Spatio-temporal optical random number generator

M. Stipčević [1,2,3*] and J. E. Bowers[1]

[1]*Department of Electrical and Computer Engineering, University of California Santa Barbara, Santa Barbara, USA*
[2]*Department of Experimental Physics, Ruđer Bošković Institute, Zagreb, Croatia*
[3]*Photonics and Quantum Optics Unit, Center of Excellence for Advanced Materials and Sensing Devices, Ruđer Bošković Institute, Zagreb, Croatia*
[*]*mario.stipcevic@irb.hr*

Abstract: We present a first random number generator (RNG) which simultaneously uses independent spatial and temporal quantum randomness contained in an optical system. Availability of the two independent sources of entropy makes the RNG resilient to hardware failure and signal injection attacks. We show that the deviation from randomness of the generated numbers can be estimated quickly from simple measurements thus eliminating the need for usual time-consuming statistical testing of the output data. As a confirmation it is demonstrated that generated numbers pass NIST Statistical test suite.

OCIS codes: (270.5585) Quantum information and processing; (270.5570) Quantum detectors; (270.5568) Quantum cryptography.

References and links

1. P. Hellekalek, "Good random number generators are (not so) easy to find," Math. Comput. Simul. **46**(5-6), 485–505 (1998).
2. I. Goldberg, and D. Wagner, "Randomness in the Netscape Browser," Dr. Dobb's, January (1996).
3. T. H. Click, A. Liu, and G. A. Kaminski, "Quality of Random Number Generators Significantly Affects Results of Monte Carlo Simulations for Organic and Biological Systems," J. Comput. Chem. **32**(3), 513–524 (2011).
4. A. Proykova, "How to improve a random number generator," Comput. Phys. Commun. **124**(2-3), 125–131 (2000).
5. A. Stefanov, N. Gisin, O. Guinnard, L. Guinnard, and H. Zbinden, "Optical quantum random number generator," J. Mod. Opt. **47**, 595–598 (2000).
6. T. Jennewein, U. Achleitner, G. Weihs, H. Weinfurter, and A. Zeilinger, "A Fast and Compact Quantum Random Number Generator," Rev. Sci. Instrum. **71**(4), 1675–1680 (2000).
7. A. Figotin, A. Y. Gordon, S. A. Molchanov, V. P. Popovich, J. E. Quinn, G. N. Stetsenko, N. M. Stravrakas, and I. M. Vitebskiy, "A random number generator based on spontaneous alpha-decay," PCT application WO0038037A1 (2000).
8. M. A. Wayne, E. R. Jeffrey, G. M. Akselrod, and P. G. Kwiat, "Photon arrival time quantum random number generation," J. Mod. Opt. **56**(4), 516–522 (2009).
9. M. Stipcević and B. M. Rogina, "Quantum random number generator based on photonic emission in semiconductors," Rev. Sci. Instrum. **78**(4), 045104 (2007).
10. J. G. Rarity, P. C. M. Owens, and P. R. Tapster, "Quantum random-number generator and key sharing," J. Mod. Opt. **41**(12), 2435–2444 (1994).
11. R. Davies, "Exclusive OR (XOR) and hardware random number generators," February 28, 2002, URL: http://www.robertnz.net/pdf/xor2.pdf
12. J. von Neumann, Various techniques for use in connection with random digits," *von Neumann Collected Works*, Vol. 5, Pergamon, 768–770 (1963).
13. M. Stipčević and D. J. Gauthier, "Precise Monte Carlo Simulation of Single-Photon Detectors," Proc. SPIE Defense, Security and Sensing, 29 April - 3 May 2013, Baltimore, Maryland, USA, also appear as arXiv:1411.3663v1 [quant-ph].
14. D. E. Knuth, *The Art of Computer Programming*, Vol. 2, *Third edition* (Addison-Wesley, Reading, 1997).
15. R. Shaltiel, "Recent developments in explicit constructions of extractors," Bull. EATCS **77**, 67–95 (2002).
16. R. Shaltiel, "How to get more mileage from randomness extractors," Random Structures Algorithms **33**(2), 157–186 (2008).

#231845 - $15.00 USD Received 5 Jan 2015; revised 8 Mar 2015; accepted 24 Mar 2015; published 24 Apr 2015

17. B. Chor, O. Goldreich, J. Hasted, J. Freidmann, S. Rudich, and R. Smolensky, "The bit extraction problem or t-resilient functions," 26th Annual Symposium on Foundations of Computer Science (FOCS), 396–407, IEEE (1985).
18. P. Lacharme, "Analysis and Construction of Correctors," IEEE Trans. Inf. Theory **55**(10), 4742–4748 (2009).
19. R. Davies, Statistics Research Associates Limited, 8 Bristol Street, Island Bay, Wellington, 6023, New Zealand (private communication 2013/2014).
20. A. Rukhin, J. Soto, J. Nechvatal, M. Smid, E. Barker, S. Leigh, M. Levenson, M. Vangel, D. Banks, A. Heckert, J. Dray, and S. Vo, "A Statistical Test Suite for the Validation of Random Number Generators and Pseudo Random Number Generators for Cryptographic Applications," NIST Special Publication 800–22rev1a (dated April 2010). URL: *http://csrc.nist.gov/groups/ST/toolkit/rng/documents/sts-2.1.1.zip*
21. A. Theodore Markettos and W. Simon, Moore. 2009. "The Frequency Injection Attack on Ring-Oscillator-Based True Random Number Generators," Proc. 11th International Workshop on Cryptographic Hardware and Embedded Systems (CHES '09), Christophe Clavier and Kris Gaj (Eds.). Springer-Verlag, Berlin, Heidelberg, 317–331 (2009).
22. P. Bayon, L. Bossuet, A. Aubert, V. Fischer, F. Poucheret, B. Robisson, and P. Maurine, "Contactless electromagnetic active attack on ring oscillator based true random number generator," Proc. Third international conference on Constructive Side-Channel Analysis and Secure Design (COSADE'12), Werner Schindler and Sorin A. Huss (Eds.). Springer-Verlag, Berlin, Heidelberg (2012).
23. M. Dichtl and J. D. Golic, "High-speed true random number generation with logic gates only," in 1248 Cryptographic Hardware and Embedded Systems (CHES), ed. by P. Paillier, I. Verbauwhede 1249 (Springer, Berlin, 2007), 45–62.
24. A. Uchida, K. Amano, M. Inoue, K. Hirano, S. Naito, H. Someya, I. Oowada, T. Kurashige, M. Shiki, S. Yoshimori, K. Yoshimura, and P. Davis, "Fast physical random bit generation with chaotic semiconductor lasers," Nat. Photonics **2**(12), 728–732 (2008).
25. D. Frauchiger, R. Renner, and M. Troyer, "True randomness from realistic quantum devices," arXiv:1311.4547 [quant-ph].

1. Introduction

The ability to generate random numbers is an important resource in many areas of science and technology: computer security, cryptography, probabilistic computation (Monte Carlo), over-Turing computing techniques (e.g. randomized algorithms), simulations, labeling of prepaid and gift cards, industrial testing, online hazard games and automata, scientific research etc. The vast majority of today's computers are deterministic (e.g. PCs, tablets and mobile phones), and so they cannot *per se* create random numbers: that task is left to a random number generator (RNG). Random number generators are usually categorized as either pseudo random (PRNG) or physical or "true" random number generators (TRNG).

A PRNG is a mathematical formula, or more generally a deterministic algorithm which, starting from a certain initial number (seed) that defines the initial state, produces a string of numbers that looks random in the sense that it possesses a certain set of desirable statistical properties, but in fact is completely deterministic and highly losslessly compressible by definition [1], neither of which should be a characteristic of a truly random sequence.

Although there is no criterion for an algorithm to be named a "PRNG", the only undisputable characteristic of any PRNG is that it is *provably non-random* because it is already known how to predict all the numbers in the pseudo random sequence, namely by using the very PRNG algorithm. Nevertheless, PRNGs are very frequently used: their popularity stems from the fact that they can be realized as a piece of software and run on a computer or programmable devices (mobile phone, smart card, etc.) thus offering an illusion that the device now also has access to random numbers without any cost in additional hardware! However, while algorithmically generated pseudo random numbers can be used for some applications, they are by construction deterministic and therefore, at least in theory, predictable which makes them risky for use in cryptography [2] as well as frequent cause of erroneous results in statistical calculations and simulations [3,4].

A non-determinism is sought in physical RNGs, notably quantum random number generators (QRNG) that extract random numbers by performing repeated measurements on a certain, specially prepared quantum systems [5–9]. The rationale behind QRNGs is that quantum theory allows the existence of fundamentally random and unpredictable physical processes such that one can, in principle, extract truly random numbers from them.

#231845 - $15.00 USD Received 5 Jan 2015; revised 8 Mar 2015; accepted 24 Mar 2015; published 24 Apr 2015

This paper is organized as follows. First, we present an original analysis of well known "spatial" and "temporal" optical quantum random number generating principles. Next, based on gained insights, we construct a new bit extracting principle with significantly improved characteristics. Its notable novelty is simultaneous extraction of two independent random bit strings one from spatial and other from temporal quantum information contained in the proposed optical system, utilizing the same set of photon detections. Finally, we build and test a physical QRNG based on the new principle. Yet another novelty in our approach is that we *prove* limits of deviation from randomness based on a set of simple measurements.

2. Beam splitter spatial method (BSR)

A well known beam splitting principle of generating random binary numbers (bits) [10], [5] is shown in Fig. 1. Ideally, a light beam ($|\psi\rangle$) is split by a balanced beam splitter (BS) causing independent, equally probable photon detections by the two identical detectors D0 and D1 sitting at each exit arm of the BS. A detection by D0 is defined as generation of bit value "0" whereas a detection by D1 is defined as generation of a bit value "1". Because the value of a random number depends on the position at which the photon is detected, this method is sometimes referred to as "spatial".

To overcame inevitably imperfect balance of the BS and detectors, in our implementation we use two neutral density filters, shown in Fig. 1, that allow us to equalize photon detection probabilities of the two detectors. The CW light source is a red LED (Hamamatsu L7868, $\lambda = 670$ nm, $\Delta\lambda = 30$ nm FWHM) powered by an adjustable current source that allows us to run the random number generator at various photon detection rates. The beam splitter BS is a fusion fiber 50:50 splitter (Thorlabs FC632-50B-FC). The two near-identical photon counting detectors D0 and D1 are based upon SLiK diodes recovered from Excelitas SPCM-AQRH modules, complemented by a home-made active quenching circuits and operated at −10 °C. They feature the dead time of 24 ns, dark count rate of ~ 250 cps, afterpulsing lifetime of $\tau_a \sim 33$ ns and visible afterpulsing probability (afterpulses visible after the dead time) of about $p_a = 0.031$.

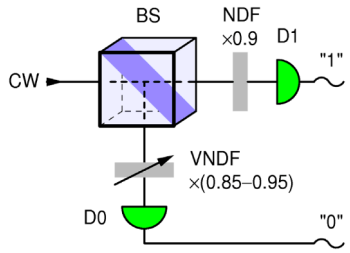

Fig. 1. Optical part of the beam splitter quantum random number generator consists of the beam splitter BS, detectors D0 and D1, fixed neutral density filter NDF and variable neutral density filter VNDF. The two filters allow for fine equalizing the probabilities of generating ones and zeros.

Even though the beam splitter method in theory generates perfect random numbers, we will show that when it is used with continuous random light (we name it the BSR method) it is quite sensitive to imperfections in the experimental setup notably those of the detectors. Bit sequences generated by any physical RNG generally feature two types of deviations from randomness which seem unavoidable: bias and correlations among bits. Bias is a measure of uneven probability of generating ones and zeros and is defined as:

$$b = \frac{p(1) - p(0)}{2} = p(1) - \frac{1}{2}. \qquad (1)$$

#231845 - $15.00 USD Received 5 Jan 2015; revised 8 Mar 2015; accepted 24 Mar 2015; published 24 Apr 2015

Following the above definition, an estimate of bias of a string of bits $x_i \in \{0,1\}$ is obtained as:

$$\hat{b} = \frac{\sum_{i=1}^{N} x_i}{N} - \frac{1}{2}. \tag{2}$$

with the variance of $1/(2\sqrt{N})$. Sources of bias in the BSR method are: (1) unbalanced beam splitter; and (2) uneven detection efficiencies of the two detectors. In the BSR random number generator, it is virtually impossible to eliminate bias by shear manufacturing precision of the components to a level that would not be (easily) detectable by statistical tests. Bias itself can be strongly reduced or eliminated altogether by post-processing techniques like Von Neumann scheme [12] but exact de-biasing works only if there are no correlations among bits.

Unlike bias, which can be expressed by a single number, correlations among bits can be arbitrarily complex and generally cannot be expressed by a single quantity. Nevertheless the description can be simplified by realizing that memory effects in detectors, that cause correlations among bits, are strongly localized in time [13]. If the mean period between detections is long enough, only neighboring bits are non-negligibly correlated. Under that assumption, correlation among bits can be well characterized by only the serial autocorrelation coefficient with lag 1. Serial autocorrelation coefficients are defined as (see Ref [14].):

$$a_k = \frac{\sum_{i=1}^{N-k} (x_i - \bar{x})(x_{i+k} - \bar{x})}{\sum_{i=1}^{N-k} (x_i - \bar{x})^2} \tag{3}$$

where a_k is a serial autocorrelation coefficient with lag k and N is number of bits in the sequence. For finite N, an estimate given by Eq. (3) has 1 sigma Gaussian statistical error of $1/\sqrt{N-k}$.

In the BSR method, correlations among bits are caused mainly by dead time and afterpulsing in detectors. To see that, let us a suppose that faint continuous Poisson random light, such as for example generated by an LED or well saturated laser, is shone upon the beam splitter causing photons to be detected by each detector with a mean detection period τ. Let us further suppose that the two detectors have identical dead time τ_d. In case that the next photon "arrives" in less than τ_d after the previous one, it can either hit the detector that is in the dead state or would get detected by the other detector thus contributing to the negative autocorrelation of the sequence of generated bits. The dead time reduces the probability of successive detections by the same detector, that is that the generated bit string misses some of the substrings "00" and "11", at random places, with respect to the string that would have been obtained from the same source of photons should the dead time be zero. Since omission of longer same-digit sequences is much les probable, the leading effect is relative excess of substrings "01" and "10" observed as a negative value of the autocorrelation coefficient a_1, henceforth denoted a and referred to as "autocorrelation". Its magnitude is approximately equal to probability p_d that a photon "falls" into the dead time:

$$a = -p_d = e^{-\tau_d/\tau} - 1 \approx -\frac{\tau_d}{\tau} \tag{4}$$

where the approximation is valid for $\tau_d << \tau$. In our case, the mean photon frequency could be 10 MHz (i.e. τ = 100 ns) and dead time τ_d = 24 ns leading to a ≈-0.24: an imperfection statistically detectable with only 17 generated bits! Even with an unrealistically short dead time, at a sufficiently high detection (i.e. bit production) rate, the autocorrelation becomes un-

#231845 - $15.00 USD Received 5 Jan 2015; revised 8 Mar 2015; accepted 24 Mar 2015; published 24 Apr 2015

tolerably high. The highest counting rate of the detector is $1/\tau_d$ and in that limit a_1 is approaches -1. To achieve lower correlation one needs to operate at a lower detection frequency, but then only a small portion of detector capabilities can be exploited for random number generation. In our previous work [13] we showed that correlation does not vanish in the low detection frequency end either: instead, it asymptotically reaches the value of the visible afterpulsing probability p_d which in our case is about 0.031.

For completeness we note that the beam splitter method can also be performed with a periodically pulsed light source, such as described in [5]. In that case a photon detection time is precisely known and a narrow gate around it can be used to strongly reduce afterpulsing and dark counts noises. Furthermore, dead time effects can be completely avoided simply by choosing pulse repetition period longer than the dead time. However, such a system does not posses time arrival randomness that we use in our COMBO method described in the Section 4.

In [13] we have already noted that positive and negative autocorrelation mechanisms in the BSR method cancel each other at a certain detection rate. Namely, following each photon detection there is a constant probability (on the order of a percent) that meta-stable deep states lying between Fermi and conductive band of the solid-state avalanche photodiode get filled. Each of these states decays randomly with a characteristic lifetime. In our case, lifetimes are short with respect to the dead time and therefore most of the afterpulses will contribute to generating substrings of "00" or "11" at random occasions times. Since the two processes are independent of each other, the resulting net correlation a is a sum of the two contributions:

$$a = p_a - \frac{\tau_d}{\tau}. \tag{5}$$

and vanishes for a detection rate $f_0 = 1/\tau = p_a/\tau_d$. For our detectors having $p_a = 0.031$ and $\tau_d = 24$ ns the vanishing point is at $f_0 \approx 1.3$ Mcps, which is quite low with respect to their counting capability of over 25 Mcps.

In this work we go further by noting an effect that, in principle, enables total elimination of the autocorrelation. First we note that by ignoring detections that came from one detector in less than a prescribed *blanking time* Δt after detection by the other detector, helps to reduce correlations significantly. Specifically, in order to completely remove the anti-correlation caused by the dead time, it would be enough to take the blanking time (Δt) any higher than the dead time because then the dead time would have no effect on photons selected for generation of random numbers. However, positive autocorrelation due to afterpulses appearing after Δt would be left over causing a small positive autocorrelation. In order to overcome that, we note that by taking a suitable Δt, slightly smaller than the dead time, one can obtain, in principle, an arbitrarily high detection frequency f_0 at which the two competing effects (negative and positive) cancel each other. Note that below f_0 the total autocorrelation is positive due to dominant effect of the afterpulsing, whereas above f_0 autocorrelation turns negative due to dominant effect of the dead time. We have therefore implemented a circuit blanking out (omitting) any event(s) that come sooner than Δt after the previous one. All electronics circuits in this study have been realized by Altera MAX3000 family CPLD chip. Due to the discrete delay times available in the CPLD chip, duration of the blanking time Δt is adjustable in the range from 5.6 ns in steps of 4.0 ns, while dead times of the two detectors D0 and D1 used in this work are fixed to 24 ns.

In our analysis we assume that higher-lag autocorrelation coefficients $a_k, k > 1$ are much smaller than a_1. We find that a detection rate of 10 Mcps is a good trade-off between a high bit production rate and appearance of longer-lag correlations due to afterpulsing. Fixing the

#231845 - $15.00 USD Received 5 Jan 2015; revised 8 Mar 2015; accepted 24 Mar 2015; published 24 Apr 2015

detection rate to 10 Mbps and varying Δt we find that the lowest autocorrelation is obtained for $\Delta t = 17.6$ ns and is then equal to $(280 \pm 32) \cdot 10^{-6}$ as evaluated by statistics of 10^9 bits. All higher lag coefficients are consistent with zero within the statistical error.

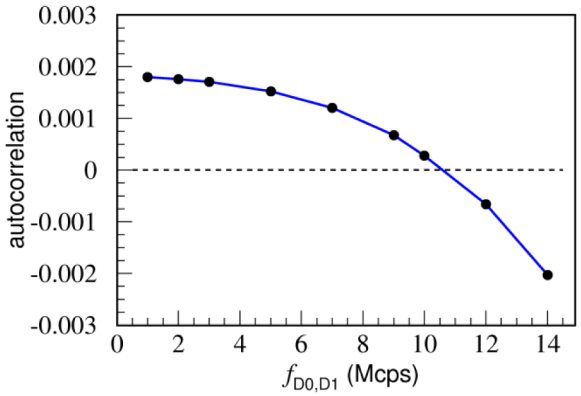

Fig. 2. Autocorrelation coefficient for the BSR method a as a function of the detection frequency f of detectors D0 and D1, for the blanking time $\Delta t = 17.6$ ns. One sigma statistical error bars are smaller than the dots sizes.

Fixing Δt to 17.6 ns the autocorrelation as a function of detection rate has been evaluated according to Eq. (3), with statistics of $N = 10^9$ bits. The result shown in Fig. 2. confirms the expected sign change near 10 Mcps.

3. Photon pair waiting times difference method (T1T2)

The biggest challenge in realization of a good quantum (or any other physical) RNG is that it is difficult to realize a setup close to the theoretical idealization, especially if there is anything that must be adjusted prior to use. For example to adjust bias in BSR methods to within $1/N$ one must generate at least $\sim N^2$ bits to test the adjustment and is therefore faced with an insurmountable time consuming task. Even if sufficiently fine adjustment could be done, there is a question of whether it would stay stable over time withstanding temperature, power supply and other variations as well as aging and wear of components.

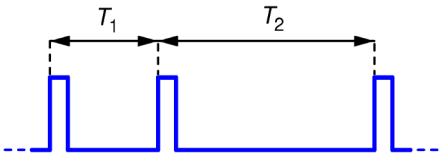

Fig. 3. Illustration of the T1T2 method. Three subsequent random events (detected photons) define two intervals: T_1 and T_2. If $T_1 > T_2$ then 0 is generated, if $T_2 > T_1$ then 1 is generated whereas if $T_1 = T_2$ then events are skipped and no random bit is generated.

Therefore, especially valuable are random number generating methods that do not require any adjustments. One such method has been proposed in [9]. A time-wise random stream of electrical (logic) pulses is obtained from a random event generator (REG) which generates electrical pulses whose waiting-times obey an exponential probability density function (p.d.f). Three consecutive pulses from the REG, as shown in Fig. 3, are used to define time intervals T_1 (between the first two) and T_2 (between second and third). A random bit is then generated by comparing the two intervals: if $T_1 > T_2$ then value 0 is generated; if $T_2 > T_1$ then value 1 is generated; if $T_1 = T_2$ (within the measurement precision) then no bit is generated. In order to

#231845 - $15.00 USD Received 5 Jan 2015; revised 8 Mar 2015; accepted 24 Mar 2015; published 24 Apr 2015

maximize the bit efficiency, the third event is taken as the first event of the next triplet, thus two events are spent to generate one random bit. If the REG source is stationary and memoryless, which is indicated by an exponential time interval distribution, the bits will be uncorrelated and, due to exchangeability of definitions of 0 and 1, the bias will be zero.

The crucial insight achieved in [9] is that clocks which measure photon time intervals must be started in synchronization with beginning of each interval, otherwise the method would produce correlated bits even if fed by perfectly random events. This was not understood in previous art, like for example in [7] where the clock was free-running which must have yielded correlated bits, but it was not noted probably because clock frequency (~10MHz) was much higher than the source mean frequency (~10kHz) and in that case correlations are small. Correlations rise quickly as the ratio of frequencies of clock and source of random events becomes smaller [7]. Afterpulsing in detectors causes some correlation, but much smaller than in BSR method because they appear at random times and merely slightly modify the exponential waiting-time distribution.

4. Combined spatio temporal method with improved randomness (COMBO)

The two above described random number generating methods offer a range of randomness quality, bits-per-photon efficiency and resilience to hardware imperfections. However, at the present level of technology, even with the most optimal method, the leftover randomness imperfections are typically larger than what is acceptable by general applications. As illustrated there is always some hardware detail that limits the randomness and which cannot be further improved with a given level of hardware technology.

In order to arrive to a better randomness of generated bits, one can use deterministic postprocessing (extractor algorithms [15,16], resilient functions [17,18]) or non-deterministic postprocessing (i.e. using additional sources of entropy). Postprocessing comes with a price of additional hardware and/or software resources. In order to simplify postprocessing or avoid it altogether, it is therefore legitimate to ask whether a better bit extraction method can be construed that would both feature lower sensitivity to hardware imperfections and provability of randomness of the extracted bits.

To that end we proceed by noting that in the BSR method (explained in the previous section), there are two independent random processes, each of which allows for random bit extraction. Namely, the time *when* a photon is detected (by either detector) is, at least in theory, completely independent of *where* it is detected (by which detector). Therefore, one can exploit *temporal* information of a train of photons detected by either detector D0 or D1 to generate a sequence of random bits via T1T2 method, and at the same time use *spatial* information of the same detected photons to generate an independent sequence of random bits via BSR method. This method we name "COMBO" as it combines the two random number generating principles. It is to be contrasted to other methods that combine two independent random strings obtained from two independent set of measurements, such as [24], because in COMBO we obtain two entirely independent random bit strings by processing the same set of measurements performed simultaneously on an optical quantum system.

The optical part of the COMBO RNG is shown in Fig. 1. Variable filter VNDF makes possible fine adjustment of the bias b_S of the BSR section to (0 ± 0.001) with stability better than ± 0.0005 during experiments.

Photon detections from the two detectors are combined by the electrical circuit shown in Fig. 4, which functions as follows. In order to extract the timing information, pulse trains from the two detectors are interleaved by a summing circuit to form a single train of pulses which is then processed according to the T1T2 method. This yields one string of random bits, let us denote it with $\boldsymbol{T}$ for "time". At the same time, the two pulse trains are processed according to the BSR method explained above. This yields the other string of random bits related to the space information, let us denote it with $\boldsymbol{S}$ for "space". Note that the very same

#231845 - $15.00 USD Received 5 Jan 2015; revised 8 Mar 2015; accepted 24 Mar 2015; published 24 Apr 2015

photo-detection signals are used to generate both strings $\boldsymbol{T}$ and $\boldsymbol{S}$. Each of these two independent bit strings will have its own bias and autocorrelation, let us denote them as (b_T, a_T) and (b_S, a_S) respectively.

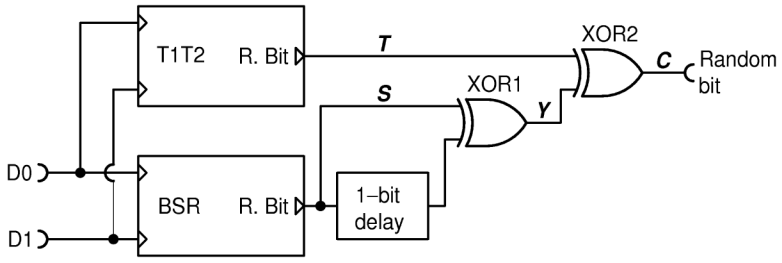

Fig. 4. Functional schematic of the circuit for realizing the COMBO random number generating method illustrating how intermediate strings $\boldsymbol{T}$, $\boldsymbol{S}$, $\boldsymbol{Y}$ and final string $\boldsymbol{C}$ are generated.

Since, in principle, the BSR method produces one bit per detected photon and T1T2 produces one bit per two detected photons, we need to decide how to combine these two bit streams. One way is to omit every second BSR bit: this would result in a string with much lower (squared) auto correlation but would not affect the bias. In order to improve both on bias and correlation, the COMBO method uses XOR of non-overlapping pairs of consecutive bits from the BSR method yielding new string $\boldsymbol{Y}(b_Y, a_Y)$. This is accomplished by the D-type flip-flop and XOR gate XOR1 shown in the Fig. 4. Finally the two random strings $\boldsymbol{T}$ and $\boldsymbol{Y}$ are XOR-ed (by the gate XOR2) to yield the final string $\boldsymbol{C}(b_C, a_C)$. In order to calculate estimates of b_C and a_C we use two assumptions discussed above: (1) both BSR and T1T2 methods suffer only from short-lag correlations; (2) random strings generated by two methods are independent of each other. Both assumptions are checked by the data themselves. Note that the T1T2 section is only fed by events passed through the blanking filter of the BSR, even though blanking is not necessary for that section, in order to prevent too often clocking of the BSR section that would induce positive correlation in the string $\boldsymbol{S}$.

For generation of random numbers, the power of the CW source is adjusted such that each detector counts with an average frequency of (10.00 ± 0.005) Mcps, in total 20 million random events per second. In principle this would result in random bit generation rate of 10 Mbps since the bit production is clocked by T1T2 section which uses 2 detections per bit, however due to the blanking the bit rate is approximately 8.0 Mbit/sec.

In order to estimate quality of randomness at the output (sequence $\boldsymbol{C}$) sequences $\boldsymbol{S}$ and $\boldsymbol{T}$ were generated the circuit shown in Fig. 4. Statistics of $2 \cdot 10^9$ bits for string $\boldsymbol{S}$ and $1 \cdot 10^9$ for string $\boldsymbol{T}$ were collected. Statistical analysis of the recorded random bits according to Eqs. (2) and (3) yielded the following estimates: $\hat{b}_S = (227 \pm 16) \cdot 10^{-6}$, $\hat{a}_S = (-149 \pm 32) \cdot 10^{-6}$ and $\hat{b}_T = (-125 \pm 16) \cdot 10^{-6}$, $\hat{a}_T = (48 \pm 32) \cdot 10^{-6}$. The error bars are plus minus 1 Gaussian sigma, estimated as explained in the Section 2. We find that second and further autocorrelation coefficients ($k \geq 2$) for either of the two strings are consistent with zero within statistical uncertainty. This is expected since the system has no intentional memory and the correlation among bits dies off very quickly with the lag. Achieved randomness merits (bias and correlation) of strings $\boldsymbol{S}$ and $\boldsymbol{T}$ are not good enough for general applications, consequently we need some kind of randomness extraction. One could, in principle, estimate min-entropy of the strings following the approach in [25], after which an efficient universal hashing extractor would be applied to the raw bits. In this work however, we use a chained XOR extractor, depicted in Fig. 4, which is much less efficient in terms of number of extracted bits

#231845 - $15.00 USD Received 5 Jan 2015; revised 8 Mar 2015; accepted 24 Mar 2015; published 24 Apr 2015

versus number of input bits, but also much simpler to realize in hardware. In order to estimate randomness merits of the COMBO RNG we start by considering the string $\boldsymbol{Y}$, derived from string $\boldsymbol{S}$, for which the following relation holds [19]:

$$b_Y = -2{b_S}^2 - 2a_S\left(\frac{1}{4} - {b_S}^2\right) \approx -2{b_S}^2 - \frac{a_S}{2} \tag{6}$$

$$a_Y = 2\frac{a_S\left(1-a_S\right){b_S}^2}{1-2\left(1-a_S\right)\left(\frac{1}{4}-{b_S}^2\right)} \approx 4a_S{b_S}^2 \tag{7}$$

where the approximations are valid when $a_S \ll 1$ and ${b_S}^2 \ll 0.25$ which is satisfied in our case. This yields: $\hat{b}_Y = (75 \pm 16)\cdot 10^{-6}$, $\hat{a}_Y = (0 \pm 21)\cdot 10^{-6}$. Again, higher lag correlations are consistent with zero within statistical error margins.

Bit-by-bit XOR-ing of two independent sequences $\boldsymbol{T}$ and $\boldsymbol{Y}$ yields the final sequence $\boldsymbol{C}$ with bias and correlation given by [11]:

$$b_C = -2b_T b_Y \tag{8}$$

$$a_C = a_T a_Y + 4\left(a_T b_Y^2 + a_Y b_T^2\right) \tag{9}$$

The above relations may give an over-optimistic result if the two sequences are correlated, that is if normalized cross-correlation coefficients defined as:

$$a_{TY}(k) = \frac{\sum_{i=1}^{N-k}\left(t_i - \overline{t}\right)\left(y_{i+k} - \overline{y}\right)}{\sqrt{\left(\sum_{i=1}^{N-k}\left(t_i - \overline{t}\right)^2\right)\left(\sum_{i=1}^{N-k}\left(y_i - \overline{y}\right)^2\right)}} \tag{10}$$

are non-zero for some k. Since the memory effects in the system (dead time and afterpulsing) are shorter than the average period of generated bits (100 ns) it is enough to check a few values of k around 0. To that end strings $\boldsymbol{T}$ and $\boldsymbol{Y}$, each $3\cdot 10^9$ bits long, were generated simultaneously and cross-correlation coefficients for lag k in the range $[-6,6]$ estimated according to Eq. (10). Obtained values shown in the Fig. 5. along with 1 sigma Gaussian error bars (evaluated as $1/\sqrt{N-k}$) are consistent with zero within statistical errors thus confirming the non-correlation hypothesis.

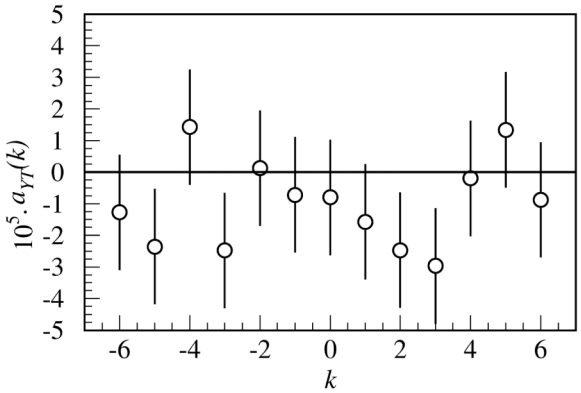

Fig. 5. Cross-correlation coefficients of spatial and temporal bit strings in the COMBO quantum random number generator.

#231845 - $15.00 USD Received 5 Jan 2015; revised 8 Mar 2015; accepted 24 Mar 2015; published 24 Apr 2015

Finally, we obtain $\hat{b}_C = (7.1 \pm 1.6) \cdot 10^{-8}$, $\hat{a}_C = (3.7 \pm 1.2) \cdot 10^{-12}$. In order to measure errors of that magnitude with 95% C.L. one would need to generate a string of at least $6 \cdot 10^{13}$ bits. Thus we prove that any generated string shorter than that is indistinguishable from a true random string and in that case there is no need to do any further tests of the generated output. This is quite different from established state of the art where randomness is only tested for strings of typically only 10^9 bits or shorter [24]. Nevertheless, as a cross-check, we also perform "standard" statistical analysis of several sequences of 10^9 bits by means of the NIST statistical test suite (STS, version 2.2.1) [20] using the default test parameters. Result obtained for a typical sequence is shown in Table 1.

Table 1. Typical results of the NIST Statistical Test Suite

Statistical test	p-value	Proportion/Threshold	Pass
Frequency	0.75186	991/980	Yes
Block frequency	0.82372	991/980	Yes
Cumulative sums	0.63859	992/980	Yes
Runs	0.67868	991/980	Yes
LongestRun	0.48464	982/980	Yes
Rank	0.35864	994/980	Yes
FFT	0.95120	992/980	Yes
NonOverlappingTemplate	0.44015	990/980	Yes
OverlappingTemplate	0.18454	992/980	Yes
Universal	0.97305	989/980	Yes
ApproximateEntropy	0.56463	992/980	Yes
RandomExcursions	0.48645	628/622	Yes
RandomExcursionsVariant	0.44883	628/622	Yes
Serial	0.47577	989/980	Yes
LinearComplexity	0.72582	989/980	Yes

5. Study of resilience against hardware failure and signal injection attacks

When a RNG is used in a critical application (e.g. for cryptographic security) it is important that it possess a resilience to common attacks and most probable hardware failure scenarios while allowing for robust monitoring of its proper functioning.

For COMBO RNG, in case that one of the detectors (D0, D1) fails completely (i.e. stops generating pulses), the BSR extraction method will be either stuck to logic 1 or logic 0, depending on whether D0 or D1 failed, respectively. However the T1T2 section will still function normally albeit at only a half rate. As a result, randomness of the output will be slightly reduced (to that of the T1T2 section) while the bit production rate will be halved. In case of partial reduction of detection rate of one of detectors, BSR section will generate a heavily biased sequence. In that case, according to Eqs. (8) and (9), randomness of output bits should be better than of the T1T2 method alone. This is indeed confirmed by an experiment in which average detection frequency of D0 was kept fixed at (10.00 ± 0.05) Mcps while that of D1 (f_{D1}) was varied from zero to 10 Mcps in order to simulate its full or partial failure. Results plotted in Fig. 6. show that, due to rate mismatch between D0 and D1, the BSR section generates highly biased and correlated output which improves as f_{D1} approaches f_{D0}. The bias b_Y is out of the scope for all points except for $f_{\mathrm{D1}} = 10$ MHz while autocorrelation a_T is shown. Even so, randomness merits of the combined output, namely a_C and b_C are improved with respect to those of the T1T2 section (a_T, b_T) except for $f_{\mathrm{D1}} = 0$ where the two are equal as expected. We further note that if both detectors would fail completely, the T1T2 section would not generate any Strobe signals and there would be no output. Therefore, the COMBO RNG is robust against detector(s) failure in the sense that as long as there is any output, its randomness quality is good.

#231845 - $15.00 USD Received 5 Jan 2015; revised 8 Mar 2015; accepted 24 Mar 2015; published 24 Apr 2015

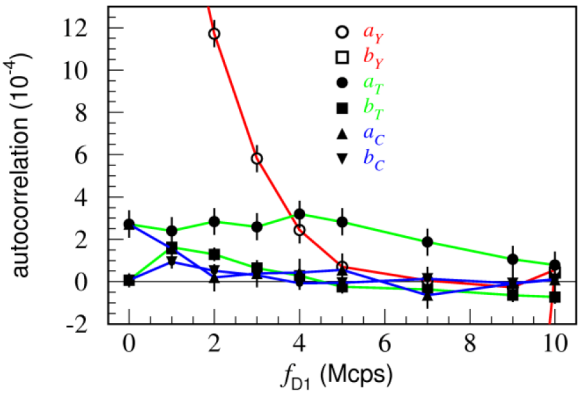

Fig. 6. Randomness merits (bias and autocorrelation) of the three sequences generated by the COMBO RNG in case when one detector is failing. Detection rate of one detector (D0) was kept at its nominal rate of 10 Mcps, while rate of the other detector (D1) was varied from 10 Mcps down to zero simulating its failure. Shown are ± 1 Gaussian sigma statistical error bars.

Signal injection attacks on RNGs are well known in the cryptographic art [21,22]. The danger stems from the fact that they can be mounted without destruction or tampering with a RNG thus leaving no physical evidence, for example via electromagnetic induction or through power supply lines. A small coil near the gadget that contains RNG, such as smart card or contactless payment card can render it unsecure. Ref. 21 demonstrates a practical attack on a RNG in an ATM machine via injection-locking which would enable unauthorized use of a stolen credit card.

Apart from security there is also a functional issue. In [23] many RNGs operate in parallel on the same chip in order to achieve a higher speed one, however their electrical interference produces such a strong mutual coupling that individual RNGs tend to synchronize and lower the total entropy to much below the expected value. Sensitivity to interference signals is especially important in mobile phones where a strong signal from the RF transmitter cannot be shielded enough by any measure (such as a Faraday cage etc.) to the level below weak (quantum) signals from which randomness is extracted.

Let us now consider the COMBO RNG under attack by injection of a periodic signal which causes fake photon detections or "injection events" in detectors. We assume that the injected pulses affects equally and simultaneously both detectors. This would in particular be the case for a RNG miniaturized to a chip level with the closely spaced detectors of nearly identical characteristics. We simulate the attack by mixing the CW light with strong laser pulses (PicoQuant PDL 800-D with laser head $\lambda = 676$ nm, 39 ps FWHM) via an additional beam splitter in front of the of the RNG setup (Fig. 1). Detection probability of a laser pulse is greater than 99.7%. In an experiment the intensity of CW light was set such that $f_{D0} = f_{D1} = 10.00 \pm 0.05$ Mcps while the periodic frequency of injected laser pulses (f_{Inject}) was varied from zero to 7 MHz. Results are shown in Fig. 7. The attack generates a deviation from otherwise exponential time interval distribution of detected events (consisting of intertwined detected photons and injection events). Since BSR section is not sensitive to the time information it is quite insensitive to the attack. We see that the randomness merits of the output string $\boldsymbol{C}$ stay zero within statistical errors for 10^9 bits up to 7 MHz. However by examining component strings $\boldsymbol{Y}$ and $\boldsymbol{T}$ which are much more sensitive to errors, we see that randomness does deteriorate with f_{Inject}. Namely, under deterministic injected signal from the detectors, BSR section is bound to generate a deterministic output. In our implementation of the BSR section, a bit is generated whenever a photon is detected by either detector and its value is identified by a state of the detector D1 (except for the blanking described above). Thus, if photon is detected by D0, state of D1 will be 0 (no detection) while if photon is

#231845 - $15.00 USD Received 5 Jan 2015; revised 8 Mar 2015; accepted 24 Mar 2015; published 24 Apr 2015

detected by D1 state of D1 is 1. This is correct for a stream of photons from the CW source, but when a signal is injected such that both detectors "fire" at the same time (within the 2 ns time resolution window of the circuit) the circuit will generate the value of "1". Thus, every injected signal generates a value of "1" in the string $\boldsymbol{S}$ driving its bias towards positive value, and therefore, according to Eq. (6), bias of the string $\boldsymbol{Y}$ will turn towards negative. This effect becomes apparent for $f_{\text{Inject}} > 4.5$ MHz, as shown in Fig. 7. The T1T2 section is by definition more sensitive to the time interval distribution and that is clearly visible through increasingly negative bias and sharp rise of autocorrelation for $f_{\text{Inject}} > 3$ MHz.

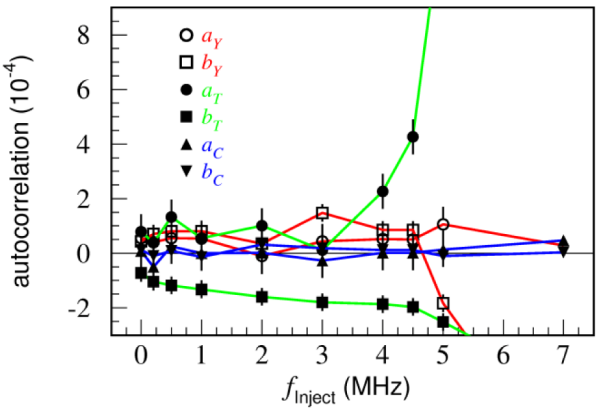

Fig. 7. Randomness merits (bias and autocorrelation) of the three sequences $\boldsymbol{Y}, \boldsymbol{T}$ and $\boldsymbol{C}$ generated by the COMBO RNG under attack by injection of a periodic signal which causes simultaneous fake photon detections in detector with a frequency f_{Inject}. Shown are ± 1 Gaussian sigma statistical error bars.

In general, favorable interplay of the two sections (BSR and T1T2) makes the COMBO RNG surprisingly robust against detector failure and signal injection attacks. However, some deterioration does appear in case of a strong attack conditions or a bad failure of detector(s). For practical applications it would therefore be important that the failure and attacks can be detected at levels that are far from causing any noticeable randomness deterioration. We studied two simple measures: bit generation rate (f_G) and blanked events rate (f_B) Blanked events rate is the rate of events that are discarded by the blanking procedure and therefore are *not* used for the random number generation. In Fig. 8 f_G (shown by quadratic markers) and f_B (shown by round markers) are plotted as a function of the degree of failure of detectors for three scenarios studied above: (*i*) D0 and D1 failing simultaneously; (*ii*) only D0 failing; and (*iii*) signal injection attack. For the first scenario (dotted line) f_D is detection rate of both detectors; for the second (dashed line) f_D is the detection rate of detector D0; and for the third (full line) f_D is the frequency of injected laser pulses.

#231845 - $15.00 USD Received 5 Jan 2015; revised 8 Mar 2015; accepted 24 Mar 2015; published 24 Apr 2015

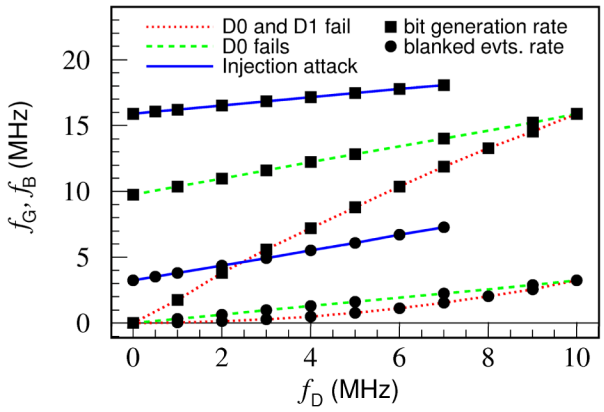

Fig. 8. Bit generation rate f_G (quadratic dots) and blanked events rate f_B (round dots) as functions of detection rates in three failure and attack scenarios (see the text). Both rates are sensitive to irregular operation of the RNG this allowing for robust monitoring and failure/attack detection.

We see that both generation (f_G) and blanked events (f_B) rates are quite sensitive to irregular operation of the RNG with a small advantage of f_G for scenario (*i*) and f_B for scenario (*iii*), whereas either is an equally good measure for scenario (*ii*). In the failure scenarios (*i*) and (*ii*) both parameters (f_G, f_B) drop below normal values whereas in the attack scenario (*iii*) both measures rise above their normal values. We thus have two measures, either of which (but preferably both) can be used to robustly alert malfunction of the COMBO RNG even at levels at which randomness is not noticeably compromised.

6. Conclusion

A RNG based on quantum effects in photonic emission and detection is presented. A mathematical framework for estimation of randomness quality based on simple measurements has been developed. The RNG is unique in several aspects: (1) the method of extraction of random bits simultaneously uses both spatial and temporal quantum information contained in the system; (2) the RNG is robust against hardware failure and signal injection attack in the sense that up to some threshold levels of detector failure or attack frequency randomness is virtually intact; (3) malfunction of the RNG, due to detector failure or signal injection, can be robustly detected at levels at which randomness is not significantly affected thus enabling protection of integrity of generated random bits. It is also shown that generated numbers pass the NIST Statistical Test Suite (STS) without the need for any further post-processing. Having in mind that partial detector failure scenario is identical to an initial or aging-related difference between detectors, we conclude that COMBO RBG is also robust to initial components variation and aging, which makes it a good candidate for mass-production or chip-level QRNG technology.

Acknowledgments

This work was supported by Fulbright program (academic year 2010/2011), Ministry of Science Education and Sports of Republic of Croatia (contract Nr. 098-0352851-2873). Extensive numerical simulations and randomness tests were made at Center for Scientific Computing UC Santa Barbara (contract Nr. NSF-CNS-0960316) and University Computing Centre (SRCE) of University of Zagreb.

#231845 - $15.00 USD Received 5 Jan 2015; revised 8 Mar 2015; accepted 24 Mar 2015; published 24 Apr 2015