J Cryptol (2019) 32:435–458
https://doi.org/10.1007/s00145-018-9291-2

Journal of
CRYPTOLOGY

From Physical to Stochastic Modeling of a TERO-Based TRNG

Florent Bernard
Laboratoire Hubert Curien, Université Jean Monnet, Member of Université de Lyon, 42000 Saint-Étienne, France
florent.bernard@univ-st-etienne.fr

Patrick Haddad
STMicroelectronics Advanced System Technology, 13790 Rousset, France
patrick.haddad@st.com

Viktor Fischer
Laboratoire Hubert Curien, Université Jean Monnet, Member of Université de Lyon, 42000 Saint-Étienne, France
fischer@univ-st-etienne.fr

Jean Nicolai
STMicroelectronics Advanced System Technology, 13790 Rousset, France
jean.nicolai@st.com

Communicated by François-Xavier Standaert.

Received 7 January 2016 / Revised 15 March 2018
Online publication 29 March 2018

Abstract. Security in random number generation for cryptography is closely related to the entropy rate at the generator output. This rate has to be evaluated using an appropriate stochastic model. The stochastic model proposed in this paper is dedicated to the transition effect ring oscillator (TERO)-based true random number generator (TRNG) proposed by Varchola and Drutarovsky (in: Cryptographic hardware and embedded systems (CHES), 2010, Springer, 2010). The advantage and originality of this model are that it is derived from a physical model based on a detailed study and on the precise electrical description of the noisy physical phenomena that contribute to the generation of random numbers. We compare the proposed electrical description with data generated in two different technologies: TERO TRNG implementations in 40 and 28 nm CMOS ASICs. Our experimental results are in very good agreement with those obtained with both the physical model of TERO's noisy behavior and the stochastic model of the TERO TRNG, which we also confirmed using the AIS 31 test suites.

Keywords. Hardware random number generators, Transition effect ring oscillator, Stochastic models, Entropy, Statistical tests.

1. Introduction

Random number generation is a critical issue in most cryptographic applications. Random numbers are used not only as confidential keys, but also as initialization vectors, challenges, nonces, and random masks in side-channel attack countermeasures. A security flaw in random number generation has a direct impact on the security of the whole cryptographic system. Unlike generators used in Monte Carlo simulations and telecommunications, those designed for cryptography must generate unpredictable random numbers—having perfect statistical properties is necessary but not sufficient.

There are two main categories of random number generators: deterministic random number generators (DRNG) and true random number generators (TRNG), which can be physical (P-TRNG) or non-physical (NP-TRNG). While deterministic generators are based on algorithmic processes and are thus not truly random, TRNGs exploit an unpredictable process, such as analog phenomena in electronic devices, to produce a random binary sequence or a sequence of random numbers. The unpredictability of DRNGs is guaranteed computationally and that of TRNGs is guaranteed physically. A good knowledge of the physical process underlying TRNG, which ensures its randomness and hence its unpredictability, is therefore necessary.

The statistical quality of TRNGs and DRNGs is usually evaluated using statistical test suites such as the one first proposed by George Marsaglia [8] and extended by NIST [10]. The goal of these suites is to detect statistical weaknesses such as non-uniformity or the appearance of patterns in a generated random sequence of only limited size. In no case can these tests guarantee the unpredictability of the random binary sequence.

As summarized by Fischer [3], the best way to evaluate unpredictability is to carefully estimate the entropy rate at the generator output. The estimation of entropy must be based on a carefully constructed stochastic model of the random number generation process. The stochastic model is a mathematical construct, which specifies the family of probability distributions that contains all possible distributions of the generated random numbers [7]. In a P-TRNG design, the model consists of a mathematical description of a link between the variations in the exploited unpredictable analog phenomena and the variations in the random binary sequence.

The main objective of using a stochastic model is to characterize the probability that an output bit is equal to one, and/or the probability that an n-bit output vector features a pattern of some sort. If the variables characterized by these probabilities are independent and identically distributed (IID), the entropy rate can be estimated from their distribution. If the variables are not IID, a conditional entropy rate based on conditional probabilities is usually computed [6].

Estimating entropy using an underlying stochastic model is mandatory in the security certification process, specially at high levels of security [7]. Stochastic models are reasonably easy to construct, but it is sometimes difficult or even impossible to check all the underlying physical assumptions. A physical model could serve as a basis for validation of these assumptions, but it is much more difficult to construct and a detailed knowledge of contributing physical phenomena is necessary.

Our objective was to model the generator recently proposed by Varchola and Drutarovsky [13], which uses a so-called transient effect ring oscillator (TERO) as a source of randomness. We chose this generator because it is small and easy to implement in

logic devices, and because it produces good statistical results. However, a satisfactory stochastic model is not yet available for this generator.

The generic stochastic model from [6] was clearly not suitable for the TERO-based TRNG. Neither were stochastic models dedicated to other existing generators, like the one proposed for the elementary ring oscillator-based TRNG in [1], nor that proposed in [12] for the TRNG using many oscillating rings as sources of randomness, nor the one proposed in [2] for the PLL-based TRNG. The models dedicated to structures with transient oscillations, which were proposed in [13] and [5], assume the distribution of generated random numbers to be Gaussian. This assumption disagreed with our own experience and even with the graphs presented in the original paper proposing TERO TRNG [13, p. 8].

For practical reasons—we had only a small number of samples, in which the TERO TRNG was implemented as an independent circuitry inside two complex logic devices, at our disposal—we could not study the design repeatability issues of the TERO TRNG architecture depending on manufacturing process conditions. Our main objective was thus to validate the proposed model and to study variation of model parameters across two different ASIC technologies at various operating conditions.

Our contributions (1) We propose and validate a novel physical TERO model including electric noises that serve as sources of randomness for a given instance of a TERO-based TRNG implemented in ASIC. (2) From the physical model, we derive a TERO stochastic model. (3) From the TERO model, we propose and validate a stochastic model of a complete TERO-based TRNG and illustrate the use of this model to estimate the entropy rate in conjunction with the output bit rate.

Organization of the paper In Sect. 2, we describe the structure of the TERO and its use in a P-TRNG. In Sect. 3, we present implementation of the TERO structure and corresponding TRNG in ASIC. The physical (electrical) and derived stochastic models of the TERO are detailed in Sect. 4. The stochastic model of the complete TERO-based TRNG is presented in Sect. 5. In Sect. 6, the effect of temperature and voltage variations on the TERO-based TRNG and on the model parameters is studied. We conclude the paper in Sect. 7 by a discussion concerning the relationship between the entropy rate and the output bit rate that can be set up using the proposed stochastic model.

2. TERO-Based RNG

TERO is an electronic circuit that oscillates temporarily. It is composed of two control gates that restart temporary oscillations and an even number of inverting logic gates connected in a loop. The number of inverting gates in the loop must be even; otherwise, oscillations would continue permanently like in standard ring oscillators.

Two typical TERO configurations are presented in Fig. 1a: in addition to two NAND gates used in both configurations, the TERO cell uses two chains of inverters (left panel) or two chains of non-inverting buffers (right panel). Consequently, the TERO can be seen as an RS latch with two inputs featuring the same voltage V_{ctr} and two different outputs V_{out1} and V_{out2}.

Figure 1b presents traces of the V_{ctr} input and V_{out1} output signal captured from oscilloscope. Following the rising edge of the V_{ctr} input, the outputs V_{out1} and V_{out2} start

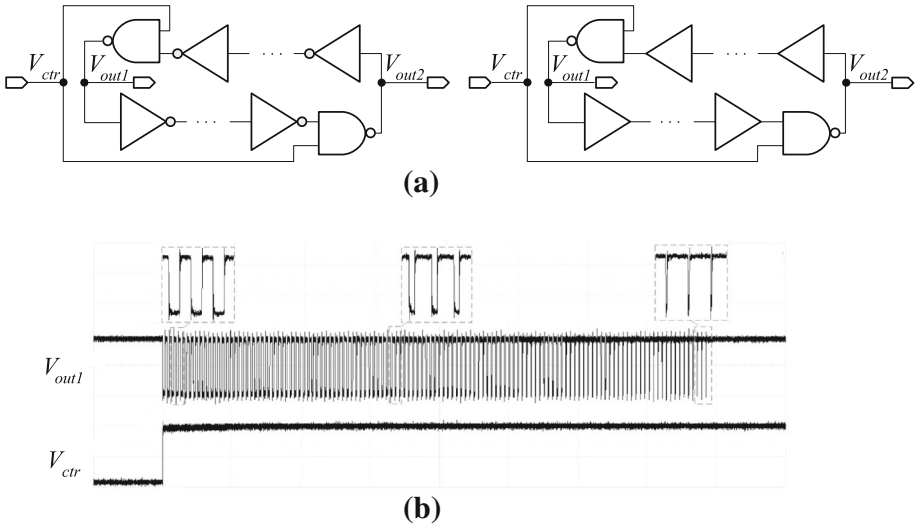

Fig. 1. Circuit diagram of two typical TERO structures (**a**) and TERO input/output waveforms (**b**) .

to oscillate: two rising edges start to propagate in the TERO cell in two opposite directions, and after traversing the NAND gate at the end of the branch, they are transformed into two falling edges, etc. Consequently, to enter the oscillatory state, the number of inverters in each branch of the TERO cell before the NAND gate must be even. Note that this condition is fulfilled automatically in the structure presented in the right panel in Fig. 1a, since each buffer present in this structure is realized in logic devices using a couple of inverters.

The oscillations obtained have a constant mean frequency, but their duty cycle varies over time: it changes monotonously, and after a certain number of periods, it reaches the rate of either 0 or 100%. At this point, outputs V_{out1} and V_{out2} stop oscillating and remain stable at two opposite logic values.

The three zooms in Fig. 1b show the changing duty cycle: immediately after the rising edge of the V_{ctr} signal, it is close to 50% and then decreases until it reaches 0%. Consequently, signal V_{out1} stabilizes at logic level 1. Of course, the signal V_{out2} behaves in the opposite way with respect to the duty cycle and stabilizes at logic level 0.

The number of oscillations before the outputs stabilize is not constant but varies because it is impacted by the electronic noises that disturb the normal behavior of transistors in the TERO structure.

The P-TRNG based on the TERO structure (TERO TRNG) is depicted in Fig. 2. The TERO circuitry is followed by an n-bit counter that counts the rising edges of the temporary oscillations. The counter output shows realizations of the random variable, i.e., the number of oscillations in successive control periods. The random binary sequence is usually obtained by successively concatenating the least significant bits of the counter, i.e., only one T flip-flop is needed in the counter.

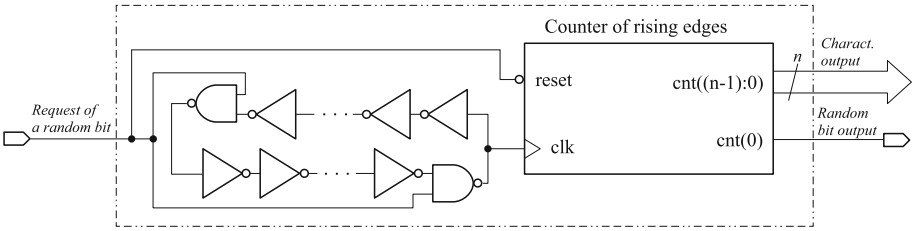

Fig. 2. True random number generator based on the TERO structure .

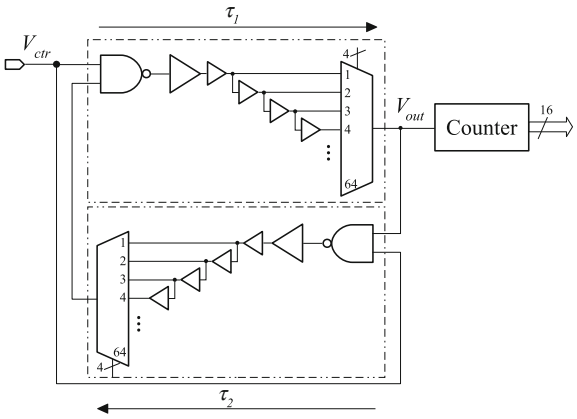

Fig. 3. TERO TRNG structure implemented in ASICs .

3. Implementation of the TERO RNG in ASIC

We implemented TERO in two of STMicroelectronics CMOS processes, with 40 and 28 nm minimum features, respectively. In order to explore the design space, we made the delays in the two TERO branches programmable, each in 64 linear steps (see Fig. 3). Each step consists of one elementary non-inverting buffer.

In the 40 nm process, the delays were programmable from 1.6 to 8 ns in 64 regularly spaced steps, resulting in oscillation frequencies in the range of 60–330 MHz. In the 28 nm process, the delays were programmable from 0.6 to 3.3 ns, resulting in oscillation frequencies in the range of 150–900 MHz. The number of oscillations was counted by a 16-bit counter.

Additional circuitry, not shown in the figure, made it possible to start the oscillations of the TERO circuitry with the *ctr* signal and to read the counter value only after the oscillation ended.

A particularly tricky issue in the physical layout consists of accounting for the routing delays, which, in such rapid processes, often dominate over the buffer delays. The multiplexers and the two NAND gates themselves add delays that also have be taken in consideration. So routing among the various multiplexers in the oscillation loop must be such that the overall delay in each of the 2 branches increases monotonously when

the number of buffers increases from 1 to 64. This requires a careful layout as well as post-layout simulations to guarantee the monotonicity.

This extra burden is only necessary when designing characterization chips. In the final design, the delays should be fixed, or with only a few adjustment steps. Nevertheless, the layout should always be undertaken with great care to control the delays as much as possible.

3.1. *Implementation Results*

We conducted extensive characterization campaigns on both processes. As expected, the adjustment of the delays τ_1 and τ_2 from Fig. 3 proved to be crucial in obtaining satisfactory results. In particular, we want to obtain a number of oscillations close to 100. With such a number of oscillations, we can assume that their variation comes mainly from the thermal noise inside transistors and that the realizations of the counter values are independent as it was shown in [4]. Indeed, we observed that for a significantly smaller number of oscillations the accumulated entropy was insufficient and for a number of oscillations too high the jitter coming from the flicker noise could cause the dependence between subsequent output samples to be non-negligible.

- When τ_1 and τ_2 are adjusted to the same value, the number of oscillations is usually extremely high, sometimes infinite (i.e., the oscillation never ends). This is of course not suitable in TRNG design. Values in which the delays differ by only 1 to 3 units (number of buffers) should also be avoided, as they are too close to infinite oscillation.
- When τ_1 and τ_2 are too different, the average number of oscillations is quite small (less than 30), usually resulting in a low entropy rate (because of a too weak jitter accumulation). This too should be avoided.

This leaves a narrow adjustment range for τ_1 and τ_2: the relative difference $\left|\frac{\tau_2-\tau_1}{\tau_2+\tau_1}\right|$[1] should not be greater than roughly 35%, yet still be greater than 5 or 10%. These ranges were observed experimentally, but it could be interesting, in a future work, to gain a full understanding of the underlying phenomenon in order to further enhance the physical model. The new model would help designers to choose appropriate values of τ_1 and τ_2 to control *apriori* expected number of oscillations.

Figure 4 shows distributions of the 8 million counter values obtained from ASIC devices in four different TERO configurations: two in the 40 nm technology (Fig. 4a, b) and two in the 28 nm technology (Fig. 4c, d). In Fig. 4a, the relative difference between the two TERO branches was 31%; in Fig. 4b, it was 35%; in Fig. 4c, it was 20%; and in Fig. 4d it was 32%. The differences between the TERO branches were obtained using the digital configurable delay chain depicted in Fig. 3.

It can be seen that in all cases the number of oscillations varied around a mean value according to a statistical law, which is apparently not a normal law. This is particularly clear in the right panels , but also observable in the left panels of the figure. One of our objectives was to determine this law and its origin.

[1] Denoted Δ_r later on the paper.

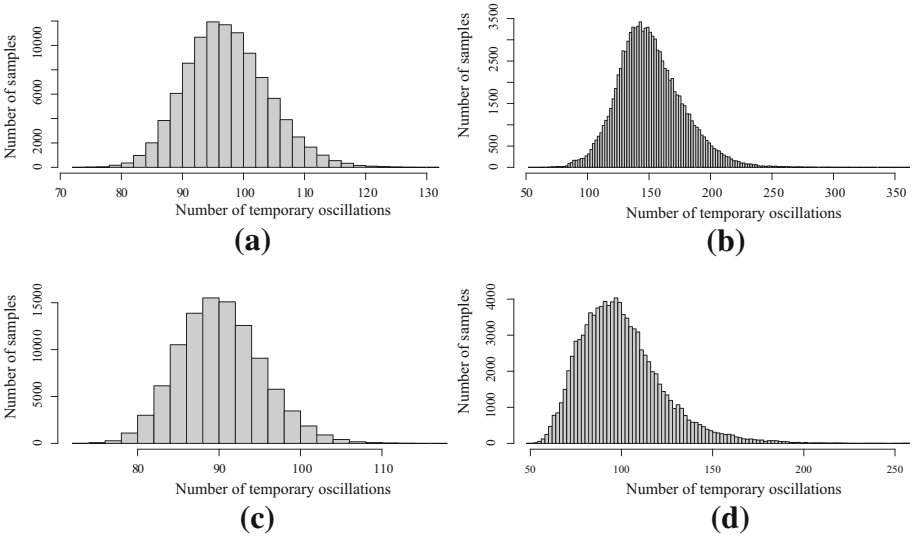

Fig. 4. Distribution of numbers of temporary oscillations for four TERO configurations—two in technology ST 40 nm (histogram **a**, **b**) and two in technology ST 28 nm (histogram **c**, **d**), with the following relative differences in delay between the two TERO branches: **a** 31%, **b** 35%, **c** 20%, and **d** 32% .

Before proceeding with the construction of the physical and stochastic models, we tested the statistical quality of the generated bit streams. The bit streams obtained by successive concatenation of the least significant bits constituted the raw binary streams, which were then tested using the AIS 31 protocol [KS11]. The data not only successfully passed all the tests of Procedure B, but also those of Procedure A aimed at testing the post-processed signals. This means that the generator is suitable for certification according to AIS 31 for PTG.1 and PTG.2 levels even without post-processing.

These good results are mitigated by the fact that they rely on accurate delay adjustments, which may not be compatible with large volume production. Extensive characterization is still needed to validate TERO usability in industrial contexts.

As explained above, successful evaluation of the output of the generator using statistical tests is a necessary but not sufficient condition to ensure the unpredictability of the generated numbers. The only way to guarantee such a property is to show the link between variations in the distribution of the raw random binary sequence and the physical phenomena that are considered as random, unpredictable, and non-manipulable. Statistical modeling of underlying analog and digital processes should make it possible to quantify the uncertainty included in the generated random sequence by estimating the entropy rate in this sequence.

4. Physical and Stochastic Models of TERO

In this section, we discuss the main processes that transform noisy electric currents into random binary sequences and explain how these phenomena are interlinked.

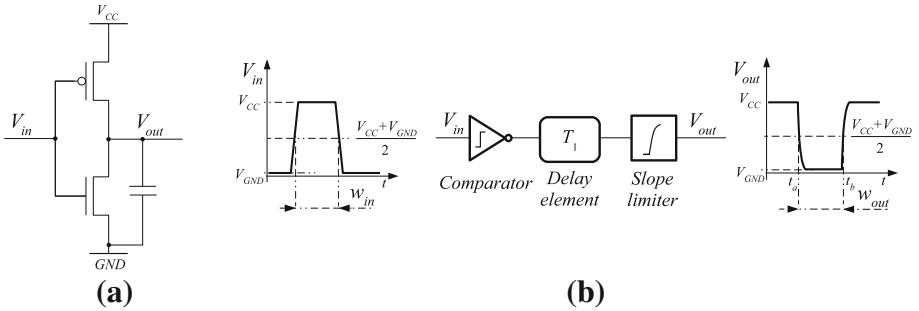

Fig. 5. Ideal noise-free CMOS inverter (**a**) and its physical model based on ideal components (**b**): comparator, delay element, and slope limiter (inverter input and output signals are also depicted) .

4.1. *Modeling the Number of Temporary Oscillations*

Our study was based on an existing physical model of RS latches published by Reyneri et al. [9]. We completed the noise- free model proposed by Reyneri et al. by taking electric noises into account.

4.1.1. *Modeling an Ideal Noise-Free Inverter*

First, we assume that TERO is built using ideal noise-free CMOS inverters as presented in Fig. 5a. This noise-free model is based on the physical model of an inverter with a variable slope published in [9]. We denote the input and output signals of such an inverter V_{in} and V_{out}, respectively. As presented in Fig. 5b, the model proposed in [9] divides the inverter into three entities:

- A comparator, which outputs V_{CC} if the input voltage V_{in} is smaller than $(V_{\text{CC}} + V_{\text{GND}})/2$; otherwise, it outputs V_{GND};
- A delay line, which delays comparator output signal by a static delay T_1;
- A slope limiter, which follows the delay line and generates the output signal V_{out}.

As depicted in Fig. 6, the model responds to a rising edge of the input signal by generating a signal that decreases linearly with the slope $-K_0$ until the output voltage reaches the value $(1 - K_0) \cdot V_{\text{CC}}$[2] after which the output decreases exponentially until it reaches the final value V_{GND}.

First, let we consider that the inverter input signal V_{in} has a linear form as presented in Fig. 5. We suppose that at $t = t_{\uparrow}$, signal V_{in} goes up from V_{GND} to V_{CC} and $\overline{t_a}$ is the time at which the output signal V_{out} is equal to $(V_{\text{CC}} + V_{\text{GND}})/2$. At time $t = t_{\downarrow}$, signal V_{in} goes down from V_{CC} to V_{GND}[3] and at $\overline{t_b}$ output V_{out} is equal to $(V_{\text{CC}} + V_{\text{GND}})/2$. Consequently, the width of the negative pulse at output V_{out} is equal to $w_{\text{out}} = \overline{t_b} - \overline{t_a}$. The output period signal is finished at $t = \overline{t_c}$, when V_{in} goes back to V_{CC}.

The authors of [9] describe the behavior of the inverter when the input signal has the same form as the described output signal. They show that in this case w_{out} can be

[2] Where K_0 is a positive real number smaller than 1.

[3] w_{in} can be defined as $w_{\text{in}} = t_{\downarrow} - t_{\uparrow}$.

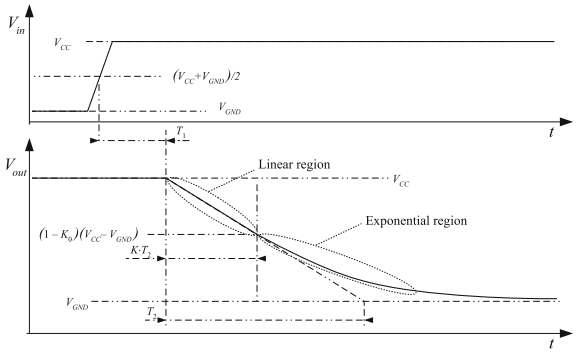

Fig. 6. Response of an ideal noise-free inverter to a step function .

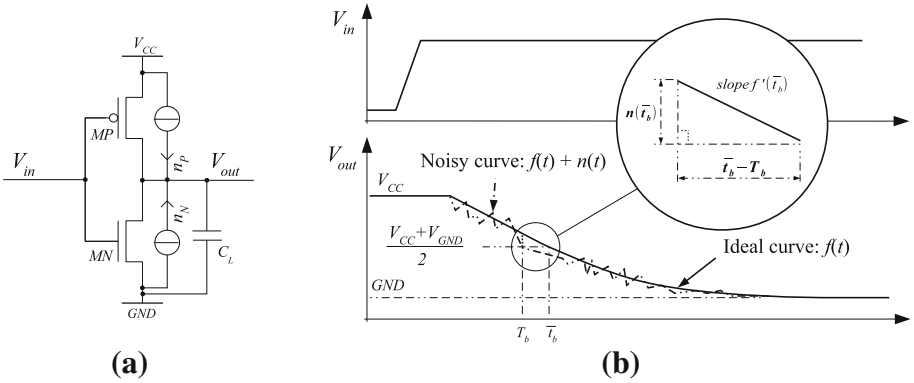

Fig. 7. Model of a noisy inverter (**a**) and its response to a step function (**b**) .

approximated by:

$$w_{\text{out}} = \frac{t_c}{2} + \left[w_{\text{in}} - \frac{t_c}{2} \right] [1 + H_d] \tag{1}$$

where $H_d = 2e^{\left(\frac{K_0 \cdot T_2 - \frac{t_c}{2}}{(1-K_0) \cdot T_2}\right)}$.

4.1.2. *Modeling a Noisy Inverter*

Noisy behavior at transistor level is modeled by noisy currents that are added to the ideal noise-free current flowing between the source and the drain. As can be seen in Fig. 7a for a CMOS inverter, these noisy currents can be represented by two sources of current n_N and n_P, which are connected in parallel to output transistors and are only active during inverter (gate) switching.

The inverter's noisy output V_{out} can be seen as the sum of two signals, $f(t)$ and $n(t)$:

– $f(t)$ represents an ideal component of the output signal, which contributes to the charge and discharge of the C_L capacitor by noise-free switching currents between the source and drain of output transistors MN and MP;

– $n(t)$ corresponds to the noisy component of the output signal, i.e., it contributes to the charge and discharge of the C_L by the noisy signals n_N and n_P.

Let t_0 be the last moment at which V_{out} is equal to V_{CC}. Since the noisy currents exist only during gate switching, $n(t_0) = 0$. It is therefore clear that:

$$n(t) = n(t) - n(t_0) = \frac{1}{C_L} \int_{t_0}^{t} [n_N(u) + n_P(u)] \mathrm{d}u$$

In the following, we assume that n_N and n_P are Gaussian random variables. This assumption is reasonable, because the noise currents can be considered as sums of random variables associated with independent quantum processes in the transistors. Consequently, $n(t)$ can be represented as a stationary Gaussian random process.[4]

Let us now analyze variations in the width of the pulse transmitted over one inverter as explained earlier in this section, but now in the presence of noisy currents. Let us consider that at $t = t_{\uparrow}$, signal V_{in} goes up from V_{GND} to V_{CC}, and we denote t_a the time, at which the signal V_{out} at the output of the inverter reaches $(V_{\text{CC}} + V_{\text{GND}})/2$. Similarly, at $t = t_{\downarrow}$, signal V_{in} goes down from V_{CC} to V_{GND} and t_b corresponds to the time at which V_{out} is equal to $(V_{\text{CC}} + V_{\text{GND}})/2$. Finally, at $t = t_{end}$ signal V_{in} goes back to V_{VCC}, ending one cycle. We denote $t_c = t_{end} - t_{\uparrow}$ the time that V_{in} needs to complete one cycle. For the sake of simplicity, we denote w_{in} the width of one (positive) pulse at signal V_{in} and w_{out} the corresponding (negative) pulse at the output of an open chain of inverters.

Proofs of the following lemma and propositions are provided in "Appendix A."

Lemma 1. *Let T_a (resp. T_b) be the random variable representing the time at which the signal V_{out} reaches $(V_{\text{CC}} + V_{\text{GND}})/2$ after a rising edge (resp. falling edge) on V_{in}. Let $\overline{t_a}$ (resp. $\overline{t_b}$) denote the ideal time at which V_{out} should reach $(V_{\text{CC}} + V_{\text{GND}})/2$ in noise-free conditions. Let W_{out} be the random variable representing the width of a pulse at signal V_{out} corresponding to a pulse of width w_{in} at signal V_{in}. Then, with the previous definitions of signals $f(t)$ and $n(t)$, we have:*

1. $T_a \sim \mathcal{N}\left(\overline{t_a}, \left(\frac{\sigma}{f'(\overline{t_a})}\right)^2\right)$ *and* $T_b \sim \mathcal{N}\left(\overline{t_b}, \left(\frac{\sigma}{f'(\overline{t_b})}\right)^2\right)$
2. *If T_a and T_b are independent,*

$$W_{\text{out}} \sim \mathcal{N}(\mu_{\text{out}}, \sigma^2_{\text{out}}) \text{ with } \begin{cases} \mu_{\text{out}} = \frac{t_c}{2} + \left(w_{\text{in}} - \frac{t_c}{2}\right)(1 + H_d) \\ \sigma^2_{\text{out}} = \sigma^2 \left(\frac{1}{(f'(\overline{t_a}))^2} + \frac{1}{(f'(\overline{t_b}))^2}\right) \end{cases}$$

where H_d is the constant introduced in Eq. (1).

[4] This may be not true at the device startup, but this assumption is reasonable after some time t_0. For each $t \geq t_0$, we assume that $n(t)$ follows a normal distribution with mean 0 and variance σ^2, denoted $n(t) \sim \mathcal{N}(0, \sigma^2)$ in the following.

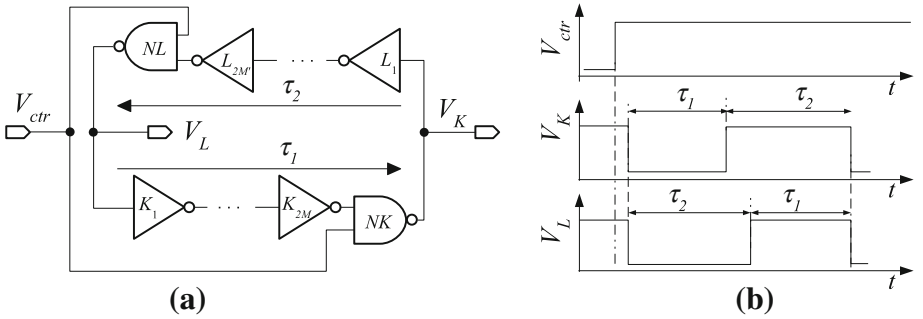

Fig. 8. TERO structure (**a**) and its initial behavior (**b**) .

4.1.3. *Shortening of the Pulse While it Traverses a Delay Chain*

Let us now consider the open chain of N inverters discussed in the previous section, where N is a nonzero positive integer. Let V_{in} be the input signal of the first inverter and V_{out_N} the output signal of the N^{th} inverter. W_{out_N} is the width of a pulse at V_{out_N} corresponding to a pulse w_{in} at signal V_{in}. The random behavior of W_{out_N} is given in Proposition 1.

Proposition 1. *If the noise source in the inverter is independent from the noise sources in other inverters, then*

$$W_{\text{out}_N} \sim \mathcal{N}(\mu_{\text{out}_N}, \sigma^2_{\text{out}_N}) \text{ with } \begin{cases} \mu_{\text{out}_N} = \frac{t_c}{2} + \left(w_{\text{in}} - \frac{t_c}{2}\right)(1 + H_d)^N \\ \sigma^2_{\text{out}_N} = \sigma^2_{\text{out}} \left(\frac{(1+H_d)^{2N}-1}{(1+H_d)^2-1}\right) \end{cases}$$

4.1.4. *Modeling Temporary Oscillations in the TERO Structure*

Let us now consider two chains of inverters, as discussed in the previous section. Let $\{K_j\}_{j=1\ldots2M}$ represent the set of inverters in the first chain and $\{L_j\}_{j=1\ldots2M'}$ those in the second chain. We denote NK and NL the two NAND gates with outputs V_K and V_L. They are connected to chains $\{K_j\}$ and $\{L_j\}$ (as depicted in Fig. 8a) and complete a TERO. If V_{ctr} is equal to V_{CC}, NK (resp. NL) can be seen as the K^{th}_{2M+1} (resp. $L^{\text{th}}_{2M'+1}$) inverter of the chain $K := \{K_j\}_{j=1\ldots2M+1}$ (resp. $L := \{L_j\}_{j=1\ldots2M'+1}$) generating the mean delay τ_1 (resp. τ_2). Theoretically, τ_1 and τ_2 can be identical, if both branches have the same topology. In practice, because of imperfections in the manufacturing process, their values always differ. Without any loss of generality, we can assume that $\tau_2 > \tau_1$.

At $t = 0$, let signal V_{ctr} go up from V_{GND} to V_{CC}. As shown in Fig 8b, this rising edge forces the outputs of NAND gates NK and NL to fall from V_{CC} to V_{GND}. The falling edge created at V_L (resp. at V_K) propagates over K (resp. L). This creates a pulse of mean width τ_1 (resp. τ_2) at V_K (resp. V_L).

The two rising edges created on V_K and V_L start to propagate over elements L and K. After a mean delay τ_2 (resp. τ_1), they cause signal V_K (resp. V_L) to fall from V_{CC} to

V_{GND}. The generated signals behave in the same way as the signals traversing set $\{I_j\}$ in the previous section with a cycle of width $t_c = \tau_1 + \tau_2$.

Proposition 2. *Let WK_0 (resp. WL_0) be the width of the pulse observed at signal V_K (resp. V_L) and WK_S (resp. WL_S) be the pulse width, once it has crossed S times over both sets K and L.*

If $WK_0 \sim \mathcal{N}(\tau_1,\ \sigma^2_{\mathrm{out}_{2M+1}})$ and $WL_0 \sim \mathcal{N}(\tau_2,\ \sigma^2_{\mathrm{out}_{2M'+1}})$ and if the noise sources in all the inverters are independent, then

$$WK_S \sim \mathcal{N}(\mu_{K_S},\ \sigma^2_{K_S}) \text{ with } \begin{cases} \mu_{K_S} = \frac{\tau_1+\tau_2}{2} + \frac{\tau_1-\tau_2}{2} R^S \\ \sigma^2_{K_S} = \sigma^2_{\mathrm{out}} \dfrac{R^{2S} R_M^2 - 1}{(1+H_d)^2 - 1} \end{cases}$$

$$WL_S \sim \mathcal{N}(\mu_{L_S},\ \sigma^2_{L_S}) \text{ with } \begin{cases} \mu_{L_S} = \frac{\tau_1+\tau_2}{2} + \frac{\tau_2-\tau_1}{2} R^S \\ \sigma^2_{L_S} = \sigma^2_{\mathrm{out}} \dfrac{R^{2S} R_{M'}^2 - 1}{(1+H_d)^2 - 1} \end{cases}$$

where $R_M = (1+H_d)^{2M+1}$, $R_{M'} = (1+H_d)^{2M'+1}$ and $R = R_M R_{M'} = (1+H_d)^{2M+2M'+2}$.

According to Proposition 2, $\mu_{L_S} + \mu_{K_S} = \tau_1 + \tau_2$, so the mean values of the duty cycles of signals V_K and V_L are always complementary. Since by definition, WK_S represents the width of the pulses observed at signal V_K and because of our assumption that $\tau_2 > \tau_1$, oscillations disappear when $\mathrm{WK}_S = 0$. Consequently, the number of oscillations N_{OSC} corresponds to the last value of S at which WK_S is positive:

$$N_{\mathrm{OSC}} = \max\{S | \mathrm{WK}_S > 0\}. \tag{2}$$

Let q be a positive integer different from zero. From Eq. (2), it follows that if N_{OSC} is greater than q, then WK_q is positive and different from zero, too. Using this fact, we can derive the probability that N_{OSC} is greater than q from Proposition 2:

$$\Pr\{N_{\mathrm{OSC}} > q\} = \Pr\{\mathrm{WK}_q > 0\}. \tag{3}$$

Then

$$\Pr\{N_{\mathrm{OSC}} > q\} = \frac{1}{\sqrt{2\pi}\sigma_{K_q}} \int_{\left[\frac{\tau_2-\tau_1}{2}\right] R^q - \frac{\tau_1+\tau_2}{2}}^{+\infty} \mathrm{e}^{-\frac{u^2}{2\sigma^2_{K_q}}}\, \mathrm{d}u, \tag{4}$$

or equivalently

$$\Pr\{N_{\mathrm{OSC}} > q\} = \frac{1}{2}\left[1 - erf\left(\frac{[\tau_2 - \tau_1]R^q - \tau_1 - \tau_2}{2\sqrt{2}\sigma_{\mathrm{out}}\sqrt{\frac{R^{2q} R_M^2 - 1}{(1+H_d)^2 - 1}}}\right)\right]. \tag{5}$$

Finally, from Eq. (5) we get the probability that N_{OSC} is smaller than or equal to q:

$$\Pr\{N_{\mathrm{OSC}} \leq q\} = 1 - \Pr\{N_{\mathrm{OSC}} > q\} = \frac{1}{2}\left[1 - erf\left(K\frac{1-R^{q-q_0}}{\sqrt{R^{2q}R_M^2 - 1}}\right)\right], \quad (6)$$

where K and q_0 are equal to:

$$K = \frac{\sqrt{R^2-1}}{2\sqrt{2}\sigma_r}, \quad (7)$$

$$q_0 = -\frac{\log(\Delta_r)}{\log(R)}, \quad (8)$$

and where

$$\sigma_r = \frac{\sigma_{\mathrm{out}}\sqrt{\frac{R^2-1}{(1+H_d)^2-1}}}{\tau_1+\tau_2} = \frac{\sigma_{\mathrm{out}_{2M+2M'+2}}}{\tau_1+\tau_2},$$

$$\Delta_r = \frac{\tau_2-\tau_1}{\tau_1+\tau_2}$$

Using Eq. (6), the probability p_q that N_{OSC} is equal to q (for $q \geq 1$) can be estimated by

$$p_q = \Pr\{N_{\mathrm{OSC}} \leq q\} - \Pr\{N_{\mathrm{OSC}} \leq q-1\},$$

$$p_q = \frac{1}{2}\left[erf\left(K\frac{1-R^{q-q_0-1}}{\sqrt{R^{2q-2}R_M^2 - 1}}\right) - erf\left(K\frac{1-R^{q-q_0}}{\sqrt{R^{2q}R_M^2 - 1}}\right)\right]. \quad (9)$$

Equation (9) is very important, because it can be used to model the distribution of the number of temporary oscillations. Its main advantage is that the parameters of the model (R, σ_r and Δ_r) are easy to quantify (see Sect. 4.2). Parameter R is the ratio of the geometric series and is related to the device technology and the number of inverters, σ_r is the relative jitter accumulated over $2M + 2M' + 2$ inverters, and Δ_r is the relative difference between TERO branches. The proposed model, as we will see later, can serve as a basis for the TERO TRNG stochastic model.

4.2. *Experimental Validation of the TERO Stochastic Model*

We validated the TERO model using the four TERO configurations presented in Sect. 2. We evaluated the appropriateness of the model using 65536 realizations $\{A_k\}_{k=1\ldots65536}$ of the TERO temporary oscillations. The model parameters R, Δ_r, and σ_r were computed from acquired data by determining K and q_0 from Eqs. (7) and (8) as follows:

1. First, the distribution of temporary oscillations N_{OSC} is obtained experimentally.

2. Equation (6) states that $\Pr\{N_{\mathrm{OSC}} \leq q\} = \frac{1}{2}$ for $q = q_0$, meaning that q_0 is the median of the distribution of temporary oscillations N_{OSC}:

$$q_0 = \mathrm{median}(N_{\mathrm{OSC}}).$$

3. The probability distribution $\Pr\{N_{\mathrm{OSC}} \leq q\}$ can be thus computed for each q:

$$\Pr\{N_{\mathrm{OSC}} \leq q\} \approx \frac{\#\{N_{\mathrm{OSC}} \mid N_{\mathrm{OSC}} \leq q\}}{\#\{N_{\mathrm{OSC}}\}}.$$

4. Then using this approximation, $Y(q) = erf^{-1}\Big(1 - 2\Pr\{N_{\mathrm{OSC}} \leq q\}\Big)$ can be computed. According to Eq. (6), $erf^{-1}\Big(1-2\Pr\{N_{\mathrm{OSC}} \leq q\}\Big) = K\frac{1-R^{q-q_0}}{\sqrt{R^{2q}R_M^2-1}}$, so

$$Y(q) \approx K\frac{1 - R^{q-q_0}}{\sqrt{R^{2q}R_M^2 - 1}}.$$

Knowing that $K = \frac{\sqrt{R^2-1}}{2\sqrt{2}\sigma_r}$ and $\sigma_r = \sigma_{\mathrm{out}}\sqrt{\frac{R^2-1}{(1+H_d)^2-1}}/(\tau_1 + \tau_2)$, K can be expressed as

$$K = (\tau_1 + \tau_2)\frac{\sqrt{(1+H_d)^2 - 1}}{2\sqrt{2}\sigma_{\mathrm{out}}} = (\tau_1 + \tau_2)\frac{\sqrt{R^{\frac{1}{M+M'+1}} - 1}}{2\sqrt{2}\sigma_{\mathrm{out}}}$$

and $Y(q)$ as

$$Y(q) = \underbrace{\frac{(\tau_1 + \tau_2)}{2\sqrt{2}\sigma_{\mathrm{out}}}}_{K'}\frac{(1 - R^{q-q_0})\sqrt{R^{\frac{1}{M+M'+1}} - 1}}{\sqrt{R^{2q}R_M^2 - 1}}. \tag{10}$$

5. Finally, the value of R is determined. Knowing that $R \sim 1$ and $R > 1$, the value R_{loop}, such that the ratio $Y(q)/Z(q)$ is almost constant (i.e., independent from q), is searched in a loop for $R > 1$ in the neighborhood of 1 . This constant named K' represents an approximation of the value $\frac{(\tau_1+\tau_2)}{2\sqrt{2}\sigma_{\mathrm{out}}}$. As mentioned above, $Y(q)$ was obtained experimentally and $Z(q)$ is derived from Eq. (10) as follows:

$$Z(q) = \frac{(1 - R_{loop}^{q-q_0})\sqrt{R_{loop}^{\frac{1}{M+M'+1}} - 1}}{\sqrt{R_{loop}^{2q}R_M^2 - 1}}. \tag{11}$$

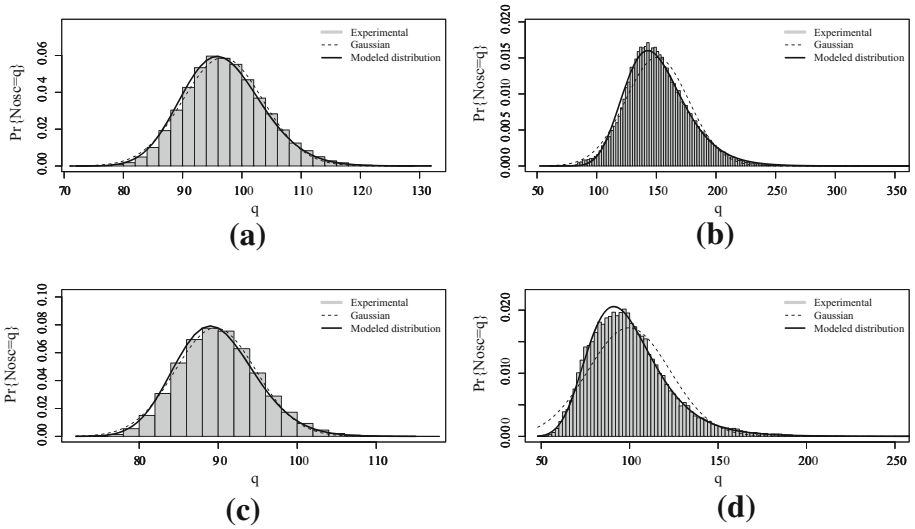

Fig. 9. Experimental validation of the model for two TERO topologies in technology ST 40 nm (graphs **a**, **b**) and ST 28 nm (graphs **c**, **d**), with the following relative differences in delay between the two TERO branches: **a** 31%, **b** 35%, **c** 20%, and **d** 32% .

Then when this particular R and the constant K' are found, we finally compute the two last parameters of the model

$$\sigma_r = \frac{\sqrt{R^2 - 1}}{2\sqrt{2}K'\sqrt{R^{\frac{1}{M+M'+1}} - 1}}$$

and

$$\Delta_r = R^{-q0}.$$

The results are presented in Fig. 9. The distribution depicted in Fig. 9a was obtained using parameter values: $R = 1.01221$; $\Delta_r = 0.3081$; $\sigma_r = 0.00205$, the distribution in Fig. 9b was modeled with parameters: $R = 1.00701$; $\Delta_r = 0.3531$; $\sigma_r = 0.00398$, the distribution in Fig. 9c had: $R = 1.01841$; $\Delta_r = 0.1936$; $\sigma_r = 0.00173$, and finally the distribution in Fig. 9d was modeled with parameters: $R = 1.01191$; $\Delta_r = 0.3171$; $\sigma_r = 0.00615$.

In the following section, we will use our model to estimate entropy at the TERO TRNG output.

5. Stochastic Model of the Complete TERO-Based TRNG

Let H_{osc} be the entropy contained in the sequence of number of oscillations N_{osc}. Since realizations of N_{osc} are assumed to be independent (the generator is restarted periodically

and is thus memory-less), this entropy is related to p_q from Eq. (9) as follows:

$$H_{N_{\text{osc}}} = -\sum_{q \in \mathbb{N}} p_q \log_2(p_q)$$

We computed the value of $H_{N_{\text{osc}}}$ for the four distributions depicted in Fig. 9. The distribution shown in Fig. 9a had the entropy rate per sample (per byte) $H_{N_{\text{osc}}} = 4.80$, that in Fig. 9b had the entropy rate $H_{N_{\text{osc}}} = 6.76$, the distribution in Fig. 9c had the entropy rate $H_{N_{\text{osc}}} = 4.39$, and in the fourth case we obtained $H_{N_{\text{osc}}} = 6.42$.

Let p_b be the probability that the least significant bit of N_{osc} is equal to 1. This probability is related to p_q from Eq. (9) as follows:

$$p_b = \sum_{k=0}^{k=+\infty} p_{2k+1}. \tag{12}$$

For each realization, we select the least significant bit of N_{osc} to form a vector $(b_{n-1} \ldots b_0)_2$. This vector can be interpreted as a binary number $B_n \in \{0, \ldots, 2^n - 1\}$. As the TRNG is restarted after each acquisition of N_{osc}, bits $(b_k)_{k=0 \ldots n-1}$ are independent. Thus, for each n-bit integer $X_n = (x_{n-1} \ldots x_1 x_0)_2$

$$p_{X_n} = \Pr(B_n = X_n) = \prod_{j=0}^{n-1} [1 - p_b]^{1-x_j} [p_b]^{x_j}.$$

If the random process associated with B_n is stationary, the entropy per bit at the generator output is equal to [11]:

$$H = \lim_{n \to +\infty} \frac{H_n}{n},$$

where

$$H_n = -\sum_{X_n \in \{0, \ldots, 2^n - 1\}} p_{X_n} \log_2(p_{X_n}).$$

Since jitter realizations are assumed to be independent, realizations of N_{osc} and b_k are also assumed to be independent. Consequently, we consider that the generator has no memory and consequently that the generated random bits do not contain any short- or long-term dependencies. The Shannon entropy per bit at the generator output derived from our model can thus be simplified as follows:

$$H_b = -p_b \log_2(p_b) - (1 - p_b) \log_2(1 - p_b).$$

We computed the entropy rate per bit for the four TERO configurations discussed in Sect. 4.2. The model parameters and entropy estimations for four TERO configurations having histograms from Fig. 4 are presented in Table 1.

Table 1. Model parameters and entropy estimation for the four TERO TRNG configurations featuring histograms from Fig. 4.

Technology	ST 40 nm		ST 28 nm	
TERO configuration	(a)	(b)	(c)	(d)
R	1.01221	1.00701	1.01841	1.01191
Δ_r	0.3081	0.3531	0.1936	0.3171
σ_r	0.00205	0.00398	0.00173	0.00615
$H_{N_{\text{osc}}}$	4.801523	6.761983	4.390844	6.423837
H_b	> 0.9999	> 0.9999	> 0.9999	> 0.9999

As can be seen, in all cases, the entropy rate at the least significant bit was higher than 0.9999, meaning that the entropy per bit exceeded the value required by AIS 31. This was in perfect agreement with the experimental results of the tests AIS 31 presented in Sect. 3.1.

Although the distribution of counter values is shown to be well characterized by our model, we are aware that this distribution itself does not stipulate that probabilities of 0's and 1's at the TRNG output are balanced. Indeed, to verify the validity of the model, we must ensure that no bit patterns or autocorrelations could be observed at the TRNG output. To check this, we computed the autocorrelation coefficients for the least significant bit of the counter for a 10,000-bit sequence, while shifting the output sequence by 1 to 40 bits. (The autocorrelation naturally decreases as the shift increases.) As can be seen in Fig. 10, the obtained autocorrelation values were close to 0 for shifts > 0 inside the confidence interval represented by the two horizontal dotted lines.

6. Impact of Temperature and Voltage Variations

The measurement results presented in the previous sections have been obtained under nominal operating conditions (voltage and temperature). In the next step, we observed generator output values and variation of the model parameters (σ_r, Δ_r and R) in varying conditions. Following our conservative approach, we wished to determine the lower bound of entropy per bit that can be achieved even in the worst case.

A TERO cell featuring $M = 18$ and $M' = 20$ with the following parameters computed under nominal conditions ($T = 25\,^{\circ}\text{C}$ and $V = 1.1\,\text{V}$):

- $R = 1.01911$,
- $\Delta_r = 0.1506238$,
- $\sigma_r = 0.000525218$,
- Mean number of oscillations: $\overline{N}_{\text{osc}} = 126$

was first placed into an environmental simulation chamber BINDER MKT 240, and we changed the temperature inside the chamber from -20 to $+65\,^{\circ}\text{C}$. Once the temperature stabilized at the given measurement step, we acquired 10,000 counter values from the device and computed the model parameters. Their evolution depending on temperature is summarized in Fig. 11.

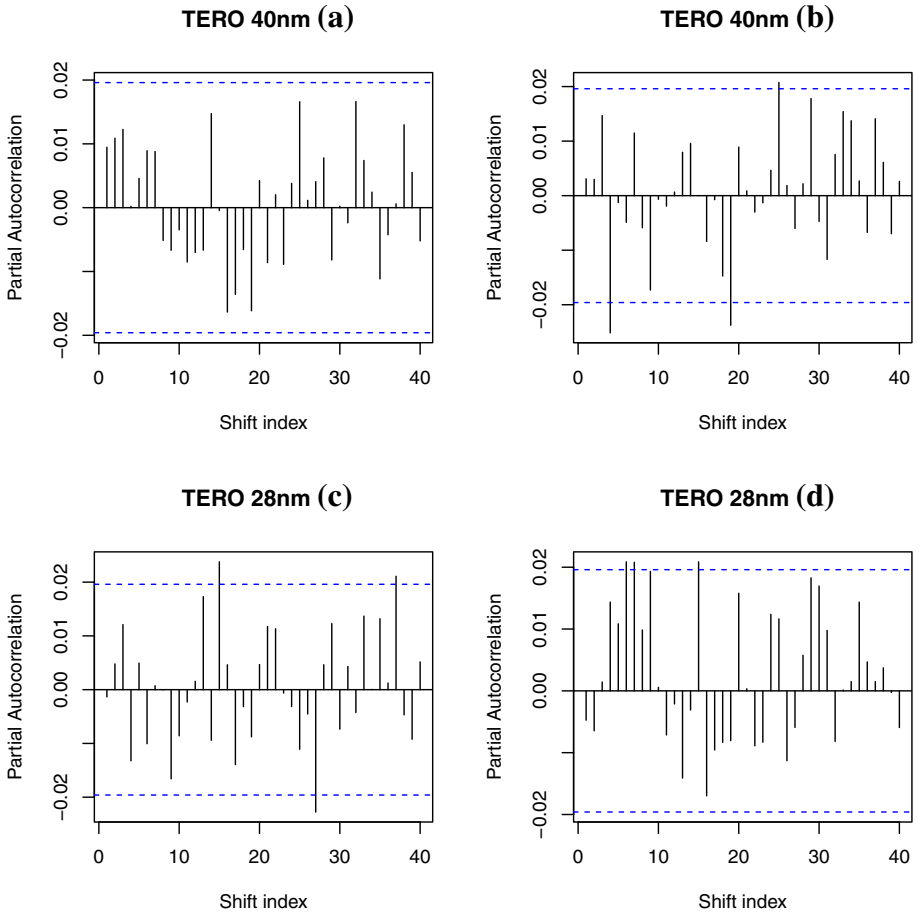

Fig. 10. Autocorrelation of the TERO-based TRNG output for four studied configurations (**a**–**d**).

Despite a relative stability of the model parameters and the output entropy rate around the nominal temperature (25 °C), we could observe that the results and in particular relative delays and transition timings (both rising and falling edges) that are represented by Δ_r and R, respectively, changed slightly with the temperature.

Following the presented conservative approach of entropy estimation, we took the minimum value of the entropy rate per output bit as a low entropy bound for the given implementation. Note that because the entropy rate depends not only on σ_r but also on Δ_r and R, this minimum entropy rate value does not necessarily correspond to the minimum value of σ_r.

We made similar experiments at various power supply voltages (from 1000 to 1200 mV by step of 10 mV) and acquired 10,000 counter values in each step to compute the model parameters. Their evolution depending on supply voltage is summarized in Fig. 12.

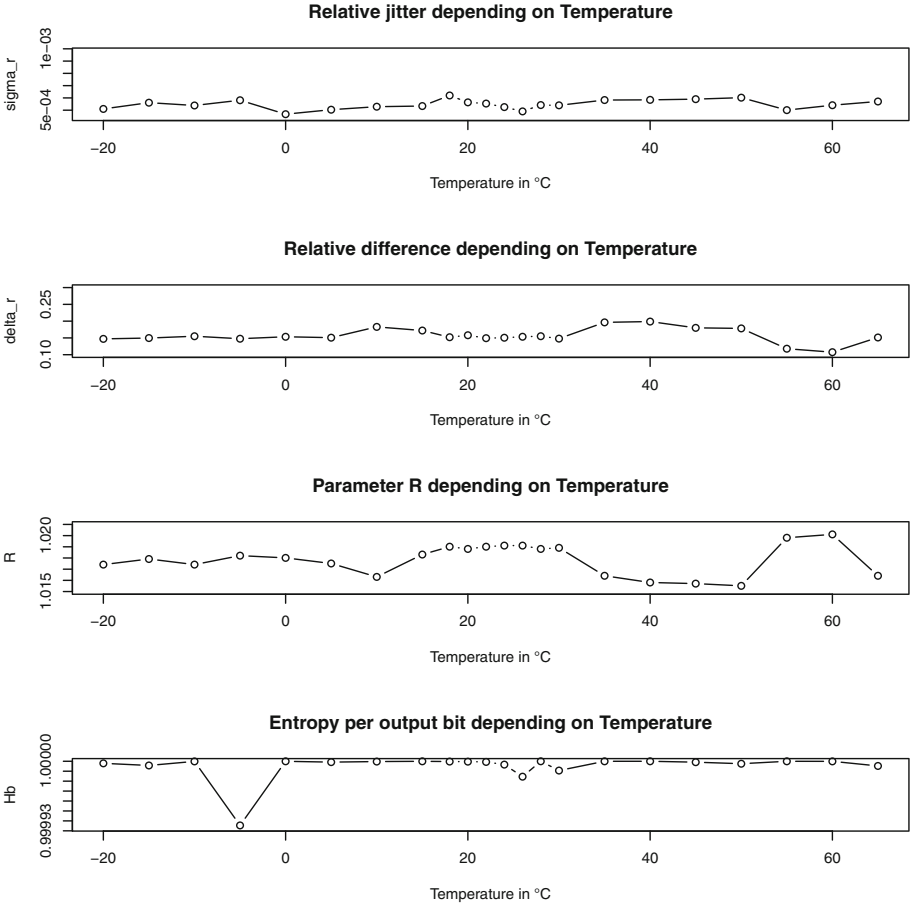

Fig. 11. Impact of the temperature on the model input parameters and output entropy rate per bit .

As can be seen, the supply voltage variation impacts the TERO structure and thus the model parameters more than the temperature variation. The parameter R is not stable around the nominal voltage any more, and it decreases regularly with the increasing voltage. This effect can be explained by the fact that the supply voltage modifies both falling and rising edge times that are modeled globally by the parameter R. Similarly as for temperature variations, we compute the entropy rate per output bit achievable in the worst case.

We could observe in this section that the model parameters are sensitive to environmental changes. These changes should be detected by some dedicated tests that should be embedded in the same device in order to signal significant deviations of security critical parameters caused by deterioration of operating conditions or some attacks.

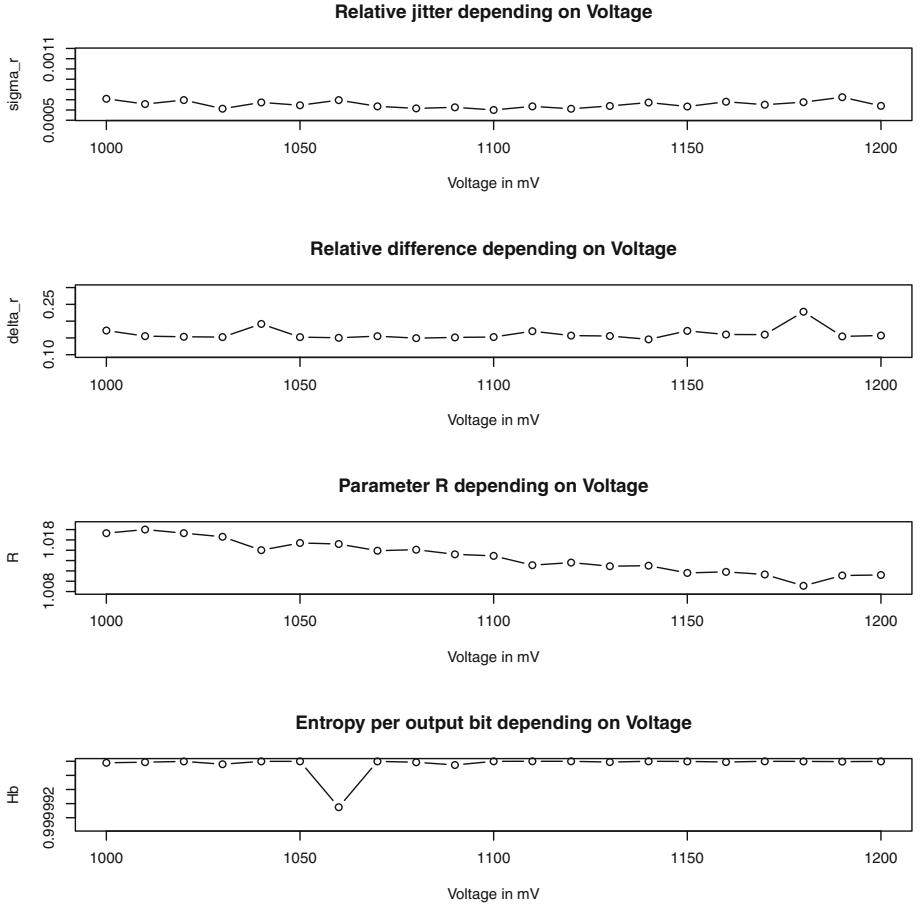

Fig. 12. Impact of the supply voltage on the model parameters and output entropy rate per bit .

7. Discussion

As we have seen above, the distribution of counter values for a given instance of the TERO-based TRNG is very well characterized by the model parameters R, σ_r, and Δ_r, and the entropy of the generated sequence depends on this distribution. Using the model, we can observe the impact of the TERO design on the distribution of random numbers and hence on entropy.

First, entropy is determined by relative jitter, i.e., by parameter σ_r. Since designers cannot directly alter the sources of thermal noise, they can only change the relative jitter by reducing the delay of the two TERO branches. This corresponds to increasing the frequency of oscillations.

Another important model parameter that determines entropy rate is the relative difference between the two TERO branches, i.e., parameter Δ_r. With smaller relative differences, TERO accumulates more jitter because it oscillates longer. As we saw in our

TERO TRNG implementations, the entropy rate per generated output byte was over 4.8, 6.7, 4.3, and 6.4, respectively. This means that if the designer only used one bit per generated byte (the counter output), they would be discarding a high percentage of usable random data. Of course, some post-processing could be used to profit from as much entropy as possible, but it would require additional silicon area, especially if a sophisticated algorithm was used (which would probably be the case in order to maintain a maximum entropy rate).

Another much more practical solution would be to unbalance the two TERO branches to the extent that the entropy rate per generated byte is sufficiently higher than 1 and then to use only one bit per generated number. Because of the difference in delays in the two branches, the TERO would oscillate for a shorter time and the output bit rate would consequently be higher. Since the entropy rate per generated number would be higher than one, each generated bit (the least significant bit of the counter) would have enough entropy and post-processing would not be necessary.

8. Conclusion

In this paper, we analyzed the processes that transform the noisy currents in the TERO circuitry into a random bit stream of the TERO-based TRNG. First, we conducted a detailed analysis of electric processes inside the TERO structure, and based on this analysis, we proposed the physical model of the TERO. We checked the model in four TERO configurations implemented in an ST 40 nm and ST 28 nm ASIC technology.

Next, based on this model, we proposed a stochastic model of a complete TERO-based TRNG. We showed that the proposed stochastic model can be successfully used to estimate the entropy rate. The entropy estimations are in perfect agreement with the results of the AIS 31 test suites.

We also showed that the proposed TRNG stochastic model can not only be used to estimate the entropy rate at the output of the generator, but also for entropy management, by setting a sufficient entropy rate while maintaining the maximum output bit rate.

Acknowledgements

This work received funds from the European ENIAC Joint Undertaking (JU) in the framework of the project TOISE (Trusted Computing for European Embedded Systems) and from the European Union's Horizon 2020 research and innovation programme in the framework of the project HECTOR (Hardware Enabled Crypto and Randomness) under Grant Agreement No. 644052. The authors wish to thank Nicolas Bruneau, Michel Agoyan, and Yannick Teglia for their help and numerous discussions.

Appendix

A. Proofs

In this section, we give proofs of Lemma 1, Propositions 1 and 2.

Proof of Lemma 1. In a neighborhood of $\overline{t_a}$, $f(t)$ can be approximated by its tangent line at time $\overline{t_a}$, giving the relation $T_a - \overline{t_a} = \frac{n(\overline{t_a})}{f'(\overline{t_a})}$. Since $n(\overline{t_a}) \sim \mathcal{N}(0, \sigma^2)$, $T_a \sim \mathcal{N}\left(\overline{t_a}, \frac{\sigma^2}{f'(\overline{t_a})^2}\right)$. The same holds for T_b in a neighborhood of $\overline{t_b}$, because $n(t)$ is stationary. By definition, $W_{\text{out}} = T_b - T_a$. If T_a and T_b are independent, W_{out} follows a normal distribution with mean $\mu_{\text{out}} = \overline{t_b} - \overline{t_a} = \frac{t_c}{2} + \left[w_{\text{in}} - \frac{t_c}{2}\right][1 + H_d]$ from Sect. 4.1 and variance $\sigma^2_{\text{out}} = \sigma^2_{T_b} + \sigma^2_{T_a} = \sigma^2\left(\frac{1}{f'(\overline{t_a})^2} + \frac{1}{f'(\overline{t_b})^2}\right)$. □

Proof of Proposition 1. (by recurrence on N)
Lemma 1 gives expression of μ_{out_N} and $\sigma^2_{\text{out}_N}$ for $N = 1$. Let $\{I_j\}_{j=1\ldots N+1}$ be a set of inverters, and let V_N be the signal between the two last inverters. Logically, the output of inverter I_N becomes the input of inverter I_{N+1}. Let V_{in} be the input signal of the first inverter I_1 and V_{out} is the output signal of the last inverter I_{N+1} in the chain. w_{in} is the width of a pulse at I_1. Let W_N be the width of the corresponding pulse appearing at signal V_N and W_{N+1} be the width of the pulse at V_{N+1}. By assumption of recurrence,

$$W_N \sim \mathcal{N}(\mu_{\text{out}_N}, \sigma^2_{\text{out}_N}) \text{ with } \begin{cases} \mu_{\text{out}_N} = \frac{t_c}{2} + \left(w_{\text{in}} - \frac{t_c}{2}\right)(1 + H_d)^N \\ \sigma^2_{\text{out}_N} = \sigma^2_{\text{out}}\left(\frac{(1+H_d)^{2N}-1}{(1+H_d)^2-1}\right) \end{cases}$$

According to Lemma 1, $W_{N+1} \sim \mathcal{N}(\mu_{\text{out}}, \sigma^2_{\text{out}})$ with $\mu_{\text{out}} = \frac{t_c}{2} + \left(w_n - \frac{t_c}{2}\right)(1 + H_d)$ where w_n is a realization of W_N. Assuming independence of noise sources in the chain, we have $\mu_{\text{out}_{N+1}} = \frac{t_c}{2} + \left(\mu_{\text{out}_N} - \frac{t_c}{2}\right)(1 + H_d)$ and $\sigma^2_{\text{out}_{N+1}} = \sigma^2_{\text{out}_N}(1 + H_d)^2 + \sigma^2_{\text{out}}$ giving

$$\begin{aligned} \mu_{\text{out}_{N+1}} &= \frac{t_c}{2} + \left(\frac{t_c}{2} + \left(w_{\text{in}} - \frac{t_c}{2}\right)(1 + H_d)^N - \frac{t_c}{2}\right)(1 + H_d) \\ &= \frac{t_c}{2} + \left(w_{\text{in}} - \frac{t_c}{2}\right)(1 + H_d)^{N+1} \end{aligned}$$

and $\sigma^2_{\text{out}_{N+1}} = \sigma^2_{\text{out}}\left(\frac{(1+H_d)^{2N}-1}{(1+H_d)^2-1}\right)(1+H_d)^2 + \sigma^2_{\text{out}} = \sigma^2_{\text{out}}\left(\frac{(1+H_d)^{2N+2}-(1+H_d)^2}{(1+H_d)^2-1} + 1\right) = \sigma^2_{\text{out}}\left(\frac{(1+H_d)^{2N+2}-1}{(1+H_d)^2-1}\right)$.
The statement in Proposition 1 is true for $N + 1$. By recurrence over N, Proposition 1 is true for any N. □

Proof of Proposition 2. Here we provide the proof for WK_S. (The same is valid for WL_S by replacing τ_1 with τ_2.)

Assuming that there is a pulse wk_{S-1} at V_K, the corresponding pulse WK_S at V_K after crossing the branches L and K (equivalent to a single chain of $2M+2M'+2$ inverters) is given as follows (according to Proposition 1 with $N = 2M + 2M' + 2$):

$$\mathrm{WK}_S \sim \mathcal{N}\left(\frac{t_c}{2} + \left(wk_{S-1} - \frac{t_c}{2}\right) R,\ \underbrace{\sigma^2_{\mathrm{out}}\left(\frac{R^2-1}{(1+H_d)^2-1}\right)}_{\sigma^2_{\mathrm{out}_{2M+2M'+2}}}\right),$$

where $R = (1+H_d)^{2M+2M'+2}$ and $t_c = \tau_1 + \tau_2$.
Thus, assuming independence of the noise sources in chains K and L, we have two relations of recurrence on $\mu_{K_S} = \frac{\tau_1+\tau_2}{2} + \left(\mu_{K_{S-1}} - \frac{\tau_1+\tau_2}{2}\right) R$ and on $\sigma^2_{K_S} = \sigma^2_{\mathrm{out}_{2M+2M'+2}} + \sigma^2_{K_{S-1}} R^2$.
It is easy to show that $\forall S \geq 1$,

$$\begin{aligned}
\mu_{K_S} &= \tfrac{\tau_1+\tau_2}{2} + \left(\mu_{K_0} - \tfrac{\tau_1+\tau_2}{2}\right) R^S = \tfrac{\tau_1+\tau_2}{2} + \tfrac{\tau_1-\tau_1}{2} R^S, \\
\sigma^2_{K_S} &= R^{2S}\sigma^2_{K_0} + \sigma^2_{\mathrm{out}_{2M+2M'+2}} \textstyle\sum_{i=0}^{S-1} (R^2)^i = R^{2S}\sigma^2_{\mathrm{out}_{2M+1}} + \sigma^2_{\mathrm{out}_{2M+2M'+2}} \tfrac{R^{2S}-1}{R^2-1}.
\end{aligned}$$

According to Proposition 1,

$$\begin{aligned}
\sigma^2_{\mathrm{out}_{2M+1}} &= \sigma^2_{\mathrm{out}} \frac{((1+H_d)^{2M+1})^2 - 1}{(1+H_d)^2 - 1} = \sigma^2_{\mathrm{out}} \frac{R_M^2 - 1}{(1+H_d)^2 - 1} \text{ and } \sigma_{\mathrm{out}_{2M+2M'+2}} \\
&= \sigma^2_{\mathrm{out}} \frac{((1+H_d)^{2M+2M'+2})^2 - 1}{(1+H_d)^2 - 1} = \sigma^2_{\mathrm{out}} \frac{R^2 - 1}{(1+H_d)^2 - 1},
\end{aligned}$$

therefore $\sigma^2_{K_S} = \frac{\sigma^2_{\mathrm{out}}}{(1+H_d)^2-1}\left(R^{2S}(R_M^2 - 1) + (R^2 - 1)\frac{R^{2S}-1}{R^2-1}\right) = \sigma^2_{\mathrm{out}} \frac{R^{2S} R_M^2}{(1+H_d)^2-1}$. □

References

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

[2] F. Bernard, V. Fischer, B. Valtchanov, Mathematical model of physical RNGs based on coherent sampling. *Tatra Mt. Math. Publ.* 45(1), 1–14 (2010)

[3] V. Fischer, A closer look at security in random number generators design, in *Constructive Side-Channel Analysis and Secure Design—COSADE 2012* (Springer, 2012), pp. 167–182

[4] P. Haddad, Y. Teglia, F. Bernard, V. Fischer, On the assumption of mutual independence of jitter realizations in P-TRNG stochastic models, in *Proceedings of Design, Automation and Test in Europe DATE 2014* (Dresden, Germany, March 2014), pp. 1–6

[5] L. Hars, Random number generation based on oscillatory metastability in ring circuits. https://eprint.iacr.org/2011/637.pdf (2011)

[6] W. Killmann, W. Schindler, A design for a physical RNG with robust entropy estimators, in Elisabeth Oswald and Pankaj Rohatgi, editors, *Cryptographic Hardware and Embedded Systems—CHES 2008, volume 5154 of LNCS* (Springer, 2008), pp. 146–163

[7] W. Killmann, W. Schindler, A proposal for: functionality classes for random number generators. https://www.bsi.bund.de (2011)
[8] G. Marsaglia, DIEHARD: Battery of Tests of Randomness. http://stat.fsu.edu/pub/diehard/ (1996)
[9] L.M. Reyneri, D. Del Corso, B. Sacco, Oscillatory metastability in homogeneous and inhomogeneous flip-flops. *IEEE J. Solid-State Circuits* **25**(1), 254–264 (1990)
[10] A. Rukhin, J. Soto, J. Nechvatal, M. Smid, E. Barker, S. Leigh, M. Levenson, M. Vangel, D. Banks, A. Heckert, J. Dray, S. Vo, A Statistical Test Suite for Random and Pseudorandom Number Generators for Cryptographic Applications—NIST SP 800-22, rev. 1a (2010)
[11] C. Shannon, A mathematical theory of communication. *Bell Syst. Tech. J.* **27**, 379–423, 623–656 July, (1948)
[12] B. Sunar, W.J. Martin, D.R. Stinson, A provably secure true random number generator with built-in tolerance to active attacks. *IEEE Trans. Comput.* 109–119 (2007)
[13] M. Varchola, M. Drutarovsky, New high entropy element for FPGA based true random number generators, in *Cryptographic Hardware and Embedded Systems (CHES), 2010* (Springer, 2010), pp. 351–365