Jitter Estimation with High Accuracy for Oscillator-Based TRNGs

Shaofeng Zhu[1,2], Hua Chen[1]✉, Limin Fan[1], Meihui Chen[1,2], Wei Xi[3], and Dengguo Feng[1]

[1] Trusted Computing and Information Assurance Laboratory, Institute of Software, Chinese Academy of Sciences, Beijing, China
{zhushaofeng,chenhua,fanlimin,chenmeihui,feng}@tca.iscas.ac.cn
[2] University of Chinese Academy of Sciences, Beijing, China
[3] Southern Power Grid Science Research Institute, Guangzhou, China
xiwei@csg.cn

Abstract. Ring oscillator-based true random number generators (RO-based TRNGs) are widely used to provide unpredictable random numbers for cryptographic systems. The unpredictability of the output numbers, which can be measured by entropy, is extracted from the jitter of the oscillatory signal. To quantitatively evaluate the entropy, several stochastic models have been proposed, all of which take the jitter as a key input parameter. So it is crucial to accurately estimate the jitter in the process of entropy evaluation. However, several previous methods have estimated the jitter with non-negligible error, which would cause the overestimation of the entropy. In this paper, we propose a jitter estimation method with high accuracy. Our method aims at eliminating the quantization error in previous counter-based jitter estimation methods and finally can estimate the jitter with the error smaller than 1%. Furthermore, for the first time, we give a theoretical error bound for our jitter estimation. The error bound confirms the 1% error level of our method. As a consequence, our method will significantly help to evaluate the entropy of RO-based TRNGs accurately. Finally, we present the application of our jitter estimation method on a practical FPGA device and provide a circuit module diagram for on-chip implementation.

Keywords: TRNG · ring oscillator · jitter · estimation · entropy.

1 Introduction

Ring oscillator-based true random number generator (RO-based TRNG) is a widely used kind of TRNGs for its simple implementation on logic devices such as FPGAs and smart cards. The elementary structure of RO-based TRNG is shown by Fig. 1. A slow clock signal (S_s) samples a fast oscillatory clock signal (S_o) generated by an oscillator composed of an odd number of inverters. Under the effect of correlated random noise (mainly low-frequency flicker noise) and uncorrelated random noise (mainly thermal noise) on the logic devices [6], the periods of the oscillatory signal will vary randomly. The deviation of the periods

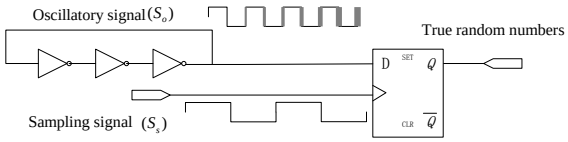


Fig. 1. Elementary structure of RO-based TRNG

is usually defined as the period jitter. So the jitter is mainly composed of thermal jitter and flicker jitter which are respectively contributed by the thermal noise and the flicker noise. Then the jitter is exploited by the TRNG to extract random numbers.

The randomness of a TRNG is mainly about the unpredictability of the generated random numbers. The unpredictability can be quantitatively measured by the entropy rate of the random numbers. Unfortunately, the traditional statistical test suites such as NIST SP800-22 [10], DIEHARD [9] merely evaluate the statistical properties of the output numbers, but can not answer whether the numbers to be tested hold enough entropy. In order to evaluate the entropy of RO-based TRNGs, several stochastic models have recently been proposed [1, 4, 7, 8], all of which show that jitter is the key parameter that directly affects the entropy rate. Consequently, it is crucial to precisely estimate the jitter.

Up to now, several jitter estimation methods have been proposed. It is quite inaccurate to estimate the jitter outside the device with measuring equipments such as oscilloscopes [13], since additional jitter would be introduced by the Input/Output circuits and pins [14]. To estimate the jitter internally, Valtchanov et al. [14] designed an embedded circuit to count the rising edges of the oscillatory signal in equal-length time intervals and took the standard deviation of the counting results as an approximate measure of the accumulated jitter in the interval. Since the counting results can only be integers, this method will introduce in quantization error when estimating the jitter. Ma et al. [7] improved Valtchanov et al.'s counter-based method by counting both the rising and falling edges of the oscillatory signal. Such improvement actually reduces the quantization step size by half, thus can decrease the quantization error. Nevertheless, the quantization error is still not eliminated. Fischer et al. [2] proposed a different method based on Monte Carlo approach, which could estimate the jitter with the error smaller than 5%. Note that all the above mentioned methods are actually to estimate the total jitter containing both thermal jitter and flicker jitter. Nevertheless, most of the stochastic models for entropy evaluation are based on the common assumption that the periods of the oscillatory are independently and identically distributed (i.i.d.) under the effect of thermal noise. This requires only the jitter contributed by the thermal noise to be used to calculate the entropy. It is known that the thermal jitter is difficult to be estimated directly. Recently, Haddad et al. [3] proposed an approach to separate the thermal jitter from the total jitter and gain the ratio of the thermal jitter in the total jitter.

Nevertheless, the estimation of the total jitter in their work is also based on a counter-based method. So quantization error will inevitably be brought in, but was not considered as well.

In this paper, we provide a highly accurate jitter estimation method for RO-based TRNGs. Our method aims at eliminating the error that exists in the previous counter-based methods. Compared to the previous ones, our method can estimate the total jitter with much lower error level, which is also confirmed by theoretical analysis.

In summary, our contributions include:

- **We propose a jitter estimation method with high accuracy for RO-based TRNGs.** As we investigated, non-negligible quantization error is introduced in the previous counter-based jitter estimation methods. After eliminating the quantization error, in the meanwhile taking the waiting time in the sampling process into account, we provide a new, more accurate estimation for the jitter with the error level below 1%, which is much lower than the previous methods. This will significantly help to evaluate the entropy of a RO-based TRNG accurately.
- **For the first time, we give a theoretical error bound for the jitter estimation.** We adopt quantization error analysis approaches and present a formal upper error bound for our jitter estimation. This error bound has confirmed the 1% error level of our method in theory.
- **With our method, we provide a practical jitter estimation on FPGA device.** We demonstrate that combined with the jitter separation approach in [3], our method can be used to estimate the thermal jitter on practical hardware platforms. We also provide a circuit module diagram of our method for on-chip implementation.

The organization of this paper is as follows: In Section 2, we introduce the preliminaries about signal model, entropy evaluation and jitter estimation. In Section 3, we analyze the error of the previous counter-based jitter estimation method given by [7] and propose our jitter estimation method. In Section 4, we give the theoretical error analysis of our method. In Section 5, we conduct a practical jitter estimation on an FPGA device with our method, and we present the circuit module diagram of our method for on-chip implementation. In Section 6, we compare our method with the previous ones and give the conclusion.

2 Preliminaries: Signal Model, Entropy Evaluation and Jitter Estimation

In this section, we first present the signal model of an elementary RO-based TRNG, where we define symbols to describe the signals. Then we introduce the entropy evaluation methods of RO-based TRNGs. The methods take the jitter as an important parameter to calculate the entropy. As a consequence, jitter estimation is crucial and will determine the accuracy of the entropy evaluation.

2.1 Signal Model

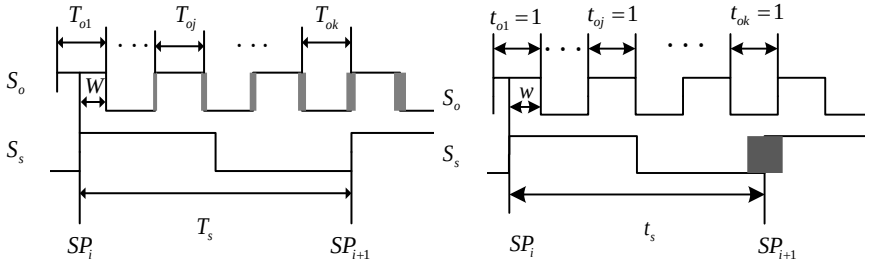
For the RO-based TRNGs, the sampling process can be approximately treated as a stationary process, so we just consider two successive samplings. Here we define symbols to describe the oscillatory signal (S_o) and the sampling signal (S_s) by Definition 1 and Fig. 2(a).

Definition 1. *The time interval between two successive samplings SP_i and SP_{i+1} is denoted by T_s. Within T_s, the edge intervals of S_o are denoted by $T_{o1} \cdots T_{oj} \cdots T_{ok}$. The standard deviation of T_{oj} is defined as the half period jitter of S_o and denoted by σ_o. (σ_o)s will be accumulated in T_s. The mean value of T_{oj} is the half mean period of S_o and denoted by μ_o. The waiting time W is defined as the time interval between SP_i and the following closest edge of S_o. According to [4] and [7], W approximately follows the uniform distribution within $[0, \mu_o]$ because of $\sigma_o \ll \mu_o$, and it is independent from the T_s in the current sampling interval.*

The μ_o can be measured from the frequency of S_o. For brevity, we normalize all the time variables with μ_o, that is $T_s \to t_s = \frac{T_s}{\mu_o}$, $T_{oj} \to t_{oj} = \frac{T_{oj}}{\mu_o}$, $\sigma_o \to \sigma = \frac{\sigma_o}{\mu_o}$, $\mu_o \to 1$ and $W \to w = \frac{W}{\mu_o} \sim \boldsymbol{U}(0,1)$. The normalized variables can be transformed back to time variables by multiplying by μ_o.

Since the jitter is relative between the two signals, an equivalent model can be presented by treating S_o as stable while S_s has period jitter. The equivalent model is illustrated by Definition 2 and Fig. 2(b).

Definition 2. *The edge intervals of S_o are $t_{o1} = \cdots = t_{oj} = \cdots = 1$. The sampling interval t_s is a random variable with mean value μ_s and standard deviation σ_s. σ_s is defined as the total jitter accumulated in the interval t_s. The jitters from thermal noise and flicker noise are respectively denoted by σ_s^{th}, σ_s^{fl}. Since the two kinds of noise are mutually independent, there is $\sigma_s^2 = (\sigma_s^{th})^2 + (\sigma_s^{fl})^2$. Besides, we still have $w \sim \boldsymbol{U}(0,1)$ and it is independent from the current t_s.*



(a) Signal Model (b) Equivalent Signal Model

Fig. 2. Signal Model of RO-based TRNG

2.2 Entropy Evaluation

Previous articles such as [1] and [7] have given the methods to evaluate the entropy of RO-based TRNGs. In order to mathematically characterize the RO signals, the articles only take the uncorrelated thermal noise into consideration. Then under the affection of thermal noise, the edge intervals $T_{o1} \cdots T_{oj} \cdots T_{ok}$ will be i.i.d. with Gaussian distribution $\boldsymbol{N}(\mu_o, \sigma_o^2)$. Correspondingly in the equivalent model, there is $t_s \sim \boldsymbol{N}(\mu_s, (\sigma_s^{th})^2)$. Under the above assumption, according to [1], the min-entropy can be calculated by (1)[4].

$$H_{min} = 1 - \frac{4}{\pi^2 \ln(2)} e^{-\pi^2 (\sigma_s^{th})^2}. \tag{1}$$

The calculated entropy is actually contributed by the thermal noise and it can be a conservative estimation for the min-entropy of RO-based TRNGs.

We can see the min-entropy is determined by the σ_s^{th} in the sampling interval t_s. Hence, it is crucial to estimate σ_s^{th} precisely for entropy evaluation.

2.3 Jitter Estimation

For a practical RO-based TRNG, if the sampling frequency is high, the accumulated jitter in t_s may be too small to be estimated accurately. So we usually estimate the accumulated jitter in a larger measuring interval. Here we denote the measuring interval by t_m ($= \frac{T_m}{\mu_o}$, T_m is time variable) with mean value μ_m and standard deviation σ_m. σ_m represents the total jitter accumulated in t_m. The thermal jitter is "sqrt" accumulated with the time interval [1], [6], [3]. So after estimating the total jitter σ_m and separating the thermal jitter σ_m^{th} from σ_m, we can calculate the needed thermal jitter σ_s^{th} accumulated in the sampling interval t_s by

$$\sigma_s^{th} = \sqrt{\frac{t_s}{t_m}} \sigma_m^{th}. \tag{2}$$

When the measuring interval is short enough so that the thermal jitter dominates over the flicker jitter, there is $\sigma_m^{th} \approx \sigma_m$, and the σ_s^{th} can also be estimated by

$$\sigma_s^{th} \approx \sqrt{\frac{t_s}{t_m}} \sigma_m. \tag{3}$$

Anyway, it is necessary to estimate the total accumulated jitter σ_m first and we focus on the estimation of σ_m as well.

3 Our Proposed Jitter Estimation Method

We present our jitter estimation method in this section. Firstly, we investigate the error of the previous counter-based jitter estimation method introduced by Ma et al. in [7]. Results show that non-negligible error exists in Ma's method. Our proposed method gives a new estimation for the total jitter and is able to achieve a much lower error level than the previous one.

[4] $(\sigma_s^{th})^2/4$ is equivalent to the quality factor Q defined in [1]

3.1 Error Investigation of Previous Counter-Based Jitter Estimation Method

We primarily investigate the previous, typical counter-based method proposed by Ma et al. [7]. Under the equivalent signal model, this method actually counts both the rising and falling edges of S_o in series of interval t_ms and approximates the variance of the counting result X to the variance of t_m:

$$\mathrm{Var}(t_m) \approx \mathrm{Var}(X). \tag{4}$$

Then the jitter σ_m is estimated by

$$\sigma_m = \sqrt{\mathrm{Var}(t_m)} \approx \sqrt{\mathrm{Var}(X)}. \tag{5}$$

The approximation between $\mathrm{Var}(t_m)$ and $\mathrm{Var}(X)$ is critical in this counter-based method, since X is measurable on the chip by edging counting.

According to Fig. 2(b) in Section 2, the edge-counting result X in the interval t_m is actually the flooring quantized value of $(t_m - w + 1)$ with the quantization size $q = 1$, that is

$$X = \lfloor t_m - w + 1 \rfloor_{q=1}. \tag{6}$$

Therefore, the waiting time factor of $(-w+1)$ and the flooring quantization will definitely introduce error in Ma's method.

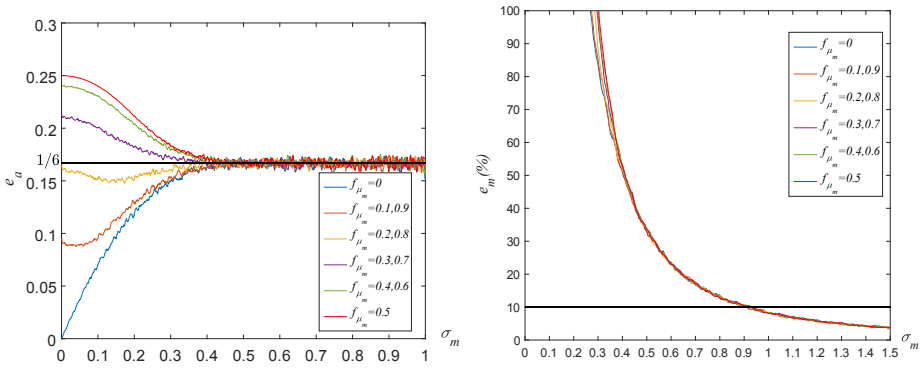
We investigate the error of Ma's method by Matlab simulation. The absolute error (e_a) and relative error (e_r) of the approximation (4) can be calculated with

$$e_a = \mathrm{Var}(X) - \mathrm{Var}(t_m), e_r = \frac{|e_a|}{\mathrm{Var}(t_m)}. \tag{7}$$

According to (5), the estimation error of σ_m (denoted by e_m) is equal to $\frac{1}{2}e_r$. e_m can be a measure of the error level of the jitter estimation method. With Matlab, we generate the instances of $t_m \sim \boldsymbol{N}(\mu_m, \sigma_m^2)$ with different size of σ_m and corresponding instances of X. Here the flicker noise is not considered, since to our knowledge, it is infeasible to be generated with simulation by now. Then we evaluate the e_a and e_m of Ma's method. The results are shown in Fig. 3. It can be seen that a $\frac{1}{6}$ absolute error always exists in the approximation (4) when $\sigma_m > 0.4$. While $\sigma_m < 0.4$, the absolute error e_a would be even larger and related with the fractional part of μ_m (denoted by f_{μ_m})[5]. The error e_m of this method is larger than 10% until $\sigma_m > 0.92$.

On one aspect, the error level of this method is certainly not low (10%), and non-negligible absolute error inherently exists in their estimation. Consequently, once adopted in entropy evaluation, this method will overestimate the jitter, and the entropy of RO-based TRNGs will be overestimated as well. On another aspect, this method requires $\sigma_m > 0.92$ to gain the 10% error level. For a practical RO, since the jitter can only be more accumulated by increasing the measuring interval, this method needs a large measuring interval to accumulate enough jitter for its accuracy.

[5] Different f_{μ_m}s are indicated by different colors as well as in following figures.



(a) Absolute error e_a (b) Error level e_m

Fig. 3. Errors evaluation of Ma's method by Matlab

3.2 New Estimation for the Jitter

In order to correct the error in Ma's method, we take a close look into the relationship between $\mathrm{Var}(t_m)$ and $\mathrm{Var}(X)$. Then we eliminate the quantization error and the effect of the waiting time factor to give an improved approximation for $\mathrm{Var}(t_m)$. Based on this approximation, we present our new, more accurate estimation for the jitter.

Firstly, we introduce the "Sheppard's correction" in quantization theory. **Sheppard's correction [11]:** For a random variable v with continuous distribution, its rounding quantized value with quantization step q can be denoted by $v_q = [v]_q$. When the variance of v is large enough, the quantization error $e_q = v - v_q$ will approximately follow uniform distribution in $(-q/2, q/2)$ and be independent from v. The first-order and second-order moments of v and v_q have the following relationships [11]:

$$\mathrm{E}(v) = \mathrm{E}(v_q), \mathrm{E}(v^2) \approx \mathrm{E}(v_q^2) - q^2/12. \tag{8}$$

In the jitter estimation case, we know that the edge-counting result in the interval t_m is

$$X = \lfloor t_m - w + 1 \rfloor_{q=1} = [t_m - w + 0.5]_{q=1}. \tag{9}$$

So according to the "Sheppard correction", when $\mathrm{Var}(t_m - w + 0.5)$ is large enough, the quantization error in the jitter estimation is

$$e_q = (t_m - w + 0.5 - X) \sim \boldsymbol{U}(-0.5, 0.5) \tag{10}$$

and e_q will be independent from $(t_m - w + 0.5)$. Besides, the equivalent signal model in Section 2 has indicated that $w \sim \boldsymbol{U}(0, 1)$ and it is independent from the current measuring interval t_m, so we have

$$\mathrm{Var}(X) = \mathrm{Var}(t_m - w + 0.5 - e_q) \approx \mathrm{Var}(t_m) + \mathrm{Var}(w) + \mathrm{Var}(e_q). \tag{11}$$

From (11) we can see the deviation between $\mathrm{Var}(t_m)$ and $\mathrm{Var}(X)$ is indeed caused by the quantization error e_q and waiting time w. Consequently, we give the new approximation for $\mathrm{Var}(t_m)$:

$$\mathrm{Var}(t_m) \approx \mathrm{Var}(X) - \mathrm{Var}(w) - \mathrm{Var}(e_q) \approx \mathrm{Var}(X) - 1/6. \tag{12}$$

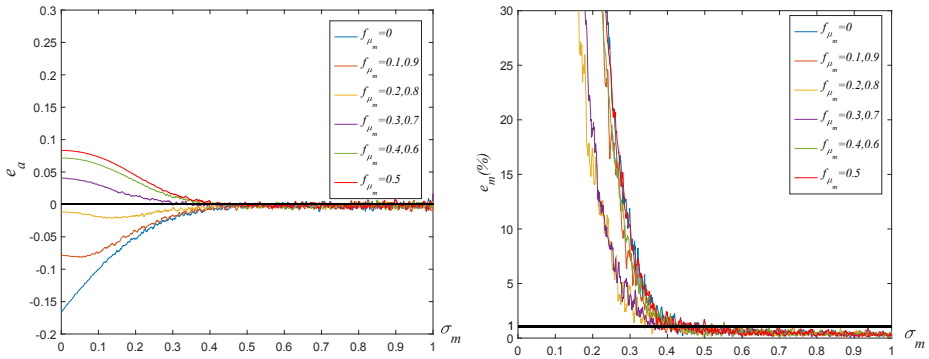
Based on the approximation (12), we present our new, more accurate estimation of σ_m by

$$\sigma_m \approx \sqrt{\mathrm{Var}(X) - 1/6}. \tag{13}$$

In the same way, the absolute and relative errors of approximation (12) can be calculated by

$$e_a = \mathrm{Var}(X) - 1/6 - \mathrm{Var}(t_m), e_r = \frac{|e_a|}{\mathrm{Var}(t_m)}, \tag{14}$$

and the error level e_m of our method is also equal to $\frac{1}{2}e_r$. By Matlab simulation, we evaluate the errors (e_a and e_m) and show them in Fig. 4. We can see our



(a) Absolute error e_a (b) Error level e_m

Fig. 4. Errors evaluation of our method by Matlab

estimation has successfully eliminate e_a when $\sigma_m > 0.4$. Correspondingly, the error level (e_m) of our method gets down to lower than 1% as long as $\sigma_m > 0.4$.

This is an obvious improvement over Ma's method. Firstly, our estimation can achieve much lower error level (1%) than Ma's method (10%). Secondly, our method can eliminate the absolute error which inherently exists in Ma's method. This will avoid overestimating the jitter. Moreover, our method needs much shorter measuring time interval, which can speed up the jitter estimation process.

3.3 An Efficient Calculation of the Variance of X

Jitter estimation should be fast for some application scenarios such as on-line health test. Considering that the calculation of $\mathrm{Var}(X)$ is the most time-consuming in counter-based jitter estimation method, we present an efficient approach to do this calculation.

As we know, if the samples of the counting result X are $x_1, \cdots, x_N$, then the ordinary variance calculating formula can be presented by

$$\mathrm{Var}(X) = \frac{\sum_{j=1}^{N} x_j^2}{N} - \Big(\frac{\sum_{j=1}^{N} x_j}{N}\Big)^2, \tag{15}$$

which needs $N+1$ multiplications. N is the sample size.

In view of modern logic devices, the jitter accumulated in the time interval t_m is usually very small, and the edge-counting results will vary slightly around the mean value $\overline{x} = \frac{\sum_{j=1}^{N} x_j}{N}$. That is, the sample space of X is small too and we denote it by $\mathcal{S}_X = \{p_i | p_i = \lfloor \overline{x} \rfloor - I + i; 1 \le i \le 2I; 5 \le I \ll N\}$. Here we recommend $5 \le I$ so that $\mathcal{S}_X$ can cover most of the counting results. Our approach is to count the number of X's samples on each sample point p_i, and the results are denoted by $c_1, \ldots, c_{2I}$. Then $\mathrm{Var}(X)$ can be calculated by

$$\mathrm{Var}(X) = \frac{\sum_{i=1}^{2I} c_i \cdot (p_i - \overline{x})^2}{N}. \tag{16}$$

Only $4I$ ($\ll N+1$) multiplications are needed in (16). Evidently, the efficiency of the jitter estimation is improved.

We present the corresponding Algorithm 1 for this approach.

Algorithm 1 Algorithm for the calculation of $\mathrm{Var}(X)$.

Input: The counting result $x_1, \cdots, x_N$. Parameters N and I.
Output: $\mathrm{Var}(X)$.

1: Calculating the mean value of $x_1, \cdots, x_N$: $\overline{x} \leftarrow \frac{\sum_{j=1}^{N} x_j}{N}$
2: Calculating the sample points of X:
  for $i = 1, \cdots, 2I$ do
  $p_i = \lfloor \overline{x} \rfloor - I + i$;
  end for;
3: Counting $x_1, \cdots, x_N$ on $p_1, \cdots, p_{2I}$:
  Set $c_1, \cdots, c_{2I} = 0$;
  for $j = 1, \cdots, N$ do
    for $i = 1, \cdots, 2I$ do
      if $(x_j = p_i)$ $c_i = c_i + 1$; end if;
    end for;
  end for;
4: Calculating $\mathrm{Var}(X)$: $\mathrm{Var}(X) \leftarrow \frac{\sum_{i=1}^{2I} c_i \cdot (p_i - \overline{x})^2}{N}$
5: **return** $\mathrm{Var}(X)$;

4 Theoretical Error Analysis

In this section, we theoretically analyze our method and give a formal error bound, which confirms the 1% error level of our method in theory.

The error e_a is affected by t_m and w. So we expand e_a in complex Fourier series based on the characteristic functions of t_m and w, then we formally express e_a and give the upper bound of the error e_m with the following steps.

Step 1. Definition of Equivalent Variable v. Firstly, we define v, its quantized value v_q ($q = 1$) and the quantization error e_q respectively by

$$v = t_m - w + 0.5 - \lfloor \mu_m \rfloor, v_q = [v] = X - \lfloor \mu_m \rfloor, e_q = v - v_q. \tag{17}$$

Step 2. Expression of e_a with v and v_q. The absolute error e_a in our estimation can be presented by

$$e_a = \mathrm{Var}(X) - \frac{q^2}{12} - \mathrm{Var}(w) - \mathrm{Var}(t_m) = \mathrm{Var}(v_q) - \mathrm{Var}(v) - \frac{q^2}{12}. \tag{18}$$

According to the "Sheppard's correction" on the first-order moment (8), mean value $\mathrm{E}(v_q)$ equals to $\mathrm{E}(v)$, so we have

$$e_a = \mathrm{E}(v_q^2) - \mathrm{E}(v^2) - \frac{q^2}{12} = 2\,\mathrm{E}(ve_q) + \mathrm{E}(e_q^2) - \frac{q^2}{12}. \tag{19}$$

Step 3. Expression of e_a in Fourier series with $\boldsymbol{W}_v(\alpha)$. The characteristic function of v is

$$\boldsymbol{W}_v(\alpha) = \int_{-\infty}^{\infty} f(v) e^{j\alpha v} dv. \tag{20}$$

Here we define $v_0 = v - \mu_v$, where $\mu_v = \mathrm{E}(v)$, then its characteristic function is

$$\boldsymbol{W}_{v_0}(\alpha) = e^{-j\alpha\mu_v} \boldsymbol{W}_v(\alpha). \tag{21}$$

According to [12] [5], the $\mathrm{E}(ve_q)$ and $\mathrm{E}(e_q^2)$ in (19) can be expressed in the form of complex Fourier series based on $\boldsymbol{W}_{v_0}(\alpha)$ and its derivation $\dot{\boldsymbol{W}}_{v_0}(\alpha)$:

$$\begin{aligned} \mathrm{E}(ve_q) = {} & \frac{q}{\pi} \sum_{k=1}^{\infty} \cos\Big(\frac{2\pi k}{q}\mu_v\Big) \dot{\boldsymbol{W}}_{v_0}\Big(\frac{2\pi k}{q}\Big) \frac{(-1)^{k+1}}{k} \\ & + \frac{q}{\pi} \sum_{k=1}^{\infty} \sin\Big(\frac{2\pi k}{q}\mu_v\Big) \mu_v \boldsymbol{W}_{v_0}\Big(\frac{2\pi k}{q}\Big) \frac{(-1)^{k}}{k}, \end{aligned} \tag{22}$$

$$\mathrm{E}(e_q^2) = \frac{q^2}{12} + \frac{q^2}{\pi^2} \sum_{k\neq 0}^{\infty} \cos\Big(\frac{2\pi k}{q}\mu_v\Big) \boldsymbol{W}_{v_0}\Big(\frac{2\pi k}{q}\Big) \frac{(-1)^{k}}{k^2}. \tag{23}$$

Step 4. Deduction of $W_v(\alpha)$. For jitter estimation, according to (17), we have

$$\mu_v = \mathrm{E}(v) = \mathrm{E}(t_m - w + 0.5 - \lfloor \mu_m \rfloor) = \mu_m - \lfloor \mu_m \rfloor = f_{\mu_m}. \tag{24}$$

Then when only considering the thermal noise, there is $t_m \sim \boldsymbol{N}(\mu_m, (\sigma_m)^2)$ and $w \sim \boldsymbol{U}(0,1)$. Their characteristic functions respectively are

$$\boldsymbol{W}_{t_m}(\alpha) = e^{j\alpha\mu_m} e^{-((\sigma_m)^2\alpha^2/2)}, \boldsymbol{W}_w(\alpha) = e^{j\alpha/2}\sin(\alpha/2)/(\alpha/2). \tag{25}$$

According to (17) and (25), we have

$$\boldsymbol{W}_v(\alpha) = \frac{2\sin(\alpha/2)}{\alpha} e^{-((\sigma_m)^2\alpha^2/2)} \cdot e^{j\alpha f_{\mu_m}}, \tag{26}$$

$$\boldsymbol{W}_{v_0}(\alpha) = e^{-j\alpha\mu_v}\boldsymbol{W}_v(\alpha) = \frac{2\sin(\alpha/2)}{\alpha} e^{-((\sigma_m)^2\alpha^2/2)} \tag{27}$$

and

$$\dot{\boldsymbol{W}}_{v_0}(\alpha) = \Big(\frac{\cos(\alpha/2)}{\alpha} - \frac{2\sin(\alpha/2)}{\alpha^2} - \frac{2\sin(\alpha/2)}{\alpha}(\sigma_m)^2\alpha\Big) e^{-((\sigma_m)^2\alpha^2/2)}. \tag{28}$$

Step 5. Formal Expression of e_a. $\boldsymbol{W}_{v_0}(\alpha)$ and $\dot{\boldsymbol{W}}_{v_0}(\alpha)$ in Step 4 will go to zero quickly when $|\alpha| > \frac{2\pi}{q}$ because of their exponent parts [5]. For example, when $q = 1$, considering the cases of $\alpha = 2\pi$ and $\alpha = 4\pi$, we have

$$e^{-((\sigma_m)^2(4\pi)^2)/2} < 10^{-25} \cdot e^{-((\sigma_m)^2(2\pi)^2)/2}. \tag{29}$$

So we just consider the terms with $k = \pm 1$ in the sums of (22), (23). By setting $q = 1$ and combining with (18), (19), (22), (23), (27), (28), we can gain the formal expression of e_a :

$$e_a \approx -\frac{1}{\pi^2}\cos(2\pi f_{\mu_m}) \cdot e^{-2\pi^2\sigma_m^2}. \tag{30}$$

e_a will reach to its maximum when $f_{\mu_m} = 0.5$:

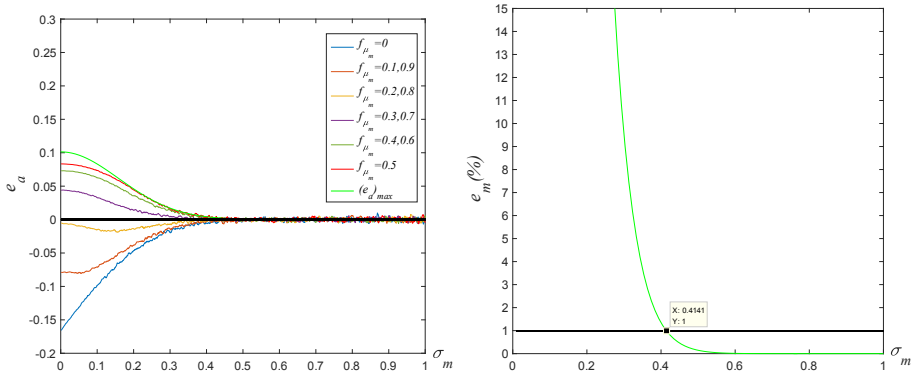
$$(e_a)_{max} \approx \frac{1}{\pi^2} e^{-2\pi^2\sigma_m^2}. \tag{31}$$

Fig. 5(a) shows the comparison between $(e_a)_{max}$ and the evaluation results of e_a got from the Matlab simulation in Fig. 4(a). Obviously, $(e_a)_{max}$ is a reasonable upper bound of e_a.

Step 6. Upper bound of e_m. According to the above theoretical analysis, upper bound of e_m in our jitter estimation method can be formally expressed by:

$$(e_m)_{max} = \frac{1}{2}(e_r)_{max} = \frac{1}{2} \cdot \frac{|(e_a)_{max}|}{\sigma_m^2} \approx \frac{1}{2\pi^2\sigma_m^2} e^{-2\pi^2\sigma_m^2}. \tag{32}$$

As we present in Fig. 5(b), the theoretical error bound is lower than 1% as long as $\sigma_m > 0.4141$. This is in accord with the Matlab simulation results shown in Fig. 4(b). In theory, the low error level of our method has been confirmed.



(a) Theoretical absolute error e_a (b) Theoretical error level e_m

Fig. 5. Theoretical error analysis of our method

5 Jitter Estimation on FPGA Device

In this section, we conduct a whole jitter estimation on a practical FPGA device. We adopt our method to estimate the total jitter of a RO-based TRNG and combine with the jitter separation approach in [3] to gain the part of the thermal jitter in the total jitter.

The oscillator is implemented on an Altera Cyclone IV FPGA. It is composed of 3 inverters and has about 305MHz frequency. Firstly, we use our method to estimate the total accumulated jitter (σ_m) in different measuring intervals (T_ms), and then the results are quadratically fitted by $\sigma_m^2 = aT_m^2 + bT_m$. According to [3], the first-order term (bT_m) is the part contributed by the thermal jitter.

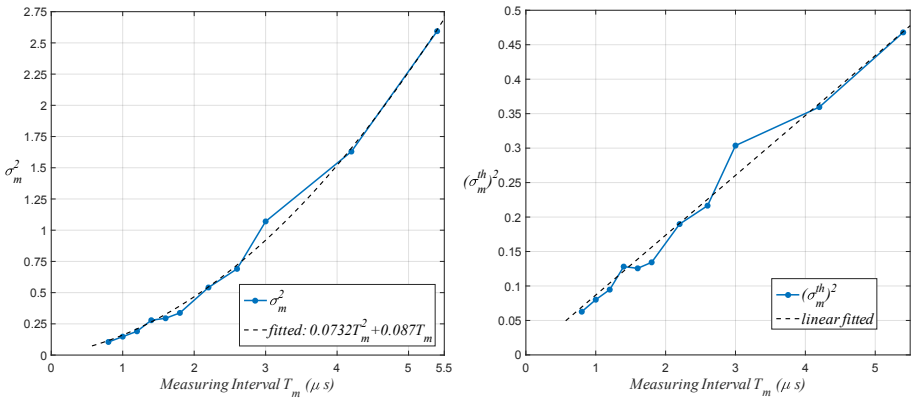
Specifically, we use a counter to count the edges of the oscillatory signal in multiple measuring intervals (T_m =0.8μs, 1.0μs, 1.2μs, 1.4μs, 1.6μs, 1.8μs, 2.2μs, 2.6μs ,3.0μs, 4.2μs, 5.4μs). For each measuring interval, we calculate Var(X) from the edge-counting results Xs and estimate the corresponding σ_m^2 by Equation (13). Then T_m and σ_m^2 is fitted by $\sigma_m^2 = 0.0732T_m^2 + 0.087T_m$, shown in Fig. 6(a). For a chosen measuring interval $T_m(\mu s)$, the ratio of the thermal jitter in the total jitter will be

$$r_{th} = \sqrt{\frac{0.087T_m}{0.0732T_m^2 + 0.087T_m}} = \sqrt{\frac{0.087}{0.087 + 0.0732T_m}}, \tag{33}$$

and the thermal jitter can be estimated by

$$\sigma_m^{th} = r_{th}\sigma_m. \tag{34}$$

We show the estimated results of $(\sigma_m^{th})^2$ in different measuring intervals by Fig. 6(b). It can be seen that the thermal jitter $(\sigma_m^{th})^2$ increases at a near-linear trend with the growth of the measuring interval. This is consistent with the fact that thermal noise is a kind of uncorrelated noise.



(a) Total jitter (b) Thermal jitter

Fig. 6. Experiment results of our jitter estimation on FPGA

For some other applications such as online health test of the entropy source, jitter estimation method on the chip should always estimate the thermal jitter in a fixed time interval. In this situation, the above multi-intervals estimation and fitting work can be regarded as a pre-calculation before implementing the online health test. Based on the pre-calculation, a ratio of the thermal jitter will be obtained and set in the implementation of the online test. During the execution phase, the online test just need to estimate the total jitter in the fixed measuring interval with our method and then extract the thermal part from the total jitter according to the ratio. For example, if the measuring interval is set fixed as $1.2\mu s$, then the ratio of the thermal jitter pre-calculated from (33) is $r_{th} = 0.706$. σ_m is the real-time total jitter estimated by our method on the chip. Then the thermal jitter can be simply calculated by $\sigma_m^{th} = 0.706\sigma_m$.

We provide the circuit module diagram of our method for on-chip implementation in Fig. 7. The sampling signal is processed by a frequency divider to generate the signal S_m which contains a series of measuring interval T_ms. Then the circuit conducts edge counting and calculates the total accumulated jitter σ_m. After multiplying σ_m by the ratio r_{th}, the circuit finally outputs the thermal jitter σ_m^{th}.

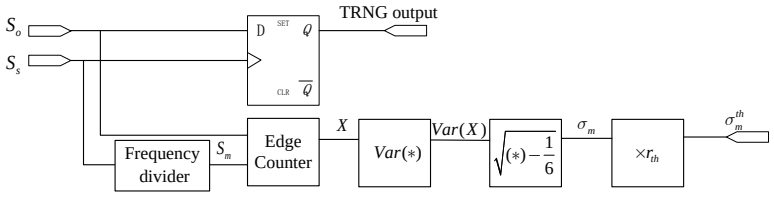


Fig. 7. Circuit module diagram for jitter estimation [6]

6 Discussion and Conclusion

We compare different jitter estimation methods in Table 1. The error levels of Ma's [7] and our method are gained from our analysis. The error level of Fischer's method was evaluated from their simulation results [2]. Note that the error levels presented in this table are given in the same condition that the flicker noise is not taken into account, but can still reflect the accuracy of different methods. In all of the methods, ours can achieve the lowest error level (1%), which is confirmed by theoretical analysis. For the methods in [7] and [2], there was no theoretical error analysis provided. Besides, compared to the method in [7], our method has reduced the requirement for the jitter σ_m, which can shorten the measuring time interval and speed up the estimation process. Taking this advantage, when our method is applied for online health test, the test can quickly assess the state of the entropy source.

Table 1. Comparisons of different methods

Methods	Error level	theoretically confirmed	Requirement for σ_m
Ma's [7]	10%	no	0.92
Fischer's [2]	5%	no	undefined
Our method	1%	yes	0.4141

In conclusion, we propose a high-accurate method to estimate the jitter of RO-based TRNGs. The error level of our method can reach to 1%, which is much lower than previous jitter estimation methods. For the first time, we give a theoretical error bound for our method, and the bound confirms the low error level. Additional advantage of our method is that it requires shorter measuring time interval, which can speed up the process of jitter estimation. Our method is to estimate the total jitter in RO-based TRNGs. When combined with the jitter separation approach in [3], our method is able to be used to estimate the thermal jitter on practical logic devices, as we presented by an experiment on FPGA. Consequently, our method will significantly help to precisely and efficiently evaluate the entropy of RO-based TRNGs.

Acknowledgments. This work is supported by the Nation Key R&D Program of China (2018YFB0904900, 2018YFB0904901) and China's National Cryptography Development Fund (No.MMJJ20170214, No.MMJJ20170211).

[6] The symbol $*$ represents the input of the module

References

1. Mathieu Baudet, David Lubicz, Julien Micolod, and André Tassiaux. On the Security of Oscillator-Based Random Number Generators. *J. Cryptology*, 24(2):398–425, 2011.
2. Viktor Fischer and David Lubicz. Embedded Evaluation of Randomness in Oscillator Based Elementary TRNG. In *Cryptographic Hardware and Embedded Systems - CHES 2014 - 16th International Workshop, Busan, South Korea, September 23-26, 2014. Proceedings*, pages 527–543, 2014.
3. Patrick Haddad, Yannick Teglia, Florent Bernard, and Viktor Fischer. On the assumption of mutual independence of jitter realizations in P-TRNG stochastic models. In *Design, Automation & Test in Europe Conference & Exhibition, DATE 2014, Dresden, Germany, March 24-28, 2014*, pages 1–6, 2014.
4. Wolfgang Killmann and Werner Schindler. A Design for a Physical RNG with Robust Entropy Estimators. In *Cryptographic Hardware and Embedded Systems - CHES 2008, 10th International Workshop, Washington, D.C., USA, August 10-13, 2008. Proceedings*, pages 146–163, 2008.
5. Istvan Kollar. Bias of Mean Value and Mean Square Value Measurements Based on Quantized Data. *IEEE Transactions on Instrumentation and Measurement*, 43(5):733–739, Oct 1994.
6. Kent H. Lundberg. Noise Sources in Bulk CMOS. `http://www.mit.edu/people/klund/papers/UNP_noise.pdf`, 2002.
7. Yuan Ma, Jingqiang Lin, Tianyu Chen, Changwei Xu, Zongbin Liu, and Jiwu Jing. Entropy Evaluation for Oscillator-Based True Random Number Generators. In *Cryptographic Hardware and Embedded Systems - CHES 2014 - 16th International Workshop, Busan, South Korea, September 23-26, 2014. Proceedings*, pages 544–561, 2014.
8. Yuan Ma, Jingqiang Lin, and Jiwu Jing. On the Entropy of Oscillator-Based True Random Number Generators. In *Topics in Cryptology - CT-RSA 2017 - The Cryptographers' Track at the RSA Conference 2017, San Francisco, CA, USA, February 14-17, 2017, Proceedings*, pages 165–180, 2017.
9. George Marsaglia. The Marsaglia random number CDROM including the DIEHARD battery of tests of randomness. *Diehard Tests*, 1995.
10. Andrew Rukhin, Juan Soto, James Nechvatal, Miles Smid, Elaine Barker, Stefan Leigh, Mark Levenson, Mark Vangel, David Banks, Alan Heckert, James Dray, and San Vo. NIST SP800-22: A Statistical Test Suite for Random and Pseudorandom Number Generators for Cryptographic Applications . `http://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-22r1a.pdf`.
11. William Fleetwood Sheppard. On the Calculation of the most Probable Values of Frequency-Constants for Data arranged according to Equidistant Division of a Scale. *Proceedings of the London Mathematical Society*, 29(1):353–380, 1897.
12. Anekal B. Sripad and Donald L. Snyder. A Necessary and Sufficient Condition for Quantization Errors to be Uniform and White. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 25(5):442–448, Oct 1977.
13. 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.
14. Boyan Valtchanov, Alain Aubert, Florent Bernard, and Viktor Fischer. Modeling and observing the jitter in ring oscillators implemented in FPGAs. In *Proceedings of the 11th IEEE Workshop on Design & Diagnostics of Electronic Circuits &*

Systems (DDECS 2008), Bratislava, Slovakia, April 16-18, 2008, pages 158–163, 2008.