OPTICAL QUANTUM RANDOM NUMBER GENERATION: APPLICATIONS OF SINGLE-PHOTON EVENT TIMING

BY

MICHAEL A. WAYNE

DISSERTATION

Submitted in partial fulfillment of the requirements
for the degree of Doctor of Philosophy in Electrical and Computer Engineering
in the Graduate College of the
University of Illinois at Urbana-Champaign, 2017

Urbana, Illinois

Doctoral Committee:

Professor Paul G. Kwiat, Chair
Professor P. Scott Carney
Associate Professor John M. Dallesasse
Professor Kyekyoon Kim

Abstract

This dissertation is the result of research which, although electrical and computer engineering in nature, also aims to improve the performance of many systems in the field of quantum information. For example, random number generators are used in almost all areas of science, and the initial portion of this work details the theory, design, and characterization of two photon-arrival-time quantum random number generators (QRNGs). After the QRNGs were completed, it was realized that their performance was severely limited both by the maximum detection rate of the single-photon detectors used, and the precision at which the arrival times could be resolved.

The single-photon detectors used for both QRNGs are single-photon avalanche photodiodes (SPADs), devices which when operated below their breakdown voltage can create a macroscopic amount of current (an avalanche) in response to a single incident photon. Some of this charge can become trapped in defects or impurities; if this trapped charge is released when the SPAD is active, a secondary 'false' detection event, or 'afterpulse' can occur. To lower the afterpulse probability to reasonable levels (< 1%), we attempted to reduce the amount of avalanche charge by halting its growth promptly with high-speed electronics, so that defects have a lower probability of becoming populated in the first place. Initial results show reductions in afterpulse probability by up to a factor of 12, corresponding to a ~20% decrease in dead time, a value that could be improved further.

We developed an FPGA-based time-to-digital converter system for use specifically with SPADs, achieving a time-bin resolution of 100 ps, with lower dead time and higher maximum detection rate than all currently available detection systems. This further allowed for the creation of a new higher-order SPAD characterization technique, which was identified previously unknown subtleties to SPAD operation.

Finally, we developed an ultra-low-latency QRNG, which was used in one of the recent loophole-free demonstrations of quantum nonlocality. The final latency was below 2.5 ns, to our knowledge the lowest latency QRNG to date. Of special interest, however, is our subsequent exploration into the characterization of its bit-probability drift using atomic clock stability techniques. By employing the Allan deviation and implementing precision feedback, the additional frequency drift caused by environmental fluctuations is reduced such that the resulting bit stream can pass cryptographic random number tests for sample sizes up to 5 Gb. This system is currently intended for the NIST random-number beacon, a world-wide trusted source of random bits.

Acknowledgments

I have struggled with how to write this section, mostly because I can't find a clear place to begin. Detailing the circumstances surrounding this journey would take many pages, and since portions of it remain unbelievable to even myself, I wouldn't expect many others to believe it either. However, there are several people who undeniably deserve to be mentioned.

I would like to profoundly thank Professor Narayana Rao for going against his friends, colleagues, and every listed rule in the ECE admissions process and giving me a chance; there is no doubt in my mind I would not be here today if it were not for you. I hope someday to be able to give that chance to others. To Professor Paul Kwiat, your patience with me has been greatly appreciated. Many students are much easier to handle I'm sure, but I am still in awe when I think about how many times you petitioned, appeared before committees, and somehow kept me afloat. You also let me do all my research in Maryland, something I've never heard of before, under Dr. Joshua Bienfang at NIST. There is no one else I would have rather occupied our rooftop lab with, and your tutelage is the only reason I consider myself an effective future Doctor of Electrical and Computer Engineering.

I am also incredibly lucky to have such an amazing family. My parents Robert and Leslie, and my brother David, have unconditionally supported me throughout all of this. Although I'm the only one with a PhD, you are all so much smarter than me, and I look forward to seeing what else we can do together. My super-talented girlfriend Fiorella has also offered endless patience and support, and has my deepest thanks. I'd additionally like to mention Donna Arnold for her help in my undergraduate degree, Mr. Akmal, Mr. Pharr, and Mr. McIntyre for being especially influential teachers in my early life, Rhea Kressman for assisting me on how to approach my anxiety, my research group for putting up with my funny noises, and all my other friends and family who helped me along the way.

Finally, I'd like to give some advice. There are many times, both in research and life, where there seems to be no path forward. The situation either seems hopeless, or the solution appears impossible. In science, I find that the most amount of information can be gained by doing something which you believe to be experimentally wrong. In life, I found this has roughly translated to being stubborn, walking an unusual path, and never, ever, ever giving up.

Contents

Chapter 1 — Introduction

1.1 Motivations

The field of quantum information involves the investigation of systems which exploit quantum mechanical processes to perform various tasks, often related to information processing (quantum computing [1]) or communications (quantum cryptography [2], teleportation [3], super-dense coding [4], etc.). While relying on quantum phenomena, these systems often require classical channels in order to measure or record their results. This demand is usually met by accompanying electrical or computer-based systems, which to be useful in modern experiments need to operate at rates near the cutting edge of current technology. The focus of this dissertation is the theory, modeling, and construction of several of these devices, as well as the design techniques that make them possible. The end result is a suite of advanced tools that facilitate further progress in quantum information processing.

Quantum computing aims to use fundamental quantum phenomena such as superposition and entanglement to solve select classes of problems faster than by classical methods. For example, Shor's quantum factoring algorithm has been shown to be able to reduce the computational complexity for factoring large numbers from sub-exponential to polynomial time [5]. The computational unit of a classical computer is a bit, a logical value stored by a digital device that exists in one of two possible states, '0' or '1'. A quantum bit (qubit), is the quantum analogue of a classical bit, and is represented by a two-level quantum system such as the polarization of a photon or the spin state of an electron. A fundamental property of qubits is that they can exist not only in one of the two binary levels, but also in a superposition of both. This is of special interest (or concern) in modern-day cryptographic applications, where products of very large prime numbers are used to encrypt messages [6]. If a quantum computer could operate with a large number of qubits, it could be used to break public-key cryptography protocols, such as RSA. Due to experimental limitations, however, Shor's algorithm has thus far been successfully used to factor 15, using 4 qubits [7], although adiabatic quantum computation was used to factor 143 [8], and mathematical tricks were used to factor numbers as large as 56153 with only four qubits [9].

Quantum communication includes such applications as quantum key distribution (QKD), where the security of a communication channel is guaranteed by the fundamental laws of physics [10]. Most QKD protocols involve two parties, referred to as Alice and Bob, with Eve being a possible eavesdropper. In the most common protocol, BB84, Alice sends single photons to Bob in one of four states of polarization (e.g., horizontal and diagonal represent a “0”, vertical and anti-diagonal a “1”), who then measures them in one of two bases. Alice and Bob then share the basis that they used for each transmitted photon. In the approximately half of the cases where they used the same basis, Alice and Bob will then have matching bit strings. Moreover, because of the no-cloning property of quantum information, it is impossible to copy an unknown quantum state without modifying the state of both the original and the duplicate. Therefore, in the presence of Eve, errors will be introduced into Alice and Bob’s key string, and the channel will be known to be insecure. While mostly used in laboratory settings, QKD has been performed over free-space distances of over 140 kilometers [11], and there are now commercially available systems.

Many quantum information systems have direct links to security applications, and therefore also require the generation of random numbers. The choice of polarization states in QKD, for example, needs to be made randomly in order to prohibit Eve from correctly guessing which was used. A quantum random number generator (QRNG) fulfills that need, producing random bits whose entropy is guaranteed not by complexity, but through the fundamental laws of quantum physics. Many other applications such as gambling, Monte Carlo simulations, and statistical sampling also employ RNGs, although not usually of the quantum variety. Initially limited to rates in the kHz range, QRNGs were not feasible for most purposes, and so software-based approximations were used. However, advances have increased QRNG operational rates to the GHz range in recent years [12]–[14], making them suitable for previously inaccessible applications.

The understanding of quantum mechanical systems gained by quantum information research has been used to explore a broad range of areas, such as the interaction of quantum states with biological systems, and even furthering the fundamental understanding of our natural world. The improvements of reliable photon sources have allowed researchers to investigate how many photons the human eye can sense [15], while enhancements to several quantum information technologies – sources, detectors, and QRNGs – were used collectively to perform the first loophole-free tests of nonlocality, known as Bell's inequalities [16]–[18]. This put to rest a half century of debate, and finally ruled out the possibility of local realistic models governing our physical reality.

1.2 Quantum Random Number Generators

Random numbers have a variety of uses (cryptography, gambling, computer simulations), and a variety of ways in which they can be produced. These methods are often grouped into two categories: pseudo-random number generators (PRNGs) and "true"-random number generators (TRNGs). PRNGs are based on mathematical algorithms which approximate the behavior of randomness, but they can suffer from a variety of problems (determinism due to the initial seed, periodicity, correlations, lack of uniformity, etc.). In contrast, true random number generators use the actual behavior of a physical process, which is chosen to produce a random output, to do the same. Unfortunately, often one must account for biases in the measurement process, such as temperature fluctuations or component aging. Due to the possibility of such a high number of these biases being present in any physical system, it is hard to place substantial trust in a TRNG actually being "truly" random. Therefore, it is often the case that an RNG, of any type, is designed to be only random enough for the application involved. This entails bounding the difficulty of exploiting a weakness in the system by the time in which such an exploit would be relevant (e.g., if it would take 2^{20} years to break a cryptographic hash function, it is probably not going to be relevant after that period).

Be that as it may, there are situations where a certifiably random number is absolutely essential, and in these cases it is essential that the source of randomness be trusted, characterized, and bounded to the best of our ability. A likely candidate for such a random source is quantum mechanics, where instead of the classical notion of a state, the possible output value of a system is characterized as a probability distribution on the set of outcomes of measurements of an observable. One example of such a system is the arrival time of a single photon, created by a laser operating well above threshold, at a point in space occupied by a detector. A short history of several such systems, or quantum random number generators (QRNGs), is the subject of Chapter 2. The design of two QRNGs, based on the randomness present in photon arrival times, is the subject of Chapter 3.

Upon construction of the first QRNG presented in Chapter 3, it was discovered that the system's performance could be greatly enhanced through improvements to various components, such as single-photon detectors and time-resolution measurement. By developing enhancements to these technologies, higher random number generation rates were achieved, and at one point were the highest achieved anywhere in the world [12], [13].

There are situations in which a truly random number is absolutely essential, one such being a loophole-free test of nonlocality [16]–[18]. Requiring two RNG systems, this experiment has the additional constraint in that it must be impossible for information from one system to have reached (and possibly influenced) the other RNG system before the measurement is finalized, if that information was traveling at the speed of light. To achieve this, a *low-latency* random number generator (LLRNG), and the subject of Chapter 6, was designed, in which the random process cannot even begin before the request of a random bit, and the whole process must have been completed before information could have reached the other LLRNG system (on the order of tens of nanoseconds). This short time window also includes every other process in the experimental measurement, so the random number generation has to be as prompt as possible. In the case of our system, the total time to generate a random bit, from request to output, was approximately 2.4 ns.

This system was extremely prone to environmental fluctuations, and great care had to be taken to temperature-stabilize the system. However, even after extensive efforts, the output bit proportion (ideally 50% one and 50% zero) would slowly fluctuate. This led to a novel characterization technique, the evaluation of the RNGs performance by an atomic clock frequency stability metric, the Allan deviation. By applying several such methods, the timescale and strength of the noise affecting the RNG was characterized. The exploration into several active feedback techniques is presented; however, work remains to be done on finding an optimal technique. Even so, this system is planned to be implemented into NIST's Random Beacon, a national random number service which provides a trusted source of entropy to users around the world.

1.3 Time-tagging

During the process of measuring the performance of the QRNGs, two major challenges were identified. First, the arrival time of the photon was registered by a "time-tagger", a device in which the arrival time of each electrical pulse output by the single-photon detector was measured to a certain degree of precision, initially 5 ns. Each time interval is referred to as a "time-bin", and starting at time-bin 0 immediately after an initial detection, the next detection event is assigned a value corresponding to however many time-bins have subsequently passed. Since this output number is the random value assigned to each photon, the more time-bins that can be squeezed into the inter-photon time-interval, the more bits that can generated per detection event.[1]

For this purpose, we developed several other time-tagging systems using field-programmable gate-arrays, or FPGAs. Utilizing the compact logic inside these chips, systems with time-bin resolution of 100 ps were developed, recording streaming data at rates of up to 400 million events per second. It was intended to combine multiple time-tagger channels and enable the option for timing resolution as low as 25 ps, paving the way for higher random number generation rates as well as more precise timing characterization of

[1] The time-bin resolution is ultimately limited by the timing resolution of the SPAD.

several other components in the overall QRNG system. Although the improved system was designed and built, a critical portion failed and thus was not completed for this work. However, the 100 ps time-tagging system, as well as the intended improvements to the hardware, is detailed in Chapter 5.

1.4 SPAD Afterpulse Reduction

The second limitation of the QRNG in Chapter 3 was that the final random number generation rate was heavily constrained by the rate at which the single-photon detector could operate. We had chosen to use single-photon avalanche diodes (SPADs), in which a single incident photon creates, through avalanche multiplication, a macroscopic charge. While these devices are convenient for laboratory use (rugged, relatively inexpensive, and requiring no cryogenic cooling), they suffer from two negative attributes which directly affect the performance of a QRNG: afterpulsing and dead time.

When a photon is absorbed by the detector, a large amount of charge is created by impact ionization and swept across the device by a strong electric field. The current quickly saturates the diode, and in the following nanoseconds, avalanche charge is held in traps within the device (due to impurities, defects, dangling bonds, etc.). This charge can later be re-released, and then has a probability of producing a second "false" avalanche, or afterpulse. There is no way to distinguish these events from original photonic events, and so reducing the probability of afterpulses is critical to our quantum random number generation scheme.[2] The most common way to do so is by imposing a dead time, often in the range of 25-50 ns, in which the SPAD is held in such a state that released avalanche charge does not trigger a spurious detection. This can reduce the afterpulsing probability to well below 1% for silicon-based devices, but has the detrimental effect of also severely limiting the detector saturation rate and, thus, in the context of QRNGs, our random number generation rate. Chapter 4 details the proposed and realized method of reducing the deadtime required, by curtailing the flow of avalanche current as soon as possible, through the use of high-speed electronics and PCB-layout techniques. By using the time-tagger system outlined in Chapter 5, precise characterization of the time-constants of the various trap levels was obtained, and through modeling of the system, an optimal dead-time period can be determined, maximizing the final random number generation rate. Additionally, the time-tagger performance characteristics resulted in the creation of a new SPAD characterization method. Now being prepared for

[2] Although presented in the context of QRNGs, afterpulsing and dead time are undesirable for most, if not all, applications using photon counters.

publication, this method is able to reveal previously undiscovered subtleties in SPAD operation, and is detailed in Section 5.5.8.

1.5 What to Expect

The research presented here was undertaken with physics-based quantum information goals in mind, and I have elaborated when necessary. However, it was also done in pursuit of a degree in Electrical and Computer Engineering, sometimes requiring advanced printed circuit board layout techniques, electrical circuit design, and other more "ECE-related" skills to accomplish said goals. Through the course of this research I have discovered that experimental physics bleeds heavily into the realm of electrical and computer engineering; e.g., knowledge of semiconductor physics has helped immensely in understanding avalanche photodiodes, and the ability to design high-speed electrical circuits has proved to be invaluable in characterizing these devices. Therefore, care has been taken to present the material with both disciplines in mind.

Chapter 2 — Quantum Random Number Generators

2.1 A Brief History

Random number generators (RNGs) are often thought to be modern day inventions, but in one form or another they have existed for much longer. Approximately 7000 years ago, religious shamans used marked objects such as fruit pits, seashells, bones, or even animal entrails to tell the future by interpreting signs from the gods. These objects, *astragali,* made their way to the layman, and may have been used to determine which hunter would go home with the best cut of meat. The bones were generally rounded or only two-sided, but eventually the rounded sides of the astragali were carved down to make them more cube-like. Thus dice were born, with their first known appearance dating back to 3000 B.C, with the discovery of the Royal Game of Ur in a Mesopotamian tomb. Early forms of roulette soon followed, with soldiers spinning marked shields (Greeks) or chariot wheels (Romans) on top of spears, and placing bets on which section the wheel would stop. Even the drawing of marked characters can be considered an RNG, with the famous Chinese general Cheung Leung using Keno to fund the army's battles and the construction of the Great Wall of China [19].

2.1.1 Pseudo-random Number Generators

"Anyone who considers arithmetical methods of producing random digits is, of course, in a state of sin."

– John von Neumann

With the age of computing came a revolution in the way random numbers were generated. Instead of using an unpredictable rudimentary physical process, RNGs began to harness the newly available computational power and shifted into the production of randomness through mathematical algorithms. These algorithms are able to produce very long strings of data that *appear* to be random but are in fact completely deterministic and chosen by an initial state, or seed. Consequently, RNGs of this type are referred to as pseudo-random number generators, or PRNGs.

The middle-square method [20] and linear congruential generator [21] were introduced in 1949, and are considered to be the first PRNGs. The middle-square method was invented by John von Neumann, a Hungarian-American scientist who was responsible for notable advances in mathematics, physics, computing, statistics, and several other fields. The algorithm consisted of several steps: creating an initial series of four-digit numbers, squaring them, and performing basic mathematical functions on the middle four digits of the result. Because it suffers from extreme periodicity and tends to converge to zero, the middle-square method is rarely used today. Even so, von Neumann and Polish mathematician Stanislaw Ulam expanded this work as part of the Manhattan Project. On one hand, the method led to the creation of the hydrogen bomb; on the other, it also led to the creation of the Monte Carlo method, one of the most widely used techniques for simulating complex systems using random numbers.

$$X_{n+1} = (aX_n + c) \ mod \ m \qquad \text{Eqn. (2.1)}$$

The linear congruential generator (LCG) was invented by American mathematician Derrick Lehmer, and its variations make up a significant fraction of modern-day PRNGs. As shown in Equation 2.1, the LCG operates by performing recursive modulo arithmetic based on a seed (X_0), multiplier (a), increment (c), and modulus (m). As shown in Figure 2.1, if poor choices are made for the internal variables, the LCG can suffer from predictable periodicity. By choosing powers of two as the modulus and large prime numbers as the multipliers and increments, an LCG can be a computationally and memory-efficient choice for a general-use PRNG; however, it is typically not considered suitable for cryptographic applications.

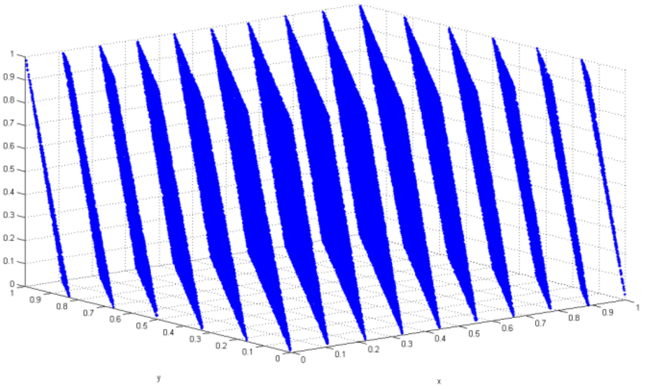


Figure 2.1. *Three-dimensional plot of the output of IBMs RANDU, one of the most popular PRNGs of the 1960s. All triplets of points {x_1, x_2, x_3} generated by RANDU follow the equation $x_3 = 6x_2 – 9x_1$. Several other popular LCGs also suffered from similar periodicity, including APPLE, ANSIC, and SIMSCRIPT [22].* Figure courtesy of *[23]*.

Another common type of PRNG is the feedback shift register (FSR), a function whose input bit is a linear function of its previous state. Figure 2.2 shows an example of a simple FSR, one whose input bit is determined by the logical XOR of some bits of the overall shift register value. The most popular PRNG in use today, the Mersenne Twister [24], is derived from a generalized FSR, although as its name implies it relies on many more complex operations and a large Mersenne Prime. Other commonly used PRNG types include stream ciphers, block ciphers, elliptical curve RNGs, and deterministic chaotic systems.

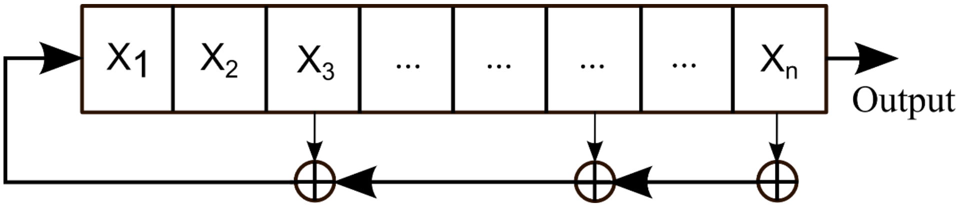


Figure 2.2. Example of simple feedback shift register. Output bit is a combination of the exclusive-or of multiple previous bits.

As techniques for developing PRNGs become more advanced, their security flaws become less apparent, but flaws have still been found. For example, the elliptical curve-based Dual_EC_DRBG PRNG was officially recognized by NIST until it was discovered (and confirmed in an Edward Snowden leak) that it contains an NSA-implanted backdoor [25]. There are cryptographically secure PRNGs available which can reliably pass random number generation test suites, but this does not make their output *random*, as the fact remains that they are inherently deterministic. For many cases this is sufficient, but some applications require *true* sources of randomness.

2.1.2 True Random Number Generators

"The generation of random numbers is too important to be left to chance." –Robert Coveyou

A true random number generator (TRNG) is a device whose entropy is derived from one or more unpredictable physical processes. Typically, this unpredictability can have one of two primary origins: complexity or actual randomness, usually associated with the uncertainty inherent to quantum mechanics. TRNGs of the first type are sometimes referred to as chaotic random number generators (CRNGs), and the second as quantum random number generators (QRNGs).

For a period after their emergence PRNGs were more efficient than their TRNG counterparts, requiring much less time to generate each random bit. The physical processes employed by early TRNG systems were limited to kilohertz frequencies [26], while PRNGs were in the MHz regime [27]. However, as the security flaws inherent in PRNGs became more noticeable, significant effort was put into designing physical systems with comparable performance.

Ignoring rudimentary examples such as dice, early TRNG systems were primarily electrical or radioactive in nature, with the first published radioactive system generating random bits at 1.2 b/s [28]. As optical physics and detection electronics advanced, it became more feasible to use these often faster processes for random number generation. Most optical-TRNGs can operate in the hundreds of MHz range [13], and recent systems have reached output rates of up to 68 Gbps [14], producing high-quality random bits which pass cryptographic test suites. This chapter presents a brief overview of select chaotic and quantum random number generation systems. For chaotic and non-optical QRNGs a small subset of the most interesting published works is summarized. As optical quantum random number generation is the focus of many of the later chapters, more detailed explanations are given of systems of this type.

2.2 Chaotic Random Number Generators

The entropy produced by a CRNG is generated from a physical process for which any small change in the system results in dramatically different results. In the case of lottery balls, for example, values are chosen after being tossed around in a turbulent cage. Ultimately obeying the predictable laws of motion, the entire process depends on the initial conditions of the system, which are assumed to be at least partially unknown. Without the totality of this information, predicting the final output grows exponentially harder as the process progresses, and quickly becomes technically infeasible.

2.2.1 Simple Examples

Dice, lottery balls, casino games, etc., are all examples of rudimentary CRNG systems, and are totally suitable for their intended use. When security is not a concern and short-term unpredictability is sufficient, it is common to use such devices or even to reuse random data. One of the most well-known examples is “A Million Random Digits with 100,000 Normal Deviates”, a book published by the RAND Corporation in 1955, and shown in Figure 2.3. Generated by an electronic roulette wheel, the book contained multiple tables of random sequences, which were used primarily for simulation or Monte Carlo purposes [29].

2.2.2 LavaRND

Designed in 2000, LavaRND is an open source, cryptographically secure CRNG based on the behavior of lava lamps.[3] A snapshot of the lamp is taken by a CCD camera, which converts 19,200 pixels into their respective Y (luminance) and UV (color) values. In the dark environment that the lava lamp is stored, the color values do not change much over time, but the Y luminance values serve as the chaotic entropy source, with an example data pattern shown in Figure 2.3. The Y data is run through an entropy extraction algorithm to separate the chaotic from non-chaotic data, and a SHA-1 hash function-based whitening

[3] An argument could be made that due to quantum shot noise in each individual semiconductor-based CCD pixel, that LavaRND could be classified as a QRNG, although the mechanism behind the shape of the ‘lava’ is primarily thermal.

process to improve the uniformity of the output bits. The LavaRND system produces bits at rates of 200 kb/s, has passed the NIST random number test suite for sample sizes of 1 Gb, and its instructions are openly available for any hobbyist to construct a system of their own [30].

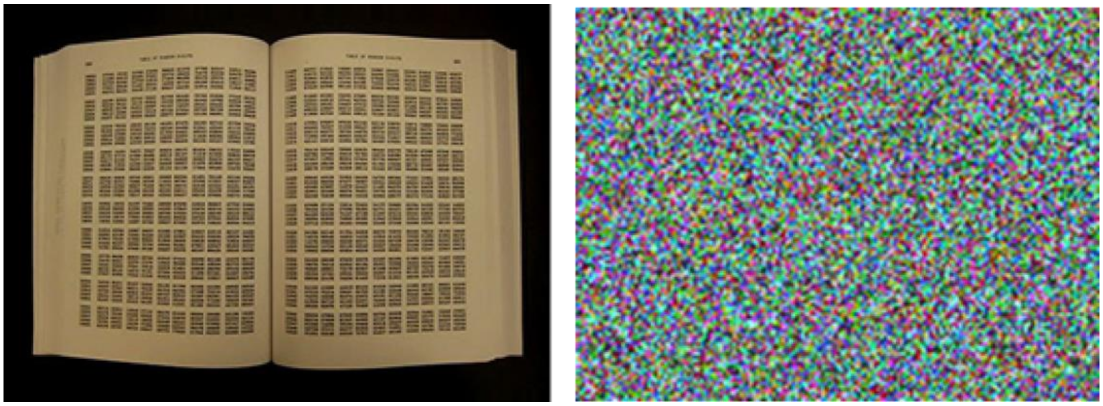


Figure 2.3. Excerpt from the book "A Million Random Digits with 100,000 normal deviates" (left), and sample data of 19,200 pixels taken from the LavaRND chaotic random number generator (right). Enclosed in a sealed can, the LavaRND CRNG generates its entropy from luminosity fluctuations of a lava lamp, while those in the book were generated from an electronic roulette wheel.

2.2.3 Atmospheric Noise

The website RANDOM.ORG is a service that generates true random numbers via atmospheric noise [31]. Readily measurable by a standard radio receiver, atmospheric noise is caused by natural processes, such as lightning discharges in thunderstorms. By digitizing and whitening the extracted chaotic data, this random service can produce high quality random bits at 3000 bits per second, per radio. However, the service itself notes that one must be careful to watch out for patterns when building a system of your own, as systematic sources (ceiling or computer fan) can introduce non-randomness. A recent paper published in 2014 [32] also used atmospheric turbulence to generate entropy, although here they specifically measured the quantum-based speckle of a coherent light source, so more information is presented in the QRNG section.

2.2.4 Clock Drift

Several popular microprocessors use internal CRNGs based on the measurement of clock drift as the source of randomness. In the Intel 82802, two independent clock crystals produce two asynchronous signals with different frequencies, one slow (≈100 Hz) and one faster (≈1 MHz). A parity check is then run on the number of fast clock events within one of the slow clock's period, and the result is whitened to remove bias. These types of CRNGs can produce several hundred bits per second, a rate sufficient for most of the microprocessors needs. Side-channel attacks have been discovered on clock-based CRNGs, such as [33], in which the crystal's frequency can be influenced by causing the CPU to overheat, and an attacker can partially control the clock skew.

2.2.5 Faster Noise-based CRNGs

The previous CRNG examples do not approach the MHz or above ranges typical of most PRNGs or modern QRNGs, and rates necessary for intensive modeling and simulation needs. A CRNG system which *can* meet these demands is one based on thermal noise, as shown in Figure 2.4. Discovered by John Johnson at Bell Labs in 1926, Johnson Noise is generated by the Brownian motion of electrons, and will cause the voltage between two terminals of a resistor to fluctuate. By amplifying this voltage, and then comparing its magnitude with a stable voltage reference, random bits can be generated at rates in the MHz regime. Although the power spectral density of Johnson Noise is approximately white, other components are introduced by the bandwidth of the amplifier or temperature fluctuations, so the final output is further processed.

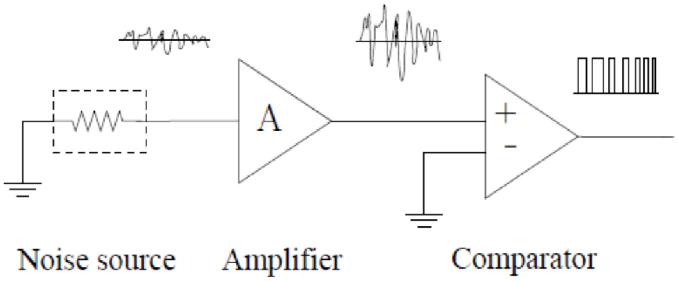


Figure 2.4. Conceptual representation of a Johnson-Noise based CRNG. The noise source, in this case the voltage across a resistor, produces a signal which is amplified and then compared against a DC value (in this case GND). The output bit is determined by which of the two terminals of the comparator is higher at a point in time *[34]*.

A photomultiplier tube (PMT) is a type of vacuum tube, and can be an extremely sensitive detector of light, capable of multiplying the current produced by the photoelectric effect by as much as 160 dB. As shown in Figure 2.5, a PMT is housed in a glass tube which contains a photocathode, multiple dynodes, and an anode. Incident photons which strike the photocathode produce electrons, which are ejected and directed towards the dynodes. Each dynode is held at a successively higher potential than the last, such that an exponentially increasing amount of electrons is produced at each stage. Finally, the large electronic signal reaches the anode, resulting in an easily detectable current pulse. The materials used for PMTs typically have a very low work function, and so randomly generated electrons have a non-zero probability of being ejected inside the tube without the presence of light. Although there exist single-photon PMTs which are used in various optical QRNGs, the dark current output by the PMT can also be used in a fashion similar to that of Figure 2.5 to produce thermal noise-based random numbers.

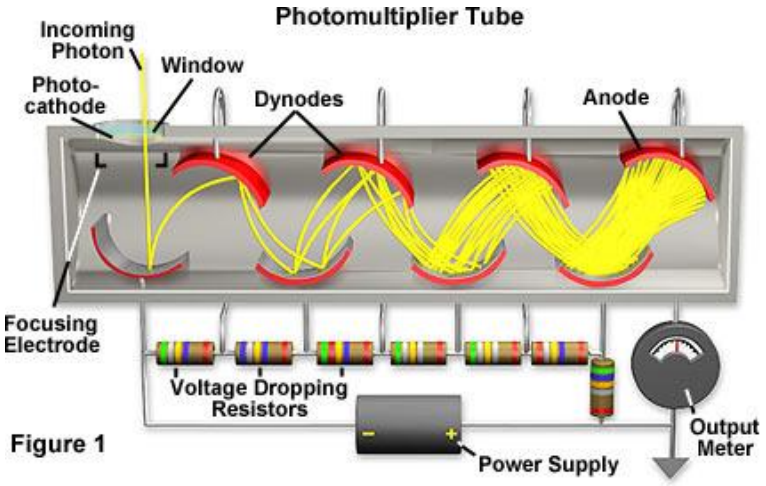


Figure 2.5 Typical internal structure of a photomultiplier tube (PMT). Incoming photons are absorbed by the photocathode and an electron can be ejected. Subsequent aligned dynodes steer the path of electrons in the multiplication region until they are finally output at the anode terminal. Figure courtesy of *[35]*.

Although systems based on analog noise can be very fast, their maximum generation rate is fundamentally limited by the bandwidth of the sampled noise. Sampling a signal at a rate significantly faster than its average rate of change will result in correlated successive bits. Therefore, the rate of acquisition must be slow enough such that any such correlations will be reduced to a negligible level, and further whitening is often necessary.

2.3 Non-Optical Quantum Random Number Generators

There exist quantum processes which, repeated many times with the exact same initial conditions, will give different results. If we operate under the assumption that the universe is governed by quantum mechanics, then it would follow that any true random number generator could be classified as quantum. However, the term QRNG is reserved here for those RNGs for which the entropy generated has been realized from primarily quantum mechanical effects. In practice this is difficult to achieve, as accompanying electronics and environmental influences create additional classical noise to be considered. In such cases it is prudent to accurately quantify and characterize possible sources of *non*-quantum bias or unpredictability, and their effects on the final random output.

2.3.1 Radioactive Decay QRNGs

The first published example of a QRNG system was in 1956 by Masatugu Isida and Hiroji Ikeda, and is based on the radioactive decay of Cobalt-60 in a Geiger-Muller tube [28]. Shown in Figure 2.6a, this QRNG measured the arrival times of electrons produced by the beta-decay process of the radioactive material, which although governed by the half-life of the isotope, cannot be predicted exactly. The times are measured by a digital counter, and the least significant bits are used as the random number. As this was the first QRNG system, it is not surprising that the final output rate is extremely slow (≈1.2 bits/s), and no random number tests were performed.

In 1970, Helmut Schmidt constructed a similar system [26] based on the radioactive decay of Strontium-90, Figure 2.6b. By this time PRNGs (and their flaws) were being investigated, and as a result this work has a much more thorough analysis of the randomness, and even possible sources of correlation. A modulo-M counter forms time-bins, and whichever bin a detection falls into is the random number. Simple XOR whitening of successive events is performed, as are basic random number tests on uniformity. Interestingly, the author used his QRNG to perform research [36] into the effects of human consciousness on RNGs at the Rhine Research Center Institute for Parapsychology. His experiments involved machines with one red and one green light, and subjects were to try to mentally influence one light to illuminate more than the other. He reported success rates 1-2% above what would be expected at random, but he always worked alone, and no one has ever been able to reproduce his experiments.

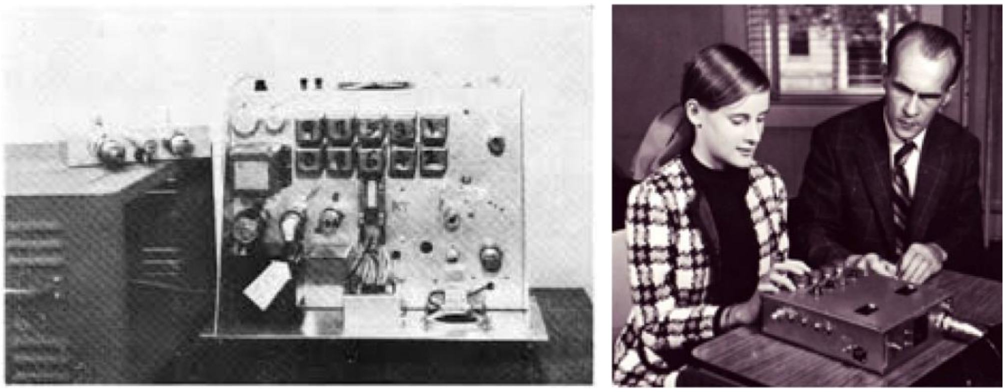


Figure 2.6. First published QRNG system (left) based on radioactive decay, and designed by M. Isida and H. Ikeda in 1956 *[28]*. Subsequent QRNGs also used this technique, such as the Strontium-90 based system by H. Schmidt (right), which was used to explore human parapsychology and whether RNGs could be mentally influenced.

QRNGs based on radioactive decay are still in use today, although they have been overshadowed by faster optical-based systems. The web-based RNG server HotBits [37] has been operating through Fourmilab in

Switzerland since 1996, and is based on the decay of Cesium-137. Hotbits compares the timing intervals between three successive electron detections, and the random bit is derived from which interval was the longest. Each hardware implementation can generate only 100 bits per second, so multiple systems are run in parallel and unused bits are stored for later use. Recently, other proposals for radioactive-decay QRNGs have been put forth, with the Geiger counters replaced by semiconductor PIN photodiodes [38].

2.3.2 Electronic Shot Noise QRNGs

In electronic circuits, random fluctuations occur due to the fact that current is actually a flow of discrete electrons. In optics, similar variations of the number of detected photons are observed due to the quantization of light. In both cases these fluctuations, or *shot noise*, can be described quantum mechanically and are considered to be governed by Poissonian statistics. Because events following a Poissonian distribution occur at an average rate and are independent of each other, the measurement of electronic shot noise is a suitable candidate for random number generation.

One of the earliest QRNGs was commissioned by the Navy in 1962 to evaluate radar detector performances through Monte Carlo simulations, and to simulate physical phenomena [27]. A thyratron is a gas-filled tube that can be used as a high-power electrical switch and rectifier. When operated as a diode, the device produces approximately white noise, although it must still be whitened for uniformity. The output is amplified, and then used to control the pulse width of a variable pulse generator. These pulses are then used to control a toggle T-type flip-flop, whose output changes upon the arrival of each random pulse. Finally, the output was then combined with a logical-AND to an accompanying synchronous 15 kHz clock, and can be sampled when needed. Fourteen devices were connected in parallel, and several different configurations could be chosen based on the user needs. Simple chi-squared and uniformity tests were performed, and passed.

Various other approaches have been published and there exist commercial products which utilize electronic shot noise for random number generation, but they will not be discussed in greater detail here. In practice quantum electronic shot noise and classical thermal noise occur simultaneously and are difficult to isolate from each other. As we will now discuss, there are, however, a few instances where rigorous quantum entropy estimation is performed, and also situations for which quantum shot noise dominates [39]–[41].

2.4 Optical Quantum Random Number Generators

Systems based on the quantum properties of light make up the majority of modern-day QRNGs. There exist multiple detection systems capable of resolving light at the single-photon level (avalanche photodiodes, superconducting nanowires, photomultiplier tubes, etc.), and there are many parameters of light which contain inherent quantum randomness (though they are all based on how many photons are detected in which optical modes). Since optical processes typically occur on much shorter time scales than those of electrical or radioactive origin, optical QRNGs have greatly surpassed the random bit generation rates of their counterparts, often reaching well into the GHz range. While the following section is by no means an exhaustive collection of all optical QRNGs, its aim is to illustrate some of the more popular systems in the literature to give an understanding of the types of processes suitable for random number generation. Particularly thorough reviews of both published and commercial products can be found in [34], [42].

2.4.1 Optical Path Generators

One of the earliest examples of an optical QRNG is based on the behavior of single-photons at a beamsplitter, where a fraction of incoming light is either transmitted or reflected through one of the two output ports, as shown in Figure 2.7. In the case of a 50/50 beamsplitter and a single incident photon, that photon has equal probability of being present in either output port. Quantum mechanically, we may write the action of the beam splitter as follows, where the factor *i* is due to the phase shift of reflection relative to transmission [43]:

$$|a\rangle \rightarrow \frac{|c\rangle + i|d\rangle}{\sqrt{2}}. \tag{2.2}$$

In the case of a perfect single-photon source, 50/50 beamsplitter, and 100% efficient single-photon detectors, this QRNG will register a count at either detector half the time for each individual photon, which can then be assigned a bit-value and used for random number generation. This type of QRNG first appeared in [44], where the beamsplitter was used to passively make the random basis choices necessary for quantum key distribution. Later, single-photon photomultiplier tubes were added to digitize the random choice for use in random number generation, achieving a rate of 1 Mb/s [45].

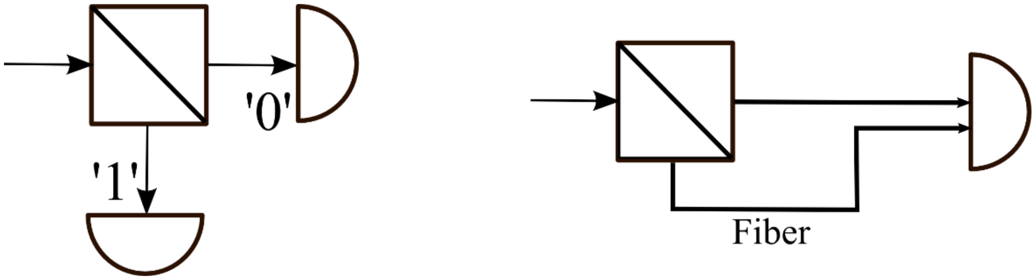


Figure 2.7 Example of a common optical-QRNG implementation (left). Photons are directed along one of two paths by a beam-splitter and are detected at either output port. In another implementation (right), bias introduced by differing detection efficiencies was eliminated by introducing a longer optical path in one arm, and binning detections by their temporal difference from the laser drive pulse.

Optical-path QRNGs suffer from a variety of problems, each of which affects the random output in a different manner. Ideally, the source would produce a single independent photon 100% of the time. A heavily attenuated laser diode or LED will approximate this *most* of the time as long as the mean photon number is low. Correlations due to thermally generated spontaneous emission events can be essentially eliminated by generating photons with a temporal separation larger than the coherence time of the source, and bias due to unbalanced beamsplitters or single-photon detectors with varying efficiencies can be reduced by post-processing, such as a hash function [6].

The detection efficiency of a single-photon detector is device-dependent and can fluctuate with time and temperature. Implementations using multiple detectors will suffer from differing probabilities of detection on either path, although this problem was alleviated in [46], where a single detector was used and photons were distinguished by an optical path delay (Figure 2.7b).

Single-photon detectors typically possess a “dead time” after a successful detection event, during which the detector has a reduced detection probability; in a two-detector beam-splitter type of QRNG, this can lead to anti-correlation of neighboring values if the repetition rate is not limited. Certain detector technologies can also produce correlations due to afterpulsing, a false secondary detection due to trapped charge from a previous detection. Alternatively, anti-correlations can result if a detection occurs in the midst of the device’s recovery period, detections referred to as “twilight” events [47]. These experimental problems all result in the loss of entropy, limiting the random bit generation rate to less than one bit per detection when only two detectors are used. There have been demonstrations of integrated systems with eight individual detectors [48], which produce nearly three bits per photon detection.

A beamsplitter is an imperfect optical element, subject to fabrication variations, temperature fluctuations, and other effects which can unbalance the relative bit probabilities of QRNGs of this type. The commercial

QRNG Quantis [49] eliminates the possibility of an unbalanced beamsplitter by removing it entirely, instead placing the detectors in positions where the amplitudes of the source are equal. The internal structure of the device is proprietary, but optical ND-filters are heavily unbalanced beamsplitters, and any QRNG which attenuates its optical state in such a manner will be subject to the same systematic effects. Although there exist promising variations on the original approach, due to the maximum operational rate of single-photon detector technologies, current path-based QRNGs are typically limited to the tens of MHz range.[4]

2.4.2 Photon Arrival Time QRNGs

The photon arrival times of heavily attenuated incoherent and coherent sources can be said to follow a Poisson distribution,[5] where events occur independent of one another and at an average rate λ. Analogous to the early radioactive decay systems of Section 2.3.1 and throughout, QRNGs of this type take advantage of faster optical processes, which allow for random bit generations at much higher rates. There are many different ways to generate randomness from these arrival times, but the timing information of successive events is typically extracted by a time-tagger, a device capable of resolving when each detection occurs with high precision. The constructions of two such arrival-time-based QRNGs [12], [13] are detailed in Chapter 3, and a high-precision FPGA-based time-tagger in Chapter 4.

One of the earliest QRNGs using photon arrival time was [50], where the intervals between successive photon detections are compared, and bit values assigned corresponding to which was longer. As shown in Figure 2.8, events from a random source are counted by both a free-running and resettable clock signal, operating at a frequency faster than the average rate of detection. If $t_1 < t_2$ a logical zero is assigned; if $t_1 > t_2$ a logical one is assigned. Previously, arrival time QRNGs of any type had only used a free-running clock to assign arrival times, but in this publication it was found that doing so introduced additional correlations. The impact of which clock signal is used helps to highlight a very important limitation for systems of this type, as we discuss.

Because the detection events are by definition random and independent of each other, there *should* be equal probability of having a $t_1 > t_2$ event versus one in which $t_1 < t_2$. In the limit of an infinitely fast clock

[4] In principle, faster detection technologies (e.g., super-conducting nanowires) could be used, but no publication of this type was found.

[5] See Chapter 3 for a short derivation of coherent vs. incoherent light statistics. In practice, other assumptions sometimes have to be made, such as sampling at time scales much longer than the coherence time of an LED to avoid correlations.

this could be achieved, but in experimental practice this is not possible. As the clock frequency is reduced, there emerges an ever-growing probability that $t_1 = t_2$, in which the random events fall in the same 'time-bin'. Additionally, the correlation between successive bits grows as the slower clock cycles corresponding to previous events start to 'bleed over' into the next detection's value. The choice of a resettable clock virtually eliminates the correlations, but does not address the coarse clock's quantization influence. In the case of an extremely slow clock signal, there would be no variation in the number of cycles per detection and there would be no random information available. It is therefore desirable to have a time-tagger which can resolve intervals as precisely as possible, although to isolate the source of randomness to the detection process the resolution should not be lower than the timing jitter of the constituent single-photon detector. As discussed in Section 2.4.1, single-photon detectors can also suffer from dead time and afterpulsing, so these characteristics must also be accounted for to avoid additional correlations.

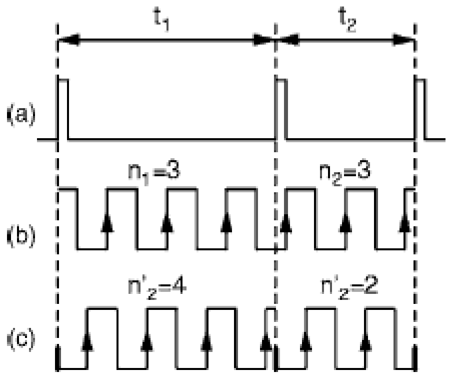


Figure 2.8. Method of extracting random bits by comparing successive time intervals (a) between photon detections. The coarseness and digitization of the free-running clock (b) led to correlations between bits, so a resettable clock (c) was used. Figure courtesy of *[50]*.

Accounting for quantization bias is usually achieved by post-processing, and QRNGs of this type have had success in literature and commercial applications. In our earlier work [12], the timing interval itself was divided into 5 ns time-bins and used as the random value, allowing for multiple bits per detection and a generation rate of 130 Mb/s. Because the waiting-time distribution of a Poisson process is a decaying exponential the entropy extracted was less than for a uniform distribution, and was whitened using the SHA-256 hash function. In [13] we reduced the amount of necessary whitening by tailoring the current driving the light source such that the waiting-time distribution of the resulting photon flux was approximately uniform (Figure 2.9b). Although residual hashing was still needed, this system was able to produce cryptographically secure random bits at rates of 110 Mb/s, the world record at the time.

There exist several other interesting variations of the above general concepts. In [51], different time-bin grouping methods are explored in order to further shape the distribution towards uniformity. The commercial *quRNG* [52] counts the number of detections in an interval of predetermined length, and assigns bit values based on whether there was an even or odd number of detections. Several generators have a hybrid optical-path / arrival time approach, such as using the two detectors in the beamsplitter QRNG to start and stop a timing interval counter [53]. The performance of arrival-time QRNGs is usually in the hundreds of MHz range, and is limited by a combination of the maximum detector count rate and the time-tagger performance[6] (both time-bin resolution and readout speed to PC).

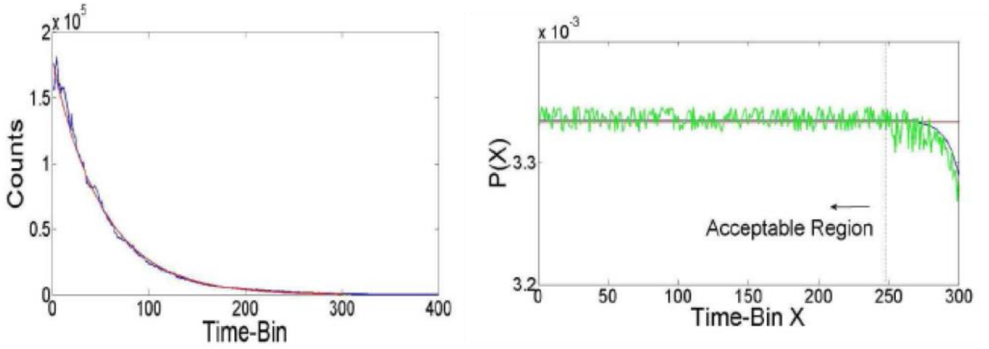


Figure 2.9. Measured (blue) and expected (red) waiting-time distribution for a constant-current laser source (left), which requires whitening. By tailoring the current such that the probability of detection is the same for each time-bin, the ideal uniform distribution can be approximated (right), although inaccuracies lead to deviation and residual hashing is still required. Figures courtesy of *[12], [13]*.

2.4.3 Laser Phase Noise QRNGs

Another optical QRNG approach which has achieved significantly higher bit rates than others is based on measuring the phase noise of lasers. Within a laser, photons due to spontaneous emission are constantly being randomly generated, and cause fluctuations in the output. The most common way to measure such phase variations in QRNGs is with an unbalanced Mach-Zehnder interferometer, as shown in Figure 2.10. In such setups, the optical path delay τ produces an amplitude fluctuation at the output proportional to the random phase change. To ensure the randomness extracted is primarily due to quantum mechanical effects, τ is chosen to be much greater than the coherence time of the source τ_c. Because the interferometer uses standard analog photodetectors they are subject to the usual classical sources of noise (thermal noise, dark current, etc). For this reason, the source is operated at low intensity near threshold, where the quantum phase noise is highest and will dominate over the photodetector noise.

[6] A summary of time-tagger types and their performance is included in Chapter 4.

Finally, the photodetectors must be sampled at rates much longer than $\tau + \tau_c$ to reduce any correlations between successive bits, although the system in [54] eliminates this by using a pulsed source.

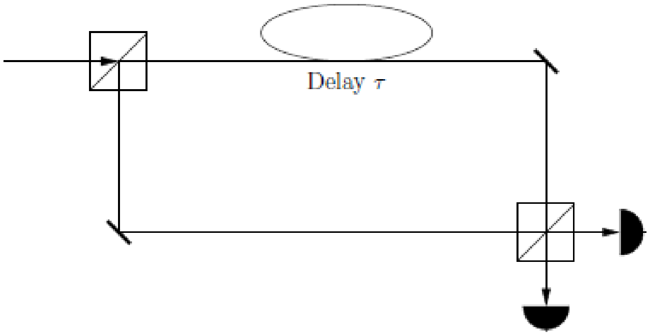


Figure 2.10. Depiction of an unbalanced Mach-Zehnder interferometer from *[34]*, where variations in phase are converted to amplitude, and detected by classical photodetectors. The delay τ and sampling periods must be chosen to be long enough to isolate the quantum mechanical effects and avoid correlations between successive bits.

Systems of this type have become increasingly popular in recent years with multiple publications being produced [14], [55]–[57]. The QRNG in [14] was shown to produce bits at a rate of 68 Gbps, a rate significantly higher than previous systems. Recent loophole-free tests of Bell's inequalities also utilized a phase-noise QRNG [57], where the fast sampling rate and low latency made it particularly suitable for locality considerations.

2.4.4 Photon-number QRNGs

As discussed in Section 2.4.2, photon arrival times can be said to follow a Poisson distribution, in which events occur independent of each other and at some average rate λ. It then follows that the *number* of photons within a certain time interval is also Poissonian, and in particular the probability of *n* photons arriving during a fixed time T is given by Equation 2.3. Although the statistics of the random source may follow a process of rate λ, what is observed is a process of rate λ_{obs}, which also takes into account events missed due to detection efficiency, dead time, optical attenuation, and other effects. The resulting process is equivalent to a sub-sampled Poissonian process, so it is still suitable for quantum random number generation.

$$P(n) = \frac{(\lambda T)^n}{n!} e^{-\lambda T} \qquad \text{Eqn. (2.3)}$$

Generators which measure photon-number typically do not use detectors with large dead times, such as commercial single-photon avalanche photodiodes, whose 50 ns recovery period would make high-speed bit-generation impossible. PMTs have been used with moderate success [58], but it is the introduction of integrated detector arrays which has made this QRNG type competitive. For example, the generator in [59] used a 10 x 10 array of silicon photon multipliers, while the commercial Micro Photon Devices generator [60] uses a 32 x 32 array of CMOS SPAD detectors. In [32], the speckle caused by atmospheric turbulence over an 143 km free-space link in the Canary Islands was used, although it was measured by a classical CCD camera.

The photon-number information can be extracted in several different ways. In systems utilizing single detectors with a small dead time, the least significant bit of the number of photons detected in a time interval can be used, or the number detected in successive intervals compared and bit values assigned as to which interval contained more counts. When arrays are used, more information becomes available, because what matters it is not only how many photons were detected, but *where* they were detected, although whitening is usually required to account for differing pixel detection efficiencies and variations in the spatial profile of the source. Although integrated arrays provide more information per interval, they are still composed of detectors with individual dead times, and consequently they are currently limited to rates within the 100 MHz range [61].

2.4.5 Amplified Spontaneous Emission QRNGs

Long-range fiber-optic communication systems require optical amplifiers, such as erbium-doped fiber amplifiers (EDFAs) or semiconductor optical amplifiers, to achieve fast data rates. For both of these technologies, input light directed into the gain medium stimulates the emission of additional photons, amplifying the intensity of the output signal. Inside this gain medium there is also the possibility that photons can be created through spontaneous emission. These photons are also amplified, resulting in a random signal of quantum origin at the output of the amplifier, known as amplified spontaneous emission (ASE).

As shown in Figure 2.11, the QRNG system in [62] used an Er/Yb-doped fiber with no input as an ASE noise source, which generates broadband optical noise. This noise is filtered, amplified, then split into two independent polarization components. Because the optical bandwidth of the ASE noise is much higher than the electrical bandwidth of the fast square-law photodetectors, band-pass and low-pass filters are chosen to achieve a signal-to-noise ratio low enough to create random bits. The photocurrent output by the detectors is mostly what is referred to as ASE-ASE beat noise, whose distribution depends on the choice of filters. The voltages of each detector are compared, and a random bit is generated depending on which is larger. The resulting bit stream contains some residual correlations between bits, so it is XOR'd with a delayed version of itself. Due to the absence of single-photon detectors and speed at which the random process changes, this QRNG was able to output random bits at a rate of 12 Gb/s.

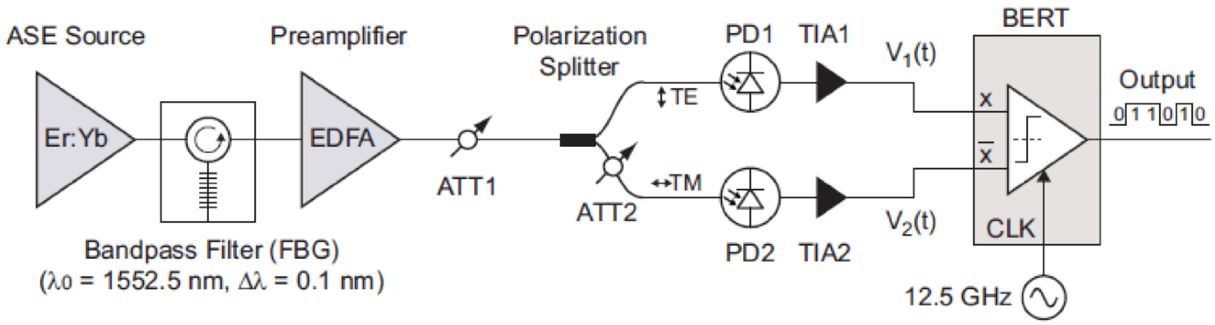


Figure 2.11. Implementation of QRNG system in *[62]*. Broadband white-noise generated by amplified spontaneous emission is filtered, split, and detected, producing a signal of random amplitude. Voltages are then compared and sampled at a rate of 12.5 GHz, which after a XOR, produce high-quality random bits. Figure courtesy of *[62]*.

Other ASE-based QRNGs extract their randomness from super-luminescent LEDs, diodes with internal optical gain [63]. The optical output of these devices is a flat spectrum over a wide range of frequencies, and noise of different frequencies can be separated and used to increase the random bit rate. The particularly interesting QRNG in [63] not only samples the ASE but achieves 20 Gb/s by multiplexing the light into several wavelength components, creating multiple parallel QRNGs from one optical source, as shown in Figure 2.12.

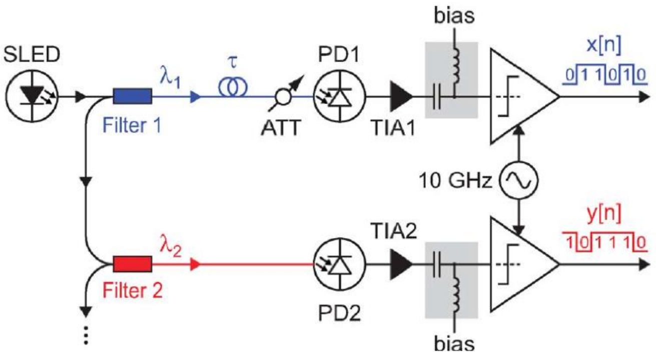


Figure 2.12*.* ASE-based system which creates multiple parallel QRNGs by measuring different frequency components of light emitted from a super-luminescent LED. Figure courtesy of *[63]*.

2.4.6 Raman Scattering QRNGs

Similar to ASE, another technique used to achieve optical amplification is Raman scattering, of which there are two types: spontaneous and stimulated. In spontaneous Raman scattering, a laser beam of frequency ω_p is incident upon a material sample, e.g., diamond. These photons can interact with the molecular lattice by either absorbing or creating a phonon, to create a photon of higher (Stokes photon, ω_s) or lower (Anti-Stokes photon, ω_{as}) frequency. In stimulated Raman scattering an additional beam of frequency ω_S is added, and when the frequency difference $\omega_p - \omega_s$ matches a particular molecular vibrational frequency, optical amplification can be achieved. The quantum fluctuations present in these processes show themselves as both phase and amplitude fluctuations in the output field, which can be measured to produce random numbers.

There are several Raman scattering QRNG systems in the published literature [64]–[66], and although the quantum process is different the techniques for measuring amplitude and phase fluctuations are standard, and the same as those presented in the systems above. One interesting exception is the influence of power fluctuations in the optical pump, which introduce non-quantum noise into the process, and can reach amplitudes that totally mask the quantum process from measurement. In this case, a portion of the optical pump power is monitored as an amplitude reference, and measurements of the Raman scattering are adjusted accordingly (Figure 2.13).

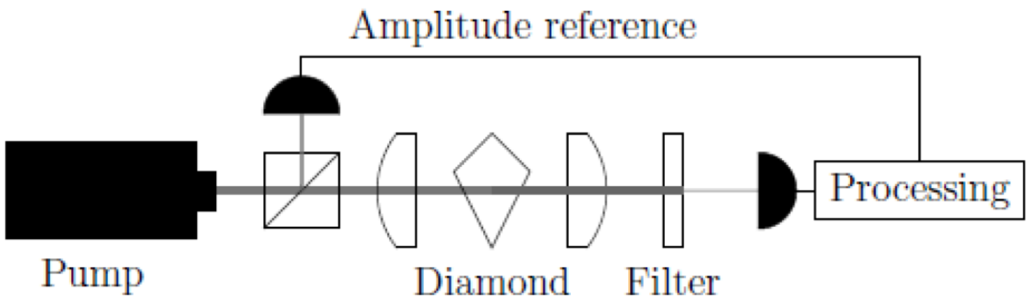


Figure 2.13. QRNG based on Raman scattering-induced amplitude fluctuations. Power variations in the pump require additional processing, where a portion of the optical signal is used to calculate the appropriate compensation factor *[34]*.

2.5 Certifying Random Numbers with Quantum Mechanics

One of the biggest problems facing random number generators of any type is that of trust. There are statistical methods to test for predictability (bias, correlations, etc.), but every random number test assumes some level of faith in the underlying device or algorithm. If an RNG fails/passes a particular test, we assign some probability that the observed statistics were from a non-random/random process. However, there is always a non-zero probability that the tested sample just happened to *appear* non-random or random. As more statistics are gathered the confidence of the test result increases, but again *only* if the underlying mechanism is trusted. There still exists the worst-case possibility of a malicious attacker who has somehow gained control of the system, and has inserted data that appears to be random, but is in fact totally controlled. This section briefly summarizes popular methods of dealing with potentially untrusted systems by using quantum mechanics.

2.5.1 Device Independent QRNGs

In the QRNGs previously presented, it has been the case that the quantum randomness has been somehow mixed-in with classical noise or other effects. For these systems, certain assumptions have to be made to make estimations on the amount of generated quantum entropy. The first method involves QRNGs in which no assumptions are made whatsoever about the generator itself. Instead, these “device-independent" QRNGs assess only the output, which comes from a system designed in such a way that certain results *guarantee* the presence of quantum-mechanical randomness.

One of the most obvious ways to certify the presence of quantum information is a Bell test, which was the approach taken using entangled ions in [67], and later with much higher rates using entangled photons in [68]. A violation of a Bell inequality guarantees that the output resulted from an entangled quantum system, and unless our fundamental understanding of quantum mechanics is wrong, contains inherent randomness. Any Bell inequality is sufficient, but in this paper the authors chose the Clauser-Horn-Shimony-Holt (CHSH) inequality, whose correlation function is given in Equation 2.4.

$$I = \sum_{x,y} (-1)^{xy} [P(a = b|xy) - P(a \neq b|xy)] \qquad \text{Eqn. (2.4)}$$

As shown in Figure 2.14., the test considers two separate systems of Yb$^+$ qubits, trapped in separate vacuum chambers. Each atom can emit a photon (entangled with its atomic qubit) which is fiber coupled and directed to the beamsplitter (BS). If these two photons are indistinguishable, their interference at the beamsplitter can be detected by the PMTs and used to herald the photon entanglement, and due to entanglement swapping, the atomic qubits as well. Two random values x and y are used to rotate each qubit in one of two ways; the ion’s fluorescence is then measured by the PMTs on the right, resulting in two bit outputs, a and b. The correlation function is then approximated by calculating the probabilities in Eqn. 2.4 after taking many measurements.

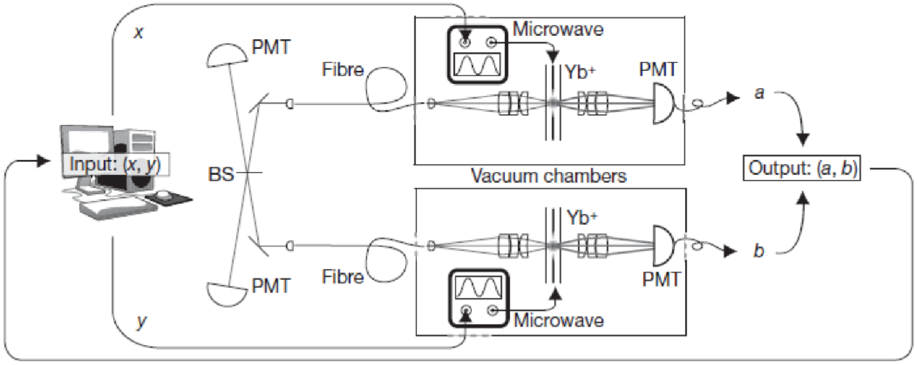


Figure 2.14. Illustration of device-independent QRNG which extracts quantum entropy through the measurement of a Bell violation. Figure courtesy of *[67]*.

For classical, local-realistic models, the value of I should never exceed 2. Any value above indicates that the measurements belong to a process of quantum mechanical origin. Beyond the scope of this dissertation, the random values which produced a particular violation contain a minimum amount of quantum entropy, which is quantified in the Supplemental Information of [67]. These measurement values are not uniform, so they are then processed by a randomness extractor. In this experiment the measurement and extraction processes were extremely slow, resulting in 42 bits of high-confidence quantum origin after approximately a month of continuous measurement.

Until recently, this type of device-independent QRNG only demonstrated a proof-of-concept. Although a perfectly constructed Bell test will guarantee the presence of quantum information, there existed imperfections in the experiment which forced additional assumptions, or loopholes. For example, in the above experiment it is possible that the ‘random’ choices x and y could be totally predetermined, and tailored in such a way to produce a false violation. It is the construction of a system addressing this issue that is the subject of Chapter 5. In 2015 several “loophole-free” tests of Bell’s Inequalities were performed [16]–[18], although they have not yet been used to extract guaranteed random bits.

2.5.2 Self-testing QRNGs

Component aging, temperature fluctuations, hardware failures, etc., can all influence the calibration of TRNGs, and affect critical output parameters such as bit uniformity. It is therefore fairly common, especially in critical security-based TRNGs, to test the quality of the generator while it is operating. This can be accomplished, for example, by including a hardware version of the NIST randomness tests, as was shown in [61]. Another option is to perform rigorous characterization of the system, and by characterizing every *identifiable* source of imperfection (and the amount in which it can affect the output bit stream), a bound can be placed on the expect amount of entropy [57].

One particularly interesting approach was taken in [69], in which a simple beamsplitter-QRNG was implemented with entangled states. This QRNG was similar to the one presented in section 2.4.1, but it also performed an on-the-fly quantum tomography of the input state from a sample of the bit stream. By extracting the effective-qubit density matrix, a bound on the fraction of quantum entropy generated was calculated and then used with a randomness extractor to prepare a shorter, higher-quality, random bit stream.

Chapter 3 — Photon Arrival Time Quantum Random-Number Generation

The beamsplitter-based system of Section 2.4.1 was arguably the first QRNG used in the context of quantum information, being integrated into an early demonstration of quantum-key distribution [44]. Although sufficient for a proof-of-concept, the limitations of this approach were immediately apparent. The bit-generation rate was limited by the dead time of the detectors and the bit-bias was influenced by both the beamsplitter transmission ratio, and varying detection efficiencies. When published as a standalone system in [45], bit-generation rates of 1 Mbit/s were shown, and the commercial product Quantis [49] is able to achieve 16 Mbit/s by packaging four 4 Mbit/s systems together. However, as long as the random "choice" remains binary, at most only one bit can be generated per photon.

Similar to the radioactive-decay QRNGs of Section 2.3.1, the arrival time of a single-photon can also be used as a quantum random variable. Because the timescales of typical optical processes can be much shorter than their radioactive counterparts, they have the potential to enable random number generation at much faster rates. The first such system was implemented in [50], in which the arrival times of successive photon detections were compared (see Figure 2.8), and bit values assigned based on which was longer. It was later realized that *multiple* bits per detection could be extracted by using the arrival time itself as the random bit values, as proposed in [70], [71].

In this chapter we discuss the theory, implementation, and results of two arrival-time optical QRNGs. The first system will be referred to as the constant-current QRNG (CCQRNG), and the second the shaped-pulse QRNG (SPQRNG). In the CCQRNG, the random information extracted from a Poisson waiting-time distribution is used to generate entropy at rates of 130 Mbit/s, although the raw timing intervals need to be whitened to remove bias [12]. In the PSQRNG, the current driving the photon source is shaped in such a way that photon statistics are tailored to approximate a uniform distribution, and by reducing the amount of hashing required, random bit generation at rates of 110 Mbit/s was achieved [13].

As the SPQRNG is in many ways an extension of the CCQRNG, much of the theory and background material remains the same. We start by explaining the concepts relevant to both systems, such as Poissonian photon statistics and entropy. The implementation-specific details of each system are shown, highlighting the improvements made in the SPQRNG. Finally, a brief introduction to random number tests is presented, as well as the results and differences in test performance for the two systems.

3.1 General Operation

Both of the QRNGs in this chapter generate their entropy in the same fashion – by extracting random information from the arrival times of single-photos. A simplified diagram of our approach is shown in Figure 3.1. Photons are generated from a source, in this case a laser diode and its corresponding driver. The optical output is strongly attenuated, and transmitted single photons can then be detected by a single-photon avalanche photodiode (SPAD). The output of the SPAD is a sequence of digital pulses, separated by an amount of time related to the photon statistics of the light source. The detector pulses are registered by a time-tagger, a device which outputs a multi-bit value corresponding to the interval between detections. These bit-values are the random timing information, but due to the shape of their probability distribution the amount of randomness is reduced. To compensate, the data is whitened to prepare values suitable for random number generation. The processed bit-values can then be transmitted to a PC for analysis or randomness testing.

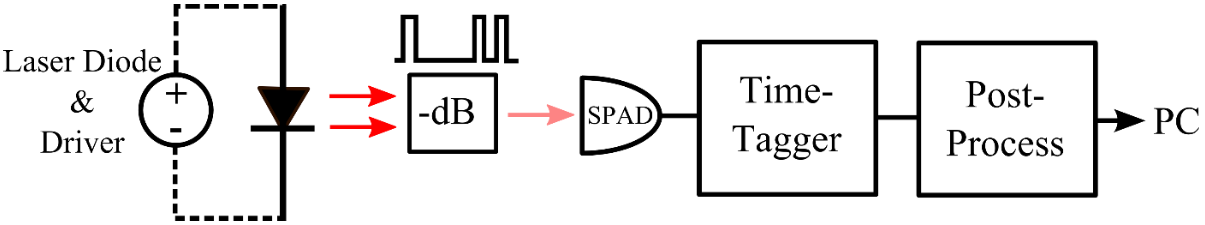


Figure 3.1. General diagram of our arrival-time QRNGs. Photons emitted from a light source are attenuated and detected by a SPAD. The resulting digital pulses are input into the time-tagger where timing-information bit-values are extracted and then post-processed using a hash function.

In order to classify a random number generator as 'quantum', a strong argument has to be made for the origin and quality of its output. Isolating and extracting information from a single physical process turns out to be extraordinarily difficult, so rigorous characterization should be performed to identify possible avenues through which additional processes can influence the random bit-stream. Even in this relatively simple system, consisting basically of a laser and a detector, there are many such noise sources, and so we must account for, or try to reduce, them as much as possible. The three major elements in our QRNG are photon emission, photon detection, and data processing, so to first-order the most prominent assumptions we have to consider are related to them. In the following sections we will address these questions, identifying possible sources of non-randomness, and how we account for them.

3.2 Poisson Processes and Photon Statistics

The Poisson process is one of the most widely encountered counting processes in statistics, frequently used to characterize events which occur continuously (with a known average rate per time interval λ), and independent of each other – properties ideal for random number generation. For a Poisson process, the probability that *k* events occur in a single time interval of length *t* is characterized by the equation

$$P[N(t) = k] = e^{-\lambda t}(\lambda t)^k / k!. \qquad \text{Eqn. (3.1)}$$

Because all events are independent, the time between arrivals is the same as the time until the first arrival, and thus the inter-arrival times, or waiting-time distribution, is given by the equation

$$P[N(t) = 0] = e^{-\lambda t}(\lambda t)^0 \,/\, 0! = \; e^{-\lambda t} \qquad \text{Eqn. (3.2)}$$

Processes for which λ is time-invariant are *homogeneous* Poissonian processes, and the waiting-time distribution takes the form of a decaying exponential with rate parameter λ. Both the expected value and variance of a Poisson process is equal to λ, so ***a probability distribution generated by a Poisson process will fluctuate around it's mean with standard deviation $\sqrt{\lambda}$.***

3.2.1 Coherent Laser Light

The monochromatic field of a single-mode laser can be described by a coherent state $|\alpha\rangle$,

$$|\alpha\rangle = e^{-0.5|\alpha|^2} \sum_n \frac{\alpha^n}{\sqrt{n!}} |n\rangle. \qquad \text{Eqn. (3.3)}$$

The intensity is proportional to the expectation value of *n,* or equivalently

$$\langle I \rangle \sim \langle \hat{n} \rangle = \langle \alpha | \hat{a}^\dagger \hat{a} | \alpha \rangle = |\alpha|^2 = \bar{n}, \qquad \text{Eqn. (3.4)}$$

where $\hat{a}$ and $\hat{a}^\dagger$are the annihilation and creation operators for that mode, and *n* is the photon number.

The probability to measure a given photon number is given by the projection

$$p(n) = \; |\langle n | \alpha \rangle|^2 = \frac{|\alpha|^{2n}}{n!} e^{-|\alpha|^2} = \frac{\bar{n}^n}{n!} e^{-\bar{n}}, \qquad \text{Eqn. (3.5)}$$

which is a Poisson distribution with mean value $\bar{n}$. Additionally, the variance of the photon number is $\Delta n^2 = \bar{n}$, also identical to the Poisson case. These quantum fluctuations are referred to as shot noise, and are due to the particle nature of light.

3.2.2 Thermal Light

When considering light emitted from thermal sources such as LEDs, the photon number follows as Boltzmann distribution, characterized by the equation

$$p(n) = \left(1 - e^{-\frac{\hbar\omega}{kT}}\right) e^{-\frac{n\hbar\omega}{kT}}. \qquad \text{Eqn. (3.5)}$$

Here the system is assumed to be in thermal equilibrium at some temperature T, ω is the angular frequency of the field, and *k* is the Boltzmann constant. For large mean photon number $\bar{n}$, the distribution can be written as

$$p(n) = \frac{\bar{n}^n}{(1+\bar{n})^{1+n}}, \qquad \text{Eqn. (3.6)}$$

which is a Bose-Einstein distribution with mean value $\langle I \rangle \propto \bar{n}$. Here, the variance of the photon number is found to be $\Delta n^2 = \bar{n}^2 + \bar{n}$. The $\bar{n}^2$ term of the variance grows as the mean photon number increases, eventually dominating completely.

3.2.3 Bunching, Antibunching, and non-Poissonian Light

There are particular types of photon sources which display non-Poissonian behavior, such as increased fluctuations or temporal correlations. A source which has a heightened probability of photons arriving closely spaced in time exhibits *bunching*, while one for which photons are more likely to be further spaced exhibits *anti-bunching*. Sources which display fluctuations above $\sqrt{\lambda}$ are *super-Poissonian* while those below are *sub-Poissonian*. A source with Poissonian statistics is neither bunched nor antibunched, with photon arrival times being distributed randomly and uncorrelated [47].

One of the most common measurements used for studying the characteristics of a single photon source is $g^{(2)}$, the second-order correlation function. Expressed in terms of time delay τ, $g^{(2)}(\tau = 0)$ gives information about a source's multi-photon emission probability. More detailed explanations can be found in [72], [73], but $g^{(2)}$ is a measure of the relative magnitude of the mean and variance of a source. For a

Poissonian source $g^{(2)}(\tau) = 1$ for all τ, as the mean is equal to the variance. Sources which display bunching have $g^{(2)}(0) > g^{(2)}(\tau \neq 0)$, while for those with antibunching $g^{(2)}(0) < g^{(2)}(\tau \neq 0)$.

3.2.4 Approximating Poissonian Light & Attenuation

In the preceding sections we have established that our ideal light source *should* emit pure coherent light, which follows Poissonian statistics and has an equal mean and variance. Thermal light has been shown to contain additional $\bar{n}^2$ fluctuations and bunching correlations, so reducing its contribution to our photon flux is critical. In this section we address the assumptions made when certifying the photon statistics of our source, a common laser diode.

As discussed in Section 3.2.1, the output of an *ideal* single-mode laser can be described as a purely coherent state, but an actual laser diode does not behave in this fashion. Under a sufficient forward bias, an electron in the conduction band has a probability to recombine with a hole in the valence band, and emits a photon at a frequency corresponding to the band gap of the semiconductor material. This is referred to as spontaneous emission, and light emitted through this process is considered thermal in nature, following the Boltzmann statistics discussed in Section 3.2.2, while light emitted by stimulated emission is considered to be in a coherent state. Because the voltage needed to raise a laser above threshold is well beyond what is required to begin spontaneous emission, a lasing-diode will necessarily emit both thermal and coherent light.

We address these unwanted photons in several ways. By forward-biasing the laser very strongly (well above threshold), the contribution from spontaneous emission becomes very small in comparison to that of stimulated emission. Photons emitted by spontaneous emission are emitted in random spatial modes, while those from stimulated emission belong to approximately one, which can be focused onto the active area of the detector. Finally, we employ approximately 120 dB of passive optical attenuation with neutral-density filters. All of these have the combined effect of reducing the final probability of a 'non-quantum' photon with thermal statistics reaching the detector to negligible levels. Finally, reducing the mean photon number through attenuation has the additional effect of reducing the $\bar{n}^2$ variance contribution, such that the light from a heavily attenuated laser diode very closely approximates Poissonian statistics**.**

3.3 Single-Photon Avalanche Photodiodes

For our QRNGs we have chosen to use single-photon avalanche photodiodes (SPADs) as our photon detection method. SPADs are semiconductor-based devices capable of resolving light at the single-photon level; they are moderately robust, operate at MHz rates, and often do not require cryogenic cooling. While convenient for laboratory use, SPADs also possess some traits which require additional assumptions to be made about their output. Here, we briefly describe some of the more common effects, and their influence on our random number generation.

3.3.1 Dead time

When a single photon is incident upon the active area of a SPAD it has a probability of creating an avalanche, a large amount of current which can then be detected. This current quickly saturates the detector, spreading charge until the device is biased below a threshold, or *breakdown voltage*. Below this threshold, the multiplication process no longer occurs, and the avalanche subsides. However, if the SPAD bias is not lowered, or *quenched*, then the avalanche will continue – eventually destroying the device.

During the below-breakdown period, or *dead time*, the SPAD is insensitive on the single-photon level and single-photon events are missed. Typical dead times are on the order of 50 ns, which limits photon detection to ≈20 MHz. Because of the random selection property of Poisson properties mentioned in Section 3.1, we can still have confidence in the randomness of the resulting distribution. However, when examining the output waiting-time distribution, the measured rate parameter will differ from what was actually output from the source, possibly affecting entropy calculations. Specifically, the correction applied to account for detector dead time is given by the equation

$$\lambda_{true} = \frac{\lambda_{measured}}{(1 - \lambda_{measured} * \tau_d)} \qquad \text{Eqn. (3.7)}$$

where λ_{true} and $\lambda_{measured}$ are the actual and measured rate parameters of the waiting time distribution, and τ_d is the dead time of the detector. A comparison of λ_{true} and $\lambda_{measured}$, for detectors with a 50 ns and 25 ns dead time, is shown in Figure 3.2.

In regards to affecting the statistics of our source, this dead time can be shown to be acceptable simply by the sub-sampling property of Poisson processes. Given a Poisson process with rate λ, if a random selection is made on each arrival with probability *p*, then the resulting distribution is also a Poisson process with rate λp. Assuming that the deletion is indeed made randomly, then the dead time (and our passive optical attenuation) can be assumed to have no negative effect, other than to reduce the overall rate, on the measured statistics.

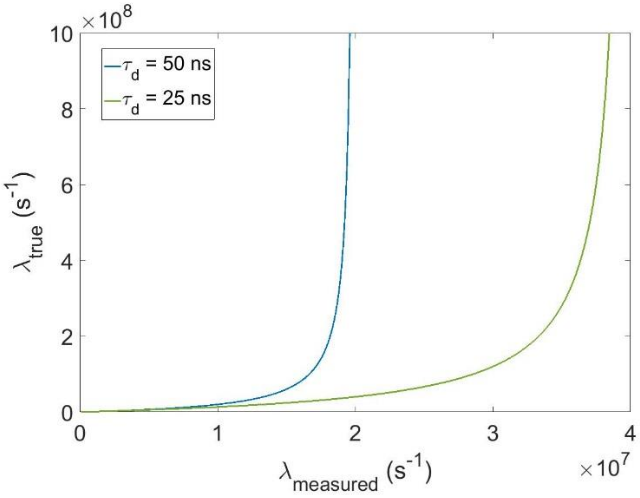


Figure 3.2. Measured vs. true rate parameters for a detector with 50 ns (blue) and 25 ns (green) dead time. At count rates approaching the detectors dead time almost every incoming photon will arrive while the detector is inactive, and higher input photon rates cause little effect on the resulting detection rate.

3.3.2 Afterpulsing

After an avalanche, excess charge quickly spreads through the device, with full saturation occurring on the hundreds-of-picoseconds scale. A portion of this charge becomes lodged in imperfections, or traps within the device. Depending on the trap lifetime τ, any particular charge carrier can be released, and if the SPAD is biased above breakdown, the released charge can trigger a secondary spurious avalanche, or *afterpulse.* In fact, one major consideration when determining the amount of dead time to apply in a SPAD is how long one has to wait until the probability of an afterpulse reaches an acceptable level. As there is no way to perfectly discriminate between an afterpulse event and an original photon detection, these events cannot be removed from the random number generation process, and must be accounted for in some fashion, since the afterpulses are clearly *correlated* with the original detection, and hence not good source of randomness. For our QRNGs, we used SPADs with a measured afterpulse probability of ≈0.1 %. Because of this low probability, and the fact that these events *originate* from a single-photon emitted by a quantum random process, we account for afterpulsing by increasing the excess entropy input into our hash function, a process described in Section 3.4.

3.3.3 Dark Count Rate

The dark count rate (DCR) of a SPAD is defined as the average amount of counts per second output by the detector when no input light is present. Largely due to thermally-generated carriers (generation-recombination processes), the rate of these events can be significantly reduced by temperature control.

Typically, a SPAD is housed in a TO-can type package and mounted on a Peltier stage. By reducing the diodes temperatures in the range of -10° to -50°C, dark count rates of 20 Hz or below are not unusual for silicon-based SPADs, with the III-V type detectors having somewhat higher dark count rates due to the higher density of impurities. For our QRNGs we utilized an ultra-low dark count detector, with a dark count rate of approximately 11 Hz. This tradeoff came with a much-reduced active area, and thus a much reduced detection efficiency, a metric largely irrelevant for our purposes. Due to the random nature of these events and the extremely low rate of occurrence, we account for these events in the same manner as afterpulsing, by allowing for extra overhead in our hash function.

3.4 Entropy

Given a QRNG's expected probability distribution P, it is necessary to accurately quantify the amount of random information available. In information theory, the Shannon entropy *S* of a distribution is the average information of all possible outcomes. Given *N* possible outcomes, the Shannon entropy is given by $S \equiv -\sum_{i=0}^{N} P_i log_2 P_i$, where P_i is the probability of a particular outcome, and *S* is measured in bits. For example, if there are four equally likely outcomes, $S = -4\left(\frac{1}{4} log_2 \left(\frac{1}{4}\right)\right) = 2$ bits.

Because RNGs are often concerned with security, it is useful to consider the most paranoid models. To that end, another related metric of interest is the min-entropy, which is the most conservative estimate of entropy, or the maximum likelihood of an attacker correctly guessing an output string. The min entropy is also measured in bits, and is given by $S_{min} \equiv -log_2 \max(\{P_i\})$. For example, if there are four outcomes with probabilities {0.3, 0.25, 0.25, 0.2}, $S_{min} \equiv -log_2(0.3) = 1.737$ bits.

One of an RNG's essential requirements is uniformity, or an output probability distribution for which every outcome is equally likely. Much like an unbalanced die, if some outcomes occur more frequently than others, the RNG is considered unfair, and on average results in less random information. For this ideal case the Shannon entropy is equal to $log_2(N)$, and since no outcome is more likely than any other, the min-entropy is also equal to the Shannon entropy. In this scenario the output is considered to be *one random bit per bit,* but for any other distribution both the Shannon and min-entropies are necessarily lessened. In such cases a whitening technique is necessary, compressing the partially random bit values into a more uniform stream, with entropy approaching one random bit per bit.

3.5 Time-Interval Measurement

Given a sequence of digital pulses output by a single-photon detector, we can extract inter-arrival times by measuring the temporal separation between successive events. To achieve this, we employ a time-to-digital converter, or 'time-tagger'. As shown in Figure 3.3, a time-tagger measures the length of time-intervals with some resolution *R* (units of seconds), where each unit of resolution is a 'time-bin'. Given a measured interval of length Δ*t*, a time-tagger will output a digital bit-value roughly equal to Δ*t/R.*

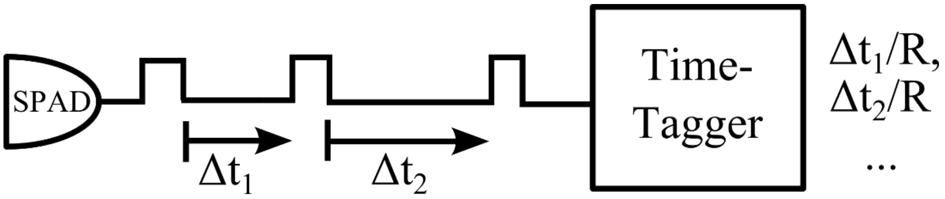


Figure 3.3. Operation of time-tagger in our QRNG systems. Digital pulses from the SPAD are translated into inter-arrival times and assigned to a time-bin of size R, the time-tagger resolution.

The time-tagging process affects the random information in several ways. Firstly, the quantization of the timing-information into bins changes the continuous probability distribution into a discrete sum. For our Poissonian case, given an average incoming photon rate λ and time-bin size Δt, the probability of a photon falling into time-bin *i* is given by the equation $P_i = \lambda \Delta t e^{-\lambda \Delta t i}$. Simulated probability distributions are shown in Figure 3.4, where the average time between detections is 15 ns and the time-bin resolution is 5 ns. In the discrete-case (right axis), the coarse resolution (relative to the average rate) causes the grouping of probabilities into large bins. We can easily calculate the expected Shannon entropy from such a distribution by using equation 3.7 and summing over all P_i.

In the case of very poor time-bin resolution, every detection would fall into one bin and the entropy would be zero. As the resolution increases, we can 'fit' more possible values into the waiting-time distribution and extract more random bits per detection. Because we represent our information as a binary value, the number of extracted bits scales with $\log_2$(N), where N is the maximum number of bins. It would seem as if arbitrarily increasing the resolution as much as possible would be beneficial, but this is not the case. Events arriving on bin-edges are assigned with some ambiguity, equal to the measurement jitter of the time-tagger. The detector also has measurement jitter, and to keep the entropy as 'quantum' as possible the time-bin resolution should not exceed this value. If kept on this time-scale, the only bit affected by the detector jitter will be the least-significant-bit, and although we can never eliminate the rare bin-boundary, we account for this by subtracting extra entropy in our hash function.

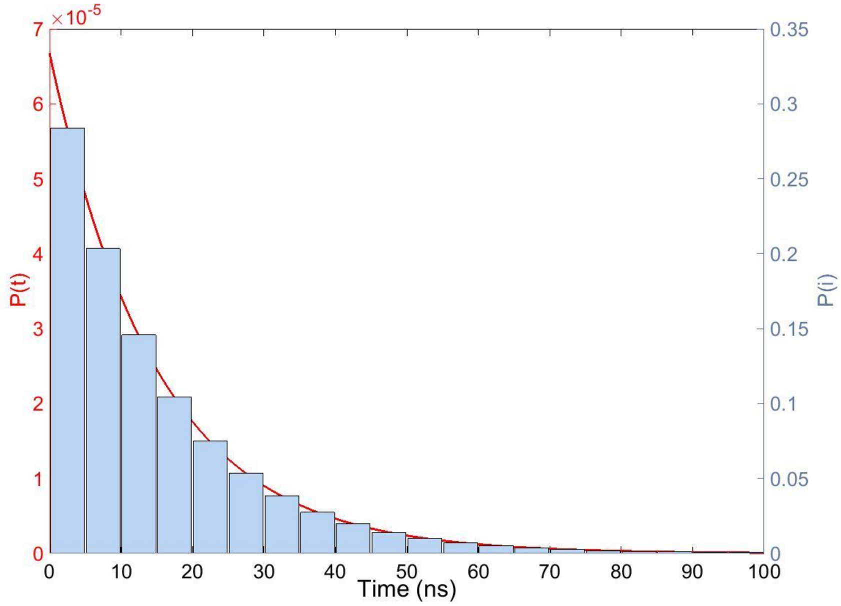


Figure 3.4. Theoretical continuous (red, left axis) and discrete (blue, right axis) exponential waiting-time distribution. Discretization and coarseness in the bin-size results in larger probabilities for earlier bins and lowered entropy.

3.6 Data Whitening

Because of the non-uniformity in our waiting-time distribution, a data-hashing process must be used to 'whiten' the raw random numbers, thereby preparing a shorter but more random string, with randomness approaching one random bit per bit. The hash function used in our QRNG implementations is the SHA-256 hash, created by NIST. The hash takes as input an N-bit block and the output is a 256-bit hash string, with each of the 2^{256} possible outputs having approximately the same probability.

A hash function is a mathematical procedure that converts a large amount of data into a smaller hash value. Often used in cryptographic applications, hash functions must satisfy two criteria: determinism and uniformity. Determinism is the property that the same input always maps to the same output, while uniformity requires that the outputs be spread across all possible values with the same probability. The SHA function belongs to a particularly robust family of hashes, referred to as cryptographic hash functions. These algorithms are extensively tested for uniformity, with hash collisions (two input values which map to the same output value) rarely occurring.

For our QRNGs we are sampling a Poisson-generated waiting-time distribution, for which the probabilities are not uniform, and the amount of entropy per bit is not one. When utilizing our hash function, care must be taken to ensure that the amount of *input* entropy is enough to ensure that the *output* entropy approaches unity. A common mistake is to *seed* a hash function with a shorter bit value and assume that the output contains perfect entropy. However, if for example the input string can only take *n* possible values, then the output will also only take *n* possible values. Extending the length of the output string does not necessarily translate into increasing the amount of entropy present.

By calculating the *average* amount of entropy per detection from our waiting time distribution, we can estimate the amount of entropy in our input string. Knowing that our hash function outputs 256 bits per message, we need to ensure that the input string contains at least 256 bits of entropy. It can be shown [74] that by saturating the input string with excess entropy, the output string will approach one random bit per bit. In particular, by adding 10 extra bits per message (i.e., inputing 266 bits of entropy), the output string will contain 0.999996 random bits per bit.

3.7 Constant-Current QRNG

In Sections 3.2-3.5 we addressed the possible sources of non-randomness in the arrival-time QRNG of Figure 3.1. The photon statistics of our laser diode can be considered to be Poissonian under heavy attenuation, and the detectors afterpulsing and dark count effects can be accounted for with a suitable hash function, if saturated with extra entropy. With strong confidence that our measurement contains essentially exclusively quantum information, we now turn to the implementation-specific details of our first system, the constant-current QRNG (CCQRNG).

For the CCQRNG the laser diode is driven by a simple current-limiting resistor and constant DC-voltage. Our single-photon detector is an idQuantique 100-MMF50-ULN [75], an ultra-low noise visible-range silicon device. To resolve time-intervals, we have utilized both an FPGA-based system with 5 ns resolution and an integrated chip (ACAM-TDX [76]), with 27 ps resolution. We only report the latter results here, while the 5 ns system was later upgraded to 50 ps, and is discussed in Chapter 5. The time-interval data was then input into a Virtex-6 FPGA [77].

Because the photon flux of a laser is directly proportional to the amount of injected current, we assume that the average rate of photon emission λ is time-invariant. Consequently, the photons directly out of the laser diode have inter-arrival times of the form exp(-λt), as discussed in Section 3.2. Although our

SPAD had a dead time of 45 ns (indicating a maximum detection rate of 22.2 MHz), internal circuitry disabled the detector at rates above 11 MHz. While we chose to operate at this speed, generating detections as fast as possible is not always optimal as the number of occupied time-bins decreases. An explanation of the trade-off between detection speed and entropy is given in Appendix A.

Given an 11 MHz detection rate and 27 ps time-bin resolution, the expected discretized probability distribution can be calculated following the process described in Section 3.4. By applying the Shannon entropy formula of Eqn. 3.7, we determine that the average amount of entropy *per detection* will be equal to ≈12.2 bits. By programming the FPGA such that it only hashes a block of data after 22 detections, we saturate the 256-bit SHA hash function with 268 bits of entropy per block. In this manner we ensure that the final hashed data has at least 0.999996 random bits per bit, as shown in Section 3.5. After post-processing was performed, the whitened values were transmitted over the PCIe bus for randomness testing, with sampled data shown in Figure 3.5. These results enabled a random number generation rate of 130 Mb/s, which at the time of publication was the world's fastest quantum random number generator [12].

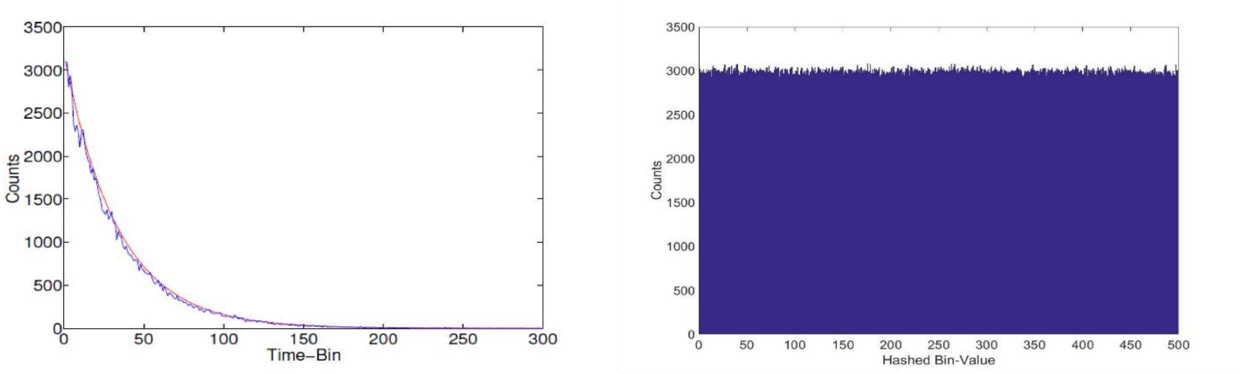


Figure 3.5. Measured (blue) versus theoretical (red) waiting time distribution. By whitening the data with a hash function, a shorter but more random string is produced, with entropy approaching one random bit per bit. The SHA-256 hash has 2^{256} possible outputs, so only a small sample (500 bins) is shown (right).

Due to the large difference between a decaying exponential and a uniform probability distribution, much less entropy was generated *per* detection, and a large amount of hashing was required. Although extensively characterized, depending on a non-quantum hash function in such a strong fashion is undesirable. To address this issue we developed the shaped-pulse QRNG, described in the following section.

3.8 Shaped-Pulse QRNG

Instead of relying on the complexity of the hash function, another approach is to modify the photon statistics themselves such that each time-bin has nearly the same probability of occurring. In this fashion we place more trust in the well-understood quantum mechanical description of the light source. To achieve this, we first consider an *inhomogeneous* Poisson process, in which the average rate of detections λ(t) varies with time. Given a waiting-time distribution of T possible time-bins, the ideal case is one for which the probability of every bin is time-independent, and exactly 1/T. Therefore, if we again assume Poisson statistics, λ(t) must be a solution to the equation

$$\lambda(t)e^{-\int_0^t \lambda(t')dt'} = \frac{1}{T}. \qquad \text{Eqn. 3.8.}$$

A rate parameter of the form λ(t) = 1/(T-t) is a solution to this equation.

Since λ(t) represents the photon arrival probability, it is dependent on the photon flux of the laser diode, which in turn has a linear relationship with the input current. Therefore, if the current is proportional to $I(t) = 1/(T-t)$, then the photon flux should be proportionally altered, achieving the ideal case.

As shown in Figure 3.6, this exact shape is impossible to produce, as the current grows rapidly (to infinity in the idealized case), thus requiring a very high bandwidth, dynamic range, and possibly damaging the diode. Approximating the shape, however, yields reasonable results, with a simulated min-entropy of approximately 0.96 random bits per bit. Addition details on our circuit realization of this pulse shape can be found in Appendix B.

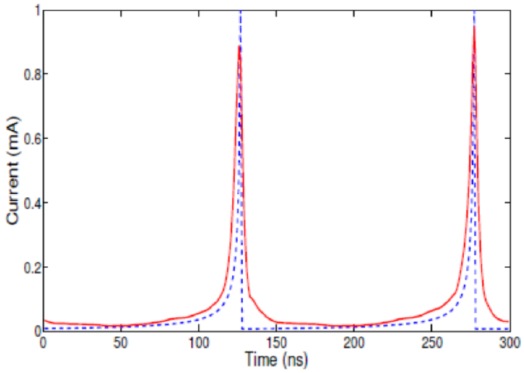


Figure 3.6. Ideal (dashed blue) and PSPICE simulated (solid red) shaped pulse of $1/(T-t)$.

The choice of the reset period T depends on several factors. The entropy per detection increases on a logarithmic scale, so an optimal reset period may not necessarily be the longest one. As seen in Figure 3.7, the optimal period is strongly dependent on the detector dead time. If we had detectors with no dead time, for example, the optimal reset period would be after 2 bins. For our 45 ns dead time, the peak generation rate corresponds to a reset period of approximately 23 ns, corresponding to a detection rate well above what our current SPADs can sustain [78].

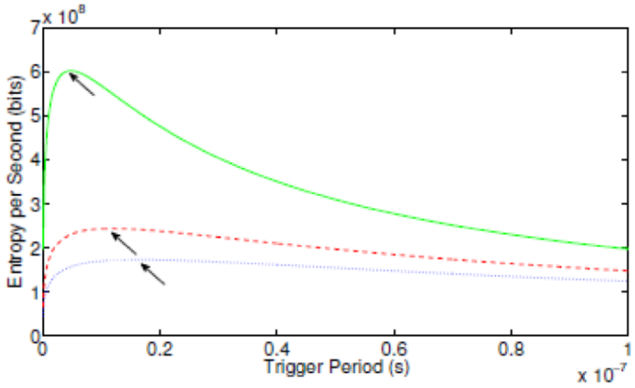


Figure 3.7. Theoretical peak min-entropy generation rate vs. trigger period T for 45 ns (short blue dash), 30 ns (long red dash), and 10 ns (solid green) dead times. Optimal trigger periods (as denoted by arrows) decrease with dead time.

Due to difficulties in approximating the ideal pulse shape at high frequencies, we were only able to operate at a reset period of 50 ns, which corresponds to a maximum min-entropy of 119 Mb/s, results of which are shown in Figure 3.8. While slightly slower than the optimal 23 ns, the resulting pulse better matched the desired shape.

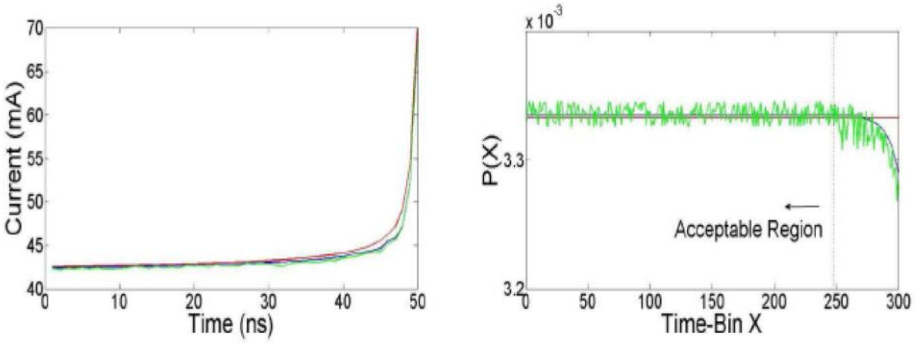


Figure 3.8. Theoretical (red), simulated (blue), and actual current pulse shape. The resulting waiting-time distribution (right) has a min-entropy of 0.90 random bits per bit; however, discarding counts to the right of the dotted line (where the driving pulse shape deviates from the ideal) results in a smaller but more random distribution, with entropy increasing to 0.9984 random bits per bit.

Due to the inaccuracy of the current shape at higher bandwidths, a portion of the random data had to be discarded. Events falling into these time bins were not input into the hash function, and not counted towards the entropy saturation requirement. The raw data's min entropy was 0.9984 random bits per bit (after truncation), a value already suitable for some applications. The saturated hash function increases this to 0.999996 random bits per bit, as well as washing out any additional effects (e.g., small differences between time-bin sizes due to clock inaccuracies) that may occur within the FPGA. Even though the raw bits were of higher quality, the slower detection rate and discarding of events resulted in a final random number generation rate of 110 Mb/s, slightly less than that of the CCQRNG.

3.9 Randomness Tests & Results

There is no true test to determine whether a sequence of bits is random, only those which test for the *appearance* of randomness. Nevertheless, by exhaustively testing against as many non-random effects as possible, we can rule out some of the more obvious scenarios, and the confidence in the final result is increased. There are several widely accepted random number test suites (i.e., NIST Statistical Test Suite [79], the DIEHARD Tests [80], and the TestU01 Alphabit Battery [81]), and here we discuss the testing results of our two QRNGs.

3.9.1 Testing Strategy and Interpretation of Results

In general, a random number test is designed to evaluate a specific null hypothesis – that the bit-sequence under test is random. Random number test suites are comprised of a collection of sub-tests, looking for correlations, frequency dependence, or an unbalanced proportion of bit values. For proper statistical significance, each test requires a large set of data (> 100s of Mb) – which is broken up into M-bit blocks. Typical values of M are of a size such that the data set is divided into 100-1000 blocks, with slight variations depending on the test. Thus for a data set as a whole, each test is run at least M times.

Each of the tests creates a test statistic, which is then used to calculate an associated *p-value*, a number related to the strength of the evidence against the null hypothesis. For example, the frequency test takes the sum of a binary bit-sequence, where -1 is substituted for zeros. Since each bit should be observed roughly the same amount of times, this test statistic should tend to zero for large amounts of data. The fluctuations around zero should follow a normal distribution, so the test is designed to compare against the complementary error function (*erfc*) of the sum, and a p-value can be assigned. Sequences which appear random will have a high p-value ($erfc(0) = 1$), while those displaying a large variation will result in a high test-statistic, and low p-value.

After the whole data set has been evaluated, the p-values are then judged by two criteria: proportion and uniformity. For a given test suite, a significance level (α) is chosen, usually on the order of 0.01 – 0.001. If $p \geq \alpha$, then the null hypothesis is accepted; i.e., the sequence appears to be random. Otherwise it is rejected, as the sequence appears to be *non*-random. For a random sequence, no more than a handful are expected to fail out of the total, so tests for which a disproportionate amount do not pass indicate a possible source of non-randomness. The failure limit varies for each test, but for most tests performed on 100 blocks of data, a maximum of 3-5 are expected to fail for α = 0.01.

Additionally, for a well-designed test statistic, a collection of p-values for an ideal RNG should be approximately uniform across the interval [0,1]. After all sub-blocks have been evaluated, the p-values are sorted, and a χ^2 test is performed to test for uniformity. Tests with significantly unbalanced p-values indicate the presence of non-random behavior, as even random number generators should sometimes *appear* non-random.

3.9.2 CCQRNG Results

We test the results of our constant-current QRNG by evaluating the output stream at two points. The *raw* data is the timing information extracted before the hash function, while the hashed data is that which has been post-processed. Data was recorded to files with sizes random from 10 MB to over 200 MB, and from incoming detections at rates of 1 to 11 MHz (in 1 MHz intervals), and the tests were run on each of them. The raw data contained significant bias due to its decaying exponential probability distribution, and failed the test suite. The whitened data passed every test suite it was evaluated with, and the results from both the DIEHARD and NIST STS test suites, for 100 blocks of 3-Mbit sets of data, are shown in Table 3.1.

3.9.3 SPQRNG Results

As with the CCQRNG, we tested our data both before and after the hash function. As before, the whitened data passed all random number tests in the suite, and gave a uniform distribution of p-values. While this gives a good indication that the hash used is suitable for whitening, of much more interest is how our pre-hashed raw data performed. As mentioned in section 3.7, we truncate the region of the probability distribution where the circuit does not approximate the pulse shape, and the waiting time distribution begins to diverge from the uniform case. We tested the raw data without truncating the affected area and, as expected, the random tests had a high failure rate. After truncating, the performance improved markedly, with the random data failing ≈ 5% of the time. However, the uniformity of the p-values was still not as high as in the whitened case. For some of the tests (Serial, Binary Matrix Rank, FFT tests) the uniformity was consistently *higher* than with the hashed data. An example of a sorted p-value graph for two of the NIST STS tests is shown in Figure 3.9. This could suggest that, while not as uniform as the hashed data, our raw data contained fewer correlations than the hashed output, as the mentioned tests look for frequency effects. While of marginally lower quality than the hashed data, the 5% failure rate could be improved with greater precision on the shaped-pulse circuit, eventually leading to performance commensurate with the post-processed data, which passes over 99% of the time [13].

Table 3.1. Results from the DIEHARD and NIST testsfor 300 MB of CCQRNG data, with M = 100, and α = 0.01.

Test	**DIEHARD result**	**Test**	**SP-822 result**
Birthday Spacing	PASS	Frequency	PASS
Overlapping 5	PASS	Block Frequency	PASS
Binary Rank Test 31	PASS	Cumulative Sum	PASS
Binary Rank Test 32	PASS	RUNS	PASS
Binary Rank Test 6	PASS	Long RUNS	PASS
Craps Test	PASS	Rank	PASS
RUNS Test	PASS	DFFT	PASS
Overlapping Sums	PASS	Non-Overlapping	PASS
Squeeze	PASS	Overlapping	PASS
3-D Spheres	PASS	Universal	PASS
Min Distance	PASS	Approx. Entropy	PASS
Parking Lot	PASS	Serial	PASS
Count the 1-s	PASS	Linear Complexity	PASS
OPSP	PASS	Random Excursions	PASS
BitStream	PASS	Serial	PASS

Although the purpose of the SPQRNG was to eliminate the need for post-processing altogether, it is *always* the case that uncontrolled variables will influence the experiment during operation. In fact, as discovered later in the research of Chapter 6, these influences can be significant, sometimes effecting the bit-probability by 20% or more. For this reason, if an accurate estimation is made of the entropy lost during post-processing, using a whitening function or randomness extractor is prudent.

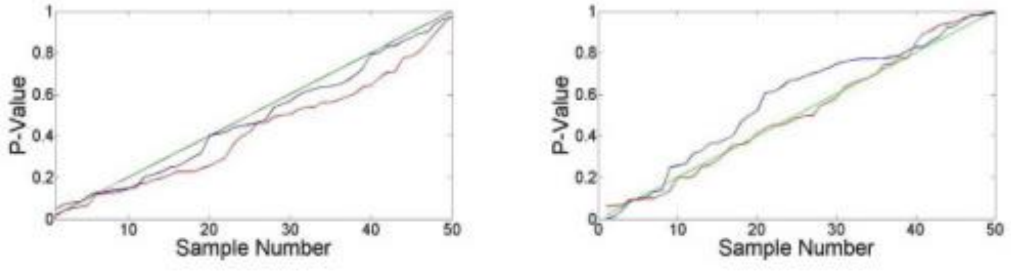


Figure 3.9. Example of sorted p-values for (red) raw and (blue) whitened random data vs. the expected straight line (green). Data from the hashed performed better in the approximate entropy test (a) while in the FFT test (b) the unhashed data actually performed better.

3.10 Conclusion and Future Improvements

In this chapter we summarized the implementation, theory, and results of two photon-arrival time QRNGs, the CCQRNG and SPQRNG. Although recent 68 Gbps QRNGs have greatly surpassed their output bit rate [14], this approach has many places for improvement. In the next chapter we present the results of our afterpulsing reduction efforts in SPADs. Since afterpulsing is often the metric which sets the dead time of a detector, reducing the afterpulse probability has a direct effect on the maximum detection speed of a SPAD. In Chapter 5 we outline our improved FPGA-based time-tagger. Although the 50 ps time-bin resolution is higher than the 27 ps resolution of the ACAM chip [76], our 11 MHz detection rate was at the limit of the ACAMs capabilities. Our FPGA system has almost no dead time, and is able to resolve events 1.6 ns after an initial start pulse. These improvements are unlikely to facilitate rates comparable to 68 Gbps; however, improvements to detection technologies have direct applications to the field of quantum information as a whole.

Chapter 4 — SPAD Afterpulse Reduction

In many quantum information systems, the characteristics of the single-photon detectors directly affect the performance of the overall experiment. For example, the maximum bit rates of the QRNGs discussed in Chapter 3 are heavily limited by the SPADs dead time, the period after a photon detection when the detector is disabled. In the case of the CCQRNG, the 50 ns dead time imposes a fundamental maximum conceivable detection rate of 20 Mc/s, which is further limited by internal protection circuitry to ≈11 Mc/s. The resulting 130 Mb/s random bit-generation rate is not sufficient for many experiments, such as some high-speed quantum key distribution (QKD) schemes where random basis settings must be chosen at gigahertz frequencies [11]. This chapter describes a method for reducing the amount of necessary dead time for three commercially available silicon SPADs: the SAP-500 [82], C30902H [83], and τ-SPAD [84]. By reducing the amount of charge allowed to flow following a single-photon detection, the total afterpulse probabilities of these devices are reduced by up to an order of magnitude. This allows for a reduction in dead time, while keeping the same original afterpulsing probability, and results in a larger possible maximum detection rate [85].

4.1 Common Charge Reduction Techniques

As introduced in Section 3.3.2, for a SPAD operating in Geiger mode, a single photo-generated charge pair can result in the growth of a macroscopic current, or avalanche. During this process some proportion of the avalanche charge can become trapped in localized device defects, and if released while the device is active, can initiate another avalanche, or 'afterpulse'. These spurious detections are undesirable, so a dead time is imposed on the SPAD by means of external circuitry. During the dead time the SPAD is *quenched*, or held below its breakdown voltage for a predetermined duration, and any trapped charge released during this time does not result in an avalanche. Afterwards, the SPADs bias voltage is restored and it is again able to detect single photons. While other motivations — such as preventing thermal overheating — can contribute slightly, the duration of the dead time is primarily chosen to reduce the afterpulse probability to a tolerably low value, typically less than one percent.

It has been recognized that an effective way to reduce the total afterpulse probability is to reduce the amount of charge allowed to flow through the SPAD after an avalanche, and thus reduce the amount of filled traps that can later contribute to afterpulsing [86]. This proves to be problematic, however, as the impact ionization events occur very rapidly, and most devices are assumed to be almost completely saturated with charge on the sub-nanosecond scale [87]. This time-scale makes it extraordinarily difficult to promptly respond to avalanches with most electronics. Because the current grows exponentially, however, even modest improvements to the response time can result in significant charge reduction.

4.1.1 Passive & Active Quenching

The most common methods of curtailing the avalanche current, or quenching, involve some combination of passive and active techniques [88]. In passive quenching, the diode is placed in series with a large load resistance R_L, chosen to be much greater than the series resistance[7] of the diode R_d. During an avalanche the SPAD becomes conducting and its resistance drops by many orders of magnitude. This change in the voltage divider between R_D and R_L causes the bias across the SPAD to drop until it reaches the breakdown voltage V_{BR}. At this point the avalanche process can no longer persist, the diode stops conducting, and the bias voltage is restored on a time scale (typ. ≈ 1 μs) proportional to the RC time constant at the SPAD-R_L node. Although simple to implement, the slow recovery times of passively-quenched SPADs generally prohibit them from operating at rates above the hundreds of kHz range. Additionally, because the probability of impact ionization is proportional to the excess bias voltage (how far above V_{BR} the SPAD is biased at), a passively quenched SPAD will display a time-varying detection efficiency until fully recovered.

In active quenching the avalanche is sensed with additional electronics, e.g., a comparator, and the diode bias is altered reactively. This can be accomplished with multiple transistors to actively lower and restore the diode voltage on time scales much faster than with passive quenching, although detection events occurring near the active reset time can cause atypical device behavior [47]. If the active quenching circuitry fails, the avalanche can persist and destroy the device, so in practice both approaches are often combined to offer additional protection to the sensitive diode. A simplified schematic of a mixed quenching circuit, as well as a measured oscilloscope trace of the C30902H SPAD is shown in Figure 4.1. Ideally the SPAD would be quenched immediately after the onset of an avalanche, but the response time of this particular circuit introduces an additional ≈ 20 ns latency. This does little to reduce the total avalanche current, as the device is essentially fully saturated with charge when the active response is triggered.

[7] The series resistance of a SPAD is defined as the resistance of the device while it is conducting (avalanching). Typical values are in the 100-1000 Ohm range, while the resistance of the SPAD while not conducting is in the giga-Ohm range.

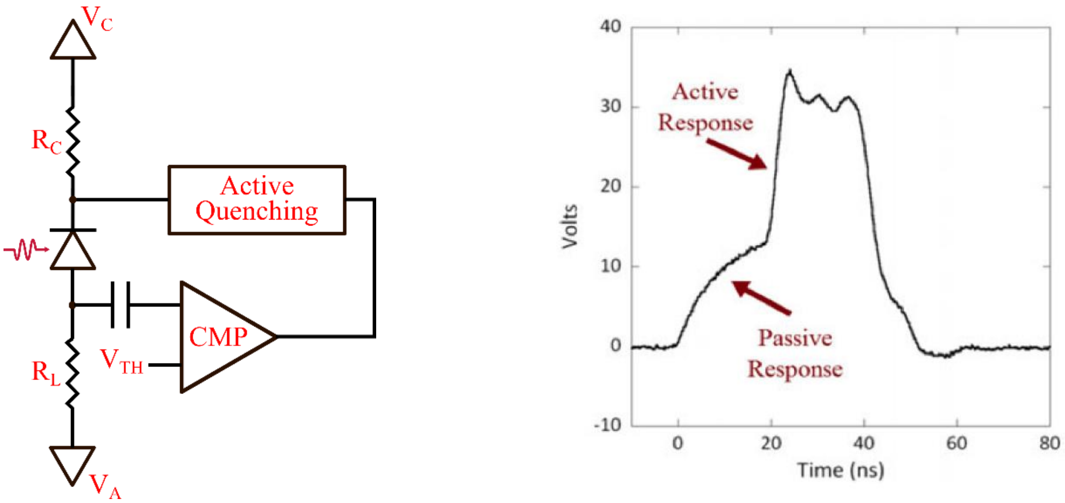


Figure 4.1. Simplified circuit schematic and measured response of typical mixed-quenching SPAD. When the AC-coupled comparator senses an avalanche, the active quenching circuitry responds by lowering the bias voltage ($V_A + V_C$) below breakdown, and then restoring it after an appropriate dead time. The 20 ns response time of this device does little to reduce the total avalanche charge, and thus has little effect on the resulting afterpulsing probability.

4.1.2 High-Speed Periodic Gating

Notable performance improvements (primarily in InGaAs/InP) have been demonstrated by rapidly gating a SPAD in a manner that strongly limits the total avalanche charge [89]–[91]. As shown in Figure 4.2a, RF frequency sinusoidal signals can be used to periodically raise and lower the bias around the breakdown voltage. In this fashion the SPAD can be quenched on sub-nanosecond timescales and significant improvements to afterpulsing probabilities have been shown.

Because the gating is periodic, if the light stimulus is asynchronous to the gating control signal, the SPAD may be inactive when a photon arrives. This excludes many applications, particularly the free-running CCQRNG of Section 3.7, unless extra synchronization efforts are made. Additionally, the ultra-short (≈ 200 ps) gate durations are on the order of the timing jitter of the avalanche itself, which can introduce a time-varying detection efficiency. In this regime of operation, the performance of the SPAD can depend on *where* in a gate period a detection is registered, as shown in Figure 4.2b.

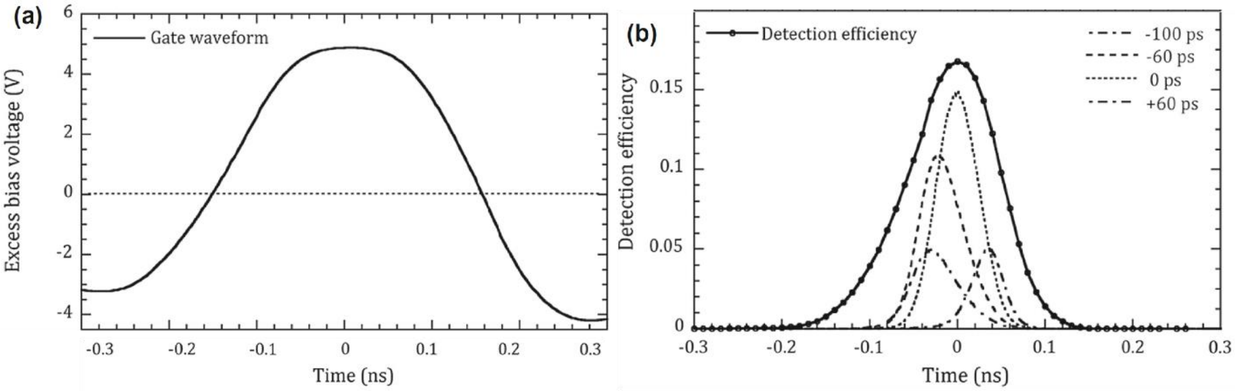


Figure 4.2. Gate bias voltage (a) and detection efficiency (b) versus time for a 1.25 GHz periodically gated InGaAs detection system. The detector is biased above breakdown (excess bias voltage > 0) for roughly 320 ps. A short (< 30 ps) optical stimulus is stepped through the gate to measure detection efficiency, and also shown are four time-correlated single-photon counting histograms for four different times relative to the peak. The timing jitter and detection profile clearly change based on *when* in the gate the photon is absorbed *[47]*.

As of yet, periodic gating approaches have mostly been employed with InGaAs/InP SPADs, and rarely with Si devices. Due to their III-V composition, InGaAs/InP devices have a higher concentration of impurities, and generally much higher intrinsic afterpulsing than their Si counterparts. Using a clean Type-IV material reduces the total number of traps and even allows high detection efficiency in the near-infrared regions, but the added thickness of the absorption region requires much larger bias voltages (≈ 20 – 30 V) to properly quench; separating the relatively small avalanche signal from the large gate transient at the input to the comparator remains a challenge. Finally, the electrical packaging for commercially available Si SPADs typically has much higher capacitance than available InGaAs devices. This makes it difficult for researchers without access to fabrication facilities or bare devices to design and test RF-frequency gating circuits.

4.2 Periodic-Mixed Quenching

The passive-then-active quenching approach of Section 4.1.1 provides little charge reduction with conventional electronics. The periodic gating approach of Section 4.1.2 has been shown to result in excellent charge reduction, but has not been significantly explored at high gating frequencies for Si-based devices due to the difficulty of achieving the higher necessary bias swings on sub-nanosecond time scales. As advances in electronics are made, however, these RF-gating approaches may become plausible. In this section a hybrid periodic-mixed quenching (PMQ) approach is introduced to explore the possible charge reduction effects of RF-frequency avalanche quenching.

The circuit schematic of the PMQ system is shown in Figure 4.3. This circuit uses a load resistance R_L = 200 kΩ, similar to the passive-then-active quenching system shown in Figure 4.1. Feedback that would normally condition the application of an active quench cycle upon sensing an avalanche has been removed and has been replaced by periodic control signals from synchronized pattern and function generators. Regardless of whether or not an avalanche has occurred, these signal generators will induce two quench cycles (quench, dead time, and reset), as illustrated in Figure 4.3.

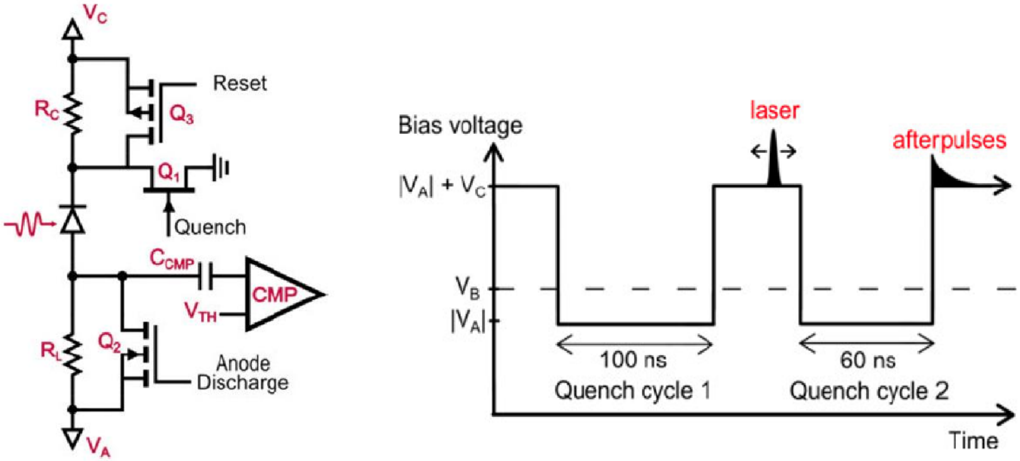


Figure 4.3. Simplified schematic of PQM periodic-quenching circuit (left) and timing diagram of the afterpulse characterization measurement (right). During both quench cycles the voltage across the SPAD is quenched below breakdown to V_A, and then reset back to $V_A + V_C$. Quench cycles are repeated every 2 µs and placement of the laser stimulus allows for afterpulse measurements of arbitrarily short quenching latencies *[85]*.

When fully armed, the voltage across the SPAD is V_C - V_A, where V_A is negative, in magnitude a few volts smaller than the breakdown voltage V_{BR}. The value of V_C is chosen such that the SPAD is biased with an appropriate amount of excess bias voltage while active, and is $|V_A|$ during the dead time. To achieve fast voltage transitions for the quench we use a wideband GaN transistor [92], denoted by Q1 in Figure 4.3, to lower the cathode to the ground potential for the duration of the holdoff. This device is capable of a fast slew rate (> 25 V/ns), that rapidly lowers the SPADs bias to below V_{BR}. While much faster than typical switching MOSFETS, its drain-source leakage current is significantly larger and while inactive it still has a relatively low impedance (≈ 50 kΩ), making it unsuitable for use at a high impedance node. For this reason, a small R_C = 200 Ω is placed at the SPAD cathode and a secondary lower-speed switching MOSFET Q2 is used to discharge the anode shortly after the quench is applied to the cathode. After the duration of the dead time another transistor, Q3, resets the cathode voltage to V_C. Finally, a high-speed comparator is

AC-coupled to the anode to sense avalanches for measurement, and a latch control signal is used to make the comparator insensitive to the quench and reset signal transitions that couple through the diode.

The measurement period of the system is approximately 2 μs, and for the majority of that time the SPAD bias is fully charged unless there is an avalanche. Photon detections occurring outside of a quench cycle, such as dark counts or afterpulses, are quenched passively and the SPAD slowly recharges through R_L. To ensure that the diode is in a well-known state (and not recovering due to a previous passively quenched event), an initial cycle (Quench cycle 1) is applied. The interval between the end of the first and the beginning of the second quench cycle forms a 50 ns window in which an attenuated laser pulse is applied at a time determined by the pattern generator. The temporal delay of the optical pulse can be adjusted arbitrarily, providing complete control of the interval between the onset of an avalanche and the application of a quench transition. After the end of the second quench cycle, afterpulses are recorded over the remaining 1.79 μs, conditional on a detection event due to the laser. All detection events are recorded with 100 ps timing resolution by the custom Virtex-6 FPGA-based time-tagging system discussed in Chapter 5. Recorded events are passed over a PCIe bus interface for continuous analysis on a computer.

4.3 Theoretical Modeling

To determine the expected amount of charge reduction through promptly quenching an avalanche, accurate estimations of both the avalanche current and characteristic trap lifetimes within the SPAD must be made. Without access to fabrication details, estimating these internal parameters can be difficult, but we find excellent agreement with data taken from relatively straight forward measurement techniques.

4.3.1 Avalanche Current

To model the expected avalanche current flow we use the simple circuit-level model given in [85], and shown in Figure 4.4. In this model the SPAD is approximated as a parallel combination of its series capacitance C_D and a time-dependent series resistance $R_D(t)$ that changes from many GΩs when not avalanching to a smaller (typ. 300-1000 Ω) series resistance upon photon detection. Accurate measurements of the additional capacitances at the anode (C_{AS}) and cathode (C_{CS}) due to the surrounding circuit and electrical packaging must also be made. The avalanche build-up process has been modeled previously with a complex time-dependent behavior, but for this measurement simply assuming that the series resistance decays exponentially at a time scale much faster (≈ 50 – 100 ps) than the subsequent RC decay give excellent agreement with experimental results.

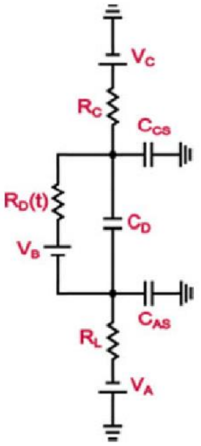


Figure 4.4. Equivalent SPICE model of a SPAD circuit *[85]*.

The series resistance R_D is measured by reducing the load resistance R_L to 1 kΩ (a value that will not cause passive quenching), and disabling the reset at the end of Quench cycle 2. In this mode the SPAD is only biased above breakdown for the short interval between the two quench cycles and when an avalanche occurs the voltage at the cathode drops to a value determined by the voltage divider between R_C and R_D. The beginning of Quench cycle 2 terminates the flow of current, and this steady-state cathode voltage is measured with a high-impedance probe.

The series capacitance C_D of typical reach-through SPADs is small (< 5 pF), and must be measured when the device is biased near (but not past) the breakdown voltage. For accuracy C_D is measured using two different techniques in independent test circuits. In the first, the RC time constant of the cathode recharge through a known resistance is measured after the abrupt turn-off of a switching MOSFET. In this measurement the additional capacitance of the test circuit and device packaging was measured using an empty package of the same type, and was subtracted off to extract C_D. In the second method a sinusoidal signal at various frequencies ranging from 100 to 300 MHz was ac-coupled to the anode, and the cathode was connected to a spectrum analyzer. With knowledge of the sinusoidal amplitude at the anode, the capacitance was determined from the signal strength measured by the analyzer. The results of both methods showed good agreement, and the values obtained in the latter method, as well as their operating conditions are shown later in Table 4.2.

Finally, SPICE modeling of the circuit in Figure 4.4 results in the diode currents of Figure 4.5, for the three tested commercial devices. The current profiles for the SAP-500 and C30902H are similar, and were much faster than the τ-SPAD. The time scales suggest that in order to achieve significant charge reduction these devices must be fully quenched in the few-nanosecond regime, whereas for the τ-SPAD much larger reductions can be achieved with quenching times at slower time scales.

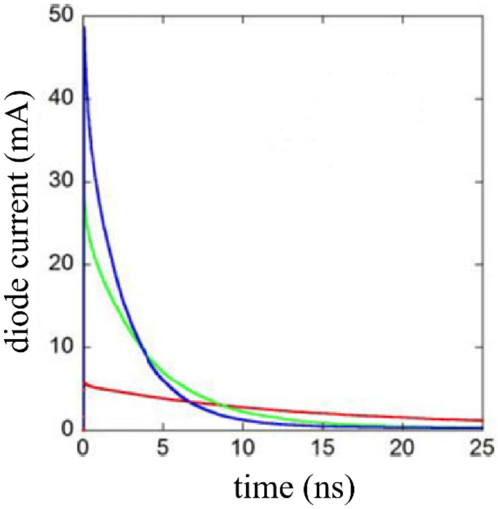


Figure 4.5. *Theoretical avalanche diode currents for the C30902 (green), SAP-500 (blue), and τ-SPAD (red)[85].*

4.3.2 Charge & Afterpulse Reduction

To model the expected afterpulse reduction as a function of quench delay we start with the assumption that the observed afterpulse probability is proportional to the number of charge carriers *N* that are trapped in the device when it is reactivated at the end of the dead time [86]. SPADs (and semiconductors in general) often exhibit various trap levels [93], and the total number of trapped charges is modeled as $N = \sum_i n_i$, where n_i is the population of the *i*th trap level. For each trap level we solve the rate equation

$$\frac{dn_i}{dt} = a_i \frac{I(t,t_Q)}{q_e} - \frac{n_i}{\tau_i} \qquad \text{(Eqn. 4.1)}$$

Here $I(t,t_Q) = I(t)(1 - H(t - t_Q)$ is the avalanche current flowing through the SPAD as a function of time, starting at t = 0 and terminating at the quench delay t_Q (H(t) is the Heaviside function), q_e is the electron charge, and τ_i and a_i are the lifetime and relative weighting of each trap level *i*, respectively. Qualitatively, this rate equation expresses two competing terms: the population of traps due to current flow, and the depopulation of traps according to their natural lifetime. The time-dependent current flowing through the SPAD, I(t), is calculated from a simulation of the circuit in Figure 4.4, following the approach in the preceding section.

The trap lifetimes τ_i and their relative weighting a_i for each device were extracted from measurements of the afterpulse probability versus dead time. The relative weighting term is related to the trap density and the volume of the active region, parameters which were unavailable for our commercial devices. To perform this measurement an optical pulse is positioned ≈ 5 ns from the start of Quench cycle 2 (Figure 4.3), and the total afterpulse probability is measured as the duration of the dead time is varied from 10 ns to 1 µs. The dead time was controlled with the pattern generator and the afterpulse probability recorded with the time-tagging system described Chapter 5. The resulting afterpulse probability is fit with a multi-term exponential of the form $\sum_i a_i \exp(-\frac{t}{\tau_i})$. This procedure was carried out for all three SPADs, and the lifetimes and relative weightings are listed in Table 4.1. The C30902 required three trap levels to accurately fit these data, and the shortest lifetime agrees with the measurement reported in [94]. The SAP-500 required only two trap levels, and the τ-SPAD required only one for accurate fitting. Because the dead time was only measured out to 1 µs, the uncertainty of the longer trap lifetimes is higher.

***Table 4.1**. Measured trap lifetimes and their relative weightings for each SPAD. Uncertainties indicate one standard deviation.*

SPAD	τ_1 (ns)	a_1	τ_2 (ns)	a_2	τ_3 (ns)	a_3
C30902	9.06 ± 0.10	0.99 ± 0.04	689.4 ± 77.5	0.01 ± 0.004		
SAP-500	16.39 ± 0.43	0.70 ± 0.01	71.63 ± 4.13	0.22 ± 0.03	485.3 ± 5.1	0.07 ± 0.05
τ-SPAD	58.88 ± 1.47	1				

Equation 4.1 is solved numerically for n_i using the parameter of Table 4.1 and the SPAD circuit model of Figure 4.4, for each value of the quench delay t_Q. The resulting individual n_i are then summed to give the total number of trapped charges at the end of the dead time N, as a function of t_Q. An example of the afterpulsing vs. hold off measurement for the C30902 is shown in Figure 4.6. We assume the relationship between the total number of trapped charges and the afterpulse probability to take the form of a simple

multiplicative constant, and with this single free parameter the model agrees very well with the observed afterpulse probability.

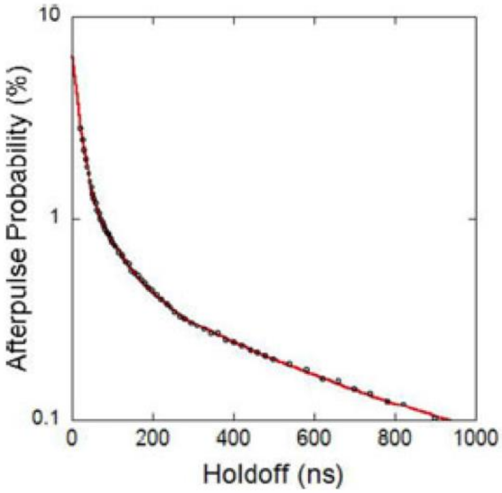


Figure 4.6. Total afterpulse probability versus holdoff for the C30902 at -10 °C, V_E = 15V, and a quench delay of ≈ 5 ns. By fitting multiple decaying exponentials (red) to experimental data (black), trap lifetimes and relative weights are extracted *[85]*.

4.4 Experimental Results

The periodically-quenched method described earlier in Section 4.2 was used to study the effects of prompt quenching in three different commercially available reach-through SPADs. The C30902 and SAP-500 were housed in a transistor-outline package, while the τ-SPAD was removed from its module for testing. All SPADs were operated at a temperature of -10 °C. The laser was an 850 nm gain-switched VCSEL which produces < 30 ps pulses that are coupled to a single-mode fiber, attenuated to low mean photon number, and focused on the SPADs active area. A 10 nm bandpass filter was also placed in front of the SPAD to suppress background photons. The mean photon number of the optical pulse was adjusted to produce a count rate of approximately 50 kHz, and was always less than 0.3 photons per pulse.

Each SPAD has a different design and detection efficiency at 850 nm so it is not appropriate to choose a single overvoltage at which to operate all three devices, as the amount of excess bias voltage required may put some SPADs out of their normal operating regime. For this reason, an excess bias voltage in the upper range of where the device is typically operated was chosen: 15 V for the SAP-500 and C30902, and 7.5 V for the τ-SPAD. The breakdown voltage and background (non-illuminated) count rate was experimentally measured, and along with the measured resistances and capacitances mentioned in Section 4.3.1, are listed in Table 4.2.

***Table 4.2**. Circuit parameters for each SPAD measured at (-10 ± 0.1) °C. V_{EX} is the excess bias voltage and Back is the number of counts measured when the laser is turned off; R_D, V_{BR}, and C_D are the series resistance, breakdown voltage, and series capacitance of each particular SPAD. Uncertainties in R_D and V_{BR} represent one standard deviation, while for C_D it encompasses the full range of variation observed over all measurements.*

SPAD	R_D (Ω)	C_D (pF)	V_B (V)	V_{EX}(V)	Back (s^{-1})
C30902	525 ± 25	1.85 ± 0.2	216 ± 0.1	15	2000
SAP-500	315 ± 15	2.27 ± 0.5	398 ± 0.1	15	1000
τ-SPAD	1300 ± 65	1.66 ± 0.2	111 ± 0.1	7.5	200

4.4.1 Observed Afterpulse Reduction

To characterize afterpulsing probability as a function of quench delay, the arrival time of the optical pulse is moved incrementally towards the beginning of Quench cycle 2, starting at 25 ns before the quench, and ending when avalanche events can no longer be detected by the comparator. At each quench position, all counts recorded after the end of Quench cycle 2, conditional on a detection event having previously occurred at the arrival time of the optical pulse, are recorded. The background count probability (see Table 4.2) is subtracted from the measured conditional count probability, and gives the final afterpulse probability. The resulting afterpulse probabilities versus quench delay for the three SPADs are shown below, as well as the predictions from the numerical model described in Section 4.3.2.

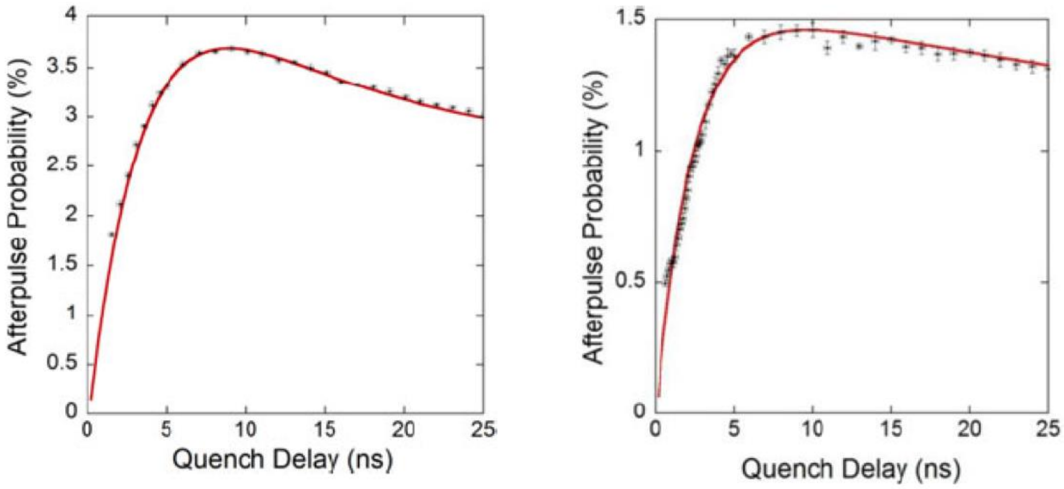


***Figure 4.7**. Afterpulsing versus quench delay for the C30902 (left) and SAP-500 (right); model (red line) and data (black dots). Error bars represent one standard deviation. The avalanche signals were sensed with a 5 pF capacitor.*

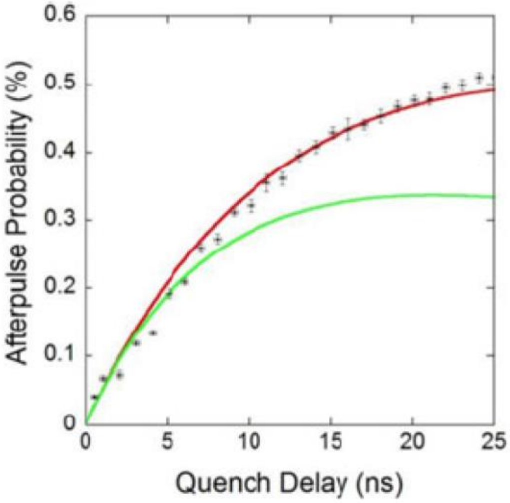


***Figure 4.8**. Afterpulsing probability versus quench delay for the τ-SPAD; model (red line) and data (black dots). Due to the smaller avalanche signal, the avalanche was sensed with a 10 pF capacitor. A theoretical estimate of the afterpulsing behavior with a 5 pF is shown in green. Error bars represent one standard deviation.*

As shown in Figure 4.7 and Figure 4.8 the afterpulsing probability exhibits a sharp decrease at short quench delays. This behavior is directly related to the current profile *I*(t), which as discussed in Section 4.3.1 is assumed to rise in sub-nanoseconds, a time scale significantly shorter than the minimum achievable quench delays used here. The minimum useable quench delay varies from one SPAD to another, but it is determined primarily by the lowest operable comparator threshold, which in turn is determined by the ringing from the control signal transitions.

As shown in Table 4.2, the product of the series resistance and capacitance for the SAP-500 and C30902 are of comparable value, and both devices were operated with a 5 pF sensing capacitor on the comparator at the anode. As a result, the current profiles for these devices are similar (Figure 4.5), and hence the quench delays over which the afterpulsing is sharply reduced are also similar. However, the magnitude of the afterpulse probability in these devices differs by more than a factor of two. This may be due to a difference in the trap density, or to a difference in the active volume of the devices.

In contrast, the τ-SPAD exhibits a significantly lower slope in its afterpulse probability versus quench delay (Figure 4.8), which agrees with the significantly larger measured series resistance and the slower decay of the theoretical current profile. Additionally, the avalanche signal was smaller for this device and required the use of a larger (10 pF) sensing capacitor at the anode, which further increased the time constant of the current decay. The projected performance of the τ-SPAD with a 5 pF capacitor is also included in Figure 4.8, but nonetheless it exhibited the lowest overall afterpulse probability as well as the largest relative reduction in afterpulsing. The lower overall magnitude is likely due to unavailable design parameters but the larger relative reduction can be attributed to the slower decay of this device's current profile. For the C30902 and the SAP-500, most of the current has already flowed after ≈ 5ns, but for the τ-SPAD this process takes much longer, allowing for larger reductions with more modest quench delays. These data show that through prompt quenching it is possible to lower the afterpulse probability by factors of at least 2, 3, and 12, for the C30902, SAP-500, and τ-SPAD, respectively.

4.4.2 Corresponding Reduction in Dead Time

To determine how much an improvement in afterpulsing would reduce dead time, we assume that the afterpulse probability decays exponentially, and that a time t_{DT} is the dead time necessary for the afterpulse probability to decay to some value AP (e.g., 1%): $AP = Ce^{\frac{-t_{DT}}{\tau}}$, where C is some constant related to the total amount of charge.

If, with prompt quenching, the afterpulse probability can be reduced by a factor of X, then the dead time can be shortened to $t_{DT} - t_{SH} = \tau \ln(X)$, at which point the afterpulse probability is once again AP.

For example, with a dead time of 60 ns, the SAP-500 displays its worst-case afterpulse probability (≈ 1.45%) when the quench delay is 8 ns. The afterpulsing is reduced (see Figure 4.7) by a factor of ≈3 with prompt quenching. According to Eqn. 4.3 and considering only the dominant trap level, the holdoff could be reduced from 55 ns to 42 ns before the afterpulse probability would exceed the value it had with an 8 ns quench delay; this implies a marginally higher device saturation rate. The afterpulsing reduction for the τ-SPAD is reduced by a much larger factor (≈ 12x), and highlights the strong relationship between dead-time reduction and current flow. In order to achieve significant reductions in dead time, the quenching must be applied at time scales significantly shorter than the decay of the avalanche current profile. This requirement is challenging, and although only modest gains were reported here, more significant dead-time reductions could be achieved with improved circuit design or faster components.

4.5 Secondary Benefits of Afterpulse Reduction

Although acceptable (< 1 %) afterpulsing probabilities can be achieved with easier quenching methods, there are other advantages which can arise by reducing it even further. Increasing the excess bias voltage above breakdown increases the electric field, which raises the velocity at which carriers are swept across the device. This has the positive effect of increasing both the probability of impact ionization (which increases the detection efficiency), as well as increasing the timing resolution due to faster avalanche build-up. Unfortunately, increasing the overvoltage also increases the dark count rate. The increased multiplication raises the probability of a thermally-generated carrier causing an avalanche, as well as other interesting mechanisms which contribute, such as tunneling into the multiplication region and trap-assisted tunneling due to mid-gap defects [47].

Without any additional charge reduction effort, increasing V_{EX} would quickly result in unacceptable noise levels. However, by promptly quenching the SPAD and reducing the avalanche charge, the device can be operated at a higher V_{EX} and still maintain the original level of afterpulsing (without prompt quenching). Figure 4.9 displays the total dark count rate versus excess bias voltage of the C30902 for an operating temperature of -10 °C. Note that regardless of charge reduction, the device cannot be operated above ≈25 V of excess voltage, at which point the noise levels sharply increase. This effect occurs at roughly the same excess bias voltage regardless of temperature, and could be due to the high excess bias voltage increasing the depletion regions by enough such that the low-gain absorption and high-gain multiplication regions overlap, or by secondary breakdown effects.

In Figure 4.10 the detection efficiency and timing resolution of the C30902 is also shown, for excess bias voltages beginning at its typical operating range of 10 V and not exceeding 25 V. As shown, the detection efficiency increases from ≈ 37 % to ≈ 53% while the full-width-half-max of the measured timing resolution decreases from ≈ 1.1 ns to ≈ 450 ps. For the random number generation methods of Chapter 3 these metrics are largely irrelevant, as the largest improvement to bit-generation rate can be gained from faster detection speeds. However, for experiments such as a Bell test [17] where current repetition rates are in the kHz range, these metrics are more significant.

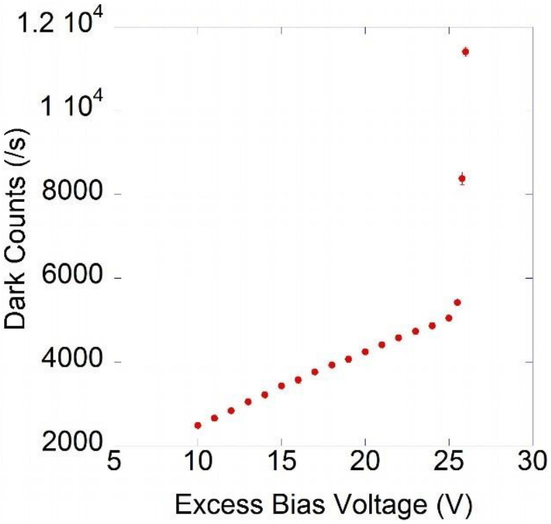


***Figure 4.9**. Measured dark count rate (s^{-1}) versus excess bias voltage (V) for the C30902 operated at -10 °C. Excess bias voltages exceeding 25V result in greatly increased background noise rate. Error bars represent one standard deviation.*

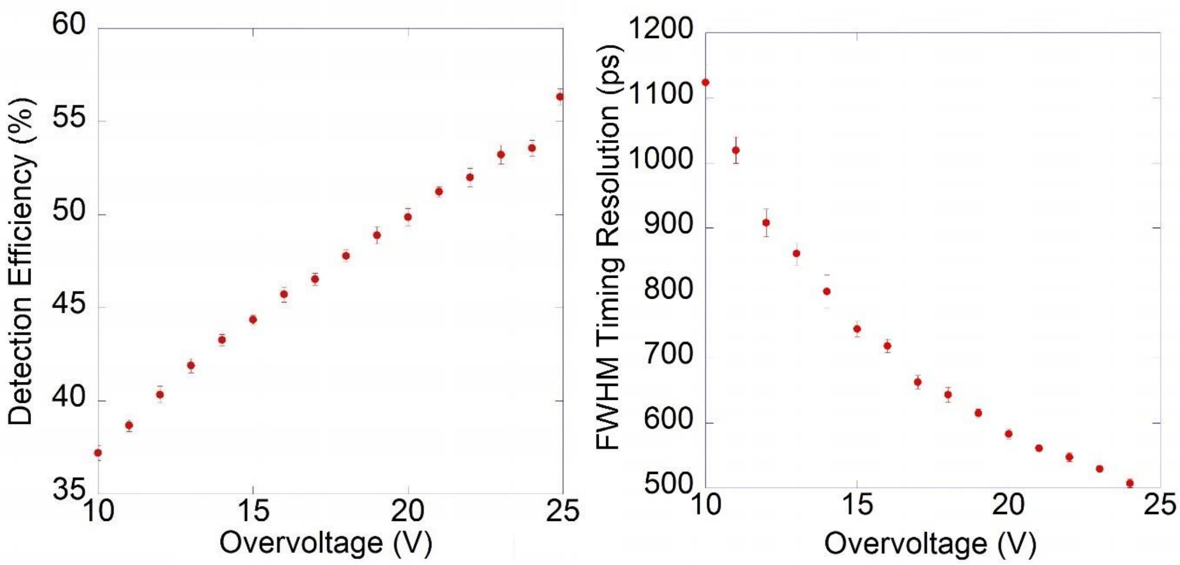


Figure 4.10. Detection efficiency (left) and timing resolution (right) for the C30902 operated at -10 °C. A higher excess bias voltage results in a higher probability of multiplication, and increased detector performance. Error bars represent one standard deviation.

4.6 Conclusions

In conclusion, this chapter has described the potential benefits of reducing avalanche charge by applying prompt quenching to a variety of commercially available reach-through SPADs. The data show that significant reductions in overall afterpulse probability are achievable if quenching is applied at time scales significantly shorter than the time scale of the current decay. This requirement is challenging, and the resulting increase in maximum count rate is likely to be modest, due to the logarithmic relationship between afterpulsing and dead time. Nonetheless, the results show that further improvements may be achievable with faster electronics. Additionally, this afterpulsing method can also allow operation of the SPAD under higher excess bias voltage, which may improve other performance metrics such as timing resolution and detection efficiency; detector dark counts, however, would be increased under higher bias.

Chapter 5 — Time-to-Digital Conversion

The ability to measure the arrival-times of events with high precision and at high frequencies is necessary in many areas of science, such as positron emission tomography (PET), time of flight measurements for high-energy physics experiments, front-end sampling for oscilloscopes and measurement equipment, and of relevance for this dissertation, single-photon detector characterization and quantum random number generation.

The first time-tagging device was invented in 1940, a coincidence detector circuit for estimating the mean lifetime of mesotron decay [95]. Devices created in this period utilized exclusively analog components, and are referred to as time-to-amplitude converters (TACs). It was later realized that more robust measurements could be taken with digital electronics, so recent time-taggers are generally referred to as time-to-digital converters (TDCs). While TACs can achieve very good timing resolution, TDCs are much less sensitive to external influences, e.g., temperature variations. Digital circuits are also much easier to implement in ASIC ICs , as well as requiring much less area when designing multiple-channel devices.

In this chapter a brief history of early time-tagging systems is presented, along with recent improvements to some of the approaches. The theory behind a system we developed specifically for quantum information applications is introduced, as well as details on its implementation, relevant performance characteristics, and sample data. Due to the designed system's capabilities, we also developed a new higher-order SPAD characterization technique. This new technique allows for the discovery of previously unknown effects with a straight-forward analysis of recorded time-tags, and will allow for a more complete SPAD characterization, a process crucial for high-accuracy quantum information experiments. Our results are being prepared for submission for publication.

5.1 Analog Time-to-Digital Converters

Although the bit-value output is ultimately digital, many early TDC systems used analog signal-processing techniques for the time-interval measurement. In one such system (Figure 5.1), start and stop events are input to a pulse generator which forms a digital pulse of width proportional to the time-interval between them. This time-modulated pulse is integrated by an analog op-amp based circuit, which ideally forms a voltage ramp. Upon the arrival of the stop-event trigger, the ramps magnitude is translated by the analog-to-digital converter (ADC) to give the final time interval in digital bit form.

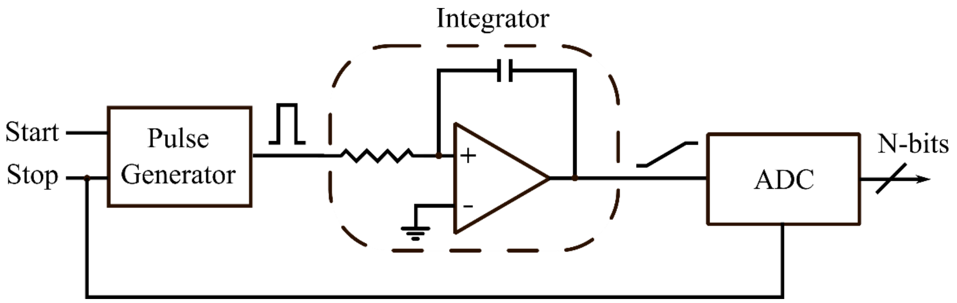


Figure 5.1. Illustration of an early analog-based TDC. A digitally-formed voltage pulse is integrated and input into an ADC, which outputs an N-bit time-tag corresponding to the interval between the start and stop pulses.

TDCs of this type have several disadvantages that can significantly degrade performance. The pulse generator, integrator, and ADC must all exhibit perfect linear behavior in order for the distribution of timing measurements to also remain linear. The finite bandwidths of the pulse generator and operational amplifier can cause frequency-dependent measurement errors, and the bandwidth of the operational amplifier can significantly limit the minimum measurable time interval. All semiconductor-based electronics are susceptible to non-linearity at some level, but many analog components can be especially sensitive. In this particular example, the resistance and capacitance values of the RC-integrator must be precisely known to determine the timing resolution. As these components are all temperature-sensitive, frequent calibration is often necessary.

ADCs are typically fairly linear devices, but they have a finite N-bits of output range. Therefore, a design choice must be made between the maximum measurable time interval (when the ADC rolls over), and the desired minimum timing interval. For sub-nanosecond time-resolutions, even a 16-bit ADC will only allow measurement of intervals shorter than a few microseconds. In one more advanced approach [96], two parallel systems are used to integrate up and down periodically. When the output of both integrators are equal (measured by a voltage comparator), they are reset and an additional digital counter is incremented. This separation of the timing method into coarse and fine portions is referred to as the Nutt Method [97], and is used by many modern TDC approaches. The coarse component can be a simple digital counter, operating at a low frequency, while the fine component is tuned such that its maximum timing interval is equal to one period of the coarse component. While these two systems running in parallel can ease the maximum time interval disadvantage, TDCs of this type will always suffer from the finite-bandwidth issues stemming from their analog components. Non-linearity can be measured with a thorough characterization of the physical system, but this can be a tedious process. For this reason, the designs of recent TDC implementations have been almost exclusively digital.

5.2 Early Digital Time-to-Digital Converters

Designed explicitly to correct for slight non-idealities in analog-signals, digital-based circuitry is well-suited for the robust and repeatable measurements needed for precision time-interval measurement. In this section some of the earlier approaches are discussed. Although later systems improved on these designs with integrated ASICs or FPGAs, the underlying time-conversion mechanisms remain largely the same.

The most basic digital-TDC approach is to use an N-bit binary counter which increments upon every rising-edge of a high-speed clock, as shown in Figure 5.2. The first incarnation of the quantum random number generator in Chapter 3 used this method, achieving 5 ns timing resolution with a 16 bit digital counter and a 200 MHz clock. The finite range of the counter allowed for a maximum measurable time interval of approximately 327 µs, before reaching 2^{16} and the counter rolled over to zero. With FPGAs available at that time, increasing the counters bit-width past 16 would not meet the FPGAs timing specifications. Even with today's smaller fabrication processes, FPBA-based digital counters will not greatly exceed an 800 MHz count rate with sixteen bits of resolution. Additionally, because the input is totally asynchronous to the counter's clock, an error up to one clock cycle can be introduced with each measured interval.

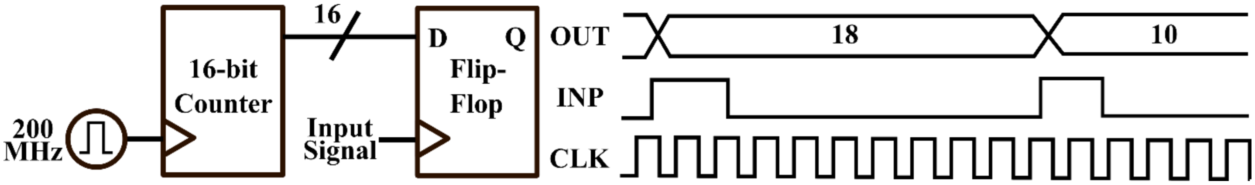


Figure 5.2. Illustration of simple counter-based digital TDC. The output of the 16-bit counter is latched by the D-type flip-flop upon the arrival of an input signal. The counter is reset in the illustrated logical waveform to give absolute timing measurements, but it can also be free-running and differences extracted in post-processing.

To help alleviate both the maximum timing interval issues arising from finite counter widths and the minimum timing resolution from clocking requirements, a Nutt Method-based approach similar to the dual-integrator of Section 5.1 is often used. One such implementation is shown in Figure 5.3, where the fine portion of the measurement is made by sampling a chain of digital buffers (or inverters) with flip-flops. Upon the arrival of a start pulse, the signal propagates down the buffer chain. Digital buffers are fairly simple devices, and can have very low propagation times. When the stop pulse arrives, it triggers the simultaneous clocking of the flip-flops, and their output value forms a thermometer code (e.g., 11111111000000) corresponding to how far down the chain the signal was able to travel. If no stop signal has arrived by the end of the chain (which is tuned to be one increment of the coarse counter), the chain is reset and the signal starts from the beginning. Assuming the counter increments every Δt_c (resulting in a bit-value N), and the propagation delays of the simple buffers to be identical and equal to Δt_f, then the final time bin measurement Δt can be calculated by the equation $\Delta t = N * \Delta t_c + \Delta t_{f.}$ By transferring much of the workload from the buffer chain to the coarse digital counter, the maximum measurable timing interval can be greatly increased. However, with discrete IC components, systems of this type cannot reach the desired few-picosecond resolution or hundreds of MHz maximum detection speeds required of many quantum information experiments.

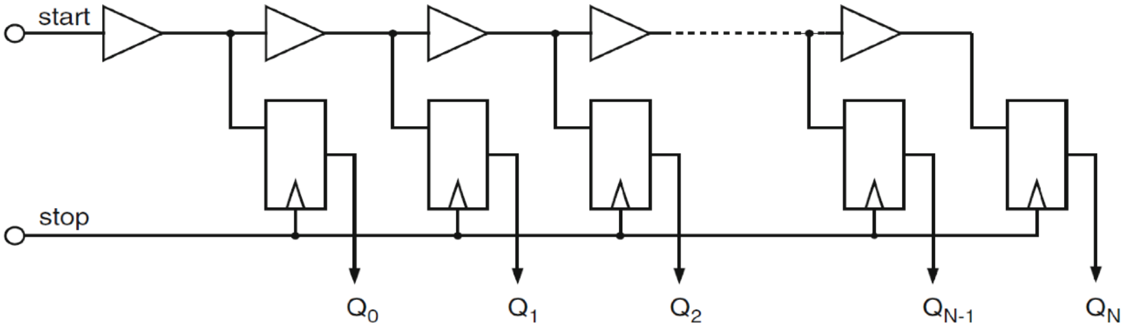


Figure 5.3. Simple buffer-based digital TDC. The fast propagation delay of the digital buffers allows for very fine timing resolution; complemented with a lower-speed digital counter (not pictured), this greatly extends the maximum timing interval of earlier approaches. Figure courtesy of *[98]*.

5.3 Novel Sub-Gate-Delay Time-to-Digital Converters

As of yet, the TDCs covered in this chapter have been limited to a timing resolution equal to the propagation delay through a single component, i.e., an inverter or a digital buffer. Designs of this type are fundamentally constrained to the fastest available digital technology, and will only improve with corresponding fabrication processes. For example, the delay of a digital CMOS gate is linearly proportional to the transistor width, and as current semiconductor-based transistors approach widths near the size of actual atoms themselves, estimates put Moore’s Law on a timeline of one to two decades. In this section we explain two approaches which use clever design techniques to achieve timing resolution better than a single component, or sub-gate-delay TDCs (SGD-TDCs).

5.3.1 Vernier Delay Line

One of the most popular SGD-TDC methods, and a modification of the delay line of buffers discussed previously, the Vernier method can achieve sub-gate delay timing resolution by placing multiple delay lines in parallel, as shown in Figure 5.4. Each delay line receives a different input (start or stop), and are constructed using components with differing delays. The start and stop paths are comprised of buffers with delay τ_1 and τ_2, where $\tau_1 > \tau_2$. The stop signal ‘catches up’ to the start with every logic element it passes through, by a value $\tau_d = \tau_1 - \tau_2$. Neglecting variations between individual components and assuming τ_d to be constant, τ_d then sets the timing resolution of this type of design. Eventually the two signals are phase aligned, a condition which can be detected by a single flip-flop, and given N elements before phase alignment occurs, the time difference between the two is given by $N*\tau_d$.

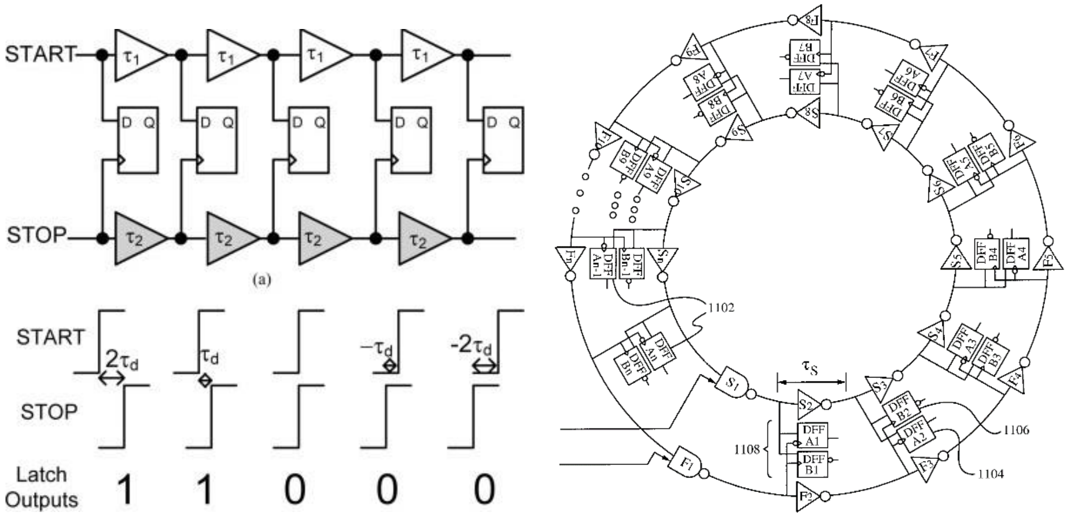


Figure 5.4. Illustration of Vernier delay line in serial (left) and ring (right) configurations. Because *$\tau_1 > \tau_2$,* the slight difference in each path’s propagation delay causes a phase difference which can be measured by flip-flops. Longer time-interval measurements require numerous components in the serial configuration, a constraint lessened in the ring configuration. Figures courtesy of *[99], [100]*.

With careful tuning of τ_d, the timing resolution of Vernier delay Lines can reach very low values, with typical systems residing in the 3-50 ps range. As τ_d decreases, however, the number of elements required to construct a chain capable of measuring longer time intervals increases. For the trap characterization application of Chapter 4, accurately measuring the longest trap lifetimes requires a maximum delay in the few-µs range, an interval which would require hundreds of thousands of delay elements in series. This heavily constrains the required area, as well as increasing the power requirements of such a circuit substantially. Recent implementations have improved performance by arranging the delay lines in a ring configuration [100], or even a 3d structure [101]; connecting the beginning and end of the delay chains together and adding an additional coarse counter allows elements in the chain to be reused. This application of the previously discussed Nutt Method allowed for an 5 ps timing resolution in one such system [99], a value highly-competitive with other SGD-TDCs.

5.3.2 Pulse-Shrinking TDC

Another notable approach that can achieve sub-gate delays with digital logic is the pulse-shrinking technique, as shown in Figure 5.5. In TDCs using this design, the first stage is a module which forms a single pulse, with rising- and falling-edges defined by the arrival of the start and stop signals respectively (similar to the pulse-forming module in Figure 5.1). Afterwards, the resulting time-modulated pulse travels through a delay chain in which each element reduces the original width by an amount equal to the timing resolution of the TDC, τ_d. When the pulse has fully disappeared, the timing information can then be extracted by counting the number of elements traversed.

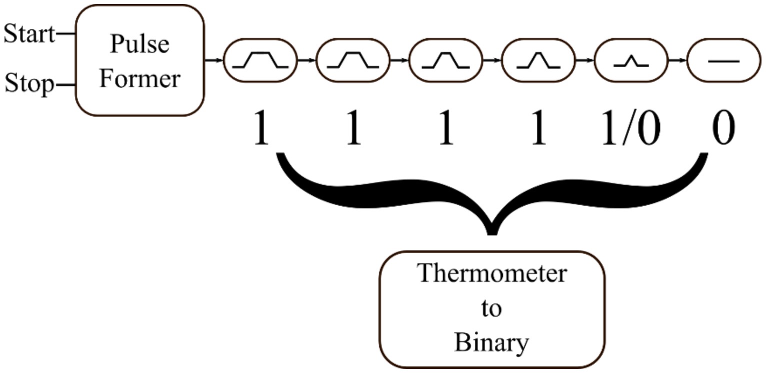


Figure 5.5. Illustration of pulse-shrinking TDC operation. An initial pulse is formed with width proportional to the time between the start and stop pulses. At each stage the pulse is shrunk by τ_d until it has disappeared. Voltage levels near the edge can be nominal and are prone to glitches.

Pulse-shrinking TDCs can be both power and area intensive, but this can be lessened with a ring approach. The delay chain output is a thermometer code, which must be converted to a straight binary representation. Because the final pulses tend to be very small and may not exceed the flip-flops required input voltage ratings, values near the detection edge are prone to glitches.

Pulse-shrinking TDCs suffer from the interesting disadvantage that the latency from pulse detection to time-tag output is directly proportional to the measured interval time itself, since the initial pulse-forming circuit must wait until the stop signal before starting its output, which makes interfacing with data-recording hardware difficult. Because of this, pulse-shrinking TDCs typically have very large dead times and are not as common as Vernier-based systems. One of the more recent implementations [102] was able to achieve 6 ps timing resolution, but was only able to record a maximum timing interval of only 4.5 ns.

Both of these SGD-TDC designs could theoretically achieve arbitrarily high timing resolution by placing multiple systems in parallel, but this can pose additional problems. Driving a large amount of digital elements with a single signal places a very large capacitive load on the circuit, directly limiting the rise- and fall-times of clocking signals required to store the data. Data throughput requirements for large parallel systems can also quickly outpace most computer's data storage or recording capabilities. The necessity for both low-capacitance integrated systems and a platform on which to perform large amounts of repeatable data processing has accelerated a shift from discrete IC-based systems to the much more compact FPGA-based systems, as discussed next.

5.4 FPGA-based TDCs

A Field-Programmable Gate Array (FPGA) is a collection of logic gates, flip-flops, interconnects, and other components, all in a single integrated circuit chip. The FPGA is designed to be configured, usually with a hardware description language (HDL), by a programmer to fit their desired needs. Modern FPGAs can contain a large amount of logic, with one commercially available chip having roughly two million logic cells, 68 Mb of integrated block ram, and 1,200 individual I/O pins [103]. This dense concentration lends itself to many TDC approaches, as thousand-component-long delay lines can be constructed with relative ease, the propagation delay between each cell is much smaller than would be allowed otherwise, and power consumption is much less. Much progress has been made in FPGA-based TDCs and both Vernier lines and pulse-shrinking type TDCs have already been implemented. Vernier lines in particular have seen great improvement, with one implementation having 1.58 ps timing resolution and an operating range of 59.3 minutes [104].

Internally, FPGAs can be depicted as arrays, with a grid composed of individual logic cells. In the architecture of Xilinx FPGAs [103], the array is composed of logic slices, with each slice containing four logic cells. The slices are connected in both horizontal and vertical directions, but the vertical path also consists of an additional fast carry-chain path. The cell-to-cell delay within a slice is in the 5-20 ps range, a value well-suited for high-resolution TDC operation. The delay from slice-to-slice, however, can be 130 ps or longer, resulting in ultra-wide time-bins with a much longer τ_d. Besides these physical path length differences, even the same-slice delays are both temperature and voltage dependent, which can result in even worse timing non-uniformity.

By applying an input signal uncorrelated with the TDC's clock, a histogram corresponding to the widths of each time bin can be extracted. If every time bin were the same length, then given a large enough number of samples, each bin should receive the same amount of hits. An example of one such system's time-bin non-uniformity is shown below in Figure 5.6. This effect has been lessened in the so-called 'wave-union' approach discussed in [105], a variation of the single delay-chain approach of Figure 5.3. By forming a multi-pulse train out of the original single-pulse signal, the various rising edges, or waves, of the signal are measured as they travel through the delay chain. With carefully tuned offsets between waves, signals which would otherwise fall into an ultra-wide bin can now be registered in a combination of many. This approach effectively makes multiple measurements on the same original input, and can greatly reduce the non-uniformity effect as shown in Figure 5.6.

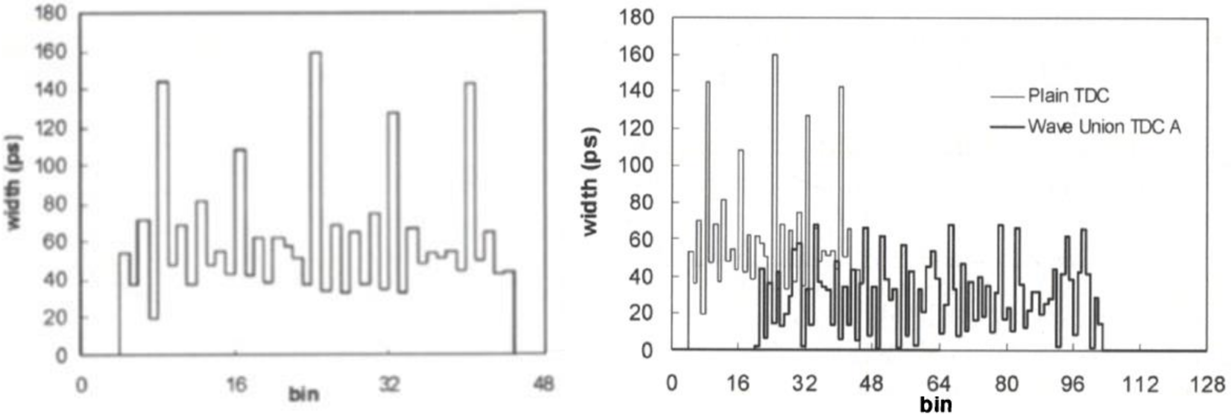


Figure 5.6. Time-bin uniformity of simple delay line implemented in an FPGA (left). Longer connections between logic slices cause some bins to be much longer than others. By forming a pulse-train, or 'wave' from a single input pulse, the same input can be measured multiple times – and not always fall into an ultra-wide bin. Figure courtesy of *[105]*.

Although the wave-union improved the average timing-resolution, there still exist bins which are several times longer than others. Another variation of the delay line was developed for use at CERN, the GANDALF 128-Channel TDC [106]. Instead of a single clock controlling a serial chain of delay elements, GANDALF uses sixteen equidistant phase-shifted clocks, each driving a single flip-flop, to sample the input in parallel. In the single-clock method, the choice of the first element of the chain sets the delay for every following element, without any possibility of modification. With the multiple-clock method, however, the user only has to choose single elements with delays matching the desired τ_d, a process which can be automated with the software development tools. FPGAs generally have very precise clock manager resources, so generating the sixteen different phase shifts is relatively trivial. Of course, temperature and voltage fluctuations are still present, but as shown in Figure 5.7, the time-bin uniformity of the channels is greatly increased. Here the unit of measurement is differential non-linearity (DNL) in terms of least-significant-bits (LSBs), with 1 LSB being equal to a time difference of τ_d. Because each delay chain only requires

16 flip-flops and an associated lower-speed counter, the GANDALF system was impressively able to pack 128 channels into a single FPGA.

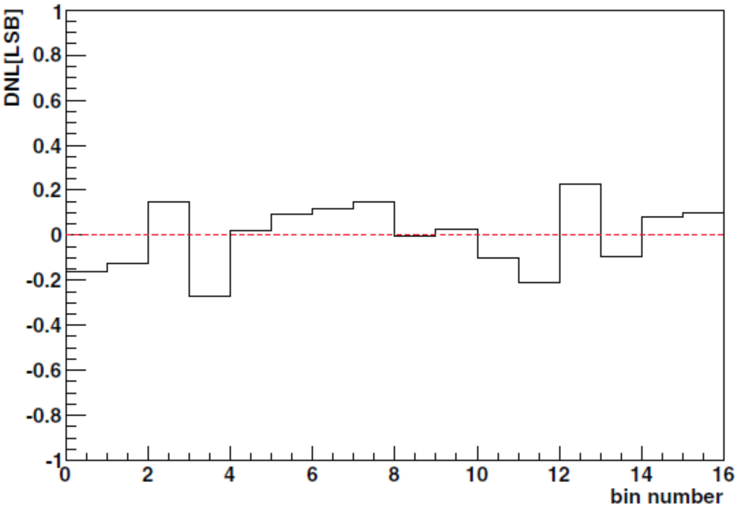


Figure 5.7. Differential nonlinearity (DNL) of one of the 128 channels in the GANDALF FPGA-based TDC. Automated placement tools allowed for optimal placement of each of the sixteen phase-shifted flip-flops, resulting in a much more uniform time-bin width. A DNL of one would correspond to a bin of width $2\tau_d$. Figure courtesy of *[106]*.

FPGAs often have integrated high-speed serial pins with dedicated resources for applications such as PCIe or Ethernet interfaces. The rated maximum single-pin serial bit rate for one FPGA is listed at 1.2 Gbps, corresponding to a time-bin width of 833 ps. While this is much larger than the TDCs listed above, interfacing with these pins is straight-forward, and cores for communicating with PCs for data storage are often already made. The wide variety of implementation options has led to the emergence of FPGAs as the *logical* platform for TDC design. In the remainder of this chapter we present the implementation details and results of a TDC designed for SPAD characterization and random number generation.

5.5 The Wayne-Tagger TDC

Both the QRNGs of Chapter 3 and the trap lifetime characterization efforts of Chapter 4 required accurate timing information of photon arrival times, as detected by their respective SPADs. Commercially available TDCs can be expensive, with some systems costing upwards of $80,000. Additionally, because the range of possible applications for TDCs is so broad, their specifications do not necessarily meet those required for some quantum information experiments. For example, one commercial system has 1 ps timing resolution but an 80 ns dead time – much longer than a typical SPAD's recovery time. For this reason, a custom TDC was constructed, dubbed the "Wayne-Tagger" (WTDC) by J.C. Bienfang. Developed with quantum information in mind, the WTDC's specifications were specifically chosen to interface with typical SPADs. In this section we explain the implementation and performance of the WTDC, and give suggestions for future improvements.

5.5.1 Design Requirements

There were four main performance requirements for the WTDC system: time-bin uniformity, recovery time, maximum detection rate, and timing resolution. Improvements in one category typically lead to detriments in another, e.g., ultra-fine timing resolution results in more bits-per-detection and a lower maximum detection rate, so a balance had to be found to create a suitable final system.

- **Maximum Detection Rate** – Because the WTDC is designed to primarily measure timing information from single-photon detections, a maximum rate consistent with typical SPAD detection speeds was chosen. The rate of 100 Mtags/s encompasses commercially available actively quenched SPADs as well as the lower-end of some of the gated-mode devices as well.
- **Timing Resolution** -- The timing resolution would ideally be as small as possible, but in practice having a τ_d less than the timing resolution of the SPAD would blur the origin of entropy extracted with our RNGs. E.g., if the CCQRNG's SPAD had 100 ps of jitter and the arrival times were resolved with 1 ps time bins, many of the least significant bits in the timing information would then be significantly affected by the classical noise in the SPAD's output electronics, not the actual intended quantum phenomena. Typical devices have a timing jitter on the order of 80 ps, so a modest 100 ps τ_d was chosen.
- **Recovery time** -- The shorter (< 10 ns) trap lifetimes present in the characterized silicon SPADs required that the system be able to detect events occurring very close to each other. Here the choice of a maximum dead time of less than 5 ns was chosen because it is well below the fastest rate at which our characterized Si SPADs can detect, and could be useful in the future with some of the higher-speed harmonic-subtraction SPADs, as in [91].
- **Time-bin Uniformity** -- Differences in adjacent time-bins will result in some values being more likely to occur. This would result in less entropy per detection for the random number generation systems, and less accurate trap lifetime measurements. Typically, this metric is measured in terms of the least-significant-bit (LSB), with exemplary systems having 0.1 LSB or less across their operating range. For our 100 ps τ_d, a DNL of 0.1 LSB corresponds to a time-bin size of 100 ps ± 10

ps. Because our τ_d is much larger than in usual systems, 0.1 LSB is not an appropriate threshold, so a LSB of 0.02, or a maximum 2 ps variation in time-bin uniformity was chosen.

5.5.2 Implementation Details

The WTDC is composed of components used for three primary operations: data parallelization, processing, and storage. The data parallelization stage consists of a 1:16 demultiplexer which transforms the high-speed data into a lower-speed parallel stream. The data processing is done internally on an FPGA, where the demultiplexer's output is processed into useable timetags, and the PCI-express interface and drivers make up the data storage stage. A conceptual overview of the system is shown in Figure 5.8, and an explanation of the data flow is given next.

WTDC Data Flow:

- The 1:16 demultiplexer (DMUX) is an Adsantec 2011 DC-17 Gbps digital deserializer [108] which receives two inputs: the data to be deserialized ('Input'), and the high-speed serial clock, which for this system has a frequency of 10 GHz. The clock was provided by two different sources in the various revisions of the WDTC; one was an external clock multiplication board which is summarized in Appendix C, and the other was a commercial RF oscillator. The commercial oscillator had better stability and was ultimately used for the final design.
- At each rising edge of the 10 GHz clock the data is sampled by the DMUX and held internally until 16 samples have been registered, at which point all 16 bits are output simultaneously synchronous with a lower-speed 625 MHz clock (which is derived from a divide-by-16 counter and the 10 GHz clock). These 17 signals (clock and data) are differential LVDS, and all 34 signals are input into a HiTech Global Virtex 6 PCIexpress Evaluation Board. This evaluation board is populated with an LXT6V365 -3 speed grade FPGA, which has a maximum input clock frequency of 800 MHz.

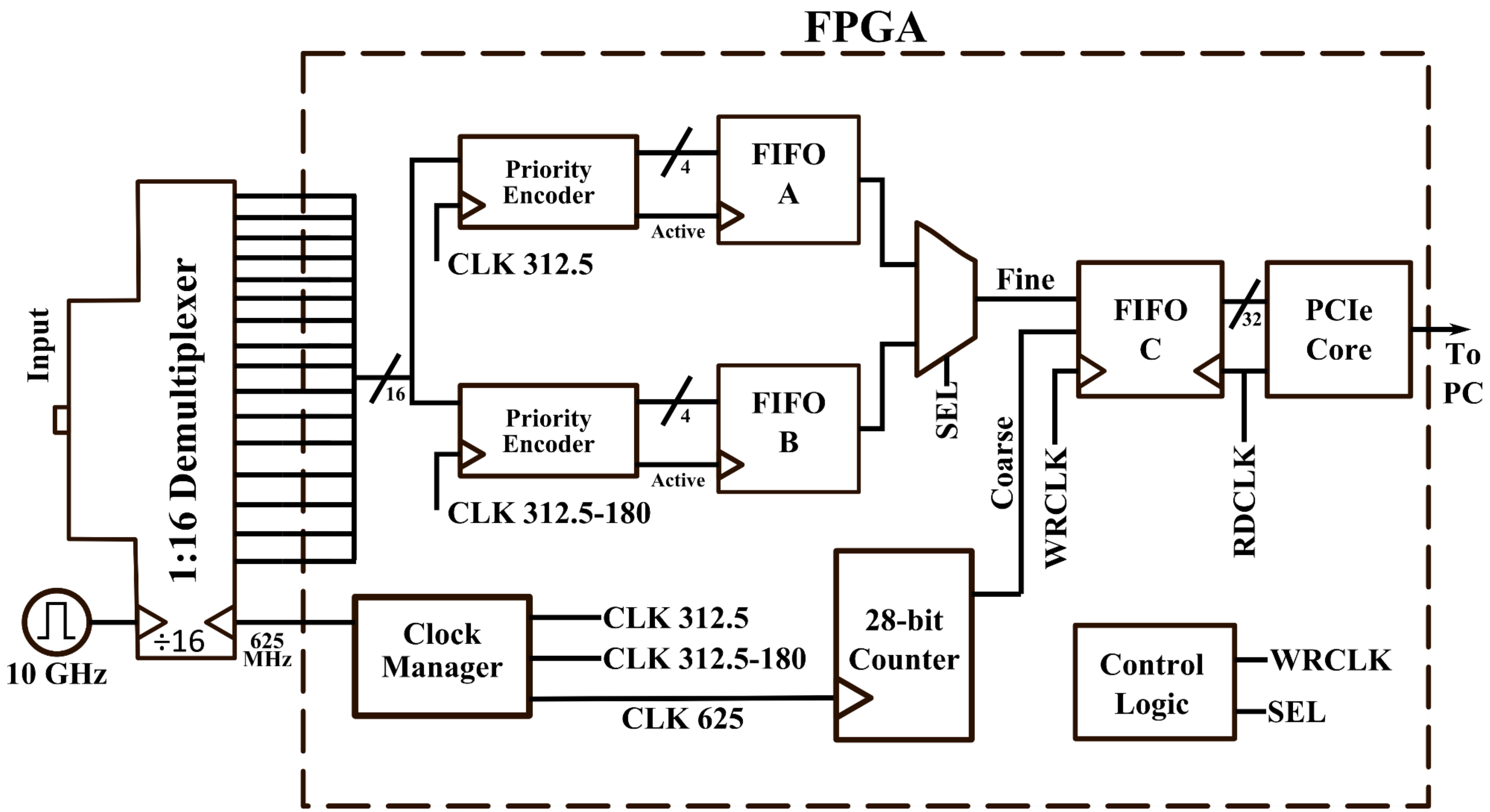


Figure 5.8. Simplified schematic of the WTDC. Signals are input to the 1:16 DMUX which samples at 10 GHz. A 16-bit word is output every 16 samples, synchronous with a 625 MHz clock (10 GHz / 16), and input into the FPGA. To ease timing requirements, two priority encoders clocked in DDR mode (CLK312 and CLK312 shifted by 180 degrees) perform the fine 100 ps portion of the timing measurement, converting the 16-bit word into a 4-bit binary representation. An additional 28-bit counter is running in parallel, forming the 1.6 ns coarse portion of the measurement, which is combined with the fine portion to create a 32-bit timetag. The control logic shown is simplified, but dictates when the first-in-first-out (FIFO) memories read and write. The 32-bit tags are stored and read when available by the Xillybus PCIe *[107]* core, at rates up to 400 MB/s.

- Although 625 MHz is below the maximum frequency allowed by the FPGA, the amount of logic operations required for each 16-bit word is enough to nominally violate the FPGAs timing constraints. For this reason, the DMUX's output is split into two channels. Channel A is sampled by a 312.5 MHz clock which is phase aligned with the original 625 MHz signal, while Channel B is sampled by a 312.5 MHz clock which has a phase shift of 180°. These two clocks sampling in parallel result in a total sampling frequency of 625 MHz, a common clocking technique known as dual data rate (DDR). Sampling the DMUX's input in this staggered lower-frequency fashion eases the timing constraints on the FPGAs logic and increases throughput. The CLK625 and both CLK312.5 signals are derived from a mixed-mode clock manager (MMCM) tile, a component internal to the FPGA.
- Each individual DDR channel is composed of one priority encoder, one 2 kB deep first-in-first-out (FIFO) buffer, and intermediate pipeline chains (not shown). The priority encoder inputs the 16-bit thermometer code of the DMUX and outputs a 4-bit binary representation of the highest priority logically-high bit, as well as an 'active' signal. Here priority is given to bits which correspond to events which occur earlier in time, so for long input pulses multiple words will have bits which are all high. By sampling on the rising edge of the 'active' signal, the location of the rising edge, and thus the arrival time of the signal, is stored in the FIFO. Each four-bit value forms the 'fine' portion of the timing measurement, with a resolution of 100 ps.
- Parallel to the priority encoder channels is a 28-bit synchronous counter clocked at a frequency of 625 MHz, the output frequency of the demultiplexer. Its output forms the coarse portion of the timing measurement, with a resolution of 1.6 ns.
- Additional control logic senses both when a rising edge has occurred on the input, and on which priority encoder path the fine measurement was located. At this point the SEL signal chooses which path FIFO C should accept as its input, and the whole 32-bit timetag (28-bits coarse, and 4-bits fine) are concatenated and stored in FIFO C.
- The 'PCIe core' block encapsulates all the logic necessary to interface with the PCI express interface on the Windows-based PC. Initially PCI and PCI-X cores were independently developed but the maximum transfer rates allowed by their respective standards were too low to meet the 100 Mtags/second proposed requirement. Xillybus, an open-source PCIe core, was used for the final WDTC implementation, which allowed for data transfers at the required rate [107]. The Xillybus driver automatically detects whenever any data is available on FIFO C and independently asserts the RDCLK signal (synchronous to the PCIe 100 MHz clock) to read from the FIFO.

- The output time-tags are read out 32 bits at a time and are stored in a binary file through a simple C command line program. With the 100 ps resolution and 32-bit range, this allows for a maximum timing interval of roughly half a second. Because SPAD dark count rates are typically in the ≈ 100 – 500 counts / second range, the probability of no count occurring in this time frame (and timing ambiguity being introduced) is acceptably small.

5.5.3 WTDC Measured Performance: Time-bin Uniformity

Any potential non-uniformity can be measured by connecting the WTDCs input to a signal asynchronous to its internal time reference. A variety of samples were analyzed ranging from single-photon detections at an assortment of count rates, to the output of a fast pulse-pattern generator. Assuming that the input counts have absolutely no correlation to the sampling rate of the DMUX, then no time-tag should have a higher probability of occurring than any other. Upon analyzing the time-tags it was quickly determined that there was a slight periodic structure on the order of 16-bits. As special effort was taken to eliminate any non-uniformities rising from the FPGA code itself, it is highly probable that the nonlinearities that are observed are due to small differences inside the external demultiplexer itself. As was described earlier, this is a common effect in TDC systems, either due to fabrication process variations or slight path length differences.

A very large (2 Gb) amount of time tags were taken and their last four-bits (corresponding to the 'fine' portion measured by the DMUX), were histogrammed into 16 bins. The expected value of each bin, assuming perfect uniformity, would therefore be 125 million. Assuming Poissonian statistics, the random fluctuations (std. dev. ≈ 2700) are much less than the expected number of hits in each bin, and are neglected. The frequency of the 10 GHz oscillator was measured and observed to be stable over long periods of time, so for this measurement a τ_d of 100 ps is assumed; a more detailed analysis of the clock's frequency characteristics is given in Appendix C. Therefore, we can estimate the duration of each time bin as the simple formula

$$Timebin\ Duration = \frac{Measured\ Counts}{Expected\ Counts} * \tau_d. \qquad \text{Eqn. (5.1)}$$

The results of Figure 5.9 were similar across multiple datasets. While this effect could certainly be due to other unknown external influences acting on both the DMUXs clock and the input source, both the 16-bin periodicity and the fact that applying slight changes in temperature to the DMUX chip itself causes the effect to be more pronounced gives strong confidence that it is indeed due to the internal structure of the demultiplexer. Nevertheless, the ≈ 1 ps maximum variation across bins is a value highly competitive across all systems reviewed in this work.

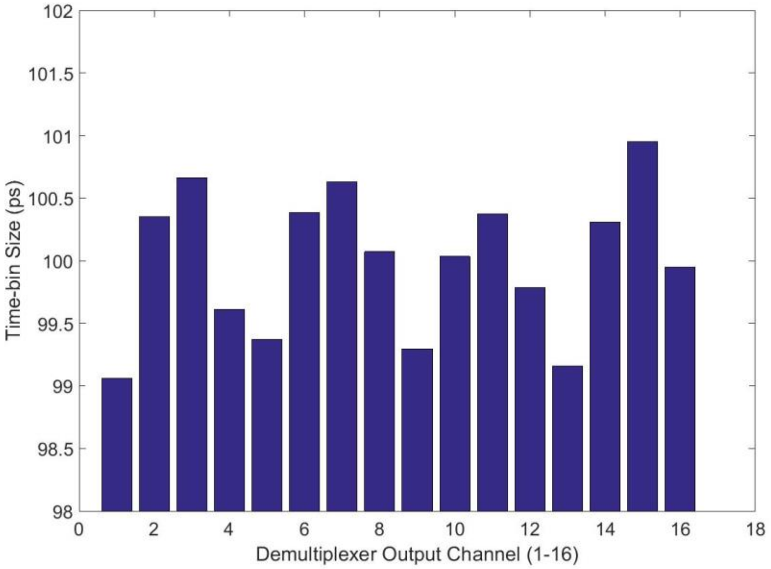


Figure 5.9. Measured periodic time-bin uniformity (or lack thereof) of the WTDC system over all sixteen demultiplexer channels.

5.5.4 WTDC Measured Performance: Dead Time, Detection Rate, and Pulse Width

To measure the dead time of the WTDC, a periodic pulse-train was input into the demultiplexer and the time intervals between one million tags measured. Initially, the input pulse-pattern consisted of a single 1.6 ns long pulse at an input frequency of 19.5 MHz. All the measured time-differences of this data set were 51.28 ns ± 0.1 ns, as expected (Figure 5.10, top). A second pulse was then inserted into the middle of the pattern and moved sequentially closer to the pattern's beginning. This kept the total input rate at ≈ 39 MHz (well below the expected limit), but caused the interval between the first two pulses to become smaller and smaller. The middle graph is the timing interval data when the first two pulses were ≈ 5.2 ns apart, and are being counted correctly (500,000 counts in each bin). When the pulses were 3.2 ns apart, the time intervals became erroneous, with almost all of the tightly-spaced pulses becoming invisible to the time-tagger and not being registered at all. This resulted in some intervals being correctly measured at 3.2 ns (red arrow, ≈10,000 out of 1,000,000), but the rest being missed and thus the next reported interval was the full 52 ns period. This effect is shown in the bottom graph and was very abrupt, not occurring at an interval of 3.3 ns, only 100 ps more. It is hypothesized (and simulations agree), that this is due to the priority encoder having a two-clock-cycle latency of 3.2 ns. This could presumably be corrected for with better routing of logic resources, but is below the 5 ns dead-time specification, and is not a priority at this time.

The maximum detection rate was measured by inputting a clock of a known frequency and measuring the interval between registered time tags. When the clock reached ≈ 104 MHz, the time intervals become erroneous in a fashion similar to Figure 5.10 bottom. At 104 Mtags/s and 4 bytes/tag, the 416 Mbyte/s maximum rate is very close to the reported 400 Mbyte / s rate specified in the Xillybus PCIe core [107]. The minimum pulse width was also measured by inputting pulses of lessening width. At a width of ~100 ps some counts were not recorded, due to the demultiplexer sampling every 100 ps, and missing some proportion of the counts if they do not cross a sampling clock-edge.

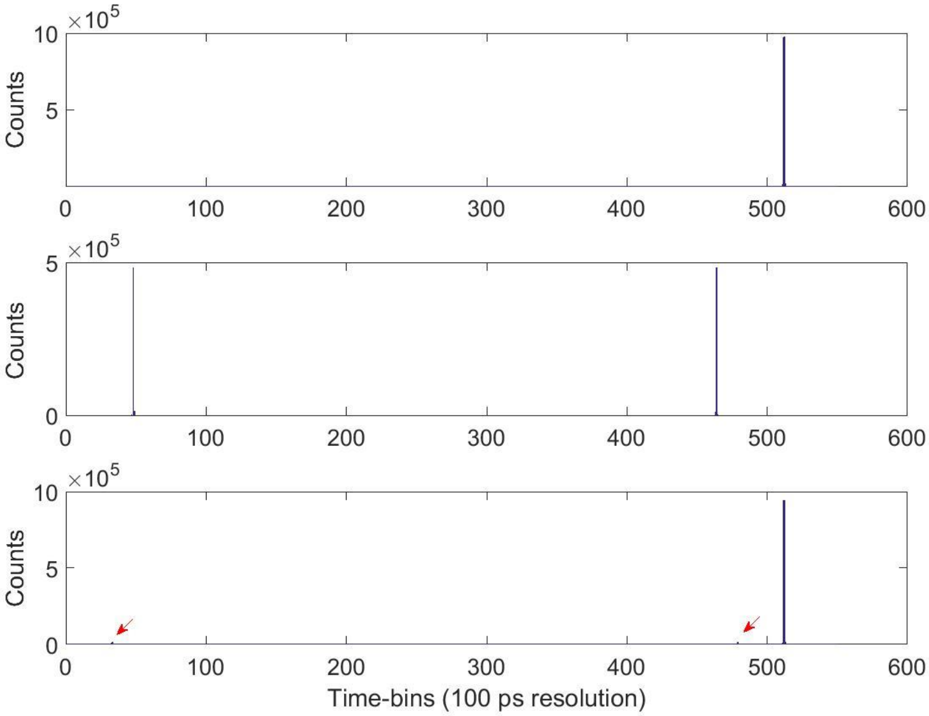


Figure 5.10.Timing interval histogram illustrating the WTDC's 3.2 ns dead time. When a single pulse of frequency 19.5 MHz is input, all tags are correctly binned ≈ 52 ns apart (top). When a second pulse is inserted 5.4 ns after the first, half of the counts correctly go in the 5.4 ns range, and the other half into the ≈ 46 ns range. Once the interval is less than 3.2 ns the second pulse is only registered by the WTDC with ~ 1% probability.

5.5.5 Physical Implementation

For initial testing and proof of concept, a commercial demultiplexer evaluation board was used. Each individual differential data channel and the low-speed clock from the DMUX are routed to SMA connectors, which must then interface to the FPGA. Because the FPGA board only has two general purpose I/O SMA connectors, an additional breakout board was designed to route all 34 signals to a single high-density QSE-type connector on the FPGA. Care was taken to ensure the impedance of each trace was 50 Ω, as well as to match each trace's length to within 1 mil, corresponding to a maximum propagation delay difference of 1.45 ps. As shown in Figure 5.11, this design was cumbersome, requiring many SMA cables, multiple external power supplies for the demultiplexer (15 V oscillator, 3.3 V DMUX supply, threshold for negative data input if single-ended input), and a mechanical mount.

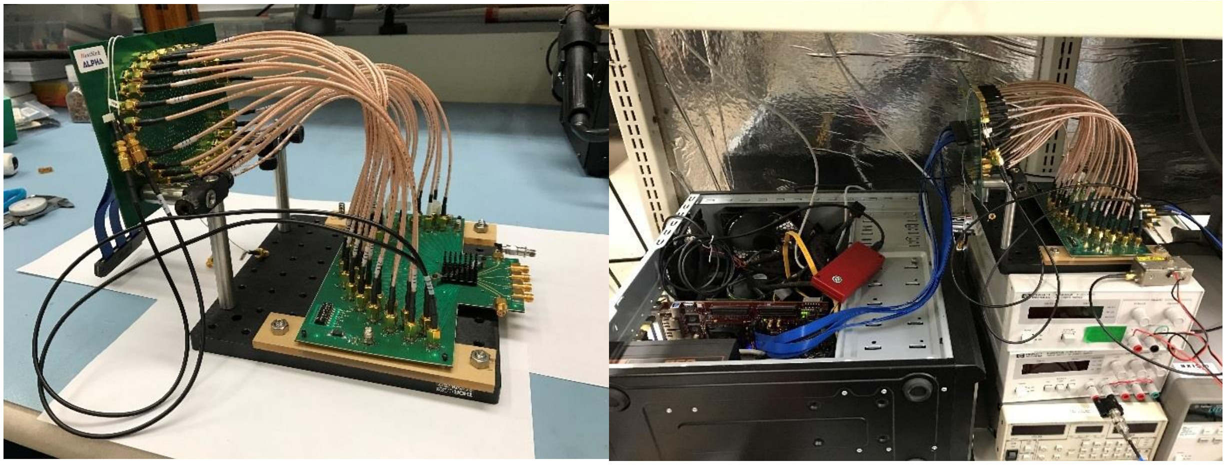


Figure 5.11. First iteration of WTDC time-tagging system. Standalone DMUX evaluation board and breakout adapter (left), and while interfacing with the PC through the PCIe-FPGA (right).

Besides the awkward physical size, this system had several disadvantages which required additional effort in the FPGA code. Upon every rising edge of the 625 MHz clock, all 16 data channels (32 differential) had to be sampled in a reliable fashion. Although carefully matched propagation delays were a priority for the designed boards, it was apparently not for the commercial FPGA evaluation board. Several of the FPGA data traces were length-mismatched and because the data channels could be in transition during each clock rising-edge, erroneous values resulted. Consequently, the clock had to be delayed (longer looped SMA cables in Figure 5.11) by approximately 90° (400 ps) so that all of the data would be stable for the input registers. The clock-alignment process was tedious and sometimes unstable. Physically disturbing the SMA-cables could cause in minute changes to the clock waveform, a situation unacceptable in our experimental environment.

To alleviate the physical space required and instability of the above system, an all-in-one solution was designed. The revised WTDC board had two individual demultiplexers, allowing for two independent 100 ps channels. The capability to combine both channels into a single 50 ps channel is discussed in Section 5.5.9. There was no necessity for external power supplies (although the option was implemented[8]), as both the DMUX's power and negative data threshold were supplied by the FPGA itself. Jumper resistors also enabled optional clock-phase alignment, removing the need for the tedious procedure described earlier. As shown in Figure 5.12, the whole system was very compact and fit on a single FMC connector on the FPGA (left). Because the PCIe-mounted FPGA board is housed in the PC's case and can sometimes be inaccessible, another FMC-to-QSE breakout board (right) was designed to allow for table-top operation. Either configuration also allows for interfacing with the clocking board presented in Appendix C, and screenshots of each boards PCB layout are included in Appendix D.

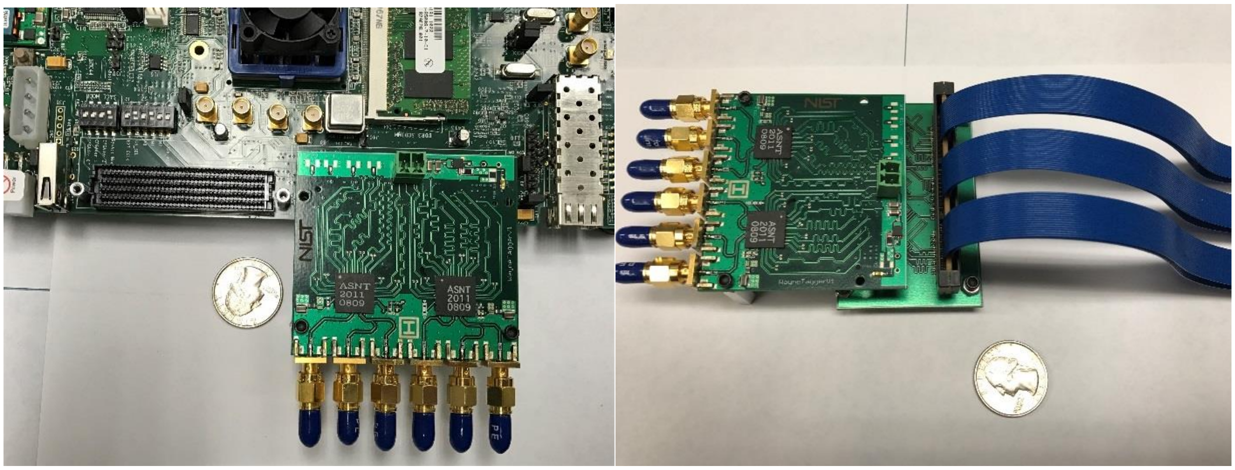


Figure 5.12. Improved revision of the WTDC interfacing through a mounted FMC connector on the FPGA (left, requires PCIe access)***,*** or the blue QSE-cable (right) for table-top use. 68 differential signals from two independent DMUXs are routed to the FPGA through length-matched traces, resulting in a much more stable system than the previous version.

Each individual DMUX channel was characterized in a manner identical to that in 5.5.3 and 5.5.4, and reported similar results. Ultimately, the right DMUX in Figure 5.12 was rendered useless due to an inconsistency in the package footprint supplied by the manufacturer, causing the chip to short on the PCB; another iteration of the board is required to correct. The PCIe interface remains an issue, constraining the location of the FPGA to inside the PCs case this. This lowers its ability to be easily transported between computers, and the eventual transition to a USB 3.0 interface is a future option. Nevertheless, this new

[8] The amount of current drawn by the WTDC board was near the maximum allowed by the on-board FPGA power supply, so an extra option was added.

design greatly increases the convenience of the WTDC system, in addition to enhancing its capabilities, through an additional channel.

5.5.6 Sample WTDC Histogram

To illustrate the intended use of the WTDC, it was integrated into the simple experimental setup shown in Figure 5.13. A commercial Perkin-Elmer SPCM SPAD module was illuminated by a CW-driven 850 nm VCSEL. The beam was collimated, focused, and attenuated with neutral density filters until the average detections per second were roughly one million. A sample of 10^6 time-tags were recorded by the WTDC, stored in a file on a PC, and their differences were used to populate a histogram, as shown in Figure 5.14.

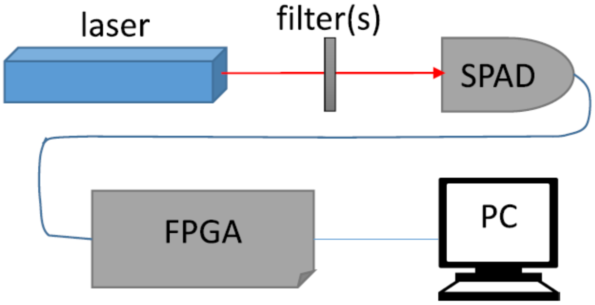


Figure 5.13. Depiction of simple experimental WTDC setup used to characterize SPADs.

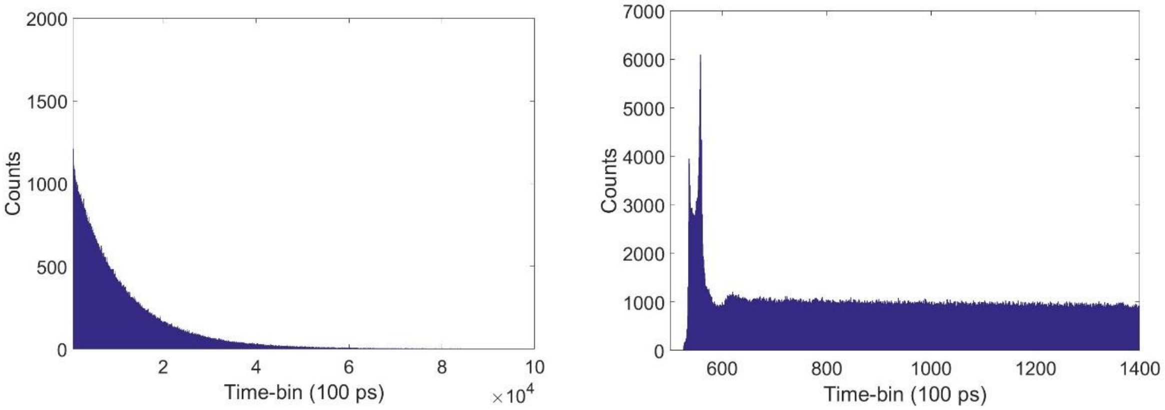


Figure 5.14. Full-range (left) and zoomed in (right) histogram of time-tags corresponding to SPAD detection waiting-times. Fitting a multi-exponential to the full data set yields important information about trap lifetimes while focusing on the region immediately after the dead time reveals subtle non-idealities.

At first glance, the waiting-time distribution looks roughly Poissonian, with a decaying exponential shape of time constant ≈ 10^{-6} s. Upon zooming in, however, additional features are revealed – giving important clues to the intricacy of the SPADs internal operation. Immediately visible is the SPADs dead time, which is measured to be 52.4 ± 0.1 ns. There is a slight slope when the SPAD first recovers, presumably due to the rise-time of the SPADs reset electronics, and there is some oscillatory structure, which is due to ringing on the reset transition. Both of these effects translate into slight nanosecond-scale variations on the voltage across the SPAD, which then result into differing detection efficiencies during these times. The large double-peak structure is due to the pile-up of events that occur after the SPAD is active, but before the sensing comparator is unlatched and sensitive to detections. The trench immediately afterwards is due to the suppressed detection probability from the double-peak region. This highlights an important point: that while the properties of a SPAD module are very much due to the absorption and gain characteristics of the SPADs material, they are also tightly linked to the behavior of the control and readout circuitry used by the accompanying electronics.

5.5.7 First- and Second-Order SPAD Characterization

It is generally the case that a SPAD's characterization consists of measuring its dark count rate, dead time, detection efficiency, and afterpulsing [47]. Many of these quantities can be easily estimated – i.e., the dark count rate as the rate of detections under no light stimulus, and the detection efficiency as the ratio of registered detections to incoming photons. These are first-order approximations, however, and can introduce systematic errors as they ignore non-idealities of the SPAD's performance. The detectors dead time (the period immediately after a detection where the SPAD is disabled) and afterpulsing probability (excess counts due to trapped charge from a previous avalanche) are effects that are correlated with a prior detection event. Therefore, most SPAD characterization is performed with a second-order detector model.

In a second-order characterization, the SPAD is illuminated by a light source with a known repetition rate and average power, a series of time-tags is recorded, and a histogram is formed from the differences between adjacent time-tag. An example of measured time-tag differences for two SPADs, taken at a variety of count rates, is shown in Figure 5.15.

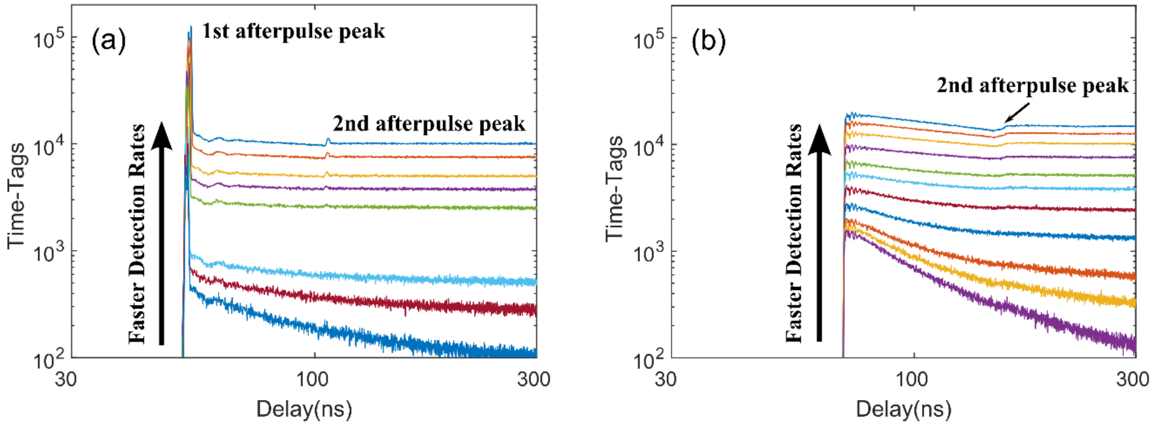


Figure 5.15. Time tag differences from two different SPADS, a SPCM (left) and MPD (right). The various line colors denote samples taken at different average detection rates. Dead times are visible at ≈ 52 ns for the MPD detector and ≈ 70 ns for the SPCM. The SPCM has a much more pronounced afterpulse peak immediately after recovering, and both display second-afterpulse peaks at twice the recovery time, ≈ 104 ns and ≈ 140 ns respectively.

The dead times for both devices can be calculated as the earliest time-bin in which a detection is registered. The dark count rate can be measured under no illumination, and the recovery process temporal structure can be determined by examining the interval directly after the dead time. Given a characterized input stimulus with a well-defined temporal position, the afterpulse probability can be extracted from second-order correlations conditional on a first event occurring in the input window. The detection efficiency can then be accurately calculated by subtracting off the measured afterpulse probability. Additionally, the characteristic trap lifetimes can be determined according to the manner described in Chapter 4.

5.5.8 Third-Order SPAD Characterization

The second-order model is reasonably effective in describing SPAD operation, but it relies on the implicit assumption that the device's behavior depends only the time passed since its previous detection. Given typical SPAD count rates of 1-20 MHz and trap lifetimes which are largely less than 100 ns, this is not unreasonable. However, this only considers the optoelectronic material properties of the diode itself, and not the subtler effects that can be present either in the SPAD or the detection circuitry.

Due to the WTDCs low dead time, it is possible to both examine operation at detection rates near the upper threshold of SPAD operation (≈ 10-20 MHz), as well as not have any detections occur so fast that the time-tagger cannot record them. The ability to not discard any information has allowed for the exploration of higher-order correlations at faster detection speeds. If the SPAD truly had no memory after one detection, then the second-order correlation would be equal to the higher-order correlations. The afterpulse of an afterpulse, for instance, should not depend on the initial first-afterpulse-causing event. Therefore, given a large enough sample of time tags, the behavior of the 2nd and 3rd order correlations (first and second afterpulses), should be identical. To verify this, a third order autocorrelation is employed, which we define as

$$C^{(3)}\left(\Delta t^{(12)}, \Delta t^{(13)}\right) \equiv \int_{-\infty}^{+\infty} f(t+\Delta t^{(12)}) f\left(t+\Delta t^{(13)}\right) f(t) dt. \qquad \text{Eqn. (5.2)}$$

Here, $f(t)$ is a function which represents the electronic output of the SPAD, $\Delta t^{(12)}$ is the time interval between a first and second count, and $\Delta t^{(13)}$ time interval between a first and third. Experimentally, $f(t)$ is measured from a series of discrete time-tags and is not continuous, so the integration is performed only over the length of the dataset. Two SPADs, a Perkin-Elmer SPCM and a MPD PDM, were characterized with the WTDC, and their time tags were analyzed for third-order correlations.

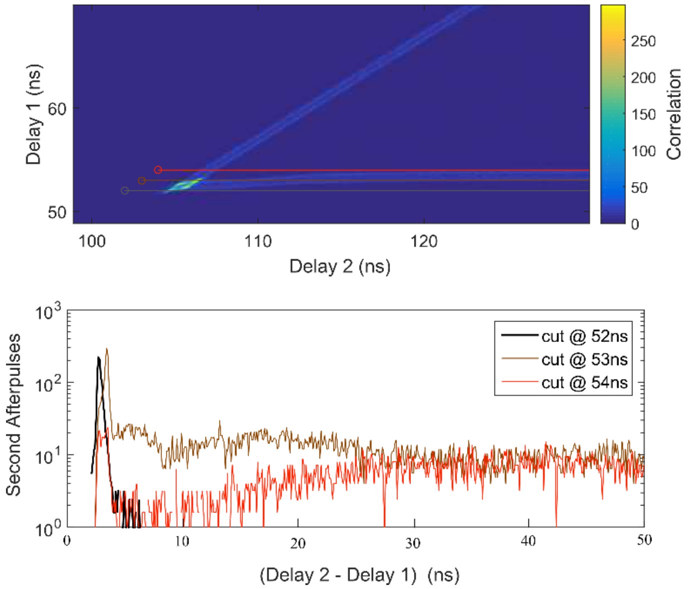


Figure 5.16. Third-order correlation measurements for the Perkin-Elmer SPCM. The top graph is the $C^{(3)}$ correlation measurement, which has been normalized to zero. The dead time of this SPAD is 51 ns, and there are clear regions of increased correlation when both afterpulses occur immediately after the recovery time (Delay 1 = 51 ns, Delay 2 = 102 ns) The bottom figure illustrates several horizontal slices taken from the top graph, which show the relationship between second afterpulse behavior and the relative delays at which they were detected.

Figure 5.16 top shows the measured $C^{(3)}$ autocorrelation function for the Perkin-Elmer SPCM, and the horizontal lines represent particular slices, which are shown in the bottom graph. For this SPAD, the second-order model is valid for most detection times, with the majority of the top figure being roughly equal to zero. Of special interest, however, is the region corresponding to detections which occur immediately after the recovery time of 52 ns. For this detector it is observed that if a first afterpulse occurred just after the reset time ($\Delta t^{(12)}$ = 52-54 ns), then the probability of a second afterpulse ($\Delta t^{(13)}$ = 104-106 ns) is significantly higher, as illustrated by the data in the bottom graph.

During the reset, there is a few-nanosecond period for which the SPAD is biased above the breakdown voltage but the active-quenching circuitry has not yet fully recovered. The device is sensitive on the single-photon level, but the circuity is not ready and therefore responds when it is armed – a time slightly later than under normal circumstances. Therefore, directly supported by the conclusions of Chapter 4, the avalanche current through the SPAD can persist longer, more traps are populated, and the afterpulsing profile is increased. This effect was undiscovered until this type of higher-order characterization, which was only possible due to the WTDC's performance characteristics.

For the MPD SPAD, a much different type of correlation is revealed, as shown in Figure 5.17. To better accentuate the correlations, the middle graph shows the difference between the measured $C^{(3)}$ and the noise floor (what would be expected from a second-order model) on a logarithmic scale. Much like the Perkin-Elmer SPAD, the MPD has abnormal behavior when the first and second afterpulses occur immediately after the dead time of ~70.6 ns. The measured difference in the second graph is negative during these instances, corresponding to a *decreased* probability of afterpulsing.

In some cases, the SPAD's operation is affected on time scales far outside those expected from a fully recovered detector. In the horizontal slice of the bottom graph, which corresponds to a first afterpulse occurring immediately after the SPAD is reactivated (@ 71 ns), the extra recovery delay can be as much as 25 ns (total dead time of ~96 ns), more than a third longer than typical operation. While this effect has not been totally explored, it is highly likely that it is also due to the active quenching circuitry not being totally recovered when the first afterpulse occurs, as was the case with the Perkin-Elmer. This data illustrates that while the 2nd order model is useful for most SPAD characterization purposes, there exist subtle effects which must be fully modeled for high-accuracy measurements. The WTDC is well-suited for this application, and the amount of information gained by such a simple experimental setup will make future characterization efforts easily accessible.

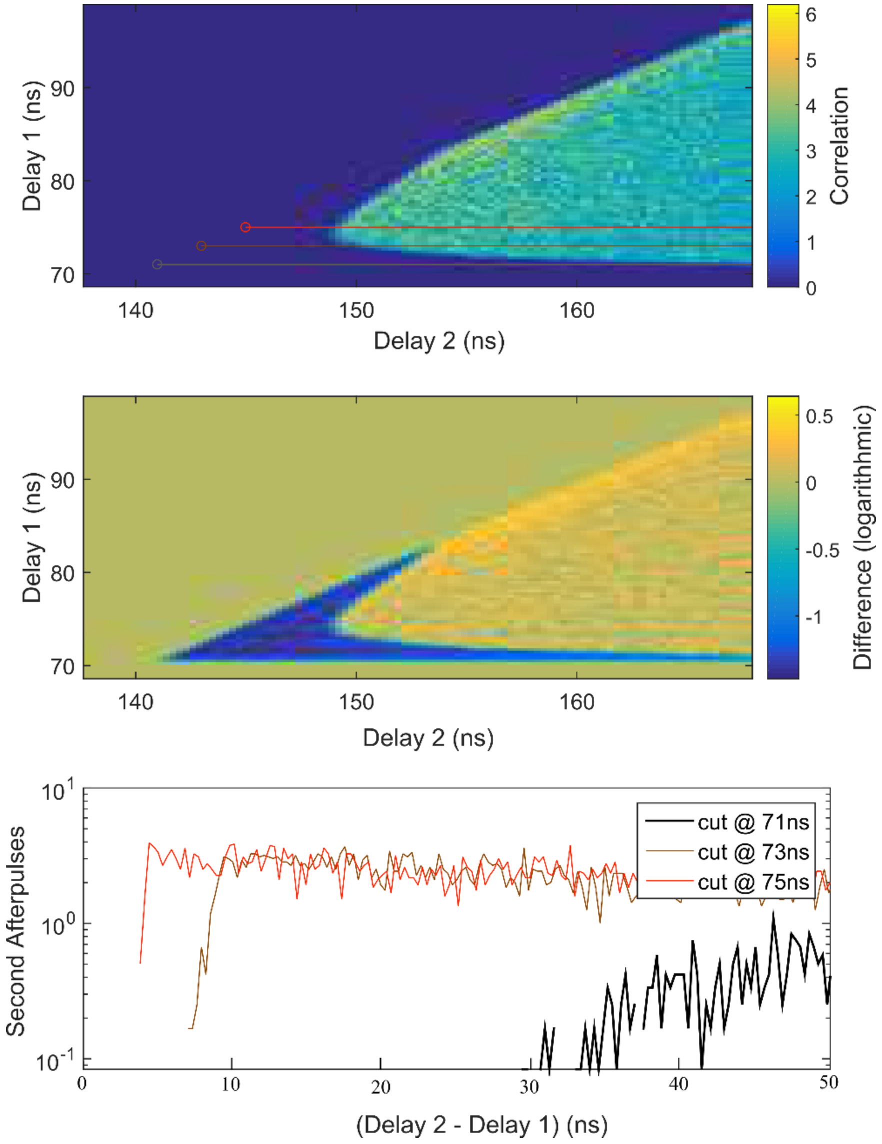


Figure 5.17. Third order correlation measurements for the MPD-PDM SPAD. The top graph shows the measured $C^{(3)}$, with slices taken directly after the ~72 ns dead time shown in the bottom graph. To accent the second afterpulsing behavior, the middle graph shows the difference between $C^{(3)}$, and what would be expected from purely a second-order model.

5.5.9 – Future Improvements: Combined Channel Operation

One of the intentions for the revised two-channel WTDC system in Section 5.5.5 was to include the capability to improve the timing resolution. Although the 100 ps resolution is on the order of the timing jitter for typical commercially available SPADS and currently sufficient, there have been experimental demonstrations of devices with jitter on the order of 25-30 ps [109]. Although constrained to proof-of-concepts for the time being, as these devices become available for sale it will be important to have a time-tagging system with the capability to characterize them effectively.

The concept of phase-shifted clock-sampling has been utilized in previous time-tagging systems[106], and its concept has been applied to the WTDC. For a single WTDC system, the time tag corresponding to an input event corresponds to which cycle of the high-speed 10 GHz clock an event took place in. Splitting the input and sampling it with systems driven by the same clock should provide nominally identical results – assuming each clock signal reaches its respective system simultaneously. However, if we drive multiple systems with phase-shifted versions of the same clock, we can extract additional information.

The simplest case — two clocks phase-shifted by 180° — is shown in Figure 5.18. If only Channel A were enabled, and the input clock frequency was 10 GHz, then the timing position of an input event could only be determined to within 100 ps. If, however, an additional clock is delayed by 50 ps (or 180°), and used to take an independent measurement with another WTDC, then the timing position of an event can be determined to within 50 ps (denoted by the red X's). In theory this process can be repeated an arbitrary number of times, by combining additional channels with equidistant phase-shifted clocks. Assuming M equidistant clocks, the timing resolution can theoretically improve by a factor of $\log_2(M)$.

As the number of channels grows by a factor of two, an additional output bit is required to convey the increased amount of timing information, which increases the data throughput requirements, and lowers the maximum detection rate. Each DMUX of this type require 34 differential inputs, a commodity on FPGAs which quickly runs out as the number of channels increases. Additionally, to achieve good time bin uniformity, the clocks must be kept phase aligned. As multiple clocks are added, the few-ps jitter quickly becomes commensurate with the phase shift itself, and small temperature variations or voltage fluctuations can significantly skew the data.

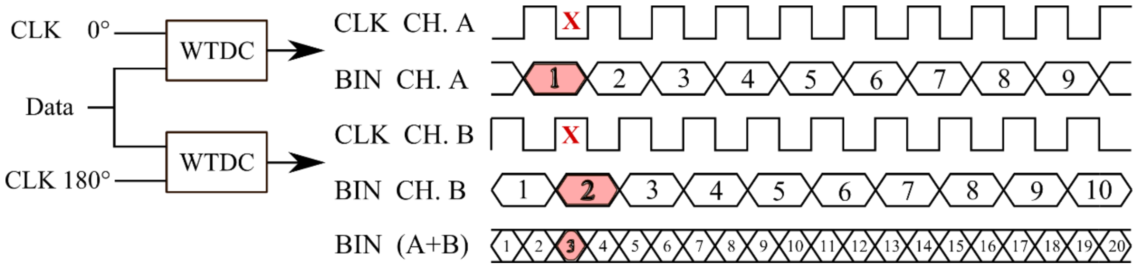


Figure 5.18. Conceptual illustration of the combined-channel operation of the WTDC. An input event (denoted by the red X) is sampled by two synchronous phase-shifted clocks. If the clocks are delayed by equidistant intervals, the sum of the recorded time bins can be taken to extract additional information.

We initially intended to realize the two-channel and increased resolution operation depicted in Figure 5.18. Unfortunately, due to experimental difficulties at the time of this dissertation's submission, the combined functionality of the WTDC has only been partially tested. A mismatch in the dimensions of the supplied PCB-footprint and the one which populates the PCB itself caused some of the DMUXs signals to occasionally short to ground if not perfectly aligned and soldered. Because of this, the right-most DMUX of the revised WTDC shown in Figure 5.12 was destroyed. Additionally, the clocking circuit explained in Appendix C had extraneous frequency components present in its output, which although reduced by filtering, were not sufficient for this mode of operation.

To attempt to illustrate this possible mode of functionality, the single-DMUX system of Figure 5.11 was input into one connector of the FPGA, while a single-DMUX of the revised system in Figure 5.12 (of which one DMUX was operational) was also sampled. The external 10 GHz commercial clock was split and delayed with cable lengths, and fine tuning to achieve the 180° phase shift was done with mechanical trombone delay lines. Unfortunately, due to the physical distances between each DMUXs input clock on the FPGA, the clock manager resources required for accurate phase shifting could not be used, and 50 ps operation was not possible, although each channel worked independently at 100 ps resolution.

5.5.10 Conclusions

In this chapter we presented the design and characterization of the WTDC time-tagging system. At 100 ps timing resolution, 3.2 ns dead time, and 400 MB/s data transfer speed, it is ideal for recording timing information from SPADs operating at current detection speeds, and integration into quantum information experiments. The ability to catch nearly all detection events, due to the time-tagger's short recovery time, has also facilitated the discovery of a new higher-order SPAD characterization technique, which in turn has revealed additional, previously unseen, information about the SPAD operation. Although the dual-channel operation of the WTDC was not implemented, we believe that the addition of digital clock phase-shifters for better data / clock alignment, as well as a correction of the DMUXs footprint, will make the 50 ps operation possible.

Chapter 6 — Low-Latency QRNG & Corrective Feedback

The theories of classical Newtonian physics and relativity are inherently deterministic, and given enough information about a physical system, its outcomes can be definitively predicted. This is often not the case for quantum mechanical systems, however, where the result of a measurement is described by a probability distribution. In an effort to reconcile these two models it was postulated by Einstein in 1927 that perhaps quantum systems were controlled by 'hidden-variables', quantities unknown to us that determine the outcomes of measurements [110]. If information about these hidden-variables became known, then quantum measurements could be predicted with certainty.

There were many early attempts at hidden-variable theories, with the first being Louis de Broglie's 1927 theory, which postulated that every particle had an associated 'pilot-wave', a hidden-variable that guided its trajectory through space [111]. A surprising feature of this theory was the prediction that hidden variables in one location could instantly change state due to events happening elsewhere. This violated the locality principle of classical physics, which states that objects cannot signal each other faster than the speed of light. This non-local phenomena, something Einstein later famously referred to as 'spooky action at a distance', was popularized as the 'EPR paradox' in 1935, and is now referred to as quantum entanglement [112]. The theories of hidden-variables, locality, and realism (the notion that a particular property or observable has a predefined value before any measurement of it is made) were generally accepted until 1964, when Irish physicist John Bell created a theorem which fundamentally altered the understanding of quantum mechanics . Bell's theorem, a centerpiece of modern quantum information theory [113], does not prove the validity of quantum mechanics, but rather constrains the observable correlations with any local realistic theory. An experimental demonstration of Bell's theorem are referred to as Bell tests; and a violation of Bell's inequality eliminates all local realistic explanations of the observed correlations.

In a typical two-party Bell test, a source generates entangled particles and sends them to two parties, Alice and Bob. Alice and Bob independently and randomly measure particular properties of their respective particles, and the joint probability distributions of their measurements are calculated. For entangled quantum particles, the joint probabilities will be constrained differently than for a system obeying local realism; in the latter, the joint probability distribution will *factor* into a product of the distribution for Alice and Bob. The result of a Bell test is an inequality that will be obeyed for local realistic systems, and violated by certain entangled quantum particles. There have been various experimental violations of Bell inequalities [114], but until recently the tests performed were forced to make additional assumptions to validate their results; these provided ‘loopholes’, preventing one from unambiguously disproving local realistic models. Therefore, the pursuit of a ‘loophole-free’ Bell test has been underway for some time.

The random number generator described in this chapter was primarily designed to help close the ‘locality’ loophole. Bell’s theorem requires that the measurement choices of Alice and Bob are made independently of each other, i.e., Bob’s measurement choice cannot influence Alice’s. For local theories, the timescales on which a local-hidden-variable at Bob could conceivably signal one at Alice depends on the physical distance between them, and the speed of light. Therefore, to close the locality loophole, the random measurement choice and the completion of the measurement itself must be done on time-scales less than the times their physical separation allows. If a signal from Alice cannot reach Bob before he has both made his random measurement choice and performed his measurement, they are considered to be ‘space-like separated’. For the physical distances in our Bell test, this required that the total time between random bit-request and bit-output, and the time required to complete the measurement, be on the few-nanosecond scale. In this chapter the design of one such system, a low-latency quantum random number generator (LLQRNG) is discussed. By utilizing a simple physical process and compact electronics, the total bit-generation time is measured to be only 2.4 ± 0.2 ns. The LLQRNG was used in one of the three recent loophole-free demonstrations of Bell’s test, landmark experiments in quantum physics [16]–[18].

During its design, it was discovered that the LLQRNGs output was prone to a slight instability, presumably due to a combination of environmental and yet undiscovered origins. Although accounted for in the Bell experiment, attempts at correction led to additional explorations into various active-feedback approaches. In particular, used the Allan Variance, a quantity used in atomic clock stability analysis, to estimate the origin and strength of the instability. This led to a feedback approach which attempts to minimize the statistical distance between the corrected probability distribution and the expected random normal distribution. To our knowledge, this type of characterization has not been performed on random number generators before, and reveals possible new standards by which to judge the quality of an RNG’s output.

6.1 Experimental Operation

Single-photon detection of optical states in the high-loss regime is a commonly used process for quantum random number generation, e.g., the QRNGs of Chapter 3 [13], [78]. In systems of this type, entropy is extracted by measuring some aspect of an optical state (spatial mode, phase, photon number, etc.), and then used to form a random bit string. Although entropy is present in many natural processes, the measurement by which it is generated may not result in information which is suitable for immediate use. For example, the waiting-time distribution of Poissonian photon detections has a probability distribution which takes the shape of a decaying exponential. As random number applications generally require that all outputs have the same probability of occurring, the unbalanced 'raw' distribution must be post-processed in some fashion, e.g., by a hash function or randomness extractor.[9] However, these algorithms are often complex and can take a substantial amount of time to complete. For example, many of the trusted cryptographic hash functions took upwards of 200 clock cycles (≈ 50 ns) on a supercomputer to output one byte of information [115]. When computed using the slower clock of an FPGA, the time required to complete any substantial post-processing would quickly violate the space-like separation requirement imposed by the Bell test. Therefore, any Bell test-suitable QRNG must either have an output which is immediately of sufficient quality, or must be able to produce bits at a fast enough rate so that a rudimentary whitening process can be quickly applied on a few number of bits to produce one, higher quality bit.

To reduce the latency as much as possible, a simple, fast process was chosen: the sampling of the output of a single-photon detector over an interval of time. An illustration of the experimental setup is shown in Figure 6.1.

[9] A randomness extractor is a mathematical function that when applied to the output of an entropy source, generates an output which appears to be independent from the source and uniformly distributed. Some cryptographic hash functions are also suitable for this purpose.

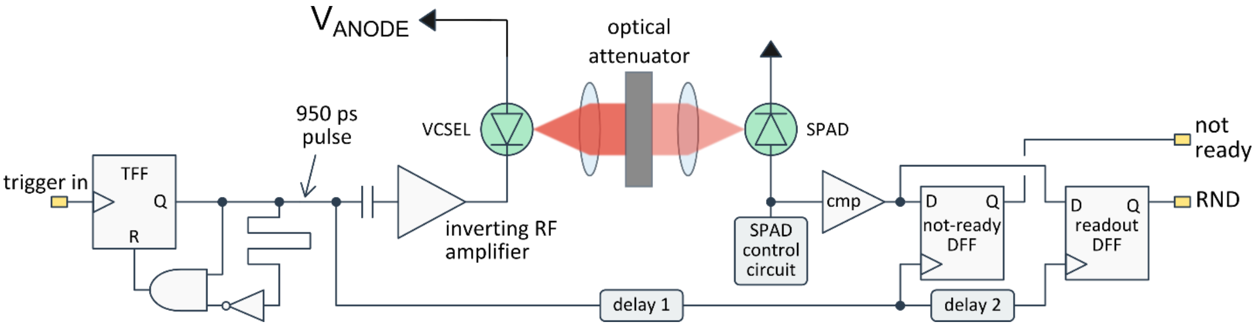


Figure 6.1. Illustration of the LLQRNG. Upon the clocking of the T-type flip-flop by the ‘trigger in’ signal, an optical pulse is created, whose amplitude is determined both by the amplified electrical pulse coupled to the VCSEL’s cathode and the DC-bias on the anode, V_{ANODE}. A temporal window is formed by the clocking of the two D-type flip-flops, and the final output bit-value is determined by whether or not a photon is detected in the window.

Upon the arrival of an electrical trigger, a T-type flip-flop produces a short (~950 ps) electrical pulse which is amplified, and drives the cathode of an 850 nm gain-switched VCSEL. The resulting 1 ns laser pulse is collimated, attenuated, and focused onto a Perkin-Elmer SPCM single-photon avalanche photodiode. The avalanche is sensed by a ADCMP582 high-speed comparator directly coupled to the data-inputs of two D-type flip-flops. Output bit-values are determined by whether or not a detection occurred in an interval whose beginning and end are defined by the clocking of the two flip-flops. A logical-1 from the ‘not-ready’ flip-flop indicates that at the start of the timing window the SPAD was disabled due to an event occurring before the arrival of the bit-request. As this condition could arise from photons arriving a full dead time (~55 ns) before, these detections cannot be guaranteed to satisfy space-like separation, and are discarded. The clocking of the ‘readout’ flip-flop indicates the end of the timing interval, at which time the state of the detector is sampled as the value of the output random bit, ‘RND’. The output random bit-value is ‘1’ if the SPAD registered a detection in the timing window, and ‘0’ if there was no detection.

The value of the final random bit depends on several parameters. The integrated area of the electrical pulse driving the VCSEL determines the amount of electrons injected, and the number of photons output is proportional to the laser’s quantum efficiency. The transmission of photons by the neutral density filters is analogous to the behavior of an unbalanced beam-splitter, one of the first optical-QRNG approaches [44]. The subsequent detection by the SPAD depends on its detection efficiency, and the probability of that detection being within the timing window is affected by the growth process of the impact ionization-driven avalanche, which determines the SPAD’s timing jitter. Therefore, the final probability of the ‘readout DFF’ flip-flop clocking a logical-1 bit-value is, to first order, the product of all these values.

Some of these parameters can be tuned, although some with greater ease than others. The amplitude and duration of the electrical pulse driving the VCSEL is set by the difference between the short direct-path and the longer inverted-path to the AND-gate of Figure 6.1, which resets the T-type flip-flop. As these are copper traces on the laser-driver PCB-board discussed in Appendix F, adjusting their length requires significant PCB-rework; after determining a length which produced a useable optical pulse, this was left untouched. The optical attenuators are enclosed in lens tubes, and replacing them physically disrupts the experimental setup; once a coarse detection probability of ~45-55% is achieved, these are also unchanged. The SPAD's detection efficiency (~45% @ 850 nm) depends both on the photon absorption probability in the diode, and the probability of the resulting photoelectron creating an avalanche detectable by the SPAD's electronics. Without device fabrication capabilities, the absorption probability is an untunable parameter, and although the avalanche probability can be adjusted by increasing the SPADs overvoltage, this is outside the intended use of the commercial SPAD module, and has the potential to destroy the device.

This leaves two conveniently tunable parameters: the temporal length of the detection window, and the forward bias across the laser. The detection window's length depends on the delay between the clocking of the 'not-ready' and 'readout' flip-flops. The length of 'delay 1' sets the start of the window and was tuned only once to match the first possible detection due to the triggered stimulus. The end of the window was coarsely adjusted with differing cable lengths on the 'delay 2' path. The final detection probability is directly proportional to the integrated area under this pulse, so the arrival of this signal is chosen to be in a region for which small perturbations due to the flip-flop's jitter will have as little effect as possible. This was accomplished by analyzing the temporal distribution of registered SPAD detections using an Agilent DSO-type oscilloscope; a histogram is shown in Figure 6.2.

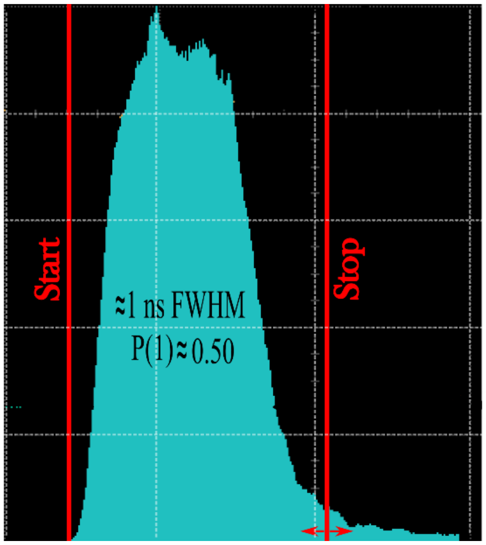


Figure 6.2. Histogram of photon-arrival times as detected by the LLQRNGs SPAD, measured with a 2 GHz bandwidth oscilloscope. The end of the window (Stop) is set by delay 2 (see Figure 6.1), and is adjusted by cable length. The probability of a detection occurring in this window is coarsely tuned to be approximately 50%.

Above threshold, the probability of the VCSEL producing a photon is directly proportional to the bias across the laser, and it is this parameter which has the most flexibility. The initial electrical pulse is coupled to an RF-amplifier; a device whose operation strongly depends on the bias applied at its output. As the amplifier's output is directly coupled to the VCSEL's cathode, any change to this node also causes a strong, often non-linear, change in the amplitude of the driving-pulse. Therefore, the fine-tuning of the VCSEL's bias is done on the anode. Initially the bias was provided by a standard benchtop Agilent E3613A triple power supply. Sections 6.2 and 6.3 detail the results of this system which, partly due to the instabilities of the power supplies, displayed noticeable drift. Circuit schematics and PCB-board layouts of the pulse-forming, VCSEL-driving, and timing-window circuits are shown in Appendices E, G, and H.

6.2 Initial Implementation & Results

Photographs of the first iteration of the LLQRNG are shown in Figure 6.3 and Figure 6.4. The circuitry containing the 'not-ready' and 'data' flip-flops, as well as logic translation, power supplies, and other auxiliary circuitry is labeled 'Readout and Timing Window' in Figure 6.3. The green looped cable is used to set 'delay 2', or end of the timing window. The VCSEL and circuitry which forms its amplified electrical driving pulse, are mounted on the temperature-controlled aluminum plate in Figure 6.4.

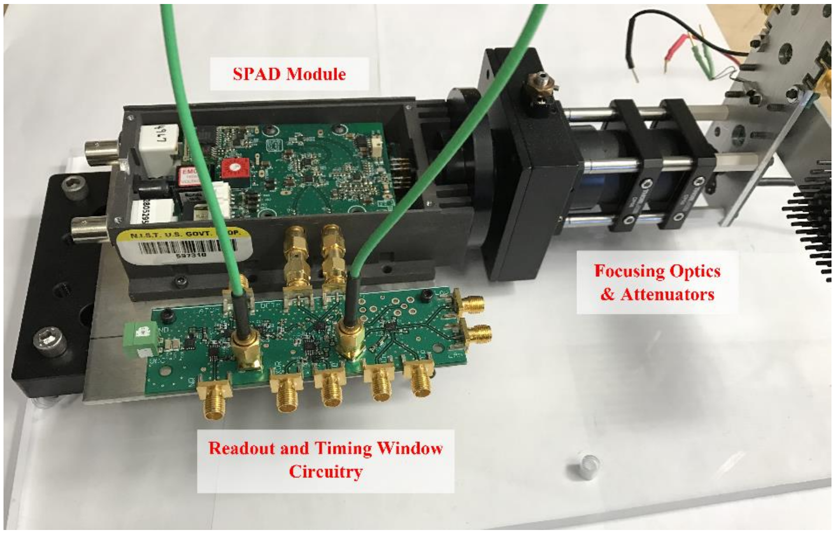

SPAD Module
Focusing Optics & Attenuators
Readout and Timing Window Circuitry

Figure 6.3. Side-view of initial LLQRNG implementation. The input laser pulse is formed by electronics located behind the aluminum plate (far-right) and output light travels through the short lens-tube assembly. The avalanche is detected by the SPAD module and output through the side of the module's box into the readout and timing board. The duration of the timing window is set by the looped green cable (length ~0.3 m), and bit-values are output through the SMA connectors.

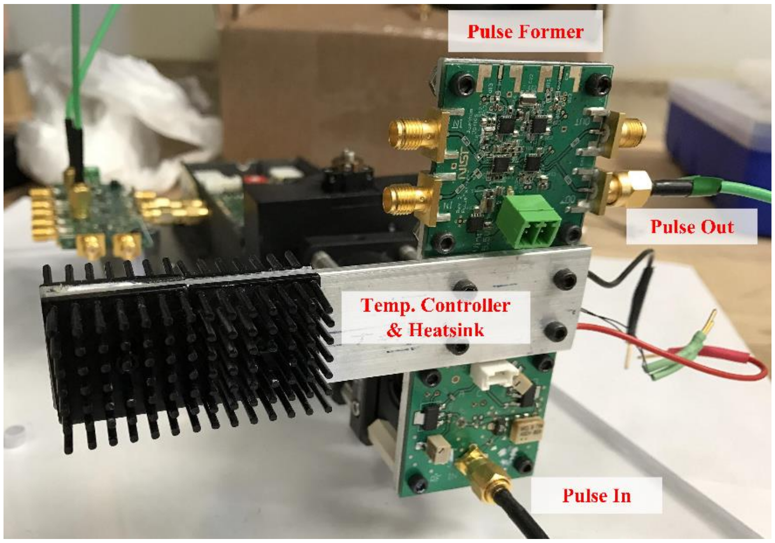


Figure 6.4. Front view of the initial LLQRNG implementation, depicting the circuitry which forms the electrical pulse which drives the VCSEL. The green SMA cable is used to synchronize the arrival of photon detections with the response time of the readout board from Figure 6.3. Due to the extreme temperature sensitivity of these electronics, both boards were mounted on a thermally-controlled aluminum plate.

Triggered at a bit-request rate of 99.1 kHz, a sample of random bits was recorded for approximately three days. The probability of a random '1' bit-value was calculated by the equation $\frac{P(1)}{P(1)+P(0)}$, where $P(1)$ and $P(0)$ are the number of recorded random '1' and '0' outputs, respectively. Several times in this chapter,

$P(1)$ is also referred to as the “bit-probability” of the LLQRNG. The per-second samples were then averaged into 100-sample blocks to reduce noise, the results of which are shown in Figure 6.5.

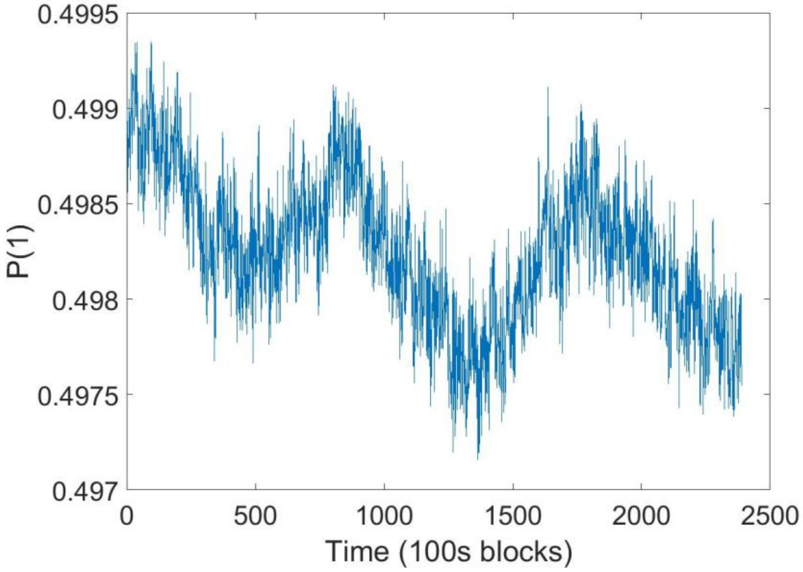


Figure 6.5. Sample 100-second averages from initial LLQRNG implementation, measured over three days. Instabilities are seen in the form of a slow drift in recorded bit-probabilities.

Immediately apparent in the above data is the tendency of the system to drift over time (with a typical level of ≈ 0.1% per day). The coarseness of the initial tuning done by the bench-top power supply prevented us from starting at P(1) = 0.500. Slight long-term drifts due to aging are not uncommon in electronics, but the observed periodic structure did not indicate that aging was the most likely cause. We performed various tests, but the most revealing came in the form of an autocorrelation analysis.

The autocorrelation function is a measure of the correlation of a signal with a delayed copy of itself. Given a time lag τ, the autocorrelation function $R(\tau)$, of a signal $f(t)$, is given by the equation

$$R(\tau) = \int_{-\infty}^{+\infty} f(t) * f(t-\tau)\, dt. \qquad \text{Eqn. (6.1)}$$

For a “white-noise” random signal, there should be no correlation between samples taken at various times. Therefore, the autocorrelation function of a random signal, taken over a series of time lags, should only contain correlations at a time lag of zero, in the case where the signal is compared directly to itself. An example of an autocorrelation from a pseudo-randomly generated signal, and the autocorrelation of the 3-day data set in Figure 6.5, are shown in Figure 6.6.

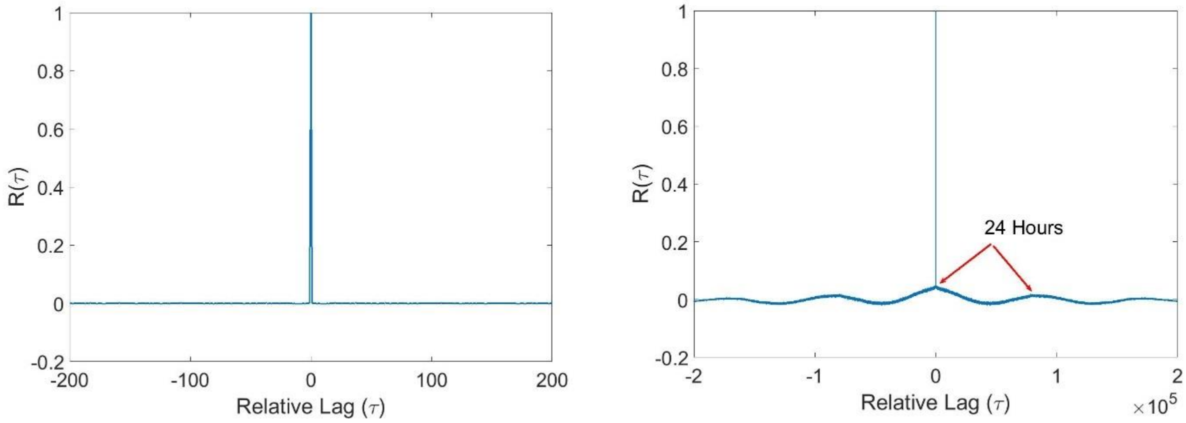


Figure 6.6. Autocorrelation analysis of a pseudo-random number generator (left), and the LLQRNG (right). The only peak for the former is one at a time lag of 0 seconds, consistent with a bit stream with no frequency correlations. The LLQRNG's data clearly displays correlations, with an oscillatory structure of 24 hours ± 5 minutes.

The autocorrelation of the white-noise signal contains only a single peak, as expected. However, clearly visible in the LLQRNG data are substantial correlations, denoted by the non-flat baseline of $R(\tau)$. Analysis revealed that the period of this structure was (24 ± 0.1) hours, compelling evidence that the bit-probability was being influenced by environment conditions.

By systematically applying temperature gradients to various physical areas of the LLQRNG and monitoring its output, it was determined that the most temperature-sensitive components were the VCSEL itself, and the electronics responsible for the preparation of the VCSEL-driving pulse. This was not particularly surprising, as the lasing threshold-current for a VCSEL has a quadratic temperature dependence [116], and the number of emitted photons per pulse is directly proportional to the number of injected electrons. However, it did highlight the need for a much higher degree of control over the system's environment.

We attempted to further temperature stabilize the system by enclosing the VCSEL and its driving-electronics in passive thermal insulation, and thermally connecting each to an actively-controlled TEC controller. This only slightly improved performance, and as shown in Figure 6.7, there was still a clear correlation between temperature and bit-probability.

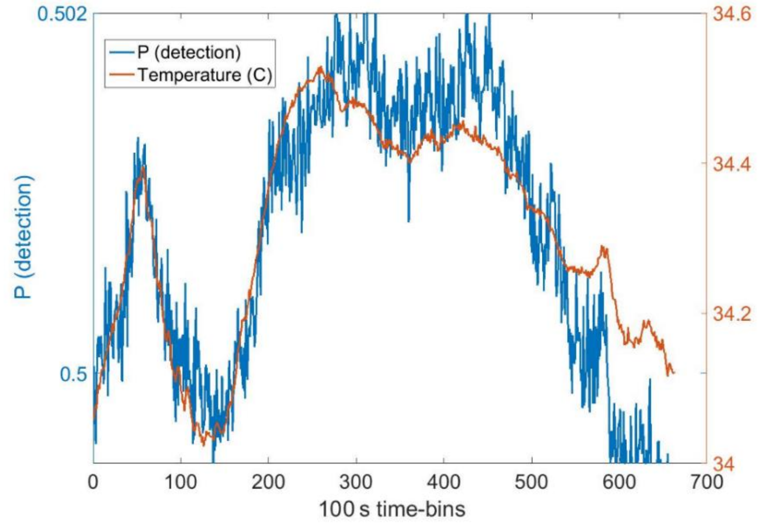


Figure 6.7. Bit-probability measurement for LLQRNG system with the VCSEL and pulse-forming electronics temperature controlled and enclosed in passive-insulation. Although the rate of change over time is slightly less, there is still a clear temperature dependence.

A more thorough characterization revealed that essentially every component was, at some level, temperature sensitive. The resistors used in every constructed circuit board were of the thick-film variety, which typically have a temperature coefficient of ≈ 100 ppm / °C. When used to set the thresholds on the timing-window electronics, for example, a temperature gradient of 1°C was enough to change the duration of the timing-window by tens of picoseconds. The performance of the optical attenuators (enclosed in lens tubes) is also affected, as slight changes in the room temperature affect the absorption coefficient of the filter's glass [117].The precision voltage regulators used to generate the power supply for every digital chip were also temperature sensitive, with variations on the 10^{-4} V scale. Shown in Figure 6.8 is the output of one such voltage regulator over approximately 7 hours. It was determined that the period of output oscillations was synchronous with the timing of the AC-cooling unit in the lab, approximately 20 minutes. As changing the power supply (Vcc) to a digital chip modifies both its timing characteristics and logic-switching levels, the behavior of the whole circuit was adversely affected.

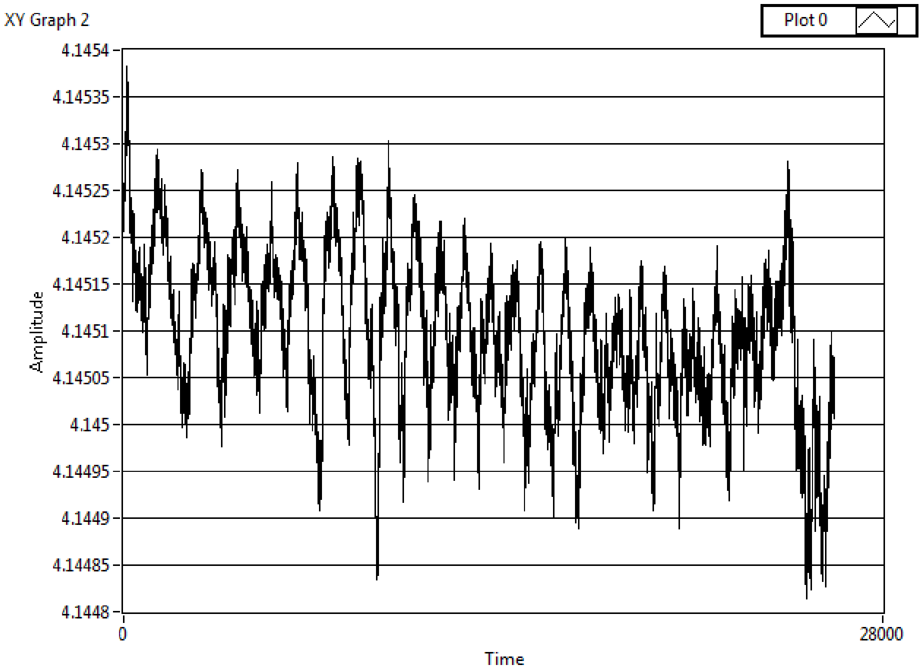


Figure 6.8. Measured voltage output of a precision voltage regulator over an approximately seven-hour period. Sub-degree changes in room temperature due to the AC-unit caused oscillations of the same frequency in the output voltage, adversely affecting the timing behavior of the LLQRNG.

To further decrease the sensitivity to the outside environment, the entirety of the LLQRNG was enclosed in an acrylic box. The outside was covered with adhesive passive insulation, and the top was sealed with silicone gel after the device was initially tuned. Every connection to the outside is achieved through SMA-connectors fed through holes in the box, which are later sealed. Every constructed PCB board, as well as the SPAD, was thermally connected to an 1/8” thick aluminum plate which was controlled by a high-power TEC stage located under the SPAD module. The resistors on the more sensitive portions were replaced with thin-film components, reducing their temperature coefficient by approximately a factor of four. In this way, the temperature influences were minimized, and every component can be stabilized by a single temperature controller. An external and internal view of the improved temperature-stabilized LLQRNG is shown in Figure 6.9.

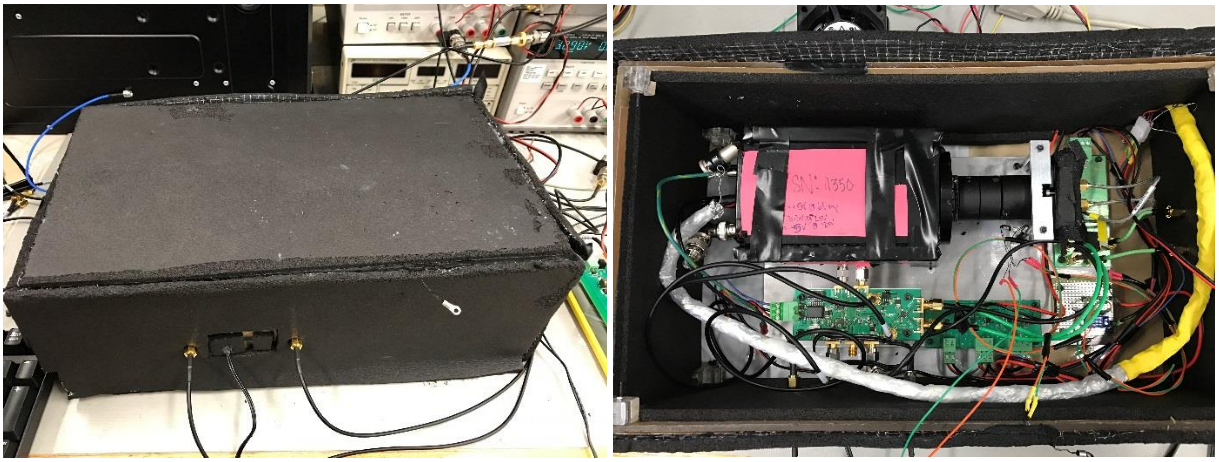


Figure 6.9. Exterior (left) and interior (right) of insulated and temperature-controlled LLQRNG system. Clockwise from top-left is the SPAD module (pink label), focusing optics in lens tubes, and the laser vertically mounted on an alignment slip-plate. The pulse forming PCB is the green PCB in the upper right, while three thermistors interfaced with an exterior microcontroller through the breadboard in the lower right corner. The voltage regulators for each component (bottom-right-middle) are bolted to the aluminum plate for temperature stabilization, as well as the SPAD readout board (lower-left).

6.3 Temperature Stabilization Results

The LLQRNGs results were improved when the entirety of the system was temperature stabilized, with plots of output probabilities over a 12-hour period shown in Figure 6.10 in one- and one-hundred-second averages. The 24-hour structure of Figure 6.6 was not present in autocorrelation analysis, but a clear trend was still visible. We hypothesized that this was due to the interior of the box needing to thermalize and reach equilibrium, but the effect persisted even after being allowed to stabilize for several days.

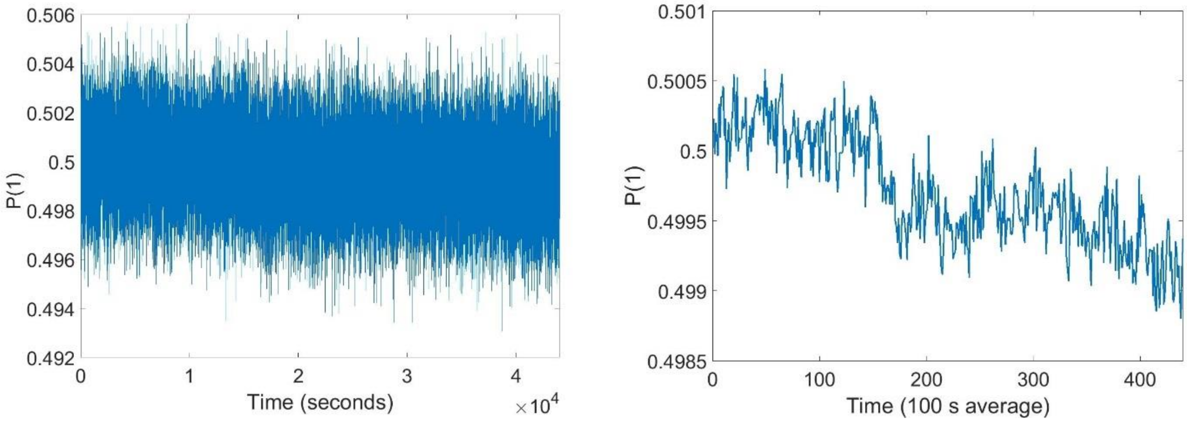


Figure 6.10. Bit-probability measurement of enclosed temperature-stabilized LLQRNG in 1-second (left) and 100-second (right) averages. Although significantly reduced, the presence of long-term drift is still apparent.

To determine if this drift could still be primarily attributed to the outside temperature, two measurements were performed. For the first, we opened the box and disabled the temperature controller. A temperature sensor was placed inside the box and the bit-probability measured over another 12-hour period. For the second, the box was re-sealed as before, allowed to equalize for over 24 hours, and three temperatures were recorded: two inside the sealed box and one outside in the uncontrolled environment. Figure 6.11 shows plots of the bit bias versus temperature (averaged over 100 seconds), for the uncontrolled and controlled setups.

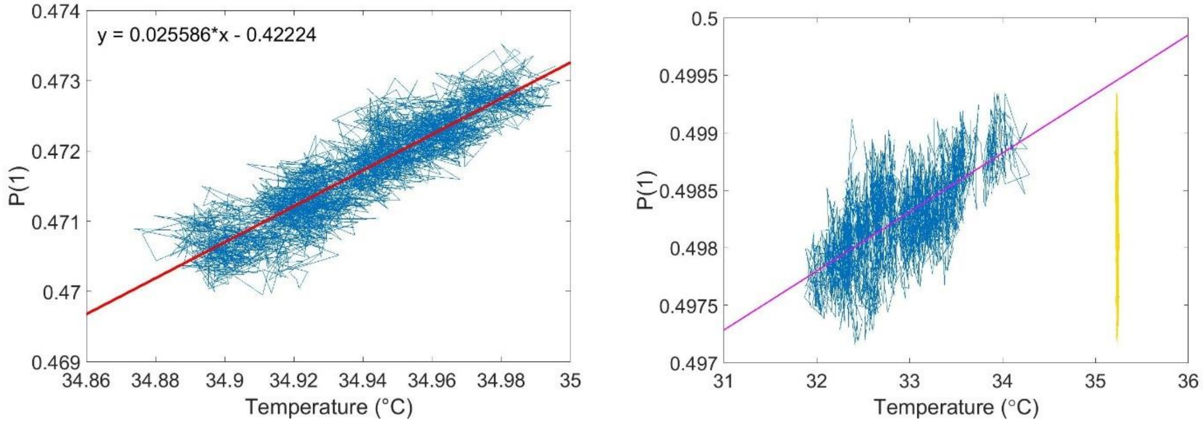


Figure 6.11. Temperature / bit-probability relationship for the uncontrolled (left) and insulated, temperature-controlled (right) LLQRNGs. The slope of the linear-fit estimates the magnitude of the outside temperatures influence, which is reduced by several orders of magnitude under careful environmental control.

The uncontrolled implementation shows a strong linear-relationship (≈ 2.5% / °C) between temperature and bit-probability, a situation clearly not acceptable for laboratory operation. The temperature-controlled setup displays a greatly reduced correlation, with the temperature outside the box (blue) having an estimated ≈ 0.051% /°C magnitude of influence when a linear-fit is used. The VCSEL's mounting block (yellow) appears to be very temperature stable. Over the entire three-day dataset, the full range of recorded temperatures was [35.21, 35.24] °C, and there was no discernable bit-probability relationship as there is with the outside temperature.

6.4 Precision-Feedback Capability

Even after extensive temperature-stabilization efforts, the performance of the LLQRNG was not sufficient. Long-term drift was visible (Figure 6.10), and there was no clear relationship between recorded temperatures and the bias-drift. Although it is certainly the case that temperature-dependent effects are present, they have seemingly been reduced to levels at the limit of our current capability to control. What is also likely, however, is that some combination of unknown effects remains (i.e., µV-scale drift of the SPAD bias, mechanical instabilities, etc.). As it is unrealistic to fully model every intricacy of a complex physical system, compensating for small-scale inaccuracies is often done with some degree of active feedback.

The application of feedback must be done in a fashion which is repeatable and minimally disrupts the existing temperature stabilization. While the final probability of detection could certainly be tuned by changing the amount of optical attenuation, this would require opening the enclosure and removing the lens tubes, a process clearly not suitable for uninterrupted operation. The length of the timing window could be adjusted with the aid of digitally programmable delay chips, but we have found these devices to be especially unstable. Therefore, for our active-feedback purposes we have chosen to use the DC-bias across the VCSEL, which has a linear relationship to the output photon flux for a laser operating well-above threshold.

To implement the feedback, we added a precision digital-to-analog converter (DAC), a microcontroller (µC), and a summing amplifier (+) to the original design, as shown in Figure 6.12. Widely used in precision voltage control, a DAC can tune its output-voltage level in a linear and repeatable fashion. We employed an AD5663-R nanoDAC, a device with 5 volts of range, 16-bits of resolution, and a 5-ppm/°C temperature coefficient. This allows for tuning of the VCSEL's bias with a resolution of $(5 / 2^{16}) \approx 76$ µV.

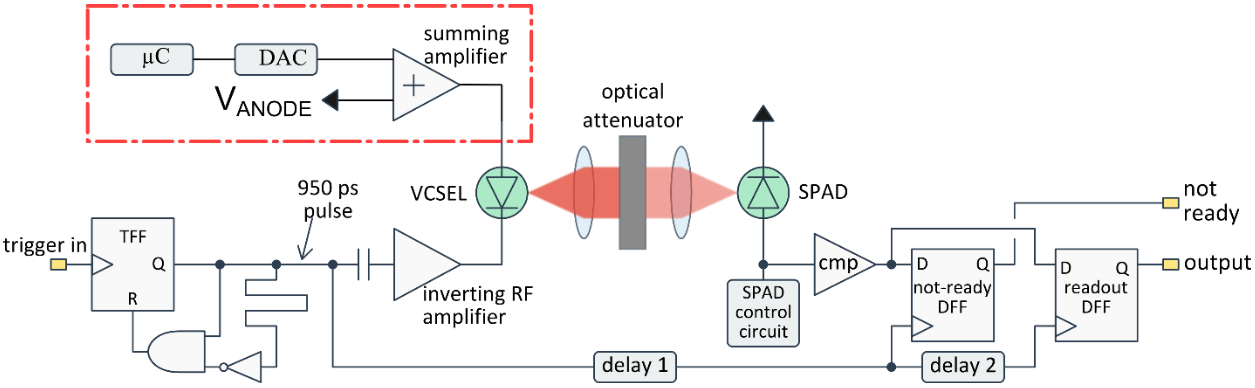


Figure 6.12. Illustration of modified LLQRNG setup. The amount of active-feedback required is calculated on the microcontroller (µC), which programs the digital-to-analog converter (DAC). This allows for fast and repeatable tuning of the VCSEL's forward-bias, and thus the probability of detection.

In the initial system, the VCSEL was operated with both the anode and cathode DC-biased at ~4.1 V. The total bias of zero ensured that no light would be emitted until the presence of the amplified electrical pulse. The choice of ~4.1 V was made after observing the optimal behavior of the GALI-51 inverting RF-amplifier, a device whose amplification-level depends on the DC-bias at its output, the cathode. Because any change in the cathode's bias causes a non-linear response magnitude of the amplified pulse, the DAC was coupled directly to the anode.

In this configuration, a 10 mV change in the anode bias results in a many-percent change in the probability of detection, so tuning the bit-probability with the complete 0 – 5 V range of the DAC is unnecessary. An op-amp-based summing amplifier was added, and the DAC's output divided down by a factor of sixteen. This reduced the range to 0 – 312 mV, but increased the resolution to 4.75 µV / bit. The digital bit-value was set with a Teensy v3.2 microcontroller, and was easily adjusted with already existing Python code-libraries. In this configuration, with V_{ANODE} kept at ~3.8 V, the output of the summing amplifier (over the full 65536-bit range of the DAC) is 4.100 ± 0.156 V. A circuit schematic of the improved laser-driver circuit, and details on the summing amplifier, are given in Appendix E.

This improved circuitry resulted in the capability to tune the average probability of detection by ~0.00013%, a measurement of which is shown in Figure 6.13. The bit-request rate was set at 100 kHz, and the DAC internal bit-value was varied over its 65535-bit (~312.5 mV) range. As the DAC-bit increased, the forward-bias across the VCSEL, the output photon flux, and thus the final probability of detection increases. The linearity was diminished at high count rates (> 90 kHz) as the laser began to saturate, and for device safety, values which resulted in a forward-bias near the maximum-rated specification were not tested. The y-intercept of -2 in the linear fit indicates that the voltage divider could be increased even more and still encompass the full range of probabilities, but this improvement has not yet been implemented.

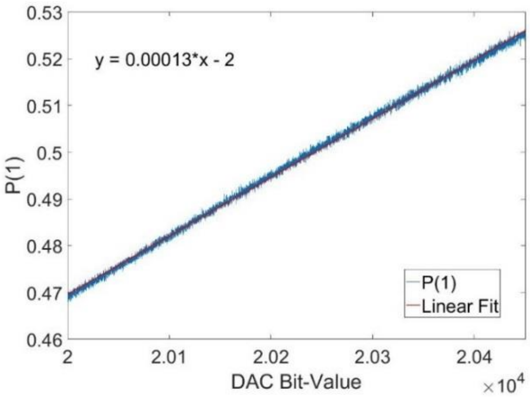


Figure 6.13. Measured relationship between DAC bit-value and output bit-probability over typical operating ranges. With the aid of an op-amp summing amplifier and a divided-down DAC, fine-tuning of the detection probability is possible with ~0.013% accuracy.

6.5 Evaluation of Bit-probability

To determine if the probability of detection is properly balanced, the output of the LLQRNG is coupled to a microcontroller. An interrupt-handling routine was created to respond to rising-edges of both random-bit value channels. Every 100,000 bits, the bit-probability was calculated by the simple formula $\frac{P(1)}{P(0)+P(1)}$.

As the LLQRNG outputs can only take two values, each bit is a Bernoulli random variable, with possible output values of logical-1 or logical-0. Assuming that each bit is independent and identically distributed, a collection of 100,000 bits is a binomial random variable with N = 100,000 and probability of success *p*. If this measurement is repeated many times and the LLQRNG is properly balanced, then a histogram of bit-probability measurements will take the shape of a binomial distribution with μ = 50,000 and σ = $\sqrt{100000(0.5)(1-0.5)}$ ≈ 158.1139, as shown in Figure 6.14. Due to the wide range of 100,000-bit sums that even a correctly-balanced random process can generate, measuring the bit-probability from the random output itself results in an extremely noisy measurement.

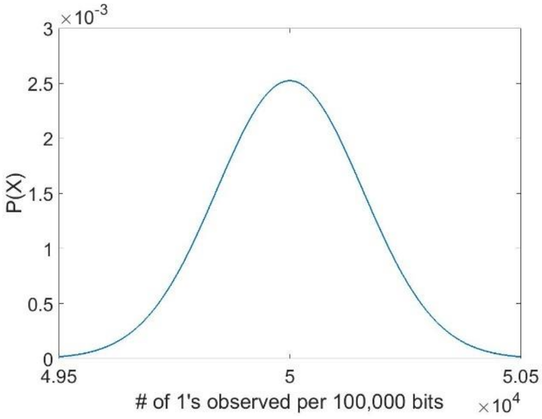


Figure 6.14. Expected probability distribution function for a single 100,000-bit sample of a properly balanced LLQRNG.

The degree of confidence in our bias-estimate can be improved by averaging over multiple samples, a direct application of the central limit theorem. If instead of only considering single measurements of 100,000-bits, we instead average them into n-sample blocks, an estimation of μ can be made with less uncertainty. In particular, the distribution of n-sample averages will retain the original mean μ, but the new standard deviation σ_n will be reduced by the factor $\sigma_n = \sigma/\sqrt{n}$. An example of the expected distributions from a perfectly-balanced binomial process taken over 1, 10, and 100 sample averages is shown in Figure 6.15.

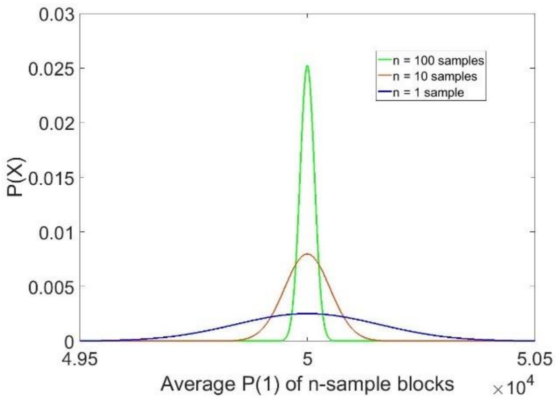


Figure 6.15. Expected probability distribution function for a binomial-distribution when averaged into 1, 10, and 100-sample blocks.

While it is certainly true that averaging for longer periods on an exclusively-binomial process will always give a more accurate estimation of the bit-probability, this is not necessarily the case for our LLQRNG. Due to the additional long-term drift, there an additional influence also affecting the mean. At some point, the increased confidence gained by longer averaging periods will be overshadowed by the error introduced by the instability. Therefore, a careful characterization must be made of the level of drift present.

6.6 The Allan-Variance

The classical N-sample variance, σ^2, of a set of discrete data points *y,* is given by the equation

$$\sigma^2 = \frac{\sum_i (y_i - \mu)^2}{N}, \qquad \text{Eqn. (6.2)}$$

and is the expected value of the squared deviation from a sample's average μ. Although widely used in statistics to characterize how much a variable tends to change, it is not as useful in frequency analysis. In the presence of zero-mean noise, i.e., oscillatory centered around μ, then the classical variance will converge. However, if the noise is divergent, as is the case with the LLQRNG's drift, then a running-average of the process's mean will also be divergent, and not provide a stable point of reference.

Modern frequency stability analysis emerged in the mid 1960s with the invention of several improved analytical and measurement techniques. In particular, the two-sample variance [118] was introduced by David Allan, formerly of NIST's Time and Frequency Division. Later named the Allan-variance (ADEV), it is one of the most common time-domain measures of frequency stability. There have been many subsequent variations (Hadamard [119], overlapping, total, etc. [120]), but the original non-overlapping Allan variance is defined as

$$\sigma_y^2(\tau) = \frac{1}{2(M-1)} \sum_{i=1}^{M-1} [y_{i-1} - y_i]^2, \qquad \text{Eqn. (6.3)}$$

where y_i is the *i*th of M fractional frequency values averaged over a sampling interval, τ. Also related to the Allan variance is the Allan deviation (ADEV), $\sigma_y(\tau)$, which is the square root of the Allan variance. The strength of σ_y lies in the fact that it is no longer referenced to a single mean μ, but rather differences are taken at a variety of averaging intervals. By analyzing the variances at different τ, the time-scales on which the signal's stability are affected can be determined.

The most straight forward example of the Allan deviation comes from measuring zero-mean white noise. In this instance, there are no correlations between data taken at differing averaging intervals, and the Allan deviation at τ = 1 is equal to the standard deviation. As τ is increased, and longer averages are considered, the deviation is reduced by a factor of $\sqrt{\tau}$, identical to the normally distributed values in Figure

6.15. If the square-root dependence on τ holds, then a log-log plot of the Allan deviation versus τ will display a slope of -0.5. The Allan deviation of MATLAB's pseudo-random number generator, grouped into 100,000 bits, is shown in Figure 6.16 top, with the difference between the measured Allan deviation and the ideal line indicative of white noise shown on the bottom graph. The slope of the PRNGs $\sigma_y(\tau)$ was approximately -0.5, as expected, up to the maximum measured averaging period of 10000.

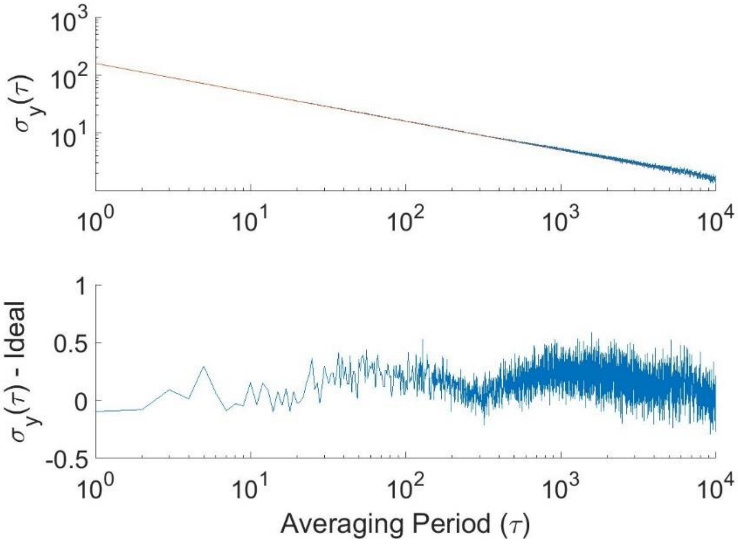


Figure 6.16. Allan variance of MATLABs PRNG (top), and the difference between a white-noise indicative, log-log line of slope -0.5 (bottom), when considered as samples of 100,000 bits.

We now consider the random '1' output of our LLQRNG as a clock-type source with expected frequency (per 100,000 bits) of 50,000, and one which should ideally only display white-noise-type characteristics.

The same analysis was performed on multiple large data sets taken from our LLQRNG, without the active-feedback enabled. The amount of deviation from the characteristic white-noise (red-line) gave important clues towards the strength and origin of the long-term drift. A plot of the Allan deviation of one 7-day dataset is shown in Figure 6.17, out to an averaging interval of 10,000.

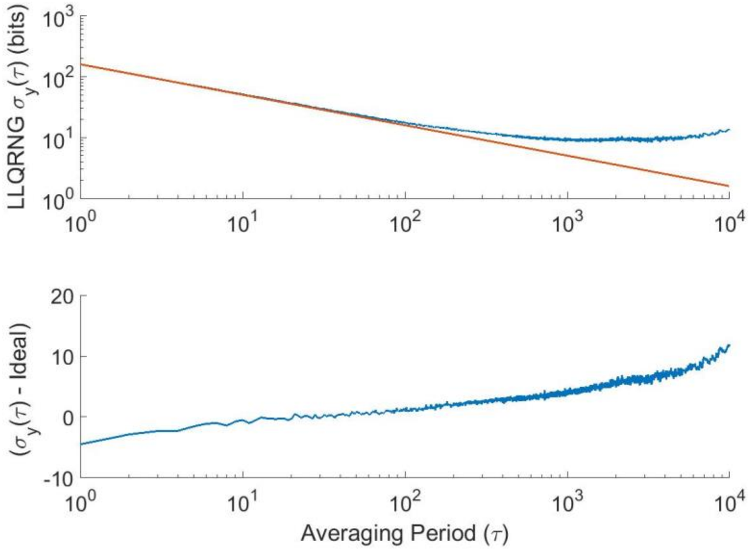


Figure 6.17. Allan deviation of the LLQRNGs random-bit output. The Allan deviation at τ = 1 is equal to the classical standard deviation, which is referenced to the overall sample mean and is divergent for certain types of signals.

Both the PRNG's and LLQRNG's Allan variance have a $\sigma_y(\tau = 1)$ of ≈ $\sqrt{100000 * (0.5)(1 - 0.5)}$ ≈ 158.11. In fact, when comparing the measured data to the expected white-noise line on the top log-log plot, there is no significant visual difference until averaging periods greater than 100 samples. When the residual is examined, however, a clear trend appears: the Allan variance of the measured random data immediately diverges from that of white noise, and seems to grow with an approximate $\sqrt{\tau}$ dependence.

6.7 Drift Characterization

Throughout extensive characterization of many frequency sources, it has been found that most instabilities can be approximated by a combination of four noise types: white noise, flicker (pink) noise, random walk (brown) noise, and flicker-walk (black) noise [120]. Ideally, the output of the LLQRNG would be exclusively white noise, however, there are clearly other effects present. Determining the cause of our bias instability is crucial when determining how to effectively apply any corrective active feedback.

6.7.1 Overall Allan Deviation

The Allan deviation is only an estimator, but it can be extremely useful in identifying the different components of a time-domain signal. Each of the common noise types previously mentioned have different frequency characteristics, and contributes to the ADEV in a specific fashion [120]. By assuming that the measured output of the LLQRNG is the sum of white noise and some other unknown noise, the

ADEV of our LLQRNG can then be examined to estimate which noise-type most closely matches the cause of our bias-drift. The ADEV of Figure 6.17 has a clear region of slope close to -0.5, indicating the time scales at which white-noise dominates. There is a region in which the slope is approximately zero, which is indicative of the time-scales at which the white-noise and the other, unknown, noise are of approximately equal magnitude, but the behavior afterwards was unclear. Another ADEV analysis was performed, on the same dataset, out to a longer τ of 50,000 seconds. As the total data set was only ≈ 600,000 seconds, there is a high degree of statistical uncertainty at the larger averaging periods, but the value of $\sigma_y(\tau)$ does seem to converge to a line of slope +0.5 on the log-log graph of Figure 6.18,[10] a behavior characteristic of 'random-walk' frequency noise [120].

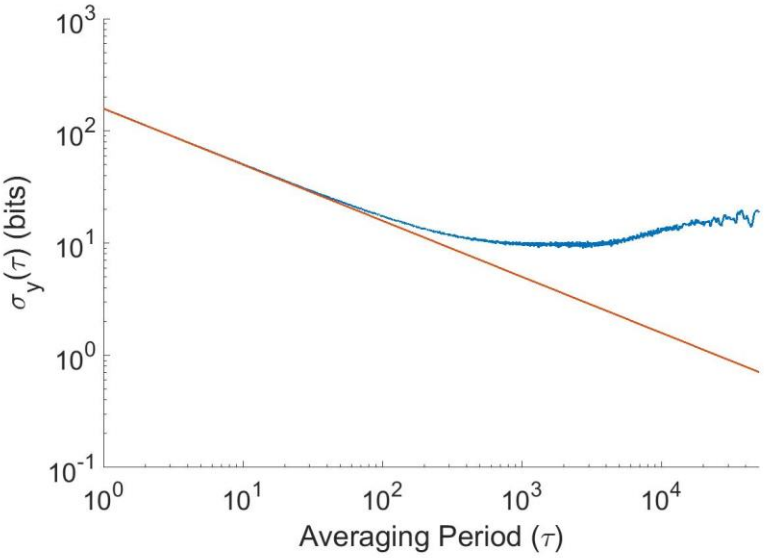


Figure 6.18. ADEV analysis of the dataset in Figure 6.17, averaged to a maximum τ of 50,000 seconds. Although the total number of data points limits the statistical certainty at larger τ, the slope of the ADEV seems to trend towards +0.5, a slope characteristic of random-walk frequency noise.

Random-walk frequency noise (brown-noise) is a random process, modeled by Brownian motion, defined in continuous time as $y(t) = \int_0^t \beta(t')dt',\ t > 0$, where $\beta(t)$is white Gaussian noise with zero mean and autocorrelation function $R_\beta(t_1, t_2) = q_3\delta(t_1 - t_2)$. In other words, a random walk can be modeled as the integral of white noise. Perhaps a more intuitive description is that a random walk is a process for which the current value $y(t)$ is the sum of a past value $y(t-1)$, and a 'walk' term w, or

[10] A best-fit was performed on the positive-slope section of Figure 6.18. The estimated (log-log) slope was 0.34 ± 0.04.

$$y(t) = y(t-1) + w, \qquad \text{Eqn. (6.4)}$$

where w is a value sampled from a white-noise process, with zero-mean and variance q_3.

The expected value of the Allan deviation for individual noise components has been analytically calculated [121]. In particular, the expected values of the ADEV estimator for white-noise and random-walk processes, with respective variances q_2 and q_3, are defined to be:

$$\sqrt{\frac{q_2}{\tau}} \qquad \text{Eqn. (6.5)}$$

$$\sqrt{\frac{q_3\tau}{3}}. \qquad \text{Eqn. (6.6}$$

The measured ADEV of the LLQRNG was fit to the sum of Eqn. 6.4 and Eqn. 6.5. using MATLABs curve-fitting toolbox, where the value of q_2 was assumed to be the expected variance of 25,000. A comparison of the measured LLQRNGs ADEV results (red) and the MATLAB best-fit (blue), as shown in Figure 6.19. According to these results, the initial estimate is that the LLQRNGs frequency measurement is the result of a combination of white frequency noise (with std. deviation $\sqrt{q_2} \sim 158.11$ bits) and a random-walk process (with std. deviation $\sqrt{q_3} \sim 0.07$ bits).

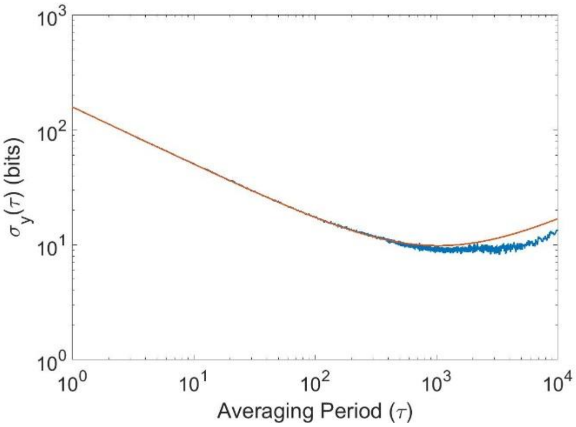


Figure 6.19. Allan deviation of a LLQRNG bit-probability measurement (blue) versus the simulated sum of a white-noise and random-walk process.

6.7.2. Dynamic Allan Variance

The theoretical random-walk + white-noise model of Figure 6.19 fits reasonably well with our observed results, but there is clearly a divergence at longer averaging periods. Just as the classical variance's disadvantage lies in its reference to a total sample-mean, the overall Allan variance is also an averaged quantity. If there exist short-term effects within relatively large datasets, there is the possibility that they can become distorted, or even averaged out, when longer averaging periods are considered. For example, the signal of Figure 6.20 is a stationary Gaussian white-noise type signal, and its Allan deviation has the expected slope of -0.5. In the signal of Figure 6.21, however, there is clearly a time-varying component to the variance; however, when averaged over the whole data set is essentially identical to that of Figure 6.20, and the calculated ADEV of the signal does not show any evidence of the increased-variance region.

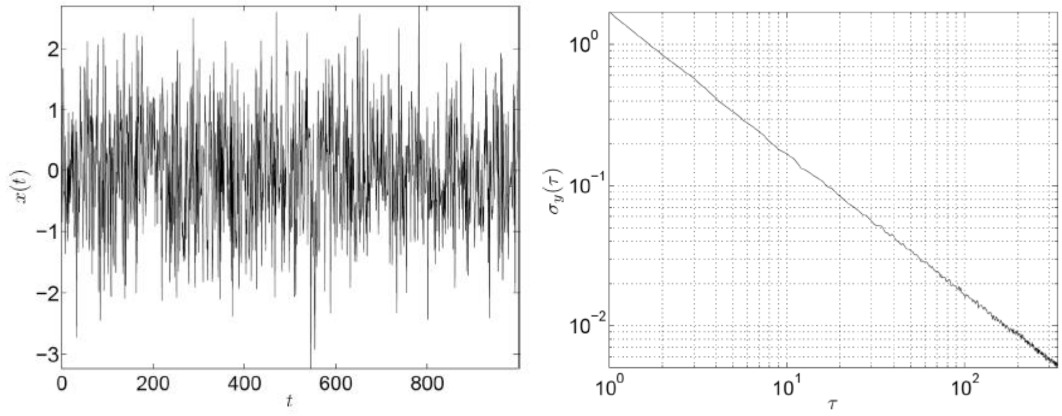


Figure 6.20. Constant-variance white-noise signal (left) and its calculated ADEV (right). Figures courtesy of *[122]*.

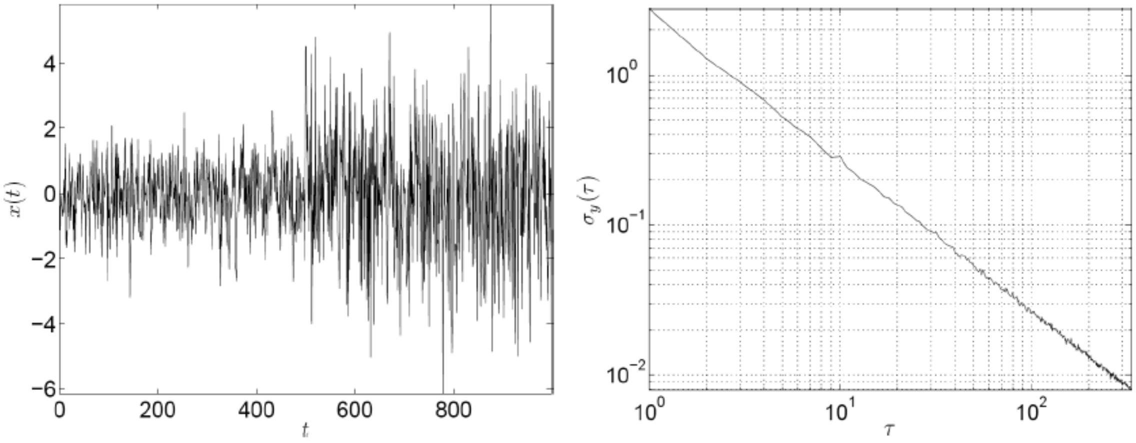


Figure 6.21. White-noise signal with differing variances (left). Calculated ADEV (right) does not reveal this effect. Figures courtesy of *[122]*.

Also developed for use with atomic clocks, the *dynamic* Allan variance (DAVAR), is a quantity that can better represent the *time-varying* stability of a frequency source. Because physical devices are influenced by environmental conditions (such as temperature and humidity), they can undergo sudden failures, or simply age with time.

The calculations for the DAVAR are identical to the ADEV, but are repeated multiple times to consider a single dataset in terms of shorter windows. Specifically, given a signal $x(t)$ and window size T, the standard ADEV $\sigma_y(\tau)$ is calculated for only the first T samples in $x(t)$. The original signal is then again considered, but the window is shifted by T+1, and $\sigma_y(\tau)$ is again calculated.[11] Therefore, the DADEV is a series of ADEVs taken at different points in the overall signal. By only considering small portions of $x(t)$ at a time, the averaging-effect seen with the standard ADEV can be reduced, and short-term instabilities can be characterized. An example of the DADEV, obtained from the signal of Figure 6.21 is shown in Figure 6.22. Here, the non-stationarity of the variance is clearly visible, although the short window-size of 100 samples introduces a high level of uncertainty in the measurement.

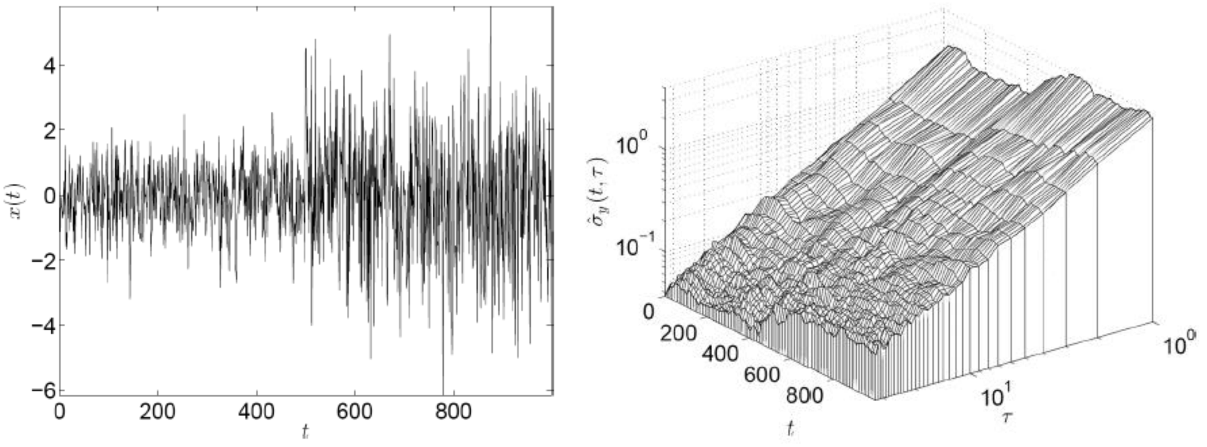


Figure 6.22. White-noise showing an increased variance in the latter-half of the signal (left). By considering the signal in terms of smaller 100-sample windows, the DAVAR calculation clearly shows the increased variance, although embedded in the uncertainty due to the small window size. Figures courtesy of *[122]*.

We calculate the DAVAR over a series of LLQRNG datasets, at various window sizes. If there were any sudden changes in bit-probability, then these types of effects would be revealed as sharp discontinuities in the DAVAR plots. If, however, the calculation produced roughly the same results over time, then it is

[11] The choice of whether to use overlapping or non-overlapping windows, as well as the optimal window size T, is a process that is still being explored.

not unreasonable to assume that the noise-type present in our system does not vary suddenly, but is rather a slow, gradual drift. The result of the DAVAR calculation, for a data set approximately 7 days long, is shown in Figure 6.23. In the 3-D mesh plot, 60 windows, each containing 100,000 1-second-samples are evaluated. The windows shift by 10,000 samples each, so there is overlap; a balance had to be found between window-length and window-overlap in order to reduce the uncertainty at longer averaging periods τ. The -0.5 slope characteristic of white noise seems to hold for all windows until averaging periods of τ > 100, at which point the random-walk noise begins to dominate.

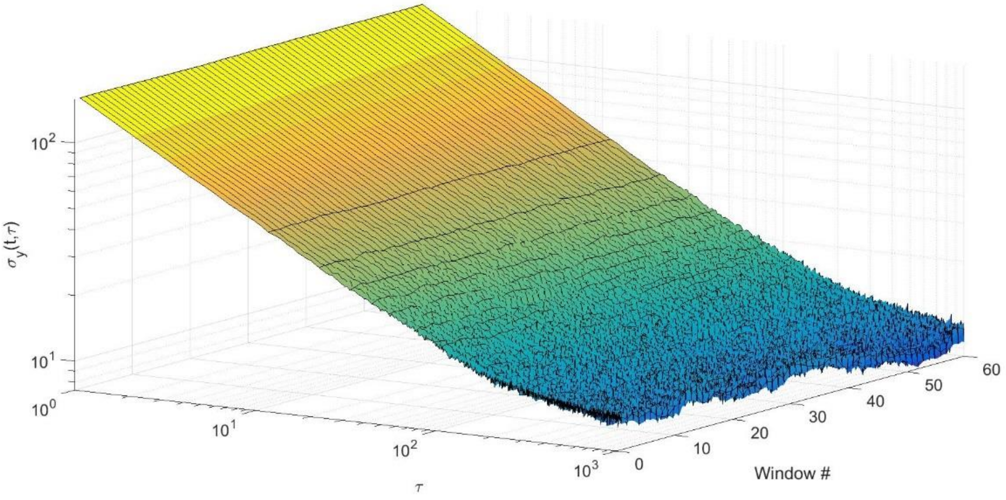


Figure 6.23. Dynamic Allan deviation of the LLQRNG's output. Although deviation from white-noise is clearly visible, the magnitude appears to remain approximately constant over the entire dataset.

A 2-D collection of each individual slice in Figure 6.23 is shown in Figure 6.24, to further illustrate the consistency over the total data set. Again, each individual slice was fitted against the theoretically expected combination of white noise and random walk noise,

$$\sqrt{\frac{q_2}{\tau}} + \sqrt{\frac{q_3\tau}{3}}. \qquad \text{Eqn. (6.7)}$$

Here, $\sqrt{q_2}$ was assumed to be ≈ 158.11 as before, and the best-fit was configured to solve for q_3. Over all windows, $\sqrt{q3}$ resided in the range of [0.0416, 0.1037] bits, with an average of 0.07.

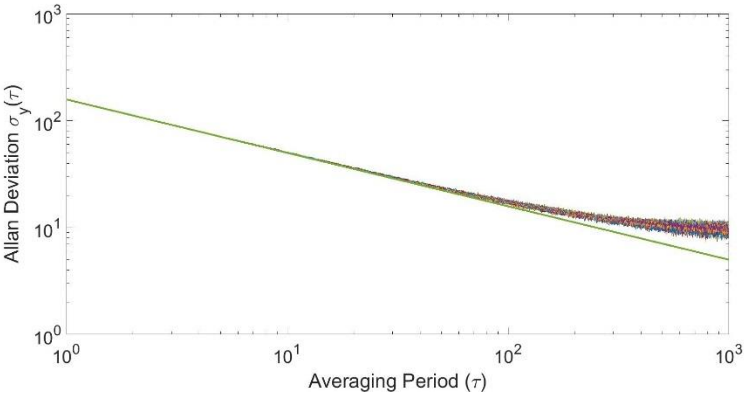


Figure 6.24. 2-d depiction of each 100,000-second line in the DADEV plot of Figure 6.23.

The behavior of several other long data sets, over a month of non-contiguous samples, showed similar results. Therefore, if our observations are indicative of the LLQRNG's behavior under ordinary conditions, we can assume several things. First, the additional noise we observe is gradual, with no sudden variations being revealed with our DAVAR analysis. Secondly, the noise is primarily of the random-walk type, the integral of a white-noise-type process itself, with no immediately observable correlation to other measurements we have made, i.e., temperature. Lastly, the influence of the noise on the bit-probability probability is very weak, affecting the bit-probabilities by only 10^{-6}-10^{-7} per second.

6.8 Application of Active-Feedback

The presence of weak, uncorrelated, random-walk type noise presents exceptional difficulties when trying to accurately apply correctional feedback. The only measurement currently used to determine the LLQRNG's status is repeated sampling of its random output, which is itself very noisy. As discussed in Section 6.5, the averaging of multiple samples improves confidence in the measurement, but must be done on time-scales for which the contribution from the drift does not overly skew the measurement. Also, we must take care that the averaging window is not too short, lest end up *reducing* the randomness.

For example, averaging a white-noise process for time *t* improves confidence in the estimate of the mean by a factor of $\sqrt{t}$. For our system, this is done by repeatedly measuring the number of random '1's in approximately one second of data, a number which ideally belongs to an approximately normal distribution, with mean $\mu = 50000$ and standard deviation $\sigma \approx 158.11$ bits. For such a distribution, 99.7% (3 standard deviations) of measurements will reside in a 950-bit range around the mean, so a single second of data does not reveal very much. However, in order to average for long enough to be over 99% confident that the current per-second rate of our random process is within one bit (or 50,000 ± 1 bit / s), an averaging period of 225,000 seconds (2.6 days), is required. This time-scale is clearly much too long, and the above calculation applies for a solitary white-noise process. The presence of our additional drift would clearly cause a greater than one-bit shift over that time-scale. In fact, given that the random walk process's variance grows with $\sqrt{t}$ dependence, while the white noise's decreases with $\sqrt{t}$, the point at which their variances become equal is approximately 600 seconds, as shown in Figure 6.25. Conceptually, this means that a 600-second running average of the LLQRNG's output will contain as much information from the white-noise process as it will from the random-walk, and averaging for longer periods becomes less effective.

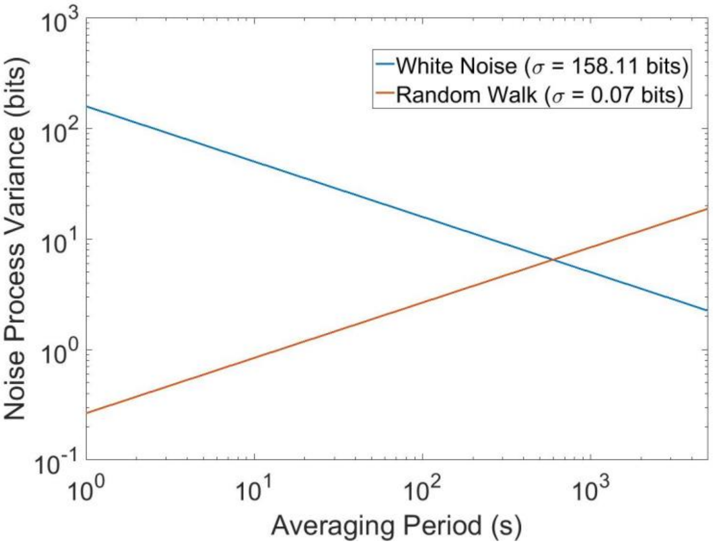


Figure 6.25. Variances of random-walk and white-noise processes when their distributions are averaged over time.

6.8.1 Moving Average Filter

With the relative strengths and time-scales of each process, one of the first feedback schemes we tried was a simple moving-average filter, an integral-based feedback. Given a measured bit-sequence $x(t)$ and window length $100s$, the current mean μ_{100} of the random process was estimated by the average of 100 successive bit-samples, each one second long. Given an expected bit-probability of 0.5, an error term was calculated by taking the difference, and a gain constant was applied to translate the error into an appropriate DAC-bit value. The amount of feedback applied was given by the simple formula

$$DAC_{fb} = gain(50000 - \mu_{100}). \quad \text{Eqn. (6.8)}$$

This scheme was applied multiple times, with variations to the window length, gain, and the amount each window overlapped with the next. Initially, the results looked quite good, as shown in Figure 6.26.

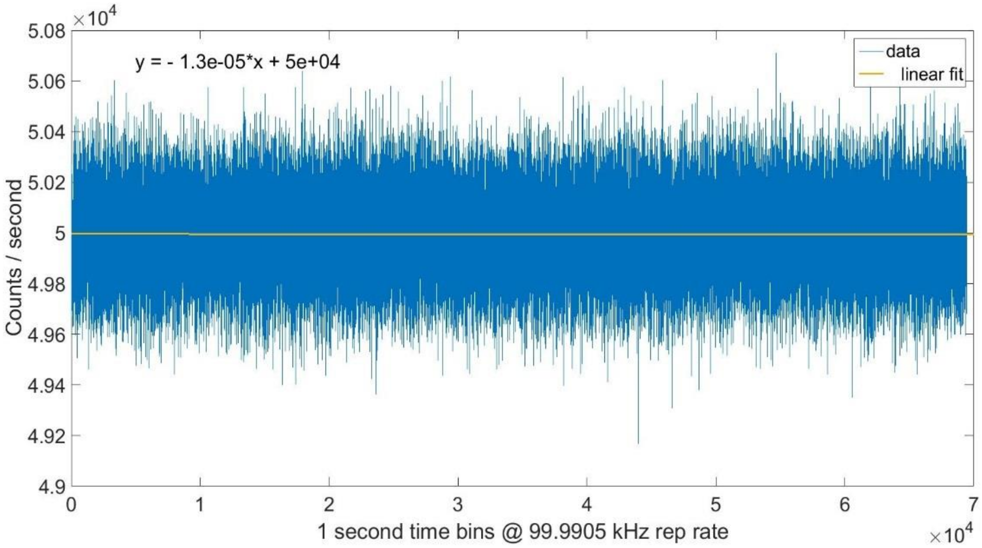


Figure 6.26. Measured bit-probability with a 100-second moving-average feedback filter applied. The small slope and y-intercept of the linear fit indicates a correctly tuned bit-probability.

For this ≈ 20 hour dataset, the measured mean bit-probability was 0.49999. For a stationary random process of mean 50,000, standard deviation ≈ 158.11, and averaging time of ≈20 hours, the 1.15-bit difference was 1.924 standard deviations away from the mean – a perfectly acceptable result. This dataset met the 50% bit-probability requirement, but additional analysis revealed a problem.

As shown in Figure 6.27, an autocorrelation measurement on the random bit stream showed correlations. As discussed in Section 6.2, a totally uncorrelated (white-noise) signal will have a single, one-sample wide peak at a relative lag of zero samples, i.e., when the signal is compared with itself. This type of integral filter, and any filter where the feedback applied is proportional to past measurements, will introduce correlations on time-scales relative to its averaging period.

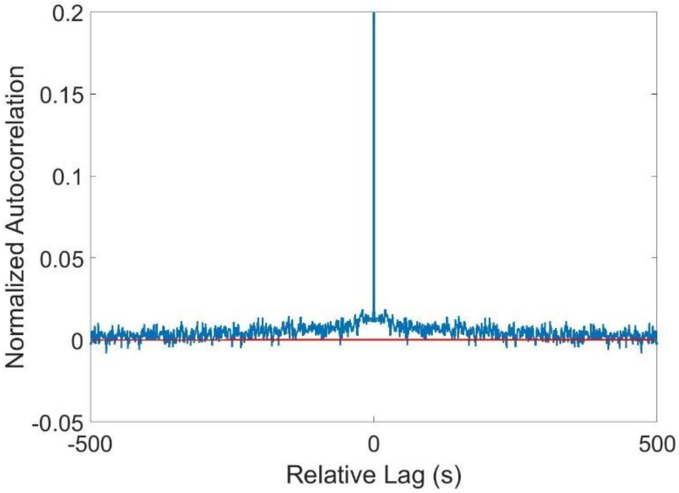


Figure 6.27. Autocorrelation of the LLQRNG with feedback applied according to a 100 sample-wide moving average filter. Ideally the value at all lags other than zero would be at zero (red-line).

6.8.2 Proportional Feedback

With the usefulness of averaging seemingly removed due to introduced correlations, a simple proportional-type feedback based on a single, one-second sample was attempted. This filter-type is analogous to a moving-average filter with window size equal to one, and was applied in the same fashion as described in Equation 6.8.

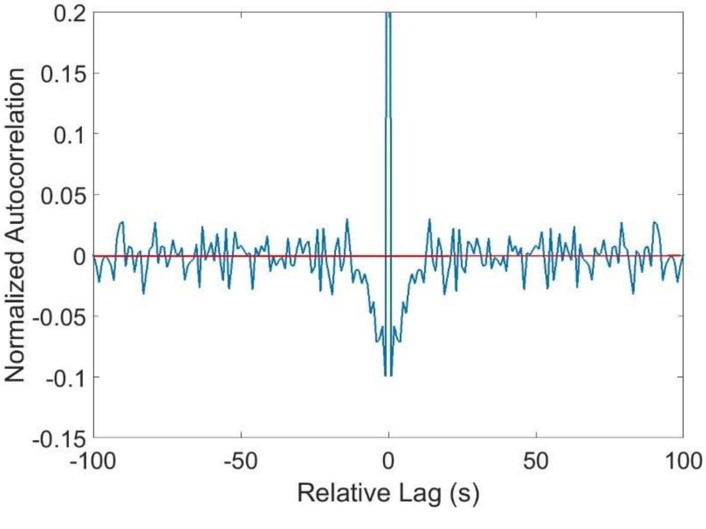


Figure 6.28. Autocorrelation of LLQRNG with simple proportional-feedback. Although the correlation between bits is larger, the time-scale at which they are reduced is much faster.

With a gain coefficient of 0.15 (1-bit of measured error results in 0.15 bits of correction), this feedback scheme performed similar to the moving-average. Although the time-scale at which bits remained significantly correlated was shortened (Figure 6.28), the ADEV analysis revealed a lower-than-expected variance at higher averaging periods, and a higher-than-expected variance at smaller averaging periods (Figure 6.29). Roughly speaking, this effect occurs because the feedback is overcorrecting the random bit stream. The overall probability distribution is widened, and the classical variance (Allan variance at τ = 0) is larger as a result. However, the unconditional action of the simple proportional-type gain also causes fewer extreme values; those located in the tails of the probability distribution, and the lower than expected variance at longer averaging periods. To study this further, we employed another random number test on the data.

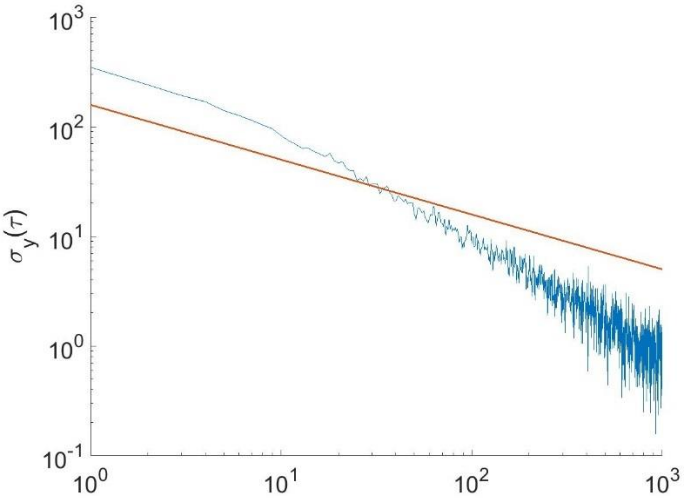


Figure 6.29. Allan deviation of LLQRNG output with proportional-only feedback applied.

A 4.99-Gb sample of LLQRNG data, resulting from a proportional-type filter was evaluated with the frequency 'monobit' test of the NIST STS suite [79]. This test calculates the probability that the mean of a measured bit-sample would belong to the output distribution of a correctly balanced RNG. The total 4.99-Gb sample was broken into 10-Mb blocks, and a p-value is assigned after the evaluation of each block. A high value (near 1) denotes a sample which has a bit bias near 0.5, and a low one (near 0), for a sample with a bit-probability far from 0.5. Given 499 blocks and a confidence interval of α = 0.01, the test expects at least 495/499 to pass, a requirement our results met with 499/499. What is also required, however, is that the p-values be uniformly distributed. A random number generator should, at times, appear to be unrandom. A histogram of our measured p-values, separated into 10 bins, is shown in Figure 6.30.

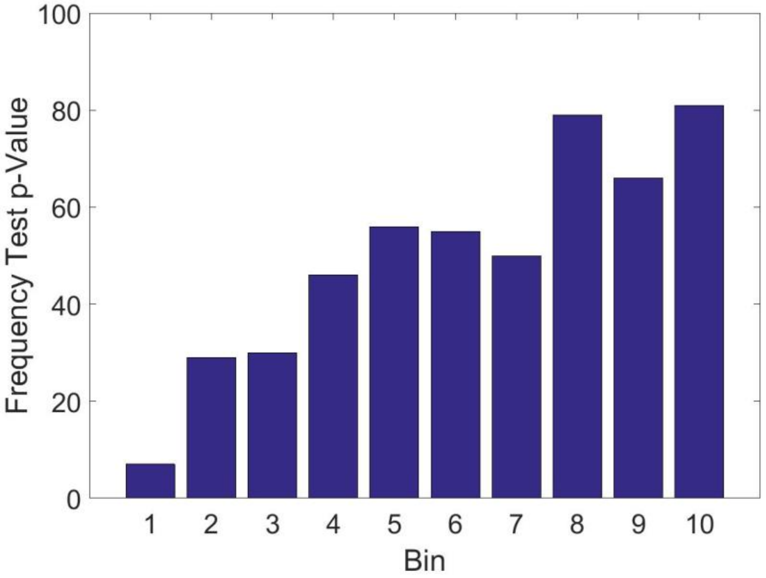


Figure 6.30. Histogram of p-values for the NIST STS frequency test. Bin 1 corresponds to p-values in the range [0, 0.1), Bin 2 [0.1, 0.2), etc. The test is designed so that a normally distributed random variable will display a uniform distribution of p-values, i.e., they should all occur with approximately the same frequency.

The absence of lower p-values in the frequency test denotes the lack of the more 'non-random' frequency values that should occur. Although the sampling of the LLQRNGs random process results in a normal distribution which is *centered* about the mean of 50%, there are extreme values near the tail-ends of the curve which should, albeit rarely, appear. This imbalance between high- and low-values comes from the unconditional action of the proportional feedback. Even if a bit-sample results in a fractional frequency which is very close to 50%, this scheme will, although weakly, apply negative feedback to tune the random process in the opposite direction. This overcorrection results in the more extreme values being less likely to occur, and thus the failure of the random number tests in the fashion above.

6.8.3 Shaped-gain Proportional Feedback

The approach, which has yielded the best results thus far, is more intuitive in nature. If averaging-type schemes are excluded due to persistent correlations, then the error term must be calculated from a single-sample. The linear-type proportional feedback over-constrains the distribution, resulting in a lack of extreme values, and a lower-than-expected variance at longer averaging periods.

We have thus far assumed that each individual LLQRNG-bit is a Bernoulli random variable, an experiment with a single-bit Boolean-valued outcome, whose value is determined by a time-varying probability of success p. The repeated measurement of many bits, in our case 100,000 per second, results in values taken from a binomial probability distribution function, with N = 100,000. Because the most probable result of any 100,000-bit sample is the mean, N*p, we take each bit-probability measurement to be a direct indication of the current value of p. However, the probability of measuring a value other than the mean does not decrease linearly (as the linear proportional-type feedback would suggest), but rather decreases relative to the shape of the binomial probability distribution. Therefore, if the amount of corrective feedback applied is shaped in a fashion that reflects the probabilities of the expected binomial distribution, perhaps the statistics of the resulting bit-stream will deviate less from the expected.

Mathematically, we define the strength of this type of applied feedback as

$$Feedback = f_{max}(1 - bino(N,p)), \qquad \text{Eqn. (6.9)}$$

where *bino(*N,p) is the normalized probability distribution of a binomial process with *N* trials and a probability of success p, and f_{max} is a tunable gain-like term. The binomial probability distribution is normalized to have a maximum value of 1, so the choice of f_{max} sets the maximum amount of feedback that is applied at any one step.

Our DAVAR analysis predicted a random walk with an average variance of 0.0049 bits, and because the variance estimates how far a set of random numbers is spread out from its mean, we chose 0.0049 as the initial value of f_{max}. The proposed feedback for N = 100,000, p = 0.5, and $f_{max} = 0.0049$ bits is shown in Figure 6.31.

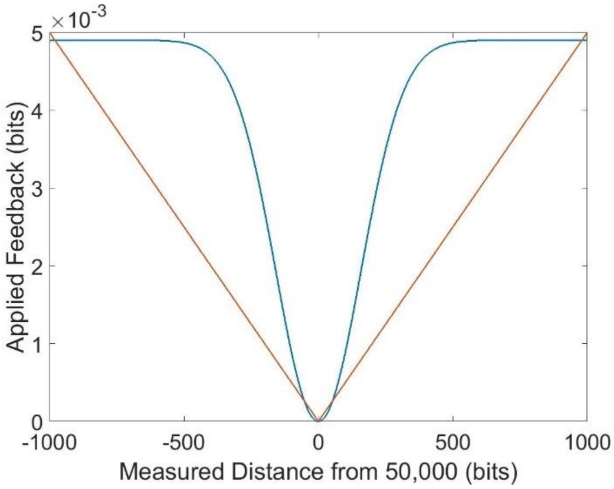


Figure 6.31. Proposed binomially-shaped (blue) and previously used linear-type (red) feedback scheme for a 100,000-bit LLQRNG sample.

Two problems arose when trying to implement this feedback-scheme. First, the desired bit-probability adjustments (0.005 c/s) are well outside the capabilities of our current DAC resolution ≈ 0.13 c/s. Secondly, the calculation of the shaped-probability distribution is complex, and cannot be done in real-time with our current microprocessor. However, to test the expected performance of such a feedback scheme it was simulated in MATLAB.

6.8.4 Simulated-Shaped Feedback

The shaped-gain feedback scheme was applied in a retroactive fashion to the same dataset used in the DAVAR analysis of Section 6.7.2, a collection of approximately 7 days in length, taken under typical environmental conditions, and with no correctional feedback. Before feedback, bias-drift is clearly evident in Figure 6.32, where samples have been averaged in 100-sample blocks.

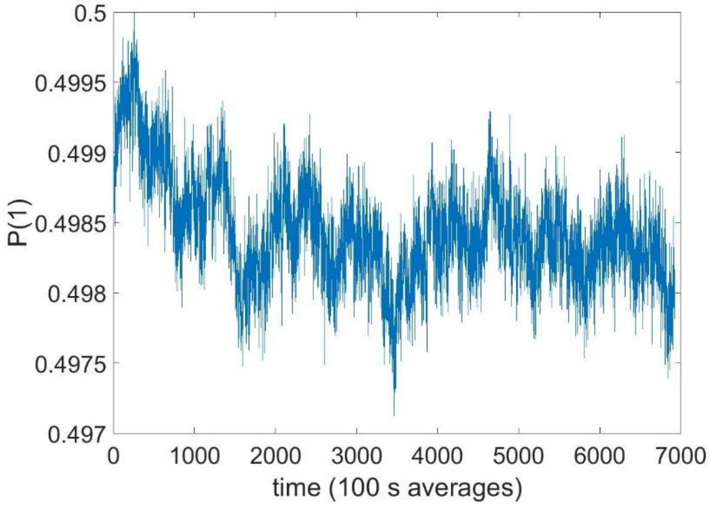


Figure 6.32. Uncorrected LLQRNG output, taken over approximately eight days.

At each point in the uncorrected dataset, the feedback that would have been applied via Equation 6.8 is calculated, with f_{max} equal to the average variance of the observed random walk (0.0049 bits), and the following data point was adjusted by that amount (here we assumed that adjusting the laser bias according to Eq. 3.8 would directly change the measured detection probability).

This type of feedback performed very well, with the simulated bit-probability and amount of applied feedback shown in Figure 6.33. Here a period of 10,000 seconds has been removed from the beginning of the simulation, a time during which the bit-probability was allowed to settle to 50%.

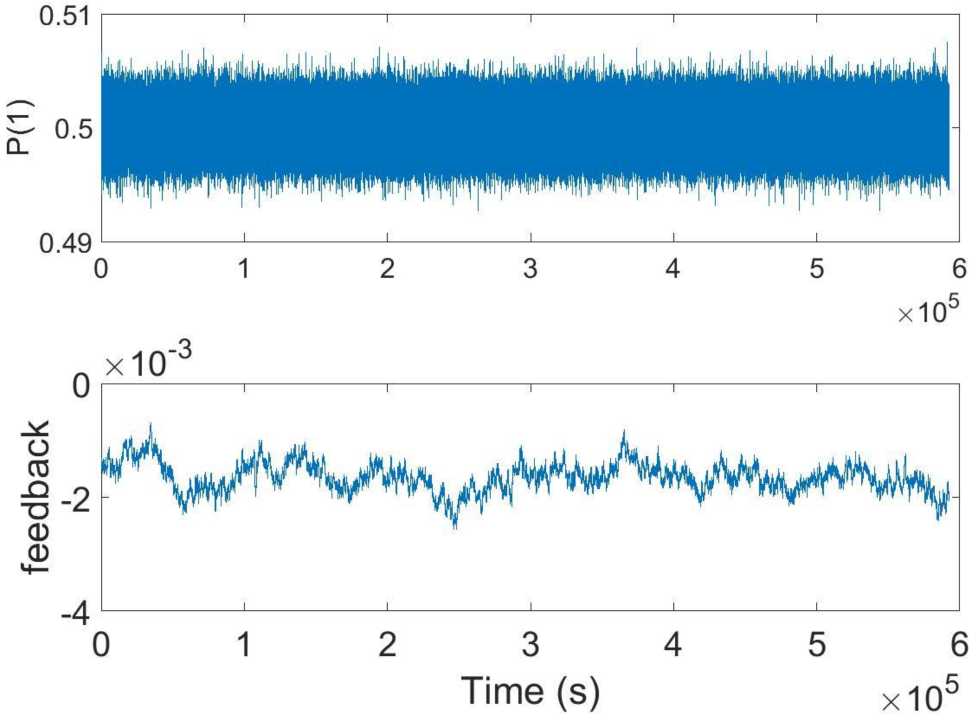


Figure 6.33. Simulated bit-probability (top) and amount of correctional feedback applied (bottom) for the shaped-gain approach.

The ADEV was also improved (Figure 6.34), with the lower-than-expected variance problem significantly alleviated. The correction caused a higher-than-expected deviation of ≈ 2-bits at some time-scales, but settled to a value which seems to be approaching zero. Statistically relevant calculations at longer averaging periods would require many months of data, and is planned for future exploration.

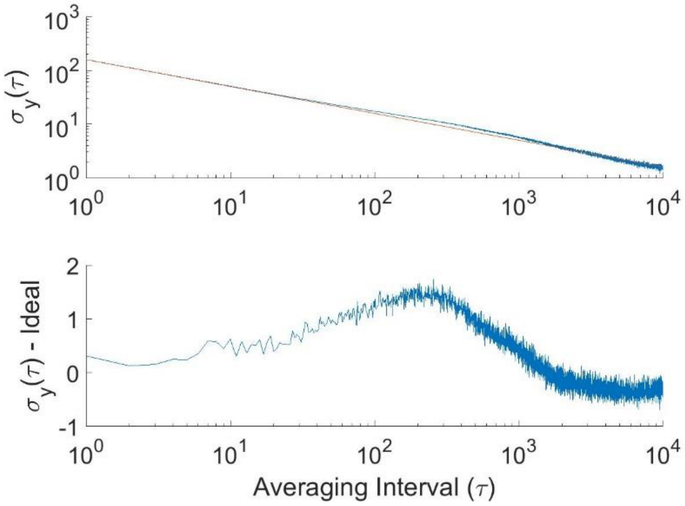


Figure 6.34. Allan deviation of simulated LLQRNG data with shaped-feedback applied (top) and the difference from what is expected from white-noise (bottom).

In terms of ADEV analysis, the choice of f_{max} was particularly sensitive. The average random-walk variance from the DAVAR analysis was 0.0049 bits, the value which was used in the data depicted in Figure 6.34. Shown in Figure 6.35, however, is the ADEV analysis for f_{max} equal to 0.001 (green) and 0.01 (blue) bits.

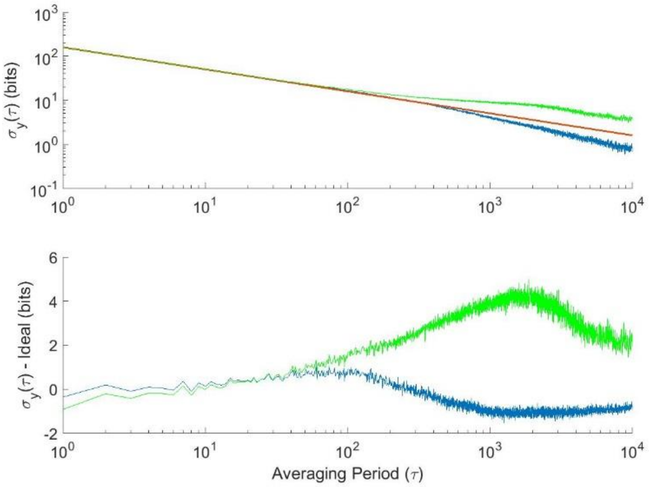


Figure 6.35. ADEV results for simulated shaped-gain application, with estimated random-walk strengths of 0.001 (green) and 0.01 (blue).

By underestimating the strength of the random walk (green line), the signal is not corrected enough, resulting in more extreme values and an under-correction at longer averaging periods. By overestimating the random walk, the signal is over-corrected, less extreme values occur, and a lower than expected variance is observed at longer averaging periods. If the magnitude of observed drift varies significantly over time, this single estimation of f_{max} may not be sufficient. Therefore, a more robust implementation would make repeated estimations in real-time, rather than a single one based on an observed average.

Autocorrelation analysis revealed correlations similar to the linear-type proportional feedback shown in Figure 6.28, but the both their time-scales and magnitude were reduced, as shown in Figure 6.36.

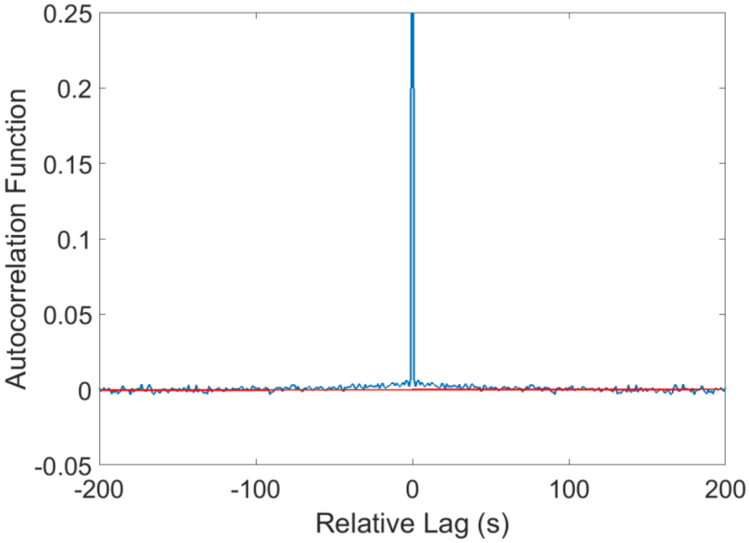


Figure 6.36. Autocorrelation function of simulated bit-stream after shaped-gain feedback is applied. Although slight correlations are observed on the ~25-s time-scale, they are much less than observed with either the linear or integral-type feedbacks.

6.8.5 Coarsely-Approximated Shaped Feedback

The continuous distribution used above is too computationally complex to be used with our microcontroller, so a binned-approach was implemented as a coarse approximation. The distribution of Figure 6.31 was integrated, starting from the center, until the total feedback contribution was equal to the bit-resolution of our DAC (~0.13 bits). This occurred after a bit-probability difference of approximately one standard deviation, or 158 bits (P(1) = 0.50158) from the mean. As an initial attempt, the amount of feedback applied (in bits) was set to be directly proportional to the number of standard deviations away from the mean, or $feedback = floor(\mu/158)$, an illustration of which is shown in Figure 6.37.

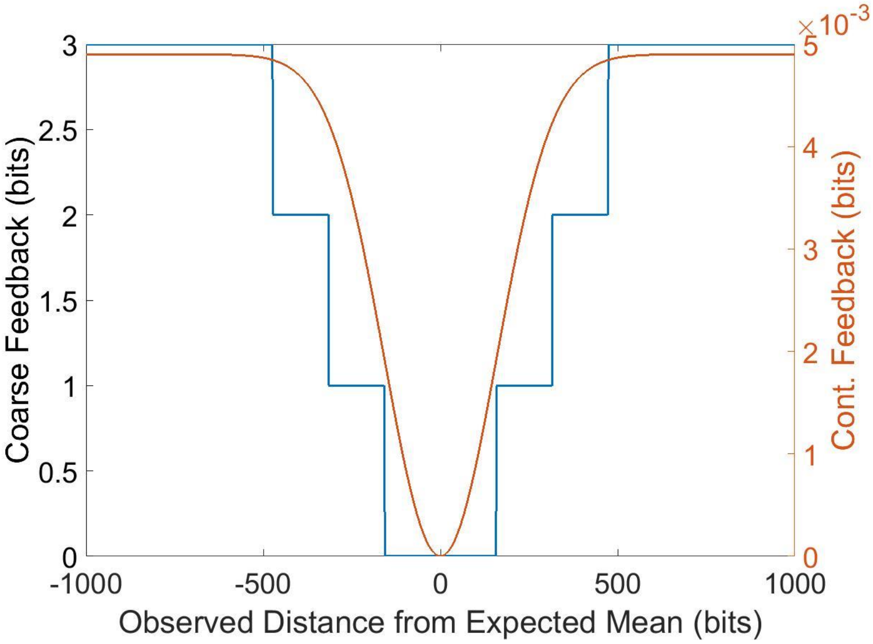


Figure 6.37. Coarse approximation (blue) to the continuous shaped-feedback approach (red).

To protect against overcorrection, a maximum bit-change per measurement was set at three bits. The microcontroller was programmed to apply this binned-type feedback, once a second, after every 100,000 recorded bits. The system was left to run for 13 days, under typical environmental conditions, and the random-bit output was analyzed as before.

Both the autocorrelation (Figure 6.38) and ADEV (Figure 6.39) were taken and showed similar results when compared to the simulation, although the ADEV analysis seems to indicate a feedback scheme which performs slightly better at longer τ. It is unclear precisely why this is the case. It is possible that the strength of the random walk noise had changed in between the taking of the no-feedback data set (in which it was estimated to be $q_3 = 0.0049$), and the data set in which feedback was applied according to the initial estimation.

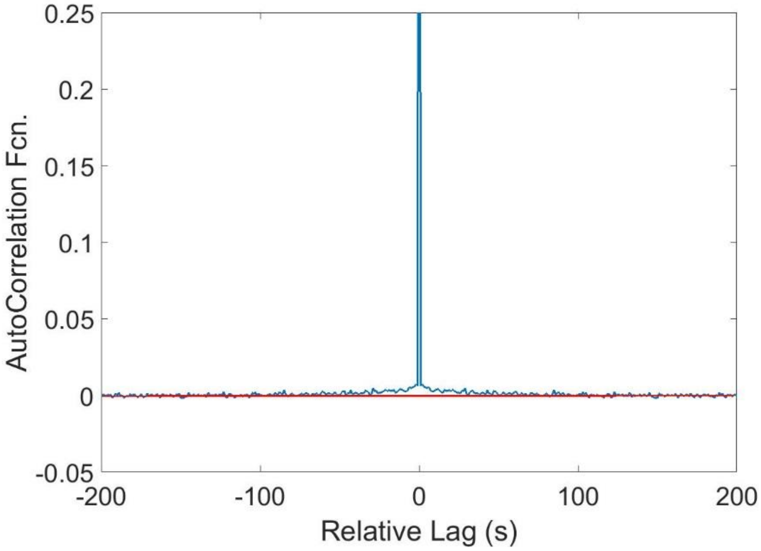


Figure 6.38. Autocorrelation results for dataset with feedback applied according to the probability distribution of Figure 6.37, a binned approximation to the continuous distribution of Figure 6.31.

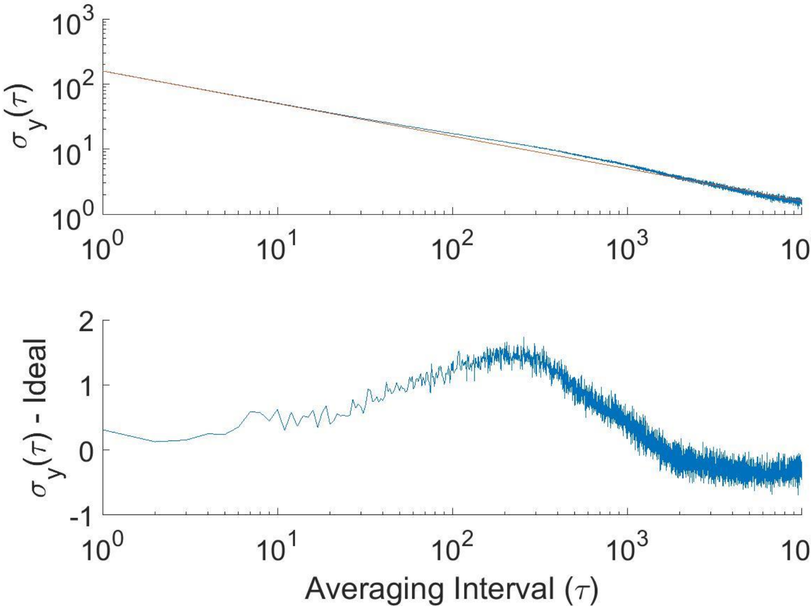
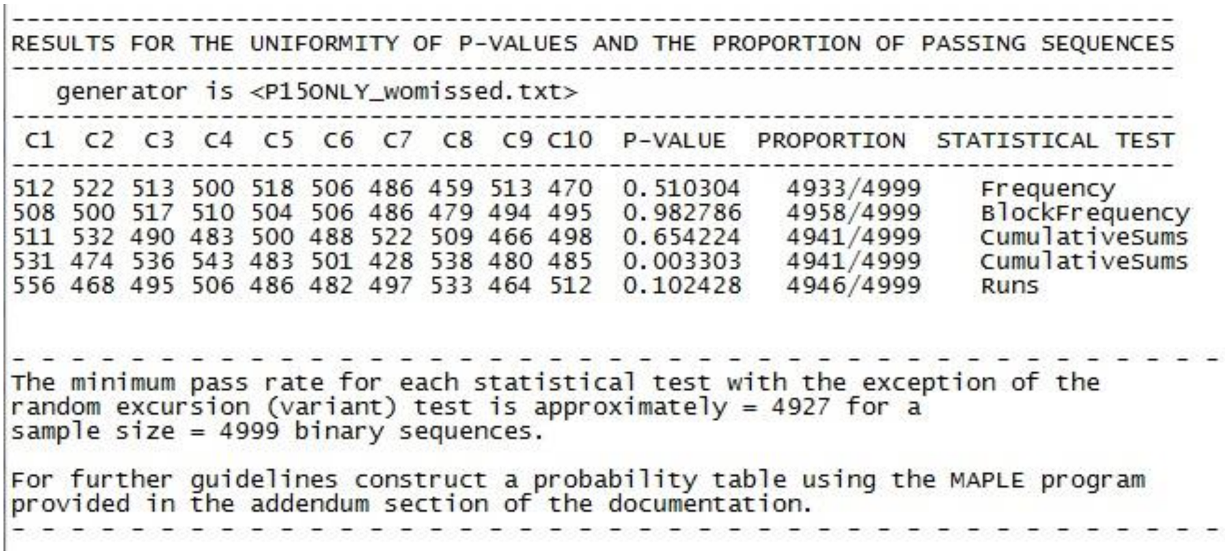


Figure 6.39. ADEV results for feedback applied according to the probability distribution of Figure 6.37, a binned approximation to the continuous distribution of Figure 6.31.

6.8.6 Frequency-Test Results

The dataset resulting from the binned-feedback approach of Section 6.8.5 was broken up into 5-Gb portions, and analyzed using a subset of the NIST STS Random Number Generator Test Suite [79]. Each individual 5-Gb file was then analyzed in 100,000-bit blocks. Under these conditions, the tests would routinely pass, with an example output shown in Figure 6.40.



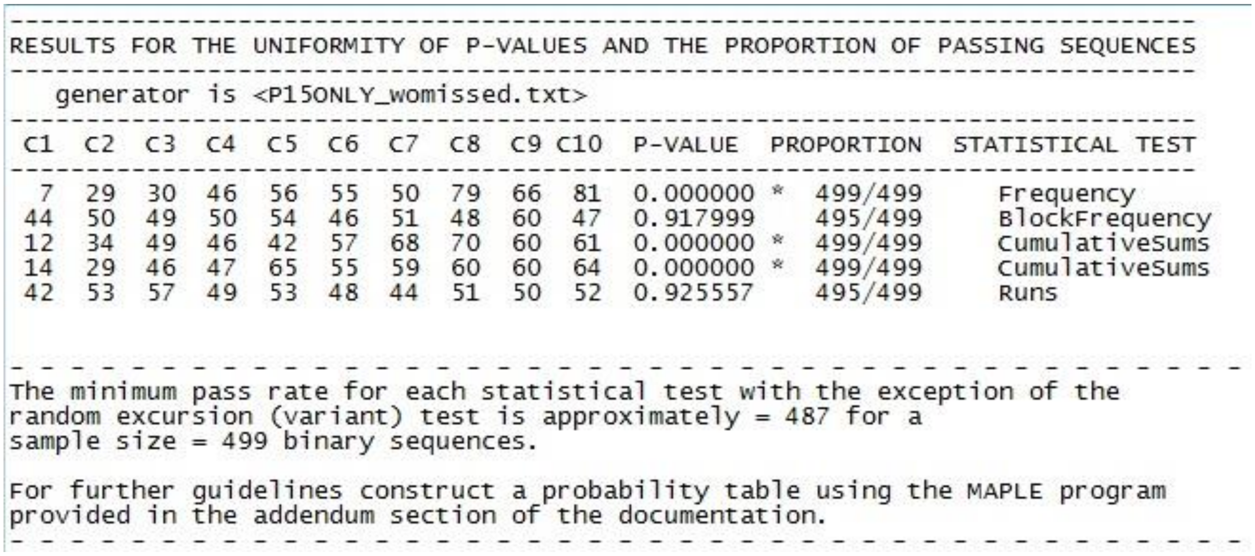
```
------------------------------------------------------------------------------
RESULTS FOR THE UNIFORMITY OF P-VALUES AND THE PROPORTION OF PASSING SEQUENCES
------------------------------------------------------------------------------
   generator is <P15ONLY_womissed.txt>
------------------------------------------------------------------------------
 C1  C2  C3  C4  C5  C6  C7  C8  C9 C10  P-VALUE  PROPORTION  STATISTICAL TEST
------------------------------------------------------------------------------
512 522 513 500 518 506 486 459 513 470  0.510304   4933/4999    Frequency
508 500 517 510 504 506 486 479 494 495  0.982786   4958/4999    BlockFrequency
511 532 490 483 500 488 522 509 466 498  0.654224   4941/4999    CumulativeSums
531 474 536 543 483 501 428 538 480 485  0.003303   4941/4999    CumulativeSums
556 468 495 506 486 482 497 533 464 512  0.102428   4946/4999    Runs


- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
The minimum pass rate for each statistical test with the exception of the
random excursion (variant) test is approximately = 4927 for a
sample size = 4999 binary sequences.

For further guidelines construct a probability table using the MAPLE program
provided in the addendum section of the documentation.
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
```

Figure 6.40. Example results of random-number tests when analyzed in blocks of 100,000 bits.

For block sizes of 1,000,000 to 50,000,000 Mb, they would routinely fail, as shown by the asterisks in Figure 6.41, denoting the non-uniformity of p-values for three particular tests.



```
------------------------------------------------------------------------------
RESULTS FOR THE UNIFORMITY OF P-VALUES AND THE PROPORTION OF PASSING SEQUENCES
------------------------------------------------------------------------------
   generator is <P15ONLY_womissed.txt>
------------------------------------------------------------------------------
 C1  C2  C3  C4  C5  C6  C7  C8  C9 C10  P-VALUE  PROPORTION  STATISTICAL TEST
------------------------------------------------------------------------------
  7  29  30  46  56  55  50  79  66  81  0.000000 *  499/499     Frequency
 44  50  49  50  54  46  51  48  60  47  0.917999    495/499     BlockFrequency
 12  34  49  46  42  57  68  70  60  61  0.000000 *  499/499     CumulativeSums
 14  29  46  47  65  55  59  60  60  64  0.000000 *  499/499     CumulativeSums
 42  53  57  49  53  48  44  51  50  52  0.925557    495/499     Runs


- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
The minimum pass rate for each statistical test with the exception of the
random excursion (variant) test is approximately = 487 for a
sample size = 499 binary sequences.

For further guidelines construct a probability table using the MAPLE program
provided in the addendum section of the documentation.
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
```

Figure 6.41. Example results of random-number tests when analyzed in blocks of 1,000,000 bits.

For each of the performed tests, the result is referenced by comparing different aspects of the bit-probability from what is expected in a normal distribution. As shown in Figure 6.38 bottom, the difference between the measurement and a white-noise distribution increases as larger bit-samples are considered.

At τ = 1, — blocks of 100,000 bits — the estimated difference is close to zero, and the suite's analysis succeeds when considering the data on these timescales. When considering block sizes where the observed difference in variance was higher, the tests would routinely fail. If the difference between observed and measured variances is indicative of the test suite's performance, then blocks taken over τ = 1000 would also pass. This could open up a possible avenue towards 'fooling' the random test suite; however, the amount of raw bit-data that this requires for accurate statistics (>= 500 Gb), has not yet been taken.

6.9 Results and Future Improvements

Currently, the output bit-latency of the LLQRNG is 2.4 ± 0.2 ns, well within the requirements set by our Bell test. The 0.13 c/s bit-resolution of the DAC does not allow for the fine adjustments needed for the shaped-gain approach, but this can be improved by adding DACs with finer resolution. While the development of an optimal active-feedback scheme is still being investigated, the results presented thus far indicate that the frequency analysis revealed by the dynamic Allan variance is of use in characterizing the type and strength of the drift. This type of frequency characterization is typically not done on random number generators, and many physical RNGs use some form of active stabilization. Modern QRNGs can generate bits in the ~Gb/s range, and their output characterization is often only performed over a few Gb, or tens of seconds, sample. Therefore, the DAVAR analysis presented here could reveal new information about their performance, especially over longer time scales.

It should be stressed that the exploration into feedback characterization schemes is by no means complete. One proposed performance goal is to keep the bit-probabilities tuned to the 10^{-6} level, or for our 100 kHz repetition rate – the mean of our measured binomial distribution is kept to (50,000 $\pm$ 1) c/s. With a single measurement of the bit-probability, averaging down enough to have this level of confidence would take several days, but we are currently looking into adding multiple sensors (i.e., temperature, classical photo diodes which monitor a portion of the VCSEL's output, etc.). With several independent measurements taken simultaneously, each with a specific relationship to the bit-probability, attaining this threshold can be achieved at faster time scales. Also used in GPS location tracking, a Kalman filter would be appropriate for such an endeavor, and exploration into this has already begun.

Also of possible interest is the combination of two LLQRNG systems, in parallel. This would be implemented by an XOR of their outputs, for which the measured latency of a typical logic gate (NB7L86M) would add another ≈ 200 ps. This would presumably improve the bit-probability results to that of other

QRNG systems, but as our research aimed to optimize a single system, we have not yet implemented such a scheme. The two systems would also have to be actively-controlled with feedback, as we can make no assumptions on how far the additional random walk will ‘travel’ over exceptionally long time scales. However, as this QRNG is set to be used in NISTs random number beacon, a world-wide random number generation service, this upgrade will be performed as more systems are made.

Chapter 7 — Conclusions & Future Work

In this dissertation we considered various ultra-high-speed electronics critical for burgeoning applications in the new field of quantum information processing. We presented the theory and implementation of two quantum random number generators. The constant-current QRNG of Section 3.7 sampled the waiting-time distribution of photon arrival times and generated random bits at 130 Mb/s, which at the time was the world's fastest quantum random number generator [78]. In an effort to reduce, or even eliminate, the amount of necessary whitening, we designed the shaped-pulse QRNG of Section 3.8, which tailored the current in such a way that the probability of photon detection over a predefined interval was roughly uniform for each time-bin. The finite-bandwidth of the operational amplifiers ultimately limited the performance, allowing only a portion of the waiting-distribution to be used; nevertheless, a random-bit generation rate of 110 Mb/s was achieved [13]. The performance of both systems was limited by the maximum detection speed of the SPADs (~11 MHz), and the minimum resolvable time-bins of 5 ns.

In Chapter 4 a new afterpulse reduction technique was introduced [85], which for the first time directly characterized the effects of prompt quenching on several commercial SPADs. Through a measurement of the SPAD's series resistance and parasitic capacitances, the expected behavior of the avalanche current flow was able to be modeled. By then collecting a series of time-tags at different quenching latencies and hold-off periods, we could estimate both the characteristic trap lifetimes and their proportionality constants. Through a simple differential rate-equation, we modeled the effect of reducing the charge flow after each avalanche, and experimentally verified our prediction. Although the experimental setup was not suitable for standalone SPAD operation, it was shown that the afterpulsing probabilities could be significantly reduced, in one case by up to a factor of 12. When integrated into a fully-function SPAD circuit, this would allow a marginally reduced dead time (~20%), but further circuitry improvements could yield better results.

In part to address the time-bin resolution limitation discussed above, we developed our own time-tagging system, built specifically for use with SPADs. With the aid of a fast demultiplexer and FPGA, our system is able to consistently record timing information at detection rates up to 100 Mc/s, with a time-bin resolution of 100 ps. The performance characteristics of the time-tagger also allowed for characterization of SPAD behavior at previously unexplored detection rates. By examining the third-order correlations of a sequence of time tags, a complete SPAD characterization can be performed; such an analysis even revealed some previously undiscovered subtleties which we currently attribute to the device's electronics. This new technique is currently being prepared for publication.

In Chapter 6 the design and characterization of a low-latency QRNG was discussed, which was used in one of the first experimental 'loophole-free' demonstrations of a Bell test [18]. A fundamental test in physics and a cornerstone of quantum information, performing a Bell test without any additional assumptions, has been an objective for over 50 years. In order to close the locality loophole, a random basis setting had to be provided in as short a time as possible, a requirement the LLQRNG was able to fulfill with a latency of (2.4 ± 0.1) ns. The LLQRNG's performance was adversely affected by the presence of a long-term bit-probability drift; we explored the drift's strength and origin using atomic clock frequency stability techniques. By adding the capability to fine-tune the bit-probability with a precision digital-to-analog converter, the drift's influence can be minimized through corrective feedback. Initial feedback schemes have been moderately successful, with some output streams passing cryptographic random number tests; however, further work into finding an optimal feedback scheme is underway. The LLQRNG is currently planned to be used as part of NIST's random-number beacon, a world-wide service providing a safe, and trusted, source of random bits.

As experiments move from relatively simple proof-of-principle investigations toward real-world applications, there will be an increasing need for precise, high-speed electronics. In this thesis, we have focused on pushing the forefront of three of these – detectors, time-taggers, and quantum random number generators – which will play an important role in the future of photonic quantum information.

Appendix A — Detection Speed vs. Entropy

While increasing the time-bin resolution (i.e., shorter resolvable bins) results in a logarithmic increase in the rate of entropy generation, increasing the source rate has a somewhat different result. Although a rate increase results in more detections per second, it also lowers the available entropy per detection, as the average time-bin value decreases. As shown in Figure A.1, increasing the detection rate causes the waiting-time distribution to shrink and become more predictable. In the limiting case, all detections fall into the first time-bin and no entropy is generated.

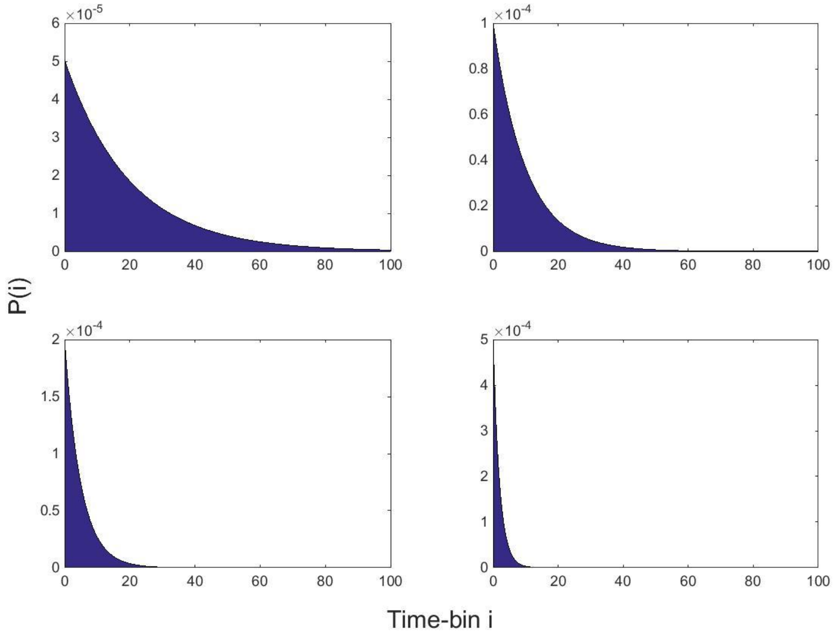


Figure A.1. Theoretical Poissonian waiting-time distributions for the CCQRNG. Average detection rates of 5 MHz, 10 MHz, 20 MHz, and 50 MHz result in probability distributions of decreasing entropy, i.e., less uniform.

To calculate the maximum amount of entropy generated per second, we first calculate the available entropy per detection. For an average incoming photon rate λ and time-bin size Δt, the probability of a photon falling into time-bin *I* is given by the equation $P_i = \lambda \Delta t e^{\lambda \Delta t i}$. The total entropy per detection is then calculated by summing over all P_i according to the Shannon entropy equation

$$S = -\sum_{i=0}^{N} P_i log_2(P_i). \quad \text{Eqn. (A.1)}$$

Given a detector with dead time τ_d, the click-rate CR, or rate at which photons are registered, is given by the equation

$$CR = \frac{1}{(\tau_d + \frac{1}{\lambda})}. \qquad \text{Eqn. (A.2)}$$

Therefore, given an entropy value and average rate of detection, we can multiply the two together to determine the available entropy per second. Assuming a fixed time-bin resolution of 27 ps from the ACAM TDC-GPX [76], we then calculate the entropy per second for varying detection speeds, as shown in Figure A.2. The peak detection speed of 19 MHz is above our detectors capability, but would produce a random number generation rate of approximately 182 Mb/s.

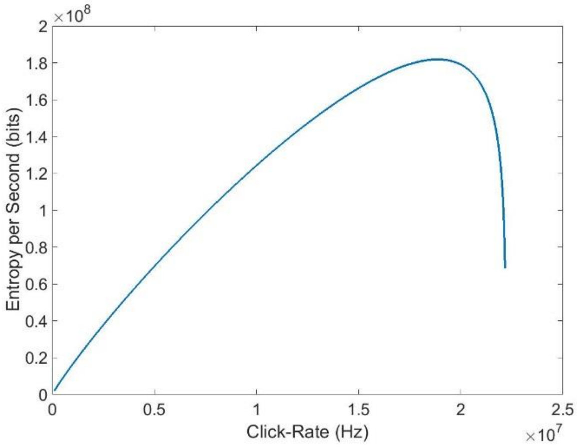


Figure A.2. Entropy per second vs. click-rate for a detector with 45 ns dead time and 27 ps time-bin resolution. As the detection speed reaches the inverse of the dead-time, the available entropy per second quickly decreases, as the detector is firing almost immediately after recovering.

Appendix B — Shaped-Pulse Circuit

The $1/(T-t)$ pulse shape was approximated with a circuit containing three major components: a saw tooth generator, logarithmic converter, and differentiator. The sawtooth generator provides the $(T-t)$ shape, the logarithmic converter transforms this into $\ln(T-t)$, and the differentiator creates the final $\sim 1/(T-t)$ waveform, as indicated in Figure B.1.

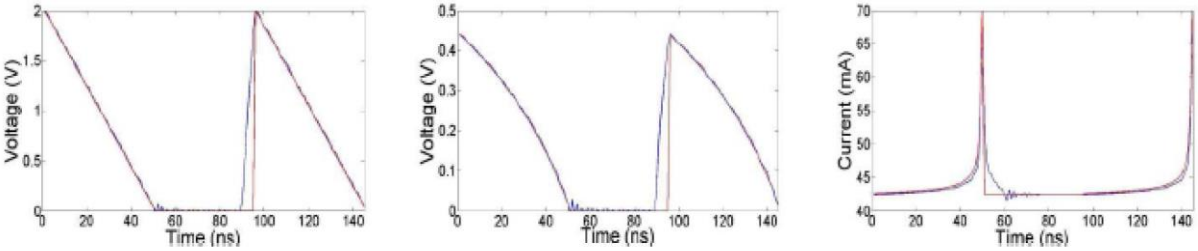


Figure B.1. Ideal (red) and simulated SPICE (blue) waveforms of the sawtooth, logarithmic converter, and differentiator stages of the pulse-shaping circuit, assuming a 50 ns reset period. The delay between pulses is to accommodate the 45 ns dead time of the SPAD, and the offset on the final waveform is to keep the laser diode operating above threshold.

The majority of the circuitry was implemented using OPA847 wideband high-speed operational amplifiers and 1N4148W small-signal diodes. The sawtooth generator was constructed from a triangle-wave generator circuit, with capacitors used to control the speed of the rising and falling edge. The logarithmic converter used a 1N4148W diode as the feedback element around an OPA847 operational amplifier, and biased in the logarithmic region of the diode. The differentiator was a simple RC op-amp implementation, but also the most problematic, as abrupt changes in the waveform resulted in spikes in the derivative. The current from the circuit was used to drive the light source, a 650 nm laser diode operating well above threshold. The laser diode output was attenuated to the single-photon level with standard neutral density filters, and the signal was detected by a fast id100 avalanche photodiode [75]. The detector output was then processed by a high-speed Virtex 6 ML605 FPGA.

Appendix C — WTDC Clock Sources

Two options were explored for providing the 10 GHz clock necessary for operating the WTDC time-tagging system of Chapter 5: a commercial RF oscillator and a custom-built digital clock multiplication circuit. The clock's frequency directly determines the time-bin resolution τ_d, and if more than one resolution is desired then the availability of multiple clock sources contributes to the overall system's feasibility as a standalone solution. In this section the design and frequency characteristics of a configurable digital clocking source (DCS) are presented.

C.1 Digital Source Design Flow and Overview

Both a simplified flow diagram and the physical circuit are shown in Figure C.1. The DCS takes as input a differential clock signal from an onboard 625 MHz MEMS-based temperature-controlled oscillator for 10 GHz operation with the Wayne-Tagger. If an external source is desired, it can be provided on the CLKIN SMA connectors. The input clock is then multiplied by the two GaAs active frequency multiplier ICs, labeled 'x16'. Because the input clock is differential, the two 10 GHz output clocks will automatically be 180 degrees out of phase, useful for the improved timing resolution method explained in Section 5.5.9, in which they are used to clock two different time tagging channels. If the onboard oscillator is used, a differential-input to single-output buffer provides a reference of the same frequency, for possible use as the FPGA internal clock. The circuit operates on a single 5.5V supply, draws ~ 150 mA, and each individual chip's power is provided by the 3.3V and 5.5V regulators.

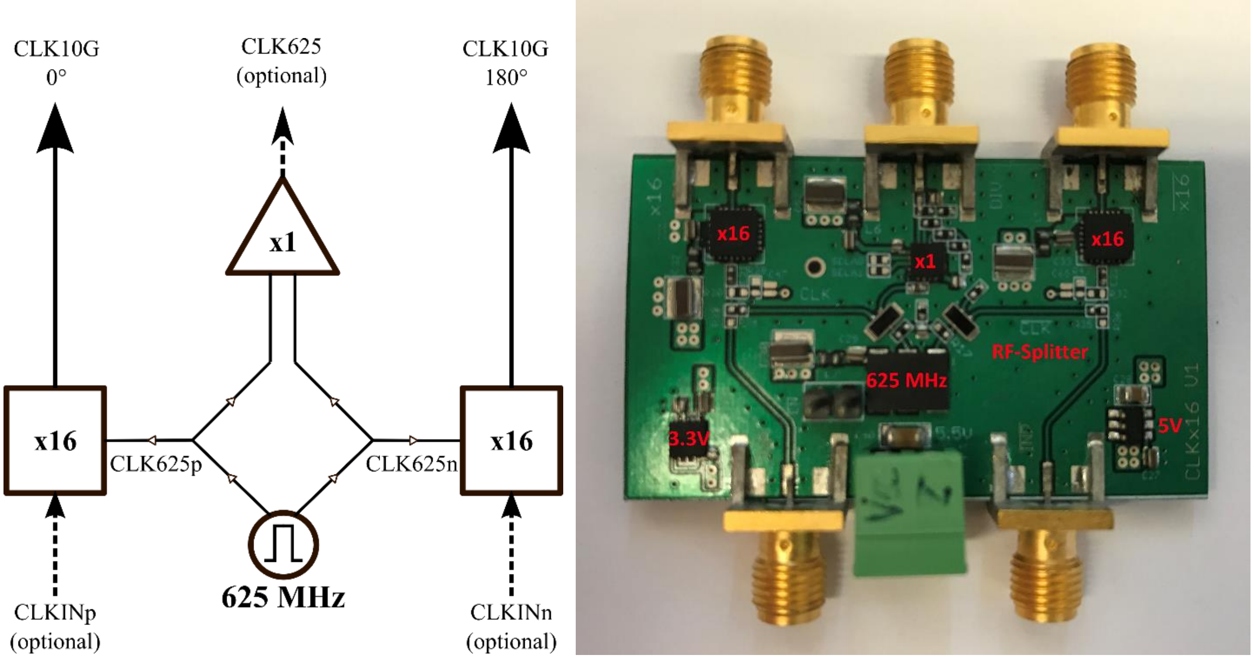


Figure C.1. Conceptual design flow (left) and physical circuit (right) for the DCS board. Each differential half of either an the onboard oscillator or an external source is multiplied by 16, resulting in two higher frequency clocks with 180 degrees of phase difference. The x1 buffer is optional for driving the FPGAs internal logic.

C.2 Measured Performance

The 10 GHz output clocks of the DCS were unfortunately very noisy. Although the center frequency was measured to be 9.999996 ± 0.000005 GHz, there were substantial other frequency components present. A spectrum analyzer was used to identify any extraneous frequencies, and a small span (2 MHz) is shown in Figure C.2. Additional harmonics were present, and because the spectrum analyzer could not resolve the full range in a single sample, a figure from the clock multiplier's datasheet is also provided. The resulting output clock signal was totally unsuitable for use with the time-tagging system, as the FPGA's clocking resources were unable to lock onto the resulting 625 MHz clock.

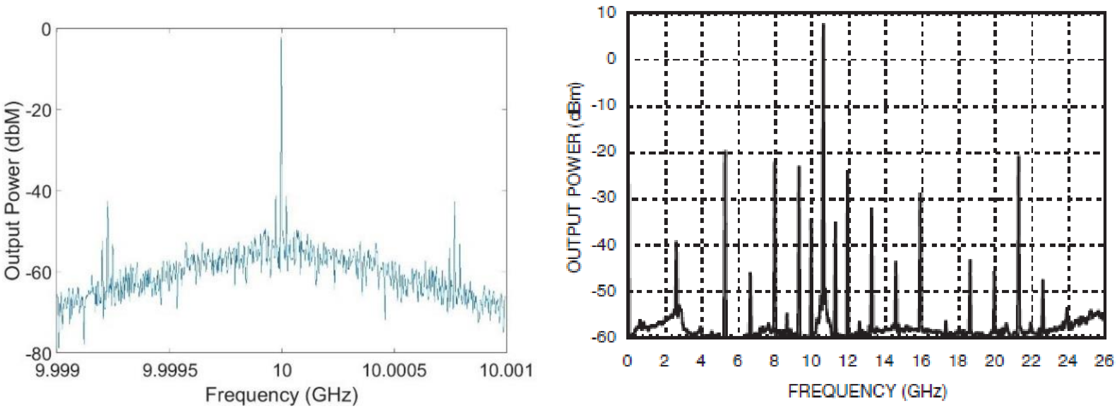


Figure C.2. Measured 2 MHz span of the DCS 10 GHz output. Clear harmonics are present, with additional components (not shown) similar to those specified in the x 16 multipliers datasheet *[123]*.

To try to reduce these additional harmonics, filtering the output was attempted. Commercial RF-frequency bandpass filters are very expensive, so a custom design was attempted. Popular among amateur radio operators, a pipe-cap filter such as the one shown in Figure C.3 has been an inexpensive filtering option for several decades [124], [125]. The screw and plumbing pipe-cap act as the center and outer conductors of a coaxial quarter-wave resonator. The depth at which the screw penetrates the cap sets the resonance frequency, and the two stripped SMA-cables act as probes for input and output, primarily capacitively coupling to the tuning screw. The pass-band frequencies were extremely sensitive both to the depth of the tuning screw as well as the depths of the SMA probes. Differences of 1/16” could cause 10 dB of attenuation, and since they were individually soldered each time, perfecting the design was difficult.

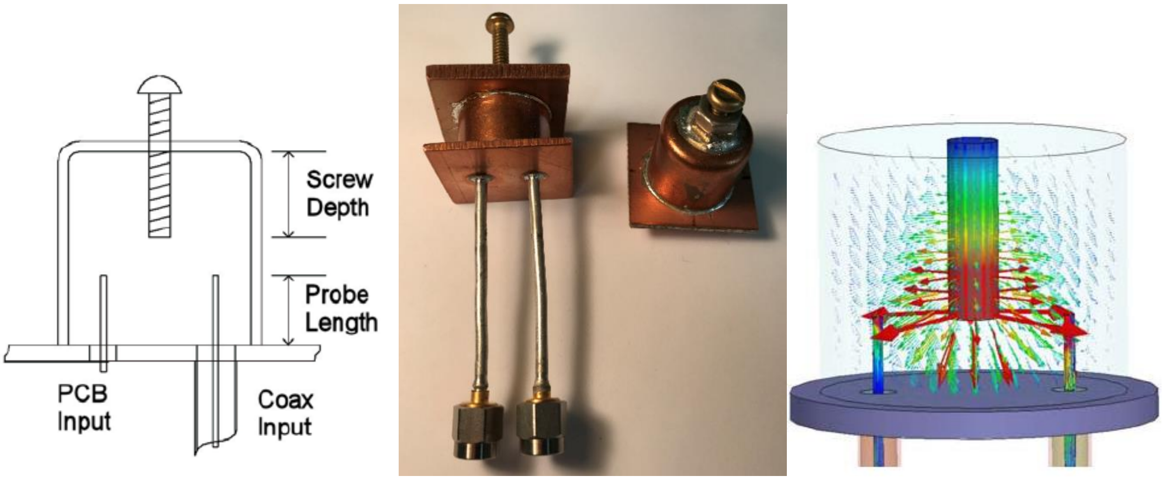


Figure C.3. Conceptual side-view (left), actual implementation (middle), and EM simulation (right) of 10 GHz bandpass pipe cap filters designed for filtering the high-speed WTDC clock. The tuning screw sets the quarter-wave resonance and is capacitively coupled to the input probes. Figures courtesy of *[124]*.

The transfer function of the filter was measured using a network analyzer, and the S21 parameter is shown in Figure C.4 (left) over a 10 GHz span. Although there exists a clear peak in the desired passband, it is relatively wide -- not reaching high levels of attenuation for many hundreds of MHz, and the minimum 5 dB of loss reduces the output signal strength significantly. The same spectral analysis as in Figure C.2 was taken again with the filtered circuit, and is shown in Figure C.4 (right). Although the two major side lobes were reduced (as were many of the frequency components not shown), there still exist additional harmonics particularly at higher frequencies. Additionally, the peak output power of roughly -20 dbM (~ 22 mV) is not enough to trigger the accompanying electronics, and as such this clocking approach was not used. Several filters could be used in series to reduce other frequencies, but even then amplifiers would have to be introduced between stages.

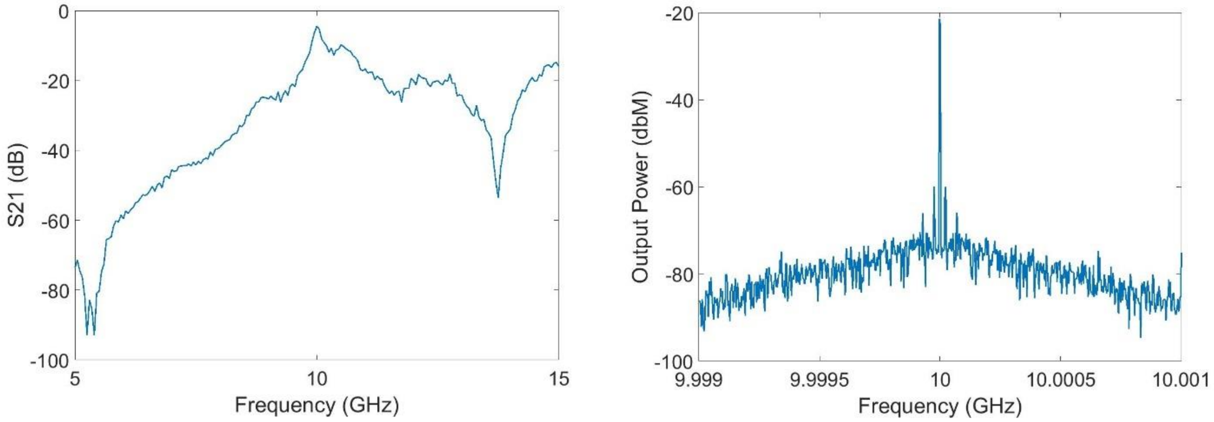


Figure C.4. Measured S21 of the pipe-cap filter (left) and spectral analysis (right) of the resulting 10 GHz filtered output. Although the two shown side lobes were eliminated, others exist at higher frequencies. Additionally, the peak power of -20 dbM makes the signal unusable without amplification.

Ultimately, for the final single-channel WTDC design a commercial 10 GHz oscillator (Jersey Microwave, FRDR0 – 10000 – 13) was used, with its measured spectral components shown in Figure C.5. This proved to be a more reliable option, and there were no significant additional harmonics at other frequencies as with the DCS, although the commercial oscillator requires an awkward 15 V supply, dissipates a large amount of heat, and the measured frequency was marginally offset (10.01690 ± 0.000001 GHz).

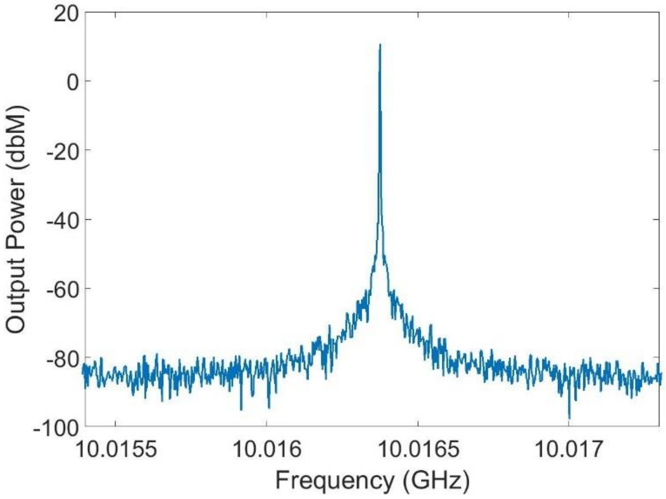


Figure C.5. Spectral analysis of the commercial 10 GHz oscillator used in the WTDC system.

Appendix D — WTDC Board Layouts

All of the printed circuit board layouts shown here were designed with the Eagle CAD design system. As there were no bottom or internal signals, the QSE-adapter of Figure D.1 has had its top layer flooded (solid red area), while Figures D.2 and D.3 have not in order to illustrate the additional signals.

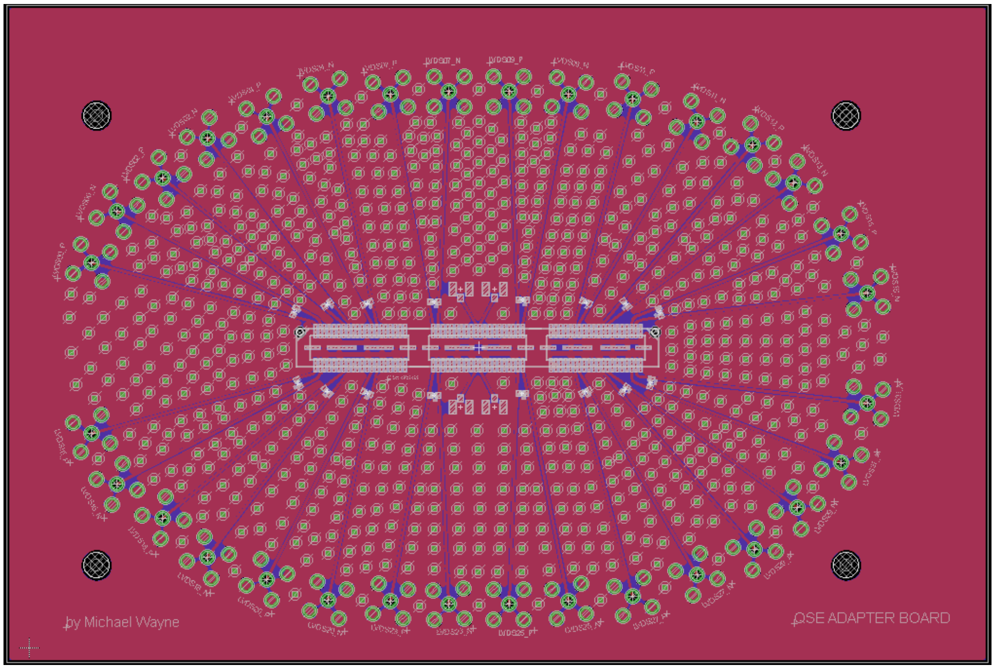


Figure D.1. PCB board design of the DMUX-to-QSE adapter for the Wayne-Tagger.

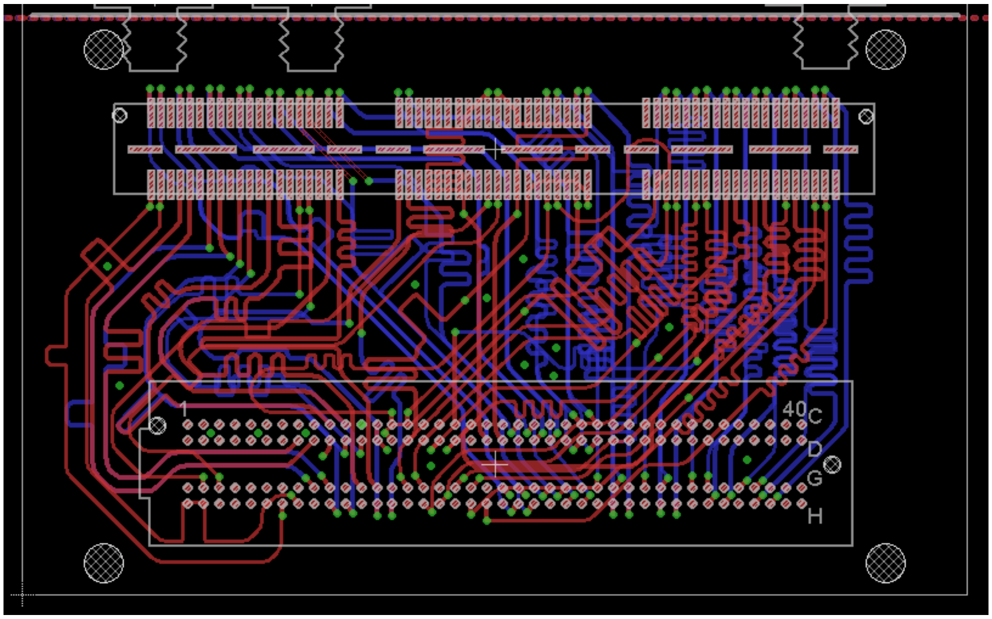


Figure D.2. PCB board design of the QSE-to-FMC adapter for the revised Wayne-Tagger. This allowed the WTDC to interface with one of our higher speed FPGAs. Floods are left un-filled to illustrate internal or bottom level signals.

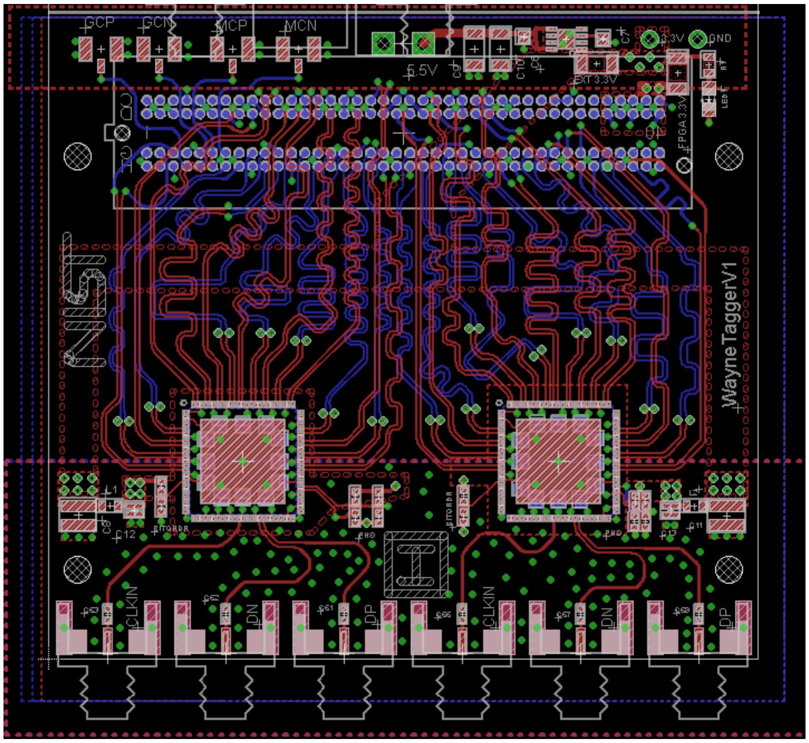


Figure D.3. PCB board design of the revised Wayne-Tagger system. Two 10 GHz DMUXs (large red squares) are able to interface with the FPGA through the length-matched traces. Floods are left un-filled to illustrate internal or bottom level signals.

Appendix E — Voltage-Summer / Pulse-Forming Circuit

The precision control of the VCSEL's bias voltage allows for ultra-fine tuning of the low-latency quantum random number generator's bit-probability. This is accomplished with the aid of an op-amp-based summing amplifier, the details of which are explained here. Currently, one bit of resolution in the summing amplifier corresponds to a 4.75 µV change in the VCSEL's forward-bias, which results in a bit-probability change of 1.3e-6.

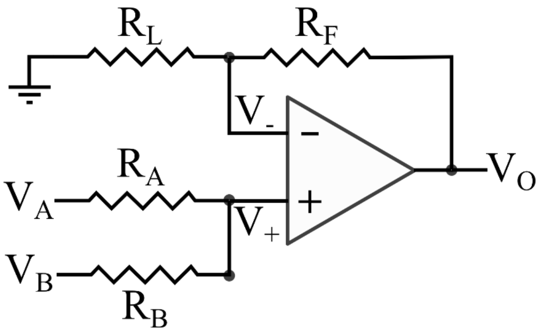


Figure E.1. Schematic of op-amp-based summing amplifier.

First consider the general summing amplifier structure, shown in Figure E.1. The application of Kirchoff's Current Law at V_+ and V_- results in equations (E.1) and (E.2), while equation (E.3) is one of the characteristic properties of an ideal op-amp:

$$\frac{V_A - V_+}{R_A} = \frac{-(V_B - V_+)}{R_B} \qquad \text{Eqn. (E.1)}$$

$$\frac{-V_-}{R_1} = \frac{V_- - V_o}{R_f} \qquad \text{Eqn. (E.2)}$$

$$V_+ = V_-. \qquad \text{Eqn. (E.3)}$$

Solving this series of equations for V_O leads to the general form of the output:

$$V_O = \left(1 + \frac{R_f}{R_L}\right)\frac{V_A R_B + V_B R_A}{R_B + R_A}. \qquad \text{Eqn. (E.4)}$$

Assuming that R_A and R_B are chosen to be equal,

$$V_O = \left(1 + \frac{R_f}{R_L}\right)\frac{V_A + V_B}{2}, \qquad \text{Eqn. (E.5)}$$

and if $R_f = R_L$,

$$V_O = V_A + V_B. \qquad \text{Eqn. (E.6)}$$

The result of equation (E.6) is a circuit that, with appropriately chosen resistors, will have an output equal to the sum of the two individual input voltages, $V_A$and V_B.

Borrowing from the 'coarse' and 'fine' portions of the Nutt Method [97] used in the time-tagger of Chapter 5, we can now use this circuit to precisely control our VCSEL's bias-voltage, an expanded schematic is shown in Figure E.2.

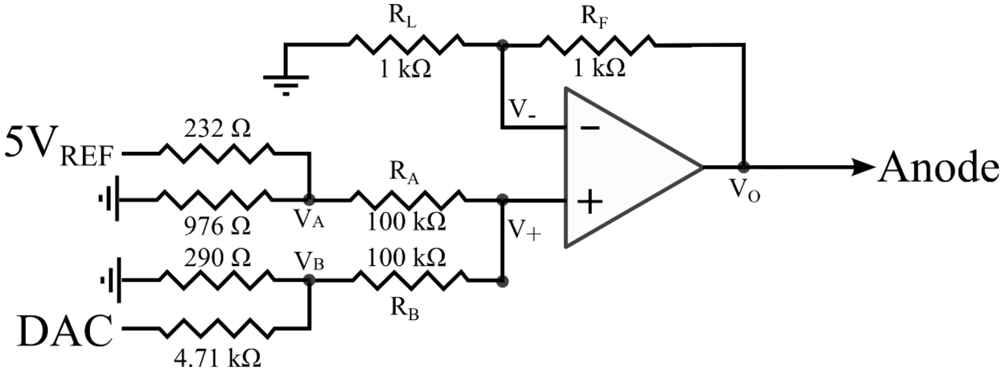


Figure E.2. Summing-amplifier as used in the LLQRNG bias-circuit. The 5V reference is divided down to 4.04 V to provide a stable DC offset, while the divided-down DAC output allows for tuning of the forward bias in 4.75 µV steps.

A constant DC offset of 4.04 V is provided by the divided-down precision 5V reference, which has a temperature coefficient of 5 ppm/°C. The cathode (not shown) is also kept at approximately this level. This ensures that when the circuit is first activated, that the total bias across the laser is approximately zero. The DAC has an output range of 0-5V and 16-bits of resolution, which would normally correspond to a voltage-per-bit of 76.2 µV. By inserting a voltage-divider at its output V$_B$, the output range is reduced to 0-290 mv; increasing the resolution to 4.43 µV / bit. The high values of R$_A$ and R$_B$ (100 kΩ) are chosen to isolate each voltage-divider from the other; if they were of comparable resistance to the voltage divider's, then a change in one would affect the divider at the other, causing a non-linear DAC response.

The electrical pulse which drives the laser was generated using the simple approach in Figure E.3. A bit is signaled by a rising-edge on the input of a 1:2 digital fanout. One copy is sent to the SPAD-readout board to set the position of the timing windows, while the other is coupled to a mismatched AND gate. One input is inverted, and the path-length difference between the two results in the creation of a ~ 950 ps positive pulse. This pulse is AC-coupled to the input of a GALI-51F inverting RF-amplifier, whose output is directly coupled to the VCSEL cathode, and is biased up to 4.1 V (approximately the same DC-level as the anode). The result is a 1.4-V negative pulse at the cathode, resulting in an approximately 1 ns long optical pulse. Driving the VCSEL directly with a pulse-pattern generator was attempted, but this resulted in additional fluctuations due to slight long-term variations in pulse amplitude. The AND-gate circuit is much less sensitive, and ensures that the same pulse is output every trigger's rising edge. The entire circuit was designed in the Eagle PCB-layout software, and its diagram, full schematic and board layout are shown in Figures E.3, E.4, and E.5 respectively.

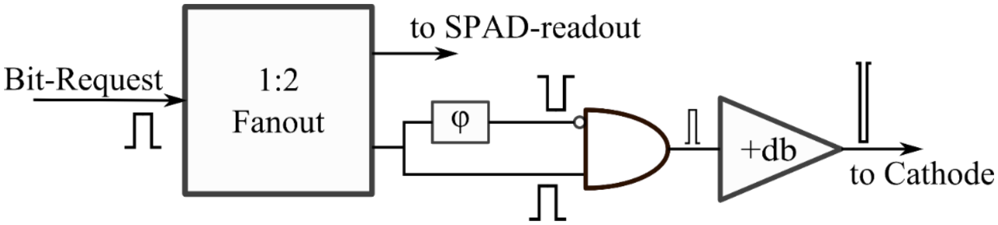


Figure E.3. Pulse-forming circuitry for the LLQRNG. The delayed-inverted path of the AND-gate causes a short electrical pulse to be output upon every rising edge of a bit-request. After amplification, its magnitude is ≈ 1.4 V peak-to-peak, enough to drive the VCSEL well above threshold.

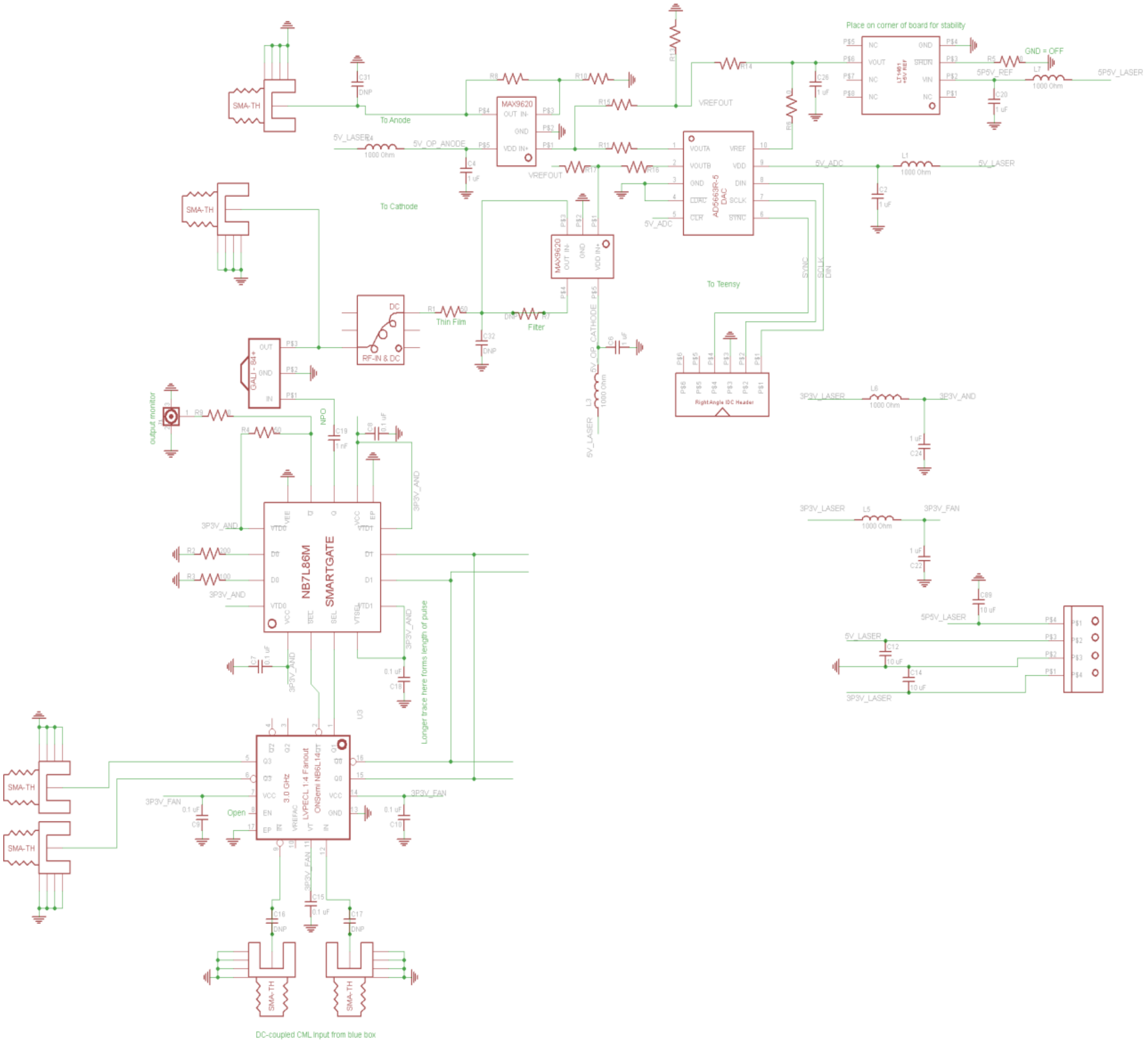


Figure E.4. Schematic of the op-amp summing amplifier circuit. The LLQRNG’s VCSEL anode is biased by the output of the summing amplifier, while the cathode is biased by the output of the RF-amplifier (‘GALI-51F’). Upon a bit request, the two logic gates at the bottom form a short (~ 950 ps) electrical pulse, which is input into the RF-amplifier.

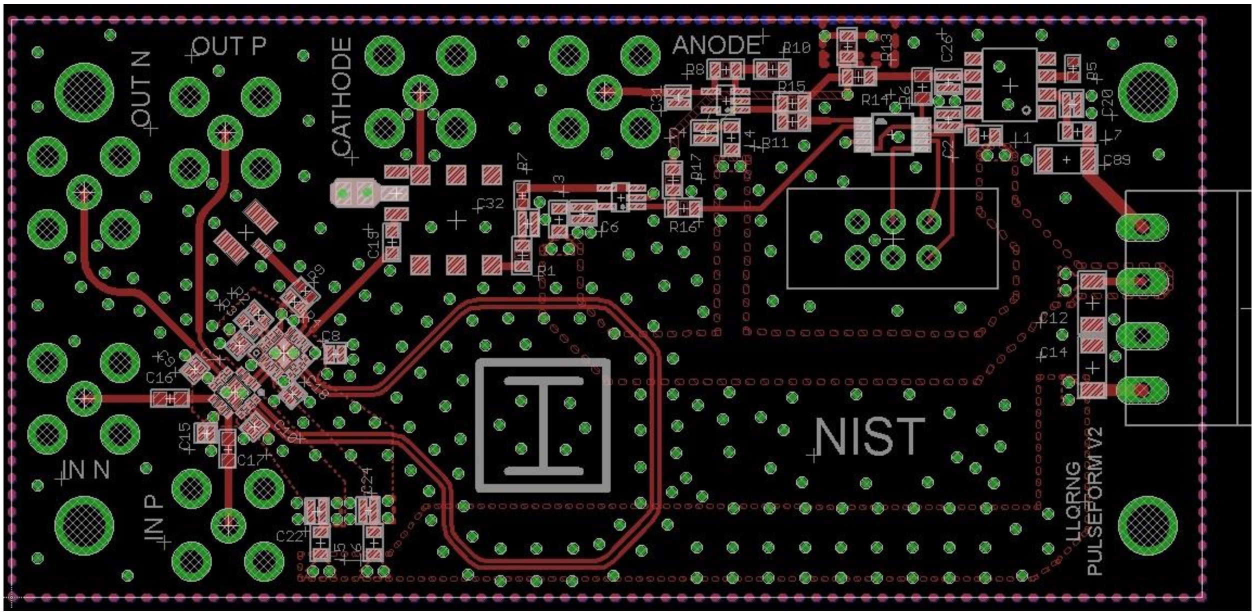


Figure E.5. Board-layout of the summing-amplifier / pulse-forming circuit.

Appendix F — LLQRNG Voltage Supplies

As shown in Figure 6.8, the voltage regulators used in the initial LLQRNG design were temperature sensitive, with typical environmental conditions resulting in voltage output inconsistencies at the 100 µV level. To alleviate this problem, every voltage supply was moved onto a single board, which was then bolted to a temperature-controlled aluminum plate and housed in the LLQRNG's insulated box. This resulted in a stable power supply for every component in the experiment.

There were three PCBs which required power supplies: the readout board of Appendix X (+3.3 V), the pulse-forming board of Appendix E (+5 V and +3.3 V), and the SPAD module itself (+5 V, +3.3 V, and -5 V). For the positive-voltage regulators, two metrics of interest are I_S, the amount of current it can source (or sink if negative), and V_{DO}, the dropout voltage. Given a specified output voltage V_{OUT}, the dropout voltage is how much additional voltage must be supplied in order for the output to be stable. For example, our 3.3 V regulator had V_{DO} = 130 mV, so in order for the output to be properly regulated at 3.3 V, the input power supply had to be above 3.43 V. The negative regulator was a charge-pump inverter type, which does not have a relevant V_{DO}. The voltage regulators used and their related metrics are listed in Table F.1.

Table F.1. Voltage regulators used for the LLQRNG system, and performance metrics.

Device Name	**V_{OUT} (V)**	**V_{DO} (V)**	**I_S (A)**
TPS73733	3.3	0.130	1.0
TPS79650	5.0	0.220	1.0
LTC1983-5	-5.0	N/A	0.1

By condensing all of the regulators onto one PCB, the entire LLQRNG system was able to run off of a single +5.5 V benchtop power supply. Schematics and board layouts were done in Eagle PCB design software, and are shown in Figures F.1 and F.2. The wider +5 V trace was to ensure a low-inductance path for the SPAD's power supply, which drew ~ 950 mA. As this was very close to the maximum rating of 1 A, this supply dissipated a lot of heat. In the future, a regulator capable of sourcing more current will be used.

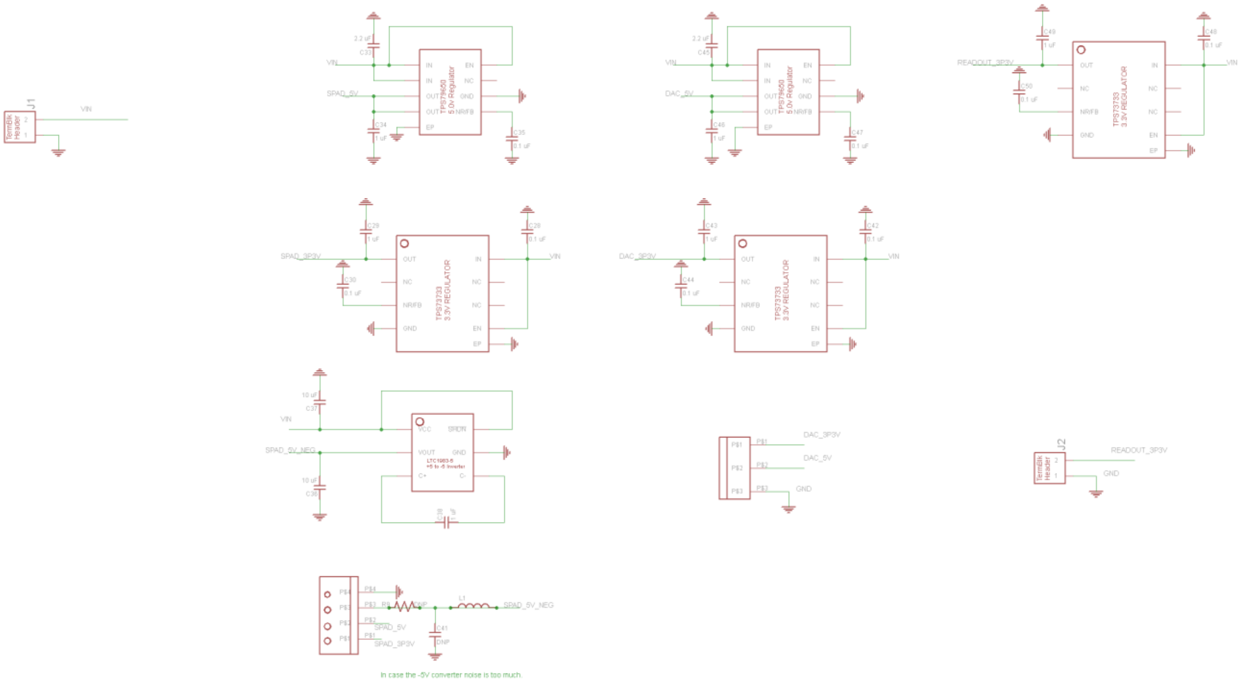


Figure F.1 Schematic of LLQRNG power-supply PCB. By temperature controlling all voltage regulators, the entire experiment's supply is kept stable, and only requires a single +5.5 V source.

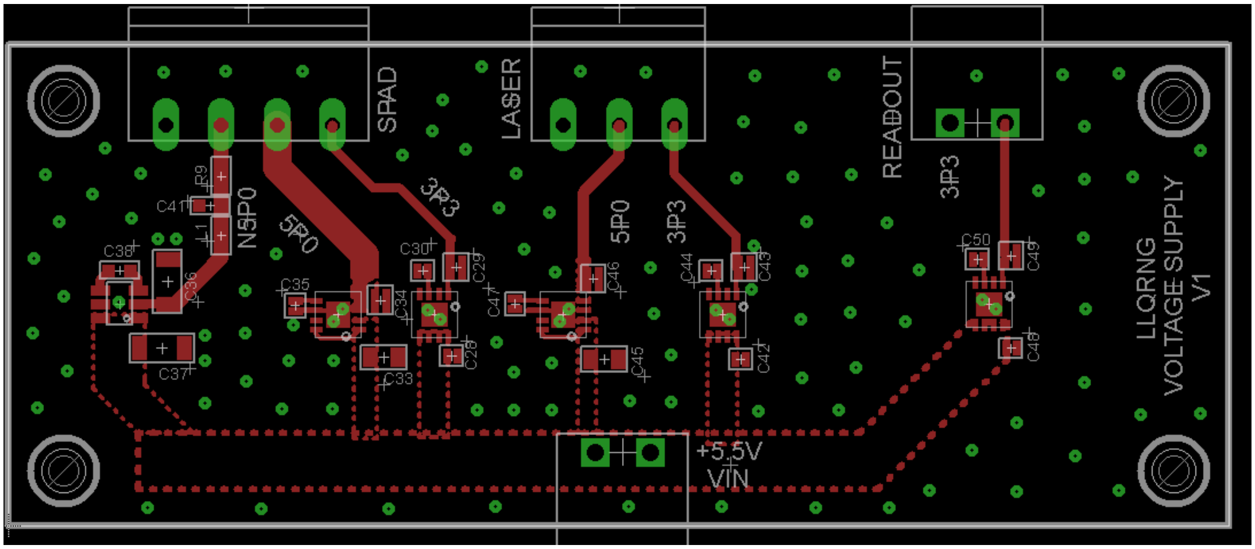


Figure F.2. PCB-board layout of the LLQRNG voltage-supply board. A single +5.5 V ('VIN') source supported power for all three sub-boards. The SPAD +5V required a large amount of current, so a low-inductance (wide) trace is used.

Appendix G — Laser Mounting PCB

The LLQRNG's VCSEL is vertically mounted on the end of a slip-plate, a mechanical element that allows for coarse optical alignment. Besides the VCSEL itself, there are no active electronics on this board. There are two SMA-connectors to couple to the anode and cathode bias voltages supplied by the voltage-summer board, and a small capacitor (1 μF) on the anode for bypass filtering purposes. The schematics and PCB board layouts are shown in Figures G.1 and G.2. The anode trace is essentially DC, only changing when the active feedback is applied. The cathode trace, however, must support the sub-ns laser-driving pulse, and its larger width reflect it being designed for an impedance of 50 Ω. The large holes around the outside of the board (green circles) are to accommodate the structure of the slip-plate and for mounting purposes.

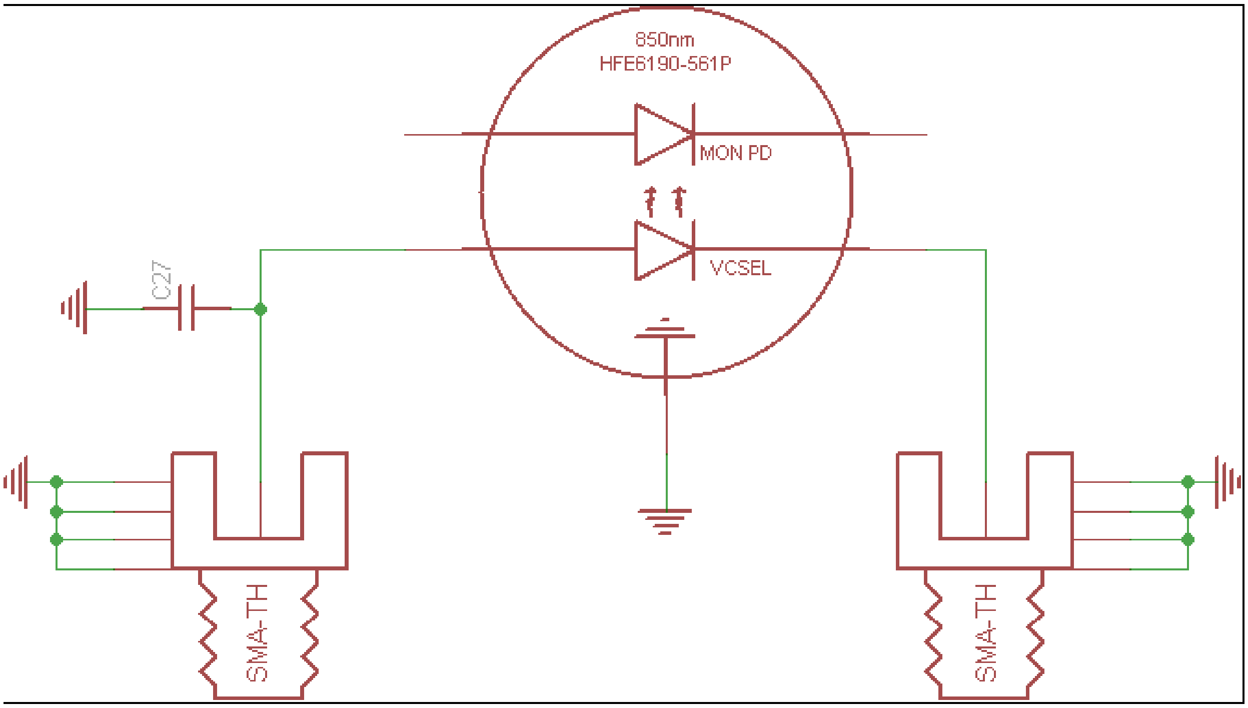


Figure G.1. Schematic of the VCSEL mounting PCB. The monitor photodiode on the VCSEL was left unconnected.

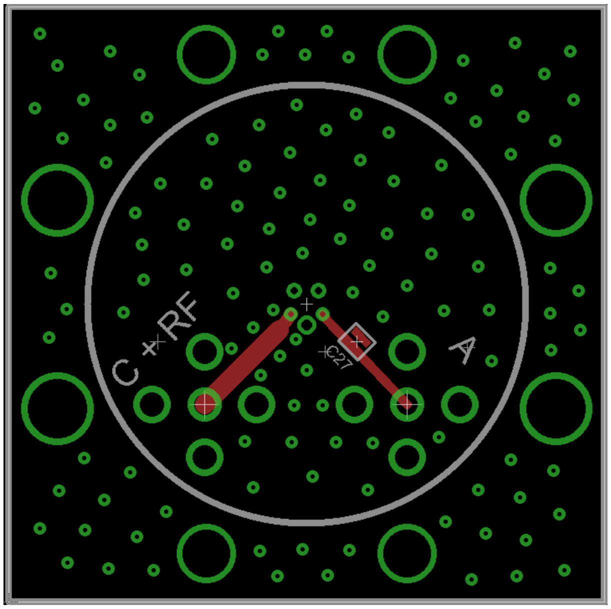


Figure G.2. PCB-layout of the VCSEL mounting PCB. The anode ('A') bias is mostly DC and has a small bypass capacitor for stability. The cathode trace must support the RF-frequency driving pulse and is designed for 50 Ω impedance.

Appendix H — SPAD Readout PCB

The SPAD-readout circuit board is responsible for sampling the state of the LLQRNG's SPAD during the correct timing interval; a conceptual logic diagram of its operation is shown in Figure H.1. There are two inputs: the SPAD's output (signaling a single-photon detection), and 'Trigger-In'. A copy of the original bit-request, Trigger-In is one of the outputs of the digital-fanout in the Voltage-Summer / Pulse-Forming circuit of Appendix F.

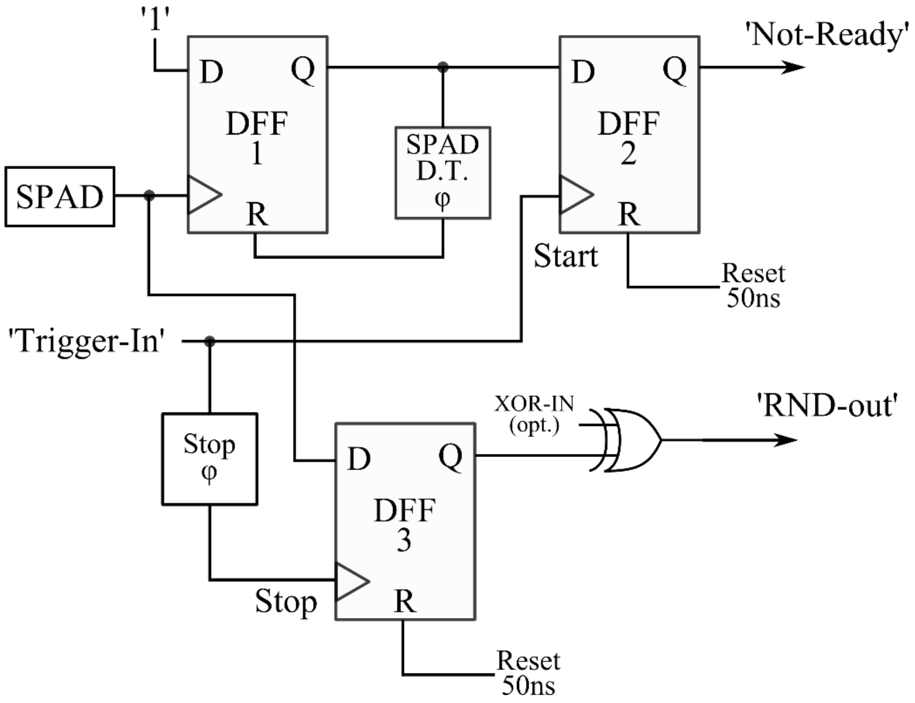


Figure H.1. Conceptual illustration of the LLQRNG's readout electronics. The arrival of a bit request signals the clocking of two flip-flops, at times corresponding to the start and stop of the timing window. The 'not-ready' output denotes whether the SPAD was in the process of recovering when a bit request arrived, and the 'RND-out' output corresponds to whether or not a valid detection arrived in the timing window.

The presence of a detection at the output of the SPAD causes the state of DFF1 to go high. The length of DFF1's reset delay (ϕ) is set to be equal to the dead time of the constituent SPAD, a value typically in the range of 50 ns. This results in a logical-high pulse at the input of DFF2 for the full dead time, and when clocked by the arrival of a trigger, determines whether or not the SPAD was active when a bit was requested. If a detection occurred in the ≈ 50 ns prior, then the SPAD would still be recovering when the trial began, and the 'Not-Ready' output off DFF2 reflects that condition.

The arrival of the trigger at DFF2 sets the start of the LLQRNG's timing window, and the clocking of DFF3 sets the end, or 'stop'. The stop delay is ~ 1 ns, and can be set with cable or a tunable delay line. Therefore, the value of 'RND-out' reflects whether or not a detection arrived in the time interval which begins at 'start' and ends at 'stop', and is the output of random-bit output of the LLQRNG. Each flip-flop is set to reset 50 ns after the arrival of Trigger-In. This gives the option for the LLQRNG to either hold state between requests, or output a 50 ns pulse, i.e., for use with a time-tagger. The addition of an exclusive-or gate allows for additional RNG systems to be combined with the output of the LLQRNG, presumably improving the output bit-quality. The PCB board layout and schematics are shown in Figures H.2. and H.3.

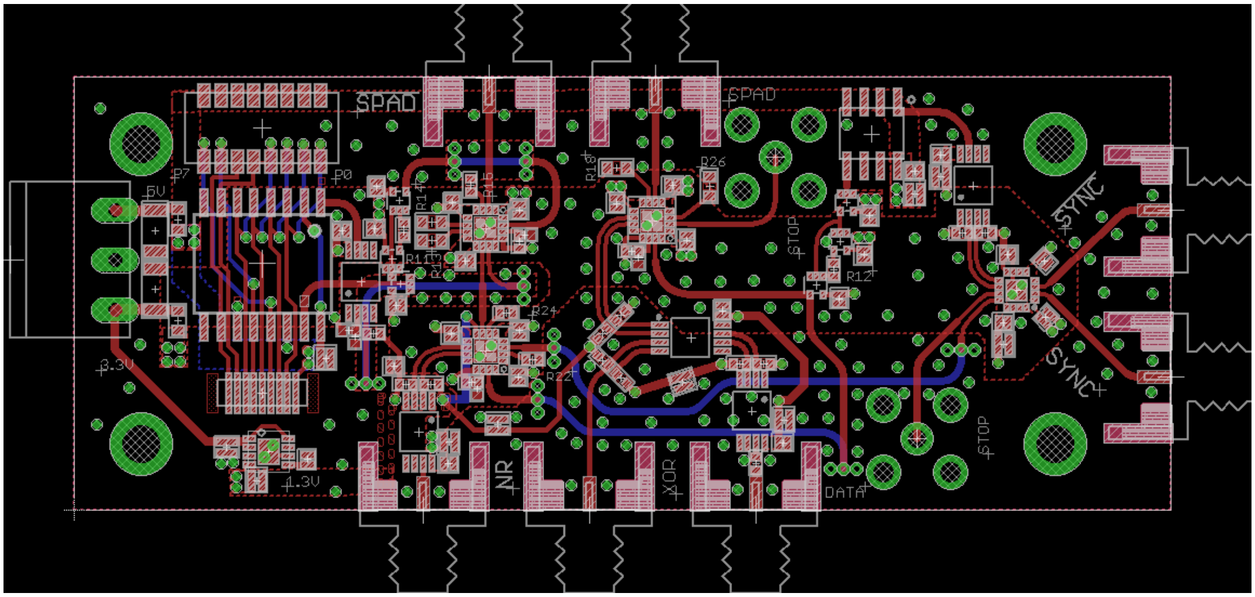


Figure H.2. Board layout of LLQRNG's readout electronics. Differing color traces and hatched polygons denote signals on internal layers.

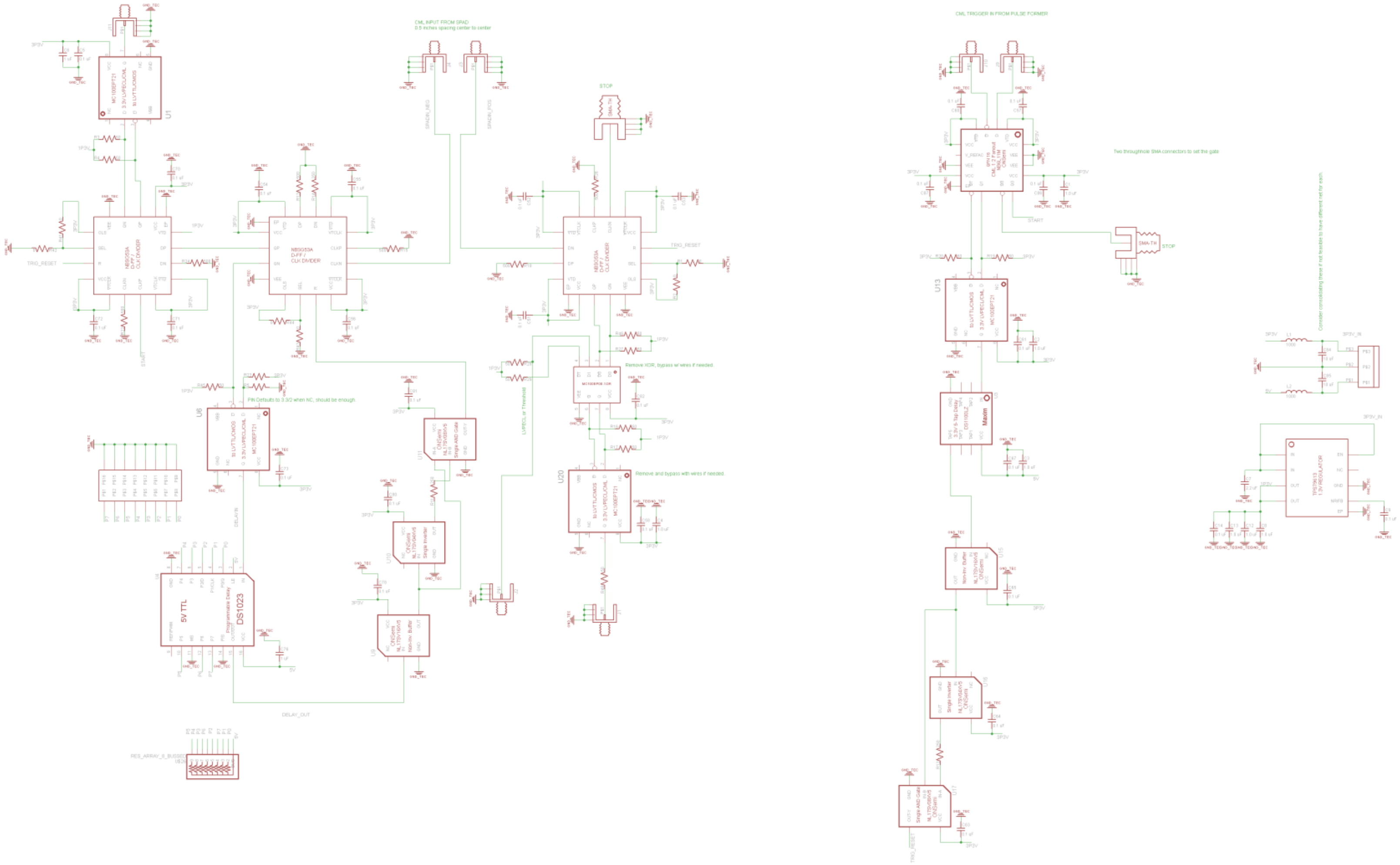

CML INPUT FROM SPAD
0.5 inches spacing center to center
STOP
SMA-TH
CML TRIGGER IN FROM PULSE FORMER
Two throughhole SMA connectors to set the gate
START
STOP
U1
MC100EPT21
NB6S53A
D-FF /
CLK DIVIDER
TRIG_RESET
Remove XOR, bypass w/ wires if needed
Remove and bypass with wires if needed
MC100EP08
U20
U13
U6
PIN Defaults to 3.3V when NC, should be enough
LVPECL or Threshold
5V TTL
Programmable Delay
DS1023
DELAY_OUT
RES_ARRAY_8_BUSSED
Maxim
DS1100LU
5-Tap Delay
Single AND Gate
Single Inverter
Non-Inv. Buffer
ONSemi
Consider consolidating these if not feasible to have different rail for each
REGULATOR

Figure H.3. Schematic diagram of LLQRNGs readout electronics.

Appendix I — Python Interface

After sampling 100,000 events, several pieces of data are sent from the Teensy-microcontroller to a PC over the USB interface. Specifically, the number of recorded random bits, the number of not-ready instances (and the state of the RND output for that bit), the current state of the active feedback in the form of the DAC-bit, and the values of three separate thermistors which are used to monitor the internal box-temperature. A Python program reads these values and updates an animated graph in real time. As the repetition rate of the experiment is 99.1 kHz, this results in the graph being updated approximately once a second. A sample of this output is shown in Figure I.1, although here the temperature data has been removed and instead replaced with a running average of the random-1 output in 60 sample blocks. This real-time monitoring allows for an immediate notification if the system starts operating in an extreme manner.

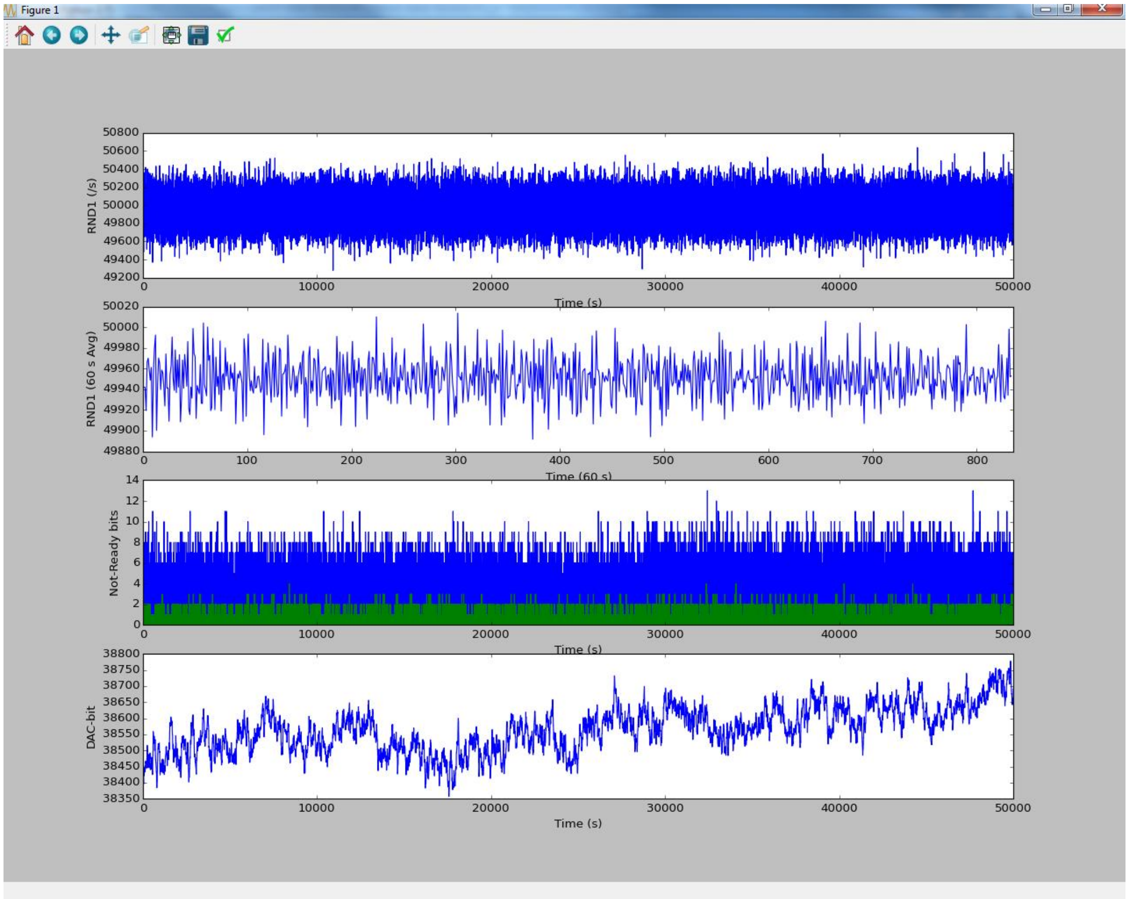


Figure I.1. Screenshot of real-time LLQRNG monitoring software. The Teensy microcontroller samples the statistics of the LLQRNG and outputs a result every 100,000 samples. This allows for monitoring the active-feedback, running bit probabilities, and internal box temperature (not shown).

References

[1] M. A. Nielsen and I. L. Chuang, *Quantum Computation and Quantum Information*, 10th Anniversary ed. Cambridge University Press, 2011.

[2] E. Jeffrey and P. G. Kwiat, "Delayed Choice Quantum Cryptography," in *Proceedings of QELS*, 2003.

[3] T. M. Graham, H. J. Bernstein, T. C. Wei, M. Junge, and P. G. Kwiat, "Superdense teleportation using hyperentangled photons.," *Nat. Commun.*, vol. 6, 2015.

[4] J. T. Barreiro, T. C. Wei, and P. G. Kwiat, "Beating the channel capacity limit for linear photonic superdense coding.," *Nat. Phys.*, vol. 23, 2008.

[5] P. W. Shor, "Polynomial-Time Algorithms for Prime Factorization and discrete Logarithms on a Quantum Computer," *ArXiv Quantum Phys. E-Prints*, 1995.

[6] "Secure Hash Standard (SHS)," *Fed. Inf. Process. Stand. Publ.*, vol. 180, no. 4, 2012.

[7] L. Vandersypen, M. Steffen, G. Breyta, et. al., "Experimental realization of Shor's quantum factoring algorithm using nuclear magnetic resonance," *Nature*, vol. 414, no. 6866, pp. 883–887, 2001.

[8] X. Nanyang, J. Zhu, D. Lu, et. al., "Quantum Factorization of 143 on a Dipolar-Coupling Nuclear Magnetic Resonance System," *Phys. Rev. Lett.*, vol. 108, no. 13, 2012.

[9] D. Nikesh and N. Bryans, "Quantum factorization of 56153 with only 4 qubits.," *arXiv:1411.6758*, 2014.

[10] C. Bennett and G. Brassard, "Quantum Cryptography: Public key distribution and coin tossing.," *Proc IEEE Conf Comput. Syst. Signal Process.*, 1984.

[11] J. C. Bienfang, "Quantum key distribution with 1.25 Gbps clock synchronization," *Opt. Express*, vol. 12, no. 9, pp. 2011–2016, 2004.

[12] M. A. Wayne and E. Jeffrey, "Photon arrival time quantum random number generation," *J. Mod. Opt.*, vol. 56, no. 4, pp. 516–522, 2009.

[13] M. A. Wayne and P. G. Kwiat, "Low-bias high-speed quantum random number generator via shaped optical pulses," *Opt. Express*, vol. 18, no. 8, 2010.

[14] Y.-Q. Nie, L. Huang, and Y. Liu, "The generation of 68 Gbps quantum random number by measuring laser phase fluctuations.," *Rev. Sci. Instrum.*, vol. 86, no. 063105, 2015.

[15] R. Holmes, B. Christensen, R. Wang, et. al., "Testing the limits of human vision with single photons," presented at the Frontiers in Optics, San Jose, CA.

[16] B. Hensen, H. Bernien, and A. Dreau, "Loophole-free Bell inequality violation using electron spins separated by 1.3 kilometers," *Nature*, vol. 526, pp. 682–686, 2015.

[17] M. Giustina, M. Versteegh, and S. Wengerowsky, "Significant-Loophole-Free Test of Bell's Theorem with Entangled Photons," *Phys. Rev. Lett.*, vol. 115, no. 25, 2015.

[18] L. K. Shalm, E. Meyer-Scott, and B. Christensen, "Strong Loophole-Free Test of Local Realism," *Phys. Rev. Lett.*, vol. 115, no. 25, 2015.

[19] M. J. Evans and J. S. Rosenthal, *Probability and Statistics: The Science of Uncertainty*, 2nd ed. W. H. Freeman and Company, 2010.

[20] J. von Neumann, "Various techniques used in connection with random digits," *J. Res. Natl. Bur. Stand.*, no. 3, pp. 36–38, 1951.

[21] D. H. Lehmer, "Mathematical methods in large-scale computing units," presented at the Second Symposium on Large-Scale Digital Calculating Machinery, Cambridge, Mass., 1949, pp. 141–146.

[22] G. Marsaglia, "Random Numbers Fall Mainly in the Planes," *Proc. Natl. Acad. Sci.*, vol. 61, no. 1, pp. 25–28, 1968.

[23] "https://en.wikipedia.org/wiki/RANDU."

[24] M. Matsumoto and T. Nishimura, "Mersenne Twister: a 623-dimensionally equidistributed uniform pseudo-random number generator," *ACM Trans. Model. Comput. Simul.*, vol. 8, no. 1, pp. 3–30, 1998.

[25] K. Zetter, “How a crypto ‘backdoor’ pitted the tech world against the NSA,” *Wired*, 2013.

[26] H. Schmidt, “Quantum-Mechanical Random-Number Generator,” *J. Appl. Phys.*, vol. 41, no. 462, 1970.

[27] G. M. Dillard and R. E. Simmons, “An electronic generator of random numbers,” *Res. Dev. Rep. 1152 Navy Electron. Lab*, 1962.

[28] M. Isida and H. Ikeda, “Random Number Generator,” *Ann. Inst. Stat. Math.*, vol. 8, no. 1, pp. 119–126, 1956.

[29] Rand Corporation, *A million random digits with 100,000 normal deviates.* Glencoe, Ill: Free Press, 1955.

[30] “LavaRND,” *www.lavarnd.org*, 2000.

[31] “True Random Number Service,” *www.random.org*.

[32] D. G. Marangon, G. Vallone, and P. Villoresi, “Random bits, true and unbiased, from atmospheric turbulence.,” *Sci. Rep.*, vol. 4, p. 5490, 2014.

[33] S. Murdoch, “Hot or Not: Revealing Hidden Services by their Clock Skew,” presented at the ACM CCS, 2006.

[34] M. Herrero-Collantes and J. C. Garcia-Escartin, “Quantum Random Number Generators,” *arXiv:1604.03304v1*, 2016.

[35] “Concepts in Digital Imaging Technology,” *hamamatsu.magnet.fsu.edu/articles/photomultipliers.html*, Aug. 2016.

[36] H. Schmidt, “PK Effect on Pre-Recorded Targets,” *J. Am. Soc. Psych. Res.*, vol. 70, 1976.

[37] J. Walker, “HotBits: A random number service,” *www.fourmilab.ch/hotbits/*.

[38] R. Duggirala, A. Lal, and S. Radhakrishnan, “Radioisotope Decay Rate Based Counting Clock,” *MEMS Ref.*, vol. 6, pp. 127–170, 2010.

[39] “Comscire: Pure Quantum True Random Number Generators,” *www.comscire.com*, 2014.

[40] P. I. Somlo, “Zener-diode noise generators,” *Electron. Lett.*, vol. 4, no. 14, p. 72, 1975.

[41] M. Stipcevic, “Fast nondeterministic random bit generator based on weakly correlated physical events,” *Rev. Sci. Instrum.*, vol. 75, no. 11, p. 4442, 2004.

[42] M. Stipcevic and C. K. Koc, *True Random Number Generators*. Switzerland: Springer International Publishing.

[43] G. Weihs and A. Zeilinger, “Photon statistics at beam-splitters: an essential tool in quantum information and teleportation,” in *Coherence and Statistics of Photons and Atoms*, J. Perina, Ed. Wiley, 2001.

[44] J. G. Rarity, P. C. M. Owens, and P. R. Tapster, “Quantum Random-number generation and Key Sharing,” *J. Mod. Opt.*, vol. 41, no. 12, pp. 2435–2444, 1994.

[45] T. Jennewein, U. Achleitner, G. Weihs, H. Weinfurter, and A. Zeilinger, “A fast and compact quantum random number generator,” *Rev. Sci. Instrum.*, vol. 71, p. 1675, 2000.

[46] A. Stefanov, N. Gisin, L. Guinnard, and H. Zbinden, “Optical quantum random number generator,” *J. Mod. Opt.*, vol. 47, no. 4, pp. 595–598, 2000.

[47] A. Migdall, S. V. Polyakov, J. Fan, and J. C. Bienfang, Eds., *Single-photon generation and detection: physics and applications*, vol. 45. Academic Press, 2013.

[48] M. Grafe, R. Heilmann, and A. Perez-Leija, “On-chip generation of high-order single-photon W-states,” *Nat. Photonics*, vol. 8, no. 10, pp. 791–795.

[49] “Quantis QRNG.” www.idQuantique.com

[50] M. Stipčević and B. M. Rogina, “Quantum random number generator based on photonic emission in semiconductors,” *Rev. Sci. Instrum.*, vol. 78, no. 4, p. 45104, 2007.

[51] L. M. Yu, M. J. Yang, and P. X. Wang, “A sampling method for quantum random bit generation,” *Rev. Sci. Instrum.*, vol. 81, no. 4, p. 046107, 2010.

[52] “quRNG,” *www.qutools.com*, 2012.

[53] S. Li, L. Wang, and L. A. Wu, “True random number generator based on discretized encoding of the time interval between photons,” *J. Opt. Soc. Am.*, vol. 30, no. 1, p. 124, 2013.

[54] M. Jofre, M. Curty, F. Steinlechner, et. al., “True random numbers from amplified quantum vacuum,” *Opt. Express*, vol. 19, no. 21, pp. 20665–20672, 2011.

[55] H. Guo, W. Tang, Y. Liu, and W. Wei, “Truly random number generation based on measurement of phase noise of a laser,” *Phys. Rev. E*, vol. 81, no. 5, p. 051137, 2010.

[56] F. Xu, B. Qi, and X. Ma, “Ultrafast quantum random number generation based on quantum phase fluctuations,” *Opt. Express*, vol. 20, no. 11, p. 12366, 2012.

[57] C. Abellan, W. Amaya, and M. Mitchell, “Generation of Fresh and Pure Random Numbers for Loophole-Free Bell Tests,” *Phys. Rev. Lett.*, vol. 115, p. 250403, 2015.

[58] M. Furst, H. Weier, and H. Weinfurter, “High speed optical quantum random number generation,” *Opt. Express*, vol. 18, no. 12, pp. 13029–13037, 2010.

[59] Y. Jian, M. Ren, and H. Zeng, “Two-bit quantum random number generator based on photon-number-resolving detection,” *Rev. Sci. Instrum.*, vol. 82, 2011.

[60] Micro Photon Devices, “Quantum random number generator,” *Wwwmicro-Photon-Devicescom*, 2014.

[61] Q. Yan, B. Zhao, and H. Yang, “High-speed quantum random number generation by continuous measurement of arrival time of photons,” *Rev. Sci. Instrum.*, vol. 86, p. 073113, 2015.

[62] C. R. S. Williams, J. C. Salevan, and T. E. Murphy, “Fast physical random number generator using amplified spontaneous emission,” *Opt. Express*, vol. 18, no. 23, p. 23584, 2010.

[63] X. Li, A. Cohen, and R. Roy, “Scalable parallel physical random number generator based on a superluminescent LED,” *Opt. Lett.*, vol. 36, no. 6, pp. 1020–1023, 2011.

[64] P. J. Bustard, D. Moffatt, and B. J. Sussman, “Quantum random bit generation using stimulated Raman scattering,” *Opt. Express*, vol. 19, no. 25, pp. 25173–25180, 2011.

[65] P. J. Bustard, D. G. England, and J. Nunn, “Quantum random bit generation using energy fluctuations in stimulated Raman scattering,” *Opt. Express*, vol. 21, no. 24, p. 29350, 2013.

[66] D. G. England, P. J. Bustard, and B. J. Sussman, “Efficient Raman generation in a waveguide: A route to ultrafast quantum random number generation,” *Appl. Phys. Lett.*, vol. 104, no. 5, p. 051117, 2014.

[67] S. Pironio, A. Acin, and C. Monroe, “Random numbers certified by Bell’s theorem,” *Nat. Lett.*, vol. 464, 2010.

[68] B. Christensen, K. T. McCusker, J. B. Altepeter, and P. G. Kwiat, “Detection-Loophole-Free Test of Quantum Nonlocality, and Applications,” *Phys. Rev. Lett.*, vol. 111, 2013.

[69] M. Fiorentino, C. Santori, and R. G. Beausoleil, “Secure self-calibrating quantum random-bit generator,” *Phys. Rev. A*, vol. 75, p. 032334, 2007.

[70] N. Lutkenhaus, J. Cohen, and H.-K. Lo, “Efficient use of detectors for random number generation,” US Patent 7197523 B2.

[71] J. Altepeter, E. Jeffrey, and P. Kwiat, “Quantum random number generator.”U.S. Patent Application 20060010182, January 12, 2006.

[72] R. Loudon, *The Quantum Theory of Light*, 3rd ed. Oxford, New York: Oxford University Press, 2000.

[73] M. Stevens, “Photon Statistics, Measurements, and Measurement Tools,” in *Single-Photon Generation and Detection*, vol. 45, Academic Press, 2013, pp. 25–66.

[74] E. Jeffrey, “Advanced Quantum Communication Systems,” PhD Dissertation, University of Illinois, 2007.

[75] idQuantique - ID100 Single Photon Detectors, “http://www.idquantique.com/wordpress/wp-content/uploads/id100-specs.pdf.”

[76] ACAM, TDC-GPX time-to-digital converter, “www.acam-usa.com/Content/English/gpx/gpx_1.html.”

[77] Xilinx - Virtex 6 FPGA, “http://www.xilinx.com/support/documentation/data_sheets/ds150.pdf.”

[78] M. A. Wayne, “Photon Arrival Time Quantum Random Number Generation,” master's thesis, University of Illinois, 2009.

[79] NIST, “A Statistical Test Suite for Random and Pseudorandom Number Generators for Cryptographic Applications,” *NIST Spec. Publ.*, 2009.

[80] G. Marsaglia, “DIEHARD Random Number Tests,” *www.phy.duke.edu/~rbg/General/dieharder.php*

[81] P. L’Ecuyer and R. Simard, “TestU01: A C library for empirical testing of random number generators,” *ACM Trans. Math. Softw.*, vol. 33, no. 4, p. 2007.

[82] “SAP500 Product Datasheet.” [Online]. Available: http://www.lasercomponents.com/us/product/silicon-apds-for-photon-counting/. [Accessed: 16-Jan-2014].

[83] “C30902 and C30921 Series SPAD,” *www.excelitas.com/downloads/DTS_C30902_C309021.pdf*.

[84] “Tau-Spad (discontinued),” *www.picoquant.com*.

[85] M. A. Wayne, A. Restelli, J. C. Bienfang, and P. G. Kwiat, “Afterpulse Reduction Through Prompt Quenching in Silicon Reach-Through Single-Photon Avalanche Diodes,” *J. Light. Technol.*, vol. 32, no. 21, pp. 4097–4103, 2014.

[86] K. E. Jensen *et al.*, “Afterpulsing in Geiger-mode avalanche photodiodes for 1.06μm wavelength,” *Appl. Phys. Lett.*, vol. 88, no. 13, p. 133503, Mar. 2006.

[87] A. Ingargiola, M. Assanelli, A. Gallivanoni, and S. Cova, “Avalanche buildup and propagation effects on photon-timing jitter in Si-SPAD with non-uniform electric field,” in *Proceedings of SPIE*, 2009, vol. 7320.

[88] S. Cova, M. Ghioni, A. Lacaita, C. Samori, and F. Zappa, “Avalanche photodiodes and quenching circuits for single-photon detection,” *Appl. Opt.*, vol. 35, no. 12, pp. 1956–1976, Apr. 1996.

[89] N. Namekata, S. Adachi, and S. Inoue, “Ultra-Low-Noise Sinusoidally Gated Avalanche Photodiode for High-Speed Single-Photon Detection at Telecommunication Wavelengths,” *IEEE Photonics Technol. Lett.*, vol. 22, pp. 529–531, 2010.

[90] Z. L. Yuan, B. E. Kardynal, A. W. Sharpe, and A. J. Shields, “High Speed Single Photon Detection in the Near Infrared,” *Appl. Phys. Lett.*, vol. 91, p. 041114, 2007.

[91] A. Restelli, J. C. Bienfang, and A. Migdall, “Single-Photon Detection Efficiency up to 50% at 1310 nm with an InGaAs/InP Avalanche Diode Gated at 1.25 GHz,” *Appl. Phys. Lett.*, vol. 102, p. 141104, 2013.

[92] “Nitronex NPTB00004 Datasheet.” [Online]. Available: http://www.nitronex.com/pdfs/NPTB00004.pdf. [Accessed: 24-Mar-2014].

[93] S. Cova, A. Lacaita, and G. Ripamonti, “Trapping phenomena in avalanche photodiodes on nanosecond scale,” *IEEE Electron Device Lett.*, vol. 12, no. 12, pp. 685–687, Dec. 1991.

[94] M. Stipčević, D. Wang, and R. Ursin, “Characterization of a commercially available large area, high detection efficiency single-photon avalanche diode,” *J. Light. Technol.*, vol. 31, no. 23, pp. 3591–3596, Dec. 2013.

[95] B. Rossi, N. Hilberry, and B. Hoag, “The Variation of the Hard Component of Cosmis Rays with Height and the Disintegration of Mesotrons,” *Phys. Rev.*, vol. 57, no. 6, pp. 461–469, 1940.

[96] E. Raisanen-Ruotsalainen, E. Rahkonen, and T. Kostamovaara, “An Integrated time-to-digital converter with 30-ps single-shot precision.,” *IEEE J. Solid-State Circuits*, vol. 35, no. 10, pp. 1507–1510, 2000.

[97] J. Kalisz, M. Pawlowsky, and R. Pelka, “Error Analysis and Design of the Nutt Time-Interval Digitiser with Picosecond Resolution,” *J Phys E Sci Instrum*, vol. 20, pp. 1330–1341, 1987.

[98] S. Henzler, *Time-to-Digital Converters*, 1st ed. Springer International Publishing, 2010.

[99] C.-T. Ko, K.-P. Pun, and A. Gothenberg, “A 5-ps sub-ranging time-to-digital converter with DNL calibration.,” *Microelectron. J.*, vol. 46, no. 12, pp. 1469–1480, 2015.

[100] J. Yu, F. F. Dai, and R. C. Jaeger, “A 12-bit Vernier Ring Time-to-Digital Converter in 0.13 um CMOS Technology.,” *IEEE J. Solid-State Circuits*, vol. 45, no. 4, pp. 830–842, 2010.

[101] J. Yu and F. F. Dai, “A 3-dimensional Vernier ring time-to-digital converter in 0.13 um CMOS,” presented at the Custom Integrated Circuits Conference, 2010.

[102] Y. Liu, U. Vollenbruch, Y. Chen, and R. Weigel, “A 6 ps resolution pulse-shrinking time-to-digital converter as phase detector in multi-mode transceiver.,” presented at the IEEE Xplore Conference: Radio and Wireless Symposium, 2008.

[103] “Xilinx Virtex-6 Family Overview.” [Online]. Available: http://www.xilinx.com/products/silicon-devices/fpga/virtex-6/. [Accessed: 16-Jan-2014].

[104] P. Chen, P.-Y. Chen, J.-S. Lai, and Y.-J. Chen, “FPGA Vernier Digital-to-Time Converter with 1.58 ps Resolution and 59.3 Minutes Operation Range,” *IEEE Trans. Circuits Syst. I*, vol. 57, no. 6, 2010.

[105] J. Wu, “On-Chip processing for the wave union TDC implemented in FPGA,” *Real Time Conf.*, 2009.

[106] M. Buchele, H. Fischer, F. Herrmann, K. Konigsmann, C. Schill, and S. Schopferer, “The GANDALF 128-Channel Time-to-Digital Converter,” presented at the 2nd International Conference on Technology and Instrumentation in Particle Physics, Freiburg, Germany, 2011.

[107] E. Billauer, “Xillybus PCIe Core,” *www.xillybus.com*.

[108] “Adsantec ASNT2011-PQA Demultiplexer.” [Online]. Available: http://www.adsantec.com/26-asnt2011-pqa.html. [Accessed: 16-Jan-2014].

[109] M. Hsu, H. Finkelstein, and S. Esener, “A CMOS STI-Bound Single-Photon Avalanche Diode with 27-ps Timing Resolution and a Reduced Diffusion Tail,” *IEEE Electron Device Lett.*, vol. 30, no. 6, pp. 641–643, 2009.

[110] P. Holland, “What’s Wrong with Einstein’s 1927 Hidden-Variable Interpretation of Quantum Mechanics?,” *Found. Phys.*, vol. 35, no. 2, pp. 177–196, 2005.

[111] L. de Broglie, “La mécanique ondulatoire et la structure atomique de la matière et du rayonnement,” *J. Radium Phys.*, vol. 8, no. 5, pp. 225–241, 1927.

[112] A. Einstein, B. Podolsky, and N. Rosen, “Can Quantum-Mechanical Description of Physical Reality be Considered Complete?,” *Phys. Rev.*, vol. 47, no. 10, pp. 777–780, 1935.

[113] J. Bell, “On the Einstein Podolsky Rosen Paradox,” *Physics*, vol. 1, no. 3, pp. 195–200, 1964.

[114] A. Aspect, P. Grangier, and G. Roger, “Experimental Tests of Realistic Local Theories via Bell’s Theorem,” *Phys. Rev. Lett.*, vol. 47, no. 460, 1981.

[115] “eBACS: ECRYPT Benchmarking of Cryptographic Systems.” https://bench.cr.yp.to

[116] Finisar, “Application Note: Modulating VCSELs.”2007.

[117] S. Kaplan, L. Hanssen, A. Migdall, and G. Lefever-Button, “Characterization of High-OD Ultrathin Infrared Neutral Density Filters,” in *Proceedings of SPIE 3425*, 1998.

[118] D. Allan, “The Statistics of Atomic Frequency Standards,” *Proc IEEE*, vol. 54, no. 2, pp. 221–230, 1966.

[119] W. K. Pratt, J. Kane, and H. C. Andrews, “Hadamard Transform Image Coding,” *Proc IEEE*, vol. 57, no. 1, pp. 38–67, 1969.

[120] W. J. Riley, *Handbook of Frequency Stability Analysis*. 2008.

[121] L. Galleani, “The Dynamic Allan Variance III: Confidence and Detection Surfaces,” *IEEE Trans. Ultrason. Ferroelectr. Freq. Control*, vol. 58, no. 8, 2011.

[122] L. Galleani, “The Dynamic Allan Variance,” *IEEE Trans. Ultrason. Ferroelectr. Freq. Control*, vol. 56, no. 3, 2009.

[123] Analog Devices, “HMC445 Active x16 Frequency Multiplier,” *Wwwanalogcommediaentechnical-Doc.-Sheetshmc445pdf*.

[124] P. Wade, “Pipe-Cap Filters Revisited,” *www.w1ghz.org*.

[125] K. Britain, “Cheap Microwave Filters,” in *Proceedings of Microwave Update ’88*, 1988, pp. 159–163.