IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received March 16, 2020, accepted April 1, 2020, date of publication April 20, 2020, date of current version May 4, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.2988658

Entropy and Randomness: From Analogic to Quantum World

EMIL SIMION
Center for Research and Training in Innovative Techniques of Applied Mathematics in Engineering, University Politehnica of Bucharest, 060042 Bucharest, Romania

e-mail: emil.simion@upb.ro

This work was founded by University Politehnica of Bucharest.

ABSTRACT This work is dedicated to the construction and evaluation of random number generators used in cryptography. The critical element on which the security of information is based is the cryptographic key (usually a binary sequence). In order to be resistant to brute force attacks it is necessary that it be made up of random variables with a certain degree of randomness and independence. Formally, this comes back to generate the cryptographic key through the systems which ensures a certain minimum level of entropy. The observer has access to a sample, of a certain size, and based on it he will estimate the minimum value of the entropy, in the situations in which the variables resulting from the measurement process are independent. In the situation where these variables are not independent, complex mathematical procedures also allow estimation of the minimum entropy. This article is a review of how mathematical entropy can be estimated and evaluated, of the construction mode (from technologies based on analogue procedures: thermal noise in a transistor to modern procedures: quantum devices), as well as to evaluate the security of binary sequence generators used for generating cryptographic keys or critical security parameters related to new technologies based on quantum principles. The techniques and methods used to generate binary random values as well as the methods of statistical and informational validation (Shannon entropy) are exemplified in this paper.

INDEX TERMS Entropy, randomness, statistical estimation, quantum.

I. INTRODUCTION

Nowadays our real life has two components: physical and virtual. Due to the dynamics, the usual activities that we carry out in the physical world will be relocated in the virtual world: interaction with the local public authorities, electronic payments, civic actions (vote, surveys, donations etc.), computerization of the financial field, navigation systems, computer activities, gaming, socializing etc.

For this reason, the virtual component necessitates implementation of safety measures for the protection of personal data, as well as electronic communications. In high-confidence encryption, generating "random" numbers is essential in order to ensure the strength of the cryptographic key, based on provable secure algorithms or physical processes.

There are two types of elementary methods used to generate these numbers: DRBG deterministic methods (based on the algorithm and its initialization) and non-deterministic TRBG methods (based on physical properties: thermal noise from a transistor, ring oscillator, quartz, magnetic RAM etc.). Systems based on deterministic and non-deterministic combinations are called mixed methods.

In the present paper we focus on the problem of constructing, testing and validating random number generator devices based on deterministic and / or non-deterministic components. In order to estimate the quality of these devices, it is necessary to evaluate the entropy of the mentioned components.

Section 2 is dedicated to entropy like a measure of randomness. After a brief introduction of Shannon entropy and conditioned entropy we present several statistical tests concerning the diversity of a population and regarding the diversity *for m* independent populations.

Section 3 is focused on the manner random numbers can be obtained: by deterministic or non-deterministic methods. In this section there are presented the techniques and methods based on physical principles that allow the generation of random numbers, as well as the implementation principles of the technologies based on TRBG.

Section 4 is widely dedicated to the validation of the mathematical model that underlies the TRBG implementation,

The associate editor coordinating the review of this manuscript and approving it for publication was Jiafeng Xie.

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

entropy estimation methods, in case the random variables that model the TRBG output are independent and correlated respectively.

Section 5 presents the statistical tests used to validate these categories of equipment and devices.

In Section 6 we present a series of such hardware devices that are available for purchase or used in various product categories. A comparative study has been realized from the point of view of their security.

Section 7 hands over the conclusions.

II. ENTROPY

Assume that we have the set of possible outcomes $\{\alpha_1, \ldots \alpha_m\}$ with the probability of occurrence $(p_1, \ldots, p_m)$. Let us consider the random variable X with the probability:

$$P(X = \alpha_i) = p_i \quad i = 1, ..m \qquad (1)$$

The entropy of the random variable X is defined by:

$$H(X) \equiv H(p_1, \ldots, p_m) = -\sum_{i=1}^{m} p_i \log p_i$$

Assume that Y is a random variable with the probability:

$$P(Y = \beta_j) = q_j, \quad j = 1, \ldots, n \qquad (2)$$

Denote by $q_{j|i} = P(Y = \beta_j | x = \alpha_i)$ the conditioned probability of $Y = \beta_j$ given $X = \alpha_i$ and by $p_{i|j} = p(X = \alpha_i | y = \beta_j)$ the conditioned probability of $X = \alpha_i$ given $Y = \beta_j$. Then:

$$r_{ij} = P(X = \alpha_i, Y = \beta_j) = p_i q_{j|i} = q_j p_{i|j}, \quad i = 1, ..m, \ j = 1, .., n. \qquad (3)$$

The joint entropy H(X,Y) of (X,Y) is defined such as $H(X,Y) = H(r_{11}, \ldots, r_{1n}, r_{21}, \ldots, r_{mn})$.

Definition 2: For the joint variables (X,Y) with the probabilities (1), (2), respectively (3) the conditioned entropy H(X|Y) of X by Y is defined as:

$$H(X, Y) = -\sum_{I,J} r_{ij} \log \frac{r_{ij}}{q_j}$$

Using the properties of the entropy we get H(X|Y) = H(X,Y)-H(Y). All the above definitions can be extended to continuous random variables. For example, definition 1 becomes:

Definition 3: Entropy of the continuous d dimensional $X = (X_1, \ldots, X_d)$ random variable with the density $\rho(x)$ is defined by:

$$h(X) \equiv h(\rho) = -\int_{R^d} \rho(t) \log(\rho(t) dt$$

if the integral exists. This entropy is called the differential entropy. We introduce now the relative entropy. Intuitively this has the significance of the similarity between the two distributions. Let μ and ν two probabilistic measures defined over the measurable space (X, B(X)). The measure μ is absolute continuous reported to ν and write $\mu < \nu$ if $\nu(A)=0 \Rightarrow \mu(A)\ \forall\ A \in B(X)$. Radon-Nikodym theorem states that μ is absolute continuous relative to ν then exists a function $\varphi(x)$ ν-integrable such that:

$$\mu(A) = \int_A \varphi(t) d\nu(t) \quad \forall\ A\ \in B(X)$$

The Radon-Nikodym derivative $\varphi(x)$ is given by:

$$\varphi(t) = \frac{d\mu}{d\nu}(t).$$

Definition 4: For the probability measures μ and ν, the relative entropy $H(\mu;\nu)$ is defined as:

$$H(\mu; \nu) = \int_X \log \frac{d\mu}{d\nu} d\mu(t) \text{ if } \mu < \nu$$

and $H(\mu;\nu) = \infty$ if μ is not absolute continuous relative to ν.

The measure ν is called the reference measure. In the discreet case consider μ and ν probability measures defined on the space $X = \{\alpha_1, \ldots, \alpha_m\}$. The measures μ and ν are specified by $\mu(\{\alpha_i\}) = p_i$ and $\nu(\{\beta_i\}) = q_i$ $(i = 1, \ldots, m)$. If $p_i = 0$ when $q_i = 0$, then $\mu < \nu$ and the Radon-Nikodym derivatives becomes:

$$\frac{d\mu}{d\nu}(t) = \frac{p_i}{q_i}$$

if $x = \alpha_i (i = 1, \ldots, m)$. The relative entropy becomes:

$$H(\mu\nu) = \sum_{i=1}^{m} p_i \log \frac{p_i}{q_i}$$

$$= -H(p_1, \ldots, p_m) - \sum_{i=1}^{m} p_i \log p_i.$$

If the reference measure is the uniform distribution ν_0 given by $\nu_0(\{\alpha_i\}) = 1/m$, $i = 1, \ldots, m$ then:

$$H(\mu; \nu_0) = -H(p_1, \ldots, p_m) + \log m.$$

Let us consider the continuous case. If μ and ν are two continuous distribution on R^d with the density functions $p(\mathbf{x})$ and $q(\mathbf{x})$. Assume that $\mu < \nu$. In this case the entropy $H(\mu;\nu)$ becomes:

$$H(\mu; \nu) = \int_{R^d} p(t) \log \frac{p(t)}{q(t)} dx.$$

If

$$h(p) = \int p(t) \log p(t) dx < \infty$$

then:

$$H(\mu; \nu) == h(p) - \int_{R^d} p(t) \log q(t) dx.$$

Consider the case when p(t) is vanished except a domain G of R^d with a fixed volume |G| and ν_0 is the uniform distribution on G with the density ν_0. The relative entropy H(μ;ν_0) is:

$$H(\mu;\nu_0) = -h(p) - \int_{R^d} p(t)\log q_0(t)dt$$
$$= -h(p) + \log|G|.$$

We have the following results.

Theorem 1: If μ and ν are two probability measures then H(μ;ν) $\geq$0 with equality if and only if $\mu = \nu$.

Theorem 2: If X=(X$_1$,…,X$_d$) is a random vector d-dimensional with a normal distribution N(a, Γ), the entropy of X is given by:

$$h(X) = \frac{1}{2}\log\{(2\pi e)^d|\Gamma|\}.$$

In the particular case when X is a N(a, σ^2) we have

$$h(X) = \frac{1}{2}\log(2\pi e\sigma^2).$$

Based on these results, it can be demonstrated that the normal distributions with a given covariance matrix have the maximum entropy. As the entropy measures the degree of "disorder" within a system, it is natural to impose the following restrictions of normality on the noise affecting the channel and to the messages transferred across the channel.

Theorem 3: If g(X) is the density function of a normal distribution N(a, Γ) and p(t) the density function of a continuous distribution with covariance matrix Γ then h(g)$\geq$h(p).

The following theorem gives a similar result for the relative entropy.

Theorem 4: If μ and ν are two d-dimensional gaussian distribution with the repartition N(a, Γ) respective N(I, Δ). If γ is a continuous distribution, with the same mean value and the same covariance matrix Γ like μ, then H(μ;ν) $\leq$ H(γ;ν).

Now we shall present some results regarding the asymptotic behavior of the entropy. We present the (h, φ)- entropies such generalizations of φ-entropy, Havrda-Charvat entropy and Renyi entropy etc. For these distributions we shall indicate the asymptotic distribution in the simple and stratified selections. Assume we have a population with N individuals classified in M classes x$_1$,…,x$_M$ according to a certain process X, the population divided in r levels such that the diversity and the variance in each class is minimum relative to the diversity, respective total variance. If N$_k$ is the size of level k, p$_{ik}$ the probability to select an individual in the level k which belongs to class x$_i$, and p$_I$ he probability of an individual to belongs to class x$_{i,}$ then we obtain:

$$\sum_{k=1}^{r} N_K = N, \quad \sum_{i=1}^{M} p_{ik} = \frac{N_k}{N} \quad \text{for k} = 1,\ldots,\text{r}$$
$$\sum_{i=1}^{M}\sum_{k=1}^{r} p_{ik} = 1, \quad p_i. = \sum_{k=1}^{r} p_{ik} \quad \text{for i} = 1,..,\text{M}$$

Under these circumstances the population (h, φ)- entropy is

$$H_h^{\varphi}(p) = h(\sum_{i=1}^{M}\phi(p_{i.})) = h(\sum_{i=1}^{M}\varphi(\sum_{k=1}^{r}p_{ik}))$$

If the probability is proportional of volume n and independent at each level, the estimation of (h, φ)-entropy is defined as:

$$H_h^{\varphi}(f) = h(\sum_{i=1}^{M}\varphi(f_{i.}) = h(\sum_{i=1}^{M}\varphi(\sum_{k=1}^{r}f_{ik}))$$

where f$_i$ is the relative frequency in the class x$_i$ in the total selection, f$_{ik}$ is the relative frequency of the elements which belongs to class x$_i$ in the level k, and n$_k$ is the size of the sample at level k. In this situation we obtain:

$$\frac{n_k}{n} = \frac{N_k}{N} \quad \text{for k} = 1,\ldots,\text{r}\sum_{k=1}^{r}n_k = n$$
$$\sum_{i=1}^{M}f_{ik} = \frac{n_k}{n} \quad \text{fork } = 1,\ldots,\text{r}\, f_{i.} = \sum_{k=1}^{r}f_{ik}$$

Theorem 5: In the case of simple selection, H$_h^{\varphi}$(s) has the following asymptotic behavior:

$$\sqrt{n}(H_h^{\varphi}(f) - H_h^{\varphi}(p)) \xrightarrow{L} N(0,\sigma^2)$$

if

$$\sigma^2 = \sum_{i=1}^{M}[\varphi'(p_{i.})h'(\sum_{i=1}^{M}\varphi(p_{i.}))]^2 p_{i.}$$
$$-[\sum_{i=1}^{M}p_{i.}\varphi'(p_{i.})h'(\sum_{i=1}^{M}\varphi(p_{i.}))]^2 > 0$$

There is a similar result in the stratified selection but using a different estimator for the variance:

$$\sigma^2 = \frac{1}{N}\sum_{k=1}^{r}N_k\{\sum_{i=1}^{M}[h'(\sum_{i=1}^{M}\varphi(p_{i.}).\varphi'(p_{i.})]^2\frac{N}{N_k}p_{ik}$$
$$-[\sum_{i=1}^{M}\frac{N}{N_k}p_{ik}h'(\sum_{i=1}^{M}\varphi(p_{i.}))\varphi'(p_{i.})]^2\}$$

Theorem 5 allows performing several statistical tests:

T1. H$_0$:H$_h^{\varphi}$(p) = D$_0$ the diversity of a population is constant. The test statistic is:

$$Z = \frac{\sqrt{n}(H_h^{\varphi}(p') - D_0)}{V} \sim \text{N(0, 1)}$$

where V is the standard deviation σ or σ_{St} when p is replaced by the estimation p'.

T2. H$_0$: H$_h^{\varphi}$(p)=H$_h^{\varphi}$(q) for two independent populations with the same variance. The test statistic is:

$$Z = \frac{H_h^{\varphi}(p') - H_h^{\varphi}(q')}{\sqrt{\frac{V_1^2}{n_1} + \frac{V_2^2}{n_2}}} \sim \text{N(0, 1)}$$

where V_i^2 is the variance σ^2 or σ_{St}^2 when p(q) is replaced by the estimation with the estimations p'(q').

T3. H_0: $H_h^\varphi(p^{(1)}) = H_h^\varphi(p^{(2)}) = \ldots = H_h^\varphi(p^{(m)})$ for m independent populations with the same variance.

These tests allow us to draw some conclusions about the entropy of a pseudorandom generator and even estimate this entropy. Moreover, we can even estimate the size of the encryption key that is generated by such a generator. The estimation method based on stratified selection is more efficient than in the case of simple selection because $\sigma_{St}^2 \leq \sigma^2$. These two estimation methods give the same variance if r=1 or

$$\sum_{i=1}^{M} \frac{N_k}{N} p_{ik} h'(\sum_{i=1}^{M} \varphi(p_{i.})).\varphi'(p_{i.})$$

is independent of k. A particular case is φ- entropy whose general expression is:

$$H_\varphi(p) = \sum_{i=1}^{M} \varphi(p_{i.})$$

For several functions φ(x) we get:

1) Shannon entropy (Shannon 1948): $\varphi(x) = -x \log x$
2) Genetic entropy (Latter 1973): $\varphi(x) = x - x^2 - x^2 (1-x)^2$
3) Hypo entropy (Ferrari 1980):

$$\varphi(x) = \frac{1}{M}(1+\frac{1}{\lambda})\log(1+\lambda) - \frac{1}{\lambda}(1+\lambda x)\log(1+\lambda x).$$

For these entropies, corresponding asymptotic results are obtained. It must be mentioned that similar results can also be obtained for relative entropy. Shannon's entropy will be used to define mutual information and the capacity of the communication channel.

III. SOURCES OF RANDOMENSS

A. DETERMINISTIC RANDOMNESS SOURCES

An easy way to produce „random" bits is based on an iterative algorithm, denoted by Gen:

$$x[t] = \text{Gen}[\text{x[t-1]}, \ldots \text{x[t-L]}],$$

which is setup-ed with some initial entropy bits:

$$\text{x[0]}, \ldots \text{x[L-1]]}.$$

From Shannon's point of view this type of randomness sources is not secure, thus is not used in application where it is needed a high level of confidence.

Using deterministic random bit generators, NIST has proposed in [1] some recommendations for the construction and usage of random number generators. The proposed procedures are based on the use of the HMAC concept and the block encryption algorithms.

In order to generate pseudorandom numbers, it is used a secret value (seed) and two functions that are hard to reverse for updating this value, respectively generating output.

NIST 800-90A (version from 2006) has some flows in the cryptographic definition of the dual elliptic curve DRBG [9]. The random numbers it has generated were slightly distorted, which raises the question of whether the NSA hides a secret backdoor in Dual_EC_DRBG. In 2013, internal notes released by Edward Snowden, a former NSA contractor, indicate that the NSA has indeed created a backdoor in Dual_EC_DRBG.

Also E. Snowden revealed the existence of NSA's Bullrun program. One of the purposes of Bullrun is *"to covertly introduce weaknesses into the encryption standards followed by hardware and software developers around the world."* The New York Times states that *"the NSA had inserted a back door into a 2006 standard adopted by NIST… called the Dual EC DRBG standard."* To restore confidence in encryption standards, NIST reopened the public analysis process for NIST SP 800-90A.

Because the elliptic curves are an algebraic structure, we will explain how to introduce hatches by using modular computation. This makes the presented scheme easier to understand. Let us assume that the PRG is specified by prime number p, and two integers g, h that are both less than p. The algorithm has an internal state s that satisfies $1 \leq s < p$. In one iteration, the following steps are performed:

$r = g^s \bmod p$, $s' = g^r \bmod p$ (update the state to s'),
$t = h^r \bmod p$ (output).

The designer of the algorithm will set g, h by a backdoor specified by a secret number e such that $g = h^e \bmod p$. Due to the complexity of the problem of the discrete logarithm, the direct connection between g and h cannot be proved. Knowing the secret hatch allows to find out the internal state of the generator. $t : t^e = (h^r)^e = h^{re} = (h^e)^r = g^r = s'$ (mod p).

B. NON DETETERMINISTIC RANDOMNESS SOURCES

An RNG must be specified by its security proofs, which are defined in relation threat models based on the impact assessment due to the likelihood of a threat being exploited by an identified vulnerability (risk analysis). The RNG limit must be designed to mitigate these threats, using physical and / or logical mechanisms.

There are several types of physical phenomena which are usually used to produce randomness: quantum based, thermal phenomena and clock drifts.

The quantum based methods for producing randomness are usually based on: Poisson noise (quantum mechanical noise source in electronic circuits, the nuclear decay of a radiation source, photons traveling through a semi-transparent mirror, amplification of the signal produced on the base of a reverse biased transistor, fluctuations in vacuum energy (based on Heisenberg's energy-time uncertainty principle).

Examples of thermal phenomena are thermal noise from a resistor, avalanche noise (generated from an avalanched diode) and atmospheric noise. There are several other related wonders where a clock does not run at the very same rate as a reference clock.

In order to validate the RNG, it is necessary to fulfill certain security requirements. Some of the security requirements are the following:

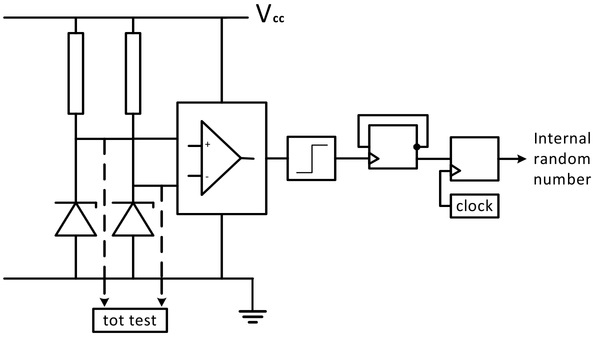

FIGURE 1. A RNG based on Zener diode.

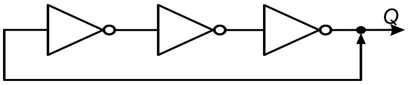

FIGURE 2. A ring oscillator with 3 gates.

- resistance to prediction (the tradeoff previously or present will not bargain the future yield) of the RBG;

- backtracking resistance (the compromise at some point will not compromise the past) of the RBG outputs.

Also it is recommended to implement the additional measures:

- the output will be modified by an approved post-processing method;

- performing validation and health testing.

1) ZENER BASED

Using the properties of the electromagnetic field, an opponent can manipulate the behavior of a Zener diode. Solutions based on two diodes mitigate this type of attack. In [17] it is proposed a solution based on two Zener diodes with a reverse avalanched effect (fig. 1). The outputs of the two diodes constitute the input to an operational amplifier that amplifies the voltage difference. The output of the operational amplifier is fed into a Schmitt trigger. The mean voltage of the amplifier output signal is about the middle of the two threshold values of the Schmitt trigger.

Because of the precarious edges of the information and use of the 0-1-upcrossings, only the hysteresis impact ought to be immaterial. The proposed structure just endeavors 0-1-intersections, the yield sign of the Schmitt trigger comprises of zeros and ones. The proposed stochastic model is the time lengths of these 0-1 switching.

2) RING OSCILATOR

A natural model for jittered oscillators is based on the fact that half-periods of the durations $X_k = T_{k+1} - T_k$, between the flipping times T_k ($k \geq 0$) of the signal, are independent and identically distributed random variables. Another way is to consider a family of model where the phase φ of an oscillator is analogue to a (stationary) one-dimensional Brownian motion. A ring oscillator (fig. 2) is a gadget made out of an odd number of NOT gates in a ring, whose output oscillates between two voltage levels, representing true and false.

In a ring oscillator TRNG each oscillator's phase relative to the sample clock drifts over time. Most of the drift is due to the difference between oscillator and sampling frequencies, but some is caused by jitter. In 2007 Sunar proposed in [23] a random number generation structure based on 114 ring oscillators, which at that time was appreciated, considering that it was easy to implement using a FPGA structure. Even though it was designed on a security model, it was based on two difficult to meet hypotheses: the independence of the rings and the fact that the modulo 2 (XOR) summations operation is performed fast enough to maintain the entropy generated by the rings. Without a rigorous demonstration of security [24], [25], the Sunar model is nothing more than one of the generators that pass the statistical tests [4], [7], [8] without having a guaranteed minimum entropy.

3) QUARTZ

Some general techniques to generate numbers are based on quartz crystals located on the mother board and/or others hardware devices for operation and timing.

4) MRAM

Another way to produce random numbers is by using the random switching behavior of Magnetic Tunnel Junctions under low write current [22].

5) A PROPOSAL FOR TRNG

Given that there are several categories of models on which we can build a TRNG, one solution is to implement all four technologies (Zener diode, ring oscillator ring, Quartz and MRAM) to increase the degree of entropy of the solution. How they can be combined requires a separate analysis. A possible solution, with demonstrable security, is to carry out XOR between the outputs of the four categories of generators. However, this approach causes the generator to take the speed of the slowest device, which is why it is necessary for the four types of technologies to have balanced speeds.

6) POST PROCESSING

The output of the random number generator may have a certain deviation from the ideal value of 0.5 (if we are talking about random generators of binary sequences). In this situation, to "center'' the generator, we use post processing after the output. Examples of such post processing can be data block encryption using a block encryption algorithm (a situation that allows output to lead to input data) applying a non-invertible function (such as the von Newman decorrelation procedure: sequences 00 and 11 are discarded and the sequences 01 and 10 are transformed into 1, respectively 0. The process is fast, considering the fact it halves the amount of output data).

IV. VALIDATION OF RANDOMNESS AND STANDARDS

According to Viktor Fisher [15] there are two types of approaches among in order to study this problem: mathematical and physical. In the mathematical approach we considered

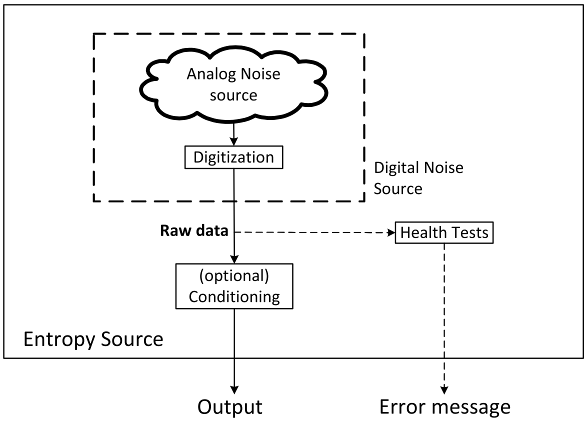

FIGURE 3. General model of entropy source [2].

an ideal TRNG and focus on obtaining entropy rate of ten bits per trial. In the physical approach we need to say what can be the frequency of trials and what (physically) means 'fair tossing' and 'fair coins'. The design of a TRNG is rather a physical than a mathematical project. The physical parameters of the source of randomness must be thoroughly evaluated: distribution of random values (bias), correlation, dependence (if there are many sources involved), manipulability and agility (spectrum). In [2] NIST made recommendations for the entropy sources used for random bit generation. For a set $A=\{x_1,x_2,\ldots,x_k\}$, with probability $Pr(X=x_i) = p_i$ for $i = 1,\ldots,k$, the min-entropy [13] of an independent discrete random variable X is defined as:

$$H = \min(-\log_2 p_i) = -\log(\max p_i).$$

NIST proposed in SP 800-90B [2] the model presented in figure 3. To validate the compliance of the entropy source with the standard requirements, after collecting the data, we are in the situation of deciding whether the analyzed samples come from independent and identically distributed random variables (IIDs). In the situation the analyzed samples come from IID variables, the evaluation methodology is relatively simple as opposed to the case where these variables are not IIDs.

Health tests are an integral part of the design of the entropy source and ensure that the noise source works according to the specifications. A reference standard for estimating the entropy quantity used by a random number generator is specified in SP 800-90C [3]. In the favorable situation when the random variables are IID, the min-entropy is estimated by the most common value estimate. In reality, many of the noise sources do not produce random IID variables and we use in this case a complex set of entropy tests. The tests are not independent and they do not overestimate the entropy at a significance level of 0.005. The source code for min-entropy estimation is available on [16].

A Common Criteria [18] approach regarding functionality classes for random number generators is proposed by Wolfgang Killmann and Werner Schindler in [17]. This approach helps developers to design random number generators, which are CC compliant and can be subject of evaluation by a CC accredited laboratory. The proposed statistical tests are similarly to NIST SP 800-22 [4], [7], [8], described in section 5.

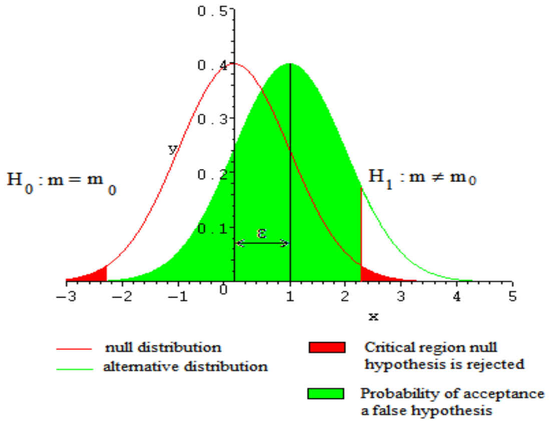

FIGURE 4. The critical region of a statistical test with $\alpha = 0.01$.

For security validation, two categories of standard ISO 19790 (similar to FIPS 140-2) and ISO 15408 (similar to Common Criteria) are used. The two standards differ in the approach to validation: the ISO 19790 standard is a qualitative standard and the ISO 15408 quantitative standard in terms of security justifications.

V. STATISTICAL TESTS USED FOR VALIDATION

The validation of the statistical hypotheses is carried out through the samples and has a determining role in the decision process regarding the parameters of the theoretical distribution of a population (most often these parameters are the average or the population dispersion).

In the case of statistical testing of the cryptographic algorithms, the samples are obtained from the output of the algorithm having plain text and keys strongly auto correlated or correlated. We model the decision making-process with two statistical hypotheses: the null hypothesis denoted by H_0 - in this case, the sample does not indicate any deviation from the theoretical distribution - and the alternative hypothesis H_A - when the sample indicates a deviation from the theoretical distribution. Due to the sample selection mode, the statistical estimates are subject to measurement errors. These are of two types: the probability of rejecting a true hypothesis, respectively the probability of accepting a false hypothesis. If figure [3] we present a graphical interpretation of the connection between these two types of errors.

The effective implementation of a statistical test includes, among others, decision rules for rejecting the null hypothesis. These rules can be described in two ways: decisions based on P-value, respectively decisions based on "critical region". Because the sample data are aggregated to be reported at the significance level of the statistical test, it is necessary to specify (calculate) the minimum sample volume. At the same time, the data that make up the sample must be obtained

TABLE 1. Reference distribution of NIST statistical tests.

Test	Distribution
Frequency (Monobit) Random Excursions Variant	half normal
Runs Discrete Fourier Transform Maurer's "Universal Statistical" Cumulative Sums (Cusum)	normal
The Longest Run of Ones in a Block Frequency within a Block Non-overlapping Template Matching Overlapping Template Matching Linear Complexity A Approximate Entropy	χ^2
Binary Matrix Rank	$\chi^2(2)$
Random Excursions	$\chi^2(5)$
Serial	$\chi^2(2^{m-1})+\chi^2(2^{m-2})$

as achievements of randomly distributed and identically distributed variables. One of the reference standards for testing the quality of binary strings is the NIST Statistical Test Suite standard specified in SP 800-22. The standard is composed of fifteen statistical tests, which can reveal various deviations from randomness of the binary strings. Within the standard are specified the reference implementations, the test vectors for them, the mathematical justification of the tests, as well as a series of test strategies.

Depending on the sample analyzed, the function of the test f is determined for each test and compute the P-value $=\Pr(f|\mathrm{H}_0)$ that summarizes the strength of the evidence against the null hypothesis. If the P-value $> \alpha$, then H_0 is accepted otherwise reject H_0.

In the specialized literature there are some comments about possible weaknesses (or need clarification) within the methodology of statistical testing SP 800-22: quantifying errors, the power of the test suite, statistical assumptions, the test dependencies/correlations [14] and inadmissible tests.

The tests specified in SP 800-22 are not independent, which makes it hard to calculate a general rejection rate (test power) [11], [12]. Table 1 presents the statistical distribution used in each test.

We assume that if we sum all the P-values of the statistical tests we shall obtain a normal distribution. As we can see, there are three types of distributions: normal, half normal and χ^2. If we suppose that all the χ^2 distributions are independent and compute the sum of all corresponding P-values of these distributions, we shall obtain a χ^2 distribution with the number of degree freedom greater then 30, which is well approximated by the normal distribution. Thus, if we presume that all the statistical tests are independent, then the sum of all P-values will go after the normal distribution. This overall distribution will be the distribution of the hole NIST statistical test suite.

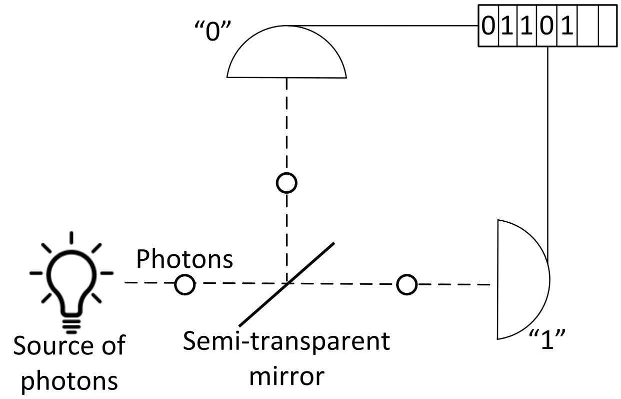

FIGURE 5. Quantis RNG principle [5].

In [10] is presented the extension of the NIST statistical tests to an arbitrary level of significance α, also being computed the value of the second error order probability.

As an example, for the frequency test we have for n>30 the following formula for second error probability:

$$\beta = \phi\left(\sqrt{\frac{p_0q_0}{p_1q_1}}\left(u_{1-\frac{\alpha}{2}} - \frac{n(p_1-p_0)}{\sqrt{np_0q_0}}\right)\right) - \phi\left(\sqrt{\frac{p_0q_0}{p_1q_1}}\left(u_{\frac{\alpha}{2}} - \frac{n(p_1-p_0)}{\sqrt{np_0q_0}}\right)\right).$$

VI. REAL LIFE

A. QUANTIS-QUANTIC TECHNOLOGY

Quantis [5] is a group of equipment arbitrary number generators which abuse basic quantum optical procedures as a source of true randomness. Quantis RNG (QRNG) abilities are:

- true hardware random number generator;
- trusted and certified source of quantum randomness;

-consistent status check and disappointment recognition component;

- instant entropy, adaptable for various applications;
- propelled functionalities, for example, scaling and randomness extraction.

Device is largely used in different sectors: Swiss Lottery, IDQ-Certes Solution Brief (security solution), PokerMatch, NS&I (banking sector).

QRNG is based on quantum physics: photons - light particles - are sent one by one onto a semi-transparent mirror and detected. The exclusive events (reflection - transmission) are associated to ≪ 0 ≫ - ≪ 1 ≫ bit values.

Quantis is the most guaranteed genuine RNG in the market. It has effectively passed the accompanying accreditations or government approvals: NIST SP800-22 Test Suite Compliance, METAS Certification [19], CTL Certification, Several iTech Labs singular Certificates and consistence with the BSI's AIS31 standard.

B. INTEL TRUE RANDOM NUMBER GENERATOR

The Random Number Generator (BA431) is a basic IP center for all FPGA and SoC structures that target cryptographically

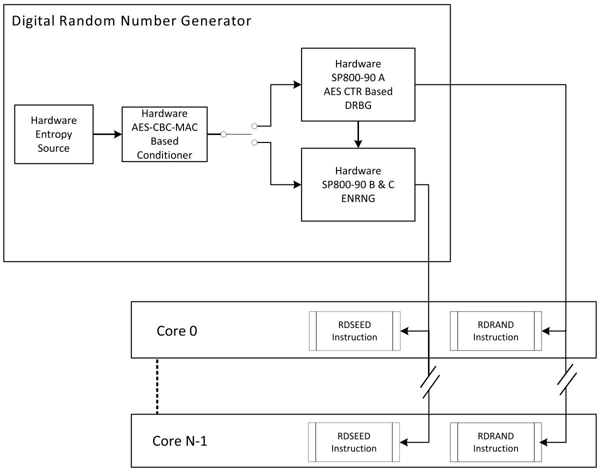
Digital Random Number Generator
Hardware Entropy Source
Hardware AES-CBC-MAC Based Conditioner
Hardware SP800-90 A AES CTR Based DRBG
Hardware SP800-90 B & C ENRNG
Core 0
RDSEED Instruction
RDRAND Instruction
Core N-1
RDSEED Instruction
RDRAND Instruction

FIGURE 6. Intel Digital Random Number Generator design [20].

made sure about applications. The BA431 incorporates a True Random Generator (TRNG) as the source of entropy. The discretionary Deterministic Random Bit Generator (DRBG) can be furnished with the core. The entropy source (yield an irregular stream of bits at the pace of 3 GHz.) and the DRBG are intended for consistence with the NIST 800-90A and NIST 800-90B. The entropy source runs asynchronously on a self-planned circuit and uses thermal noise within the silicon [20], [21].

It is easily portable to any Intel FPGA device (including SoC). The IP core effectively passes AIS31 and NIST 800-22 test suites and has passed FIPS 140-2 certification. A portion of the end advertise for the utilization of this gadget is on car, communication, computer and storage, purchaser, industrial, medical, military, test and measurement, wireline and so forth.

C. QUINTESSENCE LABS

Starting from the Kerckhoff principle, the strength of a cryptographic system depends only on the key, the device for generating random numbers qStream designed by Quintessence Labs [6] produces with a speed of 1Gbit / s maximum entropy cryptographic keys. The device is built on quantum principles. qStream provides random numbers for generating cryptographic keys, as well as other critical security parameters. Other applications include Entropy as a Service (EaaS), simulations, modeling and computer games. Operating systems use algorithmically generated random numbers. Random numbers are used to start the operating system as well for performing cryptographic operations (SSL / TLS, SSH or PKI protocols). The entropy pool ensures the generation of random numbers considering the fact that the programs on the computer are deterministic. Each operating system has an entropy pool (without limiting the generality we will refer to the LINUX operating system). This entropy pool is powered by various random sources such as motion mouse. The entropy pool has 4096 random bits (the highest possible entropy) which makes a private key of 2048 bits random. The kernel is responsible for maintaining an acceptable level of this pool. More precisely, as the bits in the pool are used the entropy is reduced, but if the operating system finds good random events the pool is full. qRand developed by QuintessenceLab is a daemon for Linux systems that is configured to monitor requests for randomness and if entropy falls below a specified limit qRand full the pool back quantum random number generator.

VII. CONCLUSION

Random numbers are necessary ingredients in ensuring information security in electronic format. Their generation is accomplished through techniques and hardware means and involves validating the quality (level of disorder) of the noise source. The disorder level estimation (entropy) is performed through statistical tests.

REFERENCES

[1] *Recommendation for Random Number Generation Using Deterministic Random Bit Generators*. Accessed: Dec. 23, 2019. [Online]. Available: https://csrc.nist.gov/publications/detail/sp/800-90a/rev-1/final

[2] *Recommendation for the Entropy Sources Used for Random Bit Generation*. Accessed: Dec. 14, 2019. [Online]. Available: https://csrc.nist.gov/publications/detail/sp/800-90b/final

[3] *Recommendation for Random Bit Generator (RBG) Constructions*. Accessed: Nov. 10, 2019. [Online]. Available: https://csrc.nist.gov/csrc/media/publications/sp/800-90c/draft/documents/draft-sp800-90c.pdf

[4] *A Statistical Test Suite for Random and Pseudorandom Number Generators for Cryptographic Applications*. Accessed: Jan. 10, 2020. [Online]. Available: https://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-22r1a.pdf

[5] *Random Number Generation White Paper*. Accessed: Nov. 29, 2019. [Online]. Available: https://marketing.idquantique.com/acton/attachment/11868/f-0226/1/-/-/-/-/What%20is%20the%20Q%20in%20QRNG_White%20Paper.pdf

[6] *Quantum Random Number Generator Sheet*. Accessed: Jan. 10, 2020. [Online]. Available: https://www.quintessencelabs.com/wp-content/uploads/2016/02/qStream_RNG_Product_Sheet.pdf

[7] *NIST Standards*. Accessed: Dec. 5, 2019. [Online]. Available: http://www.nist.gov/

[8] *Randomness Testing of the Advanced Encryption Standard Candidate Algorithms*, document NIST IR 6390, Sep. 1999.

[9] D. R. L. Brown and K. Gjosteen, "A security analysis of the NIST SP 800-90 elliptic curve random number generator," Cryptology ePrint Archive, Tech. Rep. 2007/048. Accessed: Oct. 10, 2019. [Online]. Available: https://www.iacr.org/archive/crypto2007/46220459/46220459.pdf

[10] A. Oprina, A. Popescu, E. Simion, and Gh. Simion, "Walsh-Hadamard Randomness Test and New Methods of Test Results Integration," *Bull. Transilvania Univ. Bragov*, vol. 2, no. 51, pp. 93–106, 2009.

[11] C. Georgescu and E. Simion, "New results concerning the power of NIST randomness tests," *Proc. Romanian Acad. A*, vol. 18, 2017, pp. 1–8.

[12] S. Murphy, "The power of NIST's statistical testing of AES candidates," Tech. Rep., Jan. 2000. Accessed: Oct. 18, 2019. [Online]. Available: http://www.ma.rhul.ac.uk/~sean/StatsRev.pdf

[13] J. Kelsey, K. A. McKa, and M. Sönmez Turan, "Predictive models for min-entropy estimation," *Hardware and Embedded Systems*, vol. 9293, T. Gäneysu and H. Handschuh, eds. Berlin, Germany: Springer, 2015.

[14] A. Doäanaksoy, F. Sulak, M. Uäuz, O. Åžeker, and Z. Akcengiz, "Mutual correlation of NIST statistical randomness tests and comparison of their sensitivities on transformed sequences," *TURKISH J. Electr. Eng. Comput. Sci.*, vol. 25, pp. 655–665, 2017.

[15] V. Fisher, "Sources of randomness in digital devices and their testability," in *Proc. NIST Random Bit Generation Workshop*, 2016. Accessed: Oct. 5, 2019. https://csrc.nist.gov/csrc/media/events/random-bit-generation-workshop-2016/documents/abstracts/viktor-fischer-abstract.pdf

[16] *Source Code*. Accessed: Oct. 10, 2020. [Online]. Available: https://github.com/usnistgov/SP800-90B_EntropyAssessment

[17] W. Killmann and W. Schindler, "A design for a physical RNG with robust entropy estimators," *Cryptographic Hardware and Embedded Systems*, vol. 5154, E. Oswald and P. Rohatgi, Eds. Berlin, Germany: Springer, 2008.

[18] *A Proposal For: Functionality Classes for Random Number Generators*. Accessed: Dec. 15, 2019. [Online]. Available: https://cosec.bit.uni-bonn.de/fileadmin/user_upload/teaching/15ss/15ss-taoc/01_AIS31_Functionality_classes_for_random_number_generators.pdf

[19] Accessed: Dec. 19, 2019. [Online]. Available: https://www.metas.ch/metas/en/home.html

[20] Accessed: Dec. 5, 2020. [Online]. Available: https://software.intel.com/en-us/articles/intel-digital-random-number-generator-drng-software-implementation-guide

[21] Accessed: Jan. 3, 2020. [Online]. Available: https://www.intel.com/content/www/us/en/programmable/solutions/partners/partner-profile/barco-silex/ip/true-random-number-generator–trng-.html

[22] K. Yang, Q. Dong, Z. Wang, Y.-C. Shih, Y.-D. Chih, J. Chang, D. Blaauw, and D. Svlvester, "A 28NM integrated true random number generator harvesting entropy from MRAM," in *Proc. IEEE Symp. VLSI Circuits*, Jun. 2018, pp. 171–172.

[23] B. Sunar, W. Martin, and D. Stinson, "A provably secure true random number generator with built-in tolerance to active attacks," *IEEE Trans. Comput.*, vol. 56, no. 1, pp. 109–119, Jan. 2007.

[24] N. Bochard, F. Bernard, V. Fischer, and B. Valtchanov, "True-randomness and pseudo-randomness in ring oscillator-based true random number generators," *Int. J. Reconfigurable Comput.*, vol. 2010, pp. 1–13, 2010.

[25] M. Baudet, D. Lubicz, J. Micolod, and A. Tassiaux, "On the security of oscillator-based random number generators," *J. Cryptol.*, vol. 24, no. 2, pp. 398–425, Apr. 2011, doi: 10.1007/s00145-010-9089-3.

EMIL SIMION graduated from the Faculty of Mathematics, University of Bucharest, in 1994. He received the master's and Ph.D. degrees from the University of Bucharest, in 1995 and 2000, respectively, with a thesis dedicated to the statistical analysis of cryptographic systems.

From 1994 to 2014, he worked as a Researcher at the Advanced Technologies Institute. From 2014 to 2016, he was a Lecturer with Paris 2, Sorbonne. Since 2012, he has been an Associate Professor with the University Politehnica of Bucharest, Romania. He published over 30 articles in journals or conference proceedings.

Dr. Simion is an Active Member of International Association Cryptographic Research.