# Entropy Evaluation for Oscillator-Based True Random Number Generators

Yuan Ma*, Jingqiang Lin**, Tianyu Chen, Changwei Xu,
Zongbin Liu, and Jiwu Jing

Data Assurance and Communication Security Research Center,
Chinese Academy of Sciences, Beijing, China
State Key Laboratory of Information Security,
Institute of Information Engineering, Chinese Academy of Sciences, Beijing, China
{yma,linjq,tychen,xuchangwei,zbliu,jing}@is.ac.cn

**Abstract.** True random number generators (TRNGs) are crucial to the implementations of cryptographic algorithms and protocols. The quality of randomness directly influences the security of cryptographic systems. Oscillator-based sampling is popular in the design of TRNGs due to its nice properties of elegant structure and high speed. However, the credibility of randomness generated from high-speed oscillator-based TRNGs, especially ring oscillator-based (RO-based) ones, is still in controversy. This is mainly because pseudo-randomness is hardly distinguished from true randomness and RO-based TRNGs are susceptible to external perturbations. In this paper, we present a stochastic model to evaluate the entropy of oscillator-based TRNGs, and then deduce the requirement of design parameters (including the sampling interval) for sufficient entropy per random bit, i.e., to ensure true randomness. Furthermore, we design a jitter measuring circuit to verify the theory, and the theoretical results are confirmed by both the simulation and practical experiments. Finally, we apply the stochastic model to analyze the effect of deterministic perturbations, and demonstrate that the randomness of RO-based TRNGs (under deterministic perturbations) can be overestimated and predicting the "random" bits could be possible.

**Keywords:** True random number generators, ring oscillators, sufficient entropy, perturbation, stochastic model.

## 1 Introduction

True random number generators are employed in many cryptographic applications such as key generation, digital signature and key exchange, and their

* The authors were partially supported by the National 973 Program of China under award No. 2013CB338001. Yuan Ma, Jingqiang Lin, Tianyu Chen and Jiwu Jing were also partially supported by the National 973 Program of China under award No. 2014CB340603.

** Corresponding author.

L. Batina and M. Robshaw (Eds.): CHES 2014, LNCS 8731, pp. 544–561, 2014.

security is crucial for cryptographic systems. The oscillator-based TRNG has been widely employed due to its nice properties of elegant structure and high speed. In oscillator-based TRNGs, a fast oscillator signal is sampled by a slow one which is generated by another oscillator or an external crystal oscillator, and the timing jitter in the signals is the entropy (randomness) source.

Randomness evaluation is important for both the design and the use of TRNGs. In general, there are two methods for randomness evaluation: black-box statistical tests and white-box stochastic models. The existing statistical tests, such as FIPS 140-2 [11], NIST 800-22 [16] and Diehard [14] measure the balance and independence of random bits through various test items. However, passing these statistical tests can only be considered as a necessary condition for true randomness (as deterministic sequences with good statistical properties can also pass these tests). Therefore, it seems extremely difficult to test the true randomness only from the outputting sequences of TRNGs. For this reason, it is necessary to evaluate TRNGs from stochastic models, which are directly related to the entropy of TRNGs.

In addition, from the white-box stochastic models, it is feasible to derive the requirements for the design parameters of TRNGs. In oscillator-based TRNGs, one of the most important parameters is the sampling interval, which determines the generation speed of TRNGs. To model oscillator-based TRNGs, Killmann and Schindler [12] used a common stochastic model, where the flipping times are independent and identically distributed (i.i.d.), and provided a tight lower bound for the entropy of the TRNG. Yet, the model is not able to provide a precise entropy, or the probabilities of outputting certain bit patterns. Using a phase-oriented approach, Baudet et al. [2] provided a more comprehensive model and calculated the precise entropy for RO-based TRNGs. The model also allowed for computing the maximal bias on a short vector and recovering the main stochastic parameters of a TRNG. Amaki et al. [1] proposed a stochastic behavior model using Markov state transition matrix to calculate the state probability vector. Some other related works for TRNG modeling are presented in [15,5,3].

Another issue for modeling the stochastic behavior of RO-based TRNGs is deterministic perturbations. In general, the perturbations can be generated from an unstable switching power, or another oscillator inside the chip. They can even be injected by attackers [13]. The effect of deterministic perturbations has been discussed in the literature. The process of injecting deterministic perturbations is simulated in [4], and the authors observe that the engagement of perturbations makes it easier to pass statistical tests due to the joining of pseudo-randomness. The improvement of statistical properties was also investigated by the theory and the experiment in [1]. Baudet et al. [2] presented a differential measurement method to acquire non-deterministic jitter, and concluded that the deterministic perturbations do not undermine the randomness of a TRNG by itself, but can lead to a dangerous overestimation of randomness jitter. In addition, the effect of deterministic perturbations on the inherent randomness was discussed in [13,7].

In this paper, by improving the stochastic model in [12], we propose a more precise and comprehensive stochastic model for evaluating the entropy of oscillator-based (more precisely, RO-based) TRNGs, and theoretically give the required parameters for sufficient entropy per bit. In order to verify the theory, we design a novel jitter measuring circuit by employing an internal measuring method. The theoretical results are verified with both simulation and practical experiments. Meanwhile, the consistencies with the previous models are also investigated. Furthermore, we apply the model to analyze and explain the effect of deterministic perturbations. We demonstrate that the randomness of RO-based TRNGs under deterministic perturbations can be overestimated, and it could be possible to predict the "random" bits.

In summary, we make the following contributions.

- We propose a new modeling method for stochastic behaviors to evaluate the entropy of oscillator-based TRNGs, and deduce recommended design parameters for sufficient entropy.
- We design a novel jitter measuring circuit by employing an internal measuring method to verify the theory, which is crucial and helpful in acquiring the design parameters of the TRNGs.
- We perform a comprehensive study on the effect of deterministic perturbations, and point out that deterministic perturbations make it possible to predict the generated random sequences, though the sequences under the effect are easier to pass statistical tests.

The rest of the paper is organized as follows. In Section 2, we present the stochastic model for oscillator-based TRNGs. In order to verify the theory, we design a novel jitter measuring circuit for experimental verification, and discuss the modeling assumption in Section 3. In Section 4, we verify the theoretical results and give the requirement of parameters. We analyze the effect of deterministic perturbations in Section 5. In Section 6, we conclude the paper.

## 2 Stochastic Model

A typical example of oscillator-based TRNG is shown in Figure 1. A stable slow clock signal samples an unstable fast oscillator signal to generate random bits. As the sampling interval increases, the jitter of the fast oscillator signal are accumulating. The foundation of generating random bits is the unpredictability of the number of fast signal periods (more precisely, half-periods) in the duration of a single slow signal period.

**Definitions.** The important notations in oscillator-based TRNGs are shown in Figure 2, where the half-periods $X_k$ is the time interval between two flopping times. In this paper, we assume that $X_k$ are i.i.d., and the reason is discussed in Section 3.4. The mean and variance of half-periods are denoted as $\mu$ and $\sigma^2$, respectively, i.e. $\mu = E(X_k)$ and $\sigma^2 = \mathrm{Var}(X_k)$. The sampling time with the equal interval $s$ are represented as $s_0, s_1, ..., s_i$, i.e. $s_i = is$. The waiting time $W_i$ denotes the timing distance of $s_i$ to the following closest edge.

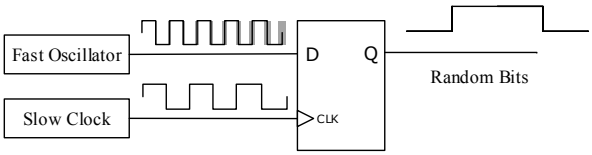



**Fig. 1.** Oscillator-based TRNG

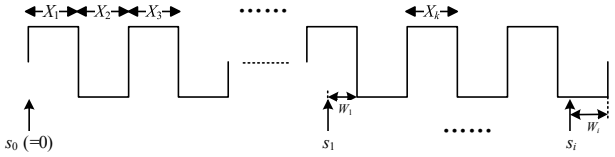



**Fig. 2.** Definitions of oscillator-based TRNGs

The number of edges within $(s_{i-1}, s_i]$ is denoted by $R_i$, then the $i$th sampling bit $B_i$ is represented as $B_i = (B_{i-1} + R_i) \bmod 2$.

Note that the operation of adding $R_i$ with $B_{i-1}$ can be treated as a type of post-processing, which is not considered in this paper; the operation causes no impact on the information entropy, thus we take $B_i = R_i \bmod 2$ in the remainder for convenience.

### 2.1 Preliminary Analysis of the Stochastic Model

We briefly summarize some important results from [12] on probability calculation of sampling bits, which is the base of our work.

Let $R_i = \min\{k \mid T_k > s\}$, where $T_k = X_1 + X_2 + ... + X_k$, meaning $R_i$ is the first increasing $k$ ensuring that $T_k$ is larger than $s$. The probability

$$\text{Prob}(R_i = k+1) = \text{Prob}(T_k \le s) - \text{Prob}(T_{k+1} \le s). \tag{1}$$

The distribution of $T_k$ is derived from the central-limit theorem (CLT), so it is deduced that

$$\text{Prob}(\frac{T_k - k\mu}{\sigma\sqrt{k}} \le x) \to \Phi(x), k \to \infty, \tag{2}$$

where $\Phi(x) = \int_{-\infty}^{x} e^{-t^2/2} dt/\sqrt{2\pi}$ denotes the cumulative distribution function of the standard normal distribution $N(0,1)$. Then we have

$$\begin{aligned} \text{Prob}(R_i = k+1) &= \text{Prob}(T_k \le s) - \text{Prob}(T_{k+1} \le s) \\ &\approx \Phi((v-k) \cdot \frac{\mu}{\sigma\sqrt{k}}) - \Phi((v-k-1) \cdot \frac{\mu}{\sigma\sqrt{k+1}}), \end{aligned} \tag{3}$$

where $v = s/\mu$ represents the frequency ratio. Then the probability distribution of sampling bit $B_i$ is

$$\begin{aligned} \text{Prob}(B_i = b_i) &= \text{Prob}(R_i \bmod 2 = b_i) \\ &= \sum_{j=1}^{\infty} \text{Prob}(R_i = 2j - b_i) \ for \ b_i \in \{0, 1\}. \end{aligned} \tag{4}$$

### 2.2 Improved Model for RO-Based TRNGs

In oscillator-based TRNGs, especially in RO-based TRNGs, the amount of jitter is very small [6], i.e., $\sigma/\mu \ll 1$. The possible values of $k$ are restricted in a small interval zone near the mean $v$. In addition, as the fast oscillator signal is dozens of times faster than the slow clock, $v$ is not a small value. Therefore, it is reasonable to assume that $\sqrt{k} \approx \sqrt{k+1} \approx \sqrt{v}$.

Setting $q = \sigma\sqrt{v}/\mu$ as the quality factor which is used to evaluate the quality of TRNGs, we have

$$\begin{aligned} \text{Prob}(R_i = k+1) &\approx \Phi((v-k) \cdot \frac{\mu}{\sigma\sqrt{k}}) - \Phi((v-k-1) \cdot \frac{\mu}{\sigma\sqrt{k+1}}) \\ &\approx \Phi(\frac{v-k}{q}) - \Phi(\frac{v-k-1}{q}). \end{aligned} \tag{5}$$

For the probability of $B_i = 1$, we have

$$\text{Prob}(B_i = 1) = \sum_{j=1}^{\infty} \text{Prob}(R_i = 2j-1) \approx \sum_{j=1}^{\infty} (\Phi(\frac{v-2j}{q}) - \Phi(\frac{v-2j-1}{q})),$$

which can be described as the sum of the interleaved column areas below the normal distribution curve in Figure 3.

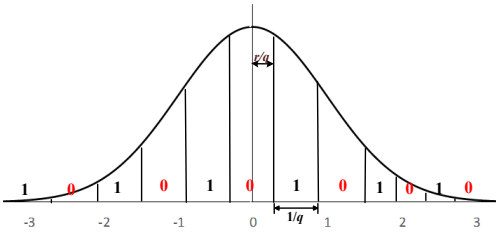



**Fig. 3.** The probability distribution ofthe sampling bit ($W_i = 0$)

In Figure 3, $W_i$ is set to 0 for convenience. The area between the normal distribution curve and $x$ axis (equaling to 1) is divided at $1/q$ interval, and the area of each column corresponds to the probability of $R_i$ equaling to each $k$. The larger $q$ is, the finer the column is divided, which means that the areas of

'0' and '1' are closer. Another observation is that, besides $q$, the value of $r$ also affects the bias of the sampling bit. Variable $r$ is the fractional part of $v$, i.e., $r = v \mod 1$. The dividing position is determined by $r/q$, as shown in Figure 3. Obviously, when $W_i = 0$ and $r = 0$, the areas of probabilities '0' and '1' are equal regardless of $q$. The most unbiased case is $r = 0.5$ when $W_i = 0$, where the distance between probabilities '0' and '1' becomes largest compared to the other cases with the same $q$. Therefore, a robust TRNG design should have sufficient entropy even in the worst (most unbiased) case.

**The Probability Distribution of the Waiting Time.** In consecutive sampling, two adjacent sampling processes are dependent as the waiting time $W_i$ generated by the $i$th sampling affects the $(i + 1)$th one. Referring to renewal theory, the probability of $W_i$ is

$$P_W(y) = \text{Prob}(W_i \leq y) = \frac{1}{\mu} \int_0^y (1 - P_X(u))du, \tag{6}$$

where $P_X(\cdot)$ denotes the cumulative distribution function of half-periods $X_i$. Furthermore, because $\sigma \ll \mu$, $P_W(y)$ is approximated to

$$P_W(y) \approx \begin{cases} \frac{1}{\mu} \int_0^y 1du = \frac{y}{\mu}, & 0 \leq y \leq \mu; \\ 1, & y > \mu \end{cases} \tag{7}$$

which can be treated as the uniform distribution on the interval $[0, \mu]$.

**Sampling Process Approximation.** Inspired by Equations (5) and (7), we approximate the consecutive sampling described in Figure 1 to the following process - a slow signal with jitter sampling a fast stable signal.

- The fast oscillator signal is stable.
- The slow oscillator signal which sampling the fast signal is unstable with jitter. The periods follow $(v\mu, v\sigma^2)$ normal distribution.

Easy to verify that the probability distributions for $R_i$ and $W_i$ under the model are corresponding with Equations (5) and (7), respectively. Therefore, the approximated model is equivalent to the original one under the assumption of small jitter. In fact, the approximated process is also a common type in oscillator-based TRNGs. The stochastic behavior of the approximated process is easier to model, so we use it as an improved model to calculate and evaluate the entropy of TRNGs.

### 2.3 Entropy Calculation

The improved model for consecutive sampling is described in Figure 4. For normalization, we define $W_i'$ as the ratio of the $W_i$ to the mean $\mu$. We calculate the probability of $B_{i+1} = b_{i+1}$ under the condition of $W_i' = w_i'$,

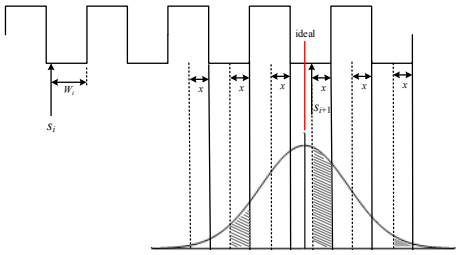



**Fig. 4.** The new model for entropy calculation

$$\begin{aligned}\text{Prob}(b_{i+1}|w'_i) = &\sum_{i=-\infty}^{+\infty}(\Phi(\frac{2i+1-c_i}{q}) - \Phi(\frac{2i-c_i}{q})) \\ &:= J_{i+1}(w'_i) \\ &(c_i = (v - w'_i - (1-b_{i+1})) \bmod 2).\end{aligned} \tag{8}$$

From Figure 4, we have

$$\begin{aligned}\text{Prob}(W'_{i+1} \le x, b_{i+1}|w'_i) = &\sum_{i=-\infty}^{+\infty}(\Phi(\frac{2i+1-c_i}{q}) - \Phi(\frac{2i-c_i+1-x}{q})) \\ &:= F_{i+1}(x, w'_i),\end{aligned}$$

which is the area of the shaded part in Figure 4.

By defining $G_i(x) := \text{Prob}(W'_i \le x|b_i, \ldots, b_1)$, we have the conditional probability of sampling bits

$$\text{Prob}(b_{i+1}|b_i, \ldots, b_1) = \int_0^1 J_{i+1}(x)G_i(dx) := K(b_{i+1}). \tag{9}$$

Due to the uncertainty of the initial sampling position, we assume the distribution of $W'_0$ is also uniformed in $(0, 1)$. Therefore,

$$G_1(x) = \text{Prob}(W'_1 \le x|b_1) = \int_0^1 \text{Prob}(W'_1 \le x|b_1, w'_0)dw'_0 = \int_0^1 \frac{F_1(x, w'_0)}{J_1(w'_0)}dw'_0.$$

Then, using the property of the Markov process

$$\text{Prob}(b_{i+1}|w'_i, b_i, w'_{i-1}, ...) = \text{Prob}(b_{i+1}|w'_i),$$

we calculate the following $G_i(x)$:

$$G_i(x) = \int_0^1 \frac{F_i(x, w'_{i-1})}{K(b_i)}G_{i-1}(dy).$$

Then we get the $n$-bit probability distribution for certain bit patterns

$$p(\boldsymbol{b}) = \text{Prob}(b_n, \ldots, b_1) = \prod_{i=1}^{n} K(b_i), \tag{10}$$

and the $n$-bit entropy

$$H_n = \sum_{\boldsymbol{b} \in \{0,1\}^n} -p(\boldsymbol{b}) \log p(\boldsymbol{b}). \tag{11}$$

## 3 Experiment Design for Model Verification

In this section, using an internal measuring method we design an improved jitter measurement circuit to verify the stochastic model. The advantage of the circuit is that it is able to acquire the approximated quality factor while the sampling bits are generated, which is useful to verify the stochastic model.

### 3.1 Dual-Counter Measurement Circuit

The ring oscillator is formed by a set of inverters that are chained into a ring, while the number of the inverters must be an odd number. A typical RO structure in FPGAs is shown in Figure 5, where these inverters are implemented by Look-Up Tables (LUTs) in FPGAs. The ideal period of the oscillator signal is represented as $2X$, where $X$ is the delay of all the RO components, i.e., the half-period.

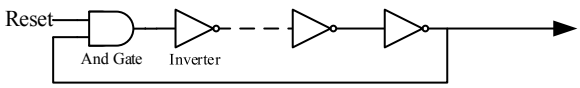



**Fig. 5.** Ring oscillator

In order to measure the jitter more accurately, we improve the internal measurement circuit [18]. In contrast to the only one positive or negative edge counter used in [18], two voltage-crossing counters are utilized in our measurement method, as shown in Figure 6. Besides improving the sensitivity to jitter accumulation, this method helps us directly obtain the sampling bits from the counting results. The counting process is the (delayed) renewal process, so the variance with the interval of $s$ is represented as $s(\sigma^2/\mu^3) + o(s) = q^2 + o(s)$, where $o(s) \to 0$ when $s \to \infty$. Therefore, by calculating the standard variance of the counting results, we can acquire the approximated quality factor $q$. It should be noticed that, when the interval is not large enough, $q$ is overestimated, since $o(s)$ cannot be ignored under the interval.

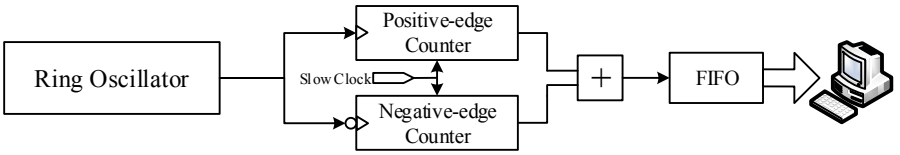



**Fig. 6.** Dual-counter measurement circuit

In the improved measuring method, two counters are employed to measure the number of positive edges and negative edges in the duration of a single slow clock period, respectively. Then, the two counter results are added to form the outputting values. After each count finishes, the counters should be cleared to start the next count. The clear signal is generated through the clear circuit which is driven by both the ring oscillator signal and the slow clock. The counting process of the positive-edge counter with the sampling interval of $s$ is depicted in Figure 7. Between the two adjacent counts, the clear signal lasts accurately one period of the oscillator signal by using the clear circuit. If the oscillator frequency is too high to clear the counters within one cycle, the number becomes two or three.

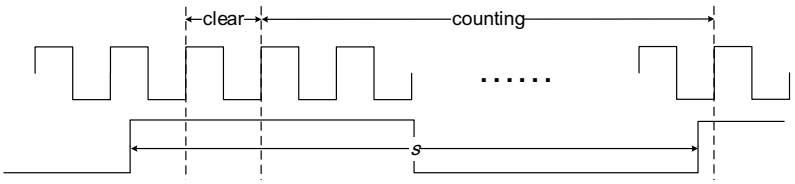


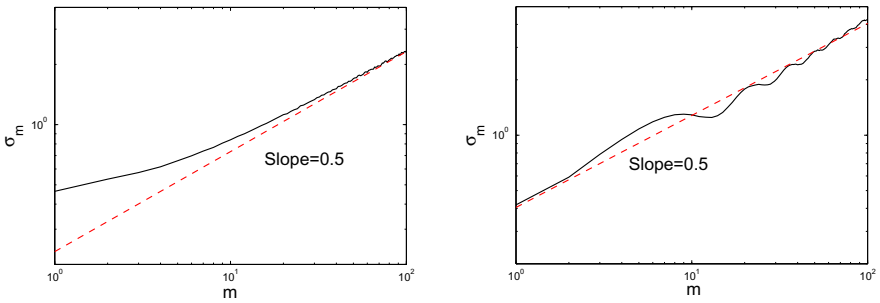

**Fig. 7.** The counting process (positive edge)

Consecutive sampling is adopted in the measurement, and the sampling type is useful to simplify the counting process, because we just need to do the counting collection only once for the longer sampling intervals of $ms$, rather than do $m$ times. After getting numbers of count results in the duration of $s$, we can sum the $m$ non-overlapping results to obtain the number of edges in the duration of $ms$, then we can figure out the quality factor under the interval of $ms$ by calculating the standard variance of these sums. Although the clear mechanism makes all sums smaller than the real values by $m - 1$, it has no impact on calculating the variances of these values.

### 3.2 Jitter Measurement

We implement the circuit with 3-inverter RO on Xilinx Virtex-5 FPGA. The RO frequency is about 484 MHz, and the slow clock is a 5 MHz crystal oscillator signal, and the circuit output is the number of RO edges within the duration of

$s = 200\ ns$. Having numbers of outputting values in the interval $s$, we can figure out the number of edges within the sampling interval $ms$ by $m$-time accumulating. For the sampling interval $ms$, we can calculate the standard deviation $\sigma_{ms}$ of the accumulation results. From the renewal theory under i.i.d. assumption, $\sigma_m = \sqrt{ms}(\sigma/\mu^{3/2}) = \sqrt{m}\sigma_1,\ s \to \infty$, where $\sigma_m$ denotes the standard variance under the interval of $ms$.




(a) Simulation results with white noises (b) Practical measuring results in FPGA

**Fig. 8.** The measuring results with ideal vs. practical noises

The simulation and practical results for the measurement method at logarithmic coordinates are shown in Figure 8, whose $x$-axis is $m$ and $y$-axis is standard deviation $\sigma_m$. In Figure 8(a), with $m$ increasing, the slope of the standard deviation curve is approaching to 0.5, which is consistent with the theory. As mentioned, if $ms$ is not large enough, meaning the accumulated jitter is small, the measuring result is larger than the real value. Fortunately, we observe that the overhead will be no more than 10% when the measuring standard deviation is larger than 0.8, so these results are available.

Surprisingly, the practical measuring result is quite different, as shown in Figure 8(b). We find the existence of deterministic (sinusoidal) perturbations which make the $\sigma_m$ curve form a wavy pattern of rising. In addition, when the sampling interval $ms$ is large (about $m > 50$), we also observe the existence of correlated noise, under which the standard variance increases faster and the slope becomes larger than 0.5.

### 3.3 Filtering Deterministic Jitter

Deterministic perturbations make an overhead for the estimation of random jitter. In order to filter deterministic jitter, a measurement method using dual oscillators was presented in [8]. The method is based on the fact that the effect of deterministic perturbations is global. We use a 15-inverter RO signal as the slow clock to filter the perturbations and measure the random jitter of fast oscillator signal. In contrast to the clock measuring result, the RO measuring result does

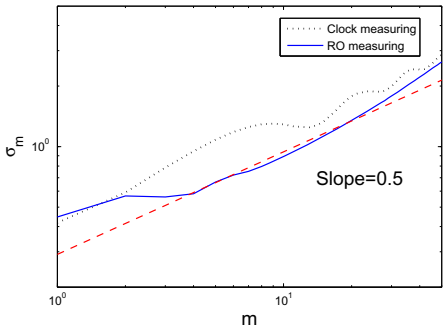



**Fig. 9.** RO measuring result

not display an obvious wavy pattern of rising, as shown in Figure 9. Therefore, we obtain the data $R_i$ without the perturbations, which are the experimental data base for verifying the theory.

### 3.4 Discussion for Modeling Assumption

In our stochastic model, we assume that the jitter or the noises are i.i.d., but the correlation is observed in the experiment when the sampling interval is long. According to [10], correlated noise (such as $1/f$ noise) is embodied at low frequency in oscillators, while the noise at high frequency is white (or independent). The correlated noise was also observed in [19] which suggested that the sampling frequency should be fast enough to avoid the influence of correlated noise. In our proposed TRNG model, the focused sampling interval is $m < 12$ (see Section 4.2) where the accumulated jitter is insufficient or almost sufficient, so the effect of correlated noise is weak in this region. Therefore, for simplicity, we do not involve the modeling for correlated noises or jitter in the stochastic model of the TRNG.

Correlated noise makes the jitter and the counting results have long-term dependence, which also affects sampling bits, so it shall be noted that the effect of correlated noise (especially mixed with white noise) on sampling bits in RO-based TRNG is actually an open problem due to the complexity and variety of correlated noise. As a preliminary analysis, we do not observe the correlation inherited in the sampling bits under correlated noise when accumulated independent jitter is sufficient (see Figure 10).

## 4 Entropy Evaluation

In this section, using the formula of entropy calculation, we deduce the requirement of RO-based TRNGs parameters for sufficient entropy per bit. The results are verified by experiments, and the comparison with other work is also presented.

### 4.1 Parameters for Sufficient Entropy

In consecutive sampling, $H_n$ can be derived from Equation (11). The bit-rate entropy is denoted as $H = H_n/n$. According to the experimental result in [12], the threshold value of bit-rate entropy is chosen as 0.9999, i.e., $H$ should be larger than 0.9999 to achieve sufficient security. We calculate the bit-rate entropy in term of $q$ for various $r$ from 0 to 0.5 using Matlab numerical calculation (shown in Figure 11). The required $q$ values for different $r$ to achieve sufficient entropy (0.9999) are listed in the second row of Table 1.

In contrast to the example of $W_i = 0$ in Figure 3, the consecutive sampling has the worst balance at $r = 0$, because the waiting time $W_i$ has a uniform distribution in consecutive sampling. In the case of $r = 0$, when $q$ is larger than 0.9264, the bit-rate entropy is sufficient. On the contrary, the generator with $r = 0.5$ is easiest to acquire sufficient entropy, and the required $q$ is only 0.6511. Given the parameters $\sigma$ and $\mu$ of the fast oscillator signal , we can figure out the required sampling interval for sufficient entropy.

**Table 1.** The required $q$ to achieve sufficient entropy for different $r$

| Req. $q$ \ $r$ | $r$=0 | $r$=0.1 (0.9) | $r$=0.2 (0.8) | $r$=0.3 (0.7) | $r$=0.4 (0.6) | $r$=0.5 | Remark |
|---|---|---|---|---|---|---|---|
| Theory | 0.9264 | 0.9209 | 0.9029 | 0.8673 | 0.7895 | 0.6511 | $H > 0.9999$ |
| Sim. Measured | 0.9778 | 0.9392 | 0.9198 | 0.8759 | 0.7928 | 0.7002 | passing FIPS 140-2 |

### 4.2 Experimental Verification

In order to verify the parameter requirement, we use the statistical tests FIPS 140-2 [11] to test the sampling bits, including monobit test, poker test, runs test and longest run test. We record the required $q$ values for the sampling bits passing all items of FIPS 140-2, and compare them with the theoretical ones.

**Matlab Simulation.** We first use Matlab simulation to verify the theoretical results, as the environment can be ideal as expected. In the simulation, the half-periods of the fast oscillator signal are set to $(1.125, 0.017^2)$ i.i.d. normal distribution. Using the measuring method under a preset sampling interval, we can get the counting results, whose standard variance and LSBs can be treated as $q$ and sampling bits, respectively. With the sampling interval increasing, the passing point for each $r$ can be observed, as shown in the third row of Table 1. As we mentioned in Figure 8, the measured $q$ values are a little larger than the real values when $m$ is small. Therefore, the simulation results approximately match with the theory in Table 1, especially in the aspect of variation tendency. The difference between these two results is because that the criteria of the theoretical entropy and FIPS 140-2 are not completely consistent.

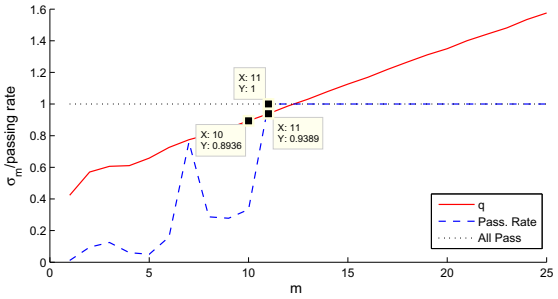



**Fig. 10.** Results of measured $q$ and FIPS 140-2 tests in FPGA

**Practical Experiment.** We also implement the measurement circuit in the FPGA platform. The measuring and test results are shown in Figure 10, where the passing rate means the ratio of the number of passed test items to the number of all items. We observe that the passing point lies in the interval $q \in [0.8936, 0.9389]$, which nearly corresponds with the simulation and theory. However, it seems infeasible to measure the right $r$ at this point to do a further verification, since a tiny measuring error will make the measured $r$ totally different in such a high frequency of the fast oscillator signal. In addition, it should also be noticed that correlated noise makes an overestimation for thermal jitter, especially when $m$ is large. One can employ the method presented in [9] to measure the thermal noise contribution to the jitter.

### 4.3 Comparison with Previous Work

For the entropy evaluation of oscillator-based TRNGs, a tight lower bound was provided in [12], and the bit-rate entropy was calculated in [2] by using a phase-oriented method. The main results of [12] and [2] are presented as Equations (12) and (13), respectively.

$$H(B_i|B_{i-1},\ldots,B_1) \geq H(B_i|W_{i-1}) \approx \int_0^s H(R^{(s-u)} \mod 2) P_W(du) \quad (12)$$

$$H_n \approx n - \frac{32(n-1)}{\pi^4 \ln(2)} \cos^2(\pi r) e^{-\pi^2 q^2} \quad (13)$$

In Equation (12), $R^{(s-u)}$ represents the number of crossing edges in the duration of $(s-u)$, and the variables in Equation (13) have been converted for the correspondence of definitions.[1] Our estimated bit-rate entropy is larger than the lower bound of [12] as expected, and is almost identical to the result of [2] at the worse cases ($r = 0,\ 0.1,\ 0.2$), as shown in Figure 11.

[1] The quality factor $Q$ defined in [2] equals to $q^2/4$.

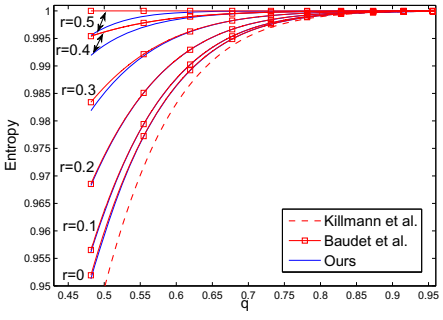



**Fig. 11.** Comparison result for entropy estimation

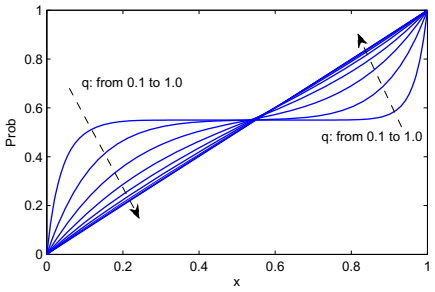



**Fig. 12.** $\text{Prob}(W_i \leq x|b_i)$ for different $q$ at $r = 0$

However, there are some inconsistencies in the comparison of our result with [2] when $r \geq 0.3$, especially at $r = 0.5$. According to Equation (13), $H_n$ approximately equals to $n$ when $r = 0.5$, meaning that the bit-rate entropy $H$ achieves the maximum value 1. That is to say, so long as the sampling interval $s$ satisfies that $(s \mod \mu)/\mu = r = 0.5$ in consecutive sampling, the bit-rate entropy is close to 1 regardless of $q$. Nonetheless, the conclusion is not confirmed in both our theory and simulation experiment. In our opinion, $r = 0.5$ can only guarantee the balance of sampling bits[2], rather than the independence. Therefore, when $r = 0.5$ the generated sequences can pass the statistical tests once the independence of sampling bits is satisfied. That is why the generators with $r = 0.5$ are easier to acquire sufficient entropy. Obviously, when $q$ is small, the correlation of sampling bits cannot be eliminated, thus the $n$-bit entropy cannot approximately equal to $n$. The sampling correlation is further illustrated via the following independence condition.

### 4.4 Independence Condition

The sampling correlation is derived from the transfer of the waiting time $W_i$ which affects the $(i+1)$th sampling result. Therefore, the independence of sampling bits should satisfy

$$\forall b_i \in \{0,1\}, \text{Prob}(W_i \leq x|b_i) = \text{Prob}(W_i \leq x) = \frac{x}{\mu}.$$

For various $q$ values at $r = 0$, the conditional probability distributions $\text{Prob}(W_i \leq x|b_i)$ are shown in Figure 12, where the curves from outside to inside correspond to the $q$ values from 0.1 to 1 at 0.1 interval. Note that $r$ does not make the conditional distribution become uniform easier, but only affects the cross position of these probability curves. Therefore, we only present the result of $r = 0$. When

[2] The balance holds only when $W_i$ is uniformly distributed, which just requires a very small $q$ (about 0.1).

$q$ is less than 0.5, the probability distribution is non-uniform, meaning that the correlation still exists. Until $q$ is approximately larger than 0.6, the distribution becomes uniform and the correlation is almost eliminated, which is consistent with the calculation results in Table 1. In addition, the experimental result in the next section also confirms the independence condition.

## 5 The Effect of Deterministic Perturbations

In this section, we show that the deterministic perturbations make the sampling bits appear to be more "random" and easier to pass statistical tests. More importantly, we point out that the seemingly random sequence actually has a vulnerability which makes it possible to predict the sequence.

### 5.1 The Effect on the Statistical Test

In order to analyze the effect on the statistical test, we carry out the measurement and FIPS 140-2 statistical tests with deterministic jitter. Under deterministic perturbations, the TRNG is easier to pass the test, as shown in Figure 13, where the passing position is $m = 9$ and the other is $m = 11$.

It is interesting that the passing rate of RO sampling has an abrupt rise at $m = 7$, which is precisely the position of the crest of the perturbations, meaning that the sampling sequence suddenly becomes more "random". The reason is that the deterministic jitter is not completely filtered out by the dual-oscillator method, since the perturbation effects on the two oscillators cannot be exactly identical, though they have been placed as close as possible. Moreover, the observation validates the fact that injecting deterministic jitter does improve the randomness of outputting sequences. However, note that the deterministic perturbation in our experiment is slight and balanced. Once the perturbation becomes strong, it will reduce the amount of inherent independent jitter; once it becomes biased, it will degrade the quality of sampling bits.

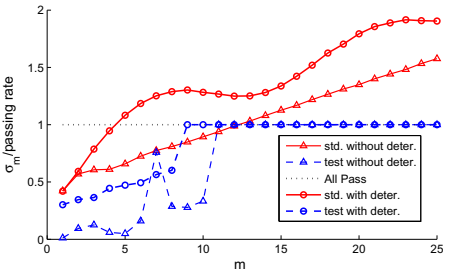



**Fig. 13.** Measuring results with and without deterministic perturbations

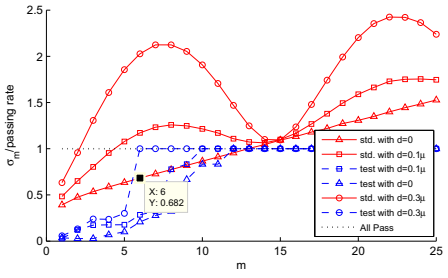



**Fig. 14.** Simulation results with varying deterministic jitter

### 5.2 The Bound for the Randomness Improvement

Increasing the amplitude makes it easier to pass the statistical tests. However, when we keep increasing the amplitude more than $0.3\mu$, the passing position does not move up any more, as shown in Figure 14. The final position stops at $m = 6$, and the current standard deviation caused by random jitter is 0.682, which is consistent with the independence condition. Therefore, we can infer that the engagement of deterministic perturbations causes little impact on the correlation of sampling bits but improves the balance of sampling sequences. With the deterministic jitter increasing, the sequences can pass the statistical test when the dependence condition holds.

However, though the balance is achieved for sampling sequences, for each sampling bit the balance is insufficient, because the jitter accumulation for each sampling has not been enough. This causes some security problems, such as predicting the sampling bits.

### 5.3 Predicting the "Random" Bits

The deterministic perturbation is assumed as sinusoidal signal $D(t) = A\sin(\frac{2\pi t}{T_D} + \phi_0)$. The half-period after perturbing becomes $X_i' = \int_{T_i}^{T_{i+1}}(1 + D(t))dt$. we have the following reasonable physical assumptions for deterministic perturbations [2]: $T_D >> \mu$ (slow variations of $D(t)$) and $X_i' \approx X_i$ (small deterministic jitter). Therefore, it is easy to deduce that the uniform distribution in $[0, \mu]$ still approximately holds for the new waiting time. Furthermore, compared with the sampling interval $s$ in the model without perturbations, the mean of the new $i$th interval is equivalent to $s - d_i$ to apply the model in Section 2, where $d_i = \int_{s_{i-1}}^{s_i} D(t)dt$. As we mentioned, it is useful to improve the balance of the whole sequence, however, the impact is very limited on a given sampling bit, which allows us to predict the seemingly random bits. The probability of the $i$th bit equaling to $b_i$ can be derived from the total probability formula $\text{Prob}(b_i) = \int_0^1 \text{Prob}(b_i|w_i)P_W(du)$, where $\text{Prob}(b_i|w_i)$ can be calculated from Equation (8) using the modified sampling interval $s - d_i$.

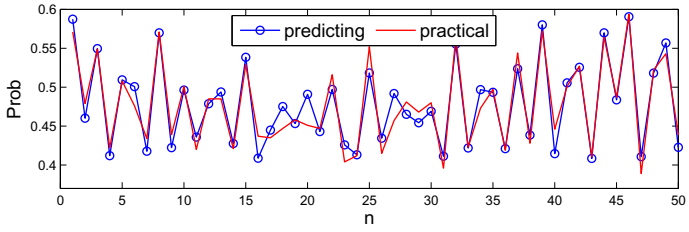



**Fig. 15.** The comparison of predicting and practical probabilities

Therefore, if precisely knowing the mean $\mu$, standard variance $\sigma$, and the behaviors of deterministic jitter, one can precisely compute the probabilities of sample bits in advance. We perform a prediction simulation, and compare the predicting probabilities with the practical ones in Figure 15. The practical probabilities come from the statistics of 1000 simulation samples that can pass FIPS 140-2. It is shown that the two sets of probabilities are consistent with each other in most sampling bits. Using the predicting results, one can optimize brute-force attacks to significantly reduce the breaking complexity. In practical terms, the more precise parameters of TRNGs one knows, the more effective attacks one can perform.

Though the TRNG output can pass the statistical tests under the perturbations, with environmental factors (such as supply voltage) changing, the frequency and amplitude of the perturbation might change to the values that no longer help to improve the "randomness" (e.g. the frequency changes to the multiples of the sampling frequency). Therefore, one way to guarantee the security of under-perturbation TRNGs is to keep the entropy sufficiency in each sampling bit, i.e. the $q$ should be large enough. As $d_i << s$, the requirement for $q$ value is approximately identical to that without deterministic perturbations at the worst case $r = 0$.

## 6 Conclusion and Future Work

In this paper, we propose an improved modeling method for oscillator-based TRNGs, and deduce the requirement for the parameters of security TRNGs. In order to verify the theory, we design an improved measuring circuit for acquiring the TRNG parameters. The measuring circuit can also be integrated into hardware for online tests and inner tests of the TRNGs. Furthermore, we apply the stochastic model to analyze the TRNGs with deterministic perturbations. We investigate the positive effect of perturbations on the statistical tests, and also provide the bound for the randomness improvement. By performing a simulated attack, we demonstrate that predicting the random bits could be possible. In future work, we will further analyze the accuracy of the measuring method and extend our stochastic model to the multiple-RO structures [17], especially for those injection-locked oscillators.

**Acknowledgements.** The authors would like to acknowledge the contribution of Dr. Wei Gao from Tsinghua University. We also thank the anonymous reviewers for their invaluable suggestions and comments to improve the quality of this paper.

## References

1. Amaki, T., Hashimoto, M., Mitsuyama, Y., Onoye, T.: A worst-case-aware design methodology for noise-tolerant oscillator-based true random number generator with stochastic behavior modeling. IEEE Transactions on Information Forensics and Security 8(8), 1331–1342 (2013)

2. Baudet, M., Lubicz, D., Micolod, J., Tassiaux, A.: On the security of oscillator-based random number generators. J. Cryptology 24(2), 398–425 (2011)
3. Bernard, F., Fischer, V., Valtchanov, B.: Mathematical model of physical RNGs based on coherent sampling. Tatra Mountains Mathematical Publications 45(1), 1–14 (2010)
4. Bochard, N., Bernard, F., Fischer, V., Valtchanov, B.: True-randomness and pseudo-randomness in ring oscillator-based true random number generators. Int. J. Reconfig. Comp. 2010 (2010)
5. Bucci, M., Germani, L., Luzzi, R., Trifiletti, A., Varanonuovo, M.: A high-speed oscillator-based truly random number source for cryptographic applications on a smart card IC. IEEE Transactions on Computers 52(4), 403–409 (2003)
6. Coppock, W.R., Philbrook, C.R.: A mathematical and physical analysis of circuit jitter with application to cryptographic random bit generation. Major qualifying project report, Worcester Polytechnic Institute (2005)
7. Dichtl, M., Golić, J.D.: High-speed true random number generation with logic gates only. In: Paillier, P., Verbauwhede, I. (eds.) CHES 2007. LNCS, vol. 4727, pp. 45–62. Springer, Heidelberg (2007)
8. Fischer, V., Bernard, F., Bochard, N., Varchola, M.: Enhancing security of ring oscillator-based trng implemented in FPGA. In: FPL, pp. 245–250 (2008)
9. Haddad, P., Teglia, Y., Bernard, F., Fischer, V.: On the assumption of mutual independence of jitter realizations in P-TRNG stochastic models. In: IEEE Design, Automation and Test in Europe Conference and Exhibition (DATE), pp. 1–6 (2014)
10. Hajimiri, A., Limotyrakis, S., Lee, T.H.: Jitter and phase noise in ring oscillators. IEEE Journal of Solid-State Circurts 34(6) (1999)
11. Information Technology Laboratory: FIPS 140-2: Security Requirement For Cryptographic Modules (2011)
12. Killmann, W., Schindler, W.: A design for a physical RNG with robust entropy estimators. In: Oswald, E., Rohatgi, P. (eds.) CHES 2008. LNCS, vol. 5154, pp. 146–163. Springer, Heidelberg (2008)
13. Markettos, A.T., Moore, S.W.: The frequency injection attack on ring-oscillator-based true random number generators. In: Clavier, C., Gaj, K. (eds.) CHES 2009. LNCS, vol. 5747, pp. 317–331. Springer, Heidelberg (2009)
14. Marsaglia, G.: Diehard Battery of Tests of Randomness, `http://www.stat.fsu.edu/pub/diehard/`
15. Petrie, C., Connelly, J.: A noise-based IC random number generator for applications in cryptography. IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications 47(5), 615–621 (2000)
16. Rukhin, A., et al.: A statistical test suite for random and pseudorandom number generators for cryptographic applications. NIST Special Publication 800–22, `http://csrc.nist.gov/publications/nistpubs/800-22-rev1a/SP800-22rev1a.pdf`
17. Sunar, B., Martin, W.J., Stinson, D.R.: A provably secure true random number generator with built-in tolerance to active attacks. IEEE Transactions on Computers 56(1), 109–119 (2007)
18. Valtchanov, B., Aubert, A., Bernard, F., Fischer, V.: Modeling and observing the jitter in ring oscillators implemented in FPGAs. In: DDECS, pp. 158–163 (2008)
19. Valtchanov, B., Fischer, V., Aubert, A., Bernard, F.: Characterization of randomness sources in ring oscillator-based true random number generators in FPGAs. In: DDECS, pp. 48–53 (2010)