Silicon avalanche photodiodes for satellite based quantum communication

by

Ian DSouza

A thesis
presented to the University of Waterloo
in fulfillment of the
thesis requirement for the degree of
Master of Science
in
Physics

Waterloo, Ontario, Canada, 2018

Author's Declaration

This thesis consists of material all of which I authored or co-authored: see Statement of Contributions included in the thesis. This is a true copy of the thesis, including any required final revisions, as accepted by my examiners.

I understand that my thesis may be made electronically available to the public.

Statement of Contributions

For chapter 2, I and Dr. Jean Philippe Bourgoin conducted the experiments and analyzed the data. Dr. Brendon Higgins provided software support. Prof. Thomas Jennewein supervised the experiment.

For chapter 3, I, Dr. Higgins and Dr. Bourgoin prepared for the experiment. Dr. Higgins provided software support. All three of us and Jin Gyu Lim, Ramy Tannous and (now Dr.) Sascha Agne conducted the experiments at the Tri-University Meson Facility (TRIUMF). The personnel at TRIUMF were also very helpful in providing technical support. I analyzed the data with support from Dr. Bourgoin and Dr. Higgins especially for their algorithm on afterpulsing using efficiency measurement data. I authored a new algorithm for analyzing the afterpulsing probability under encouragement and preliminary discussions with Prof. Jennewein. Prof. Jennewein provided helpful input during weekly group meetings, and Dr. Bourgoin and Dr. Brendon also provided helpful input on the final algorithm. Dr. Bourgoin and I generated graphs for the data and deduced results. Prof. Vadim Makarov helped with high level planning of the experiment. Prof. Jennewein supervised the experiments.

For chapter 4, Dr. Bourgoin and Dr. Higgins contributed on the overall planning and I prepared for the experiments with Dr. Bourgoin. I, Dr. Bourgoin and Jin conducted the experiments at the David Florida Laboratory (DFL). The personnel at DFL were also very helpful in providing technical support. Dr. Higgins provided software support. Paul Marchand from Neptec Design Group Ltd. provided key support for the thermal aspects of this project. Prof. Jennewein supervised the experiments.

For chapter 5, for the derivation of the theory of the afterpulsing algorithm, Prof. Jennewein and I had preliminary discussions on formulating the theory. I developed the theory. For the observed afterpulsing theory, I developed the theory and Dr. Bourgoin provided feedback on the final theory.

For chapter 6, I want to thank Prof. Michal Bajcsy for helping me brainstorm potential projects for the final project of QIC 750 (Implementation of Quantum Information Processing) and also for advice to model the photon statistics of the laser source. The

initial part of the simulation (before the QIC 750 course) was jointly done by me and Dr. Bourgoin who aided me in simulating a part of the APD parameters. I want to also thank Dr. Bourgoin for preliminary discussions regarding the overview of the quantum key distribution protocol. For the remainder of the laser source and detector modeling as well as all of the QKD protocol and attack modeling, generating the graphs and analyzing the results, I am the author. Dr. Bourgoin helped me in analyzing the recharge time graph.

Abstract

Single photons detectors are required for applications like quantum key distribution and testing Bell's inequalities. Silicon avalanche photodiodes (Si APD's) have advantages like low dark count rate. This thesis aims to experimentally characterize Si APD's under conditions experienced in low earth orbit on board a satellite and also develop supporting algorithms and simulations.

In chapter 1, I motivate the use of Si APD's as single photon detectors on a satellite. I also briefly explain the how these APD's work and the related electronics.

In chapter 2,I describe the test setup used in the laboratory at the Institute for Quantum Computing to characterize the APD's. I then explain how one goes about measuring and/or calculating the different characteristic parameters of the APD. These characteristics include breakdown voltage, output pulse characteristics, recharge time, dark counts, detection efficiency, timing jitter of APD, saturation value and afterpulsing probability. The aim of the lab characterization is to form a baseline measurement under nominal conditions before we introduce the APD to conditions of low Earth orbit.

In chapter 3, the APD's are exposed to proton radiation which is expected in low Earth orbit. Such radiation causes displacement defects in the APD's substrate and increases dark counts. Previous work [1] has shown that thermal annealing (increasing the APD temperature for a period of time) is successful in decreasing the dark count rate to operational range. It is anticipated that multiple annealing instance would be needed in operation under constant bombardment of protons. The work in this thesis explores possibility of using annealing repeatedly by alternating instances of proton irradiation and annealing. Other parameters of the APD are also under scrutiny to see if they change over the course of this experiment. A secondary objective explores if it is better to anneal only when the dark count rate drops below a threshold value or anneal after fixed intervals of time regardless of the dark count rates. A novel algorithm is also developed to analyze the afterpulsing probability. The algorithm is applied to simulated thermal count and afterpulse timetags to demonstrate its effectiveness. It is also applied to the data set drawn from the experimental tests of proton irradiation and thermal annealing to calculate the afterpulsing probability.

In chapter 4, the APD's are exposed to the vacuum conditions that they would encounter in low Earth orbit. The bracket and radiator are set to different temperatures to simulate thermal variations in orbit. The detectors are then characterized for the aforementioned parameters to ensure they are in operable range.

In chapter 5, I distinguish between the usual notion of afterpulsing probability which relates to the probability of trapping of charge carriers and the observed afterpulsing probability which differs from the former due to dead time and recharge time. I then use first principles to derive the observed afterpulsing probability as a function of charge trapping probability, charge detrapping lifetime, dead time and thermal count rate. I further go on to lay the theoretical ground work the afterpulsing algorithm developed in section 3.3.8 by deriving the analytical expression for the histogram between time differences of consecutive time tags and then justifying the linearity of the thermal region of the histogram.

Finally, in chapter 6, I devise a numerical experiment where I simulated a quantum link to implement the quantum key distribution. I model the photon statistics of a weak coherent pulsed laser and the parameters of a Si APD (the one mentioned above). I then model the coincidence algorithm to determine photon coincidences and model a simple intercept and resend attack. I calculate statistical quantities of quantum bit error ratio and sifted key generation rate.

Acknowledgements

I would like to thank Ramy Tannous, Jin Gyu Lim and Dr. Sascha Agne for helping out with conducting the radiation experiments at TRIUMF. I would like to thanks Dr. Brendon Higgins for his software support for the experiments regarding thermal vacuum and radiation tests as well as doing a lot of high level planning for these experiments. I would like to especially thank Dr. Jean-Philippe Bourgoin (JP) for overseeing my research work especially in the experiments and also for doing much high level planning for the experiments. I would also like to mention a special thanks to him for the numerous discussions I have had with him regarding technical aspects of these projects. I would also like to thank Brian Moffat for doing the administrative work surrounding the experiments and making sure we were on schedule. I want to thank Paul Marchand from the Neptec Design Group for his efforts towards the thermal vacuum tests and Excelitas Technologies Corp. for designing and manufacturing the silicon avalanche photodiodes. I would also like to thank Prof. Vadim Makarov for his personal guidance and motivation through all the experimental projects. Finally, I would like to thank Prof. Thomas Jennewein who was my supervisor for my research in the masters program here at the University of Waterloo and the Institute for Quantum Computing. His guidance and oversight of my research has been instrumental over the past two years. I would also like to thank the funding agencies: Canadian Space Agency, Natural Sciences and Engineering Research Council of Canada, Canada Foundation for Innovation and Ontario Research Fund for their financial support towards the projects.

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 The need for remote key generation

When a person, say Alice, wants to communicate with another person, say Bob, they can do so in person and ensure their conversation is secure from an eavesdropper. If the two parties are situated in geographically distinct regions and cannot physically meet, the next best choice is to send a message over a secure channel that they believe an eavesdropper, say Eve, doesn't have access to it. However, if an eavesdropper does manage to gain access to the channel, she can listen to Alice's message. A worse situation is where no secure channel is available for communication. Such an unsecure channel is easily prone to an attack. In such cases, Alice might want to modify the message such that the modified message isn't very meaningful even if Eve gains access to it. When Bob receives the message, he needs to decipher the message. If Alice uses a certain mapping to convert the original message to the modified message, Bob would need to perform the inverse mapping on the modified message to recover the original message. In such a case, Alice and Bob both need to agree on the mapping they use. This is fine but if multiple messages are sent through the unsecure channel, Eve could over a period of time build up partial knowledge of the mapping used especially if the original message has a familiar structure like the English language. To prevent Eve from doing this, Alice and Bob could change the mapping every time they send a message, so that Eve doesn't get many samples of the modified message that use the same mappings to decipher the mappings used. A neat way to do this is for Alice and Bob to agree on a general class of mappings usually called a protocol and change some parameter of the protocol such that Alice and Bob both know the value of the parameter in each instance. The sequence of values of the parameter is called the key, with Alice and Bob each having a copy of the key. A simple example would be that Alice

wants to send Bob a series of numbers. For each number, Alice and Bob could agree on the mapping to be the multiplication operator. Alice and Bob then agree on a ordered set of rational numbers such that when Alice wants to send the first number, she multiplies it with the first rational number in the set an sends the product instead. Bob then has to divide the received number by the first rational number to decipher Alice's original number. There exist protocols like the RSA which uses one key to encrypt and another key to decrypt the message. It mainly relies on the computational complexity of hard problems like factoring a very large number into primes. However, such systems are not guaranteed secure [2]. Given enough time, they can can be broken. This is fine for simple conversations or even a bank transaction. However, government secrets especially in the matters of national security, for example, need to be kept secret for decades altogether. Also, with the advent of quantum computers certain quantum based algorithms like the Shor's algorithm [3] can solve these hard problems including discrete logarithm problem and the elliptic-curve discrete logarithm problem [4] quite efficiently making the need to guaranteed secure protocols. A protocol called the one time pad protocol [5] uses the modular addition as the class of mappings. It adds the bits of the message to the bits of the key to get the encrypted message. The one time pad protocol is considered unbreakable under the assumption that the key cannot be reused. However, Alice and Bob cannot possibly hold the same of copy of an infinitely long key. Thus, they must meet in person again to generate another key once the first one is used up. If this is impossible and they must risk reusing the key and sacrifice security of the protocol. It is highly desirable to have a method where one can generate key remotely.

1.2 Quantum Key Distribution

To prevent attacks by quantum algorithms, many post quantum cryptographic (PQC) systems have been proposed. A thesis [6] by a masters student at the Institute for Quantum Computing briefly describes PQC systems.It was mentioned in the previous section (1.1) that the one time pad protocol is a classical protocol that guarantees secured if key are not re-used. Quantum key distribution (QKD) is a protocol that uses the laws of quantum mechanics, in particular, the random nature of the collapse of a quantum state, to generate generate exact copies of a secure key. A few different protocols have been proposed in this domain. The first one was the BB84 protocol [7]. Here, Alice first prepares single photons using a source like a weak coherent pulsed (WCP) laser source in specific polarization states. These polarization states are the source of information in this protocol. Since the quantity of polarization has quantum properties, these polarization states are called

qubits. Alice has two basis sets, rectilinear HV and diagonal DA basis set. She randomly chooses a basis set and then randomly one of the basis states in H, V, D or A in that basis set. She sends the series of photons over to Bob who then measures it in one of the same two basis sets. When Bob uses the same basis set as Alice did for a specific photon, the state that Bob detects is exactly the same as the state that Alice encoded the photon in. After the series of measurements, Alice and Bob publicly share their measurement basis set choice but not the preparation / measurement state. They then retain only those bits where the choice of same basis set was used by them. In the absence of an eavesdropper or more general noise, Alice and Bob both have the same copy of the key. A more detailed discussion of this protocol is presented in chapter 6. Using this key a communication protocol like the one time pad can be used to guarantee secure communication. Similar variants of the BB84 protocol like the BB92 [8] and E91 [9] exist.

1.3 Satellites for quantum key distribution

The photons need to be transmitted from Alice to Bob. One choice is the use of optical fibers. However, fibers experience losses that scale exponentially with distance [10]. Currently, distances of upto 420 km have been reached [11]. Another choice is atmospheric free space. Here too there exist some minorly contributing losses that scale exponentially with distance. However, the leading losses like beam divergence scale quadratically [10]. This makes free space more tractable. At least as of 2007, the maximum distance for transmitting single photons through atmosphere is 144 km [12]. However, in free space, one is still limited by the curvature of the Earth which limits the distance for communication. One way to get over this is to build a set of intermediate nodes which are owned by trusted entities [13]. The issue with trusted nodes is that if one in a network is compromised, the entire network is compromised. Quantum repeaters use entanglement to extend the distance of communication [14] but the technology is still in its infancy. Another approach to get over the line of sight issue is to build tall towers to increase the distance covered by line of sight. However, the height of the towers is still limited by the strength of the foundation and the building material. This insight of increasing the height above the ground can be motivated in the use of a satellite instead of a tower. This increases the range of communication by itself and when used in a network of satellites, it can cover large distances [15]. Also, losses due to atmospheric turbulence are absent in the vacuum of outer space.

There are two choices with respect to transmitter and receiver when using satellites. One can use the transmitter on the satellite and the receiver on ground. The other is to

use the source on the ground and the receiver on satellite. The transmitter like a laser source can often tend to be more bulky than the receiver like a detector unit, giving an advantage to using the receiver on the satellite [16]. Also, especially for science satellites (which are used to test scientific hypotheses), different experiments can require very diverse kinds of sources like weak coherent pulsed laser, entangled photon source. If one places the source on the satellite, all sources that are used in the intended experiment have to be on the satellite simultaneously, increasing the payload. Alternatively, with a limited payload capacity, one must restrict the variety of experiments that one can perform. One the other hand, most experiments like QKD or Bell's inequality tests require more or less the same kinds of detectors. Thus, if one places the detectors on board the satellite, it allows for a larger variety of experiments with a smaller payload. This is true for communication satellites if they want to utilize different communication or encryption protocols.

1.4 Silicon based avalanche photo diodes as detectors

We need the capability to detect single photons. Many detectors are commercially available for this purpose. Among them are Silicon Avalanche Photodiodes (APD), InGaAs APD and superconducting single photon detectors (SPD). Silicon APD's have low dark count rates, high detection efficiency, high saturation value (maximum detector count rate) and do not have the requirement of cryogenic cooling [17]. These advantages make them suitable for space applications where cryogenic cooling can technologically challenging. Dark counts are of primary concern in Si APD detectors. Before going into this, I would like to address the general working of an APD.

APD is generally $p-n$ junction with a depletion layer. The depletion layer is usually devoid of mobile charge carriers. If one applies a reverse bias to the $p-n$ junction - applying the positive terminal of power source to the n-type region and the negative terminal of power source to the p-type region, there is a minimum bias voltage to be applied before the APD can conduct and charge carriers flow across the junction. This threshold voltage is called the breakdown voltage. If the bias voltage is larger than the breakdown voltage and a mobile charge carrier exists in the junction, the electric filed in the junction can cause it to knock charge carriers of out of the neighboring atoms which in turn knock off more charge carriers. This process is called an avalanche.If the device is constructed such that the initial mobile charge carrier is knocked off by an incident photon, the $p-n$ junction can be used to detect light. If the bias voltage is slightly larger than the breakdown voltage, the field strength is minimal and the avalanche created by an incident photon eventually dies down. The current so generated by a single photon is extremely small - off the order

of tens to hundreds of electrons [18] - which is difficult to detect. If multiple photons are incident, the avalanche current so generated is large enough to be detected. This current tends to be linearly related to the incident optical intensity. So, this is a good device to measure optical intensity . However, it cannot be used to detect single photons.

Instead if the bias voltage is much larger than the breakdown voltage (in our experiments, we use 20 V), then the field strength is high enough to cause a runaway or self-sustaining avalanche even when the avalanche is triggered by a single photon [19]. This means that whether one or multiple photons are incident of the APD, you get pretty much the same avalanche because it is self sustained by the mobile charge carriers knocking off other charge carriers. So, such a device cannot be use to measure optical intensity and we say that the APD is not photon resolving. However, it can be used in detecting single photons. Hence in this mode (also called the Geiger mode), it is called a single photon avalanche diode.

Since you have a runaway effect of avalanche current (which stabilizes to a maximum value), once a avalanche is triggered, the APD cannot be used to detect any further photons. The avalanche current must be reduced to zero before a second detection can be made. We call such the process of killing the avalanche current as a quenching process. A circuit external to the APD must do this. As previously noted, an avalanche can only occur when the bias voltage is above the breakdown voltage. So, one way to stop a runaway avalanche is to decrease the bias voltage below the breakdown voltage. If a resistor (called the ballast resistor) placed in series with the APD, then when no avalanche current exists, the bias voltage applied across the series combination effectively acts on the $p-n$ junction. However, when an avalanche is triggered and a current flows, the current through the resistor induces a voltage drop across it. Since the resistor is in series with the $p-n$ junction and the voltage across the series combination doesn't change, the effective voltage across the $p-n$ junction decreases [20]. If the resistance value of the ballast resistor is chosen chosen high enough, the voltage across the $p-n$ junction could drop below the breakdown voltage following an avalanche. Below the breakdown voltage, the charge carriers are not aided by the internal electric field strength to sustain the avalanche. Hence the avalanche stops. The voltage across the $p-n$ junction then rises to the original bias voltage on a time scale which is determined by the product of the ballast resistance and the total capacitance (sum of the diode capacitance and any capacitance) [21]. We use passive quenching in the APD's in our laboratory. There are also active quenching methods [22] which involve a dedicated circuitry to detect the leading edge of an avalanche and decrease the bias voltage.

Dark counts are avalanches that occur when no photon is incident of the APD. They consist of two types [23]. One is the thermal counts. These are thermally triggered

avalanches that occur due to certain defects in the crystal - the rate of thermal counts increases linearly with temperature. The thermal count rate also increases with over voltage (bias voltage minus breakdown voltage) because the stronger internal field strength inside APD increases the chances of creating the initial mobile charge carrier that triggers the avalanche. The second is afterpulses that are avalanches correlated with a specific avalanche that occur previously. There are defects in the APD substrate such that when an avalanche occurs, one of the mobile charge carriers can get trapped in these deep levels. Once a charge carrier (usually an electron for Si APD) is trapped, it gets detrapped a finite time later which follows and exponentially decaying time distribution [24]. A detrapped electron can then serve to trigger another avalanche which is termed as an afterpulse. The number of afterpulses generated by any given avalanche generally depends on the number of deep level defects in the APD as well as the number of charge carriers in an avalanche. This means that the longer an avalanche exist for, the greater the expected number of trapped charge carriers. Here, a fast quenching circuit is helpful to quell the avalanche before many mobile charge carriers have a chance of getting trapped in deep levels. If the number of deep levels are so high in the substrate that even a fast quenching circuit cannot mitigate the likelihood of charges getting trapped, a longer duration of quenching might help. Here, the idea is not necessarily in preventing the trapping of charge carriers but instead preventing the detrapped charge carrier from triggering another avalanche by maintaining the bias voltage at or below the breakdown voltage. This of course limits the maximum detector count rate and for deep levels with significantly large expected detrapping times, it becomes impractical to use such a technique.

1.5 The problem of radiation damage in avalanche photo diodes

Although low dark count rates in freshly manufactured Si APD's are desirable, these devices are prone to radiation damage caused by protons emitted by the Sun [25]. The protons displace atoms/molecules in the APD's substrate creating defects. Such defects can potentially increase the dark count rate [1] - especially the rate of thermal counts. The energy of the incident proton is related to the likelihood it will create these displacements [26]. The slower moving protons (lower energy) will transfer more energy towards these displacements. The energy of the proton that goes into displacements is characterized by a quantity called the non ionizing energy loss. Decreasing the temperature of the detector can decrease the likelihood of generating initial mobile charge carriers that trigger the avalanche [27]. Also, increasing the temperature temporarily can fix the displacement defects caused by incident protons [27]. However, repeatedly irradiating and annealing

has not been researched. This will be addressed in chapter 3.

1.6 Nominal Values of detector parameters

Finally, I wanted to address some of the nominal values of the parameters of the Si APD that we are looking for. A detailed discussion about what these parameters mean and how they are measured will be given in section 2. The thesis [1] by Elena Anisimova describes in detail the detector requirement for a satellite system with the detector on board the satellite. A link analysis was performed [28] to determine how the dark count rate affected the secure key generation rate. For QKD with a WCP source pulsing at 300 MHz with a 100 Hz dark count rate, a secure key was generated at a rate of 3.109 MBit per month whereas when the dark count rate was 1 kHz, 2.92 MBit per month of secure key was generated. When the dark count rate was increased to 10 kHz, the secure key rate was negligible. The aforementioned dark count rates are with respect to the link and hence have contributions from each of the four detectors that are proposed for QKD operation. With this in mind, a dark count rate of 200 Hz per detector (and hence 800 Hz overall) was aimed for as a maximum threshold, going into our experiments.

The secure key length was also obtained for different wavelengths of the laser source on the ground [28]. 785 nm was found to be generate the longest secure key for both WCP source and entangled source. However, the secure key length was greater with the WCP source. Hence, for the experiments in chapters 2, 4 and 3, we use a WCP source with wavelength 785 nm.

Ref [29] gives further guidelines in selecting detector parameters. A request for proposal [30] of the detector project with the Canadian Space Agency details the requirements of the detector. A timing jitter of less than 1 ns (also stems from ref [28] which mandated the system to have a time resoultion of about 0.5 ns), a detection efficiency of 25% and an afterpulsing probability of the order of 1% was aimed for in each detector.

Chapter 2

Characterization of Single Photon Detectors

2.1 Test Setup

The QKD protocol requires the ability to detect photons in the four different polarization states - horizontal, vertical, diagonal and anti-diagonal. Correspondingly there four optical paths with an APD at the end of each. The APD is housed inside a detector module (DM) which keeps the APD's physically aligned parallel to each other. The prototype DM has four APD's while later on, an additional APD was added just in case one of the APD's suffered a failure when the DM was transported to the launch site. Figure 2.1 shows the DM as was used in the radiation tests (which will be detailed in chapter 3). The DM is a stand-alone unit with fiber optic cables sticking out of it to transmit photons to the APD's and coaxial electric cables to apply a bias voltage on the APD as well as to read the output electric signal corresponding to the APD's avalanche. Apart from this, the DM houses the electronics like the quenching circuit, etc. to operate the detector.

The temperature of the APD is of crucial concern because the thermal count rate which constitutes the background noise is temperature dependent. In general, higher temperatures result in greater thermal count rates, all other experimental parameters kept constant. It is thus beneficial to lower the temperature of the detector to lower the background noise. The signal photon rate is limited by the specifics of experimental set up. There is also a lower bound on the signal-to-noise ratio which is in turn related to the final key rate and the quantum bit error ratio. This imposes an upper bound on the thermal count rate. Thus, cooling the detector helps keep the thermal count rate below this limiting value. A temperature of $-80\,^{\circ}\mathrm{C}$ at the APD was aimed for. The DM houses a thermo-electric cooler (TEC) which uses the Peltier effect maintain a temperature difference between its ends. The cooler end thermally interfaces with the APD's substrate. The hotter end is

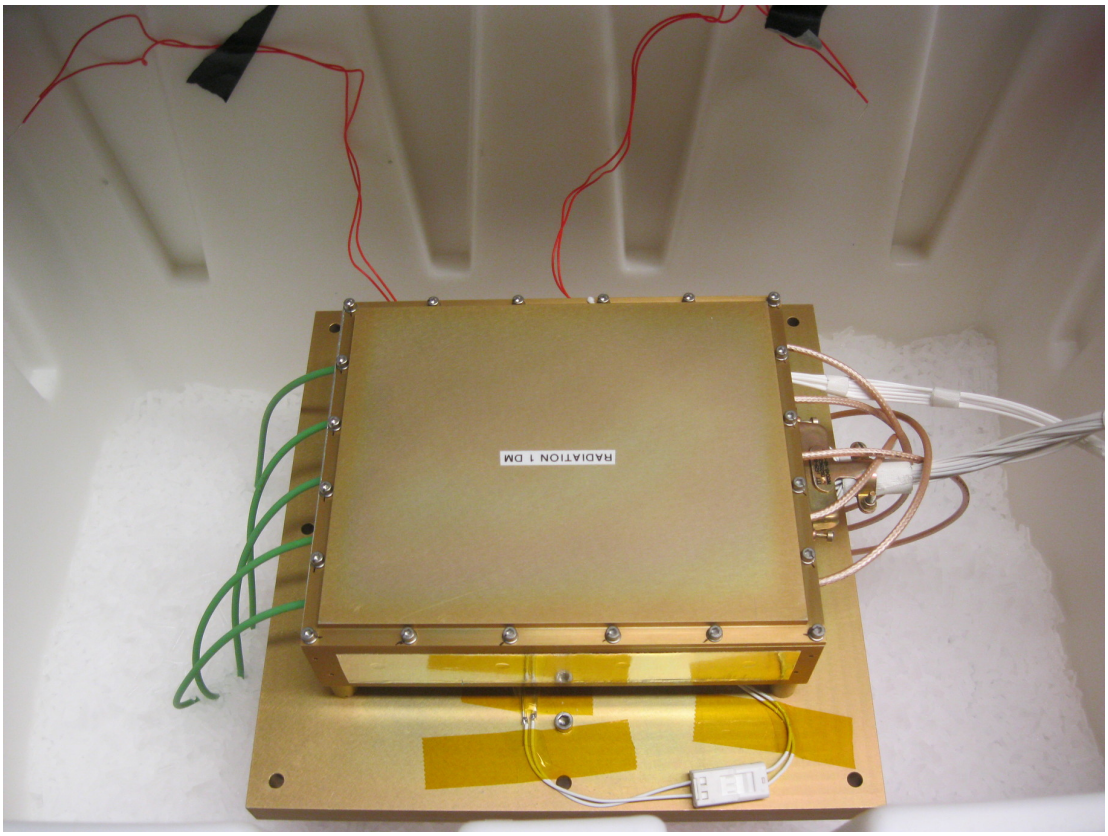


Figure 2.1: This is a photograph of the detector module which houses the avalanche photodiodes. It is attached to the top plate of the cold finger. The bottom plate (not in view) is immersed in dry ice inside a cooler to cool the radiator of the DM

interfaced with a heat sink called the bracket. The bracket thermally interfaces with the radiator. The radiator is modeled after a black body and is pointed to deep space(away from the Sun). In our lab, if the bracket is exposed to room temperature, the TEC would not be able to cool the APD to $-80\,^{\circ}\mathrm{C}$. In fact the TEC only maintains a temperature difference of around a few 10's of degrees between its ends. This is fine in outer space because when the DM is facing away from Sun, the bracket is able to cool down to $-70\,^{\circ}\mathrm{C}$ by radiative heat dissipation in the absence of the Sun's radiative heat input. In our lab, we cannot thermally expose the bracket to ambient room temperature to maintain the APD at $-80\,^{\circ}\mathrm{C}$. Instead we cool the bracket down to $-70\,^{\circ}\mathrm{C}$ using the freezer and let the TEC cool the APD further down to $-80\,^{\circ}\mathrm{C}$. The bracket also has a TEC whose hot end interface with the radiator (and ultimately the external environment) and the cold end interfaces with the bracket.

A discriminator circuit is present in the time tagger. The longer the time that elapses between two consecutive output pulses, the smaller is the amplitude of the second pulse. We use the discriminator circuit to ignore the second pulse if its amplitude is smaller than some fixed value. This is primarily done to prevent any fluctuations in output voltage due to noise from being time tagged. For our device, we used a discriminator voltage threshold of about 50 mV because we anticipate noise in the output to have smaller amplitudes than this value.

Figure 2.2 shows a schematic diagram of the test set up that was used in the laboratory at the Institute for Quantum Computing (IQC). The main purpose of this set up was to characterize the detectors under nominal conditions. In outer space, the DM would have to face away from the Sun or be in the Earth's shadow to perform the QKD protocol. Under such conditions, the DM gets very cold due to radiative heat dissipation. In order to recreate such cold conditions, the DM was placed in a freezer whose temperature could be regulated. The freezer was set to $-80\,^{\circ}\mathrm{C}$.

A thermistor is connected to each APD in the DM to measure the temperature at its detector's substrate. The resistance on the thermistor can be read out using a multimeter. Later on, a proprietary software was developed which was able to electronically read out and display the APD's temperature on a computer.

A 5-channel high voltage (HV) supply is connected via coaxial cables to the DM. This is used to apply a bias voltage to the APD's. The HV supply has knobs which can be used to adjust the bias voltage. The HV supply is also connected to the computer where the applied voltage can be read out on the software. Later on, the software was also able to set the bias voltage from within the user interface.

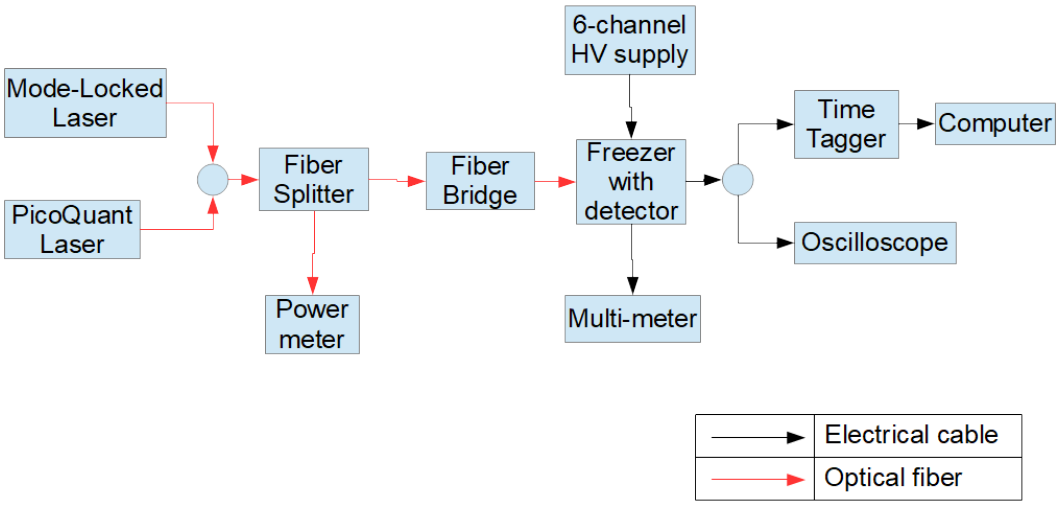


Figure 2.2: A schematic diagram of the test setup that was used in the laboratory at the Institute for Quantum Computing. The detector was placed inside a freezer to thermally simulate space environment. Either the mode-locked laser or the PicoQuant laser was used as a source of single photons. The output of the detectors can be visualized on to an oscilloscope or time stamped using a time tagger and stored in a data file which can be post-processed.

The coaxial cables that connect to the DM also serve to read out the electrical output signal. These coaxial cables can be connect to an oscilloscope that can visualize the output pulse to determine output pulse characteristics and also measure the recharge time. Alternatively, the coaxial cables can be connected to a time tagger that time stamps each output pulse. The time stamps are then stored in a data file and saved on a computer. From this time tag file, one can extract information about thermal count rate, detection efficiency, detector timing jitter, saturation and afterpulsing.

Two different laser sources were used, a mode-locked laser and the PicoQuant laser. The details of this will be explained in section 2.2.7. The fiber splitter diverts a part of the optical beam to the optical power meter whose reading can then be used to predict the power in the main optical path. The fiber bridge then distributes the optical per to each the detectors.

2.2 Detector Parameters

There are many parameters of the APD to characterize. Firstly, the break down voltage is measured (section 2.2.2). Then, a bias voltage of 20 V above the breakdown voltage is applied to the APD. Next, the output pulse is read out on the oscilloscope to measure the pulse amplitude and width (section 2.2.3). The oscilloscope is set to persistent mode and the recharge time is measured (section 2.2.4). The coaxial cables are then connected to the time tagger and the thermal count rate is measured in the absence of ambient light (section 2.2.5). Next, the signal count rate is measured and the data is post-processed to calculate the detection efficiency (section 2.2.6). The maximum count rate (also called the saturation) of the APD is measured (section 2.2.8). Finally, the afterpulsing probability is calculated from the time tag files (section 2.2.9).

2.2.1 Operating Temperature

When creating the prototype of the DM, a thermistor was used to get information about the APD's temperature. The thermistor has a thermally varying resistance. The resistance is a monotonic function with respect to temperature. Thus, measuring the resistance of the thermistor gives information about its temperature. The thermistor's resistance was measured using a multimeter. Simple calculations then let one arrive at the temperature of the APD. Later on, a proprietary software was developed so as to read out the temperature directly on to a computer screen. The temperature of the APD and the bracket is

individually controlled using separate closed loop feedback circuits. The temperature set points to be achieved on the TEC and bracket can be set using the software. In practice, the bracket temperature is set and we wait for it to stabilize. Then the APD temperature is set. Once the APD temperature stabilizes, one can begin the process of characterization.

2.2.2 Breakdown Voltage

As explained in section 1, the silicon APD is used in reverse bias. Bias voltage is the potential difference between the ends of the APD. There is a value of bias voltage below which the APD does not produce an avalanche. This value is called the breakdown voltage. The first task is to detect the breakdown voltage by applying a low bias voltage and increasing the bias voltage in steps of 20 V initially gradually decreasing the steps as the breakdown voltage is reached. The coaxial cable of one of the APD's is connected to an oscilloscope via a SubMiniature version A (SMA) port. The bias voltage can be adjusted either by knobs analog knobs on the HV power supply unit or digitally using the proprietary software. It is important not to increase the bias voltage in large steps because the energy in the avalanche increases with the over-voltage (bias voltage − breakdown voltage) and large energies involve imply greater heat generation in the substrate. The breakdown voltage is reached when the first signs of an output signal are noticed on the oscilloscope. The breakdown voltage is noted. The breakdown voltage increases with temperature. Therefore,it is essential to measure the breakdown voltage of an APD before measuring any of its other parameters. Otherwise, it might alter the true over voltage on the APD and alter other detector parameters like output pulse amplitude and thermal count rate. This is repeated for each of the APD's.

2.2.3 Output Pulse Characteristics

The bias voltage is set to 20 V above the breakdown voltage of the particular APD. By setting an appropriate trigger on the oscilloscope, the out pulse can be visualized on its screen. A representative picture of the output pulse is shown in figure 2.3. The output pulse rises from zero with time reaches a maximum value and decreases back to zero. The trailing edge is a little longer than the leading edge. The reason for this lies in the electronics external to the APD. The rise in the output voltage is due to increase in avalanche current after the avalanche is triggered. As part of the quenching circuit, a large resistance called the ballast resistance is connected in series with the APD. As the current increases through both the APD and the ballast resistance, the voltage drop across

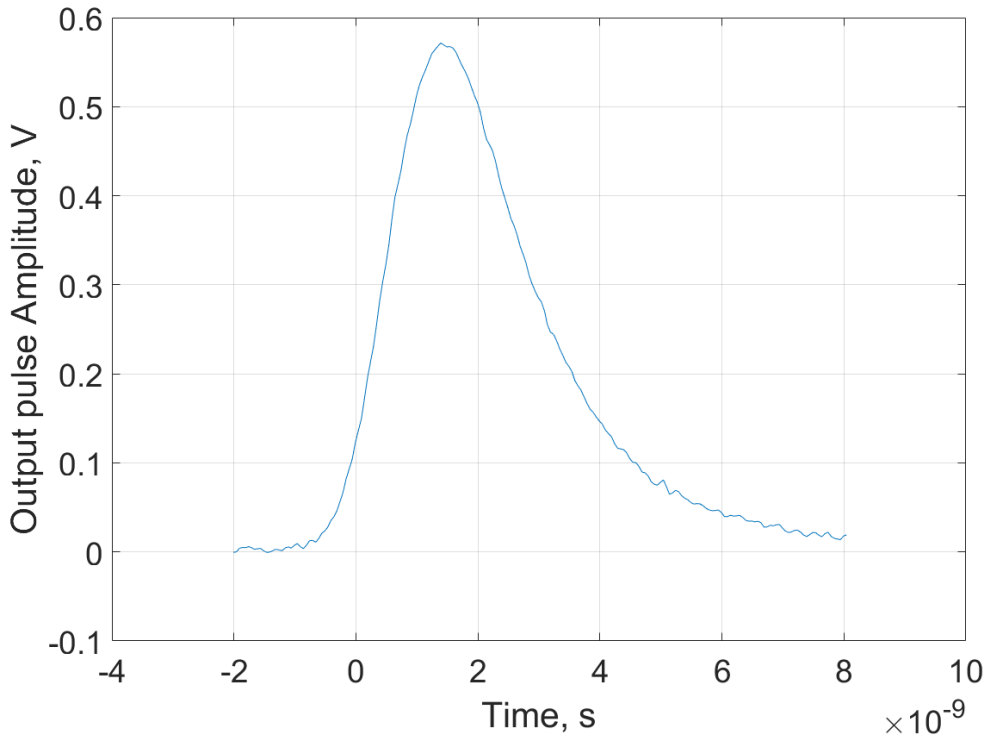


Figure 2.3: Sample output pulse trace of a silicon avalanche photodiode. This particular pulse was measured at 20 V above the breakdown voltage at a detector temperature of −80 °C

the ballast increases. Because a constant voltage is applied across the series combination, the voltage across the APD decreases and when it falls below the breakdown voltage, the avalanche current cannot self sustain and the current decreases. However, the effective diode capacitance and the parasitic capacitance from the wiring slow down the rate at which the current decreases. Hence the output voltage decreases slowly.

The maximum value of the output voltage is called the amplitude of the output pulse. The pulse amplitudes measured usually range from 550 mV to 660 mV. Two different output pulses can have different widths even if they have the same amplitudes. This can happen if the energy in the output avalanche changes. Thus, the pulse amplitude alone is not enough to characterize the output pulse. The width of the pulse at half the amplitude value must also be visually measured from the oscilloscope. This value is called the full width at half maximum (FWHM) value of the pulse. The pulse FWHM values usually range from 2.2 ns to 2.9 ns. Usually the energy in the avalanche depends on the over voltage. Also, the area under the curve is representative of the energy of the avalanche. Thus, the pulse amplitude and FWHM are negatively correlated to each other.

2.2.4 Recharge Time

The further apart two consecutive output pulses are from each other in time, the greater is the amplitude of the second output pulse. The amplitude asymptotes to a finite value. The curve is of the form

$$V = V_0 \left(1 - e^{-\frac{\Delta t}{\tau_R}}\right) \tag{2.1}$$

- Δt is the time elapsed since the dead time period induced by the first output pulse ended
- V is the amplitude of the second output pulse
- V_0 is the asymptotic value of the output pulse amplitude of the second pulse, i.e. value of output pulse amplitude as $\Delta t \to \infty$
- τ_R is called the recharge time. It is the characteristic parameter of the curve and determines the time scale over which the second output pulse's amplitude asymptotes.

If the data points that constitute the curve, recharge time can be calculated as follows. Consider the data point at corresponding to $\Delta t = \tau_R$, i.e., elapsed time is equal to the

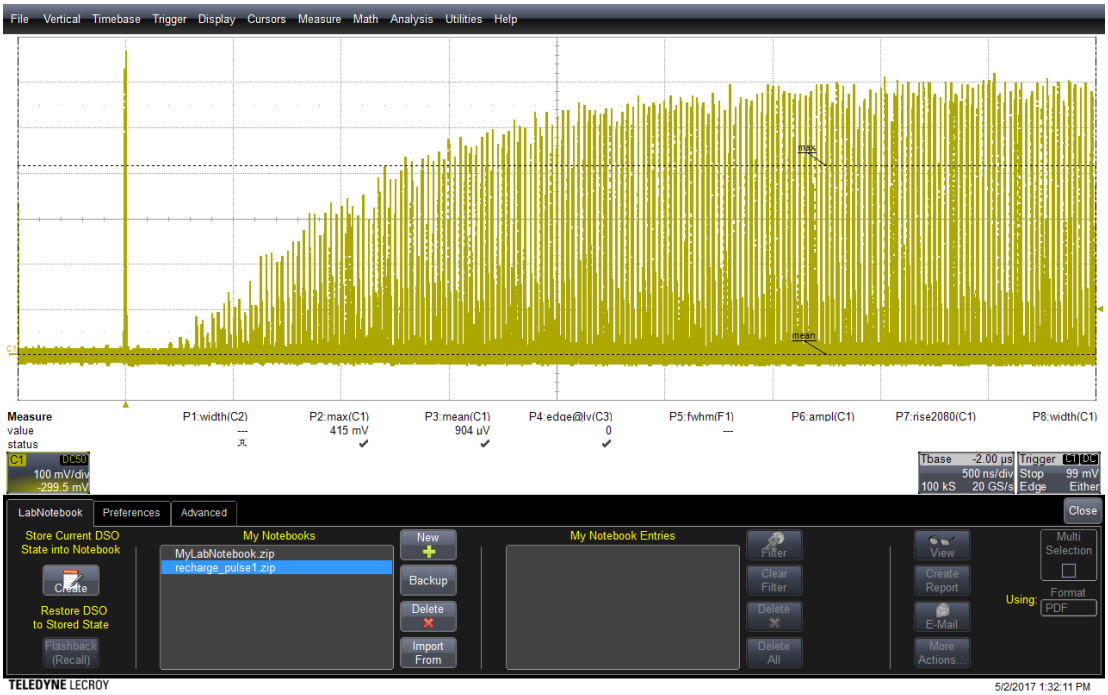


Figure 2.4:

recharge time. Let's consider the following ratio .

$$\left.\frac{V}{V_0}\right|_{\Delta t=\tau_R} = 1 - e^{-1} = 0.6321 \tag{2.2}$$

Thus the recharge time is the elapsed time when the voltage on the recharge curve in equation 2.1 has reached 63.21% of its maximum value.

To find the recharge time experimentally, the oscilloscope is set to persistent mode and one waits for some time for the curve to get populated with data points. Figure 2.4 shows a screenshot of the oscilloscope screen. The envelope of the yellow region represents the recharge curve. The x-axis is representative of Δt and the y-axis is representative of V. The tall vertical yellow line on the left of the screenshot in figure 2.4 is representative of the first output pulse. Basically each pulse shows up as a vertical line on the right side. Afterpulse that vertical line is plotted, the trigger goes back to the tall vertical left line and the next pulse is again plotted on the right side. It takes about 1-2 minutes for the graph to be populated with the data points. The time elapsed since the left yellow line when then the recharge curve attains 63.2% of its maximum value is deemed the recharge time for the APD. The values of recharge time usually range from 0.9 µs to 1.35 µs.

2.2.5 Thermal Counts

Thermal counts are detector clicks in the absence of signal photons. Also, each thermal count is not correlated to any other detector click. The measurement of thermal count rate must be done in the absence of ambient light in order to prevent stray photons from the environment reaching the APD. Also, it helps to cover all fiber optic cables that lead into the DM with a black cloth to prevent stray photons from entering the the fiber optic cables through its jacket. The SMA output from each detector is connected to the time tagger. The time tagger time stamps each detector output that passes through the discriminator circuit and sends it to the computer. A program in LabVIEW was developed by Dr. Brendon Higgins, a post doctoral fellow at IQC to store the time stamps in a data file. The data file can then be post-processed to get the thermal count rate. To clarify terminology, I will use the term dark counts to comprise of thermal counts and their associated afterpulses. When afterpulsing is negligible, the dark count rate is equal to the thermal count rate. In general,

$$\begin{aligned} \text{Dark count rate} &= \frac{\text{number of time tags in file}}{\text{Time duration of measurement}} \\ &= \frac{\text{number of time tags in file}}{\text{Time stamp of last time tag - Time stamp of first time tag}} \end{aligned} \tag{2.3}$$

An assumption is that the time tags in the file are sorted by increasing value of time stamps. The thermal count rate increases with APD temperature.Also, under the assumption that afterpulsing is negligible, equation 2.3 gives the thermal count rate as well.

At APD temperature of $-20\,^{\circ}\mathrm{C}$, the thermal count rate is approximately 10 - 30 Hz where as at an APD temperature of $-80\,^{\circ}\mathrm{C}$, the thermal count rate is of the order of 1 Hz. This is assuming that the APD hasn't been exposed to radiation which can damage the detector substrate and increase thermal count rate. Radiation damage will be addressed in section 3.3.4. Another assumption is that the afterpulsing probability is negligible. If the afterpulsing probability is substantial, this means that a good chunk of the detector clicks will be avalanches due to afterpulses. If p_{ap} is the afterpulsing probability, then true thermal count rate is given by

$$\text{Thermal count rate} = \text{dark count rate}\ (1 - p_{ap}) \tag{2.4}$$

Equation 2.4 can be derived as follows. In this experiments there are only thermal counts and afterpulses. By the definition of afterpulsing probability, each thermal count

count gives rise to p_{ap} first generation afterpulses on average. Each one of the p_{ap} first generation afterpulses also gives rise to p_{ap} second generation afterpulses on average because each avalanche is considered to be identical to every other. Thus each thermal count gives rise to p_{ap}^2 second generation afterpulses on average. Similarly, each thermal count gives rise to p_{ap}^3 second generation afterpulses on average and so on. This is illustrated in equation

$$1 \rightarrow p_{ap} \rightarrow p_{ap}^2 \rightarrow p_{ap}^3 \rightarrow p_{ap}^4 \rightarrow ..... \tag{2.5}$$

Assuming $p_{ap} < 1$, which it must be to prevent a runaway effect of detector clicks, equation 2.5 is a geometric series with first element as 1 and ratio of consecutive terms as p_{ap}. Thus the average total number of detector clicks due to one thermal pulse (including the thermal pulse itself) is $\frac{1}{1-p_{ap}}$. For a thermal count rate of d Hz, d thermal pulses occur in 1 s on average.Thus due to the afterpulsing probability, a total of $\frac{d}{1-p_{ap}}$ clicks occur on average in 1 s. Thus, the total click rate, i.e., the dark count rate, is $\frac{d}{1-p_{ap}}$. Hence equation 2.4 follows. A special mention must be made that the count rates mentioned in equation 2.3 must be evaluated on time scales much larger than the average detrapping time of a trapped charge carrier as well as the average time period of the thermal counts. Otherwise, the calculated value of count rates will depend on where in the time tag file you took your data set from.

2.2.6 Detection Efficiency

Not all photons that fall on the APD's substrate are detected. For example, physical APD's may not have 100% detection efficiency. Even if the APD had 100% detection efficiency (in asymptotic time), in the presence of dead time and recharge time, the APD will not detect some of the photons falling on it. The detection efficiency corresponding to some value of optical input power to the detector can be calculated. To do this, we first set the laser frequency to 5 MHz. For our experiments, we aimed to have APD click rates of about 100 kHz. This required us to maintain a certain photon rate hitting the APD. Experimentally, this was achieved by maintaining the optical power at the input side of the APD in the range of 100 fW. This value depends on the wavelength of the photon which affects its energy. Please note that when one changes the optical power on the laser, its pulse rate (5 MHz) is unaffected. Instead the average number of photons per pulse changes. Thus, we chose to stabilize the input photon rate to the APD, let's call it s_{in}, by setting the optical power to 100 fW. We can the calculate s_{in} as follows:

The wavelength of each photon generated by the laser was 785 nm. The energy of photon, E_{photon} is given by

$$E_{photon} = \hbar\ \omega = h\ f = \frac{h\ c}{\lambda} \tag{2.6}$$

- $\hbar$ is the reduced Planck constant and h is the Planck constant
- ω is the angular frequency corresponding to the photon and f is the corresponding frequency
- c is the speed of light
- λ is the wavelength corresponding to the photon

Since the total optical power hitting the APD is 100 fW, the corresponding photon rate is given by

$$\text{Photon rate hitting the APD, } s_{in} = \frac{100\,\text{fW}}{E_{photon}} \tag{2.7}$$

If the corresponding APD output click rate is s_{out}, Then photon detection efficiency η_{photon} of the APD is given by

$$\eta_{photon} = \frac{s_{out}}{s_{in}} \tag{2.8}$$

Note 1: It helps to perform the measurement without ambient light. So, the lights in the laboratory are turned off and all optical fibers are covered with a black cloth to prevent any photons from entering the optical path between the power meter and the APD, through the jacket of the optical fiber.

Note 2: It is assumed that thermal count rate is orders of magnitude less than 100 kHz. Also, the afterpulsing probability is negligible. This ensures that the majority of the output pulses are due to photon hitting the APD's substrate.

2.2.7 Timing Jitter

The laser is set to a fixed pulse rate. During each pulse a certain number of photons are released. These photons reach the APD and trigger avalanches in the detector's substrate which are eventually time tagged. Let's assume that exactly one photon is released during

each laser pulse. One can then ask if the time stamps in the detector's time tag file are evenly spaced, i.e., periodic. IF one analyses the file, he will noticed that the time stamps aren't evenly spaced. T Here is some uncertainty of time stamps about the expected periodic spacing. his uncertainty is called the timing jitter of the system. There are many causes for this.

Firstly, there is a reference pulse that signals the release of the photon in the laser. Ideally the references pulses should be evenly space in time by about 0.2 µs (corresponding to 5 MHz). However, each pulse occurs slightly before or slightly after the time instant is expected to occur. One can make a histogram of the position in time of the actual electric pulse relative to its expected time instant. One will find that the resulting time distribution is a log-normal curve whose trailing edge is longer than its leading edge. This curve is approximated as a Gaussian for simplicity. Also, the photon is not released exactly when the reference pulse occurs. In fact, the reference pulse triggers laser pulse time window. The photon is released at some time instant within that window. There is some uncertainty associated with the laser pulse time window. Both aforementioned uncertainties contribute to the laser's timing jitter. The jitter of the PicoQuant laser used in our experimental set up is about 465 ps. I will explain later how this was measured.

Secondly, when the photon hits the detector, there will be a small delay before an avalanche is generated and the electric signal (indicating an avalanche) is available at the output of the associated circuitry. One can find out the expected value of this delay experimentally. Then if one starts to measure the actual delay for many detector click, one will also find that the true delay is distributed as a log normal curve which can then also be approximated as a Gaussian. Our experiments aim to characterize the timing jitter of the detectors.

Finally, we use time taggers to time stamp the emission of photons on the transmitting side and detection of photons on the receiving side. When the electrical pulse that indicates the emission or detection of a photon reaches the time tagger, the time tagger time stamps it. One must keep in mind that the time tagger has a finite resolution - the time taggers used in our experiments had a resolution of 78.125 ps. This means that two electric pulses that occur closer in time than a threshold value t_{tag} are not guaranteed to be ascribed two distinct time stamps by the time tagger. To explain this in more detail, the time tagger divides the time axis into blocks of width t_{tag}. If two electric pulses occur within the same block, the tagger outputs only one time stamp. We say that those two electric pulses are not resolved. To be resolved, the electric pulses should arrive at the tagger in two separate blocks. We need to calculate the timing jitter of the time tagger. The time period of our laser is 0.2 µs. However, the timing jitter of the detector (which will be calculated shortly)

is at least three orders of magnitude lesser. Thus photons in consecutive laser pulses should arrive at the detector toughly spaced apart in time by 0.2 µs. The time taggers we use in our experiments have resolution $t_{tag} = 78.125$ ps which can easily resolves between them. Thus, realistic a small and negligible fraction of the photons in consecutive pulses will not be resolved. We can therefore ignore this effect. On the other hand, we can look at a single electric pulse arriving at the input of the detector. Say it arrives in some time block. The tagger ascribes it a time stamp corresponding to the time instant at the start of that time block. Looking at the time stamp, one cannot know with a 100% certainty where within the time block the electric pulse arrived especially because t_{tag} <<0.2 µs. This induces some uncertainty. Because one has no knowledge of where within the time block the electric pulse arrived, we can model the true arrival of the electric pulse as a uniform distribution over the time block. The standard deviation for a uniform distribution defined over a finite range t_{tag} =78.125 ps is given by $\frac{t_{tag}}{\sqrt{12}}$ =22.553 ps. Therefore, this is the timing jitter associated with time tagger.

We just mentioned a few different processes with associated uncertainties. But what about the overall uncertainty? If one looks at the time stamps on the time tagger that time stamps the APD's out put pulses, the calculated uncertainty in the time stamps cannot be explained by the time tagger alone. This is because along the way the detector and the laser circuitry have each also imparted some uncertainty. In probability theory,

In our case, there are uncertainties associated with the laser circuit, detector circuit and two time taggers (for photon emission and detection). A key theorem in probability called the central limit theorem [31, 32] states that if you have independent random variables and look at the sum or average of them, the result random variable approaches a Gaussian distribution as the number of samples increases, even if the individual random variables are not Gaussian distributed to begin with. Also, the variance of the resulting random variable is equal to the sum of variances of the individual random variables. For two random variables $X$and Y which are independent of each other with each variable having its own uncertainty, the joint uncertainty is calculated as follows:

$$\begin{aligned} Var(X,Y) &= Var(X) + Var(Y) \\ \sigma_{X,Y} &= \sqrt{\sigma_X^2 + \sigma_Y^2} \end{aligned} \tag{2.9}$$

If we calculate the time difference between the time-tagged photon detection and the corresponding time-tagged laser reference pulse and create a histogram over a large number of instances of different photons, the resulting histogram should approach a normal

distribution according to the central limit theorem since the laser reference pulse, time tagging and photon detection events occur independently of each other. Thus the combined uncertainty of the system σ_{sys} is given by

$$\sigma_{sys} = \sqrt{\sigma_{laser}^2 + \sigma_{det}^2 + 2\sigma_{tag}^2} \tag{2.10}$$

- σ_{laser} is the uncertainty in the laser and its associated circuitry
- σ_{det} is the uncertainty in the detector and its associated circuitry
- σ_{tag} is the uncertainty associated with the time tagger

Please note that σ represents the standard deviation of the respective distributions.

When one analyzes the time stamps in the time tagger that is connected to the detector, they contain uncertainty from the laser, detector and time tagger because all these processes are sequential and any uncertainty accumulated in one process carry forward. Thus, the uncertainty in the time stamps of the time tagger connected to the detector is the overall system's timing jitter, σ_{sys}. To do this, one calculates the time difference between the given stamp t_1 in the detector's time tagger and the time stamp in the laser's time tagger that corresponds to t_1 and creates a histogram across many photons. A program to do this was created by Dr. Brendon Higgins. I created a program in Mathematica to take as input the histogram just mentioned and curve fit the data points using a Gaussian equation. The analytical expression of this Gaussian is obtained and the standard deviation of the Gaussian is found out. This standard deviation is the system's timing jitter σ_{sys}.

One now is in a position to calculate the timing jitter of the detector as follows:

$$\sigma_{det} = \sqrt{\sigma_{sys}^2 - \sigma_{laser}^2 - 2\sigma_{tag}^2} \tag{2.11}$$

We then converted this standard deviation of the Gaussian for the detector to a FWHM value of the same Gaussian. The conversion is as follows for a Gaussian:

$$\text{Full width at half maximum for a Gaussian, FWHM} = 2\sqrt{2log_e(2)}\sigma \tag{2.12}$$

The typical FWHM values measured were measured to be around 650 ps

I had earlier mentioned that the laser's timing jitter was measured to be 465 ps. Here's how. We used a mode-locked laser which has a very short laser pulse width (the time

window in which the photon is released). The pulse width of the mode locked laser is around 50 fs and hence the jitter associated with it is orders of magnitude smaller than the jitter of the detector and hence it can be ignored. To calculate the jitter of the APD, one needs to look at multi-photon events. For example, let's look at two photon events where two photons are produced in the same laser pulse. One of the photons hits detector 1 and the other hits detector 2. Whatever the jitter of the laser, it is safe to assume that they impinge on the detectors simultaneously. Since the two detectors are independent entities and the output signals are processed independently, one would expect the time stamps due to the two photons to be different. The difference between the time stamps will be related to the detector's inherent jitter. The greater the jitter, the greater is the expected absolute difference. The time difference in the time stamps corresponding to those two photons can be histogrammed for many distinct two photon events in detectors 1 and 2. The uncertainty in the resulting distribution is calculated as σ_{12}. Similarly, one can analyze two photon events between detectors 2 and 3 and also between between detectors 3 and 1, resulting in corresponding uncertainties σ_{23} and σ_{31}. Let σ_1, σ_2 and σ_3 be timing jitters of detectors 1, 2 and 3, respectively. The aforementioned uncertainties are approximated in terms of the detector timing jitters as follows:

$$\begin{aligned} \sigma_{12} &= \sqrt{\sigma\prime_1^2 + \sigma\prime_2^2} \\ \sigma_{23} &= \sqrt{\sigma\prime_2^2 + \sigma\prime_3^2} \\ \sigma_{31} &= \sqrt{\sigma\prime_3^2 + \sigma\prime_1^2} \end{aligned} \tag{2.13}$$

Now we use an arithmetic trick to find the timing jitter of detector 1 as follows:

$$\begin{aligned} \sigma_{12}^2 + \sigma_{31}^2 - \sigma_{23}^2 &= \sigma\prime_1^2 + \sigma\prime_2^2 + \sigma\prime_3^2 + \sigma\prime_1^2 - \sigma\prime_2^2 - \sigma\prime_3^2 \\ &= 2\sigma\prime_1^2 \end{aligned} \tag{2.14}$$

Therefore,

$$\sigma\prime_1 = \sqrt{\frac{1}{2}(\sigma_{12}^2 + \sigma_{31}^2 - \sigma_{23}^2)} \tag{2.15}$$

Actually $\sigma\prime_1$ is not the inherent timing jitter of the detector because there is some contribution from the time tagger itself. This is because $\sigma\prime_1$ is derived from the $\sigma\prime_{ij}$'s,

which are calculated from time stamps and hence include uncertainty contributions from the time tagger. The inherent timing jitter of detector given by σ_1 can be expressed as

$$\sigma_1 = \sqrt{\sigma t_1^2 - \sigma_{tag}^2} \tag{2.16}$$

Finding the timing jitter of the detectors in our laboratory at IQC was ideal. However, as will be seen in chapters 4 and 3, we need to characterize the timing jitter of the APD's at off-site locations. This poses a problem as the mode-locked laser is not portable. Thus use a more portable laser from PicoQuant. However, the PicoQuant has a wider pulse width which induces which contributes towards the overall system jitter. So, the first task is to characterize the PicoQuant's jitter. To do this, we use the PicoQuant laser and find the resulting system timing jitter by analyzing the time difference between the corresponding time stamps of photon emission and detection. Then, we get can calculate the PicoQuant laser timing jitter by the equation

$$\sigma_{laser} = \sqrt{\sigma_{sys}^2 - \sigma_1^2 - 2\sigma_{tag}^2} \tag{2.17}$$

It is important to characterize the PicoQuant's jitter soon after we characterize the detector's jitter with the mode-locked laser in order to rule out the possibility of the detector's jitter varying in between - although the detector's jitter isn't generally expected to vary under nominal conditions.

2.2.8 Saturation

When one changes the input signal photon rate, the output detector click changes. However, the value of the click rate and how fast it changes with respect to input photon rate can vary. To explore this, we conduct an experiment where we increase the laser power and observe the output detector click rate. Figure 2.5 shows the time-tagged APD click rate as a function of reference power.

At low reference power, we find that there is almost a linear relationship between APD click rate and reference power. This is expected because the detection efficiency is more or less constant and greater the input photon rate, greater the output APD click rate.

As the input signal photon rate hitting the APD further increases, the output click rate does increase. However, we must keep in mind that there is a finite recharge time. The more temporally dense the detector's output clicks, the lower is the detection efficiency when any given photon impinges on the APD's substrate. This is because according to equation

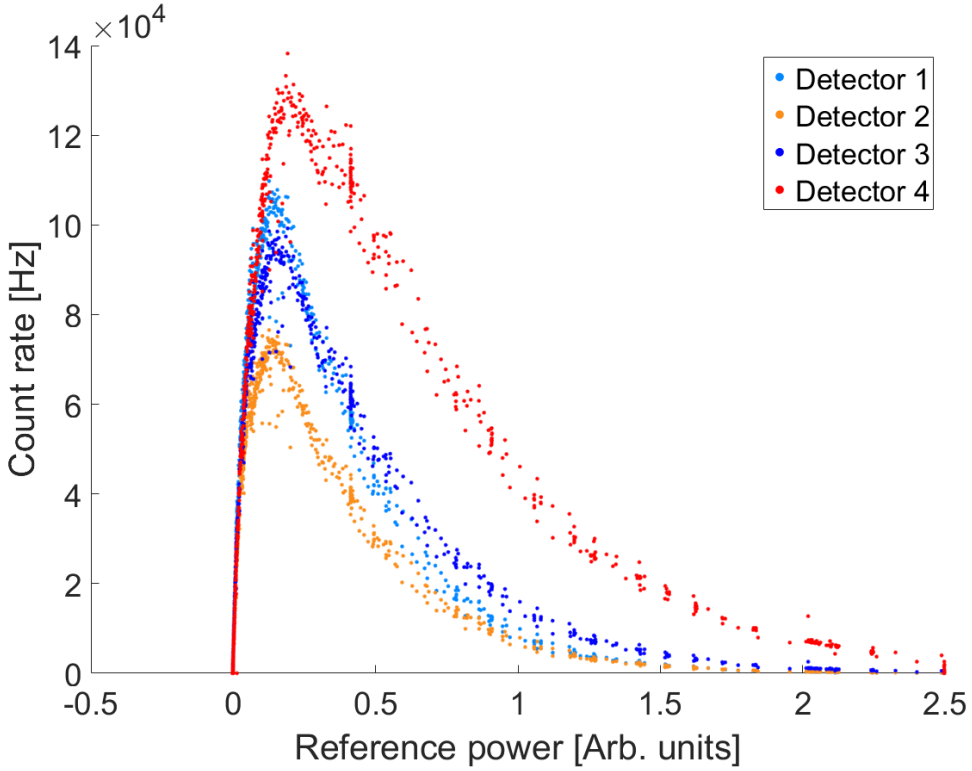


Figure 2.5: The laser power was increased from zero till the detector count rate increased from the dark count rate to a maximum and fell back down to zero. This maximum value is called the saturation value of the detector. The reference power was measured on the optical meter and the units as such are not important to calculate the saturation value. This data set is from the prototype of the detector which had four APD's. The temperature of the APD's was set to $-40\,^{\circ}\mathrm{C}$. The threshold on the discriminator circuit was set to $50\,\mathrm{mV}$.

2.1, the recharge time curve doesn't have enough time to asymptote to the maximum bias voltage and the detection efficiency $\eta_{recharge}$ due to recharge is dependent on the bias voltage. The overall time dependence of the $\eta_{recharge}$ on time Δt elapsed after the previous avalanche is given by

$$\eta_{recharge} = \frac{1}{1 - e^{-\frac{20}{8.5}}}(1 - e^{-(1-e^{-\frac{\Delta t}{\tau_R}})\frac{20}{8.5}}) \tag{2.18}$$

Equation 2.18 assumes that the detector has a maximum time asymptotic efficiency of 100%. Thus the probability that a given photon is detected by the APD is low when the input signal photo rate is high. As the signal photon increases, the loss of detection efficiency catches up and the saturation curve in figure 2.5 starts to increase at a sublinear rate and eventually starts decreasing in value.

Furthermore, there is a finite dead time. When the time period of the input signal photons approaches the dead time, we should see the recharge time curve asymptotes to a value equal to $\frac{1}{\text{dead time}}$. However, we do not see this in the experimental curve. This is because we have in place a discriminator circuit that does not let and output APD signal of voltage less than 50 mV to pass through to the time tagger. When the output APD avalanche rate increases, the amplitude of the output signal of the APD decreases in accordance with the recharge time curve. Thus, the discriminator circuit blocks some of these low voltage APD output signals. Hence instead of the expected asymptotic behavior, the saturation curve starts to decrease in value. Eventually at very high input signal photon rate, the APD avalanches at a high rate and almost all its output pulses have amplitudes lesser than the discriminator's threshold value of 50 mV, causing the time tagger to not register any clicks. Thus, the saturation curve eventually drops to zero.

This behavior is interesting. But from a practical point of view, we are interested in the maximum count rate, i.e., the maximum of the saturation curve also called the saturation value of the APD. In QKD operation, we would like to operate the APD at input powers lower than the power corresponding to the maximum output count rate.

In our experiments, the saturation value is in the range 0.6 - 1.1 MHz.

2.2.9 Afterpulsing

During an avalanche, electrons can get trapped at trap sites in the APD's substrate. Some time later, the electron gets trapped and causes a second avalanche. This second avalanche is called an afterpulse. This is undesirable because afterpulses tend to increase noise in the

time tag file. The average number of afterpulses directly generated by a given avalanche is called the afterpulsing probability.

There are multiple ways to calculate the afterpulsing probability. One of the ways was elucidated by Dr. Elena Anisimova [1]. The experimental data in taken from the dark count measurements elucidated in section 2.2.5. It involves forming a histogram of all detector clicks following an arbitrary click (also called the start click). The bins used increase exponentially in width - in specific the base of the exponent is 10. This means that bins towards the left have smaller width and hence can better resolve more dynamic behavior in this region whereas bins towards the right have larger width and tend to average out any interesting behavior. Therefore, the histogram has a smooth tail because the bins towards to the far right are so wide that calculating the average count rate in each bin gives rise to more or less the same value. The bins towards the left of the graph show that series of afterpulses. The area under the curve above the steady state value is the afterpulsing probability. Ursin and Peev [33] had initially suggested this approach but with linear binning (equal width bins).

I worked on another method to calculate the afterpulsing probability. This method relies on the dark count experiment performed in section 2.2.5. If the afterpulsing probability were zero, there would only be thermal counts. One could find the time difference between consecutive clicks and create a histogram of these time differences. Thermal counts follow a Poisson distribution and the time elapsed between consecutive clicks follows an exponentially decaying distribution. Therefore, the aforementioned histogram should follow this exponential curve. However, under non-zero afterpulsing probability, there would be afterpulses in the time tag file. If one creates a similar histogram of time difference between consecutive detector clicks, one should expect a deviation from the exponentially decaying curve. Tee greater the afterpulsing probability, the greater this deviation. By quantifying this deviation, one can measure the afterpulsing probability. This forms the intuition for the algorithm.

Assume zero dead time. Assume no stray photons from the environment enter the optical fibers

A key assumption is time scale of the afterpulsing (afterpulsing time constant) is of the same order as the average time period of the thermal counts. I chose to have exponentially bins so that the tail of the histogram will be smooth. This is required because we would later try to curve fit this region. Since the afterpulsing occurs on a smaller time scale than the thermal counts, the tail of the histogram primarily consists of thermal counts only making the tail nearly exponential (a decaying exponential) in behavior. However, the region towards the left of the histogram consists of both thermal counts and afterpulses.

Here we would expect a deviation the exponential curve. By curve fitting the tail to an exponential curve, one can extrapolated curve fitted exponential towards the left of the histogram. The extent that the histogram deviates from this curve fitted exponential as measured by the area under the histogram but above the curve fitted exponential is representative of the afterpulsing probability.

Since the tail of the histogram consists mostly of thermal counts, it follows the p.d.f. y of the next thermal count

$$y = d\ e^{-d\Delta t} \tag{2.19}$$

- d is the thermal count rate
- Δt is the time elapsed since the previous detector click
- y is the value of the histogram's p.d.f.

Taking logarithm to the base e on both sides of equation 2.19 we get

$$\log y = \log d - d\Delta t \tag{2.20}$$

Equation 2.20 is a linear equation in $\log y$ and Δt. If you extrapolate this linear curve towards the left of the graph, the area under the extrapolated curve is due to thermal counts because thermal counts occur as the next click right from $\Delta t = 0$. Then, any area under the histogram's envelope but above this linear curve fitted line is due to afterpulses, assuming there are no photons from ambient light reaching the detector.

It is important to figure out the linear region towards the right of the graph when $\log y$ is plotted with respect to Δt during linear curve fitting. If data points in the non-linear region are used towards curve fitting, the resulting curve fitted line will not be representative of the thermal counts. One way to detect the linear region is for a human to visually determine it from the graph. However, the transition from linear to non-linear can happen gradually and every time a human tries to determine the boundary between the two, you might get slightly different answers.

It would be ideal to automate the process of determining the linear region. One way to do this would be to start from the right side of the graph and select, say, the 10 rightmost data points. Then linear curve fit these data points and find out the sum of squares of deviations of the data points from the curve fitted straight line to give a measure of the

how much that data points deviate from the straight line. If all the selected data points are within the linear region, this deviation should be small. Next consider the 11 rightmost points, then 12 rightmost and so on. Eventually with enough iterations, you will start considering points in the non-linear region as well. When this happens the deviation of the data points from the curve fitted straight line should be larger. One can then set a threshold for this deviation to determine the linear region. One problem with this approach is that the threshold has to be fixed. If one knew the true afterpulsing probability of a data set, one can play around with the threshold value and select an appropriate value to get the correct afterpulsing probability. Then one can use this value of threshold to work with data sets of unknown afterpulsing probability. However, it is difficult to know what an appropriate threshold is because one doesn't really know the true afterpulsing probability of the data set to begin with. So, it is difficult to know what range of thresholds gives an accurate afterpulsing probability. Also, the longer the linear region considered, the better is the linear curve fit because there tends to be a small amount of noise even thin this region. So, the extent of statistical noise in the tail puts a lower bound on the size of the region to be considered as linear. The trade-off of considering larger regions is that one runs the risk of considering points in the non-linear region. Since we are using exponential binning, any error in selecting the linear region contributes exponentially towards errors in the afterpulsing probability because the afterpulsing probability is related to the Riemann integral of the histogram curve.

I decided to use another approach towards selecting the linear region. There are two ways by which one can find the afterpulsing probability. One is as described above - select the linear region towards the right of the graph, linear curve fit this region, extrapolate the linear curve fitted line towards the left of the graph and find the are under the histogram but above the linear curve fitted line to get the afterpulsing probability. Another way is to first find the thermal count rate. This can be done because when one curve fits the linear region, the algorithm outputs the parameters of the fitted straight line. Equation 2.20 relates the parameters of the straight line to the thermal count rate d. However, one must keep in mind that equation 2.20 is a normalized p.d.f. But since our entire histogram itself is normalized, the fitted straight line is not normalized because there are afterpulses as well. The exactitude of the calculation will be detailed later. The afterpulsing probability can be calculated as follows:

$$\text{Afterpulsing proability} = \frac{\text{afterpulse rate}}{\text{overall count rate}} = \frac{o - d}{o} \tag{2.21}$$

where o is the overall count rate which can be calculated from the time tag file as follows

$$\text{overall count rate, } o = \frac{\text{number of time tags in file}}{\text{Time duration of measurement}} \tag{2.22}$$

The two methods of calculating the afterpulsing probability are completely independent of each other. They should give the same answer, provided the linear region is selected properly. When the non linear region is used to linear curve fit as well, the answers got from the two methods start to deviate. One can then put a threshold on this deviation to ensure that the linear region is selected properly. But again, we have the same issue of selecting the correct threshold. To overcome this problem, I instead iterated through all possible continuous regions in the graph and used these regions in each iteration towards linear curve fitting. I then selected a threshold e_{ap} for the extent of the mismatch between afterpulsing probabilities as calculated by the two methods. If the mismatch is smaller than the threshold e_{ap}, I store in an array, $array_{ap}$, the average of the afterpulsing probability values due to the two methods. Then I create a histogram H_2 of the afterpulsing probability values stored in $array_{ap}$. The intuition here is that when the linear region is selected accurately (either the entire linear region or a part of it), the afterpulsing probability values due to the two methods more or less match and is accurate. Since many of the choices of linear region are considered correct, there will be multiple terms in $array_{ap}$ which have more or less the correct afterpulsing probability and hence these values are clustered together in the histogram H_2. When the linear region is selected incorrectly, the afterpulsing probability values due to the two methods mismatch and even if the extent of mismatch is less than the threshold, the average of the two values is more or less random. Thus, the corresponding entries in histogram H_2 are scattered. H_2 should have a clear peak which is representative of the afterpulsing probability. Changing the threshold should only change the height of the bins in histogram H_2 but the peak should still be present in the same location. This makes the afterpulsing probability slightly independent of the threshold value. I say "slightly" because you couldn't choose the threshold infinitely small or large. There is a range of thresholds that work and this range is quite large. Also, another benefit to this method is that the way you know a threshold doesn't work is due to the absence of a clear peak in H_2. Similar information was not present in the previous methods, where a bad choice of threshold or linear region just gave a wrong answer without indicating whether the answer was reliable or not.

I will now explain the method in more detail. First, the following bit of code in MATLAB reads the time tags in the data file and stores them in an array.

```
1
2 selected_channel = 1    ;
3 fid = fopen('C:\Users\QPL\Desktop\2017-05-04 01-37-37 ...
      Timetags.dat','r');%%TRIUMF Rad 2 DM Test 16 Det#5
4 ticks = [];
5 i = 1;
6 [channel, count] = fread(fid,1,'*uint8');
7 while count == 1
8     [tick, count] = fread(fid,1,'*uint64');
9     if count == 1
10        if channel == selected_channel
11            ticks(i) = tick;
12            i = i + 1;
13        end
14        [channel, count] = fread(fid,1,'*uint8');
15    end
16 end
```

Next I calculated the time difference between consecutive detector clicks. The following bit of code does this.

```
1
2 TTres = 78.125e-12;
3 array = ticks * TTres;
4 length(array(1,:))/(array(1,end)-array(1,1))
5
6 differenceConsecutiveTTs = array(1,2:end)-array(1,1:end-1);
```

I then created a histogram of these time differences. To create the histogram, I used bins of exponentially (to the base 10) increasing width. Then, I normalize the histogram such that the sum of counts in all bins is equal to 1. Next, I compute the bin count rate corresponding to the normalized bin count by dividing the normalized bin count in each bin by the bin's width. Thus the area under the normalized bin count rate histogram is 1. The following is the corresponding code.

```
1
2 TTres = 78.125e-12;
3 array = ticks * TTres;
4 length(array(1,:))/(array(1,end)-array(1,1))
5
6 differenceConsecutiveTTs = array(1,2:end)-array(1,1:end-1);
```

```
7
8  % differenceConsecutiveTTs = (array(2:end)-array(1:end-1))';
9
10 timeTaggerRes = 78.125e-12;
11 endTime = max(differenceConsecutiveTTs)+timeTaggerRes;
12 minTime = min(differenceConsecutiveTTs)-1e-13;%Make sure the leftmost ...
       bin edge be smaller than the smallest difference
13
14 % binEdges = [minTime:timeTaggerRes:endTime];%linear binning
15 binEdges = logspace(log10(minTime),log10(endTime),1000);%logarithmic ...
       binning
16
17 binCount = zeros(1,numel(binEdges)-1);
18 for i = 1:length(differenceConsecutiveTTs)
19 %     if rem(i,length(differenceConsecutiveTTs)/100) == 0
20 %         sprintf('i=%s',i)
21 %     end
22     if differenceConsecutiveTTs(i)≤binEdges(end)
23         binNum = sum((binEdges-differenceConsecutiveTTs(i))<0); ...
               %Compute the index of bin where histVariable(i) belongs
24         binCount(binNum) = binCount(binNum) + 1;
25     end
26 end
27 %normalize bin Count
28 binCount = binCount/(length(differenceConsecutiveTTs));
29 binCountRate = zeros(1,length(binCount));%should matter for linear x ...
       scale
30 for i = 1:length(binCount)
31     binCountRate(i) = binCount(i) / (binEdges(i+1)-binEdges(i));
32 end
```

Figure 2.6 shows the envelope of a histogram where the logarithm of the normalized bin count rate is plotted against time differences between consecutive detector clicks. The data set used in 2.6 is taken from the radiation tests that will be detailed in chapter 3. In particular, it is from the dark count measurements done on the $DM2$ in test $12a$ for detector #3. As we can see, the left most region of the graph (in the first 1 ms or so) is non-linear and contains both afterpulses (which occur on a short time scale) and thermal counts. Towards the right of this region, is a large linear region which is mostly due to thermal counts only. Also, since we used exponentially growing bins, the normalized bin count rate is calculated by taking more detector clicks into account making this region relatively smooth. However, eventually beyond 8 ms or so the detector clicks become fewer because they obey an exponentially decaying p.d.f. represented by equation 2.19. This is

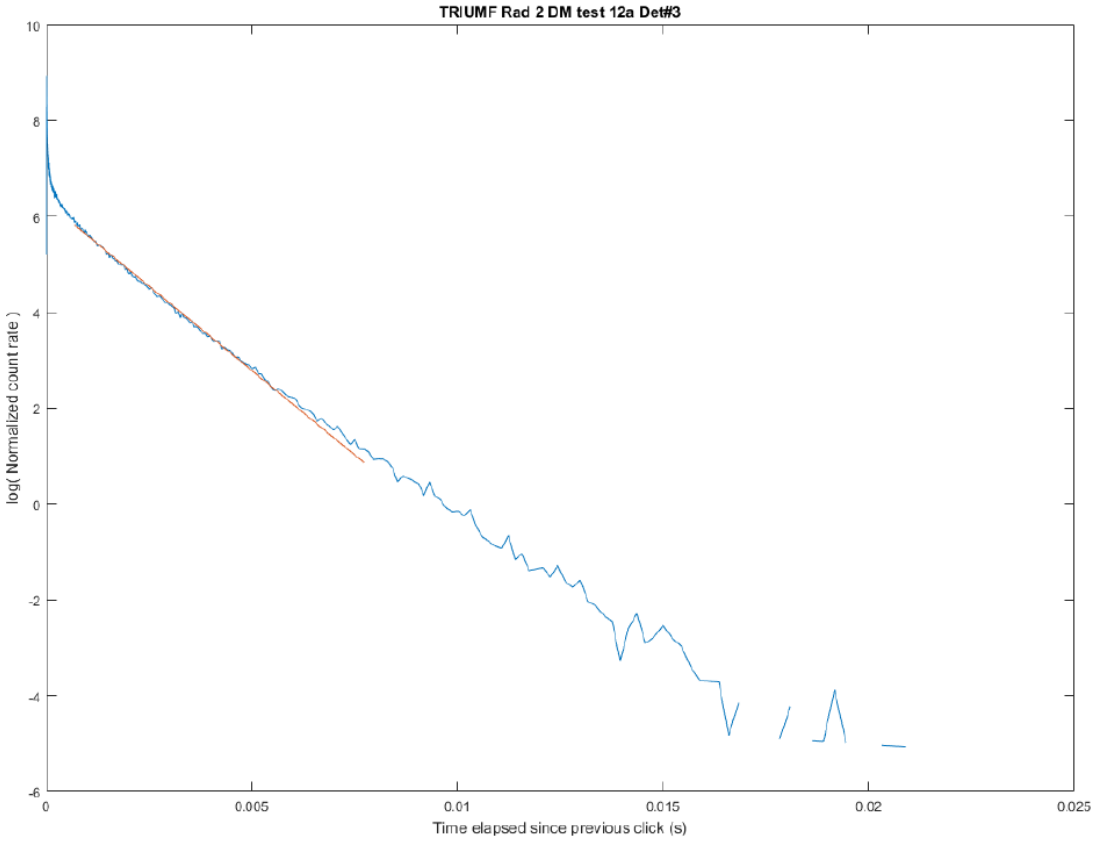


Figure 2.6: Time difference between consecutive detector clicks is calculated and then histogrammed with exponentially increasing bin widths. The histogram is normalized so that the sum of bin counts is unity. Then, the bin count rate is calculated for each bin by dividing the value in each bin by the corresponding bin width such that the area under the histogram is unity. The logarithm of the bin count rate is then plotted with respect to time difference between consecutive clicks in the figure above. The data set is obtained from the dark count measurement in the radiation test of $DM2$ in test $12a$ for detector #3

an issue of lack of statistics due to finite time of measurement. Thus, the very end of the tail of the graph becomes noisy and ideally should not be used during linear curve fitting. By visual inspection, I selected a linear region in this graph to illustrate the process of curve fitting. The data points in the linear region are curve fitted to a straight line using a linear regression function in MATLAB called $fitlm()$. However, not all data points in figure 2.6 are equally reliable. Once the number of normalized bin counts in a given bin goes very low, one needs a much longer measurement time to ensure good statistics. Thus bins with low normalized bin counts tend to behave nosily in the normalized bin count rate histogram. The $fitlm()$ can take in weights for each of the data points. Therefore, I set the weight for each data point in figure 2.6 as the bin's corresponding normalized bin count. This noise actually also shows up if you plot the normalized bin count with respect to time difference between consecutive detector clicks. Figure 2.7 does just this with a logarithmic axes to visually detail the behavior of the graph better. One can see that the when the graph drops below about 2×10^{-5} counts, it starts getting noise, not only in the right end of the graph but also in the left end which is part of the region containing both afterpulses and thermal counts. The reason one get so few counts here is because of the recharge time curve which is of the order of 1 µs resulting in low detection efficiency in this region.

The following bit of code linear curve fits the data points in the deemed linear region.

```
leftBound = 6.74e-4;
rightBound = 7.614e-3;

% Find linear curve fit between leftBound and rightBound
Min = sum((binEdges-leftBound)<0) + 1;
Max = sum((binEdges-rightBound)<0) + 1;
X = binEdges(Min:Max);
Y = log(binCountRate(Min:Max));
weights = binCount(Min:Max); %Curve fit the thermal count region ...
    weighted by the number of data points in each bin because that's ...
    how much (relatively) reliable the value counRate value in a bin is.
% Delete entries
X(Y==Inf|Y==-Inf)=[];
weights(Y==Inf|Y==-Inf)=[];
Y(Y==Inf|Y==-Inf)=[];

linearModel = fitlm(X,Y,'Weights',weights);
```

Now, on to selecting the linear region. As described before, I consider all possible

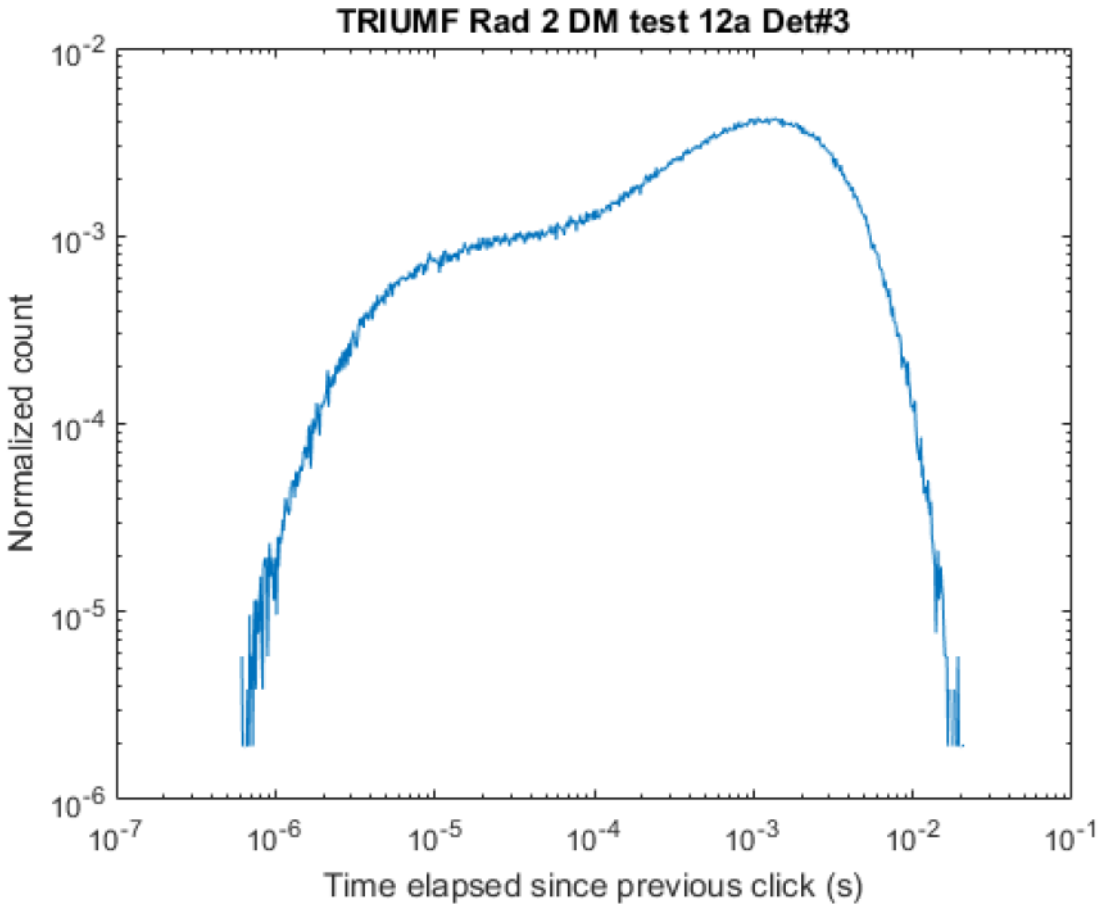


Figure 2.7: Time difference between consecutive detector clicks is calculated and then histogrammed with exponentially increasing bin widths. The histogram is normalized so that the sum of bin counts is unity. The bin counts are then plotted with respect to time difference between consecutive clicks in logarithmic axes in the figure above to visually visually detail the behavior of the graph. So, the normalized counts on the y- axis form the probability mas function of the next click as a function of time elapsed since an arbitrary click. The data set is obtained from the dark count measurement in the radiation test of $DM2$ in test $12a$ for detector #3

continuous regions in the graph containing at least two bins as the linear region. I start with left bound of the linear region as the left most bin. Then in successive iterations, I consider the right bound as each of the bins to wards the right of the left bound. Then I select the left bound as the bin adjacently right of the current left bound and repeat the process over and over again till the bin adjacently left of the rightmost bin is selected as the left bound. This process considers all possible continuous regions of the graph. In each iteration, the linear region is curve fitted using the $fitlm()$ function. This function outputs two parameters which characterize the curve fitted straight line. Given the set of data points that constitute to histogram of normalized bin count rate and the parameters o the curve fitted straight line, one can compute the ares under the histogram but above the straight line as follows in MATLAB. The variable $weightedSum$ gives the value of afterpulsing probability

```
1 weightedSumArray = (binEdges(2:end) - binEdges(1:end-1)) .* ...
      (binCountRate - exp(linearModel.Coefficients.Estimate(2) * ...
      binEdges(1:end-1) + linearModel.Coefficients.Estimate(1)));
2 weightedSum = sum(weightedSumArray);
```

Next, the thermal count rate is found from the slope of the linear region. This is because the linear region is governed by the equation]5.42.

$$d = -(\text{slope of linear region}) \tag{2.23}$$

The overall count rate is calculated using equation 2.22. Then the afterpulsing probability is calculated using 2.21.

Chapter 3

Repeated Irradiation and Annealing

3.1 Introduction

The advantages of using silicon avalanche photodiodes have been explained in section 1. We had a custom made Si APD from Excelitas Technologies Corp. A brand new detector can have thermal count rate under 1 Hz. However, in space operation, the detectors that are placed on the satellite have the disadvantage of being bombarded by protons primarily from the Sun [25]. Protons incident on the detector's substrate will dissipate some of their energy in the substrate. This causes defects in the substrate. The thermal count rate increases as a result of these defects. This poses a problem because there is an upper limit to the signal photon rate. The thermal counts appear as noise when trying to determine photon coincidences which is a part of the QKD protocol. Higher thermal count rate decreases the signal to noise ratio, thus increasing QBER. If the QBER exceeds a certain threshold value, the resulting key becomes unreliable in the sense that the two copies of the key that are held by Alice and Bob tend to have more bit mismatches. We aim for the satellite to have a two year life expectancy. During this period, the APD's are constantly exposed to protons and the thermal count rate is expected to keep increasing. It is desirable to decrease the thermal count rate of the APD's. Thermal annealing is a process of heating the substrate of the APD to a high temperature for a period of time. It was previously shown that thermal annealing was effective in decreasing thermal counts[1]. But this study performed the annealing only once on a irradiated APD However, it was know if and how effective thermal annealing would be when used repeatedly on the same ADP in separate instances. The purpose of the following experiments was to explore this aspect.

We irradiated two DM's with protons at the Tri-University Meson Facility (TRIUMF) near Vancouver in incremental doses up to a cumulative dosage equivalent to two years

in low earth orbit (LEO). Each DM had about five different APD's and housed all the supporting electronics. After radiation dose, we characterized the APD's of each DM for detector parameters much like the experiments in section 2. One of the DM's was thermally annealed after each radiation dose while the other DM was annealed only when the post irradiation thermal count rate exceeded a threshold value. These two different annealing strategies were employed to see if there was a distinct advantage on one over the other. After each annealing phase, the DM's were characterized again. The reason for the characterizations was primarily to monitor dark count rates and see how the radiation and annealing affect it because thermal counts were expected to vary. But other parameters were also monitored just in case they changed.

3.2 Test Setup

The Si APD detectors and it's associated circuitry are enclosed in a compact casing. The unit is called the detector module and is expected to be deployed on-board a satellite. Since thermal counts appear as noise during the photon coincidence matching phase of the QKD protocol, we would like to keep the thermal count rate low. One way to do this is to decrease the temperature which in turn decreases thermal count rate. Thermo electric coolers (TEC) are used to keep the detector's substrate at low temperatures. Section 2 explains how the TEC's are used and also the role of the bracket and radiator. In outer space, the radiator is expected to be at a temperature of about $-70\,^{\circ}\mathrm{C}$. In our experiments, we would like to recreate this thermal condition.

A cooler filled with dry ice was used to to maintain a low temperature. Dry ice has a sublimation temperature of about $-78.5\,^{\circ}\mathrm{C}$. A mechanism was needed to transfer dissipate out the heat from the DM into the dry ice. The DM is a compact unit with small surface area. This doesn't allow the radiate temperature to drop down to $-70\,^{\circ}\mathrm{C}$. A cooling finger was designed and fabricated in the machine workshop at the Department of Physics and Astronomy at the University of Waterloo. It consists of a flat rectangular plate at the bottom and a relatively smaller flat rectangular plate at the top. Each of the plates have a hole drilled at the center. Each end of a cylindrical rod is pressure fitted into the hole of one the plates. To do this, cylindrical rod was machined to a radius slightly greater than the radius of the the hole in the plate. Then, the end of the cylindrical rod was immersed in liquid nitrogen for at least an hour. The metal of the cylindrical rod has a positive coefficient of thermal expansion, i.e., its dimensions shrink when cooled. Thus the radius of the rod at room temperature was such that upon cooling with liquid nitrogen, its radius became less than the radius of the hole of the plate. Thus, the rod could be

fitted into the hole after the cooling process. Once fitted, the rod-plate combination was left at room temperature so that the rod heated up to room temperature. In this process, the rod's radius increases. Since its radius was machined to be greater the hole's radius, the rod exerts a large pressure on the inner surface of the hole. This achieves a thermally conductive contact between the rod and the plate. The procedure is repeated at the other end of the rod. Holes are drilled into smaller and the DM is mounted onto this plate by means of screws - actually the holes are drilled prior to the pressure fitting but I thought it was easier to explain it this way. The larger plate is kept in contact with the dry ice to cool the cold finger and consequently the DM. The bracket TEC which interfaces the bracket and the radiator was then able to cool the bracket down to $-70\,^{\circ}\mathrm{C}$. The detector TEC which interfaces each APD with the bracket was then able to cool the detector further down to $-80\,^{\circ}\mathrm{C}$. The TEC's were used in closed loop control to stabilize the detector temperature during characterization.

An initial room temperature detector characterization was performed at TRIUMF where the external surface of the DM was maintained at room temperature. Using the bracket TEC and detector TEC's, the detector temperature was maintained at $-20\,^{\circ}\mathrm{C}$. Then the DM was placed in the cooler with dry ice to decrease the radiator temperature. Then with the help of the TEC's, the detector temperature was maintained at $-80\,^{\circ}\mathrm{C}$ and a second characterization was performed. The aim of these two characterizations was to have baseline (especially the cold temperature one) to to compare and contrast how the detector parameters change with the upcoming radiation and annealing stages. Also, since similar tests were carried out at the laboratory at the Institute for Quantum Computing, it gives an indication if there was any damage during the shipping process from IQC to TRIUMF.

Much of the remaining setup to support the characterization is very similar to the lab characterization process at IQC as described in section 2.1. The DPCU supplies regulated bias voltage to the 5 APD's in each module. The DPCU also controls the bracket TEC and detector TEC's. A proprietary software on a computer interfaces with the DPCU and where one can set values of bias voltages and target temperatures of the TEC's. A portable laser from the manufacturer PicoQuant was used to supply signal photons to the DM via optical fibers and a fiber bridge. An optical power meter is placed at the fiber splitter and calibrated to measure the optical power of the photons hitting the detectors. The output electric pulse from each APD in the DM can be read out via SMA cables on an oscilloscope or can be time tagged on a time tagger like in the IQC lab characterization. The detector characterization is carried as explained in chapter 2. In particular, the breakdown voltage (explained in section 2.2.2), output pulse amplitude and FWHM (explained in

section 2.2.3), recharge time (explained in section 2.2.4), thermal counts (explained in section 2.2.5), detection efficiency (explained in section 2.2.6), timing jitter (explained in section 2.2.7), saturation value (explained in section 2.2.8) and afterpulsing probability (explained in section 2.2.9) were determined. The characterization was performed after each irradiation phase and each annealing phase. All the aforementioned parameters except recharge time and saturation were determined during each characterization. This was because these two measurements took time to execute and were not expected to be affected by either proton radiation or thermal annealing. They were only performed at the lab in IQC before leaving and after returning from the TRIUMF tests, during the baseline room temperature and cold tests at TRIUMF and at the end of the two year LEO equivalent cumulative radiation dosage at TRIUMF.

The two year equivalent cumulative proton fluence in LEO is estimated to be around $4 \times 10^9\,\mathrm{p/cm^2}$. This was initially based on a detailed modeling and simulation done by a partner company COM DEV (now acquired by Honeywell). More recently, Jin in his thesis [6] derived this again. With a total spherical aluminum shielding of 30mm, he calculates the total displacement damage dose (DDD) after 2 years due to radiation to be $1.27 \times 10^6\,\mathrm{MeVg^{-1}}$. Elena Anisimova also does a similar calculation in her PhD thesis [1] and arrives at the result that the two year equivalent dosage in LEO for 100 MeV protons is $4 \times 10^9\,\mathrm{p/cm^2}$. At TRIUMF, a beam of protons was aimed at the DM which was placed inside the cooler. Some of the protons from the beam incident on the cooler were expected to be absorbed by the cooler itself. The remaining were incident on the DM. So, we have some extra shielding in our experiments that are not expected to be present during operation in a space environment on board a satellite. Two masters students at IQC did some calculations regarding the extra shielding factors. One of them, Jin Gyu Lim, has detailed this in his master's thesis [6]. The primary outcome was that we needed to increase the exposure of the DM plus cooler assembly to a proton fluence slightly larger than $4 \times 10^9\,\mathrm{p/cm^2}$ as measured outside the cooler, so that the APD's physically experience a cumulative fluence of $4 \times 10^9\,\mathrm{p/cm^2}$. In specific, the shielding ratio was calculated to be around 1.052. This means that the DM plus cooler assembly had to be exposed to $1.052 \times 4 \times 10^9 = 4.208 \times 10^9\,\mathrm{p/cm^2}$ in order for the APD's to experience $4 \times 10^9\,\mathrm{p/cm^2}$ of proton fluence. Henceforth all the mentioned proton fluences will be with respect to what the APD's physically experience. If one wants to calculate the proton fluence outside the cooler, one should multiply the mentioned value by 1.052.

One of the DM's, labeled DM1, was irradiated with protons in incremental doses of fluence $6.6 \times 10^8\,\mathrm{p/cm^2}$ (corresponding to 4 month equivalent fluence in LEO) up to a cumulative proton fluence of $4.03 \times 10^9\,\mathrm{p/cm^2}$ (2 year equivalent fluence in LEO). DM1 was

annealed after each incremental dose of proton radiation. Another DM, labeled DM2, was irradiated with protons in incremental doses of $3.3 \times 10^8\,\mathrm{p/cm^2}$ (2 month equivalent fluence in LEO) or $6.6 \times 10^8\,\mathrm{p/cm^2}$(4 month equivalent fluence in LEO) up to a cumulative proton fluence of $4.03 \times 10^9\,\mathrm{p/cm^2}$ (2 year equivalent fluence in LEO). However, DM2 was further irradiated beyond for cumulative radiation doses beyond the 2 year expected lifetime of the satellite to see the effects of irradiation and annealing on longer operation lifetimes. In particular, DM2 was further irradiated to a cumulative proton fluence of $2 \times 10^{10}\,\mathrm{p/cm^2}$ (10 year equivalent fluence in LEO). Beyond the 2 year lifetime, larger incremental doses were used - between $2 \times 10^9\,\mathrm{p/cm^2}$ (1 year equivalent fluence in LEO) and $6 \times 10^9\,\mathrm{p/cm^2}$ (3 year equivalent fluence in LEO). DM2 was however only annealed when the dark count rate exceeded a predetermined threshold of 2 kHz. This threshold was mostly fixed by link analysis calculations [10]. The reason for using the smaller dose of $3.3 \times 10^8\,\mathrm{p/cm^2}$ for DM2 was to give us better resolution when the dark count rate approached the 2 kHz threshold, so that we don't overshoot it by a large amount. This was done because it reflects space operation where the proton radiation occur on a longer time scale than provided by the proton beam at TRIUMF; so in space one would be able to more or less detect exactly when the dark count rate threshold is exceeded. A third DM was used as a control and did not get irradiated or annealed. However, it underwent the same handling and storage processes as DM1 and DM2 to account for factors extraneous to the experimental setup affecting detector characteristics. Each annealing phase comprised of heating the APD's (using the TEC's) to 80 °C for a period of 1 hour. The DM's were characterized after each radiation and annealing phase. During characterization, the APD's were set to a temperature of −80 °C.

3.3 Results and Discussions

Breakdown voltage, output pulse amplitude, output pulse FWHM, recharge time, detection effciency, detector timing jitter and saturation values were largely unaffected by either radiation or annealing. The dark counts did change usually increasing with irradiation and decreasing with annealing. The afterpulsing probability tended to increase with irradiation but annealing had no effect on it.

3.3.1 Breakdown Voltage

The breakdown voltage was measured during each characterization because it is temperature dependent. This ensured that a known over-voltage is applied since other detector

parameters are significantly affected by the over-voltage - if the same bias voltage was used for each characterization instance and the breakdown voltage changed, this would effectively change the over-voltage without our notice.

Figures 3.1 and 3.2 show the measured breakdown voltages after a radiation exposure (before annealing), marked by an x, and after annealing, marked by an o. The $\triangle$ marker at zero cumulative radiation fluence shows the breakdown voltage measured during the baseline test before the experiments began. All characterizations were performed with detector temperature set at $-80\,^{\circ}\mathrm{C}$. The breakdown voltage shows no correlation with respect to cumulative proton radiation fluence. However, in the vast majority of instances, breakdown voltage measured after annealing is higher than the corresponding value before annealing. But in a few cases like test $5a$ Detector 3 on DM1 and test $15a$ on Detector 1 on DM2, the post-annealed value was lower than the pre-annealed value. The amount by which the post annealed breakdown voltage on average was higher compared to its pre-annealed value is more or less within the bounds of statistical uncertainty. However, for a purely statistical case, on average, one would expect that half the instances would show that the pre-annealed value was higher (or lower) than the post-annealed value. Since, the majority of cases have the post-annealed breakdown voltage as the higher one, it might be some indication that annealing may have increased the breakdown voltage, although more tests would be needed to ascertain this. The breakdown voltage has a positive correlation with temperature. Although the measured ADP temperatures do not explain the effect of higher post annealed breakdown voltages, it may be possible that the physical substrate of the APD may be at a slightly higher temperature because of the high temperature annealing phase that just preceded these measurements. One observation to support this might be that if a detector in a given test showed a higher post annealed breakdown voltage, the probability that the other detectors in the same test also showed a higher post annealed breakdown voltage was greater than if that first detector showed a lower post annealed breakdown voltage. Since the main parameter that changes the breakdown voltage is the APD substrate's temperature, it is possible that the annealing may have slightly increased this temperature. Nonetheless, more tests are need to ascertain this.

3.3.2 Output Pulse Shape

The over-voltage on each APD was set to 20 V above its respective breakdown voltage. The output pulse was displayed on an oscilloscope. The pulse amplitude was visually measured off the oscilloscope screen. Figure 3.3 shows the output pulse amplitudes for each of the 5 detectors in DM1 for the entirety of the radiation experiments. Figure 3.4 shows the

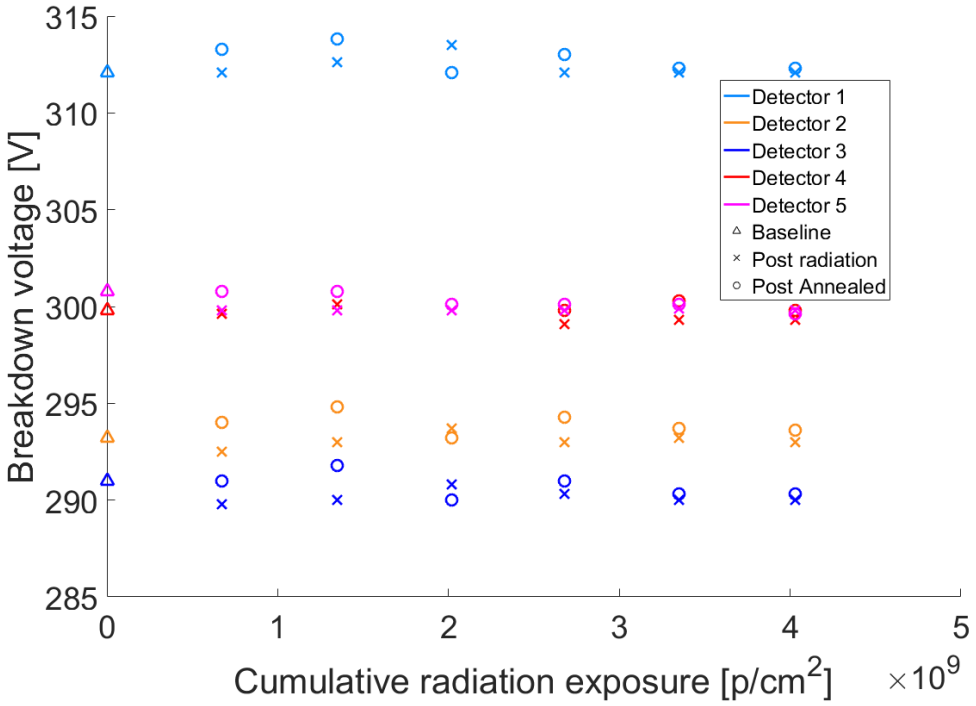


Figure 3.1: The breakdown voltage is plotted with respect to cumulative proton radiation fluence for DM1. The breakdown voltage at the baseline test before any irradiation began is plotted with t $\triangle$. After every incremental radiation exposure, the breakdown voltage was measured and this is plotted with an $\times$ marker. Also, after every annealing phase, the breakdown voltage was measured and this is plotted with an $\circ$ marker. Detector temperature of $-80\,^{\circ}\mathrm{C}$ was used for all tests. The post annealed breakdown voltage was generally slightly higher for all tests except test 5a

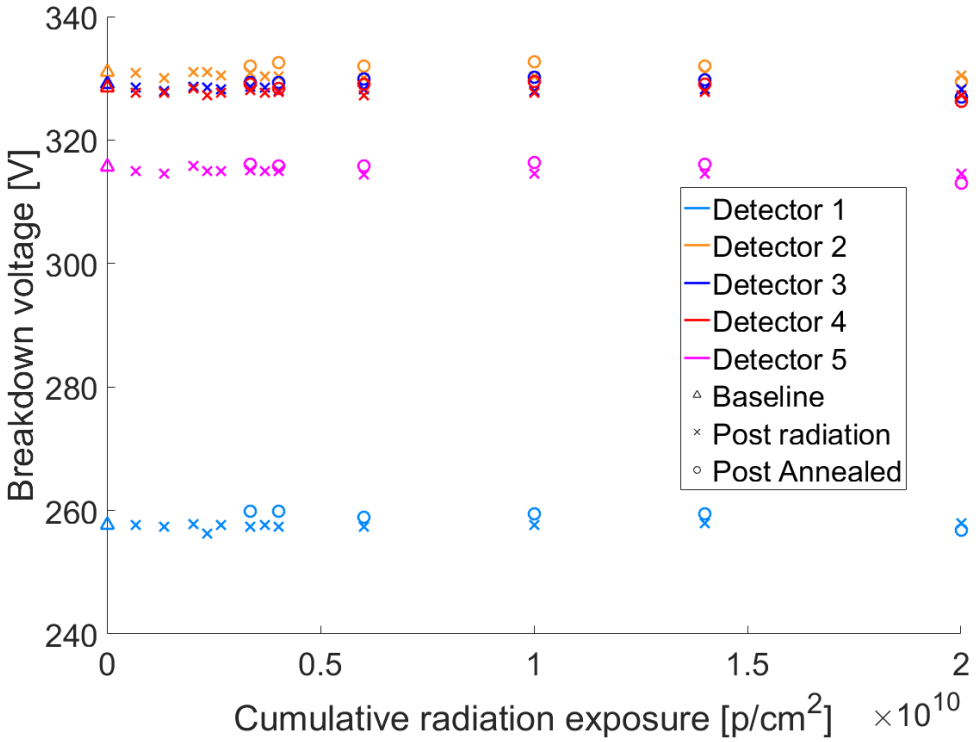


Figure 3.2: The breakdown voltage is plotted with respect to cumulative proton radiation fluence for DM2. The breakdown voltage at the baseline test before any irradiation began is plotted with t $\triangle$. After every incremental radiation exposure, the breakdown voltage was measured and this is plotted with an $\times$ marker. Also, after every annealing phase, the breakdown voltage was measured and this is plotted with an $\circ$ marker. Detector temperature of $-80\,^{\circ}\mathrm{C}$ was used for all tests. The post annealed breakdown voltage was generally slightly higher for all tests except test 15a

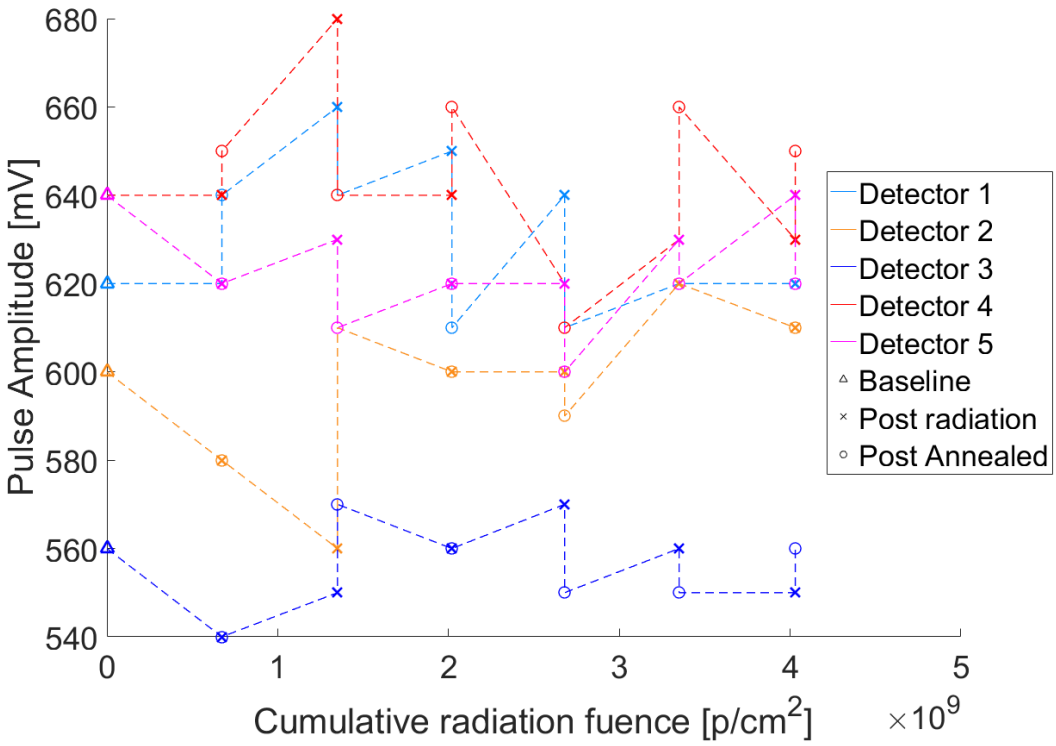


Figure 3.3: Output pulse amplitude of the 5 detectors in DM1 are plotted against cumulative proton radiation fluence for all characterization tests. The pulse amplitude at the baseline test before any irradiation began is plotted with a $\triangle$. After every incremental radiation exposure, the pulse amplitude was measured and this is plotted with an $\times$ marker. Also, after every annealing phase, the pulse amplitude was measured and this is plotted with an $\circ$ marker. All characterization tests were performed at a APD temperature of $-80\,^{\circ}\mathrm{C}$

corresponding values for DM2. Most detectors on both DM's have output pulse amplitudes in the range from 550 mV to 650 mV except for detector 1 on DM2 which has output pulse amplitudes in the range 700 mV to 750 mV. All characterization tests in these two figures were performed at a APD temperature of $-80\,^{\circ}\mathrm{C}$.

There was no observed correlation between the output pulse amplitudes of the APD's and the cumulative radiation exposure. Also, annealing did not seem to have any definitive effect on the pulse amplitudes. The maximum variation in the measured pulse amplitude on a given detector was 70 mV. This variation is attributed to measurement uncertainty due to statistical uncertainties arising from the nature of the experimental setup as well as the act of visually reading off the amplitude from the oscilloscope.

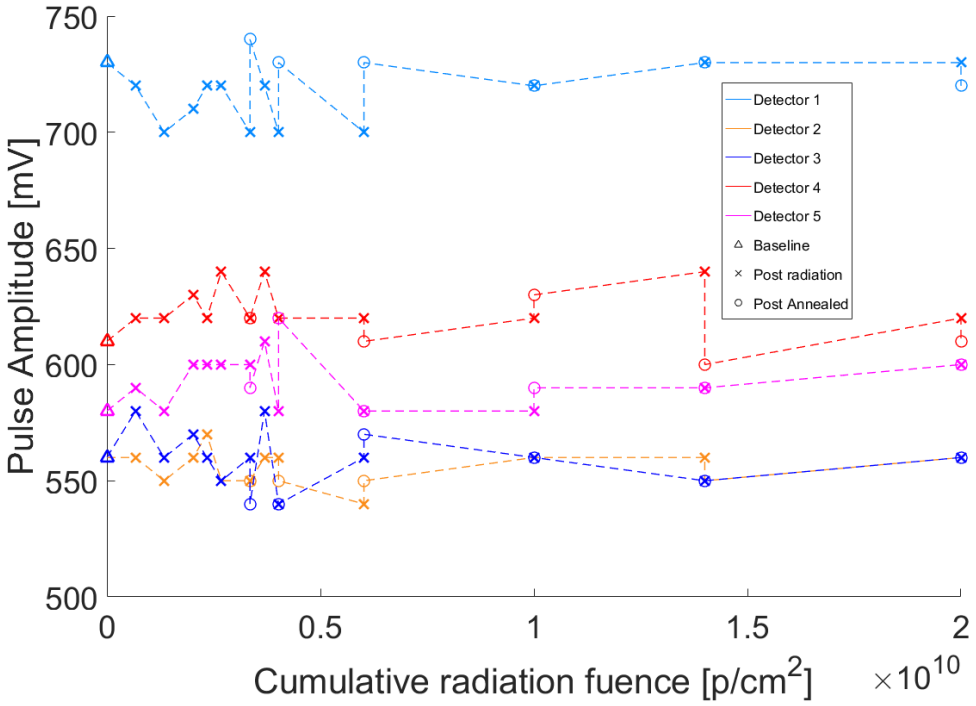


Figure 3.4: Output pulse amplitude of the 5 detectors in DM2 are plotted against cumulative proton radiation fluence for all characterization tests. The pulse amplitude at the baseline test before any irradiation began is plotted with a $\triangle$. After every incremental radiation exposure, the pulse amplitude was measured and this is plotted with an $\times$ marker. Also, after every annealing phase, the pulse amplitude was measured and this is plotted with an $\circ$ marker. All characterization tests were performed at a APD temperature of $-80\,^{\circ}\mathrm{C}$

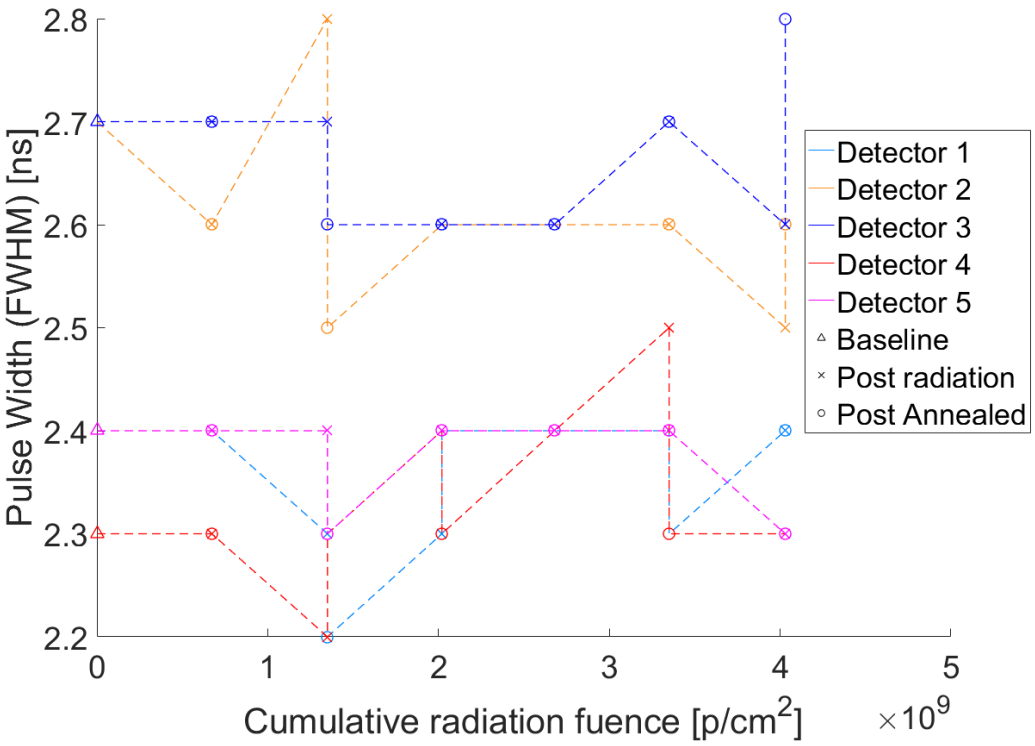


Figure 3.5: Output pulse FWHM of the 5 detectors in DM1 are plotted against cumulative proton radiation fluence for all characterization tests. The pulse FWHM at the baseline test before any irradiation began is plotted with a $\triangle$. After every incremental radiation exposure, the pulse FWHM was measured and this is plotted with an $\times$ marker. Also, after every annealing phase, the pulse FWHM was measured and this is plotted with an $\circ$ marker. All characterization tests were performed at a APD temperature of $-80\,^{\circ}\mathrm{C}$

The width of the output pulse at a voltage half the amplitude of that pulse is read out visually from the oscilloscope. The oscilloscope has a couple of horizontal and vertical crosshairs to help with the measurement. Figures 3.5 shows the FWHM values of the output pulses of the 5 detectors in DM1 for all characterization tests. Figure 3.6 shows the corresponding values for DM2. Most FWHM values are in the range 2.2 ns to 2.8 ns with a few outliers.

No correlation was found between output pulse FWHM values and the cumulative radiation fluence. Also, annealing did not affect the pulse FWHM. The maximum variation in the measured pulse amplitude on a given detector was under 0.4 ns for most of the detectors except for detectors #2 and #3 on DM2 which had a variation of 0.7 ns and 0.6 ns respectively. This variation can also attributed to measurement uncertainty especially because the measurement of the FWHM value accumulates uncertainty from the amplitude

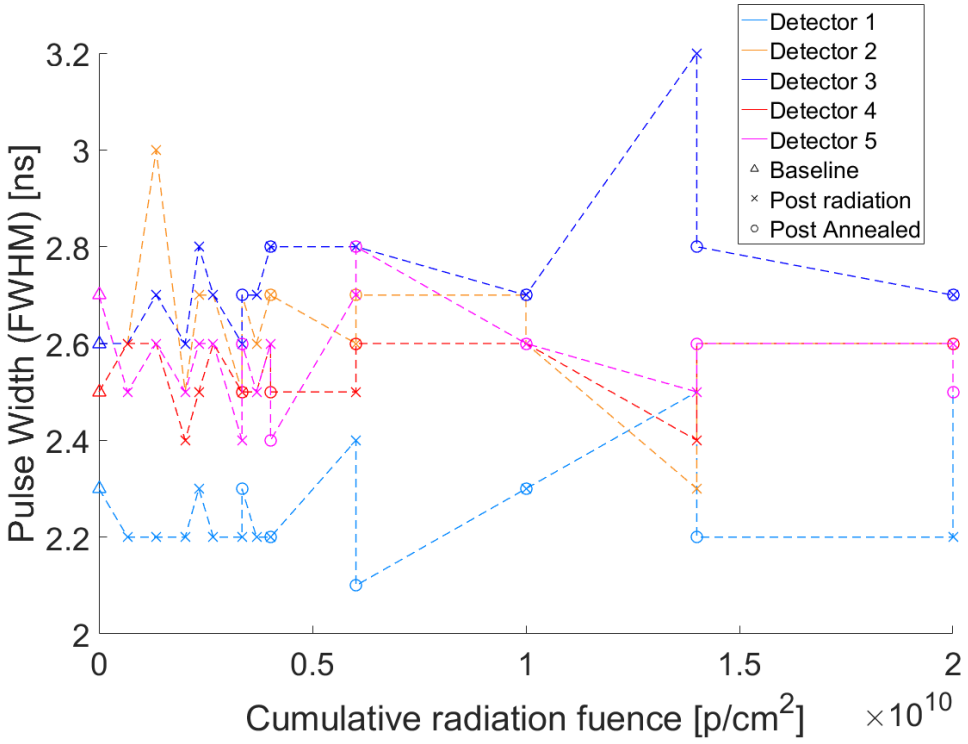


Figure 3.6: Output pulse FWHM of the 5 detectors in DM2 are plotted against cumulative proton radiation fluence for all characterization tests. The pulse FWHM at the baseline test before any irradiation began is plotted with a $\triangle$. After every incremental radiation exposure, the pulse FWHM was measured and this is plotted with an $\times$ marker. Also, after every annealing phase, the pulse FWHM was measured and this is plotted with an $\circ$ marker. All characterization tests were performed at a APD temperature of $-80\,^{\circ}\mathrm{C}$

measurement as well in addition to visually reading off the width.

3.3.3 Recharge Time

The recharge time was measured for both DM's at the lab in IQC (test #00), at TRIUMF (at room temperature and cold - tests #01 and #02) before irradiating the DM's, at the end of the two year LEO equivalent cumulative radiation dose (tests #08 and #08a for DM1 and test #10a for DM2) and finally back at IQC after returning from TRIUMF (test#13 for DM1 and test #17 for DM2). The primary reason the remaining characterization tests did not include recharge time measurements was because that it took substantial time (2 - 3 minutes) to populate the recharge curve with output pulses. Since time was limited at TRIUMF and recharge time as such was not expected to change either with irradiation or annealing, this measurement was only performed during the aforementioned tests.

Figures 3.7 and 3.8 show the recharge time constants as measured from the oscilloscope for DM1 and DM 2 respectively and how they vary with respect to cumulative radiation fluence. Since the recharge time isn't expected to vary with temperature, I used all data points from the aforementioned tests with all the tests having a detector temperature of $-80\,^{\circ}\mathrm{C}$ except one of the baseline characterization tests (test #01) at TRIUMF which had the detector temperature set at $-20\,^{\circ}\mathrm{C}$. The recharge time does not seem to be correlated to cumulative radiation fluence. The maximum variation of recharge times measured on a given detector was within 200 ns for most detectors except for detectors #1 and #3 which had a variation of 300 ns and 250 ns respectively. This uncertainty in measurement is accountable due to the fact the the recharge times were visually determined from the oscilloscope screen. Hence, it was concluded that there was no dependence of recharge time on cumulative radiation fluence. As far as the effect of annealing goes, only DM1 was characterized for recharge time at the end of the two year LEO equivalent cumulative radiation fluence before and after the corresponding annealing phase - tests #08 and #08a. Three of the detectors did not show any change in recharge time whereas two of the detectors showed an increase/decrease of 50 ns after the annealing phase. Although this was only one instance of annealing, from the outcome of the characterization tests, there is no reason to believe annealing might have any effect on recharge time.

3.3.4 Dark Counts

Dark counts are defined as detector clicks which occur when no photon is incident on the APD. This includes thermal counts which are random detector click due ti imperfections in

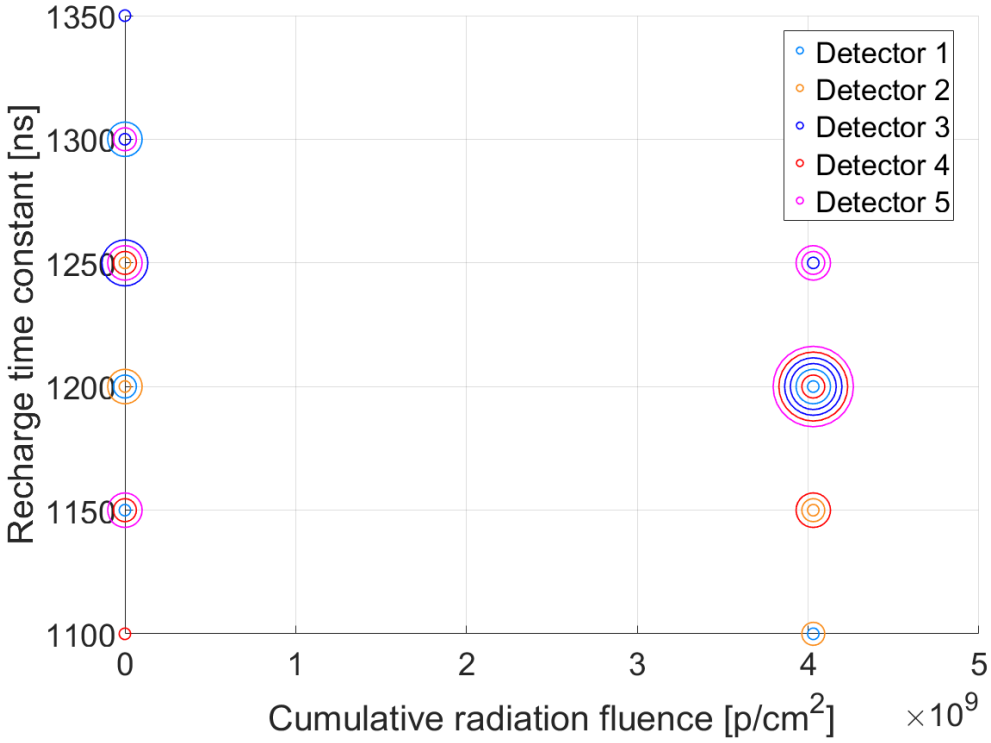


Figure 3.7: Recharge time was visually measured off the oscilloscope and this is a plot of recharge times for DM1 against cumulative radiation fluence. The recharge time of each APD detector was measured at the lab in IQC before going to TRIUMF (test #00), at TRIUMF (test #01 and test #02) before irradiating the DM's, at the end of the two year LEO equivalent cumulative radiation dose (tests #08 and #08) and finally back at IQC after returning from TRIUMF (test#13). All tests were conducted at an APD temperature of $-80\,^{\circ}$C except test #01 which was conducted at an APD temperature of $-20\,^{\circ}$C. The different sizes of concentric circles are merely to illustrate the frequency of occurrence of a given value of recharge time - different sizes of circles at the same point are not be differentiated between, except that they belong to different tests. There was no observed correlation of recharge time with respect to cumulative radiation fluence or annealing.

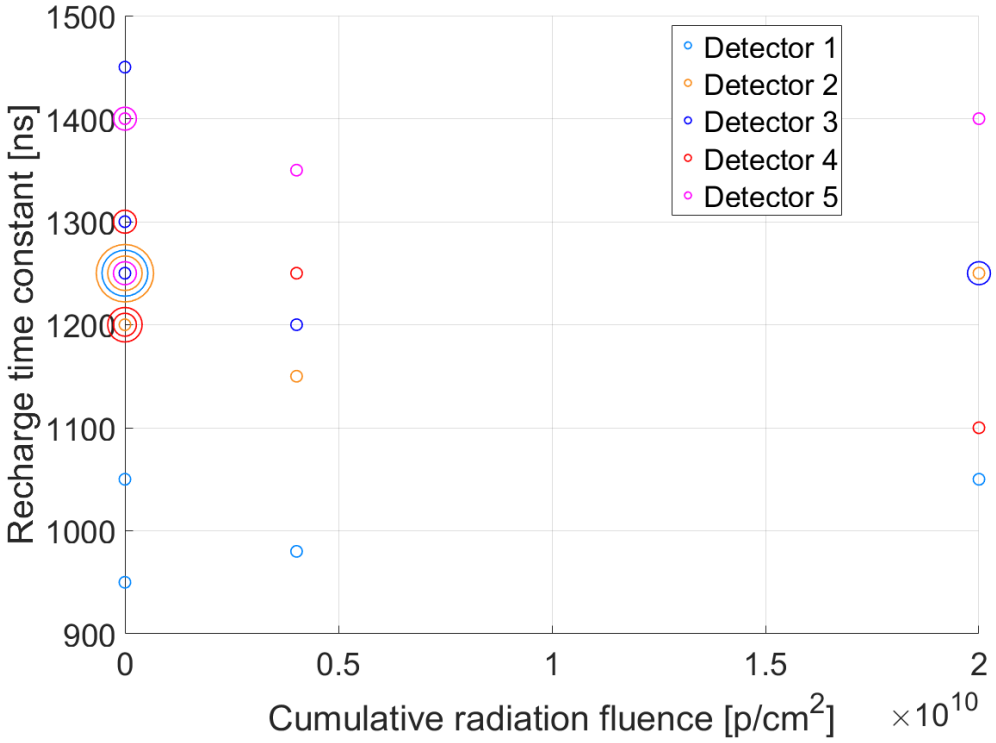


Figure 3.8: Recharge time was visually measured off the oscilloscope and this is a plot of recharge times for DM2 against cumulative radiation fluence. The recharge time was measured at the lab in IQC before going to TRIUMF (test #00), at TRIUMF (tests #01 and #02) before irradiating the DM's, at the end of the two year LEO equivalent cumulative radiation dose (test #10a) and finally back at IQC after returning from TRIUMF (test #17). All tests were conducted at an APD temperature of $-80\,^{\circ}\mathrm{C}$ except test #01 which was conducted at an APD temperature of $-20\,^{\circ}\mathrm{C}$. The different sizes of concentric circles are merely to illustrate the frequency of occurrence of a given value of recharge time - different sizes of circles at the same point are not be differentiated between, except that they belong to different tests. There was no observed correlation of recharge time with respect to cumulative radiation fluence or annealing.

the substrate of the APD. Dark counts also include the afterpulses of thermal counts. The results mentioned in this subsection are inferences about dark counts and do not isolate the thermal counts from their chain of afterpulses. In the absence of ambient light (laser off and covering the optical fibers with a black cloth), the output detector clicks were time tagged after each incremental radiation dose and each annealing phase. DM1 was annealed after every exposure to radiation whereas DM2 was annealed only when the post irradiation dark count rate exceed 2 kHz.

Figure 3.9 shows how the dark count rate varies with cumulative radiation fluence as well as annealing, with detector temperatures at $-80\,^{\circ}\mathrm{C}$. It is seen that in general every incremental dose of radiation increases the dark count rate. After the first incremental proton radiation dose of $6.72 \times 10^8\,\mathrm{p/cm^2}$, seven of the ten APD's registered a dark count rate exceeding 500 Hz, with three of the detectors exceeding 1 kHz. This surpassed the expected increase in dark count rate which was projected to be well under 200 Hz. Only three of the ten detectors did the first proton irradiation cause the highest increase in dark count rate per unit increase in cumulative radiation fluence. There were however anomalies where post radiation dark count rates were lower than the corresponding pre-radiation dark count rates. This happened only on DM2 for Detector #1 on test #04, detectors #2, #3 and #4 on test #06, detectors #1 and #3 on test #7, detectors #1 and #2 on test #08 and finally on detector #1 on test #10. The anomaly could possibly be due to statistical uncertainty. However, since the dark counts were time tagged and measured over a course of 600 s, one must still question if there is an underlying cause for the anomaly. The cause, if any, is unknown at the moment.

Figures 3.11 and 3.12 show the change in dark count rate (DCR) per unit proton fluence for DM1 and DM2 respectively. This is calculated by looking at how much the dark count rate changes before and after an instance of incremental proton irradiation and then dividing this change by the value of incremental proton fluence. As such there is no correlation between the DCR change per unit fluence. However, one must note that there is a tighter grouping of DCR change per unit fluence at higher cumulative fluence for DM2. My opinion is that this is due to the fact that large incremental radiation doses were used in these tests and that the grouping as such isn't related to the cumulative fluence itself. This might suggest that the tighter grouping is a result of better averaging over larger incremental fluences in turn suggesting that the scattering of the points towards the left of the graph are due to statistical uncertainty which get averaged out when large incremental fluences are evaluated. This might also have implications for the aforementioned anomaly of decrease of dark count rates with proton fluence being of statistical nature.

Annealing decreased the dark count rate in general. Figures 3.13 and 3.14 show the dark

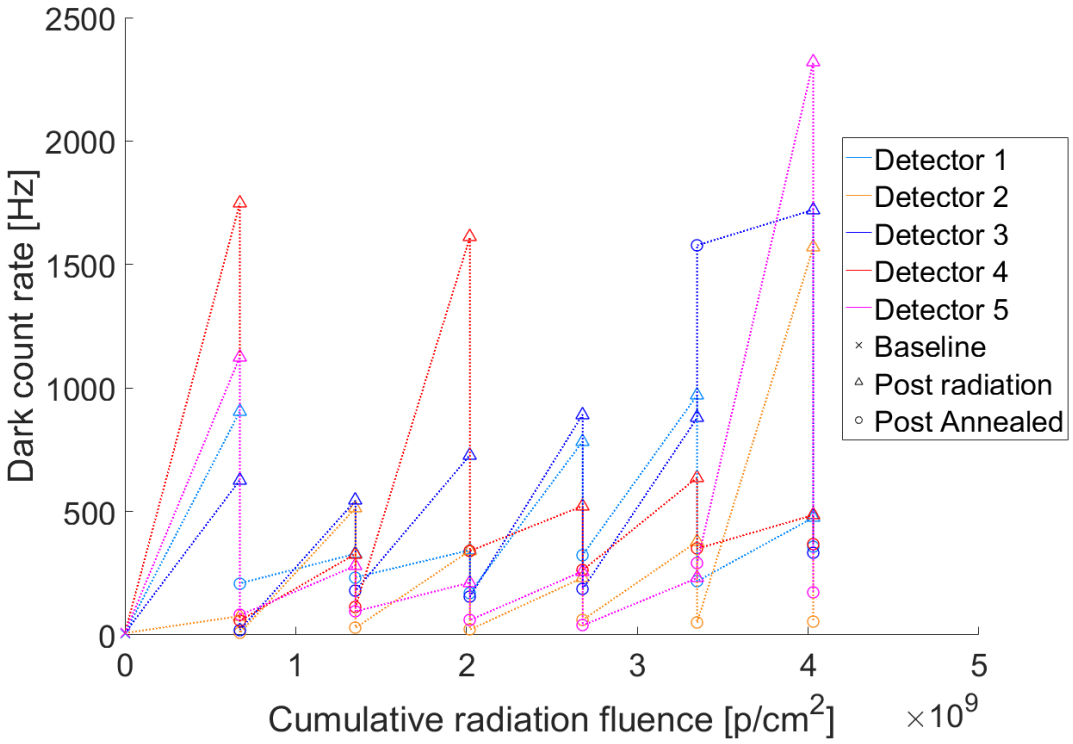


Figure 3.9: The dark count rate characterized after each radiation dose and annealing phase is plotted against the cumulative radiation fluence for DM1. All tests were conducted at an APD temperature of $-80\,^{\circ}\mathrm{C}$. A general trend is that radiation increases dark count rate and annealing decreases dark count rate. There are some anomalies however.

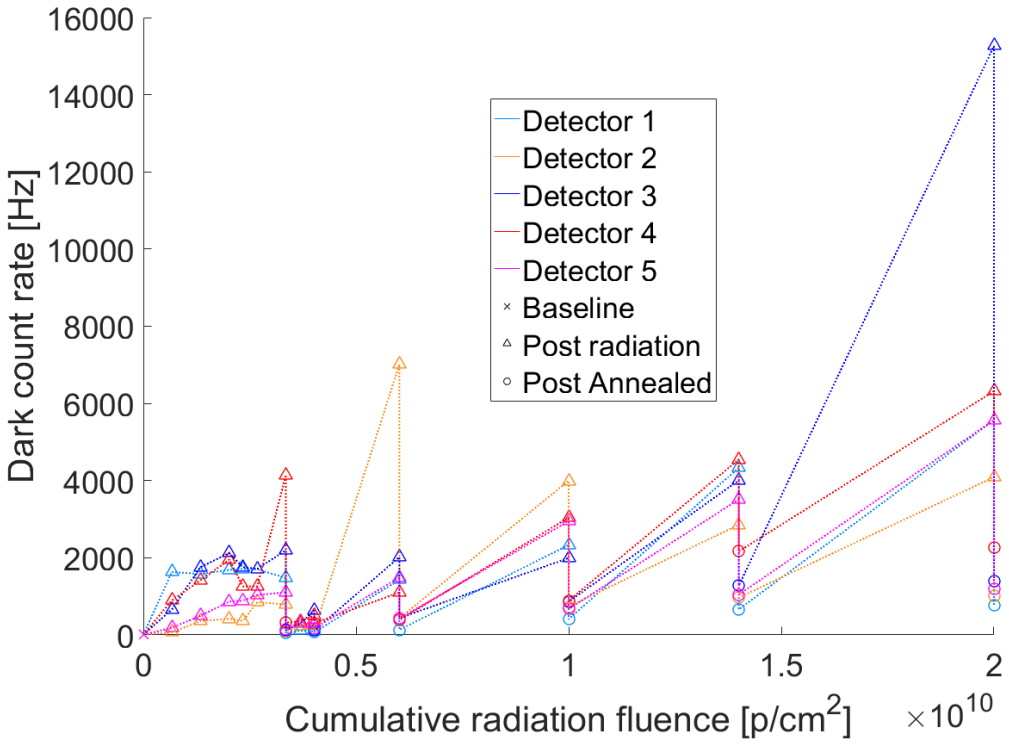


Figure 3.10: The dark count rate characterized after each radiation dose and annealing phase is plotted against the cumulative radiation fluence for DM2. All tests were conducted at an APD temperature of $-80\,^{\circ}\mathrm{C}$. A general trend is that radiation increases dark count rate and annealing decreases dark count rate. There are some anomalies however.

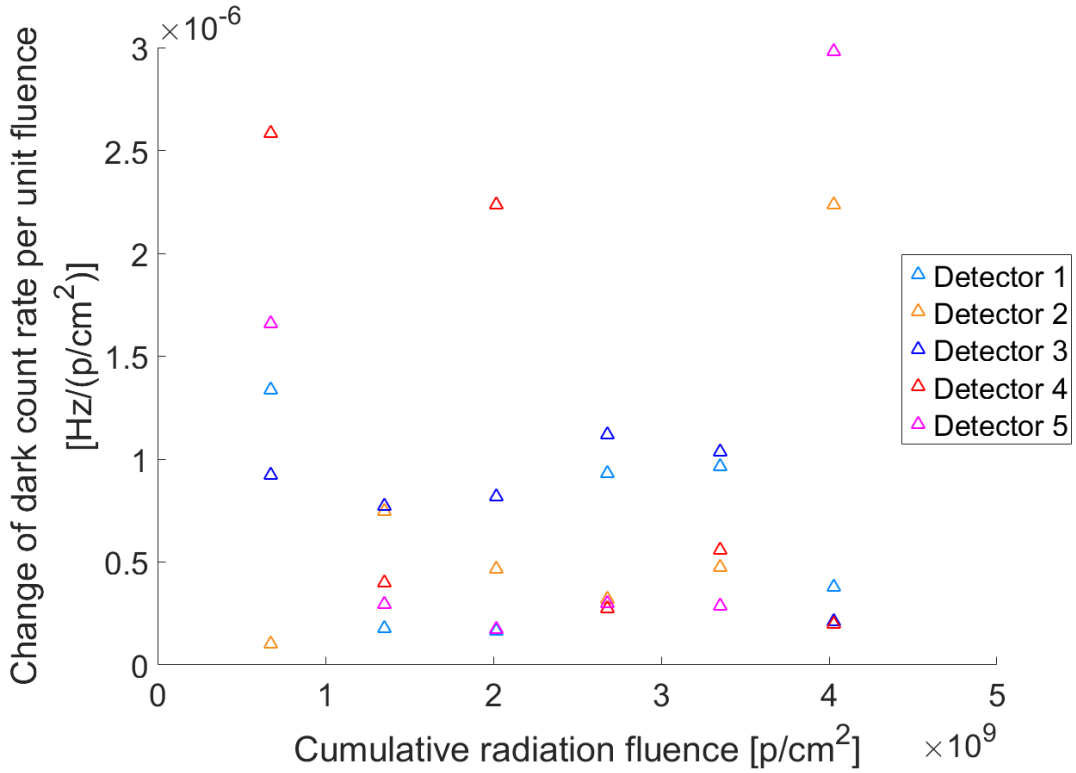


Figure 3.11: The rate of change of dark count rate with respect to incremental proton fluence is plotted against the cumulative proton fluence. For any given instance of incremental irradiation phase, we calculate the ratio of the change in dark count rate from before to after irradiation, to the quantity of incremental radiation experienced by the APD. This ratio is plotted against the cumulative radiation value at the end of a given incremental radiation phase. This gives an idea of the contribution of proton radiation towards increase in dark count rate. All tests were conducted at an APD temperature of $-80\,^{\circ}\mathrm{C}$.

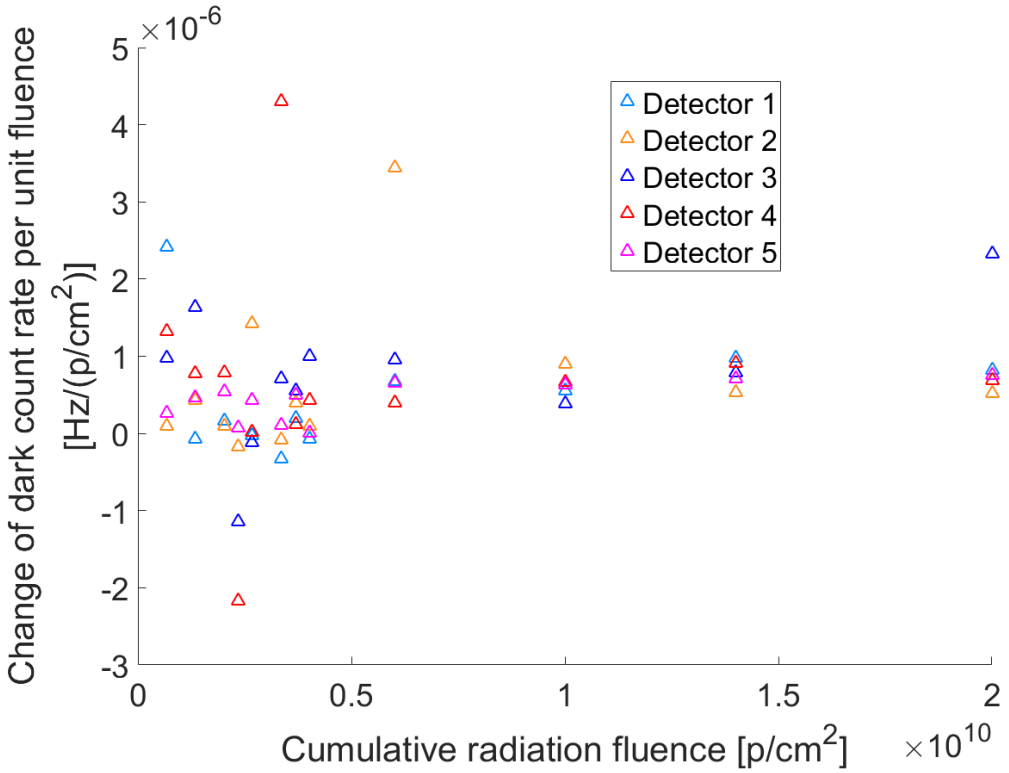


Figure 3.12: The rate of change of dark count rate with respect to incremental proton fluence is plotted against the cumulative proton fluence. For any given instance of incremental irradiation phase, we calculate the ratio of the change in dark count rate from before to after irradiation, to the quantity of incremental radiation experienced by the APD. This ratio is plotted against the cumulative radiation value at the end of a given incremental radiation phase. This gives an idea of the contribution of proton radiation towards increase in dark count rate. All tests were conducted at an APD temperature of $-80\,^{\circ}\mathrm{C}$.

count rate measured after each annealing process. This is important from an operational point of view because we would like to keep the dark count rate low to lower the signal to noise ratio and annealing is a proposed method to lower dark count rate and mitigate the adverse effects of proton radiation. Annealing manages to keep the dark count rate below 500 Hz up to 6×10^9 p/cm^2 cumulative radiation fluence (3 year LEO equivalent). This is under the threshold dark count rate suggested by the link analysis [28]. The only exception here is detector #3 on DM2 which registered a post-annealed dark count rate of 1577 Hz at the 3.35 p/cm^2 cumulative radiation fluence mark. It is important to note that the post-annealed dark count rate tends to increase slightly as the cumulative radiation fluence increases. This shows annealing is increasingly unable to mitigate the effects of proton radiation. This will be a factor in limiting the lifetime of a QKD satellite. Although annealing generally decreased dark count rates, detectors #3 and #5 on DM1 in test #7a at 3.35 p/cm^2 mark registered higher dark count rates annealing the DM. Since there were only two instances over the entire course of the experiments, it is difficult to predict why this anomaly occurred.

When the dark count rate after radiation is high, annealing can lower it. One way to quantify the effectiveness of annealing is the look at the difference between the pre-annealed and post-annealed dark count rates. However, dark counts occur due to the presence of impurities in the APD substrate and have a positive correlation with the density of impurities. Annealing "fixes" these impurities, hence decreasing dark count rates. It does this by increasing the temperature of the APD substrate. In doing so, each site of impurity is maintained at a higher temperature. Assuming that each impurity is identical to every other and whether or not an impurity gets fixed is independent of what happens to any other impurity, the ratio of the the dark count rate before annealing to that after annealing is a better measure for the effectiveness of annealing. To gain better insight into the effectiveness of annealing using this reduction factor of annealing as a function of the cumulative radiation fluence is plotted against cumulative radiation fluence. Figures 3.15 and 3.16 plots this data for DM1 and DM2 respectively. For eight out of the ten detectors, annealing reduction factor was the highest the first time it was used. This shows that annealing was the effective the first time around. However, in subsequent instances of annealing, we do not see any correlation in the annealing reduction factor with respect to cumulative radiation fluence.

Since annealing "fixes" impurities in the APD substrate and dark count rate is strongly related to the density of these impurities, it might be more beneficial to see how the annealing reduction factor changes when the dark count rate before annealing varies. Figures 3.17 and 3.18 plot the annealing dark reduction factor with respect to the dark count rate mea-

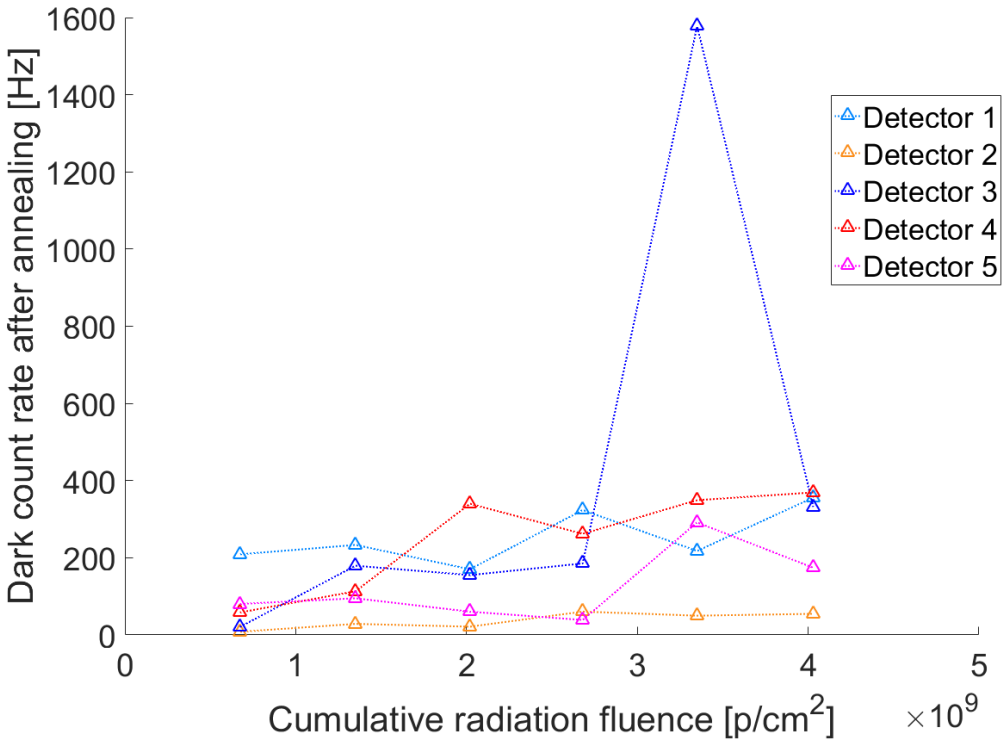


Figure 3.13: The dark count rate measured after annealing is plotted against cumulative radiation fluence for DM1. The curve doesn't start at zero radiation because DM1 was not annealed at baseline zero radiation. All tests were conducted at an APD temperature of $-80\,^{\circ}\mathrm{C}$.

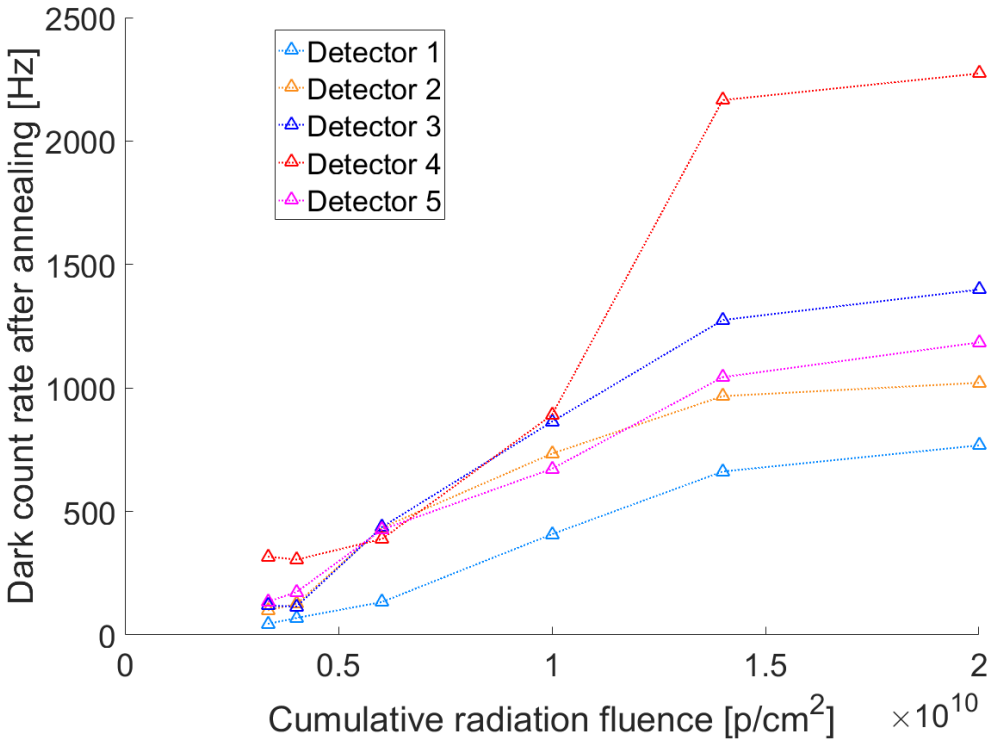


Figure 3.14: The dark count rate measured after annealing is plotted against cumulative radiation fluence for DM2. The curve doesn't start at zero radiation because DM2 was not annealed at baseline zero radiation. All tests were conducted at an APD temperature of $-80\,^{\circ}\mathrm{C}$.

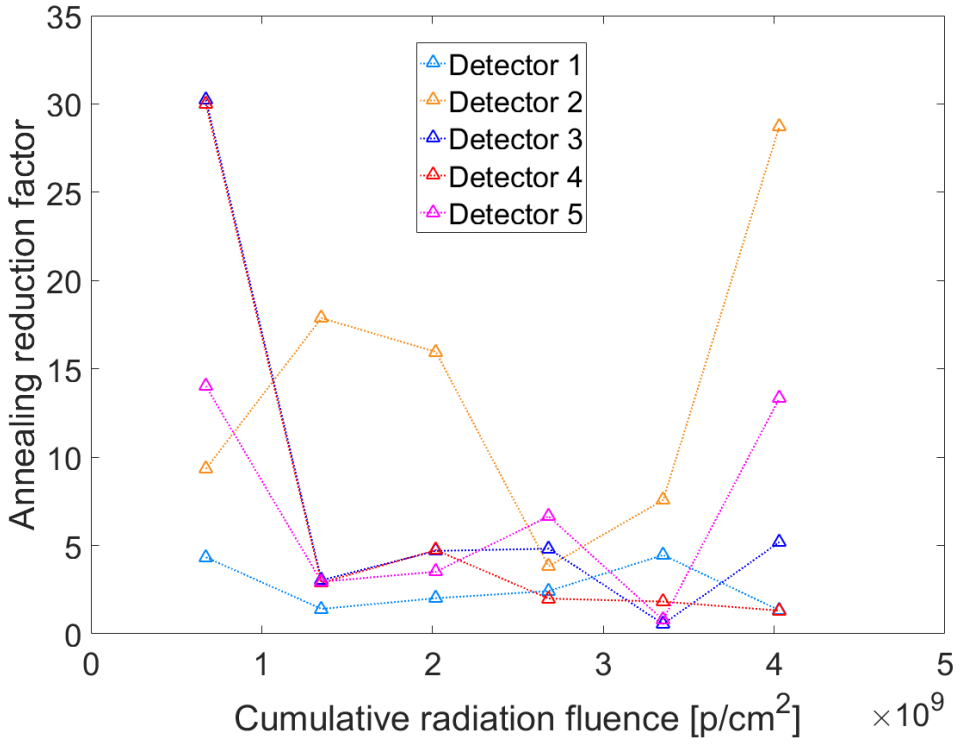


Figure 3.15: The annealing reduction factor is plotted against cumulative radiation fluence for DM1. The curve doesn't start at zero radiation because DM1 was not annealed at baseline zero radiation. All tests were conducted at an APD temperature of $-80\,^{\circ}\mathrm{C}$.

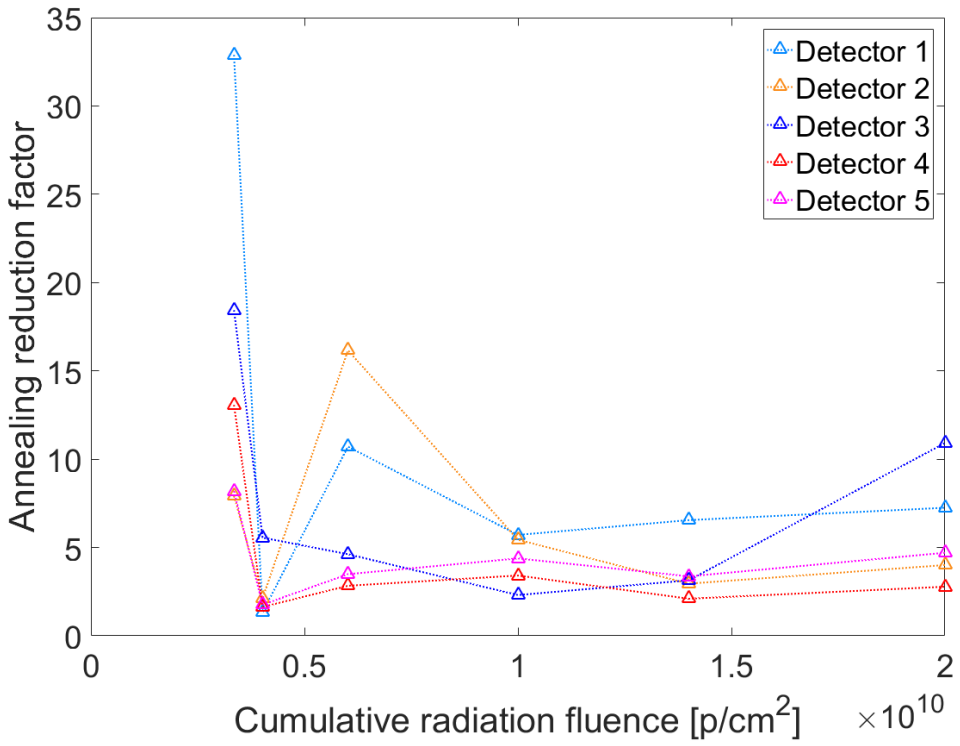


Figure 3.16: The annealing reduction factor is plotted against cumulative radiation fluence for DM2. The curve doesn't start at zero radiation because DM2 was not annealed at baseline zero radiation. All tests were conducted at an APD temperature of $-80\,^{\circ}\mathrm{C}$.

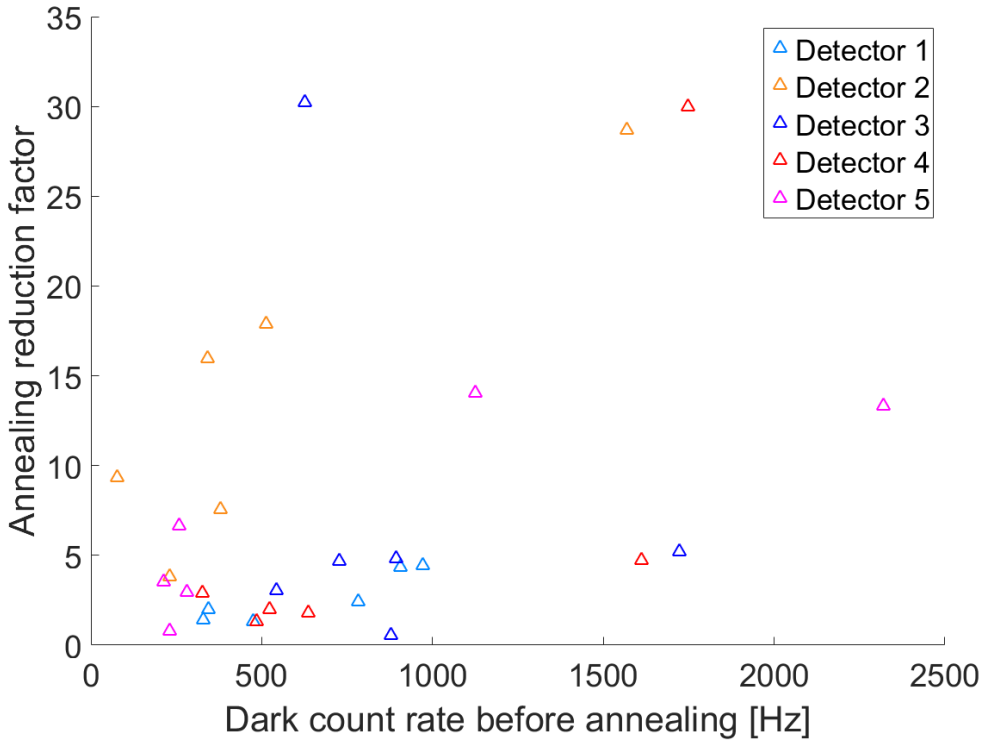


Figure 3.17: The annealing reduction factor is plotted against dark count rate before annealing for DM1. All tests were conducted at an APD temperature of $-80\,^{\circ}\mathrm{C}$.

sured before the annealing phase. There doesn't seem to be a strong correlation between the annealing reduction factor and the pre-annealed dark count rates. This may reinforce the assumption that whether or not an impurity gets fixed by annealing is independent of what happens to any other impurity during the annealing process, making the annealing reduction factor independent of the density of impurities and hence the dark count rate.

Since the annealing reduction factor (ARF) is independent of the pre-annealed dark count rate as well as the cumulative radiation fluence, one can think about how tightly grouped are the ARF values. To do this, it might be instructive to create a histogram of the ARF values. Figures 3.19 and 3.20 show the histograms of the annealing reduction factors for DM1 and DM2 respectively. Most annealing reduction factors assumes values between 2 and 5.

As explained earlier, two annealing strategies were adopted. DM1 was annealed after every incremental radiation exposure (about $6.67 \times 10^8\,\mathrm{p/cm^2}$ - 4 month LEO equivalent). DM2 was annealed only when the post irradiation dark count rate exceeded 2 kHz. If you look at figures 3.19 and 3.20, you can see that DM2 has slightly higher ARF values

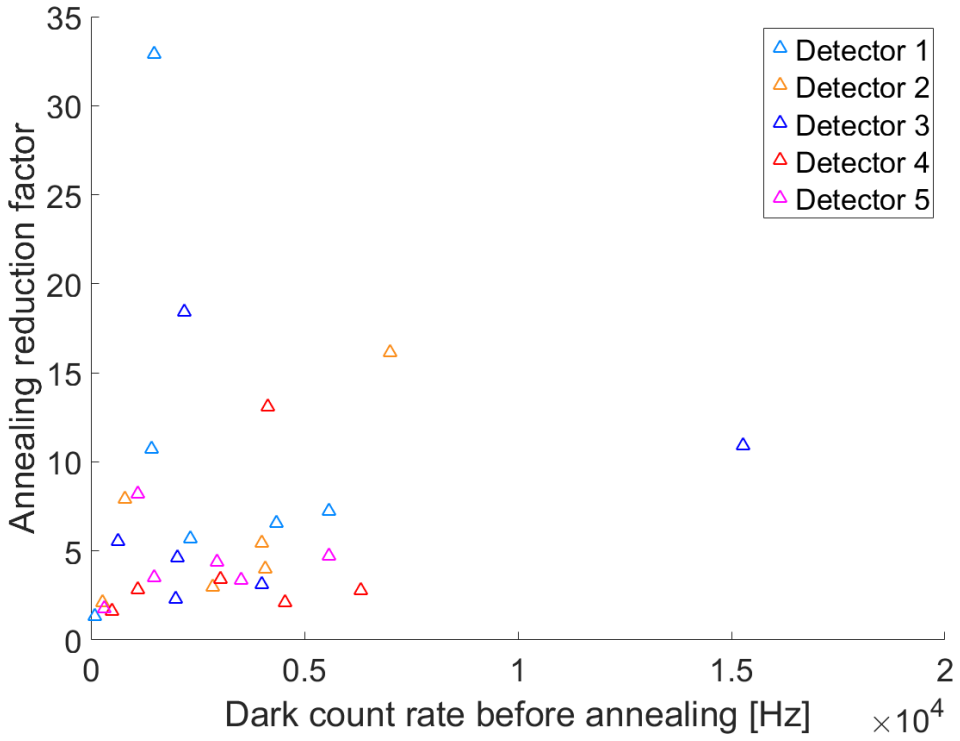


Figure 3.18: The annealing reduction factor is plotted against dark count rate before annealing for DM1. All tests were conducted at an APD temperature of $-80\,^{\circ}\mathrm{C}$.

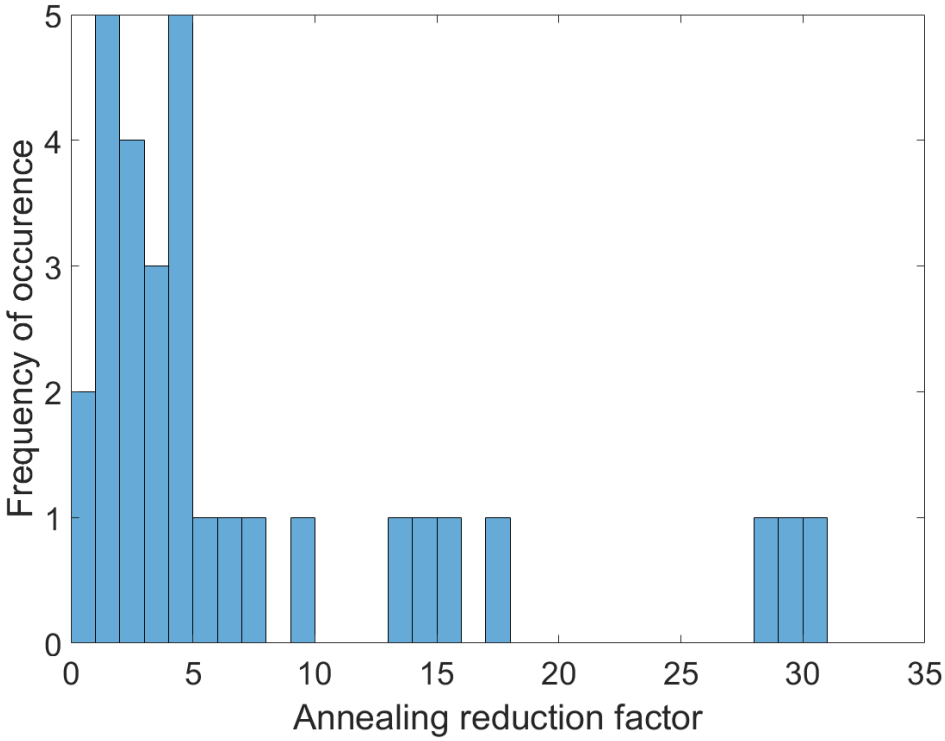


Figure 3.19: Histogram of the annealing reduction factor for DM1. All tests were conducted at an APD temperature of $-80\,^{\circ}\mathrm{C}$. Values cluster between 0 to 4

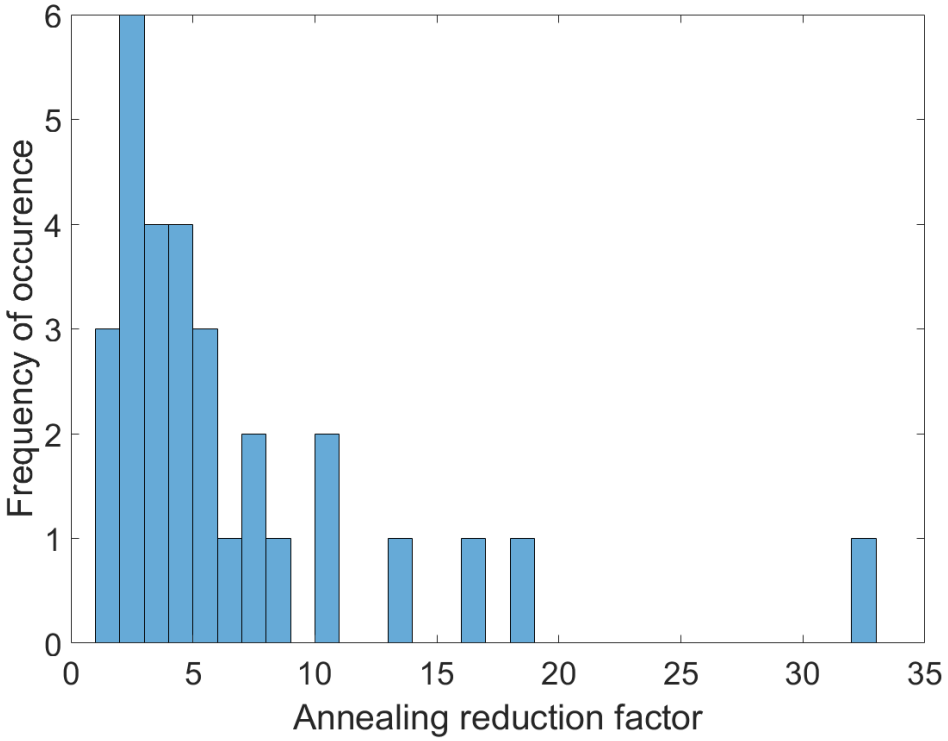


Figure 3.20: Histogram of the annealing reduction factor for DM2. All tests were conducted at an APD temperature of $-80\,^{\circ}\mathrm{C}$. Values cluster between 1 to 5

suggesting that the on-demand annealing strategy (only when dark count rate exceeds threshold) may have a slight advantage. This can be further seen in figure 3.21 which plots the dark count rates after annealing for both DM1 and DM2 up to a cumulative radiation fluence of 4.03×10^9 p/cm^2 (2 year LEO equivalent). Although the dark count rate is not believed to be correlated to the cumulative radiation dose, it might still be informative because, the absolute value of dark count rate is a key factor in satisfying the signal-to-noise ratio threshold. There are only two instances where annealing was performed on both DM's at the same cumulative radiation fluence mark. This was tests #07a and #08a for DM1, and tests #08a and #10a for DM2. DM2 achieves lower dark count rates on most detectors - Detector #2 was the only detector that registered higher dark count rates on DM2 for both tests. The ratio of the average dark count rate for DM1 on all five detectors and both tests to the same value for DM2 is about 2.5 . This might suggest that the on-demand strategy of annealing has a slight advantage over the time- periodic strategy (annealing after every irradiation dose). However, keeping in mind that there were only two tests in which the annealing strategies could be directly compared, more tests would be needed to ascertain if this advantage is outside the bounds of statistical uncertainty.

A one hour annealing time was used for most instances of the annealing process. However, we also employed annealing times of 20 min and 40 min. Figure 3.22 shows how the dark count rate on the five detectors of DM2 vary with time of annealing. It starts out with the data of test #11 (6.02×10^9 p/cm mark). DM2 is annealed for 20 min and the dark count rates on the detectors were measured. Then DM2 was annealed for a further 40 min before another dark count rate measurement was performed. The decrease in dark count rate is more significant during the first 20 min than it was for the next 40 min. This agrees with previous measurements done on such devices. The decrease in dark counts tends to be more significant earlier in the annealing phase rather than later. Eventually, increasing the annealing time will not decrease the dark count rate substantially.

Annealing changes the structure of the detector substrate to decrease the dark count rate. Another method of decreasing dark count rate without making changes to the APD substrate is to decrease the operating temperature of the APD. $-80\,^\circ$C was the lowest temperature was achievable with operating the detector TEC in closed loop control. After the two year LEO equivalent cumulative radiation fluence of 4.03×10^9 p/cm^2, the dark count rate was measured at $-80\,^\circ$C - test#08a. Cooler detector temperatures could be achieved but this involved running the detector TEC in open loop. This means that a definite value of voltage can be applied to the TEC. However, there is no information about the value of the temperature that the detectors operated at and hence to feedback to stabilize the temperature a a specific value. Nonetheless, it is plausible that when a

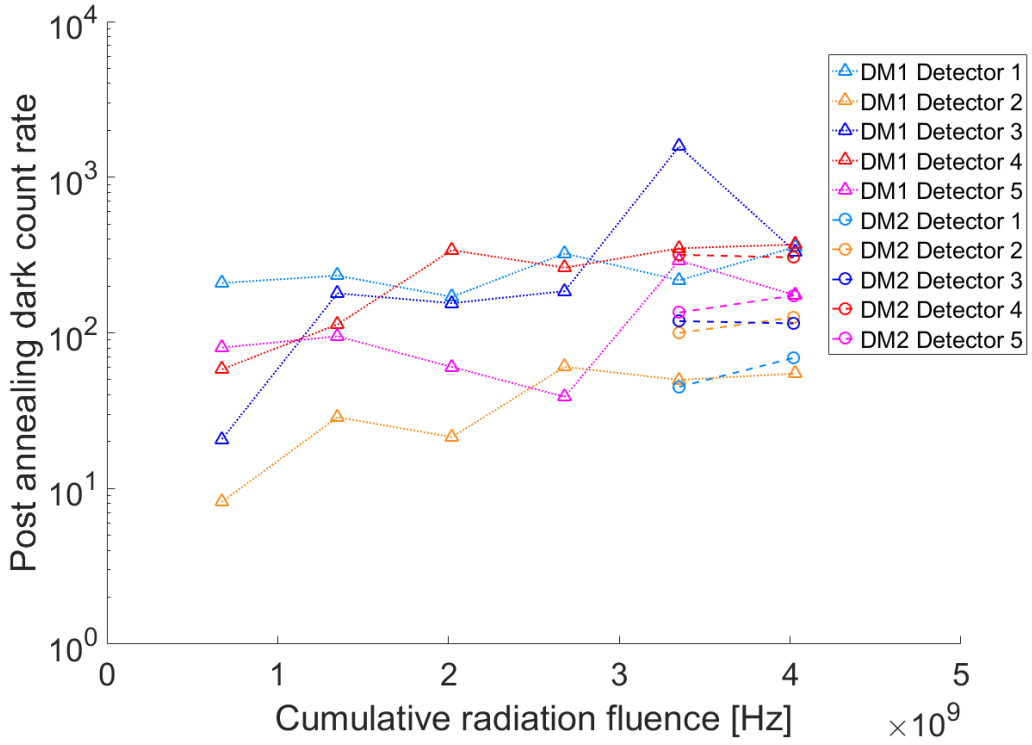


Figure 3.21: The dark count rate measured after annealing is plotted against cumulative radiation fluence for both DM's. The dark count rates were measured at an APD temperature of $-80\,°C$. DM2 shows slightly lower post annealing dark count rates suggesting that annealing only when the post annealed dark count rate exceeds a fixed threshold might be more advantageous than annealing after fixed instants of time regardless of the dark count rate.

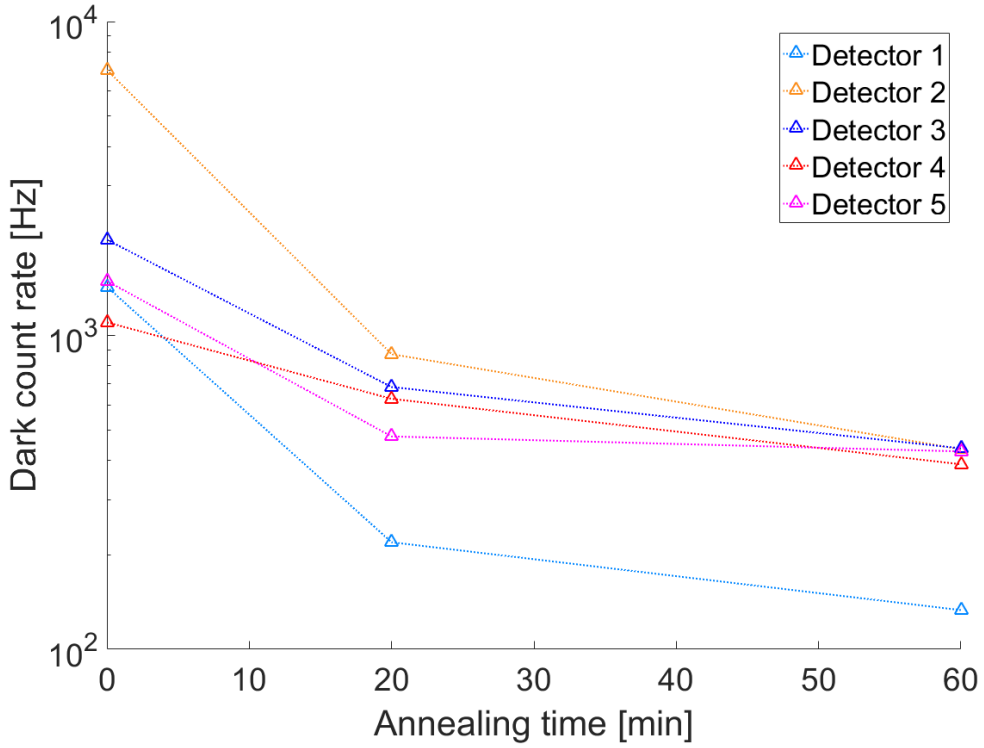


Figure 3.22: Dark count rate measured as a function fo annealing time. During annealing periods the APD temperature was held at 80 °C. In between annealing periods, the APD's were cooled to −80 °C and the dark count rates were measured. The rate of decrease of dark count rate itself decreases over time

specific voltage is applied to the TEC in open loop, a more or less definite value of detector temperature is achieved. For test #09, a maximum possible voltage of 4.43 V was applied to the detector TEC in open loop - test#09. The temperature reading displayed on the proprietary software stalled at −91.5 °C. This was not the true detector temperature as the thermistor was incapable of measuring lower temperatures than −91.5 °C - the true detector temperature would have been lesser than −91.5 °C. Nonetheless, this was the coldest temperature the detectors could be operated at with the current test setup. After a dark count measurement, the voltage to the detector TEC was changed to 1.92 V -test #10. This technically should achieve a higher temperature than test #09. The thermistor reading still capped of at −91.5 °C, meaning that the true detector temperature was still less than −91.5 °C. After another dark count measurement, the detector temperature was set to −70 °C (test #11) and then to −60 °C (test #12), both tests using using closed loop control to stabilize the detector temperature. The results of the dark count rates over the course of the aforementioned tests in plotted in figure

3.3.5 Detection Efficiency

The detection efficiency is calculated by first estimating the number of photons impinging the detector using a calibrated optical power meter just before the DM along the optical path from the laser to the DM. The ratio of the detector click rate to the impinging photon rate is called the detection efficiency. The details of this calculation are elucidated in section 2.2.6. Figure 3.23 shows the calculated detection efficiency of each of the 5 APD's in DM1 over the course of all the characterization tests. Figure 3.24 shows the corresponding detection efficiency values for DM2. No correlation is apparent between detection efficiency and cumulative radiation fluence.

Most of the detectors had a maximum detection efficiency of under 50% except for detectors #2, #4 and #5 on DM1. The maximum variation of detection efficiency on a given detector over the course of the tests was about 20% with the exception of Detector #2 on DM1 because it registered a abnormally high detection efficiency of 85% on test#05. However, there was no particular trend in the detection efficiency as the cumulative radiation fluence increased; hence the two were deemed more or less independent of each other. Since the maximum detection efficiency as specified by the manufacturer is 61%, it is unlikely that the value of 85% of Detector #2 on DM1 is real. There is a possibility that the afterpulsing probability could have increased the overall detector click rate which could increase the observed detection efficiency. However, afterpulsing calculations did not show any aberrations in test #05 on detector #2. Another possibility : there is still some

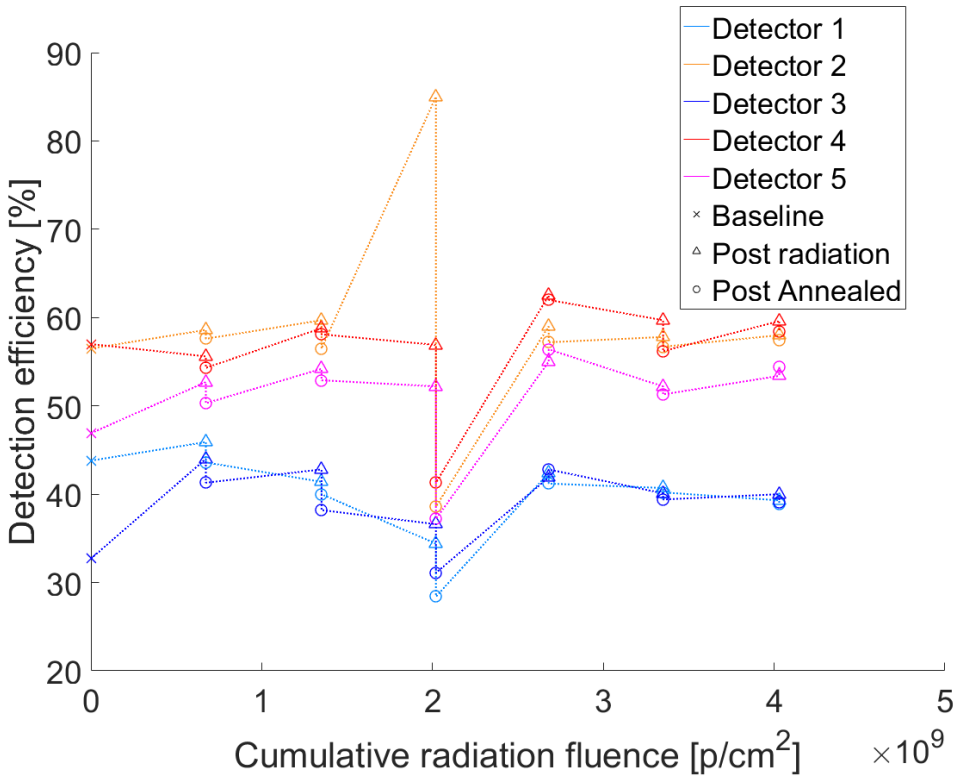


Figure 3.23: Detection efficiency of APD's is plotted against cumulative radiation fluence for DM1. All tests were conducted at an APD temperature of −80 °C.

optical path between the power meter and the DM. The ratio of the optical power input to output of this path was determined before each characterization test began. However, if this ratio changed only for the specific path leading to detector #2 (say, the black cloth had displaced a little bit, letting stray photons to enter the path through the optical fiber's jacket), the actual rate of photons impinging the detector could have increased thus increasing the detector output click rate. This would eventually lead the apparent detection efficiency to increase. The results were not reproducible and we cannot ascertain the cause of this anomaly.

3.3.6 Timing Jitter

The portable PicoQuant laser was used to generate photons which were then transmitted to the DM via a fiber bridge and optical fibers. The timing jitter was calculated for each detector by first histogramming the time delay of the photon detections with respect to the laser clock. Figure 3.26 shows the timing jitter histogram performed for test #04 again with the five detectors at −80 °C.

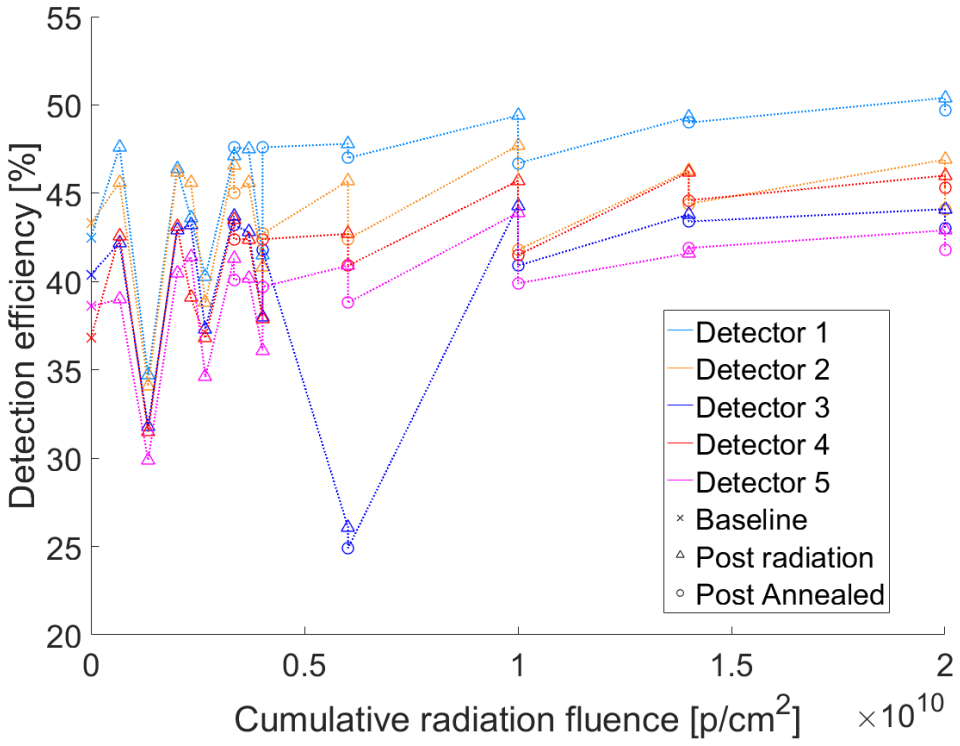


Figure 3.24: Detection efficiency of APD's is plotted against cumulative radiation fluence for DM2. All tests were conducted at an APD temperature of $-80\,^{\circ}\mathrm{C}$.

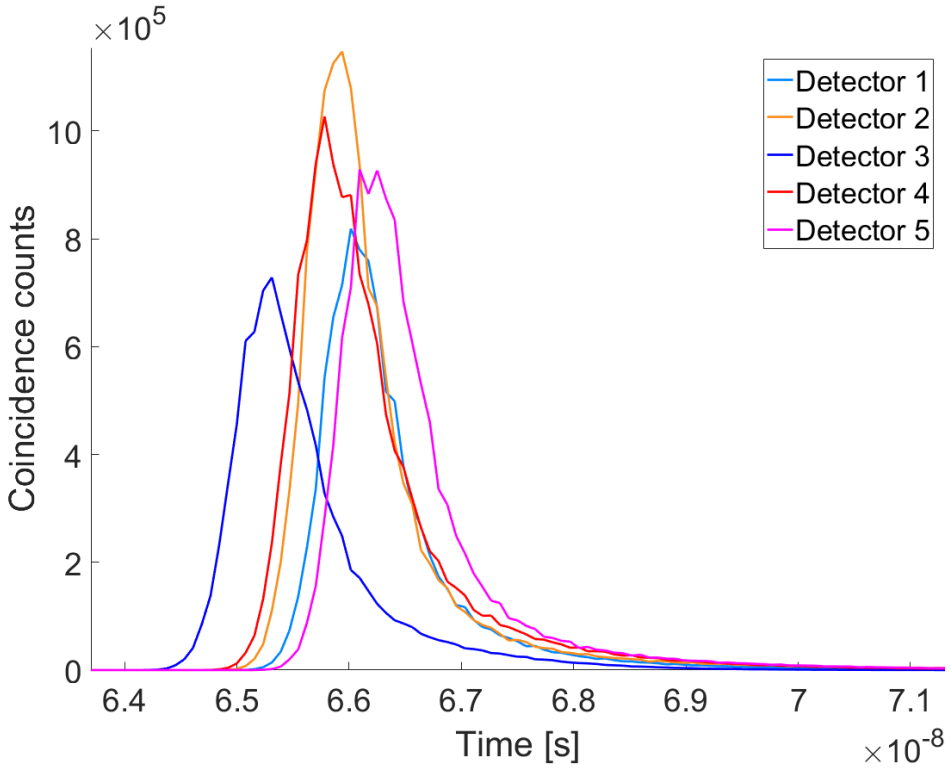


Figure 3.25: The time delay between a photon detection time tag and the corresponding PicoQuant laser clock pulse is histogrammed in the figure above. The uncertainty in the envelope of the histogram (shown as the curves) is a measure of the timing jitter of the system.This particular data was taken from DM1 on characterization test #07 at TRIUMF with a detector temperature of $-80\,^{\circ}\mathrm{C}$

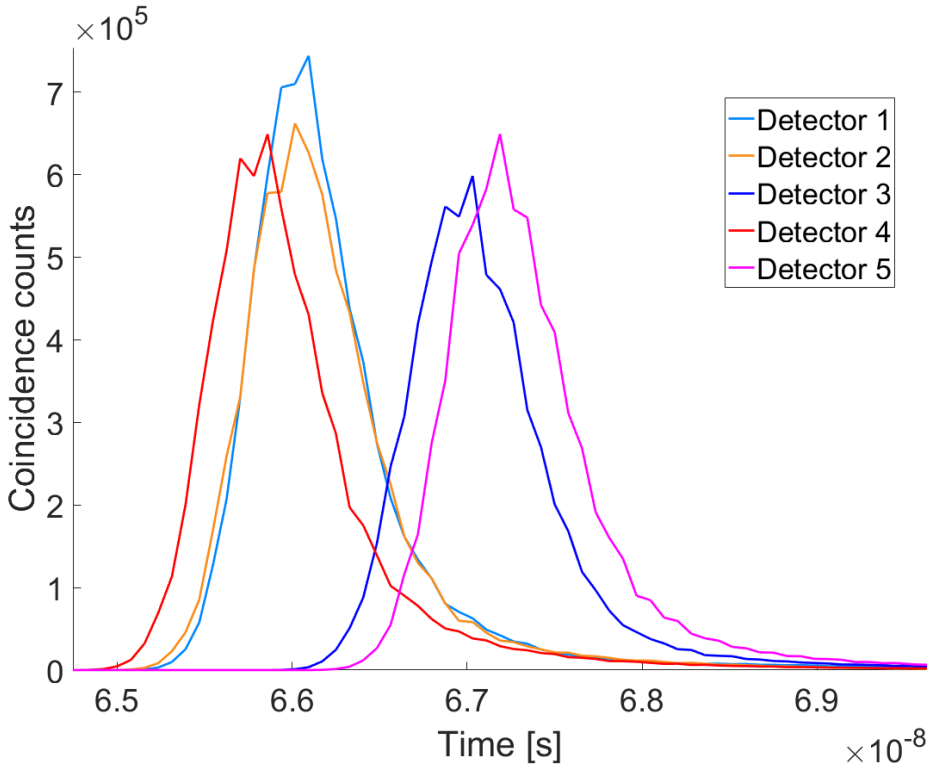


Figure 3.26: The time delay between a photon detection time tag and the corresponding PicoQuant laser clock pulse is histogrammed in the figure above. The uncertainty in the envelope of the histogram (shown as the curves) is a measure of the timing jitter of the system. This particular data was taken from DM2 on characterization test #04 at TRIUMF with a detector temperature of $-80\,^{\circ}\mathrm{C}$

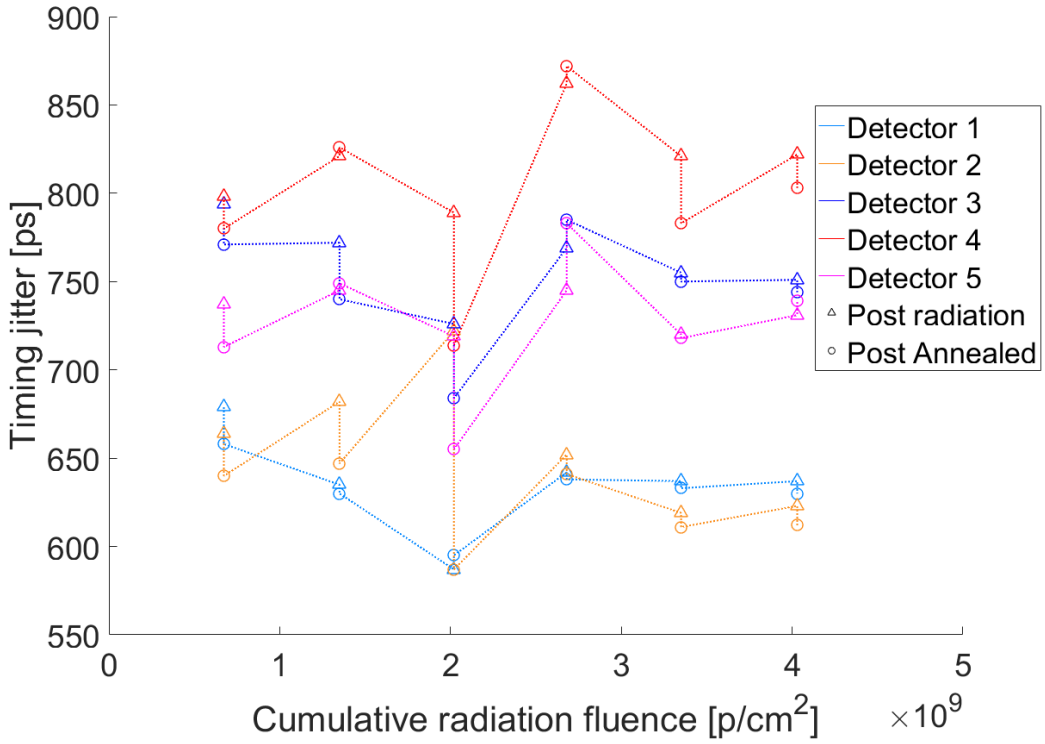


Figure 3.27: Timing jitter of APD's are plotted against cumulative radiation fluence for DM1. All tests were conducted at an APD temperature of −80 °C.

The resulting curves in figures 3.25 and 3.26 are approximated as Gaussians. The width of these Gaussians are the timing jitters of the system with respect to each detector. The detector's timing jitter is then calculated by accounting for the laser's timing jitter (measured at 465 ps) and the time taggers' timing jitter (measured at 78.125 ps) which contribute towards the overall system jitter. The details of this calculation are listed in section 2.2.7. Figures 3.27 and 3.28 show the timing jitter values for DM1 over the course of the radiation experiments. A special mention must be made that some of the tests on DM2 failed to time-tag the clock pulse due to which timing jitter information was not available. The cold baseline test at TRIUMF wa one of these. Thus, I included the cold test performed at IQC before DM2 was shipped to TRIUMF.

The timing jitter values for both DM's were independent of cumulative radiation fluence. Also, annealing didn't have an effect on the timing jitter. The timing jitter of the detectors was in the range 650 ps to 850 ps at TRIUMF. The maximum variation of timing jitter on a given detector was around 150 ps with only detector #3 on DM2 varying by nearly 150 ps. A special mention must be made that the test at IQC, both before leaving to and after returning from TRIUMF, yielded consistently lower timing jitter values

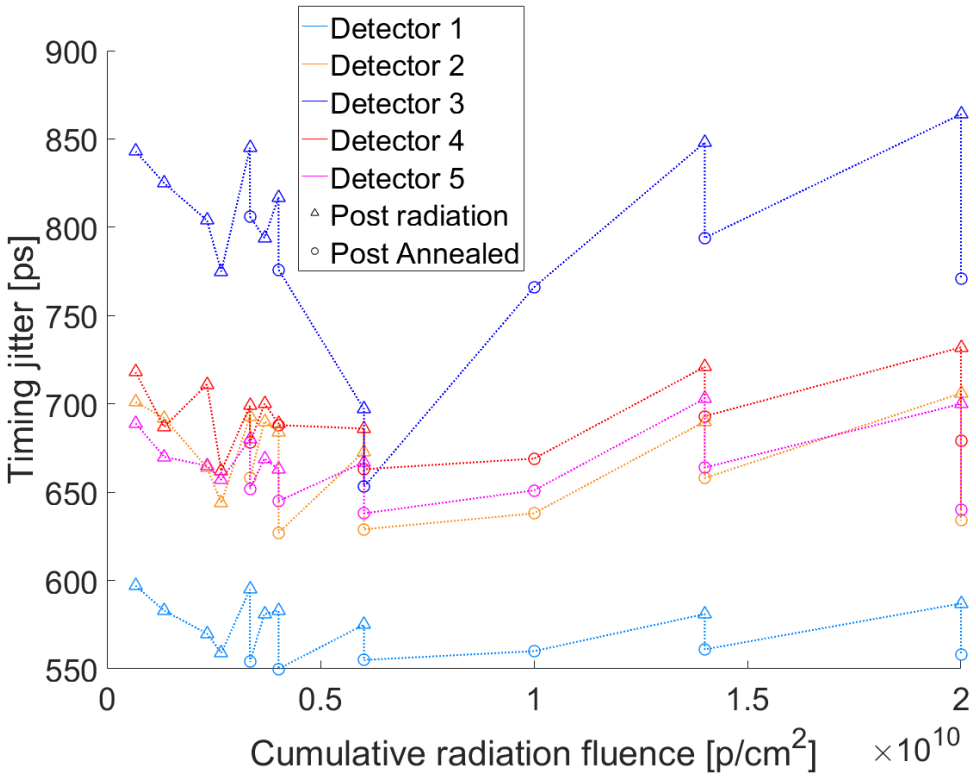


Figure 3.28: Timing jitter of APD's are plotted against cumulative radiation fluence for DM2. All tests were conducted at an APD temperature of $-80\,^{\circ}\mathrm{C}$.

compared to the TRIUMF results. The tests at IQC before leaving TRIUMF yielded timing jitter values in the range 525 ps to 665 ps whereas the IQC tests after returning from TRIUMF yield jitter values in the range 330 ps to 610 ps. Since the lower values were observed both before and after the TRIUMF tests but not at TRIUMF, it is believed that the difference is due to the experimental set up rather than arising from any changes in the APD's. However, one cannot draw a conclusion if the values observed at IQC or TRIUMF are more accurate. Nonetheless, the independence of timing jitter on radiation fluence and annealing can be ascertained due to the non-observance of any particular trend during the TRIUMF tests.

3.3.7 Saturation

When laser power is increased from zero, the output detector click rate increases reaches a maximum value and decreases to zero. From this point if the laser power is decreased, the aforementioned output trend reverses. The maximum possible detector click rate is called the saturation value. The saturation measurements were performed only during the pre-radiation cold test at IQC (tests #00), the baseline tests at TRIUMF (test #01 and #02), at the two-year LEO equivalent cumulative radiation fluence mark (tests #08 and #08a for DM1 and test #10a for DM2) and the post radiation cold test at IQC (test #13 for DM1 and test #17 for DM2). The reason that the other tests at TRIUMF did not include saturation measurement was that it takes a few minutes to do the measurement (and time was key at TRIUMF) and saturation level wasn't expected to be affected by radiation of annealing.

The saturation value for each detector was rerecorded on each test. Figures 3.29 and 3.30 plot the saturation values for DM1 and DM2, respectively, over the course of the different tests. The maximum count rate of a given detector is not correlated to cumulative radiation fluence. Although, there was only one instance where the pre-annealed saturation value was compared with the post-annealed saturation value, it is projected that saturation value is not affected by annealing either. The saturation values were in the range 166 kHz to 960 kHz. The reason for this large range is that different detectors had very widely spaced saturation values even in the same characterization test. The reason for this has not been determined. However, it is important to note that the maximum variation of saturation values on a given detector across all tests was 250 kHz.

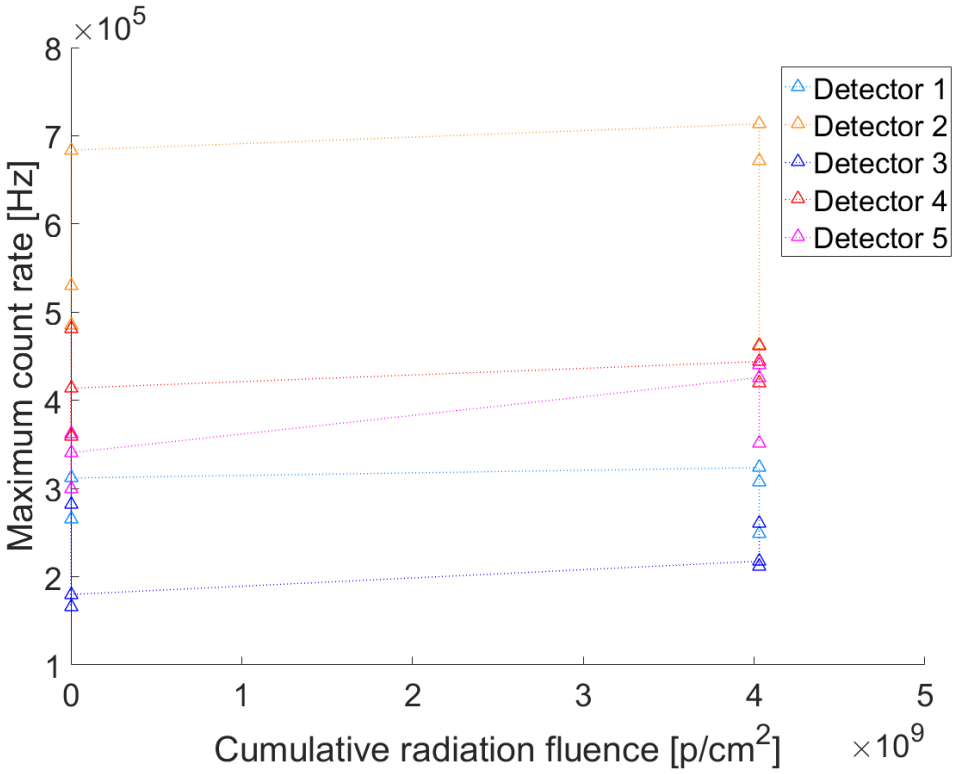


Figure 3.29: The saturation curve for each detector on each test is plotted for DM1. The the maximum value (i.e., the maximum count rate) of each curve is then tabulated and plotted in the figure above. All tests were conducted at an APD temperature of $-80\,^{\circ}\mathrm{C}$. The saturation values are not dependent on cumulative radiation fluence nor on annealing

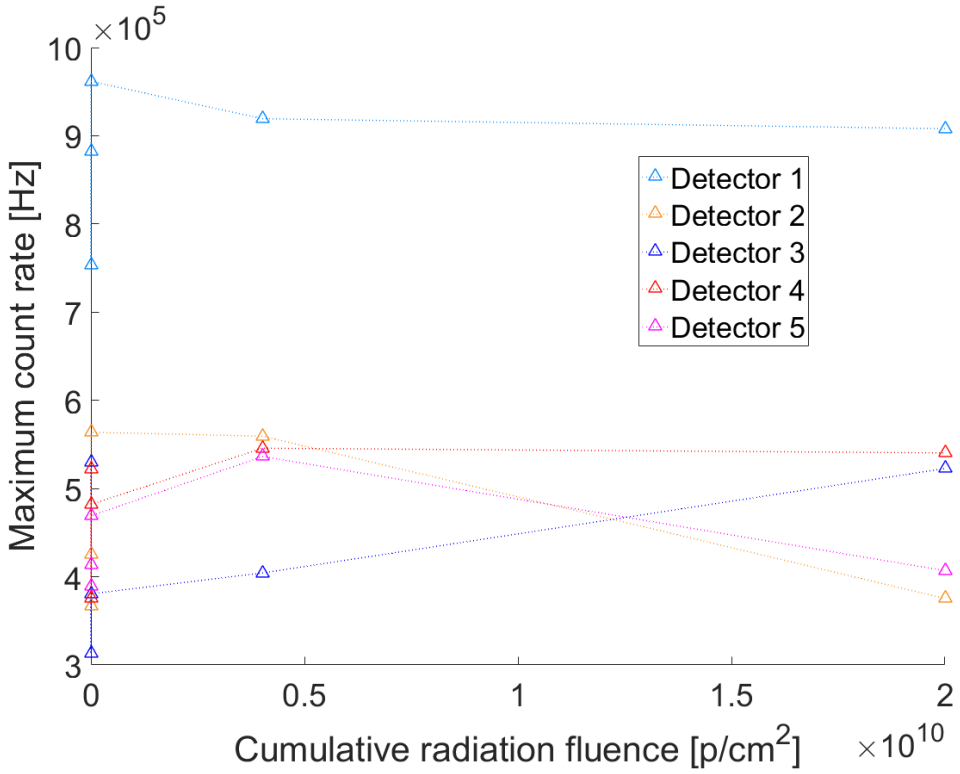


Figure 3.30: The saturation curve for each detector on each test is plotted for DM1. The the maximum value (i.e., the maximum count rate) of each curve is then tabulated and plotted in the figure above. All tests were conducted at an APD temperature of $-80\,^{\circ}\mathrm{C}$. The saturation values are not dependent on cumulative radiation fluence nor on annealing

3.3.8 Afterpusling

When an avalanche occurs in the APD's substrate, there is some chance that an electron can get trapped in one of the trap sites. This electron when detrapped triggers a second avalanche called an afterpulse. The probability with which a detector click gives rise to an avalanche is called the afterpulsing probability.

Elena Anisimova, a former doctoral student at the Institute for Quantum Computing developed a program in Python to calculate the afterpulsing probability of the APD's using the measurement data from the dark count experiments. Details of this calculation are given in section 2.2.9. Figure 3.31 shows the histogram output of the Python program. The program requires the user to specify a point on the histogram which separates the afterpulsing region to its left from the steady state region to its right. It then calculates the area under the histogram but above the y value of the selected point in the afterpulsing region only (region to the left of the selected point). The user visually determines this point on the histogram. There will be some error on his part. The program uses exponentially increasing bin size and the afterpulsing probability which is the area under the curve picks up greater and greater areas as you go towards the right (for same bin heights). Due to this, any error on the visual location of the point separating the afterpulsing region from the steady state region contributes exponentially towards errors in afterpulsing probability. This makes the algorithm sensitive to user error. This effect is magnified in tests where the transition from afterpulsing to steady state region is gradual.

In order to make the output of the algorithm less sensitive to user choices, I automated the process of the the region demarking point. Moving from the rightmost bin of the histogram, I checked if the bin to its left deviated from the bin value of the rightmost bin by more than a certain percentage threshold fixed in the algorithm. If it didn't, I averaged bin values of the two bins by weighting them by their true bin size. Then I checked the value of the third bin from the right to see if it deviates by more than the same percentage threshold from the aforementioned calculated weighted average. If it didn't, I calculated the weighted average of all three bins. The idea is that the weighted average represents the steady state value towards the right of the histogram. Upon repeated iterations, when the bin value deviates by more than the percentage threshold of the weighted average, the algorithm deems this points as the region demarking point. This modification didn't require any input from the user and hence the program outputted the same value of afterpulsing probability during every instance of program execution.

However, this then shifted the burden of choice from the user to the programmer who selects the aforementioned percentage threshold. Changing this threshold changes the

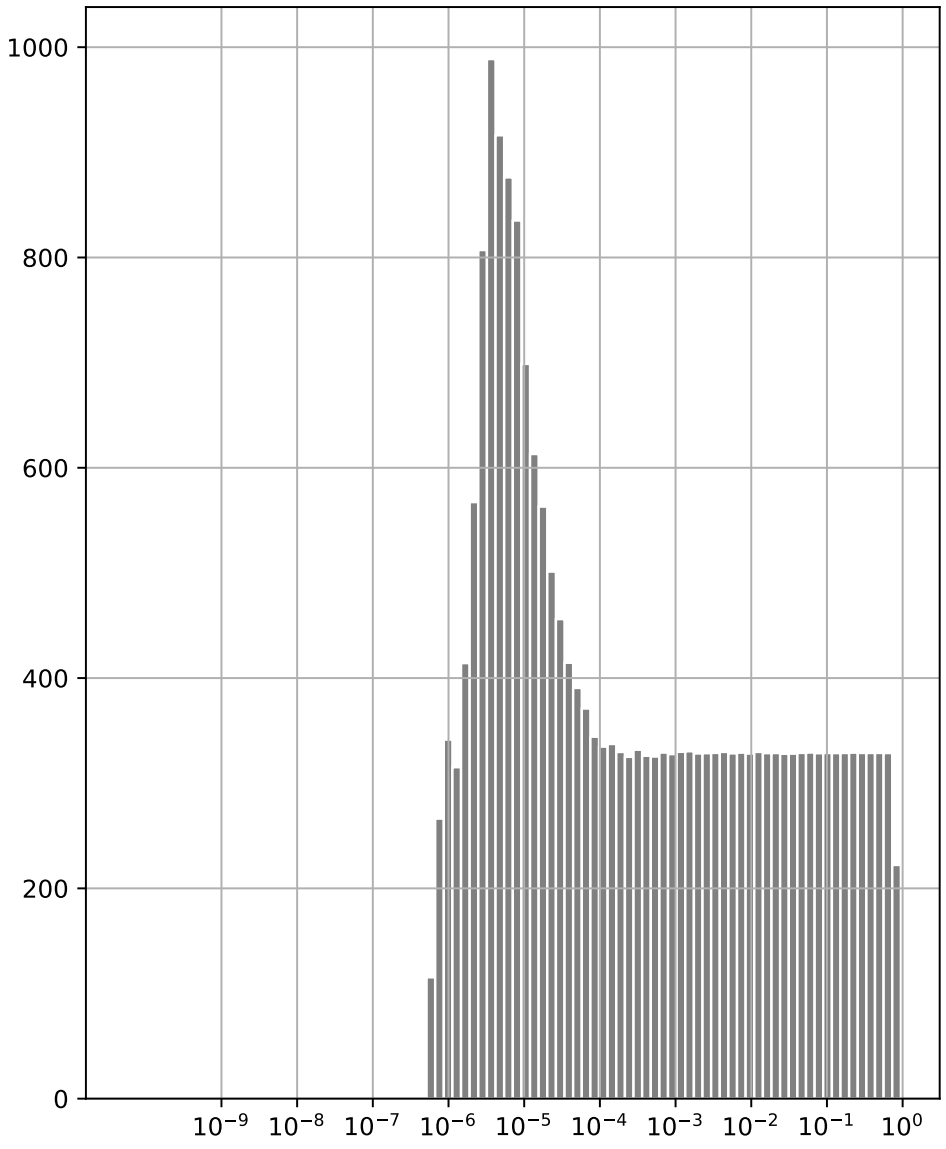


Figure 3.31: Output of Elena Anisimova's Python algorithm [1]. The histogram plots the time difference of all time tags following the start time tag. Data is from DM 1 test 04 Detector 1 with a detector temperature of $-80\,^{\circ}\mathrm{C}$

afterpulsing probability. The redeeming factor though is that a data set with a higher true afterpulsing probability will yield a higher output value from the Python algorithm than that from a dataset with a lower true afterpulsing probability. So, it is possible to compare two data sets and ascertain which one has the higher probability.

But what about the accuracy of the absolute value of the calculated afterpulsing probability? Using the QKD simulation described in chapter 6, I was able to generate thermal counts and their chain of afterpulses (see sections 6.4.1 and 6.4.2 for details). Using this, I am able to play around with data sets of known afterpulsing probabilities. The programmer can then fine tune the value of the percentage threshold using these data sets so as to get a output value for afterpulsing probability within a required percentage of the true afterpulsing probability. This then solves the issue of accuracy to some extent.

There is a problem even with automating the selection of the region demarking point. There is some statistical noise during the region transition. This introduces some noise in the location of the region demarking point as determined by the algorithm as mentioned above. Since this noise occurs at the rightmost end of the afterpulsing region and since the rightmost bins have wide time intervals, the error in selecting the region demarking point results in large errors in the calculated value of afterpulsing probability because it uses the Riemann integral. So, although for a particular experimentally generated data set, the output of the program does not change with multiple executions, if one experimentally generates multiple data sets using the APD in similar conditions (same cumulative radiation,detector temperature, etc.), one will find that the statistical noise in the region transition cause the program to output different values of afterpulsing probability for each of these data sets.

One way to get around this problem is to curve fit the envelope of the histogram to smooth out the noise and to use the aforementioned methods on the curve fitted histogram. I didn't proceed with this primarily because I wanted to know the analytical solution of the histogram envelope to see what effect the process of curve fitting might have. Also, I am not sure how to recreate this noise in my simulate data especially because the afterpulsing peak contains all afterpulses in the chain of afterpulse generated by the start time tag (on avalanche gives rise to an afterpulse which in turn gives rise to another afterpulse, etc.) This is possibly an avenue for future research. However, even without an analytical solution, it should be possible to curve fit the histogram to arrive at a stable output relatively insensitive to the noise.

Figures 3.32 and 3.33 show the afterpulsing probability as calculated by the modified python algorithm. The percentage threshold to determine the region demarking point was set to 4%. The afterpulsing probability towards the later tests tend to be higher than

the initial few ones. However, the impact of irradiation and annealing is hard to decipher with different detectors showing different trends. For example the majority of tests on detect #5 in DM1 irradiation increases afterpulsing probability while annealing decreases afterpulsing probability. On the other hand, the majority of tests on detector #2 in DM2 show that irradiation decreases afterpulsing probability while annealing increases afterpulsing probability. I would say that one cannot draw any conclusive correlation between afterpulsing probability and irradiation / annealing, except for the fact that most detectors ended up with higher afterpulsing probabilities at the end of the series of experiments. Since this happens for most of the detectors, there is some reason to believe that afterpulsing probability increases with irradiation or annealing or a combination of both. Finally, if you notice, some of the afterpulsing probabilities are greater than 100%. By our assumption that each avalanche cannot on average give rise to more than one first generation afterpulse (which is a good assumption considering we don't have a runaway effect of detector click rate), the afterpulsing probability cannot theoretically be greater than 100%. My contention is that all the successive generations of afterpulses of the start time tag occur in the afterpulsing peak. The n^{th} generation afterpulse is connected to the $(n+1)^{th}$ generation afterpulse through an exponential decaying function in time. Due to these correlations between consecutive generations of afterpulses, they accumulate in the peak of the histogram. Given this, the program is outputting the average of total number of afterpulses (all generations) produced by an arbitrary time tag. This value can indeed be greater than 1 (100%). The conversion between the afterpulsing probability p according to our definition and the value outputted by the algorithm p' is given by

$$p' = \frac{p}{1-p} \tag{3.1}$$

The details of how equation 3.1 is partially derived is given in section 2.2.5.

Altogether another approach was suggested by Jean-Philippe Bourgoin and Brendon Higgins, both Post-doctoral fellows at IQC and members of Prof. Thomas Jennewein's group. The efficiency measurements with the laser on, were used for this calculation. The idea is that laser photons generate detector clicks. It is possible to know with a high probability which detector click is caused by a laser photon by having access to the time-tagged laser clock pulse and knowing the approximate value of the optical path time - the time taken by the photon to travel from the laser to the detector. The photon clicks generate afterpulses. After accounting for the thermal counts and their afterpulses, one can calculate the afterpulsing probability. Details of these calculations are given in section 2.2.9. This approach has the advantage that it does not require the user or programmer

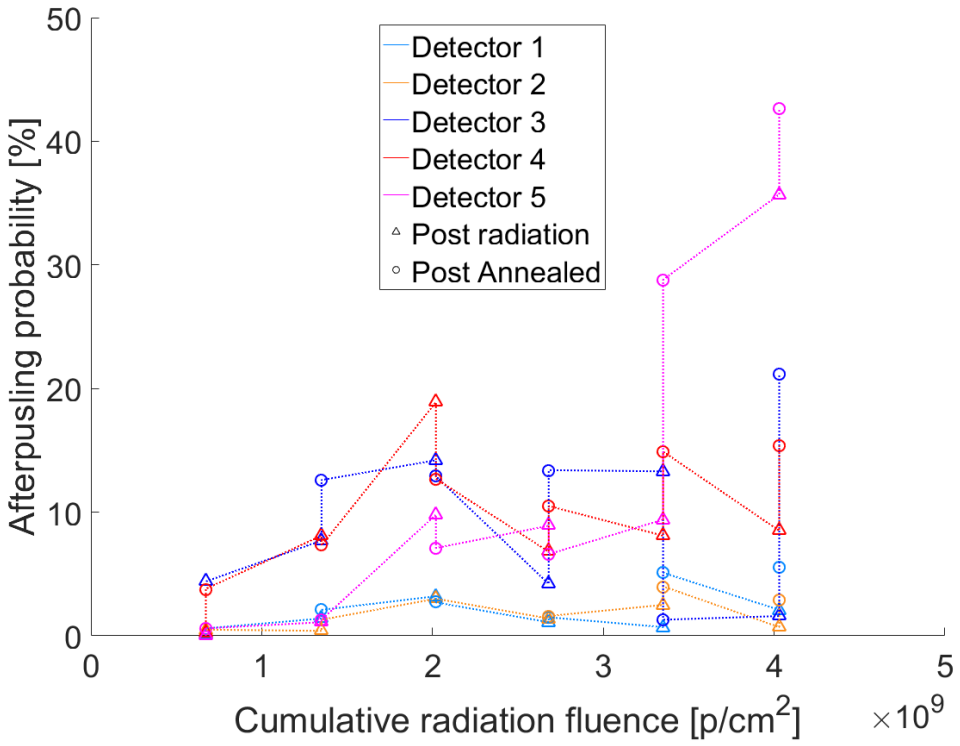


Figure 3.32: Afterpulsing probability is plotted against cumulative radiation fluence for DM1 using dark count measurement data in Elena's python program [1]. All tests were conducted at an APD temperature of $-80\,^{\circ}\mathrm{C}$.

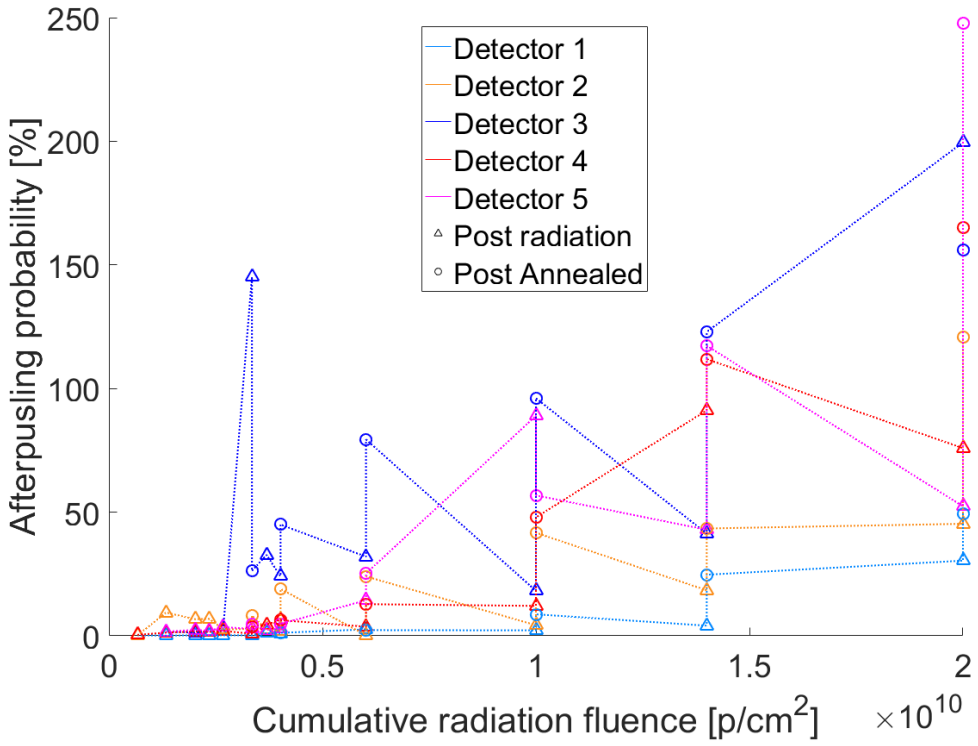


Figure 3.33: Afterpulsing probability is plotted against cumulative radiation fluence for DM2 using dark count measurement data in Elena's python program [1]. All tests were conducted at an APD temperature of $-80\,^{\circ}\mathrm{C}$.

to personally choose the value of a program variable, thus making it immune to human discretion.

My contribution here was to implement this algorithm in Microsoft Excel and calculate the afterpulsing probability from the efficiency measurement data for all the tests conducted at TRIUMF. Figures 3.34 and 3.35 show the afterpulsing probability plotted as a function of cumulative radiation fluence. Here too, one notices that the afterpulsing probability somewhat tends to increase towards the later tests compared to the initial few ones. Again, there isn't a definitive trend with respect to irradiation or annealing increasing or decreasing the afterpulsing probability. However, a key difference from the Python algorithm is that the afterpulsing values are smaller especially for the later tests. For example, the maximum afterpulsing probability. For example, the maximum afterpulsing probability outputted by the method that uses efficiency measurement data is 26% where as the afterpulsing probability p calculated by converting the output of the Python algorithm has a maximum value of 71.2% which occurred in test 15 on detector #5 on DM2. It is likely that this method is more accurate than the Python algorithm because an afterpulsing probability of 71.2% would have surely altered the efficiency value of this particular detector in that particular test. However, the detection efficiency of detector #5 on DM2 in test 15 was 42.9% which was lower than the detection efficiency of detector #1 on DM2 in test 02 registering a value of 47.6% efficiency. However, afterpulsing probability p of detector #1 on DM2 in test 02 as determined (and converted) by the Python algorithm was 0.2%. If the Python algorithm was accurate, the detection efficiency of detector #1 on DM2 in test 02 should have been significantly higher than that of detector #5 on DM2 in test 15 because the generations of afterpulses due to a photon click would have increased, and the detection efficiency calculations did not subtract out the afterpulses due to photons. Although the method of afterpulsing probability calculation that used the efficiency measurement data also ascribed detector #5 on DM2 in test 15 an afterpulsing probability of 14.9% compared to 0.5% for detector #1 on DM2 in test 02, this difference is smaller and may be more representative of the true afterpulsing probability.

The main disadvantage with the two methods of calculating afterpulsing probability is that one doesn't quite know if the outputted value of the method is ultimately correct or not.

I spent some time working on a third method which first plots a histogram of the time difference between consecutive detector clicks. Ursin and Peev [33] have a similar concept but I have made many changes and slightly different assumptions. The idea here is that thermal counts obey a Poisson distribution in time. Accordingly the time difference between two thermal clicks follows an exponentially decaying distribution. If no afterpulses

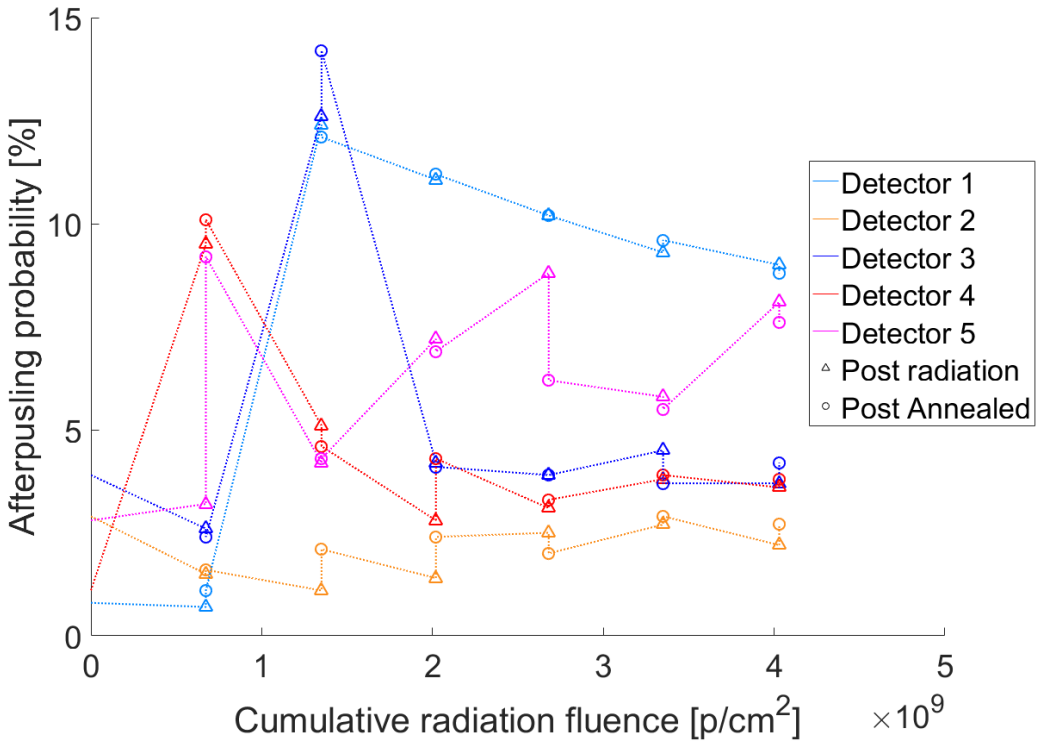


Figure 3.34: Afterpulsing probability is plotted against cumulative radiation fluence for DM1 using efficiency measurement data. All tests were conducted at an APD temperature of $-80\,^{\circ}\mathrm{C}$.

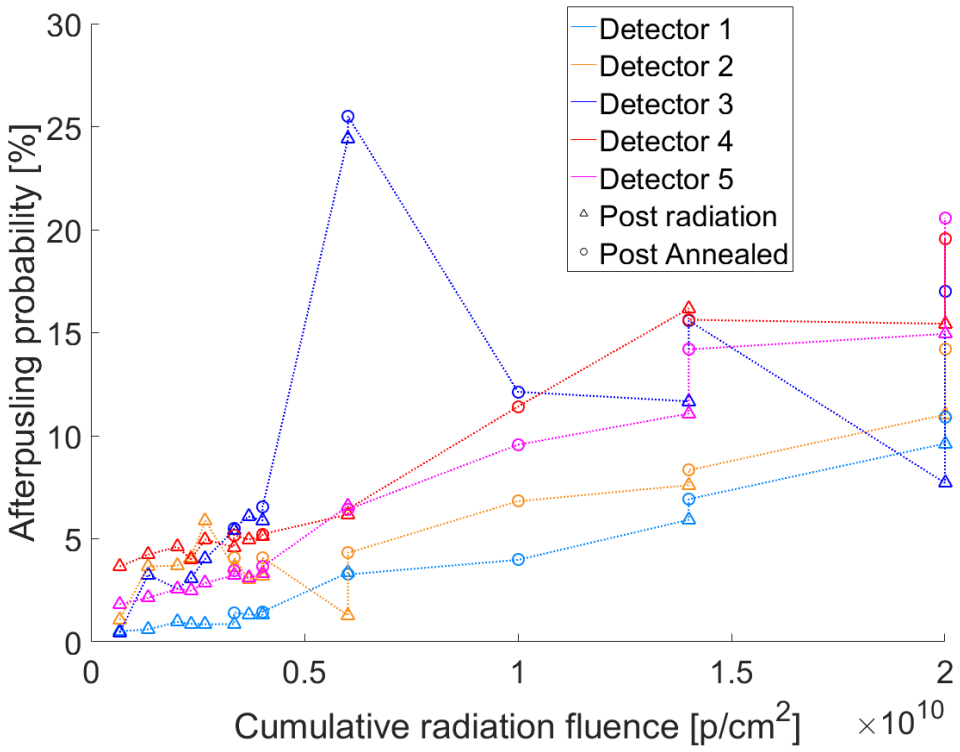


Figure 3.35: Afterpulsing probability is plotted against cumulative radiation fluence for DM2 using efficiency measurement data. All tests were conducted at an APD temperature of $-80\,^{\circ}\mathrm{C}$.

are present, this is the distribution of time difference between time tags. However, due to the presence of afterpulses, the distribution of time difference between time tags deviates from the exponentially decaying one. The extent of deviation helps to quantify the relative number of afterpulses and correspondingly the afterpulsing probability. If the average time period of the thermal counts is orders of magnitude greater than the average detrapping time of a trapped electron, i.e., the inverse of the afterpulsing time constant, the region to the right of the histogram of time differences between consecutive detector clicks should comprise primarily of thermal counts and should be approximately exponentially decaying with time constant equal to the average frequency of thermal counts. By taking the logarithm of y-axis of the histogram, the rightmost region becomes a straight line. One can then use curve fitting algorithms to get the slope of the line which is related to the thermal count rate. Knowing the thermal count rate and the overall detector click rate, one can find the afterpulsing rate and eventually the afterpulsing probability. Alternatively, the straight line region to the right of the histogram can be extrapolated towards the left side and the area under the histogram but above the extrapolated straight line gives the afterpulsing probability as well. Different region demarking points are selecting and a histogram of the average of the values of afterpulsing probabilities calculated by the two aforementioned approaches is histogrammed. Details of this entire method are given in section 2.2.9. The key assumption here is that the frequency of thermal counts is orders of magnitude lesser than the afterpulsing time constant. Given this assumption is satisfied in experimental dark count measurement data, this method has a huge advantage in that its output is a histogram of possible afterpulsing values. The histogram should have a distinctive peak which represents the afterpulsing probability of the APD. The sharpness of the peak is a measure of how reliable the method is for that particular data set- the sharper the peak, the more reliable the output. This is somewhat of an advantage because, the previous two methods (the Python and Excel programs) don't give an indication about the reliability of their output. And since all three methods give different answers, this is important.

A note about the scalability of the program. As the size of the time tag file increases, it takes more time to read the time tags as this is done in a for loop. Once the time tag file is read and the time tags are stored in a MATLAB array, plotting the histogram is fast since, getting the values of time difference between consecutive time tags is easily implemented by the following code

```
1 timeDifferences = timeTagArray(2:end) - timeTagArray(1:end-1)
```

Getting the set of time difference values is quick. Similarly, creating the histogram is rather quick as well and scales more slowly with the size of the time tag file than the reading of file. Once the histogram of time differences is created, the computation time for remainder of the calculations is not affected by the size of the time tag file because the size of the time tag file only affects the height of the bins of the histogram and not the number of bins. So, say, compared to Python algorithm, this method scales much more nicely with the size of time tag file.

To test this algorithm, I simulated thermal counts and afterpulses in MATLAB. Dead time and recharge time were also incorporated in the simulated dataset. I was also able to keep a track of the thermal counts and afterpulses in my simulation and calculate the observed afterpulsing probability as the ratio of afterpulses to total number of time tags. I used a dark count rate of 200 Hz, recharge time of 1×10^{-6} s, afterpulsing probability 25%, afterpulsing time constant $1 \times 10^{6}\,\mathrm{s}^{-1}$ and dead time 1×10^{-7} s. I was able to keep track of the time tags by labeling them as thermal counts or afterpulses. Because of the dead time and recharge time, the observed afterpulsing should decrease. The observed afterpulsing is defined as the ratio of the number of afterpulses to the total number of detector clicks. A more detailed treatise on why the afterpulsing probability decreases is given in section 5.2. For this particular choice of detector parameters, the observed afterpulsing probability was 15.29%. I then stored these time tags in a data file and used the afterpulsing algorithm I developed here to analyze the afterpulsing probability. Figures 3.36 shows the PMF of time difference between consecutive detector clicks. Exponential (to the base 10) increasing bin widths were used and the time difference values in each bin was counted. The bins values were then normalized to unity (sum of bin values = 1). The hump on the left contains both thermal counts and afterpulses where as the hump to the right contains only thermal counts. Figure 3.37 shows the corresponding PDF got by dividing the value of each bin by the bin width. An important point to note here is that although the PMF shape of the PMF depends on the type of binning used (linear or exponential), the PDF does not (with enough samples). If you notice, there is an initial rise in the PDF. This is due to the recharge time where the detection efficiency is increasing with time elapsed since the start time tag. The thermal region to the right of the histogram here is not linear because the x-axis was logarithmic (to view the details of the curve). In linear x-axis, the thermal region is linear. As described, I calculated the afterpulsing probability two ways and set a threshold such if the ratio the smaller value to the large value of afterpulsing probability is higher than a threshold value (which is less than unity), I consider the average of the two afterpulsing probability values. I then histogram all these averages. Figure 3.38 shows the histogram of the average of the two afterpulsing probability values when the threshold

ratio ratio is set to 0.99. I used linear binning of bin width 0.5% There are two peaks, a larger one centered around 15.25% and a smaller one at 10.25%. When the threshold ratio is decreased to 0.95 in figure 3.39, a clear peak emerges at 15.25%. Further decreasing the ratio down to 0.75 in steps of 0.05 in figures 3.40, 3.41, 3.42 and 3.43 shows that neither the peaks position or the height changes. In fact, if the threshold ratio ratio is driven down all the way to 0, effectively not setting a threshold, as in figure 3.44, one sees that sees the same peak in position and height. I hadn't initially expected this because, even if the two afterpulsing probability values differ by orders of magnitude, they are still plotted in the histogram. I think this result is because the cases where the two values differ a lot are cases where the values are not really emerging from any underlying structure in the data set. They can be seen as nearly random and are dispersed throughout the histogram, hence not being able to form a peak. The values that do emerge from the structure of the afterpulsing characteristics of the data set are the values that cluster together forming a peak. Experimental data sets tend to have a lot of noise and I found that the histogram which does not use any threshold often leads to more reliable results than using a non-zero threshold value.

I then proceeded to calculate the afterpulsing probability for the test conducted as part of the radiation experiments. Figures 3.45 and 3.46 show the P.M.F and P.D.F. of the time distribution for the next click for Radiation DM1 test#06 detector #03. Using the P.D.F., i followed the procedure of the afterpulsing algorithm to create histograms of potential afterpulsing probability values with different thresholds on the ratio of the values obtained from the two methods of calculating the afterpulsing probability. Figures 3.47, 3.48 3.49, 3.50, 3.51 and 3.52 show the histograms with threshold values 0.99, 0.95, 0.90, 0.85, 0.80 and 0.75, respectively. Finally, I also created a histogram of potential afterpulsing probability values without any threshold which is shown in figure 3.53. I also similar results for Radiation DM2 test #14a detector #3. Figures 3.54 and 3.55 show the P.M.F and P.D.F. of the time distribution for the next click for Radiation DM1 test#06 detector #03. Figures 3.56, 3.57, 3.58, 3.59, 3.60and 3.61 show the histograms with threshold values 0.99, 0.95, 0.90, 0.85, 0.80 and 0.75, respectively. Finally, I also created a histogram of potential afterpulsing probability values without any threshold which is shown in figure 3.62. One notices that when we compare the two tests for the histogram of potential afterpulsing probabilities at the same threshold value, the "width" the peak for Radiation DM2 test #14a detector #3 is greater. If the width is very large, it becomes difficult to find the exact location of the peak, making the program unreliable. Therefore, the width of the peak can be used as a measure of the reliability of the final value of afterpulsing probability outputted by the algorithm. This measure of reliability wasn't there in previous analysis

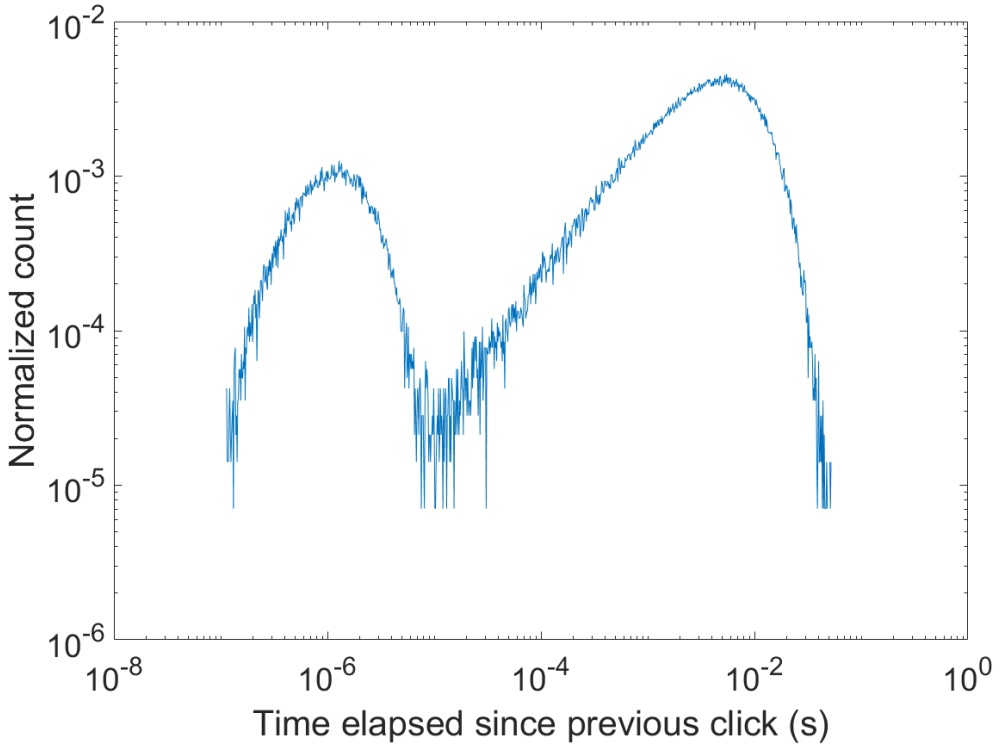


Figure 3.36: Probability mass function of time difference between consecutive time tags using simulated dark count measurement data. The time tag file was simulated in MATLAB. A dark count rate of 200 Hz, recharge time of 1×10^{-6} s, afterpulsing probability 25%, afterpulsing time constant $1 \times 10^{6}\,\mathrm{s}^{-1}$ and dead time 1×10^{-7} s were used to generate the time tag file. Due to dead time and recharge time, the observed (effective) afterpulsing probability was calculated by the simulation algorithm itself to be 15.29%.

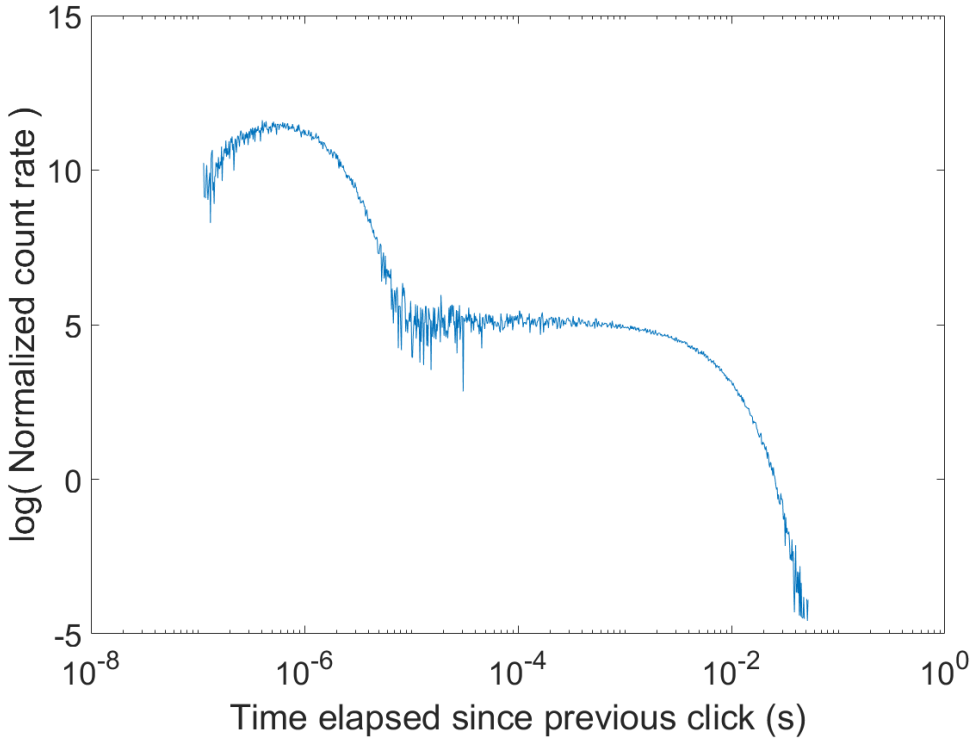
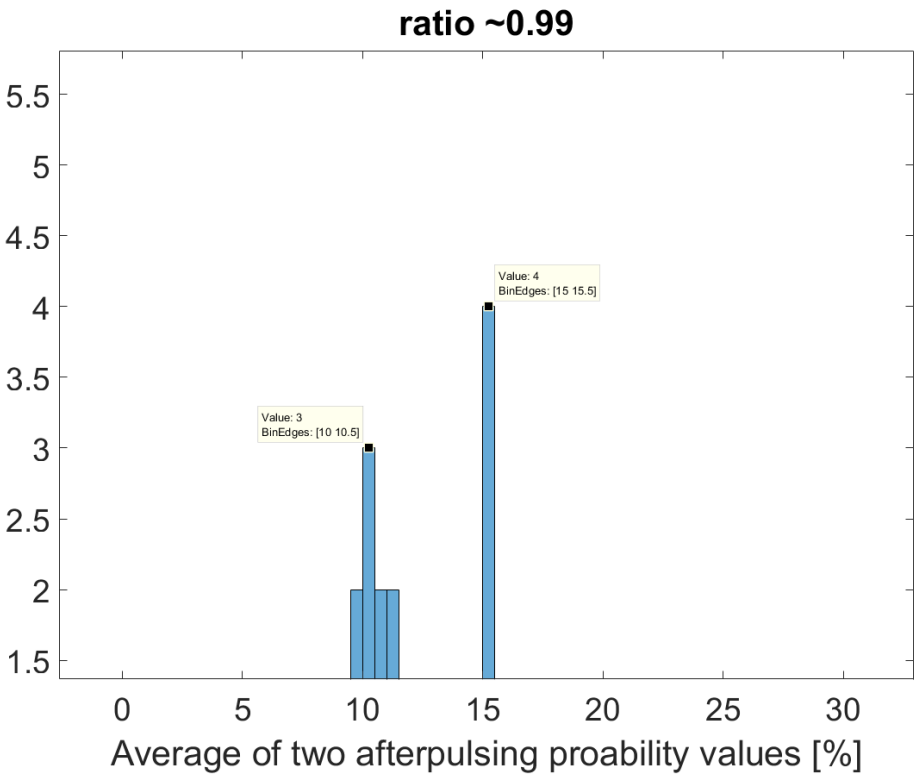


Figure 3.37: Probability density function of time difference between consecutive time tags using dark count measurement data. The time tag file was simulated in MATLAB. A dark count rate of 200 Hz, recharge time of 1×10^{-6} s, afterpulsing probability 25%, afterpulsing time constant $1 \times 10^{6}\,\mathrm{s}^{-1}$ and dead time 1×10^{-7} s were used to generate the time tag file. Due to dead time and recharge time, the observed (effective) afterpulsing probability was calculated by the simulation algorithm itself to be 15.29%.



99.png

Figure 3.38: Histogram of potential values of afterpulsing probability with the ratio of the values derived from two different approaches set to a threshold value of 0.99. The average of the two values is plotted.The time tag file was simulated in MATLAB. A dark count rate of 200 Hz, recharge time of 1×10^{-6} s, afterpulsing probability 25%, afterpulsing time constant $1 \times 10^{6}\,\mathrm{s}^{-1}$ and dead time 1×10^{-7} s were used to generate the time tag file. Due to dead time and recharge time, the observed (effective) afterpulsing probability was calculated by the simulation algorithm itself to be 15.29%.

95.png

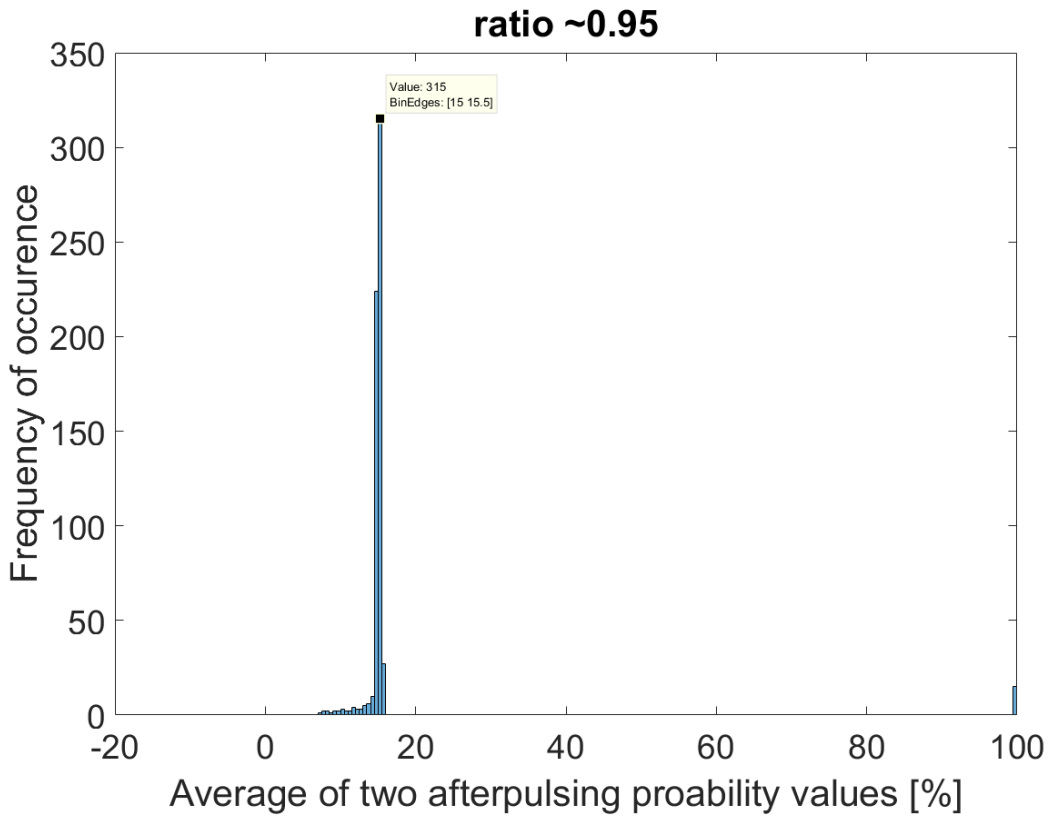
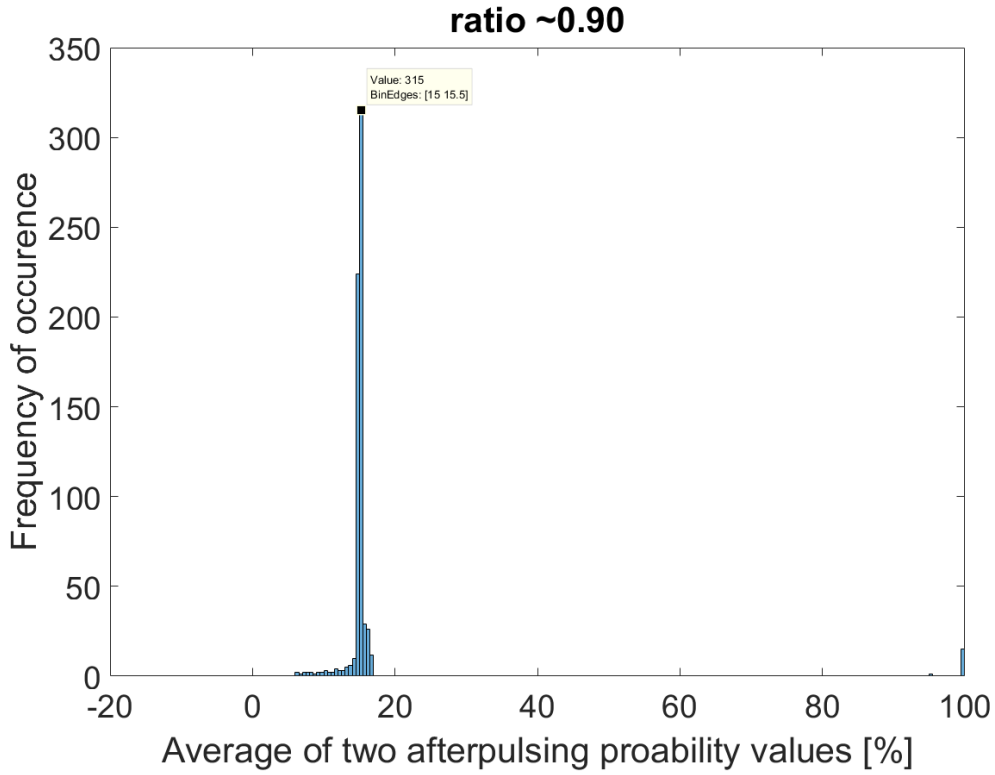
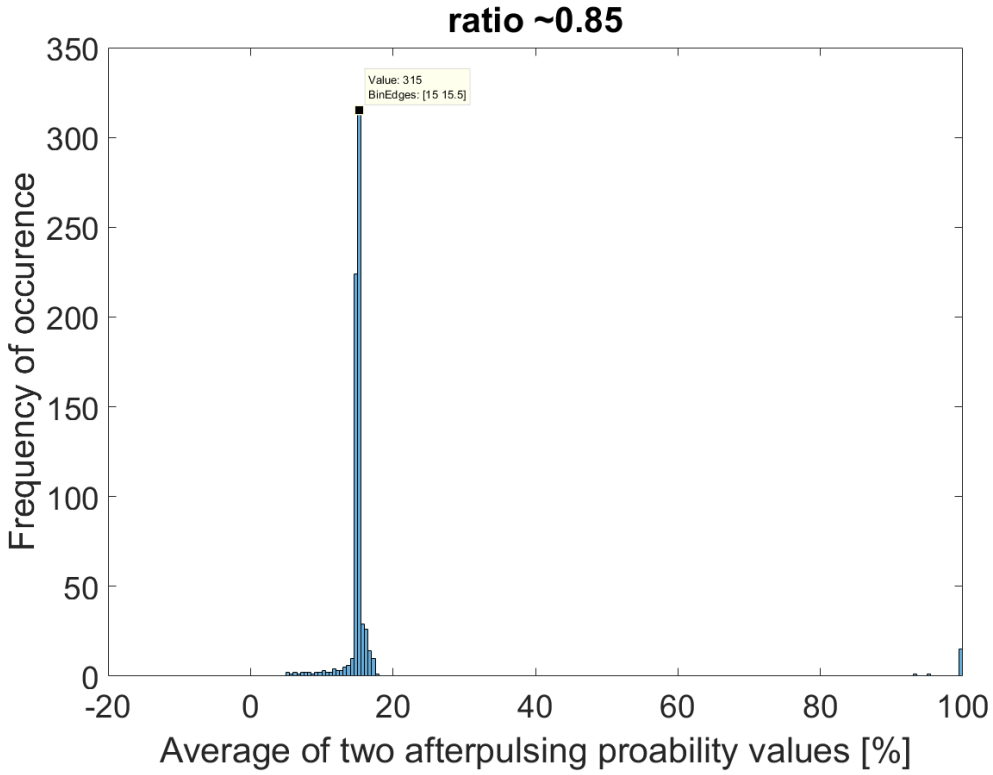


Figure 3.39: Histogram of potential values of afterpulsing probability with the ratio of the values derived from two different approaches set to a threshold value of 0.95. The average of the two values is plotted.The time tag file was simulated in MATLAB. A dark count rate of 200 Hz, recharge time of 1×10^{-6} s, afterpulsing probability 25%, afterpulsing time constant $1 \times 10^{6}\,\mathrm{s}^{-1}$ and dead time 1×10^{-7} s were used to generate the time tag file. Due to dead time and recharge time, the observed (effective) afterpulsing probability was calculated by the simulation algorithm itself to be 15.29%.



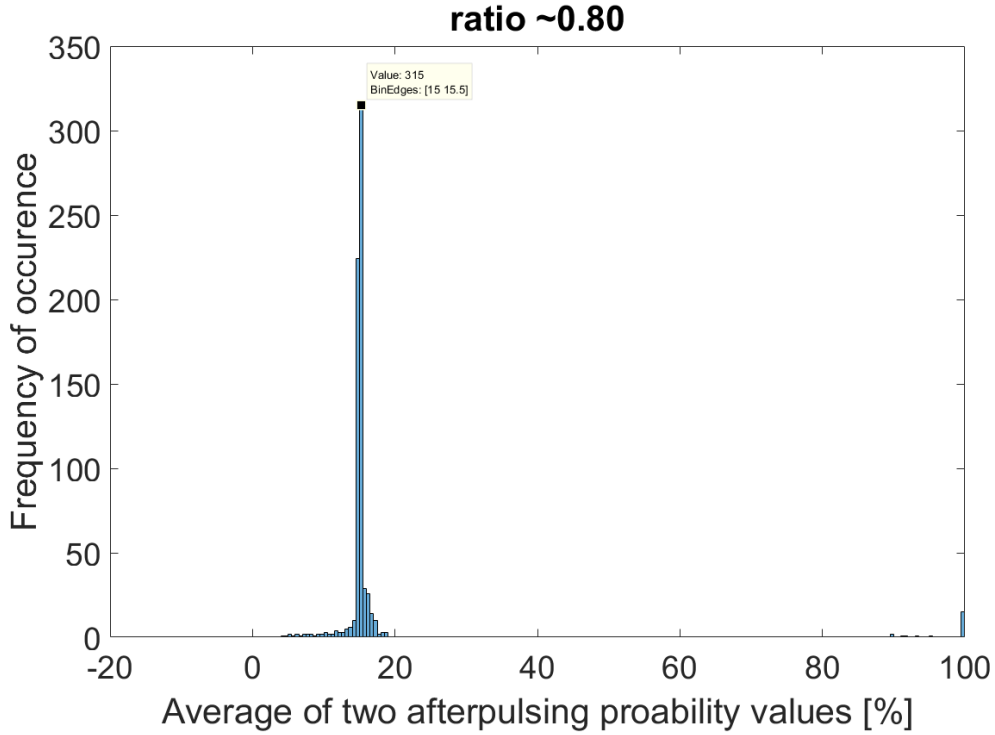
90.png

Figure 3.40: Histogram of potential values of afterpulsing probability with the ratio of the values derived from two different approaches set to a threshold value of 0.90. The average of the two values is plotted.The time tag file was simulated in MATLAB. A dark count rate of 200 Hz, recharge time of 1×10^{-6} s, afterpulsing probability 25%, afterpulsing time constant $1 \times 10^{6}\,\mathrm{s}^{-1}$ and dead time 1×10^{-7} s were used to generate the time tag file. Due to dead time and recharge time, the observed (effective) afterpulsing probability was calculated by the simulation algorithm itself to be 15.29%.



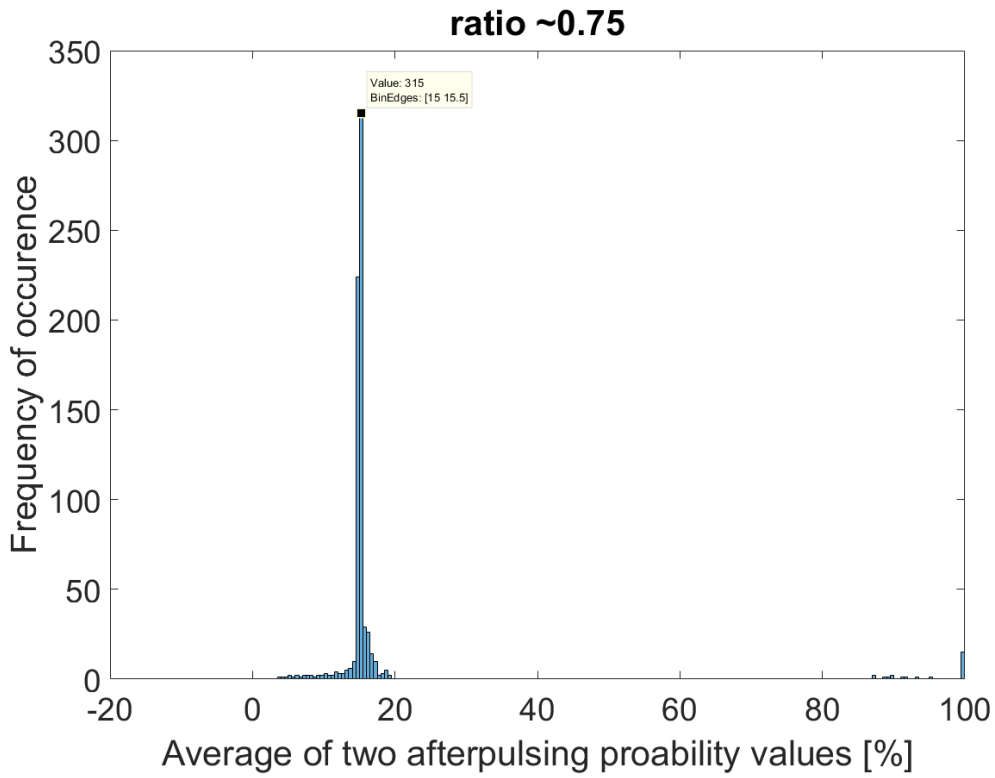
85.png

Figure 3.41: Histogram of potential values of afterpulsing probability with the ratio of the values derived from two different approaches set to a threshold value of 0.85. The average of the two values is plotted.The time tag file was simulated in MATLAB. A dark count rate of 200 Hz, recharge time of 1×10^{-6} s, afterpulsing probability 25%, afterpulsing time constant $1 \times 10^{6}\,\mathrm{s}^{-1}$ and dead time 1×10^{-7} s were used to generate the time tag file. Due to dead time and recharge time, the observed (effective) afterpulsing probability was calculated by the simulation algorithm itself to be 15.29%.



80.png

Figure 3.42: Histogram of potential values of afterpulsing probability with the ratio of the values derived from two different approaches set to a threshold value of 0.80. The average of the two values is plotted.The time tag file was simulated in MATLAB. A dark count rate of 200 Hz, recharge time of 1×10^{-6} s, afterpulsing probability 25%, afterpulsing time constant $1 \times 10^{6}\,\mathrm{s}^{-1}$ and dead time 1×10^{-7} s were used to generate the time tag file. Due to dead time and recharge time, the observed (effective) afterpulsing probability was calculated by the simulation algorithm itself to be 15.29%.



75.png

Figure 3.43: Histogram of potential values of afterpulsing probability with the ratio of the values derived from two different approaches set to a threshold value of 0.75. The average of the two values is plotted.The time tag file was simulated in MATLAB. A dark count rate of 200 Hz, recharge time of 1×10^{-6} s, afterpulsing probability 25%, afterpulsing time constant $1 \times 10^{6}\,\mathrm{s}^{-1}$ and dead time 1×10^{-7} s were used to generate the time tag file. Due to dead time and recharge time, the observed (effective) afterpulsing probability was calculated by the simulation algorithm itself to be 15.29%.

threshold; avg of ap1 and ap2.png

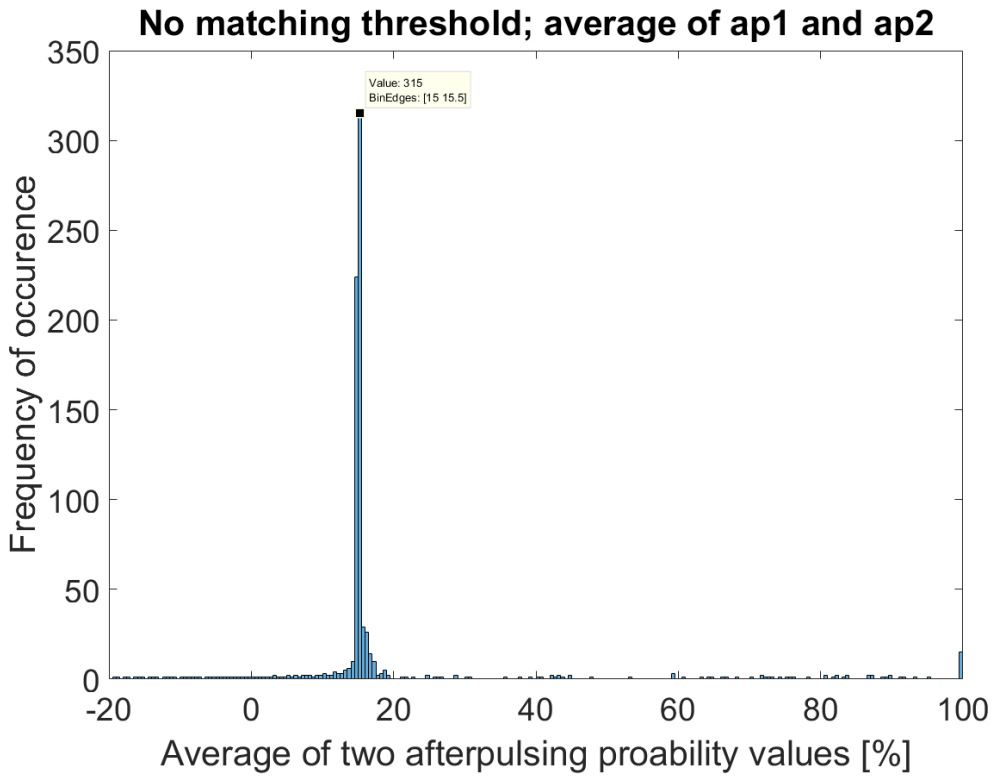


Figure 3.44: Histogram of potential values of afterpulsing probability with the ratio of the values derived from two different approaches not set to a threshold value. The average of the two values is plotted.The time tag file was simulated in MATLAB. A dark count rate of 200 Hz, recharge time of 1×10^{-6} s, afterpulsing probability 25%, afterpulsing time constant $1 \times 10^{6}\,\mathrm{s}^{-1}$ and dead time 1×10^{-7} s were used to generate the time tag file. Due to dead time and recharge time, the observed (effective) afterpulsing probability was calculated by the simulation algorithm itself to be 15.29%.

algorithms for afterpulsing probability. This was important because each one of them gave slightly different results (greater than any statistical uncertainty). To quantify this measure of reliability, I would suggest calculating the standard deviation of the peak at a fixed threshold and comparing these standard deviations values for different data sets. The smaller the standard deviation, the more reliable the final value of afterpulsing probability. I haven't implemented this yet in my algorithm but this is an area for future work.

Another observation in the results of the experimental tests tests is that the P.D.F. of next click seems to qualitatively have a different form from the one generated by simulation data. This could possibly be due to a simplistic afterpulsing model employed by the simulation data. One aspect is probably the assumption of at most one charge carrier being trapped in deep levels during an avalanche. The other aspect might be that there are multiple types of deep levels each with their own afterpulsing time constants and hence with different average detrapping life times. Nonetheless, these two aspects may not necessarily undermine the validity of the algorithm because I haven't assumed any particular afterpulsing model for the algorithm. Instead the only assumption was that the time scale for afterpulsing be orders of magnitude smaller than the time scale of thermal counts. This seems to be true in the experimental data as can be seen in figures 3.45 and 3.54. The location of the peaks along the time axis are roughly the time scales of the afterpulsing and thermal counts. So, I think the algorithm is still applicable.

Yet another observation is the the histogram of potential afterpulsing probabilities with no threshold seems to have a peak on the negative x-axis as well. This would correspond to negative afterpulsing probabilities which are not physical. Hence, I don't think this arises from any structure in the data. However, figuring out why this peak occurs might tell us something about how the algorithm interacts with the data. Also, one notices that the bin height at zero afterpulsing probability is negligible and the peak in the negative region is almost mirrored off the y-axis from the peak in the positive region. All these factors can be further studied to help improve the algorithm.

Figures 3.63 and 3.64 show the afterpulsing probability plotted against cumulative radiation fluence. DM1 which was irradiated up to $4 \times 10^9\,\mathrm{p/cm^2}$ doesn't show any correlation with respect cumulative radiation fluence. Also, there is no discernible trend with respect to annealing or incremental irradiation. However, DM2 shows an increase in afterpulsing probability with respect to cumulative radiation fluence which is evident when it is radiated well beyond the two year LEO cumulative proton fluence mark - it was irradiated up to $2 \times 10^{10}\,\mathrm{p/cm^2}$. Also, an interesting observation is that although afterpulsing probability showed a decrease in other cases, the decrease was very small and could be explained by statistical uncertainty in the data or calculation errors by the algorithm. However, all

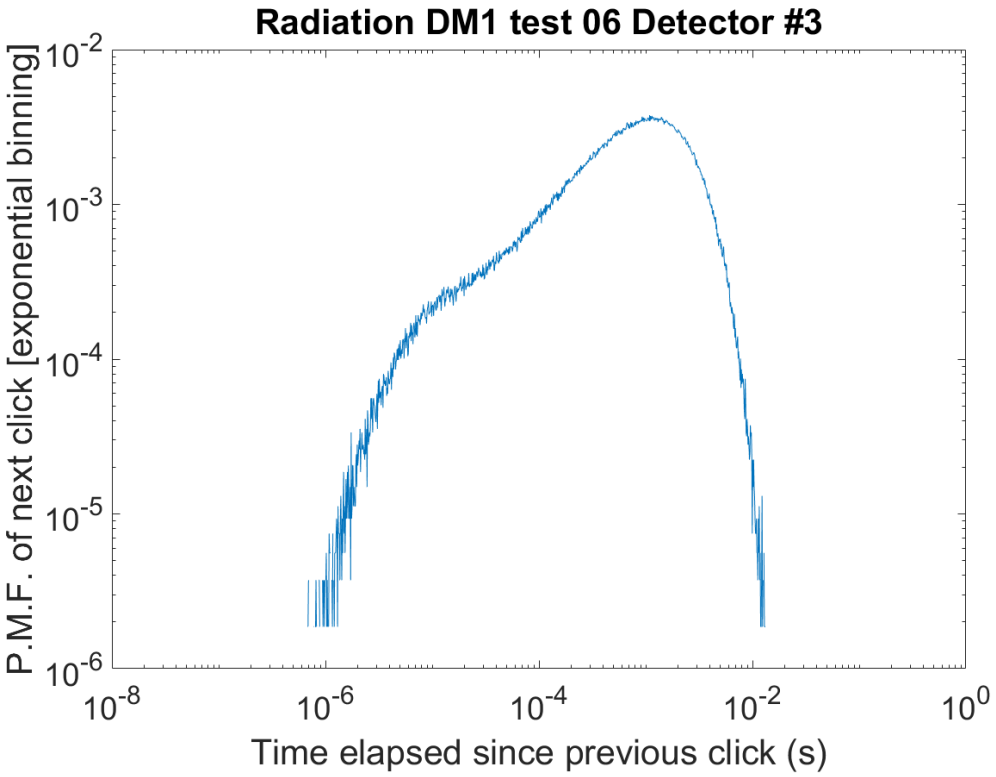


Figure 3.45: Probability mass function of time difference between consecutive time tags for Radiation DM1 test #06 detector #3. This test was conducted at the 2.68×10^9 p/cm^2 cumulative proton fluence mark at a detector temperature of $-80\,^\circ$C. The measured dark count rate (thermal count rate plus afterpulse count rate) was 892.6 Hz.

cases where afterpulsing probability strongly changed with annealing were increases. It is possible that annealing does increase the afterpulsing probability.

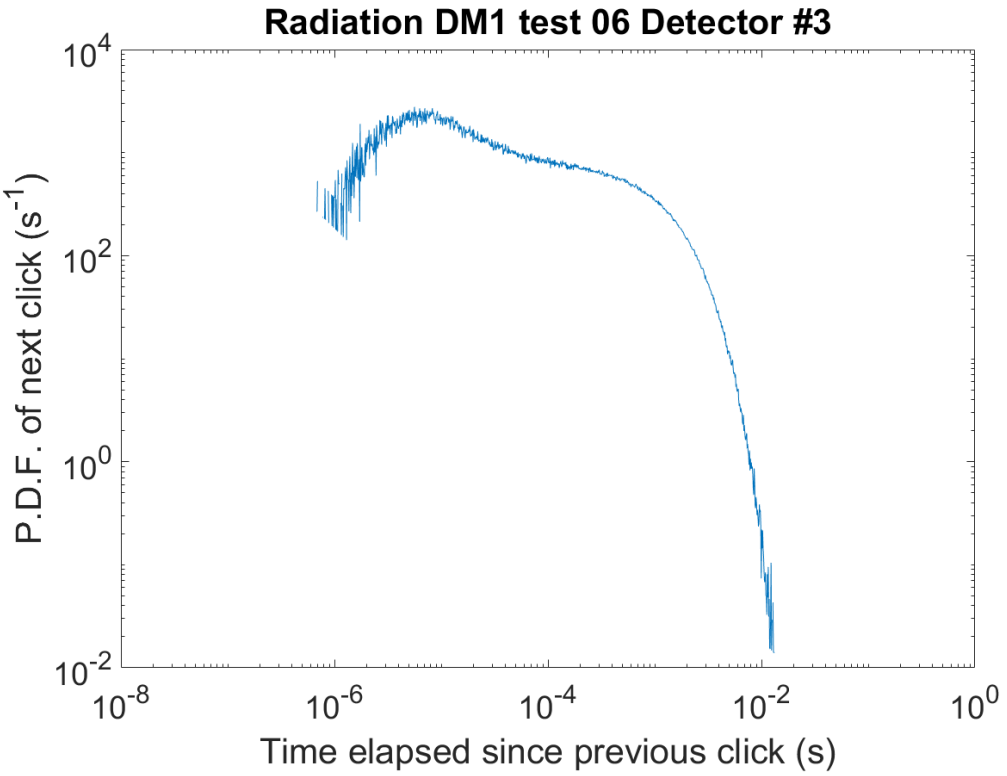


Figure 3.46: Probability density function of time difference between consecutive time tags for Radiation DM1 test #06 detector #3. This test was conducted at the $2.68 \times 10^9\,\mathrm{p/cm^2}$ cumulative proton fluence mark at a detector temperature of $-80\,^{\circ}\mathrm{C}$. The measured dark count rate (thermal count rate plus afterpulse count rate) was 892.6 Hz.

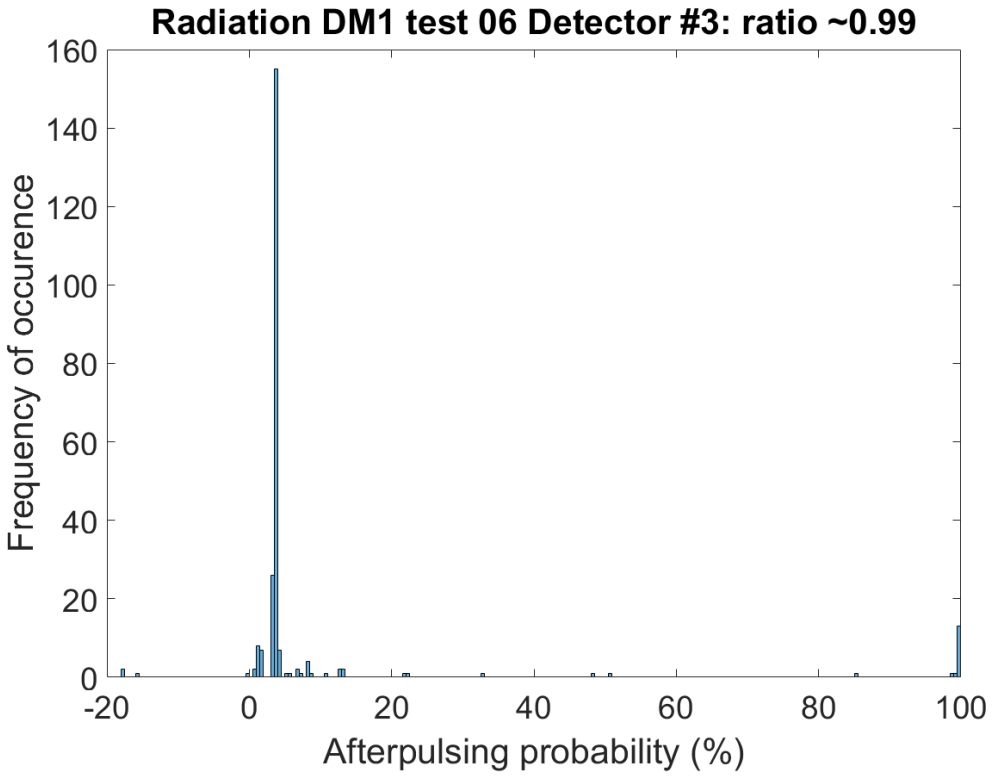


Figure 3.47: Histogram of potential values of afterpulsing probability with the ratio of the values derived from two different approaches set to a threshold value of 0.99. The average of the two values is plotted. This test was conducted at the $2.68 \times 10^9\,\mathrm{p/cm^2}$ cumulative proton fluence mark at a detector temperature of $-80\,^{\circ}\mathrm{C}$. The measured dark count rate (thermal count rate plus afterpulse count rate) was 892.6 Hz.

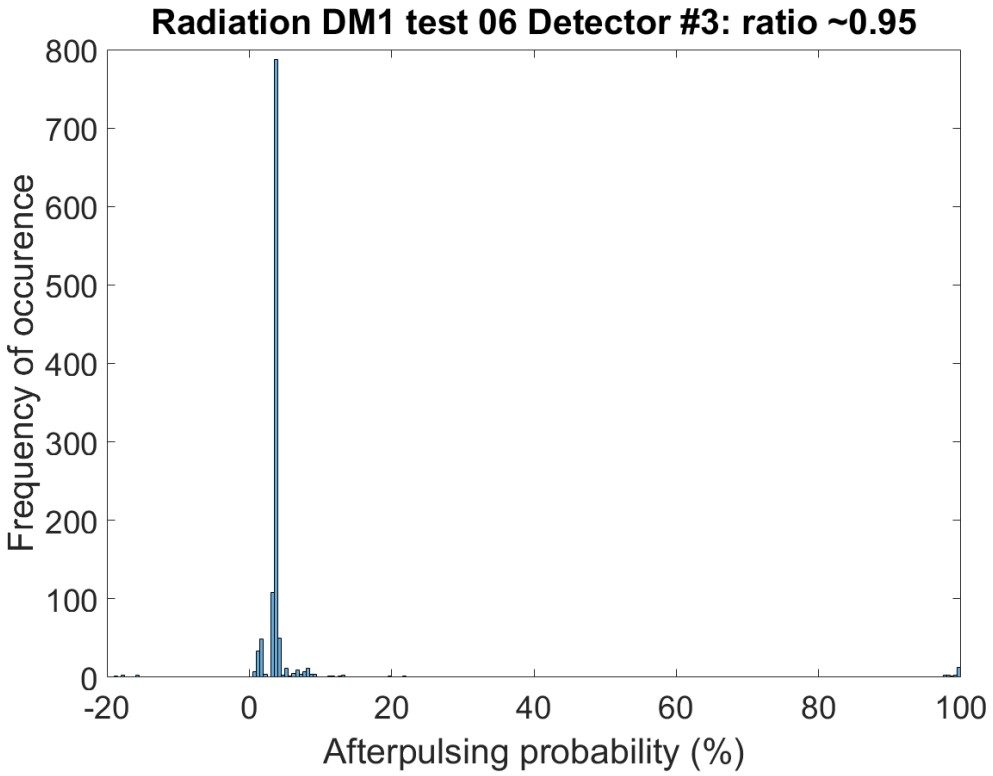


Figure 3.48: Histogram of potential values of afterpulsing probability with the ratio of the values derived from two different approaches set to a threshold value of 0.95. The average of the two values is plotted. This test was conducted at the $2.68 \times 10^9\,\mathrm{p/cm^2}$ cumulative proton fluence mark at a detector temperature of $-80\,^{\circ}\mathrm{C}$. The measured dark count rate (thermal count rate plus afterpulse count rate) was 892.6 Hz.

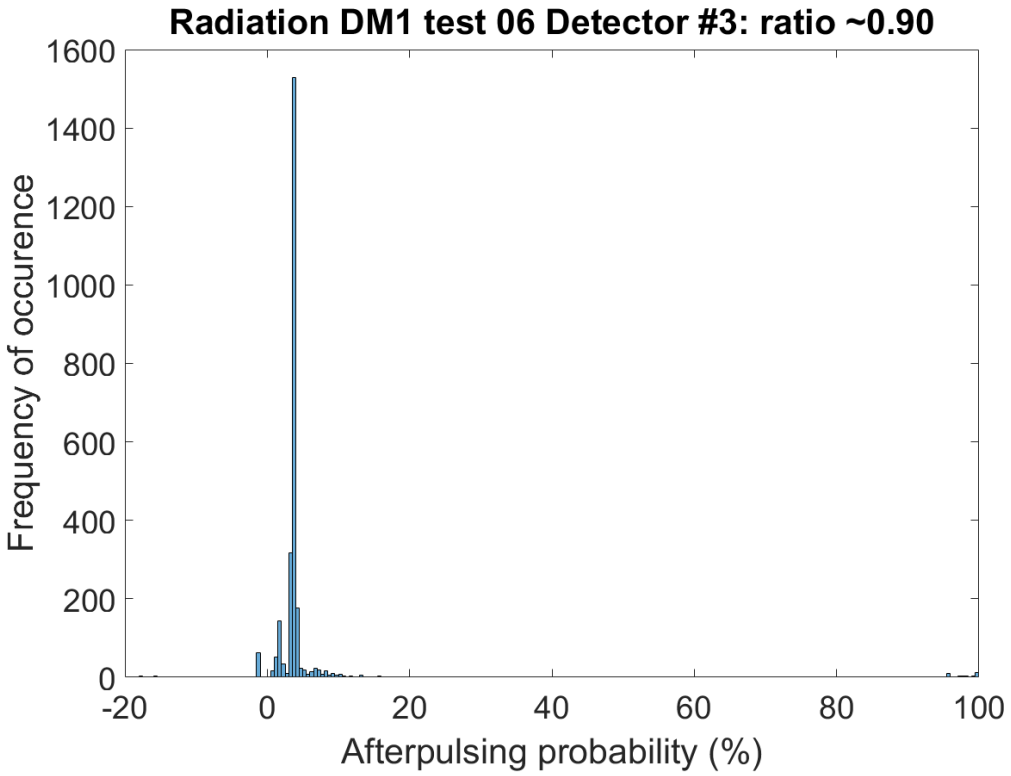


Figure 3.49: Histogram of potential values of afterpulsing probability with the ratio of the values derived from two different approaches set to a threshold value of 0.90. The average of the two values is plotted. This test was conducted at the $2.68 \times 10^9\,\mathrm{p/cm^2}$ cumulative proton fluence mark at a detector temperature of $-80\,^{\circ}\mathrm{C}$. The measured dark count rate (thermal count rate plus afterpulse count rate) was 892.6 Hz.

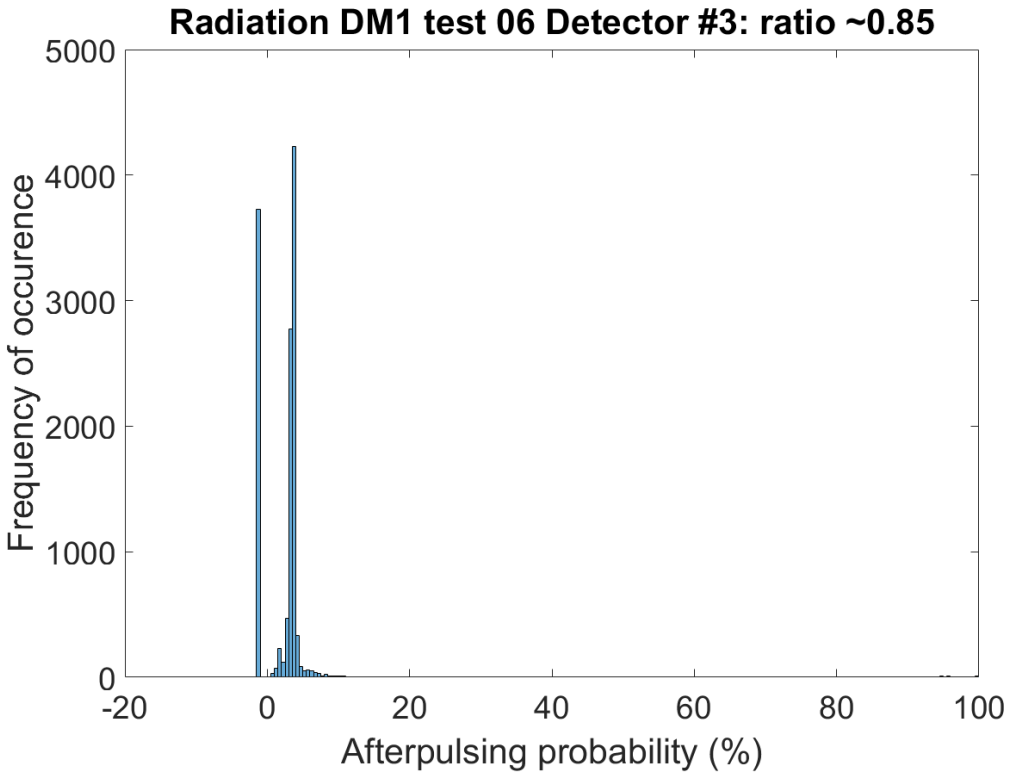


Figure 3.50: Histogram of potential values of afterpulsing probability with the ratio of the values derived from two different approaches set to a threshold value of 0.85. The average of the two values is plotted. This test was conducted at the $2.68 \times 10^9\,\mathrm{p/cm^2}$ cumulative proton fluence mark at a detector temperature of $-80\,^{\circ}\mathrm{C}$. The measured dark count rate (thermal count rate plus afterpulse count rate) was 892.6 Hz.

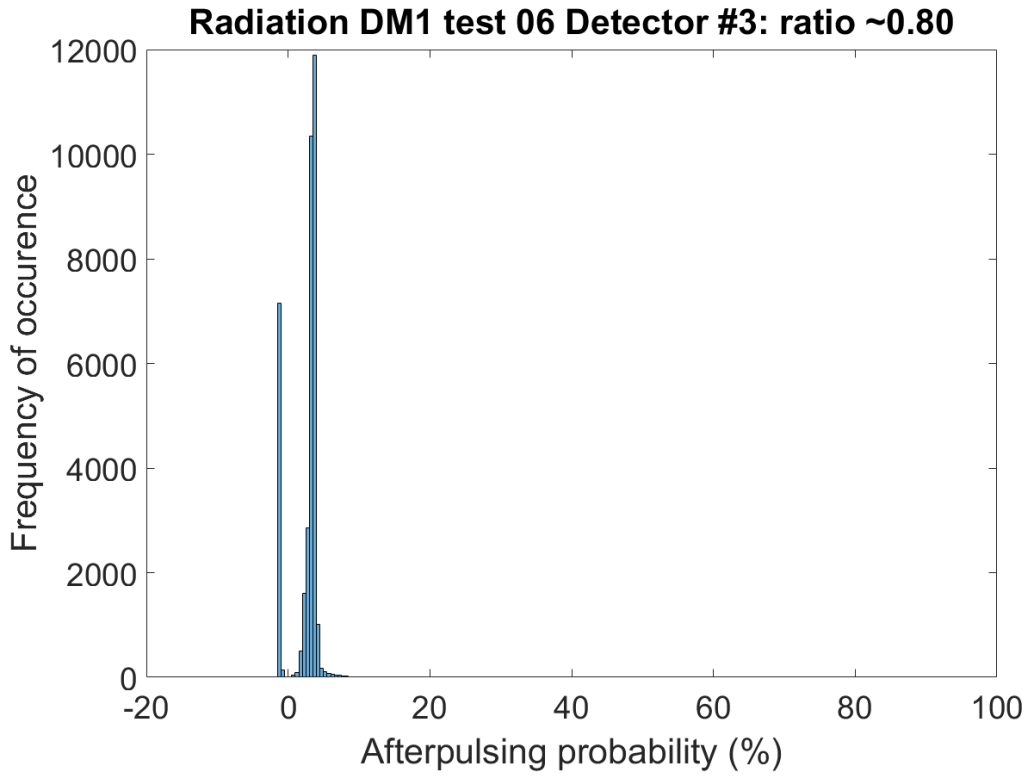


Figure 3.51: Histogram of potential values of afterpulsing probability with the ratio of the values derived from two different approaches set to a threshold value of 0.80. The average of the two values is plotted. This test was conducted at the $2.68 \times 10^9\,\mathrm{p/cm^2}$ cumulative proton fluence mark at a detector temperature of $-80\,^{\circ}\mathrm{C}$. The measured dark count rate (thermal count rate plus afterpulse count rate) was 892.6 Hz.

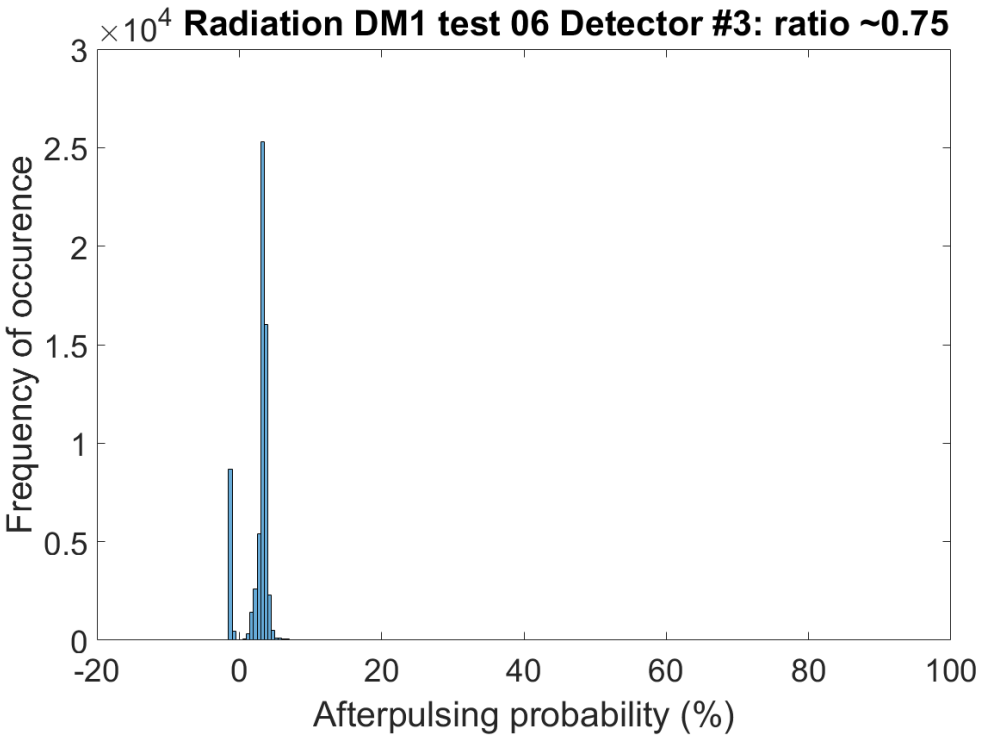


Figure 3.52: Histogram of potential values of afterpulsing probability with the ratio of the values derived from two different approaches set to a threshold value of 0.75. The average of the two values is plotted. This test was conducted at the 2.68×10^9 p/cm^2 cumulative proton fluence mark at a detector temperature of $-80\,^{\circ}$C. The measured dark count rate (thermal count rate plus afterpulse count rate) was 892.6 Hz.

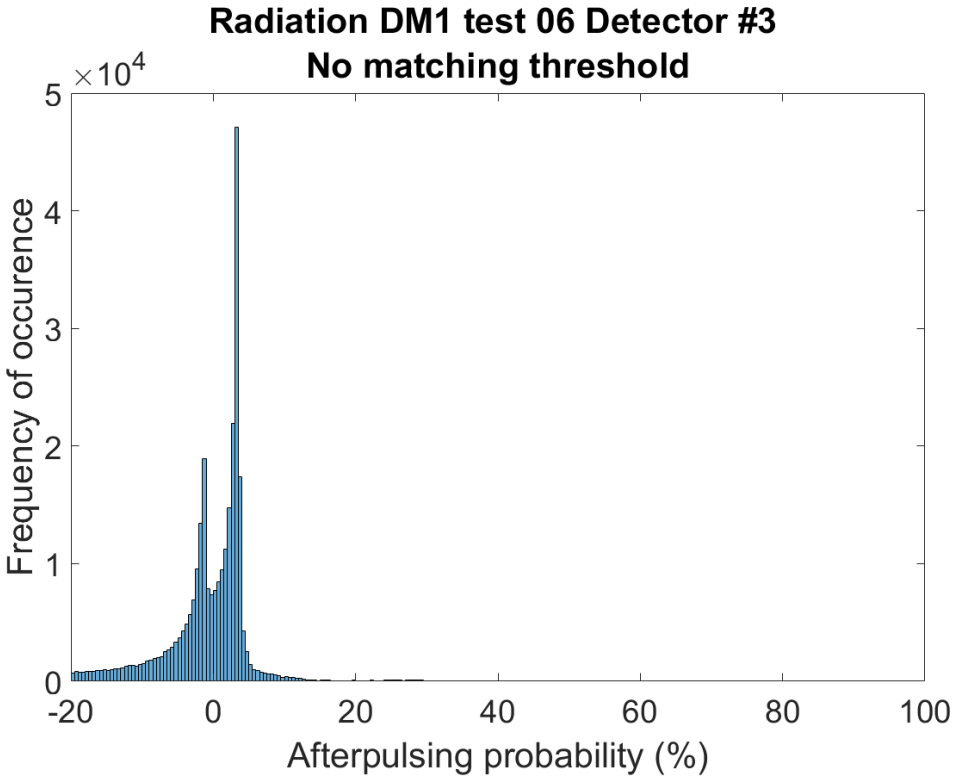


Figure 3.53: Histogram of potential values of afterpulsing probability with the ratio of the values derived from two different approaches set with no imposed threshold value on the ratio. The average of the two values is plotted. This test was conducted at the 2.68×10^9 p/cm^2 cumulative proton fluence mark at a detector temperature of $-80\,^{\circ}$C. The measured dark count rate (thermal count rate plus afterpulse count rate) was 892.6 Hz.

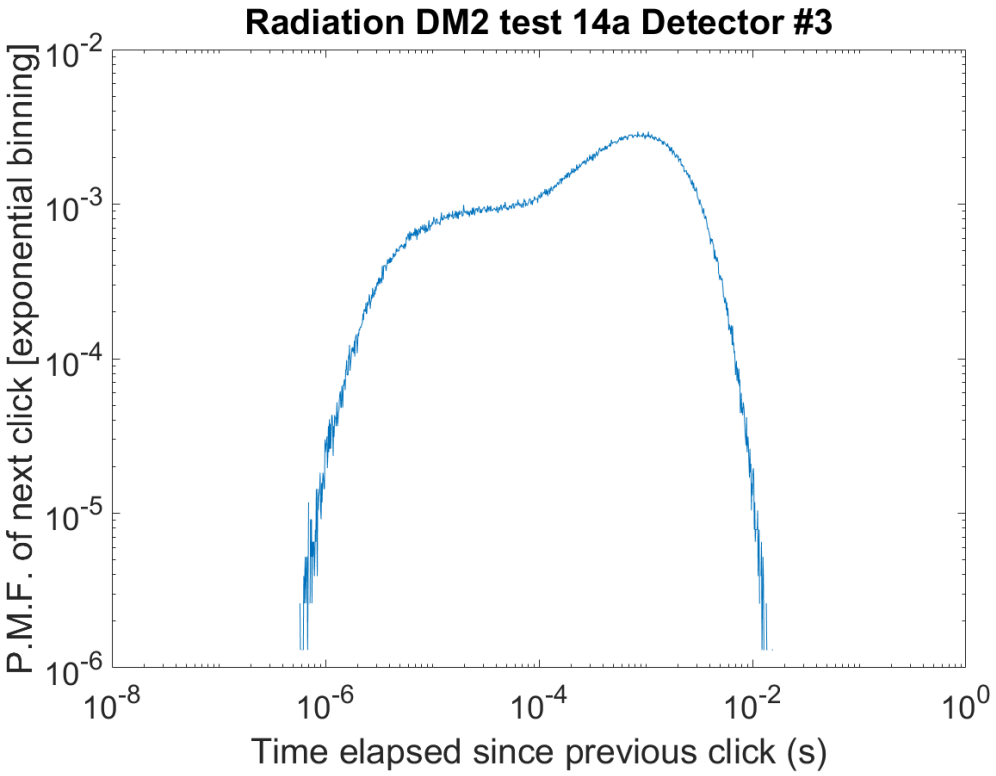


Figure 3.54: Probability mass function of time difference between consecutive time tags for Radiation DM2 test #14a detector #3. This test was conducted at the $1.4 \times 10^{10}\,\mathrm{p/cm^2}$ cumulative proton fluence mark at a detector temperature of $-80\,^{\circ}\mathrm{C}$. The measured dark count rate (thermal count rate plus afterpulse count rate) was 1275.3 Hz.

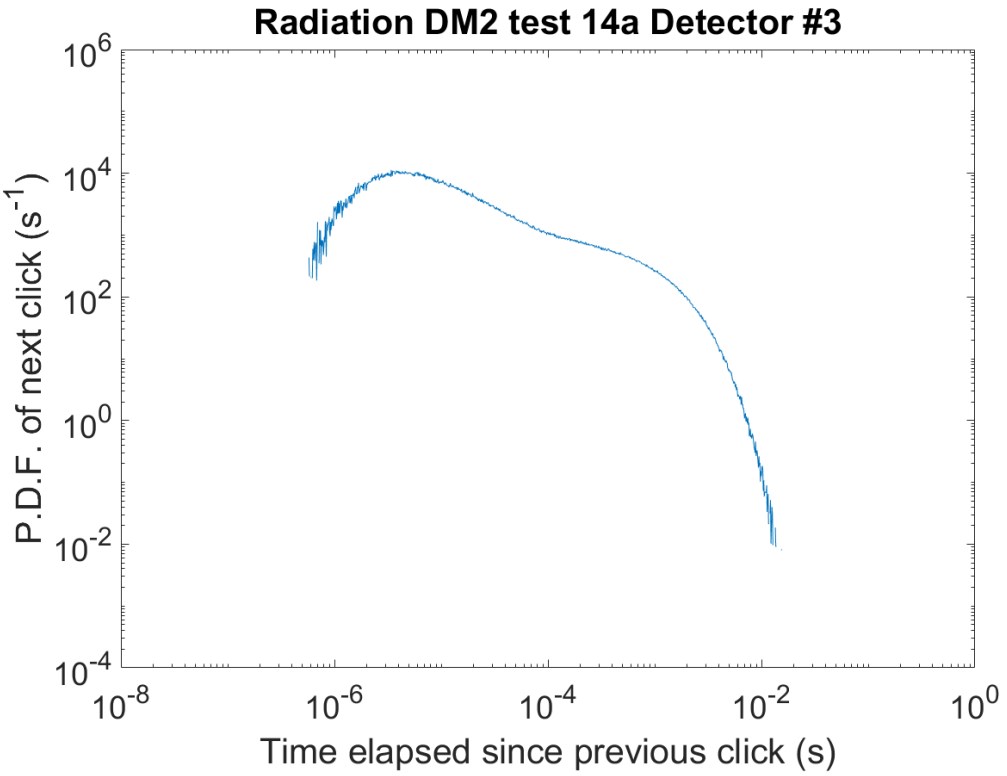


Figure 3.55: Probability density function of time difference between consecutive time tags for Radiation DM2 test #14a detector #3. This test was conducted at the 1.4×10^{10} p/cm^2 cumulative proton fluence mark at a detector temperature of $-80\,^{\circ}$C. The measured dark count rate (thermal count rate plus afterpulse count rate) was 1275.3 Hz.

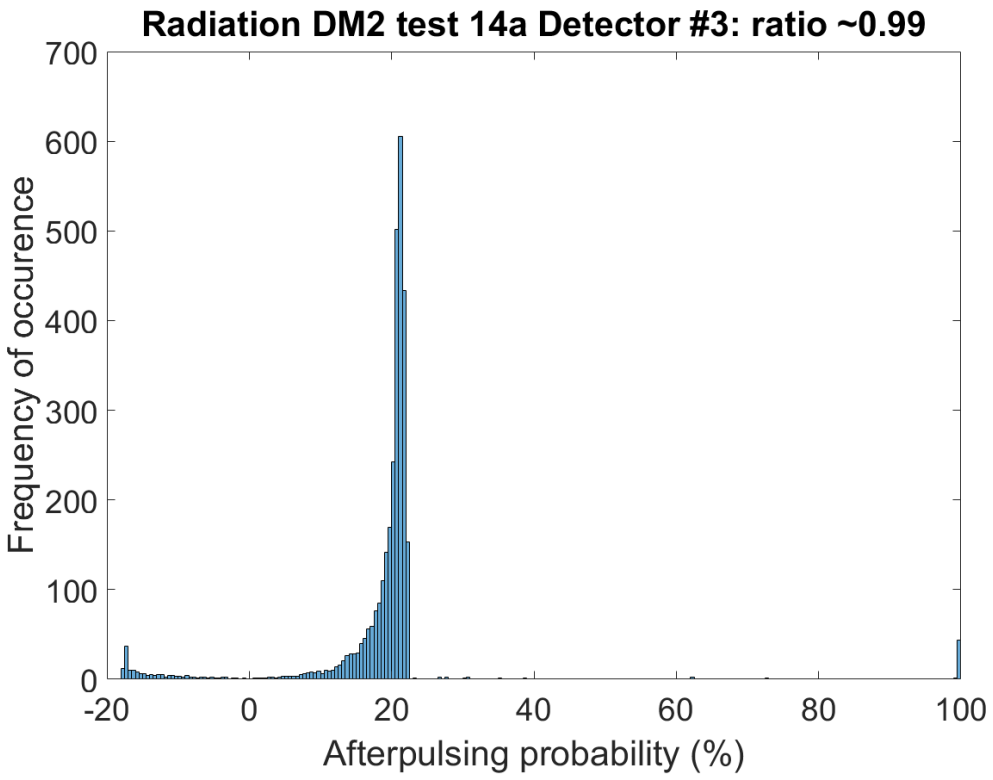


Figure 3.56: Histogram of potential values of afterpulsing probability with the ratio of the values derived from two different approaches set to a threshold value of 0.99 for Radiation DM2 test 14a detector 3. The average of the two values is plotted. This test was conducted at the $1.4 \times 10^{10}\,\mathrm{p/cm^2}$ cumulative proton fluence mark at a detector temperature of $-80\,^{\circ}\mathrm{C}$. The measured dark count rate (thermal count rate plus afterpulse count rate) was 1275.3 Hz.

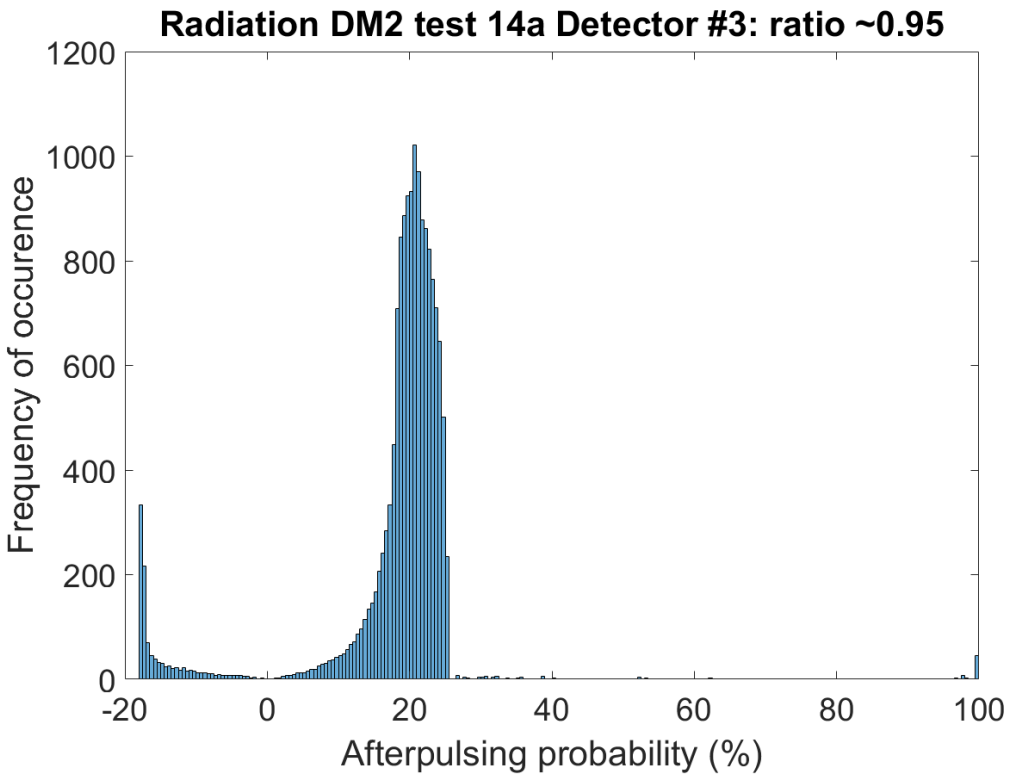


Figure 3.57: Histogram of potential values of afterpulsing probability with the ratio of the values derived from two different approaches set to a threshold value of 0.95 for Radiation DM2 test 14a detector 3. The average of the two values is plotted. This test was conducted at the $1.4 \times 10^{10}\,\mathrm{p/cm^2}$ cumulative proton fluence mark at a detector temperature of $-80\,^{\circ}\mathrm{C}$. The measured dark count rate (thermal count rate plus afterpulse count rate) was 1275.3 Hz.

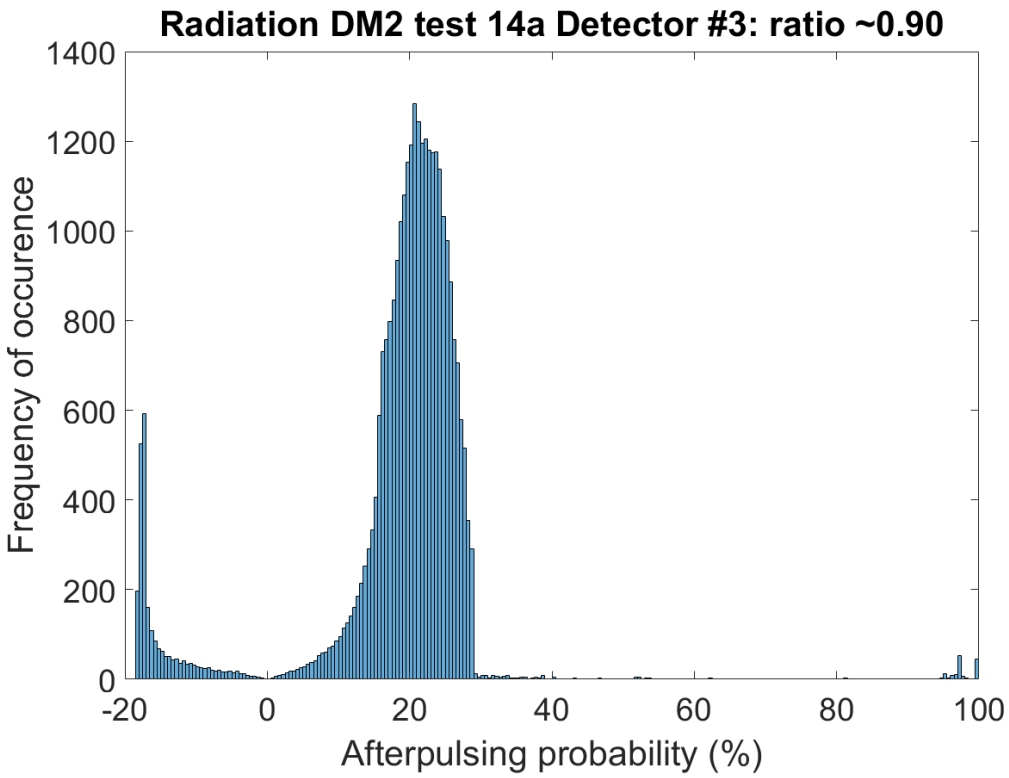


Figure 3.58: Histogram of potential values of afterpulsing probability with the ratio of the values derived from two different approaches set to a threshold value of 0.90 for Radiation DM2 test 14a detector 3. The average of the two values is plotted. This test was conducted at the $1.4 \times 10^{10}\,\mathrm{p/cm^2}$ cumulative proton fluence mark at a detector temperature of $-80\,^{\circ}\mathrm{C}$. The measured dark count rate (thermal count rate plus afterpulse count rate) was 1275.3 Hz.

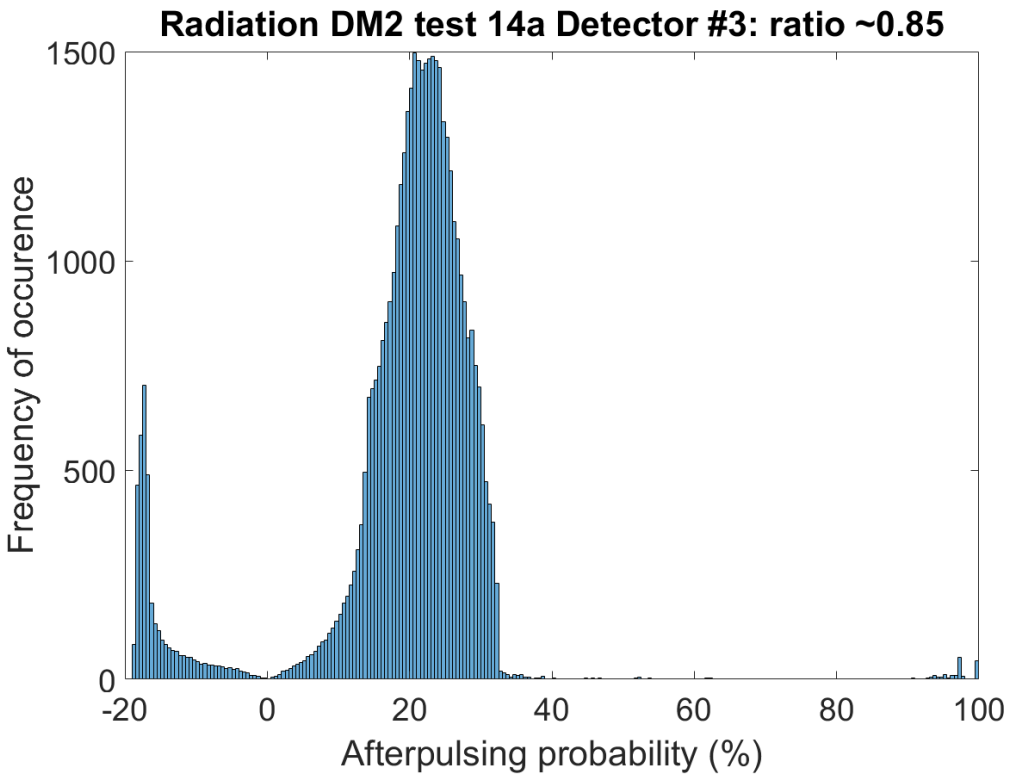


Figure 3.59: Histogram of potential values of afterpulsing probability with the ratio of the values derived from two different approaches set to a threshold value of 0.85 for Radiation DM2 test 14a detector 3. The average of the two values is plotted. This test was conducted at the $1.4 \times 10^{10}\,\mathrm{p/cm^2}$ cumulative proton fluence mark at a detector temperature of $-80\,^{\circ}\mathrm{C}$. The measured dark count rate (thermal count rate plus afterpulse count rate) was 1275.3 Hz.

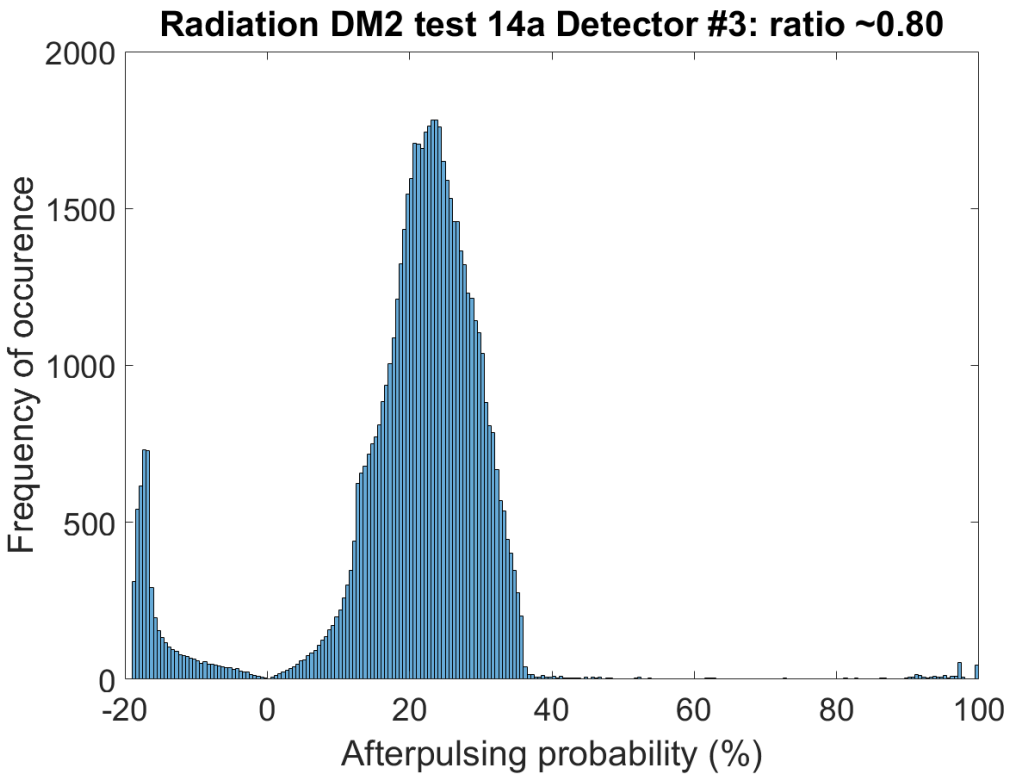


Figure 3.60: Histogram of potential values of afterpulsing probability with the ratio of the values derived from two different approaches set to a threshold value of 0.80 for Radiation DM2 test 14a detector 3. The average of the two values is plotted. This test was conducted at the $1.4 \times 10^{10}\,\mathrm{p/cm^2}$ cumulative proton fluence mark at a detector temperature of $-80\,^{\circ}\mathrm{C}$. The measured dark count rate (thermal count rate plus afterpulse count rate) was 1275.3 Hz.

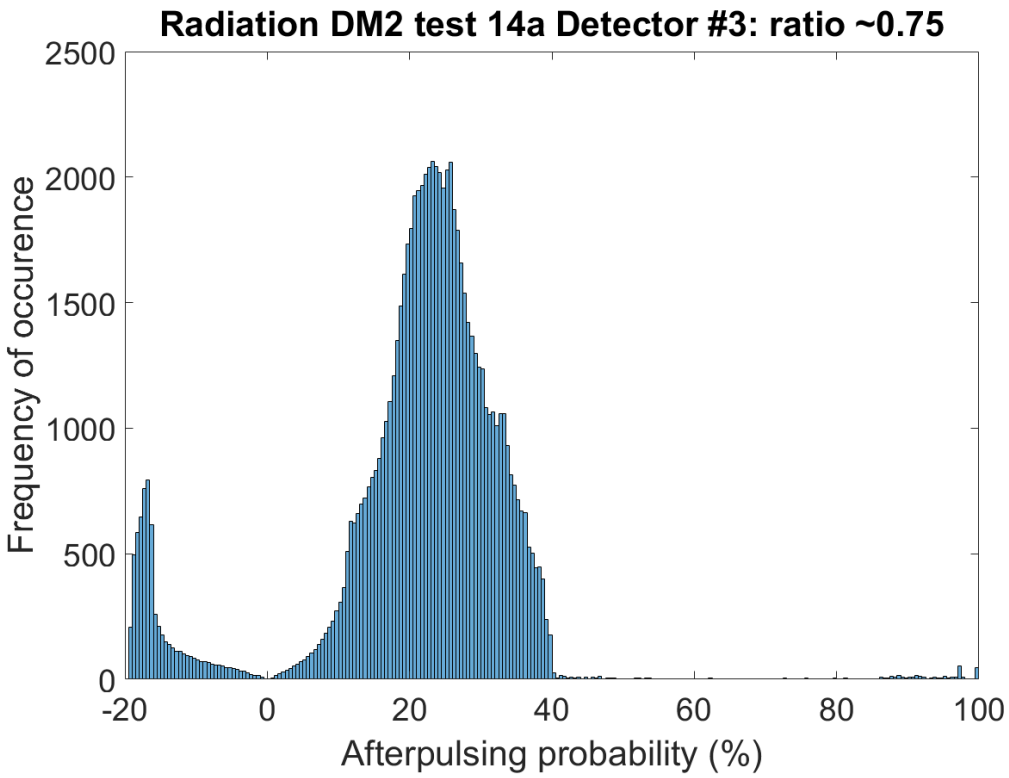


Figure 3.61: Histogram of potential values of afterpulsing probability with the ratio of the values derived from two different approaches set to a threshold value of 0.75 for Radiation DM2 test 14a detector 3. The average of the two values is plotted. This test was conducted at the $1.4 \times 10^{10}\,\mathrm{p/cm^2}$ cumulative proton fluence mark at a detector temperature of $-80\,^{\circ}\mathrm{C}$. The measured dark count rate (thermal count rate plus afterpulse count rate) was 1275.3 Hz.

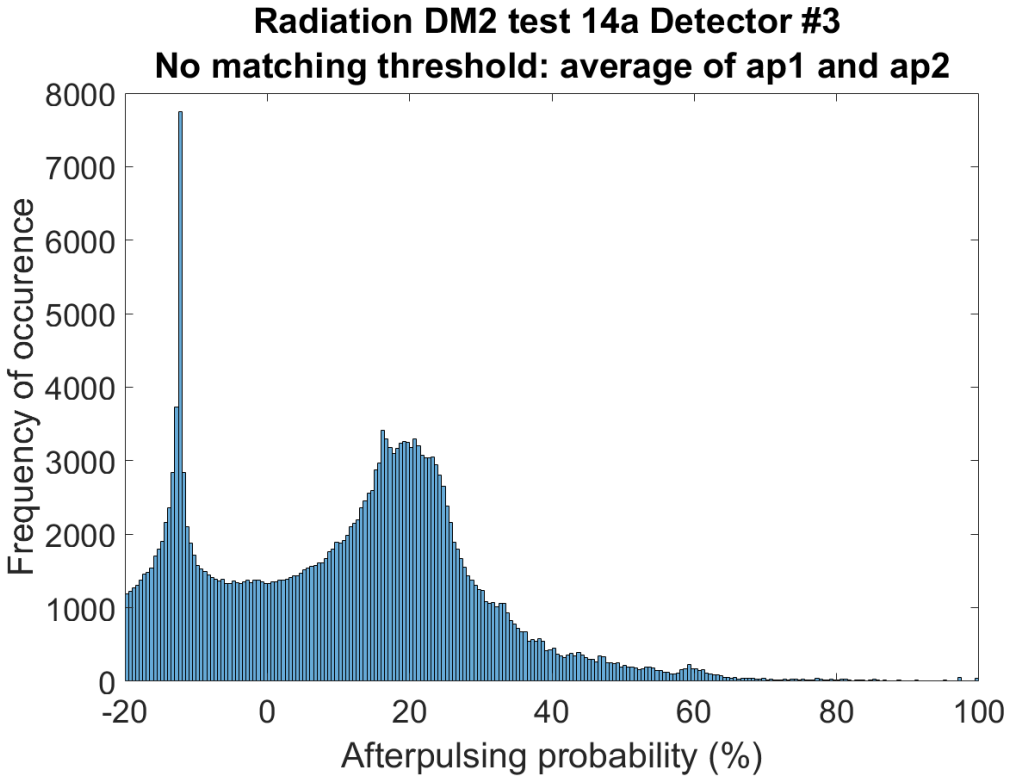


Figure 3.62: Histogram of potential values of afterpulsing probability with the ratio of the values derived from two different approaches with no threshold value imposed on the ratio, for Radiation DM2 test 14a detector 3. The average of the two values is plotted. This test was conducted at the $1.4 \times 10^{10}\,\mathrm{p/cm^2}$ cumulative proton fluence mark at a detector temperature of $-80\,^{\circ}\mathrm{C}$. The measured dark count rate (thermal count rate plus afterpulse count rate) was 1275.3 Hz.

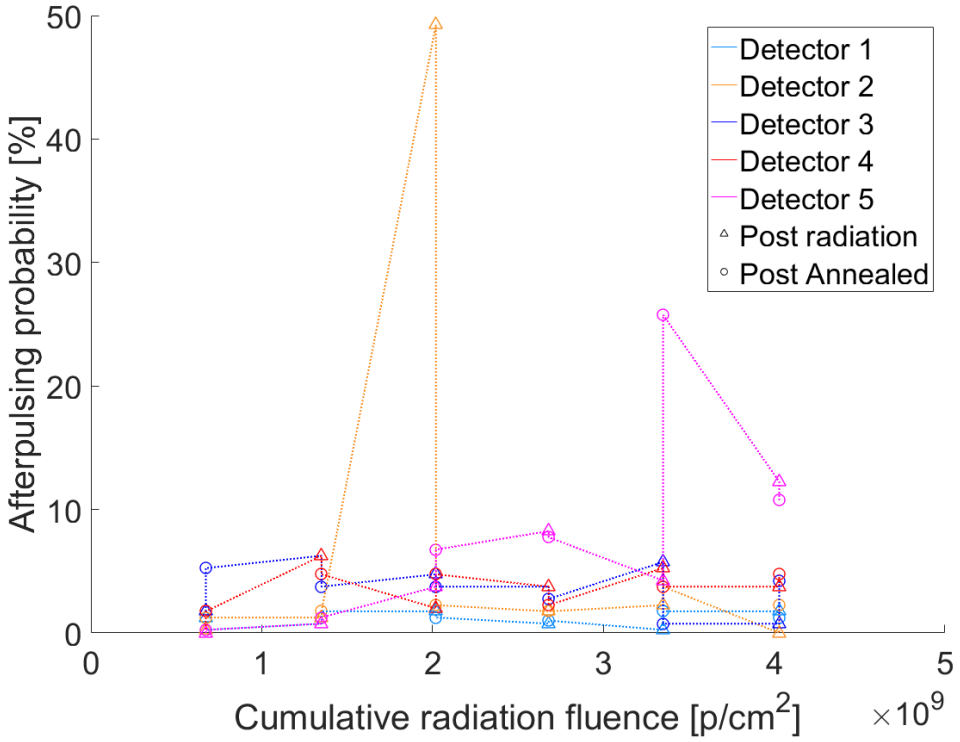


Figure 3.63: Afterpulsing probability is plotted against cumulative radiation fluence. Time difference between consecutive time tags is histogrammed using dark count measurement data for DM 1. An algorithm calculates afterpulsing probability. All tests were conducted at an APD temperature of $-80\,°\mathrm{C}$.

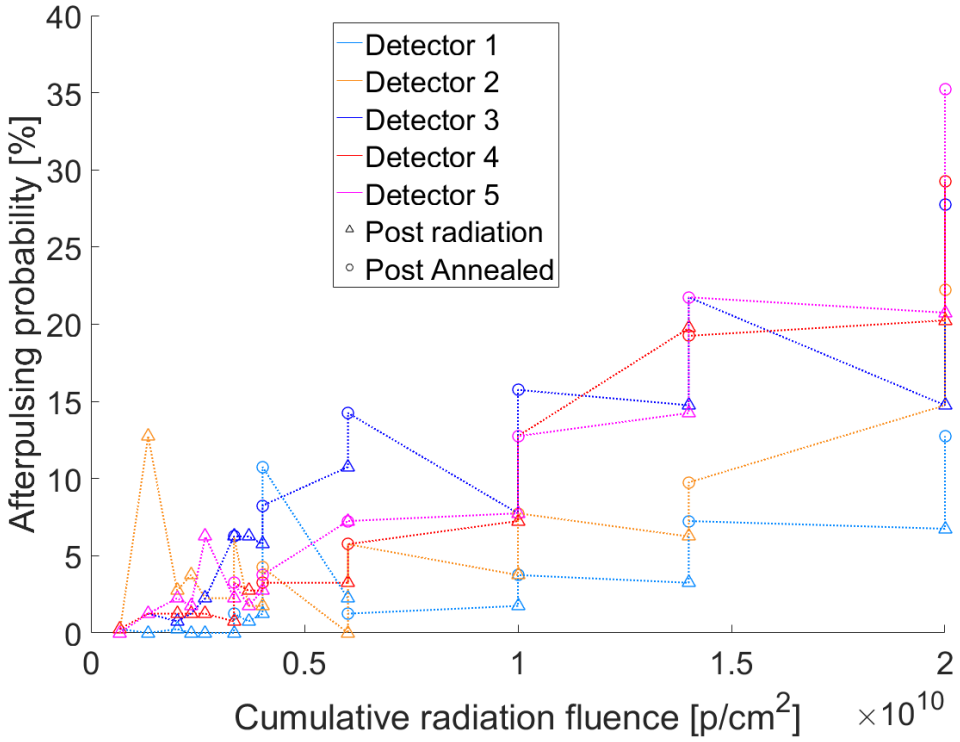


Figure 3.64: Afterpulsing probability is plotted against cumulative radiation fluence. Time difference between consecutive time tags is histogrammed using dark count measurement data for DM 2. An algorithm calculates afterpulsing probability. All tests were conducted at an APD temperature of $-80\,^{\circ}\mathrm{C}$.

3.4 Conclusion

The effects of repeated application of proton radiation and annealing on Silicon avalanche photodiodes was studied. Each of the five detectors in the DM was characterized for different parameters. Of these, the output pulse characteristics (i.e., the amplitude and FWHM), recharge time, detection efficiency, timing jitter and saturation value were not affected by annealing or proton radiation. The breakdown voltage was found to have a positive linear correlation with respect to detector temperature. Three different afterpulsing algorithms were discussed. In general, the afterpulsing probability increased with respect cumulative radiation fluence. However, it was difficult to ascertain if the radiation for annealing itself caused this increase. The algorithm that used the time difference between consecutive time tags especially shows that annealing might have a tendency to increase afterpulsing probability. The algorithm that used the efficiency measurement data also seems to corroborate this, though to a slighter lesser extent.

The dark count rates increased with proton radiation with a few instances of exception to this rule. The increase in dark count rate per unit fluence of proton radiation was shown to be independent of the cumulative radiation fluence mark - the quantity remained more or less the same over the course of the tests. Dark count rates in general decreased with thermal annealing, again with a few exceptions to this rule. Annealing was successful in lowering the dark count rates below 500 Hz on most detectors up to the 3 year LEO equivalent cumulative radiation fluence (6×10^9 p/cm^2), well beyond the projected two year lifetime of the satellite. However, the post-annealing dark count rates steadily increase with cumulative proton radiation, showing that annealing was not able to completely mitigate the effects of radiation. The annealing reduction factor (ARF) was used as a measure to gauge the effectiveness of annealing. The ARF value was the highest the first time annealing was used on a given detector. Subsequent instances had ARF values uncorrelated with cumulative proton fluence as well as pre-annealed dark count rates. Typical ARF values of 2-5 were observed. Finally, lower temperatures (as low as $-110\,^\circ$C) were successfully explored as a means to decrease dark count rates to below 100 Hz at the 1×10^{10} p/cm^2 mark. The current temperature feedback device however will need to be modified in order to operate at such low temperatures in closed loop control.

Two approaches towards annealing were considered - one where the DM was annealed after every incremental exposure to proton radiation and the other where the DM was annealed only if the post-radiation dark count rate increased beyond a threshold value of 2 kHz. The approach of annealing when the dark count rate exceeded the threshold values showed slightly lower post-annealing dark count rates as well as slightly higher ARF values

suggesting that frequent annealing might not be beneficial. Since this advantage was small (less than an order of magnitude) and could only be compared on two tests, more tests would be needed to make a strong inference. Dark count rates seemed to decrease the fastest during the earlier in the annealing phase rather than later.

The detector parameters seem to be in operating range for the entirety of the experiments. The main conclusion was that although the dark count rates increased with temperature, annealing was effective in mitigate this, and so was lower operating detector temperatures. Some anomalies were detected, the effect of detector circuit temperature on the detector parameters must be explored in detail.

Chapter 4

Characterization of Avalanche Photo Diode in Thermal Vacuum

4.1 Motivation

The APD's are expected to operate in outer space as part of a QKD satellite. The satellite is proposed be in a low Earth orbit (LEO). This means that the DM will be exposed to vacuum. One needs to consider how the DM as a whole would behave in vacuum. AS such the APD's parameters aren't expected to change in vacuum. However, it is still important to ensure that these parameters don't drift under in vacuum and that any inferences drawn from the parameters measured at our laboratory at IQC in section 2 or during the radiation experiments mentioned in section 3 are valid in vacuum as well. Towards this end, a thermal vacuum (TVAC) experiment was devised to test the effects of vacuum on the DM's operation. The word *thermal* denotes that in vacuum, we have the capability of controlling the temperature of the radiator or exterior of the DM to simulate the thermal conditions when the DM on the satellite is facing away from the Sun or is in the shadow of the Earth. Some aspects of the setup which could potentially behave differently is the absence of convective cooling. In the gaseous medium of the Earth's atmosphere, convection aids to cool the DM - the APD's and their associated circuitry. However, in vacuum convective cooling is absent. Depending on the rate of radiative heat dissipation and internal heat generation, this could potentially be a concern. Also, there DM has high voltage circuit - we apply a bias voltage to the APD of the order of 300 V. Under such circumstances, there is a small likelihood to have arcing. Primarily arcing requires a gaseous medium - although vacuum arcing does exist, but this would be common to both atmospheric conditions and vacuum conditions. In the absence of a gaseous medium, the

chances for arcing are lesser. This is beneficial but it would still be nice to experimentally see the effects of vacuum.

4.2 Experimental Setup and test sequence

The thermal vacuum tests were conducted at the David Florida Laboratory (DFL) in Kanata near Ottawa. A vacuum chamber was used to create the conditions of vacuum. Pressures of the order of 1×10^{-8}Torr, about 10 billion times lesser than atmospheric pressure, were achieved inside the chamber. The experimental set up is nearly the same as the one used at IQC laboratory described by the schematic diagram in figure 2.2. A key difference is that the TVAC DM has five detectors unlike the prototype that was used in section 2. Also, the freezer was not used to change the radiator's temperature because the TVAC chamber had a plate inside the chamber whose temperature can be controlled. Also, only the PicoQuant laser was used because it was portable - the mode locked laser could not be transported offsite. The rest of the set up is nearly identical to the one used in the IQC laboratory.

A few room temperature tests at ambient pressure were performed first at a partner company Neptec (test #00) and then at DFL (test #01). Then the DM was sealed in the vacuum chamber and vacuum pressure was achieved. Here, another room temperature test was performed - test 02. For these tests, a detector temperature of $-20\,^{\circ}\mathrm{C}$ was used. The purpose of these room temperature tests was mainly to make sure the the DM is functional and there isn't any problem with the associated electronics along the way. Dark counts tend to increase with APD temperature. Hence, cooling the detectors with TEC's is one way to decrease dark counts. So the detectors were then cooled to $-80\,^{\circ}\mathrm{C}$. From here on different combination of temperatures for the bracket and radiator were tried under vacuum conditions while maintaining the APD temperatures at $-80\,^{\circ}\mathrm{C}$. Finally, the DM was brought back to ambient pressure and a room temperature test with the detectors at $-20\,^{\circ}\mathrm{C}$ was performed. For all tests,a bias voltage of 20 V was applied to APD. The discriminator voltage was set to 50 mV. This means that the discriminator circuit on only allows output pulses with a voltage amplitude greater than 20 V to reach the time tagger / oscilloscope.

Note: The DM used in the TVAC tests had not been irradiated with protons.

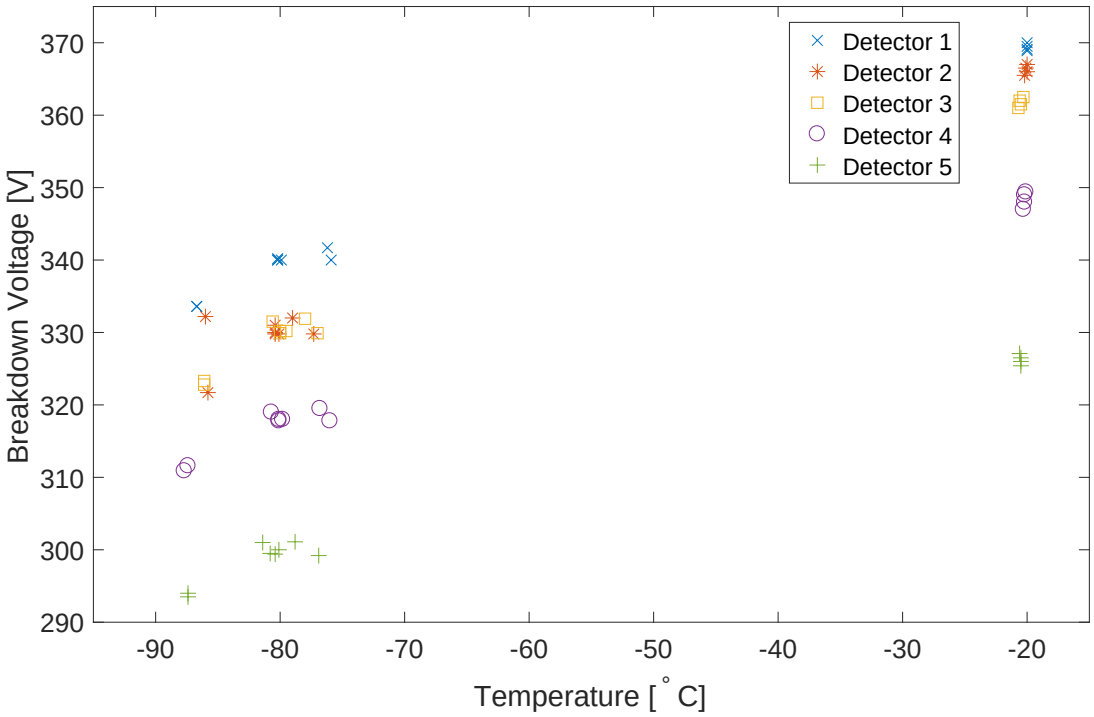


Figure 4.1: Breakdown voltage of APD is plotted as a function of temperature. A linear dependence is found.

4.3 Results and Discussions

The APD's were characterized during each test for breakdown voltage, output pulse characteristics, recharge time, dark counts, detection efficiency, saturation value and afterpulsing probability. The details of how each of these parameters were measured or calculated is given in section 2.

4.3.1 Breakdown Voltage

The breakdown voltage was measured during each test. Results are shown in figure 4.1. The breakdown voltage shows a nearly positive linear correlation wit APD temperature. This temperature dependence again emphasizes the importance of determining the breakdown voltage before every test even if all the conditions of the experimental set up remain the same - small variations in the breakdown voltage can effectively change the over-voltage on the detectors.

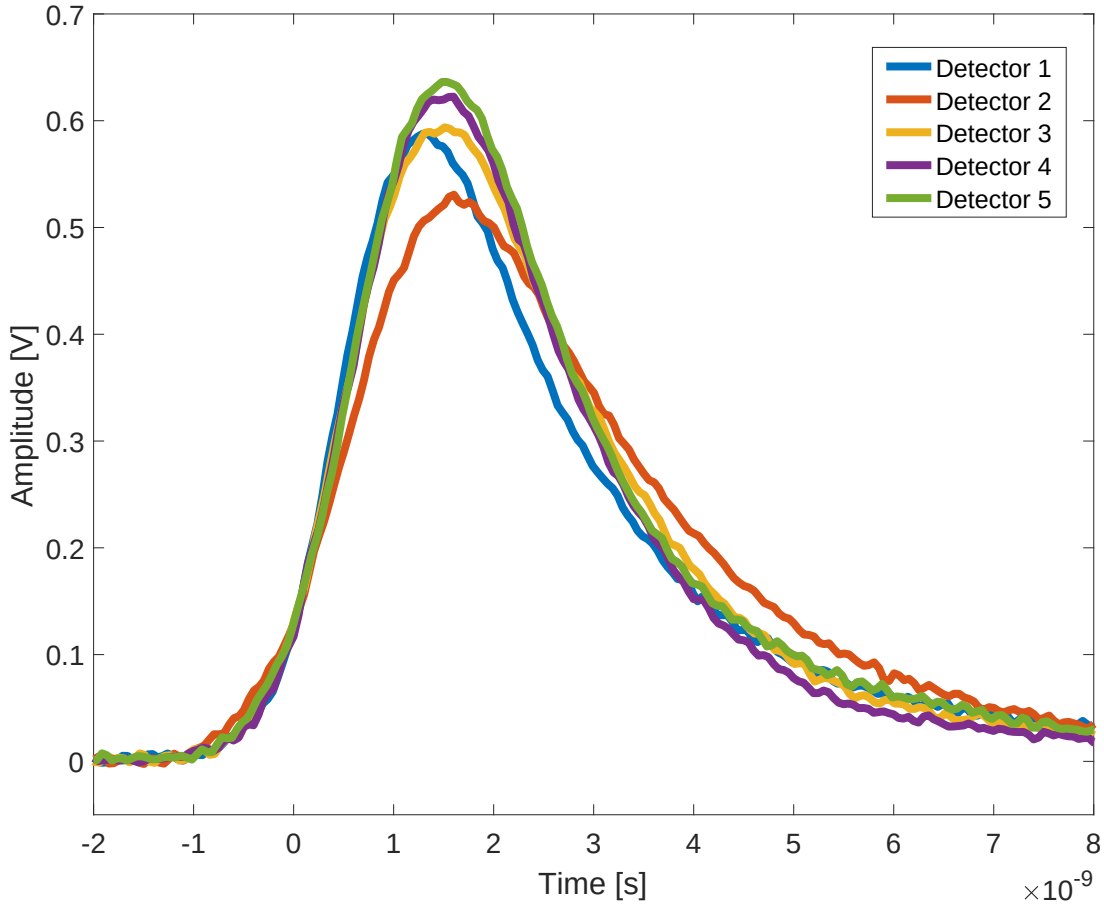


Figure 4.2: Output pulse shape for test #05 for all five detectors. The detector temperature was $-80\,^{\circ}\mathrm{C}$.

4.3.2 Output Pulse Shape

The output from the APD's were connected to an oscilloscope. The output pulse was visualized on the screen. A sample of the output pulse for all five detectors on test #05 is shown in figure 4.2. The amplitude and full width at half maximum (FWHM) of the pulse was visually determined. The amplitude of the output pulse showed a tendency to increase while the FWHM value showed a tendency to decrease. Upon further investigation, the area under the curve calculated by the Riemann integral does not show a tendency for increase/decreases. The area under the curve is directly related to the energy of the output pulse which is expected to be constant because of the constant input voltage. The constancy of the area under the pulse curve shows that the energy of the output pulse is more or less constant. However, the reason for the changes in amplitude and FWHM is unknown. Such changes were not observed in the laboratory at IQC.

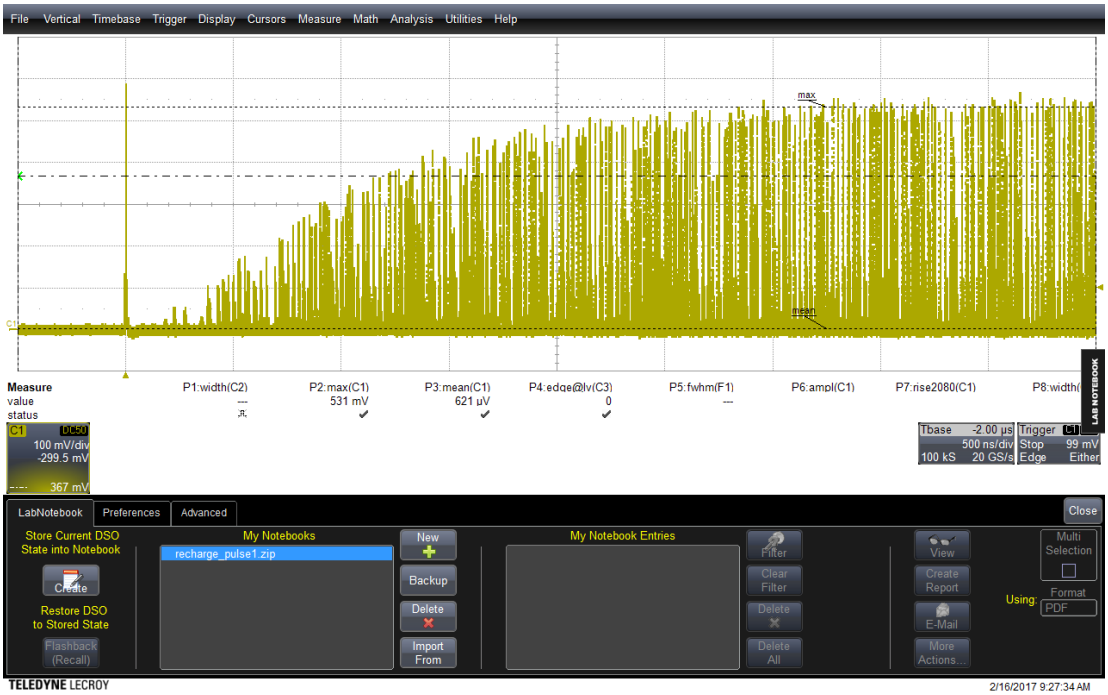


Figure 4.3: Screenshot of oscilloscope while calculating recharge time in TVAC experiments. The oscilloscope was set to persistent mode and about 2-3 minutes were needed for the graph to get populated with data points. The amplitude of the next output pulse following any given output pulse increases (and asymptotes) as the time difference between the two pulses increases.

4.3.3 Recharge Time

The oscilloscope is set to persistent mode and the recharge curve is populated with dark counts. The laser can be turned on if the dark count rate is low and one wants to speed up the process of populating the recharge curve. A screen shot of the oscilloscope screen is given in figure 4.3. Details of the process of measuring the recharge time is given in section 2.2.4. The recharge time values are plotted with respect to APD temperatures in figure 4.4. The recharge time showed a slight positive correlation with temperature. However, one must also account for the fact that the recharge time was visually measured off the oscilloscope screen and the precision was about 50 ns. So, the correlation might be less stringent than what appears directly in figure 4.4.

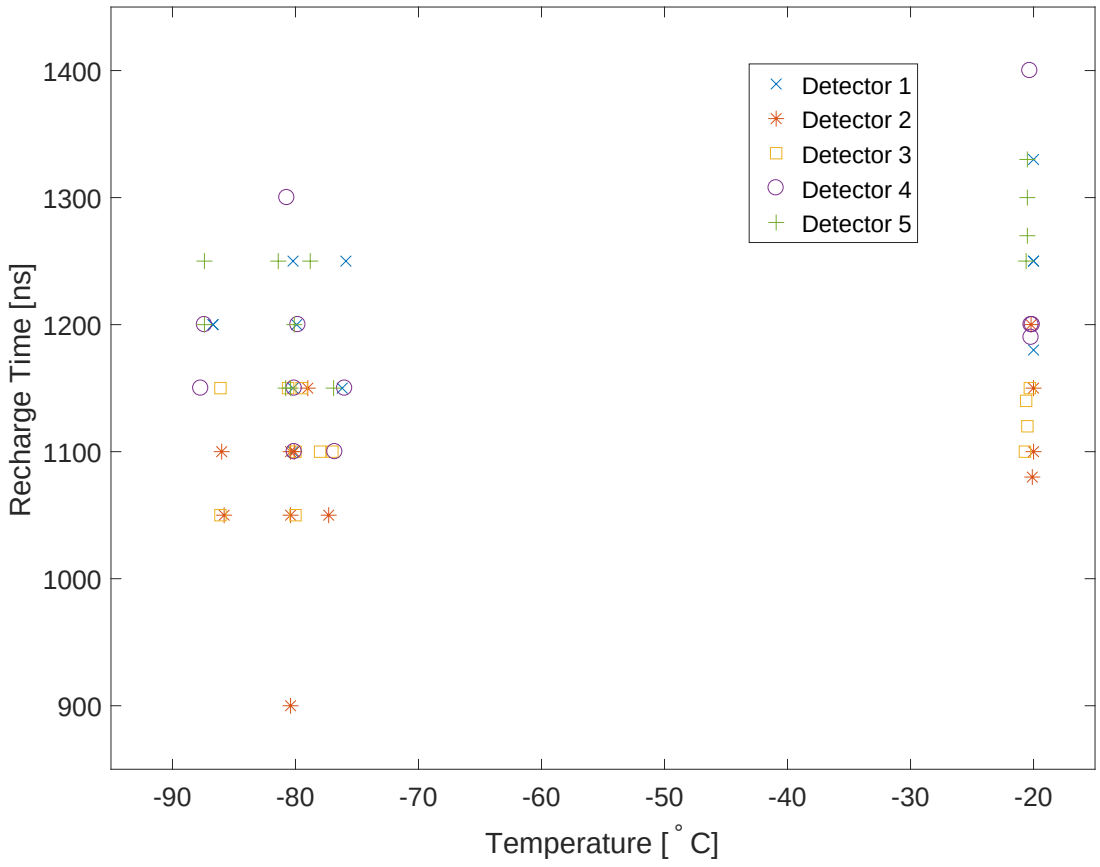


Figure 4.4: Recharge time is plotted as a function of detector temperature. The recharge time is calculated as the time taken for the recharge time curve in figure 4.3 to reach 63% of it's maximum height.

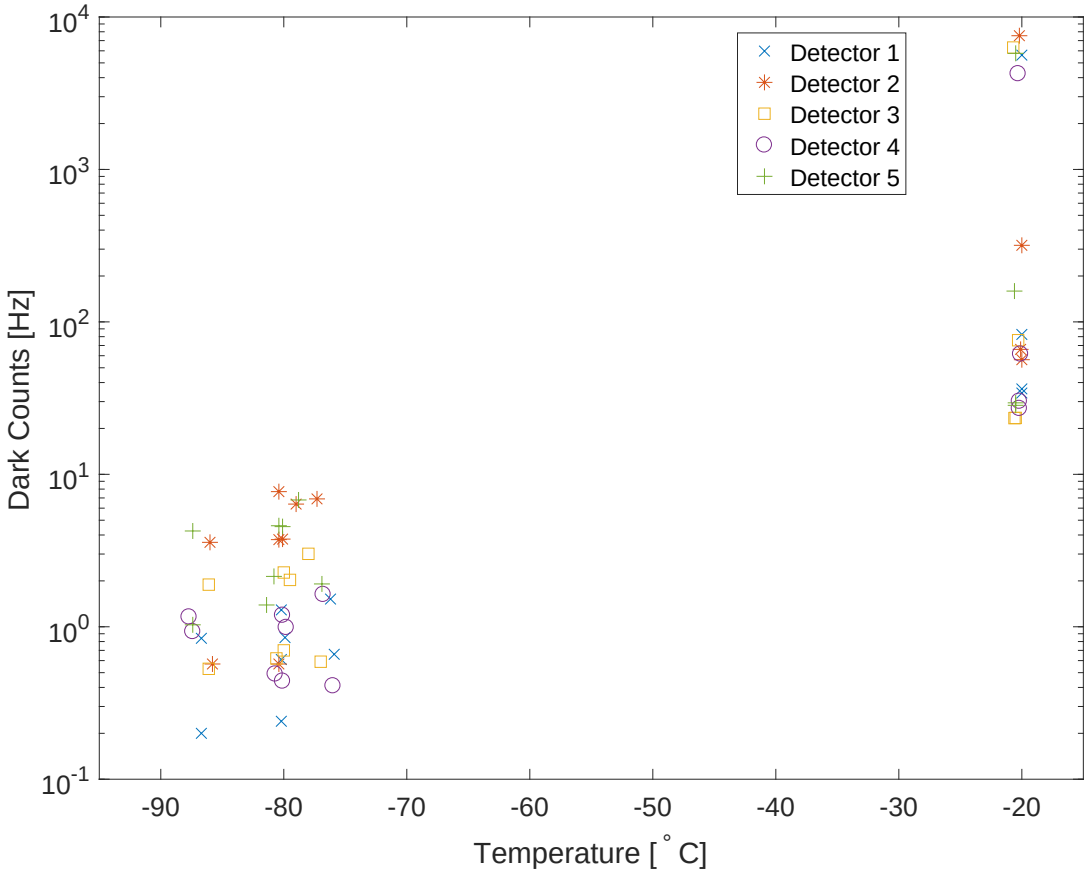


Figure 4.5: Dark count rate is plotted as a function of detector temperature. The dark count rate tends to increase linearly with detector temperature.

4.3.4 Dark Counts

The dark count rate was measured in the absence of ambient light - black cloth was used to cover the optical fibers. The laser was turned off for this experiment. The detector clicks were time tagged. The dark counts are plotted in figure 4.5 with respect to APD temperature. Dark count rate shows a nearly positive linear correlation to APD temperature. It is important to note that these dark counts included both thermal counts which are Poisson distributed in time as well as their generations of afterpulses.

4.3.5 Detection Efficiency

With the laser turned on, the optical power meter placed just before the DM in the optical path lets us calculate the rate of photons hitting the detectors. Comparing this to the detector click rate and ignoring the possibility of afterpulses in the APD's, one can calculate

the detection efficiency. Details of this calculation are explained in section 2.2.6. The detection efficiency at the room temperature tests at DFL were in the range 23% to 35%. The efficiency on test #00 was a bit higher than the aforementioned range. It was found out to be in the range 34 - 45%. This test was conducted at Neptec and changes in optical shielding - specific optical fibers and their lengths, the specific arrangement of the black light shielding cloth, etc. can easily account for this change. The manufacturer specified efficiency of the detector itself is about 60%. This does not account for optical efficiency of the optical path from laser to APD. The optical efficiency can be in the range of 50 - 70%. This satisfactorily explains the observed values of detection efficiency.

The efficiency at cold temperatures (below $-70\,^{\circ}\mathrm{C}$) drastically dropped. The vast majority of the cold temperature tests had observed detection efficiency in the range 3 - 16% with only three instances where the detection efficiency of a detector in the range 16 - 33%. This was not expected. Moreover, the decrease in efficiency was strongly correlated to the radiator temperature. Since the bracket and detector TEC's were functioning and stabilized to attain a specific bracket and detector temperature, there was some contention that the drop in efficiency was caused by something outside the DM. A series of troubleshooting tests were conducted. One included the DM being connected directly to the laser (and not using any mating sleeves) to find that the efficiency returned back to normal. Also, heating the radiator to room temperature cause the efficiency to return back to normal. It was finally determined that the mating sleeve which connects optical fibers along the optical path was not rated for use at cold temperature - it was only rated for vacuum. The mating sleeves were resting on the radiator and their temperature varied with the radiator's temperature. This was the source of loss of detection efficiency.

4.3.6 Timing Jitter

Th PicoQuant laser is used to determine the timing jitter of the detectors. The system timing jitter was first calculated by calculating the time difference between the laser clock pulse and the corresponding detector click caused by the photon hitting the detector. The time difference has a distribution shown in figure 4.6. The standard deviation of distribution gives the overall system's timing jitter. Then one can calculate the timing jitter of the detector by accounting for the time tagger's and laser source's timing jitter. Details of this calculation are given in section 2.2.7 (look for the calculations regarding the PicoQuant laser, not the mode locked laser). The calculated FWHM value (instead of standard deviation for the approximated Gaussian) of timing jitter values are plotted against APD temperature in figure 4.7. The timing jitters of the detectors were calculated

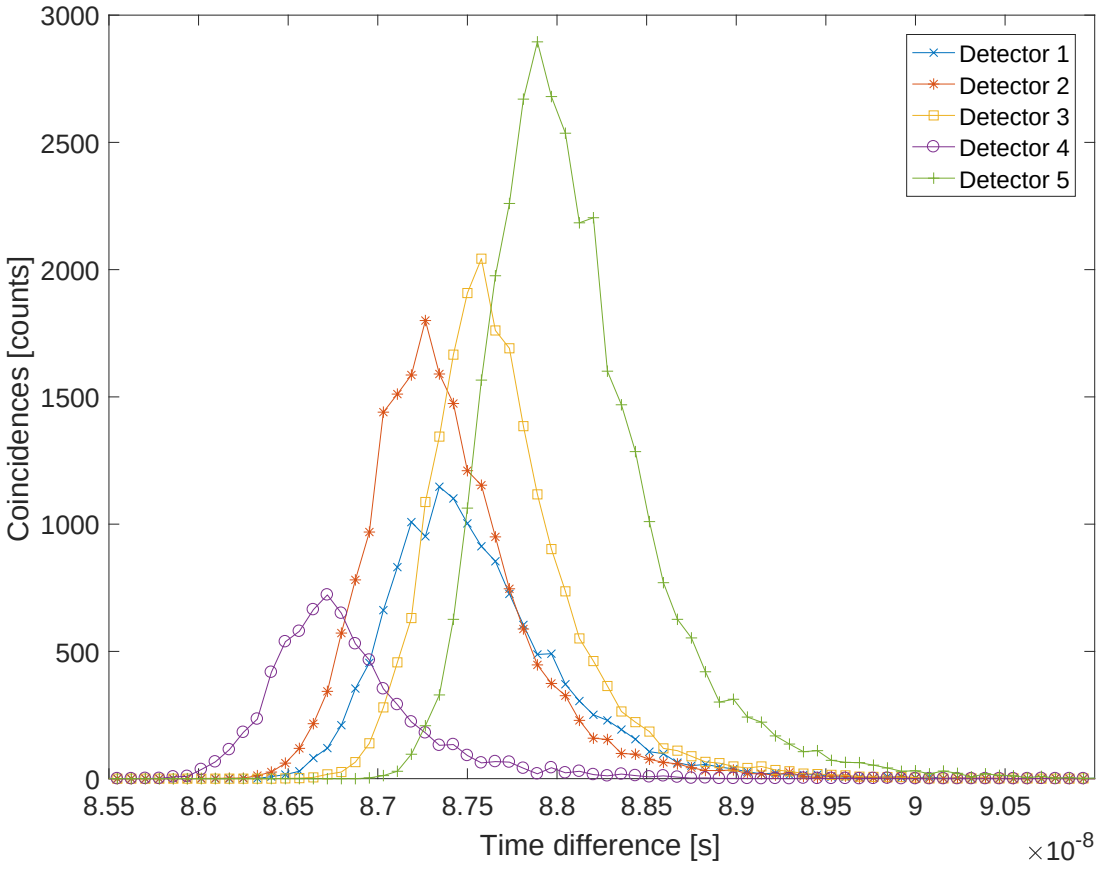


Figure 4.6: Histogram of photon coincidences for test #05 at a detector temperature of −80 °C. The timing jitter is then calculated as the width of this curve. Each curve is curve fitted using a Gaussian equation. The width (or standard deviation) of the resulting Gaussian is a measure of the timing jitter of that detector.

to be in the range 525 - 900ps. This is slightly higher than was expected because the tests with the prototype had given rise to jitter value sin the range 400 - 800 ps.

Test 00 gave rise to somewhat higher jitter values that the rest of the tests. This test was conducted at Neptec and high dark counts two orders of magnitude higher than the room temperature tests at DFL. My guess is that this increase in dark counts may have affected the algorithm that determined the coincidences between laser pulses and photon induced detector clicks. and caused a horizontal spread in the histogram in figure 4.6. Also, detector #1 on the DM showed a consistently higher jitter than the rest of the detectors. Th cause of this is unknown.

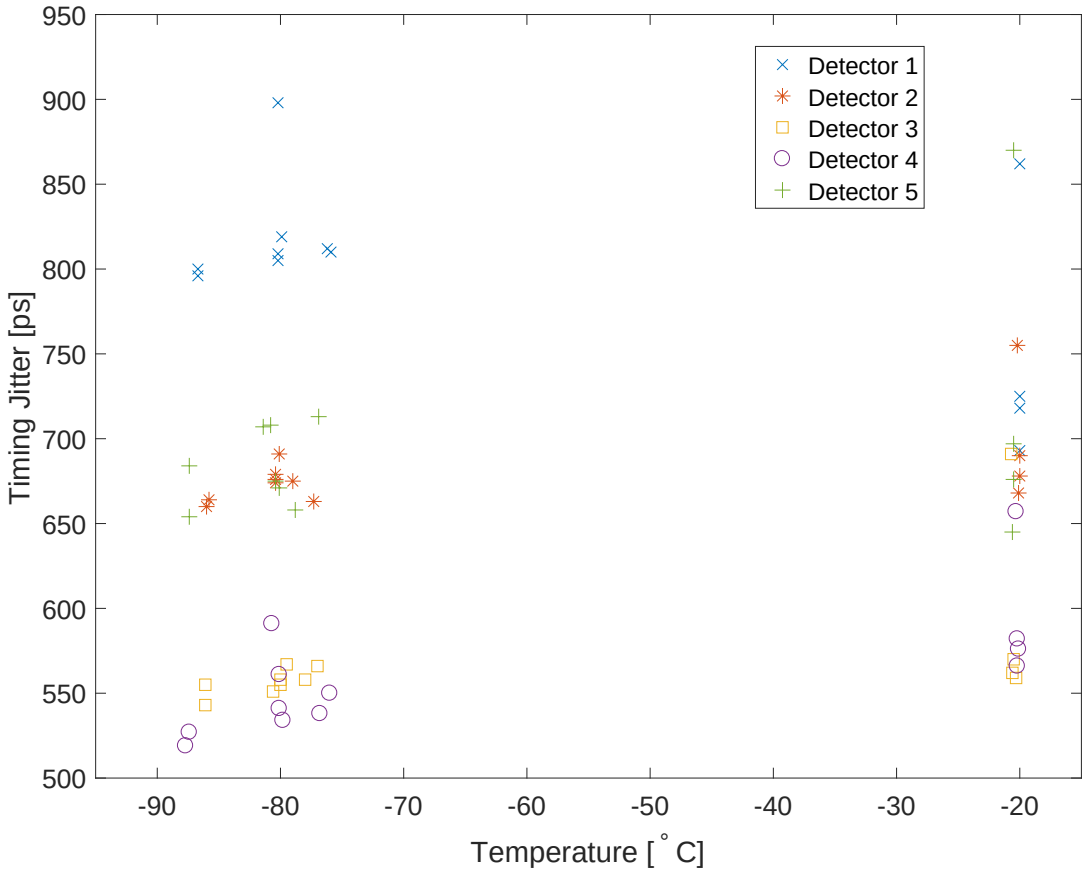


Figure 4.7: Timing jitter of detectors as a function of detector temperature in TVAC experiment

4.3.7 Saturation

The maximum count rate , or saturation value, was determined by varying the input laser power. The details of this procedure are given in section 2.2.8. The saturation values were in the range 325 - 780 kHz. One notices that the saturation values are higher at lower APD temperatures. For example, the room temperature tests had saturation value sin the range 325 - 590 kHz whereas the cold temperature tests had saturation values in the range 470 - 780 kHz. My contention is that the decreased efficiency could potentially require a larger laser power to saturate the detectors and since the optical power meter was earlier in the optical path than the mating sleeve which caused the loss of efficiency, it is possible that the optical power meter would have recorded a large power but the APD's receive a fraction of the photons.

4.4 Conclusion

The DM was characterized for a combination of different APD temperatures, bracket temperatures and radiator temperatures in the conditions of vacuum of the order of 1×10^{-8}Torr. The parameters measured were breakdown voltage, output pulse amplitude and FWHM, recharge time, dark counts, detection efficiency, timing jitter and saturation. Some of the values, especially the timing jitter were higher than expected. The reason for this has not been ascertained and future tests can look into the cause to this deviation. The output pulse amplitudes and FWHM also changed over the course of the tests. This could also be the motivation for further study. Finally, the drop in efficiency at cold temperatures was determined to be due to the mating sleeves that were not rated for cold temperatures. Most other parameters behaved as expected. The minor anomalies , for example,the increased timing jitter of detector #1 can also be explored. Of central importance, the DM survived the vacuum state in operating condition.

Chapter 5

Theoretical Analysis of Afterpulsing

5.1 Motivation

Afterpulses are output avalanches in an APD that are correlated with a previous avalanche. During any given avalanche, there is a finite probability that an electron can get trapped in a trap site in the APD and released at a later time. This probability is called the afterpulsing probability. It is generally thought that the afterpulsing probability is constant for a given spatial density of trap sites in the APD substrate and fixed operational conditions (over-voltage, etc.). Even if the probability with which an electron gets trapped remains fixed, it is conceivable that the experimentally observed afterpulsing probability may change with parameters such as thermal counts rate, signal photon rate and dead time. To give a brief intuition, the dead time plays a key role here. If a trapped electron gets detrapped during a dead time period (i.e., when the detector is inactive), it does not cause an avalanche. Hence, you do not get an afterpulse. I want to distinguish between two terms used in this chapter: *Afterpulsing probability* is the probability with which one electron gets trapped in a trap site during an avalanche. The *observed afterpulsing probability* is the probability with which an avalanche successfully gives rise to a second avalanche when the trapped electron gets detrapped.

Note: I assume that during an avalanche exactly zero or one electron gets trapped. Multiple electrons in an avalanche can't get trapped.

A high value of dead time can decrease the observed afterpulsing probability because the detector is unable to avalanche for a longer period of time, thereby increasing the chances that a trapped electron can get detrapped when APD is inactive. High thermal count rate and signal photon rate can both also decrease the observed afterpulsing probability because

the time tag file becomes increasingly cluttered with dead regions, each following every detector click. This also increases the chances that the trapped electron will get detrapped when the APD is inactive. In the following subsections, I quantitatively derive expressions for observed afterpulsing probabilities from first principles using as few assumptions as possible. In section 5.2, I derive an expression for the observed afterpulsing probability as a function of thermal count rate.

5.2 Observed afterpulsing probability as a function of thermal counts

Let the afterpulsing probability of the APD detector be denoted by p_{ap}. The detrapping time of a trapped electron is given by modeled by a random variable which follows an exponentially decaying time distribution given by

$$\text{p.d.f. of detrapping time of a trapped electron} = \lambda_{ap}\ e^{-\lambda_{ap}\ t} \tag{5.1}$$

- λ_{ap} is the detrapping time constant, also called the afterpulsing time constant because the process of trapping of electron and the process of the detrapped electron causing an avalanche are assumed to occur on negligible time scales.
- t is the time elapsed since the electron got trapped.

Equation 5.1 is actually a conditional probability where the condition is the trapping of the electron. The area under the curve from $t = 0$ to $t = \infty$ is unity, representing the fact that a trapped electron gets detrapped with 100% probability.

$$\int_{t=0}^{\infty} \lambda_{ap}\ e^{-\lambda_{ap}\ t} \mathrm{d}t = 1 \tag{5.2}$$

If one wants to find the probability density function (p.d.f.) of the time distribution for an afterpulse to occur following any arbitrary avalanche, equation 5.1 is not directly applicable because not every avalanche results in a trapped electron. Instead the probability of trapping an electron is given by p_{ap}. Thus, if one renormalizes equation 5.1 such that the area under it's curve is p_{ap} instead of unity, one gets the required p.d.f. of the time distribution for an afterpulse to occur following an arbitrary avalanche

$$\text{p.d.f. of time for afterpulse to occur} = p_{ap}\ \lambda_{ap}\ e^{-\lambda_{ap}\ t} \tag{5.3}$$

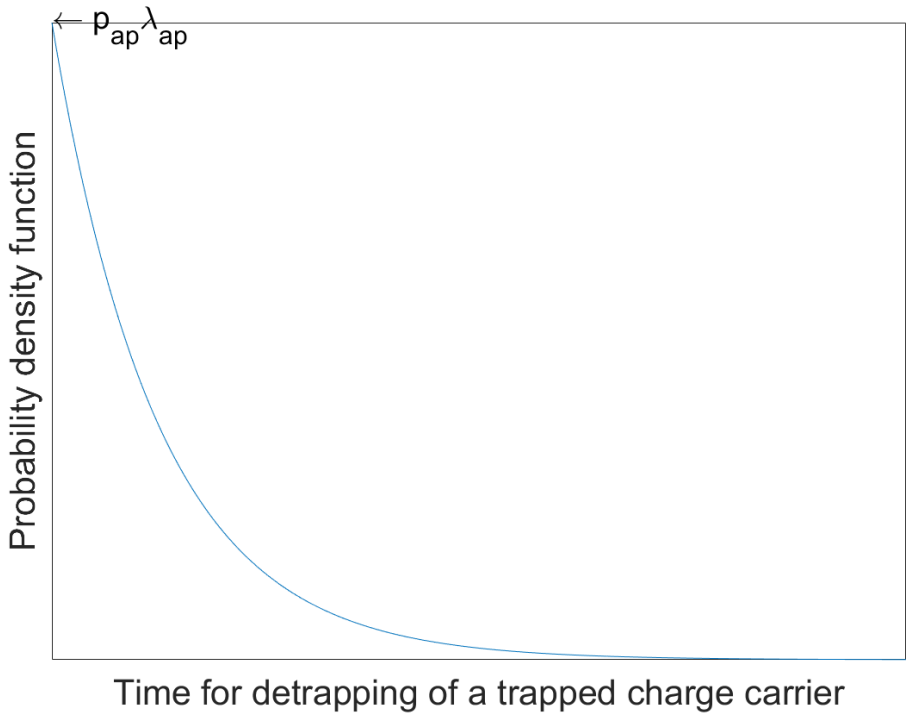


Figure 5.1: Probability density function of the time required for an afterpulse to occur following any arbitrary avalanche. The equation of the curve is of the form $p_{ap}\ \lambda_{ap}\ e^{-\lambda_{ap}\ t}$ where p_{ap} is the afterpulsing probability and λ_{ap} is the afterpulsing time constant. The area under the curve is p_{ap} because an arbitrary avalanche gives rise to an afterpulse not with 100% probability but with the afterpulsing probability p_{ap}

where t is the time elapsed after an arbitrary avalanche

One must however be careful to only deduce statistical results from this p.d.f. and not inferences about a particular avalanche because the probability p_{ap} in equation 5.3 denotes that it is applied to arbitrary avalanches which may or may not result in an afterpulse. The area under the curve is p_{ap}.

$$\int_{t=0}^{\infty} p_{ap}\ \lambda_{ap}\ e^{-\lambda_{ap}\ t}\ \mathrm{d}t = p_{ap} \tag{5.4}$$

The curve in equation 5.3 is sketched in figure 5.1

Equation 5.3 assumes zero dead time. With no dead time, every detrapped electron would produce an avalanche. Thus, the observed afterpulsing probability would be equal to p_{ap}. However, as explained earlier, the presence of finite dead time decreases the observed

afterpulsing probability because in the range $t = [\ 0,\ \infty\ )$, there could be dead time regions induced either by avalanches that occur at some time in the finite future or by the avalanche whose afterpulsing distribution we are trying to calculate itself. One would thus need to modify the p.d.f. in equation 5.3 to consider dead time. The area under this modified curve would then be the observed afterpulsing probability. One can divide the time axis into three regions:

- $t = [\ 0,\ \tau_d\ )$
- $t = [\ \tau_d,\ 2\tau_d\ )$
- $t = [\ 2\tau_d,\ \infty\ )$

where τ_d is the dead time

5.2.1 Region $t = [\ 0,\ \tau_d\ )$

The avalanche in question itself induces a dead time period which by definition runs from $t = 0$ to $t = \tau_d$. Since in this region, an afterpulse cannot occur even if the possibly trapped electron gets detrapped, the p.d.f. for the time distribution of afterpulse in equation 5.3 assumes a value of 0.

$$\text{p.d.f. of time for afterpulse to occur} = 0\ , \qquad t = [\ 0,\ \tau_d\ ) \tag{5.5}$$

5.2.2 Region $t = [\ 2\tau_d,\ \infty\ )$

Let's assume that the electron gets detrapped at some time instant $t = t_0$. In order for this detrapped electron to trigger an avalanche, the detector must be active. This means that no detector click can occur in a time window of width τ_d to the left of $t = t_0$. If such a detector click did occur, it would induce a dead time period and the time instant $t = t_0$ would fall within this period, thus the detector won't be able to avalanche when the electron gets detrapped. It is worthwhile to note that I am assuming a 100% detector efficiency, i.e., every *potential* avalanche physically occurs. In the absence of signal photons, two kinds of detector clicks can occur - thermal counts and afterpulses (not the afterpulse in question). I will only consider thermal counts in this discussion. WE would like to find the probability that no thermal counts occur in a time window of width τ_d to the left of $t = t_0$. Thermal counts occur randomly follow a Poisson distribution in time. The

probability of detecting k thermal counts is a time interval of some fixed length is given by $P(k)$ as follows:

$$P(k) = e^{-\lambda}\frac{\lambda^k}{k!} \tag{5.6}$$

where λ is the average number thermal counts in the time interval.

One must note that since the thermal counts occur randomly in time, the probability $P(k)$ depends only on the width of the time interval and is independent of the position of the time interval in the time tag file. Thus for a time window of width τ_d and an average thermal count rate of d counts/second, the average number of thermal count in the time window is given by

$$\lambda = d\ \tau_d \tag{5.7}$$

The probability of having no thermal counts in this time window can be derived from equations 5.6 and 5.7 by substituting $k = 0$ to get

$$P(k=0) = e^{-\lambda} = e^{-d\tau_d}\ , \qquad t = [\ 2\tau_d,\ \infty\ ) \tag{5.8}$$

Any point on the p.d.f. in equation 5.3 in the region $[\ 2\tau_d,\ \infty\ )$ should be multiplied by the factor $e^{-d\tau_d}$ because given knowledge that an electron has been detrapped at time $t = t_0$, it triggers an avalanche not with probability 1 but with probability $e^{-d\tau_d}$, the probability that no thermal counts occur in the aforementioned time window. Thus the modified p.d.f. in this region becomes

$$\text{p.d.f. of time for afterpulse to occur} = p_{ap}\ \lambda_{ap}\ e^{-\lambda_{ap}\ t}\ e^{-d\tau_d}\ , \qquad t = [\ 2\tau_d,\ \infty\ ) \tag{5.9}$$

5.2.3 Region $t = [\ \tau_d,\ 2\tau_d\ )$

In the region $t = [\ \tau_d,\ 2\tau_d\ )$, the same argument as in subsection 5.2.2 applies. However, the width of the time window to consider is no longer τ_d. This is because if one tries to consider a time window of width τ_d to the left of some time instant $t = t_0$ in the region $t = [\ \tau_d,\ 2\tau_d\ )$, one impinges into the region $t = [\ 0,\ \tau_d\ )$ which was already addressed in subsection 5.2.1. Instead the left edge of the time window should be at $t = \tau_d$. Since the right edge of the time window is at $t = t_0$, the width of the time window is $t_0 - \tau_d$. Let's

change the notation here a bit. Since we are looking for a modified p.d.f. of equation 5.3 which is already a function of the time variable t, the time window to consider to the left of an arbitrary time instant t in the region $t = [\ \tau_d,\ 2\tau_d\ )$ should be of width $t - \tau_d$. The average number of thermal counts occurring in this window is given by

$$\lambda = d\ (t - \tau_d)\ , \qquad t = [\ \tau_d,\ 2\tau_d\ ) \tag{5.10}$$

According to equation 5.6, the probability of having zero thermal counts is given by

$$P(k = 0) = e^{-\lambda} = e^{-d(t-\tau_d)}\ , \qquad t = [\ \tau_d,\ 2\tau_d\ ) \tag{5.11}$$

Thus the p.d.f. in equation 5.3 should be multiplied by the factor $e^{-d(t-\tau_d)}$ because this is the probability with which a detrapped electron in this region can trigger an avalanche. The modified p.d.f. in region $t = [\ \tau_d,\ 2\tau_d\ )$ is given by

$$\text{p.d.f. of time for afterpulse to occur} = p_{ap}\ \lambda_{ap}\ e^{-\lambda_{ap}\ t}\ e^{-d(t-\tau_d)}\ , \qquad t = [\ \tau_d,\ 2\tau_d\ ) \tag{5.12}$$

5.2.4 Deriving the observed afterpulsing probability from the modified afterpulsing time distribution

The modified p.d.f. of time distribution for an afterpulse to occur over the entire range $t = [\ 0,\ \infty\ )$ is now given by

$$\text{p.d.f. of time for afterpulse to occur} = \begin{cases} 0\ , & t = [\ 0,\ \tau_d\ ) \\ p_{ap}\ \lambda_{ap}\ e^{-\lambda_{ap}\ t}\ e^{-d(t-\tau_d)}\ , & t = [\ \tau_d,\ 2\tau_d\ ) \\ p_{ap}\ \lambda_{ap}\ e^{-\lambda_{ap}\ t}\ e^{-d\tau_d}\ , & t = [\ 2\tau_d,\ \infty\ ) \end{cases} \tag{5.13}$$

The observed afterpulsing probability is got by simply integrating the modified p.d.f. of afterpulsing time distribution given in equation 5.13 over the time interval $t = [\ 0,\ \infty\ )$. Let's integrate over each of the three regions separately and then add the probabilities.

Let the area under the modified p.d.f. in the region $t = [\ 0,\ \tau_d\ )$ be P_I.

$$P_I = 0 \tag{5.14}$$

Let the area under the modified p.d.f. in the region $t = [\ \tau_d,\ 2\tau_d\ )$ be P_{II}.

$$\begin{aligned} P_{II} &= \int_{t=\tau_d}^{2\tau_d} p_{ap}\ \lambda_{ap}\ e^{-\lambda_{ap}\ t}\ e^{-d(t-\tau_d)}\ \mathrm{d}t \\ &= p_{ap}\ \lambda_{ap}\ e^{d\tau_d} \int_{t=\tau_d}^{2\tau_d} e^{-(\lambda_{ap}+d)t}\ \mathrm{d}t \\ &= p_{ap}\ \frac{\lambda_{ap}}{\lambda_{ap}+d}\ e^{d\tau_d}\ (e^{-(\lambda_{ap}+d)\tau_d} - e^{-2(\lambda_{ap}+d)\tau_d}) \\ &= p_{ap}\ \frac{\lambda_{ap}}{\lambda_{ap}+d}\ e^{-\lambda_{ap}\tau_d}\ (1 - e^{-(\lambda_{ap}+d)\tau_d}) \end{aligned} \tag{5.15}$$

Let the area under the modified p.d.f. in the region $t = [\ 2\tau_d,\ \infty\ )$ be P_{III}.

$$\begin{aligned} P_{III} &= \int_{t=2\tau_d}^{\infty} p_{ap}\ \lambda_{ap}\ e^{-\lambda_{ap}\ t}\ e^{-d\tau_d}\ \mathrm{d}t \\ &= p_{ap}\ \lambda_{ap}\ e^{-d\tau_d} \int_{t=2\tau_d}^{\infty} e^{-\lambda_{ap}\ t}\ \mathrm{d}t \\ &= p_{ap}\ \lambda_{ap}\ e^{-d\tau_d}\ \frac{e^{-2\lambda_{ap}\tau_d}}{\lambda_{ap}} \\ &= p_{ap}\ e^{-\lambda_{ap}\tau_d}\ e^{-(\lambda_{ap}+d)\tau_d} \end{aligned} \tag{5.16}$$

Thus, the observed afterpulsing probability, p_{obs} is given by

$$\begin{aligned} \text{Observed afterpulsing probability, } p_{obs} &= P_I + P_{II} + P_{III} \\ &= 0 + p_{ap}\ \frac{\lambda_{ap}}{\lambda_{ap}+d}\ e^{-\lambda_{ap}\tau_d}\ (1 - e^{-(\lambda_{ap}+d)\tau_d}) \\ &\quad + p_{ap}\ e^{-\lambda_{ap}\tau_d}\ e^{-(\lambda_{ap}+d)\tau_d} \\ &= \frac{p_{ap}\ e^{-\lambda_{ap}\tau_d}}{\lambda_{ap}+d}\ (\lambda_{ap} + d\ e^{-(\lambda_{ap}+d)\tau_d}) \end{aligned} \tag{5.17}$$

Thus, every avalanche produces an afterpulse with probability p_{obs}. Figure 5.2 plots the observed afterpulsing probability p_{obs} as a function of thermal count rate d according to the equation 5.17. The afterpulsing probability p_{ap} (the probability of a charge carrier getting trapped) is fixed at 25%. The afterpulsing time constant λ_{ap} is fixed at $5 \times 10^3\,\mathrm{s}^{-1}$. Finally, the dead time τ_d was assumed to be $1 \times 10^{-5}\,\mathrm{s}$ and the recharge time τ_R was assumed negligible.

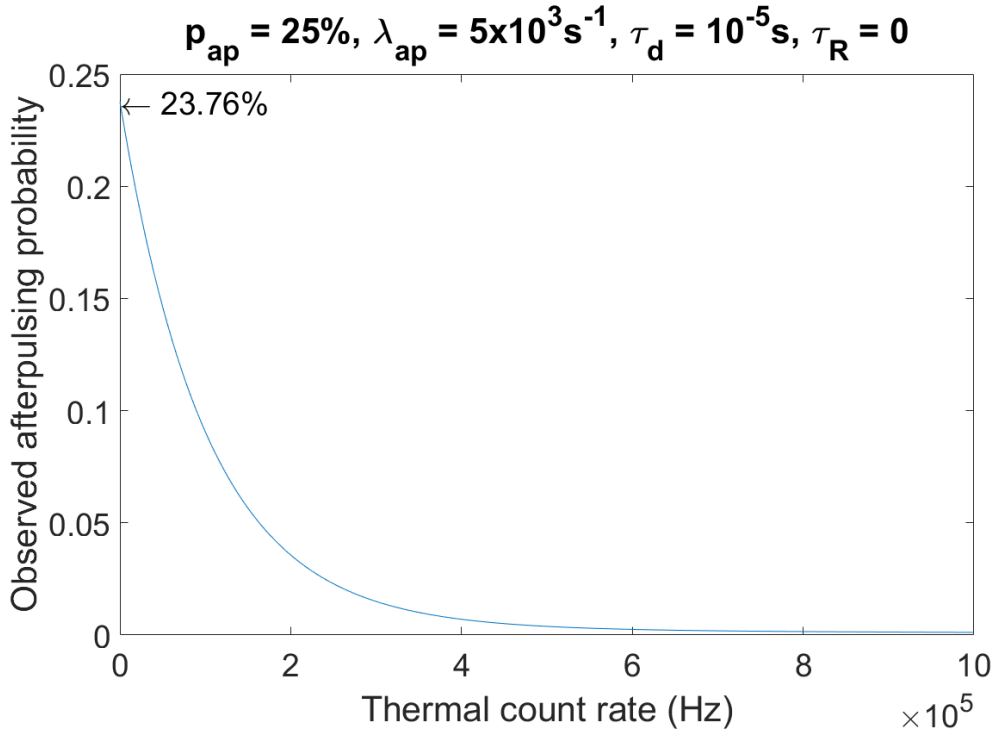


Figure 5.2: The observed afterpulsing probability is plotted against thermal count rate in accordance with equation 5.17. The afterpulsing probability p_{ap} (i.e., probability of a charge carrier getting trapped) is assuumed to be 25% with afteprulsing time constant λ_{ap} $= 5 \times 10^3\,\mathrm{s}^{-1}$ and the dead time $\tau_d = 1 \times 10^{-5}\,\mathrm{s}$

The observed afterpulsing probability decreases as the thermal count rate increases. This is in keeping with the notion that the increasing number of thermal counts induce more dead time regions in the time tag file. A charge carrier that is trapped then has an increasing likelihood of being released when the detector is in a dead time region, thus failing to trigger a second avalanche, the afterpulse. This decreases the observed afterpulsing probability. One notices that the observed afterpulsing probability at zero thermal count rate isn't 25% (the value that was ascribed to p_{ap}). This is due to the presence of dead time region induced by the start time tag itself. IF the trapped charge carrier is detrapped during this dead time, it can't trigger an afterpulse, hence reducing the observed afterpulsing probability to slightly below 25%. The graph shows that this value is 23.76%. Of course one can set $d = 0$ in equation 5.17 to get this value. But in order to get more insight, let's try to theoretically derive this value from first principles.

The area under the time distribution of detrapping of a charge carrier given by equation 5.3 is $p_{ap}\lambda_{ap}$. The dead time occurs in the interval $t = [\ 0,\ \tau_d\ )$. To evaluate the observed afterpulsing probability when no thermal counts are present, one must integrate the p.d.f. of detrapping of charge carrier between the limits $t = \tau_d$ and $t = \infty$ because it is only in this region that a detrapped charge carrier can trigger an afterpulse. Therefore,

$$\begin{aligned}
\text{Observed afterpulsing probability, } p_{obs}\Big|_{d=0} &= \int_{t=\tau_d}^{\infty} p_{ap}\ \lambda_{ap}\ e^{-\lambda_{ap}\ t}\ \mathrm{d}t \\
&= -p_{ap}\left[e^{-\lambda_{ap}t}\right]_{t=\tau_d}^{t=\infty} \\
&= p_{ap}e^{-\lambda_{ap}\tau_d} \\
&= 0.25 \times e^{-\left(5\times10^{3}\right)\times\left(1\times10^{-5}\right)} \\
&= 0.2378 = 23.78\%
\end{aligned} \tag{5.18}$$

I would like to point out that a thermal count rate of zero doesn't make sense in this scenario because this would mean that there are no counts in the time tag files (not even afterpulses which require thermal counts to occur). But equation 5.18 requires us to assume a start time tag. Hence this result can only make sense in the limit $d \to 0$.

Figure 5.3: Following an arbitrary detector click (also called the start time tag), we consider a time interval of width Δt after time t has elapsed since the start time tag.

5.3 Afterpulsing probability calculation from time tag file of APD

5.3.1 Theoretical derivation of probability density function of the next detector click

Here, I will derive an analytical expression for the p.d.f. of the time difference between consecutive detector clicks as a function of that time difference$t$. Another way to think about this graph is that it is the p.d.f. of when the next detector click occurs as a function of time elapsed t since any given detector click.

Let's consider an arbitrary detector click and call it the "start time tag". The next detector click occurs at some finite time t after the start time tag. Figure 5.3 illustrates this. Let's start by aiming to find the probability that the next click will occur in a small tie interval of width Δt whose left edge coincides with the time instant t as shown in the figure 5.3. I assume there are no signal photons of stray photons from ambient light hitting the APD. The detector clicks are due to thermal counts and afterpulses. Therefore the next detector click can either be a thermal count of an afterpulse of the start time tag. I am currently not considering afterpulses of detector clicks that are not the start time tag. The following are the requirements for the next detector click to occur in the aforementioned time interval Δt.

1. No thermal count occurs in the time interval [0 , t)

2. No afterpulse occurs in the time interval [0 , t)

3. Either a thermal count or afterpulse occurs in the time interval [t , $t + \Delta t$]

Thermal counts obey Poisson statistics. The probability of having k thermal counts in a given time interval is given by equation 5.6. If d is the thermal count rate, the average number of thermal counts λ occurring in any time interval of width $t*$ is given by $d \times t*$. The position of the time interval is not important since thermal counts occur randomly. Only the width is important.

The probability $P_{NT}^{0\to t}$ that no thermal counts occur in the time interval [0 , t) is calculated by setting $k = 0$ and $\lambda = d \times t$ in equation 5.6.

$$P_{NT}^{0\to t} = P(k = 0) = e^{-\lambda} = e^{-dt} \tag{5.19}$$

The p.d.f. of the time required for the afterpulse of the start time to occur is given by equation 5.3. The probability $P_{A}^{0\to t}$ that an afterpulse of the start time tag occurs in the interval [0 , t) is given by

$$\begin{aligned} P_{A}^{0\to t} &= \int_0^t p_{ap}\lambda_{ap}e^{-\lambda_{ap}t'}\mathrm{d}t' \\ &= p_{ap}(1 - e^{-\lambda_{ap}t}) \end{aligned} \tag{5.20}$$

Therefore, the probability $P_{NA}^{0\to t}$ of not having an afterpulse of the start time tag in the interval [0 , t) is given by

$$\begin{aligned} P_{NA}^{0\to t} &= 1 - P_{A}^{0\to t} \\ &= 1 - p_{ap}(1 - e^{-\lambda_{ap}t}) \end{aligned} \tag{5.21}$$

The probability $P_{T\mathrm{or}A}^{t\to t+\Delta t}$ of having either a thermal count or an afterpulse in the interval $[t, t + \Delta t]$ is given

$$\begin{aligned} P_{T\mathrm{or}A}^{t\to t+\Delta t} &= 1 - \text{probability of having neither thermal count nor afterpulse in interval}[t, t + \Delta t] \\ &= 1 - P_{NT}^{t\to t+\Delta t} \times P_{NA}^{t\to t+\Delta t} \end{aligned} \tag{5.22}$$

- $P_{NT}^{t\to t+\Delta t}$ is the probability of not having a thermal count in the interval $[t, t+\Delta t]$
- $P_{NA}^{t\to t+\Delta t}$ is the probability of not having an afterpulse of the start time tag in the interval $[t, t+\Delta t]$

Now,

$$P_{NT}^{t\to t+\Delta t} = P(k=0) = e^{-\lambda} = e^{-d\Delta t} \tag{5.23}$$

since $d\Delta t$ is the average number of thermal counts in the time interval $[t, t+\Delta t]$. Also, the probability $P_{A}^{t\to t+\Delta t}$ of having an afterpulse in the interval $[t, t+\Delta t]$ is given by

$$\begin{aligned} P_{A}^{t\to t+\Delta t} &= \int_{t}^{t+\Delta t} p_{ap}\lambda_{ap}e^{-\lambda_{ap}t'}\mathrm{d}t' \\ &= p_{ap}e^{-\lambda_{ap}t}(1-e^{-\lambda_{ap}\Delta t}) \end{aligned} \tag{5.24}$$

Therefore

$$\begin{aligned} P_{NA}^{t\to t+\Delta t} &= 1 - P_{A}^{t\to t+\Delta t} \\ &= 1 - p_{ap}e^{-\lambda_{ap}t}(1-e^{-\lambda_{ap}\Delta t}) \end{aligned} \tag{5.25}$$

Putting equations 5.23 and 5.25 into equation 5.22, we get

$$P_{T\mathrm{or}A}^{t\to t+\Delta t} = 1 - e^{-d\Delta t}(1 - p_{ap}e^{-\lambda_{ap}t}(1-e^{-\lambda_{ap}\Delta t})) \tag{5.26}$$

Now, to find the probability $P_{\text{next click}}^{t\to t+\Delta t}$ that the next detector click occurs in the time interval $[t, t+\Delta t]$, Ursin and Peev [33] suggests that the probability of finding the next click in the n^{th} bin involved the event of detecting a click in the n^{th} bin and the events of not detecting clicks in each of the first $n-1$ bins. Furthermore, they suggest multiplying the corresponding probabilities. For this problem, equation 5.27 would be the suggested expression for $P_{\text{next click}}^{t\to t+\Delta t}$.

$$P_{\text{next click}}^{t\to t+\Delta t} = P_{NT}^{0\to t} \times P_{NA}^{0\to t} \times P_{T\mathrm{or}A}^{t\to t+\Delta t} \tag{5.27}$$

Putting equations 5.19, 5.21 and 5.26 into equation 5.27, we get

$$P_{\text{next click}}^{t\to t+\Delta t} = e^{-dt} \times (1 - p_{ap}(1-e^{-\lambda_{ap}t})) \times (1 - e^{-d\Delta t}(1 - p_{ap}e^{-\lambda_{ap}t}(1-e^{-\lambda_{ap}\Delta t}))) \tag{5.28}$$

I then simulated thermal counts and afterpulse timetags in MATLAB and generated the histogram of time difference between consecutive detector clicks and compared it with equation 5.28 because I had access to the true values of dark count rate, afterpulsing probability and afterpulsing time constant. It was a close match but there was a tiny but noticeable deviation. After thinking about the problem at hand, when the probabilities were multiplied in equation 5.28, the underlying assumption was that they correspond to independent events. Since the dark counts were viewed as a Poisson process, this is true for the way the corresponding probabilities were formulated. However, when considering the afterpulsing probability, if the afterpulse of the start time tag is the very next click and say it occurs in the n^{th} bin, then the event that the afterpulse of the start time tag doesn't occur in the first $n-1$ bins is automatically manifested. The occurrence of the afterpulse in the n^{th} bin and the non-occurrence of the afterpulse in the first $n-1$ bins are not independent events. Hence, one cannot possibly multiply them. It only suffices to consider the probability that the afterpulse occurs in the n^{th} bin. The conditional probability of the non-occurrence of the afterpulse in the first $n-1$ bins given that the afterpulse occurred in the first $n-1$ bins is unity.

In fact, this caveat is not peculiar to the afterpulsing itself. It does exist in the thermal counts scenario in a subtle way. Let's consider the case of no afterpulses and only thermal counts. Then when a start time tag is a thermal count, the probability of the next click (a thermal count) being in the the time interval $[t, t+\Delta t]$ can be calculated by two approaches. The first involves the method we have been using - calculate the probability of occurrence of at least one thermal count in the in the interval $[t, t+\Delta t]$ (given by $1-e^{d-\Delta t}$) and the probability of non-occurrence of the thermal counts in the interval $[0, t]$ (given by e^{-dt}). Although, we have considered a start time tag, we can also zoom out and look at the whole time tag file. Then it is evident that the occurrence / non-occurrence of at at least one thermal count in any time interval is independent of the occurrence / non-occurrence of at at least one thermal count in any other time interval. Thus to find the probability $P^{t\to t+\Delta t}_{\text{next click}}$ of the next click being in the interval $[t, t+\Delta t]$, one can multiply the aforementioned probabilities as follows

$$\begin{aligned} P^{t\to t+\Delta t}_{\text{next click}} &= P^{0\to t}_{NT} \times P^{t\to t+\Delta t}_{T} \\ &= e^{-dt}(1-e^{d-\Delta t}) \end{aligned} \tag{5.29}$$

The second approach would be to consider the probability density function of the occurrence of the next click as a function of time elapsed since the previous on for a Poisson process. This is an exponentially decaying distribution given in our case by de^{-dt}.

So, if the start time tag is a thermal count, the probability $P^{t\to t+\Delta t}_{\text{next click}}$ of the next click occurring in the interval $[t, t+\Delta t]$ is given by integrating the p.d.f. between the limits of the interval

$$\begin{aligned} P^{t\to t+\Delta t}_{\text{next click}} &= \int_t^{t+\Delta t} de^{-dt}\mathrm{d}t \\ &= -e^{-d(t+\Delta t)} + e^{-dt} \\ &= e^{-dt}(1-e^{-d\Delta t}) \end{aligned} \tag{5.30}$$

We don't have to consider the probability of the non-occurrence of thermal counts in the $[0, t]$ bins because the occurrence of the thermal counts in the interval $[t, t+\Delta t]$ implies this, and were not "zoomed out" but have our reference as the start time tag. This is a sanity check that when using the p.d.f of next click of a sub process like thermal count (or afterpulse). Thus, equation 5.22 is wrong when considering the case that the next click is an afterpulse of the start time tag.

To accommodate for this, I considered three separate cases. Case 1, where only thermal counts occur in the interval $[t, t+\Delta t]$. Case 2, where only the afterpulse of the start time tag occurs in the interval $[t, t+\Delta t]$. Case 3, where both thermal counts and the afterpulse of the start time tag occur in the interval $[t, t+\Delta t]$. In all three cases, no detector clicks occur in the interval $[0, t]$

Case 1: The probability P_1 where at least one thermal count occurs in the interval $[t, t+\Delta t]$ is given by

$$\begin{aligned} P_1 &= P_T^{t\to t+\Delta t}\ P_{NT}^{0\to t}\ P_{NA}^{0\to t+\Delta t} \\ &= (1-e^{-d\Delta t})\ e^{-dt}\ \left[1-p_{ap}\left(1-e^{-\lambda_{ap}(t+\Delta t)}\right)\right] \end{aligned} \tag{5.31}$$

Case 2: The probability P_2 where the afterpulse of the start time tag occurs in the interval $[t, t+\Delta t]$ is given by

$$\begin{aligned} P_2 &= P_A^{t\to t+\Delta t}\ P_{NT}^{0\to t+\Delta t} \\ &= p_{ap}\left(e^{-\lambda_{ap}t} - e^{-\lambda_{ap}(t+\Delta t)}\right)\ e^{-d(t+\Delta t)} \end{aligned} \tag{5.32}$$

Case 3: The probability P_3 where both the afterpulse of the start time tag and at least one thermal count (they are independent events) occur in the interval $[t, t+\Delta t]$ is given

by

$$\begin{aligned} P_3 &= P_A^{t\to t+\Delta t}\ P_T^{t\to t+\Delta t}\ P_{NT}^{0\to t} \\ &= p_{ap}\left(e^{-\lambda_{ap}t} - e^{-\lambda_{ap}(t+\Delta t)}\right)\ \left(1-e^{-d\Delta t}\right)\ e^{-dt} \end{aligned} \tag{5.33}$$

The three cases correspond to to mutually exclusive events and hence the probabilities P_i add up. There the probability of the next click being in interval $[t, t+\Delta t]$ is

$$\begin{aligned} P_{\text{next click}}^{t\to t+\Delta t} &= P_1 + P_2 + P_3 \\ &= \left(1-e^{-d\Delta t}\right)\ e^{-dt}\ \left[1 - p_{ap}\left(1-e^{-\lambda_{ap}(t+\Delta t)}\right)\right] \\ &\quad + p_{ap}\left(e^{-\lambda_{ap}t} - e^{-\lambda_{ap}(t+\Delta t)}\right)\ e^{-d(t+\Delta t)} \\ &\quad + p_{ap}\left(e^{-\lambda_{ap}t} - e^{-\lambda_{ap}(t+\Delta t)}\right)\ \left(1-e^{-d\Delta t}\right)\ e^{-dt} \\ &= p_{ap}\ e^{-(\lambda_{ap}+d)t}\left(1-e^{-\lambda_{ap}\Delta t}\right) \\ &\quad + \left(1-e^{-d\Delta t}\right)\ e^{-dt}\ \left[1 - p_{ap}\left(1-e^{-\lambda_{ap}(t+\Delta t)}\right)\right] \end{aligned} \tag{5.34}$$

The p.d.f. of the next detector click as a function of time t elapsed since the start time tag is given by

$$\begin{aligned} \text{p.d.f.}_{\text{next click}}(t) &= \lim_{\Delta t\to 0}\frac{P_{\text{next click}}^{t\to t+\Delta t}}{\Delta t} \\ &= p_{ap}\ e^{-(\lambda_{ap}+d)t}\lim_{\Delta t\to 0}\frac{1-e^{-\lambda_{ap}\Delta t}}{\Delta t} \\ &\quad + e^{-dt}\ \lim_{\Delta t\to 0}\frac{\left(1-e^{-d\Delta t}\right)\ \left[1-p_{ap}\left(1-e^{-\lambda_{ap}(t+\Delta t)}\right)\right]}{\Delta t} \\ &= p_{ap}\ e^{-(\lambda_{ap}+d)t}\lim_{\Delta t\to 0}\frac{1-e^{-\lambda_{ap}\Delta t}}{\Delta t} \\ &\quad + e^{-dt}\left[(1-p)\lim_{\Delta t\to 0}\frac{1-e^{-d\Delta t}}{\Delta t} + \lim_{\Delta t\to 0}\frac{p_{ap}\ e^{-\lambda_{ap}t}\left(e^{-\lambda_{ap}\Delta t} - e^{-(d+\lambda_{ap})\Delta t}\right)}{\Delta t}\right] \end{aligned} \tag{5.35}$$

Let's compute the two limits, naming them I_1, I_2 and $I3$.

$$\begin{aligned} I_1 &= \lim_{\Delta t \to 0} \frac{1 - e^{-d\Delta t}}{\Delta t} \\ &= \lim_{\Delta t \to 0} \frac{1 - \left(1 + (-d)(\Delta t) + \frac{(-d)^2(\Delta t)^2}{2!} + ...\right)}{\Delta t} \\ &= \lim_{\Delta t \to 0} \left(d - \frac{d^2 \Delta t}{2} + \mathcal{O}\left((\Delta t)^2\right)\right) \\ &= d \end{aligned} \tag{5.36}$$

Similarly,

$$\begin{aligned} I_2 &= \lim_{\Delta t \to 0} \frac{1 - e^{-\lambda_{ap}\Delta t}}{\Delta t} \\ &= \lambda_{ap} \end{aligned} \tag{5.37}$$

$$\begin{aligned} I_3 &= \lim_{\Delta t \to 0} \frac{p_{ap}\ e^{-\lambda_{ap} t}(e^{-\lambda_{ap}\Delta t} - e^{-(d+\lambda_{ap})\Delta t})}{\Delta t} \\ &= p_{ap} e^{-\lambda_{ap} t} \lim_{\Delta t \to 0} \frac{\left(1 + (-\lambda_{ap})(\Delta t) + \frac{(-\lambda_{ap})^2(\Delta t)^2}{2!} + ...\right) - \left(1 + (-(d+\lambda_{ap}))\,(\Delta t) + \frac{(-(d+\lambda_{ap}))^2(\Delta t)^2}{2!} + ...\right)}{\Delta t} \\ &= p_{ap}\ e^{-\lambda_{ap} t} \lim_{\Delta t \to 0} \left[\left(-\lambda_{ap} + \frac{\lambda_{ap}^2 \Delta t}{2!} + \mathcal{O}\left((\Delta t)^2\right)\right) - \left(-(d+\lambda_{ap}) + \frac{(d+\lambda_{ap})^2\,\Delta t}{2!} + \mathcal{O}\left((\Delta t)^2\right)\right)\right] \\ &= p_{ap}\ e^{-\lambda_{ap} t}\left[-\lambda_{ap} - (-(d+\lambda_{ap}))\right] \\ &= p_{ap}\ d\ e^{-\lambda_{ap} t} \end{aligned} \tag{5.38}$$

Putting equations 5.36, 5.37 and eqn.third integral into equation 5.35, we have

$$\text{p.d.f.}_{\text{next click}}(t) = p_{ap}(\lambda_{ap} + d)e^{-(\lambda_{ap}+d)t} + (1 - p_{ap})de^{-dt} \tag{5.39}$$

I simulated thermal counts and afterpulses in MATLAB with dark count rate $d = 200\,\text{Hz}$, afterpulsing probability $p_{ap} = 25\%$ and afterpulsing time constant $\lambda_{ap} = 1 \times 10^6\,\text{s}^{-1}$. Dead time and recharge time were set to zero. Figure 5.4 shows the histogram of the time difference between consecutive pulses. The graph is plotted in log-log axes in order to take advantage of the fact that exponential binning was used (which was in turn used to smooth out the tail and magnify the details of the region to the left of the graph

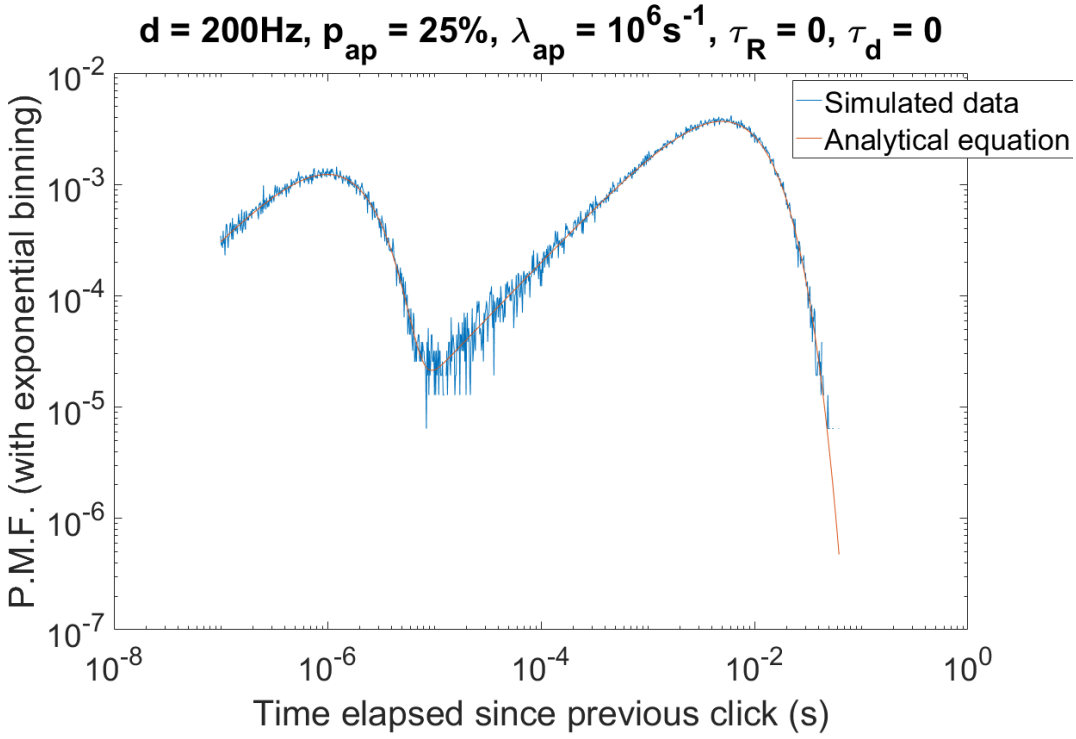


Figure 5.4: A time tag file was simulated using a dark count rate of 200 Hz, afterpulsing probability 25% and afterpulsing time constant $1 \times 10^6\,\mathrm{s}^{-1}$ was simulated. The probability mass function of the next click following an arbitrary click is plotted in blue. Correspondingly, the parameters are used to plot the analytical equation for the same P.M.F. described by equation 5.34

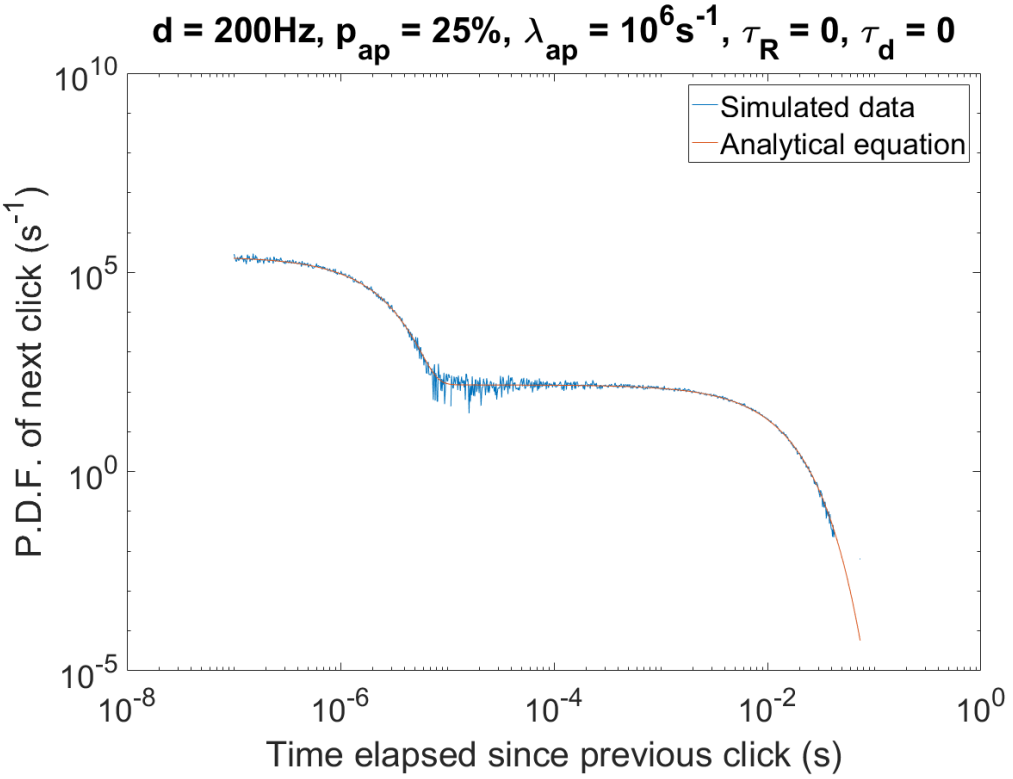


Figure 5.5: A time tag file was simulated using a dark count rate of 200 Hz, afterpulsing probability 25% and afterpulsing time constant $1 \times 10^6\,\mathrm{s}^{-1}$ was simulated. The probability density function of the next click following an arbitrary click is plotted in blue. Correspondingly, the parameters are used to plot the analytical equation for the same P.D.F. described by equation 5.39

where the afterpulses occur). The sum of the counts of all the bins is normalized to unity because there is exactly one "next click" for every start time tag. The curve in blue is the derived from the simulated data set. The hump on the left is the afterpulsing region (containing both afterpulses and thermal counts). The hump on the right comprises primarily of thermal counts. The curve in red is plotted using the analytical equation of the probability of the next click being in the time interval delimited by the edges of each bin as given by equation 5.34. Basically, $P_{\text{next click}}^{t \to t+\Delta t}$ is evaluated for each bin in the histogram with t representing the left edge of the bin and $t+\Delta t$ representing the right edge of the bin. There is a close match between the two curves indicating that the analytical expression approximates the histogram curve very closely. I say approximates because we haven't considered the possibility of afterpulses of time tags other than the start time tag as being the next click after the start time tag. This is justifiable because $\lambda_{ap} >> d$. One can use the expression $\lambda_{ap} e^{-\lambda_{ap} t}$ to show that the average time for detrapping of a trapped charge carrier is $\frac{1}{\lambda_{ap}}$. Therefore in this case the time scale of afterpulsing is much smaller than the average time period of the thermal counts. This ensures that chances all successive generation of afterpulses created by any given thermal count occur before the thermal count occurs is very high. Thus, even by not considering afterpulses of time tags other than the start time tag, one can derive a good analytical approximation to the true histogram.

Figure 5.5 shows the p.d.f. counterpart of the next click distribution. From the histogram derived from simulated data, we divide the counts in each bin by the width of that bin. This is the curve in blue. The curve is normalized automatically such that the area under the curve is unity. The curve in red is plotted using the analytical equation of the p.d.f of next click given by equation 5.39. Here too, one sees a close match between the curves.

In our experimental data (derived from physical experiments with APD's), it is often the case that λ_{ap} and d are separated by order of magnitude apart, thus making this derivation suitable for analyzing the afterpulsing probability for experimental data. In particular, for $\lambda_{ap} >> d$, we can drive the sum $\lambda_{ap} + d \to \lambda_{ap}$ and show that equation 5.39 condenses to a sum of exponential given below.

$$\text{p.d.f.}_{\text{next click}}(t)\Big|_{\lambda_{ap} >> d} = p_{ap} \lambda_{ap} e^{-\lambda_{ap} t} + (1 - p_{ap}) d e^{-dt} \tag{5.40}$$

Since $e^{-\lambda_{ap} t}$ decays faster than e^{-dt}, the right most region of the histogram is dominated

by the e^{-dt} exponential. Thus,

$$\text{p.d.f.}_{\text{next click}}(t)\Big|_{\lambda_{ap}>>d;t>>0} = (1-p_{ap})de^{-dt} \tag{5.41}$$

Taking the logarithm on both sides of equation 5.41, we get

$$log(\text{p.d.f.}_{\text{next click}}(t)) = -d \times t + log\left[(1-p_{ap})d\right] \tag{5.42}$$

This is a straight line with slope as $-d$ and y-intercept as $log\left[(1-p_{ap})d\right]$. Thus, linear curve fitting the right most region of the the log(normalized bin count rate) in the histogram derived from experimental data, one can find the thermal count rate. This fact is made use of in the afterpulsing algorithm I had worked on described in section 2.2.9.

Equation 5.42 represents a straight line with slope as $-d$ and y-intercept as $log\left[(1-p_{ap})d\right]$

Since the analytical expression agrees with the simulated data very closely towards the right of the graph which is dominated by thermal counts, one can reliably curve fit this region and compare it to equation 5.42 to calculate afterpulsing probabilities using both methods previously described.

5.4 Summary and Outlook

In section 5.2 the observed afterpulsing probability was shown to vary with detector parameters. In specific, an analytical expression was derived to show the explicit dependence of the afterpulsing probability on the APD's thermal count rate, dead time and the afterpulsing probability value without dead time. Furthermore, the observed afterpulsing probability was shown to decrease with the thermal count rate, al other parameters held constant.

If one looks at equation 5.17, one sees that there is no restriction on how big the thermal count rate can get. However, if thermal counts themselves induce dead time regions (preventing other thermal counts from occurring), the number of dead time regions in a time tag file of fixed finite size cannot be more than $\frac{\text{Time duration of time tag file}}{\text{dead time}}$. This also means the number of thermal counts cannot be more than this value. Thus the thermal count rate cannot be more than $\frac{\frac{\text{Time duration of time tag file}}{\text{dead time}}}{\text{Time duration of time tag file}} = \frac{1}{\text{dead time}}$.

So the thermal counts in section 5.2 are actually potential thermal counts and are associated with charge carriers being produced by thermal activation in defects in the APD substrate. Such events are in fact Poisson distributed in time. However, not all

these events give rise to output pulses because some of them will occur in dead time regions. Hence, there is also this notion of observed thermal counts which are not Poisson distributed in time. This affects the derivation of the analytical expression of the observed afterpulsing probability in key place - during the calculation of the probability of having no thermal counts in a time interval which was obtained from the Poisson distribution. Since the detector clicks due to thermal counts are not Poisson distributed in time, one would need to work out the corresponding probabilities.

Another source of improvement would be to incorporate the effect of afterpulses of time tags other than the start time tag in the derivation for observed afterpulsing probabilities. These afterpulses will also induce dead time regions following the start time tag and decrease the observed afterpulsing probability.

In section 5.3.1, we derived an analytical expression for the p.m.f. and p.d.f. of the time distribution of the next detector click following any arbitrary click from first principles. The resulting analytical expressions were validated against simulated data with negligible dead time and recharge time. The primary aim of the derivation was to theoretically demonstrate that the rightmost region of the graph consisted of thermal counts under the assumption that the time scale for afterpulsing was much smaller than that for thermal counts. The rightmost part of the p.d.f. curve was shown to be a simple decaying exponential which can be curve fitted. This was central to the working of the afterpulsing algorithm.

The derivation for the p.d.f. of the next click did not account for afterpulses of time tags other than the start time tag. This isn't problematic as long as the time scale for afterpulsing is much shorter than time scale for thermal counts because all successive generations of thermal counts will in all likelihood occur long before the next thermal count occurs. So, when considering an arbitrary click (thermal or afterpulse) as the start time tag, the chances that (first generation) afterpulses of clicks other than the start time tag will be the next click is very small. More importantly, this assumption allows the rightmost region to be free of afterpulsing, thus allowing for a simple exponential consisting of only thermal counts in this region. However, if the time scales of afterpulsing and thermal counts are comparable to each other, the chances that the afterpulses of time tags other than the start time tag being the next click are not negligible. One would then have to address the probabilities associated with this. More importantly, there would n't be a distinct thermal region and hence curve fitting and extrapolation is not an option. The afterpulsing algorithm itself must be altered to accommodate for this. In such a case, a successful theoretical derivation incorporating afterpulses of time tags other than start time tag might be able to provide insight on how to tackle the problem.

Recharge time and detection efficiency can also be incorporated into the derivation.

They affect the probabilities of occurrence or non-occurrence of detection events in each bin. Furthermore, the implementation of a dead time is fairly easy. The only change is to drive the p.d.f. curve to zero in the range $t = [\ 0,\ \tau_d\ )$

I have assumed a simple model of afterpulsing in this section. It contains only one exponential. I would like to review some of the afterpulsing models that have been proposed in literature. Ref [33] uses a similar assumption of a single exponential. Ref [34] suggests that there might be deep level traps of different detrapping time constants. Thus, the average detrapping time might be a result of a sum of exponentials, each having a unique characteristic time constant corresponding to the different types of deep level traps. The same paper [34] also suggests that a power law dependence on time (with a negative integer as the power) might be a good model as well. For future work, it might be worth considering the sum of exponentials as the effective time distribution for detrapping of the trapped charge carrier. The coefficients associated with the exponentials will be related to the spatial density of the each type of deep level trap in the APD's substrate. In particular in section 5.3.1, it would be interesting to see if the analytical equation of the PMF and PDF of the next click is of the same form as that for the experimental data. If the afterpulsing detrapping time distribution of a physical detector is indeed a sum of exponentials, one should be able to curve fit the experimental PMF and PDF with the resulting analytical equations.

*

Chapter 6

Quantum Key Distribution Simulation

6.1 Introduction and Motivation

When I was working on an algorithm to calculate the afterpulsing probability, I wanted to ascertain its accuracy. I had time-tag files taken from dark count measurement experiments on the DM. However, I could never really know the true afterpulsing probability of the APD's and didn't have a way to test my algorithm as I was developing it. The time tag files of dark count measurements consisted of time tags due to thermal counts and afterpulses. Both were probabilistic events drawn from certain time dependent distributions. For example, the thermal counts are a Poisson process which follow the Poisson distribution. Also, each avalanche on average produced an afterpulse with an probability called the afterpulsing probability and the detrapping time of the charge carrier followed an exponentially decaying distribution. Such distributions could be simulated in algorithm thus generating a time tag file of known parameters. I was able to do exactly this to ascertain the accuracy of my algorithm. I had initially conceived this idea and the initial part of the simulation was worked on jointly by me and Dr. Jean Philippe Bourgoin.

In the Winter term of 2018, as part of a final project of the course QIC 750 at the University of Waterloo, I decided to take this a step further and simulate a quantum link for the QKD protocol. The QKD setup consists of a laser source and single photon detectors. APD's are a good choice for the detectors for reasons as mentioned in section 1. APD's have a host of detector parameters like thermal counts, afterpulsing probability, recharge time, timing jitter, etc. Although the impact of these parameters on quantities like the Quantum bit error ratio (QBER) and key generation rate is known to some extent, in order to study the simultaneous effect of the source and detector parameters on the aforementioned quantities, physical experiments have to be conducted. Such experiments

are a crucial step in demonstrating the commercial realization of QKD. However, fine tuning the different parameters can be a laborious and time consuming task, involving redesigning the experimental set up, physical components or supporting software. A simulation of the QKD protocol can help optimize the different parameters of the experimental set up. This then helps to design physical components with these optimized parameters.

Currently, there have been attempts in literature towards simulation of QKD distribution. However, all of them are based on analytical equations theoretically deriving the secret key rate for different QKD scenarios. For example, ref [35] mentions analytical equations for secret key rate as a function of transmission of the quantum channel for discrete and continuous variable QKD protocols with and without decoy states. Ref [36] shows similar graphs for QKD protocols using entangled photon sources, again using analytical equations. However, these equations do not seem to consider many detector parameters like dead time, recharge time and afterpulsing characteristics. Even if these quantities are negligible under nominal conditions, it would still be insightful to see how the key rates may change if these values were substantial. The simulation presented in this section does not depend on analytical equations directly. Instead, it tries to simulate the individual physical processes like photon emission, afterpulsing, dead time, recharge time, etc. in code. The protocol of QKD itself is simulated and a key is generated. The key rate and QBER are then inferred from this key. With this approach, one can vary any of the modeled parameters and see its effect on the sifted key rate and QBER. Moreover, one can potentially explore the modeled parameter space to optimize QBER and sifted key rate with this simulation.

In this chapter, I have simulated a laser source (section 6.3), single photon detectors (section 6.4) and the QKD protocol (section 6.5). A brief discussion of these topics is in section 6.2 before going into details in the successive sections. An intercept and resend attack is also simulated in section 6.5.4. The QKD quantum link is simulated and the effects of the different parameters of the system on the sifted key generation rate and QBER are explored.

6.2 Methodology

I have modeled an implementation of the QKD protocol in MATLAB. I have assumed the BB84 protocol here. The sender, nicknamed Alice, prepares a photon in a certain polarization state and sends it to the receiver, nicknamed Bob, who measures the photon in a basis of his choice. The choice of polarization state by Alice and the choice of measure-

ment basis by Bob are implemented using pseudo-random numbers. Also, the process of measurement is simulated using pseudo-random numbers. The laser source used by Alice to generate single photons is modeled as a weak coherent pulsed laser. The single photon detectors used by Bob to measure the polarization state of the photon are modeled as avalanche photodiodes (APD). I have modeled the APD's thermal counts, afterpulses, dead time, detection probability and timing jitter. These quantities have been explained in chapter 2. I have also implemented a coincidence algorithm that matches the time tags corresponding to a photon being produced and measured. Using this, I generate a raw key and a sifted key. I also calculate the quantum bit error ratio (QBER) and the sifted key rate. I then vary the parameters of the laser source, detectors and the coincidence algorithm to see how the QBER and sifted key rate change in response. Finally, I simulate the presence of a attacker, nicknamed Eve, who uses an intercept and resend attack. I also see how the QBER and sifted key rate change with the probability of Eve's interception of the photon. The QBER in particular is important because this allows Alice and Bob to predict the presence of an eavesdropper. I have refrained from using MATLAB's quantum optics toolboxes because I wanted to code from scratch and also wanted to understand the details of how each component is modeled.

A note about the validity of such a simulation. I am attempting to simulate the QKD protocol on a classical computer. I want to delineate on what grounds this simulation is valid. The process of measurement in the QKD protocol is a quantum mechanical one. Bell's theorem says that the laws of quantum mechanics cannot be reproduced by using local hidden variables. However, in my simulation, I generate pseudo-random numbers in MATLAB and use this to simulate the process of measurement. Pseudo-random numbers are generated by deterministic processes and the algorithm has to keep track of the underlying variables. Hence, the simulation of measurement is not an accurate one. This means that one cannot draw conclusions about a single photon that was measured or a single bit in the final key. However, one must note that the wave function in quantum mechanics is a deterministic function. From it, the probability space over the measured outcomes can be derived in a deterministic way. This probability space is a classical object and can be modeled using pseudo-random numbers. This means that if one has a specific quantum sate, the measurement outcomes of the quantum state, on the one hand, using a quantum mechanical process of measurement, and on the other hand, using pseudo-random number generator in MATLAB, are statistically indistinguishable from each other. In this simulation, I refrained from making conclusions about single bits but only made conclusions on aggregate properties like QBER and sifted key rate because statistical properties of the key remain unchanged whether one use quantum mechanical measurement or pseudo-random

numbers. Similar arguments hold for the choice of measurement basis which is usually accomplished by a 50-50 beam splitter in experiment which uses a quantum mechanical process. However, one can still ask if a certain value of a modeled detector parameter is better than some other value. This is because the definition of the parameter does not require a quantum process. This is not true however when asking questions about a single bit in the final sifted key.

6.3 Laser Source

The laser source generates photons. It was modeled as a weak coherent pulsed laser. The laser triggers at a constant rate. For my simulation, I chose a rate of 5 MHz. Each laser trigger is one iteration in a giant *for* loop. During any laser trigger, the laser produces k number of photons, where k is a natural number. The probability distribution P over the number of photons produced during a single laser trigger is given by the Poisson distribution [37].

$$P(k) = e^{-\mu}\frac{\mu^k}{k!} \tag{6.1}$$

where

- k is the number of photons produced during a given single laser trigger
- μ is the average number of photons produced during a single laser trigger

Figure 6.1 shows a Poisson distribution with $\mu = 5$

One must note that the photons produced during a given laser trigger are produced at the same time with no delay in between them. However, they are not necessarily produced exactly when one expects the laser to trigger. There is some uncertainty which shows up in the timing jitter - this will be addressed in section 6.4.5.

In practice, one needs to carefully select the average number μ of photons during a laser trigger. If one selects very small value of μ, it results in low key generation rate. At first sight, it seems like a higher values of μ are beneficial. However, high values of μ make the protocol more prone to attacks. This is because high values of μ results in a greater proportion of multi-photon events compared to single photon event. When a multiple photons are transmitted to Bob at the same time over a quantum channel, they have the same polarization state. If an eavesdropper, Eve, can detect a multi-photon state

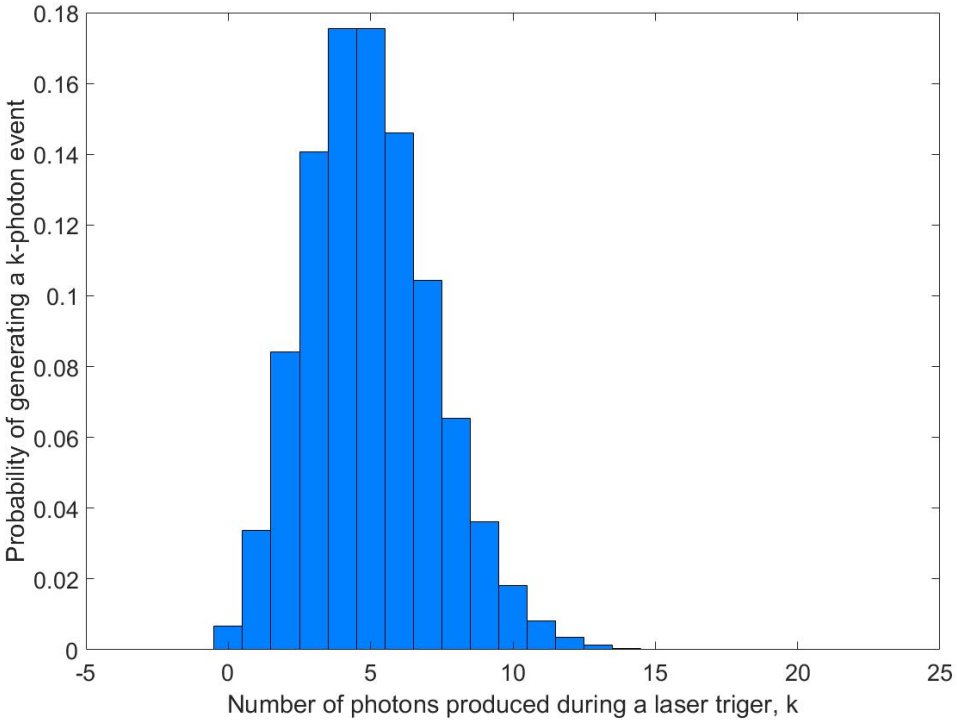


Figure 6.1: Probability distribution of generating k photons during a laser trigger is a Poisson distribution. This figure shows a Poisson distribution where the average number of photons during a laser trigger is $\mu = 5$

in the channel and seize only one of the photons, she now has access to the same quantum state that Bob will receive. Under the assumption that photons in a multi-photon event are not entangled, Eve's measurement of her seized photon doesn't affect the outcome of Bob's measurement. However, there is a caveat here in that the number of photons in a multi-photon state affects the detection probability (addressed in section 6.4.4) of the multi-photon at Bob's side. If Bob is able to detect such changes in his detection of Alice's photons, he might be able to still detect Eve's presence. In any case, Alice would want to maximize the generation of single-photon events. To see how μ affect the generation of multi-photon events, consider the following ratio

$$\frac{\text{Probability of 1-photon event}}{\text{Probability of 2-photon event}} = \frac{e^{-\mu}\frac{\mu}{1!}}{e^{-\mu}\frac{\mu^2}{2!}} = \frac{2}{\mu} \tag{6.2}$$

Thus increasing μ increases the proportion of single-photon event compared to a 2-photon event. Similar ratios can be found for n-photon events, where $n > 2$ and the same conclusion holds true.

Thus one needs to find an optimal value for μ. For my simulation, I chose $\mu = 0.5$. Please note that I haven't tried to explicitly optimize μ to arrive at this value.

So, I created one huge *for* loop to cycle through the laser triggers. During each trigger, I sampled from the Poisson distribution with mean value $\mu = 0.5$ to decide on the number of photons in each iteration of the loop. To do this I used the *poissrnd()* function in MATLAB to sample from the Poisson distribution.

6.4 Single Photon Detector

Bob needs a detector to detect individual photons that Alice sends him. There are many choices of single photon detectors. Avalanche photodiodes have distinct advantages - low dark count rates, high detection efficiency and do not require cryogenic cooling. I have attempted to model and simulate different characteristics of the avalanche photo diode.

6.4.1 Thermal Counts

APD's tend to produce output avalanches even when no photons are incident on the detector substrate. This results in background noise. These output clicks are called thermal

counts. Thermal counts occur randomly and are Poisson distributed in time. A characteristic of Poisson distributed events is that the time difference between two consecutive events follows an exponentially decaying probability distribution.

It would be interesting to discuss a little bit about the connection between a random process and the Poisson distribution. For a generic probability time distribution of the occurrence of the next click, if at a randomly chosen time instant, you were told when the previous click occurred and were also asked to generate a time distribution of the next click as a function of time elapsed since this time instant, the resulting distribution would be dependent on when the previous click occurred and in general would not be the same as the original distribution. This cannot be representative of a random process like thermal counts. Instead, if you started off with the exponentially decaying distribution and repeated the same experiment just described, the resulting time distribution would be exactly the same exponentially decaying distribution. This means that the occurrence of a given thermal count doesn't depend on when any other thermal count occurred. Consider the normalized probability time distribution P of the next thermal count

$$P(\text{next thermal count}) = de^{-dt} \tag{6.3}$$

where

- d is the thermal count rate with respect to time
- t is the time elapsed since the previous thermal count

Say, at some time instant you were told that the previous thermal count occurred a time t_0 before the the current moment. To get the time distribution of the next click as a function of time elapsed from the current moment, first translate the origin of time axis to the previous click. Then you would have to consider the curve from time $t = t_0$ onwards and renormalize the area under the curve from $t = t_0$ to $t = \infty$ because the next click will occur at some point in the future with 100% probability.

Area under the curve from $t = t_0$ onwards is given by

$$\int_{t_0}^{\infty} de^{-dt} \mathrm{d}t = e^{-dt_0} \tag{6.4}$$

To ensure you are measuring elapsed time from the present moment, translate curve in equation 6.3 towards the left by amount t_0 so that the present moment is at the origin of

the time axis. The resulting curve is $de^{-d(t+t_0)}$. Now, to renormalize this curve, divide it by the area calculated in equation 6.4 to get

$$\frac{1}{e^{-dt_0}} de^{-d(t+t_0)} = de^{-dt} \tag{6.5}$$

where t is the time elapsed since the current moment.

This is exactly the distribution we started out with. Thus, the occurrence of the next thermal count doesn't depend on how long ago the previous click occurred. Thermal counts occur randomly and independent of each other in this sense. We have established that the exponentially decaying distribution embodies this randomness. There is then a standard derivation that assumes the exponential distribution for the next click and shows that the resulting events are Poisson distributed. This is the basis for the postulate that thermal counts are Poisson distributed under the assumption that they occur randomly and independent of each other.

Since thermal counts occur randomly, I used the *rand*() function in MATLAB to randomly populate the time tag file with thermal counts. To do this, first specify the thermal count rate, say d. Then, if the length of the time tag file is t', dt' is the number of thermal counts one would expect to find in the timetag file on average. However, if one repeats this experiment multiple times, one would expect the number of counts in the time tag file to vary a tiny bit. The number of thermal counts in the time tag file will be Poisson distributed with mean value of the distribution equal to dt'. Thus the following command in MATLAB populates a time tag file of length t' with thermal count rate d

```
1 timeTage = t'*rand(1, poissrnd(d*t'));
```

The code above gives the unordered set of time tags corresponding to thermal counts. These time tags are pseudo-random in nature due to the way the *rand*() function works. But as mentioned in section 6.2, this is valid as long as we don't make conclusions on individual thermal counts but on aggregate properties like the thermal count rate. Also an interesting sanity check is using MATLAB to calculate the distribution of the time difference between consecutive clicks of such simulated thermal counts to arrive at the exponentially decaying distribution.

6.4.2 Afterpulsing

APD's generate output through avalanching. Basically one of the electrons in the substrate moves to the conduction band and physically knocks out some electrons which in turn each knock out a few more electrons. Eventually, one has an avalanche of electrons in the substrate. Sometimes, one gets a second avalanche that is correlated with a previous avalanche. This process is divided into three sub-processes:

1. The detector substrate contains deep level imperfections which can trap a moving electron. A mobile electron during an avalanche can get trapped in such an imperfection / trap-site. The expected number of electrons getting trapped in deep levels during an arbitrary avalanche is called the afterpulsing probability p_{ap}. In this simulation, I assume that at most one electron gets trapped during an avalanche. In such a scenario, the afterpulsing probability can also be defined as the probability of having a trapped electron during an avalanche.

2. An electron that gets trapped eventually gets detrapped if you wait long enough. The time distribution for detrapping as a function of time elapsed since it got trapped is given by the normalized exponentially decaying distribution

$$\lambda_{ap} e^{-\lambda_{ap} t} \tag{6.6}$$

   where

   - t is the time elapsed since the electron got trapped
   - λ_{ap} is the afterpulsing time constant

   A special mention must be made here that the exponentially decaying nature of the time distribution for detrapping makes the detrapping process of the electron random in the sense that the distribution of the time elapsed from the present moment for detrapping a trapped electron does not depend on how long ago that electron got trapped. This stems from a similar argument as the one used in section 6.4.1.

3. The detrapped electron triggers a second avalanche because it knocks neighboring electrons as it gets detrapped. This second avalanche is called an afterpulse. The time scale of a detrapped electron causing an avalanche is assumed to be negligible.

A few assumptions:

1. Less than one electron gets trapped during an avalanche on average. This is a good assumption from an operational point of view because if on average more than one electron got trapped at a trap-site, then one detector click on average will produce more than one detector click, leading to a runaway effect with the count rate eventually approaching to ∞. In my simulation, during any given avalanche, there can be exactly one or no electrons getting trapped.

2. Avalanches are indistinguishable in nature from each other. One cannot look at an avalanche and determine what caused it. A photon, thermal count and even an afterpulse can generate avalanches that look the same. Hence, any afterpulse can potentially generate a further afterpulse through the process described above with the same afterpulsing probability and time constant.

To simulate the afterpulse of any given detector click in MATLAB, I sampled the time distribution in equation 6.6 to get a value, say t_1. I then placed a time tag a time t_1 seconds after the the detector click in question. This simulates the afterpulse. If I perform this process for every time tag, I will have simulated a 100% afterpulsing probability. To simulate an afterpulsing probability of p_{ap}, I only perform this process with probability p_{ap} - not all detector clicks give rise to an afterpulse. This condition is got by the following statement in MATLAB

```
1 if rand < p_ap
```

All thermal counts, signal photon detections and previously generated afterpulses are candidates for generating an afterpulse. I also labeled each thermal count and signal photon with a unique identifier. Afterpulses inherited the label of the parent thermal count or signal photon. This way I could keep track of which click generated any given afterpulse. However, I did not use these labels to perform any computation that Alice or Bob do since they do not have access to these labels.

Note: MATLAB doesn't have an inbuilt function to sample from the exponential function. One can perform this sampling by first computing the cumulative distribution function (C.D.F.) of the probability distribution function (P.D.F) that one wants to sample. Let's call this C.D.F. as function U. One then inverts U to express t as a function of U. Then if one samples for values of U from the uniform distribution (*rand* function in MATLAB) and calculates the corresponding values of t, they will assume a distribution corresponding to the initial P.D.F. you started out with.

For example, say you want to sample from the exponentially decaying distribution $\lambda_{ap}e^{-\lambda_{ap}t}$. The corresponding C.D.F. is

$$U(t) = \int_0^t \lambda_{ap}e^{-\lambda_{ap}t'}\mathrm{d}t' = 1 - e^{-\lambda_{ap}t} \tag{6.7}$$

Then invert U to express t as a function of U

$$t = -\frac{1}{\lambda}log(1-U) \tag{6.8}$$

t ranges from 0 to ∞. Correspondingly from equation 6.7, one can see that U ranges from 0 to 1. If you then sample for values of U from the uniform distribution in the closed interval $[0, 1]$, and for each sampled value of U calculate the value of t using equation 6.8, the values of of t are distributed according to the original P.D.F. $\lambda_{ap}e^{-\lambda_{ap}t}$. The following code was used in MATLAB to sample from the exponential distribution.

```
1 - log(rand)/lambdaAfterpulse
```

Please note that I used $rand$ instead of $1 - rand$ because for a uniform distribution, U, over the closed interval $[0, 1]$, U and $1 - U$ statistically give the same samples.

Every generated detector click, be it thermal count, signal photon or an afterpulse itself, is a candidate for generating an afterpulse. I cycle through all detector clicks in the time tag file and generate chains of afterpulses conditional to the afterpulsing probability. The following code shows this. $BDet1$ is the time tag file of Bob's first detector. Its first row stores the time tags. Its second row stores the identifier corresponding to the time tag.

```
1  n = length(BDet1(1,:)); %Length of dark count + signal photonarray
2  for i = 1:n
3      apulses=BDet1(1:2,i);
4      j=1;
5      while (1)
6          if rand<p_ap
7              apulses(:,j+1)=[apulses(1,j) - ...
                   log(rand)/lambdaAfterpulse; BDet1(2,i)];
8              j=j+1;
9          else
10             break
11         end
12     end
```

```
13      m = length(BDet1(1,:));
14      if length(apulses(1,:))>1
15          BDet1(:,m+1:m+length(apulses(1,:))-1)=[apulses(:,2:end); ...
                -2*ones(1,numel(apulses(1,2:end)))];%append all afterulses ...
                of 'i'th thermal count to 'array' but do not append the ...
                thermal count itself that produce the afterpulses
16      end
17  end
18  BDet1=sortrows(BDet1',1)';
```

6.4.3 Dead Time

After an avalanche occurs, the APD is inactive for a period of time. This duration of time is called the dead time, τ_d.

Time difference between two detector clicks cannot be smaller than τ_d. This means that if a particular click is *scheduled* to occur within time τ_d following a detector click, it cannot physically occur. I term such clicks as *illegal clicks*. Also, by *scheduled*, I mean either a thermal click was supposed to occur or a photon was incident on the detector or a trapped electron got detrapped.

The simulation can simulate detector clicks in chronological order. Although it makes sense to implement it this way, I take advantage of the fact that the thermal counts can be generated in one line of code because the *rand*() function can generate an array of random numbers as this is computationally very efficient. However, in doing so, one must remember that some of these thermal counts may be illegal. One has to simulate photon clicks and afterpulses in the vicinity of a thermal click to determine whether or not it is illegal. So, I first generate each type of click in the time tag file in a batch before moving on to simulate the next type of click. Then I iterate through each click and determine whether it is illegal by checking to see if a click has occurred a value τ_d before that particular click. If it is illegal, I delete the click. I also delete all it's successive afterpulses (which have already been generated in code) since none of them should have occurred in the first place. The code that implements this is presented in section 6.4.4.

6.4.4 Detection Probability

Following the dead time period, the APD doesn't detect a scheduled detector click with 100% efficiency. Instead, the probability of detection rises and asymptotes to a maximum

value as elapsed time since the first click approaches ∞. To explain this in more detail, during the dead time period the over-voltage on the APD drops to zero. Over time, it increases and asymptotes to the set over-voltage. The variation of over-voltage with respect to time is given by the following equation

$$V = V_0(1 - e^{-\frac{\Delta t}{\tau_R}}) \tag{6.9}$$

where

- Δt is the time elapsed since the first click
- V is the over-voltage corresponding to elapsed time Δt
- V_0 is the set over-voltage
- τ_R is the recharge time which sets the time scale for the rise of over-voltage

Here, an assumption is that no other click occurs after the dead time period has passed and the scheduled click has occurred.

The efficiency of detection of a scheduled click at any given time depends on the over-voltage applied to the APD at that instant in time. The variation of detection efficiency with respect to over-voltage is given by the following equation

$$\eta_{recharge} = 1 - e^{-\frac{V}{V_0}} \tag{6.10}$$

where $\eta_{recharge}$ is the probability of detection of a somewhat ideal detector whose detection probability asymptotes to 100% over time.

Putting equations 6.9 and 6.10 together, we have the probability of detection as a function of time elapsed

$$\eta_{recharge} = \frac{1}{1 - e^{-\frac{20}{8.5}}}(1 - e^{-(1-e^{-\frac{\Delta t}{\tau_R}})\frac{20}{8.5}}) \tag{6.11}$$

However, for physical detectors the detection efficiency doesn't asymptote to 100%. The APD's that I used for the experiments had about 60% maximum efficiency. Therefore, equation 6.11 has to be rescaled.

The net probability of detection depends on the type of click. For thermal counts and afterpulses, the detection probability $\eta_{thermal/afterpulse}$ is the rescaled recharge efficiency given by

$$\eta_{thermal/afterpulse} = \eta_{recharge} * \eta_{DetMax} \tag{6.12}$$

where η_{DetMax} is the detection efficiency of a thermal count or afterpulses after an infinite time has passed.

The detection probability for a signal photon has an additional factor of optical efficiency of the path it travels from the laser to the detector. Realistically, there will be some loss of photons in the optical path and one must account for this. For a single photon, the probability of detection $\eta_{\text{single photon state}}$ is given by

$$\eta_{\text{single photon state}} = \eta_{optical} * \eta_{recharge} * \eta_{DetMax} \tag{6.13}$$

where $\eta_{optical}$ is the optical efficiency of the path of the photon en route from the laser to the detector

Note that only $\eta_{recharge}$ varies over the time tag file as it depends on the time elapsed since the previous click. The other factors are constant for a given experimental setup.

But what happens when the laser produces multiple photon in a given laser trigger. Corresponding to any k-photon state, the detector can only avalanche once because all photons are assumed to arrive at the detector simultaneously. Since each photon has some probability of detection given by equation 6.13, the probability of detection of a multi-photon state is higher than that of a single photon state because at least one of the photons must be detected to get an avalanche output. The paper [37] suggests that the detection probability $\eta_{k-photon\ state}$ of a k-photon state emitted by a WCP laser source is given by

$$\eta_{k-photon\ state} = 1 - (1 - \eta)^k \tag{6.14}$$

where k is the number of photons emitted by the WCP source in a given laser trigger

To simulate the detection probability, I use the same approach as that for the dead time. I first generate all types of detector clicks in the time tag file. I then iterate through all clicks in chronological order. To determine whether a given detector click is illegal or not, I check the time difference between that click and the previous click. This gives the

Δt factor in equation 6.11. I then calculate the probability of detection η of the second click (depending on the type of second click). Then I delete the click in consideration with a probability $1-\eta$ which is the probability of not detecting the second click. If I delete a click, I also delete all its successive afterpulses.

The following block of code in MATLAB is used to delete clicks based on both the dead time criterion and the detection probability criterion:

```
    %Delete illegal avalanches for BDet1
j=numel(BDet1(1,:));
i=2;
t0=BDet1(1,1);
flag = 0;
while i≤j
    if BDet1(1,i)≠-5 %Ian: Do not evaluate if the afterpulse has not ...
        actually occured due to a non-occurence of a previous ...
        generation avalanche
        if BDet1(1,i)-t0<deadTime
            BDet1(1,[false(1,i-1),BDet1(2,i:end)==BDet1(2,i)])=-5;%Ian: ...
                Deletes i'th avalanche and all its successive afterpulses
            flag = 1;
        else
            etaRecharge = ...
                (1-exp(-(1-exp(-(BDet1(1,i)-t0)/rechargeTime))...
                            *20/8.5))/(1-exp(-20/8.5)); % has max ...
                                value of 1(fully recharged)
            etaPhoton = etaRecharge * etBdetectorMax * ...
                etaOpticalTransmission;
            if BDet1(3,i)> 0
                if rand > 1 - (1 - etaPhoton)^Bdet1(3,i) %rand> ...
                    because probability of not detcting
                    Bdet1(1,[false(1,i-1),Bdet1(2,i:end)==Bdet1(2,i)])=-5;...
                        %Ian: Deletes i'th avalanche and all its ...
                            successive afterpulses
                    flag = 1;
                end
            else %Thermal pulse or any afterpulse
                if rand > etaRecharge  * etBdetectorMax %rand> ...
                    because probability of not detcting
                    Bdet1(1,[false(1,i-1),Bdet1(2,i:end)==Bdet1(2,i)])=-5;
                        %Ian: Deletes i'th avalanche and all its ...
                            successive afterpulses
                    flag = 1;
```

```
                end
            end
        end

        if flag == 0
            t0=Bdet1(1,i);
        end
        flag = 0;
    end
    i=i+1;
end
Bdet1=sortrows(Bdet1',1)';
numIllegalAvalanches = sum(Bdet1(1,:)==-5);
Bdet1=Bdet1(:,numIllegalAvalanches+1 :  end);
```

Please note that the third row in $Bdet1$ array stores a -1 to identify a thermal count, a -2 to identify an after pulse or a unique natural number ascribed to each photon emitted by the WCP source (thus identifying a photon click).

6.4.5 Timing Jitter

The emitted photon is time tagged by Alice. However, Alice's time tags may not be periodically spaced as one might be lead to believe from the previous sections. The laser is triggered periodically and the photon is released during a small time window following each trigger. However, there is some uncertainty in position along the window that the photon is released though a release closer to the center of the window is more probable than towards the ends of the window. Furthermore, an electric pulse is generated at the start of the window to notify that the laser has been triggered. However, there is some uncertainty associated with the exact time the electric pulse is generate relative to the start of the window. The uncertainties due to the electric pulse and photon emission is together assumed to be Gaussian by the central limit theorem. Finally, the time tagger has a finite time resolution, i.e., it cannot distinguish between events that are separated in time by a value smaller than a some fixed value, $t_{tagResA}$. When a time tag t_0 has been registered, the event takes place in the interval $[t_0, t_0 + t_{tagResA}]$. Since no knowledge can be obtained about when in that interval the event actually took place, one can assume a normalized uniform time distribution over the interval $[t_0, t_0 + t_{tagResA}]$. The central limit theorem says that the combined distribution of time tags will be Gaussian for large number of samples and the combined variance is the sum of variances of variances of the individual process

mentioned above. Thus, the combined uncertainty σ_{Alice} as

$$\sigma_{Alice} = \sqrt{\sum_{i=1}^{3} \sigma_i} \tag{6.15}$$

where different values of i corresponds to the different sources of uncertainty on Alice's side.

The expected time position of the electric trigger of Alice's source is periodic. The expected time for the photon to be released following the start of the time window is assumed to be known. The time tag is registered after an expected value of $t_{tagResA}/2$ following the electric reference pulse of the laser. Given these assumptions, the expected positions of the time tags that Alice registers are periodic. However, due to the uncertainties mentioned before, the actual positions of the registered time tags are spread out about the expected positions of the time tags according to a Gaussian distribution of standard deviation σ_{Alice}.

Similarly, Bob time tags his received photon. When the photon is incident on the detector, the avalanche may not occur instantaneously. Instead, there will be a small delay which can be modeled as a Gaussian with non-zero mean (the expected delay $t_{avalanche}$). Also as described before, the time tagger has a finite resolution $t_{tagResB}$. On average, the time tag will be registered $\frac{t_{tagResB}}{2}$ following a photon detection. Again one assumes a uniform time distribution over the time tagger's window. The combined distribtuion of timetags on Bob's side will be Gaussian distributed about the expected value again due to the central limit theorem.

The combined certainty in time tags experienced by each Alice and Bob are called timing jitters. In general, Alice and Bob experience different values of timing jitter. However, for my code, I assumed them to be equal to each other. This doesn't critically alter the nature of graphs in my results but only changes their absolute values. The timing jitter values can be easily changed. In simulation, during each laser trigger, I sampled the Gaussian distribution using the *randn*() function in MATLAB.

Note 1: The overall jitter experienced by Bob's time tags relative to Alice's time tags ultimately incorporates Alice's jitter as well.

Note 2: The optical path time for the photon is known to both Alice and Bob, and is constant throughout the QKD protocol. In fact in my simulation, I have subsumed the expected values of the aforementioned delays (e.g., delay in time tagging, delay in source emitting a photon following an electric trigger, etc.) into the optical path time variable in the code. One could keep them separate but they would eventually add up. Figure 6.2

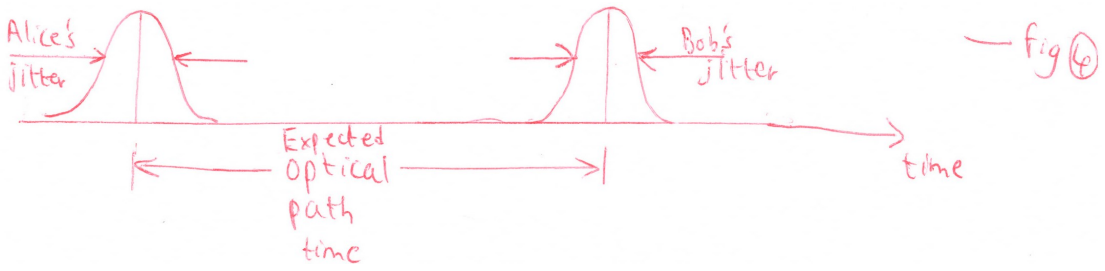


Figure 6.2: A schematic of the model of the timing jitter. The Gaussian on the left represents the net uncertainty in Alice's time tags. The Gaussian on the right represents the net uncertainty associated with Bob's time tags. The expected value of the two Gaussians are separated in time by the optical path time that the photon takes to travel from Alice's source to Bob's detector. All expected values of delay are subsumed into the optical path time

shows a schematic of the timing jitters on Alice and Bob's side, separated by an optical path time.

I have assumed that the time taken for the photon to travel from Alice's source to Bob's detector is accurately known.

The following code implements timing jitter. The expected value of the start of the laser time window in the WCP source is periodic and is represented by the *sourceTime* variable. The number of photons in a laser trigger is decided according to the Poisson distribution. Then, Alice and Bob's time tags are computed using the Gaussian to account for the timing jitter.

```
1 for sourceTime = startTime : 1/sourceFreq :  endTime
2     numPhotons = poissrnd(mu); % number of photons in a given laser ...
          pulse. poisson ditribution with mean 'mu'
3     if numPhotons > 0 %if at least 1 photon is produced in the laser ...
          pulse
4
5         alicePhotonTT = sourceTime + jitter*randn; %Time Alice ...
              PREPARES photon
6         bobPhotonTT   = sourceTime + opticalPathTime + jitter*randn; ...
              %Time Bob receives photon
7 end
```

6.5 QKD Protocol

Alice has a laboratory where she houses her WCP source. She is able to produce photons in a particular polarization state . All photons in a k-photon state are in the same polarization state. She employs two orthogonal basis sets. One basis set is the Horizontal-Vertical (HV) or rectilinear basis set whose basis states are photons whose polarization vectors point along the spatially horizontal $|H\rangle$ and vertical $|V\rangle$ directions. The other basis set she uses is the Diagonal-AntiDiagonal (DA) or diagonal basis set whose basis states are photons whose polarization vectors point along the spatially diagonal $|D\rangle$ and anti-diagonal $|A\rangle$ directions. Each basis set spans the same 2 dimensional Hilbert space and hence can be viewed as a change of coordinates in the Hilbert space. The following set of equations state the relationship between the normalized basis states of each basis set.

$$|D\rangle = \frac{1}{\sqrt{2}}(|H\rangle + |V\rangle) \tag{6.16}$$

$$|A\rangle = \frac{1}{\sqrt{2}}(|H\rangle - |V\rangle) \tag{6.17}$$

However, when one says that Alice prepares a photon in the HV basis, it means she prepares a photon in one of its two basis states, i.e., a photon with horizontal or vertical polarization. Figure 6.3 shows a schematic of the two orthogonal basis sets (reprinted from [38]).

When Alice wants to send a photon, she first chooses one of the two basis sets randomly. She then chooses at random one of the basis states in the chosen basis set. Alice then prepares k-photon state in the chosen polarization state. Because of the random choices, the photons can be either in the H, V, D or A polarization state each with probability 0.25 . She then sends this k-photon state to Bob. I simulate these random choices using the $rand()$ function in MATLAB.

When Bob receives the photon, he chooses one of the two basis sets to measure the received photon state in. He can implement this choice passively by using a 50-50 polarization independent beam splitter. He then measures the polarization state in the chosen basis set. He can implement this using a polarization dependent beam splitter. The photon then collapses to one of the polarization basis states of the chosen basis set. The basic laws of quantum mechanics corresponding to measurement and collapse of quantum state apply. For example, if Alice sends a photon in the $|H\rangle$ polarization state and Bob measures it in the HV basis set, the photon collapses to $|H\rangle$ state with 100 % probability. However, if

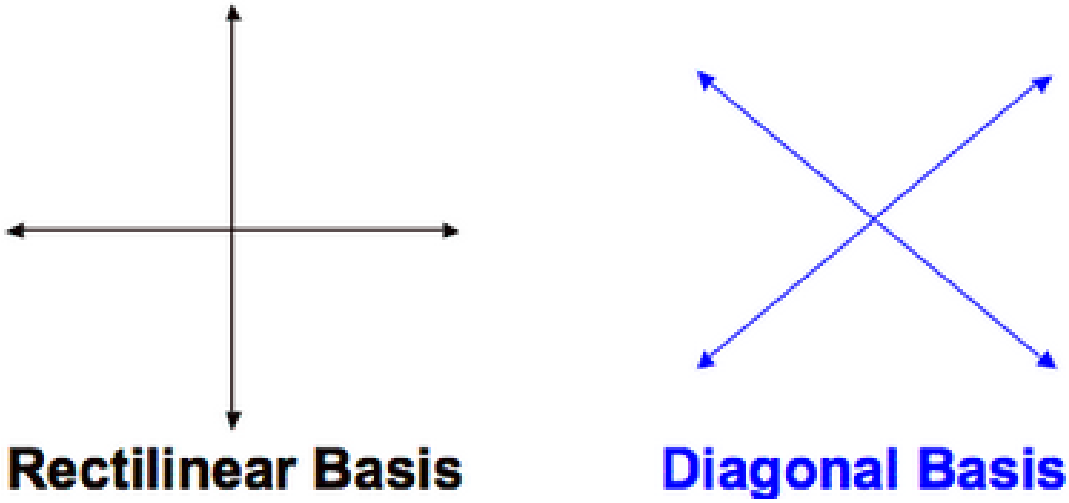


Figure 6.3: [38]A schematic of the two basis sets in the QKD protocol. The rectilinear basis set consist of basis states corresponding to photons in the horizontal and vertical polarization states. The diagonal basis set consists of basis states corresponding to photons in the diagonal and anti-diagonal polarization states

Alice sends a photon in the $|D\rangle$ polarization state and Bob measures it in the HV basis set, the photon collapses to $|H\rangle$ state with 50 % probability and to $|V\rangle$ state with 50 % probability. In general, if Alice and Bob use the same basis set to prepare and measure the photon respectively, Bob's measurement outcome will be the same polarization state as the one Alice prepared the photon in. However, if their choice of basis set is different, Bob's measurement outcome has a 50% chance of being along either of the basis states of his chosen basis set. I implement the random choice of basis set and the process of measurement by Bob using the $rand()$ function in MATLAB.

A critical assumption here is that when Alice prepares or Bob measures a photon in a particular polarization state, there is no error or offset in the direction of polarization in the 2-D Hilbert space.

An assumption I made here is that Bob has four detectors to detect the $|H\rangle$, $|V\rangle$, $|D\rangle$ and $|A\rangle$ states. In particular, this makes sense for Bob because the two beam splitters he uses creates four physically separate optical paths at the output. Hence, he needs four separate detectors to detect photons along each path. Another motivation for having four detectors is that the time tag files contain no information about the polarization state of an incident photon. So, having a different detector for each polarization state lets you identify the polarization of the photon by identifying the detector in which photon's time tag occurred.

The following block of code stores time tags corresponding to photon emission from the WCP source for Alice.

```
        numPhotons = poissrnd(mu); % number of photons in a given ...
            laser pulse. poisson ditribution with mean mu

    alicePhotonTT = sourceTime + jitter*randn; %Time Alice PREPARES ...
        photon
    aliceTimeTags(:,i) = [alicePhotonTT; indentifier; numPhotons];
    indentifier = indentifier + 1;


    num = ceil(4*rand); % Generate random number for Alice to decide ...
        which basis and what polirazation within the basis Alice ...
        prepares her photon in
    if     num == 1
        alicePreparesOutcome = 'H';
        alicePreparesOutcomes(i) = 'H';
        alicePolarDirection(i) = 1;
        i = i + 1;
    elseif num == 2
        alicePreparesOutcome = 'V';
        alicePreparesOutcomes(i) = 'V';
        alicePolarDirection(i) = 2;
        i = i + 1;
    elseif num == 3
        alicePreparesOutcome = 'D';
        alicePreparesOutcomes(i) = 'D';
        alicePolarDirection(i) = 3;
        i = i + 1;
    elseif num == 4
        alicePreparesOutcome = 'A';
        alicePreparesOutcomes(i) = 'A';
        alicePolarDirection(i) = 4;
        i = i + 1;
    end
```

The following block of code simulates time tags corresponding to photons incident on Bob's four detector. The code checks to see the true polarization state of the photon that Alice sent it in and simulates the quantum mechanical measurement process.

```
%Bob measures photon
```

```
2
3           num2 = ceil(2*rand); %Random number for passive 50-50 beam ...
                splitter to decide which basis Bob measures in
4           if num2 == 1 %Bob measures in HV basis
5               if alicePreparesOutcome == 'H' %Alice prepared phton as 'H'
6                   BDet1 = [BDet1, [bobPhotonTT; indentifier; numPhotons]];
7                   indentifier = indentifier + 1;
8                   bobMeasuresOutcomes = 'H';
9                   bobMeasuresBasis = 'HV';
10              elseif alicePreparesOutcome == 'V' %Alice prepared phton ...
                    as 'V'
11                  BDet2 = [BDet2, [bobPhotonTT; indentifier; numPhotons]];
12                  indentifier = indentifier + 1;
13                  bobMeasuresOutcomes = 'V';
14                  bobMeasuresBasis = 'HV';
15              else %Alice prepared in DA basis instead
16                  num3 = ceil(2*rand);% Random number to decide if ...
                        photon prepared as eiter D or A is measured by Bob ...
                        - collapses to H or V
17                  if num3 == 1
18                      BDet1 = [BDet1, [bobPhotonTT; indentifier; ...
                            numPhotons]];
19                      indentifier = indentifier + 1;
20                      bobMeasuresOutcome = 'H';
21                      bobPreparesBasis = 'HV';
22                  else %num3 = 2
23                      BDet2 = [BDet2, [bobPhotonTT; indentifier; ...
                            numPhotons]];
24                      indentifier = indentifier + 1;
25                      bobMeasuresOutcome = 'V';
26                      bobPreparesBasis = 'HV';
27                  end
28              end
29          else %num2 = 2 ---- Bob measures in DA basis
30              if alicePreparesOutcome == 'D' %Alice prepared phton as 'D'
31                  BDet3 = [BDet3, [bobPhotonTT; indentifier; numPhotons]];
32                  indentifier = indentifier + 1;
33                  bobMeasuresOutcomes = 'D';
34                  bobMeasuresBasis = 'DA';
35              elseif alicePreparesOutcome == 'A' %Alice prepared phton ...
                    as 'A'
36                  BDet4 = [BDet4, [bobPhotonTT; indentifier; numPhotons]];
37                  indentifier = indentifier + 1;
38                  bobMeasuresOutcomes = 'A';
```

```
39                  bobMeasuresBasis = 'DA';
40              else %Alice prepared in HV basis instead
41                  num10 = ceil(2*rand);% Random number to decide if ...
                        photon prepared as eiter H or V is measured by Bob ...
                        - collapses to D or A
42                  if num10 == 1
43                      BDet3 = [BDet3, [bobPhotonTT; indentifier; ...
                            numPhotons]];
44                      indentifier = indentifier + 1;
45                      bobMeasuresOutcome = 'D';
46                      bobPreparesBasis = 'DA';
47                  else %num10 = 2
48                      BDet4 = [BDet4, [bobPhotonTT; indentifier; ...
                            numPhotons]];
49                      indentifier = indentifier + 1;
50                      bobMeasuresOutcome = 'D';
51                      bobPreparesBasis = 'DA';
52                  end
53              end
54          end
```

6.5.1 Photon Preparation and Measurement

6.5.2 Coincidence Algorithm

As will be explained in section 6.5.3, in order to generate the sifted key, one needs to find out those preparation / measurement instances of photons where Alice and Bob both used the same choice of basis set. This means that they should first be able to match the time tag corresponding to the emission of a photon in Alice's time tag file to the time tag corresponding to the detection of the same photon in Bob's time tag file. However, both time tag files are littered with time tags from thermal counts and afterpulses in addition to other signal photon emissions/detections. It is the job of the coincidence algorithm to determine such signal photon coincidences.

Since Alice and Bob are separated in space, there is a finite time t_{trans} needed for the signal photons to travel from the WCP source to the APD detector. Thus, ideally, time tags corresponding to the same preparation / measurement instance of photons should be separated in time by an amount t_{trans}. However, one must keep in mind timing jitter in Alice's and Bob's time tags. This jitter creates some uncertainty in the value of t_{trans} for each instance.

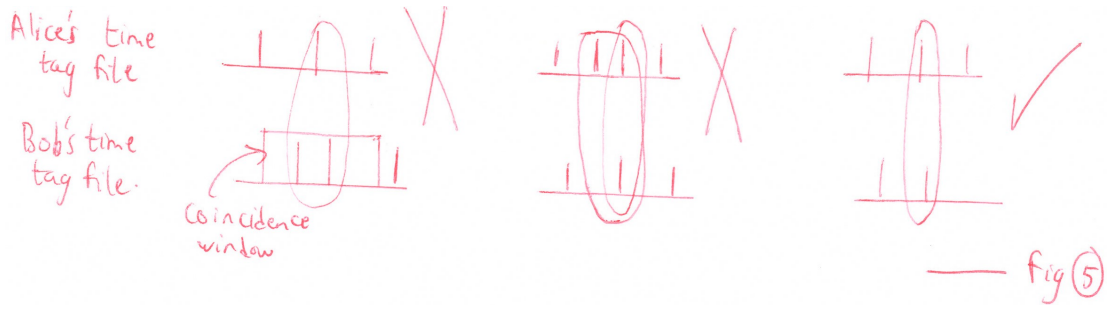


Figure 6.4: Illustration of one-to-one mapping by coincidence algorithm

If one wants to find a photon time tag in Bob's file that corresponds to a given photon time tag in Alice's file. The coincidence algorithm checks for one-to-one mappings between time tags in Alice's and Bob's files that separated by value t_{trans} in time up to t_{coinc}, where t_{coinc} is the called the coincidence threshold. To simplify the description of the coincidence algorithm, let's assume that $t_{trans} = 0$ for now. The simulation of course assigns it a positive value. The coincidence threshold is the maximum absolute time difference a time tag in Bob's file can have from Alice's given photon time tag to be considered as a potential match (or coincidence). Say, for example, one wants to find out the photon coincidence in Bob's file corresponding to a particular time tag, say a_{12}, in Alice's file. Here *12* represents the 12^{th} time tag in the Alice's sorted time tag file. If the algorithm finds multiple time tags in Bob's file within a time window of width t_{coinc} around a_{12}, it deems that there exists no (unique) coincidence of a_{12}. But let's say that we find exactly one such time tag, say b_{49}, in Bob's file. This does not yet satisfy the one-to-one mapping condition. So, the algorithm then checks the reverse way to see how many time tags exist in Alice's file that are within a time window of width t_{coinc} around b_{49}. By tautology, a_{12} should satisfy this condition. However, if the algorithm finds multiple such time tags that satisfy the condition, it deems that a_{12} has no (unique) photon coincidences in Bob's file. But if only a_{12} satisfied the condition, it deems a_{12} and b_{49} to be photon coincidences - a_{12} and b_{49} correspond to photon emission and detection, respectively.

The following block of code shows the implementation of the coincidence algorithm in MATLAB.

```
1
2 aliceTTOutcomesArray = sortrows([aliceTimeTags(1,:)...
3                                  ;alicePolarDirection...
4                                  ;aliceTimeTags(3,:)]',1)';
5 bobTTOutcomesArray   = sortrows([BDet1(1,:), BDet2(1,:), BDet3(1,:), ...
    BDet4(1,:)...
```

```
;1*ones(1,numel(BDet1(1,:))), 2*ones(1,numel(BDet2(1,:)))...
,3*ones(1,numel(BDet3(1,:))),4*ones(1,numel(BDet4(1,:)))
;BDet1(3,:), BDet2(3,:), BDet3(3,:), BDet4(3,:)]',1)';

aliceTTarray = aliceTTOutcomesArray(1,:); % adjusting for the finite ...
    time for transmission of phton from Alice to Bob
bobTTarray = bobTTOutcomesArray(1,:);

for i = 1:numel(aliceTTarray)
    A = abs(bobTTarray - (aliceTTarray(i) + opticalPathTime)) < ...
        coincidenceThreshold;
    if sum(A)== 1
        B = abs((aliceTTarray + opticalPathTime) - bobTTarray(A)) < ...
            coincidenceThreshold;
        if sum(B)== 1
            coincidenceTT = [coincidenceTT, ...
                [aliceTTarray(B);bobTTarray(A)]];
        end
    end
end
```

The coincidence threshold must be finely tuned. A small coincidence threshold might increase accuracy slightly. This because majority of the area under the Gaussian (which is used to model the jitter) is around the expected value. This increases the chances that the closer one looks around the expected value of the Gaussian, one is more likely to find signal photons when compared to thermal counts (which are more uniformly distributed in Bob's file with respect to Alice's given signal photon time tag) and afterpulsing. However, the penalty for making the window too small is that the coincidence algorithm misses photon coincidence because the jitter introduces some uncertainty around the expected value, hence reducing key rate. On the other hand, if one makes the coincidence threshold large, a one-to-one mapping of counts in Alice's and Bob's files may be more difficult to achieve.

6.5.3 Key Generation

The key is a string of zeros and ones that is used to encode a message string before transmission and is also used to decode the encoded message on the receiver end. Thus Alice and Bob need to have the copies of the same key. The temporally ordered set of photon coincidences form the what is known as the raw key. Say, for example, in the HV basis, one attributes the state $|H\rangle$ to bit *0* and the state $|V\rangle$ to bit *1*. Similarly, in the DA

basis, one attributes the state $|D\rangle$ to bit *0* and the state $|A\rangle$ to bit *1*. After the coincidence algorithm, if one looks at the temporally ordered set of photon coincidences as a string of zeros and ones, one set held by Alice and the other by Bob, one can in the ideal case expect a 25% error (mismatch) in bit value in the same index position in the two bit strings. This was partly explained in section 6.5 and is due to the orthogonality of the rectilinear and diagonal basis set.

One must then note that in the ideal case, the errors are only occur when the choice of preparation / measurement basis sets by Alice and Bob are not the same. Thus, one can eliminate this error by choosing only those photon coincidences where Alice and Bob made the same choice of preparation / measurement basis sets. This, on average, reduces the key size to half of that of the raw key. The sifted key is defined as the temporally ordered subset of the raw key containing only bits where Alice and Bob's choice of preparation / measurement basis sets matched.

Realistically, there are many sources of errors in the sifted key.

- The coincidence algorithm doesn't perfectly match photons time tags. Timetags corresponding to thermal counts and afterpulsing may also deemed as photon coincidences. This is because all sources of time tags are probabilistic events. Since thermal counts and afterpulsing are not correlated to the choice of basis sets, they can seep into the sifted key,introducing errors.

- The polarization encoding apparatus by Alice and the measurement apparatus by Bob are assumed to be accurately aligned relative to each other. Because the polarization vectors have spatial components, any mismatch of such alignment causes the expected outcomes of Bob's measurements to change. For example, if both Alice and Bob use the same basis set and Alice encodes her photon with vertical polarization $|V\rangle$ but Bob's measurement apparatus corresponding to the HV basis set is slightly offset compared to Alice's preparation apparatus, there is a non-zero probability that the measured outcome by Bob will have a polarization $|H\rangle$. The magnitude of this probability will depend on the extent of the offset. This will introduce errors in the sifted key.

- An eavesdropper, Eve, could intercept a photon sent by Alice and measure it in a basis set of her choice. If she uses a basis set different from the one chosen by Alice, she can introduce errors in the sifted key because her choice of basis set in independent of Alice and Bob's choices of basis sets. This will be addressed in detail in section 6.5.4.

Such errors can compromise ones ability to securely use the sifted key to encode/decode a message string. One needs to quantify the magnitude of the mismatch between Alice and Bob's copies of the sifted key to judge whether the key can be used securely. The quantum bit error ratio (QBER) is the expected error per bit in the sifted key. More elaborately,

$$\text{QBER} = \frac{\text{Number of mismatched bits between sifted keys}}{\text{Total number of bits in each sifted key}} \tag{6.18}$$

In order to estimate the QBER, Alice and Bob need to find out the number of bit mismatches in the sifted key. One way to do this is for Alice to transmit her copy of the key over to Bob and then Bob can calculate the number of mismatches. Since the sifted key is a classical bit string, it must be sent over the classical channel which is prone to attacks. Also, it doesn't make sense to encode this classical sifted key before sending it because you need a secure key to do this and this whole protocol is itself trying to generate one. Thus, Alice can send a small subset of her key, say, the first 5% of her copy of the key. Bob can then compare this to the first 5% of his copy of the key. Thus Bob can determine the QBER. The compared portions of the keys (the first 5%)is discarded by both. As the number of bits in the compared portion of the key increases the estimated QBER approaches the true QBER of the two copies of the key. The *estimated* QBER is calculated by the aforementioned procedure and the *true* QBER is the QBER calculated by considering the entire sifted key. If the calculated QBER is greater than a pre-decided threshold, the QKD protocol must be restarted.

Another quantity one wants to keep track of is at what rate can the sifted key be generated. The key generation rate is defined as the ratio of the number of bits in the sifted key to the number of laser trigger pulses required to generate that key. Since the laser is triggered at almost a constant rate, the key generation rate is closely linked to the number of key bits generated per unit time.

The following block of code generates the raw key and sifted key. It also calculates the true QBER and the estimated QBER.

```
1 if ¬isempty(coincidenceTT)
2     %Generate raw key
3     aliceCoincidenceIndex = ismember(aliceTTarray, coincidenceTT(1,:));
4     aliceRawKey = aliceTTOutcomesArray(2, aliceCoincidenceIndex);
5     bobCoincidenceIndex = ismember(bobTTarray, coincidenceTT(2,:));
6     bobRawKey = bobTTOutcomesArray(2, bobCoincidenceIndex);
7     fprintf('error in raw key = %d\n',sum(aliceRawKey≠ ...
          bobRawKey)/numel(aliceRawKey)*100);
```

```
8
9       % Generate sifted key
10      sameBasis = ceil(aliceRawKey/2) == ceil(bobRawKey/2); %Figures ...
            out indices in alice and Bob's raw key where they ...
            prepared/measured in the same basis
11      aliceSiftedKey = aliceRawKey(sameBasis);
12      bobSiftedKey = bobRawKey(sameBasis);
13      QBER = sum(aliceSiftedKey≠ ...
            bobSiftedKey)/numel(aliceSiftedKey)*100; %This is the true QBER
14      if isequal(aliceSiftedKey, bobSiftedKey) == 1
15          fprintf('Alice and Bob have same sifted keys\n')
16      else
17          fprintf('Alice and Bob have different sifted keys. True QBER ...
                is %d %%\n', QBER)
18      end
19      %NOTE: The second row in aliceTTBasisArray is the outcome value, ...
            isn't the basis choice
20
21      % Estimattion of QBER by Alice and Bob: Take 5% of the key and ...
            compare
22      numEstKey = ceil(numel(aliceSiftedKey)*0.05);%number of bit froom ...
            the left to devote for estimating QBER
23      aliceEstimationKey = aliceSiftedKey(1:numEstKey);%Part of alice's ...
            key devoted to estimation
24      bobEstimationKey = bobSiftedKey(1:numEstKey);%Part of bob's key ...
            devoted to estimation
25      if isequal(aliceEstimationKey, bobEstimationKey) == 1
26          fprintf('Alice and Bob estimate that they have same sifted ...
                keys \n')
27      else
28          estimatedQBER = sum(aliceEstimationKey≠ ...
                bobEstimationKey)/numel(aliceEstimationKey)*100;
29          fprintf('Alice and Bob estimate that they have different ...
                sifted keys. Estimated QBER is %d %%\n', estimatedQBER)
30      end
31
32      fprintf('Size of final key = ...
            %d\n',numel(aliceSiftedKey)-numel(aliceEstimationKey))
33
34  else
35      break
36  end
```

In my simulation I calculated the true QBER except for the case of Eve's attack, where

I calculated the estimated QBER. The primary reason for calculating the true QBER in simulation was that a large key size was required to get an accurate value of the estimated QBER. But sometimes when key rate drops as I change certain experimental parameters, I am required to run the simulation for longer periods of time but MATLAB has an upper bound on the size of arrays. However, since the estimated QBER approaches the true QBER as the number of bits in the transmitted key increases, the results of the simulation are still valid.

6.5.4 Attack

Alice sends her polarization encoded photons over a quantum channel. Under the assumption that this channel is unsecure (open), an eavesdropper, Eve, could gain access to the transmitted photons. A host of attacks could be performed by Eve to gain polarization information of the photons without being detected.

One of the simplest attacks is an intercept and resend attack. Eve is assumed to have a laboratory of her own which is located physically close to Bob's. Eve intercepts the photons that Alice sends to Bob. Eve then measures the polarization state of the photons using a detection apparatus which has 100% detection efficiency. Eve randomly choose a basis set between the HV or DA basis and proceeds to measure the polarization state of Alice's photon. Eve also has an ideal photon source which can generate photons in the same polarization state as she measured Alice's photon to be in. Eve also can detect the number of photons in Alice's multi-photon state accurately. Eve then can use her source to simultaneously generate the same number of photon she detected in Alice's k-photon state, all photons being in the same polarization state as she measured Alice's photon in.

If Eve happens to choose the same basis set as Alice does during a given laser pulse, Eve's presence is undetectable. This is because Eve's measurement doesn't change the polarization of the photon. On the other hand if Eve chooses a basis set different from Alice in a given laser pulse, there is a 50% chance of changing the polarization state of the basis set and hence changes Bob's statistical outcome. This eventually introduces errors in the sifted key. The lecture [39] states and briefly explains that the QBER introduced by the presence of Eve if she intercepts every photon is 25%. Here I attempt to describe why this is so. To probe this in a bit more detail, let's only look at cases where Alice and Bob use the same choice of basis set because these are the only instances which the sifted key retains. First let's start by looking at the case when Eve is absent. Let's assume in the ideal case, the QBER is zero. This is illustrated in figure 6.5 and is straightforward. In the presence of Eve, let's assume she intercepts every k-photon state that Alice sends. Let's look at the

case where both Alice and Bob both use the HV basis. Say, Alice generates her photon in the $|H\rangle$ state. If Eve chooses to measure in the HV basis set, she doesn't change the state. Hence, Bob also measures the polarization state of the photon as $|H\rangle$. The expected error is zero corresponding to this series of outcomes. This is illustrated in figure 6.5 in the left-most branch. The letters H, V, D and A within the circle are polarization states prepared in or measured. The number along each straight line is a conditional probability which is the probability that the next person measured the photon in the state denoted by the letter at the bottom of the line given the fact that the previous person measured / prepared the photon in the state denoted by the letter at the top of the line. On the other hand if Eve uses the DA basis to measure Alice's photon, half the time Eve measures it in the $|D\rangle$ state, the other half in the $|A\rangle$ state. When Eve measures in the $|D\rangle$, Bob has a 50-50% chance of measuring the photon in the $|H\rangle$ or $|V\rangle$. It is the measurement of the photon state as $|V\rangle$ by Bob that induces an error in the sifted key because Bob's bit in his copy of the sifted key now flips with respect to Alice's copy. The expected error due to this is $\frac{1}{8}$ as illustrated in middle leg of figure 6.5. Finally a similar calculation shows that if Eve had measured the photon as $|A\rangle$, the expected error due to this would also be $\frac{1}{8}$. The net error and hence QBER due to Eve's presence is $\frac{1}{8} + \frac{1}{8} = 25\%$. Thus, Alice and Bob can estimate their QBER by using a small fraction (for the simulation, I used 5%) of their sifted keys. If the QBER exceeds a threshold, they can determine the presence of Eve - under the assumption that experimental setup itself induces negligible errors.

Eve also has a choice of intercepting a k-photon state sent by Alice with a probability $p_{int} \leq 1$. Looking at the explanation in the previous paragraph, corresponding to $p_{int} = 0$, QBER $= 0$ and corresponding to $p_{int} = 1$, QBER $= 0.25$. Since QBER is the expected error per bit in the sifted key, one predicts $0 <$ QBER < 0.25 corresponding to $0 < p_{int} < 1$.

Note: Eve doesn't induce a time delay in the system. This is because her lab is assumed physically close to Bob's. The distance traveled by the photon from Alice to Bob doesn't change in Eve's presence. This means that the optical path time of the photon doesn't change. It is also assumed that Eve's detection of Alice's $k-$photon state and her own generation of the same k- photon state is instantaneous. All this means that Bob cannot detect Eve's presence by a shift in the expected time to receive Alice's photon. Instead, Bob must rely on QBER estimation to detect Eve's presence.

The following code in MATLAB implements Eve's intercept and resend attack with probability p_{int}

```
% Eve attacks
if eveAttacks == 1
```

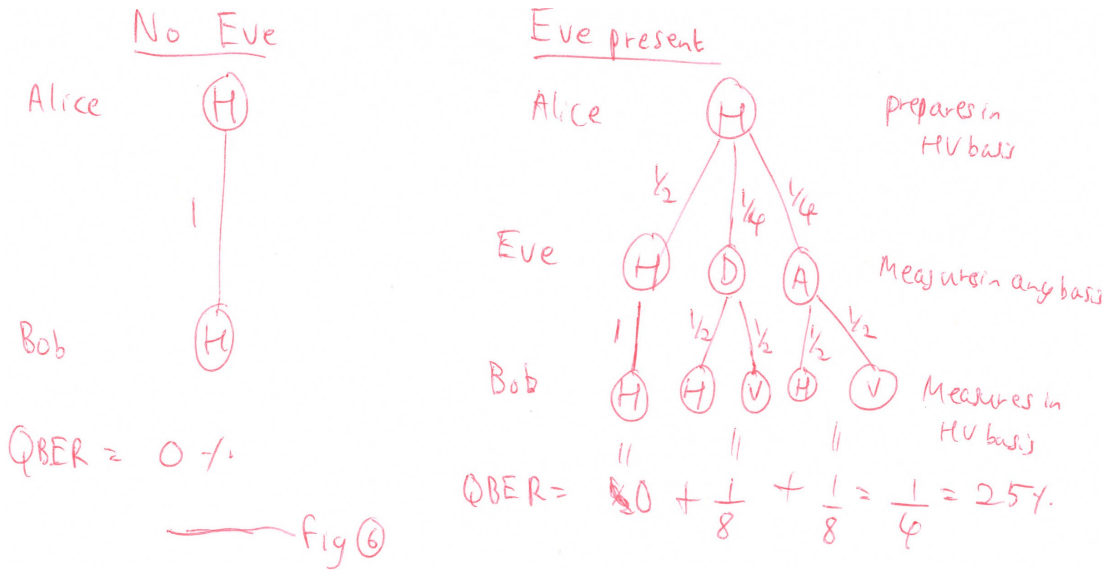


Figure 6.5: The left part shows the QBER when no eavesdropper is present. When an eaves

```
3       if rand < eveAttackProbability
4           num5 = ceil(2*rand); %Random number for passive 50-50 beam ...
                splitter to decide which basis Bob measures in
5           if num5 == 1 %Eve measures in HV basis
6               if alicePreparesOutcomes(end) == 'H' %Alice prepared ...
                    phton as 'H'
7                   EDet1 = [EDet1, evePhotonTT];
8                   alicePreparesOutcome = 'H'; %Eve's sends this state ...
                        to Bob hoping he weill think it's from Alice
9               elseif alicePreparesOutcomes(end) == 'V' %Alice prepared ...
                    phton as 'V'
10                  EDet2 = [EDet2, evePhotonTT];
11                  alicePreparesOutcome = 'V'; %Eve's sends this state ...
                        to Bob hoping he weill think it's from Alice
12              else %Alice prepared in DA basis instead
13                  num7 = ceil(2*rand);% Random number to decide if ...
                        photon prepared as eiter D or A is measured by Bob ...
                        - collapses to H or V
14                  if num7 == 1
15                      EDet1 = [EDet1, evePhotonTT];
16                      alicePreparesOutcome = 'H'; %Eve's sends this ...
                            state to Bob hoping he weill think it's from Alice
17                  else %num7 = 2
18                      EDet2 = [EDet2, evePhotonTT];
19                      alicePreparesOutcome = 'V'; %Eve's sends this ...
```

```
                        state to Bob hoping he weill think it's from Alice
20                  end
21              end
22          else %num5 = 2 ---- Eve measures in DA basis
23              if alicePreparesOutcomes(end) == 'D' %Alice prepared ...
                    phton as 'D'
24                  EDet3 = [EDet3, evePhotonTT];
25                  alicePreparesOutcome = 'D'; %Eve's sends this state ...
                        to Bob hoping he weill think it's from Alice
26              elseif alicePreparesOutcomes(end) == 'A' %Alice prepared ...
                    phton as 'A'
27                  EDet4 = [EDet4, evePhotonTT];
28                  alicePreparesOutcome = 'A'; %Eve's sends this state ...
                        to Bob hoping he weill think it's from Alice
29              else %Alice prepared in HV basis instead
30                  num8 = ceil(2*rand);% Random number to decide if ...
                        photon prepared as eiter H or V is measured by Bob ...
                        - collapses to D or A
31                  if num8 == 1
32                      EDet3 = [EDet3, evePhotonTT];
33                      alicePreparesOutcome = 'D'; %Eve's sends this ...
                            state to Bob hoping he weill think it's from Alice
34                  else %num8 = 2
35                      EDet4 = [EDet4, evePhotonTT];
36                      alicePreparesOutcome = 'A'; %Eve's sends this ...
                            state to Bob hoping he weill think it's from Alice
37                  end
38              end
39          end
40      end
41  end
```

6.6 Results and Discussions

A simulation of the QKD protocol with the source and detector was implemented in MATLAB. In this section, I will cover the results of the simulation and discuss them a bit. One of the goals of the simulation was to see how different parameters of the experimental setup affects the QBER and the sifted key generation rate. This information can in turn be used to fine tune these parameters to decrease QBER and increase key generation rate. The ideal way to go about this is to first decide on a range of acceptable QBER values for secure key generation. Then one can fine tune the experimental parameters to maximize the key

generation rate under the constraint of maintaining the QBER in operational range. The next few subsections look at how each of the individual experimental parameters affects the QBER and key generation rate.

Although the details of simulating each experimental component was covered in sections 6.3, 6.4 and 6.5, I will briefly outline the main steps of the simulation. The laser trigger frequency was set to 5 MHz. The entire algorithm lives inside a giant *for* loop with each iteration corresponding to a single laser trigger. The average number of photons emitted during a given laser trigger was set to 0.5 . The number of photons emitted during a given laser trigger was calculated by sampling a Poisson distribution with mean value 0.5 . The net timing jitter experienced by Alice in her time tags was set to 300 ps. The time stamp in Alice's file corresponding to the emission of a k-photon state is calculated by sampling a Gaussian distribution with mean value centered around the expected instant of the laser trigger and uncertainty equal to 300 ps. Eve may attack with a certain probability p_{int}. If during a laser pulse, she does intercept a photon, she measures it. The simulation of measurement has been described in section 6.5. Then Bob measures it in the same manner. The expected time taken by photons from Alice to Bob, i.e. optical path time, was set to 300 ns. Then thermal counts are generated in all detectors. I assumed a thermal count rate of 100 Hz in each detector. Then the afterpulsing on signal photons and thermal counts is generated in all detectors. I used an afterpulsing probability of 10%. The afterpulsing time constant λ_{ap} was set to $1 \times 10^6\,\mathrm{s}^{-1}$. Then the illegal detector clicks were deleted according to dead time and detection probability criteria. The dead time was set to 500 ns. The recharge time was set to 1 µs. The detector's maximum possible detection efficiency was set to 60%. The optical efficiency of the optical path that the photons take from Alice to Bob was set to 70%. The coincidence algorithm then finds the photon coincidences. The coincidence threshold was set to 10 ns. Finally, the sifted key is generated and the QBER and key rate are calculated. The QBER was calculated as the true QBER evaluated using the entire sifted key. The only exception is in section 6.6.8 which deals with the attack scenario, where the QBER is calculated as the estimated QBER.

In each of the following sections, I selected one experimental parameter and varied it over a range of values. For each value, I ran the simulation 5 separate times each time running the simulation for 10 ms. I averaged the QBER and key rate I calculated during each of the 5 runs and mapped it to a specific value of the varied experimental parameter. I chose to use 5 instances of 10 ms simulations instead of 50 ms of a certain simulation because MATLAB has an upper bound on the memory allocated to an array variable. This limits the number of elements in the time tag file array and eventually limits the simulation time for a single run when some of the parameters like thermal count rate take

extremely high values. I then plot a graph each showing how the QBER and key rate vary when that particular parameter is varied. The exact values of QBER and key rate aren't pivotal in the graphs because these depend on the values of the parameters that are kept constant. However, the form of the curve in the graph is important because it should be preserved even if the fixed parameters take a different value within a reasonable range.

Note 1: The parameters were set to the values mentioned above unless explicitly mentioned otherwise in the following sections

Note 2: One must note that when the key rate is very low, the QBER value is unreliable. This is because the QBER is a statistically aggregate value over the bits in the sifted key. If there are very few bits in the key, the QBER value can vary drastically every time the simulation is run.

Before going into the results of the simulation, table 6.1 shows a table that summarizes how each parameter was modeled as well as which ones were varied during simulation. I have also listed out the nominal values that I used for the purposes of simulation. Please note that some of these values are not realistic for satellite QKD. Changing these values to realistic ones is simple and one needs to only change the values declared in the MATLAB code before the QKD protocol is simulated. Also, in some simulations even though a certain parameter was varied continuously, I did change the value from the nominal one a little bit mostly for the sake of computational complexity or to emphasize interesting points in the resulting graphs.

6.6.1 Thermal Counts

The thermal count rate was varied from 1 Hz to 10 MHz. The QBER and key rate were calculated for each value of thermal count rate. Figure 6.6 shows how the true QBER and sifted key generation rate change as the thermal count rate in the APD detectors is varied. For this simulation, I used a coincidence threshold of 100 ns

At low thermal count rate, the QBER is low. This indicates that the coincidence algorithm does a good job of correctly identifying photon coincidences. This corresponds to a certain key rate being relatively constant in this region of the graph. As the thermal count rate increases beyond 10 kHz, we start to see a rise in the QBER. The increase in the temporal density of thermal counts in the time tag file firstly decreases the number of one-to-one mappings of the actual photon time tags. Also, the increasing thermal counts induce dead time regions in the time tag file which may prevent the detector from avalanche when a photon is incident and hence not record the photon in the first place. Thus, the number

Parameter	Model	Nominal Value	Assumptions
Laser Source	Weak coherent pulsed laser with Poisson photon statistics *poissrnd()* function in MATLAB	Pulse frequency = 5 MHz average photon number per pulse	Photons in a multi-photon event are produced simultaneously
Thermal counts*	Poisson distributed in time. Use *rand()* function to randomly populate time tag file	Thermal count rate = 100 HZ	P.D.F. of next thermal count is an exponentially decaying distribution ($d\,e^{-dt}$)
Afterpulsing probability*	Sample from an exponentially decaying distribution but do so only with probability equal to the afterpulsing probability.	p_{ap} = 10%	1) p_{ap} is the average number of charge carriers getting trapped during an avalanche. 2) pap < 1 3) At most one charge carrier gets trapped during an avalanche 4) A detrapped charge carrier triggers an avalanche subject to dead time and recharge time conditions 5) Avalanches are indistinguishable from each other
Afterpulsing time constant*	Sample from an exponentially decaying distribution of characteristic time constant as the afterpulsing time constant	$\lambda_{ap} = 10^6\ s^{-1}$	1) P.D.F. of detrapping time of charge carrier is an exponentially decaying distribution ($\lambda e^{-\lambda t}$) 2) No time delay between the detrapping and initiation of an afterpulse.
Dead time*	Generate thermal counts, photon clicks and afterpulses first. If two clicks occur less than the dead time value τ_d , delete the second click as well as all generation of afterpulses arising from the second click.	τ_d = 0.5 μs	

Recharge time*	Modeled as a $(1-e^{-t/\tau_R})$ type curve Detection efficiency due to recharge time is $\eta_{recharge} = \frac{1}{1-e^{-\frac{20}{8.5}}}\left(1-e^{-(1-e^{-\frac{\Delta t}{\tau_R}})\frac{20}{8.5}}\right)$	τ_R = 1 µs	Efficiency due to recharge is the only efficiency that varies with time
Optical efficiency		$\eta_{optical}$ = 70%	
Maximum Detector efficiency		η_{DetMax} = 60%	
Detection efficiency of thermal count or afterpulse	$\eta_{thermal\ /afterpulse} = \eta_{recharge} \times \eta_{DetMax}$ For two consecutive detector clicks with the second click being a thermal count or afterpulse, delete the second click with probability $(1 - \eta_{thermal\ /afterpulse})$. Also, delete all successive generations of afterpulses arising from the second click		Varies with time only due to $\eta_{recharge}$
Detection efficiency of single photon state	$\eta_{single\ photon} = \eta_{optical} \times \eta_{recharge} \times \eta_{DetMax}$		Varies with time only due to $\eta_{recharge}$
Detection efficiency of *k* -photon state	$\eta_{k\text{-}photon} = 1 - (1 - \eta_{single\ photon})^k$ For two consecutive detector clicks with the second click being due to detection of a *k* -photon state, delete the second click with probability $(1 - \eta_{k\text{-}photon})$. Also, delete all successive generations of afterpulses arising from the second click		1) Varies with time only due to $\eta_{recharge}$ 2) Detector is not photon resolving

Timing jitter of laser and detector*	Modeled as Gaussian. Time tag is sampled from the Gaussian distribution centered around the expected time instant of a potential photon emission/detection event	300 ps	1) Laser side timing jitter consists of the laser jitter and jitter associated with laser side time tagger. 2) Detector side timing jitter consists of the detector's jitter and jitter associated with detector side time tagger. 3) Laser side timing jitter = detector side timing jitter
Optical path time		0.3 μs	Time taken for photon to travel from Alice's to Bob's laboratory is fixed and known to both Alice and Bob
Production and measurement of quantum states	Used *rand()* in MATLAB		The measurement is done using a polarization dependent beam splitter
Choice of basis set	Used *rand()* in MATLAB		The choice of basis set is done using a 50-50 non-polarizing beam splitter
Coincidence window	A time window of fixed width t_{coinc}. Coincidence matching is successful when a count in Alice's file is within a time interval of tcoinc from a count in Bob's file. A one-to-one mapping is required to successful matching.	t_{coinc} = 10 ns	

Intercept and Resend attack (p_{int}*)	Eve intercepts a k-photon with probability p_{int}, measures it and sends the measured state to Bob	p_{int} = 0%	1) Eve has 100% detection efficiency on her detector 2) Eve induces zero time delay in the system, when photon travels from Alice to Bob. 3) Eve can detect the number of photons in Alice's multi-photon state accurately and generate a multi-photon state with the same number of photons. 4) Eve has an ideal photon source which can generate photons in the same polarization state as she measured Alice's photon to be in

* indicates that this parameter was varied at some point to find out how the QBER and sifted key rate

Table 6.1: Table listing out each modeled parameters in the quantum key distribution simulation. The second column lists how each parameter was modeled in the simulation. The third column lists out the nominal value used for that parameter. This is the default value that a parameter assumes when it is not being varied. The fourth column states assumptions relating the modeled parameter. A $*$ symbol against a parameter in the first column denotes that it was varied at some point in the simulation to correlate how the QBER and sifted key rate changed with respect to the parameter. During this process the other parameters were held fixed.

of bits in the sifted key corresponding to actual photons decreases. Also, at low thermal count rate it is improbable that thermal counts are classified as photon coincidences by the coincidence algorithm.But as the temporal density of thermal counts increases in the time tag file, it is plausible that some of these thermal counts will be classified as photon coincidences. These thermal counts will eventually seep into the sifted key since they are not correlated with Alice and Bob's choice of preparation / measurement basis sets, and on average half of these thermal counts in the sifted key will contribute to mismatched bits - this is because in a 2-bit system where each bit is chosen randomly, the probability of mismatch is 0.5 . The combination of decreased number of photons and the increased thermal counts in the sifted key increases the QBER. Also, in this region the key rate increases. If you notice the key rate doesn't increases exactly when the QBER increases. This is because as the QBER starts to increases, the decrease in the number of actual photons in the sifted key is nearly the same as the increase in the thermal counts in the key. Thus the key rate still remains the same. However, with further increase in thermal count rate, the increase in the thermal counts seeping into the key dominates the decrease in the photons in the key. This causes the key rate to increase. Hence the increase in key rate lags the increase in QBER as the thermal count rate is increased. In my simulation, I kept track of which bits in the final key were due to signal photons, thermal counts and afterpulses. This helped me explain the variations in QBER and key rate.

I was unable to simulate for thermal counts beyond 10 MHz because of the limitation of size of arrays in MATLAB . As the thermal count rate exceeds the dead time, many of the thermal counts won't actually manifest because the average time period of thermal counts will be of the order of magnitude of the dead time (during which the detector cannot avalanche). Hence, one predicts the rate of increase of QBER and key rate to drop around this region. The dead time used was 0.5 µs. This corresponds to a frequency of 2 MHz. Looking at the graph, it does seem that the rate of increase of QBER and key rate starts to drop around 2 MHz. With further increase in thermal count rate variable, the observed thermal count rate is expected to asymptote due to the time tag file being filled with dead time regions. Consequently, the key rate should asymptote as well. My guess is that the QBER will drop to zero because the photon detection will be extremely difficult due to the time tag file being filled with dead time regions. Nonetheless, one must stay away from this region during the QKD protocol in practice. However, as the thermal count rate exceeds the coincidence threshold of 100 ns (corresponding to a thermal count rate of 10 MHz), one would see a drastic drop in the key rate because the coincidence algorithm will find it more difficult to achieve one-to-one mappings for photon coincidences.

Note: I want to distinguish between *thermal count rate* and *observed thermal count rate*.

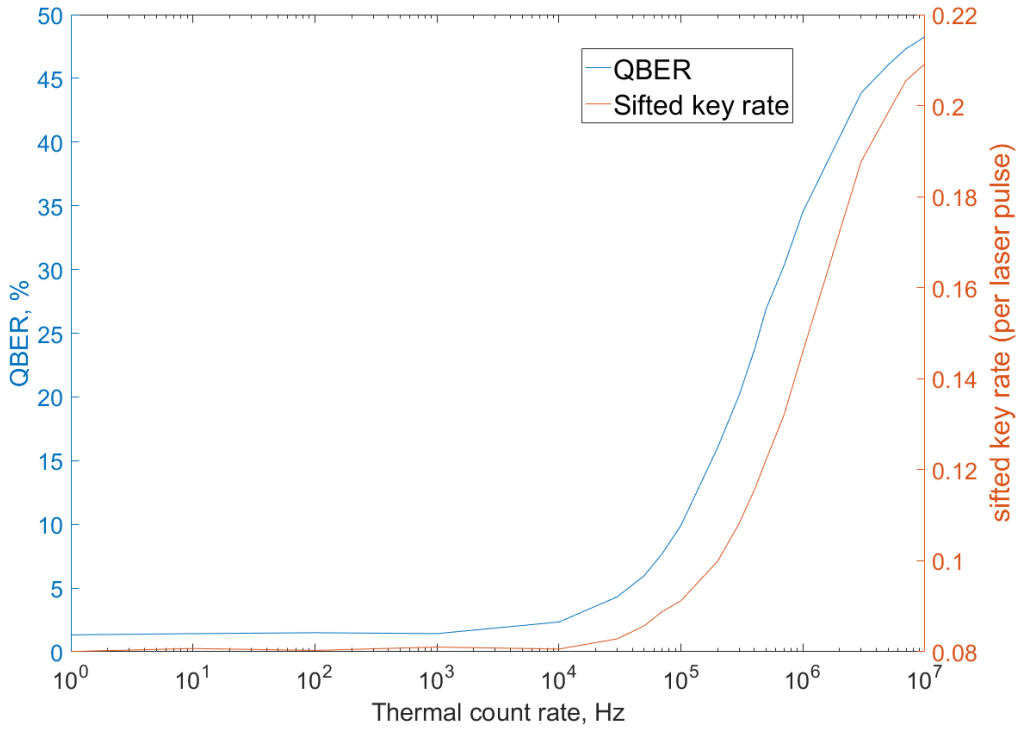


Figure 6.6: This figure shows how the QBER and sifted key generation rate vary as the thermal count rate in the APD detectors changes.

Thermal count rate is the rate at which thermal clicks are supposed to occur under the assumption of zero dead time. This corresponds to the rate at which the initial mobile charge carriers that will initiate the avalanche associated with thermal counts are produced. However, in the presence of dead time, the increase in the thermal counts causes the time tag file to be cluttered with dead time regions which increasingly prevent some of the thermal counts from manifesting as avalanches in the detector. The observed thermal count rate is defined as the rate at which avalanches which are triggered by the aforementioned initial mobile charge carriers occur. Hence, observed thermal count rate $\leq$ thermal count rate. The time tags in the file correspond to the observed thermal count rate. The graph in figure 6.6 shows the thermal count rate on the x-axis and not its observed counterpart. There should be a nice relation between thermal count rate and the observed thermal count rate and this might be a case for future work.

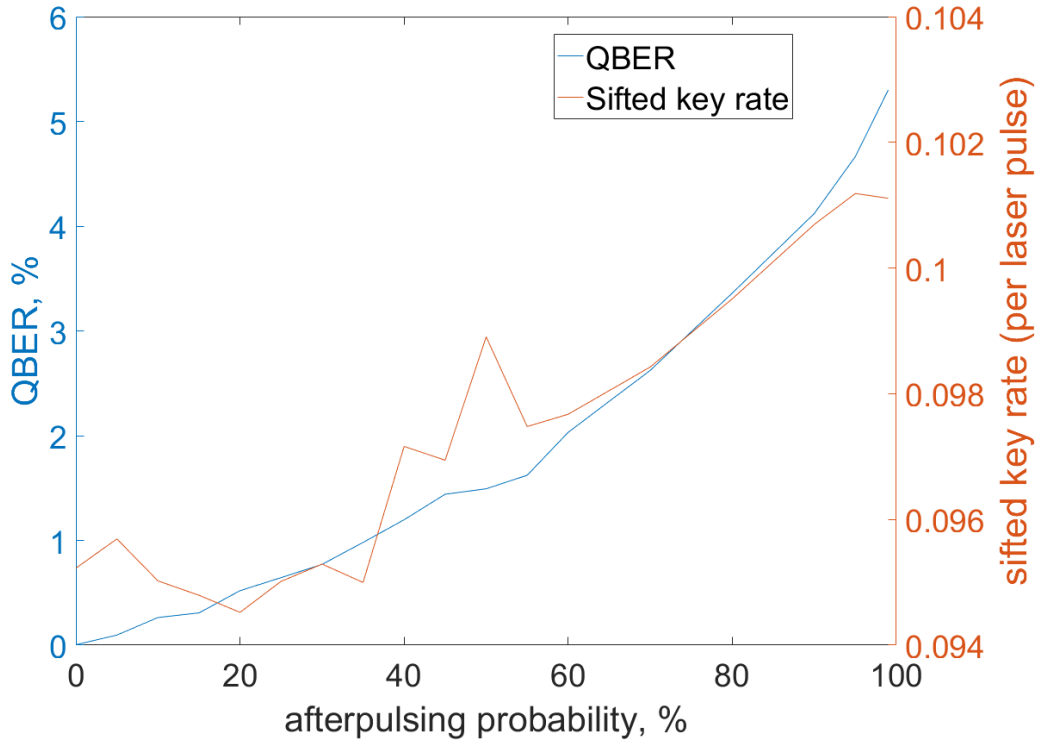


Figure 6.7: This figure shows how the QBER and sifted key generation rate vary as the afterpulsing probability in Bob's APD detector changes.

6.6.2 Afterpulsing Probability

For this simulation, I varied the afterpulsing probability, p_{ap} from 0 - 99%. I didn't simulate $p_{ap} \geq 100\%$ because this would mean every avalanche would on average give rise to at least one afterpulse. This would lead to a runaway effect with the count rate approaching ∞ far to the right of the time the file - actually due to non-zero dead time, the detector click rate would asymptote to a finite value under the assumption that every avalanche is time tagged regardless of its pulse amplitude.

Figure 6.7 shows how the QBER and sifted key generation rate vary when the afterpulsing probability in Bob's detector changes. A larger afterpulsing probability means that the temporal density of afterpulses in the time tag file is greater. As the afterpulses start to clutter Bob's time tag file, the chance of a one-to-one mapping by the coincidence algorithm to determine photon coincidences decreases. Thus, the number of bits in the sifted key corresponding to actual photons decreases. Simultaneously, it is plausible that the chances that an afterpulse is deemed as a photon coincidence increases . Since, the occurrence of an afterpulse time tag has no correlation with Alice and Bob's preparation

/ measurement basis sets, on average half of these afterpulse time tags that are deemed as photon coincidences seep into the sifted key. Also, since such afterpulse time tags could occur in any of Bob's detectors, about half of such time tags in the sifted key will result in a bit mismatch between Alice and Bob's copies of the key. Due to both the decrease of photons and the increase of afterpulses in the sifted key, the QBER increases when the afterpulsing probability is increased. The increase in QBER is greater than linear with respect to afterpulsing probability. This is likely because the QBER responds to the number of afterpulses in the time tag file and the number of afterpulses increases greater than linearly with respect to the afterpulsing probability.

The key generation rate increases with afterpulsing probability. This may be because the increase in the number of afterpulses seeping into the sifted key may outweigh the decrease in the number of photons in the sifted key, thus increasing key rate. I haven't simulated beyond 99% afterpulsing probability primarily due to space complexity but would like to make some predictions. As the afterpulsing probability approaches 100%, the sifted key rate would drop because the coincidence algorithm would find it difficult to achieve one-to-one mappings.

6.6.3 Afterpulsing Time Constant

For this simulation, I varied the afterpulsing time constant λ_{ap} from $1\,\mathrm{s}^{-1}$ to $1 \times 10^{12}\,\mathrm{s}^{-1}$. Figure 6.8 shows how the QBER and sifted key rate change with afterpulsing time constant. The average time to de-trap a trapped electron can be shown to be equal to $\frac{1}{\lambda_{ap}}$. The p.d.f. of detrapping time with a smaller λ_{ap} is more spread out meaning that afterpulses generated by this p.d.f. tend to be more spaced out. At first, this might seem like a smaller λ_{ap} is associated with lesser temporal density of afterpulses in the time tag file. However, one must note that in an infinitely long file, the smaller λ_{ap} implies that at any point in time you have afterpulsing contributions from time tags much further towards the left than if you had a larger λ_{ap}. This should imply that the smaller λ_{ap} itself does not decrease the temporal density of afterpulses and hence cannot affect the QBER. This effect is seen in the intermediate region between $\lambda_{ap} = 1 \times 10^2\,\mathrm{s}^{-1}$ to $1 \times 10^6\,\mathrm{s}^{-1}$. When $\lambda_{ap} < 1 \times 10^2\,\mathrm{s}^{-1}$, the average detrapping time becomes greater that 10 ms, which is the size of the time tag file during the simulation. Thus, the assumption of infinite sized file no longer applies. In this case most of the afterpulses occur beyond the size of the file and I didn't explicitly delete these afterpulses. Beyond the 10 ms mark, the temporal density of afterpulses drops as you go towards the right in the time tag file. This improves the accuracy of the coincidence algorithm as some of the afterpulses are now not being deemed as photon coincidence.

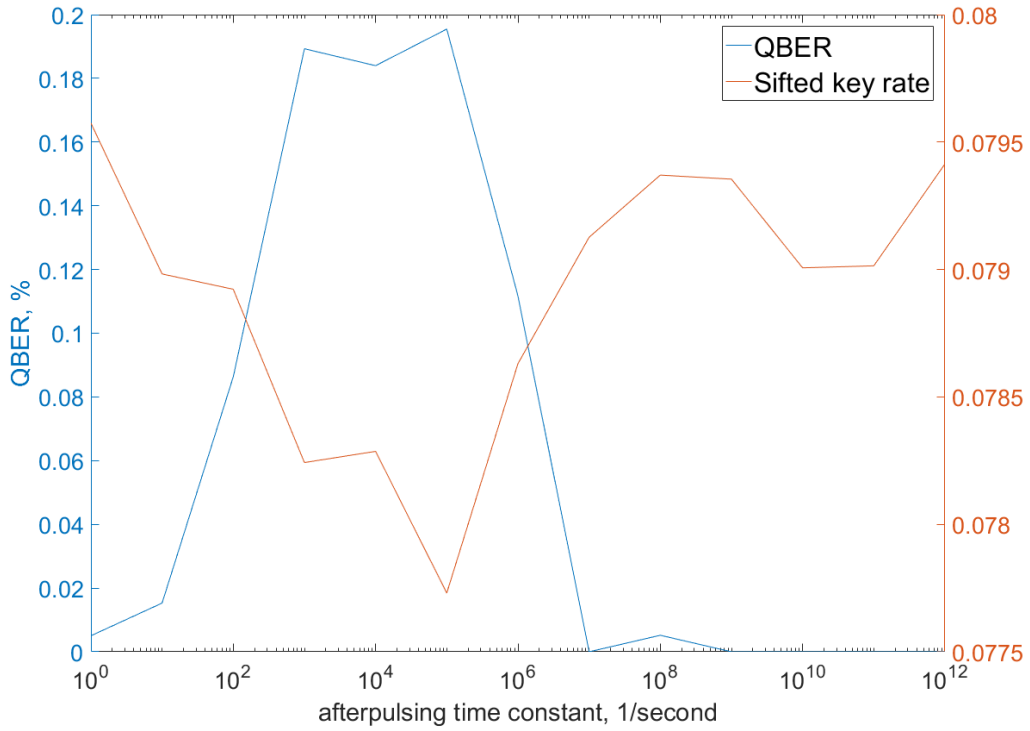


Figure 6.8: This figure shows how the QBER and sifted key generation rate vary as the afterpulsing time constant in Bob's APD detector changes.

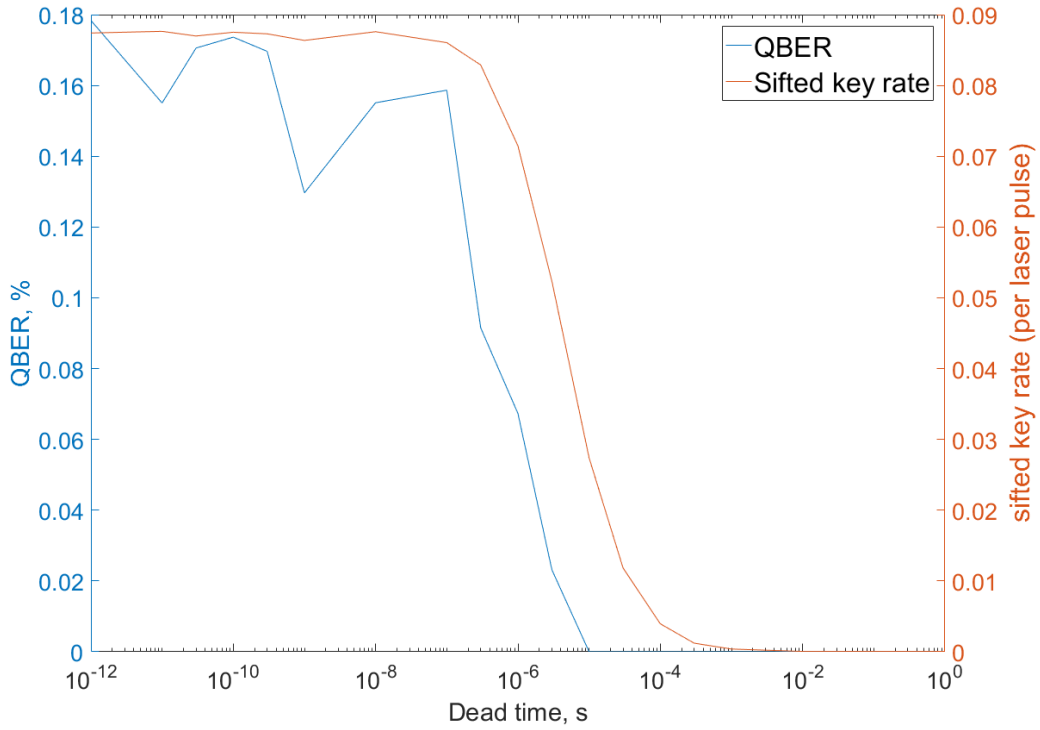


Figure 6.9: This figure shows how the QBER and sifted key generation rate vary as the dead time in Bob's APD detector changes.

Thus the QBER drops for small λ_{ap} values. When the λ_{ap} value increases beyond the dead time frequency equivalent of 2 MHz (corresponding to 0.5 µs), the trapped charge carrier is now detrapped when the detector is still in a dead time region induced by the avalanche that trapped the charge carrier in the first place. Thus the afterpulses increasingly cease to occur and cannot seep into the sifted key by mistake. Hence the QBER drops for large λ_{ap} values. The key rate doesn't itself change much with λ_{ap}.

6.6.4 Dead Time

I varied the dead time of Bob's APD detector from 1 ps to 1 s and simulated how the QBER and sifted key rate change. In figure 6.9. At low dead time values, both the key rate and QBER are largely uncorrelated with dead time. This means that the dead time isn't impeding the photon detection. As the dead time increases, it makes the detector increasingly unavailable for detection because the time tag file becomes cluttered with dead time regions. This decreases the key rate. In particular, when the dead time exceeds the time period of the laser trigger (200 ns), the key rate starts to drop. This is because any

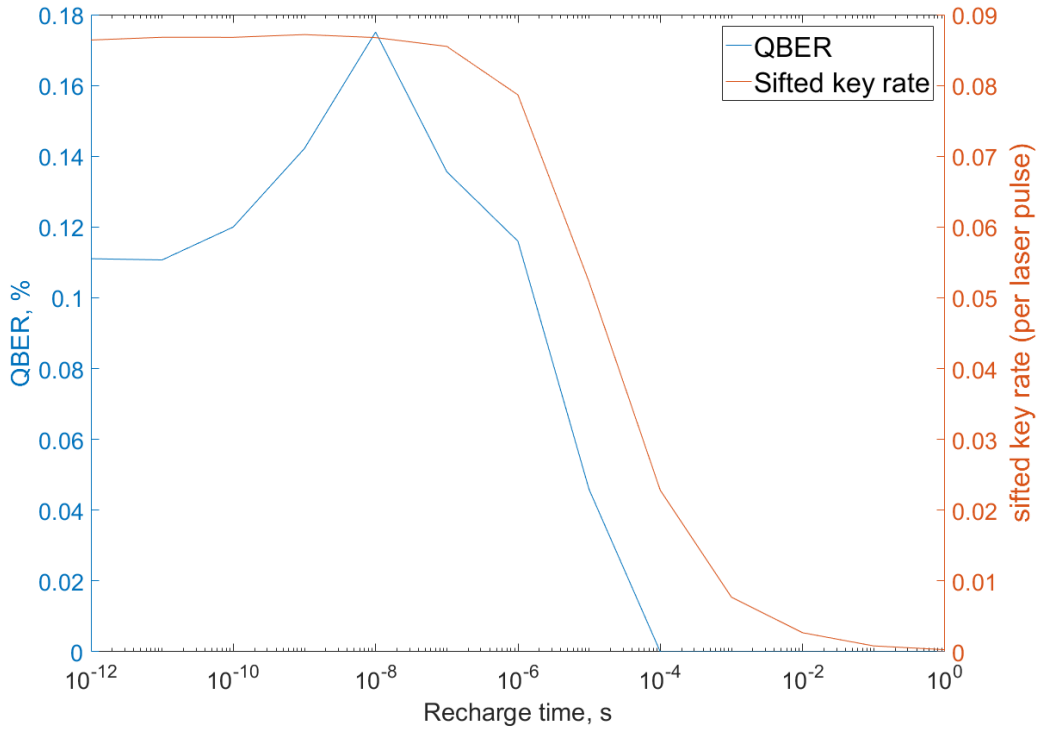


Figure 6.10: This figure shows how the QBER and sifted key generation rate vary as the recharge time in Bob's APD detector changes.

type of detector click (thermal, photon, or afterpulse) in Bob's APD induces a dead time period that is long enough that when the next photon arrives at Bob's laboratory, his APD is inactive and unable to detect the photon. Thus, one expects few photon coincidences during the 10 ms simulation and this decreases the key rate. It does seem that as the key rate drops, so does the QBER.

6.6.5 Recharge Time

I varied the recharge time from 1 ps to 1 s and simulated how the QBER and sifted key rate change. At low recharge time, the detector is at maximum efficiency when most "potential clicks" are supposed to occur. This keeps the sifted key rate more or less constant. However, when the recharge time exceeds the source clock's time period of 200 ns, the key rate starts to drop. This is because the recharge time τ_R decides the time scale for the recharge efficiency $\eta_{reacharge}$ to reach unity. As the recharge time exceeds the source's time period, the APD has a low detection probability due to which an increasing number of photons will go undetected. Hence, fewer photons end up in the key, leading to lower key rates.

Also, the afterpulses tend to occur on a time scale of 1 µs. This also means that when the recharge timeexceeds 1 µs, fewer detrapped charge carriers produce and avlanche (fewer afterpulses) which leads to smaller key rates.

At low recharge time values, the QBER is uncorrelated with recharge time because the detector detects most "potential clicks". As the recharge time exceeds the source clock time period, the QBER starts to increase. This likely because the photon detection is adversely affected as explained earlier. However, since the thermal counts occur on a larger time scale of 10 ms, they are less affected when the recharge time is around the source clock's time period. Thus as the recharge time increases, the decrease in the number of photons in the final key is greater than the corresponding decrease for thermal counts. Although the afterpulses in the sifted key also drop around 1 µs recharge time (which should decre), I have used a small afteprulsing probability (10%). Thus, the drop

6.6.6 Timing Jitter

For my simulation, Alice's jitter (due to her laser electronics and time tagger) and Bob's jitter (due to his detector and time tagger) are assumed to be equal to each other. The code can however be easily modified to make these jitters different. Please note that Bob's jitter is only the jitter imparted due to his experimental setup. The net jitter in his time tags will as contain Alice's jitter as explained in section 6.4.5. I then varied this value of jitter from 1 fs to 1 s and simulated how the QBER and sifted key rates changed. Figure 6.11 shows how the QBER and key rate change as the jitter is varied.

At low jitter, the QBER is very low. This is because the jitter is smaller than the coincidence threshold and hence the coincidence window can successfully catch true photon coincidences. However, when the jitter exceeds the coincidence threshold of 10 ns, we see that the QBER increases. This is because there occurs an increasingly larger number of instances where many true photon coincidences are not matched by the coincidence algorithm because the corresponding photon time tags in Alice's and Bob's files are separated by a value greater than the coincidence threshold (after accounting for the optical path time). This decreases the number of photons in the final sifted key. One must note that the increase in QBER occurs gradually around the coincidence threshold because the Gaussian distribution is a smooth function. Also, I didn't simulate any jitter for the thermal counts and the afterpulses. Hence, the number of thermal counts and afterpulses in the final key shouldn't be affected by changing the value of jitter. Thus, the QBER increases.

The key rate starts decreasing around the coincidence threshold because although the thermal counts and afterpulse in key remain more or less constant in the final sifted key,

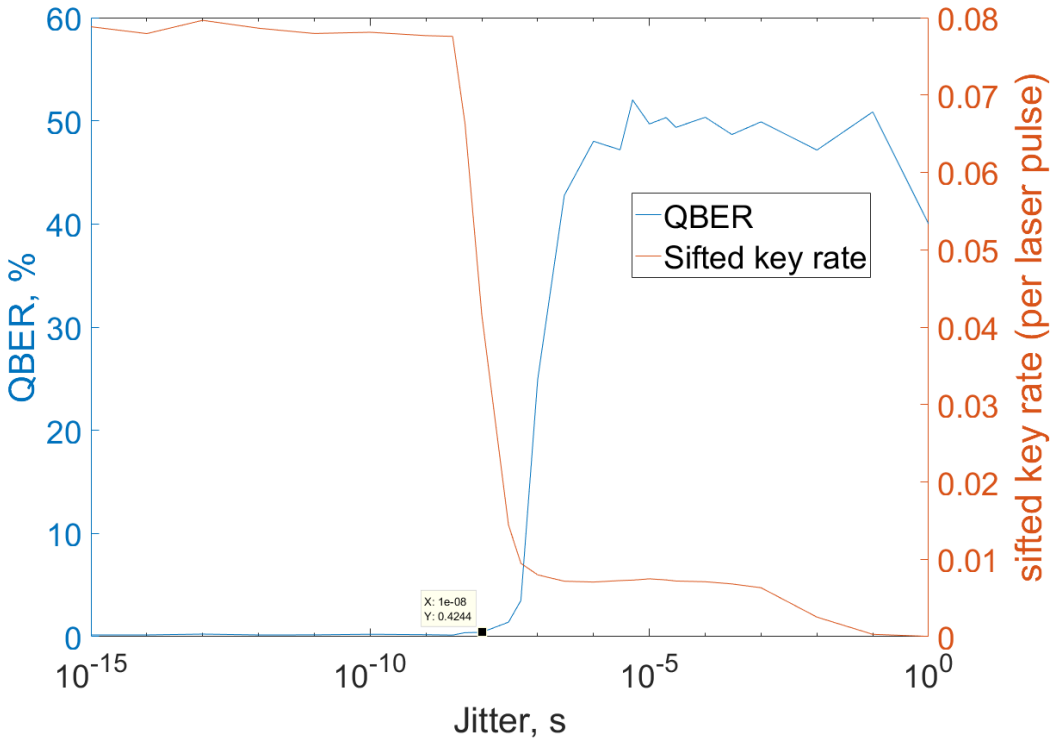


Figure 6.11: This figure shows how the QBER and sifted key generation rate vary as the jitter in in Alice's and Bob's equipment changes. Here Alice's jitter comprises of her laser electronics and time tagger whereas Bob's jitter comprises of his

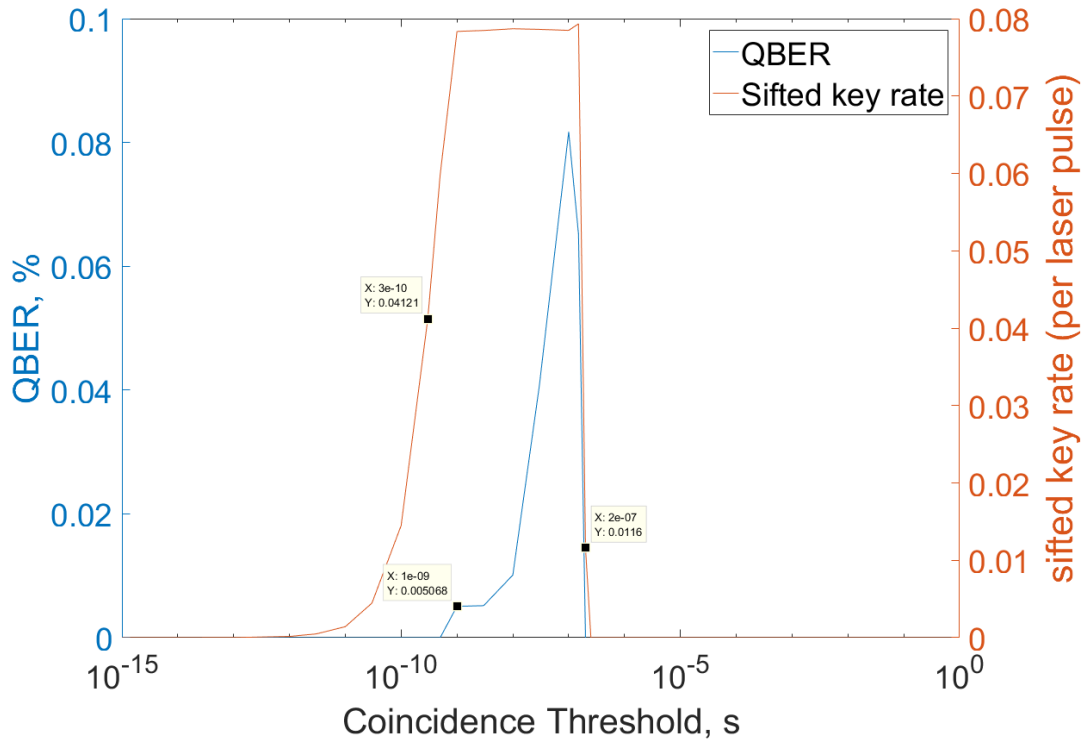


Figure 6.12: This figure shows how the QBER and sifted key generation rate vary as the coincidence threshold (width of the coincidence window) in the coincidence algorithm changes.

the photons in the key start decreasing. This decreases the overall size of the key for a fixed simulation time, hence decreasing the key rate.

A special note must be made that the calculated values of QBER at jitter $\geq 0.1\,\mathrm{s}$ are unreliable. This is because since the key rate is extremely small at these jitter values, it is difficult to get a statistically accurate value of QBER for a simulation run time of 10 ms. One would need to increase the simulation run time to get a reliable value of QBER.

6.6.7 Coincidence Threshold

I varied the width of the coincidence window, i.e., coincidence threshold, in the coincidence algorithm from 1 fs to 1 s and simulated how the QBER and sifted key rates changed. Figure 6.12 shows the results.

At low coincidence threshold, the key rate is nearly zero. This is because the coincidence threshold is significantly smaller than the jitter of 300 ps. Therefore, the time tags in

Alice's and Bob's files corresponding to photon production and detection, respectively, are too spread out compared to the coincidence threshold and hence the coincidence algorithm cannot find photon coincidences. As the coincidence threshold surpasses the jitter, we notice an increase in the key rate. The increasing coincidence threshold starts detecting more and more photon coincidences, increasing the key rate. One must note that this increase in key rate is gradual as the coincidence threshold exceeds the jitter because the jitter is modeled by a Gaussian function which is a smooth function.

As the coincidence threshold exceeded the time period of the laser (0.2 µs), we see a drop in the key rate. This is because, the coincidence window is so wide it can now pick up photons produced during consecutive laser triggers. Thus, a one-to-one mapping is more difficult to achieve in the coincidence algorithm, leading to fewer photons ending up in the final key, decreasing the key rate eventually to zero.

The QBER is nearly zero for the most part because thermal counts and afterpulses are not deemed by the coincidence threshold as photon coincidences. However, the QBER shoots up a tiny bit when the coincidence threshold exceeds 10 ns because a few thermal counts in Bob's file become deemed as photon coincidences, increasing the QBER slightly.

Thus, the coincidence has a narrow region in which it is effective. The exact value of this region depends on experimental parameters, in particular, the jitter and the source frequency.

6.6.8 Attack

An intercept and resend attack by an eavesdropper, Eve, was simulated. The probability of interception, p_{int}, of a k-photon state was varied from 0 - 100% and the corresponding QBER and sifted key rate was calculated from the simulation of the QKD protocol for each value of p_{int}. Figure 6.13 shows the results.

The QBER increases linearly with the probability p_{int} that Eve intercepts Alice's k-photon state. The QBER reaches its maximum value of 25% when $p_{int} = 100\%$. This is the expected theoretical prediction as explained in section 6.5.4. Furthermore, it is simple to see why the curve is linear. Let's assume the coincidence algorithm does a perfect job at matching true photon coincidences. Then, if you only look at the photons that Eve intercepts, all these intercepted (and resent) photons will end up in the raw key. But by the explanation in section 6.5.4, a quarter (25%) of the bits corresponding to these intercepted photons in Alice's and Bob's sifted keys will be mismatches. If Eve intercepts $p_{int}\%$ of Alice's k-photon states (k > 0), then the number of mismatched bits in the sifted keys is equal to

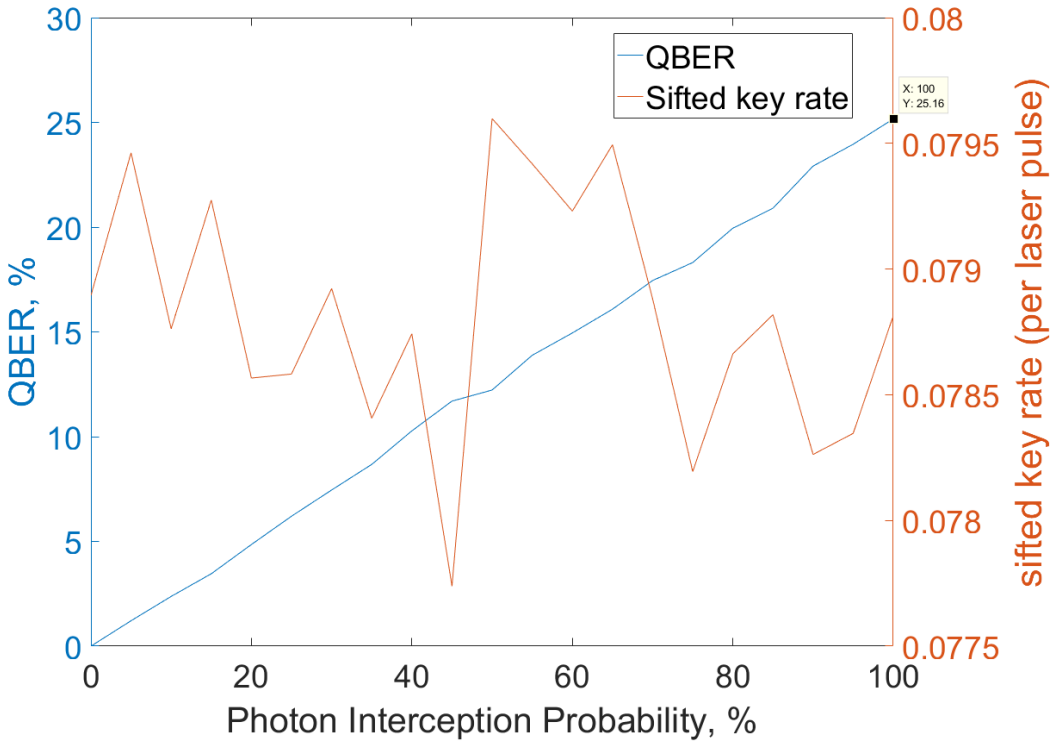


Figure 6.13: This figure shows how the QBER and sifted key generation rate change as one varies the interception probability of Alice's k-photon state in an intercept and resend attack performed by an eavesdropper.

$25\% \times p_{int} \times$ number of k-photons states sent $= 0.25\ p_{int} \times$ number of k-photons states sent. This is under the assumption that unintercepted photons produce no bit mismatches in the sifted key. Since the QBER is the number of bit mismatches per bit in the sifted keys, QBER $= 0.25\ p_{int}$, which is the equation of the QBER curve in figure 6.5.4. The greater the value of p_{int}, the more information that Alice has about the sifted key. Alice and Bob can thus put an upper bound on their estimated QBER to limit Eve's information of their keys. If the estimated QBER exceeds this threshold, Alice and Bob discard their keys and restart the protocol.

The key rate is uncorrelated to p_{int}. This is because Eve does not induce any delay in Bob's detection of Alice's k-photon state. So, it doesn't affect the coincidence algorithm and hence doesn't affect generated key size. Also, Bob's choice of measurement basis set is independent of Eve's photon interception probability. Since the sifted key is determined by those instances in which Alice's and Bob's choice of basis sets match, the intercepted photons still end up in the sifted key, thus the key size doesn't change. However, these intercepted photons in the sifted key contribute towards bit mismatches. Thus, Alice and Bob cannot detect Eve's presence by the generated key rate but only through the key's estimated QBER.

6.7 Summary and Outlook

A QKD quantum link was simulated and the dependence of sifted key generation rate and QBER on different system parameters was found. The parameters of the laser source were fixed. However, the detector parameters were varied and for each value of the detector parameter, the simulation was run for 10 ms five separate times and the sifted key generation rate and QBER values were averaged out.

Here's a brief summary of the results. As the thermal count rate is increased the QBER and sifted key rate are initially constant but both increase after a certain point (the exact point value of thermal count rate at which they increase is not extremely important to mention her because this depends on other system parameters as well). The thermal counts start being wrongly ascribed by the coincidence algorithm as being photon coincidences, causing them to seep into the sifted key, increasing the key generation rate on one hand and increasing QBER. Similarly, both the key generation rate and QBER increase with afterpulsing probability for the same reason that more afterpulses start being wrongly deemed as photon coincidences. The value of afterpulsing time constant has no effect on the key generation rate as the temporal density of afterpulses doesn't change. Also,

the QBER is not significantly affected with a maximum variation of 0.2%. At low dead times, the QBER and key rate are unaffected by changes in dead time. When the the dead time exceeds the time period of the laser trigger, the key rate drop because with each detector click inducing a dead time region as large as the time period of the laser, there is a chance that a photon arrives in this time dead time interval is huge. The QBER also simultaneously decreases.

As the recharge time is varied and increases beyond the laser clock's time period, the photon detection probability starts to drop. This decreases key rate and increases QBER because the thermal counts which occur on larger time scale sin this simulation are less affected by recharge time and their presence in the final key in unchanged till the recharge time increases to the average time period of the thermal counts. The timing jitter and the coincidence threshold are closely linked. When the jitter is larger than the coincidence threshold, the number of true photon coincidences drops, increasing the QBER and decreasing the key rate. The coincidence threshold has an additional feature that increasing its value beyond the time period of the laser clock decreases the chance one-to-one mapping of the of the true photon detections, hence decreasing key rate drastically to nearly zero. Finally an intercept and resend attack was simulated. The key rate was uncorrelated to the probability of photon interception whereas the QBER linearly increased from 0% to the theoretical maximum of 25% as the the probability of photon interception increased from 0 - 100%.

I have modeled the jitter as a Gaussian distribution function centered at zero over the real line. However, one must be careful here because while sampling the Gaussian that is used to model Bob's jitter, there is a finite probability that one obtains a value of Bob's time tag far to the left on the time axis, even before the experiment started. Although this probability is small, it is still impossible that in a physical experiment Bob's time tag of a certain photon is earlier than Alice's time tag of the same photon, the only possibility when this may occur is if the optical path time is negligible compared to time tagger's resolution - this not representative of long distance communication. Though the number of such instances are small when realistic experimental parameters are used, they may became significant at large jitter values. It would be nice to explore the possibility of putting a lower bound on the range over which Bob's jitter Gaussian is defined. An upper bound on the range of Alice's Gaussian might also be needed.

In this simulation, the time tagger jitter was modeled as a Gaussian. Instead one can explore modeling the jitter of the time tagger as a uniform distribution (a step function which takes value 1 over a range whose width is equal to the resolution of the time tagger).

I would also like to propose a new model for afterpulsing. The afterpulsing model

in this simulation assumes that every time an avalanche occurs, either no electron gets trapped in trap site in the detector's substrate or exactly one electron gets trapped in a trap site. The former event happens with probability $1 - p_{ap}$ and the latter occurs with probability p_{ap}, where p_{ap} is the afterpulsing probability of the APD. However, this may not exactly be true. If one assumes that there are M electron produced during an avalanche and there are N trap sites in the substrate, then each of these trap sites has a non-zero probability of trapping an electron. IT is completely possible that more than one of these trap sites trap an electron each. To make this idea more concrete, let's assume that the M electrons that are produced during an avalanche are uniformly (spatially) distributed in the volume of the detector's substrate. Let's also assume that the N trap sites in the substrate are far apart from each other and $M >> N$. This ensures that if an electron does get trapped in a particular trap site, the trapping event doesn't affect the probability with which a neighboring trap site might trap an electron. This means that each trap site has the same probability of trapping an electron (let's call this probability p_0) and the event that a particular trap site traps an electron is independent of whether any other trap site does a electron trapping. In such a case, one expects that during an given avalanche, anywhere between 0 and N electrons can get trapped. Let the number of electron trapped during a given avalanche be k. The probability that one gets k trapped electrons during a given avalanche is given by the binomial distribution BN, p_0. To see this, let's try to estimate the probability of getting k trapped electrons in a given avalanche. First label and order the trap sites. The probability of the first k trap sites each trapping an electron is p_0^k because the trapping events are independent of each other, hence the probabilities multiply. The probability of the last $N - k$ trap sites do not trap an electron is given by $(1 - p_0)^{N-k}$. Thus, the overall probability that the first k sites trap an electron is given by $p_0^k\ (1 - p_0)^{N-k}$. However, corresponding to any unordered combination of k out of the N sites each trapping an electron, one can go through a similar argument to show that the probability of such an event is also $p_0^k\ (1 - p_0^{N-k}$. Since there are $\binom{N}{k}$ ways of selecting unordered combinations of k out of N sites and each of the corresponding events are mutually exclusive, the total probability that exactly k sites each trap an electron is given by $P(k)$ as follows

$$P(k) = \binom{N}{k}\ p_0^k\ (1 - p_0)^{N-k} \tag{6.19}$$

This is nothing but the probability mass function of the binomial distribution. Thus in simulation, one can fix values for N and p_0. When a detector click occurs, one can sample from the binomial distribution to decide on how many electrons get trapped during the avalanche corresponding to this detector click. If k electrons get trapped, then that

detector click will produce k afterpulses. To simulate the detrapping of each of these trapped electrons, the exponentially decaying distribution given by equation 6.6 must be sampled in k separate instances giving rise to k different detector clicks.

There is a caveat here. There is a constraint of the values of N and p_0. One must note that in operation, on average, one detector click cannot give rise to more than detector clicks; otherwise there will be a runaway effect with detector click rate approaching ∞ far to the right in the time tag file (in the absence of dead time). This renders the detector nonoperational. Thus, although in particular instances of avalanches, one can have more than one afterpulse according to the binomial mass function, on average, the number of afterpulses must be less than one. The expected value $\mathbb{E}(k)$ of a binomial distribution is given by

$$\mathbb{E}(k) = N\ p_0 \tag{6.20}$$

where k is the number of sites that each trap an electron during a given avalanche.

Thus the constraint must be that $\mathbb{E}(k) < 1$, i.e,

$$N\ p_0 < 1 \tag{6.21}$$

Another aspect for improvement would be to incorporate jitter due to the time tagger. First, on Bob's side when time tagging incident signal photons, thermal counts and afterpulses. Second, on Alice side when time tagging the laser reference pulse.

Eve has other attack strategies than the intercept-resend attack mentioned in section 6.5.4. One method is called the photon number splitting attack. In this, Eve detects a multi-photon states sent by Alice. Eve keeps one photon for herself and the remaining photons on their way to Bob. Since all photons in a multi-photon state have the same polarization, Eve has an exact copy of the quantum state that Bob receives without having to clone quantum states - which is prohibited by the no cloning theorem [40]. She does this for all multi-photon states that Alice sends. Eve does not measure the photons till Alice and Bob publicly reveal their measurement bases. Since Eve knows that during the sifting process of QKD, Alice and Bob will only retain the bits where their measurement bases agreed, she can be assured that measuring the corresponding photons will give her exact information about those bits in the key. Moreover, if Eve blocks all single photon states from reaching Bob, Eve will know the vast majority of the sifted key. This kind of attack can easily be implemented in simulation. In Eve's code, one would need to declare an array which stores states $|H\rangle$, $|V\rangle$,$|D\rangle$ or $|A\rangle$, when Alice sends a k-photon state with

$k > 1$. However, when Eve sends on the remaining photons, the code will have to modify the k-photon state to a $k-1$-photon state. As such Alice and Bob's code will not have to be modified for this attack.

However, practically Alice and Bob can defend themselves from a photon number splitting attack by the use of dummy quantum states called decoy states [37, 41]. In section 6.3, it was explained that the photon number statistics across the pulses of a WCP laser (also called the signal source) was modeled as a Poisson distributed with some average photon number μ. Alice can use another source with the same spectral properties (wavelength, etc.) as that of the signal source. But this source can have Poisson photon statistics with a different average photon number. This source is called the decoy source. Alice randomly switches between source and decoy states and transmit photons to Bob. Due to different average photon number, signal and decoy states have different ratios of single photon states to multi-photon states. If Eve starts blocking out single photon states selectively, Bob will experience different losses in the signal and decoy pulses. When Alice makes her choice of signal/decoy sequence public, Bob will be able to calculate detect these statistically different losses between signal and decoy, and hence detect Eve's presence. In the simulation, the modification will have to be done to the post-processing section in the QKD protocol after the generation of sifted key. Once the details of the calculation of channel loss with respect to signal and decoy are worked out, it should be straightforward to implement it in code.

Other parameters can also be modeled into the simulation. For example, I haven't yet considered the effect that over-voltage has on thermal count rate, detection efficiency and afterpulsing rates. In our lab, we use an over voltage of 20 V. It would be nice to determine the optimum voltage in order to minimize QBER or maximize the sifter key rate. Figure 3 in ref [19] shows that the dark count rate increases with over voltage. Here, dark count rate includes both the thermal count rate and their associated afterpulses. In my simulation, I have modeled thermal counts and the afterpulsing characteristics separately. It would be preferable to first find out the explicit dependence of thermal count rate on over voltage and of afterpulsing probability on over voltage. Qualitatively, the thermal count rate increases with over-voltage due to an increased likelihood generation of initial charge carrier at generation centers but also due to the increased chance of that charge carrier triggering an avalanche. On the other hand, the afterpulsing time constant of a deep level does not depend on over-voltage. But the afterpulsing probability will increase with over-voltage because there will be more charge carriers in the avalanche which would increase the expected number of charge carriers getting trapped during an avalanche, which is exactly the afterpulsing probability. Then instead of specifying a value for the afterpulsing

probability or thermal count rate in the code, one specifies the over-voltage instead. This is also realistic because experimentally one can alter the over voltage directly but not the thermal count rate or afterpulsing characteristics. Also, figure 1 of ref [19] shows that the photon detection efficiency increases with over voltage. When the photon is incident on the APD substrate, it generate a single charge carrier. Although this process is not affected by over-voltage, a higher over-voltage increases the chances that this initial charge carrier will trigger an avalanche, hence overall increasing the photon detection efficiency. Now, an increased over-voltage increases dark count rate which increases QBER; simultaneously it increases photon detection efficiency which should likely decrease QBER (because you have more signal photons which would increase the signal to noise ratio in the sifted key). So, there could possibly be a trade-off here and one should be able to use the simulation to find a value of over-voltage that minimizes the QBER.

Also, this simulation uses a continuous mode of detector operation. Gated operation [42] can also be modeled in the future.

For simulating afterpulsing, I have assumed a single exponentially decaying p.d.f. for the detrapping of a charge carrier. In reality, there could be multiple types of deep level traps, each one with a different average detrapping life time. One could model this in two parts. Firstly, each type of deep level could have different spatial density of traps in the APD substrate. The spatial densities of each type of deep level will determine the probability with which charge carriers get trapped in a given type of deep level. Assume again just one charge carrier gets trapped during an avalanche. For every avalanche, one would have to use the $rand()$ function to determine if the charge carrier would get trapped or not according to the overall afterpulsing probability and then decide which kind of deep level trap the charge carrier would get trapped in according to the relative spatial densities of the different types of deep levels. This simulates the afterpulsing probability. The second part of the code would have to sample the exponentially decaying distribution with the average detrapping lifetime for the particular type of deep level that the charge carrier got trapped in the first part of the code. This simulates the afterpulsing time constant. The next step would be to allow for multiple charge carriers to get trapped in deep levels in any given avalanche. The total number of charge carriers getting trapped in a given avalanche would obey the binomial distribution as previously stated. Then the above procedure of afterpulsing simulation with different types of deep levels is applied to each of the charge carriers getting trapped.

A promising direction for future work is to optimize the various modeled parameters. From each graph in section 6.6, one can see that by varying a given parameter, the QBER and sifted key rate change. However, these graphs assumed that when one parameter is

varied the others are kept fixed. One can define possible ranges of all modeled parameters and discretize these ranges. One can can then select every possible ordered set of choices of parameter values, run the simulation for each ordered set and map the resulting QBER and sifted key rates to the ordered sets of parameter values. One can then optimize the parameter space by selecting the ordered set of parameter values which minimize the QBER or maximize the sifted key rate. This is a brute force way of optimization. Current multi-variable optimization techniques can also be explored to speed up computational time.

A final note I'd like to make is that this kind of simulation is conducive to incorporating new effects of mechanisms like hold off times, etc. One would have to model the new mechanism separately and then integrate it appropriately into the the code. The remaining modeled mechanisms like the dead time recharge time, etc. would not have to be changed. This is an advantage compared to the existing analytical approaches of simulation, wherein incorporating new effects would mandate a new derivation which inevitably must consider all previously modeled parameters and evaluate their effect on the QBER and key rates in light of the new mechanism that is being modeled. This is a relatively cumbersome task every time new effects need to be modeled.

Moreover, apart from the QKD protocol, one might want to see how some quantity that depends on some of the modeled parameters varies as a function of the modeled parameters. For example, in section 5.2, I derived an approximate analytical expression for the dependence of observed afterpulsing probability on thermal count rate. However, I was not able to model the fact that thermal counts induce dead time regions in between themselves. Also, afterpulses of time tags other than the start time tag were not considered. To find the observed afterpulsing probability in the simulation presented in this chapter is straightforward. One would need to switch off the QKD protocol, attack and laser source in the code. The code can incorporate thermal counts, afterpulses, dead time and even recharge time into the time tag file. The observed afterpulsing probability can then be easily calculated because I keep track of the afterpulses in the time tag file. If you run the simulation for different values of thermal count rate, you would get a more accurate curve for observed afterpulsing probability than in fig 5.2.

Chapter 7

Conclusion

In chapter 2, a scalable prototype was designed and fabricated using Si APD as the detector. The prototype included all representative electronics. The detectors were characterized at $-80\,^{\circ}\mathrm{C}$ to show that all parameters were with the predetermined requirements. A cold finger was developed to help cool the detector module at off site locations and was tested to demonstrate its efficacy with dry ice.

Previous research had shown that dark count rate increases with proton irradiation and that thermal annealing was an effective way to decrease dark count rate. In chapter 3, we demonstrate that multiple applications of thermal annealing with proton irradiation phases in between is effective in decreasing dark counts to within the operable range suggested by previous link analysis studies. However, it must be noted that the first instance of annealing on a given detector achieved the greatest dark count rate reduction factor. This makes thermal annealing a viable option in satellite orbit say as opposed to laser annealing which requires an larger additional experimental setup and hence would increase the payload of the satellite. Colder temperatures were also explored to mitigate the increase of dark count rate induced by radiation. The coldest temperature our set up could go to in closed loop control was $-80\,^{\circ}\mathrm{C}$ which showed. However, we were able to go to temperatures as low as $-110\,^{\circ}\mathrm{C}$ albeit in open loop. A new thermal sensor would need to be developed that can help decrease the temperature the detectors can operate in closed feedback control, thereby decreasing dark count rate. Two approaches to annealing were explored. It was found that annealing only when the dark count rate exceeded a threshold value achieved slightly lower post-irradiation dark counts when compared to annealing after a fixed interval of time. Future studies could look into a low intensity beam of proton impinging on detectors which have been set to anneal at the same time. Further tests would need to be done in comparing the two approaches towards annealing using more data points when both DM's

are at the same cumulative radiation fluence mark when annealing takes place. Overall it was demonstrated that cold temperature and thermal annealing are effective methods to decrease dark count rate over the proposed life time of two i low Earth orbit.

In chapter 4, it was demonstrated that the DM can survive the conditions of vacuum which it will experience in low Earth orbit. Also, almost all detector parameters were within operable range. Detection efficiency was the only parameter which showed a significant change but it did so in response to temperature because some of the mating sleeves were not rated for cold temperatures. This is an easy fix. No secondary effects of vacuum like the overheating of electronic components due to lack of convective cooling was found. This shows the readiness of the DM for satellite applications.

In chapter 5, an important distinction between the usual notion of afterpulsing probability which relates to the likelihood of the trapping of charge carriers and the observed value of afterpulsing probability was made. For specifying the afterpulsing probability of an APD, it is not enough to calculate it from experimental. I would propose two ways to specify the afterpulsing probability: one approaches where the experimentally calculated value must be stated in conjunction with charge trapping probability, charge detrapping lifetime, dead time and thermal count rate. This is because the experimentally calculated value of afterpulsing probability will not be reproducible if any of the other mentioned parameters change. Of these, the thermal count rate is most prone to changing over time and hence at a minimum, the thermal count rate must be mentioned. Alternatively, one could specify the charge trapping probability in the APD during an avalanche. This value should not change as long as the APD substrate doesn't change in form (like additional defects introduced). With this information and the knowledge of the other parameters, the observed afterpulsing probability can be predicted for any experimental setup.

Furthermore in chapter 5, the theoretical groundwork for the afterpulsing algorithm I developed in chapter 3 was laid out. Under the assumption of the afterpulsing time constant being orders of magnitude higher than the thermal count rate, this provides an almost exact analytical expression which boils down to the intuitive sum of two different exponentials. Not only does it give an output value of the afterpulsing probability but it also gives an indication as to how reliable that value is, which is advantageous because all three algorithms presenting in section 3.3.8 give slightly different outputs.

Finally, a simulation of the quantum key distribution protocol on a classical computer was performed to extract statistical values like quantum bit error ratio and sifted key generation rate. This can be used to optimize a lot of the parameters of the experimental setup before a physical experiment is made. This helps to save time and monetary resources because certain parameters like the saturation value of the APD cannot be changed. Instead

a new detector might have to be developed. Such iteration of parameter values can be run in such a simulation before proceeding to a physical demonstration.

References

[1] Anisimova, Elena. Single-photon detectors for long distance quantum communications, 2018.

[2] Ronald L Rivest, Adi Shamir, and Leonard Adleman. A method for obtaining digital signatures and public-key cryptosystems. *Communications of the ACM*, 21(2):120–126, 1978.

[3] Peter W Shor. Algorithms for quantum computation: Discrete logarithms and factoring. In *Foundations of Computer Science, 1994 Proceedings., 35th Annual Symposium on*, pages 124–134. Ieee, 1994.

[4] Wikipedia contributors. Post-quantum cryptography — Wikipedia, the free encyclopedia, 2018. [Online; accessed 16-August-2018].

[5] J. Am G. S. Vernam. In this classical cryptographic protocol the message is combined with a random key string of the same size as the message to form an encoded message which cannot be deciphered by any statistical methods. g. s. vernam, j. am. inst. electr. In *Inst. Electr. Eng*, volume 55, 1926.

[6] Jin Gyu Lim. Laser annealing irradiated silicon single-photon avalanche photodiodes for quantum satellite receiver, 2018.

[7] C. H. Bennett and G. Brassard. Quantum cryptography: Public key distribution and coin tossing. In *Proceedings of the IEEE International Conference on Computers, Systems, and Signal Processing*, pages 175–179, Bangalore, India, December 1984.

[8] Charles H Bennett. Quantum cryptography using any two nonorthogonal states. *Physical review letters*, 68(21):3121, 1992.

[9] Artur K Ekert. Ak ekert, phys. rev. lett. 67, 661 (1991). *Phys. Rev. Lett.*, 67:661, 1991.

[10] Jean-Philippe Bourgoin. Experimental and theoretical demonstration of the feasibility of global quantum cryptography using satellites, 2014.

[11] Alberto Boaron, Gianluca Boso, Davide Rusca, Cédric Vulliez, Claire Autebert, Misael Caloz, Matthieu Perrenoud, Gaëtan Gras, Félix Bussières, Ming-Jun Li, et al. Secure quantum key distribution over 421 km of optical fiber. *arXiv preprint arXiv:1807.03222*, 2018.

[12] R Ursin, F Tiefenbacher, T Schmitt-Manderbach, H Weier, T Scheidl, M Lindenthal, B Blauensteiner, T Jennewein, J Perdigues, P Trojek, et al. Free-space distribution of entanglement and single photons over 144 km. *arXiv preprint quant-ph/0607182*, 2006.

[13] Shuang Wang, Wei Chen, Zhen-Qiang Yin, Yang Zhang, Tao Zhang, Hong-Wei Li, Fang-Xing Xu, Zheng Zhou, Yang Yang, Da-Jun Huang, et al. Field test of wavelength-saving quantum key distribution network. *Optics letters*, 35(14):2454–2456, 2010.

[14] Charles H Bennett, Gilles Brassard, Claude Crépeau, Richard Jozsa, Asher Peres, and William K Wootters. Teleporting an unknown quantum state via dual classical and einstein-podolsky-rosen channels. *Physical review letters*, 70(13):1895, 1993.

[15] Richard J Hughes, William T Buttler, Paul G Kwiat, SK Lamoreuax, GL Morgan, Jane E Nordholt, and Charles G Peterson. Quantum cryptography for secure satellite communications. In *Aerospace Conference Proceedings, 2000 IEEE*, volume 1, pages 191–200. IEEE, 2000.

[16] J-P Bourgoin, E Meyer-Scott, B L Higgins, B Helou, C Erven, H Hbel, B Kumar, D Hudson, I D'Souza, R Girard, R Laflamme, and T Jennewein. A comprehensive design and performance analysis of low earth orbit satellite quantum communication. *New Journal of Physics*, 15(2):023006, 2013.

[17] J-P Bourgoin, E Meyer-Scott, B L Higgins, B Helou, C Erven, H Hbel, B Kumar, D Hudson, I D'Souza, R Girard, R Laflamme, and T Jennewein. A comprehensive design and performance analysis of low earth orbit satellite quantum communication. *New Journal of Physics*, 15(2):023006, 2013.

[18] Matthew W Fishburn. *Fundamentals of CMOS single-photon avalanche diodes*. fishburn, 2012.

[19] Sergio Cova, M Ghioni, A Lacaita, Carlo Samori, and Franco Zappa. Avalanche photodiodes and quenching circuits for single-photon detection. *Applied optics*, 35(12):1956–1976, 1996.

[20] Robert GW Brown, Kevin D Ridley, and John G Rarity. Characterization of silicon avalanche photodiodes for photon correlation measurements. 1: Passive quenching. *Applied Optics*, 25(22):4122–4126, 1986.

[21] Henri Dautet, Pierre Deschamps, Bruno Dion, Andrew D MacGregor, Darleene MacSween, Robert J McIntyre, Claude Trottier, and Paul P Webb. Photon counting techniques with silicon avalanche photodiodes. *Applied optics*, 32(21):3894–3900, 1993.

[22] Robert GW Brown, Robin Jones, John G Rarity, and Kevin D Ridley. Characterization of silicon avalanche photodiodes for photon correlation measurements. 2: Active quenching. *Applied Optics*, 26(12):2383–2389, 1987.

[23] Roland H Haitz. Mechanisms contributing to the noise pulse rate of avalanche diodes. *Journal of Applied Physics*, 36(10):3123–3131, 1965.

[24] Sergio Cova, A Lacaita, and G Ripamonti. Trapping phenomena in avalanche photodiodes on nanosecond scale. *IEEE Electron device letters*, 12(12):685–687, 1991.

[25] Jeffery C Chancellor, Graham BI Scott, and Jeffrey P Sutton. Space radiation: the number one risk to astronaut health beyond low earth orbit. *Life*, 4(3):491–510, 2014.

[26] Christian Poivey and Gordon Hopkinson. Displacement damage mechanism and effects. In *Proc. ESAEPFL Space Center Workshop*, volume 9, 2009.

[27] Yong-Su Kim, Youn-Chang Jeong, Sebastien Sauge, Vadim Makarov, and Yoon-Ho Kim. Ultra-low noise single-photon detector based on si avalanche photodiode. *Review of scientific instruments*, 82(9):093110, 2011.

[28] JP Bourgoin, E Meyer-Scott, Brendon L Higgins, B Helou, Chris Erven, Hannes Huebel, B Kumar, D Hudson, Ian D'Souza, Ralph Girard, et al. A comprehensive design and performance analysis of low earth orbit satellite quantum communication. *New Journal of Physics*, 15(2):023006, 2013.

[29] D. Bronzi, F. Villa, S. Tisa, A. Tosi, and F. Zappa. Spad figures of merit for photon-counting, photon-timing, and imaging applications: A review. *IEEE Sensors Journal*, 16(1):3–12, Jan 2016.

[30] Request for prosposal. https://buyandsell.gc.ca/procurement-data/tender-notice/PW-MTB-575-13154.

[31] Question: Why is random jitter always modeled with a gaussian distribution? https://www.advantest.com/documents/11348/66671987-6dda-4e64-bc2f-6927065c5541. Accessed: 2018-08-16.

[32] Wikipedia contributors. Central limit theorem — Wikipedia, the free encyclopedia, 2018. [Online; accessed 16-September-2018].

[33] Gerhard Humer, Momtchil Peev, Christoph Schaeff, Sven Ramelow, Mario Stipčević, and Rupert Ursin. A simple and robust method for estimating afterpulsing in single photon detectors. *Journal of Lightwave Technology*, 33(14):3098–3107, 2015.

[34] Mark A Itzler, Xudong Jiang, and Mark Entwistle. Power law temporal dependence of ingaas/inp spad afterpulsing. *Journal of Modern Optics*, 59(17):1472–1480, 2012.

[35] Valerio Scarani, Helle Bechmann-Pasquinucci, Nicolas J Cerf, Miloslav Dušek, Norbert Lütkenhaus, and Momtchil Peev. The security of practical quantum key distribution. *Reviews of modern physics*, 81(3):1301, 2009.

[36] Xiongfeng Ma, Chi-Hang Fred Fung, and Hoi-Kwong Lo. Quantum key distribution with entangled photon sources. *Physical Review A*, 76(1):012307, 2007.

[37] Xiongfeng Ma, Bing Qi, Yi Zhao, and Hoi-Kwong Lo. Practical decoy state for quantum key distribution. *Physical Review A*, 72(1):012326, 2005.

[38] A survey of the prominent quantum key distribution protocols. https://www.cse.wustl.edu/~jain/cse571-07/ftp/quantum/. Accessed: 2018-08-16.

[39] Lectures 12 & 13: Introduction to quantum key distribution. https://inst.eecs.berkeley.edu/~cs191/fa14/lectures/lectures12and13.pdf. Accessed: 2018-08-16.

[40] WK Wooters and WK Zurek. Quantum no-cloning theorem. *Nature*, 299:802, 1982.

[41] Hoi-Kwong Lo, Xiongfeng Ma, and Kai Chen. Decoy state quantum key distribution. *Physical review letters*, 94(23):230504, 2005.

[42] Gregoire Ribordy, Jean-Daniel Gautier, Hugo Zbinden, and Nicolas Gisin. Performance of ingaas/inp avalanche photodiodes as gated-mode photon counters. *Applied Optics*, 37(12):2272–2277, 1998.