Recommendations for the 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

February 22, 2024

Abstract

These Recommendations describe essential elements of the design of a secure physical true random number generator (PTRNG) integrated in an electronic device. Based on these elements, we describe and justify requirements for the design, validation and testing of PTRNGs, which are intended to guarantee the security of generators aimed at cryptographic applications.

1 Introduction

Random number generators (RNGs) are essential components of cryptographic equipment. In particular, they are used to generate keys, key identifiers, initialization vectors, and nonces, but also to protect cryptographic equipment against attacks that use side channels including power consumption, electromagnetic emanation, and sound, among others. Random number generators are vital because they are the only source of diversity in cryptographic algorithms, a diversity

whose combinatorial richness characterized by entropy (Shannon entropy or min entropy) is the only protection against brute force attacks.
In general, most 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, periodically reseeds a deterministic PRNG, which must be cryptographically secure, i.e. it must exploit a cryptographic mode with a proven security level (see [1]) or use an approved cryptographic algorithm (see [2]).
The PRNG guarantees security of the generator based on computational assumptions if the source of randomness in the generator fails to operate correctly. However, this is tolerated for short time intervals depending on the security level targeted and on the quantity of entropy accumulated in the PRNG (the PRNG must have accumulated sufficient entropy before the failure of the physical source of entropy).
The present document concerns only the TRNG part of a RNG. We define and describe the essential elements of a TRNG design approach that guarantees security of the generator and takes the most recent advances of 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 using the approach recommended here. However, we believe that for devices like random number generators which guarantee security in cryptographic applications, the developer must demonstrate its security, and we are convinced the approach we recommend provides the strongest assurance along with the fewest possible constraints on the design.
Like in cryptographic applications, in which the designer has to prove the security of protocols based on widely accepted computational assumptions, we want to prove unpredictability of the generated numbers by estimating a lower bound of the entropy rate of the TRNG based upon well established assumptions about the randomness of a number of identified physical phenomena. To this end, we consider an attacker who may have full knowledge of the TRNG design and can observe its output. We make no assumption about his computational power, which can hypothetically be unlimited.
Unidentified phenomena may contribute to the random nature of the operation of the TRNG. This is evidenced for example by statistical tests of the generator output bits that show that the output bits have higher entropy rates than could have been predicted by only accounting for identified physical noises. However, in our approach, we do not take these unidentified noises into account because it is not possible to evaluate their contribution to the security of the generator. In the following, to distinguish the entropy rate produced by identified phenomena from the rest of the entropy, we refer to it as the *proven entropy*.
We believe a TRNG that, by design, is not suitable for such an analysis, should

be considered as being unsuitable for use in high security cryptographic applications. Unfortunately, if analyzed as recommended in these recommendations, the security analysis of numerous constructions in the literature is based on dubious arguments.

Our approach is described as a series of requirements that are well-founded and argued[1]. We provide definitions that appear important but are perhaps insufficiently clear in the literature: for example, that of a stochastic model of physical noise that we distinguish from a stochastic model of the whole generator. We define the testing conditions in which 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 of no use and may 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 widely studied and best understood among those described in the literature (for an overview of TRNGs using free running oscillators as a source of randomness, see [3]). The example of application of ring oscillators presented confirms the rigor of the recommended approach, that is fully compatible with the production requirements of a data security device including a random number generator.

The document is structured as follows. Section 2 outlines the general objectives of the evaluation process described in the document and introduces basic definitions and requirements concerning the TRNG design. Section 3 provides the definitions and requirements for the model of the source of randomness. Section 4 provides the definitions and requirements for matching the model to the generator. Section 5 defines the objectives and kinds of embedded tests required. In Section 6, we compare the proposed PTRNG design and evaluation approach with the methodologies required by German AIS20/31 and American NIST SP 800-90B standards . In Section 7, we present our conclusions.

2 TRNG design – definitions and requirements

In most cases, a physical true random number generator is designed and its security is evaluated by independent institutions and companies. The certification bodies allocate a limited number of licenses to laboratories that are accredited

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

to perform security evaluations. To simplify the evaluation process, the TRNG designer, the evaluation laboratory and the certification body should use the same vocabulary as well as identical definitions.
The design and certification process should procure 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, that fits the particular constraints of their project.
Since the proof of security we wish to achieve is based on a mathematical model of the TRNG, one of the main difficulties of the security evaluation process is guaranteeing accuracy of the predictions given by the model with respect to the reality. To achieve this objective, it is very important to be able to check how the main building blocks of the model of the TRNG fit the structure of the TRNG itself.
To make this task easier, we group our definitions and requirements in four categories:

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

It is clear that the operation of a TRNG, whose essential function is to produce a series of unpredictable bits or binary numbers, necessarily relies on unpredictable physical phenomena that are referred to in the following as *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 part of a device), featuring the source (or sources) of randomness, which is delimited by a well defined *security boundary*. It is characterized by its *internal state* $E(t)$ that evolves depending on *physical random phenomena* occuring inside this area (*physical noises*).

Remark 1. As stated in Definition 2, the physical core of randomness is a device or a part of a device, in which some random phenomenon or random phenomena occur. The security boundary of this area must be clearly defined by the designer. At first glance, it may seem that this boundary can be chosen arbitrarily. For instance, it is always possible to consider that several cores of randomness constitute one 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. Most of these requirements will be easier to fulfill by choosing the smallest possible security boundary for the physical cores even if this means splitting a big core into smaller and more elementary ones. Nevertheless, designers can freely chose the boundary of the core of randomness as long as all the requirements are met.

Example 1. *The TRNG in Appendix A features two physical cores of randomness: the oscillators Osc_i ($i = 0, 1$) that 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 the oscillator Osc_i is affected by various sources of noise (thermal noise, flicker noise etc.) that produce the phase noise. The source of randomness in the cores of randomness Osc_i are the phase noises that depend 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 that uses 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 if he/she can observe the internal state of the cores of randomness (see Definition 7 of the PTRNG security model). This assumption means that the security of the PTRNG (and therefore its unpredictability) relies solely on the sources of randomness that affect the cores of randomness. As a consequence, in the security assessment of a PTRNG, only a specified list of sources of randomness shall be considered and the contribution of all other sources shall be disregarded. 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 by the designer. In other words, we assume that the attacker may have more computational power than the designer. So the best possible TRNG design is the one, about 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 that performs analog-to-digital conversion is thus an integral part of the generator. Accordingly, as shown in Fig. 1, we distinguish four basic PTRNG blocks (entities):

- Core(s) of randomness that include source(s) of randomness,
- An analog-to-digital converter (ADC),
- A post-processor,
- Embedded tests.

The PTRNG usually contains 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) that can be of low statistical quality (e.g. low entropy). In particular, it can feature a deterministic pattern. If necessary, this low statistical quality can be enhanced by an algorithmic post-processor to obtain a high-quality digital noise.

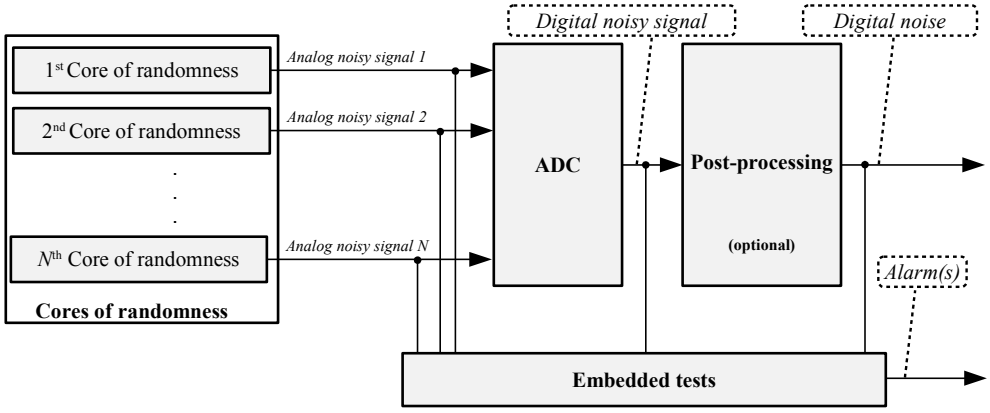

Figure 1: *General structure of a TRNG*

The quality of the generated random numbers is continuously tested by embedded tests performed according to pre-defined testing procedures. At least two testing procedures shall be defined: a 'Startup procedure' and a 'Continuous testing procedure'. If certain kinds 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 emerge in the elementary oscillator-based TRNG (EOTRNG) presented in Appendix A: two oscillators that generate jittered clocks. The ADC is composed of two parts: a frequency divider and a sampler. The division factor of the frequency divider can be chosen so that post-processing is not needed (because enough entropy has accumulated during each sampling period). The embedded parametric tests scrutinize two generated clock signals and the output of the generator. They measure the TRNG parameters used as input parameters in the stochastic model: the phase noise 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 delimited.

Requirement 3. ([Design] **Security boundary of the core of randomness**) It shall be clearly shown 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 that 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 attempted manipulation would be detected by embedded tests. In particular, the physical part of the generator that guarantees randomness (the core of randomness) shall not have any inputs that can affect the source of randomness.

Example 3. *In the EOTRNG in Appendix A, the jitter of the two clock signals comprises both deterministic and non-deterministic (random) components. The global deterministic component is reduced to negligible size using a differential principle: two identical oscillators are influenced in the same way by global deterministic signals. We assume that thermal noise is always present in semiconductor devices in normal operating conditions. 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 have no 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 in Appendix A, the randomness in the generator output depends on the relative phase noise of the two oscillators. In general, the two oscillators are not completely independent since – for instance – they may share the same power supply and be 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 noise, 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 into digital signals (numbers) is assumed 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. the behavior of 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 in Appendix A, the ADC that transforms the random phase of the clock signal into 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 used to enhance the statistical parameters of the generator. It shall not reduce the entropy rate per bit.

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

Requirement 6. ([**Design**] **Specification of the post-processing algorithm**) The post-processing algorithm and its objective shall be clearly defined. The designer shall show that the selected post-processing algorithm does not reduce 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 different ways. 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 cores 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 who has full knowledge of the PTRNG design and can freely observe its output. We make no assumptions regarding the attacker's computational power, which may be unlimited. The attacker tries to predict output bits of the PTRNG. The feasibility of the attack is closely related with the entropy rate of the PTRNG, which is the average amount of information one has to give the attacker so that he/she 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 his/her knowledge of the internal state $E(t)$.

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

- The physical noises that tend to reduce the attacker's knowledge about the state of the generator;
- Output bits that allow the attacker to access some information about the internal state of the generator.

As the attacker's incertitude concerning the bits produced by the PTRNG necessarily originates 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 that describes the effect of the physical noise on the evolution of $p_t(x)$;
- guarantee that this law remains stable over time and in prevailing environmental conditions and remains capable of continuously monitoring any 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 cover most possible PTRNG designs. In order to fulfill these assumptions, we first provide 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. The model 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 of the core of randomness C is denoted $M(t, p_1, \ldots, p_n)$.

Remark 5. The distribution $p_t^C(x)$ represents all the knowledge an attacker can have (independently of his/her computation power) concerning the internal state of the generator core at time t, based on his/her 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 changes in the knowledge an attacker can obtain concerning the internal state of the PTRNG. However, this knowledge tends to decrease over 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 to 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 also assumed to be manipulable by the attacker, but only within certain limits.
The condition that the stochastic 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 the importance of the stochastic model, we use 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 is often found in the 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 limited calculation power, with no knowledge of the key K, this distribution will be entirely indistinguishable, for example by using statistical tests, from a distribution generated by a completely unpredictable phenomenon. However, in the context of the analysis of unconditional security that concerns us here, the samples drawn from G_{AES} following the distribution $G(x, \sigma)$ leak certain information concerning key K. The attacker can recover this information (since he/she has infinite computational power) and the G_{AES} then becomes completely deterministic.
This example justifies the requirement that the stochastic model of the noise contains all the information that the attacker can gain concerning the source of randomness. If the noise model does not satisfy this condition, the model could be based on the statistical distributions that in reality (perhaps necessitating very complicated calculations) could be described by more accurate statistical laws than those given by the model or could even 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 in Appendix A, it should be noted that even if the elementary generator is not affected by physical noises, the output bit stream would still feature a pseudo-random component (a pattern) depending on the ratio of frequencies ω_1/ω_0. In a real implementation of an oscillator based TRNG, the TRNG is made of several elementary generators the output of which is collected (for instance) by way of an XOR operator. In this case, the pseudorandom behavior is even more complex and difficult to distinguish from the truly 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. In our approach, this is not a problem since we do not take an eventual pattern into account when estimating the entropy rate of the TRNG. On the contrary, if we follow the definition of a stochastic model from [1] as a family of distributions that contains the real distribution, the stochastic model would necessarily take into account the pseudorandom component that affects the output bits and causes overestimation of entropy.*

The above 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 model of

phenomena that alone can guarantee 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 may be affected by pseudorandom phenomena and hence may be difficult to model. A statistical analysis of the output bits, e.g. using statistical tests, is not sufficient since alone, such an analysis cannot 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 stochastic 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 in Appendix A depends on several parameters. One is the duty cycle α of the sampled clock signal. Beyond the duty cycle, 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. Its contribution 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. the jitter component coming from the thermal noise depends on 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 noise caused by the thermal noise: using these parameters it is possible to perfectly simulate the distribution of output TRNG values caused by the thermal noise. Stochastic models of flicker noise are currently not available.

3.2 Model of the 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 may sometimes display partially random behavior. In the first case, the converter can be described by a deterministic function (that 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 account for its possible probabilistic nature, we assume it is of the form:

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

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 $f_C : V \times S \to \{0,1\}^*$ describing this transformation shall be identified.

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

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 easily be 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 that may consist of several bits depending on the value of l_{f_C}. Recall that we denote by $p_t(x)$ the probability distribution

[2]Note that in both cases 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_C} bits, the conditional probability $p_t(x|s(t) = b)$ generally differs 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 that can be used by the attacker to extend his/her knowledge about the PTRNG and to enhance his/her ability to guess subsequent bits. In fact, the more unpredictable the output bits (i.e. the higher their entropy rate), the more information about the state of the PTRNG they provide.
This justifies 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 $p_t^*(x)$ that provides more information about the actual state of the PTRNG than the value of $p_t(x|s(t) = b)$ 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 to predict the output bits produced by the TRNG. Consequently, the proposed approach makes it possible to compute a lower bound of the entropy rate per bit of the TRNG that can be used to guarantee its security but at the expense of its throughput.

Example 10. *In the case of the elementary generator, suppose that at time t, the attacker's knowledge of 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 in the red zone, otherwise it will be in 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 areas 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 attacker also knows the internal state of the PTRNG.

e TRNG.

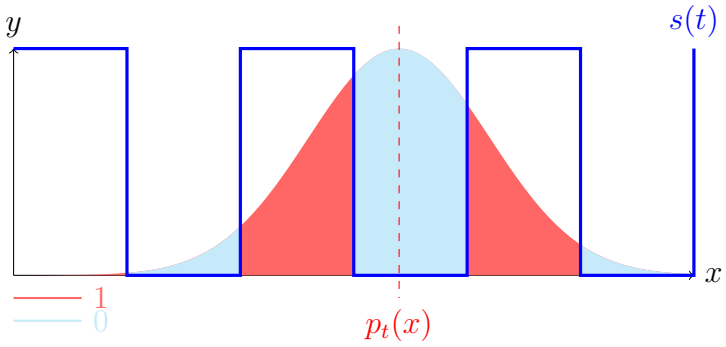

Figure 2: Effect of the analog-to-digital conversion: example considering the elementary PTRNG

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_C} 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, $i = 0, 1$ originating 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 also depends 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 the following requirement:

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

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

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

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

4 Fitting the model with the generator – 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 into the instability of delays in logic gates, which in turn, cause the phase noise of the clock signal generated by the ring oscillator. The phase noise is a complex phenomenon since it appears to be caused of different noise sources including thermal noise and flicker noise, among others. In our estimation of the proven entropy, we only take thermal noise into account.*

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

Example 13. *In the case of ring oscillator-based PTRNGs, several techniques can be used to measure the 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 that consist in reading the generator internal signals and analyzing them using external measurement equipment (e.g. an oscilloscope);*
- *internal measurement techniques that consist in analyzing the internal signals inside the device to determine the values of the input parameters of the model.*

The main advantage of external measurement techniques is that precise measuring equipment and advanced signal processing techniques can be used, e.g. advanced differential input/output techniques, high quality probes, and low noise oscilloscopes. However, these techniques have several disadvantages:

- *The input/output circuitry, transmission cables and input noise of the measuring equipment result in rather inaccurate measurements (that can however, be improved for example by using differential probes and high quality devices);*
- *the techniques are difficult to apply in a production line;*
- *testing each individual circuit can be very time-consuming and hence 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, require their own analysis. In particular, it is important to

filter out very high amplitude global deterministic noises that 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 evaluate whether the stochastic model fits the observed probability distribution and if the theory of tests of assumptions and parametric tests provide appropriate tools for this type of situation.

It is often difficult to evaluate *a priori* the measurement errors in Requirement 13. One possible approach is to make several measurements, while making sure that the experimental conditions remain stable and to estimate the standard deviation of different measurements. To compensate for the uncertainties of the different methods, it is advisable to corroborate the results using different types of experimental equipment (for example to combine various types of external measurement with various types of internal measurements).

Requirement 14. ([**Model fitting**] **Stability of parameters of the stochastic 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 respect to

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

Aging tests could also be performed.

The purpose here is to check the stability of parameters of the physical phenomena that guarantee the unpredictability of generated numbers from one device to

another as well as throughout the operating domain of the circuit to be sure that the security of the generator is maintained in the worst conditions.

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

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

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

Remark 9. Verifying the suitability of the stochastic model of sources of randomness is not straightforward. It is possible that a designer has only partial knowledge of how to model different types of physical noises. Concerning the elementary RNG, according to our recent state-of-the-art review, only thermal noise has already been modeled. It is very difficult to account for the impact of global deterministic noises on generated numbers, and, if great care is not taken, the predictions made by the PTRNG stochastic model may be rather conservative in comparison with 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 the required entropy rate.

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

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 $q_1, \ldots, q_n$ remain permanently within the bounds that guarantee a sufficient output entropy rate.

Example 15. *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 originating 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 with the stochastic model – with the parameter values that guarantee the 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 subsequently continuously.

Example 16. *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 originating from the thermal noise:*

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

Remark 11. A statistical test that cannot be interpreted as a parametric test is useless and may even be risky 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 the correct operation of the generator including all the blocks between the core of randomness and the generator output, 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 17. *Here we give as an example important classical 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 compare our proposed methodology for the evaluation of the TRNG design with the German document AIS 20/31 [1] and American standard NIST SP 800-90 [8] superseded by three other standards – NIST SP 800-90A [2], NIST SP 800-90B [9], and NIST SP 800-90C [10].
To facilitate the comparison, we firstrecall the methodology proposed in this document (illustrated in Fig. 3). Figure 3 clearly shows the relationship between the TRNG design, models, embedded tests, and testing procedures performed during security evaluation.
Recall that the TRNG hardware is composed of five types of blocks:

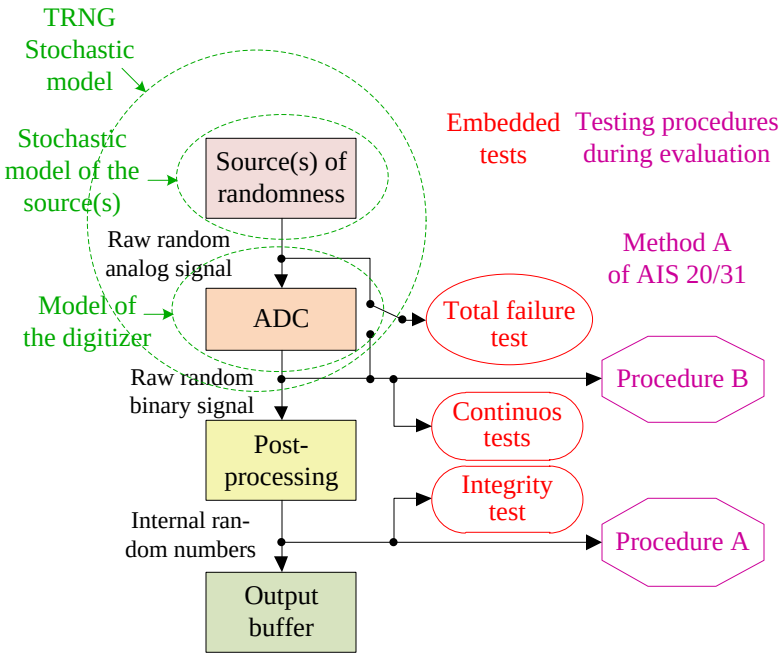

Figure 3: *TRNG design, modeling and testing as proposed in this document*

1. Block(s) containing source(s) of randomness,
2. An analog-to-digital converter (ADC),
3. A post-processing block,
4. An 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,
- A model of analog-to-digital conversion.

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

conversion is an intrinsic part of 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 the statistical parameters of the output signal. Usually, it increases the entropy rate per bit at the generator output at the expense of the output bit rate. The output buffer determines the format of the output data from the generator (i.e. the number and order of bits).
The second column of blocks presented in Fig. 3 represent embedded tests:

- A total failure test,
- Continuous test(s),
- An integrity test.

The role of the total failure test is to detect a total failure of the source of randomness as quickly as possible. Accordingly, 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 that continuously evaluate the values of the input parameters (physical quantities) of the stochastic model. If any of these parameters are smaller than what is required to obtain the expected entropy rate, the alarm shall be triggered.
The test of integrity verifies the correct operation of the whole data path between the core of randomness and the generator output, 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 three of the above mentioned types of tests. The TRNG output shall not be allowed until all the tests were successfully completed.
The last column of operations in Fig. 3 lists the testing procedures performed during the TRNG security evaluation. Since according to the requirements listed in this document, the raw binary signal (the digital noise) shall be available, only Method A of AIS 20/31 is acceptable, i.e. the digital noise shall be tested by Procedure B, which defines requirements concerning the execution of tests T6 to T8; the internal random numbers shall be tested using Procedure A, which defines requirements regarding the execution of tests T0 to T5.

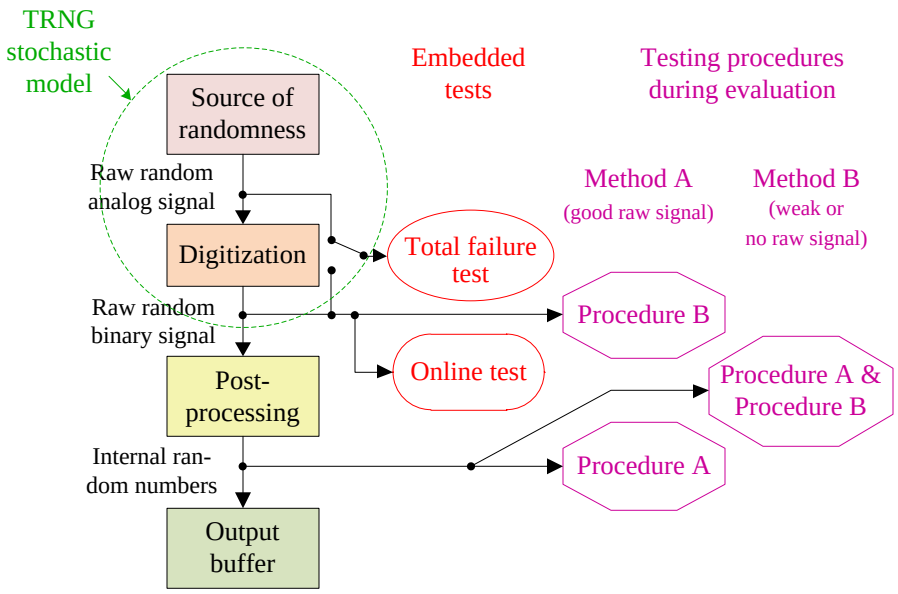

Figure 4: *TRNG design, modeling and testing according AIS 20/31*

6.1 Comparison with German AIS 20/31

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

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

The main differences can be observed by comparing Fig. 3, which illustrates the proposed methodology and Fig. 4, which presents the design, modeling and testing of TRNG according to AIS 20/31 . While the generator structure remains the same, its modeling and testing clearly differ.
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, the 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 that a good quality raw binary signal shall be available for off-line statistical testing. Comparison of Fig. 3 and Fig. 4 shows 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 also test the correct operation of the post-processing block.

6.2 Comparison with American NIST SP 800-90B

Compared with the American approach, we conclude that the methodology presented in this document is stricter. While the American approach only requires analysis of the source of randomness, (like AIS 20/31), our proposed methodology requires 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, to make the TRNG design compatible with the American approach, the required continuous tests should be added.
Comparison of Fig. 3 and Fig. 5 shows that although the structure of the generator is similar, the names of the blocks differ slightly. Namely, the ‘Source of

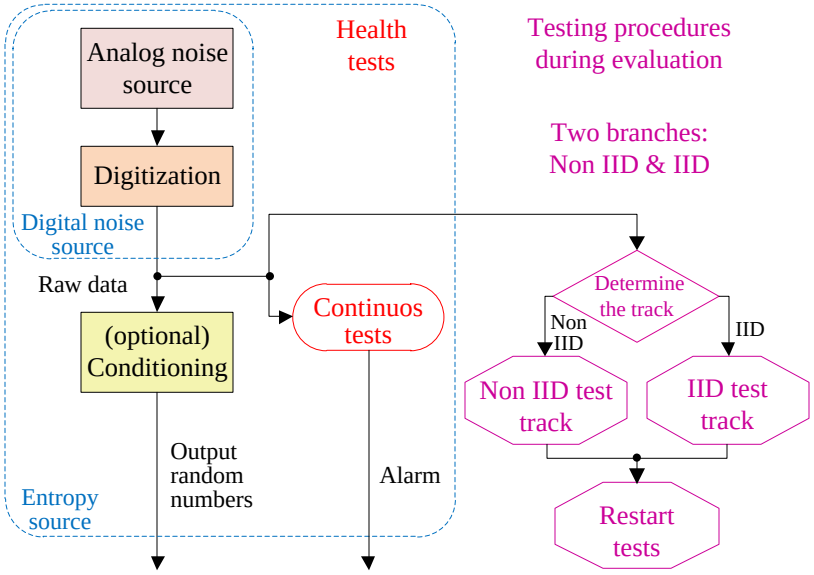

Figure 5: *TRNG design and testing according to NIST SP 800-90B*

randomness' from AIS 20/31 is called the 'Analog noise source' in NIST SP 800-90B. Although the role of the 'Conditioning element' presented in Fig. 5 is similar to that of the 'Post-processing block' in Fig. 4, (i.e. to improve the statistical quality of the output signal), the NIST SP 800-90B document recommends using one of proposed six vetted functions for this purpose (see [[9]] for more details).
The embedded tests defined by the NIST SP 800-90B document, called 'Health tests', can be divided into three categories: *start-up tests*, *continuous tests* (primarily on the noise source), and *on-demand tests*. The same test categories are considered acceptable in the present document.
The off-line testing procedures required by NIST SP 800-90B differ significantly from those required by AIS 20/31 and thus by the present document. Different tests are used for random variables that are independent and identically distributed (IID) from those which are not (Non IID). The aim of both tracks is to estimate the 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 right from the start. 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 provide recommendations for the design and security validation of a physical true random number generator implemented in an electronic device. The document begins with a review of the general objectives of the security evaluation process in which we introduce basic definitions and requirements concerning the PTRNG design. We explain, why the availability of the stochastic model of the physical phenomena used as sources of randomness is important to evaluate the security of the generator. The stochastic model of a complete PTRNG can be thus constructed as an ensemble of two models: a model of sources of randomness and a model of the analog-to-digital conversion process. We also explain that to further improve the security of the system, the embedded test must be adapted to the PTRNG stochastic model by continuously testing the values of the input parameters of the model and comparing them with thresholds that guarantee sufficient entropy rate at the generator output. Finally, the comparison with existing security standards clearly reveals the 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 the simple TRNG we use to illustrate our approach throughout this 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 the 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 noise which, in turns, determines the statistical bias of the bits of the TRNG output.
In what follows, it is assumed that Osc_1 is the oscillator that generates the sampled signal and Osc_0 produces the sampling clock signal.

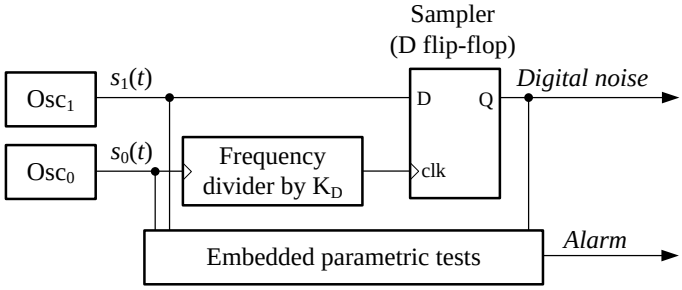

Figure 6: *Structure of an oscillator-based TRNG*

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 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 the 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, in particular ring oscillators. 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 noise. 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*.