Palacký University Olomouc
Faculty of Science

Department of Optics

Photonic detectors as quantum random number generators

Bachelor's Thesis

Monika Dovičinová

Palacký University Olomouc
Faculty of Science

Department of Optics

Photonic detectors as quantum random number generators

Bachelor's Thesis

Author:	Monika Dovičinová
Study programme:	B1701 Physics
Field of study:	Optics and Optoelectronics
Form of study:	Full-time
Supervisor:	RNDr. Miroslav Ježek, Ph.D.

Thesis submitted on:	. . . .

Univerzita Palackého
Přírodovědecká fakulta

Katedra optiky

Photonic detectors as quantum random number generators

Bakalářská práce

Autor:	Monika Dovičinová
Studijní program:	B1701 Fyzika
Studijní obor:	Optika a optoelektronika
Forma studia:	Prezenční
Vedoucí:	RNDr. Miroslav Ježek, Ph.D.

Práce odevzdána dne: ....

Abstract

In this thesis, I generate random numbers using arrival time of photons as a random variable. Single-photon detection is discussed in relevance to quantum random number generators. I demonstrate the effect of the detector properties on the resulting probability distribution and the balance of bits. The results are compared to the theoretical predictions and prove to be in good agreement. Furthermore, entropy of the output is estimated and the randomness is enhanced by a randomness extractor. Different rates of detection are compared and the highest reported speed of the generation is 160 Mb/s. Randomness is verified using the Dieharder battery of statistical tests.

Keywords

Quantum random number generator, Randomness, Random numbers, Single-photon detection

Acknowledgments

I highly appreciate the guidance and encouragement my supervisor RNDr. Miroslav Ježek, Ph.D. provided. I would like to express my deepest gratitude for his ideas, knowledge, and wit he did not hesitate to share. His hardworking, energetic personality inspired me to do my best and seize new opportunities in the field of science. Without him, it would be impossible to complete this thesis. Also, I would like to thank my parents, who supported me during my studies, and my boyfriend, who has helped me stay calm in difficult times.

Monika Dovičinová

Declaration

I hereby declare that I have written this Bachelor's Thesis—and performed all the presented research and experimental tasks—by myself, while being supervised by RNDr. Miroslav Ježek, Ph.D. I also state that every resource used is properly cited. I agree with the Thesis being used for teaching purposes and being made available at the website of the Department of Optics.

Signed in Olomouc on . . . .

Monika Dovičinová

Contents

Chapter 1

Introduction

The demand for random numbers began to emerge along with the development of cryptography, numerical simulations, and statistical analysis. Random numbers have a significant role in keeping our data secure as they are necessary for producing security keys [1]. The performance of a security key directly depends on the quality of random numbers, which is strongly affected by their generation method [2]. Another important use of random numbers is the Monte Carlo method. It involves repeated random sampling to obtain numerical results, and it is used to analyze complex mathematical systems with a large number of variables and equations. In statistical physics, it is common to use this method to simulate the behavior of a system with a large number of particles [3]. Random numbers are, among other things, necessary in areas such as lotteries and gambling [4].

The need for random numbers is often met by pseudo-random number generators (PRNG), which are algorithms outputting a sequence of numbers similar to sequences of (trully) random numbers [2]. Probably the simplest algorithms are the linear congruential generator and the middle square method invented by John Von Neumann. In the middle square method, the seed is multiple digit number, for example four-digit number. Then it is squared and the middle four digits become the next seed [5]. The linear congruential generator is described by a simple recurrence relation $x_{n+1} = (ax_n + c) \mod m$ [6]. One of the widely used algorithms today is the Mersenne twister. It is a default pseudo-random number generator on platforms like Python and Matlab and is based on linear shift feedback registers and their generalizations [7]. The advantage of generating random numbers using an algorithm is a high speed of the process. However, no deterministic algorithm can produce perfect randomness, and PRNGs have proven inefficient in many applications. With the knowledge of the seed, it is possible to reproduce the whole sequence. It makes the sequence algorithmically predictable. After a certain time, the string of numbers begins to repeat itself and it is possible to predict the next output, which is obviously problematic in areas where security and privacy of data depends on random number generation [8]. The quality of the pseudo-random numbers depends on the complexity of the algorithm and on the appropriate choice of the seed. For example, the middle square method creates strings that begin to repeat in a very short time, which is highly dependent on the seed. The seed must actually be selected very carefully in order to achieve fairly good results [9].

Computers have much greater computational potential than before and that makes it easier to decode algorithms. Therefore the need for true randomness has emerged. True random number generators (TRNG) can be divided into hardware and physical. Hardware TRNG generates random numbers using processes directly related to the computer or accessories, like current memory location, mouse movement coordinates, or thermal noise [10]. Physical random number generators rely on physical processes such as decay of radioactive material, chaotic systems, atmospheric noise, etc. In industry, we can currently find random number generators based on free-running oscillators (FRO) [11]. Its randomness is derived from electronic noise present in logic circuits and it offers easy technological implementation [12]. There are also problems tied with these methods. External influences can impair the function of the generator to the extent that statistical imperfections occur. Statistical correlations express the relation between two processes or quantities. Random numbers may exhibit dependence on previous events of the system or on some external variables [2].

The category of physical TRNGs also includes quantum random number generators (QRNG), which promise even better results. QRNGs rely on the fact that randomness is an inherent part of quantum processes [2]. Before measurement of an observable, state is represented by a linear combination of eigenstates of given observable. When measurement takes place the system transforms into one of the eigenstates with a specific probability. Therefore, we cannot predict the result of a measurement, we can only talk about probabilities of different outcomes. The probability distribution corresponds to the context of a given physical system [13]. The analogy of the classical bit is the so-called qubit. In the case of one-qubit system, it is represented by a state that is in a superposition of two eigenstates [14]. System of qubits can exhibit quantum entanglement, which is a phenomenon applicable to quantum random number generation [15].

We often encounter the use of coherent states of light or individual photons. Coherent signals can be generated by lasers and measured using homodyne detection [16]. Very efficient QRNG strategy is measuring vacuum noise [17, 18, 19]. It is realized by homodyning of the vacuum state of light, where we sample normally distributed quadrature values. The vacuum noise QRNGs have reached random bit generation rate up to 8 Gb/s [20]. They have been implemented on optics chips [21, 22]. Furthermore, random numbers can be generated using fluctuations in the phase or intensity of the laser operating near threshold level [23, 24].

There is a relatively large number of photonic implementations [2]. Some of those rely on the behavior of an incoming photon at a beam-splitter to generate data [25]. The beam splitter either has balanced ratio of reflectivity and transmittance, or it is possible to sent circurally polarized photons at a polarizing beam splitter. A value 0 is then assigned to a detection event in one detector and 1 to an event in the other detector. The ones and zeros are equaly distributed [26], but the disadvantage is that only one bit per detection is generated.

The photon arrival time can also represent a quantum random variable [25, 27]. The intervals between successive photons are governed by Poissonian statistics. The detection is provided by single-photon detectors, which affect the resulting statistics [28]. Random numbers are obtained either by binary representation of measured time differences, or according to the position in the probability distribution. Generally, the resulting sequence is not balanced, but it is possible to achieve equal distribution by post-processing. The advantage of this quantum random number generator is the ability to generate more than one bit per detection, which is not always an option in other setups [27].

Moreover, there are QRNGs based on a photon counting since some single-photon detectors are able to distinguish the number of photons arriving at the same time [29]. Another method is measuring interaction of photones with phonons, when Raman scattering takes place. Raman photons can be produced by illuminating a highly nonlinear glass [30]. Binary value can be assigned to different intervals of wavelenghts of equal probability. Then, there are attenuated pulse generators, which provide randomness by binary measurements of coherent states [31]. Randomness can be also extracted from bell equations violation, when examining pairs of entangled photons. Bell test is done repeatedly on incoming photon pairs [15].

Theoretically, quantum generators are able to produce perfect randomness. In reality, it depends on the implementation. It is important what the setup is and what we draw randomness from. It needs to be considered whether there are classical effects causing correlation and how the numbers are extracted and post-processed. The method of detection often plays a big role in the discussion of imperfections. In photonic applications, superconducting nanowire single-photon detectors (SNSPD) and single-photon avalanche diode detectors (SPAD) are used most often [32]. The most impactful imperfection of detection is the dead time of the detector. A single-photon detector needs this time to recover after detection event and thus is unable to measure until the dead time passes [33]. There are also other phenomenons such as afterpulsing. Single photon can result in more than one electrical pulse, which leads to an overestimation of the incident count rate by up to 10 % and increase of the dead time [34]. Moreover, false detections sometimes take place. They are referred to as dark counts [35].

Detection effects were investigated in the context of QRNGs and it was shown that the dead time causes greater imbalance of bits than it would be without the dead time [28]. In the probability distribution of time differences between succesive photons, from the beginning of the distribution until the deadtime, the value of probability is zero. Afterpulsing shows in the first non-zero bins in the distribution. It causes the bins to have a higher value than corresponds to the laws of the distribution. Afterpulsing cas also increase autocorrelation coefficients [25].

Randomness extractors are a way of post-processing used to increase randomness. It is a function or algorithm that erases part of the data based on some condition. Postprocessing significantly limits the speed of generation of physical generators and QRNGs. Most of the records set in the QRNG generation speed were conditioned primarily by an increase in the processing speed of the classical signal and the subsequent postprocessing of random bits [36].

The preferred type of generator must be selected according to the application. PRNGs offer the simplest and fastest generation, but it is also the most problematic and imperfect generator type. QRNGs are without a doubt the highest quality, but they are still slower than PRNGs. Currently, the fastest QRNG is based on intensity fluctuations in a laser and generates 250 000 Gb/s [37].

In this thesis, I will focus on generating random numbers using the arrival time of individual photons as a random variable. In Chapter 2, I will describe all methods I used. Specifically, I will discuss entropy estimation, bit balance of resulting binary string, correlations and algorithmic compressibility in Section 2.1. In Section 2.2, I will focus on Poissonian process and describe its properties. The aim is to derive Poisson statistics of time intervals between adjacent detections. Then, I will describe the basic principles of single-photon detection with a focus on SPAD and SNSPD detectors in Section 2.3. Among other things, I will describe the properties of the detector that affect the detection quality. In Section 2.4, I will briefly describe randomness extractors and hash functions and in Section 2.5, I will discuss randomness testing. In Chapter 3, I will present results. Firstly, the scheme of the experiment and component properties will be described in Section 3.1. Then, in Section 3.2, I will present probability distribution and discuss balance of bits for different rates. Entropy will by estimated in Section 3.3 and in Section 3.4, results of randomness testing will be presented. The experimental results agree with the theoretical models. The generation speed of 160 Mb/s is reached. It is primarily limited by a detector dead time of 28.7 ns. Chapter 4 will include recapitulation of used methods and achieved results.

Chapter 2

Methods

2.1 Random numbers

Binary random numbers are composed of ones and zeros that occur in a sequence. The values ought to be uniformly distributed over the set of ones and zeros and it should be impossible to predict future values based on past ones [38]. This preposition is related to algorithmic randomness. A sequence is algorithmically random if there is no algorithm able to generate the sequence using a shorter string of bits than the size of the original one is. This definition directly ties with the definition of complexity K proposed by Kolmogorov. The complexity K of a binary sequence of length N is the binary length of the shortest program able to recreate the sequence. If complexity K is approximately the same as N, the sequence of N bits is considered algorithmically random. Algorithmic compression refers to the case when K is smaller than N. It is important to note that K in never equal to N, because no sequence is absolutely incompressible. That is the reason why the definition of algorithmic randomness is only approximate. For quantum sources of randomness, it had been proven that complexity K actually cannot be computed, it can only be estimated [39].

Another plausible definition of randomness is statistical randomness. It takes the uniformity of the distribution and auto-correlations into account [40]. In a binary representation of values of a random variable, it is necessary to examine the proportionality of the number of zeros and ones for individual bits. The so-called least significant bits are at the end of the binary representation and are often unbalanced [41]. This means that in a certain bit, zeros prevail over ones or vice versa. The least significant bits have the least effect on the value of the binary number. For large number N of random variable values, we can introduce a probability of bit i being sampled as 1

$$\lim_{N\to\infty} \frac{n_i(1)}{N} = p(b_i = 1), \tag{2.1}$$

where $n_i(1)$ is number of ones in i-th bit position. Autocorrelation is the correlation of a string with a delayed copy of itself. It shows how much dependent the output of random number generator is on previous events in the generator.

Another important attribute of random numbers is entropy, which quantifies the average information of a source/generator. Renýi entropy is a generalized entropy in information theory and is described by

$$H_a(X) = \frac{1}{1-a} \log_2 \left(\sum_{i=1}^{N_{out}} p_i^a \right), \tag{2.2}$$

where X is a discrete random variable with N_{out} possible outcomes with corresponding probabilities p_i. For a approaching 1, we get Shannon entropy [42], which is equal to

$$H_1(X) = -\sum_{i=1}^{N_{out}} p_i \log_2 p_i. \tag{2.3}$$

Shannon entropy was named after Claude Shannon, who first introduced the idea of entropy in information theory [43]. Shannon entropy was one of the first entropy measures in this area and is widely used to this day. For a approaching infinity, we get min-entropy, which is equal to

$$H_\infty(X) = -\log_2(\max p_i). \tag{2.4}$$

It is called min-entropy because it is the smallest measure of all Renýi entropies, which also makes it the most conservative measure [42].

2.2 Poissonian process

Poissonian statistics generally apply to processes in which individual events are independent, are considered random and output consists of discrete events [44]. Detection rate R is given by the number of detection events per second. The average detection rate is described by

$$R = \frac{n}{T}, \tag{2.5}$$

where n is the number of detected photons in time T. When the rate is constant, the number n of detected photons per time T is

$$n = RT. \tag{2.6}$$

The probability of one detection event in time interval $\delta T \to 0$ is

$$p(1, \delta T) = R\delta T. \tag{2.7}$$

Considering that $p(2, \delta T) << p(1, \delta T)$, probability of no detection event in δT is

$$p(0, \delta T) = 1 - p(1, \delta T) = 1 - R\delta T. \tag{2.8}$$

Statistical independence of events will allow us to express the probability of absence of detection event in the time interval $T + \delta T$ as

$$p(0, T + \delta T) = p(0, T)p(0, \delta T), \tag{2.9}$$

which leads to differencial equation

$$\frac{d}{dT} p(0, T) = -Rp(0, T). \tag{2.10}$$

With initial condition $p(0,0) = 1$, we get

$$p(0,T) = e^{-RT}, \tag{2.11}$$

which describes the distribution of waiting times. Integrating $p(0,T)$ from zero to infinity results in $\frac{1}{R}$. Therefore, to get a normalized distribution, it needs to be multiplied by R. It is also convenient to adress $p(0,T)$ as $p(T,R)$ since the zero is fixed and the distribution depends on T as a variable and R as a parameter. The normalized distribution is then described by

$$p(T,R) = Re^{-RT}. \tag{2.12}$$

Using the probability distribution $P(T,R)$, we can analogically express the bit balance from the previous section. When measuring, we perceive time as a discrete quantity according to the time resolution. The resolution then represents the size of one time-bin.

In order to express the probability $p(b_i = 1)$ of bit i being sampled as 1 (2.1), we integrate over the probability distribution for the time-bins where the i-th bit is sampled as 1. Therefore, a selection must take place. If we multiply the probability distribution by the ones and zeros that occupy the i-th bit position, we obtain only the required bins. Using time T, the pattern of ones and zeros cas be expressed as $\lfloor \frac{T}{2^i} \rfloor \mod 2$. Then, the balance of bits is described by

$$p(b_i = 1) = \frac{\int_0^{2^N} P(T,R) \lfloor \frac{T}{2^i} \rfloor \mod 2 \, dT}{\int_0^{2^N} P(T,R)\, dT} \tag{2.13}$$

introduced in [28]. After applying $P(T,R)$ from (2.12), we obtain

$$p(b_i = 1) = \frac{1}{1 + \exp{(RT2^i)}}. \tag{2.14}$$

2.3 Single-photon detection

In order to quantify the properties of a detector, there must be some parameters defined. After every detection, there is a dead time τ_d, in which the detector is not able to detect another photon [45]. The dead time is caused more by electronics parts of the detector rather than by the detection element [32]. In general, detectors with lower dead times are preferable, because the dead time limits the detection rate according to equation

$$R_{detected} = \frac{R_{incident}}{1 + \tau_d R_{incident}}. \tag{2.15}$$

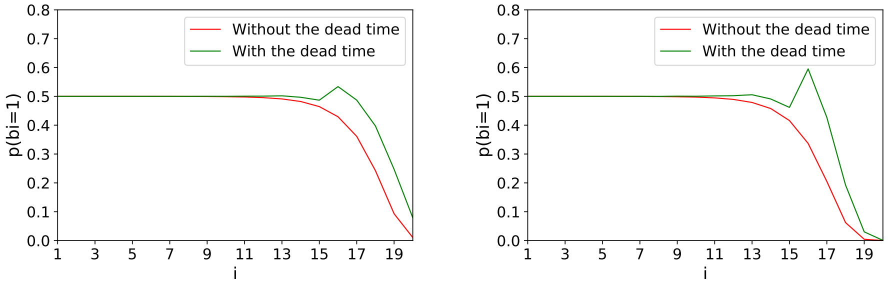
For most detector types, there is a finite probability that false detection events (dark counts) take place. Generally, detection events are referred to as counts. Dark count rate D is also a measure of quality as it is caused by the material or the susceptibility to external noise.

After the absorption of a photon, an output electrical pulse is generated. There is a time interval between those events and the variation in this time interval is the timing jitter. Timing jitter limits the timing resolution [46]. Afterpulsing is a phenomenon that occurs in SPAD detectors. It arises when a single incident photon results in more than one electrical pulse. However, this effect together with the dark counts is outweighed by the dead time effect so the detected rate is always smaller than the incident rate [34]. It may also happen, that charge accumulates in the detector during the dead time and right after the detector fully recovers, the accumulated charge creates a signal called twilight pulse.

The influence of dead time on the statistical distribution of time intervals between adjacent events should be also mentioned. The time between two detections cannot be smaller than dead time. Therefore the probability of events is zero for smaller time differences. Considering this fact, probability density of waiting times [28] is described by

$$P(T, R, \tau_d) = \begin{cases} 0, & T < \tau_d, \\ R \exp\left(-R(T - \tau_d)\right), & T \geq \tau_d. \end{cases} \tag{2.16}$$

Using (2.16) in Eq. (2.13), we obtain the probability of bit i being sampled as 1 in the case of a detector with dead time.

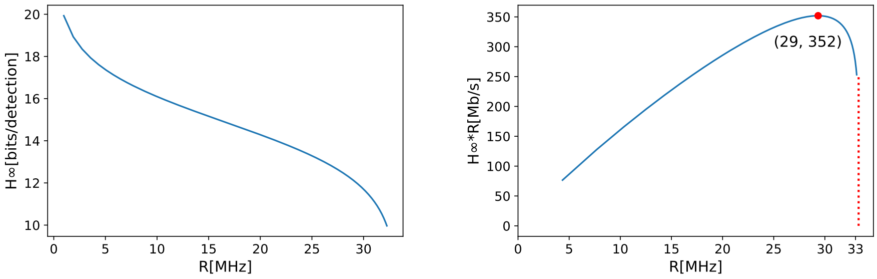


(a)

(b)

Figure 2.1: Bit balance distributions (2.13) without the dead time (2.14) and with the dead time 30 ns for time resolution 1 ps and detection rate (a) 5 MHz, (b) 15 MHz.

In Fig. 2.1, we can see the effect the dead time has on the bit balance. At first, both functions remain at value 0.5 and then the dead time causes oscillations after which the functional value decreases to zero. The oscillations are more prominent with higher rates. Without the dead time effect, we can see decrease without any oscillations. Lower rates offer more balanced bits per detection, but the detections are less frequent than in the case of higher rates. That raises the question of the optimum rate, at which it is possible to produce maximum number of balanced bits per second. Using the equation describing rate saturation (2.15), we can observe that the function of min-entropy (2.4) multiplied by detected rate R has a maximum. As we can see in equation (2.16), R is actually a maximum value in this distribution, therefore it can be used as $\max p_i$ in Eq. (2.4). For a demonstration, we set the dead time to 30 ns and show in Fig. 2.2a that the entropy per detection decreases with rate as expected. The saturation of rate creates an inflection point, where the function changes from convex to concave. In Fig. 2.2b, the maximum of min-entropy multiplied by rate is at 29 MHz. The detected rate is saturated to its final value at the rate equal to $\frac{1}{\tau_d}$, which is 33.3 MHz in this case.



(a)

(b)

Figure 2.2: Min-entropy H_∞ alone (a) and multiplied by saturated detection rate R (b) dependence on saturated detection rate R for dead time 30 ns.

Conventional single-photon detectors consist of photomultiplier tubes (PMT) or avalanche photodiodes [45]. In a photomultiplier, there are components in a vacuum glass tube. Light is absorbed on a photocathode and free electrons are generated. Electrons are emitted due to the photo effect and are accelerated with a high voltage to a first dynode, where they generate secondary electrons. This is repeated several times and at the end of the cycle a strongly amplified photocurrent can be collected [46]. PMTs offer larger active areas, more than ten millimeters. The disadvantages are that they require large operating voltages and are easily damaged and expensive. In certain types of PMT, the excess noise of the multiplication process is low enough to distinguish one or multiple photons. Photomultiplier tubes cover the spectral range of 115–1700 nm and have maximum efficiency of around 40 % at 500 nm wavelength. The highest reported count rates are up to 10 MHz, and the typical jitter is 300 ps [32]. Microchannel plate PMTs have glass capillaries coated with a secondary electron-emitting material to achieve a single continuous dynode. Microchannel plate PMTs improved timing jitter to 20 ps [32].

A single-photon avalanche diode (SPAD) is a photodiode, which operates high above the breakdown point at high voltages thus it is able to detect even very small signals such as individual photons with a delay that corresponds to pico seconds. Signal amplification through an avalanche process takes place over a distance of only a few micrometers and amplifies the photocurrent by a significant factor and thus increases responsivity. Responsivity is the ratio of generated photocurrent and incident optical power [47].

An avalanche photodiode consists of four layers: N +, P, pure semiconductor and P +. Around the N + and P layers, between which the avalanche phenomenon occurs, there is a protective ring made of an N-type semiconductor, which increases the diode's resistance to surface voltage breakdown. The incident light causes an electron-hole pair. The electron is transported to the avalanche region by a strong electric field, where it is accelerated to such a speed that a collision with the crystal lattice causes another electron-hole pair. The new electron is also accelerated by a strong electric field, and gradually, as if in a chain reaction, more and more new electron-hole pairs are formed, creating an avalanche effect. Thus, a single photon can cause ionization of the crystal lattice leading to an avalanche. Silicon single-photon avalanche photodiodes are now a well-established component in quantum measurements [47].

The Geiger mode is used when extremely high gains are required. In this mode, a voltage just above the breakdown voltage is used, where a single electron-hole pair causes a strong avalanche. Once this occurs, the external electronics need to reduce the voltage below the breakdown voltage of the diode to return it to its original state, in which it is able to detect more photons [46].

Another significant method of detection uses a superconducting nanowire [48]. Superconducting nanowire single-photon detectors (SNSPD) have lower than 50 ps timing jitter. Detection takes place due to the current density being just below the critical level, above which the wire reverts to normal resistance. After photon absorption, small spot on a wire acquires normal resistance, which causes increase of current in nearby spots and therefore resistance area forms all the way across the width of the wire. This occurrence causes sudden increase of voltage allowing detection [49].

2.4 Extractors and hash functions

It was shown that quantum processes are random by nature. However, when implementing quantum random number generator in practice, we have to deal with issues that can spoil the randomness. To deal with these negative effects, data post-processing is necessary. The first technique of randomness extraction was proposed by Von Neumann [5]. The data are divided into two bit strings. If the two bits match, no output is generated and if the bits are different, the value of the first bit is the output. The number of bits is reduced to less than a half and quality increases. Another option is a least-significant-bits operation. When the least-significant bits are disproportionate to ones and zeros, removing them from the data file can establish an unitary distribution and thus enhance the quality of the string [50]. More elaborate and newer extractor is Trevisan's. He proposed randomness extraction based on pseudorandom number generators [51].

A hash function is a mathematical function or algorithm for converting input data into a smaller number or string. Hash functions are used to search for items in a database, detect duplicate records, search for malware by an antivirus program. In the form of a cryptographic hash function, it is used to create and verify an electronic signature, ensure data integrity, protect stored passwords, etc. Any amount of input data results in the same length of output and a small change in the input data will cause a large change in the output. It is practically impossible to reconstruct the original data from the hash (fingerprint). These features ensure safety. In principle, hashing can be used as an extractor. When searching for similar data, hash from a part of the data is counted several times and the match of the fingerprints and thus matching parts of data are spotted [52, 53] and can be eliminated in order to increase randomness.

2.5 Randomness testing

Randomness tests are based on statistical testing. The null hypothesis H_0 is first established. In this case, H_0 is that the sequence is random. Then, the alternative hypothesis H_A is that the sequence is not random. It could be the other way around but most testing batteries use this setting. Test statistic T is used for the testing itself. It is a formula, a function of data that indicates how probable the measured data are if the null hypothesis applies. The zero distribution is the distribution of the test statistic with H_0 applying. Moreover, p is the probability that at H_0 applying the result of test statistic T would acquire a value that indicates that H_0 does not apply. The test level is denoted by α. It is a selected number from the interval from 0 to 1. Most often $\alpha = 0.05 = 5\%$. If $p < \alpha$, then the validity of H_0 is very unlikely and then we reject H_0 and accept H_A. Either H_0 applies, but there have been data that appear with a probability less than α (something very unlikely has happened), or indeed H_A is valid, which we are leaning towards. If $p \geq \alpha$, then this does not mean that we reject H_A, but only do not reject H_0. There are two types of errors. Type I is that we reject H_0 while it applies and type II is accepting H_0 while it is actually not true [54].

NIST (National Institute of Standards and Technology) tests and Dieharder test verify statistical randomness. NIST battery includes 15 tests. First two tests are focused on proportion of ones and zeros in the whole sequence and in blocks. Then, there is a test for number of uninterrupted sequence of identical bits. Fourier transform test detects periodic features by applying discrete Fourier transform to the sequence, linear complexity test computes the length of the shortest linear feedback shift register that generates s^n sequence as its first n output items. Other tests involve random walks across the data. Purpose and description of each test is listed in the manual together with recommended input and examples [54]. Dieharder battery offers similar tests [55]. The major disadvantage of statistical tests is that they often fail to distinguish algorithmically generated sequences from truly random ones [39].

Chapter 3

Experiment and discussion

3.1 Scheme and experiment description

Infrared light from a luminiscence diod (LED) is attenuated by neutral density filter (ND), a polarizer, and an analyzer, which allows for continuous tunability. Part of the intensity is transferred into a single-mode optical fiber connected to a SPAD detector. Then, time tagger converts detection times to digital data and streams them to a computer. To verify parameters of the time tagger such as time jitter, signal or clock generator can be used. Before the measurement depicted in Fig. 3.1, we measure the optical signal with a biased photodiode (DET36A THORLABS), which provides photocurrent, so it can be detected by a voltmeter. The reading was 2.2 mV, the multimeter had 1 MΩ internal resistance R and therefore the power P equaled 4.4 nW according to equation

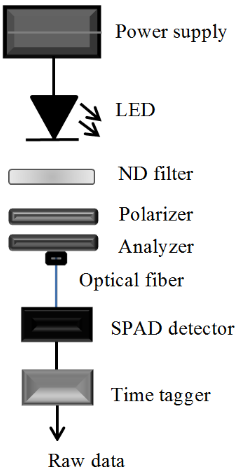


Figure 3.1: Experiment scheme. Infrared light is emitted by a LED diode, attenuated by ND filter and polarizers, and detected by a SPAD detector. Detection data are obtained using time tagger.

$$P = \frac{U}{RS}, \tag{3.1}$$

where U is the voltage and S is the responsivity of the photodiode equal to 0.5. LED diode we used emitted 810 nm wavelength. Since power represents energy per time unit, it was possible to calculate the number of photons n emitted per second as

$$n = \frac{P}{h\frac{c}{\lambda}}, \tag{3.2}$$

where denominator of the fraction represent energy of one photon (c is the light speed in vacuum, h is Planck‘s constant and λ is the wavelength). The result corresponds to approximately 10^{10} photons, which is sufficient as an estimate for the required attenuation. We used an ND filter to reduce the number to 10^8. Filter NE-530 lets through 0.9 % infrared light of given wavelength. Then, it was possible to attenuate as needed by the polarizer and the analyzer. Each lets through 42 % unpolarized infrared light at 810 nm wavelength. At a certain polarizer position, we get only the intensity corresponding to the extinction (1:10000) and after turning 90° the intensity peaks. The SPAD detector is sensitive to light and could be damaged, so it is necessary to find the position of the lowest intensity as a starting point before the main measurement. Components used in the measurement are listed with more details in Tab. 3.1 and Tab. 3.2. The stability of the experiment is affected by fluctuations of intensity of the infrared light, which depends on the stability of the power supply output. Our power supply utilizes line regulation to enhance stability. The fluctuations than make 0.01 % of the average current and voltage. Among other things, it

Component	**Type and relevant information**	
Power supply	TTi PL303QMD quad-mode dual power supply	
LED	MTE2081-OH5	810 nm wavelength
ND filter	NE530	0.9 % transmission at 810 nm
Polarizer, Analyzer	2× LPNIRE050-B	42 % transmission at 810 nm
Optical fiber	NUFERN 780-HP	Single-mode fiber
SPAD detector	EXCELITAS SPCM CD 3432 H	Serial number: 24336 Dark count: 46 Hz Light count: 31 MHz Dead time: 28.7 ns Total after-pulse: 0.1 High level voltage: 5.25 V
Time tagger	SWABIAN Time tagger 20	RMS jitter: 43 ps Transfer rate: 8.5M tags/s Digital resolution: 1 ps

Table 3.1: Components of the main measurement.

Component	**Type**
Signal generator	SMB 100A
Programmable multimeter	1705 TRUE RMS
Biased photodiode	DET36A/M Si biased detector

Table 3.2: Components used for testing.

is important to ensure that the background intensity is constant and the measurement is not disturbed by changes in lighting conditions in the laboratory. In my experiment, the background noise was slightly different in each measurement, but throughout the measurement for a given rate it stayed constant. The differences might have been caused by the light emitted by my notebook screen and its position relative to the aperture. The highest measured background rate was 340 Hz and the lowest was 250 Hz.

Maximum rate we were able to achieve was 11 MHz. By turning the polarizer, we were able to attenuate the infrared light continuously from background noise to the maximum possible rate. Saturation rate is according to equation (2.15) 34.8 MHz, maximum transfer rate of the time tagger is according to the manufacturer 8.5 MHz. The connection is provided by USB 2.0, which can transfer 60 MB/s and one tag is about 6 MB. That results in maximum transfer rate of 10 MHz, which shows that the transfer rate can be higher than the guaranteed one. In any case, the guaranteed transfer rate of the time tagger respresents the biggest limitation.

With a relatively small delay, it is possible to observe the measured rate in time tagger application. Using this application, we can set the rate by turning the polarizer. For the main measurement, time tagger Python package was installed. First, the `createTimeTagger` command must be used. Then, trigger level should be set by `setTriggerLevel`. `FileWriter` creates a file with the data and `FileReader` allows to upload them again. Then, the `getTimeStamps` command helps extracting time stamps into a separate variable. To obtain the time differences, each time stamp must be subtracted from the following one. A histogram of the time differences shows the negative exponential distribution (2.12). The graph depicting the balance of bits (2.13) in data can be obtained by creating a two-dimensional data field. The binary representation of time differences is coded in lines and columns represent the bit positions. In Tab. 3.3, there is an example of the procedure. Then the number of ones in a column is summed up and divided by the number of lines and that number represents the functional value for the given bit position.

Time difference	Binary representation					
	Bit position					
[ps]	1	2	3	4	5	...
219692	0	0	1	1	0	...
136385	1	0	0	0	0	...
51654	0	1	1	0	0	...

Table 3.3: An example of making a two dimensional data field out of time differences. The values of bit position are 1, 2, 3, 4, 5,... according to the order of the bits in the binary representation of time differences.

There are two ways to obtain random numbers from raw bit stream. The first is dividing the distribution by quantiles into sections of equal probability and then assigning each interval a binary value. For example, distribution divided by quartiles would have binary values 00, 01, 10 and 11. Generally, 2^n intervals result in n bits per detection. This method works for an arbitrary distribution, which is particularly convenient in the case of complex detection process where the accurate model is not known. However, it requires excessive computational resources. With increasing n, the process in Python is slowing down exponentially. The highest number of bits I generated using this method is 24 per a single time difference.

The other method is directly tied with the balance of bits and entropy. It is addressed as the least-significant bit operation as mentioned in Section 2.1. After examining the balance of bits, we decide how many bits per detection should maintain in the data file to keep them balanced. Acceptable deviation in balance is determined by min-entropy. It can help to decide from which bit on to cut off, since the output of min-entropy function is a number of bits per detection. Then, it is possible to implement a randomness extractor. Von Naumann extraction can be performed by reshaping random numbers into two dimensional data field with two columns and then deleting the lines if the bits match, or keeping only first bit if they differ.

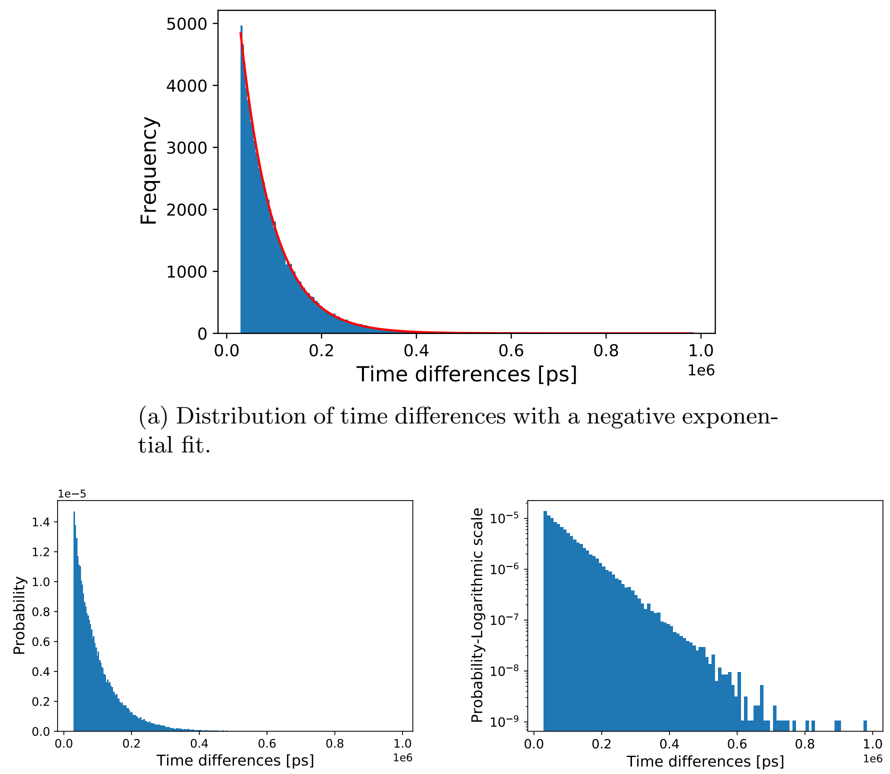
3.2 Probability distribution and balance of bits

Measurements of time differences were made for rates 100 kHz, 500 kHz, 1 MHz, 3 MHz, 5 MHz, 7 MHz, and 10 MHz. Data analysis was performed in Python, where it was possible to obtain a histogram of frequency and a probability density distribution.

Fig. 3.2a shows the typical distribution of time differences for 3 MHz rate. Horizontal axis represents the time differences in picoseconds and vertical axis the number of detection events. The negative exponential trend is visible as the fit curve traces the bin heights. The fitting function has the form $a \exp(-bx)$, where x represents the values on the x-axis and a and b are the fit parameters. Moreover, two important discrepancies are visible. The first bin is higher than it should be according to the negative exponential fit, which is caused by twilight pulses. In Tab. 3.1, we can see that the afterpulsing of the detector occupies 0.1 % of the number of counts. The percentage of counts that are above the negative exponential fit is presented for different rates in Tab. 3.4. It includes the after-pulses as well as twilight pulses. That is the reason why it is above 1 % and not just 0.1 %. Also, the percentage of after-pulses alone should not be changing with rate. Up to 7 MHz, the percentage grows and then it drops at 10 MHz by 0.5 %. It is also visible that the distribution does not start at zero. It actually starts at the dead time of the detector, which is 28.7 ns. In

R [Hz]	Percentage of after-pulses and twilight pulses
100 k	1.364 %
500 k	1.411 %
1 M	1.435 %
3 M	2.165 %
5 M	2.260 %
7 M	2.447 %
10 M	1.984 %

Table 3.4: Standard deviation of fit parameters modeling a negative exponential distribution for measured rates.



(a) Distribution of time differences with a negative exponential fit.

(b) Probability distribution of time differences.

(c) Probability distribution of time differences on logarithmic scale.

Figure 3.2: Measured distribution for 3 MHz detected rate.

Fig. 3.2b, the probability density distribution is shown. It has the same shape as the distribution of counts in Fig. 3.2a because it is just scaled, so that the sum over the whole distribution equals one.

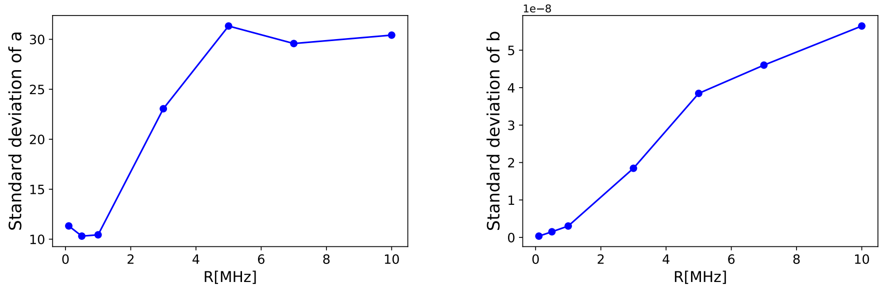
Depicting the distribution on logarithmic scale results in almost linear decrease of bin heights in Fig. 3.2c. Imperfections at the end of the distribution are also more visible.

The fitting was also done for other measured rates and standard deviation of fit parameters was estimated, see Tab. 3.5. The deviation is the measure of how much the data correspond to the negative exponential fit. Fig. 3.3 shows dependence of standard deviation of fit parameters on rate R. In Fig. 3.3b, there is a visible increase of the deviation of the parametr b depending on rate. This trend is less pronounced for a in Fig. 3.3a, but we could still make the case for increase on average. It shows that the detection errors become more significant with incresing rate since they have a greater effect on the shape of the distribution.

Then the balance of bits in the data was examined by comparing number of ones and zeros in each bit position as was described in Section 3.1. Results for rates 100 kHz, 500 kHz, 1 MHz, 3 MHz, 5 MHz, 7 MHz and 10 MHz were

R [Hz]	Standard deviation of a	Standard deviation of b
100 k	1.133	3.477×10^{-10}
500 k	1.031	1.508×10^{-9}
1 M	1.044	3.042×10^{-9}
3 M	2.305	1.847×10^{-8}
5 M	3.133	3.847×10^{-8}
7 M	2.958	4.602×10^{-8}
10 M	3.042	5.646×10^{-8}

Table 3.5: Standard deviation of fit parameters modeling a negative exponential distribution for measured rates.



(a)

(b)

Figure 3.3: Standard deviation of fit parameters a (a) and b (b) dependence on detected rate R.

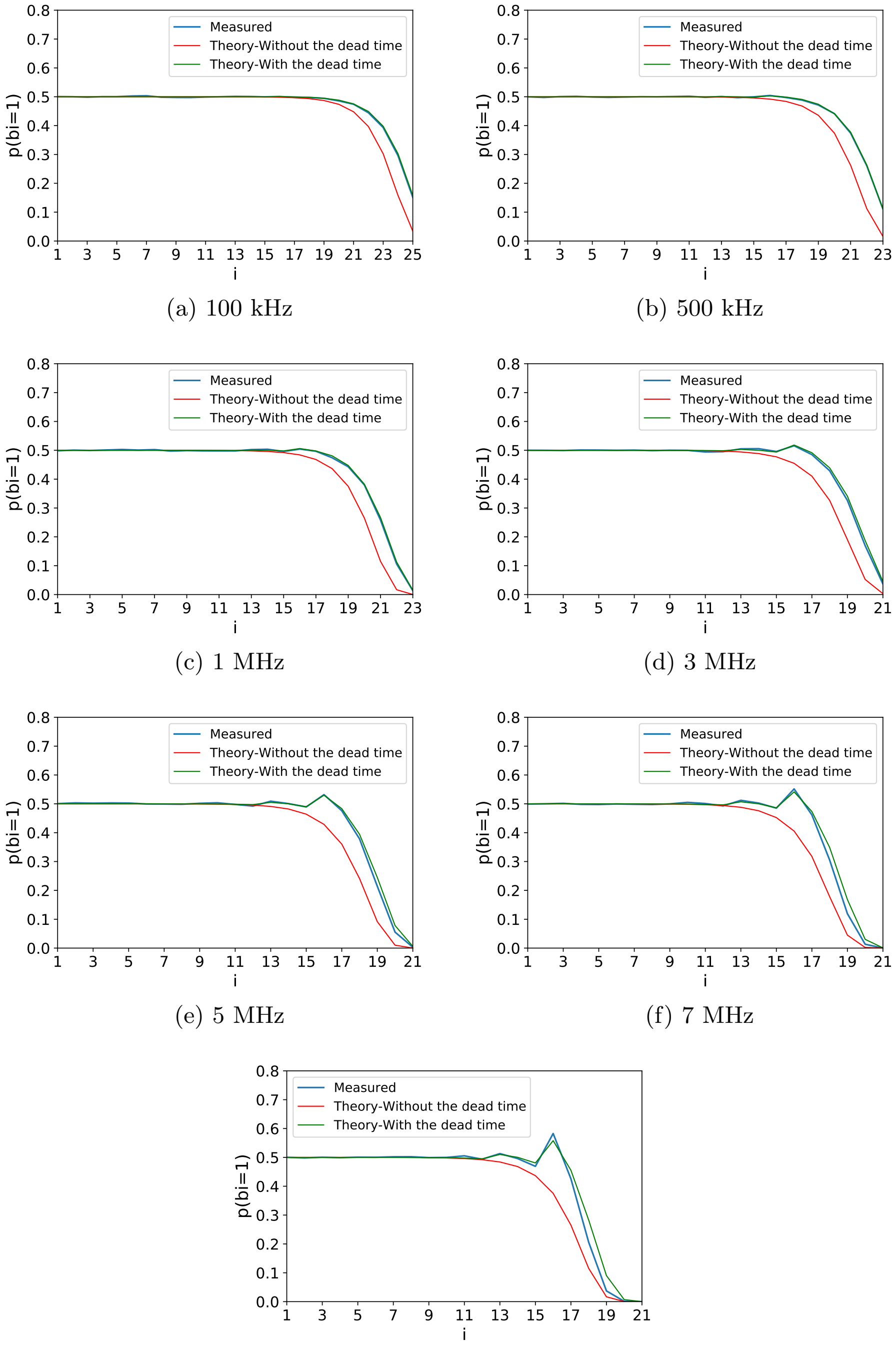
obtained and are presented in Fig. 3.4. Fig. 3.4 shows that the theoretical curve according to equation (2.13) with the dead time models the real situation very well especially for lower rates. The number of balanced bits decreases with increasing rate. As we can see, the blue curves and green curves coincide for rates up to 3 MHz. For higher rates, there is a slight shift between the blue curve and green one at the end before they drop to zero. The theory predicts more balance then there really is for these bit positions.

Both green and blue curves are slightly shifted from the red one. It shows that accounting for the dead time predicts more balance in certain bit positions.

In Fig. 3.4c, it is visible that the shape of blue and green curve is different from the red one and their functional value goes above 0.5 at a certain points, where ones are more prevalent than zeros.

Visible oscillations are formed from 3 MHz in Fig. 3.4d up and the peak is getting higher with increasing rate because the dead time effect is emphasized, when photons arrive more frequently.

Entropy then answers the question of the optimum rate, at which we obtain the highest number of balanced bits, and determines what deviation from 0.5 still constitutes as balanced.



(a) 100 kHz

(b) 500 kHz

(c) 1 MHz

(d) 3 MHz

(e) 5 MHz

(f) 7 MHz

(g) 10 MHz

Figure 3.4: Bit balance distributions (2.13) without the dead time (2.14), with the dead time, and the measured distributions.

3.3 Entropy and test results

Min-entropy was estimated from the measured probability distributions. Approximatively, $\max p_i$ is the height of the first bin in the histogram of probability density distribution, which is then applied to equation (2.4). Inaccuracies can be caused by after-pulses and twilight pulses since they make the first bin in the distribution higher than expected. We can see in Fig. 3.5 that the measured entropy (red points) approximatively corresponds to theory (blue curve). Min-entropy is a measure of generated balanced bits per detection. The min-entropy decreases with increasing rate, which corresponds to the results of the discussion about the balance of bits in previous section. In Tab. 3.6, there are results of min-entropy for the measured rates rounded to integers.

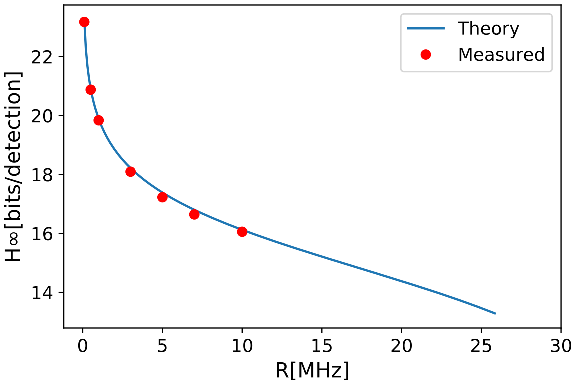


Figure 3.5: The dependence of min-entropy H_∞ on rate R.

R [Hz]	H_∞ [bits per detection]
100 k	23
500 k	21
1 M	20
3 M	18
5 M	17
7 M	17
10M	16

Table 3.6: Measured min-entropy H_∞ for different rates R.

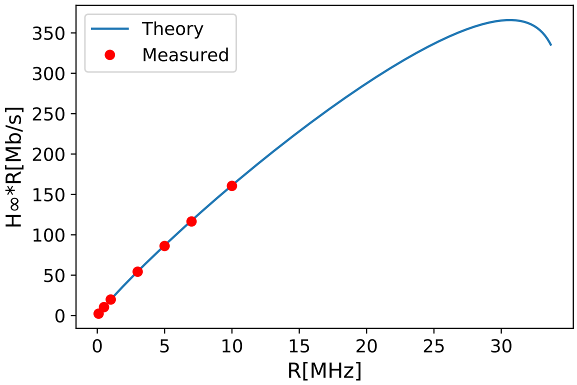


Figure 3.6: The dependence of min-entropy H_∞ multiplied by rate R on rate R.

The optimum rate can be found as the maximum of the function obtained by multiplying the min-entropy by saturated rate (2.15). This function then will transform the measure of bits per detection to bits per second, since there are $R_{detected}$ detection events per second on average.

As we can see in Fig. 3.6, the maximum is at 29.3 MHz, which is lower that the guaranteed light count rate of 31 MHz in Tab. 3.1. This rate is unfortunately unreachable because of the maximum transfer rate of the time tagger. If the detector maximum continuous rate suggested by the manufacturer is possible, we would obtain 352 Mb/s bits per second. For the maximum achievable rate of 10 MHz, we obtained 160 Mb/s.

Finally, random numbers for rates 100 kHz, 500 kHz, 1 MHz, 3 MHz, 5 MHz, 7 MHz and 10 MHz were generated. One way was dividing the probability distribution into 16 quantiles and another way was cutting the binary representation of the time differences at the value of min-entropy for that rate. That makes it 14 files in total. Then, the Von Neumenn extractor was applied. All of the files passed the Dieharder battery of tests. There was no difference between the results of the files where the extractor was applied and where it was not. It is highly likely that the tests are not sensitive enough to tell the difference or the quality of our QRNG is very high.

Chapter 4

Conclusion

The aim of the thesis was to demonstrate and compare methods of random number generation using quantum generators based on photonic detectors. Particularly, I used arrival times of individual photons as a random variable. I performed measurements with single-photon avalanche diode and analyzed the generated random bit sequences.

First, the employed methods were discussed. In Section 2.1, randomness and binary random numbers were defined together with complexity. The balance of bits was introduced in the form of probability of bit i beind sampled as 1. Moreover, Renýi family of entropies was described. Then, in Section 2.2, it was shown that the Poisson statistics of time intervals between adjacent detections is negative exponential. The Section 2.3 discussed single-photon detection, its properties affecting the detection quality. It was shown that the dead time of the detector affects the probability distribution and the balance of bits. Furthermore, the means of single-photon detection were described with a focus on SPAD and SNSPD detectors. Extractors and hash functions were introduced in Section 2.4. The randomness extractor invented by Von Neumann was described. Then, in Section 2.5, it was shown how the statistical randomness tests are designed.

Results and experiment description were provided in Chapter 3. In Section 3.1, experiment scheme and component properties were presented. Measurement and data acquisition were also described together with data processing. Tunability and stability were discussed. The transfer rate of the time tagger turned out to be the biggest limitation. It was possible to achieve only 10 MHz rate. The detector saturation, which is caused by its dead time, would result in rate limitation to 34.8 MHz. In Section 3.2, the resulting frequency and probability distributions for measured rate 3 MHz were presented and visible effects of detection properties were discussed. Together with 3 MHz, distributions for rates 100 kHz, 500 kHz, 1 MHz, 5 MHz, 7 MHz and 10 MHz were measured and plotted with a negative exponential fit. The percentage of detection events that are above the fit represent the percentage of after-pulses and twilight pulses. Standard deviation of fitting parameters are also presented. It is shown, that the standard deviation increases with increasing rate, which indicates that the distribution is more affected by detection imperfections at higher rates.

The corresponding balance of bits is presented in Section 3.3. The dead time of 28.7 ns resulted in oscillations that became more pronounced with increasing rate. Minor inaccuracies were caused by after-pulsing and twilight pulses. Otherwise, the experimental results are in good agreement with the theoretical models. The min-entropy was estimated in Section 3.3 and used to determine the number of balanced bits per detection. The entropy obtained from the data agreed with the theoretical estimate. Based on entropy, it was possible to determine the speed of random bit generation. For rate 10 MHz, the maximum speed of 160 Mb/s was achieved. The results of statistical testing are presented at the end of Section 3.3. Every generated file of random numbers passed the Dieharder battery of tests.

A time tagger with higher transfer rate could be used to improve the generator in the future. Among other things, we would reach the maximum random bit rate and we could verify the validity of the model over the full dynamic range of the detector. Post-processing options should also be further explored [50], because Von Neumann extractor is rather simple mean of randomness extraction. There are also different ways to test randomness than just statistical testing [56, 57]. These options should be examined too because statistical testing is not always efficient and sensitive enough. Apart from the time arrival method presented in this thesis, there are other methods to generate random numbers and some of them are more efficient. It is possible to circumvent some limitations by choosing homodyne detection [16].

Bibliography

[1] M. Stipcevic, "Quantum random number generators and their applications in cryptography," in *Advanced Photon Counting Techniques VI*, SPIE, 2012.

[2] M. Stipčević and Ç. K. Koç, "True random number generators," in *Open Problems in Mathematics and Computational Science*, pp. 275–315, Springer International Publishing, 2014.

[3] J. E. Gentle, *Random number generation and Monte Carlo methods*. Springer, 2003.

[4] N. Turner, "Randomness, does it matter?," *Journal of Gambling Issues*, no. 2, 2000.

[5] J. von Neumann, "Various techniques used in connection with random digits," vol. 12, pp. 36–38, 1951.

[6] P. Stoev and S. Stoilova, "Pseudo-random properties of a linear congruential generator investigated by b-adic diaphony," Author(s), 2017.

[7] K. Noel, "Analysis of random generators in monte carlo simulation: Mersenne twister and sobol," *SSRN Electronic Journal*, 2016.

[8] M. Herrero-Collantes and J. C. Garcia-Escartin, "Quantum random number generators," *Reviews of Modern Physics*, vol. 89, no. 1, 2017.

[9] E. Barker, "Recommendation for key management part 1: General," tech. rep., 2016.

[10] B. Ray and A. Milenkovic, "True random number generation using read noise of flash memory cells," *IEEE Transactions on Electron Devices*, vol. 65, no. 3, pp. 963–969, 2018.

[11] E. N. Allini, M. Skórski, O. Petura, F. Bernard, M. Laban, and V. Fischer, "Evaluation and monitoring of free running oscillators serving as source of randomness," *IACR Transactions on Cryptographic Hardware and Embedded Systems*, pp. 214–242, 2018.

[12] L. Gong, J. Zhang, H. Liu, L. Sang, and Y. Wang, "True random number generators using electrical noise," *IEEE Access*, vol. 7, pp. 125796–125805, 2019.

[13] J. J. Sakurai and J. Napolitano, *Modern quantum mechanics.* Pearson India Education Services, 2018.

[14] X. Ma, X. Yuan, Z. Cao, B. Qi, and Z. Zhang, "Quantum random number generation," *npj Quantum Information*, vol. 2, no. 1, 2016.

[15] J. E. Jacak, W. A. Jacak, W. A. Donderowicz, and L. Jacak, "Quantum random number generators with entanglement for public randomness testing," *Scientific Reports*, vol. 10, no. 1, 2020.

[16] T. Gehring, C. Lupo, A. Kordts, D. S. Nikolic, N. Jain, T. Rydberg, T. B. Pedersen, S. Pirandola, and U. L. Andersen, "Homodyne-based quantum random number generator at 2.9 gbps secure against quantum side-information," *Nature Communications*, vol. 12, no. 1, 2021.

[17] Y. Shi, B. Chng, and C. Kurtsiefer, "Random numbers from vacuum fluctuations," *Applied Physics Letters*, vol. 109, no. 4, p. 041101, 2016.

[18] Y. Shen, L. Tian, and H. Zou, "Practical quantum random number generator based on measuring the shot noise of vacuum states," *Physical Review A*, vol. 81, no. 6, 2010.

[19] R. Shakhovoy, D. Sych, V. Sharoglazova, A. Udaltsov, A. Fedorov, and Y. Kurochkin, "Quantum noise extraction from the interference of laser pulses in an optical quantum random number generator," *Optics Express*, vol. 28, no. 5, p. 6209, 2020.

[20] T. Gehring, C. Lupo, A. Kordts, D. S. Nikolic, N. Jain, T. Rydberg, T. B. Pedersen, S. Pirandola, and U. L. Andersen, "Ultra-fast real-time quantum random number generator with correlated measurement outcomes and rigorous security certification," 2018.

[21] F. Raffaelli, G. Ferranti, D. H. Mahler, P. Sibson, J. E. Kennard, A. Santamato, G. Sinclair, D. Bonneau, M. G. Thompson, and J. C. F. Matthews, "A homodyne detector integrated onto a photonic chip for measuring quantum states and generating random numbers," *Quantum Science and Technology*, vol. 3, no. 2, p. 025003, 2018.

[22] L. Huang and H. Zhou, "Integrated gbps quantum random number generator with real-time extraction based on homodyne detection," *Journal of the Optical Society of America B*, vol. 36, no. 3, p. B130, 2019.

[23] F. Xu, B. Qi, X. Ma, H. Xu, H. Zheng, and H.-K. Lo, "Ultrafast quantum random number generation based on quantum phase fluctuations," *Optics Express*, vol. 20, no. 11, p. 12366, 2012.

[24] Y.-Q. Nie, L. Huang, Y. Liu, F. Payne, J. Zhang, and J.-W. Pan, "The generation of 68 Gbps quantum random number by measuring laser phase fluctuations," *Review of Scientific Instruments*, vol. 86, no. 6, p. 063105, 2015.

[25] M. Stipčević and J. E. Bowers, "Spatio-temporal optical random number generator," *Optics Express*, vol. 23, no. 9, p. 11619, 2015.

[26] E. R. Jeffrey, *Advanced quantum communication systems.* PhD thesis.

[27] M. A. Wayne, E. R. Jeffrey, G. M. Akselrod, and P. G. Kwiat, "Photon arrival time quantum random number generation," *Journal of Modern Optics*, vol. 56, no. 4, pp. 516–522, 2009.

[28] M. Wahl, M. Leifgen, M. Berlin, T. Röhlicke, H.-J. Rahn, and O. Benson, "An ultrafast quantum random number generator with provably bounded output bias based on photon arrival time measurements," *Applied Physics Letters*, vol. 98, no. 17, p. 171105, 2011.

[29] M. Ren, E. Wu, Y. Liang, Y. Jian, G. Wu, and H. Zeng, "Quantum random-number generator based on a photon-number-resolving detector," *Physical Review A*, vol. 83, no. 2, 2011.

[30] M. J. Collins, A. S. Clark, C. Xiong, E. Mägi, M. J. Steel, and B. J. Eggleton, "Random number generation from spontaneous Raman scattering," *Applied Physics Letters*, vol. 107, no. 14, p. 141112, 2015.

[31] W. Wei and H. Guo, "Bias-free true random-number generator," *Optics Letters*, vol. 34, no. 12, p. 1876, 2009.

[32] R. H. Hadfield, "Single-photon detectors for optical quantum information applications," *Nature Photonics*, vol. 3, no. 12, pp. 696–705, 2009.

[33] L. Neri, S. Tudisco, F. Musumeci, A. Scordino, G. Fallica, M. Mazzillo, and M. Zimbone, "Dead time of single photon avalanche diodes," *Nuclear Physics B - Proceedings Supplements*, vol. 215, no. 1, pp. 291–293, 2011.

[34] A. W. Ziarkash, S. K. Joshi, M. Stipčević, and R. Ursin, "Comparative study of afterpulsing behavior and models in single photon counting avalanche photo diode detectors," *Scientific Reports*, vol. 8, no. 1, 2018.

[35] M. Hofbauer, B. Steindl, and H. Zimmermann, "Temperature dependence of dark count rate and after pulsing of a single-photon avalanche diode with an integrated active quenching circuit in 0.35 μm CMOS," *Journal of Sensors*, vol. 2018, pp. 1–7, 2018.

[36] X. Ma, F. Xu, H. Xu, X. Tan, B. Qi, and H.-K. Lo, "Postprocessing for quantum random-number generators: Entropy evaluation and randomness extraction," *Physical Review A*, vol. 87, June 2013.

[37] K. Kim, S. Bittner, Y. Zeng, S. Guazzotti, O. Hess, Q. J. Wang, and H. Cao, "Massively parallel ultrafast random bit generation with a chip-scale laser," *Science*, vol. 371, no. 6532, pp. 948–952, 2021.

[38] D. Eastlake, J. Schiller, and S. Crocker, "Randomness requirements for security," tech. rep., 2005.

[39] M. G. Kovalsky, A. A. Hnilo, and M. B. Agüero, "Kolmogorov complexity of sequences of random numbers generated in Bell's experiments," *Physical Review A*, vol. 98, no. 4, 2018.

[40] B. W. Lindgren, *Statistical theory.* Chapman & Hall, 1998.

[41] Z. Wu, “The information hiding model for speech secure communication,” in *Information Hiding in Speech Signals for Secure Communication*, pp. 27–40, Elsevier, 2015.

[42] P. Jizba and T. Arimitsu, “The world according to Rényi: thermodynamics of multifractal systems,” *Annals of Physics*, vol. 312, no. 1, pp. 17–59, 2004.

[43] C. E. Shannon, “A mathematical theory of communication,” *Bell System Technical Journal*, vol. 27, no. 3, pp. 379–423, 1948.

[44] R. Loudon and M. O. Scully, “The quantum theory of light,” *Physics Today*, vol. 27, no. 8, pp. 48–48, 1974.

[45] C. J. Chunnilall, I. P. Degiovanni, S. Kück, I. Müller, and A. G. Sinclair, “Metrology of single-photon sources and detectors: a review,” *Optical Engineering*, vol. 53, no. 8, p. 081910, 2014.

[46] G. S. Buller and R. J. Collins, “Single-photon generation and detection,” *Measurement Science and Technology*, vol. 21, no. 1, p. 012002, 2009.

[47] A. Migdall, S. Polyakov, J. Fan, and J. Bienfang, *Single-photon generation and detection: experimental methods in the physical sciences.* Elsevier/AP, Academic Press is an imprint of Elsevier, 2013.

[48] Q.-Y. Zhao, D. Zhu, N. Calandri, A. E. Dane, A. N. McCaughan, F. Bellei, H.-Z. Wang, D. F. Santavicca, and K. K. Berggren, “Erratum: Corrigendum: Single-photon imager based on a superconducting nanowire delay line,” *Nature Photonics*, vol. 11, no. 9, pp. 608–608, 2017.

[49] M. de Cea, E. E. Wollman, A. H. Atabaki, D. J. Gray, M. D. Shaw, and R. J. Ram, “Photonic readout of superconducting nanowire single photon counting detectors,” *Scientific Reports*, vol. 10, no. 1, 2020.

[50] X. Ma, F. Xu, H. Xu, X. Tan, B. Qi, and H.-K. Lo, “Postprocessing for quantum random-number generators: Entropy evaluation and randomness extraction,” *Physical Review A*, vol. 87, no. 6, 2013.

[51] L. Trevisan, “Extractors and pseudorandom generators,” *Journal of the ACM*, vol. 48, no. 4, pp. 860–879, 2001.

[52] E. W. Weisstein, “Hash function.” https://mathworld.wolfram.com/HashFunction.html, accessed 2021-04-14.

[53] J. Zbigniew, “Perfet hashing.” http://www.burtleburtle.net/bob/hash/perfect.html, accessed 2021-04-14.

[54] A. L. Rukhin, *A statistical test suite for random and pseudorandom number generators for cryptographic applications.* U.S. Dept. of Commerce, Technology Administration, National Institute of Standards and Technology, 2001.

[55] J. Shah, “Dieharder test suite,” 05 2019.

[56] P. Gács, “Uniform test of algorithmic randomness over a general space,” *Theoretical Computer Science*, vol. 341, no. 1-3, pp. 91–137, 2005.

[57] L. Bienvenu, P. Gács, M. Hoyrup, C. Rojas, and A. Shen, "Algorithmic tests and randomness with respect to a class of measures," *Proceedings of the Steklov Institute of Mathematics*, vol. 274, no. 1, pp. 34–89, 2011.