Recommendations for Design and Validation of a Physical True Random Number Generator Integrated in an Electronic Device

David Lubicz, Viktor Fischer

Recommendations for Design and Validation of a Physical True Random Number Generator Integrated in an Electronic Device

Version 1.0

David Lubicz
DGA and IRMAR, France
david.lubicz@univ-rennes1.fr

Viktor Fischer
Hubert Curien Laboratory, UJM Saint-Etienne, France
fischer@univ-st-etienne.fr

January 10, 2024

Abstract

In this Recommendations, we describe essential elements of the design of a secure physical true random number generator (PTRNG) integrated in an electronic device. On the basis of these elements, we describe and justify requirements for the PTRNG design, validation and testing, which are intended to ensure security of the generator aimed at cryptographic applications.

1 Introduction

Random number generators (RNGs) represent essential part of each cryptographic equipment. In particular, they are used to generate keys, key identifiers, initialization vectors, nonces, but also to protect cryptographic equipment against attacks using side channels like power consumption, electromagnetic emanation, sound, etc. Random number generators are important because they are the only source of diversity in cryptographic algorithms, a diversity whose combinatorial

richness characterized by entropy (Shannon entropy or min entropy) constitutes the only protection against brute force attacks.
In general, most of the state-of-the-art RNGs are hybrid: they consist of a true random number generator (TRNG) and a preudo-random number generator (PRNG). A TRNG, which exploits some physical random phenomena to guarantee unpredictability, reseeds periodically a deterministic PRNG, which must be cryptographically secure, i.e. it must exploit a cryptographic mode with a proven security (see [1]) or use an approved cryptographic algorithm (see [2]).
The PRNG ensures security of the generator based on computational assumptions if the source of randomness in the generator fails to operate correctly. However, this is accepted only for some short time interval depending on targeted security level and quantity of entropy accumulated in the PRNG (the PRNG must contain enough entropy, which was accumulated before the failure of the physical source of entropy).
This document concerns only the TRNG part of an RNG. We define and describe essential elements of a TRNG design approach that ensures security of the generator and takes the most recent advances of a RNG design into account. Indeed, it may be possible to design a generator for which it would be difficult to construct a real attack, even without adopting the recommended approach. However, we think that for devices such as random number generators which guarantee security in cryptographic applications, the developer has to provide a security demonstration, and we believe the proposed approach provides the strongest assurance together the less possible constraints on the design.
Similarly to cryptographic applications, in which the designer has to prove security of protocols on widely accepted computational assumptions, we want to prove unpredictability of generated numbers by estimating a lower bound of the entropy rate of the TRNG upon well established assumptions about the randomness of some identified physical phenomenons. For this, we consider an attacker which may have full knowledge of the TRNG design and can observe its output. We make no assumption on its computational power, i.e. it can be theoretically unlimited.
Unidentified phenomena may contribute to the random nature of the operation of the TRNG. This is evidenced for example by the statistical tests of the generator output bits which may show that the output bits have higher entropy rate than could have been predicted taking into account only identified physical noises. However, in the framework of our approach, we do not take into account these unidentified noises because it is not possible to evaluate their contribution to the security of the generator. In the following, in order to distinguish the entropy rate produced by identified phenomena from the rest of the entropy, we will call it the *proven entropy*.

We believe that a TRNG which could not, by design, be suited to such an analysis, should be considered as improper to be used in high security cryptographic applications. Unfortunately, the security analysis of numerous constructions published in the literature is based on dubious arguments if analyzed as recommended in this report.
Our approach is described in a series of requirements that are well-founded and argued[1]. We provide definitions that appear important and are perhaps insufficiently clarified in the literature: for example, that of the stochastic model of the physical noise that we distinguish from the stochastic model of the entire generator. We specify in what testing conditions the statistical tests can be used (essentially when a stochastic model of the generator is available). If these conditions are not fulfilled, the statistical tests are not useful and can even lead to incorrect conclusions.
Each requirement or concept is illustrated with the particular case of a ring oscillator based elementary random number generator. Such an elementary generator, which is described in Appendix A, has the advantage of being easy to describe and implement on all types of supports (FPGA and ASIC, including all CMOS technologies) as well as a structure that is among the most studied and best understood among those described in the literature (see [3] for an overview of TRNGs using free running oscillators as a source of randomness). Presented example of application of ring oscillators confirms rigorosity of the chosen approach, which remains compatible with the production requirements of a data security device including random number generator.
The document has the following structure. In Section 2 we give the general objectives of the evaluation process described in the document and we introduce basic definitions and requirements regarding the TRNG design. Section 3 regroups definitions and requirements for the model of the source of randomness. Section 4 gives definitions and requirements for fitting of the model with the generator. In Section 5 we specify objectives and kinds of required embedded tests. The proposed PTRNG design and evaluation approach is compared with methodologies required by German AIS20/31 and American NIST SP 800-90B standards in Section 6. We conclude the document in Section 7.

2 TRNG design – definitions and requirements

In most cases, the design of a physical true random number generator and its security evaluation are performed by independent institutions and companies.

[1] As widely accepted, the requirements of these Recommendations are indicated in the rest of the document by the word “shall”.

The certification bodies give a limited number of licenses to laboratories that are empowered to perform security evaluations. In order to simplify the evaluation process, the TRNG designer, the evaluation laboratory and the certification body should use the same vocabulary and identical definitions and terms.
The design and certification process should ensure the highest level of confidence in the design and security of the TRNG aimed at cryptographic applications while giving the designers the maximum freedom to conceive a TRNG, which fitts in with the particular constraints of their projects.
Since the security proof we want to achieve is based on a mathematical model of the TRNG, one of the main difficulties of the security evaluation process is to guarantee accordance of the model with respect to the reality. To achieve this objective, it is very important to be able to verify how the main building blocks of the model of the TRNG fit with the structure of the TRNG itself.
To make this task easier, we regroup our definitions and requirements in four categories:

- [**Design**] which deals with the principle of the TRNG itself;
- [**Model**] which deals with the stochastic model of the TRNG;
- [**Model fitting**] which deals with the verification of the model fitting in with the generator;
- [**Tests**] which deals with embedded testing strategies of the generator.

It is quite clear that the operation of a TRNG, whose essential characteristic is to produce a series of unpredictable bits or binary numbers, must necessarily rely on unpredictable physical phenomena, that we call in the following the *physical noises*, the outcome of which must be converted into a series of bits or numbers.

This motivates the following general definitions:

Definition 1. ([**Design**] **Source of randomness**) The source of randomness is an uncontrollable physical random phenomenon (or random phenomena), which is (are) transformed to electric signals featuring some random analog components – amplitude and/or phase.

Definition 2. ([**Design**] **Core of randomness**) The core of randomness is a physical area, i.e. an electronic or opto-electronic device (or its part), featuring the source (or sources) of randomness, which is delimited by a well defined *security boundary*. It is characterized by its *internal state* $E(t)$, which evolves depending on *physical random phenomena* appearing inside this area (*physical noises*).

Remark 1. As stated in the definition, the physical core of randomness is a device or its part, in which some random phenomenon or random phenomena appear. The security boundary of this area must be well defined by the designer. At first glance, it could seem that this boundary can be chosen arbitrarily. For instance, it is always possible to consider that several cores of randomness constitute a unique bigger core. Nonetheless, we will see that the interpretation of many requirements will depend on the choice of the boundaries of the cores, which include sources of randomness. These requirements will be, most of the time, easier to fulfill by choosing the smallest possible security boundary for the physical cores even if it means splitting a big core into smaller and more elementary ones. Nevertheless, the designer can chose freely the boundary of the core of randomness as long as he can fulfill all the requirements.

Example 1. *The TRNG from Appendix A features two physical cores of randomness: the oscillators* Osc_i *(*$i = 0, 1$*), which produce clock signals* $s_i(t) = f(\phi_i(t))$*. The internal state of the oscillator* Osc_i *at time* t *is characterized by its current phase* $\phi_i(t)$*. The phase* $\phi_i(t)$ *of* Osc_i *is affected by different noise sources (thermal noise, flicker noise etc.) producing the phase jitter. The source of randomness in the cores of randomness* Osc_i *are the phase jitters depending on* $\phi_i(t)$*.*

Definition 3. ([**Design**] **Physical true random number generator**) In the context of this document, a physical true random number generator (PTRNG) is a device using one or several core of randomness to generate random numbers.

Remark 2. We assume that the PTRNG output value can be computed by an attacker, assuming that he can know the internal state of the cores of randomness (see Definition 7 of the PTRNG security model). It means that the security of the PTRNG (and therefore its unpredictability) relies solely on the sources of randomness affecting the core of randomness. As a consequence, in the security assessment of a PTRNG, only a specified list of sources of randomness shall be considered and contribution of all other sources must be neglected.
It should be noted that this assumption does not imply that the output of the PTRNG is guessable from the internal states of the cores of randomness also for the designer. In other words, we assume that the attacker can have more computational power than the designer. So the best possible TRNG design is the one, for which the attacker and the designer have the same (unlimited) knowledge, i.e. for which the output bit is guessable from the internal states of the cores of randomness.

2.1 General TRNG structure and its basic parts

In the vast majority of cases, the physical random phenomena used in PTRNGs are analog. The mechanism performing analog-to-digital conversion is thus an integral part of the generator. Accordingly, we distinguish four basic PTRNG blocks (entities) as presented in Fig. 1:

- Core(s) of randomness, which include source(s) of randomness,
- Analog-to-digital converter (ADC),
- Post-processor,
- Embedded tests.

The PTRNG contains in general N cores of randomness that produce electric signals featuring some random component (amplitude and/or phase). These signals are converted into a stream of digits (bits or vectors of bits) by an analog-to-digital converter (ADC). Note that this conversion can be made intrinsically inside the core of randomness. In that case, the ADC block is represented by an identity function, in order to maintain the general structure of the PTRNG.
The ADC outputs a stream of random numbers (bits or multi-bit values), which can feature low statistical quality (e.g. low entropy). In particular, it can feature some deterministic pattern. This low statistical quality can be enhanced, if necessary, by an algorithmic post-processor to obtain a high-quality digital noise.

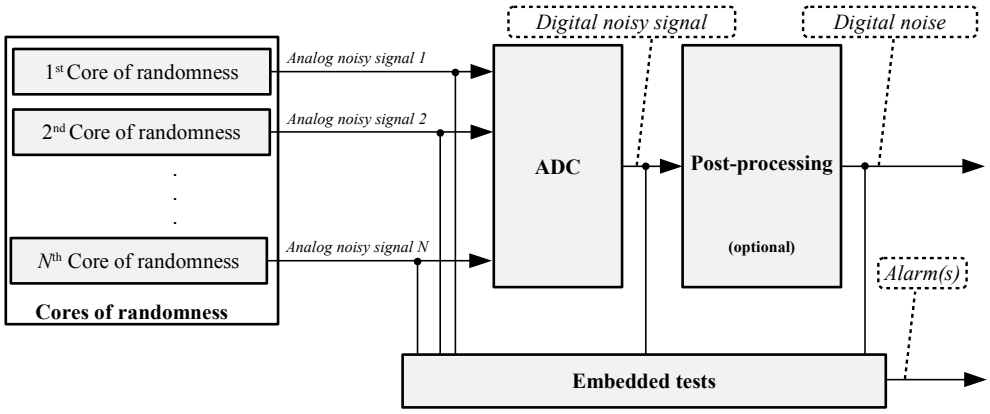

Figure 1: *General structure of a TRNG*

The quality of the generated random numbers is tested continuously by embedded tests. Embedded tests are performed according the pre-defined testing procedures. At least two testing procedures shall be defined: the Startup procedure and the Continuous testing procedure. If some kind of tests cannot be executed continuously (e.g. a known answer test of some deterministic block), an On demand testing procedure can complete the previous two testing strategies.

Requirement 1. ([**Design**] **Identification of the main TRNG entities**)
The four main entities of the generator shall be clearly identified.

Example 2. *Two cores of randomness appear in the elementary oscillator-based TRNG (EOTRNG) presented in Appendix A: two oscillators generating jittery clocks. The ADC is composed of two parts: the frequency divider and the sampler. The division factor of the frequency divider can be chosen so that a post-processing is not needed (enough entropy is accumulated during each sampling period). The parametric embedded tests scrutinize two generated clock signals and the generator output. They measure TRNG parameters, which are used as input parameters of the stochastic model: the phase jitter drift and volatility and the duty cycle of the sampled oscillator.*

Requirement 2. ([Design] **Identification of sources of randomness**) All the sources of randomness exploited in the RNG shall be identified and the security boundary of the cores of randomness featuring the sources shall be clearly determined.

Requirement 3. ([Design] **Security boundary of the core of randomness**) It shall be ensured that it is impossible for an attacker to breach the security boundary of each core of randomness.

Requirement 4. ([Design] **Unmanipulability of the source(s) of randomness**) It shall be proved that the physical random phenomena, which affect the evolution of the internal state of the core of randomness can not be influenced from outside of the security boundary or at least that any tentative manipulation would be detected by embedded tests. In particular, the physical part of the generator ensuring randomness (the core of randomness) shall not have any inputs affecting the source of randomness.

Example 3. *In the EOTRNG from Appendix A, the jitter of the two clock signals is composed of both deterministic and non-deterministic (random) component. The global deterministic component is reduced to a negligible size by using a differential principle: two identical oscillators are influenced in the same way by global deterministic signals. We assume that in normal operating conditions of semiconductor devices, the thermal noise is always present. The entropy shall then be estimated in the corner operating conditions, in which the contribution of the thermal noise to the clock jitter is minimal and cannot be further reduced during attacks. Oscillators do not have any inputs.*

Remark 3. Requirement 4 implies in particular that the robustness of all sources of randomness shall be thoroughly analyzed and evaluated.

Example 4. *In the EOTRNG from Appendix A, the randomness in the generator output depends on the relative phase jitter of the two oscillators. In general, the two oscillators are not fully independent since for instance they may share the same power supply and are subject to cross talks. Nonetheless, it is widely accepted that sources of thermal noises are independent so that if we only consider the thermal noise component of the phase jitter, the two sources of randomness in the EOTRNG can be considered to be independent.*

Definition 4. ([**Design**] **Analog-to-digital converter**) The analog-to-digital converter (ADC) is the entity that transforms noisy analog signal component into a stream of random numbers (e.g. a bit stream).

The conversion of analog signal components to digital signals (numbers) is supposed to be guessable by an attacker.

Requirement 5. ([**Design**] **Specification of the ADC**) The ADC block shall be clearly identified and its behavior shall be described mathematically.

Remark 4. If the analog-to-digital conversion is fully deterministic, it can be easily described by a mathematical function. If its behavior is partially random (e.g. its less significant bits), at least a statistical description should be provided for instance by a stochastic function.

Example 5. *In the case of the elementary generator from Appendix A, the ADC, which transforms the random phase of the clock signal to a random binary value, is composed of the frequency divider and the sampler.*

Definition 5. ([**Design**] **Post-processor**) The post-processor is a deterministic block aimed at enhancement of statistical parameters of the generator. It shall not reduce the entropy rate per bit.

The ADC output is post-processed in order to enhance the statistical properties of the generated binary signal, e.g. to increase its entropy rate per bit. The post-processing of the ADC output is not mandatory. Indeed, if the entropy rate per bit is sufficient, a post-processing is not needed.

Requirement 6. ([**Design**] **Specification of the post-processing algorithm**) The post-processing algorithm and its objective shall be clearly identified. The designer shall show that the selected post-processing algorithm does not decrease the entropy rate per bit.

3 Stochastic model – definitions and requirements

3.1 Model of the source of randomness

A very important design requirement for a PTRNG suitable for a provable security approach is the ability to identify and distinguish the sources of randomness from the rest of the PTRNG, since sources of randomness and the whole generator should be analyzed in a different manner. This motivates the following definition which represents the main design assumption in our approach:

Definition 6. ([**Model**] **Random number generator**) A random number generator is a device G composed of one or several core of randomness with global internal state $E : t \to V$ depending on time t, with the value in a space of states V and producing a series of bits $b_1(t_1)b_2(t_2)\ldots$ at given times.

The security model of the PTRNG is defined based on the definition of the attacker:

Definition 7. ([Model] **Security of a Random number generator**) We consider an attacker, which has the full knowledge of the PTRNG design and can freely observe its output. We make no assumption regarding his computational power, which can be even unlimited. The attacker tries to predict output bits of the PTRNG. We measure his effectiveness with the entropy rate of the PTRNG, which is the average amount of information one has to give to the attacker so that it can predict the generator's output bits. We assume that the output bit produced at time t can be perfectly determined by the attacker from the knowledge of the internal state $E(t)$.

The knowledge of the attacker regarding the state of the PTRNG can be described by a statistical distribution $p_t(x)$ on the space of states V. The evolution of $p_t(x)$ in time depends on two things:

- The physical noises which tend to decrease the attacker's knowledge regarding the state of the generator;
- Output bits which allow the attacker to get some information about the internal state of the generator.

As the incertitude the attacker has on the bits produced by the PTRNG comes necessarily from the incertitude on $E(t)$, in order to be able to compute the entropy rate at the PTRNG output, it is necessary to:

- have a law describing the effect of the physical noise on the evolution of $p_t(x)$;
- ensure that this law is stable with respect to time and environmental conditions and be able to follow continuously fluctuations of its parameters;
- have a precise description of the effect of sampling on $p_t(x)$;
- be able to compute the probability distribution of bit $b(t)$ from the knowledge of $p_t(x)$.

These very general assumptions should encompass most of possible PTRNG designs. In order to fulfill these assumptions, we give first the following definition.

Definition 8. ([Model] **Stochastic model of a core of randomness**)
A stochastic model of a core of randomness C is a stochastic process of time variable t and space of states V describing the evolution of the internal state $E_C(t)$ of C.
It may be given by a probability distribution $p_t^C(x) = \mathbb{P}(E_C(t)|p_1, \ldots, p_n, E(t_0))$, with $t > t_0$ on $E_C(t)$, depending on parameters $p_1, \ldots, p_n$ and initial state $E(t_0)$. Such a stochastic model is denoted $M(t, p_1, \ldots, p_n)$.

Remark 5. Distribution $p_t^C(x)$ represents all the knowledge that an attacker can have (independently from his computation power) concerning the internal state of the generator core at time t, based on his knowledge of the initial state of the generator at time t_0. So the stochastic model of the physical source of randomness can be viewed as a law of evolution of the knowledge an attacker can get regarding the internal state of the PTRNG. However, this knowledge tends to decrease in time due to different noise phenomena.

Parameters $p_1, \ldots, p_n$ can correspond to a description of the physical environment of the generator (temperature and supply voltage) or a description of the physical properties of the underlying technology. Below, they are termed *parameters* of the physical noise model. The parameters and the preconditions are assumed to be known by the attacker. The parameters are assumed to be manipulable by the attacker, but only within certain limits.
The condition that the statistical model contains all the information accessible to an attacker must be well understood and justified because it has important methodological implications.

Example 6. *To illustrate importance of the stochastic model, we take a somewhat artificial example of a random number generator G_{AES} built using an AES cipher in counter mode. This kind of generator produces a stream of bits by concatenating 128-bit blocks of the form $AES_K(Cnt)$ where AES_K is the AES function with the key K and Cnt is the counter. Using such a generator, any distribution that one could call "natural" can be simulated with arbitrary accuracy as it is often found in the domain of exploitation of random phenomena (e.g., [4]). For example, it is possible to approximate a Gaussian distribution $G(x, \sigma)$ of mean x and standard deviation σ by a binomial law that is very easy to formulate using the outputs of G_{AES}.*

For any attacker with a limited calculation power, without any knowledge of the key K, this distribution will be entirely indistinguishable, for example using statistical tests, from a distribution generated by a completely unpredictable phenomenon.
However, in the context of an analysis of unconditional security that concerns us here, the samples drawn from G_{AES} following the distribution $G(x, \sigma)$ leak certain information regarding the key K. The attacker can recover this information (since it has an infinite computational power) and then G_{AES} becomes completely deterministic.
This example justifies the necessity that the stochastic model of the noise contains all the information that the attacker can gain regarding the source of randomness. If the noise model does not satisfy this condition, the model could be based on the statistical distributions which in reality (perhaps necessitating very complicated calculations) could be described by more accurate statistical laws than those given by the model or even could be fully deterministic. This would result in overestimating the quality of random numbers produced by the generator.

Example 7. *In the case of the elementary generator from Appendix A, it should be remarked that even if the elementary generator would not be affected by physical noises, the output bit stream would feature a pseudo-random component (a pattern) depending on the ratio of frequencies ω_1/ω_0. In a real implementation of oscillator based TRNG, the TRNG is made of several elementary generators the output of which is gathered for instance by the way of an XOR operator. In this case, the pseudorandom behavior is even more complex and difficult to distinguish from the really random component, which determines the entropy rate of the output bit sequence. Moreover, from one run of the TRNG to another, as the ratio of frequencies may vary due to variations in the physical environment of the TRNG (temperature, supply voltage etc.), the pseudorandom component of the output bits may be difficult to predict and model. This is not a problem in our approach since we do not take the eventuel pattern into account in the estimation of the entropy rate of the TRNG. On the contrary, if we would follow the definition of a stochastic model from [1] as a family of distribution which contains the real distribution, the stochastic model would necessirely take into account the pseudorandom component affecting the output bits and causing entropy overestimation.*

The previous discussion illustrates the following important remark:

Remark 6. A stochastic model of the physical noise can only be deduced from a detailed description of the source of randomness and from a physical modeling of phenomena that alone can ensure that the deduced law contains all the information concerning the system. The stochastic model of a random phenomenon contributing to the entropy rate at the TRNG output is a family of distributions of really random output values. These distributions may not correspond to the observed distribution, which can be affected by pseudorandom phenomena and which may be difficult to model. A statistical analysis of the output bits, e.g. using statistical tests, is insufficient since it cannot alone distinguish between the random and the pseudorandom component of the noise affecting output bits.

With Definition 8, the following requirement can be required:

Requirement 7. ([Model] **Availability of the stochastic model of the core of randomness**) The statistical model $M(t, p_1, \ldots, p_n)$ of each core of randomness exploited by the generator for the production of the *proven entropy* shall be available.

Example 8. *The entropy at the output of the elementary generator from Appendix A depends on several parameters. One of them is the duty cycle α of the sampled clock signal. Besides the duty cycle, the entropy depends on the phase noise, which is caused by different phenomena such as random thermal noise or flicker noise. The thermal noise is known to be independent of other noises. The contribution of the thermal noise to the jitter can be modeled using a one-dimensional Wiener process given by, for $i = 0, 1$:*

$$\mathbb{P}\{\xi_i(t_0 + \Delta t) - x_0 \leq x | \xi_i(t_0) = x_0\} = G(\mu_i \Delta t, \sqrt{\Delta t}\sigma_i),$$

i.e. it depends on the drift μ_i, volatility σ_i and accumulation time Δt. The elements of the two pairs (μ_i, σ_i) are the parameters of the physical model. Article [5] shows that α and (μ_i, σ_i) form a complete set of physical parameters for the phase jitter caused by the thermal noise: using these parameters it is possible to simulate perfectly the distribution of output TRNG values due to the thermal noise. The statistical models of the flicker noise are currently not available.

3.2 Analog-to-digital converter

If the space of states of the PTRNG is continuous, an ADC must be used to produce series of bits or series of binary samples. The ADC can behave fully deterministically, but it can sometimes feature a partially random behavior. In the first case, it can be described by a deterministic function (which can be also the identity function). In the second case, its description can include a random variable.[2]
We call this function the model of the ADC. In order to take into account its possible probabilistic nature, we suppose that it is of the form:

$$f_0 : V \times S \to \{0,1\}^*,$$

where S is a probability space.

Requirement 8. (**[Model] Mathematical description of the ADC**) The ADC transforms elements of $V \times S$ into a series of bits. The model of the ADC $f_0 : V \times S \to \{0,1\}^*$ describing this transformation shall be identified.

Remark 7. In the following, we suppose that for all $x \in V \times S$, $f_0(x)$ has the same bit length that we denote by l_{f_0}.

Example 9. *In the case of an elementary generator, the k^{th} bit is generated at the instant t_k that respects $(\omega_0(t_k + \xi_0(t_k))) = k$. The value of the k^{th} bit is $f_\alpha(\omega_1(t_k + \xi_1(t_k)))$ so that in this case the model of the ADC is just f_α.*
The design of an elementary generator can be easily modified to make the ADC nondeterministic. For instance, we can take for S the set $\{0,1\}$ with the equidistributed probability and the function: $f_0 : V \times S \to \{0,1\}$, $(x,s) \mapsto s \oplus f_\alpha(\omega_1(t_k + \xi_1(t_k)))$ where $\oplus$ represent the XOR operation.

Let us denote by $s(t)$ the output of the PTRNG at time t, representing the output sample value of the ADC (which can consist of several bits depending on the value of l_{f_0}). Recall that we denote by $p_t(x)$ the probability distribution

[2]Note that in any case the function of the analog-to-digital converter is assumed to be computable by the attacker.

$\mathbb{P}(E(t)|p_1, \ldots, p_n, E(t_0) = \ldots)$. Let b be a sample of l_{f_0} bits, the conditional probability $p_t(x|s(t) = b)$ is in general different from $p_t(x)$. This is due to the fact that the output values $s(t)$ convey some information about the state of the PTRNG, which can be used by the attacker to enlarge his knowledge regarding the PTRNG and to enhance his capacity to guess subsequent bits. In fact, the more the output bits are unpredictable (i.e. the higher their entropy rate is), the more information about the state of the PTRNG they give.
This justifies the need for the following requirement:

Requirement 9. ([**Model**] **Effect of the analog-to-digital conversion**) For all t, b, the conditional distribution of probability $p_t(x|s(t) = b)$ should be either computed or at least a conservative approximation of it should be obtained. By the conservative approximation, we mean a distribution of probability $p_t^*(x)$ which gives more information about the actual state of the PTRNG than the value of $p_t(x|s(t) = b)$ that is such that for all x such that $p_t^*(x) > 0$, we have $p_t^*(x) \geq p_t(x|s(t) = b)$.

It is clear that the use of a conservative approximation of $p_t(x|s(t) = b)$ gives more power to the outside attacker in his ability to predict the output bits produced by the TRNG. So the proposed approach allows to compute a lower bound of the entropy rate per bit of the TRNG, which can be used to guarantee its security at the expense of its throughput.

Example 10. *In the case of the elementary generator, suppose that at time t the knowledge of the attacker regarding the state of the PTRNG is represented by the distribution $p_t(x)$ depicted in Fig. 2. It is clear that if $s(t) = 1$ then the attacker knows that the actual state is inside the red zone, otherwise it is inside the blue zone. So the probability distribution $p_t(x|s(t) = 1)$ (resp. $p_t(x|s(t) = 0)$) is precisely given by the relative surface of the red and blue zones.*
Therefore, in this case, a convenient conservative approximation of the distribution $p_t(x|s(t) = b)$ would mean that by knowing the TRNG output at time t, the observer knows also the internal state of the PTRNG.

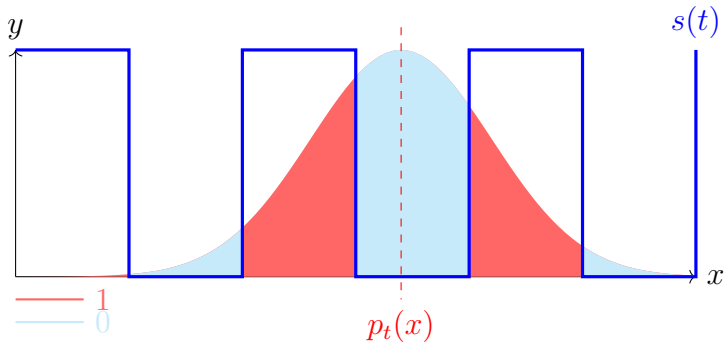

Figure 2: Effect of the analog-to-digital conversion: example considering the elementary PTRNG

Definition 9 (**[Model] Stochastic model of the TRNG**). A stochastic model of the PTRNG is a mathematical description of the process of data generation, which is written in a simplified form depending on the statistical assumptions: knowledge of the stochastic model of the physical noise $M(t, p_1, \ldots, p_n)$, the way the physical noise is converted to numerical values in discrete time instants $(t_1, \ldots, t_k)$. The stochastic model specifies the distribution $D(k, q_1, \ldots, q_m, p_1, \ldots, p_n)$ of the set of k output values, each composed of l_{f_0} bits. The model depends on parameters $p_1, \ldots, p_n$ of the source of randomness, but also on parameters $q_1, \ldots, q_m$, which are the parameters of the TRNG.

Certain parameters $q_1, \ldots, q_m$ may be adjustable during the design of the generator, but must not be manipulable by the attacker.

Example 11. *In the case of an elementary PTRNG, the parameters of the model of the source of randomness (the phase noise of oscillator Osc_i, $i = 0, 1$ coming from the thermal noise) are the two pairs (μ_i, σ_i) specifying the drift and volatility of a Wiener process. The distribution of PTRNG output values depends also on the duty cycle α_1 of the sampled clock and on an additional parameter, the value of the frequency divisor K_D, which is the only tunable parameter of the PTRNG.*

Definition 9 leads to setting up the following requirement:

Requirement 10. ([Model] **Availability of the stochastic model of the PTRNG**) A stochastic model for the PTRNG shall be available.

This assumes that all the previous requirements given in Sec. 3 (and in particular the requirement of availability of a stochastic model of the physical source of randomness) shall be fulfilled.

Requirement 11. ([Model] **Consistency of the statistical model of the PTRNG**) The stochastic model of the PTRNG shall be obtained:

- by respecting Requirement 7 to specify the law of $p_t(x)$ depending continuously on the time;
- by respecting Requirement 9 to specify the effect of sampling of $p_t(x)$ in discrete time intervals.

4 Fitting the model with the generator – definitions and requirements

Requirement 12. ([Model fitting] **Identification of physical random sources contributing to random number generation**) The physical phenomena responsible for the unpredictable nature of generator operation shall be clearly identified.

Example 12. *In the case of the elementary random number generator, the analog electric noises are transformed to the instability of delays in logic gates, which cause the phase jitter of the clock signal generated by the ring oscillator. The phase jitter is a complex phenomenon since it appears to be caused of different noise sources: thermal noise, flicker noise, etc. In our estimation of the proven entropy, we only take into account the thermal noise.*

Requirement 13. ([**Model fitting**] **Evaluation of parameters of the physical noises**) One shall be able to evaluate experimentally the parameters $p_1, \ldots, p_n$ of the statistical model for physical noise $M(t, p_1, \ldots, p_n)$. One shall be able to evaluate the measurement errors of these parameters.

Example 13. *In the case of ring oscillator-based PTRNGs, several techniques can be exploited to measure parameters associated with the phase noise model based on the thermal noise* (μ, σ).
They can be broken down into two main groups:

- *external measurement techniques, which consist in reading the generator internal signals and analyzing them with an external measurement equipment (e.g. an oscilloscope);*
- *internal measurement techniques, which consist in analyzing the internal signals inside the device, in order to determine the values of input parameters of the model.*

The main advantage of external measurement techniques is that precise measurement equipment and advanced signal processing techniques such as advanced differential input/output techniques, high quality probes, and low noise oscilloscopes can be used. However, these techniques bring a number of disadvantages:

- *Input/output circuitry, transmission cables and input noise of the measurement equipment result in fairly inaccurate measurements (which can be improved for example by using differential probes and high quality devices);*
- *they are difficult to be performed on a production line;*
- *testing of each individual circuit can be very long and thus impractical.*

In any case, one of the main difficulties in satisfying Requirement 13 results from the need to filter various types of noises which, since they are described by different statistical laws, must undergo specific analysis. In particular, it is important to filter out the global deterministic noises which have very high amplitude and do not contribute to the unpredictable nature of the generator

(see for example [5]). One approach is to perform differential measurements on a device containing only a PTRNG, in the measurement campaign with a carefully prepared experimental setup [6] (stabilized power supply, controlled electromagnetic environment, controlled temperature, etc.).

Remark 8. To fulfill Requirement 13, the use of statistical tests is perfectly legitimate, since they can help to evaluate the fitting of the stochastic model with the observed probability distribution and the theory of tests of assumptions and parametric tests provide tools that are well adapted to this type of situation.

It is often difficult to evaluate *a priori* the measurement errors in Requirement 13. One possible approach is to repeat several measurements, while ensuring that the experimental conditions are maintained stable and to calculate an estimator of the standard deviation of different measurements. To compensate the uncertainties of the different methods, it may be interesting to corroborate the results with different experimental equipment (for example to combine some external measurement with various types of internal measurements).

Requirement 14. ([**Model fitting**] **Stability of parameters of the statistical models of the physical noises**) The stability of parameters $p_1, \ldots, p_n$ of the stochastic model shall be evaluated for the physical noise with regard to

- physical environmental operating conditions of the RNG: temperature, supply voltage, electromagnetic environment;
- operation of the RNG inside the system: operation of surrounding circuits (e.g. a cipher) and their impact on the generation of random numbers;
- variations of production parameters depending on the target technology.

Aging tests could also be performed.

The purpose is to evaluate the stability of parameters of the physical phenomena guaranteeing unpredictability of generated numbers from one device to another and throughout the operating domain of the circuit to verify that the security of the generator is ensured in the worst conditions.

Example 14. *The frequency of the clock signals generated by ring oscillators depends significantly on the temperature (the frequency decreases with the temperature), the supply voltage (the frequency increases with the power supply voltage), as well as the integration of other functional blocks in the device (this depends of course on the placement-and-routing of the oscillators and surrounding blocks).*

Verification of Requirements 13 and 14 can be called the technology qualification stage. This qualification could require some additional dedicated circuitry (e.g. differential inputs/outputs) to ensure that:

- the measurements are as accurate as possible;
- the circuit can be tested in the most unfavorable environmental conditions possible.

Example 15. *For the last point, in the case of an elementary generator, its operation could be tested and compared with other oscillators generating clocks with frequencies close to those of ring oscillators.*

Remark 9. Verification of suitability of the stochastic model of sources of randomness is not straightforward. It is possible that the designer has only a partial knowledge of how to model physical noises. When considering the elementary RNG, according to the recent state of the art, only the thermal noise is known to be modeled. It is very difficult to take into account the impact of global deterministic noises on generated numbers. Thus, if care is not taken, predictions given by the PTRNG stochastic model will in general be rather conservative comparing to experimental results.

Requirement 15. ([**Model fitting**] **Management of the PTRNG entropy rate**) Using the stochastic model of the TRNG, it shall be possible to adjust parameters $q_1, \ldots, q_n$ to obtain required entropy rate.

Remark 10. To verify the previous requirement, the use of statistical tests is once again entirely legitimate since the theoretical corpus of the theories of tests of assumptions can be used for example.

5 Requirements on the embedded tests

Definition 10. ([**Tests**] **Embedded tests**)
A parametric test of a PTRNG, which is described by a stochastic model $d(k, q_1, \ldots, q_m, p_1, \ldots, p_n)$, is a test that verifies that parameters $p_1, \ldots, p_n$ and $q_1, \ldots, q_n$ are permanently inside the bounds that ensure sufficient entropy rate at generator output.

Example 16. *In the elementary oscillator based RNG, the output entropy rate depends on the size of the jitter and on the mean periods of the two generated clock signals (T_1 and T_0). Consequently, embedded test should measure the jitter coming from the thermal noise, for example as presented in [7], and the periods T_1 and T_0. The measured values should be compared with thresholds obtained from the stochastic model – the parameter values ensuring sufficient entropy rate required by the targeted security level depending on the application.*

From Definition 10, the following requirement can be deduced:

Requirement 16. ([**Tests**] **Execution of embedded tests**) Parametric embedded tests shall run at startup and continuously.

Example 17. *In the case of an elementary PTRNG, the following parametric tests could be used to evaluate the parameters of the clock signal including the jitter coming from the thermal noise:*

- *the frequency (monobit) test can be used as an estimator of duty cycle α;*
- *article [7] shows that the auto-correlation test can be used as an estimator of volatility σ;*
- *an additional method described in [7] can be used to calculate drift μ_i for $i = 0, 1$.*

Remark 11. A statistical test that cannot be interpreted as a parametric test is useless and even dangerous in certain circumstances.

The following definitions and remarks apply in the case that the ADC is deterministic.

Definition 11. ([Tests] **Test of integrity of the PTRNG data path**) A test of integrity of the whole PTRNG data path is any test that verifies correct operation of the generator including all the blocks between the randomness core and the generator output and in particular between analog-to-digital conversion and post-processing.

With this definition, we can write:

Requirement 17. ([Tests] **Verification of integrity of the PTRNG data path**) Correct operation of all the blocks between the core of randomness and the generator output shall be verified using the PTRNG integrity tests.

Example 18. *We can give here as an example important classic deterministic tests of bonding between the core of the generator and the output.*

6 Comparison of the proposed methodology with existing standards and recommendations

In this section we will compare the proposed methodology of evaluation of the TRNG design with the German document AIS 20/31 [1] and American standard NIST SP 800-90 [8], which has been superseded by three other standards – NIST SP 800-90A [2], NIST SP 800-90B [9], and NIST SP 800-90C [10].
To make comparisons easier, we recall the methodology proposed in this document, which is illustrated in Fig. 3. This figure shows clearly the relationship between the TRNG design, models, embedded tests, and testing procedures performed during security evaluation.
We recall that the TRNG hardware is composed of five types of blocks:

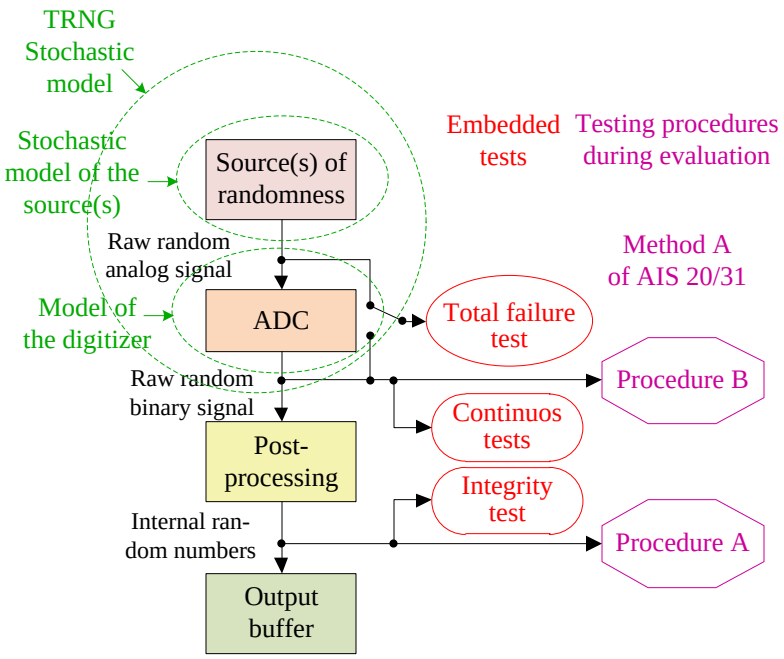

Figure 3: *TRNG design, modeling and testing according to this document*

1. Block(s) containing source(s) of randomness,
2. Analog-to-digital converter (ADC),
3. Post-processing block,
4. Output buffer,
5. Embedded tests.

The stochastic model of the generator of the raw random digital signal (digital noise) is based on two underlying models:

- Stochastic model(s) of source(s) of randomness,
- Model of the analog-to-digital conversion.

Hardware block(s) containing source(s) of randomness define security boundaries of the sources of randomness. Analog-to-digital converter converts targeted analog signal components (e.g. the analog timings) to digital values. If this

conversion is included intrinsically in the random signal generation mechanism (i.e. it is performed inside the block containing the source of randomness), the analog-to-digital conversion is replaced by an identity function.
The role of the post-processing block is to enhance statistical parameters of the output signal. Usually, it increases entropy rate per bit at generator output at expense of the output bit rate. The output buffer determines output data format of the generator (number and order of bits).
The second column of blocks presented in Fig. 3 represent embedded tests:

- Total failure test,
- Continuous test(s),
- Integrity test.

The role of the total failure test is to detect as soon as possible a total failure of the source of randomness. Therefore, it should test the signal as close as possible to its source, i.e. even before the analog-to-digital conversion, if this conversion can be separated from the source of randomness.
Continuous test(s) shall be parametric stochastic tests, which evaluate continuously the value of input parameters (physical quantities) of the stochastic model. If the size of any of these parameters is smaller that required to gain expected entropy rate, the alarm shall be triggered.
The test of integrity verifies correct operation of the whole data path between the randomness core and the generator output, and in particular the analog-to-digital converter and the post-processing block. If possible, this test should be executed continuously. If this test is realized as a known answer test (KAT), it shall be executed at least during startup tests and periodically on demand.
The startup procedure shall be composed of at least all three above mentioned types of tests. The TRNG output is not allowed until all the tests were successfully finished.
The last column of operations in Fig. 3 represents testing procedures performed during the TRNG security evaluation. Since according to requirements of this document, the raw binary signal (the digital noise) shall be available, only the Method A of AIS 20/31 is acceptable, i.e. the digital noise shall be tested by Procedure B, which defines requirements concerning execution of tests T6 to T8 and the internal random numbers shall be tested using Procedure A defining, which defines requirements regarding execution of tests T0 to T5.

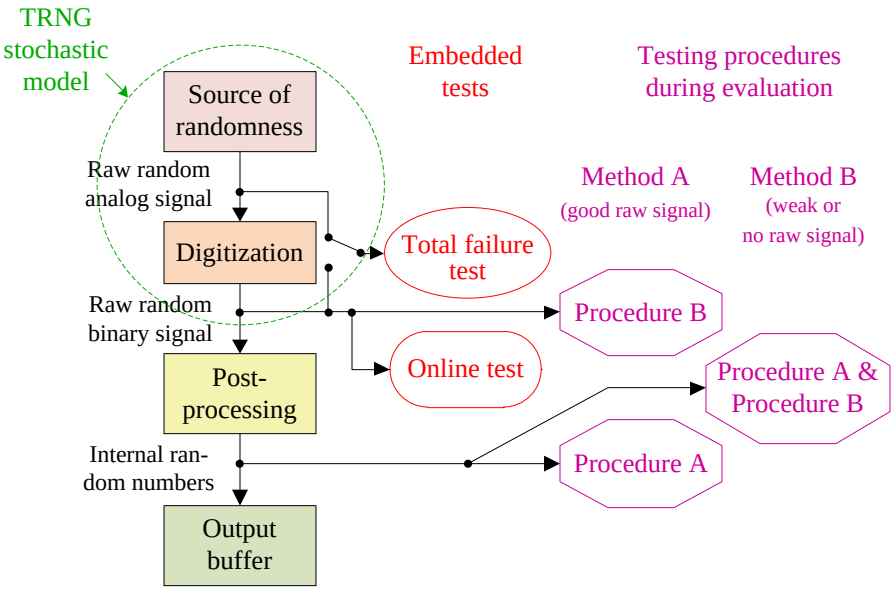

Figure 4: *TRNG design, modeling and testing according AIS 20/31*

6.1 Comparison with German AIS 20/31

As explained before, the TRNGs validated using methodology proposed in this document correspond to security level PTG.2 (i.e. without the cryptographic post-processing). All TRNGs, which will be compliant with the methodology proposed in this document will be compliant with AIS 20/31 for security level PTG.2. However, not all designs compliant with AIS 20/31 at security level PTG.2 would necessarily be compliant with this document. In particular, Method B of evaluation of the PTRNG specified in AIS20/31 is not applicable according to these Recommendations.
The following list of requirements, which are not explicitly specified in AIS 20/31, shall be satisfied according to present document:

- **Requirement 7** – Availability of the stochastic model of the core of randomness
- **Requirement 8** – Mathematical description of the analog-to-digital conversion
- **Requirement 9** – Characterization of the effect of the analog-to-digital conversion

- **Requirement 11** – Consistency of the stochastic model of the PTRNG
- **Requirement 13** – Evaluation of parameters of the physical noises
- **Requirement 14** – Stability of parameters of the stochastic models of physical noises
- **Requirement 17** – Verification of integrity of the whole datapath between outputs of sources of randomness and output of the digital noise.

The main differences can be observed by comparing Fig. 3 illustrating the proposed methodology with Fig. 4, which presents the TRNG design, modeling and testing according AIS 20/31 . It can be observed that while the generator structure remains the same, its modeling and testing are different.
Unlike the AIS 20/31 procedure, the present document requires models of both main parts of the generator core: the sources of randomness and the ADC. While the embedded online test required by AIS 20/31 can be launched on demand or run continuously or periodically, embedded parametric tests required by the proposed document shall run continuously.
During the TRNG security evaluation using the off-line black-box tests, only Method A can be applied, since this documents requires a good quality raw binary signal to be available for off-line statistical testing. Comparing Fig. 3 and Fig. 4, it can be observed that according to AIS20/31, the post-processing block is not necessarily tested . Since the present document requires (Requirement 17) the whole datapath to be tested, the designer shall propose a method to test also the correct operation of the post-processing block.

6.2 Comparison with American NIST SP 800-90B

Comparing with the American approach, we can conclude that the methodology presented in this document is more stringent. While the American approach requires only some analysis of the source of randomness, the proposed methodology requires (as AIS 20/31) the use of a stochastic model to estimate the output entropy rate. Another important difference is that the American standard requires simple black box statistical tests (Repetition count and Adaptive proportion tests) to be applied continuously on the digital noise signal. Consequently, in order to make the TRNG design compatible with the American approach, the required continuous tests should be added.
Comparing Fig. 3 and Fig. 5, we can observe that although the structure of the generator remains similar, the naming of blocks is slightly different. Namely, the Source of randomness from AIS 20/31 is called the Analog noise source in NIST

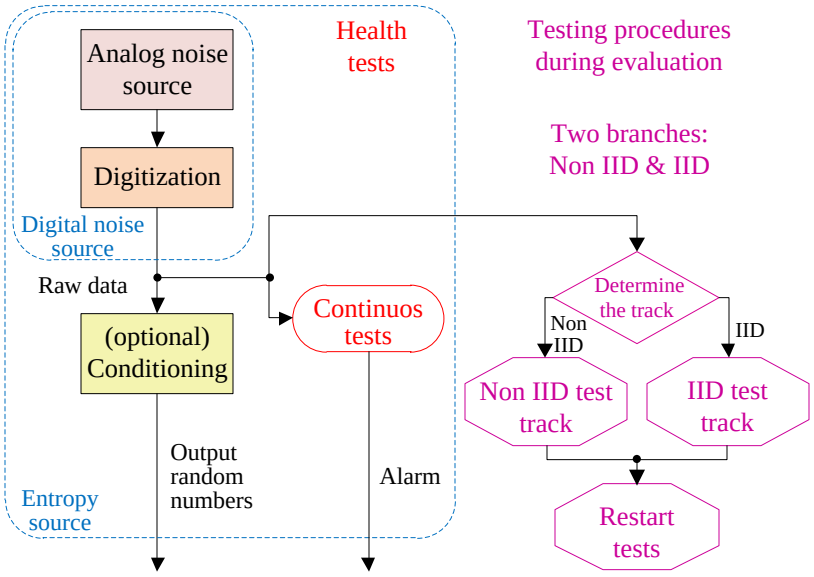

Figure 5: *TRNG design and testing according to NIST SP 800-90B*

SP 800-90B. Although the role of the Conditioning element presented in Fig. 5 is similar to that of the Post-processing block from Fig. 4, (i.e. to increase statistical quality of the output signal), the NIST SP 800-90B document recommends to use one of proposed six vetted functions to do so (see [[9]] for more details).
The embedded tests defined by the NIST SP 800-90B document, which are called Health tests, can be divided to three categories: *start-up tests*, *continuous tests* (primarily on the noise source), and *on-demand tests*. The same test categories are accepted in the present document.
Off-line testing procedures required by NIST SP 800-90B are very different from those required by AIS 20/31 and thus by the present document. Different tests are used for random variables, which are independent and identically distributed (IID) and for those which are not (Non IID). The aim of both tracks is to estimate entropy rate at the output of the digital noise source. This entropy rate must be confirmed by the Restart tests, which should prove that the generator behaves correctly and that it outputs different random numbers from the very beginning. The stochastic model is not required, although its use is recommended. Nevertheless, even if the model was used to estimate the output entropy rate, the certified entropy rate is always the lowest bound of min-entropy given by the

corresponding testing track (IID or Non IID) and confirmed by the Restart tests.

7 Conclusions

In this document, we presented recommendations for the design and security validation of a physical true random number generator implemented in an electronic device. We started the document with a specification of general objectives of the security evaluation process and we introduced basic definitions and requirements concerning the PTRNG design. We explained, why availability of the stochastic model of the physical phenomena used as sources of randomness is important for security evaluation of the generator. The stochastic model of a complete PTRNG can be thus constructed as an ensemble of two models: model of sources of randomness and model of the analog-to-digital conversion process. Moreover, we explained that to further increase security of the system, the embedded test must be adapted to the PTRNG stochastic model by testing continuously the values of input parameters of the model and comparing them with the thresholds guaranteeing sufficient entropy rate at generator output. Final comparison with existing security standards show clearly advantages of the proposed approach.

References

[1] W. Killmann and W. Schindler. A proposal for: Functionality classes for random number generators (AIS 20 / AIS 31), Version 2.0. BSI, Germany. [online] Available at https://www.bsi.bund.de, 2011.

[2] E. Barker and J. Kelsey. Recommendation for Random Number Generation Using Deterministic Random Bit Generators, NIST Special Publication 800-90A Rev. 1. [online] Available at https://csrc.nist.gov/publications/detail/sp/800-90a/rev-1/final, 2015.

[3] V. Fischer, P. Haddad, and K. Cherkaoui. Ring oscillators and self-timed rings in true random number generators. In Y. Nishio, editor, *Oscillator Circuits: Frontiers in Design, Analysis and Applications*, pages 267–292. The Institution of Engineering and Technology (IET), 2016.

[4] D. E. Knuth. *The Art of Computer Programming. Vol. 2.* Addison-Wesley Publishing Co., Reading, Mass., second edition, 1981. Seminumerical algorithms, Addison-Wesley Series in Computer Science and Information Processing.

[5] M. Baudet, D. Lubicz, J. Micolod, and A. Tassiaux. On the security of oscillator-based random number generators. *Journal of Cryptology*, 24(2):398–425, 2011.

[6] E. Noumon Allini, M. Skorski, O. Petura, F. Bernard, M. Laban, and V. Fischer. Evaluation and monitoring of free running oscillators serving as source of randomness. *Transactions of Cryptographic Hardware and Embedded Systems (TCHES)*, 2018(3):214–242, 2018.

[7] V. Fischer and D. Lubicz. Embedded evaluation of randomness in oscillator based elementary TRNG. In Lejla Batina and Matthew Robshaw, editors, *Cryptographic Hardware and Embedded Systems (CHES 2014)*, volume 8731 of *LNCS*, pages 527–543. Springer, 2014.

[8] E. Barker and J. Kelsey. Recommendation for Random Number Generation Using Deterministic Random Bit Generators, NIST Special Publication 800-90 (Revised). [online] Available at http://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-90r.pdf, 2007.

[9] M.S. Turan, E. Barker, J. Kelsey, K.A. McKay, M.L. Baish, and M. Boyle. Recommendation for the Entropy Sources Used for Random

Bit Generation, NIST Special Publication 800-90B. [online] Available at https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-90B.pdf, 2018.

[10] E. Barker, J. Kelsey, K. McKay, A. Roginsky, and M.S. Turan. Recommendation for Random Bit Generator (RBG) Constructions, NIST Special Publication 800-90C (Third Public Draft). [online] Available at https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-90C.3pd.pdf, 2022.

Appendix

A Elementary oscillator-based TRNG

This appendix describes simple TRNG on which we illustrate our approach throughout the document. An *elementary oscillator-based TRNG* is composed of two oscillators Osc_i for $i = 0, 1$, a frequency divider and a sampler (see Figure 6). The output of one of the oscillators determines sampling times of the output of the second oscillator via a sampling unit that includes a D-type flip-flop. The frequency of the sampling oscillator is divided by a factor K_D. Division ratio K_D determines the time interval required to accumulate sufficient phase jitter which determines statistical bias of the bits of the TRNG output.
In what follows, it is supposed that Osc_1 is the oscillator that generates the sampled signal and Osc_0 produces the sampling clock signal.

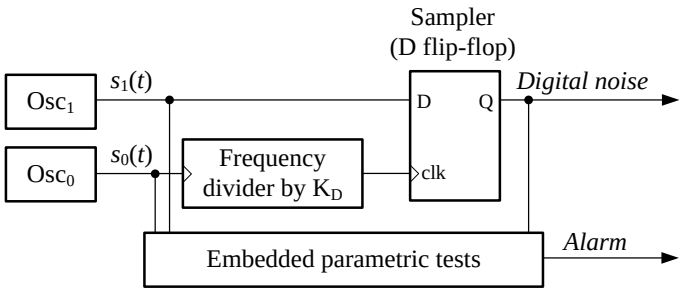

Figure 6: *Structure of an oscillator-based TRNG*

For $i = 0, 1$, the output of the signal of Osc_i is given by a periodic function of time t of the form

$$s_i(t) = f(\omega_i(t + \xi_i(t))), \tag{1}$$

where f can be any real function of period 1. In our case, we can assume that we are dealing with the integration of a TRNG on a logic device and thus for $\alpha \in [0, 1)$, we define f_α as the function of real value of period 1 such that $f_\alpha(x) = 1$ for $0 < x < \alpha$ and $f_\alpha(x) = 0$ for $\alpha < x < 1$, and $f_\alpha(0) = f_\alpha(\alpha) = 1/2$. We use f_α as a suitable model for a clock signal produced by clock generators and ring oscillators in particular. The clock edge is not necessarily centered on the half-period, since the oscillators often have unbalanced half-periods.
In practice, the frequencies of the two signals $s_i(t)$, $i = 0, 1$, fluctuate due to phase jitter. Thus ω_i is the *average frequency* of the signal $s_i(t)$, $(\omega_i(t + \xi_i(t)))$

is the *phase* of the oscillator and function $\xi_i(t)$ represents the *absolute phase drift*.