Laser QKD BB84 Protocol

Pratham Gujar

May 2025

1 Abstract

This report presents a classical optical implementation of the Quantum Key Distribution (QKD) BB84 protocol using laser modules, polarizing optical components, and Arduino-based control logic. The BB84 protocol, developed by Bennett and Brassard, enables two parties to securely exchange encryption keys while detecting potential eavesdroppers by utilizing principles from quantum mechanics. In this setup, laser beams are polarized with half-wave plates (HWPs) and polarizers to simulate quantum bits (qubits), and beamsplitter cubes serve as polarizing beamsplitters (PBS) for measurement. The system is calibrated to distinguish between horizontal/vertical (**H/V**) and diagonal/anti-diagonal (**D/A**) polarizations, offering a clear and hands-on illustration of the protocol. While this demonstration uses classical light rather than single photons, it still captures essential features such as basis mismatches and eavesdropper detection. The setup performed reliably under ambient lighting, although issues such as polarizer nonuniformity and stray light are noted. Suggested improvements include adding additional HWPs for easier basis switching and testing longer bit sequences to evaluate detection accuracy. This implementation provides a practical and educational method for exploring quantum cryptography without requiring quantum hardware.

2 Introduction

Quantum computing has been presented with the promise of solving several computationally heavy problems. These include complex simulations for fusion reactors, materials science, and drug development. Other problems include traffic analysis and machine learning. However, the focus here will be on cryptography. Quantum cryptography seeks to implement new methods utilizing the natural, occurring, and immutable laws of quantum mechanics [1]. Despite being a new concept, this technology has the potential to be significantly more secure than classical protocols and may theoretically be unhackable. It is also capable of breaking modern-day encryption methods such as RSA using Shor's algorithm. These advancements, however, will take time to develop due to hardware limitations. Advanced quantum algorithms require stable qubits and increased scalability. This report will focus on a simpler encryption protocol that can be demonstrated classically using lasers: the Quantum Key Distribution BB84 protocol.

The Quantum Key Distribution (QKD) is the most common type of quantum cryptography, theorized by Charles Bennet and Gilles Brassard in 1984 [1]. This protocol was designed to exchange a secure key between two parties by collaboratively constructing a shared private key. These secured keys can be used for classical symmetric key encryption methods. These methods use a secure key to both encrypt and decrypt data. Examples include Vigenére and Caesar encryption methods. These methods are much simpler and easier to break compared to asymmetric methods such as RSA encryption since the hacker only needs to obtain the secure key.

They can do this by intercepting it through a communication channel or by analyzing the encrypted key for patterns. Assuming the hacker knows how the two parties will communicate their shared private key, the hacker will pick the easier approach to intercept the key. For the convention, Alice will generate the secure key and send it to Bob. Eve will be the eavesdropper and will attempt to steal the key. Eve wants to remain undetected because she will need Alice and Bob to think that nobody stole their shared private key so that they will use their key to encrypt and decrypt sensitive data. So, Eve needs to ensure Bob receives the secure key that Alice generates. Alice's key, no matter what she chooses, will be serialized into a bit string so she can send it to Bob digitally. If Alice transmits her key classically, Eve can capture and copy each bit and then send the copied bit to Bob. Classical bits do not decohere, thus allowing for nearly perfect cloning for each bit. So, classically implementing this would be advantageous to Eve since Alice and Bob will not know if their key has been stolen until their encrypted data is leaked. However, quantum mechanics changes the tide to the advantage of Alice and Bob.

3 QKD BB84 Protocol

Drawing on the laws of quantum mechanics, Alice and Bob are now able to detect with high probability if someone is intercepting their shared secret key. The QKD protocol allows Alice to encode her secure key into photons and transmit them to Bob who then measures the photons [1]. The photons are called quantum bits or "qubits". They differ from classical bits since photons, depending on environmental interferences and measurement devices, decohere. So, if Eve measures the photons Alice sent, there is a 50% chance Eve will measure the correct bit and will transmit the correctly encoded photon to Bob. The 50% comes from how Alice prepares her photon and how Eve measures the photon. Suppose Alice can pick between two different transmission bases. Considering she is using photons, she will determine these two bases to be polarization bases: horizontal/vertical ($\mathbf{H/V}$) and diagonal/diagonal ($\mathbf{D/A}$) bases. Using the convention, $\mathbf{H}, \mathbf{D} \rightarrow 0$ and $\mathbf{V}, \mathbf{A} \rightarrow 1$, Alice will prepare her photon according to her transmission basis.

Suppose Alice's key is n bits long, for each i-th bit, she will randomly select a transmission basis for each. For simplifying notation, she decides to use $+$ for **H/V** and $\times$ for **D/A** [2]. So a sample bit string for $n = 3$ might be `001` and her transmission basis string will be $+ \times +$. Then, she will prepare her photons as **HDV** where the left-most photon corresponds to the left-most bit in the string. If Eve wants to intercept the bit, she will need to measure each photon. For simplicity, assume Alice announces she will send n bits so that both Eve and Bob know. Eve and Bob will randomly generate a measurement basis for each photon. Since there are only two possible bases, Eve and Bob each have a 50% chance of picking the same basis Alice used to encode her photon.

Using their selected bases, Eve and Bob will use a polarizing beamsplitter cube (PBS) to measure the photon. The PBS cube splits light into two different paths based on their polarizations (See Figure 1). If Eve uses a PBS in the **H/V** basis and a **H** photon enters the cube, then the photon will only pass through the **H** side of the cube (consider this to be the P-polarization in the figure below). Similarly, if the photon was in the **V** polarization, then the photon would only come through the S-polarization side of the cube. However, what if Alice prepared her photon in the **D/A** basis but Eve measured in the **H/V** basis?

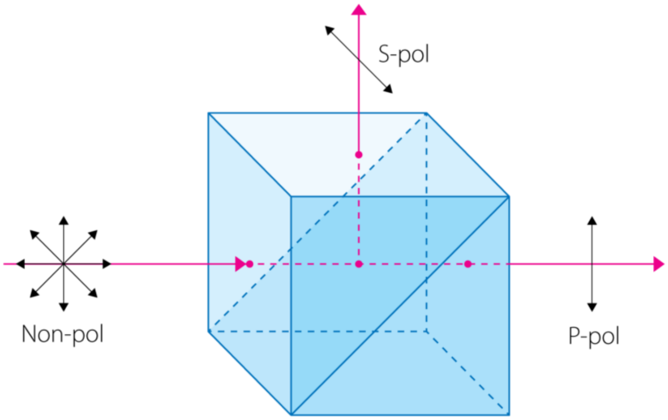


Figure 1: Diagram of PBS splitting non-polarized light into S-polarized and P-polarized light [3]. The S and P polarizations could be **H** or **V** in the **H/V** basis or vice versa in the **D/A** basis.

If Alice sent a **D** or **A** polarized photon, then Eve would see the photon come from the P or S polarization sides with 50% probability. This is because both the **D** and **A** polarizations have components of **H** and **V**. The **D** polarization is the linear combination of **H** and **V** components. So, there are equal components of **H** and **V** which causes the 50% chance of the photon coming from either side. If Alice used a polarized laser beam instead of a photon, the results would average and cause Eve to see the laser beam split and emerge from both sides simultaneously. The resulting intensities would be halved.

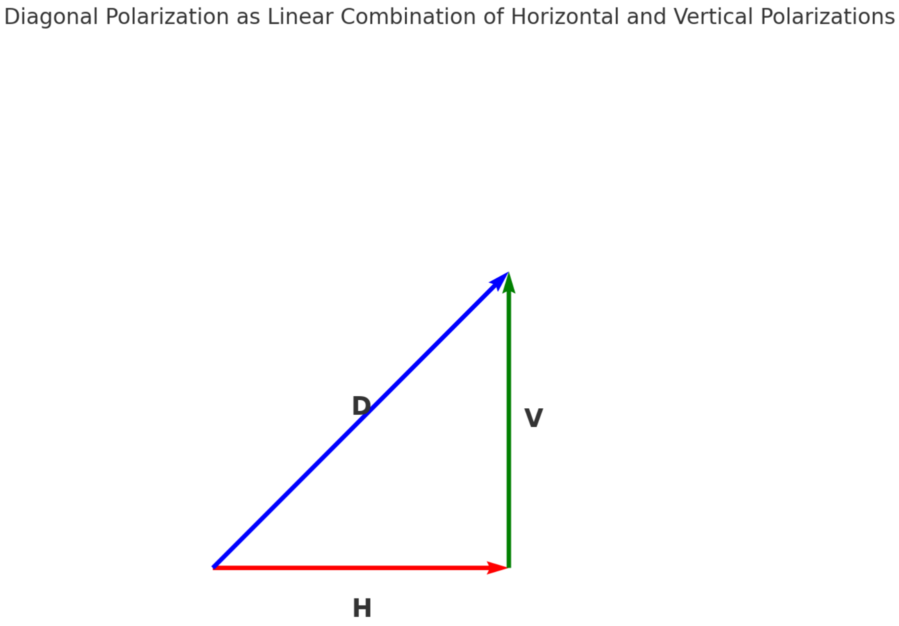
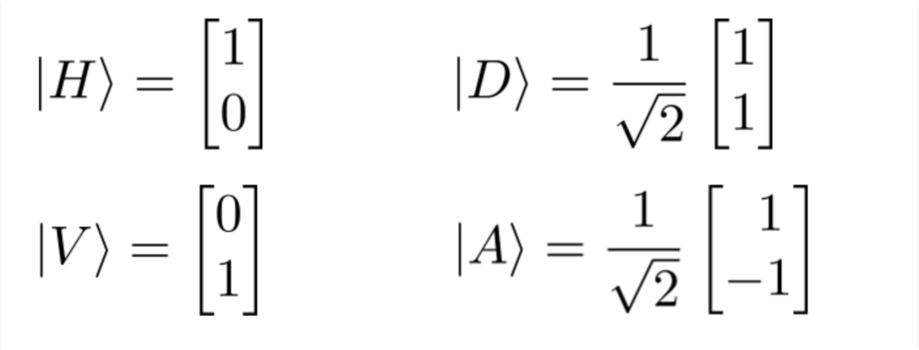

Diagonal Polarization as Linear Combination of Horizontal and Vertical Polarizations
D
V
H

Figure 2: **D** polarization represented as the linear combination of **H** and **V** polarization vectors. The same happens with **A** polarization with **V** going the opposite direction.



$$|H\rangle = \begin{bmatrix}1\\0\end{bmatrix} \qquad |D\rangle = \frac{1}{\sqrt{2}}\begin{bmatrix}1\\1\end{bmatrix}$$

$$|V\rangle = \begin{bmatrix}0\\1\end{bmatrix} \qquad |A\rangle = \frac{1}{\sqrt{2}}\begin{bmatrix}1\\-1\end{bmatrix}$$

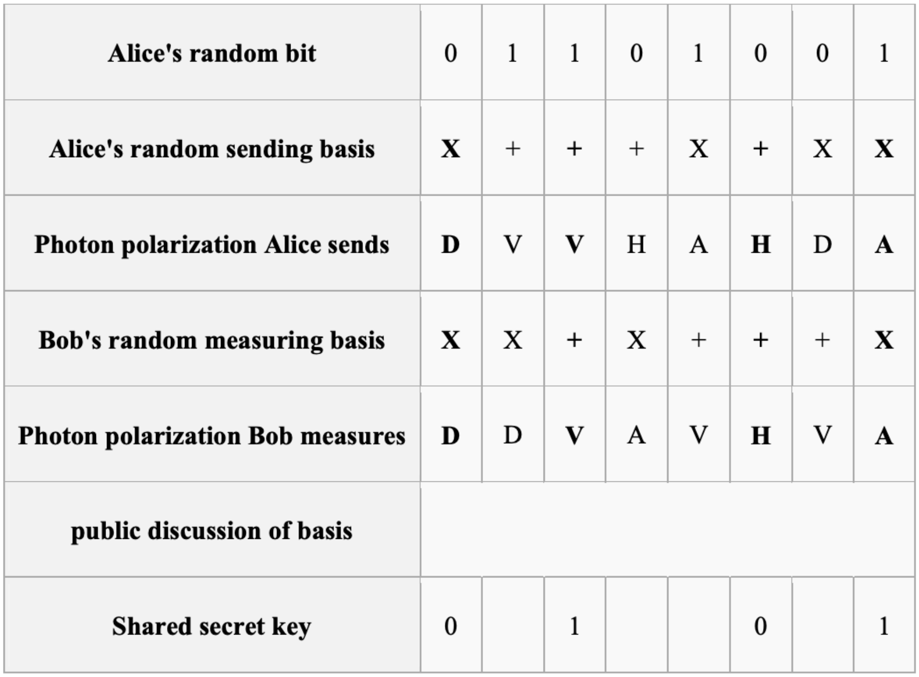
Figure 3: All four polarizations are represented as vectors (taken from the poster).

After the photon emerges, Eve will count it using a photon detector. Based on which photon detector registered a count, Eve will assume the polarization Alice prepared and then send her prepared photon to Bob. So, if Alice sent a **D** photon and Eve measured **H** in the **H/V** basis, then Eve will send a **H** photon to Bob thinking that is what Alice sent. Bob will measure with his own PBS configured to his chosen measurement basis and then record his result. So, if Alice and Bob picked the right basis for the current bit but Bob measured a different bit than what Alice transmitted, then Alice and Bob can suspect that someone, Eve, intercepted Alice's photon but measured it on the wrong basis. Thus, Eve measured the wrong bit or measured the correct bit and prepared it on the wrong basis causing the mismatch between Alice and Bob.

This is what allows Alice and Bob to detect eavesdroppers in their communication channel. Now, in the case of photons, there are chances of decoherence due to environmental factors or false counts from the dark count probabilities of their photon detectors. This arises from the quantum nature of the photons and can still cause mismatches between Alice and Bob if they selected the same basis and there was no eavesdropper. So, Alice and Bob establish a Quantum Bit Error Rate (QBER) that accounts for this and serves as the threshold as to when a channel is no longer secure. This decoherence also fully prevents Eve from perfectly cloning Alice's photon.

After Alice transmits all n bits to Bob, Alice and Bob will sample some number of their bits [2]. Consider they sample the first 10% of their bits. They will publicly announce what bases they selected for each of those sampled bits. Alice and Bob will then sift their sampled bases to find which bases matched for which bits. They will denote the discarded bases with a $-$. They will then reveal the associated bits they have for the matching bases. Alice and Bob compare their bits and count the number of discrepancies. They use the discrepancies to calculate the QBER and see if it exceeds their threshold. If it does, then they abort communication on this channel and find a different way. This highlights the entire QKD BB84 protocol. There are other QKD protocols such as B92, E91, and COW but these will not be discussed.

The implemented QKD BB84 protocol discussed in this report utilizes lasers instead of photons. So, the quantum behavior of the photon decoherence vanishes due to the averaging of the photons in the laser beam. This demonstration aims to show the essence of the BB84 protocol without quantum fluctuations in the measurements. With this implementation, one can clearly and visually see how Alice transmits her bits to Bob and how Eve attempts to steal each bit and go unnoticed.



Alice's random bit	0	1	1	0	1	0	0	1
Alice's random sending basis	**X**	+	**+**	+	X	**+**	X	**X**
Photon polarization Alice sends	**D**	V	**V**	H	A	**H**	D	**A**
Bob's random measuring basis	**X**	X	**+**	X	+	**+**	+	**X**
Photon polarization Bob measures	**D**	D	**V**	A	V	**H**	V	**A**
public discussion of basis								
Shared secret key	0		1			0		1

Figure 4: Example of a sampled bit string and bases [2]. If their QBER is under their threshold, then Alice and Bob will use their generated shared secret key for symmetric key encryption protocols.

4 Materials

- 2 KY-008 laser modules (630 nm)
- 4 Laser receiver sensor modules
- 2 25 mm Beamsplitter cubes
- Polarizers
- 2 25 mm Half-wave plates (HWP) [4]
- Arduino Uno
- Arduino Uno Power Supply Module
- 1 1' by 2' by 1.5" wooden board
- 4 2" by 2" by 0.5" wooden blocks

5 Methods

Using the materials above, I implemented the BB84 protocol split into two phases (See figures below).

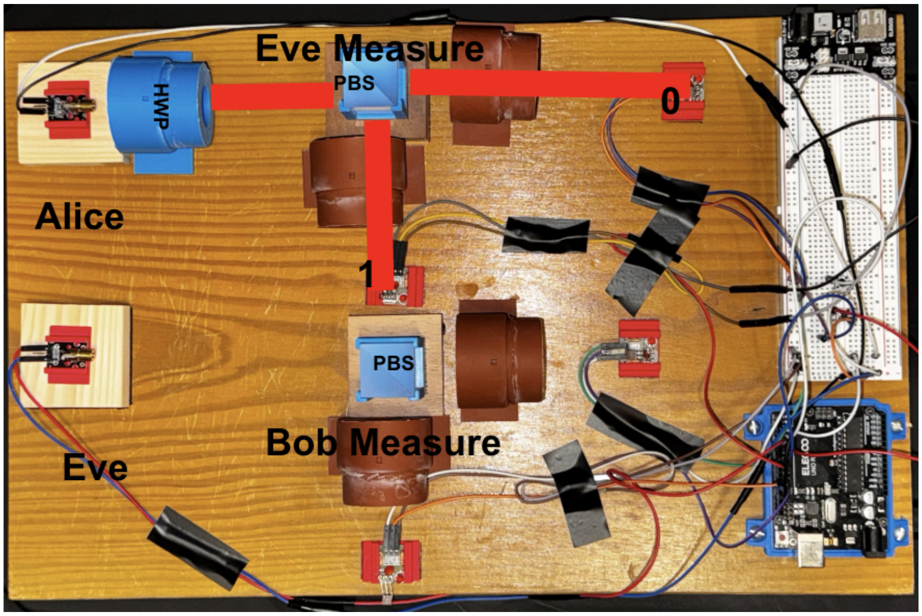


Figure 5: Phase 1 consists of Alice transmitting her bit to Bob and Eve intercepts this bit and measures it.

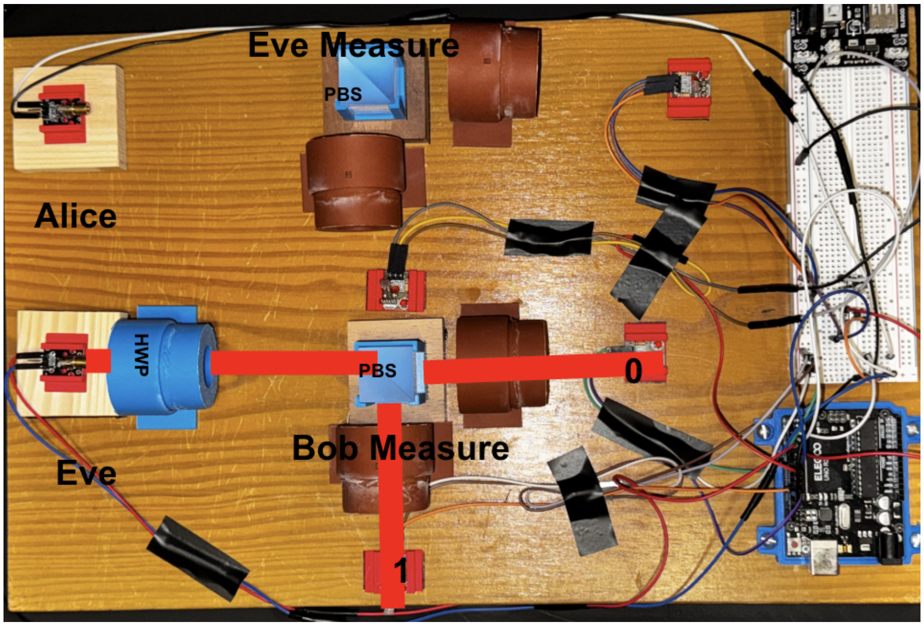


Figure 6: Phase 2 consists of Eve using her measured bit and chosen basis to assume what Alice sent. She then prepares the bit using her measured basis and transmits it to Bob. Bob measures the bit and records his measurement. The cycle continues until Alice has sent all of her bits.

In the figures above, the wooden blocks mentioned in Section 4, were used to mount the lasers and beamsplitters (labeled as PBS in the images). The lasers were placed into 3D printed mounts as shown in the image with the red mount base around the laser diode. The blue and brown rings (the blue one is labeled as HWP) are custom 3D-printed optical holders (my CAD files will be available on my GitHub [5]). Inside the blue optical holder is the HWP mentioned in Section 4. Inside the brown holders are cutouts of a non-adhesive polarizing film capable of filtering out **H**, **V**, **D**, **A** polarizations. The beamsplitters are inserted into another custom 3D-printed mount to prevent any scratches. The beamsplitters and polarizers combined together work as a makeshift PBS.

As for the setup and usage, the setup is quite challenging as the demonstration is sensitive to ambient light and requires much precision. The board should be arranged as it is in the images above for optimal usage. The laser must pass through the beamsplitter and the beam(s) must pass through the polarizers (brown optical holders) and must land on the detectors. Although counterintuitive, the laser beams mustn't land on the center of the detectors (small circular section). Several factors including the ambient light and performance of the HWP and polarizers cause the detector to read false positives because of the extremely faint laser beam landing on the center. For calibration, I adjusted the laser detector instead of the laser transmitter as it allowed for much better precision and accuracy.

Further calibration methods include using the Arduino buttons to activate the individual phases. The detector counts and measured bit should be analyzed to determine if calibration is complete. I followed the following steps for efficient calibration. Note that the optical holders have degree markings on them and are accurate to ±3°.

Step 1: Set up the KY-008 laser modules (wiring discussed later) and aligned them with the laser receiver sensor modules.

Step 2: Place HWP in front of the laser for Phase 1 (Alice's laser) and set it to 0°.

Step 3: Set the polarizers around the beamsplitter to the **H/V** basis. 0° for the side entering the `0` bit detector and 90° for the side entering the `1` bit detector (See Figure 5 for reference of the `0` and `1` bit detectors).

Step 4: Connect the buttons to the Arduino as indicated in the `QKD_Control.ino` and also upload this program to the Arduino Uno.

Step 5: When ready, press the designated button for Phase 1 to begin the phase.

Step 6: Read the output using the Arduino Serial Monitor (or Python).

Step 7: If the measured bit is `0` and `Count0` > `Count1` (See Section 5.2 on how to determine this inequality), then the measurement is a success. If not, adjust the laser detector and try again.

Step 8: Repeat all previous steps but change the HWP angles for each step. HWP at 0° keeps the laser in the **H** state. At 22.5°, the laser is in the **D** state. At 45°, the laser is in the **V** state. Lastly, at 67.5°, the laser is in the **A** state.

Step 9: Confirm the measurements and determine if they are as expected. Note that `Count0` corresponds to the number of counts measured by the `0` detector and likewise for `Count1`.

Step 10: After all HWP angles are completed, return to 0° and then change the polarizers into the **D/A** basis. Both polarizers must be at 45°.

Step 11: Repeat steps 4-8.

Step 12: After Phase 1 calibration is complete, move the HWP to Eve's laser in Phase 2 and repeat all steps but for Phase 2.

Step 13: After all calibration has been completed, upload `QKD_Control2.ino` to the Arduino Uno and begin the demonstration.

5.1 Wiring

Properly wiring the laser components, especially the laser detector, will ensure the components remain unharmed. On the KY-008 laser transmitter, the **S** terminal is the voltage source terminal, the middle terminal is the output terminal, and the **-** terminal is the **GND** terminal (Refer to the datasheet [6]). The output terminal on the transmitter module outputs a voltage if the laser is supplied by power and turns on. This can be used for laser diagnostics. The transmitter should be supplied by at most 5V. Look at either `QKD_Control.ino` or `QKD_Control2.ino` for the **S** wiring. The **S** should be connected to an Arduino Uno numbered pin. The number of the pin is mentioned in either of the two codes mentioned earlier.

The laser detector, similar to the laser transmitter, has three pins but all three pins *must* be connected (See the datasheet [7]). Unlike the laser transmitter, the source pin must be connected to a steady 5V power supply. The constant 5V supply can come from the Arduino Power Supply Module (See module on the top of the breadboard in 5). The middle output pin must be connected to the Arduino numbered pins and the specific number is found in the code. The detectors associated with Eve will be mentioned as `EVE_DETECT0` or `EVE_DETECT1` (similarly for Bob). The last remaining pin should be connected to **GND**.

For the Arduino buttons, one pin on one side of the button (consider the left side) should be connected to its designated PIN on the Arduino as stated in only `QKD_Control.ino`. The other pin on the right side should be connected to **GND**.

5.2 Count Similarity Threshold

The count similarity between `Count0` and `Count1` is determined by their ratio with the smaller quantity as the numerator. If the ratio is greater than or equal to 80%, then `Count0` ≈ `Count1`. Now, to save on computation, it's not necessary to know which count is the smaller quantity. For simplified notation, let `Count0` be a and `Count1` be b. Using the logic above,

$$\frac{\min(a,b)}{\max(a,b)} \geq 0.8 \Rightarrow a \approx b \tag{1}$$

To remove the logic of determining the minimum and maximum count, I rewrite Eq. 1 as

$$\frac{\max(a,b)-\min(a,b)}{\max(a,b)} \leq 0.2 \tag{2}$$

$$\tag{3}$$

$$\max(a,b)-\min(a,b) = |a-b| \tag{4}$$

To simplify the denominator, I use a bounding trick.

$$\max(a,b) \geq \frac{a+b}{2} \tag{5}$$

$$\tag{6}$$

$$\frac{1}{\max(a,b)} \leq \frac{2}{a+b} \tag{7}$$

$$\tag{8}$$

$$\Rightarrow \frac{|a-b|}{\max(a,b)} = \frac{2|a-b|}{a+b} \tag{9}$$

Putting this into Eq. 2, this gives the final result

$$\frac{|a-b|}{a+b} \leq 0.1 \Rightarrow a \approx b \tag{10}$$

This threshold is used in the code and should also be used in the calibration steps to determine if each phase is calibrated.

6 Essential Optics for QKD

My implementation utilized the HWP to prepare the lasers into particular polarizations. This can be confirmed mathematically by using the Jones matrix for the HWP with θ relative to the fast-axis. The fast axis is the directional axis within the HWP medium where light travels fastest. The Jones matrix is given by

$$\mathbf{HWP}(\theta) = \begin{pmatrix} \cos(2\theta) & \sin(2\theta) \\ \sin(2\theta) & -\cos(2\theta) \end{pmatrix}$$

The KY-008 laser is by default in the **H** polarization. Using the Jones matrix for the HWP, at $\theta = 45°$ the calculation will yield the **V** vector shown in Figure 3. All angles described in the calibration steps can be confirmed mathematically using this Jones matrix applied to the **H** vector.

6.1 Demonstration Operation

Once calibration is complete, make sure that the `QKD_Control2.ino` program is loaded onto the Arduino Uno. Then, follow the instructions on the GitHub [5] to start the QKD Control Center front end and back end. In the back end, the `PORT` variable in the back end should be updated to the directory listed at the bottom right of the Arduino application window when the Arduino is connected. If they do not match, the program will not work.

To run the demonstration, look at the basis in the Alice panel. Then look at the right-most bit directly above the basis. Set the HWP accordingly with this information and then look at the Eve panel. Similarly, look at the basis for the Eve panel and set Eve's polarizers accordingly. Now Phase 1 is ready to start. Activate Phase 1. After completion, use the bit Eve measured and the basis in which Eve measured to configure the HWP for Eve's laser. Then, look in Bob's panel and configure Bob's polarizers according to the basis mentioned in his panel. After this is done, start Phase 2. Repeat this until Alice has no more bits to send. After all bits have been transmitted, click on `Analyze` at the bottom right to get the results of the demonstration.

7 Results And Discussion

After constructing the demonstration, the results were better than expected even in strong ambient lighting. Unfortunately, due to the nature of the demonstration, there are no figures to demonstrate the effectiveness of the demonstration. I did notice that when setting the laser polarization for Alice to **A** and if Eve was measuring in the **H/V** basis, then Eve would measure **H** close to 100% of the time. I determined the polarizers were not uniform and had imperfections at certain angles. If it had been uniform, then the same issue would persist for Alice transmitting the **D** polarization and Eve measuring in the **H/V** basis.

Other than this, the demonstration seems to be working quite well in classroom lighting. If in the presence of sunlight or higher-intensity lights, the detectors pick up stray readings and measure incorrectly. Ultimately, this demonstration can concretely show the QKD BB84 protocol using classical optics. It demonstrates Alice and Bob's advantage in detecting any eavesdroppers on their communication channels.

8 Conclusion

The demonstration successfully shows how the QKD BB84 protocol is performed between two parties and how it may be intercepted. The key result is that this implementation shows how the communicating parties can detect eavesdroppers. Some further improvements for this implementation would be to increase the number of HWPs so that the polarizer angles need not be changed but rather the HWPs for configuring the measurement bases. Another addition would be to gather test results in large bit streams (more than 20 bits) to determine how successful Alice and Bob are in detecting Eve. This would provide more concrete results on the effectiveness of this protocol. A succeeding, but significantly more advanced, demonstration can be built using photon generators and photodetectors to compare with the classical implementation of the protocol. The comparison of the results should be interesting to see how quantum effects enhance the protocol.

9 Acknowledgments

I would like to acknowledge Dr. Aida Torabi for her mentorship and assistance in ensuring all materials required for the apparatus were delivered on time.

She also greatly assisted in offering several resources to help manufacture the apparatus components such as the optical holders and laser/detector mounts. I would also like to thank Dr. Dan for his mentorship and assistance with the PMA Student Machine Shop equipment. His experience and guidance were invaluable and allowed me to precisely machine my wooden blocks for the lasers and beamsplitters. I would also like to acknowledge the CNS MakerSpace for their 3D printing training and guidance as it allowed me to create mounts for my lasers and beamsplitters as well as my optical holders. I would finally like to thank my classmates for their encouragement and peer review as their insight helped me construct my demonstration.

References

[1] IBM Quantum, *Quantum cryptography*, Retrieved May 2, 2025, IBM, (2023) https://www.ibm.com/think/topics/quantum-cryptography.

[2] M. Chekhova, *Lecture 12: quantum key distribution*, Retrieved May 2, 2025, Max Planck Institute for the Science of Light, (n.d.) https://mpl.mpg.de/fileadmin/user_upload/Chekhova_Research_Group/Lecture_4_12.pdf.

[3] Altechna, *Diagram of polarizing cubes for high energy applications*, Image available at: https://www.altechna.com/wp-content/uploads/2018/10/diagrama-polarizing-cubes-01-1-1024x637.png. Retrieved May 2, 2025, (2018) https://www.altechna.com/products/polarizing-cubes/polarizing-cubes-for-high-energy-applications/.

[4] Edmund Optics, *Polymer retarder film*, Retrieved May 2, 2025, (n.d.) https://www.edmundoptics.com/f/polymer-retarder-film/14827/.

[5] Psonu2003, *Laser-qkd-bb84*, Accessed: 2025-05-02, (n.d.) https://github.com/Psonu2003/Laser-QKD-BB84.

[6] ArduinoModulesInfo, *Ky-008 laser transmitter module*, Accessed: 2025-05-02, (n.d.) https://arduinomodules.info/ky-008-laser-transmitter-module/.

[7] T. Hareendran, *The mysterious laser receiver sensor module!*, Accessed: 2025-05-02, (2019) https://www.codrey.com/electronic-circuits/the-mysterious-laser-receiver-sensor-module/.