Towards an Oscillator Based TRNG with a Certified Entropy Rate

David Lubicz and Nathalie Bochard

Abstract—We describe a practical and efficient method to estimate the entropy rate of a TRNG based on free running oscillators that does not require outputting and analyzing the clock signals with external equipment. Rather it relies on very simple computations that can be embedded in any logic device such as FPGA or ASIC. The method can be used for the calibration of an oscillator based TRNG or for online certification of its entropy rate. Our approach, which is inspired by the coherent sampling method, works under the general assumption that the period jitter is small compared to the period of the generated clock signal. We show that, in this case, it is possible to measure the relative phase between clocks of two oscillators with far higher precision than the time resolution given by the period of any internal clock signal. We use this observation to recover, under some reasonable heuristics, the distribution of the random walk component of the jitter, from which it is possible to obtain a lower bound on the entropy rate of the TRNG. Our method was thoroughly tested in simulations and in hardware. At the end of the paper, we draw some conclusions and make recommendations for a reliable implementation of TRNGs in cryptographic applications.

Index Terms—Hardware random number generators, ring oscillators, jitter model, entropy, statistical tests

1 Introduction

Random number generators (RNGs) are crucial components of cryptographic systems—typical applications include key generation, initialization vectors and even countermeasures against side-channel attacks. In addition to providing output bitstreams with good statistical properties, RNGs used for cryptographic applications must fulfill additional security requirements: they should be testable in real time and their security should be proved under thoroughly tested physical assumptions. In general, cryptographic RNGs comprise two stages: a *true random number generator* (TRNG in the following) that produces the entropy, and cryptographic post-processing, used to obtain a certain level of security even in the case of an undetected failure of the underlying TRNG (see [10]). In this paper, we are only concerned by the TRNG part of a RNG.

For cryptographic applications, the security of a TRNG depends on its *entropy rate* (or min-entropy rate), which is the measure of unpredictability of the TRNG. The usual way to evaluate the security of a TRNG is to use a general purpose statistical test suite (see for instance [13]). However, it is not possible to determine the entropy rate of a TRNG solely from knowledge of a long output sequence of the generator: the entropy rate has to be estimated using *statistical model* of the TRNG. The statistical model describes the distribution of the output sequence of a TRNG from knowledge of physical parameters. By approximating this distribution by a source of information, it is possible to recover the entropy rate of the TRNG.

A source of randomness commonly used in TRNGs implemented in logic devices such as field programmable gate arrays (FPGAs) and digital application specific integrated circuits (ASICs), is the instability of the signal propagation time across logic gates. This instability produces a *jitter* which is, by definition [11, p. 122], a short term deviation of a signal's transition time from its ideal position in time. This jitter is typically accumulated in so-called *ring oscillators*, which consist in a series of inverters or delay elements connected in a ring. The jitter can be extracted by means of a sampling unit, triggered by a reference clock, which can be an external clock signal or the output from another ring oscillator. This simple structure, which in this paper, we call an *elementary true random number generator* (elementary TRNG), and the underlying physical phenomena are widely reported in the literature, since the elementary TRNG is often used as a building block for on-chip TRNGs [3], [4], [15]. There are two classical ways to describe the jitter of a clock signal either as a edge-to-edge jitter or a edge-to-reference jitter: in the first approach, the edge timing is referenced to the preceding edge (*period jitter*) or to the Nth-preceding (*N-period jitter*), while in the other approach, the edge position is referenced to a separate reference signal (see [11, p. 127]).

The jitter is a complex phenomenon that results from the superposition of different noise sources. Following [16], it is important to distinguish the local component of the jitter, which is the entropy source of the TRNG, from the global deterministic noises, which can be manipulated from outside the device. But even the local component of the jitter is made of a combination of different types of noise with different statistical properties (see [14, p. 23]). In [1], the authors proposed a statistical model for an elementary TRNG, that computes the entropy rate originating from the

- *D. Lubicz is with the Ministère de la Défense, DGA-Maîtrise de l'information, BP 7419, F-35174 Bruz, France, and the Institut de Mathématiques de Rennes, Université de Rennes 1, Campus de Beaulieu, F-35042 Rennes, France. E-mail: david.lubicz@univ-rennes1.fr.*
- *N. Bochard is with the Laboratoire Hubert Curien/UMR CNRS 5516, Université de Lyon, 18 rue du Professeur Benoit Lauras, 42000 Saint-Etienne, France. E-mail: nathalie.bochard@univ-st-etienne.fr.*

Manuscript received 25 July 2013; revised 17 Dec. 2013; accepted 29 Dec. 2013. Date of publication 26 Feb. 2014; date of current version 13 Mar. 2015. Recommended for acceptance by B. Parhami.
For information on obtaining reprints of this article, please send e-mail to: reprints@ieee.org, and reference the Digital Object Identifier below.
Digital Object Identifier no. 10.1109/TC.2014.2308423

random walk phase noise. This statistical model has two parameters: the drift and the volatility parameters of the Wiener stochastic model. With knowledge of these two parameters, that characterize the random walk component of the jitter, it is possible to recover a lower bound on the entropy rate of an elementary generator. A usual way to characterize the jitter is to output the signal produced by the ring oscillator and to analyze it with an oscilloscope or a spectrum analyzer [11]. The problem with this technique is that it introduces extra jitter and distortions in the measured signal coming from the data acquisition chain. Moreover, it does not provide a simple method for quantifying parameters of individual chips on a production line or for checking the proper behavior of the circuitry when in operation. The purpose of this paper is to present a simple method, easy to implement, to measure the drift and volatility of the random walk component of the jitter of an elementary TRNG inside the chip.

Our method can be seen as a special purpose statistical test applied on the bitstream of an elementary TRNG (composed of two ring oscillators). Unlike other similar designs, we use TRNG with a high sampling frequency (i.e., comparable with the frequency of the sampled signal) and the statistical parameters of the generated bitstream are thus biased, since the variance of the jitter accumulated between two output bits is very small. We recover the statistical parameters of the source by analyzing the bias of the generated bitstream. Knowing these parameters, we can then compute a lower bound on the entropy rate of the source corresponding to the random walk component of the jitter. Using the lower bound on entropy rate and the statistical model in [1], designers can set the frequency of the sampling clock in order to guarantee the required entropy rate at the TRNG output. Our technique is a simple extension of the classical design of an elementary TRNG. It can be easily implemented in FPGA and ASIC and used to calibrate an elementary TRNG or to implement efficient dedicated online tests (specific to the TRNG principle) as required by AIS31 [10].

The paper is organized as follows: in Section 2, we detail the statistical parameters we want to measure. In Section 3, we analyze a simple intuitive method that can be used to measure them. We then explain why the results of this straightforward method are not completely satisfactory, in order to introduce more sophisticated approaches. In Section 4, we explain how to deduce the relative phase of a couple of oscillators from knowledge of the output sequence of an elementary TRNG. We use these results to present an algorithm aimed at computing statistical parameters of the jitter in Section 5. In Section 6 we present the experimental results and in Section 7 we describe some possible applications.

Notations used in the remainder of this paper. For all $x \in \mathbb{R}$ and $T \in \mathbb{R}$, we let $x \bmod T = x - \max\{i \in \mathbb{Z} | x - iT \geq 0\}T$. For interval I and $t \in \mathbb{R}$, $I + t$ is the interval $\{x + t | x \in I\}$. If I, J are intervals, $I + J$ is the interval $\cup_{t \in J} I + t$. We have to consider intervals that are invariant under translation by $T \in \mathbb{R}$. Thus, if $I \subset \mathbb{R}$ is an interval, we let $I_T = \cup_{n \in \mathbb{Z}}(I + nT)$. For instance, $[0,1)_2 = \cup_{i \in \mathbb{Z}}[2i, 2i+1)$. If $I = [x, y]$ is an interval, by convention, we set $I = \emptyset$ if $x > y$, and we have the obvious extension for open or semi-open intervals. For $x, y \in \mathbb{R}$, we let $d(x,y)_T = \min(|u - v|, u \in x + \mathbb{Z}T, v \in y + \mathbb{Z}T)$ be the distance modulo T.

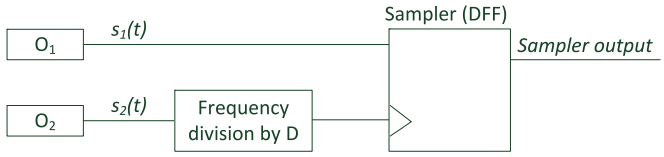

Fig. 1. Block diagram of an elementary TRNG.

2 The Period Jitter of an Elementary TRNG

We consider outputs of two oscillators O_i for $i = 1, 2$. They can be described by periodic functions of time t having the form

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

where f can be any real-valued function with period 1. For $i = 1, 2$, $\phi_i(t) = \omega_i(t + \xi_i(t))$ is the *total phase* of f and $\xi_i(t)$ is the phase shift due to the jitter. In the following, for $\alpha \in [0, 1)$, we define f_α as the unique real-valued one-periodic function such that $f_\alpha(x) = 1$ for all $0 < x < \alpha$, $f_\alpha(x) = 0$ for $\alpha < x < 1$, and $f_\alpha(0) = f_\alpha(\alpha) = 1/2$. We use f_α as a convenient model for the clock signal produced by a ring oscillator, which generally has imbalanced half periods ($\alpha \neq 1/2$). The reason we chose $1/2$ for the value of $f_\alpha(0)$ (rising edge of the clock signal) and $f_\alpha(\alpha)$ (falling edge) is to be coherent with the notations of [1] but in fact, this is of little concern for the results presented in this paper. We do not consider amplitude fluctuations since, as explained in [11, p. 134], their contribution to the jitter is negligible in the case of a clock signal. For $i = 1, 2$, ω_i is the mean frequency of the signal $s_i(t)$ and the function $\xi_i(t)$ models the absolute phase drift of this signal. Let $T_i = 1/\omega_i$ for $i = 1, 2$ be the mean period of $s_i(t)$.

An elementary TRNG is composed of two oscillators. The output of one is used to sample the output of the other by means of a sampling unit, for instance a D flip-flop (see Fig. 1). The frequency of the sampling oscillator is divided by D. The parameter D is very important since it makes it possible to reduce the sampling frequency and thus to accumulate the jitter for a longer time in order to increase the entropy rate of the output bit stream of the TRNG (while decreasing the bit rate). In the following, we assume that the output clock of oscillator O_1 is sampled at moments determined by the output of oscillator O_2. As we are mainly concerned with the relative phase between O_1 and O_2 output clocks, we make the simplifying assumption that O_1 is a perfectly stable oscillator and that the whole phase drift of the elementary TRNG comes from O_2 so that we have $\xi_1 = 0$ and we would like to characterize $\xi_2 = \xi$.

We make the assumption that the evolution of the total phase of O_2 can be modeled by an *ergodic stationary Markov process* $\Phi(t)$: for any time t, t_0, such that $t \geq t_0$, the phase $\Phi(t)$ conditioned by the value $\Phi(t_0) = x_0$ follows a probability distribution depending only on $\Delta t = t - t_0$ with mean $x_0 + \mu(\Delta t)$ and variance $\lambda(\Delta t)$ where λ, μ are real valued functions. In the following, we only consider a realization $\phi(t)$ of $\Phi(t)$ and use the ergodicity of the

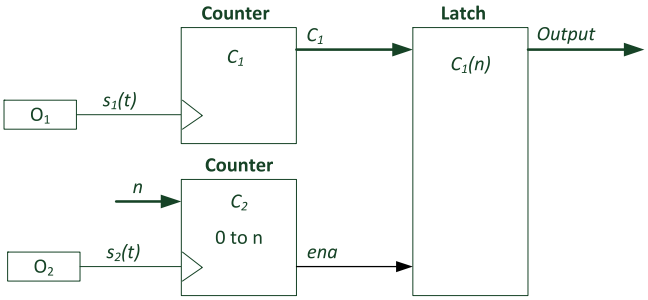

Fig. 2. Computation of Q and ν: the simple counter method.

process to compute probabilities that are independent of the time from the knowledge of the realization. For instance, as $\mathbb{P}\{\Phi(t_0+\Delta t)-x_0 \le x|\Phi(t_0)=x_0\}$ is independent of t_0 (it depends only on Δt), this cumulative distribution function can be computed by taking the probability of the realization over t_0: $\mathbb{P}_{t_0}\{\phi(t_0+\Delta t)-\phi(t_0)\le x\}$.

As $s_2(t)=f_\alpha(\omega_2(t+\xi(t)))$, where ω_2 is the mean frequency of s_2, we deduce that $\mu(\Delta t)=\omega_2\Delta t$. Thus, if the Markov process is Gaussian (i.e. $\frac{d}{dx}\mathbb{P}\{\Phi(t_0+\Delta t)\le x|\Phi(t_0)=x_0\}$ is a Gaussian distribution), it is completely determined by ω_2 and $\lambda(\Delta t)$. The random walk component of the jitter is produced by noise sources that affect each transition *independently*. It is described by a probability distribution $\frac{d}{dx}\mathbb{P}\{\Phi(t_0+\Delta t)\le x|\Phi(t_0)=x_0\}$ of mean $x_0+\omega_2\Delta t$ and variance $\sigma_0^2\Delta t$. Recall that D is the factor of the frequency divider of the elementary TRNG (see Fig. 1). Using [1, Proposition 1], it is possible, under this model, to compute the entropy rate of an elementary TRNG produced by the random walk jitter from the knowledge of $Q(D)=\frac{\sigma_0^2}{T_1^2}T_2D$ and $\nu=T_2/T_1$, where Q is the *quality factor* and ν is the *frequency ratio* introduced in [1, Section 2.4]. In this paper, we propose a method to determine the values of $Q=Q(1)$ and ν in order to obtain the entropy lower bound for the total entropy rate at the output of the generator. As Q and ν are dimensionless quantities, there is no a priori obstacle to computing these values from the data of an output bit stream of an elementary TRNG.

Other closely correlated noise sources, such as the $1/f$ noises, also contribute to the jitter. In this case, the variance of the jitter is in the form $\sigma_{1/f}(\Delta t)^2$. In practice, both uncorrelated and correlated noise sources exist and a typical log-log plot of $\lambda(\Delta t)$ versus the measurement delay Δt will demonstrate regions with slope 1 and 2 as explained in [7] (or see Fig. 3 below): for a shorter delay Δt the variance of the jitter grows linearly with Δt while for higher value of Δt the $1/f$ component of the jitter becomes dominant and the variance follows a quadratic law.

3 The Counter Method

Here we present a simple approach, introduced in [16], we call the counter method, to measure Q and ν inside the device. For $i=1,2$, the output signal of O_i increments counter C_i. When C_2 reaches a certain n, the value of C_1 is

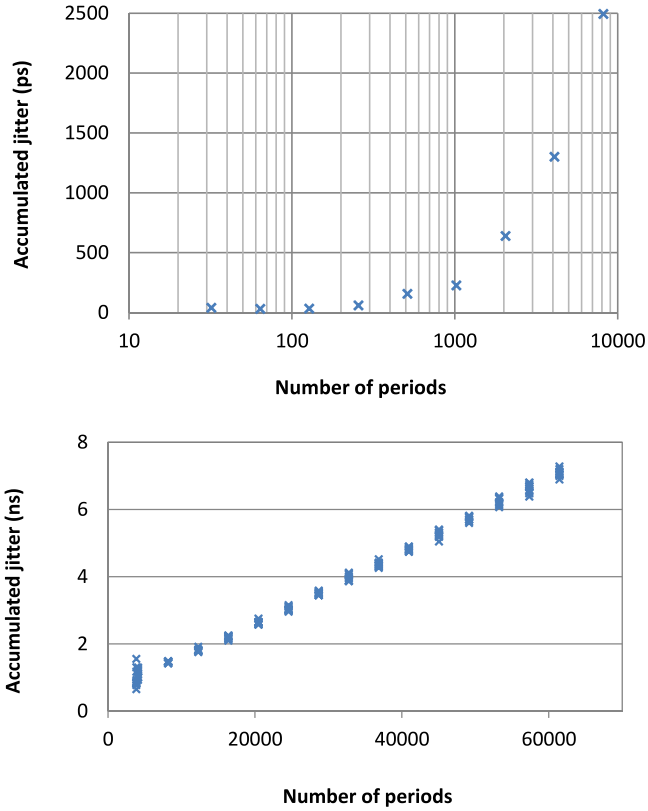

Fig. 3. Period jitter measured with an oscilloscope (top) and with the counter method (bottom).

stored (see Fig. 2). When this experiment is repeated, the different outcomes of the value of C_1, can be modeled by a random variable $\mathbf{C_1}(n)$. According to our model for the random walk noise of the jitter, the variance of $\mathbf{C_1}(n)$ is well approximated by $\frac{\sigma_0^2}{T_1^3}T_2n$ as long as n is sufficiently small that the $1/f$ component of the jitter is negligible. Thus, by computing the variance of $\mathbf{C_1}(n)$ for different values of n that satisfy the above condition, we obtain a linear law with the slope Q.

Denote by $E(\mathbf{C_1}(n))$ the expected value of $C_1(n)$. The expected time for the counter C_2 to reach the value n is nT_2. Thus we have $\lfloor n\frac{T_2}{T_1}\rfloor \le E(\mathbf{C_1}(n)) < \lfloor n\frac{T_2}{T_1}\rfloor+1$ from which we deduce that

$$\left|\frac{E(\mathbf{C_1}(n))}{n}-\frac{T_2}{T_1}\right| < \frac{1}{n}. \tag{2}$$

Denote by $\sigma_C(n)$ the standard deviation of $\mathbf{C_1}(n)$. Suppose that we obtain N outcomes $C_{1,i}$ for $i=1,\dots,N$ of the experiment. We can then evaluate $E(\mathbf{C_1(n)})$ by way of the estimator $\mathscr{E}(n)=1/N\sum_{i=1}^N C_{1,i}$ which follows a probability distribution with the standard deviation $\sigma_C(n)/\sqrt{N}$. By Bienaymé-Chebychev's inequality, for all $\epsilon>0$, $\mathbb{P}\{|\mathscr{E}(n)-E(\mathbf{C_1}(n)|>\epsilon\}\le\frac{\sigma_C(n)^2}{N\epsilon}$. By taking $\epsilon=1$ in the preceding equation and by combining it with Equation (2), we obtain:

$$\mathbb{P}\left\{\left|\frac{\mathscr{E}(n)}{n}-\frac{T_2}{T_1}\right|>\frac{2}{n}\right\}\le\frac{\sigma_C(n)^2}{N}. \tag{3}$$

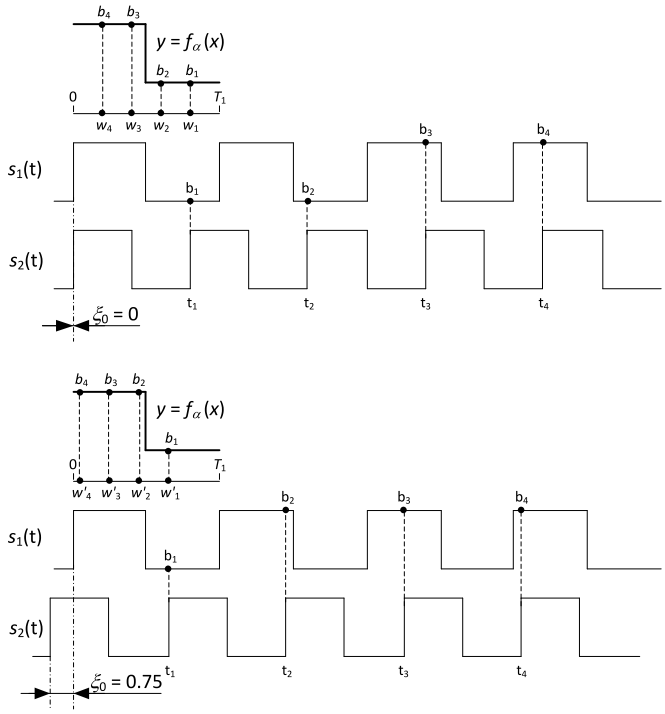

Fig. 4. If $\xi_0 = 0$ (top), we have $[b_1, \ldots, b_4] = [0, 0, 1, 1]$ and if $\xi_0 = 0.75$ (bottom), we have $[b_1, \ldots, b_4] = [0, 1, 1, 1]$.

We will see from experiments that we obtain an upper bound for $\sigma_C(n)$. As a consequence, Equation (3) shows that by ensuring that n and N are big enough, we can obtain an evaluation of T_2/T_1 with high probability and an arbitrarily small error.

This method was implemented using an evaluation board dedicated to TRNG benchmarking [5]. Different families from the three main manufacturers (Actel-Microsemi, Altera and Xilinx) were tested and gave similar results. In the following, we consequently present only one representative experiment featuring Altera Cyclone-III FPGA. The two oscillators were built using seven delay elements connected in a loop: one inverter and six non-inverting delay elements were mapped to LUT-based logic cells (LCELL) from the Altera library. The mean frequencies of the two oscillators were measured using a Lecroy WaveRunner 640ZI oscilloscope with D420 differential probes. The value measured on LVDS output was approximately 186 MHz. Fig. 3 (bottom) shows the standard deviation of $\mathbf{C_1}(n)$ as a function of the number n of accumulated periods T_2. We observe that under $8{,}000$ accumulated periods (corresponding to an accumulation time of less than $40\ \mu$s) the standard deviation of $\mathbf{C_1}(n)$ cannot be measured with the counter method since it is smaller than T_1. On the other hand, if the number of accumulated periods is over 8,000, the distribution of $\mathbf{C_1}(n)$ can be measured using the counter method, but the standard deviation increases linearly with the accumulation time. This means that the measurements are made in the frequency range in which the random walk noise we want to measure is dominated by $1/f$ noises.

Our experiments showed that the standard deviation of the period jitter accumulated during time nT_2 is much smaller than T_1 for n in the range 1 to 10^3. From Equation (3) we deduce that it is possible to measure T_2/T_1 with an error of less than $2 \cdot 10^{-3}$ with only one outcome of the counter method with $n = 10^3$. On the other hand, to produce an observable distribution of the outcomes of C_2, we have to wait for such a long time that the random walk noise is dominated by $1/f$ noise even if O_1 has the highest possible frequency achievable inside the chip (oscillator O_1 with one inverter and two delay elements). We conducted many experiments to identify the range of accumulation times in which the random walk noises are still measurable: by plotting the accumulated standard deviation of the period jitter as a function of the time in log scale (see Fig. 3 top), we saw that the order of magnitude of the accumulation time can not exceed a few hundred periods of the ring oscillators. In conclusion, the counter method is a simple and efficient way of precisely assessing the ratio T_2/T_1 but does not provide a way to obtain Q.

The time resolution of the jitter measurement can be improved by using the coherent undersampling method. The idea behind this approach is to magnify the distribution of the jitter by taking $T_1 = T_2 + \delta T$ with a very small δT, resulting in a temporal aliasing of the clock signal of the sampled oscillator. Unfortunately, this method requires a fine control of T_1 and T_2 in order to obtain sufficiently small δT. This is very difficult to obtain in practice and especially in FPGA, since it implies perfect control of the placement and routing of the ring oscillators. In the following, we explain how to tweak the coherent undersampling method so that it works under very general assumptions for any ratio T_2/T_1.

4 Relative Phase Measurement

In this section, we come back to the elementary TRNG described in Fig. 1. We set $D = 1$ so that the mean frequency of the sampling signal is ω_2. We denote by $(t_i)_{i\in\mathbb{N}}$ (resp. $(b_i)_{i\in\mathbb{N}}$) the time sequence (resp. the output bit sequence) corresponding to the rising edges of O_2. We have $t_i = iT_2 - \xi(t_i)$. For $i \in \mathbb{N}$, let $w_i = iT_2 \bmod T_1$.

Let $N \geq 1$ be an integer. A first important observation is that for $k \in \mathbb{N}$, the data of the pattern $[b_k, \ldots, b_{k+N}]$ gives a precise evaluation of the value $\xi(t)$ for $t \in [t_k, t_{k+N}]$. To this end, we first suppose that $\xi(t)$ is a constant ξ_0 in the time interval $[t_k, t_{k+N}]$. We denote by ρ_k (or simply by ρ when there is no ambiguity) the unique permutation of the set $\{0, \ldots, N\}$ such that for all $i, j \in \{0, \ldots, N\}$ with $i < j$, we have $w_{k+\rho_k(i)} < w_{k+\rho_k(j)}$. Then fact 1 says roughly that the pattern $[b_{k+\rho(0)}, \ldots, b_{k+\rho(N)}]$ is of the form

$$[1, \ldots, 1, 0, \ldots, 0, 1, \ldots],$$

where all ones (resp. zeros) are grouped together and the position of the transition from one to zero (resp. from zero to one) corresponds to the time of a falling edge (resp. a rising edge), from which we can deduce the phase ξ_0.

Before stating and proving Fact 1, we give a concrete example to illustrate it. Let $k = 1$ and $N = 3$, and suppose that $\xi(t) = \xi_0$ in the time interval $[0, 5.T_2]$. In addition we suppose that $\alpha = 1/2$ and that $T_2 \bmod T_1 = 4/5 \cdot T_1$ so that for $j = 1, 2, 3, 4$, $w_j = (5 - j)/5.T_1$ as it is at the top of Fig. 4. We have by definition $b_i = f_\alpha(w'_i)$ where $w'_i = w_i - \xi_0$. It is clear that for $i = 0, 1, 2, 3$, $\rho(i) = 3 - i$. By

looking at the position of the falling edge of $s_1(t)$ in Fig. 4, we see that $\xi_0 \in (-0.5, 0.5)$ if $[b_{1+\rho(1)}, \ldots, b_{1+\rho(4)}] = [1,1,0,0]$ and $\xi_0 = (0.5, 1.5)$ if $[b_{1+\rho(1)}, \ldots, b_{1+\rho(4)}] = [1,1,1,0]$. In other words, the data of the pattern $[b_{1+\rho(1)}, \ldots, b_{1+\rho(4)}]$ allows the value of ξ_0 to be recovered with a known error margin. This is exactly the content of Fact 1 except that in Fact 1, in order to recover the relative phase ξ_0, we consider the rising edge rather than the falling edge.

Fact 1. Suppose that for all $i \in \{0, \ldots, N-1\}$,

$$|w_{k+\rho(i)} - w_{k+\rho(i+1)}| < \min(\alpha T_1, (1-\alpha)T_1).$$

Suppose that there exists a couple of integers $i, j \in \{k, \ldots, k+N\}$ such that $b_i \neq b_j$. Then there exists $i_u \in \{0, \ldots, N\}$ such that

$$b_{k+\rho(i_u)} = 0 \text{ and } b_{k+\rho((i_u+1) \bmod (N+1))} = 1. \tag{4}$$

Set $x = w_{k+\rho(i_u)}$ and $y = w_{k+\rho((i_u+1) \bmod (N+1))}$, then we have $\xi_0 \in (x, y)_{T_1}$ if $i_u \in \{0, \ldots, N-1\}$ else $\xi_0 \in (x - T_1, y)_{T_1}$. Moreover, i_u satisfying condition (4) is unique.

Proof. It is clear that, under the condition of the statement, there exists $i_u \in \{0, \ldots, N\}$ such that $b_{k+\rho(i_u)} = 0$ and $b_{k+\rho((i_u+1) \bmod N+1)} = 1$. Set $K = k + \rho(i_u)$, $L = k + \rho((i_u + 1) \bmod (N+1))$. As $b_K = f_\alpha(\omega_1(x - \xi_0)) = 0$, by definition of f_α, we have $x - \xi_0 \in (\alpha T_1, T_1)_{T_1}$. In the same way, as $b_L = f_\alpha(\omega_1(y - \xi_0)) = 1$, $y - \xi_0 \in (0, \alpha T_1)_{T_1}$. From the above, we deduce that $\xi_0 \in (x, x + (1-\alpha)T_1)_{T_1} \cap (y - \alpha T_1, y)_{T_1}$. Thus, we have $\xi_0 \in \cup_{k \in \mathbb{Z}} I_k$ where $I_k = (\max(x, y - \alpha T_1 + kT_1), \quad \min(x + (1-\alpha)T_1, y + kT_1))_{T_1}$. Suppose first that $i_u \in \{k, \ldots, k+N-1\}$, then we have $x < y$ by definition of ρ. Thus, as by assumption $|x - y| < \min(\alpha T_1, (1-\alpha)T_1)$, we have

$$\max(x, y - \alpha T_1 + kT_1) = \begin{cases} x \text{ if } k \leq 0, \\ y - \alpha T_1 + kT_1 \text{ otherwise,} \end{cases}$$

and

$$\min(x + (1-\alpha)T_1, y + kT_1)_{T_1} = \begin{cases} y + kT_1 \text{ if } k \leq 0, \\ x + (1-\alpha)T_1 \text{ otherwise.} \end{cases}$$

We deduce that $I_0 = (x, y)_{T_1}$ and $I_k = \emptyset$ if $k \neq 0$, so that $\xi_0 \in (x, y)_{T_1}$. The case $x > y$ can be treated in the same way. The uniqueness of i_u is a consequence of the fact that each pair in the set of intervals $(w_{k+\rho(i)}, w_{k+\rho(i+1)})_{T_1}$ for $i \in \{k, \ldots, N+k-1\}$ and $(w_{k+\rho(N)} - T_1, w_{k+\rho(0)})_{T_1}$ has an empty intersection. Thus ξ_0 can only belong to a unique interval. □

It is possible to obtain a similar, more precise result by taking into account the small variability of $\phi(t)$ in the time interval $[t_k, t_{k+N}]$ (see the Appendix).

To exploit the above results, we need to compute the permutation ρ_k. A first remark is that for all $i \in \{0, \ldots, N\}$, we have $w_{k+i} = (w_k + w_i) \bmod T_1$ so that for all $k, k' \in \mathbb{N}$, we have $\rho_k = \rho^c \circ \rho_{k'}$ where ρ^c is a circular permutation. But the statement of Fact 1 depends only on the way we arrange the w_i for $i \in \{k, \ldots, k+N\}$ along the fixed circle $[0, T_1]$ (where we merged 0 with T_1) which, as a consequence, are independent of the permutation ρ^c. Thus, it is sufficient to be able to compute ρ_0. To compute ρ_0, one can obtain the value of w_i for $i \in \{0, \ldots, N\}$ and then use a sorting algorithm. Finally, as $w_i = iT_2 \bmod T_1$, the whole problem is reduced to the precise evaluation of $\zeta = T_2/T_1 \bmod 1$, which can be accomplished with the simple counter method described in Section 3. Knowing ζ, the simple Algorithm 1 makes it possible to recover the permutation ρ_0.

input : $\zeta = T_2/T_1 \mod 1$, N – the length of patterns.
output: The unique permutation ρ_0 of the set $\{0, \ldots, N\}$ such that for all $i, j \in \{0, \ldots, N\}$ with $i < j$, we have $w_{\rho_0(i)} < w_{\rho_0(j)}$.

Compute the list $L = [(0,0), (\zeta, 1), \ldots, (k\zeta \mod 1, k), \ldots, (N\zeta \mod 1, N)]$;
return L sorted by the first column ;

Algorithm 1: *Algorithm to compute the permutation* ρ_0

5 Algorithm of the Jitter Computation

We want to measure the distribution of the jitter accumulated during M periods of O_2 from the knowledge of a sample $[b_1, \ldots, b_n]$ of the output bit sequence of the elementary TRNG. We choose an $N < n$ and using the counter method, we can recover $\zeta = T_2/T_1 \bmod 1$. We define ρ as the unique permutation of the set $\{0, \ldots, N\}$ such that for all $i, j \in \{0, \ldots, N\}$ such that $i < j$, we have $(\rho(i)\zeta \bmod 1) < (\rho(j)\zeta \bmod 1)$. Let M be an integer and we would like to compute the cumulative distribution function $f_{MT_2}(x)$ given by $\mathbb{P}_t\{\xi(t + MT_2) - \xi(t) \leq x\} = \mathbb{P}_k\{\xi(t_k + MT_2) - \xi(t_k) \leq x\}$. We take a sufficiently small M so that the contribution of the $1/f$ noises to this distribution is negligible. Under this condition, as we explained in Section 3, the standard deviation of f_{MT_2} is small compared to T_1. To compute it, we only need to recover the values of $\xi(t) \bmod T_1$.

We consider patterns of the form $[b_{k+\rho(0)}, \ldots, b_{k+\rho(N)}]$. Under the assumption that $\sqrt{\lambda(NT_2)}$ is small, we know by Fact 1 (or by Fact 2 in the Appendix) that there will be only one $i \in \{0, \ldots, N\}$ such that $b_{k+\rho(i)} = 0$ and $b_{k+\rho((i+1) \bmod N+1)} = 1$. Still, by Fact 1, we know that there exists a $t \in [t_k, t_{k+N}]$ such that $\xi(t) \bmod T_1 \in (w_{k+\rho(i)}, w_{k+\rho((i+1) \bmod (N+1))})$. In the same way there exists a unique $j \in \{0, \ldots, N\}$ such that $b_{k+M+\rho(i)} = 0$ and $b_{k+M+\rho((i+1) \bmod N+1)} = 1$ and we know that there exists a $t \in [t_{k+M}, t_{k+M+N}]$ such that $\xi(t) \bmod T_1 \in (w_{k+M+\rho(j)}, w_{k+M+\rho((j+1) \bmod (N+1))})$. As a consequence, we approximate the value $\xi(t_k + MT_2) \bmod T_1 - \xi(t_k) \bmod T_1$ by $w_{k+M+\rho(j)} - w_{k+\rho(i)} = (w_M + w_{k+\rho(j)}) \bmod T_1 - w_{k+\rho(i)}$.

This principle is used by Algorithm 2 to compute the cumulative distribution function $\mathbb{P}_t\{\xi(t + MT_2) - \xi(t) \leq x\}$.

We define δ as

$$\min_{i \in \{0, \ldots, N\}}(d(w_{k+\rho(i)}, w_{k+\rho(i \bmod N+1)})_{T_1}).$$

The method works if $\sqrt{\sigma_0^2 NT_2} < \delta/2$ (see Appendix) so that we can apply Fact 2 and we also need $\sqrt{\sigma_0^2 MT_2} > \delta$ so that we can observe a distribution. Consequently, we must have $M > 4N$. As we will see in the next section, in practice these conditions are always fulfilled.

The method works better if we can assume that the sequence $(w_{\rho(i)})_{i=0,\ldots,N}$ is almost regularly spaced in the interval $[0, T_1]$. This means that there exists a constant $\delta_0 \approx 1/N \in \mathbb{R}$ such that for all $i \in \{0, \ldots, N\}$, we have $d(w_{\rho(i)}, w_{\rho((i+1) \bmod N+1)})_{T_1} = \delta_0$. It is always possible to choose N such that this condition is fulfilled. The theory of continued fraction [9, Theorem 16] states that for N it suffices to take the denominator of one of the convergents of the continued fractions development of $T_2/T_1 \bmod 1$. These denominators can be efficiently computed with the extended euclidean algorithm.

input: The output sequence $[b_0, \ldots, b_n]$ of an elementary TRNG with $D = 1$, the permutation ρ, $N < n$ the length of the patterns, M an integer such that we compute the period jitter accumulated during time MT_2.
output: A list of real numbers $L = [x_0, \ldots, x_\lambda]$, where $\lambda \in \mathbb{N}$ and x_i is an approximation of $\xi(t_k + MT_2) \mod T_1 - \xi(t_k) \mod T_1 + C$ for some k and the constant $C \in [0, T_1]$.

$k \leftarrow 0$;
$L \leftarrow []$;
while $k \leq n - M - N$ **do**
  Let i_u be the index such that $b_{k+\rho(i_u)} = 0$ and $b_{k+\rho((i_u+1) \bmod (N+1))} = 1$;
  Let j_u be the index such that $b_{k+M+\rho(j_u)} = 0$ and $b_{k+M+\rho((j_u+1) \bmod N+1)} = 1$;
  $L \leftarrow L + [w_{\sigma(j_u)} - w_{\sigma(i_u)}]$;
  $k \leftarrow k + M$;
end
return L ;

Algorithm 2: *Algorithm to compute the cumulative distribution function* $\mathbb{P}_t\{\xi(t + MT_2) - \xi(t) \leq x\}$

6 Experiments and Results

We thoroughly tested in simulations and in hardware the method of computing the relative jitter of two ring oscillator clocks described in this paper. In this section, we present the details and outcomes of our experiments.

Our demonstrator consists of an FPGA based hardware dedicated to TRNG evaluation composed of two parts: 1) a mother board that contains a Cypress USB interface device powered by an isolated low noise power supply, 2) a daughter module containing the FPGA component plugged into the motherboard [5].

Different daughter modules based on FPGA from three main manufacturers (Actel-Microsemi, Altera and Xilinx) are available. All modules have the same topology. This made it possible to compare the results of different technologies with maximum objectivity. The elementary TRNG was implemented in the daughter FPGA module and the output bit sequence was transmitted via USB interface to be analyzed by a software running on an external computer. All algorithms were implemented in Python.

Next, we describe step by step the application of our method to the elementary TRNG in Fig. 1 implemented in Altera Cyclone III FPGA (other families were tested but are not presented here as they produced similar results). The mean period of the sampled oscillator O_1 and sampling oscillator O_2, measured with an oscilloscope, was $T_1 = 11.335\,\text{ns}$ and $T_2 = 8.712\,\text{ns}$, respectively. We acquired a sequence $(b_i)_{i \in I}$ of 819,200 bits from the output of the elementary TRNG with $D = 1$.

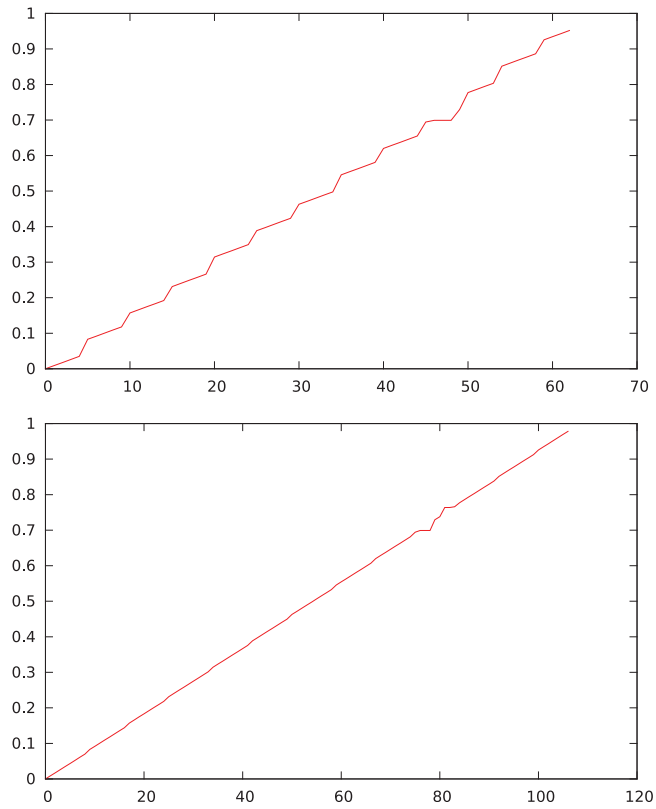

Fig. 5. Graph of the ordered sequence of $i\zeta \bmod T_1$ for $i = 1, \ldots, N$ for $N = 64$ (top) and $N = 108$ (bottom).

Knowing ζ, obtained with the counter method, we can recover the permutation ρ_0 by Algorithm 1. For patterns of length $N = 64$, the permutation is given by the following list L such that $L[i] = \rho_0(i)$:

```
0 13 26 39 52 9 22 35 48 61 5 18 31 44 57 1 14 27 40 53
10 23 36 49 62 6 19 32 45 58 2 15 28 41 54 11 24 37 50
63 7 20 33 46 59 3 29 42 16 55 12 25 38 51 8 21 34 47
60 4 17 30 43 56
```

The first 64 bits of the sequence $(b_i)_{i \in I}$ are

```
0011001100011001        1001100111001100
0011001100011001          1001100111001100
```

If we apply the permutation ρ_0 on this pattern, we obtain

```
                          ↓
0000000000000000 0000011111111111
1111111111111111 1111↓000000000000
```

Note that the ones and zeros in the pattern are grouped together as predicted by Fact 1 and Fact 2 and the arrows show the position of the rising and falling edges, from which we can evaluate $\xi(t) \bmod T_1$ for $t \in [0, NT_2]$.

We now explain how to choose N so that the values w_i for $i \in \{0, \ldots, N\}$ are regularly spaced. To this end, we used the list K of $w_i = i\zeta \bmod T_1$ for $i = 1, \ldots, N$ sorted by ascending order and plotted the graph of $(i, K[i])$. For $N = 64$, in the top part of Fig. 5 the bumpy curve shows that

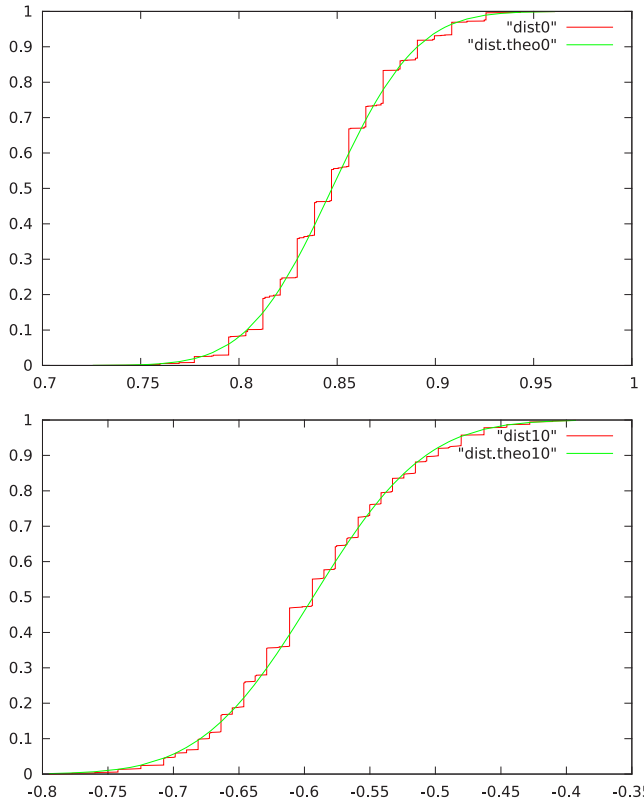

Fig. 6. Cumulative distribution function of the output of Algorithm 2 for $M = 400$ (top) and $M = 900$ (bottom); the horizontal time scale is such that 1 corresponds to $1/2\,T_1$.

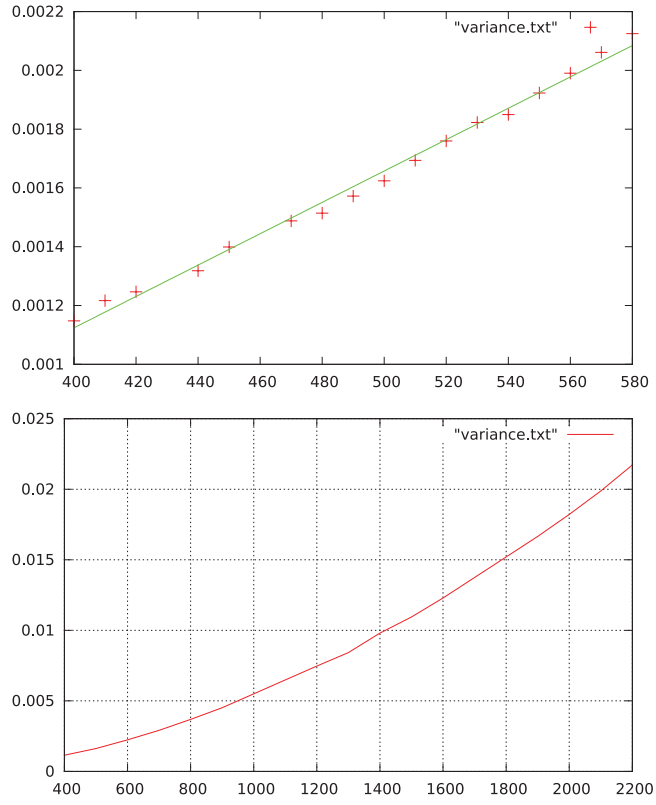

Fig. 7. $V(M)$ as a function of M.

the w_i are not regularly spaced in the interval $[0, T_1]$. We computed the convergents of the continued fraction development of ζ, that is $[0, 1/4, 3/13, 25/108, 28/121, \ldots]$ and for instance select $N = 108$. We obtained an almost regularly spaced sequence of w_i for $i \in \{0, \ldots, 108\}$ (see bottom of Fig. 5).

We used Algorithm 2 to recover a list $L(M)$ of points corresponding to approximations of $(\xi(t + MT_2) \bmod T_1) - (\xi(t) \bmod T_1)$ for some t. In Fig. 6, we show the cumulative distribution graph of $L(M)$ for $M = 400$ and $M = 900$. If $L(M) = [x_1, \ldots, x_\lambda]$, the graph of the cumulative distribution function is given by the points $(x_i, \#\{z \in L(M) | z \leq x_i\}/\lambda)$ for $i = 1, \ldots, \lambda$. This function corresponds to the cumulative distribution function of the Gaussian distributions shown in Fig. 6. We can then obtain the mean $E(M)$ and variance $V(M)$ of these distributions with the usual estimators that is, keeping the above notations : $E(M) = 1/\lambda \sum_{i=1}^{\lambda} x_i$, $V(M) = 1/\lambda \sum_{i=1}^{\lambda} x_i^2 - E(M)$.

Fig. 7 shows the graph of $V(M)$ as a function of M in the interval $[400, 580]$. In this interval, we see that $V(M)$ is well approximated by a linear function, whereas when the accumulation time is longer, the influence of the $1/f$ noise becomes noticeable (see the bottom graph). As Algorithm 2 computes distances between points of the form $w_i = iT_2/T_1 \bmod 1$, the slope of the graph of $V(M)$ for M in the domain where the random walk is dominant is a dimensionless quantity, which is nothing but $\frac{\sigma_0^2}{T_1^2} T_2$ where σ_0^2 is the variance of the jitter accumulated during one unit of time.

The fact that the output of our algorithm has no dimension is intrinsic to our method since we used only a bit sequence as input, which does not provide any way to measure the time. But the quantities $Q = \frac{\sigma_0^2}{T_1^2} T_2$ and T_2/T_1 are exactly what we needed to estimate the entropy rate of a TRNG as a function of the value of the frequency divider D with the statistical model of [1]. For TRNG implementation in Atera Cyclone III, we obtained $\frac{\sigma_0^2}{T_1^2} T_2 = 5.33484\,10^{-6}$.

To check that the proposed method gives a good approximation of $\frac{\sigma_0^2}{T_1^2} T_2$, we simulated an elementary TRNG using a VHDL simulator. In simulations, we dynamically modified the timing of two ring oscillators by adding a random jitter. The size of the jitter was used as an input parameter in the simulations. Using the proposed method, we were able to calculate the size of the jitter and compare it with the input jitter value. We used ModelSim to simulate two oscillators O_1 and O_2 with a Gaussian jitter added to each period. The output sampled bits were written into an output file during the simulations. The mean clock period of the sampled oscillator O_1 was $T_1 = 9{,}050\,\text{ps}$ and that of the sampling oscillator O_2 was $T_2 = 9{,}100\,\text{ps}$. For $i = 1, 2$, the frequency of the output clock signal of O_i was given by $f_i = f_{1/2}(1/T_1(t + \xi_i(t)))$, where ξ_i is the absolute jitter such that $\frac{d}{dx}\mathbb{P}\{\xi_i(t + T_i) \leq x | \xi_i(t)\}$ follows a Gaussian distribution of mean 0 and standard deviation σ_c. This is equivalent to oscillator O_1 with a fixed period and oscillator O_2 with a relative jitter $\xi(t)$ such that $\frac{d}{dx}\mathbb{P}\{\xi(t + T_2) \leq x | \xi(t)\}$ is a Gaussian distribution with standard deviation $\sigma_{T_2} = \sqrt{2}\sigma_c$

(see [1, Appendix C] for a justification). The quantity that our method approximates is thus $\sqrt{2}\sigma_c/T_1$. Note that our simulation does not take the $1/f$ noises into account.

For different values of σ_c we obtained output bit sequences $(b_i)_{i\in I}$ of length 197,780 bits. Next, as before, we can trace the graph of $V(M)$ as a function of M, which is well approximated by a linear function with slope a. Then we can compare $\sqrt{2}\sigma_c/T_1$ with $\sqrt{a}$. The results are summarized in Table 1, showing that we were able to recover the expected noise parameter with good precision.

We end this section with some concluding remarks about some implementation issues as well as some applications for our method. It should be noted that the implementation of the algorithms presented in this paper is quite straightforward if a basic floating point arithmetic is used. The only non-trivial part is the sorting algorithm included in Algorithm 1. Another important fact is that our algorithm can be implemented without changing the elementary TRNG and it can be executed continuously when the TRNG is in operation. The parameter D can be set to a higher value to achieve the required entropy rate at the output of the TRNG and *at the same time*, all the output bits of the generator can be used to internally compute the statistical parameters of the random walk component of the jitter (using $D = 1$ for computations).

The computation of $T_2/T_1 \bmod 1$ can also be used to compute small common approximate harmonics of T_1 and T_2. By small common approximate harmonics of T_1 and T_2, we mean small integers $\lambda_1, \lambda_2 \in \mathbb{Z}$ such that $|\lambda_1 T_1 + \lambda_2 T_2| < \epsilon$, where $\epsilon > 0$ is a small real number. It is interesting to be able to compute these harmonics in relation with the frequency lock phenomenon. It was shown in [12] that ring oscillators are sensitive to signal injection attack which lead to synchronization of the ring oscillators and a dramatic reduction in the entropy rate of the TRNG. Further analyses (see [2], [6]) showed that even in the absence of signal injection, ring oscillators can lock on small common harmonics of their frequency. It so happens that it is possible to internally compute small common harmonics of a couple of ring oscillators and, as a consequence, predict and detect the most probable frequency-lock occurrences. Indeed, by definition of small harmonics, we look for couple of integers (λ_1, λ_2) such that $|\lambda_1/\lambda_2 + T_2/T_1|$ is small. We know how to internally evaluate $T_2/T_1 \bmod 1$. So the problem is reduced to the computation of good rational approximation (with small integers) of a real number. But again the theory of continued fractions provides very simple and efficient algorithms to obtain such approximations (see [9, Theorem 16]).

7 Conclusion and Further Works

In this paper, we presented a simple method to recover the statistical parameters of the random walk component of the jitter of an elementary TRNG. Using the recovered parameters, it is possible to determine a lower bound on the generator's entropy rate, which is essential for TRNG security evaluation and certification. The main advantage of our method is that it does not require modification of the elementary TRNG and that the computations can be performed while the TRNG is in operation. Depending on the entropy assessment proposed, the generator's output bit rate can be set to a value that guarantees the entropy level specified by the security requirements.

TABLE 1
Simulation Results: Error Percentage

σ_c	a	$\sqrt{2}\sigma_c/T_1$	$\sqrt{a}$	Error percetage
$10\,ps$	$6.82494\,10^{-6}$	0.00156	0.00155	6%
$15\,ps$	$1.45836\,10^{-5}$	0.00234	0.00227	3%

The proposed method was validated in simulations and in hardware. Up to now, the analysis of the output bit sequence has been performed outside the device using a computer. However, we believe that it can be easily implemented in an FPGA or ASIC using common floating point arithmetic blocks. The use of integer arithmetic is also under consideration. The accuracy of our method has been evaluated experimentally, but it is possible to carry out exact computations using the techniques of [1]. We now plan to apply our technique as a countermeasure against attacks that attempt to lock the TRNG ring oscillators together or to some external frequencies.

Appendix

In Fact 1, we made the assumption that the jitter is constant in the time interval $[0, NT_2]$.We want to justify this assumption by establishing a similar result taking into account the small variability of $\xi(t)$ in the time interval $[t_k, t_{k+N}]$. Indeed, from experiments, we know that the standard deviation of the jitter accumulated during T_2, i.e., $\sqrt{\lambda(T_2)}$, is very small compared to T_1. For N, k integers, we define ρ as above and let $\delta = \min_{i\in\{0,\ldots,N\}}(d(w_{k+\rho(i)}, w_{k+\rho((i+1) \bmod N+1)})_{T_1})$. We assume that we have

$$\sqrt{\lambda(NT_2)} \approx \sqrt{\rho_0^2 NT_2} < \delta/(2\kappa), \tag{5}$$

where κ is an integer that will be set later on. The last approximation of (5) is true because we assume that N is small enough that the random walk model of the noise is valid.

As the Markov process $\xi(t)$ is well approximated in the time interval $[t_k, t_{k+N}]$ by a Wiener process, by the reflection principle [8, p. 346 (3.1)], we have

$$\mathbb{P}\left\{\max_{t\in[t_k,t_{k+N}]}(\xi(t)-\xi(t_k)) > \delta/2|\xi(t_k) = x\right\} = 2\mathbb{P}\{\xi(t_{k+N}) - \xi(t_k) > \delta/2|\xi(t_k) = x\}. \tag{6}$$

On the other hand, as $t_{k+N} - t_k \approx NT_2$ (we could be more precise here by saying that with probability near to 1 we have $t_{k+N} - t_k \approx NT_2$ still using the fact that the jitter is small in the time interval $[t_k, t_{k+N}]$), by Chebyshev's inequality and (5), we have

$$\mathbb{P}\{\xi(t_{k+N}) - \xi(t_k) > \delta/2|\xi(t_k) = x\} < 1/\kappa^2. \tag{7}$$

From (6) and (7), we deduce

$$\mathbb{P}\left\{\max_{t,t'\in[t_k,t_{k+N}]}(\xi(t)-\xi(t'))\ >\ \delta\right\}\ <\ 1/\kappa^2. \tag{8}$$

With this assumption, and keeping the notations of Fact 1, we can state

Fact 2. Suppose that for all $i\in\{0,\dots,N-1\}$,

$$|w_{k+\rho(i)}-w_{k+\rho(i+1)\ \bmod\ (N+1)}|\ <\ \min(\alpha T_1,(1-\alpha)T_1).$$

Let I_u be the set of $i_u\in\{0,\dots,N\}$ such that

$$b_{k+\rho(i_u)}=0 \text{ and } b_{k+\rho((i_u+1)\ \bmod N+1)}=1. \tag{9}$$

For $i_u\in I_u$, set $x=w_{k+\rho(i_u)}$ and $y=w_{k+\rho((i_u+1)\bmod(N+1))}$. Then, we have

$$\begin{aligned}\mathbb{P}\{\exists t\in[kT_1,\dots,(k+N)T_1]\text{ s.t. }\xi(t)\notin(x-\delta,y+\delta)_{T_1}\mid\\ I_u\neq\emptyset,i_u\in\{0,\dots,N-1\}\}\ <\ 1/\kappa^2,\end{aligned} \tag{10}$$

and

$$\begin{aligned}\mathbb{P}\{\exists t\in[kT_1,\dots,(k+N)T_1]\text{ s.t. }\xi(t)\notin(x-T_1-\delta,y+\delta)_{T_1}\mid\\ I_u\neq\emptyset,i_u=N\}\ <\ 1/\kappa^2.\end{aligned} \tag{11}$$

Moreover, $\mathbb{P}\{\#I_u\geq 2|I_u\neq\emptyset\}\ <\ 1/\kappa^2$.

Fact 2 says that *with a high probability* the pattern $[b_{k+\rho(0)},\dots,b_{k+\rho(N)}]$ is of the form $[1,\dots,1,0,\dots,0,1,\dots]$ where the transition from 0 to 1 corresponds to the time of a rising edge which gives a bound on the variation of $\xi(t)$ during the time interval $[t_k,\dots,t_{k+N}]$.

Proof. The proof follows the same reasoning as Fact 1. Assuming that $I_u\neq\emptyset$, let $i_u\in I_u$, we set $K=k+\rho(i_u)$, $L=k+\rho((i_u+1)\bmod N)$. As $b_K=f_\alpha(\omega_1(x-\xi(t_K)))=0$, by definition of f_α, we have $x-\xi(t_K)\in(\alpha T_1,T_1)_{T_1}$. In the same way, as $b_L=f_\alpha(\omega_1(y-t_L))=1$, $y-\xi(t_L)\in(0,\alpha T_1)_{T_1}$. Thus, $\xi(t_K)\in(x,x+(1-\alpha)T_1)_{T_1}$ and $\xi(t_L)\in(y-\alpha T_1,y)_{T_1}$. Using (8), we deduce that we have with a probability greater than $1-1/\kappa^2$, for all $t\in[t_k,t_k+N]$,

$$\begin{aligned}\xi(t)\in&((x,x+(1-\alpha)T_1)_{T_1}+[-\delta,\delta])\\ &\cap((y-\alpha T_1,y)_{T_1}+[-\delta,\delta]).\end{aligned}$$

If $i_u\in\{0,\dots,N-1\}$, we have $((x,x+(1-\alpha)T_1)_{T_1}+[-\delta,\delta])\cap((y-\alpha T_1,y)_{T_1}+[-\delta,\delta])=(x,y)_{T_1}+[-\delta,\delta]$ whence (10). If $i_u=N$, we have $((x,x+(1-\alpha)T_1)_{T_1}+[-\delta,\delta])\cap((y-\alpha T_1,y)_{T_1}+[-\delta,\delta])=(x-T_1,y)_{T_1}+[-\delta,\delta]$ from which we deduce (11).

In the case that $\#I_u\geq 2$, it means that there exists, $\{j_0,(j_0+1)\bmod(N+1),j_1,(j_1+1)\bmod(N+1)\}\subset\{0,\dots,N\}$, four distinct integers such that, for all $t\in[t_k,t_k+N]$, for $a=0,1$, we have

$$\xi(t)\in(w_{k+\rho(j_a)}-\alpha(j_a)T_1,w_{k+\rho((j_a+1)\ \bmod\ (N+1))})_{T_1}+[-\delta,\delta],$$

where $\alpha(j_a)=0$ for $j_a\in\{0,\dots,N-1\}$ and $\alpha(N)=1$.

The intersection of two such intervals is either empty, and in this case we are done, or is contained in an interval of the form $(w_{k+\rho(j)},w_{k+\rho((j+1)\ \bmod\ (N+1))}$ for $j=(j_0+1)\bmod(N+1)$ or $j=(j_1+1)\bmod(N+1)$. In the second case, for all $t\in[t_k,t_k+N]$, $\xi(t)\in(w_{k+\rho(j)},w_{k+\rho((j+1)\bmod(N+1))}$ but this is a contradiction with the hypothesis that $b_{k+\rho(j_0)}\neq b_{k+\rho((j_0+1)\bmod(N+1))}$. □

ACKNOWLEDGMENTS

The authors would like to thank Viktor Fischer for his assistance during the set up of the experiments as well as his many advices and careful reading of earlier version of the manuscript.

REFERENCES

[1] M. Baudet, D. Lubicz, J. Micolod, and A. Tassiaux, "On the security of oscillator-based random number generators," *J. Cryptology*, vol. 24, pp. 398–425, 2011.

[2] P. Bayon, L. Bossuet, A. Aubert, V. Fischer, F. Poucheret, B. Robisson, and M. Philippe, "Contactless electromagnetic active attack on ring oscillator based true random number generator," in *Proc. 3rd Int. Conf. Constructive Side-Channel Anal. and Secure Des.*, 2012, pp. 151–166.

[3] H. Bock, M. Bucci, and R. Luzzi, "An offset-compensated oscillator-based random bit source for security applications," in *Proc. 6th Int. Workshop Cryptographic Hardware Embedded Syst.*, 2004, pp. 268–281.

[4] M. Epstein, L. Hars, R. Krasinski, M. Rosner, and H. Zheng, "Design and implementation of a true random number generator based on digital circuit artifacts," in *Proc. 5th Int. Workshop Cryptographic Hardware Embedded Syst.*, 2003, pp. 152–165.

[5] EvaristeII. (June 2013). A modular hardware system for development and evaluation of cryptographic functions and random number generators. [Online]. Available: http://labh-curien.univ-st-etienne.fr/wiki-evariste-ii/index.php/Main_Page

[6] V. Fischer, "A closer look at security in random number generators design," in *Proc. 3rd Int. Conf. Constructive Side-Channel Anal. and Secure Des.*, 2012, pp. 167–182.

[7] A. Hajimiri, S. Limotyrakis, and T. Lee, "Jitter and phase noise in ring oscillators," *IEEE J. Solid-State Circuits*, vol. 34, no. 6, pp. 790–804, Jun. 1999.

[8] S. Karlin and H. M. Taylor, *A First Course in Stochastic Processes.*, second ed., New York, NY,USA: Academic Press, 1975.

[9] A. Ya. Khinchin, *Continued Fractions*. Chicago, IL, USA: Univ. of Chicago Press, 1964.

[10] W. Killmann and W. Schindler, "A proposal for: Functionality classes for random number generators, version 2.0," , Bundesamt fur Sicherheit in der Informationstechnik (BSI), Bonn, Switzerland, Tech. Rep., AIS20/AIS31 21.09.2011, Sep. 2011.

[11] W. Maichen, "Frontiers in Electronic Testing," *Digital Timing Measurements: From Scopes and Probes to Timing and Jitter*. New York, NY, USA: Springer, 2010.

[12] A. Theodore Markettos and S. W. Moore, "The frequency injection attack on ring-oscillator-based true random number generators," in *Proc. 11th Int. Workshop. Cryptographic Hardware Embedded Syst.*, 2009, pp. 317–331.

[13] *Recommendation for Random Number Generation Using Deterministic Random Bit Generators (Revised)*, NIST SP800-90, Mar. 2007.

[14] E. Rubiola, *Phase Noise and Frequency Stability in Oscillators*, (The Cambridge RF and Microwave Engineering Series). Cambridge. U.K.: Cambridge Univ. Press, Nov. 2008.

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

[16] B. Valtchanov, A. Aubert, F. Bernard, and V. Fischer, "Modeling and observing the jitter in ring oscillators implemented in FPGAs," in *Proc. IEEE 11th Workshop Des. and Diagnostics of Electron. Circuits and Syst.*, 2008, pp. 1–16.

David Lubicz received the PhD degree in mathematics from the University of Bordeaux in 1998, and then he joined the Cryptography Laboratory of DGA Information Superiority—the technical expert of the French Ministry of Defence in the fields of information superiority. Since 2008, he has been an associated researcher at the Institute of Mathematics of Rennes and is currently the head scientist of the cyber-defense division of DGA Information Superiority. His research interests include algorithmic number theory, asymmetric cryptography, cryptographic protocols, hardware random number generators, pseudorandom generators, and statistical tests.

Nathalie Bochard received the master's degree in electronic engineering in 1996 and the diploma of technological research (DRT) in vision, telecommunications and instrumentation from the University of Lyon, in 1997. She joined the CNRS (the official French research institution) in 1998, first at its laboratory Service Central d'Analyses in Lyon, and since 2001 at the Hubert Curien Laboratory in St-Etienne. Currently, her main research interests include embedded hardware cryptographic architectures for configurable logic devices and especially design, implementation and evaluation of true random number generators aimed at cryptographic applications.