Applying Multi-Parity Code To The Quantum Security Protocol BB84 Under Different Types Of Attacks

Dr. A. I. A. Jabbar
Assistant Professor
Department of ElectricalEngineering
Mosul university

Ahmed I. A.
M.Sc. Student
Department of ElectricalEngineering
Mosul university

Abstract

Quantum Key Distribution QKD mechanism is based on the principles of quantum mechanics to guarantee the secure exchange of secret keys between users. In this paper, the BB84 protocol is simulated, and the enhancement of the protocol using multi-parity instead of single parity is introduced with the error correction unit. The sub-block length is changed dynamically in an adaptive way and according to the QBER values, also this study takes into account the channel effect on the protocol by applying four types of channels (perfect channel, low noise channel, medium noise channel and high noise channel) and all channels are assumed to be lossless. The study includes also the effect of three types of attacks (PNS, IR, PNS & IR attacks) onto the protocol performance. The simulation results show that (IR) attack have the strongest effect on the BB84 performance.

Keywords : BB84, QKD, Multi-Parity, Quantum Cryptography, Security, PNS, IR, PNS & IR, attack.

تطبيق طريقة المماثلة المتعددة على بروتوكول السرية الكمية BB84 تحت انواع مختلفة من الهجمات

احمد إبراهيم احمد
طالب ماجستير
قسم الهندسة الكهربائية
جامعة الموصل

د. عبد الإله عبد الجبار
أستاذ مساعد
قسم الهندسة الكهربائية
جامعة الموصل

الخلاصة

ميكانيكية توزيع المفتاح الكمي تعتمد على مبادئ ميكانيكا الكم لتضمن تبادل امن للمفاتيح السرية بين المستخدمين. في هذا البحث تم محاكاة البروتوكول BB84 وتحسين ادائه عن طريق استخدام المماثلة المتعددة بدلا من المماثلة المنفردة في مرحلة تصحيح الاخطاء، كذلك فان اطوال المجاميع المحتسبة تكون متغيرة ديناميكا بطريقة انتقائية وبحسب معدل الخطأ في وحدة المعلومة الكمية QBER. تم في هذه الدراسة ايضا اخذ تأثير القناة الكمية على اداء البروتوكول BB84، عن طريق تطبيق اربعة انواع مختلفة من القنوات (قناة مثالية ، قناة ذات ضوضاء منخفضة، قناة ذات ضوضاء متوسطة وقناة عالية الضوضاء) بافتراض ان جميع القنوات لا تحوي على خسائر. هذه الدراسة تضمنت كذلك اخذ تأثير ثلاثة انواع من الهجمات على اداء البروتوكول BB84 (هجوم تقسيم عدد الفوتونات PNS، هجوم الاعتراض واعادة الارسال IR وهجوم مزيج من الهجومين السابقين PNS & IR). حيث اظهرت نتائج المحاكاة ان هجوم الاعتراض واعادة الارسال هو الاكثر تأثيرا على اداء البروتوكول BB84.

Received: 3-12 - 2012 Accepted: 7– 3 - 2013

1. Introduction

Public-key encryption systems such as RSA [1] starts to be unable to withstand the attack of intruders due to the development of new cryptanalyst is methods and the huge increment of the processors computing power, it is obvious that if quantum computers are applied practically, then most of the current public-key cryptosystems would be vanished [2]. Conventionally one-time pad provides perfect information security but it has a big problem concerning the length of the random key which should be equal to the length of the plaintext. [3, 4, 5].

Quantum security is considered as an excellent and unbreakable way of encryption, it is based on the laws of quantum mechanics, which deals with elementary particles that do not have a precise location and speed simultaneously. This means that obtaining a particle's location will destroy the information concerning its speed – and vice versa – (*Heisenberg uncertainty principle*)[6].

Bennett and Brassard [3] developed the protocol BB84 which is based on the uncertainty principle, it provides guaranteed security during the exchange of a secret key, it is worth to mention that according to this principle, it is impossible to know everything about the photons that carry the key bits. This means that intruders will certainly impose errors on the transmission channel that can be detected by the sender and receiver sides [6].

Many researchers have investigated this subject, for example:

Stefan Rass and Christian Kollmitzer present an improvement to the error correction facility which yields a significant improvement on the BB84 protocol performance [7]. The classical cryptographic protocols, which are required for quantum key distribution are documented by Nikolaos Papanikolaou and Rajagopal Nagarajan then they suggest a model for the analysis of these protocols and verification of relevant security requirements. Alan Mink, Sheila Frankel and Ray Perlner [8] prepare an overview of the quantum key distribution (QKD) showing how QKD could be applied within security systems.

In this paper, a new method for error correction is proposed using Multi-Parity instead of single parity and a comparison between three types of attacks on BB84 is included.

2. The Principles of the ProtocolBB84

Figure (1) shows a simple QKD system which is the main task to be executed in the quantum security system. Figure(2) shows the sequence diagram of BB84, given that k_a is the pre-shared secret key required for authentication and K_f is the final key which is generated after the execution of the protocol BB84.

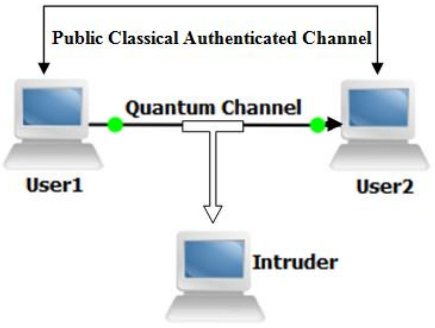


Figure (1) – simple representation of the QKD system

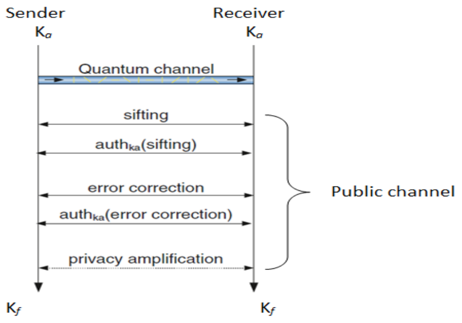


Figure (2) - abstract sequence diagram of BB84

According to BB84 protocol, a user must apply the following steps:

1. The secret key (Raw Key) will be transmitted as photons, which is the physical representation of the qubits (shorthand for Quantum Bit), from the sender to receiver as shown in Figure(3).
2. Switch the system to the public channel .
3. Authenticate the exchanged messages and check for message integrity.
4. The sender and receiver negotiate which bits are used and which bits are to be neglected (Public Discussion stage), depending on their selected prepare/ measurement basis for each transmitted / received photon.
5. The function of the sifting stage is to neglect all results which based on different basis, they also neglect the bit if the receiver's detector failed to detect the photon. The key in this stage is called sifted key as shown in Figure(3).
6. Move to error correction stage.
7. Because of the noise in the quantum channel and the intruder attack, the end users will not share the same identical key string (small errors in the receiver's key string), hence the error in the key must be corrected in the error correction stage. Again intruder scan modify messages during this stage. Therefore, end users must authenticate this stage [9].
8. The sender and receiver now share an initially identical key string. But this key will not be used yet because there is a possibility that an intruder may have some information about it [3].
9. The function of the privacy amplification stage is to minimize the effect of Intruders during the error correction stage, and maybe also during the quantum transmission. The users must map their strings via a function to a smaller subsets, so that the intruder's task will be more complicated and consequently the end users will share a secret key which is known by them only[10].
10. Since users will chose two bases (for transmission and measurement) randomly and independently then the final usable key length will be further reduced, the probability of using the same basis is 1/2 * 1/2 + 1/2 * 1/2 = 1/2 [11]. This means that the length of the key after the sifting stage is no more than 1/2 of the received photons as it will be verified later by the results of simulation, the error correction and privacy amplification stages will introduce further length reduction.

These steps are shown in Table (1) which summarize the BB84 performance steps in the absence of intruders.

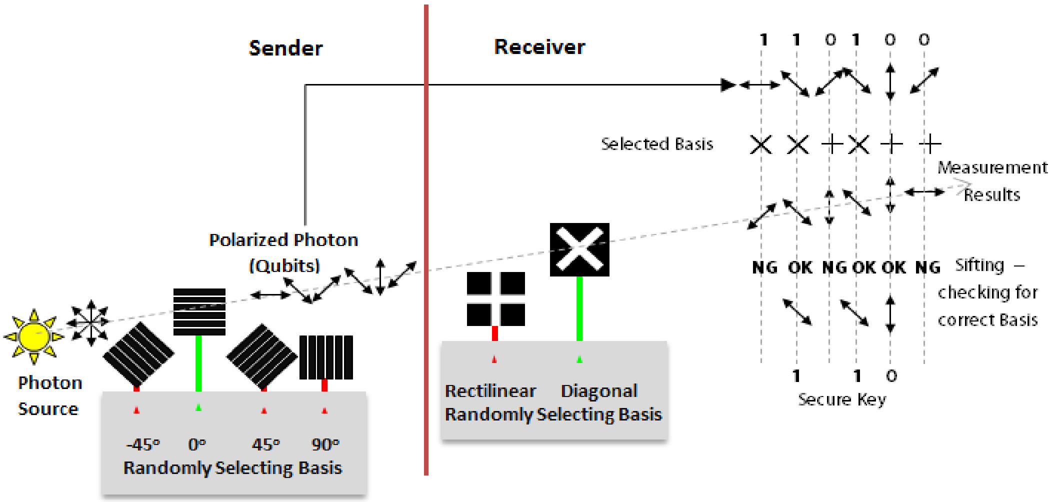


Figure (3) – shows the BB84 after sifting stage

Table (1) – shows the different steps of BB84 protocol

Sender's Random Bits	0	0	1	1	1	1	0	0
Sender's Random Basis	+	×	+	×	+	+	×	+
Sender's States	↑	↗	→	↘	→	→	↗	↑
Receiver's Random measuring Basis	+	+	+	×	+	×	+	×
Receiver's photon polarization Measures	↑	→	→	↘	→	↗	↑	↗
Public Discussion of Basis	✓	✗	✓	✓	✓	✗	✗	✗
Key after sifting	↑		→	↘	→			
Shared secret key	0		1	1	1			
Error in key	✓		✓	✓	✓			

3. Attack Strategies on QKD Protocols

In this study, and in order to check the immunity of the quantum protocol against attack, three types of attacks are applied: Photon number splitting attack PNS, Intercept & resend attack IR and a combination of the Photon number splitting and intercept & resend attack [11].

3.1Intercept& Resend Attack

In this attack, the intruder [12]:

1. Intercepts the transmitted photon and measures it by choosing one of the two basis (Rectilinear or Diagonal) independently and randomly.
2. Resend a new photon (its polarization identical to his measurement) to the receiver.
3. Listen to the public channel during sifting stage.

The QBER caused by this attack will be about 25%.

3.2Photon Number Splitting Attack

This type of attack is considered as the strongest attack, because the photon source is not perfect and the emitting pulses will contain more than one photon in the same pulse, and have the same polarization. This attack has no effect on the key (QBER=0) [8, 12, 13, 14, 15].

The intruder will apply these steps as follows:

1- All pulses which contain no photon are neglected.
2- All pulses which contain more than one photon will attack (by sniff-off) one photon from the pulse and let the other photons in the pulse pass to the receiver without disturbing their polarization.
3- All pulses which contain one photon will pass to the receiver.

3.3Photon Number Splitting and Intercept & Resend Attack

This attack includes the Photon number splitting in the case of multi photon source plus the emitted pulses that contain one photon are subjected to the Intercept & resend attack.

4 The Proposed Model of Key Exchange

4.1 Estimation of Quantum Bit Error Rate (QBER)

Initially, the Sender and receiver must estimate the Quantum Bit Error Rate (QBER), this can be achieved by revealing a portion of the sifted key between them, the computed QBER is used to detect a possible eavesdropper. After that the revealed bits will be discarded from the sifted key, this will cause a reduction in the length of the final key. Finally the computed QBER will be compared with a threshold error or Max Error level and if the QBER >error$_{max}$ they stop the communication and repeat the processes on a different channel.

The revealing portion can be calculated as follows:

$$RevealingPortion = 0.08 * Length(Sifted\ Key) \tag{1}$$

$$QBER = \frac{N_{wrong}}{N_{total}} \tag{2}$$

Where, N$_{wrong}$ Represents the number of different bits in the Revealing portion, N$_{total}$ represents the revealing portion [11].

4.2 Selection of a Suitable Block Length

Now both sender and receiver will divide the sifted key into SubBlocks, the length (L) of each SubBlock will be dynamically chosen depending on the QBER and also depending on the round number. in this study, the Block length is (5 ≤ L ≤ n) where n ≤sifted key length. They exchange the parities between them and apply the Bisective search algorithm to find the error. This requires many rounds and in each round a new Block length is applied.

The formula that used in this simulation to calculate the SubBlock length is given by :

$$L = \left(\frac{1}{QBER}\right) * \left(0.3 * j^{1.2}\right) \tag{3}$$

Where, j represents the current Round number

This formula is suitable and effective after testing it in many cases of simulation and it gives a good result and correct the errors in few rounds.

It's worth to mention in this simulation that after applying 20 rounds on the sifted key the sender and receiver can correct all errors in the key successfully. The basic steps that take place in the error correction stages for the first four rounds are shown in Flowchart (A.3) given in appendix (A).

4.3 Error Correction of the Quantum Key

Quantum key distribution must be common and secret. The errors must be corrected, whether they are caused by intruders or by imperfections during the setup process or by the quantum channel. To achieve this target, secret-key distillation techniques from classical information theory must be used, taking into account that even a professional intruder will have no information about the key.

Unlike the traditional way of using single parity, odd and even parity bits are used in this paper, even parity bit is obtained from bits with even locations in the current subblock while odd parity bit is obtained from bits with odd locations in the current subblock. After that the sender and receiver will exchange the computed parity bits. The minimum subblock length with multi-parity, could be estimated from the case with minimum length using single parity, (the subblock length with single parity could be 2 bits). In this study, the subblock length is proposed to be at least 5 bits to make the key guessing by the intruder difficult and to prevent information leakage to intruder during this stage .

A comparison between the single parity and multi-parity are shown in Figure(5), it is possible to conclude that the multi-parity will correct the errors faster than the single-parity, this means also that the number of rounds needed will be less than the number required by the single parity technique.

4.4Estimate Intruder's Information

In this study, two types of photon sources are used, the perfect single photon source and the weak coherent pulse WCP (which transmits a pulse contains no photon, one photon and more than one photon according to the passion distribution). Practically there is no perfect photon source with the current technology, so the intruder can use the imperfections in the sources and apply the PNS attack on the raw key or even apply the IR attack. After that the sender and receiver should estimate the intruder information about the raw key and remove it from the final key to prevent the intruder from getting any information about the final key.

The estimated bits by the intruder during quantum transmission can be calculated as follows[16]:

1. In the case of Intercept & Resend attack

$$W = N_s\left(\frac{4}{\sqrt{2}}\right) * QBER + 5\sqrt{N_s\left(4+2\sqrt{2}\right) * QBER} \qquad (4)$$

2. If Photon Number Splitting attack is present :

$$W = N_s\,\mu + 5\sqrt{N_s\mu(1-\mu)} \qquad (5)$$

3. The estimated bits with Photon Number Splitting and Intercept & Resend attack will be given by:

$$W = N_s\,\rho + 5\sqrt{N_s\left(N_s\mu(1-\mu) + \left(4+2\sqrt{2}\right) * QBER\ \right)} \qquad (6)$$

$$\rho = \mu + \frac{4}{\sqrt{2}} * QBER \qquad (7)$$

Where (μ) represents the mean photon number (Pulse Intensity), N_s : number of successful pulses (Sifted Key length).

4.5 Privacy Amplification Stages

In this stage, both users should calculate the safety parameter (S) to remove the dissimilar bits from the obtained key after the error correction stage, the safety parameter is used to reduce the intruder's information about the key that may be obtained during the Error correction stages to a minimum value. In this simulation, the safety parameter calculated depends on the QBER value and the length of key in the error correction stage. It is assumed that the safety parameter is in a relationship with QBER to make the estimation of the key by intruder more difficult . The following proposed equation can be used to calculate the safety parameter :

$$S = QBER * 100 + \left(\frac{QBER}{2}\right) * Length(Key\ after\ error\ correction\ ) \qquad (8)$$

The Privacy Amplification apply hash function on the Corrected key to get the final key, the hash function after dividing the corrected key into subblocks will calculate the even or odd parity but this time keep the result without exchanging it, the parity calculation depends on the length of the subblock as illustrated in the following algorithm:

```
If ( length ( current subblock ) is even ) then
        Calculate the even parity ( current subblock)
Else
        Calculate the odd parity (current subblock)
end
```

So the final key length will be :

$$Final\ Key\ = \frac{1}{2} Raw\ Key - Revealing\ Portion\ to\ estimate\ the\ QBER$$
$$- Number\ of\ Discarded\ Bits\ during\ error\ correction - W - S$$

Where S is the safety parameter and W is the estimated Intruder information about the raw key

5. Model Assumptions

The proposed model is based on BB84 protocol, it will be simulated according to the following assumptions:

1. It uses four Quantum States to send qubits, see Table (2).
2. Four photon polarizations will be used (0° , 45° , 90° and 135°) to represent the Quantum States, see Table (3).
3. Two types of Basis will be used to generate/measure the following Polarized photons:
   - i. Diagonal Base (D) X. it is used to generate/measure the |45>& |135> states.
   - ii. Rectilinear Base (R) +. it is used to generate/measure the |0>& |90> states.
4. Perfect and different noise level channels are used.
5. Three types of attacks are used, as follows:
   - i. Photon Number Splitting attack.
   - ii. Intercept & Resend attack.
   - iii. Photon Number Splitting with Intercept & Resend attack.

Table (2) – shows the symbols used to represent the four quantum states

Rectilinear & Diagonal States		Symbol
|0>	→	H
|90>	↑	V
|45>	↗	X
|135> = |-45>	↘	Z

Table (3) – shows the decoding of the four quantum states into binary values

Rectilinear	Diagonal	Value Bit
|0> = |H>	|135> = |Z>	1
|90> = |V>	|45> = |X>	0

6. Simulation Results

To validate the proposed algorithm, it is required first to simulate the protocol BB84, Flowchart(A.1&A.2) which are given in appendix (A) details the simulation steps of this protocol. Figure(4) shows a simple system (Two personal computers and a switch) which is used to simulate the quantum protocol BB84.

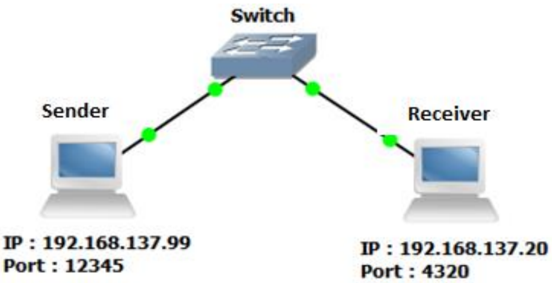


Figure (4) – Schmatic diagram of the hardware implementation of BB84

Figure(5) shows a comparison between the proposed Multi-parity and Single-parity, it is possible to notice that the proposed method eliminates the errors in the key faster than the classical method. Figure(6) shows the GUI of the multi-parity error correction stage being used with BB84 protocol. Note that the error will be reduced to less than 1% at the fourth round and the difference in the key length is 191 bits . but after few rounds the error becomes zero.

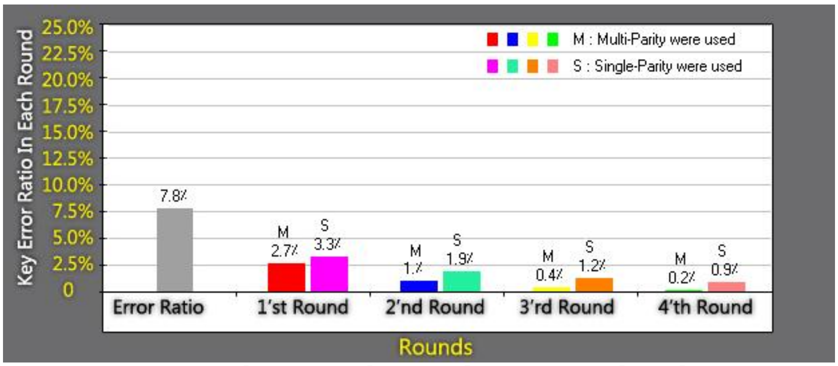
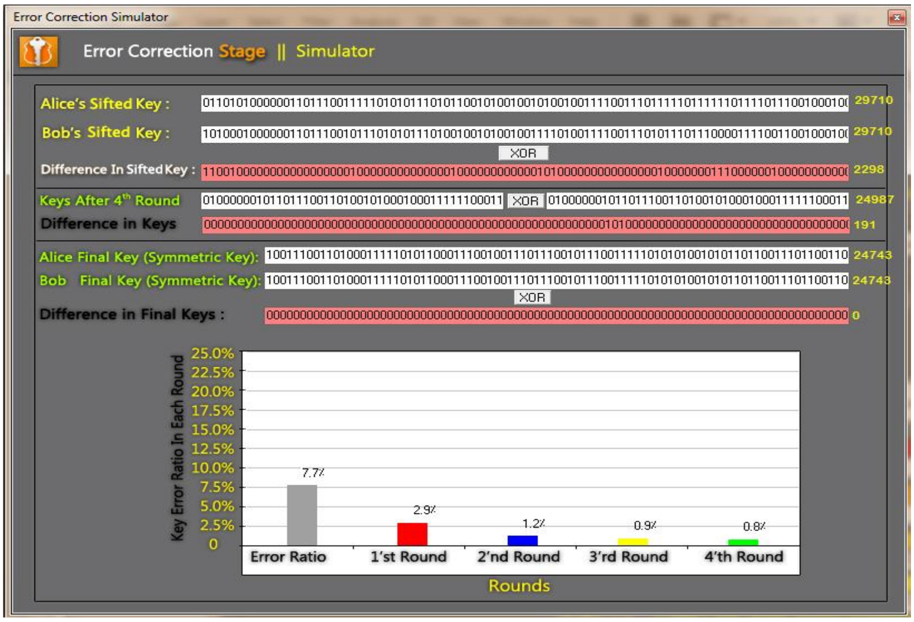


Figure (5) – shows a comparison between the Single-parity and Multi-Parity codes for the first four rounds of simulation (Raw Key = 60000Bits).


Error Correction Simulator
Error Correction Stage || Simulator

Alice's Sifted Key : 0110101000000110111001111101010111010110010100100101001001111001110111110111111011110111001000100(29710

Bob's Sifted Key : 1010001000000110111001011101010111010010010100100111101001111001110101110111000011110011001000100(29710

XOR

Difference In Sifted Key : 1100100000000000000000100000000000000010000000000000101000000000000000100000001110000001000000000(2298

Keys After 4th Round 0100000010110111001101001010001000111111100011 XOR 0100000010110111001101001010001000111111100011 24987

Difference in Keys 000000000000000000000000000000000000000000000000000000101000000000000000000000000000000000(191

Alice Final Key (Symmetric Key): 100111001101000111110101100011100100111011100101110011111010101001010110110011101100110 24743

Bob Final Key (Symmetric Key): 100111001101000111110101100011100100111011100101110011111010101001010110110011101100110 24743

XOR

Difference in Final Keys : 0000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000 0

Figure (6) – shows the error correction simulator, the error ratio between the two key is 7.7%, the Sifted Key length is 29710 bits and the Number of photons being sent is 60000 photons.

Figure(7) & Figure(8) show the final key length for the different channel types. Figure(9) shows the GUI of BB84 simulator under IR attack, while Figure(10) shows the relationship between the safety parameter and the QBER. It is clear that the length of the safety parameter will be increased if the QBER increases and vice versa. Figure(11-A) shows the relationship between intruder's information and pulse intensity under PNS attack. It can be concluded that as the pulse intensity increases, the intruder information increases this is related to the fact that the transmitted pulses will contain more than one photon. Figure(11-B) shows a relationship between QBER and intruder information under PNS attack. it can be easily noticed that the Intruder information doesn’t change even if QBER changes (see eq(5)). Figure(11-C) shows a relationship between the Raw key length and the intruder information under PNS attack, againit is possible to conclude that the intruder information will be increased if the raw key length increases too. While Figure(12-A) shows the Max. and Min. QBER obtained under Intercept & Resend attack. Figure(12-B) shows the relationship between the Pulse intensity and the Intruder information about the raw key. It is obvious that the intruder information doesn’t change when the pulse intensity changes this is related to the intruder which deals with all types of pulses as a single entity. Figure(12-C) shows the relationship between QBER and Intruder information, one's can notice that the Intruder information increases when the QBER increases.

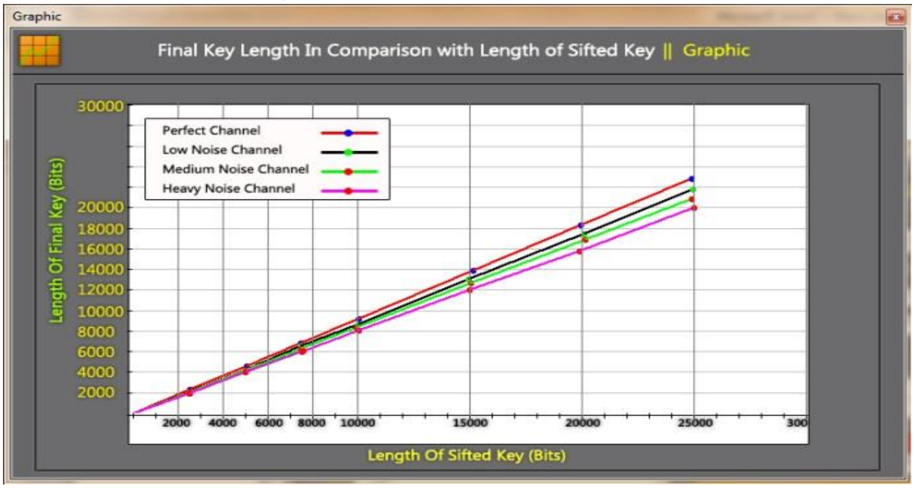


Figure (7) – shows the final key length as a function of the length of the sifted key and for the four channel types.

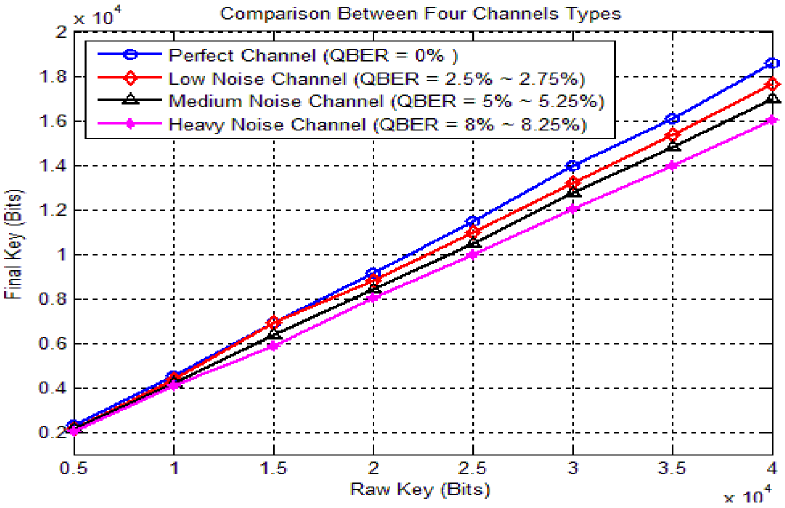
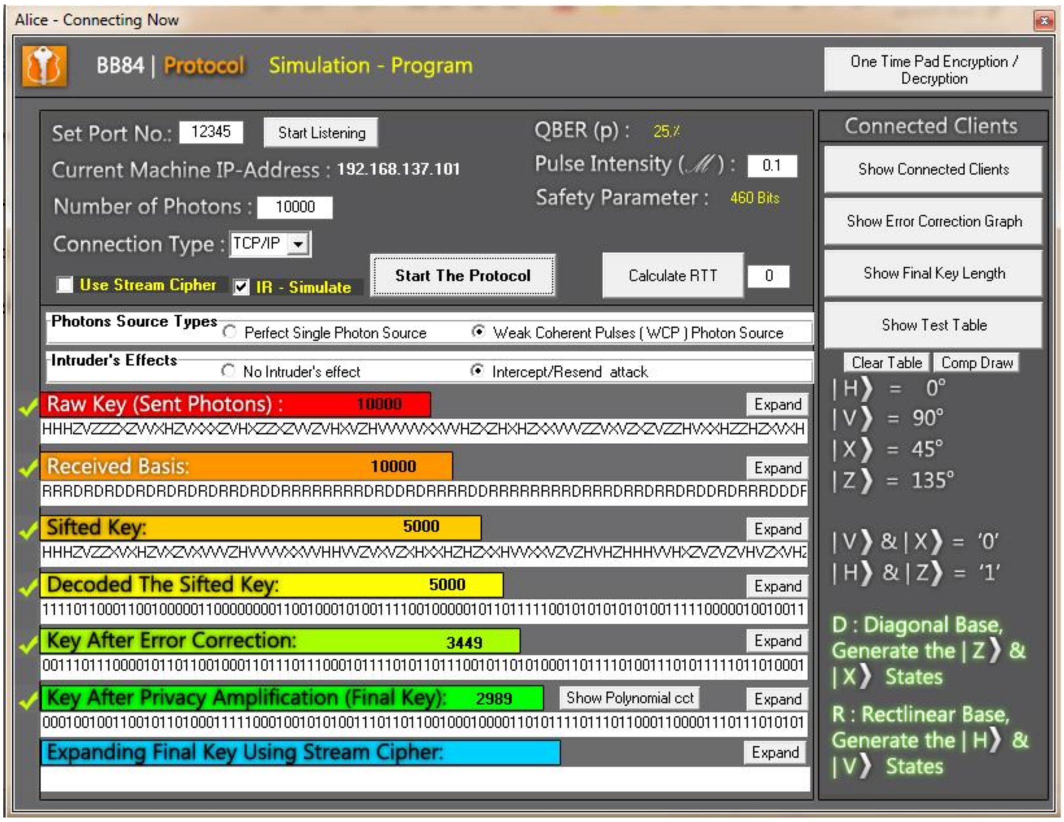


Figure (8) – shows the final key length as a function of the length of the Raw key and for the four channel types.



Alice - Connecting Now

BB84 | Protocol Simulation - Program

One Time Pad Encryption / Decryption

Set Port No.: 12345 Start Listening

Current Machine IP-Address : 192.168.137.101

Number of Photons : 10000

Connection Type : TCP/IP

☐ Use Stream Cipher ☑ IR - Simulate

Start The Protocol

QBER (p) : 25 %

Pulse Intensity ($\mathcal{M}$) : 0.1

Safety Parameter : 460 Bits

Calculate RTT 0

Photons Source Types ○ Perfect Single Photon Source ● Weak Coherent Pulses (WCP) Photon Source

Intruder's Effects ○ No Intruder's effect ● Intercept/Resend attack

✓ Raw Key (Sent Photons) : 10000 Expand

HHHZVZZZXZVVXHZVXXXZVHXZZXZVVZVHXVZHVVVVVXXVVHZXZHXHZXXVVVZZVXVZXZVZZHVXXHZZHZXVXH

✓ Received Basis: 10000 Expand

RRRDRDRDDRDRDRDRDRRDRDDRRRRRRRRDRDDRDRRRRDDRRRRRRRRDRRRDRRDRRDRDDRDRRRDDDF

✓ Sifted Key: 5000 Expand

HHHZVZZXVXHZVXZVXVVZHVVVVXXVVHHVVZVXVZXHXXHZHZXXHVVXXVZVZHVHZHHHVVHXZVZVZVHVZXVHZ

✓ Decoded The Sifted Key: 5000 Expand

111101100011001000001100000000011001000101001111001000001011011111001010101010100111110000010010011

✓ Key After Error Correction: 3449 Expand

001110111100001011011001000110111011100010111101011011100101101010001101111010011101011111011010001

✓ Key After Privacy Amplification (Final Key): 2989 Show Polynomial cct Expand

000100100110010110100011111000100101010011101101100100010000110101111011101100011000011101111010101

Expanding Final Key Using Stream Cipher: Expand

Connected Clients

Show Connected Clients

Show Error Correction Graph

Show Final Key Length

Show Test Table

Clear Table Comp Draw

$|H\rangle$ = 0°

$|V\rangle$ = 90°

$|X\rangle$ = 45°

$|Z\rangle$ = 135°

$|V\rangle$ & $|X\rangle$ = '0'

$|H\rangle$ & $|Z\rangle$ = '1'

D : Diagonal Base, Generate the $|Z\rangle$ & $|X\rangle$ States

R : Rectlinear Base, Generate the $|H\rangle$ & $|V\rangle$ States

Figure (9) – shows the GUI of BB84 Simulator under Intercept & Resend attack (Raw Key length = 10000 Bits, and the QBER = 25%). It is clear that the length of the key becomes 5000 bits after the sifting stage.

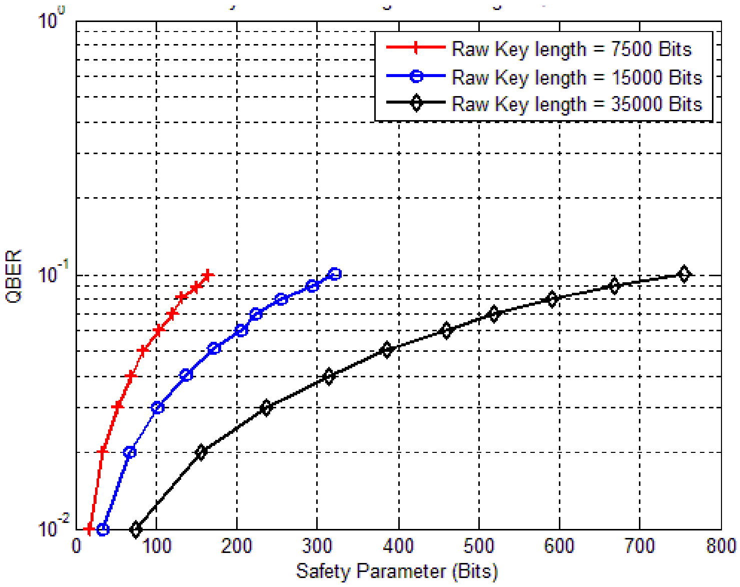


Figure (10) – shows the relationship betweensafety parameter and QBER for different raw key lengths.

PNS - Attack

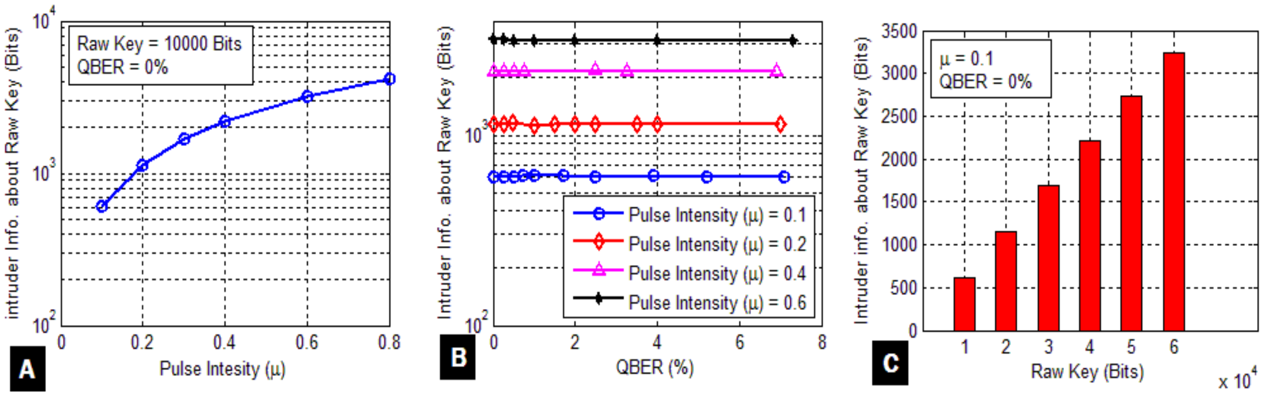


Figure (11) – A- shows the relationship between the Pulse intensity and the Intruder information about the raw key. B- shows the relationship between QBER and Intruder information. C- shows the relationship between the Raw key length and the intruder information.

IR Attack

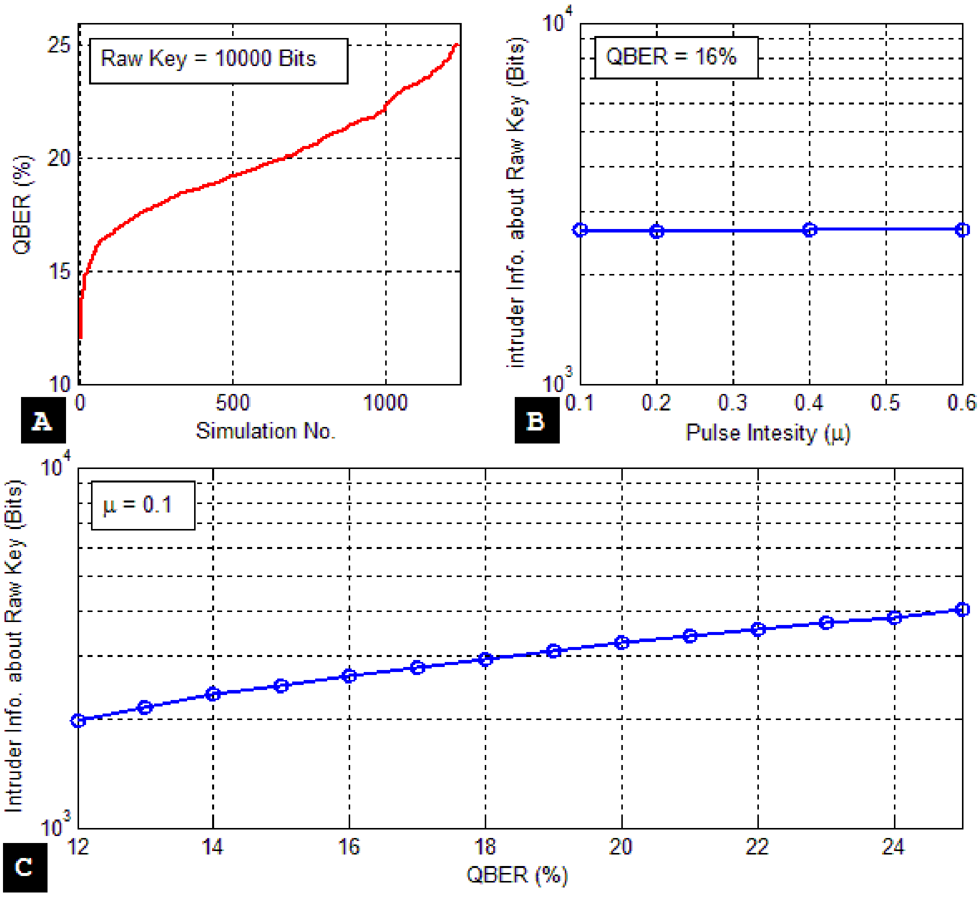


Figure (12) – A- shows the relationship between QBER and the number of simulation repetition. B- shows the relationship between the Pulse intensity and the Intruder information about the raw key. C- shows the relationship between QBER and Intruder information.

Figure (13-A) shows the relationship between Raw key length and the intruder information and it is obvious that if the raw key length increases then the intruder information will increase too. Figure(13-B) shows the final key length as a function of the Raw key length. Finally Figure(13-C) shows the relationship between the QBER and the final key length, it can be noticed that the final key length will decrease when the QBER increases. Figure (14) –Shows the relationship between the QBER and Intruder information for different values of Pulse intensity, again the Intruder information will increase when the QBER or pulse intensity increases too.

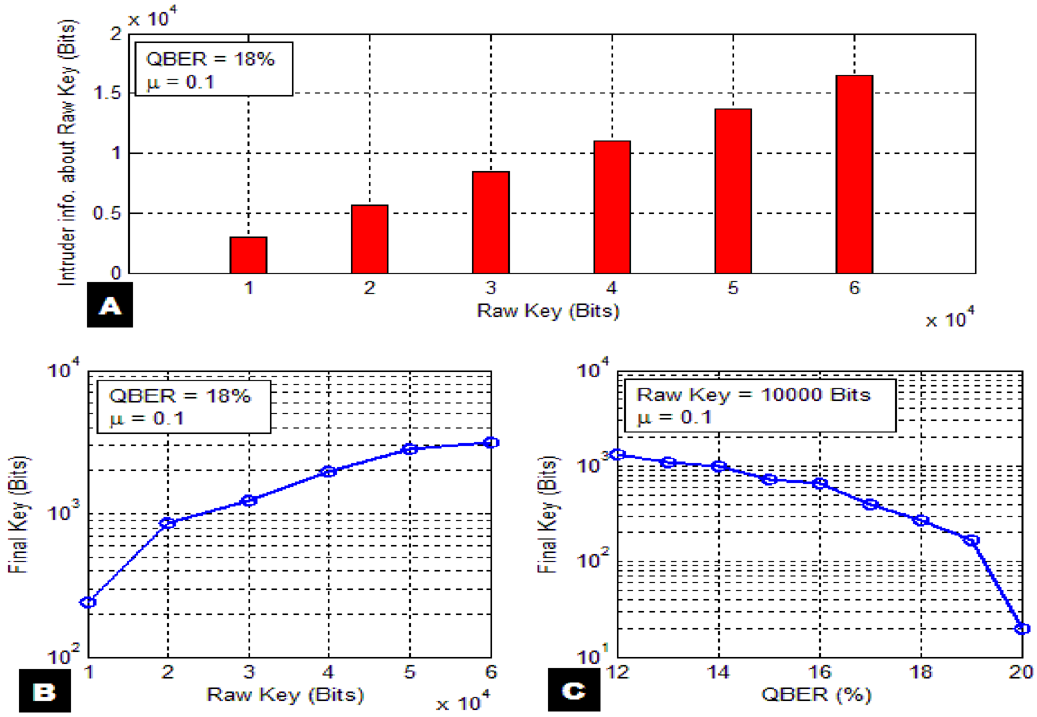


Figure (13) – A- shows the relationship between Raw key length and the intruder information. B- Shows the final key length as a function of the Raw key length. C- Shows the relationship between the QBER and the final key length.

PNS &IR Attack

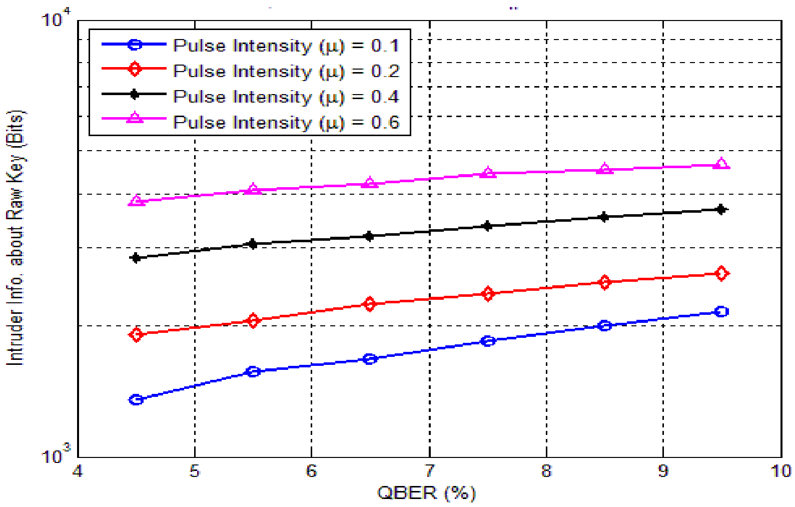


Figure (14) –Shows the relationship between the QBER and Intruder information for different values of Pulse intensity.

Figure(15) show the effect of PNS & IR attack on the BB84 protocol. Finally Figure (16) – shows the final key length as a function of the Raw key length under different attack strategies. The result shows that the IR attack has the strongest effect on the final key length, followed by the PNS & IR attack, the PNS attack has the minimum effect on the final key length relative to the other attack strategies.

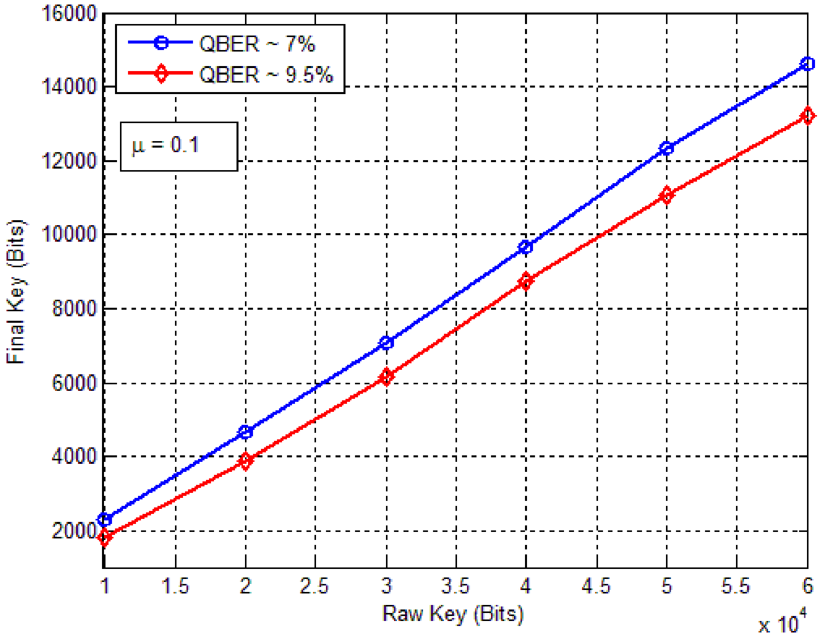


Figure (15) –shows the final key length as a function of the Raw key length for two different values of QBER.

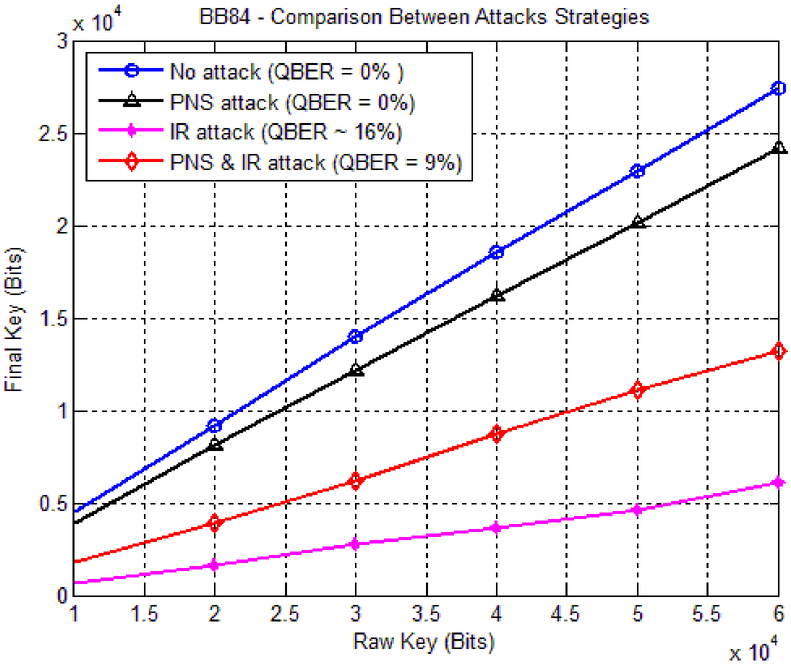


Figure (16) – shows the final key length as a function of the Raw key length for the different attack strategies.

7. Conclusion

This paper shows that the application of Multi-Parity will enhance the performance of BB84 protocol as far as the error correction is concerned, (instead of the single parity which is used by the original error correction scheme). It is found that the key length at the end of the protocol was less than the sent photons (Raw Key), in other word the usable key length reduced if there is an eaves dropping or a noisy channel. In this paper, the channel impairments took into account are (perfect, low noise, medium noise and high noise channels), and the channel is assumed to be lossless, while the detector is considered to be perfect (without loss). It is worth to mention that each channel is simulated separately and a comparison between the four types is performed, with perfect channel, the key should be symmetric after the key sifting stage with a reduction in length due to the estimation of the QBER. Also in this paper intruder's effect is taken into consideration and three types of attack are applied. It is found that the strongest attack which affect the final key is the IR attack followed by the mixed attack and finally the PNS attack. The study shows that the serious attack is PNS because it didn’t cause any error to be discovered by the sender and receiver. If some of the pulses have no more than one photon then intruders will be unable to apply this attack especially if they use perfect single photon sources. It is easy to conclude that the errors resulted from applying combined attack strategies will still be in the allowable error range<$\text{error}_{\text{max}}$.

References

[1] R. L. Rivest, A. Shamir, and L. Adleman, “A method for obtaining digital signatures and public-key cryptosystems”, Comm. Of the ACM, Vol.21, No.2, pp. 120-126, Feb. 1978.

[2] P. W. Shor, “Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer”, SIAM Jour. on Computing, Vol.26, No.5, pp. 1484-1509, 1997.

[3] C. H. Bennett and G. Brassard, “Quantum cryptography: public key distribution and coin tossing”, Proc. of IEEE Int. Conf. on Comp., Syst. and Sig. Proc., pp. 175-179, Bangalore, India, Dec. 10-12, 1984.

[4] D. Mayers, “Unconditional security in quantum cryptography”, Jour. of the ACM, Vol.48, No.3, pp. 351-406, May 2001, arXiv e-Print quant-ph/9802025; preliminary version in D. Mayers, “Quantum key distribution and string oblivious transfer in noisy channels” Adv. in Cryptology - Proc. of Crypto, pp. 343-357, 1996.

[5] P. W. Shor and J. Preskill, “Simple proof of security of the BB84 quantum key distribution protocol”, Phys. Rev. Lett., Vol.85, No.2, pp. 441-444, July 2000.

[6]G. V. Assche, “Quantum Cryptography And Secret-Key Distillation”, Lect. Notes Phys. 797 (Cambridge University Press), 2006.

[7] S. Rass and C. Kollmitzer, “Improved Error Correction in Quantum Key Distribution Protocols”,Institute of Applied Informatics, System Security Group, Klagenfurt University, University ätsstrasse 65-67, 9020

[8] A. Mink, S. Frankel and R. Perlner, "Quantum Key Distribution (QKD) and Commodity Security Protocols: Introduction and Integration", International Journal of Network Security & Its Applications (IJNSA), Vol.1, No.2, 2009.

[9] C.H.Bennett, “Quantum cryptography using any two nonorthogonal states”, Phys. Rev. Lett. **68**(21), 3121–3124. DOI 10.1103/PhysRevLett.68.3121 23, 1992.

[10] D. Bruss, “Optimal eavesdropping in quantum cryptography with six states”, Phys. Rev. Lett **81**(14), 3018–3021, 1998.

[11] C. Kollmitzer and M. Pivk (Eds.), "Applied Quantum Cryptography", Lect. Notes Phys. 797 (Springer, Berlin Heidelberg), DOI 10.1007/978-3-642-04831-9, 2010.

[12] G. Brassard, N. Lutkenhaus, T. Mor, B. Sanders, “Limitations on Practical Quantum Cryptography”, Phys. Rev. Lett. 85(6), pp.1330-1333, Aug 2000.

[13] E. Diamanti, "Security and implementation of differential phase shift quantum key distribution systems", Ph.D. thesis, Stanford University, 2006.

[14] N. Rafiei, "Quantum Communication Networks", M.Sc. Thesis, Stockholm University, May 2008.

[15] L. I. A. Ghazali, A. F. Abas, W. A. Adnan, M. Mokhtar, M. A. Mahdi, M. I. Saripan, "Security Proof of Improved-SARG04 Protocol Using the Same Four Qubit States", International Conference On Photonics 2010, IEEE DOI: 10.1109/ICP.2010.5604403, 978-1-4244-7187-4/10/$26.00 © 2010 IEEE, pp.1-4, 2010.

[16] S. Faraj, "Error Elimination and Privacy Amplification in Quantum Cryptosystems", Ph.D. thesis, Nahrain University, September 1999.

Appendix (A)

Simulation Software of BB84 Protocol Work as Follows:

Flowchart (A.1) - Initialization a TCP connection

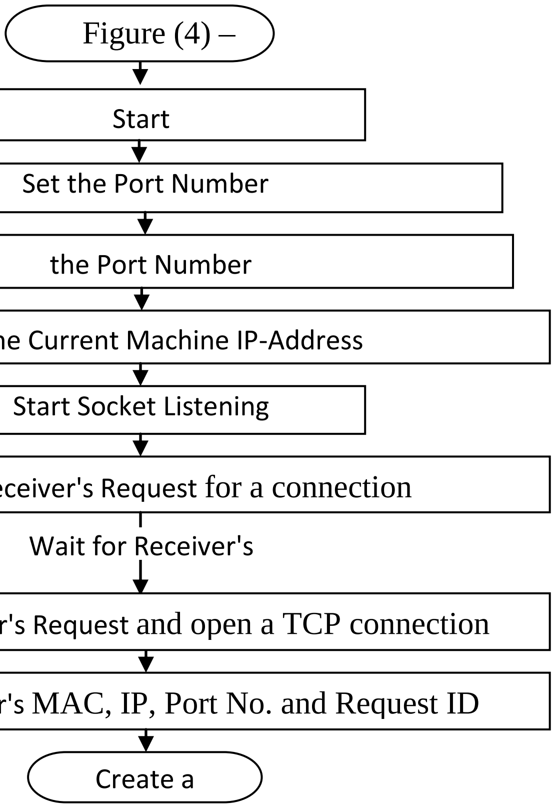
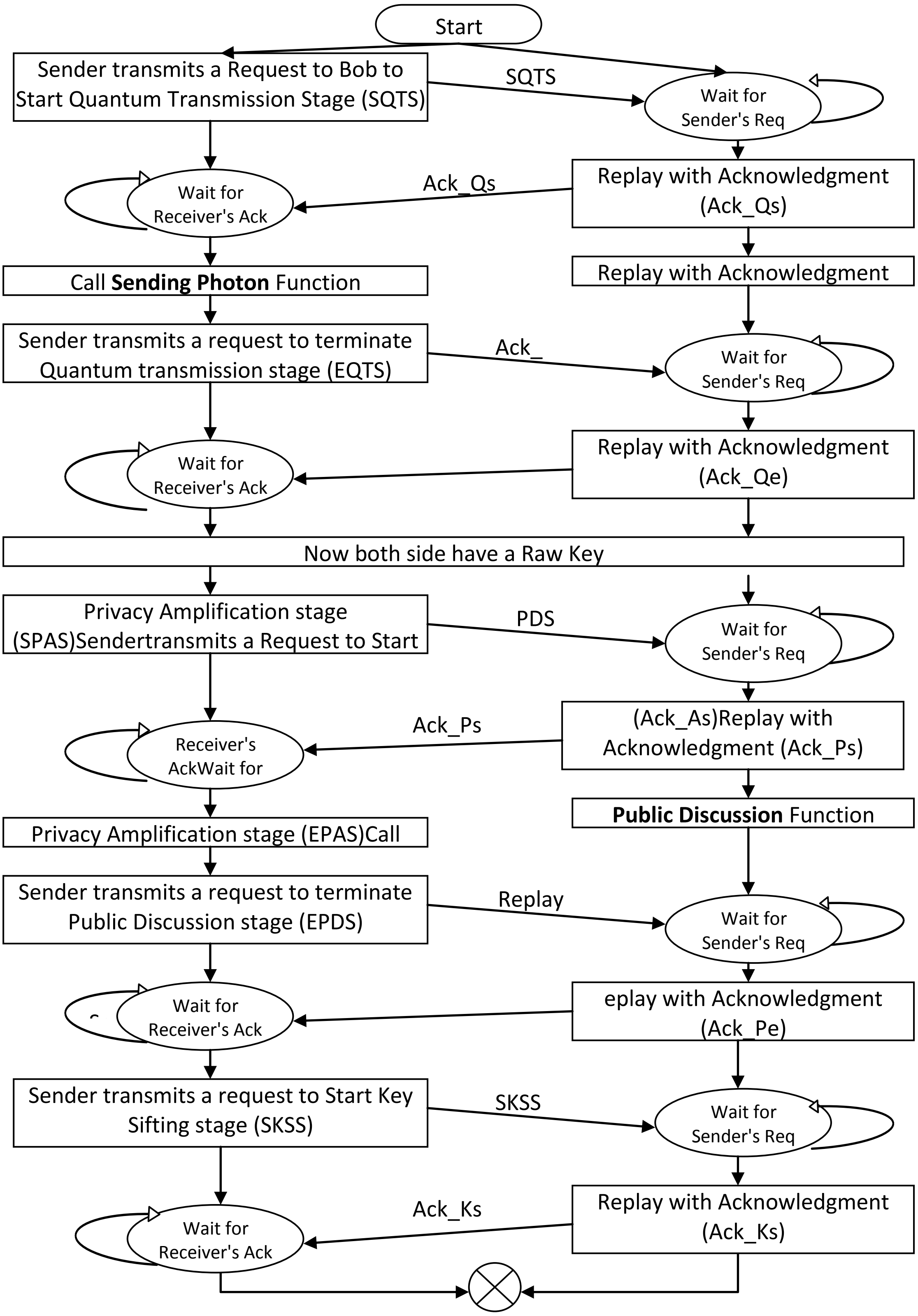
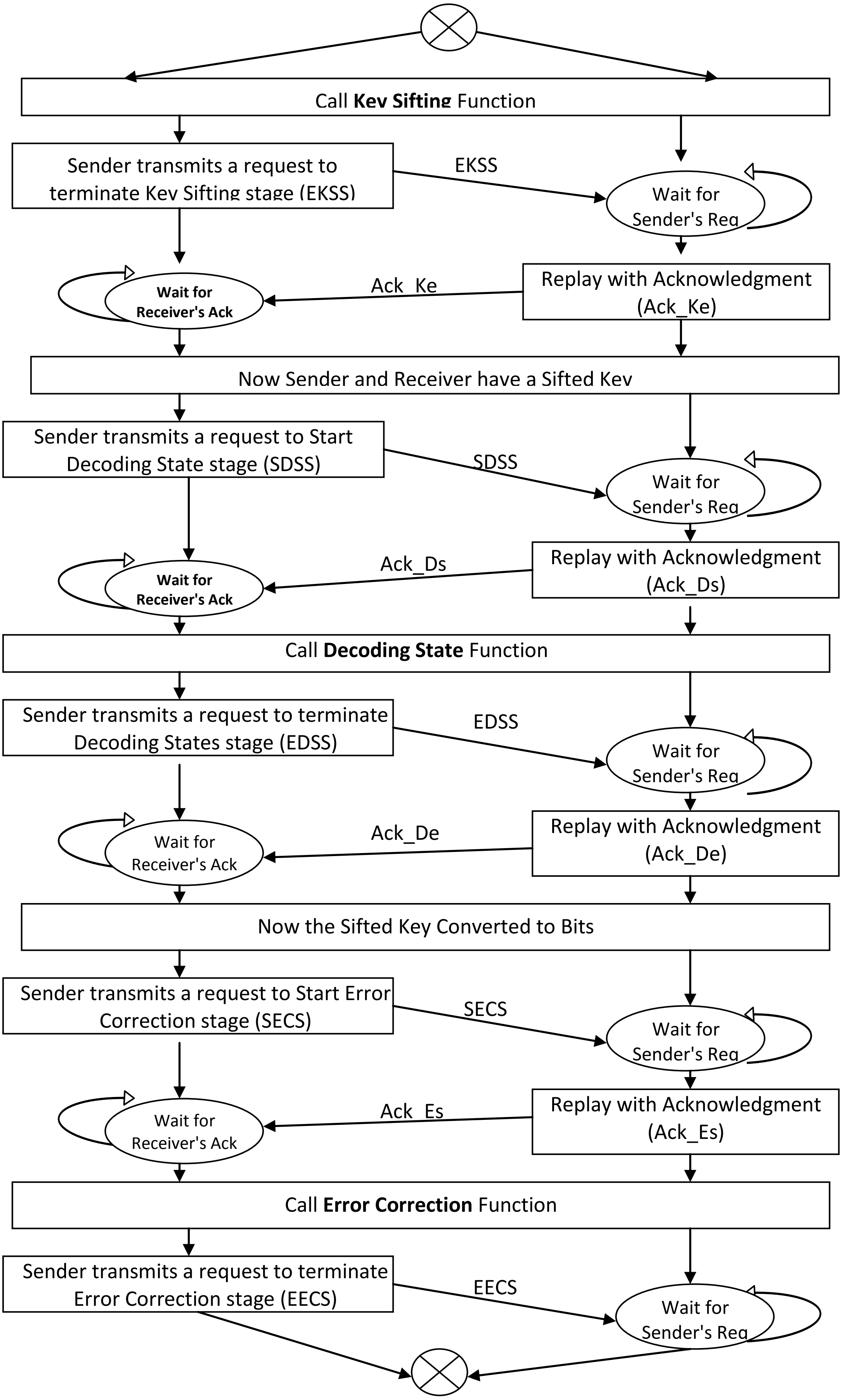
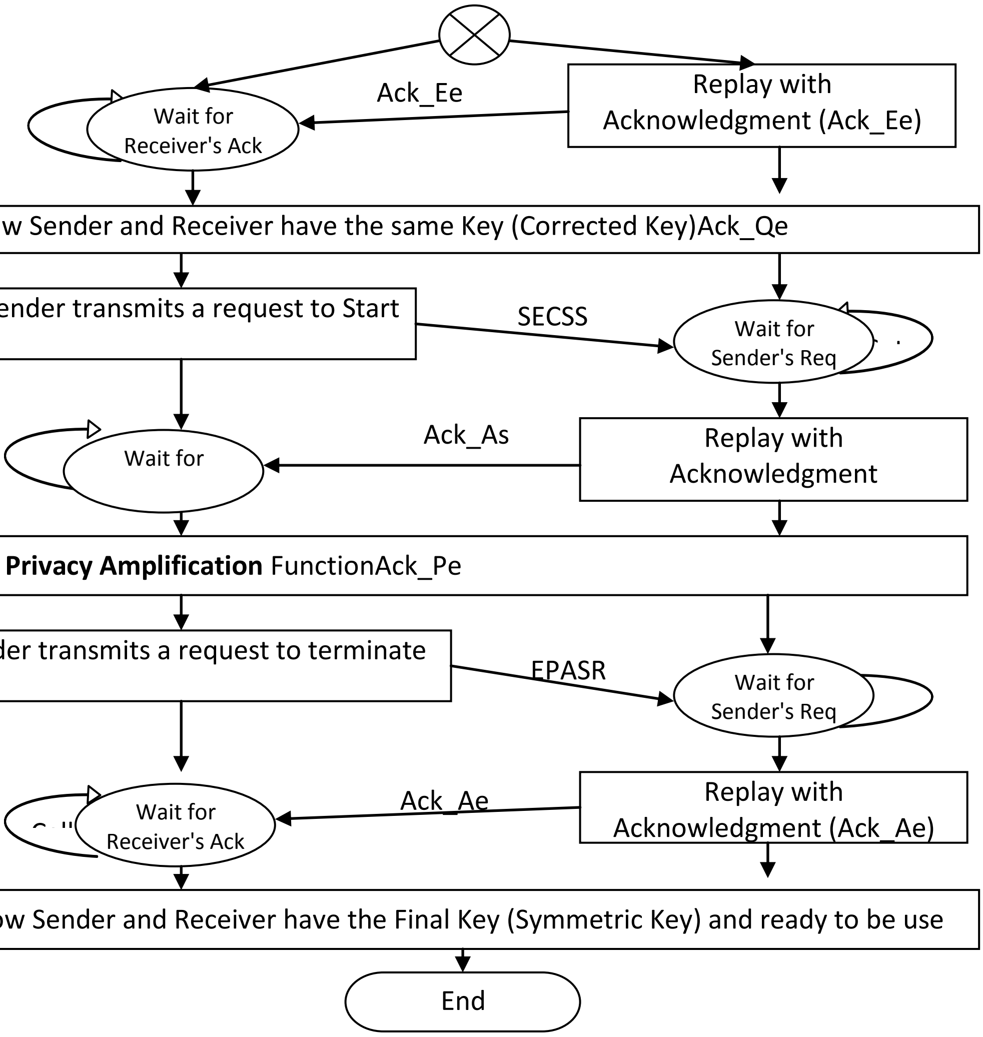


Figure (4) –
Start
Set the Port Number
the Port Number
Load the Current Machine IP-Address
Start Socket Listening
Wait for Receiver's Request for a connection
Wait for Receiver's
Agree on Receiver's Request and open a TCP connection
Register Receiver's MAC, IP, Port No. and Request ID
Create a

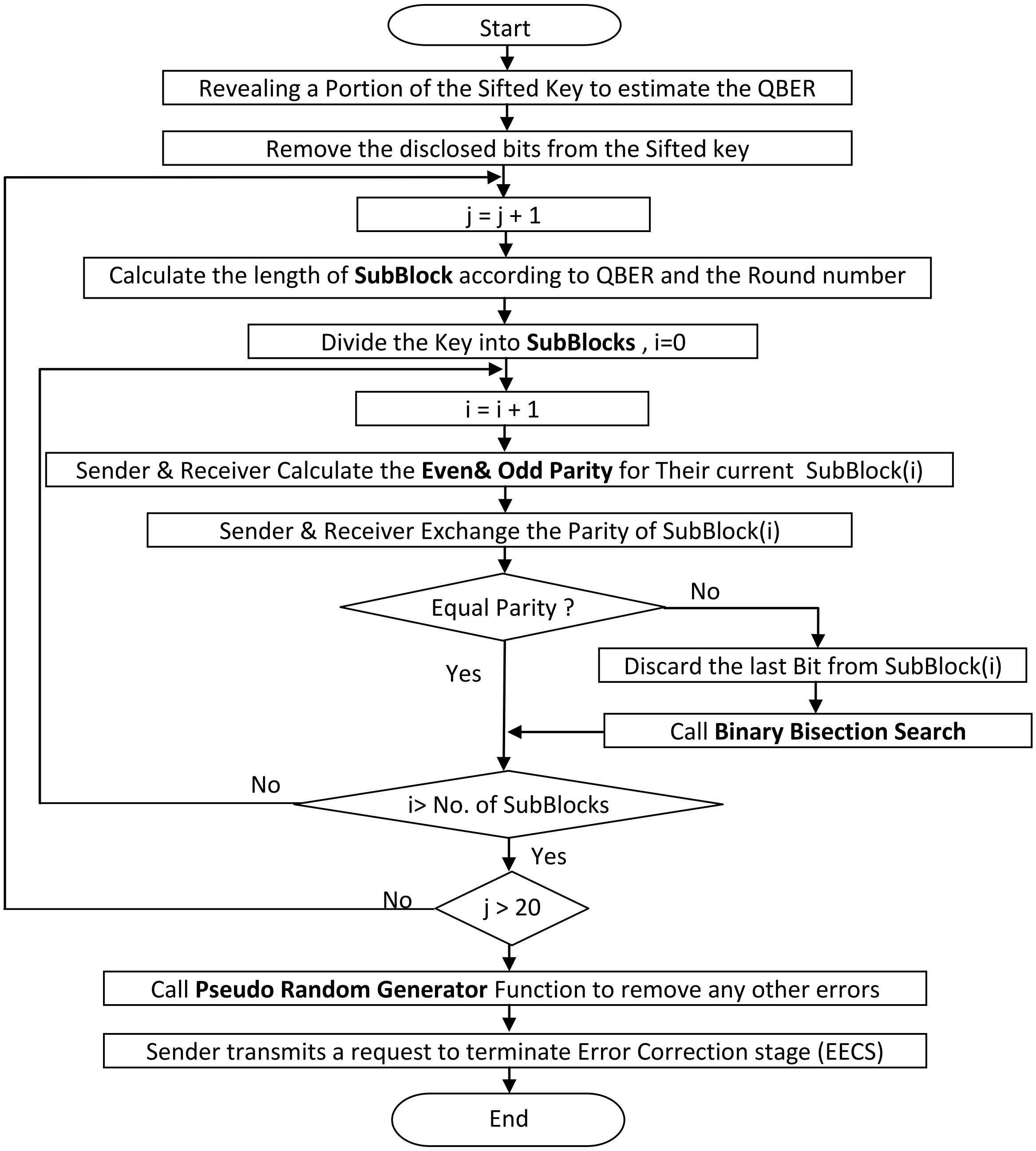
Flowchart (A.2) - The Protocol BB84


Start
Sender transmits a Request to Bob to Start Quantum Transmission Stage (SQTS)
SQTS
Wait for Sender's Req
Replay with Acknowledgment (Ack_Qs)
Ack_Qs
Wait for Receiver's Ack
Call Sending Photon Function
Replay with Acknowledgment
Sender transmits a request to terminate Quantum transmission stage (EQTS)
Ack_
Wait for Sender's Req
Replay with Acknowledgment (Ack_Qe)
Wait for Receiver's Ack
Now both side have a Raw Key
Privacy Amplification stage (SPAS)Sendertransmits a Request to Start
PDS
Wait for Sender's Req
(Ack_As)Replay with Acknowledgment (Ack_Ps)
Ack_Ps
Receiver's AckWait for
Public Discussion Function
Privacy Amplification stage (EPAS)Call
Sender transmits a request to terminate Public Discussion stage (EPDS)
Replay
Wait for Sender's Req
eplay with Acknowledgment (Ack_Pe)
Wait for Receiver's Ack
Sender transmits a request to Start Key Sifting stage (SKSS)
SKSS
Wait for Sender's Req
Replay with Acknowledgment (Ack_Ks)
Ack_Ks
Wait for Receiver's Ack


Call Key Sifting Function
Sender transmits a request to terminate Key Sifting stage (EKSS)
EKSS
Wait for Sender's Req
Wait for Receiver's Ack
Ack_Ke
Replay with Acknowledgment (Ack_Ke)
Now Sender and Receiver have a Sifted Key
Sender transmits a request to Start Decoding State stage (SDSS)
SDSS
Wait for Sender's Req
Wait for Receiver's Ack
Ack_Ds
Replay with Acknowledgment (Ack_Ds)
Call Decoding State Function
Sender transmits a request to terminate Decoding States stage (EDSS)
EDSS
Wait for Sender's Req
Wait for Receiver's Ack
Ack_De
Replay with Acknowledgment (Ack_De)
Now the Sifted Key Converted to Bits
Sender transmits a request to Start Error Correction stage (SECS)
SECS
Wait for Sender's Req
Wait for Receiver's Ack
Ack_Es
Replay with Acknowledgment (Ack_Es)
Call Error Correction Function
Sender transmits a request to terminate Error Correction stage (EECS)
EECS
Wait for Sender's Req


Ack_Ee
Replay with Acknowledgment (Ack_Ee)
Wait for Receiver's Ack
Now Sender and Receiver have the same Key (Corrected Key)Ack_Qe
Sender transmits a request to Start
SECSS
Wait for Sender's Req
Ack_As
Replay with Acknowledgment
Wait for
Call **Privacy Amplification** FunctionAck_Pe
Sender transmits a request to terminate
EPASR
Wait for Sender's Req
Ack_Ae
Replay with Acknowledgment (Ack_Ae)
Wait for Receiver's Ack
Now Sender and Receiver have the Final Key (Symmetric Key) and ready to be use
End

Flowchart (A.3) – The Error Correction Stage


Start
Revealing a Portion of the Sifted Key to estimate the QBER
Remove the disclosed bits from the Sifted key
j = j + 1
Calculate the length of **SubBlock** according to QBER and the Round number
Divide the Key into **SubBlocks** , i=0
i = i + 1
Sender & Receiver Calculate the **Even& Odd Parity** for Their current SubBlock(i)
Sender & Receiver Exchange the Parity of SubBlock(i)
Equal Parity ?
No
Discard the last Bit from SubBlock(i)
Yes
Call **Binary Bisection Search**
No
i> No. of SubBlocks
Yes
No
j > 20
Call **Pseudo Random Generator** Function to remove any other errors
Sender transmits a request to terminate Error Correction stage (EECS)
End

The work was carried out at the college of Engineering. University of Mosul