

**COMPACT QUANTUM RANDOM NUMBER GENERATOR BASED ON BALANCED
DETECTION OF SHOT NOISE IN AN OPTOCOUPLER**

CROSS-REFERENCE TO RELATED APPLICATION

[0001] This application claims priority to U.S. Provisional Patent Application No. 63/517,045, filed August 1, 2023, which is incorporated by reference in its entirety.

FIELD

[0002] A system and method for compact Quantum Random Number Generator (QRNG) based on balanced detection of shot noise in an optocoupler.

BACKGROUND

[0003] Ideal random number generators generate unbiased unpredictable numbers that lack any pattern. One such solution is a QRNG that utilizes fundamentally probabilistic processes in quantum mechanics to generate numbers which cannot be predicted. Existing QRNG solutions sometimes use lasers or multipixel photosensor as detectors, or both. These solutions are bulky, expensive, require specialized electronics and accurate alignment to operate properly.

SUMMARY

[0004] In one aspect, the present disclosure relates to a QRNG system including an optocoupler including at least one light emitter and at least two light receivers that are electrically coupled to each other in a configuration that rejects common mode noise. In embodiments, the system also can include a controller configured to control the at least one light emitter to emit photons randomly towards the at least two light receivers, measure a shot noise signal generated by the at least two light receivers in response to receiving the photons, and extract a random number from the shot noise signal.

[0005] In embodiments of this aspect, the disclosed system according to any one of the above example embodiments, wherein the at least one light emitter is a light emitting diode (LED), and at least two light receivers are photodiodes.

[0006] In embodiments of this aspect, the disclosed system according to any one of the above example embodiments, wherein the shot noise is measured from a common node where the at least two light receivers are electrically coupled to each other.

[0007] In embodiments of this aspect, the disclosed system according to any one of the above example embodiments can include an amplifier for amplifying the shot noise signal generated by the at least

two light receivers, and a digitizer for digitizing the amplified shot noise signal, wherein the controller can be configured to extract the random number from the digitized shot noise signal.

[0008] In embodiments of this aspect, the disclosed system according to any one of the above example embodiments, the controller can be configured to extract the random number from the digitized shot noise signal by modifying a probability distribution of the digitized shot noise signal, and extracting the random number from the modified probability distribution.

[0009] In embodiments of this aspect, the disclosed system according to any one of the above example embodiments, the controller can be configured to modify the probability distribution and extract the random number from the modified probability distribution by applying a randomness extractor to the digitized shot noise signal.

[0010] In embodiments of this aspect, the disclosed system according to any one of the above example embodiments, the controller can be configured to modify the probability distribution by modifying (i.e., converting) the probability distribution due to the digitized shot noise signal into a Uniform probability distribution for use in a software application.

[0011] In embodiments of this aspect, the disclosed system according to any one of the above example embodiments, wherein the digitized shot noise signal has a Uniform probability distribution.

[0012] In embodiments of this aspect, the disclosed system according to any one of the above example embodiments, can include a quantum random number utilizer (QRNU) device coupled to the QRNG. The QRNU device may execute a software application that utilizes the extracted random number.

[0013] In embodiments of this aspect, the disclosed system according to any one of the above example embodiments, the controller can be configured to control a size of the random number extracted from the shot noise based on a software application that utilizes the extracted random number.

[0014] In one aspect, the present disclosure relates to a quantum random number generator (QRNG) method includes controlling, by a controller, at least one light emitter to emit photons towards at least two light receivers that are electrically coupled to each other in a configuration that rejects common mode noise, measuring, by the controller, a shot noise signal generated by the at least two light receivers in response to receiving the photons, and extracting, by the controller, a random number from the shot noise signal.

[0015] In embodiments of this aspect, the disclosed method according to any one of the above example embodiments includes controlling, by the controller, a light emitting diode (LED) as the at least one light emitter, and measuring, by the controller, the shot noise signal from photodiodes as the least two light receivers.

[0016] In embodiments of this aspect, the disclosed method according to any one of the above example embodiments includes measuring, by the controller, the shot noise signal from a common node where the at least two light receivers are electrically coupled to each other.

[0017] In embodiments of this aspect, the disclosed method according to any one of the above example embodiments includes amplifying, by an amplifier, the shot noise signal generated by the at least two light receivers, digitizing, by a digitizer, the amplified shot noise signal, and extracting, by the controller, the random number from the digitized shot noise signal.

[0018] In embodiments of this aspect, the disclosed method according to any one of the above example embodiments includes extracting, by the controller, the random number from the digitized shot noise signal by modifying a probability distribution of the digitized shot noise signal, and extracting the random number from the modified probability distribution.

[0019] In embodiments of this aspect, the disclosed method according to any one of the above example embodiments includes modifying, by the controller, the probability distribution, and extracting the random number from the modified probability distribution by applying a randomness extractor to the digitized shot noise signal.

[0020] In embodiments of this aspect, the disclosed method according to any one of the above example embodiments includes modifying, by the controller, the probability distribution by converting the probability distribution due to the digitized shot noise signal into a Uniform distribution for use in a software application.

[0021] In embodiments of this aspect, the disclosed method according to any one of the above example embodiments includes outputting, by the controller, the digitized shot noise signal having a Uniform probability distribution.

[0022] In embodiments of this aspect, the disclosed method according to any one of the above example embodiments includes executing, by a quantum random number utilizer (QRNU) device coupled to the QRNG, a software application that utilizes the extracted random number.

[0023] In embodiments of this aspect, the disclosed method according to any one of the above example embodiments includes controlling, by the controller, a size of the random number extracted from the shot noise signal based on a software application that utilizes the extracted random number.

BRIEF DESCRIPTION OF THE DRAWINGS

[0024] So that the way the above-recited features of the present disclosure can be understood in detail, a more particular description of the disclosure, briefly summarized above, may be made by reference to example embodiments, some of which are illustrated in the appended drawings. It is to be noted, however, that the appended drawings illustrate only example embodiments of this disclosure and are therefore not to be considered limiting of its scope, for the disclosure may admit to other equally effective example embodiments.

[0025] FIG. 1 shows a block diagram of a QRNG circuit, according to an example embodiment of the present disclosure.

[0026] FIG. 2 shows a circuit diagram of an opto-coupler of the QRNG circuit, according to an example embodiment of the present disclosure.

[0027] FIG. 3 shows a circuit diagram of an amplifier of the QRNG circuit, according to an example embodiment of the present disclosure.

[0028] FIG. 4 shows a flowchart of a randomness extractor of the QRNG circuit, according to an example embodiment of the present disclosure.

[0029] FIG. 5 shows a block diagram of a controller of the QRNG circuit, according to an example embodiment of the present disclosure.

[0030] FIG. 6 shows a flowchart of operation of the QRNG circuit, according to an example embodiment of the present disclosure.

[0031] FIG. 7A shows an experimental setup of the optocoupler connected to a microcontroller for ADC extraction, according to an example embodiment of the present disclosure.

[0032] FIG. 7B shows a histogram of the QRNG observed by the experimental setup, according to an example embodiment of the present disclosure.

[0033] FIG. 7C shows the variance of the output noise signal observed by the experimental setup, according to an example embodiment of the present disclosure.

[0034] FIG. 7D shows the autocorrelation coefficient of a sequence of bits for observed by the experimental setup, according to an example embodiment of the present disclosure.

[0035] FIG. 7E shows a plot of the p-values from the output of a dieharder test run observed by the experimental setup, according to an example embodiment of the present disclosure.

DETAILED DESCRIPTION

[0036] Various example embodiments of the present disclosure will now be described in detail with reference to the drawings. It should be noted that the relative arrangement of the components and steps, the numerical expressions, and the numerical values set forth in these example embodiments do not limit the scope of the present disclosure unless it is specifically stated otherwise. The following description of at least one example embodiment is merely illustrative in nature and is in no way intended to limit the disclosure, its application, or its uses. Techniques, methods, and apparatus as known by one of ordinary skill in the relevant art may not be discussed in detail but are intended to be part of the specification where appropriate. In all the examples illustrated and discussed herein, any specific values should be interpreted to be illustrative and non-limiting. Thus, other example embodiments may have different values. Notice that similar reference numerals and letters refer to similar items in the following figures, and thus once an item is defined in one figure, it is possible that it need not be further discussed

for the following figures. Below, the example embodiments will be described with reference to the accompanying figures.

[0037] The disclosed methods, devices and systems herein overcome the limitations of the existing systems by generating random numbers from a stochastic process based on utilizing shot noise of a simplistic light source such as a light emitting diode (LED) source for use in resource constrained applications. The disclosed methods, devices and systems detect (i.e., measure) the shot noise of an LED via a balanced photo detector (e.g., photodiode) arrangement that eliminates (i.e., rejects) classical common mode noise (e.g., power supply noise, etc.) detected by the photo detectors. In embodiments, the disclosed methods, devices and systems include an optoelectronic system including a power source, linear balanced optocoupler, and amplifiers as the entropy source, and at least one entropy processor including a digitizer and a randomness extractor generating nonbiased random numbers with full entropy at high speed. The disclosed methods, devices and systems operate as an efficient and cost effective QRNG to produce unbiased unpredictable numbers that lack any pattern due to the fundamentally probabilistic processes in quantum mechanics.

[0038] For photodiode of quantum efficiency η and photon flux Φ , the photocurrent is noted as shown in equation (1)

$$i = \eta e \phi = \eta e \frac{P}{\hbar \omega} \quad (1)$$

where e is the charge of an electron, P is the power of the beam, and ω is the angular frequency.

[0039] The photocurrent produced by the photodiode fluctuates because of the underlying fluctuations in the impinging photon number. Photon number fluctuations are reflected in photocurrent fluctuations with a fidelity determined by η . The time-varying photocurrent $i(t)$ can be broken into a time-independent average current $\langle i \rangle$ and time-varying fluctuations $\Delta i(t)$ as shown in equation (2):

$$i(t) = \langle i \rangle + \Delta i(t) \quad (2)$$

[0040] The average value of $\Delta i(t)$ is zero but the average of square of Δi , $\langle (\Delta i(t)^2) \rangle$, is not zero. Considering a load resistor R_L for the photodiode measuring shot noise, the time-varying noise power is computed as shown in equation (3):

$$P_{noise}(t) = (\Delta i(t))^2 R_L \quad (3)$$

[0041] For an optical field with Poisson photon number statistics are shown in equations (4) and (5):

$$P(n) = \frac{\bar{n}^n}{n!} e^{-\bar{n}}, \quad n = 0, 1, 2, \dots \quad (4)$$

$$(\Delta n)^2 = \bar{n} \quad (5)$$

where Δn is standard deviation in the number of photons and $\bar{n}$ is the mean. The photoelectron statistics also follow a Poisson distribution as shown in equation (6):

$$(\Delta N)^2 = \langle N \rangle \quad (6)$$

[0042] Since $i(t)$ is proportional to the number of photoelectrons generated per second, the photocurrent variance satisfies equation (7):

$$(\Delta i)^2 \propto \langle i \rangle \quad (7)$$

[0043] It is noted that the variance of current fluctuations in the photocurrent is directly proportional to the average value of the photocurrent. In a balanced configuration, variance of the photon number distribution impinging on the two photodiodes adds up and the mean is cancelled since the mean is the same for both PD1 and PD2. Therefore, a Gaussian distribution is found at the output such that the variance increases linearly with a linear increase in the intensity of the optical field.

[0044] In embodiments, the disclosed methods, devices and systems include the following stages for generating quantum random numbers: an optoelectronic stage, an amplification stage and an entropy processing stage. Each stage is described below.

[0045] In embodiments, the optoelectronic stage provides raw entropy in the form of an analog current signal generated by a balanced linear optocoupler receiving photons from a light source. The analog current signal includes the shot noise from the balanced detection of photons being transmitted (i.e., emitted) from an LED light source (i.e., emitter), while excluding classical common mode noise caused by various factors such as power supply fluctuations and thermal noise introduced by electronic devices such as the LED emitter and the photodetectors of the optocoupler. In other words, the balanced circuit configuration of the linear optocoupler cancels out the classical common mode noise, such as from the driving current of the LED while maintaining the shot noise. It is noted that shot noise is due to the discrete nature of photons which results in a random amount of photons arriving to the photodetectors in a light beam having Poisson statistics. Therefore, the photocurrent generated by photodetectors of the optocoupler fluctuate due to the underlying fluctuations in the impinging number of photons (i.e., the number of photons impinging on the photodetectors fluctuates over time). In practice, noise created by other electronic devices (e.g., amplifier) after the optocoupler (e.g., amplifier noise or thermal noise) may also be present in the signal along with the shot noise.

[0046] In embodiments, the optocoupler may be an arrangement of an LED and photodiode pair electrically connected in a balanced configuration to eliminate common mode noise from the power source and the DC mean of the output current. The entropy of the detected analog current signal is derived from the fluctuations of the number of photons impinging on each photodiode over time due to the discrete nature (i.e., shot noise) of the photons which is fundamentally probabilistic and cannot be

predicted. In other words, the number of photons received by each of the photodiodes over a given time interval is random resulting in a generated current signal that randomly fluctuates. The current signal output by the photodiode pair is then input to the amplifier stage for amplification and conversion.

[0047] In embodiments, the amplifier stage includes two parts: a transimpedance amplifier and a voltage amplifier. The transimpedance stage includes a transimpedance amplifier with a suitable transimpedance gain to convert shot noise, which may be an electrical current signal, into an electrical voltage signal. Voltage amplifiers are then used to amplify the voltage signal appropriately according to the digitizer parameters (e.g., digitizer range) to eliminate gaps in the digitized values.

[0048] The entropy processing system that receives the amplified signal includes an entropy digitizer and a randomness extractor. The entropy digitizer may be an analog to digital converter (ADC) that converts (i.e., digitizes) the amplified analog voltage signal to digital code. In order to remove bias from the entropy in the digital code, the randomness extractor (e.g., a processor) may convert (i.e., modify) a probability distribution (e.g., Poissonian distribution) of the digital code into another probability distribution (e.g., Uniform distribution) that makes every digital code equally probable and therefore more suitable for generating random numbers for use in applications. In embodiments, a Toeplitz matrix extractor may be used as a randomness extractor.

[0049] The disclosed methods, devices and systems herein then output the generated quantum random numbers (e.g., the generated digital codes in binary form or the generated digital codes converted to another number base) to a random number application device (e.g., smart device, personal computer, microcontroller, etc.). The random number application device may use the random numbers in various executed software applications including but not limited to cryptography, token generation, Monte-Carlo simulations, fundamental physics experiments, and gaming applications.

[0050] In embodiments, the QRNG device may be a standalone device that transmits the random numbers to an external application device. In another example, the QRNG device may be integrated directly into the application device itself as an integrated circuit.

[0051] Benefits of the disclosed methods, devices and systems described herein include but are not limited to generating a quantum random number unbiased by classical common mode noise in a cost effective and efficient manner. The disclosed methods, devices and systems provide a standalone or fully integrated solution that can be easily manufactured and tailored to specific random number based applications.

[0052] FIG. 1 shows a block diagram of a QRNG circuit 100 coupled to a quantum random number utilizer (QRNU) device such as random number application device 114. QRNG circuit 100 generally includes power source 102, opto-coupler 104, amplifier 106, digitizer 108, randomness extractor 110, and an optional QRNG controller 112. Although not shown, power source 102 is connected to and applies power (e.g., Direct Current (DC)) to each component of QRNG circuit 100 including opto-

coupler 104, amplifier 106, digitizer 108, randomness extractor 110, and optional QRNG controller 112. In another example, power source 102 may be external to QRNG circuit 100.

[0053] In a first stage of QRNG circuit 100, opto-coupler 104 produces a quantum random signal (e.g., voltage signal or current signal) representing shot noise. As shown in FIG. 2, the light source of optocoupler 104 may include LED 202 (i.e., light emitter) that is powered by power source 102 via wires 204 and 206. LED 202 transmits photons along paths 202A and 202B towards two light receivers (e.g., photodiodes 208 and 210) which are powered by power source 102 via wires 212 and 216 and generate the quantum random signal on common node connection 214 in response to receiving a different number of photons from LED 202. In other words, the quantum random signal is produced because of the particle nature of photons. Due to a difference in the number of photons travelling along paths 202A and 202B, the photocurrent produced by photodiodes 208 and 210 are different (e.g., in amplitude). The light receivers (e.g., photodiodes 208 and 210) are connected in a balanced bias configuration by common node connection 214 such that classical common mode noise (i.e., noise common to both photodiodes 208 and 210) is eliminated while the uncorrelated shot noise (power) generated by photodiodes 208 and 210 in response to the impinging photons accumulates.

[0054] For example, LED 202 might generate noise on the generated light beam due to thermal noise, power supply noise or electromagnetic interference. In addition, the average light received by photodiodes 208 and 210 produces a common DC bias in the current signals produced by photodiodes 208 and 210. By connecting photodiodes 208 and 210 in the manner shown in FIG. 2, the common DC bias and other common mode noises are effectively eliminated leaving just a current signal representing the shot noise (i.e., a shot noise signal) at 214.

[0055] In other words, the current signal output on common node connection 214 is the quantum random signal which is a current signal that randomly fluctuates due to the inherent shot noise from the particle nature of photons and the difference in the number of photons emitted along paths 202A and 202B over time without being biased by common mode noise. This quantum random signal based solely on shot noise forms the basis for the entropy source to generate the quantum random numbers.

[0056] It is noted that although FIG. 2 shows an opto-coupler configuration including a single LED and a pair of photodiodes, other configurations are of course feasible. For example, the opto-coupler may include more than one LED and more than two photodiodes. In addition, the LEDs and photodiodes may be replaced by other light sources and light detectors. It is also noted that the LEDs and photodiodes may operate in any suitable wavelength including wavelengths in the infrared light spectrum.

[0057] In a second stage of QRNG circuit 100, amplifier 106 amplifies and converts the quantum random signal produced by the photodiodes of opto-coupler 104 for further processing by QRNG circuit 100. As shown in FIG. 3, amplifier 106 may include an operational amplifier 302 and a resistor 306

electrically coupled together to form a transimpedance amplifier for converting the analog current signal output by opto-coupler 104 into an analog voltage signal. Amplifier 106 may also include an operational amplifier 304 acting as a voltage amplifier for amplifying the analog voltage signal. The quantum random signal is input as an analog current signal from opto-coupler 104, converted to an analog voltage signal and then amplified for further processing by QRNG circuit 100. It is noted that the range of the amplified signal can be set to correspond to the range of digitizer 108 to produce efficient extraction. In other words, the amplification factor is set such that the full range of the digitizer may be utilized.

[0058] For example, optocoupler 104 outputs photoelectrons corresponding to the shot noise detection by the circuit. Each photoelectron (or a predetermined number of photoelectrons) may lead to a change of the output code by the least significant bit (LSB) of the digitizer. Therefore, the amplification factor results in the resolution of the digitizer to map bit values to each photoelectron (or a predetermined number of photoelectrons) resulting from the shot noise. Stated differently, amplifier 106 maps the current signal from the photodiodes to the voltage range of the digitizer.

[0059] In a third stage of QRNG circuit 100, digitizer (e.g., ADC) 108 digitizes the analog voltage signal output from amplifier 106. Digitizer 108 may have an input voltage range as discussed above, an input voltage resolution and a sampling period that are preset or adjustable. Generally, the input voltage range may be from 0 volts to +X volts and have a voltage resolution of $X/2^N$ volts, where N is the number of bits supported by the ADC. The sampling rate may be set to a value F_s for producing a random sample of bits every sampling period $T_s=1/F_s$ seconds.

[0060] For example, the ADC may have a range from 0 volts to 3.3 volts and have 16-bits of resolution such that the voltage resolution is 0.05mV. In this example, the ADC may output 65,536 possible bit sequences that may represent the quantum signal and therefore digitized quantum random numbers. It is noted that the ADC output may be a sequence of bits (e.g., 16 bits) output either in parallel or serially from the ADC. In either case, the bit sequence is sent to the randomness extractor for further processing.

[0061] In a fourth stage of QRNG circuit 100, a randomness extractor 110 is implemented to extract random numbers from the output of the digitizer such that the random numbers output by QRNG circuit 100 have a uniform probability distribution. This is beneficial because the output of digitizer 108 is based on shot noise which has an inherent Poissonian probability distribution with a natural bias for certain ranges of numbers. Such biased probability distributions do not lend themselves to most random number applications which desire a random number having a uniform probability distribution. In other words, the digitized quantum random signal output by ADC 108 is random but is also biased due to the inherent Poissonian probability distribution of the shot noise. This bias is compensated for by randomness extractor 110 prior to outputting a final random number to random number application device 114.

[0062] It is noted that the randomness extractor can take many forms. In embodiments, randomness extractor 110 may be implemented by a Toeplitz matrix having coefficients set to extract the digitized samples according to a uniform distribution. In practice, the random bits can be arranged in vector form and multiplied by the Toeplitz matrix. The result of this multiplication is extracted digitized samples according to a uniform distribution. Of course, multiplication does not actually have to be performed, as a look-up-table for the products may be pre-loaded into randomness extractor 110.

[0063] FIG. 4 shows a flowchart 400 of an example randomness extractor 110 of the QRNG circuit 100. In this example, randomness extractor 110 receives a digital output (bit sequence representing random voltage values) from digitizer 108 in step 402. In step 404, the entropy $H_{\min}$ of the random bit sequence is computed. In embodiments, entropy $H_{\min}$ of the distribution is calculated by equation (8) below which quantifies the amount of randomness of a distribution X (e.g., distribution of output values from digitizer 108) on $\{0,1\}^n$. The entropy of given random number samples X is determined by sample point x with maximal probability, given that the distribution is Poissonian. In other words, a set (e.g., histogram) of outputs from digitizer 108 is operated on to determine $H_{\min}$. It is noted that frequency of computing $H_{\min}$ may be set based on various factors and may coincide with startup of the random number generating device, may be performed cumulatively in real-time, may be performed periodically, or may be performed during a self-testing phase.

$$H_{\min} = -\log \left(\max_{x \in \{0,1\}^n} P_r[X = x] \right) \quad (8)$$

[0064] In step 406, randomness extractor 110 extracts one of the digitized samples from the multiple digitized samples. The number of bits to be extracted from the digital sample is determined to be either $H_{\min}-1$ or $\text{floor}(H_{\min})$. In other words, $H_{\min}$ of the distribution is computed in digital units (i.e., bits) and then an extractor is used to extract either $H_{\min}-1$ bits or $\text{floor}(H_{\min})$ bits from each sample from the digitizer. This entropy-based extraction process ensures that digitized samples are random and not influenced by classical sources of entropy. In other words, this extraction process produces a uniformly distributed sequence that is random and is derived from quantum random processes. In embodiments, a sample of X bits can be multiplied with a uniformly distributed matrix of 1s and 0s, in such a fashion as to result in the output sequence of $H_{\min}-1$ bits. Therefore, for every sample of X bits, an output of $H_{\min}-1$ bits is received. For example, if there is a 12-bit sample and the $H_{\min}-1$ is 4 bits, multiplication of two matrices may be used such that an input matrix of dimensions 1×12 is multiplied with a uniform matrix of dimensions 12×4 . This results in an output matrix of dimensions 1×4 , which is the 4-bit output. In embodiments, the uniform matrix is modulo 2 and may include an equal number of 0s and 1s uniformly distributed in the matrix. In either case, if more digitized samples are needed in step 408, the process loops back and repeats step 402. If the digitized samples are adequate, then they are output to random number application device 114 in step 410. Randomness extractor 110 extracts and outputs

random numbers based on the entropy from a Poisson distribution and outputs uniformly distributed digitized samples to be used as random bits by random number application device 114. The output bit sequences may be used as separate random numbers by the random number application device 114, or may be combined (e.g., appended to one another) by random number application device 114 to produce a larger size random number. For example, if the extractor outputs 16-bit sequences, but random number application device 114 requires 256-bit sequences, random number application device 114 can append sixteen 16-bit sequences together to form the desired 256-bit sequence.

[0065] It is noted that the operation of opto-coupler 104, amplifier 106, digitizer 108 and randomness extractor 110 in FIG. 1 and described above may be automatic upon receiving power from power source 102 (which may or may not be internal to QRNG circuit 100) or may be controlled by optional QRNG controller 112, or by random number application device 114. In addition, operational parameters of opto-coupler 104 (e.g., LED brightness), amplifier 106 (e.g., amplification range, bias, etc.), digitizer 108 (e.g., ADC range, bias, resolution, sampling frequency, etc.) and randomness extractor 110 (e.g., entropy extraction, Toeplitz matrix parameters, etc.) may be preset in the respective devices upon manufacturing or may be adjustable by optional QRNG controller 112 or by random number application device 114.

[0066] FIG. 5 shows a block diagram of a processor system that may represent the hardware present in optional QRNG controller 112 of the QRNG circuit 100 and random number application device 114. The optional QRNG controller 112 and the random number application device 114 may generally include a processor 502, memory device 504, QRNG circuit input/output (I/O) interface 506 and user I/O interface 508.

[0067] In embodiments, when the QRNG circuit 100 includes optional QRNG controller 112, processor 502 of optional QRNG controller 112 may control the operation of QRNG circuit 100 via interface 506 according to computer code stored in memory device 504, and/or user input received via user I/O interface 508. Processor 502 may then output the generated random bits to random number application device 114 via I/O interface 506.

[0068] In another example, when the QRNG circuit 100 does not include optional QRNG controller 112, processor 502 of random number application device 114 may control the operation of QRNG circuit 100 via QRNG circuit I/O interface 506 according to computer code stored in memory device 504, and/or user input received via user I/O interface 508. Processor 502 may then extract the generated random bits from QRNG circuit 100 via I/O interface 506.

[0069] FIG. 6 shows a flowchart 600 of an example operation of the QRNG circuit 100. In step 602, the light source (e.g., LED) of opto-coupler 104 randomly emits photons to the light receivers (e.g., photodiodes) of opto-coupler 104. The light receivers of opto-coupler 104 then convert the photons to

an electrical signal (e.g., electrical current) proportional to the shot noise due to the difference in the numbers of photons given the particle nature of light received between the light receivers. As mentioned above, the generated electrical signal does not include classical noise due to the balanced connection between the light receivers which cancels common mode noise. In step 604, amplifier 106 and digitizer 108 amplify and digitize the analog signal respectively to produce raw bits representing the shot noise. Specifically, transimpedance amplifier 302 converts the analog current signal to an analog voltage signal, while voltage amplifier 304 amplifies the voltage output by transimpedance amplifier 302. The amplified voltage signal output by voltage amplifier 304 is then sampled by digitizer 108 at a sampling frequency and converted to a bit sequence representing the amplitude of the sampled voltage. The sampling frequency, range and resolution of digitizer 108 may be preset or be set based on the requirements of random number application device 114. In step 606, randomness extractor 110 performs an extraction process on the output of digitizer 108 and outputs uniformly distributed random numbers. Specifically, randomness extractor 110 computes entropy of the digitized samples. This entropy is used to determine the number of bits that can be extracted for randomness. The reason for this extraction is to ensure a uniform distribution of random bit sequences being output by the system and eliminate influence of classical noise sources. In step 608, if more random bits are needed, then the steps are repeated. For example, the output randomness extractor 110 may include samples of N-bits in length. However, random number application device 114 may be utilizing the random bits for cryptography purposes where M-bits (where $M > N$) are needed to perform a certain application function (e.g., cryptographic function). In this example, the method may repeat steps 602-608 several times until M random bits total are generated. In other words, each sample of N-bits is accumulated and combined to form an M-bit word for use in a practical application. These random bits are then output to random number application device 114 in step 610.

[0070] FIGS. 7A-7E illustrate experimental results related to the operation of an experimental QRNG circuit. Specifically, FIG. 7A depicts an example of an experimental test bench setup 700 for the QRNG circuit. FIGS. 7B, 7C, 7D, and 7E show various statistical results obtained from the operation of the test bench in FIG. 7A. These figures provide insights into the performance and characteristics of an example QRNG circuit under experimental conditions. It should be noted that this experimental test bench and the resultant experimental results are provided as an example only and are not intended to limit the scope of the disclosure. The QRNG circuit may operate with different circuitry and/or parameters or produce different results in other implementations or configurations.

[0071] In an example experimental setup shown in FIG. 7A, an optocoupler 702 is coupled to an amplifier 704 and a microcontroller 706. During the experiment, LED 702A illuminates two photodiodes 702B and 702C which generate a shot noise signal at the output. Shot noise manifests as a weak current noise signal and therefore needs high amplification and converted to voltage for further

processing. The amplification is chosen to preserve the bandwidth while providing a high gain to utilize ADC 706A effectively.

[0072] ADC 706A may be integrated into the microcontroller or it can be a discrete component. In either case, ADC 706A converts the analog signal to digital codes that can then be run through an extractor algorithm. Raw ADC codes are then transferred to the computer for this experiment which implements an extractor such as Toeplitz extractor 706B. An instance of a Toeplitz extractor 706B was also implemented on the microcontroller to run standalone. It is noted that Toeplitz extraction is just one example of an extraction algorithm. Other extraction algorithms may be used.

[0073] The noise signal at the output of the voltage amplifier is classified into two categories (i.e., quantum noise and classical noise). As previously mentioned, quantum noise is the noise generated by the optocoupler which corresponds to the shot noise of the LED 702A. Classical noise is any noise that is not shot noise. This includes noise such as thermal noise, detector dark noise, and amplifier noise. The contribution of shot noise to the output signal at the voltage amplifier is found to be more significant than classical noise from the system. To characterize this relationship, quantum to classical noise ratio (QCNr) is calculated, where QCNr is the ratio of shot noise from the optocoupler to the overall classical noise in the system. The experiment also calculates the min-entropy which is the minimum entropy of the raw signal. The output distribution from the voltage amplifier is Poisson.

[0074] As mentioned above, the light receivers produce signals corresponding to the shot noise generated by the light receivers. FIG. 7B shows a histogram showing a comparison between the histogram of classical noise when the light source of optocoupler is turned OFF and the histogram of the combination of classical noise and shot noise when the light source of optocoupler is turned ON. Classical noise is Gaussian in nature, and therefore does not have a mean but does have a variance. Once the LED is turned on, a Poisson distribution due to the shot noise is added to the Gaussian distribution due to the classical noise. Due to balanced detection, the average photocurrent of both the photodiodes cancel each other out thereby eliminating the mean from the Poisson distribution. However, the overall variance increases as the intensity of the light beam from the LED increases due to increased forward current. This indicates that the variance of the histogram distribution increases due to the shot noise. As mentioned above, due to the common connection between the light receivers, the classical common mode noise voltage is eliminated leaving only the voltage from the shot noise for use by QRNG circuit.

[0075] Specifically, in FIG. 7B, the output statistics 710 of the noise signal captured by the ADC are shown. The statistics 710A are captured with the LED off, this is the overall classical noise in the system. The LED of the optocoupler is then turned ON (e.g., 20mA of forward current in the test bench) to capture further statistics 710B which are a mixture of quantum and classical noise in the system. Since the variance of the quantum noise is an order of magnitude more significant than the classical noise, it

is noted that the quantum noise in the output signal dominates. To calculate the QCNR, the variance of Classical noise ($\sigma_{classical}^2$) and quantum noise ($\sigma_{Quantum}^2$) may be utilized in equation (9):

$$QCNR = 20 \log \left(\frac{\sigma_{quantum+classical}^2 - \sigma_{classical}^2}{\sigma_{classical}^2} \right) \quad (9)$$

where $\sigma_{Quantum+classical}^2$ is the variance of the output noise signal with the LED ON given in 710B, and $\sigma_{classical}^2$ is the variance of the output noise signal with the LED OFF given in 710A. The QCNR for the device was computed as 32 dB. It is also beneficial to calculate the min-entropy of the output sequence. The min-entropy is the minimum information-theoretic random bits in a sequence. The minimum entropy is used to define the extraction ratio which bounds the maximum output bits from the extractor. To calculate the min-entropy from the output statistics, equation (8) is used which quantifies the amount of randomness of a distribution X on $\{0,1\}^n$. The entropy of given sequence X is determined by sample point x with maximal probability.

[0076] The extraction ratio of the device was computed to be 7 bits for a 12-bit ADC on a RP2040 microcontroller used in the test bench. The ADC on the RP2040 microcontroller has an effective number of bits (ENOB) of 8 which is the maximal entropy that can be generated from the device. Putting a sufficient clearance to this limit the extractor is used to extract 5 bits per sample using a Toeplitz extractor.

[0077] FIG. 7C shows a data plot depicting a relationship between the amplitude of variance with respect to the mean. For a Poisson distribution, the variance changes linearly with the mean. In other words, as forward current to the LED increases, the average photocurrent of the photodiodes also increases and a linear increase in the variance of the noise results. Since classical noise does not depend on the LED intensity, the variance of the classical noise voltage remains constant even with increasing forward current driving the LED. In other words, the variance of the shot noise voltage (not the variance of the classical noise voltage) continuously increases with increasing amplification current of the LED, and since the output distribution is linear with the mean, it is a Poisson distribution.

[0078] Specifically, FIG. 7C depicts the variance 712 of the output noise signal for different values of forward current to the LED. In this example, 3 runs of varying the value of the DAC-controlled current source from 0 to 2000 were executed to check that the output distribution is Poissonian. A linear fit of the data along line 712B was performed to determine that the variance change is linear to the change of the mean value of the optical field. In FIG. 7C, the experiment varies the forward current of the LED increasing the intensity of the output and measuring the variance of the output signal. Changing the forward current of the LED linearly increases the output intensity of the optical field. Since photodiodes are linear, this manifests as a linear change in the mean of the output signal. Therefore, a linear increase in forward current is shown as a linear increase in the mean value of the output of the photodiodes. Due

to the balanced scheme this mean is cancelled out electrically leaving access to the noise signal. For a Poisson distribution, the variance of the output noise signal should vary linearly with the mean and therefore the forward current. The onboard digital to analog converter (DAC) current source was controlled using the microcontroller. Therefore, by linearly stepping the DAC value, the forward current linearly increases. It is shown in FIG. 7C that the variance values 712A are increasing with a linear trend for 3 separate runs of the experiment. This trend would be quadratic for Super-Poissonian distribution if the classical noise in the system were to dominate. This can be used as a health test for the device before entropy is collected and processed.

[0079] For the trials shown in FIGs. 7B and 7C for the example test bench, 1.2 Gb of random bits were used and divided into 4 batches of 300 Mb each at an acquisition rate of 181 Kb/sec of raw entropy. A Toeplitz extractor was then run on the captured data.

[0080] To test the statistical randomness of the generated results, the experiment ran the output of the QRNG through the dieharder test suite and also ran autocorrelation tests. The result of the dieharder test is visualized in a KS test shown as data lines 716A in plot 716 of FIG. 7E where the KS plot of the p-values from the output of a dieharder test were run on 900 Mbit data, divided into 3 chunks and tested individually for statistical randomness. The output of the KS test should be a uniform distribution of p-values for truly random bit, shown by the dotted line 716B. In this analysis, the Cumulative Density Function (CDF) of the p-values from the test is compared to the CDF of an ideal distribution of p-values for statistically random data. It is shown that the trend for each run of the 300 Mb file is linear which is what is expected for a truly uniform output. An autocorrelation on 5 Mbit of extracted data is calculated by performing a cross-correlation of a sequence with itself lagged bitwise. If x and y are 1-D arrays, the equation used for the cross-correlation is given below as equation (10):

$$Z[k] = (x * y)(k - N + 1) = \sum_{l=0}^{|x|-1} x[l]y_{l-k+N-1} \quad (10)$$

for $k = 0, 1, \dots, |x|+|y|-2$, where $|x|$ is the length of x, $N = \max(|x|, |y|)$, and y_m is 0 when m is outside the range of y.[12]

[0081] This autocorrelation 714 is shown in FIG. 7D where inset 714A shows Pearson product moment coefficients for autocorrelation of 1450 chunks of the autocorrelation data 714B. The line indicates the linear fit of the points, where the average value should be close to zero, indicating no positive or negative correlations. The Pearson product moment correlation coefficients were calculated for 5000-bit chunks and the autocorrelation was run on 100 chunks of data. The standard deviation of the coefficients for the data size had a confidence interval of 0.013 (shown in shaded region of 714A) and a three-sigma confidence interval of 0.039 (shown in shaded region of 714A).

[0082] It is observed that there is only one peak which corresponds to when the lag is zero, or when cross-correlation of the sequence is performed with itself. Correlations inside the chunks would show

up as points lying outside the three-sigma bound for the correlation coefficients, and as the size of the data is increased to run the correlation test, the value of the correlation coefficients would also decrease.

[0083] While the foregoing is directed to example embodiments described herein, other and further example embodiments may be devised without departing from the basic scope thereof. For example, aspects of the present disclosure may be implemented in hardware or software or a combination of hardware and software. One example embodiment described herein may be implemented as a program product for use with a computer system. The program(s) of the program product defines functions of the example embodiments (including the methods described herein) and can be contained on a variety of computer-readable storage media. Illustrative computer-readable storage media include, but are not limited to: (i) non-writable storage media (e.g., read-only memory (ROM) devices within a computer, such as CD-ROM disks readably by a CD-ROM drive, flash memory, ROM chips, or any type of solid-state non-volatile memory) on which information is permanently stored; and (ii) writable storage media (e.g., floppy disks within a diskette drive or hard-disk drive or any type of solid-state random-access memory) on which alterable information is stored. Such computer-readable storage media, when carrying computer-readable instructions that direct the functions of the disclosed example embodiments, are example embodiments of the present disclosure.

[0084] It will be appreciated by those skilled in the art that the preceding examples are exemplary and not limiting. It is intended that all permutations, enhancements, equivalents, and improvements thereto are apparent to those skilled in the art upon a reading of the specification and a study of the drawings are included within the true spirit and scope of the present disclosure. It is therefore intended that the following appended claims include all such modifications, permutations, and equivalents as fall within the true spirit and scope of these teachings.

CLAIMS

What is claimed is:

1. A quantum random number generator (QRNG) system comprising:
 - an optocoupler including at least one light emitter and at least two light receivers that are electrically coupled to each other in a configuration that rejects common mode noise; and
 - a controller configured to:
 - control the at least one light emitter to emit photons randomly towards the at least two light receivers,
 - measure a shot noise signal generated by the at least two light receivers in response to receiving the photons, and
 - extract a random number from the shot noise signal.
2. The system as in claim 1, wherein the at least one light emitter is a light emitting diode (LED), and at least two light receivers are photodiodes.
3. The system as in claim 1, wherein the shot noise is measured from a common node where the at least two light receivers are electrically coupled to each other.
4. The system as in any one of the preceding claims, further comprising:
 - an amplifier for amplifying the shot noise signal generated by the at least two light receivers;
 - and
 - a digitizer for digitizing the amplified shot noise signal,
 - wherein the controller is further configured to extract the random number from the digitized shot noise signal.
5. The system as in claim 4, wherein the controller is further configured to extract the random number from the digitized shot noise signal by:
 - modifying a probability distribution of the digitized shot noise signal, and
 - extracting the random number from the modified probability distribution.
6. The system as in claim 5, wherein the controller is further configured to modify the probability distribution and extract the random number from the modified probability distribution by applying a randomness extractor to the digitized shot noise signal.

7. The system as in claim 5, wherein the controller is further configured to modify the probability distribution by converting the probability distribution due to the digitized shot noise signal into a Uniform probability distribution for use in a software application.
8. The system as in claim 4, wherein the digitized shot noise signal has a Uniform probability distribution.
9. The system as in any one of the preceding claims, further comprising a quantum random number utilizer (QRNU) device coupled to the QRNG, the QRNU device executing a software application that utilizes the extracted random number.
10. The system as in any one of the preceding claims, wherein the controller is further configured to control a size of the random number extracted from the shot noise based on a software application that utilizes the extracted random number.
11. A quantum random number generator (QRNG) method comprising:
 - controlling, by a controller, at least one light emitter to emit photons towards at least two light receivers that are electrically coupled to each other in a configuration that rejects common mode noise;
 - measuring, by the controller, a shot noise signal generated by the at least two light receivers in response to receiving the photons; and
 - extracting, by the controller, a random number from the shot noise signal.
12. The method as in claim 11, further comprising:
 - controlling, by the controller, a light emitting diode (LED) as the at least one light emitter; and
 - measuring, by the controller, the shot noise signal from photodiodes as the least two light receivers.
13. The method as in claim 11, further comprising:
 - measuring, by the controller, the shot noise signal from a common node where the at least two light receivers are electrically coupled to each other.
14. The method as in any one of the preceding claims, further comprising:
 - amplifying, by an amplifier, the shot noise signal generated by the at least two light receivers;
 - digitizing, by a digitizer, the amplified shot noise signal; and
 - extracting, by the controller, the random number from the digitized shot noise signal.
15. The method as in claim 14, further comprising:
 - extracting, by the controller, the random number from the digitized shot noise signal by:

modifying a probability distribution of the digitized shot noise signal, and
extracting the random number from the modified probability distribution.

16. The method as in claim 15, further comprising:

modifying, by the controller, the probability distribution, and extracting the random number from the modified probability distribution by applying a randomness extractor to the digitized shot noise signal.

17. The method as in claim 15, further comprising:

modifying, by the controller, the probability distribution by converting the probability distribution due to the digitized shot noise signal into a Uniform distribution for use in a software application.

18. The method as in claim 14, further comprising:

outputting, by the controller, the digitized shot noise signal having a Uniform probability distribution.

19. The method as in any one of the preceding claims, further comprising:

executing, by a quantum random number utilizer (QRNU) device coupled to the QRNG, a software application that utilizes the extracted random number.

20. The method as any one of the preceding claims, further comprising:

controlling, by the controller, a size of the random number extracted from the shot noise signal based on a software application that utilizes the extracted random number.