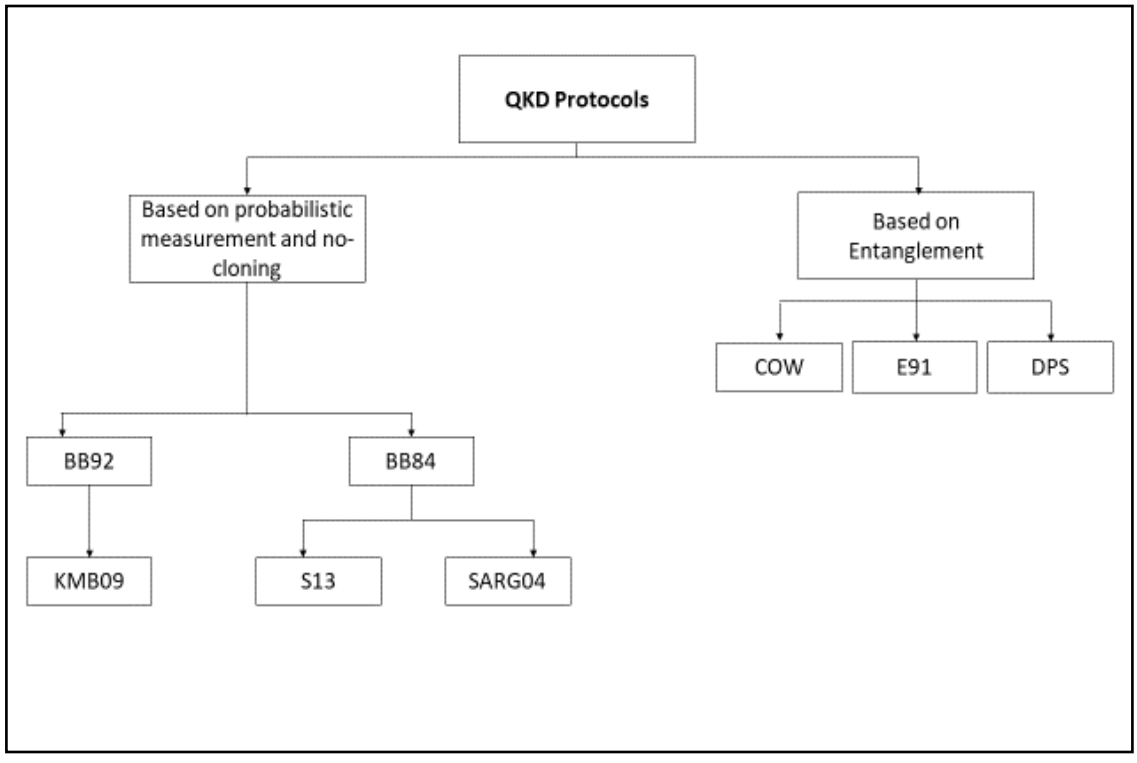

Figure 5. Hierarchy of QKD Protocol[24]

1.8: Motivation of proposed thesis:

After reviewing all the QKD protocols, basic quantum key distribution protocols based on no-cloning theorem and probabilistic measurement are selected for the mathematical modelling. As quantum cryptography is device specific and can only be done by using quantum computers. These quantum computers are large, expensive and complex to use and there is limited access to quantum computers. Considering these limitations, there is a requirement of some simulation and emulation models that can check the working of key distribution algorithms. Some generalized emulators have been designed by the researchers but standalone dedicated system is not available for BB84 and BB92 protocol. The emphasis of the proposed thesis was on

- The discussion of Quantum computing.
- The exploration of Quantum cryptography. The main focus of proposed thesis was specifically on basic quantum key distribution (QKD) protocols.
- The simulation of BB84 and BB92 Quantum key distribution protocols on visual C++.
- The designing of resource efficient emulator based on FPGA that primarily emulates the protocols based on superposition and probabilistic measurement. It will help to test the performance of quantum cryptography protocols and benefits the future researchers to implement already tested quantum key distribution protocols for making the IoT (Internet of Things) communication reliable and secure.
- Performance analysis of actual results, simulation and emulation of basic QKD protocol.

Chapter 2: Literature Review

Quantum computing was first introduced by a U.S physicist named “Richard Feynman” in 1980. Quantum computing uses the laws of quantum physics to perform calculations faster than digital computers. Due to its efficient speed of calculation, quantum computers are beneficial for various number of field such as healthcare, industries, meteorology and most of the above security. In 1982, Paul Benioff developed the first model of quantum Turing machine by describing Schrodinger’s equation [25]. There are seven quantum technologies that are used to realize the quantum computer. These quantum computers can be categorized into four generations. First generation of quantum computers are implemented by using ion trap. Second generation was realized by using distributed diamonds, optical technologies and superconducting quantum circuits. Monolithic diamonds, quantum dots and donor technologies are used for implementing third generation quantum computers and topological quantum technologies are used for fourth generation quantum computers[5]. These days, Google, IBM, Intel and Microsoft are trying to launch quantum computer that can be commercially available[6]. Bernard and Brassard in 1984 gave an idea of using quantum techniques in cryptography for enhancing the security of data. These protocols were based on the property of superposition and probabilistic measurement. These quantum cryptographic techniques ensure the secure transmission of key without letting eavesdropper to gain any information about the key[26]. For this purpose, quantum cryptography uses $qubit$ to transfer information of key between transmitter and receiver.

2.1. Basics of $\boldsymbol{Qubit}$:

$Qubit$ is a unit of information which defines two dimensional quantum system. It works on principles of quantum laws such as superposition, probabilistic measurement etc. As the superposition defines that a qubit is present in both 0 and 1

state until it is measured. After measurement on receiver side, it collapses into 0 or 1. This property secure the key as it does not exist until it is observed so it is hard to get any information about the key while it is travelling[27]. The law of superposition can also be defined by the spinning property of electron that was described by Stern Gerlach experiment performed in 1922 by shooting beam of electrons in the presence of magnetic field. This experiment showed that before interaction of electron with the magnetic field, we cannot predict if it is spin up or spin down but when the spinning particle is measured by interacting it with magnetic field, it can be found in one of the two states; either spin up or spin down[28].

$Qubits$ also have a geometrical representation by using Bloch sphere that was named after the Swiss-American physicist Felix Bloch. Bloch sphere represents the $qubit$ as a point in a unit sphere. The north pole of sphere corresponds to state $|0\rangle$ and south

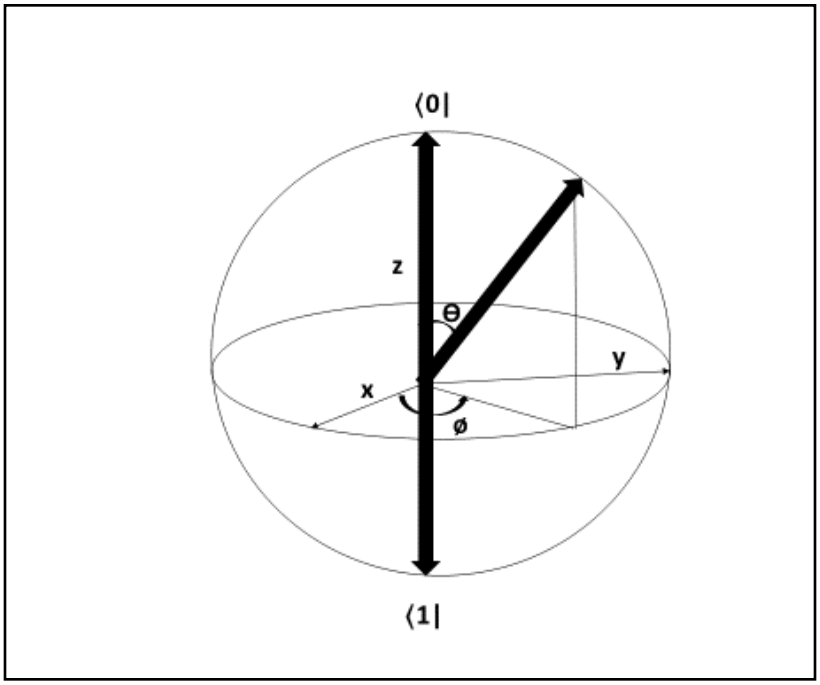

Figure 6. Geographical representation of Qubit by using Bloch Sphere

Pole represents to state $|1\rangle$. A $qubit$ can be mapped to an arrow from origin to the three dimensional sphere as stated in figure 6. Every $qubit$ is represented by two angles; latitude (Θ) and longitude (ϕ). Longitude (ϕ) is the angle that the projection of $|\psi|$

makes from x along the equator and latitude(θ) is half of the angle that ψ makes with the z axis.

Following is the mathematical derivation for conversion of qubit representation to Bloch sphere representation. We start with the generic form of $qubit$:

$$|\psi| = C_0|0\rangle + C_1|1\rangle \qquad \text{Eq.1}$$

Where C_0 and C_1 are complex numbers and $|C_0|^2 + |C_1|^2 = 1$

Rewriting the $qubit$ in polar form, we get the equation:

$$|\psi| = r_0 e^{i\emptyset 0}|0\rangle + r_1 e^{i\emptyset 1}|1\rangle \qquad \text{Eq.2}$$

Where $C_0 = r_0 e^{i\emptyset 0}$ and $C_1 = r_1 e^{i\emptyset 1}$

There are four real parameters r_0, r_1, $\emptyset_0$, $\emptyset_1$ in the equation, to get it more simplified it is multiplied by an arbitrary complex number $e^{-i\emptyset 0}$ as we know that a quantum physical state remains same if we multiply its vector by an arbitrary complex number.

The equation becomes

$$e^{-i\emptyset 0}\,|\psi| = e^{-i\emptyset 0}\left(r_0 e^{i\emptyset 0}\langle 0| + r_1 e^{i\emptyset 1}\langle 1|\right) = r_0\langle 0| + r_1 e^{i\emptyset 1 - i\emptyset 0}\langle 1| \qquad \text{Eq.3}$$

Now there are three real parameters r_0, r_1 and $\emptyset = i\emptyset_1 - i\emptyset_0$. This equation becomes more better if we use the fact

$$1 = |C_0|^2 + |C_1|^2 = \left(r_0 e^{i\emptyset 0}\right)^2 + \left(r_1 e^{i\emptyset 1}\right)^2 \qquad \text{Eq.4}$$

We get that $|r_0|^2 + |r_1|^2 = 1$. We can rename r_0 and r_1 as $r_0 = \cos\theta$ and $r_1 = \sin\theta$. Putting the values in equation 1, the canonical representation of $qubit$ becomes

$$|\psi| = \cos\theta\,\langle 0| + e^{i\emptyset} sin\theta\,\langle 1| \qquad \text{Eq.5}$$

The final equation contains only two real parameters $\emptyset\ and\ \theta$ that are necessary to represent a $qubit$ in three dimensional sphere called Bloch sphere[28].

When a $qubit$ is measured in the standard basis, it will collapse to a bit which will be equal to the North pole or South pole in Bloch Sphere. On which pole the $qubit$ will collapse depends on the latitude or angle Θ. If the $qubit$ is present at the equator of Bloch sphere, then there is 50-50 chance of collapsing it to either north pole or south pole.

2.2. Related Work:

Quantum computers is finding its applications in the field of meteorology, finance, marketing, healthcare, logistics and most of the above security. Many researchers are contributing in the development of quantum protocols for providing security. Bernard and Brassard in 1984 gave an idea of using quantum techniques in cryptography for enhancing the security of data. After the development of BB84 and BB92 quantum cryptography protocols, many other researchers have contributed towards QKD protocols. In 1991, Artur Ekert developed another quantum key distribution protocol that was different from BB84 and BB92 as it was based on the property of entanglement[29]. It works on the statement that "when one of the $qubits$ is measured, they both will collapse to the same value". In this type of protocol, each time a sender wants to send a secret key to receiver, a sequence of entangled pair of $qubits$ is generated and given to sender and receiver. When they are ready to communicate, one of them measures it, and it will collapse $qubit$ to the same random value on sender and receiver side. To check that the pairs are still entangled, some tests are performed based on John Bell's famous Bell's inequality that describes that if the pairs are independent of each other, they are not entangled and measurement will satisfy the Bell's inequality. Another

QKD protocol called "quantum teleportation" was introduced in 1993 by Bennett which was based on the teleportation process by which the state of an arbitrary $qubit$ can be transported from one location to another. The teleportation algorithm works

with two entangled $qubits$. One of them is held by sender and other is held by receiver. These protocols were the base protocols.

Based on these techniques, some latest quantum key distribution protocols have been built in 20th century. In [30], the author proposed an alternative quantum cryptographic protocol known as KMB09 protocol. This protocol uses two mutually unbiased polarization states, one for the encoding of '1' and the other for the encoding of '0'. Minimum Index Transmission Error Rate (ITER) and Quantum Bit Error Rate (QBER) that were introduced by eavesdropper are the major cause of security in this scheme. Another quantum key distribution protocol known as S09 protocol is introduced in [31] for transmitting data on public channel securely. The security of this scheme depends on the fact that the transmitter and receiver use private keys in multiple exchange of qubits. The way it is different from BB84 is that the transmitted $qubit$ can be in any arbitrary direction whereas in BB84, the $qubit$ was encoded in one of the four different polarization states. S13 is also a latest quantum key distribution protocol described in [32] based on random seed. It's working is identical to BB84 protocol, the only way it is different from BB84 is that it uses private reconciliation between a random seed and asymmetric cryptography. Hence it generates a larger secure key. Another author in [33] introduced a quantum key distribution protocol based on pseudorandom bases abbreviated as PRB Protocol. This scheme uses pseudorandom bases generated by pseudorandom number generators (PRNGs) instead of using random bases in order to eliminate the problem of sifting key and losing half key. This protocol strongly needs only one photon source of light. Table.2 shows the list of defined QKD protocols along with the quantum principles on which they are based.

Table 2. Quantum Key Distribution Protocols with their Quantum principles [22]

BASE QUANTUM KEY DISTRIBUTION PROTOCOLS					
No	**Name of Protocol**	**Year**	**Author**	**Principles**	**Applications**
1.	BB84 Protocol	1984	Charles Bennett and Gilles Brassard	Heisenberg Uncertainty principle	It uses two orthogonal basis for secure communication.
2.	ERP Protocol	1991	Arthur K. Ekert	Entanglement	It uses entangled pairs of $qubits$ for secure communication.
3.	BB92 Protocol	1992	Charles Bennett	Heisenberg Uncertainty principle	It uses only one non-orthogonal basis instead of two orthogonal basis used in BB84 protocol.
4.	Teleportation protocol	1993	Charles Bennett	Entanglement and teleportation	It works with two entangled $qubits$
LATEST QUANTUM KEY DISTRIBUTION PROTOCOLS					
No.	**Name of Protocol**	**Year**	**Author**	**Principles**	**Applications**
5.	KMB09 Protocol	2009	Muhammad Mubashir Khan, Michael Murphy and Almut Beige	Heisenberg Uncertainty Principle	Two different bases are used for encoding '0' and encoding '1' rather than using two directions of a single base.
6.	S09 Protocol	2012	Eduin Esteban Hernadez Serna	Public private key cryptography	Source and transmitter each use secret

					key in multiple exchange of $qubit$.
7.	S13 Protocol	2013	Eduin H.Serna	Heisenberg Uncertainty Principle	It uses private reconciliation from random seed and asymmetric cryptography for the generation of larger secure keys.
8.	PRB Protocol	2018	A.S. Trushechkin, P.A. Tregubov, E.O. Kiktenko, Y.V. Kurochkin, A.K. Fedorov	Heisenberg Uncertainty Principle	It uses pseudorandom bases generated by pseudorandom number generators (PRNGs) instead of using random bases in order to eliminate the problem of sifting procedure and losing of half key

After the development of several quantum key distribution protocols, many researchers performed experiments to compare the performance of classical and quantum cryptography to see how beneficial is quantum cryptography as compared to classical one. An author in [21] compared the common algorithms of quantum and classical cryptography on the basis of encryption and decryption time and avalanche effect. Another researcher in [34] presented the benefits of quantum cryptography over

classical cryptography. An author in [35] surveyed different quantum cryptography protocols to list down their benefits and concluded that data secured by quantum cryptography algorithms have low probabilities of being intercepted by the third party.

As Quantum computers are large, expensive and complex to use and we do not have access to quantum computers. Considering these limitations, researchers started designing some simulation and emulation models that can check the working of key distribution algorithms. In [36], BB84, BB92 and E91 were compared by using quantum simulator named QuVis to show that BB92 performed best on the basis of error performance. In [37], a Brazilian researcher performed an experiment to demonstrate the working of BB84 protocol by using intense beam light. In [24], an author presented the simulation process of quantum key distribution protocol by using object orient approach. An efficient reconciliation method for quantum key distribution protocol was proposed in [38]. An author in [39] presented the quantum key distribution model that introduced two way quantum channel rather than using one way quantum channel to increase the capacity and for error reduction. Quantum information exchange computer emulator is proposed by an author in [40].An author in [41], proposed a hardware emulator based on FPGA that was capable of running quantum algorithms. Another FPGA based quantum emulator was presented in [42] that describes a major advantage in performance over simulators by emulating quantum protocols at high level instead of simulating individual gate operations. In [43], projectQ was introduced that was a high performance simulator with emulation capabilities. This framework enables the testing of quantum protocols by using simulation properties and enables running these algorithms on quantum hardware by using back-end connecting to the IBM Quantum Experience cloud service. In [44], FPGA based quantum emulator was presented that can use 32-bit floating-point arithmetic in order to emulate an entangled 4-$qubit$ quantum system by using Quantum Fourier Transform, Grover's Search algorithms on a single FPGA node. One of the authors of Oxford shire in [45] introduced QuESTlink that

is a mathematical package for emulating quantum circuits. An author of [46] used an approach of mixing series and parallel processing on FPGA to propose software-hardware system for quantum emulation.

Table 3. List of already existed simulators and emulators for QKD protocols

S.No.	Emulator/Simulator	Basics
1.	QuVis quantum simulator	Compared BB84, BB92 and E91 by using quantum. [36]
2.	Laser beam	Intense laser beam is used to explain the working of BB84 protocol.[37]
3.	Turbo codes	Turbo codes are used to construct an efficient reconciliation method for BB84 protocol.[38]
4.	Emulator program of the Qt Creator C ++	Predicted noise and attenuation of quantum channel in key exchange process. [40]
5.	FPGA based scalable quantum emulator	Quantum computer emulator that can emulates the behavior of real quantum systems. [42]
6.	Emulator based on FPGA in combination with Vivado.	It can emulate the desired algorithms written in C++. [42]
7.	ProjectQ simulator	It allows testing of key distribution protocols. It uses back end connection to the IBM Quantum Experience cloud service to run protocols on quantum hardware.[43]
8.	FPGA based quantum emulator	Quantum emulator that allows the modelling of quantum system consist of 4 fully entangled qubits.[44]

9.	QuESTlink, a mathematical solution consist of state vectors and density matrices.	It can manipulate quantum circuits and do rapid simulation by using remote hardware.[45]
10.	FPGA based emulator based on series-parallel architecture	Series-parallel technique is used to reduce the resource utilization of QKD emulator.[46]

2.3. Limitations of basic quantum key distribution protocols:

Although the security of basic quantum key distribution protocols is unbreakable in theory but when it comes to practical life application, there are number of attacks that leads to the security loopholes. When this protocol is applied to the real world, there is a tradeoff between the security of BB84 and key generation rate. This limits the distribution of infinitely long keys. The key generation rate must be in certain limit to claim its security[47]. The most realistic photon detector used for detecting the photon contains the time interval right after the detection of photon in which a detector recovers and do not detect more incoming photons. Therefore, if the transmission rate will be high, detector will be in recovering phase and more photons will reach to the detector which cannot be detected and results in the leakage of photons. Thus, BB84 and BB92 is not suitable for high transmission rates [48]. An attack that BB84 and BB92 can encounter is photon number splitting (PNS) attack. This attack explains that in BB84 and BB92, Alice has to send a single photon towards the Bob at a time but due to the limitation of single photon generator, weak coherent pulses are used to implement these basic QKD protocols. These are the signal states with a probability of generating more than one photon at a time. These generated photons are exactly the same. Whenever Eve detects the multiple photons, she splits the photon and pass only one. In this way, information can be retrieved by Eve. The security of BB84 and BB92 against this attack can be improved by using decoy states [49]. Another possible attack is called

Trojan horse attack in which an Eve can send a bright light towards the polarizer of transmitter and tries to leak information from the reflected light coming from the polarizer of transmitter [50].

Considering these limitations, there are certain requirements that needs to be taken care of when designing the hardware for quantum key distribution protocols:

- There should be no physical access to Eve on polarization device of transmitter and receiver.
- The authentication of classical channel used for communication must be done by using certain authentication schemes.
- The pseudo random number generator used by transmitter and receiver should be truly random.
- Transmission key rate should be kept under certain limit in order to retain the security of BB84 and BB92 protocols[51].

Many researchers have contributed in designing generalized emulators and simulators. The emphasis of the proposed thesis is on the designing of dedicated emulator that primarily emulates the protocols based on superposition and probabilistic measurement. This resource efficient standalone system will help to test the performance of quantum cryptography protocols and benefits the future researchers to implement already tested quantum key distribution protocols for making the IoT (Internet of Things) communication reliable and secure.

Chapter 3: BB84 Protocol

3.1. Background of BB84 Protocol:

BB84 was the first QKD protocol presented by Charles Bennett and Gilles Brassard in 1982. It was based on probabilistic measurement. Key Exchange between two users is done by using $qubits$ which make the key secure as the eavesdropper cannot make a perfect copy of the stream of $qubits$ according to the no-cloning property of quantum theory[52]. This protocol uses four different polarization states $\binom{1}{0} as\ 0°, \binom{0}{1}$ as $90°$, $\binom{-1}{1} as\ 45°$, $\binom{1}{1}\ as\ 135°$ or two orthogonal bases: rectilinear (+) and diagonal base (x) to convert the string of bits that is in the form of $0,1$ into string of $qubits$ in the form of $|\psi| = C_0|0\rangle + C_1|1\rangle$ that can be sent to the other user from quantum channel[53]. The mathematical description of these bases are:

$$|+> = \frac{1}{\sqrt{2}}\binom{1}{0} + \frac{1}{\sqrt{2}}\binom{0}{1} \qquad (0°, 90°)$$

$$|\times> = \frac{1}{\sqrt{2}}\binom{1}{1} + \frac{1}{\sqrt{2}}\binom{-1}{1} \qquad (45°, 135°)$$

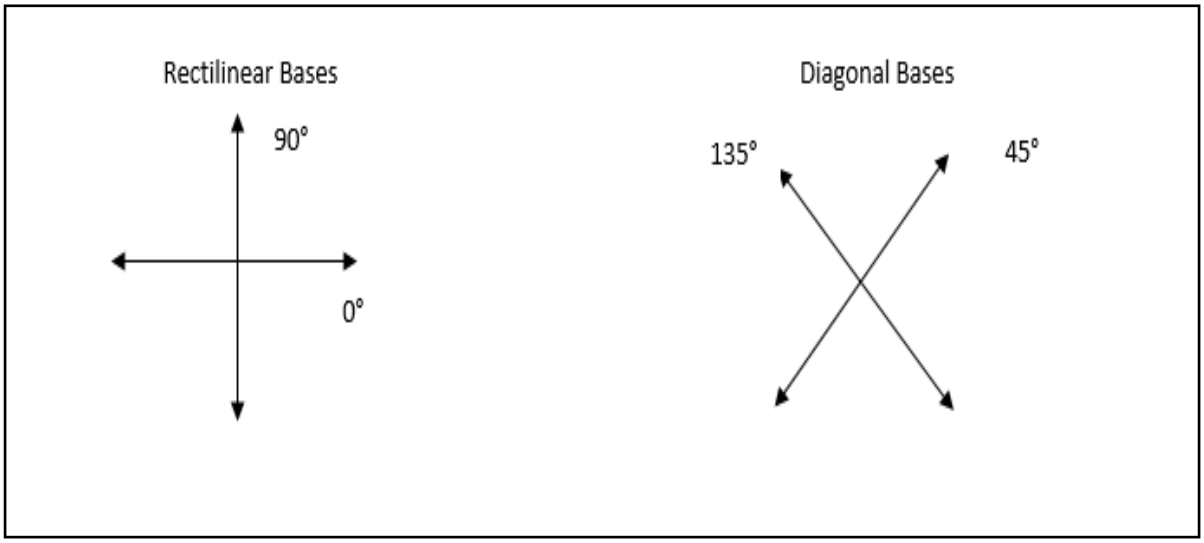

Figure 7. Orthogonal bases used for BB84 Protocol[54]

3.2. Description of BB84 Protocol:

1. Alice randomly generates a bit string of n random bits.
2. For each bit, Alice chooses a base from diagonal or rectilinear randomly to encode a bit into orthogonal bases.
3. Alice transmit these randomly chosen bases to Bob.
4. Bob receives the n $qubits$ and measures each in rectilinear or diagonal bases randomly as Bob does not know which bases Alice has chosen for which bit.
5. Bob notify Alice about the bases he has chosen for measuring $qubits$.
6. Alice compares his bases with the bases that bob has chosen to encode bits and notify the results to Bob.
7. Both discard the bits corresponding to $qubits$ that they measured with different bases.
8. Remaining bits are the final bits called sifted key.

As they both choose bases randomly there is a chance that for about half of the time, Bob's bases will be same as Alice's bases. Hence there is a probability of getting $n/2$ bits as the sifted key. Therefore, if Eve tries to sniff or measure the $qubit$, it will affect the Bob's chances of agreement with Alice negatively[55, 56].

The working of BB84 Protocol is defined in the table 4.

Table 4.BB84 Description

Alice random bits	1	0	0	1	1	0	0	1
Alice random bases	↑	→	→	↖	↑	→	→	↖
Bob received bits	1	0	0	1	1	0	0	0

Bob chosen bases	↖	→	↗	↑	↑	→	→	↗
Result after comparing	1	0	1	0	1	0	0	1
Common Bases of Alice and Bob		→			↑	→	→	
Sifted Key corresponding to common bases.		0			1	0	0	

3.3. Simulation of BB84 Protocol:

In order to evaluate the performance of BB84 QKD Protocol, several simulators have been used. In the proposed thesis, Object Oriented approach is selected to design the simulation of BB84 protocol. Flow chart defined in figure 8 shows the working of BB84 QKD Protocol.

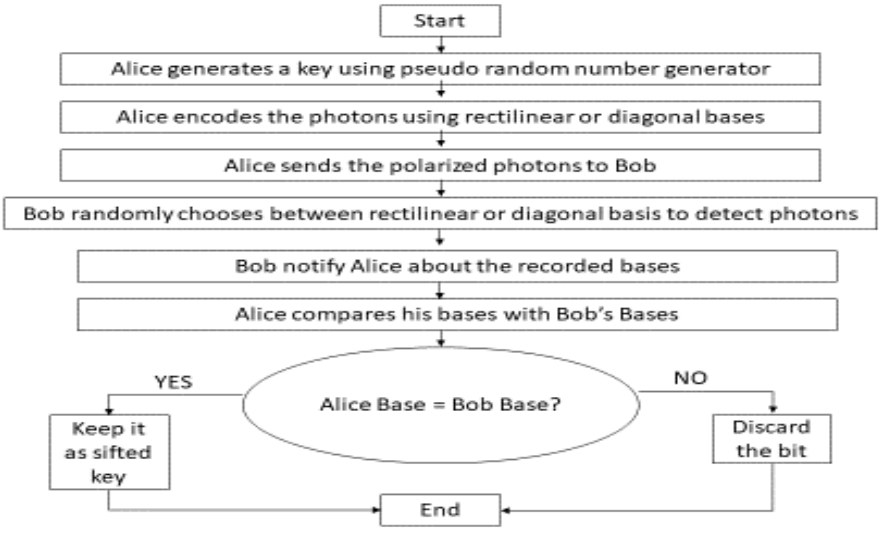

Figure 8. Flowchart of working of BB84 Protocol

The main module is defined as the sender and receiver is defined by a function in C++. The sender inputs random sequence of 8 bits and encodes them into diagonal or rectilinear bases by using pseudorandom number generator. These bases are defined as complex array 1. These are sent to the receiver. Receiver side defined in code as Bob function measures the $qubit$ according to his own chosen bases defined as complex array 2 in the code. At the end the bases of sender and receiver are compared to find the sifted key. The simulation is repeated 100 times and successful bit retrieval on each time is defined in figure 9.

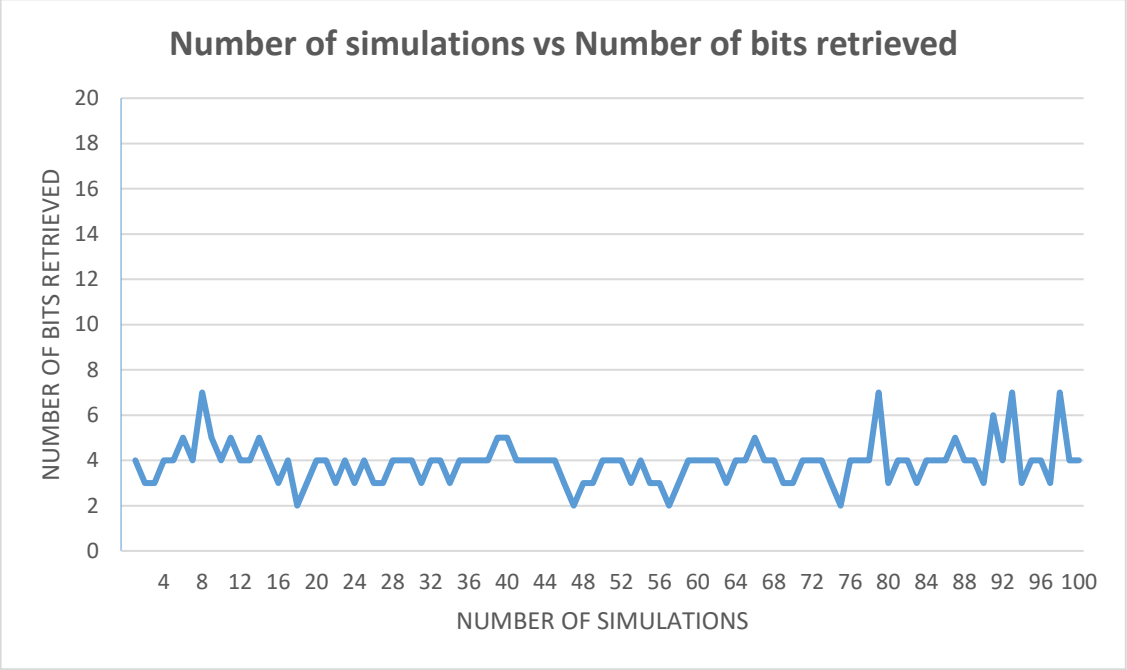

Figure 9. Graph of simulation of BB84 Protocol

3.4. Emulation of BB84 Protocol:

For emulation, FPGA method is used to implement coding of BB84 Protocol. The hardware implementation of BB84 protocol is designed on Xilinx by using Verilog programming language. The simulation results of BB84 protocol are evaluated by using

ISE simulator. The architectures of transmitter and receiver of BB84 protocol are defined below:

3.4.1. Architecture of BB84 Transmitter side:

Step 1: 8-bit binary key is randomly generated by the transmitter as an input.

Step 2: 8-bit random number is generated by the pseudo random number generator.

Step 3: After getting strings of 8-bit key and random numbers, each bit of binary number and each bit of random number is fed into $3x8$ mux in sequence which converts the input bits into $qubits$ according to the bases defined in step 4. 3-bit counter is used to select the bit of binary key and random number from their 8-bit strings. Counter enables the $3x8$ mux to take the bits from binary key and random number string sequentially and apply protocol operation on the respective bits.

Step 4: Four polarization states i.e. $0°(\rightarrow), 90°\ (\uparrow), 45°(\nearrow)$ and $135°(\nwarrow)$ are defined by $2x4$ mux. The values assigned to each orthogonal state are defined in the table below. To show the respective output, we need to assign different values to each orthogonal state so that we can differentiate between the output obtained from the emulation circuit.

Binary key bit[k], Random number[counter]	Output
00	$0°(\rightarrow)$
01	$90°\ (\uparrow)$
10	$45°(\nearrow)$
11	$135°(\nwarrow)$

Step 5: These transformed $qubits$ are stored in 8-bit qubit storage memory and send towards the receiver side. The architecture of BB84 transmitter is depicted in the figure.

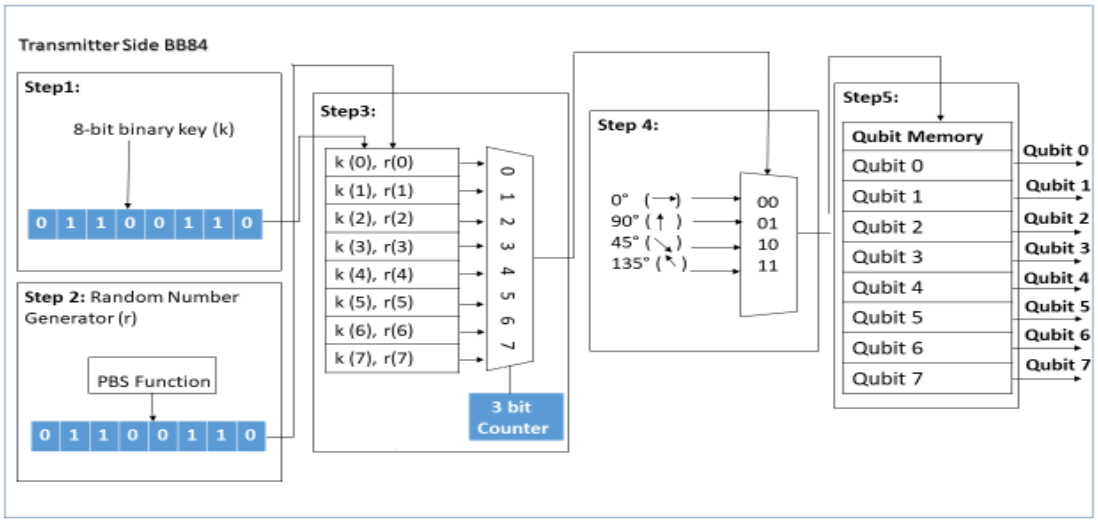

Figure 10. Architecture of BB84 Transmitter side

3.4.2. Architecture of BB84 Receiver side:

The architecture of receiver side of BB84 protocol is defined in steps.

Step 1: The $qubits$ generated by transmitter as an output are fed into the receiver as an input.

Step 2: Two 8-bit random numbers (r1 & r2) are generated by pseudo random number generator.

Step 3: Each bit of binary number and each bit of random number r_1 is fed into $3x8$ mux in sequence by using counter which converts the $qubits$ into output bits according to the mapping defined in step 4.

Step 4: If the base of qubit and base selected by random number are same, the output bit will be same as that of the bit received by transmitter. If the bases are different then norm will be calculated and compared with the random number r_2. If norm is greater than r2, output will be 0 otherwise 1.

if random number is 0, and $qubits$ is at polarization state 0° means $\binom{1}{0}$, the mux will give the output 0 and if $qubit$ is $\binom{0}{1}$, output will be 1. For $qubit$ $\binom{-1}{1}$ and $\binom{1}{1}$, norm will be calculated and compared with the r_2 by using comparator to get the desired output. If r_2<norm, output will be 0 otherwise 1.

If random number is 1, and $qubit$ is $\binom{-1}{1}$, the mux will give the output 0 and if $qubit$ is $\binom{1}{1}$, output will be 1. For qubit $\binom{0}{1}$ and $\binom{1}{0}$, norm will be calculated and compared with the r_2 by using comparator to get the desired output. If r_2 <norm, output will be 0 otherwise 1. Outputs are defined in the table 5.

Table 5. Outputs of BB84 Receiver

Counter	$\boldsymbol{r_1[counter]}$	$\boldsymbol{C_0}$	$\boldsymbol{C_1}$	**Output**
000	0	01000000	00000000	0
001	0	00000000	01000000	1
010	0	00101101	00101101	r_2< norm: 0 r_2> norm: 1
011	0	11010011	00101101	r_2< norm: 0 r_2> norm: 1
100	1	01000000	00000000	r_2< norm: 0 r_2> norm: 1
101	1	00000000	01000000	r_2< norm: 0 r_2> norm: 1

110	1	00101101	00101101	0
111	1	11010011	00101101	1

The architecture of BB84 receiver is depicted in the figure below.

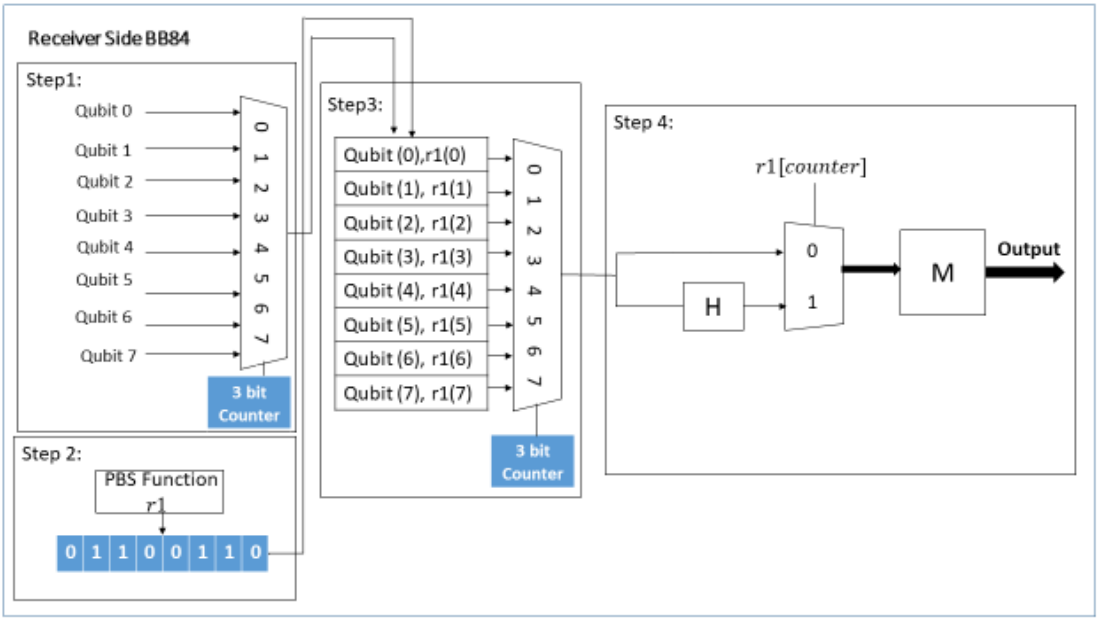

Figure 11. Architecture of BB84 Receiver Side

Chapter 4: BB92 Protocol

4.1: Background of BB92 Protocol:

BB92 is the simplified version of BB84 published by Charles Bennett in 1992. This algorithm states that the use of two different orthogonal bases or four polarization states is redundant for key exchange. Key exchange can be done by using only one non-orthogonal base that is a pair of $\binom{1}{0}$ as 0° and $\frac{1}{\sqrt{2}}\binom{1}{1}$ as 45°. Rest of the protocol is same as that of BB84[57]. The mathematical description of non-orthogonal base can be written as:

$|\nearrow> = \frac{1}{\sqrt{2}}\binom{1}{0} + \frac{1}{\sqrt{2}}\binom{1}{1}$ (0°, 45°)

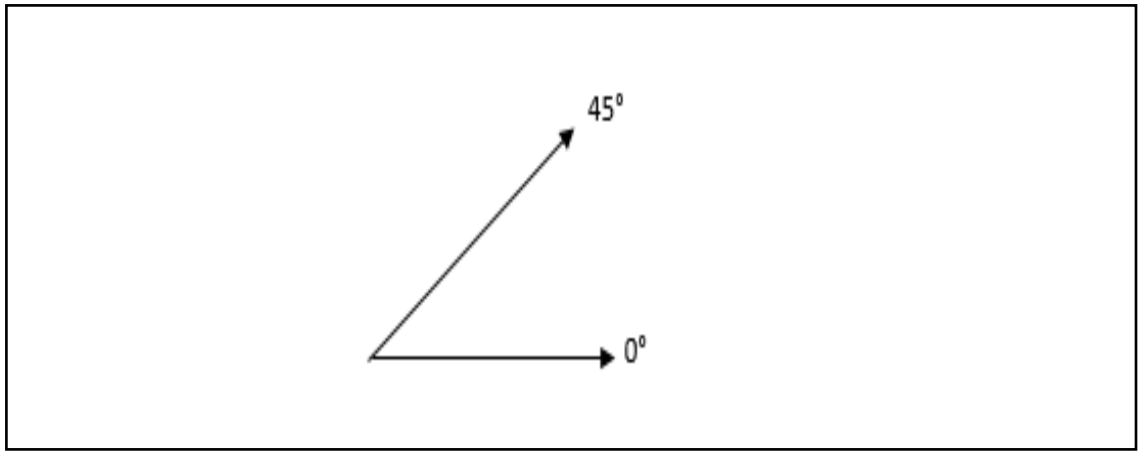

Figure 12. Non orthogonal base used for BB92 Protocol[57]

4.2. Description of BB92 Protocol:

BB92 Protocol can be stated as follows:

1. Alice generates bit string containing n random bits.
2. Alice measures 0 in $0°$ polarization state and 1 in $45°$ polarization state.
3. Alice transmit these chosen bases to Bob.

4. Bob receives the n $qubits$ and measures each in rectilinear or diagonal bases randomly as Bob does not know which bases Alice has chosen for which bit.
5. Bob notify Alice about the bases he has chosen for measuring qubits.
6. Alice compares his bases with the bases that bob has chosen to encode bits and notify the results to Bob.
7. Both discard the bits corresponding to $qubits$ that they measured with different bases.
8. Remaining bits are the final bits called sifted key.

The working of BB92 Protocol is defined in the table.6.

Table 6. BB92 Protocol Description

Alice random bits	1	0	0	1	1	0	0	1
Alice random bases	↗	→	→	↗	↗	→	→	↗
Bob received bits	1	0	0	0	0	0	0	1
Bob chosen bases	↗	↗	→	↗	→	→	→	↖
Result after comparing	1	0	0	1	0	0	0	0
Common Bases of Alice and Bob	↗		→	↗		→	→	

Sifted Key corresponding to common bases.	1		0	1		0	0	

4.3. Simulation of BB84 Protocol:

In order to evaluate the performance of BB92 QKD Protocol, several simulators have been used. Object Oriented approach is selected to design the simulation of BB92 protocol. Flowchart defined in figure 13 shows the working of BB92 QKD Protocol.

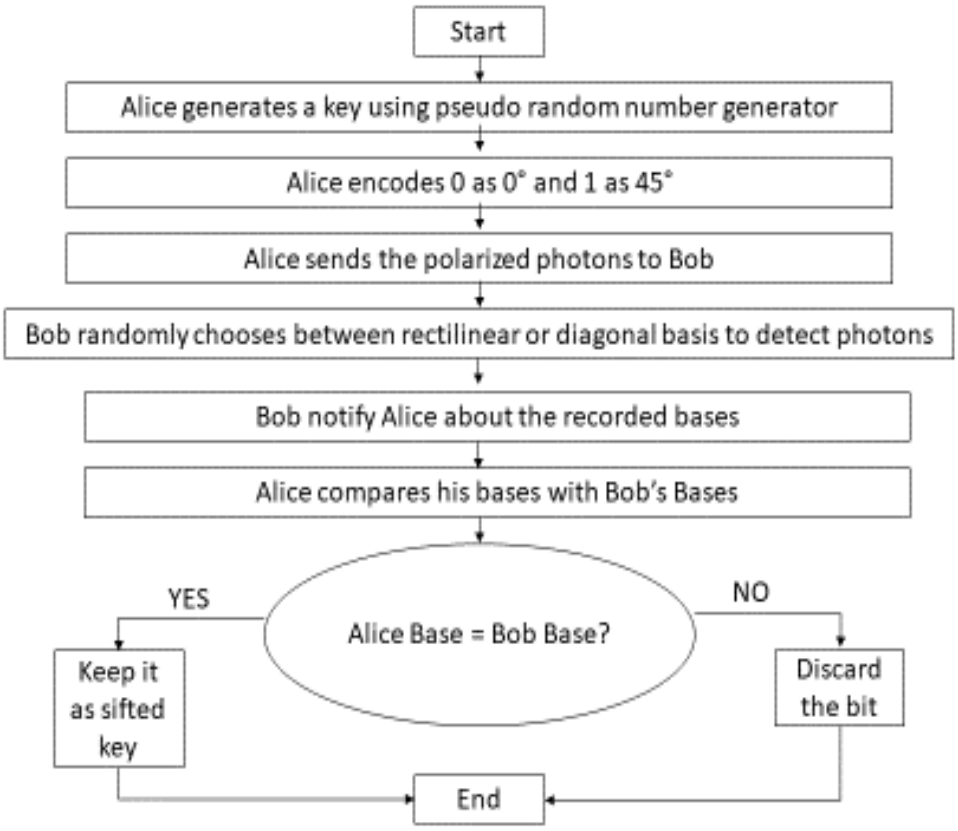

Figure 13. Flowchart of working of BB92 Protocol

The main module is defined as the sender and receiver is defined as a function in C++. The sender inputs random sequence of 8 bits and encodes 0 as $0°$ and encodes 1 as $45°$. These bases are defined as complex array 1. These are sent to the receiver. Receiver side defined in code as Bob function measures the qubits according to his own chosen bases defined as complex array 2 in the code. At the end the bases of sender and receiver are compared to find the sifted key. The simulation is repeated 100 times and successful bit retrieval on each time is defined in figure 14.

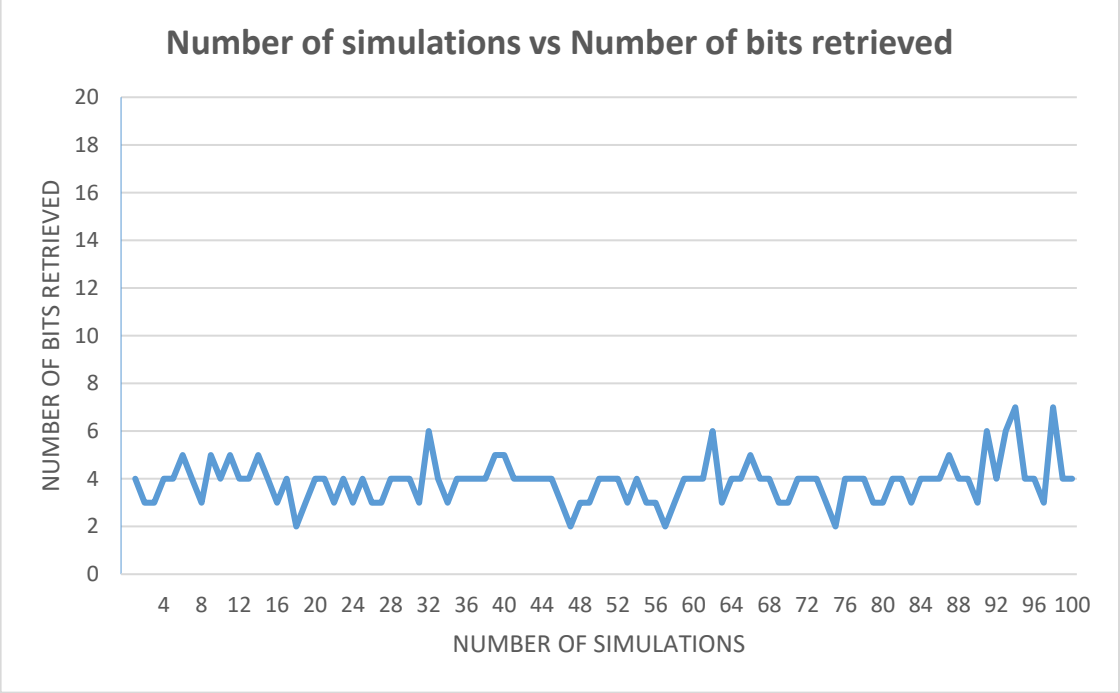

Figure 14. Graph of simulation of BB92 Protocol

4.4. Emulation of BB92 Protocol:

For emulation, FPGA method is used to implement coding of BB92 Protocol. The hardware implementation of BB92 protocol is designed on Xilinx by using Verilog programming language. The simulation results of BB92 protocol are evaluated by using

ISE simulator. The architectures of transmitter and receiver of B92 protocol are defined below:

4.4.1. Architecture of BB92 Transmitter side:

Step 1: A random 8-bit binary key is generated by the transmitter as an input.

Step 2: Each bit of binary number is fed into $3x8$ mux in sequence which converts the input bits into $qubits$ according to the bases defined in step 3. 3-bit counter is used to select the bit of binary key and random number from their 8-bit strings. Counter enables the $3x8$ mux to take the bits from binary key and random number string sequentially and apply protocol operation on the respective bits.

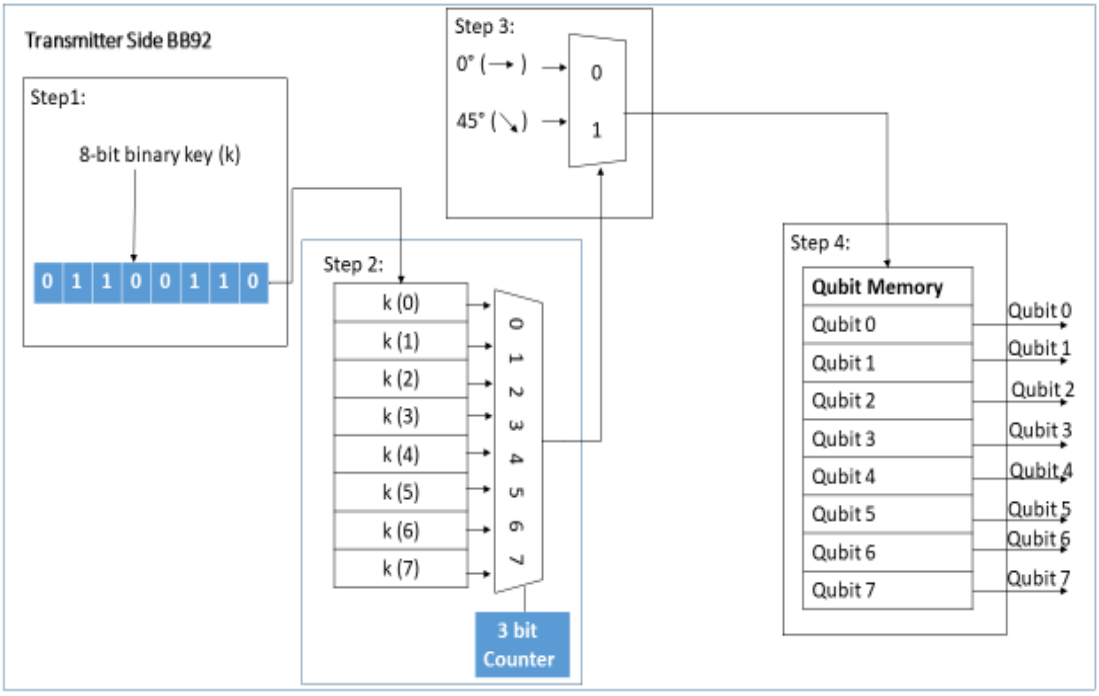

Figure 15. Architecture of transmitter of BB92 Protocol

Step 3: Two polarization states i.e. $0°(\rightarrow)$ and $45°(\nearrow)$ are defined by $2x1$ mux. These states are defined in $2x1$ mux in the table below. To show the respective output, we

need to assign different values to each orthogonal state so that we can differentiate between the output obtained from the emulation circuit.

Random key bit (k)	Output
0	$0°(\rightarrow)$
1	$45°(\nearrow)$

Step 4: These transformed $qubits$ are stored in four 8-bit qubit storage memory and sent towards the receiver side.

The architecture of BB92 transmitter is depicted in the figure 15.

4.4.2. Architecture of BB92 Receiver side:

The architecture of receiver side of BB84 protocol is defined in steps.

Step 1: The $qubits$ generated by transmitter as an output are fed into the receiver as in input.

Step 2: Two 8-bit random numbers (r_1 & r_2) are generated by pseudo random number generator.

Step 3: Each bit of binary number and each bit of random number $r1$ is fed into $3x8$ mux in sequence which converts the $qubits$ into output bits according to the outputs defined in step 4.

Step 4: If the receiver chooses same orthogonal state as that of transmitter, it will get the output defined for respective bit. If it chooses different base, there will be no output.

Table 7. Outputs of BB92 Receiver

Counter	$R1$	C_0	C_1	**Output**
00	0	01000000	00000000	0
01	0	00101101	00101101	X (no output)
10	1	01000000	00000000	X (no output)
11	1	00101101	00101101	1

The architecture of BB92 receiver is depicted in the figure below.

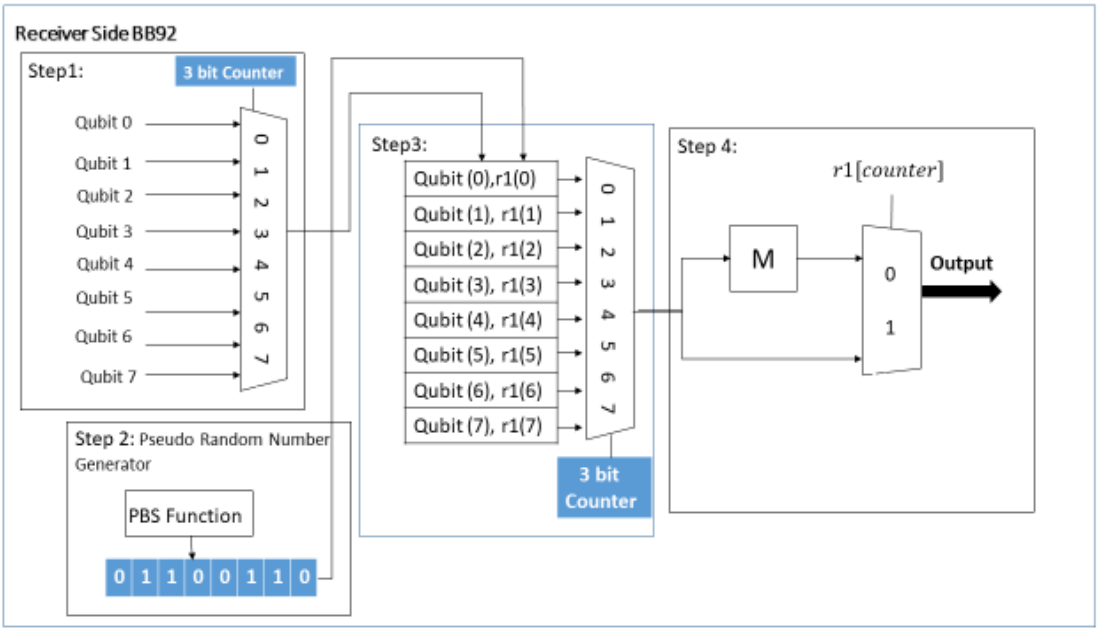

Figure 16. Architecture of receiver of BB92 Protocol

Chapter 5: Analysis and Results

Quantum cryptography uses laws of quantum physics and takes the advantage of unusual and unique behavior of small microscopic elements which enables the user to develop the secret keys securely and enables the user to sense the presence of eavesdropper trying to steal the message that was not possible in the previous cryptography techniques. These techniques are drawing attention of researchers. Many researchers have contributed towards the development of quantum key distribution protocols that are based on some basic protocols e.g. BB84, BB92 and E91.

The proposed thesis has provided the detailed study, simulation and emulation of the most prominent protocols e.g. BB84 and BB94. The simulation of these protocols was done by using visual C++. $Qubits$ were represented by using four floating point numbers in the form of matrix. One floating point of datatype 'double' occupies 64 bits in computer memory. Hence $64x4 = 256$ total bits were used to represent one $qubit$ in visual C++. Random number between 0 and 1 was also generated and represented by using floating point number. The accuracy of floating point numbers represented by datatype double is about 15 decimal points. These evaluation parameters of simulator are summed up in table 8.

Table 8. Performance analysis parameters of simulator

Performance analysis parameters of simulator		
1.	Software used for simulation	Dev C
2.	Language used for simulation	Visual C++
3.	Number of floating points to represent 1 qubit	4
4.	Data type used to represent qubit values	Double

5.	Number of bits to represent 1 qubit	256
6.	Accuracy up to decimal point	15
7.	Length of Key size	$8\ bit$
8.	Number of random numbers generated	02
9.	Data type to represent 1st random number(randNum)	Int
10.	Data type to represent 2nd random number(random)	Float
11.	Compilation time of BB84	$0.91s$
12.	Compilation time of BB92	$0.86s$

The emulation of BB84 and BB92 protocols was done on FPGA by using Spartan3 device. The language used for writing the code was Verilog. The emulator was designed to exchange 8-bit binary key between transmitter and receiver. Four floating numbers were used to represent $1\ qubit$. The notation used to represent floating numbers was $Q2{,}6$ notations. Each $qubit$ was represented by four floating numbers represented by $Q2{,}6$ notation. Hence, 32 number of bits were used to represent each $qubit$. The 8-bit random number used to choose the bases was generated by pre-defined pseudo random number generator. Another 8-bit random number for the measurement block was also generated by pre-defined pseudo random number generator. The bases used for these protocols were z-bases. For other bases, we used measurement block to decode the bit from $qubit$. These all parameters of emulator are summed up in table.9

Table 9. Performance analysis parameter for emulator

Performance analysis parameters of emulator		
1.	Device used for emulation	FPGA Spartan3
2.	Language used for simulation	Verilog
3.	Total float numbers used to represent one $qubit$	4

4.	Number of bits per float	8
5.	Number of bits to represent $1\ qubit$	32
6.	Notation used to represent float number	$Q(2,6)$
7.	Length of Key size	$8\ bit$
8.	Number of random numbers generated	2
9.	Bits of random number generated	8
10.	Number of registers used to store each $qubit$	4
11.	Number of bits stored on each register	8
12.	Total 8-bit numbers used by 8-bit registers to store $8\ qubits$	32
13.	Total number of bits used to store $8\ qubits$	256

Based on the experimental results of simulation and emulation, the performance of BB84 and BB92 protocol have been compared on the basis of bit error probability, success probability rate and number of lookup tables used.

5.1. Performance Analysis and Results of BB84 and BB92:

In both BB84 and BB92 protocols, receiver is choosing the bases randomly, so there is 50% chance that receiver will choose the same bases as that of transmitter and 50% chance that receiver will choose bases different from transmitter. When receiver is choosing different bases, there is still half of the probability of getting the correct bit by measuring qubit. So, for an ideal system, there is 75% chance of getting correct bit at the receiver side, hence the success probability of ideal system is 75%.

5.1.1. Experimental results for performance analysis of BB84 QKD protocol:

For performance analysis of BB84 protocol, transmitter has generated random 8-bit key and compared the output received on the receiver side with the transmitted key to find the frequency of successful transmission of bits and compared the successful probability of BB84 protocol. This experiment was performed 100 times on simulator and emulator

and successful bit retrieval of each event was recorded. The collective data of number of successful bit retrieval of all the events is provided in the table 10 and shown in figure 17.

Table 10. Experimental data of simulation of BB84 QKD protocol

Number of successful bits retrieved	Count of successful bit received by simulator	multiply	Count of successful bit received by emulator	multiply
0	0	0	0	0
1	1	1	0	0
2	2	4	1	2
3	4	12	5	15
4	20	80	16	64
5	23	115	24	120
6	31	186	40	240
7	13	91	11	77
8	6	48	3	24
Sum	100	537	100	542
Average		5.37		5.42
success probability		67.125		67.75
bit error		7.875		7.2

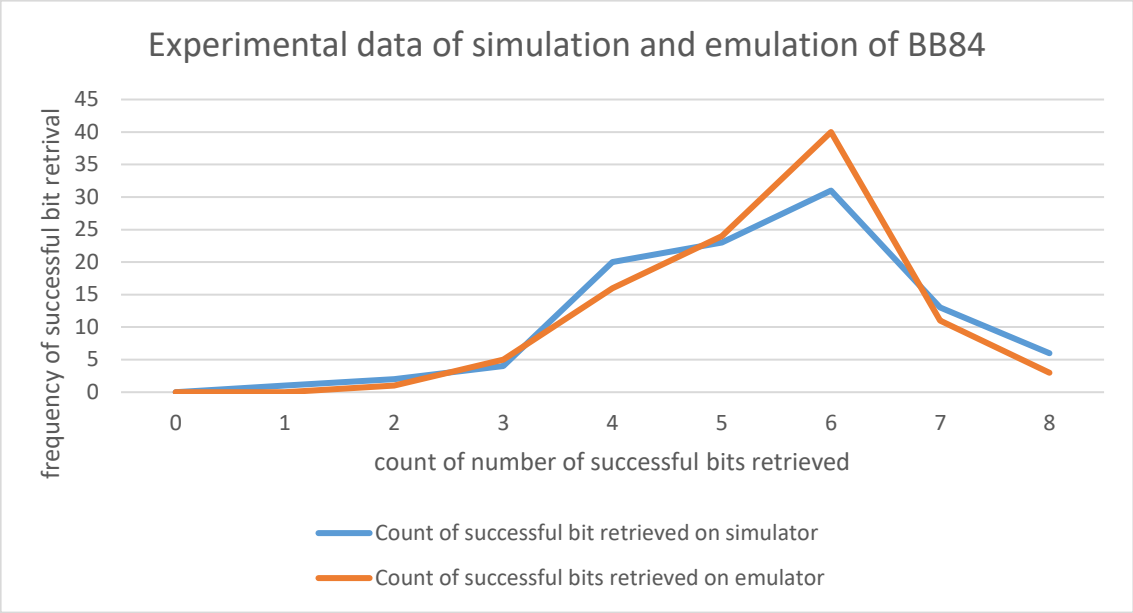

Figure 17. Experimental data of simulation and emulation of BB84

5.1.2. Experimental results for performance analysis of BB92 QKD protocol:

For performance analysis of BB92 protocol, transmitter has generated random 8-bit key and compared the output received on the receiver side with the transmitted key to find the frequency of successful transmission of bits and compared the successful probability of BB92 protocol. This experiment was performed 100 times on simulator and emulator and successful bit retrieval of each event was recorded. The collective data of number of successful bit retrieval of all the events is provided in the table 11 and shown in figure 18.

Table 11. Experimental data of simulation of BB92 QKD protocol

Number of successful bits retrieved	Count of successful bit received by simulator	multiply	Count of successful bit received by emulator	multiply
0	0	0	0	0
1	0	0	0	0
2	2	4	2	4
3	6	18	4	12

4	24	96	14	56
5	25	135	25	125
6	29	174	41	246
7	11	77	9	63
8	3	24	5	40
Sum	100	528	100	546
Average		5.28		5.46
success probability		66		68.25
bit error		9		6.75

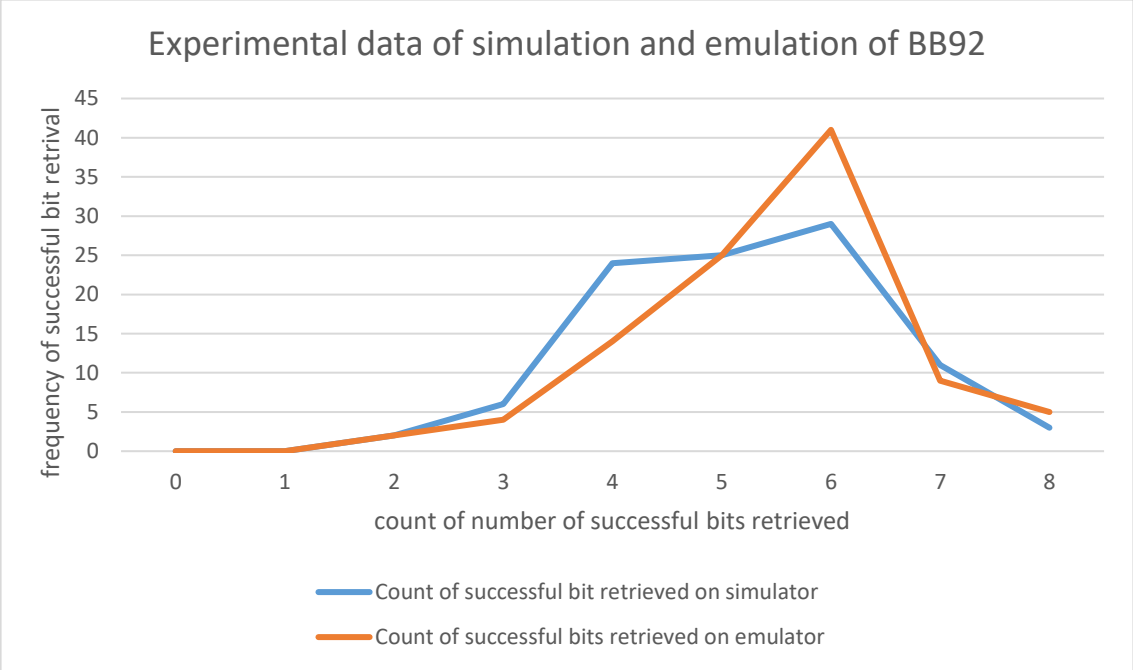

Figure 18. Experimental data of simulation and emulation of BB92

The successful bit retrieval rates and bit error rates obtained by the performed experiments are compiled in table 12. Figure 19 shows the cumulative result of all the experiments performed for analyzing the performance of BB84 and BB92 simulators and emulators.

Table 12. Successful Bit Rate and Bit Error Rates of BB84 and BB92 Protocols

S.No.	Simulator/emulator	Protocol Name	Successful bit retrieval rate	Bit Error
1.	C language	BB84	67.13%	7.87%
2.	C language	BB92	66%	9%
3.	FPGA	BB84	67.75%	7.2%
4.	FPGA	BB92	68.25%	6.75%

The experimental results of Verilog simulation showed that BB84 system has 67.75% success probability with 7.2% –bit error. BB92 system has 68.25% success probability

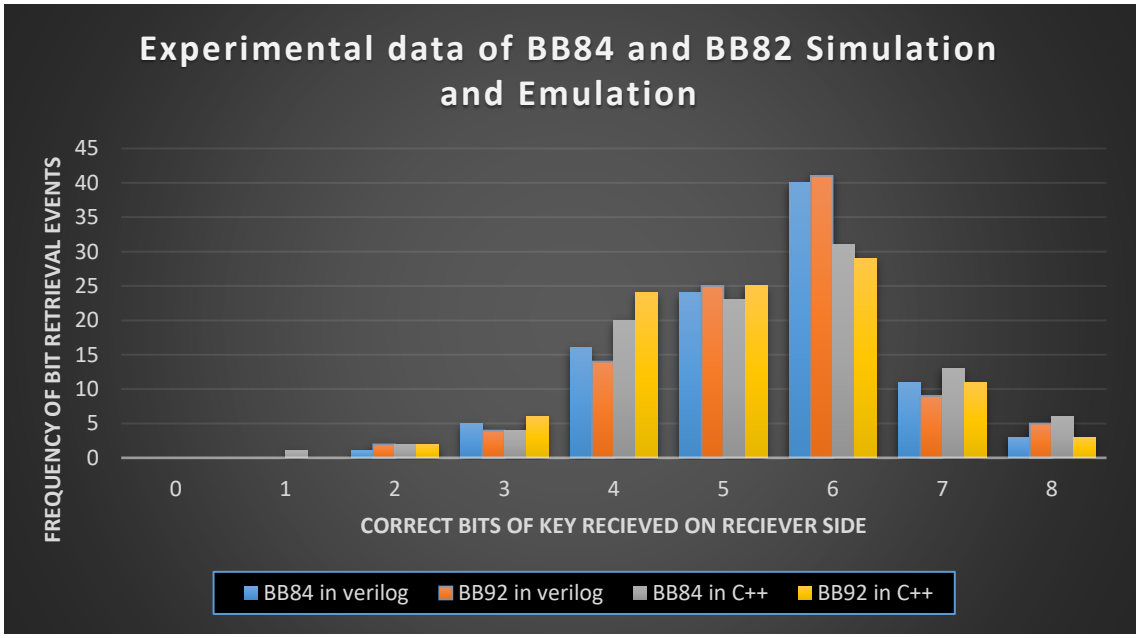

Figure 19. Graph of experimental data of BB84 and BB92 simulation and emulation

with bit error of 6.75%. The simulation results of C++ showed that BB84 system has 67.13% success probability with bit error of 7.87% and BB92 system has 66% success probability with 9%-bit error. The transmitter of BB84 and BB92 protocols utilizes 45 lookup tables and 24 lookup tables for architecture design. The receiver side of BB84 uses 15 lookup tables while the receiver side of BB92 utilizes 17 lookup tables. The summary of resources used for designing architectures of proposed protocols are shown in the table below. It shows that these standalone systems of BB84 and BB92 QKD protocols are cost efficient but increased the error rates.

Table 13. Resources used for BB84 and BB92 protocols

Protocol	**Emulation cost in terms of Lookup tables**	
	Transmitter	Receiver
BB84	45	15
BB92	24	17

Chapter 6: Conclusion

In near future, quantum computers will be in use and study of quantum cryptography will become inevitable. Researchers have already started working on the quantum cryptography protocols. As quantum cryptography is device specific and can only be done by using quantum computers. These quantum computers are large, expensive and complex to use and we have limited access to quantum computers. Considering these limitations, we need some simulation and emulation models that can check the working of key distribution algorithms in the meantime when quantum computers are ready to use. Many researchers have contributed in designing generalized emulators and simulators. The proposed thesis has designed the dedicated emulator that primarily emulates the protocols based on superposition and probabilistic measurement. We have designed a resource efficient standalone system that will help to test the performance of quantum cryptography protocols. Performance analysis of these simulators and emulator is done against the ideal results to test the behavior of quantum key distribution protocol on two different platforms. Simulator was designed by using C language while the emulator was designed on FPGA.

The performance of these QKD protocols was analyzed on the basis of successful bit retrieval probability and cost efficiency. As the ideal cryptography system has bit retrieval rate of 75%, the experimental results of Verilog simulation showed that BB84 system has 67.75% successful bit retrieval probability and BB92 system has 68.25% successful bit retrieval probability. The simulation results of C++ showed that BB84 system has 67.13% −bit retrieval probability and BB92 system has 66% successful bit retrieval probability. The transmitter of BB84 and BB92 protocols utilized 45 and 24 lookup tables for architecture design. The receiver side of BB84 used 15 lookup tables while the receiver side of BB92 utilized 17 lookup tables out of total 2400 lookup tables.

The comparison of simulation results of these protocols in Verilog proved that BB92 has more successful bit retrieval probability as compared to BB84. The success probability of BB84 protocol is better than BB92 protocol in C++. We have checked the feasibility of emulators for basic QKD protocols and obtained encouraging results. The performance of these emulators proved that we can test the quantum key distribution protocols by using digital computers. Now we are one more step closer to quantum computing in implementation and architectures design.

6.1. Future Work:

The proposed thesis has provided the details, simulation and emulation of main quantum key distribution protocols based on probabilistic measurement. The testing and detailed analysis of emulation and simulation of BB84 and BB92 can be done. In the future, we can change the bit size to decrease bit error.

References

1. Cao, Y., et al., *Quantum chemistry in the age of quantum computing.* Chemical reviews, 2019. **119**(19): p. 10856-10915.
2. Shalf, J.M. and R. Leland, *Computing beyond moore's law.* Computer, 2015. **48**(12): p. 14-23.
3. Theis, T.N. and H.-S.P. Wong, *The end of moore's law: A new beginning for information technology.* Computing in Science & Engineering, 2017. **19**(2): p. 41-50.
4. Waldrop, M.M., *The chips are down for Moore's law.* Nature News, 2016. **530**(7589): p. 144.
5. Gyongyosi, L. and S. Imre, *A survey on quantum computing technology.* Computer Science Review, 2019. **31**: p. 51-71.
6. Sadiku, M.N., M. Tembely, and S.M. Musa, *Quantum computing: A primer.* International Journal of Advanced Research in Computer Science and Software Engineering, 2017. **7**(11): p. 129-130.
7. Sehgal, S.K. and R. Gupta. *A Comparative Study of Classical and Quantum Cryptography*. in *2019 6th International Conference on Computing for Sustainable Global Development (INDIACom)*. 2019. IEEE.
8. Nannicini, G., *An introduction to quantum computing, without the physics.* arXiv preprint arXiv:1708.03684, 2017.
9. MARTIN, E., *4 Amazing Quantum Computing Applications.* 2018, April 20.
10. Gossett, S. *8 QUANTUM COMPUTING APPLICATIONS & EXAMPLES*. 2020; Available from: https://builtin.com/hardware/quantum-computing-applications.
11. Mohseni, M., et al., *Commercialize quantum technologies in five years.* Nature, 2017. **543**(7644): p. 171-174.
12. Jackson, M. *6 Things Quantum Computers Will Be Incredibly Useful For*. 2017, June 25; Available from: https://singularityhub.com/2017/06/25/6-things-quantum-computers-will-be-incredibly-useful-for/.
13. Pawar, H.R. and D.G. Harkut. *Classical and Quantum Cryptography for Image Encryption & Decryption*. in *2018 International Conference on Research in Intelligent and Computing in Engineering (RICE)*. 2018. IEEE.
14. Sam, S., *Cryptography: What are Encryption Keys-Symmetric vs Asymmetric.* 2020.
15. Mavroeidis, V., et al., *The impact of quantum computing on present cryptography.* arXiv preprint arXiv:1804.00200, 2018.
16. Brandon, J.A. and T. Shelar, *Shor's Algorithm Simplified1.*
17. Fluhrer, S.R., *Reassessing Grover's Algorithm.* IACR Cryptol. ePrint Arch., 2017. **2017**: p. 811.
18. Sprenkels, D., *Grover's algorithm.*
19. Trizna, A. and A. Ozols, *An overview of quantum key distribution protocols.* Inf. Technol. Manage. Sci, 2018. **21**.
20. Li, J., et al., *A survey on quantum cryptography.* Chinese Journal of Electronics, 2018. **27**(2): p. 223-228.

21. Lakshmi, P.S. and G. Murali. *Comparison of classical and quantum cryptography using QKD simulator*. in *2017 International Conference on Energy, Communication, Data Analytics and Soft Computing (ICECDS)*. 2017. IEEE.
22. Singh, H., D. Gupta, and A. Singh, *Quantum key distribution protocols: a review.* Journal of Computer Engineering, 2014. **16**(2): p. 1-9.
23. Bennett, C.H. and G. Brassard, *Quantum cryptography: Public key distribution and coin tossing.* arXiv preprint arXiv:2003.06557, 2020.
24. Kalra, M. and R.C. Poonia, *Simulation of BB84 and proposed protocol for quantum key distribution.* Journal of Statistics and Management Systems, 2018. **21**(4): p. 661-666.
25. Poornima, A., N. Naghabhushana, and R. Ujjinimatad, *Matrix Representation of Quantum Gates.* Int. J. Comp. App.(0975-8887), 2017. **159**.
26. Lopes, M. and N. Sarwade, *Cryptography from quantum mechanical viewpoint.* arXiv preprint arXiv:1407.2357, 2014.
27. Hassanien, A.E., M. Elhoseny, and J. Kacprzyk, *Quantum computing: an environment for intelligent large scale real application*. 2018: Springer.
28. Yanofsky, N.S. and M.A. Mannucci, *Quantum computing for computer scientists*. 2008: Cambridge University Press.
29. Polyakov, K.O. and N.G. Butakova. *Comparative Analysis of Post-Quantum Key Transfer Protocols Using Mathematical Modeling*. in *2020 IEEE Conference of Russian Young Researchers in Electrical and Electronic Engineering (EIConRus)*. 2020. IEEE.
30. Khan, M.M., M. Murphy, and A. Beige, *High error-rate quantum key distribution for long-distance communication.* New Journal of Physics, 2009. **11**(6): p. 063043.
31. Giampouris, D., *Short review on quantum key distribution protocols*, in *GeNeDis 2016*. 2017, Springer. p. 149-157.
32. Serna, E.H., *Quantum Key Distribution from a random seed.* arXiv preprint arXiv:1311.1582, 2013.
33. Trushechkin, A., et al., *Quantum-key-distribution protocol with pseudorandom bases.* Physical Review A, 2018. **97**(1): p. 012311.
34. Patil, P.A. and R. Boda, *Analysis of Cryptography: Classical verses Quantum Cryptography.* International Research Journal of Engineering and Technology (IRJET), 2016. **3**(05).
35. Moizuddin, M., J. Winston, and M. Qayyum. *A comprehensive survey: quantum cryptography*. in *2017 2nd International Conference on Anti-Cyber Crimes (ICACC)*. 2017. IEEE.
36. Nurhadi, A.I. and N.R. Syambas. *Quantum key distribution (QKD) protocols: a survey*. in *2018 4th International Conference on Wireless and Telematics (ICWT)*. 2018. IEEE.
37. Camargo, A., et al., *Simulation of the BB84 protocol of quantum cryptography by using an intense laser beam.* Revista Brasileira de Ensino de Física, 2017. **39**(2).
38. Benletaief, N., H. Rezig, and A. Bouallegue, *Toward efficient quantum key distribution reconciliation.* arXiv preprint arXiv:2002.04887, 2020.
39. Kalra, M. and R.C. Poonia, *A Two Way Synchronized Quantum Channel Quantum Key Distribution Protocol.* Recent Patents on Computer Science, 2018. **11**(4): p. 255-261.
40. Arshinov, N.A. and N.G. Butakova. *Modeling of quantum channel parameters impact on information exchange security*. in *2018 IEEE Conference of Russian Young Researchers in Electrical and Electronic Engineering (EIConRus)*. 2018. IEEE.

41. Pilch, J. and J. Długopolski, *An FPGA-based real quantum computer emulator.* Journal of Computational Electronics, 2019. **18**(1): p. 329-342.
42. Silva, A. and O.G. Zabaleta. *FPGA quantum computing emulator using high level design tools*. in *2017 Eight Argentine Symposium and Conference on Embedded Systems (CASE)*. 2017. IEEE.
43. Steiger, D.S., T. Häner, and M. Troyer, *ProjectQ: an open source software framework for quantum computing.* Quantum, 2018. **2**: p. 49.
44. Mahmud, N. and E. El-Araby. *A scalable high-precision and high-throughput architecture for emulation of quantum algorithms*. in *2018 31st IEEE International System-on-Chip Conference (SOCC)*. 2018. IEEE.
45. Jones, T. and S.C. Benjamin, *QuESTlink–Mathematica embiggened by a hardware-optimised quantum emulator.* Quantum Science and Technology, 2020.
46. Lee, Y.H., M. Khalil-Hani, and M.N. Marsono, *An FPGA-based quantum computing emulation framework based on serial-parallel architecture.* International Journal of Reconfigurable Computing, 2016. **2016**.
47. Iwakoshi, T., *Trade-off between key generation rate and security of BB84 quantum key distribution.* Tamagawa University Quantum ICT Research Institute Bulletin, 2015. **5**(1): p. 1-4.
48. Burenkov, V., et al., *Security of high speed quantum key distribution with finite detector dead time.* arXiv preprint arXiv:1005.0272, 2010.
49. Zhao, Y. *Development of Quantum Key Distribution and Attacks against It*. in *Journal of Physics: Conference Series*. 2018.
50. Brassard, G., et al., *Limitations on practical quantum cryptography.* Physical Review Letters, 2000. **85**(6): p. 1330.
51. Brassard, G., et al. *Security aspects of practical quantum cryptography*. in *International conference on the theory and applications of cryptographic techniques*. 2000. Springer.
52. Foong, O.-M., T.J. Low, and K.W. Hong, *Simulation study of single quantum channel BB84 quantum key distribution*, in *IT Convergence and Security 2017*. 2018, Springer. p. 159-167.
53. Li, H.-F., et al. *The improvement of QKD scheme based on bb84 protocol*. in *2016 International Conference on Information System and Artificial Intelligence (ISAI)*. 2016. IEEE.
54. Upadhyay, L. *Quantum Cryptography: A Survey*. in *International Conference on Innovations in Bio-Inspired Computing and Applications*. 2018. Springer.
55. Su, H.-Y., *Simple analysis of security of the BB84 quantum key distribution protocol.* Quantum Information Processing, 2020. **19**: p. 1-15.
56. Zisu, L., *A method to improve the BB84 protocol.* Scientific Bulletin" Mircea cel Batran" Naval Academy, 2019. **22**(1): p. 1-7.
57. Kour, J., S. Koul, and P. Zahid, *A SURVEY ON QUANTUM KEY DISTRIBUTION PROTOCOLS.*

Thesis

ORIGINALITY REPORT

13%	9%	9%	7%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	Submitted to University of Bristol Student Paper	1%
2	docplayer.net Internet Source	1%
3	export.arxiv.org Internet Source	<1%
4	www.hindawi.com Internet Source	<1%
5	etheses.whiterose.ac.uk Internet Source	<1%
6	thesai.org Internet Source	<1%
7	www.ll.mit.edu Internet Source	<1%
8	Submitted to Riga Technical University Student Paper	<1%
9	Submitted to University of Liverpool Student Paper	<1%

10	Laszlo Gyongyosi, Sandor Imre. "A Survey on quantum computing technology", Computer Science Review, 2019 Publication	<1%
11	Submitted to Myongji University Graduate School Student Paper	<1%
12	Submitted to GEMS Wellington Academy Student Paper	<1%
13	onlinelibrary.wiley.com Internet Source	<1%
14	www.comp.leeds.ac.uk Internet Source	<1%
15	Naveed Mahmud, Esam El-Araby. "A Scalable High-Precision and High-Throughput Architecture for Emulation of Quantum Algorithms", 2018 31st IEEE International System-on-Chip Conference (SOCC), 2018 Publication	<1%
16	Naveed Mahmud, Bennett Haase-Divine, Annika Kuhnke, Apurva Rai, Andrew MacGillivray, Esam El-Araby. "Efficient Computation Techniques and Hardware Architectures for Unitary Transformations in Support of Quantum Algorithm Emulation", Journal of Signal Processing Systems, 2020 Publication	<1%

17	www.spiedigitallibrary.org Internet Source	<1%
18	ethesis.nitrkl.ac.in Internet Source	<1%
19	Melissa de Oliveira Santos, Welerson Santos Souza, Thiago de Andrade Bragagnolle, Leonardo Diogo Bueno Bobadilla Ivan Aldaya et al. "All-optical Spectral Shuffling Applied to 16-QAM Signals", 2019 SBFoton International Optics and Photonics Conference (SBFoton IOPC), 2019 Publication	<1%
20	Submitted to University of West London Student Paper	<1%
21	Matthew N. O. Sadiku, Chandra M. M. Kotteti, Sarhan M. Musa. "Nano Computing: An Introduction", International Journal of Advances in Scientific Research and Engineering, 2019 Publication	<1%
22	www.edadesignline.com Internet Source	<1%
23	Submitted to British University in Egypt Student Paper	<1%
24	Submitted to Heriot-Watt University Student Paper	<1%

25	www.eurekaselect.com Internet Source	<1%
26	Meenakshi Sharma, Sonia Thind. "A Quantum Key Distribution Technique Using Quantum Cryptography", International Journal of Distributed Artificial Intelligence, 2019 Publication	<1%
27	Submitted to University of Oxford Student Paper	<1%
28	Deepti Raj, A. B. Kalpana, Manoj Kumar Singh. "State encoding for low power in FSM using non-oscillating self-adaptive particle swarm optimization (NOS-SAPSO)", Journal of Information and Optimization Sciences, 2020 Publication	<1%
29	"Quantum Computing:An Environment for Intelligent Large Scale Real Application", Springer Science and Business Media LLC, 2018 Publication	<1%
30	Submitted to University of Hertfordshire Student Paper	<1%
31	perso.telecom-paristech.fr Internet Source	<1%
32	Submitted to University of Western Sydney Student Paper	<1%

33	quantum-journal.org Internet Source	<1%
34	depcom.uqac.ca Internet Source	<1%
35	www.ijcse.net Internet Source	<1%
36	scholarworks.bridgeport.edu Internet Source	<1%
37	Submitted to University of Greenwich Student Paper	<1%
38	www.mhsl.uab.edu Internet Source	<1%
39	www.sopmac.de Internet Source	<1%
40	Submitted to University of Warwick Student Paper	<1%
41	Yao Zhang, Qiang Ni. "Recent Advances in Quantum Machine Learning", Quantum Engineering, 2020 Publication	<1%
42	Submitted to Queensland University of Technology Student Paper	<1%
43	"Communications, Signal Processing, and	

	Systems", Springer Science and Business Media LLC, 2020 Publication	<1%
44	en.misis.ru Internet Source	<1%
45	Submitted to Canterbury Shool Student Paper	<1%
46	Paolo Zuliani. "Reasoning about faulty quantum programs", Acta Informatica, 2009 Publication	<1%
47	Submitted to University of Wales Institute, Cardiff Student Paper	<1%
48	livrepository.liverpool.ac.uk Internet Source	<1%
49	Submitted to Imperial College of Science, Technology and Medicine Student Paper	<1%
50	"Artificial Intelligence and Evolutionary Computations in Engineering Systems", Springer Science and Business Media LLC, 2020 Publication	<1%
51	Submitted to Melbourne Institute of Technology Student Paper	<1%

52	Submitted to The Manchester College Student Paper	<1%
53	www.osti.gov Internet Source	<1%
54	Submitted to Queen Mary and Westfield College Student Paper	<1%
55	Klaas Gunst, Frank Verstraete, Sebastian Wouters, Örs Legeza, Dimitri Van Neck. "T3NS: Three-Legged Tree Tensor Network States", Journal of Chemical Theory and Computation, 2018 Publication	<1%
56	Nikita A. Arshinov, Natalia G. Butakova. "Modeling of quantum channel parameters impact on information exchange security", 2018 IEEE Conference of Russian Young Researchers in Electrical and Electronic Engineering (EIConRus), 2018 Publication	<1%
57	Submitted to uhasselt Student Paper	<1%
58	link.springer.com Internet Source	<1%
59	www.ejournal.org.cn Internet Source	<1%

60	arxiv.org Internet Source	<1%
61	Submitted to Northwest Missouri State University Student Paper	<1%
62	Reju, V.G.. "Partial separation method for solving permutation problem in frequency domain blind source separation of speech signals", Neurocomputing, 200806 Publication	<1%
63	www.mysciencework.com Internet Source	<1%
64	eprints.gla.ac.uk Internet Source	<1%
65	M. Tchoffo, A.G. Tene. "Privacy amplification of entanglement parametric-down conversion based quantum key distribution via quantum logistic map for photon bases choice", Chaos, Solitons & Fractals, 2020 Publication	<1%
66	vufind.katalog.k.utb.cz Internet Source	<1%
67	www.nature.com Internet Source	<1%
68	ijrcct.org	

	Internet Source	<1%
69	www.crossref.org Internet Source	<1%
70	Ramesh C. Poonia, Vijander Singh, Linesh Raja. "Special Issue on Smart Technologies in Engineering-PART 1", Recent Patents on Computer Science, 2018 Publication	<1%
71	Kirill O. Polyakov, Natalia G. Butakova. "Comparative Analysis of Post-Quantum Key Transfer Protocols Using Mathematical Modeling", 2020 IEEE Conference of Russian Young Researchers in Electrical and Electronic Engineering (ElConRus), 2020 Publication	<1%
72	mindsofmexico.org Internet Source	<1%
73	Harshad R. Pawar, Dinesh G. Harkut. "Classical and Quantum Cryptography for Image Encryption & Decryption", 2018 International Conference on Research in Intelligent and Computing in Engineering (RICE), 2018 Publication	<1%
74	H.F. Chau. "Unconditionally Secure Key Distribution in Higher Dimensions by	<1%

Depolarization", IEEE Transactions on Information Theory, 2005
Publication

75	worldwidescience.org Internet Source	<1%
76	creativecommons.org Internet Source	<1%
77	Zhe Chen, Hang Wong, Jun Xiang. "A Polarization Reconfigurable Dielectric Resonator Antenna with Agile Switch", 2019 IEEE International Conference on Computational Electromagnetics (ICCEM), 2019 Publication	<1%
78	explora.unex.es Internet Source	<1%
79	Nedasadat Hosseinidehaj, Zunaira Babar, Robert Malaney, Soon Xin Ng, Lajos Hanzo. "Satellite-Based Continuous-Variable Quantum Communications: State-of-the-Art and a Predictive Outlook", IEEE Communications Surveys & Tutorials, 2019 Publication	<1%
80	Nedasadat Hosseinidehaj, Zunaira Babar, Robert Malaney, Soon Xin Ng, Lajos Hanzo. "Satellite-Based Continuous-Variable Quantum Communications: State-of-the-Art and a	<1%

Predictive Outlook", IEEE Communications Surveys & Tutorials, 2018
Publication

---

81 Jakub Pilch, Jacek Długopolski. "An FPGA-based real quantum computer emulator", Journal of Computational Electronics, 2018 — <1%
Publication

---

82 www.utwente.nl — <1%
Internet Source

---

83 Manish Kalra, Ramesh C. Poonia. "Simulation of BB84 and proposed protocol for quantum key distribution", Journal of Statistics and Management Systems, 2018 — <1%
Publication

---

84 Phuc V. Trinh, Thanh V. Pham, Ngoc T. Dang, Hung Viet Nguyen, Soon Xin Ng, Anh T. Pham. "Design and Security Analysis of Quantum Key Distribution Protocol Over Free-Space Optics Using Dual-Threshold Direct-Detection Receiver", IEEE Access, 2018 — <1%
Publication

---

85 ira.lib.polyu.edu.hk — <1%
Internet Source

---

86 Yong Min Lee, Seong Pal Lee. "A study on the preliminary RF design of payload for satellite communication (SATCOM) system", The 7th International Conference on Advanced — <1%

Communication Technology, 2005, ICACT 2005., 2005
Publication

87	www.science.gov Internet Source	<1%
88	Peng-Yong Kong. "A Review of Quantum Key Distribution Protocols in the Perspective of Smart Grid Communication Security", IEEE Systems Journal, 2020 Publication	<1%
89	Jeong San Kim, Gilad Gour, Barry C. Sanders. "Limitations to sharing entanglement", Contemporary Physics, 2012 Publication	<1%
90	Vasileios Mavroeidis, Kamer Vishi, Mateusz D., Audun Jøsang. "The Impact of Quantum Computing on Present Cryptography", International Journal of Advanced Computer Science and Applications, 2018 Publication	<1%

Exclude quotes	Off	Exclude matches	Off
Exclude bibliography	Off		