IACR Transactions on Cryptographic Hardware and Embedded Systems
ISSN 2569-2925, Vol. 2024, No. 1, pp. 207–228. DOI:10.46586/tches.v2024.i1.207-228

Low Cost and Precise Jitter Measurement Method for TRNG Entropy Assessment

Florent Bernard[1], Arturo Garay[1,2], Patrick Haddad[2], Nathalie Bochard[1] and Viktor Fischer[1,3]

[1] Hubert Curien Laboratory, Université Jean Monnet, Member of Université de Lyon, 42000, Saint-Etienne, France
{florent.bernard,arturo.garay,nathalie.bochard,fischer}@univ-st-etienne.fr
[2] STMicroelectronics, Advanced System Techonology, 13790 Rousset, France
{arturo.mollinedogaray,patrick.haddad}@st.com
[3] Faculty of Information Technologies, Czech Technical University in Prague, 160 41, Prague, Czech republic

Abstract. Random number generators and specifically true random number generators (TRNGs) are essential in cryptography. TRNGs implemented in logic devices usually exploit the time instability of clock signals generated in freely running oscillators as source of randomness. To assess the performance and quality of oscillator-based TRNGs, accurate measurement of clock jitter originating from thermal noise is of paramount importance. We propose a novel jitter measurement method, in which the required jitter accumulation time can be reduced to around 100 reference clock periods. Reduction of the jitter accumulation time reduces the impact of the flicker noise on the measured jitter and increases the precision of the estimated contribution of thermal noise. In addition, the method can be easily embedded in logic devices. The fact that the jitter measurement can be placed in the same device as the TRNG is important since it can be used as a basis for efficient embedded statistical tests. In contrast to other methods, we propose a thorough theoretical analysis of the measurement error. This makes it possible to tune the parameters of the method to guarantee a relative error smaller than 12% even in the worst cases.

Keywords: Random number generation · TRNGs · Jitter characterization · Embedded jitter measurement

1 Introduction

Random number generators (RNGs) are essential in cryptography – they provide the random numbers used as encryption keys, nonces, and padding values in cryptographic protocols or as random masks in countermeasures against side channel attacks. RNGs can be implemented as pseudo-random number generators (PRNGs), that are easy to implement in logic systems, reach higher output bitrates and produce random numbers with excellent statistical properties. However, if the algorithm and internal state of the PRNG is known, it is potentially predictable and hence vulnerable. Physical true random number generators (TRNGs), which rely on some physical random phenomena, are by nature unpredictable, but they usually offer a lower output bitrate and generate numbers of lower statistical quality. The main difficulty with the TRNG design is finding a robust and quantifiable physical source or sources of randomness and an efficient method of converting random physical quantities into a stream of generated numbers (for example, a bitstream).

Traditionally, the quality of generated numbers has been evaluated using a suite of general-purpose statistical tests. However, black box statistical tests can not detect

Received: 2023-07-15 Accepted: 2023-09-15 Published: 2023-12-04

dependencies and pseudo-randomness, which weaken the robustness of the generator. Therefore, according to modern security standards [KS11, ISO19], the quality of generated numbers and their unpredictability should be assessed using an output entropy rate that is estimated by a stochastic model. The aim of the stochastic model is to describe the distribution of output random numbers depending on the entropy extraction method used (usually some kind of analog-to-digital conversion) and on the set and size of physical quantities, that are used as the input parameters of the model.

One of the most frequently used source of randomness in logic devices is time instability (i.e. the jitter) of the clock signals generated in freely running oscillators, e.g. ring oscillators (RO). This kind of jitter depends on non-manipulable random physical sources like thermal noise, but also on global sources, that can be manipulated by an attacker. To eliminate the possibility of jitter manipulations, a differential principle based on a couple of ROs is used in the so-called elementary RO-based TRNG (ERO-TRNG) [BLMT11]. According to the proposed stochastic model of the ERO-TRNG, the entropy (and hence the unpredictability) of the generated numbers depends on three model input parameters [BLMT11, Appendix C]: 1) the *duty cycle* of the sampled clock signal; 2) the *drift* of the Wiener process; and 3) the *volatility* of the Wiener process, which is linked to the variance of the resulting clock signal. While the duty cycle and the drift (mean frequency) are easy to measure inside or outside the device, the volatility of the Wiener process should only reflect the contribution of thermal noise to jitter and thus to entropy, since the sources of thermal noise are mutually independent, uncorrelated in time and not manipulable. Consequently, accurate measurement of the jitter component originating from the thermal noise is indispensable to assess the performance and the quality of oscillator-based TRNGs and of the ERO-TRNG. The main danger is overestimating jitter that can lead to overestimating entropy and thus to reducing security.

Several jitter measurement methods have been published. In [SMS07], Sunar *et al.* measure the jitter outside the device using standard probes and an oscilloscope. In [VFAB10], the authors use a differential data interface and differential oscilloscope probes to increase measurement precision. Their results show that Sunar *et al.* overestimated the jitter more than five fold, which, in practice, would lead to catastrophic overestimation of entropy. However, a differential data interface is not always available. Moreover, the use of external measurement equipment would exclude online measurement of jitter to detect attacks and could add error caused by the acquisition chain.

Clearly, embedded jitter measurement methods are preferable. In these methods, like in random number generation itself, it is widely accepted that jitter is measured using a differential method exploiting two oscillators, which helps reduce the impact of global perturbations. In [HTBF14], the authors use a counter method to quantify jitter. However, to be able to measure jitter, long jitter accumulation times are necessary (thousands of the reference clock periods in [HTBF14]). But the long accumulation times mean the contribution of flicker noise to jitter will dwarf that of thermal noise. It is important to note that flicker noise is known to be auto-correlated [HLL99] and can lead to non-stationarity of random variables. Therefore, reducing the measurement time to reduce the impact of the flicker noise on the measured jitter is of the utmost importance.

In [FL14], the authors propose a jitter measurement method that requires jitter accumulation times of about 300 reference clock periods. The method presented in [YRG^{+}17] reduces the jitter accumulation time required but at the cost of huge hardware resources. In [GBF^{+}23], the authors point out that the precision of a measurement method should be thoroughly evaluated and demonstrated before the method is used.

In this paper, we propose a novel jitter measurement method, that can be embedded in logic devices and does not require any external measurement or any prior characterization to work properly. The proposed method requires only short (shorter than in [FL14]) jitter accumulation times meaning the contribution of flicker noise to jitter is smaller or even

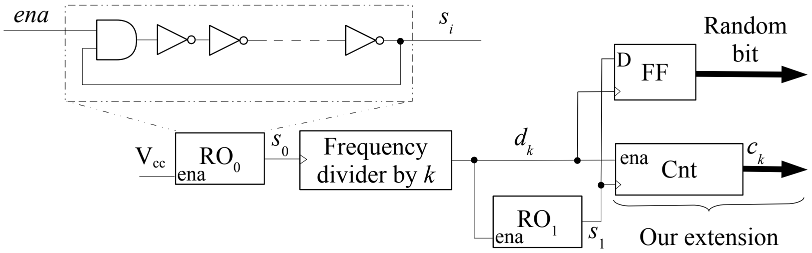

Figure 1: Circuitry of the ERO-TRNG with an additional counter aimed at the jitter measurement.

negligible, and uses far fewer hardware resources than in [YRG+17].

Our methodology is different from other published methods: we analyse and explain in detail how different parameters determine the accuracy of the method. We also offer precise formulas to compute the upper bound of the measurement error to avoid overestimating jitter in the measurement process. The resulting measurements can consequently be used to confidently estimate the entropy rate.

The paper is organized as follows: in Section 2, we present the theoretical background and definitions. In Section 3, we explain the principle of the novel method. In Subsection 3.2, we present a way to calculate the upper bound of the measurement error and in Subsection 3.3, we verify our analysis of measurement errors using simulations and in Subsection 3.4, we illustrate how the method works. In Section 4, we describe hardware implementation of the method and its implementation constraints. We then verify the validity of assumptions made in Section 3 and present results of the jitter measurements performed in three different FPGA families. In Section 5, we compare our method with other published jitter measurement methods. The objectivity of the comparison is guaranteed by performing the jitter measurements on the same couple of ROs and using the same environmental conditions in all the methods evaluated. Finally, in Section 6, we conclude the article and present our plans for the future.

2 Theoretical background and definitions

Our objective is to evaluate the source of randomness in an ERO-TRNG (Elementary Ring Oscillator-based TRNG), that is, the clock jitter accumulated in a couple of ring oscillators. The jitter measurement circuitry should be based on the ERO-TRNG architecture with only minor modifications. The proposed architecture is shown in Fig. 1. The ring oscillator (RO) is composed of a chain of an odd number of inverters connected in a ring using an AND gate as shown in the upper part of Fig. 1. The AND gate added to the ring can be used to restart the oscillations. The output signal s_i of the ring oscillator RO$_i$ is a square wave whose average frequency depends on the sum of delays added by the individual inverters. The frequency divider determines the jitter accumulation time by dividing the reference clock frequency by k. The sampler (D flip-flop) is the standard part of the ERO-TRNG that generates random bits at its output. The counter Cnt is the only additional component needed to measure the jitter. The counter output c_k gives the number of rising edges of s_1 that appear during the jitter accumulation time d_k.

Let us consider the timings in the jitter measurement circuitry as presented in Fig. 2. Time periods T_0 and T_1 represent the average periods of signals s_0 and s_1, respectively. Similarly, $E_0(n)$ and $E_1(n)$ are the arrival times of the $(n+1)$-th rising edge of these signals, while their initial edges arrive at time $E_0(0)$ and $E_1(0)$, respectively. Finally, we denote φ_0 the initial phase shift between the generated clocks, i.e. $\varphi_0 = E_1(0) - E_0(0)$.

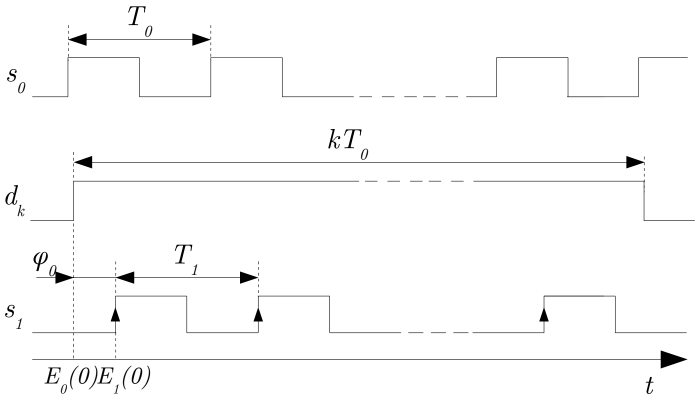

Figure 2: Signal timings in the jitter measurement circuitry.

In a real oscillator, the arrival time of clock edges fluctuates. It will be recalled, that thanks to the differential measurement principle [VABF08], which is based on the use of two identical oscillators, the contribution of the global and deterministic noises to the clock jitter is negligible. Therefore, we can consider that the fluctuation of the edges is mainly due to local noises such as thermal noise and flicker noise. What is more, with short accumulation times, the influence of flicker noise is reduced and thermal noise dominates [HTBF14]. In this case, Baudet *et al.* [BLMT11] suggest to model the evolution of the phase of the oscillators by a Wiener process of drift nT_i and volatility $\sqrt{n}\sigma_i$ where σ_i is the period jitter caused by thermal noise of RO_i.

If we consider the output signal of a couple of ring oscillators as shown in Fig. 2, we can further simplify our analysis. Like [BLMT11], we can assume that RO_0 is an ideal oscillator and that the measured jitter, which includes the contribution of both RO_0 and RO_1, is only present in the output of RO_1, i.e.:

$$\begin{aligned} E_0(n) &= E_0(0) + nT_0, \\ E_1(n) &= E_1(0) + nT_1 + q_n. \end{aligned} \tag{1}$$

where q_n follows a normal distribution $\mathcal{N}(0, \sigma_n^2)$ with $\sigma_n^2 = na_{th}^2$. Where a_{th} is the equivalent jitter representing the thermal noise contribution of both RO_0 and RO_1, i.e.

$$a_{th} := \sqrt{\sigma_1^2 + \frac{T_1}{T_0}\sigma_0^2}\,. \tag{2}$$

3 The new jitter measurement method

3.1 Principle

We recall that our objective is to propose a precise measurement method of the jitter coming from thermal noise, that can be embedded in logic devices as a basis for future embedded online tests.

As described in the previous section, we use the circuit shown in Fig. 1, which includes two ring oscillators that can be restarted by means of the *ena* signal. Ring oscillator RO_0 feeds a frequency divider. Its output, denoted d_k, determines the jitter accumulation time kT_0. Since RO_0 is considered as an ideal oscillator, according to Eq. (1) the accumulation interval stops at $t = E_0(k)$. The output of RO_1 is used as the clock signal of a counter. Since RO_1 produces jittered rising edges, for a given frequency division factor k, the

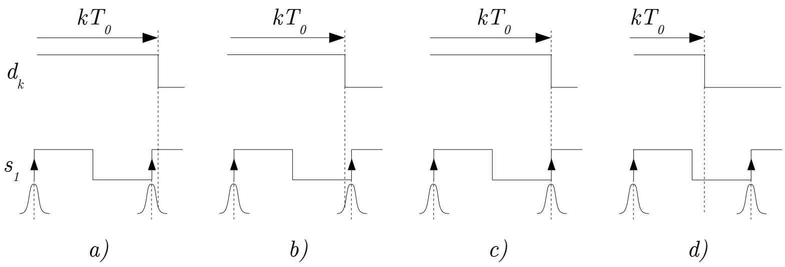

Figure 3: Illustration of the position of the last edge of the clock signal s_1 at the end of the measurement interval: before (a), after (b), at (c), and far from (d).

counter produces random values denoted c_k corresponding to the number of rising edges of RO_1 during kT_0. The C_k set is composed of N random values c_k: $C_k = \{c_k\}_{i=1,\ldots,N}$.

Depending on the initial phase φ_0, the jitter accumulation factor k and clock periods T_0 and T_1, four cases can occur at the end of the counting interval, as shown in Fig. 3:

1. The measurement interval ends up in the vicinity of the rising edge area of signal s_1, slightly after the mean position of the edge.

2. The measurement interval ends up close to the rising edge area of signal s_1, slightly before the mean position of the edge.

3. The measurement interval closes at exactly the mean position of the rising edge of s_1.

4. The measurement interval ends up far from the rising edges of signal s_1.

In Case d), the counter values c_k are constant so the jitter could not be measured. In Case c), when the measurement interval stops at exactly the mean position of the rising edge of signal s_1, two counter values appear at its output, each with the same probability. In Cases a) and b), two counter values appear just like in Case c), but one of the two values would have higher probability than the other. In Case a), a higher counter value would be more probable than a lower one (the next rising edge would be counted more often). In Case b), the probability of the two counter values would be inverted (the next rising edge would be missed more often).

To confirm our analysis, we implemented the circuit in Fig. 1 in hardware. In our experiments, we observed behavior of counter values c_k for different values of k. Namely, we varied k from 1 to 255 and acquired a set C_k in $N = 4\,096$ measurements for each k. Choosing a k value above 300 was not useful (and probably dangerous due to the influence of flicker noise), because for these values of k, the method presented in [FL14] was shown to produce satisfactory results. Figure 4 depicts the variance $Var(c_k)$ resulting from these sets of measurements. Note that the variance of a random variable X that only has two outcomes with probability p_0 and p_1 respectively, is given by $Var(X) = p_0 p_1 = p_0(1 - p_0)$. Its maximum value is 0.25 for $p_0 = p_1 = 0.5$.

The experiments confirmed our expectations. Most often, and especially with short accumulation times, the counter values were constant. This corresponds to Case d) shown in Fig. 3, in which the variance of N counter values $Var\,(c_k) = 0$ and no information regarding the jitter can be obtained from the counter output.

Nonzero variances of counter values in Fig. 4 correspond to Cases a) to c) in Fig. 3. However, in Case c), the probabilities of the two possible counter values are identical and consequently, for any value of the jitter, the variance $Var\,(c_k)$ is equal to 0.25. This case does not provide any information about the jitter and cannot be used for jitter measurement. Fortunately, this balanced case is very rare and easy to detect.

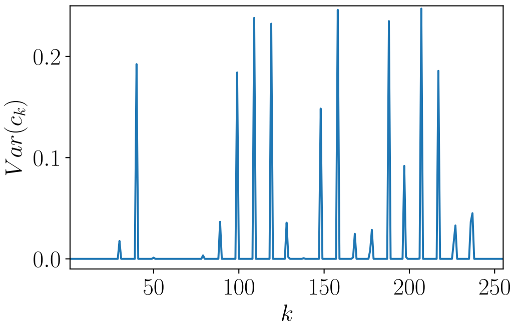

Figure 4: Variance of counter values c_k as a function of k generated by the circuit shown in Fig. 1 implemented in hardware.

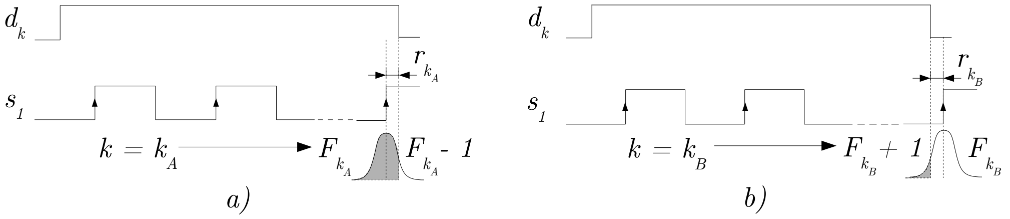

Figure 5: Illustration of the case where the measurement interval ends in the vicinity of the edge of the clock signal s_1, whereas it most probably ends: a) after the edge of the clock signal s_1; b) before the edge of the clock signal s_1. r_{k_A} (resp. r_{k_B}) denotes the distance between the last rising edge of s_1 and the end of the measurement window in Case a) (resp. in Case b)).

Consequently, we focus on Cases a) and b), where the two counter values do not have the same probability. Because of the jitter, the measurement interval most often ends after (Case a)) or before (Case b)) the last rising edge of s_1. This edge has a small but non-negligible probability of sometimes ending before (in Case a)) or after (in Case b)) the last rising edge of s_1. The main idea of the proposed method is to exploit the imbalance in the distribution of this last edge to retrieve information about the jitter. These cases occur when there are only two different and consecutive values in the set C_k and when their probability of appearance is imbalanced. In such cases the following conditions hold:

$$\begin{aligned} &\max(C_k) - \min(C_k) = 1, \\ &0 < Var(c_k) < 0.25. \end{aligned} \tag{3}$$

We denote the most likely outcome of the counter by

$$F_k := c_k \in C_k | \#\{c_k \in C_k\} > \frac{N}{2} \ . \tag{4}$$

We denote $M_{(k,N)}$ the number of samples in C_k equal to the maximum c_k in the set,

$$M_{(k,N)} := \# \{c_k \in C_k | c_k = \max(C_k)\} \ . \tag{5}$$

In the left panel of Fig. 5, the event corresponding to the end of the measurement interval kT_0 most probably occurs just after the rising edge of RO_1. Nevertheless, because of the jitter, the probability that this event occurs just before the edge of RO_1 is not negligible. It corresponds to Case a) and, in this case, we denote the value of k as $k = k_A$.

Recalling that F_{k_A} corresponds to the number of rising edges (so $F_{k_A} - 1$ is the number of complete periods T_1 counted in this measurement interval), from the left panel in Fig. 5, the following equation can be deduced:

$$\varphi_0 + (F_{k_A} - 1)T_1 + r_{k_A} = k_A T_0. \tag{6}$$

We then observe that:

$$c_k = F_{k_A} \Leftrightarrow q_{F_{k_A}} \leq r_{k_A}, \tag{7}$$

where $q_{F_{k_A}}$ is the clock jitter accumulated during F_{k_A} clock periods, as described in Eq. (1). Consequently,

$$Pr\,(c_k = F_{k_A}) = Pr\left(q_{F_{k_A}} \leq r_{k_A}\right). \tag{8}$$

But because $q_{F_{k_A}} \sim \mathcal{N}\left(0, \sigma^2_{F_{k_A}}\right)$,

$$Pr\,(c_k = F_{k_A}) = Pr\left(\frac{q_{F_{k_A}}}{\sigma_{F_{k_A}}} \leq \frac{r_{k_A}}{\sigma_{F_{k_A}}}\right) = \Phi\left(\frac{r_{k_A}}{\sigma_{F_{k_A}}}\right), \tag{9}$$

where $\Phi(\cdot)$ is the normal cumulative distribution function. The $Pr\,(c_k = F_{k_A})$ can be estimated using the proportion of a large number of counter values c_k equal to F_{k_A}. This proportion converges to $Pr\,(c_k = F_{k_A})$ while the number of observed counter values tends to infinity. It corresponds to the shadowed area in the left panel of Fig. 5 and is denoted $\mathcal{A}_{k_A}$. Note that in this case, $\mathcal{A}_{k_A} > 0.5$. Considering Eq. (5), we get:

$$Pr\,(c_k = F_{k_A}) = \lim_{N\to\infty} \frac{M_{(k_A,N)}}{N} = \mathcal{A}_{k_A} = Pr\left(q_{F_{k_A}} \leq r_{k_A}\right), \tag{10}$$

and then by combining Eq. (9) and (10), we obtain:

$$\frac{r_{k_A}}{\sigma_{F_{k_A}}} = \Phi^{-1}\left(\mathcal{A}_{k_A}\right), \tag{11}$$

where $\Phi^{-1}(\cdot)$ is the inverse normal cumulative distribution function. Finally, Eq. (11) can be applied in Eq. (6) to obtain:

$$\varphi_0 + (F_{k_A} - 1)T_1 + \Phi^{-1}\left(\mathcal{A}_{k_A}\right)\sigma_{F_{k_A}} = k_A T_0. \tag{12}$$

Let us now consider Case b) shown in the right panel of Fig. 5. We can proceed in the same way as in Case a). In Case b), the end of the measurement interval most probably occurs before the rising edge of RO$_1$. Again, because of the jitter, the probability that the end of the measurement interval arrives just after the rising edge of RO$_1$ is not negligible and corresponds to Case b). In this case, we denote $k = k_B$. From the right panel of Fig. 5, we can deduce the following equation:

$$\varphi_0 + F_{k_B} T_1 - r_{k_B} = k_B T_0. \tag{13}$$

Further, we observe that:

$$c_k = F_{k_B} + 1 \Leftrightarrow q_{F_{k_B}+1} \leq -r_{k_B}, \tag{14}$$

where $q_{F_{k_B+1}}$ is the clock jitter accumulated during $F_{k_B} + 1$ clock periods. Consequently,

$$Pr\,(c_k = F_{k_B} + 1) = Pr\left(q_{F_{k_B}+1} \leq -r_{k_B}\right). \tag{15}$$

But because $q_{F_{k_B}+1} \sim \mathcal{N}\left(0, \sigma^2_{F_{k_B}+1}\right)$,

$$Pr\left(c_k = F_{k_B} + 1\right) = Pr\left(\frac{q_{F_{k_B}+1}}{\sigma_{F_{k_B}+1}} \le -\frac{r_{k_B}}{\sigma_{F_{k_B}+1}}\right) = \Phi\left(-\frac{r_{k_B}}{\sigma_{F_{k_B}+1}}\right). \tag{16}$$

As before, the $Pr\left(c_k = F_{k_B} + 1\right)$ can be estimated using the proportion of a large number of counter values c_k equal to $F_{k_B} + 1$. It corresponds to the shadowed area in the right panel of Fig. 5 and is denoted $\mathcal{A}_{k_B}$. Note that in this case, $\mathcal{A}_{k_B} < 0.5$. By taking into account Eq. (5), we can write:

$$Pr\left(c_k = F_{k_B} + 1\right) = \lim_{N\to\infty} \frac{M_{(k_B,N)}}{N} = \mathcal{A}_{k_B} = Pr\left(q_{F_{k_B}+1} \le -r_{k_B}\right). \tag{17}$$

By combining Eq. (16) and (17) we obtain

$$-\frac{r_{k_B}}{\sigma_{F_{k_B}+1}} = \Phi^{-1}\left(\mathcal{A}_{k_B}\right). \tag{18}$$

Equation (18) can be applied in Eq. (13) to obtain:

$$\varphi_0 + F_{k_B} T_1 + \Phi^{-1}\left(\mathcal{A}_{k_B}\right) \sigma_{F_{k_B}+1} = k_B T_0. \tag{19}$$

The initial phase shift (φ_0) is an unknown variable. We can use Eq. (12) and (19) to form a system of equations where the only independent unknown is the equivalent thermal contribution to the jitter. Recalling that $\sigma_{F_k} = a_{th}\sqrt{F_k}$ (see Eq. (1)), we can write

$$\begin{cases} \varphi_0 + (F_{k_A} - 1)T_1 + \Phi^{-1}\left(\mathcal{A}_{k_A}\right) a_{th}\sqrt{F_{k_A}} & = k_A T_0, \\ \\ \varphi_0 + F_{k_B} T_1 + \Phi^{-1}\left(\mathcal{A}_{k_B}\right) a_{th}\sqrt{F_{k_B} + 1} & = k_B T_0. \end{cases} \tag{20}$$

As it will be explained in Subsection 4.2, we can assume that φ_0 and the ratio T_0/T_1 are constant throughout the measurement. Thus, by combining equations from Eq. (20), we obtain

$$\frac{a_{th}}{T_1} = \frac{\frac{T_0}{T_1}\left(k_A - k_B\right) - \left(F_{k_A} - F_{k_B} - 1\right)}{\Phi^{-1}\left(\mathcal{A}_{k_A}\right)\sqrt{F_{k_A}} - \Phi^{-1}\left(\mathcal{A}_{k_B}\right)\sqrt{F_{k_B} + 1}}. \tag{21}$$

The ratio T_0/T_1 cannot be measured directly. However, it can be approximated by the ratio c_L/L, where c_L is the counter value obtained using the circuit in Fig. 1 with a big frequency divider factor L. Another factor that determines the precision of the method is the number of measurements N. Indeed, $\mathcal{A}_k$ is approximated by $\frac{M_{(k,N)}}{N}$ when N is sufficiently big. The value a_{th}/T_1 is thus approximated by $\widetilde{a}_{th}/T_1$, which depends only on measurable parameters:

$$\frac{a_{th}}{T_1} \approx \frac{\widetilde{a}_{th}}{T_1} := \frac{c_L/L\left(k_A - k_B\right) - \left(F_{k_A} - F_{k_B} - 1\right)}{\Phi^{-1}\left(\frac{M_{(k_A,N)}}{N}\right)\sqrt{F_{k_A}} - \Phi^{-1}\left(\frac{M_{(k_B,N)}}{N}\right)\sqrt{F_{k_B} + 1}}. \tag{22}$$

3.2 Analysis of the measurement error

As pointed out in [GBF+23], the parameters that affect the precision of the method, need to be clearly identified and their impact on the precision thoroughly evaluated. In this sense, we identified the parameters, that determine the precision of the proposed measurement method. In contrast to the state-of-the-art methods, we evaluated theoretical bounds of the relative error and determined the values of parameters (L, N, k_A and k_B) that minimize the measurement error.

According to the previous analysis, the relative measurement error of ${}^{a_{th}}\!/_{T_1}$ is given by:

$$\left| \frac{\frac{a_{th}}{T_1} - \frac{\widetilde{a}_{th}}{T_1}}{\frac{a_{th}}{T_1}} \right| = \left| 1 - \frac{\widetilde{a}_{th}}{a_{th}} \right| . \tag{23}$$

By replacing ${}^{a_{th}}\!/_{T_1}$ from Eq. (21) and $\frac{\widetilde{a}_{th}}{T_1}$ from Eq. (22) in Eq. (23) we obtain a cumbersome expression that can be upper bounded according to the following inequality:

$$\left| 1 - \frac{\widetilde{a}_{th}}{a_{th}} \right| \leq \sqrt{\frac{\max(F_{k_A}, F_{k_B} + 1)}{\min(F_{k_A}, F_{k_B} + 1)}} \left(|\alpha_{0,1}| + |\alpha_{AB}| + |\alpha_{0,1} \cdot \alpha_{AB}| \right) . \tag{24}$$

This inequality represents the upper bound of error of our method. It is governed by two main factors: α_{AB}, which is related to the error made when approximating $\mathcal{A}_k$ to $\frac{M_{(k,N)}}{N}$, equal to

$$\alpha_{AB} := \frac{\Phi^{-1}\left(\mathcal{A}_{k_B}\right) - \Phi^{-1}\left(\frac{M_{(k_B,N)}}{N}\right) - \left(\Phi^{-1}\left(\mathcal{A}_{k_A}\right) - \Phi^{-1}\left(\frac{M_{(k_A,N)}}{N}\right)\right)}{\Phi^{-1}\left(\frac{M_{(k_A,N)}}{N}\right) - \Phi^{-1}\left(\frac{M_{(k_B,N)}}{N}\right)} , \tag{25}$$

and $\alpha_{0,1}$, which is related to the measurement error of the ratio $\frac{T_0}{T_1}$, equal to

$$\alpha_{0,1} := \frac{(k_A - k_B) \cdot \left(\frac{T_0}{T_1} - \frac{c_L}{L}\right)}{(k_A - k_B) \cdot \frac{T_0}{T_1} - (F_{k_A} - F_{k_B} - 1)} . \tag{26}$$

We analyze the possible origins of these two errors in the two following subsections.

3.2.1 Error due to area approximation

The error due to approximation of the area under the Gaussian curve in Fig. 5 is represented by α_{AB}. It is related to a general problem of area approximation using a Monte-Carlo method. Following this principle, we obtain an absolute upper bound of $|\alpha_{AB}|$ depending on N, but not on other parameters of the method. We recall that

$$\mathcal{A}_{k_A} > 0.5 \implies \Phi^{-1}(\mathcal{A}_{k_A}) > 0 \ ; \ \mathcal{A}_{k_B} < 0.5 \implies \Phi^{-1}(\mathcal{A}_{k_B}) < 0 . \tag{27}$$

However, we approximate $\mathcal{A}_{k_A}$ and $\mathcal{A}_{k_B}$ by $\frac{M_{(k_A,N)}}{N}$ and $\frac{M_{(k_B,N)}}{N}$, respectively. Because of these approximations, if $\mathcal{A}_{k_A}$ or $\mathcal{A}_{k_B}$ is close to 0.5, $\frac{M_{(k_A,N)}}{N}$ can be < 0.5 or $\frac{M_{(k_B,N)}}{N}$ can be > 0.5. Hence, the denominator of Eq. (25) can be close to 0, making α_{AB}, and consequently the error, diverge. Moreover, due to the divergence of the function $\Phi^{-1}(\cdot)$ in 0 and 1, a minor error in the approximation of $\mathcal{A}_k$ with $\frac{M_{(k,N)}}{N}$ will be strongly amplified by $\Phi^{-1}(\cdot)$ when $\frac{M_{(k_A,N)}}{N} \approx 1$ or $\frac{M_{(k_B,N)}}{N} \approx 0$. For this reason, to find an upper bound of $|\alpha_{AB}|$, we need to bound $\frac{M_{(k_A,N)}}{N}$ and $\frac{M_{(k_B,N)}}{N}$.

To obtain these bounds, we simulated the counting process for $\frac{r_{k_A}}{\sigma_{F_{k_A}}}$ (resp. $\frac{r_{k_B}}{\sigma_{F_{k_B}+1}}$) representing the normalized difference between the position of the last rising edge of s_1 in Case a) (resp. in Case b)) and the end of the measurement window. They both varied independently from $r_{min} = 0$ to $r_{max} = 5$ in steps of 0.01. We computed the theoretical values, $\Phi^{-1}(\mathcal{A}_{k_A})$ and $\Phi^{-1}(\mathcal{A}_{k_B})$, and the experimental ones, for N repetitions, $\Phi^{-1}\left(\frac{M_{(k_A,N)}}{N}\right)$ and $\Phi^{-1}\left(\frac{M_{(k_B,N)}}{N}\right)$, to compute α_{AB} for each case. The maximum evaluated value $r_{max} = 5$ was chosen because the probability that the last edge of s_1 falls outside this normalized interval is negligible, i.e. for $\frac{r_k}{\sigma_{F_k}} \geq 5$ we obtain $1 - \Phi(5) = 2.87 \times 10^{-7}$.

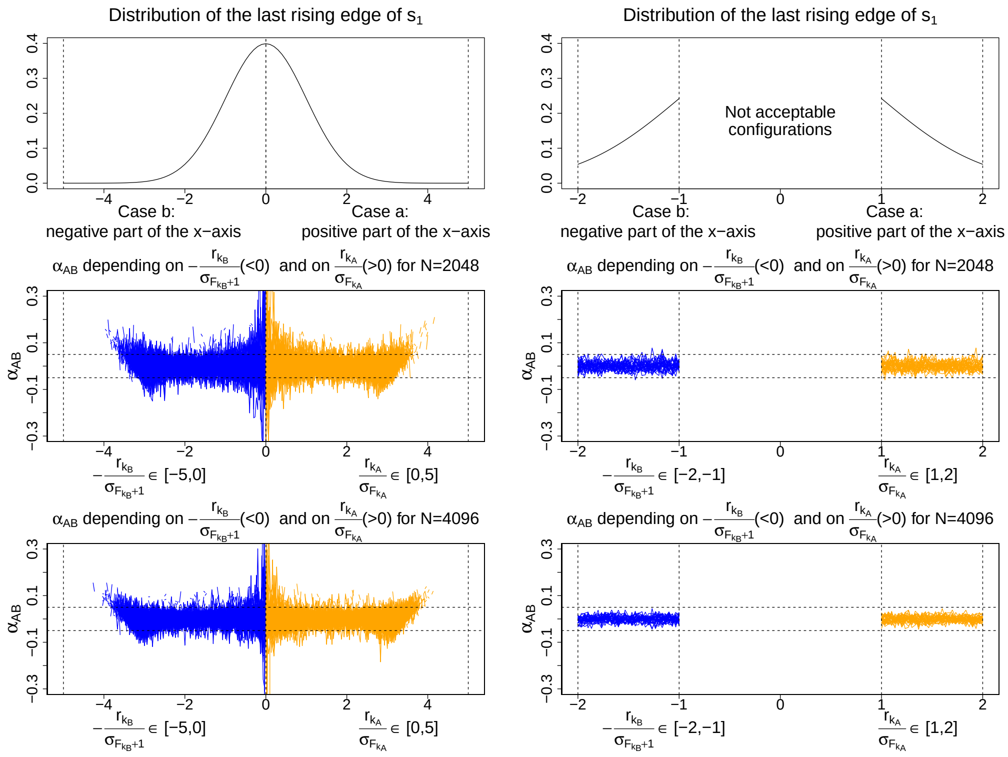

Figure 6: Variations of $|\alpha_{AB}|$ depending on the selection of suitable (right panels) or unsuitable (left panels) relative positions between the last rising edge of s_1 and the end of the measurement interval, for $N_{min} = 2\,048$ and $N_{min} = 4\,096$, respectively. In orange color for Case a) (resp. blue color for Case b)), for **a** given $^{r_{k_A}}/_{\sigma_{F_{k_A}}}$ (resp.$^{r_{k_B}}/_{\sigma_{F_{k_B}+1}}$) $\in [r_{min}; r_{max}]$, α_{AB} is computed for **all** values of $^{r_{k_B}}/_{\sigma_{F_{k_B}+1}}$ (resp.$^{r_{k_A}}/_{\sigma_{F_{k_A}}}$) $\in [r_{min}; r_{max}]$. So for each x-axis, all possible values α_{AB} are plotted.

The results are presented in Fig. 6. The top part of the figure represents the distribution of the last rising edge of s_1. If there is no restriction (left panel) on the relative position of the end of the measurement window with respect to the last rising edge ($0 \leq \frac{r_k}{\sigma_{F_k}} \leq 5$), then $|\alpha_{AB}|$ cannot be reasonably bounded (middle left panel) even for large values of N (bottom left panel).

However, in the right panel, by setting r_{min} relatively far from 0 (e.g. $r_{min} = 1$) and r_{max} relatively far from 5 (e.g. $r_{max} = 2$), we exclude some relative positions of the last rising edge of s_1 with respect to the end of the measurement window (upper right panel) hence $|\alpha_{AB}|$ does not diverge. Indeed, the middle right and bottom right panels of Fig. 6 show the influence of N on the amplitude of $|\alpha_{AB}|$. Because $\lim_{\to\infty} \frac{M_{k,N}}{N} = \mathcal{A}_k$, for any given threshold $\varepsilon > 0$, there exists an integer N_{min} such that for any $N \geq N_{min}$, $|\alpha_{AB}| \leq \varepsilon$. We chose $\varepsilon = 0.05$. As shown in Fig. 6 (right middle and bottom panel), we need $N_{min} = 4\,096$, to ensure $|\alpha_{AB}| \leq 0.05$. Note that in the right panel, both $^{r_{k_A}}/_{\sigma_{F_{k_A}}}$ and $^{r_{k_B}}/_{\sigma_{F_{k_B}+1}}$ are in $[r_{min}; r_{max}] = [1; 2]$.

Using Eq. (11) and (18), the following thresholds for $\frac{M_{(k_A,N)}}{N}$ and $\frac{M_{(k_B,N)}}{N}$ (that depend

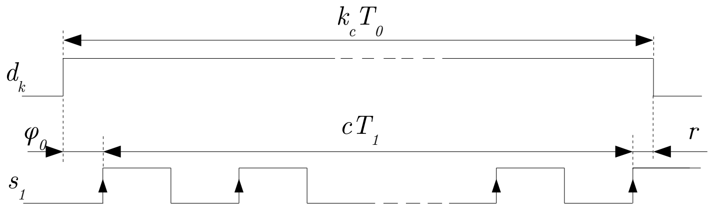

Figure 7: Waveform used to deduce a precise measurement of the ratio T_0/T_1.

on N, r_{min} and r_{max}) can be computed:

$$\left\{\begin{array}{rcl} r_{min} & \le \Phi^{-1}\left(\frac{M_{(k_A,N)}}{N}\right) \le & r_{max}, \\ -r_{max} & \le \Phi^{-1}\left(\frac{M_{(k_B,N)}}{N}\right) \le & -r_{min}, \end{array}\right. \Leftrightarrow \left\{\begin{array}{rcl} \Phi(r_{min}) & \le \frac{M_{(k_A,N)}}{N} \le & \Phi(r_{max}), \\ \Phi(-r_{max}) & \le \frac{M_{(k_B,N)}}{N} \le & \Phi(-r_{min}). \end{array}\right. \tag{28}$$

The variance $Var(c_k) = \frac{M_{(k,N)}}{N} \cdot \left(1 - \frac{M_{(k,N)}}{N}\right)$ is an increasing function of $\frac{M_{(k,N)}}{N}$ in the interval $[0; \frac{1}{2}]$ and a decreasing one in $[\frac{1}{2}; 1]$. Therefore, the thresholds in Eq. (28) can be used to get bounds of $Var(c_k)$. Recalling that $\Phi(-x) = 1 - \Phi(x)$, we get

$$\underbrace{-\Phi(r_{max})^2 + \Phi(r_{max})}_{\approx 0.0222 \text{ for } r_{max}=2} \le Var(c_k) \le \underbrace{-\Phi(r_{min})^2 + \Phi(r_{min})}_{\approx 0.1335 \text{ for } r_{min}=1} . \tag{29}$$

Note that we do not need to compute the variance. Using Eq. (28), for a given N, we can obtain suitable thresholds for $M_{(k_A,N)}$ and $M_{(k_B,N)}$ that bound $|\alpha_{AB}|$.

For example, for $N = 4\,096$, $r_{min} = 1$ and $r_{max} = 2$, the following inequalities ensure that $|\alpha_{AB}| \le 0.05$:

$$\begin{array}{c} \underbrace{N \cdot \Phi(r_{min})}_{3\,446} \le M_{(k_A,4\,096)} \le \underbrace{N \cdot \Phi(r_{max})}_{4\,003}, \\ \underbrace{N \cdot \Phi(-r_{max})}_{93} \le M_{(k_B,4\,096)} \le \underbrace{N \cdot \Phi(-r_{min})}_{650}, \end{array} \tag{30}$$

which is useful when the goal is to embed the method in logic devices, since only a comparator is needed.

If not enough values of k satisfy these criteria, it is possible to decrease r_{min} and increase r_{max} to find more. Although the error of the method would be increased, its value could still be evaluated precisely.

3.2.2 Error due to approximation of the average periods ratio

Next we examine the origins of $\alpha_{0,1}$. We perform the counting experiments from Fig. 1 for one large L. We denote the obtained value c_L. The waveform of this experiment is given in Fig. 7 and these parameters are related such that $\varphi_0 + c_L \cdot T_1 + r = L \cdot T_0$. Because $0 \le \varphi_0 < T_1$ and $0 \le r < T_1$,

$$0 \le \frac{T_0}{T_1} - \frac{c_L}{L} \le \frac{2}{L} . \tag{31}$$

By choosing a large L, we can get as close as needed to the ratio $\frac{T_0}{T_1}$, from only one, but sufficiently long counting process.

Furthermore, using Eq. (20), we get

$$(k_A - k_B) \cdot \frac{T_0}{T_1} - (F_{k_A} - F_{k_B} - 1) = \Phi^{-1}(\mathcal{A}_{k_A}) \frac{a_{th}}{T_1} \sqrt{F_{k_A}} - \Phi^{-1}(\mathcal{A}_{k_B}) \frac{a_{th}}{T_1} \sqrt{F_{k_B} + 1} . \tag{32}$$

From the study of α_{AB}, we conclude that suitable cases (Case a) and Case b)) should be such that $\Phi^{-1}(\mathcal{A}_{k_A})$ and $-\Phi^{-1}(\mathcal{A}_{k_B})$ are greater than r_{min}. Therefore,

$$(k_A - k_B) \cdot \frac{T_0}{T_1} - (F_{k_A} - F_{k_B} - 1) \geq r_{min} \cdot \frac{a_{th}}{T_1}(\sqrt{F_{k_A}} + \sqrt{F_{k_B} + 1}) \,. \tag{33}$$

Finally, by combining Eq. (31) and (33), we get

$$|\alpha_{0,1}| \leq \frac{2|k_A - k_B|}{L \cdot r_{min} \cdot \frac{a_{th}}{T_1}(\sqrt{F_{k_A}} + \sqrt{F_{k_B} + 1})}. \tag{34}$$

In contrast with the absolute upper bound of $|\alpha_{AB}|$ found, which depended only on N, the upper bound of $|\alpha_{0,1}|$ depends on several parameters. For a given (intrinsic) jitter $^{a_{th}}/_{T_1}$ we need to choose: a large L; a distance $|k_A - k_B|$ as small as possible; and, because k_A (resp. k_B) and F_{k_A} (resp. $F_{k_B} + 1$) have the same order of magnitude, a relatively big k_A and k_B.

To give an example, for an intrinsic jitter $^{a_{th}}/_{T_1}$ greater than 0.5‰, by choosing $L = 65\,535$, and $F_{k_A} \approx F_{k_B} \approx 100$ (to assume thermal noise dominates), we obtain $|\alpha_{0,1}| \leq |k_A - k_B| \cdot 0.003$. Hence, if we want $|\alpha_{0,1}| \leq 0.05$, we have to choose $|k_A - k_B| \leq \frac{0.05}{0.003} = 16.6$. From now on, we impose $|k_A - k_B| \leq 16$. Again, depending on the desired upper bound for $|\alpha_{0,1}|$, we can define a condition for k_A and k_B. In other words, we can tune the condition on $|k_A - k_B|$ to find more couples (k_A, k_B) at the cost of increasing the error on $|\alpha_{0,1}|$ and always find the best trade-off between the number of couples (k_A, k_B) and the error made.

3.2.3 The upper bound of the measurement error

Given the analysis of $|\alpha_{0,1}|$ and $|\alpha_{AB}|$, the relative error computed with Eq. (24) is

$$\left|1 - \frac{\widetilde{a}_{th}}{a_{th}}\right| \leq \underbrace{\sqrt{\frac{\max(F_{k_A}, F_{k_B} + 1)}{\min(F_{k_A}, F_{k_B} + 1)}}}_{\approx\sqrt{\frac{116}{100}}<1.1} \left(\underbrace{|\alpha_{0,1}|}_{0.05} + \underbrace{|\alpha_{AB}|}_{0.05} + \underbrace{|\alpha_{0,1} \cdot \alpha_{AB}|}_{0.0025}\right) < \underbrace{12.3\%}_{\delta_W} \,. \tag{35}$$

We can therefore conclude that the theoretical maximum error (denoted by δ_W) of the jitter measurement achievable with our method is 12.3%.

It will be recalled that these parameters are determined to measure a jitter $^{a_{th}}/_{T_1}$ >0.5‰. They have to satisfy the following criteria:

- $N = 4\,096$; $L = 65\,535$,
- $3\,446 \leq M_{(k_A,N)} \leq 4\,003$; $93 \leq M_{(k_B,N)} \leq 650$; $|k_A - k_B| \leq 16$.

According to Eq. (34), the order of magnitude of the jitter will impact the upper bound δ_W – it will decrease with an increase in $^{a_{th}}/_{T_1}$. It will then be possible to compute the maximum measurement error for a given order of magnitude of the jitter, which depend on the technology used.

In our error analysis we assumed the worst-case scenarios. Therefore, the boundary value $\delta_W = 12.3\%$ would be very difficult to reach. Nevertheless, a conservative approach can use this δ_W to get a lower bound of the measured jitter, i.e.

$$\frac{a_{th}}{T_1} \geq \frac{1}{1 + \delta_W} \cdot \frac{\widetilde{a}_{th}}{T_1} \,. \tag{36}$$

Using this lower bound of the jitter value as an input for a stochastic model, like the one from [BLMT11], the designer can reduce the risk of overestimation of the entropy rate.

For this reason, the analysis of the measurement error is of utmost importance and is one of the main contributions of this article.

Algorithm 1 Algorithm of the new jitter measurement method.

```
procedure JITTER MEASUREMENT($N = 4\,096, L = 65\,535$)
    Initialise $C_k$, ListCase$k_A$, ListCase$k_B$ and List$\widetilde{a_{th}}/T_1$ as empty lists
    for $k := 1$ to 255 do
        Empty $C_k$
        for $i := 1$ to $N$ do
            $c_k \leftarrow$CountingExperience($k$)
    ▷ Get a counter value from the circuit in Fig. 1 with a frequency division factor $k$.
            $C_k$.append($c_k$)
        end for
        $F_k \leftarrow c_k \in C_k | \#\{c_k \in C_k\} > N/2$
        $M_{(k,N)} := \#\{c_k \in C_k | c_k = \max(C_k)\}$
        if $3446 \leq M_{(k,N)} \leq 4003$ then                                ▷ Case $k = k_A$
            ListCase$k_A$.append($\left[k, F_k, M_{(k,N)}\right]$)
        else
            if $93 \leq M_{(k,N)} \leq 650$ then                               ▷ Case $k = k_B$
                ListCase$k_B$.append($\left[k, F_k, M_{(k,N)}\right]$)
            end if
        end if
    end for
    $c_L/L :=$ CountingExperience($L$)$/L$                    ▷ Measurement of the fraction $T_0/T_1$.
    for all $\left[k_A, F_{k_A}, M_{(k_A,N)}\right]$ of ListCase$k_A$ do
        for all $\left[k_B, F_{k_B}, M_{(k_B,N)}\right]$ of ListCase$k_B$ do
            if $|k_A - k_B| \leq 16$ then
                $\frac{\widetilde{a_{th}}}{T_1} := \frac{c_L/L(k_A - k_B) - \left(F_{k_A} - F_{k_B} - 1\right)}{\Phi^{-1}\left(\frac{M_{(k_A,N)}}{N}\right)\sqrt{F_{k_A}} - \Phi^{-1}\left(\frac{M_{(k_B,N)}}{N}\right)\sqrt{F_{k_B}+1}}$ .
                List$\widetilde{a_{th}}/T_1$.append($\widetilde{a_{th}}/T_1$)
            ▷ List of the measured $\widetilde{a_{th}}/T_1$ calculated with every found $(k_A, k_B)$ couple
            end if
        end for
    end for
end procedure
```

3.3 Discussion

If we take the conditions established in Subsection 3.2 into account, the novel jitter measurement method can be described by Algorithm 1. Note that the only operation that needs to be performed in hardware is the acquisition of the counter values for different jitter accumulation times $k \cdot T_0$.

To further confirm the precision of the method, we simulated the measurement process in software. We generated examples following the pseudo-random normal law to simulate the behavior of oscillators affected by thermal noise, like in [GBF+23]. Namely, for given values of φ_0, T_0, T_1 and k, we generated pseudo-random counter values c_k. In our simulations, we randomly selected the ring oscillator average periods ($T_1 = 7\,940$ ps, $T_0 = 7\,462$ ps). We then set a_{th}/T_1 to 1.39‰ to be consistent with jitter sizes published in the state-of-the-art jitter measurement methods [FL14, YRG+17] and greater than 0.5‰ used to compute the error bounds. Algorithm 1 was simulated 100 times, each time, a list of simulated jitter measurements was obtained. Figure 8 shows the results of the simulations, i.e. the distribution of the simulated jitter measurements. The average measured value is represented by the vertical black dashed line and is equal to 1.387‰. The simulated a_{th}/T_1 is represented by the vertical red dashed line. The average error is

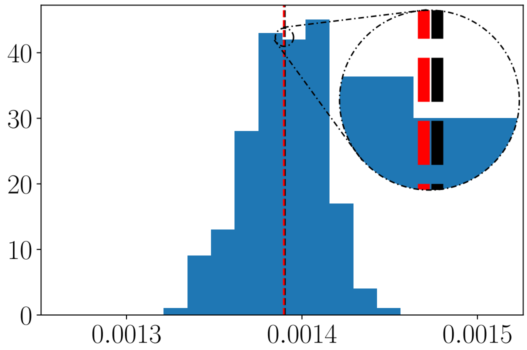

Figure 8: Histogram of the simulated jitter measurements. The value of the jitter entered in the simulator is represented by the vertical black dashed line. The mean measured value obtained in simulations is represented by the vertical red dashed line. The two values are very close, thereby confirming the accuracy of the method.

Table 1: Example of simulation results obtained using Algorithm 1.

k	$Var(C_k)$	$(c_{k,1}, \# \{c_{k,1} \in C_k\})$	$(c_{k,2}, \# \{c_{k,2} \in C_k\})$	Case
53	0.1094	(50, 3 584)	(49, 512)	A
70	**0.1249**	**(65, 3 497)**	**(66, 599)**	B
86	**0.0245**	**(81, 3 993)**	**(80, 103)**	A
120	0.0666	(112, 3 802)	(113, 294)	B
169	**0.0526**	**(159, 3 868)**	**(158, 228)**	A
170	**0.0321**	**(159, 3 960)**	**(160, 136)**	B
252	**0.0641**	**(237, 3 814)**	**(236, 282)**	A
253	**0.0724**	**(237, 3 774)**	**(238, 322)**	B

0.04% and the maximum error is 4.97%.

The results clearly highlight the accuracy and precision of our method. Indeed, the maximum measurement error is much smaller than the upper bound of 12.3% found in Subsection 3.2. As confirmed by our simulations, this theoretical error bound is very conservative and is difficult to reach.

3.4 Illustration of the method

Here we demonstrate how the proposed method works and illustrate its high precision with the following example. We simulated the method based on Algorithm 1. In our simulations, line 24 of the algorithm was executed independently of the previous condition (line 23). The frequency division factor k varied from 1 to 255 in steps of 1. For each k, we acquired $N = 4\,096$ samples of c_k to form a set C_k. Like in Subsection 3.3, we set $\varphi_0 = 6\,335$ ps, $T_0 = 7\,462$ ps, $T_1 = 7\,940$ ps, $L = 65\,535$ and ${}^{a_{th}}/_{T_1}$ =1.39‰.

For certain values of k, two different c_k appeared in the C_k set. Table 1 shows the simulated data for these cases. If we consider the conditions presented in Subsection 3.2, only the six bold entries highlighted in Table 1 can form couples (k_A, k_B) with a small upper bound of the error. Table 1 shows: the frequency division factor k, the variance of the set C_k, the different c_k encountered in C_k and the number of times they appeared in the set, $(c_{k,1}, \# \{c_{k,1} \in C_k\})$ and $(c_{k,2}, \# \{c_{k,2} \in C_k\})$ where $c_{k,1}$ is the most often encountered c_k and $c_{k,2}$ is the second most often encountered c_k, and the case identified (according to Fig. 5), where A indicates that $c_{k,1} > c_{k,2}$ and B indicates that $c_{k,2} > c_{k,1}$.

Since in simulations, the phase of the oscillators is accurately and permanently known,

Table 2: Error analysis of three suitable and two unsuitable (in grey) couples (k_A, k_B) from Table 1.

k_A	k_B	F_{k_A}	F_{k_B}	$M_{(k_A,N)}$	$M_{(k_B,N)}$	${}^{c_L}/_{L}$	${}^{\widetilde{a}_{th}}/_{T_1}$	$\alpha_{0,1}$	α_{AB}	δ_W	$\left\|1-\frac{\widetilde{a}_{th}}{a_{th}}\right\|$
86	**70**	**81**	**65**	**3 993**	**599**	**0.93977**	**1.390‰**	**1.08%**	**0.94%**	**2.25%**	**0.14%**
169	170	159	159	3 868	136	0.93977	1.391‰	-0.04%	-0.11%	0.15%	0.07%
252	253	237	237	3 814	322	0.93977	1.348‰	-0.04%	-3.19%	3.24%	3.15%
53	253	50	237	3 589	322	0.93977	1.510‰	-12.37%	-2.46%	33.03%	8.46%
252	70	237	65	3 814	599	0.93977	1.235‰	10.47%	-0.62%	21.14%	11.27%

we can apply Eq. (24) to calculate a more stringent error bound for any couple (k_A, k_B). Table 2 shows how to use the equations presented in Subsection 3.2 to estimate the maximum error of each measurement. For the three couples (k_A, k_B) listed in Table 1, the upper part of Table 2 shows the resulting values $({}^{\widetilde{a}_{th}}/_{T_1})$. The lower part of Table 2 shows the exact evaluation of the different factors of Eq. (24). It then shows the value of a stringent upper error bound using Eq. (35) starting with the second column. The last column shows the relative measurement error $\left|1-\frac{\widetilde{a}_{th}}{a_{th}}\right|$ that can be computed correctly because as it is a simulation the injected jitter is known. To illustrate the importance of the condition $|k_A - k_B| \leq 16$, we added two unsuitable couples (k_A, k_B) (in grey) that do not satisfy this requirement and whose error upper bound is above 12.3%.

For the three well selected couples (k_A, k_B) we can confirm, Eq. (35) is always verified and its stringent upper bound is far below the worst-case very conservative upper bound of 12.3%. Because this upper error bound comes from the exact evaluation of Eq. (24) it can be very small compared to the absolute bound of 12.3% while still being very conservative.

Our aim is to reduce accumulation time as much as possible while simultaneously minimizing the error. The first row in Table 2 is in bold because we consider it is the optimal jitter measurement. Its relative error is bounded by 2.25% which is not the smallest but allows jitter measurements in the shortest possible accumulation time (86 reference clock periods) which is crucial to limit the influence of flicker noise in real experiments.

In hardware implementations, we cannot compute $\alpha_{0,1}$ as precisely as in simulations. In these cases, we use the obtained k_A, k_B, F_{k_A} and F_{k_B} and suppose ${}^{a_{th}}/_{T_1} \geq 0.5‰$, to compute $|\alpha_{0,1}|$ using Eq. (34). We illustrate this approach in Subsection 4.3.

4 Hardware implementation

Figure 9 illustrates acquisition of a set C_k for $k = 3$ and $N = 2$. The ring oscillator RO_0 is permanently enabled and its output is used as a reference clock for the whole system. The frequency divider determines the length of the measurement interval ($d_k = 1$), during which the number of periods of signal s_1 is counted. The measurement interval is thus synchronous with RO_0. The counter value reached at the end of the measurement interval is sent to the PC and processed using Algorithm 1. The bottom panel in Fig. 9 defines the exact definition of the measurement interval and important timing requirements.

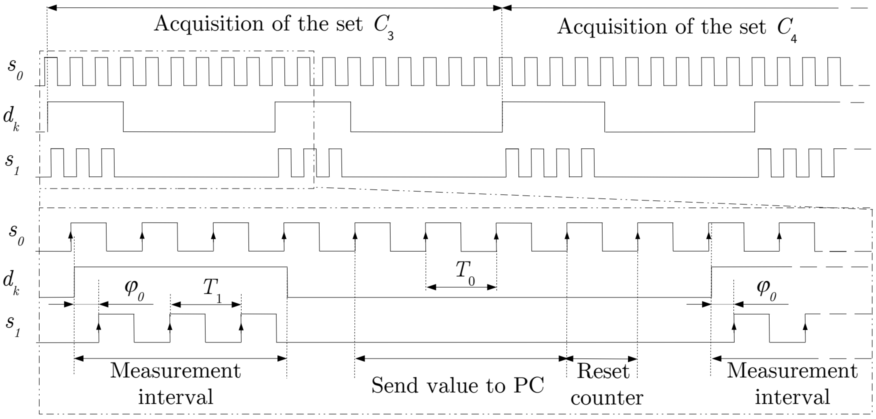

Figure 9: Acquisition of an example set C_3 with $N = 2$ in the jitter measurement circuitry.

4.1 Implementation constraints

Although the circuit shown in Fig. 1 is easy to implement in hardware, two hardware constraints have to be respected to guarantee the precision of the method: routing of the control signal d_k and precision of the counter with respect to possible violation of its setup-and-hold time.

To avoid the possibility that some rising edge of signal s_1 appears after the measurement interval (once the signal d_k goes down), the routing delay between the output of the frequency divider and the counter has to be shorter than the delay between the divider and the control input of RO_1. This is easily achieved by careful placement and routing of the divider, RO_1, and counter.

Another phenomenon that can alter the counting result is violation of the setup-and-hold time of the counter. Considering the principle of the method, the counter stops counting very close to the arrival of the last edge os s_1, hence, violation of the setup-and-hold time of counter flip-flops occurs very frequently.

When a synchronous counter is used, all the flip-flops of the counter are concerned. Therefore, we observed that for a given k, the counter values differed in more than one, unlike what was expected based on Eq. (3).

To demonstrate this phenomenon, we first used the synchronous counter in our measurement hardware. We analyzed a total of 680 000 sets of counter values C_k with a non-zero variance, i.e. $Var(C_k) > 0$. We then calculated the maximum differences between the counter values C_k we obtained using Eq. (3). The observed maximum differences are shown in Fig. 10. The differences were higher than one, meaning that one or more bits within the counter were affected by the time violation. Fortunately, this problem can be solved by using an asynchronous counter as can be seen in Fig. 11 – the difference in counter values was always equal to one for the same set of experiments.

4.2 Assumptions regarding the stability of the measurement conditions

When obtaining Eq. (21) from (20), we assumed that φ_0 and the ratio T_0/T_1 remained constant during acquisition of all the sets $C_k\, \forall k \in [1, 255]$. In our hardware implementation the measurement process included: acquisition of the sets C_k with $N = 4\,096$ and $\forall k \in [1, 255]$; the time to send the counter values to a PC; the time to reset the counter, and the time to measure the ratio T_0/T_1 with $L = 65\,535$. The process lasted at most 2.3 s because it took 3.2×10^5 periods of RO_0 with a mean clock frequency of 136 MHz. Hence, our assumption concerning the stability of φ_0 and that of the ratio T_0/T_1 should be

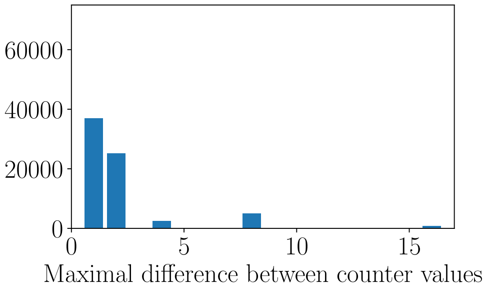

Figure 10: Histogram of maximum differences between counter values obtained using a synchronous counter.

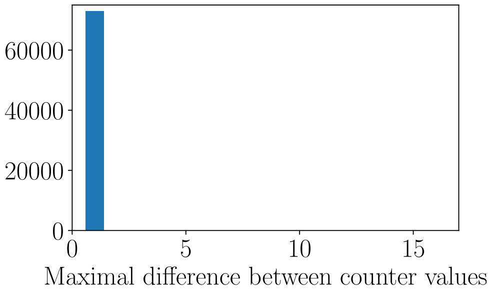

Figure 11: Histogram of maximum differences between counter values obtained using an asynchronous counter.

valid for at least 2.3 s. We measured φ_0, T_0 and T_1 using a LeCroy WaveRunner 9254M oscilloscope at a 40 GS/s sampling rate for a period of 10 s. In order to stabilize the temperature of the board and hence the frequency of oscillators, we let the oscillators run freely for 10 minutes before the measurements. RO_0 and RO_1 were composed of a series of 19 and 20 buffers implemented using LCELL structures ([Int20]) respectively, looped back by a NAND gate. Both rings were manually placed and routed. We observed: a mean of 7.32 ns and standard deviation of 4.4 ps for T_0; a mean of 7.9 ns and standard deviation of 4.8 ps for T_1 and a mean of 0.6 ns and standard deviation of 1.9 ps for φ_0. The values of the standard deviations are very small, indicating that φ_0, T_0 and T_1 are very stable over the 10 s, which largely covers the duration of the measurement. We deduced that the mean of $^{T_0}/_{T_1}$ was stable, too. Our assumptions about stability were thus validated.

The duration of our jitter measurement can be parameterized for a particular implementation of the method. Let us denote k_{max} the largest value of k that we use to look for a suitable couple (k_A, k_B). If k ranges from 1 to k_{max}, the total measurement duration (t_m) can be calculated in terms of N and k_{max},

$$t_m = T_0 \left(N \left(k_{max} \frac{k_{max}+1}{2} + I_c \right) + L + I_c \right) . \tag{37}$$

I_c is the number of cycles of RO_0 that must pass between each acquisition of a counter value. Our stability assumption must hold during t_m. The time interval t_m can be reduced at the cost of reducing N and thus also reducing the precision of the measurement.

4.3 Measurement results

We implemented the circuit in Fig. 1 in three FPGA from different manufacturers : Intel, Xilinx and Microsemi. Subsequent counter values were acquired for k going from 1 to 255 in steps of 1 and processed using Algorithm 1. We set $N = 4\,096$, $L = 65\,535$.

The implementation results for the three FPGA are shown in Table 3. We only show the results for the smallest couple found (k_A, k_B) for each family. The implementation results show that the method only needs very short jitter accumulation times, indeed, around 100 reference clock edges were needed.

We also used the results of the analysis performed in Subsection 3.2 to find the upper bound of the error caused by approximation of $^{a_{th}}/_{T_1}$ by $^{\widetilde{a}_{th}}/_{T_1}$. As we set $N = 4\,096$, we can consider that $|\alpha_{AB}| \leq 0.05$. The upper bound of $|\alpha_{0,1}|$ presented in the first column of Table 3 was then found by applying Eq. (34), based on the assumption that $^{a_{th}}/_{T_1} \geq 0.5‰$. The second column shows δ_W, i.e. the worst case value of the relative measurement error, computed from Eq. (24). Note that because the conditions determined in Subsection 3.2 are respected, $\delta_W \leq 12.3\%$. The last column shows corrected measurement results that

Table 3: Results of jitter measurements of $\widetilde{a_{th}}/T_1$ in three different FPGA families.

FPGA	k_A	k_B	F_{k_A}	F_{k_B}	$\widetilde{a_{th}}/T_1$	$\lvert\alpha_{0,1}\rvert$	δ_W	$\frac{1}{1+\delta_W} \cdot \frac{\widetilde{a_{th}}}{T_1}$
Cyclone V	112	99	105	92	0.9425‰	3.98%	9.76%	0.8586‰
Spartan 6	117	102	103	89	1.087‰	4.66%	10.58%	0.9836‰
SmartFusion 2	115	103	107	72	0.9491‰	3.63%	9.31%	0.8683‰

come from decimating $\widetilde{a_{th}}/T_1$ with δ_W, calculated like in Eq. (36). Thanks to our error analysis we can be sure that the corrected measurement does not overestimate a_{th}/T_1.

5 Comparison with state-of-the-art methods

To be sure our comparison of the proposed method with other published jitter measurement methods ([VABF08], [VFA09], [YRG+17], [FL14]) is fair, we implemented all of them in a Cyclone V FPGA. We measured the jitter of the same couple of ring oscillators using different methods. Both ring oscillators were composed of 20 buffers implemented in logic cells (LCELL) looped back by a NAND gate. The rings were placed and routed manually and they generated clocks with mean frequencies of about 112 MHz. The jitter measurement methods were implemented one after the other in a few hours. During each measurement, we let the oscillators run freely for 10 minutes before beginning the acquisitions so that the clock periods were stable. We acquired the amount of data we needed to make at least 75 jitter measurements using each method. We processed the acquired data in a PC and obtained the boxplots presented in Fig. 12. Although the sources of the jittered clocks (the two rings) were always the same, certain aspects of their implementation were specific to each method:

- [VABF08]: the jitter accumulation time was set to 200 000 cycles of RO_0. We acquired and processed 4 096 counter values per jitter measurement using a PC.
- [VFA09]: we measured an average period difference of 109 ps between the two oscillator clocks using the oscilloscope. We acquired 4 096 counter values per jitter measurement and sent them to a PC for processing.
- [YRG+17]: the accumulation time was set to 344 ns using a precise external quartz oscillator. This time was chosen so that the falling edges of both oscillator clocks always arrived in about the middle of two delay lines implemented using CARRY ([Int20]) chains. The length of the delay lines was set up empirically. To be sufficiently long, each chain had to be composed of 1 000 elements. We acquired and processed 100 000 couples of snapshots per jitter measurement, taken at the same time. As pointed out by the authors, we had to clean the glitches appearing in the snapshots by reordering their bits during data processing.
- [FL14]: we let M vary from 150 to 300 with a 5-unit step and assumed $N = 100$ as recommended by the authors. We acquired and processed 4 096 counter values per M. As pointed out by the authors, some sets of counter values whose mean values were close to N or 0 had to be filtered out during data processing.
- Our method: we varied k from 20 to 175. Because we obtained on average 1.5 jitter measurements per k sweep, we varied k 52 times to obtain 77 jitter measurements, each time, $N = 4\,096$. Every set of counter values was sent to a PC for processing. In order to evaluate the performance of the method regarding k, we filtered the obtained (k_A, k_B) couples so that $(k_A, k_B) \leq 70$. This time, we obtained on average

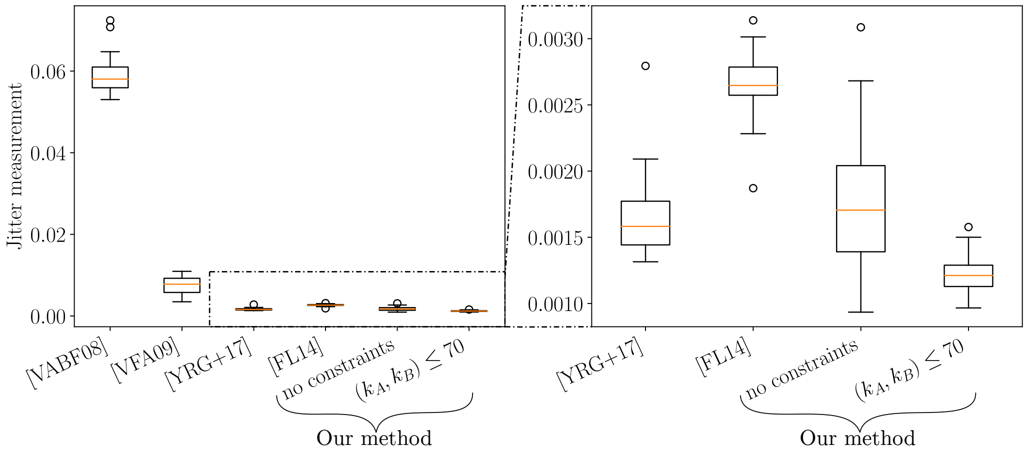

Figure 12: Boxplots of 75 jitter measurements of the five compared methods.

0.4 jitter measurements per k sweep and varied k as many times a necessary to obtain 75 jitter measurements.

Table 4 shows different aspects of the hardware implementations of the five jitter measurement methods. The first column presents the average jitter value obtained using different methods. On one hand, it is clear that the methods in [VABF08] and [VFA09] greatly overestimate the jitter, probably because of much longer jitter accumulation times, which increases the impact of the flicker noise. On the other hand, the methods in [YRG$^+$17] and [FL14] yield results comparable with our novel method. Our method and the method in [YRG$^+$17] produce the lowest mean jitter values. However, in our method, we did not decimate measurements using δ_W, this would yield even smaller and more conservative measurements. Using the proposed error analysis method, we verified that the error of our measurements was less than 6.08%. We noticed that the method in [FL14] yields a slightly higher average jitter measurement than ours or that of [YRG$^+$17] but also uses accumulation times of 300 clock cycles. This may originate from the fact that flicker noise becomes non negligible even with accumulation times as short as 300 clock cycles on the Cyclone V FPGA. This hypothesis is in agreement with the comparison between the last two boxplots in Fig. 12. When the jitter measurements come from accumulation times shorter than 70 cycles the obtained mean measurement is smaller. In other words, the longer we accumulate jitter the more we overestimate the thermal component of jitter, probably because of the influence of flicker noise.

We evaluated the *precision* of the methods through the standard deviation of their measurements. The results are listed in the second column of Table 4. The method in [FL14] is the most precise with a standard deviation of 2×10^{-4}. However, our method can greatly reduce the standard deviation at the cost of acquiring more data per measurement. For example, if we impose the condition k_A, $k_B \leq 70$, the measurements have a standard deviation of 1.24×10^{-4}. Hence, our method is more precise at the cost of obtaining less measurements per k sweep.

Although none of the methods we evaluated was fully embedded in hardware, we can compare the *area* of the measurement block of the method expressed by a number of ALMs. The results are presented in the third column of Table 4. Note the method in [YRG$^+$17] is the most expensive, requiring at least six times the area of our method or that of [FL14].

The *measurement rate* of the methods can also be expressed as the number of reference clock periods, needed to obtain a single jitter measurement. Based on this criterion, the method [FL14] requires the smallest number of periods, followed by the method [YRG$^+$17], with 4.1×10^5 and 4.3×10^6 cycles respectively. The fifth column in Table 4 indicates

Table 4: Measurement and implementation results of the five jitter measurement methods while measuring the same jittered clocks in a Cyclone V FPGA.

Method	Average jit.	Std. dev.	ALMs	RO_0 cycles	P. C.	Power [mW]
[VABF08]	58.6‰	3.63×10^{-3}	163	8.19×10^8	No	4.4
[VFA09]	7.47‰	2.09×10^{-3}	127	4.5×10^5	Yes	4.4
[YRG+17]	1.63‰	2.43×10^{-4}	1759	4.3×10^6	Yes	20.9
[FL14]	2.66‰	2×10^{-4}	266	4.1×10^5	Yes	9.9
Our method	1.73‰	4.48×10^{-4}	260	6.15×10^7	No	8.8
$k_A, k_B \leq 70$	1.22‰	1.24×10^{-4}	260	5.12×10^5	No	8.8

whether the method needs a prior calculation (P.C.) before the jitter can be measured. From this point of view, in method [VFA09], the difference between the average periods must be sufficiently small as detailed in [GBF+23]. The method [YRG+17] requires the delay chains to be characterized, which takes more than 2.58×10^7 RO_0 cycles. Finally the method [FL14] requires verifying that the couple of ring oscillators used is suitable for the measurement. This should be done by decomposing their average periods using continuous fractions. Finally, our new method requires 6.15×10^7 RO_0 cycles to measure the jitter because many k values have to be skipped, as shown in Fig. 4. However, our new method does not require any prior calculations nor does it impose any particular constraints.

In the sixth column of Table 4 we compare the power consumption of the FPGA when instantiating different evaluated methods. The method in [YRG+17] consumes the most power, and is prone to consume exponentially more if the oscillator frequency is increased because the oscillating signals must pass across very long delay chains.

6 Conclusion and future work

In this paper, we presented a new method for measuring clock jitter that can be easily embedded in logic devices. The method allows very accurate measurements and jitter accumulation times as short as 100 reference clock periods. We conducted a thorough study including analyzing timing constraints and the precision of the method. We identified and quantified possible sources of errors and their effects on jitter measurement. All the parameters that determine measurement error were optimized theoretically and demonstrated practically. We also show how to set up the parameters of the method to achieve the desired error level and to determine the precise upper bound of the error. We stand by the principle that the jitter measurement results do not overestimate the jitter. Consequently, they can be safely used to compute the entropy rate, for example using the stochastic model from [BLMT11], to guarantee the security of the ERO-TRNG.

According to the simulations made in [GBF+23], other methods ([FL14],[YRG+17]) can attain a mean error of about 10%. Our simulations show that our method can reach a mean error as low as 0.04% and a maximum error of up to 5%. This is consistent with our very conservative upper bound of the error of 12.3% – the errors we found were always much smaller.

From the analysis presented we can draw several conclusions: first, jitter measurement methods based on long accumulation time greatly overestimate the thermal component of the jitter, so methods based on short accumulation time should be largely preferred. Second, although our method can be easily embedded in logic devices, like that of [FL14], it also uses very short accumulation times, like that of [YRG+17]. Our new method is thus the best compromise between accuracy and ease of hardware implementation. Moreover, while our method has relatively slow measurement rate, it does not require any previous calculation.

In our theoretical analysis and simulations, we assumed that only thermal noise affected the stability of oscillator clocks. However, as explained in [HLL99], flicker noise becomes increasingly important at higher frequencies as modern transistor channels are continuously shrinking, hence overcoming the thermal jitter component faster. This can lead to overestimating entropy and can compromise the security of the entire cryptographic system. For example, according to our implementation results, the method from [FL14] yields a slightly higher average jitter measurement than ours or that of [YRG+17], but uses accumulation times about three times longer than our method. It is thus more impacted by the jitter coming from the flicker noise than other two methods.

To avoid overestimating entropy, two options remain: avoiding the impact of flicker noise on the measured jitter by reducing the jitter accumulation time or including the impact of flicker noise, including autocorrelation, on the total measured clock jitter and on the stochastic model. The novel method we propose in this paper will be further improved regarding both these aspects.

References

[BLMT11] Mathieu Baudet, David Lubicz, Julien Micolod, and André Tassiaux. On the security of oscillator-based random number generators. *Journal of Cryptology*, 24(2):398–425, April 2011.

[FL14] Viktor Fischer and David Lubicz. Embedded evaluation of randomness in oscillator based elementary TRNG. In Lejla Batina and Matthew Robshaw, editors, *CHES 2014*, volume 8731 of *LNCS*, pages 527–543. Springer, Heidelberg, September 2014.

[GBF+23] Arturo Mollinedo Garay, Florent Bernard, Viktor Fischer, Patrick Haddad, and Ugo Mureddu. An evaluation procedure for comparing clock jitter measurement methods. In Ileana Buhan and Tobias Schneider, editors, *Smart Card Research and Advanced Applications - CARDIS 2023*, pages 167–187. Springer International Publishing, 2023.

[HLL99] Ali Hajimiri, Sotirios Limotyrakis, and Thomas H. Lee. Jitter and phase noise in ring oscillators. *IEEE J. Solid State Circuits*, 34(6):790–804, 1999.

[HTBF14] Patrick Haddad, Yannick Teglia, Florent Bernard, and Viktor Fischer. On the assumption of mutual independence of jitter realizations in P-TRNG stochastic models. In Gerhard P. Fettweis and Wolfgang Nebel, editors, *Design, Automation & Test in Europe Conference & Exhibition, DATE 2014, Dresden, Germany, March 24-28, 2014*, pages 1–6. European Design and Automation Association, 2014.

[Int20] Intel. *Cyclone V Device Handbook*. Intel Corporation, July 2020.

[ISO19] Information technology – Security techniques – Test and analysis methods for random bit generators within ISO/IEC 19790 and ISO/IEC 15408. Technical report, International Organization for Standardization, October 2019.

[KS11] Wolfgang Killmann and Werner Schindler. A proposal for: Functionality classes for random number generators, AIS20/31. Technical report, Bundesamt für Sicherheit in der Informationstechnik (BSI), Bonn, 2011.

[SMS07] Berk Sunar, William J. Martin, and Douglas R. Stinson. A provably secure true random number generator with built-in tolerance to active attacks. *IEEE Trans. Computers*, 56(1):109–119, 2007.

[VABF08] Boyan Valtchanov, Alain Aubert, Florent Bernard, and Viktor Fischer. Modeling and observing the jitter in ring oscillators implemented in FPGAs. In Bernd Straube, Milos Drutarovský, Michel Renovell, Peter Gramata, and Mária Fischerová, editors, *Proceedings of the 11th IEEE Workshop on Design & Diagnostics of Electronic Circuits & Systems (DDECS 2008), Bratislava, Slovakia, April 16-18, 2008*, pages 158–163. IEEE Computer Society, 2008.

[VFA09] Boyan Valtchanov, Viktor Fischer, and Alain Aubert. A coherent sampling based method for estimating the jitter used as entropy source for True Random Number Generators. In *International Conference on Sampling Theory and Applications - SAMPTA 2009*, 2009.

[VFAB10] Boyan Valtchanov, Viktor Fischer, Alain Aubert, and Florent Bernard. Characterization of randomness sources in ring oscillator-based true random number generators in FPGAs. In Elena Gramatová, Zdenek Kotásek, Andreas Steininger, Heinrich Theodor Vierhaus, and Horst Zimmermann, editors, *13th IEEE International Symposium on Design and Diagnostics of Electronic Circuits and Systems, DDECS 2010, Vienna, Austria, April 14-16, 2010*, pages 48–53. IEEE Computer Society, 2010.

[YRG+17] Bohan Yang, Vladimir Rozic, Milos Grujic, Nele Mentens, and Ingrid Verbauwhede. On-chip jitter measurement for true random number generators. In *2017 Asian Hardware Oriented Security and Trust Symposium, AsianHOST 2017, Beijing, China, October 19-20, 2017*, pages 91–96. IEEE Computer Society, 2017.