Ratio-Based Ring Oscillator PUF Statistical Model

Ondřej Staníček, Jiří Buček, and Róbert Lórencz

Faculty of Information Technology, Czech Technical University in Prague,
Prague, Czech Republic
{staniond,jiri.bucek,robert.lorencz}@fit.cvut.cz

Abstract. A ratio-based ring oscillator physical unclonable function (ROPUF) provides high reliability and is capable of extracting multiple bits from a single ring oscillator pair. However, the derivation of the response bits from such PUFs and the assessment of their randomness have relied only on empirical statistical tests.

This paper introduces a novel, mathematically derived closed-form statistical model that characterizes the counter value distribution of the ratio-based ROPUF and formally describes the behavior of this PUF design. We empirically validate the model's assumptions using a large-scale real-world ring oscillator measurement dataset of 218 Xilinx Artix-7 FPGAs.

We demonstrate that this statistical framework enables theoretical quantification of response entropy of the ratio-based ROPUF. Furthermore, the model unlocks the use of advanced bit derivation methods, such as Equiprobable Quantization and Two-Metric Helper Data schemes. These were previously inapplicable to this PUF architecture, and we empirically show on the real-world dataset that they achieve better combinations of PUF reliability, randomness, and bit yield than the original counter-based derivation.

Keywords: physical unclonable function · ring oscillator · FPGA · statistical model · bit derivation

1 Introduction

Physical unclonable functions (PUFs) serve as essential cryptographic primitives, addressing critical information security challenges such as device authentication, secure identification, and cryptographic key generation. By acting as a unique hardware fingerprint, a PUF leverages the uncontrollable and random process variations inherent in semiconductor manufacturing to establish a distinct identity for each physical chip [14].

Designing a PUF that consistently extracts this uniqueness with high reliability and security remains a significant engineering challenge. Among the various proposed architectures, a ring oscillator-based physical unclonable function

(ROPUF) has emerged as a widely adopted solution due to its robust properties and ease of implementation on both application specific integrated circuits (ASICs) and field programmable gate arrays (FPGAs) [14].

A ring oscillator (RO) consists of an odd number of inverters connected in a feedback loop. The frequency of a RO is dictated by the cumulative propagation delays of the inverters, which are inherently tied to the physical manufacturing variations unique to each chip. A ROPUF uses the unique frequencies to extract the PUF responses — the device fingerprint.

The derivation of the PUF response is highly dependent on the distribution of the underlying gate delay variations. Therefore, understanding and accurately representing the statistical behavior of the oscillators is critical to evaluating the overall security and reliability of ROPUFs. Moreover, knowing the distribution of the ROPUF enables the use of more sophisticated bit derivation methods (such as Two-Metric Helper Data (TMHD) or Equiprobable Quantization schemes) that increase response size or reliability [2, 16].

1.1 Our Contribution

The ratio-based PUF was first introduced in [5]. This paper introduces a novel statistical model of this PUF construction. We mathematically derive the model from the ROPUF design and justify the assumptions that the model makes about ROs. We show interesting properties of the ratio-based PUF that are revealed by the model.

We verify the model and its assumptions based on a large-scale dataset of real-world measurements of RO frequencies on the 28 nm Xilinx Artix-7 FPGAs [3], while identifying future work needed in this area.

We then discuss why having a statistical model is useful, showing that it enables the evaluation of the entropy of the ROPUF responses. This in turn justifies how the PUF response is formed, which has previously been validated only empirically [6]. Furthermore, the statistical model enables the use of more sophisticated bit derivation methods that help increase the number of response bits and reliability, such as TMHD or Equiprobable Quantization schemes [2, 16]. We empirically compare these schemes against the original counter-based derivation on the real-world dataset using standard PUF quality metrics, showing improved reliability, randomness and bit yield.

1.2 Organization of the Paper

The rest of this paper is organized as follows. Section 2 reviews the existing work on ROPUF modeling and bit derivation schemes. Section 3 details the ratio-based ROPUF architecture which is modeled in this paper. In Section 4, we mathematically derive the proposed statistical model and discuss its properties. Section 5 validates the model and its underlying assumptions using a dataset of real-world FPGA measurements. Finally, Section 6 explores practical

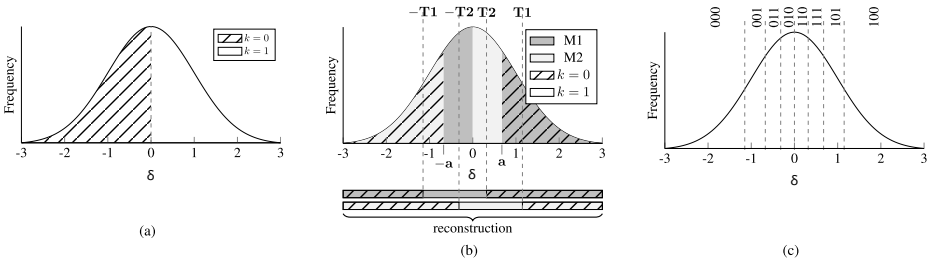


Fig. 1. Bit derivation schemes: (a) Sign-based bit derivation, (b) Two-Metric Helper Data scheme according to [2], (c) Equiprobable Quantization with eight intervals using a 3-bit Gray code. [15]

applications of the model, including theoretical entropy quantification, the implementation of advanced bit derivation methods, and an empirical comparison of their resulting PUF quality metrics on real-world data.

2 Related Work

This section reviews existing approaches to the statistical modeling of ROPUFs. Furthermore, it outlines the state-of-the-art bit derivation schemes used to extract bits of the PUF response.

2.1 Existing Work on ROPUF Modeling

Typically, ROPUFs generate a bit of their response using a frequency difference (comparison) $\delta = f_x - f_y$ of a pair of ROs. The resulting bit is then taken as the sign of δ [14].

This process is known as sign-based bit derivation. It is commonly modeled by assuming that f_x and f_y are independent normal random variables with the same mean. Then δ is also normal with zero mean and the probabilities of deriving a 0 or 1 bit are 50 %, as illustrated in Fig. 1a [1, 11, 12].

Another relevant ROPUF architecture is the Loop PUF. It is composed of a single configurable RO. The delay difference δ is obtained by subtracting the RO frequency for a challenge $\mathcal{C}$ and its complement $-\mathcal{C}$. The authors also experimentally show that δ is normally distributed [1].

In [11], the authors model the delays of the Loop PUF's n configurable delay elements as normal random variables. They show that n bits of entropy can be obtained using n challenges if and only if the challenges constitute a Hadamard code. They derive a closed-form expression for the entropy of the PUF response. They also show that the total entropy can be greater than n bits if more than n challenges are used.

2.2 Existing Bit Derivation Schemes

Beyond simple sign-based bit derivation, more sophisticated schemes can be used when the distribution of δ is known.

The Equiprobable Quantization scheme, first introduced in [16], divides the distribution of δ into n intervals of equal probability. Each interval is assigned a symbol, which then makes up a part of the PUF response. Using a Gray code, so that the adjacent intervals differ by only one bit, reduces bit errors caused by small variations when δ is near the interval boundaries. This way, one of n symbols can be extracted from each realization of δ, compared to only 1 bit for sign-based bit derivation, although the reliability of the PUF decreases with increasing n. See Fig. 1c for an example with $n = 8$, creating 3 bits per realization of δ using a Gray code.

The sign-based bit derivation can be interpreted as Equiprobable Quantization with $n = 2$ with the interval $(-\infty, 0)$ assigned to '0' and $(0, \infty)$ assigned to '1'.

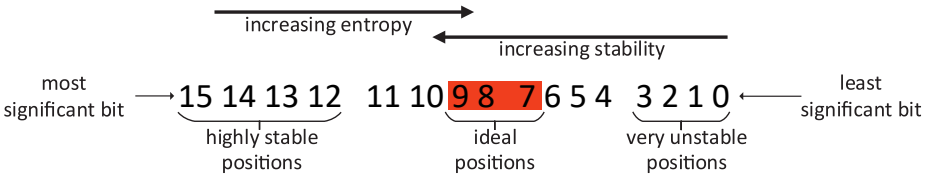


Fig. 2. Counter-based bit derivation scheme. Based on [6].

To address the reliability of PUFs, helper data schemes can be used to discard unreliable bits. However, this filtering approach reduces the PUF response size and exposes the PUF to helper data manipulation attacks, where an adversary can iteratively flip reliable and unreliable bits in the helper data to deduce the response bits.

To secure delay PUFs and increase their reliability without discarding any response bits, Danger et al. [2] proposed a Two-Metric Helper Data (TMHD) scheme. Assuming δ follows a normal distribution, they define two distinct derivation metrics, M_1 and M_2, which map different sub-intervals of the δ distribution to 0 and 1 bits. During the enrollment phase, the metric that places δ farthest from the decision boundaries, thus maximizing reliability, is selected and recorded as helper data for that specific δ (see Fig. 1b). Because maliciously altering the chosen metric yields exactly a 50 % probability of flipping the resulting bit regardless of its original value, the scheme is natively robust against helper data manipulation attacks. It also increases reliability compared to sign-based bit derivation.

A fundamentally different approach is the counter-based bit derivation, first introduced by Kodýtek and Lórencz in [5] for a ratio-based ROPUF. Instead of evaluating a frequency difference, this architecture accumulates the RO pair

response in a hardware counter, producing an integer counter value c (the ratio-based counterpart of δ, detailed in Section 3). Bits at suitable positions of c are then selected as the PUF response. Positions close to the least significant bit (LSB) have the highest entropy but are unreliable, whereas bits close to the most significant bit (MSB) are more stable but with low entropy. Therefore, a suitable range of positions in the middle of the counter has to be selected (see Fig. 2).

3 The Ratio-Based ROPUF Design

In this paper, we develop a statistical model for the ratio-based ROPUF architecture first introduced in [5]. Building on the counter-based derivation scheme, it operates by evaluating pairs of ROs selected from two distinct sets.

To generate a part of the PUF response, the oscillations of each RO in a chosen pair are tracked by two independent counters. Both counters stop as soon as the first of the two overflows. The final counter value c is then read from the non-overflowed counter.[1] The overall architecture of this design is illustrated in Fig. 3.

Mathematically, the counter value c is a scaled ratio of the two RO frequencies. It can be calculated if the frequencies f_x and f_y are known:

$$c \cong \frac{\min(f_x, f_y)}{\max(f_x, f_y)} \cdot 2^l \tag{1}$$

where l is the bit width of the counters. In [13], it was experimentally confirmed that c can be calculated with sufficient precision from f_x and f_y using Equation (1). It was also shown in [7] that the ratio-based ROPUF is more reliable than the difference-based ROPUFs, mainly under varying environmental conditions (temperature and voltage).

Bits of the PUF response are then calculated from c using the counter-based derivation scheme described in Section 2.2. In [5, 6], 2 to 4 bits were extracted from c from positions 7 to 10 of a 16-bit counter.

However, in these previous works, determining which specific bit positions are suitable was entirely empirical, relying on randomness tests of the PUF responses. This highlights the need for a statistical model to more rigorously guide bit selection, quantify theoretical entropy of the response, and provide deeper insights into the overall behavior of ratio-based PUFs.

4 Statistical Model of Ratio-Based ROPUF

The primary contribution of this paper is a statistical model of the ratio-based PUF described in Section 3. This means an exact derivation of a closed-form

[1] The random variable C (with its realization c) for ratio-based PUFs is equivalent to the random variable δ in the difference-based PUFs.

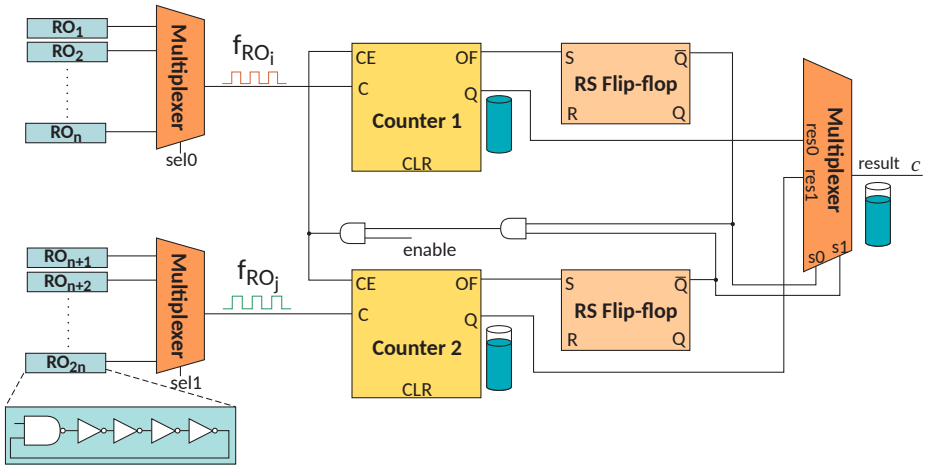


Fig. 3. The ratio-based ROPUF design diagram. [6]

formula for a distribution of the counter value c from Equation (1), based on reasonable assumptions that fit the actual hardware behavior.

In order to derive the explicit distribution of c, we assume two independent normally distributed random variables X and Y.

$$X \sim \mathcal{N}(\mu_x, \sigma_x^2), \quad Y \sim \mathcal{N}(\mu_y, \sigma_y^2) \tag{2}$$

The oscillator pair draws its frequencies f_x and f_y at fabrication time from X and Y. The hardware measures f_x and f_y and calculates the resulting value of c at the PUF evaluation time accordingly.

It is important to note that the model does not account for the measurement noise affecting an individual RO at evaluation time. Instead, it captures how the counter value c varies across a population of devices and RO pairs. It therefore describes the distribution of the noise-free PUF response — the entropy of the correctly reconstructed response. This is a property of the population distribution, whereas reliability depends on the evaluation-time noise. The model nevertheless informs reliability indirectly, since the decision boundaries for bit derivation it exposes govern the entropy–reliability trade-off discussed in Section 6.

The problem then reduces to deriving the distribution of a random variable C, where:

$$C := \frac{\min(X, Y)}{\max(X, Y)} \cdot 2^l \tag{3}$$

and the model parameters are μ_x, μ_y, σ_x and σ_y.

Justification of the model assumptions for real hardware and experimental verification on real-world data are given in subsequent Sections 4.3 and 5.

4.1 Normal Ratio Distribution

We first show a distribution of a simpler random variable

$$Z := \frac{X}{Y} \tag{4}$$

and afterwards we derive the distribution of C using Z.

Since X and Y are independent and effectively positive random variables (in real hardware f_x and f_y are always strictly positive), Z follows a noncentral uncorrelated normal ratio distribution. The probability density function (PDF) of Z is known and was derived by Hinkley in [4].

Using helper functions:

$$\begin{aligned} a(z) &:= \sqrt{\frac{z^2}{\sigma_x^2} + \frac{1}{\sigma_y^2}} \\ b(z) &:= \frac{\mu_x z}{\sigma_x^2} + \frac{\mu_y}{\sigma_y^2} \\ \kappa &:= \frac{\mu_x^2}{\sigma_x^2} + \frac{\mu_y^2}{\sigma_y^2} \\ d(z) &:= \exp\left(\frac{b^2(z) - \kappa a^2(z)}{2a^2(z)}\right) \end{aligned}$$

the PDF of Z, denoted as f_Z, is expressed as: [4]

$$f_Z(z) = \frac{b(z)d(z)}{a^3(z)\sqrt{2\pi}\sigma_x\sigma_y}\left[2\Phi\left(\frac{b(z)}{a(z)}\right) - 1\right] + \frac{1}{a^2(z)\pi\sigma_x\sigma_y}e^{-\frac{\kappa}{2}} \tag{5}$$

where Φ denotes the cumulative distribution function of the standard normal distribution.

4.2 Counter Value Distribution C

We will now derive the exact distribution of C using Z and its PDF. First, we look at C without the scaling factor 2^l. We define:

$$R := \frac{\min(X,Y)}{\max(X,Y)} = \min\left(\frac{X}{Y}, \frac{Y}{X}\right) = \min\left(Z, \frac{1}{Z}\right) \tag{6}$$

Now consider the cumulative distribution function (CDF) of R, defined as $F_R(r) := P(R \leq r)$. The minimum of two values is less than r if and only if at least one of the values is less than r. Applying this property and using R written in terms of Z from Equation (6), we obtain:

$$P(R \leq r) = P\left(Z \leq r \cup \frac{1}{Z} \leq r\right) = P\left(Z \leq r \cup Z \geq \frac{1}{r}\right) \tag{7}$$

The domain of r is restricted to $(0, 1]$ by the definition of R. Consequently, the events $Z \leq r$ and $Z \geq \frac{1}{r}$ are mutually exclusive and the probability of their union is simply the sum of their individual probabilities:

$$P(R \leq r) = P(Z \leq r) + P\left(Z \geq \frac{1}{r}\right) \tag{8}$$

We can rewrite Equation (8) into CDFs:

$$F_R(r) = F_Z(r) + \left(1 - F_Z\left(\frac{1}{r}\right)\right) \tag{9}$$

and obtain a PDF of R by differentiation of both sides with respect to r:

$$f_R(r) = f_Z(r) + \frac{1}{r^2} f_Z\left(\frac{1}{r}\right) \tag{10}$$

The final step is to scale R to obtain the final C:

$$C = R \cdot 2^l \tag{11}$$

Using the scaling property of PDFs ($f_C(x) = \frac{1}{|2^l|} f_R(\frac{x}{2^l})$), we obtain the final PDF of C:

$$f_C(x) = \frac{1}{2^l}\left[f_Z\left(\frac{x}{2^l}\right) + \left(\frac{2^l}{x}\right)^2 f_Z\left(\frac{2^l}{x}\right)\right] \tag{12}$$

We finally arrive at a PDF of C expressed only in terms of f_Z, which we already know from Equation (5). Using f_C and the parameters of the ROs (μ_x, μ_y, σ_x and σ_y), it is now possible to calculate the theoretical distribution of the PUF counter values using a closed-form formula.

Symmetric Case Simplification The ratio-based PUF construction is also unique in that it allows the RO pairs to be asymmetric [6]. This is also captured by the model, as the parameters for the two RO sets are separate (although the ROs have to be symmetric within each set).

However, ROPUF constructions typically use only fully symmetric ROs. In this case $\mu_x = \mu_y$ and $\sigma_x = \sigma_y$ and the expression for f_C can be greatly simplified.

In the symmetric case Z and its reciprocal are equal in distribution:

$$Z = \frac{X}{Y} \stackrel{d}{=} \frac{Y}{X} = \frac{1}{Z} \tag{13}$$

The PDF of Z can then be written as:

$$f_Z(x) = f_{\frac{1}{Z}}(x) = \frac{1}{x^2} f_Z\left(\frac{1}{x}\right) \tag{14}$$

and we can substitute it into Equation (12) and simplify:

$$f_C(x) = \frac{1}{2^l}\left[f_Z\left(\frac{x}{2^l}\right) + \underbrace{\left(\frac{2^l}{x}\right)^2 f_Z\left(\frac{2^l}{x}\right)}_{\text{substitute } f_Z(x)=\frac{1}{x^2}f_Z\left(\frac{1}{x}\right)}\right] \tag{15}$$

$$f_C(x) = \frac{1}{2^l}\left[f_Z\left(\frac{x}{2^l}\right) + f_Z\left(\frac{x}{2^l}\right)\right] = 2^{1-l} f_Z\left(\frac{x}{2^l}\right) \tag{16}$$

Three Modes of the C Distribution Depending on the input parameters, the exact distribution of C exhibits three distinct behaviors (see Fig. 4). When $\mu_x \gg \mu_y$ (or vice versa), the min/max 'sorting' operations are effectively static and C behaves as a simple ratio of independent normals, closely approximating a normal distribution. However, as the means move closer together and the underlying distributions overlap, the 'sorting' logic causes the resulting distribution to fold against the upper limit 2^l, producing an approximation of a reversed folded normal distribution. In the symmetric case, the distribution is folded around the middle, resulting in a reversed half-normal approximation anchored at 2^l.

It is important to note that because the true distribution of C is fundamentally a ratio, it is not mathematically identical to the normal, folded normal, or half-normal families. However, these standard distributions serve as highly accurate approximations when the coefficient of variation is small. As shown in Fig. 4 for $\text{CV} = \frac{\sigma}{\mu} = 0.1$, the normal/folded normal/half-normal distributions closely follow the exact distributions of C, though the minor deviations inherent to the true ratio remain visible.

4.3 Equivalence of Normal Frequency and Period

To be able to explicitly derive the distribution of C, we assumed that the frequencies f_x and f_y are normally distributed.

However, the hardware behavior is different. In reality, the period of the ROs is normally distributed and therefore the frequencies have a reciprocal normal distribution. This comes from the normal distribution of delays in MOS transistors selected at the time of fabrication [9, 10]. The RO period can be calculated as a sum of the normally distributed delays d_i of the n inverter stages [17], so the central limit theorem dictates that the RO period T_{RO} will also be normally distributed.

$$T_{\text{RO}} = 2\sum_{i=1}^{n} d_i \tag{17}$$

Although this invalidates our initial assumption regarding the distributions of f_x and f_y, the underlying model remains correct, even though the PUF hardware measures RO frequencies, which are not normally distributed. We will demonstrate that the computation of the counter value c is invariant whether evaluated using frequency or period values.

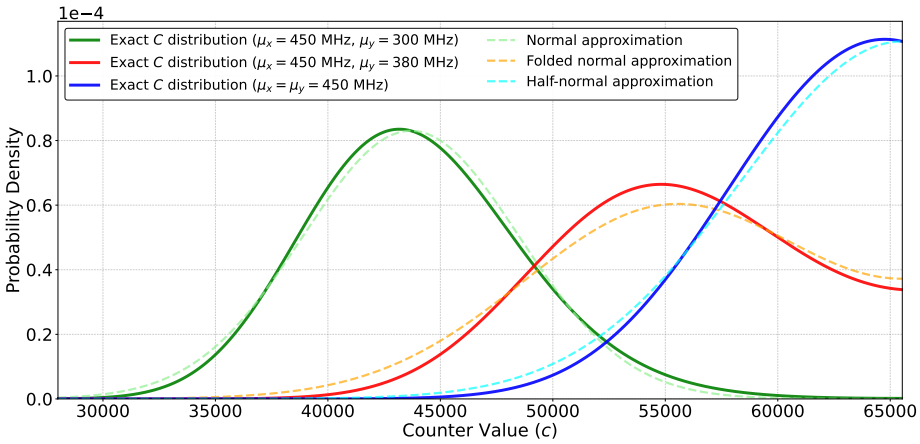


Fig. 4. Three modes of the C distribution based on different parameters, with normal, folded normal and half-normal approximations. CV $= \dfrac{\sigma}{\mu} = 0.1$ in all three cases and σ calculated accordingly.

Because frequency (f) and period (T) are inversely proportional ($f = 1/T$), and the transformation $g(x) = 1/x$ is strictly decreasing for positive arguments, the min and max operators invert as follows:

$$\min(f_x, f_y) = \min\left(\frac{1}{T_x}, \frac{1}{T_y}\right) = \frac{1}{\max(T_x, T_y)} \tag{18}$$

$$\max(f_x, f_y) = \max\left(\frac{1}{T_x}, \frac{1}{T_y}\right) = \frac{1}{\min(T_x, T_y)} \tag{19}$$

Substituting these relationships into Equation (1) yields:

$$c = \frac{\min(f_x, f_y)}{\max(f_x, f_y)} \cdot 2^l = \frac{\frac{1}{\max(T_x, T_y)}}{\frac{1}{\min(T_x, T_y)}} \cdot 2^l = \frac{\min(T_x, T_y)}{\max(T_x, T_y)} \cdot 2^l \tag{20}$$

Although the physical PUF hardware evaluates c using frequencies f_x and f_y, we proved that this operation is equivalent to calculating c based on the oscillator periods. Therefore, to satisfy the assumption of normality of the underlying model, the random variables X and Y must represent the RO periods. As a result, the correct model parameters (μ_x, μ_y, σ_x and σ_y) should be derived from period measurements rather than frequencies. Consequently, the theoretical model remains mathematically exact while accurately reflecting the hardware reality of normally distributed oscillator periods.

4.4 Parametrization by Coefficient of Variation

The distribution of C is parametrized by the four values μ_x, μ_y, σ_x and σ_y derived from the RO periods. However, we can also show that C can be described using a different set of parameters. This will be useful later in Section 5.

General Case We can rewrite X and Y from Equation (2) using standard normal variables $U, V \sim \mathcal{N}(0,1)$:

$$X = \mu_x + \sigma_x U, \quad Y = \mu_y + \sigma_y V \tag{21}$$

With a coefficient of variation $\mathrm{CV} := \dfrac{\sigma}{\mu}$, we can express Z as:

$$Z = \frac{X}{Y} = \frac{\mu_x + \sigma_x U}{\mu_y + \sigma_y V} = \frac{\mu_x \left(1 + \frac{\sigma_x}{\mu_x} U\right)}{\mu_y \left(1 + \frac{\sigma_y}{\mu_y} V\right)} = \frac{\mu_x}{\mu_y} \cdot \frac{1 + \mathrm{CV}_x \cdot U}{1 + \mathrm{CV}_y \cdot V} \tag{22}$$

From Equation (22) we can see that Z can also be parametrized using:

1. $\dfrac{\mu_x}{\mu_y}$ – the ratio of means of X and Y
2. CV_x – the coefficient of variation of X
3. CV_y – the coefficient of variation of Y

As $C = \min(Z, \frac{1}{Z}) \cdot 2^l$ (from Equations (6) and (11)) and the min and scaling operations are deterministic, C can also be parametrized by these three parameters.

Symmetric Case If the RO pairs are symmetric, we know that:

$$\mu := \mu_x = \mu_y, \quad \sigma := \sigma_x = \sigma_y, \quad \mathrm{CV} := \mathrm{CV}_x = \mathrm{CV}_y \tag{23}$$

Then Z from Equation (22) can be simplified as:

$$Z = \frac{\mu}{\mu} \cdot \frac{1 + \mathrm{CV} \cdot U}{1 + \mathrm{CV} \cdot V} = \frac{1 + \mathrm{CV} \cdot U}{1 + \mathrm{CV} \cdot V} \tag{24}$$

And therefore, in the symmetric case, the only parameter of the distribution C is the coefficient of variation $\mathrm{CV} = \frac{\sigma}{\mu}$ of the ROs.

5 Model Validation

Section 4 derived the theoretical distribution of the counter values C. This model yields an exact distribution, provided that its underlying assumptions hold. That is, the frequencies (or rather periods, as discussed in Section 4.3) of the RO pairs are independent and normally distributed.

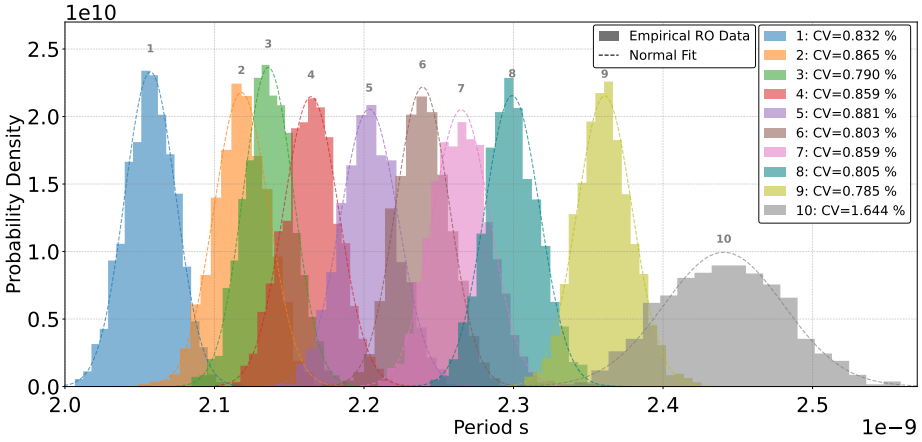


Fig. 5. Ten histograms (one per FPGA) of 1600 mean RO periods each. Each device is represented by a distinct color with a fitted normal distribution and annotated with its coefficient of variation (CV). Device 10 is a notable outlier.

To evaluate whether these assumptions are satisfied in practice, this section analyzes a public dataset of RO measurements collected from 218 Xilinx Artix-7 XC7A35T FPGAs [3]. The 1600 ROs of type q_L (notation from [3]) from each FPGA were taken from the dataset. The period of each RO was calculated by averaging all 100 provided measurements taken over an interval of 143.45 µs. This specific duration was chosen because it closely approximates the PUF measurement time for a 16-bit counter that was used in previous implementations [5–7]. Therefore, the value of $l := 16$ is used for all experiments.

Only the mean period for each RO is considered, as the model does not take into account noise during individual PUF measurements, but rather models the variations across a population of ROs and devices, which is consistent with the defined model scope (Section 4).

Note that the physical positions of the ROs were not taken into account and the measurements were captured sequentially rather than in parallel, which deviates from true PUF hardware behavior. This is a limitation of the public dataset used, and investigating the impact of these factors is left for future work.

The q_L ROs are all implemented and routed exactly the same on the FPGA, so we evaluate the symmetric case for C, where the RO pairs are drawn from the same normal distribution ($X \overset{d}{=} Y$).

A histogram of 1600 RO periods plotted per FPGA (for a subset of 10 FPGAs) together with a normal fit for each is visible in Fig. 5. The data demonstrate that each FPGA produces a distinct but normal distribution of the RO periods. This means that the model works if the parameters are measured and used for each FPGA individually.

However, Section 4.4 demonstrated that in the symmetric case, the model's sole parameter is the coefficient of variation (CV). As shown in Fig. 5, most

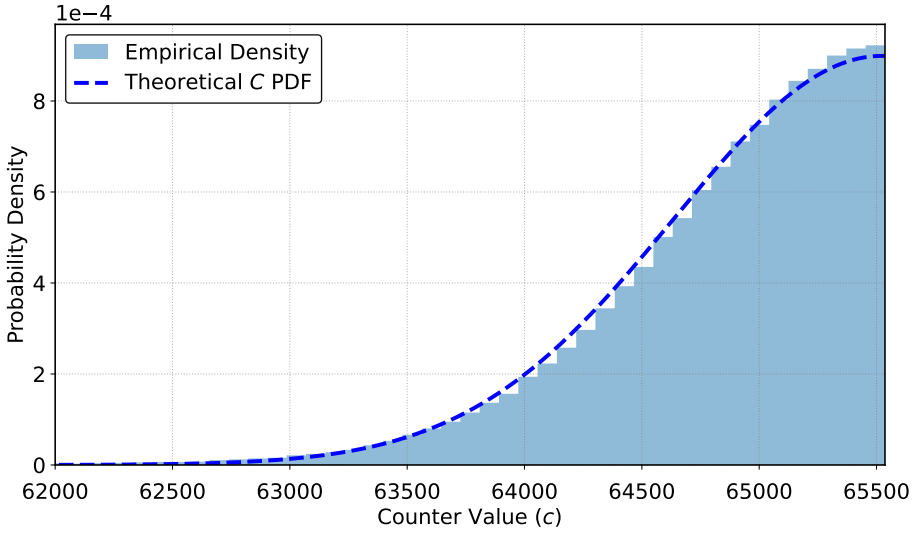


Fig. 6. Theoretical C distribution PDF with a histogram of 174400 counter values c calculated from 800 RO pairs per FPGA and from a total of 218 FPGAs.

empirical CVs cluster around 0.85 %. Consequently, a single model can accurately describe the behavior of all FPGAs with a similar CV value.

Fig. 6 illustrates this unified behavior. It plots a theoretical C PDF parametrized by $\mu_x = \mu_y = 2.194\,\text{ns}$ (calculated as a mean of all RO period measurements) and $\sigma_x = \sigma_y = 21.571\,\text{ps}$ (calculated as the average standard deviation of RO periods across all FPGAs). The empirical data are represented as a histogram of counter values c, calculated from pairs of RO frequencies using Equation (1). The RO pairs were formed within the same FPGA (yielding 800 pairs from the 1600 ROs per device), and the results were aggregated across all 218 FPGAs. The theoretical C PDF aligns closely with the empirical density calculated from the dataset.

However, this global applicability is possible only in the symmetric case, assuming the CV is consistent across the device population. In the general asymmetric case, applying a single model across all devices becomes significantly more challenging. The asymmetric model relies on three parameters ($\frac{\mu_x}{\mu_y}$, CV_x, and CV_y), which come from the two different RO distributions. The asymmetric ROs may exhibit different sensitivities to variations in the manufacturing process and environmental fluctuations. This would result in the parameters being different per-device and in a changing environment (temperature, voltage). This assumption is in line with the findings in [7], where it was experimentally confirmed that the symmetric ROs perform better than the asymmetric ones.

It is important to note that not all FPGAs in the dataset have a similar CV value. We observed a small number of outliers with CVs almost double the

nominal 0.85 %. For example, Fig. 5 highlights one such outlier device (number 10 in gray) with a CV of 1.644 %. For these anomalous devices, the generalized global model would serve as a poor predictor of hardware behavior. It is not clear why some devices exhibit such different CV values, though their period distribution is still normal. Therefore, the model can still be fitted directly to these devices.

Beyond these device-specific outliers, this empirical validation is fundamentally limited to a single dataset from one specific FPGA family. The statistical behavior observed here may differ for other FPGA architectures, varying process nodes, or ASIC implementations. Furthermore, only the normality assumption was verified, but the independence of the RO pair was not. While assuming independence of the ROs is standard practice in existing literature [1, 11, 12], physical validation remains critical. Although the model works well on the presented dataset (Fig. 6), the RO pairs were not measured in parallel and their exact position on the chip was not taken into account. Possible intra-die RO correlations might be accounted for by introducing a correlated normal ratio for Z [4]. More work needs to be done in this direction.

Consequently, before relying on this theoretical model, any specific PUF implementation must verify that its target hardware satisfies the assumptions of the model. Expanding this empirical evaluation remains an important direction for future work.

6 Discussion and Model Applications

While the presented statistical model provides important theoretical insights into the behavior of the counter distribution, its primary value lies in its practical application. By characterizing this distribution, the model enables quantifying the entropy of the PUF response when using the original counter-based bit derivation scheme. It also enables the use of other bit derivation techniques that were not possible before. We conclude by empirically comparing all three schemes on real-world data (Section 6.4).

6.1 Model-Guided Counter-Based Bit Derivation

The ratio-based PUF presented in [5] extracts response bits from specific positions of the counter value c (see Fig. 2). Previously, selecting these suitable bit positions was a purely empirical process based on randomness and reliability testing [5–7].

Because our model explicitly defines the distribution of counter values, the theoretical entropy of the response bits can now be calculated directly from the underlying RO parameters. This makes the important design decision of which bit positions to select significantly more rigorous. Furthermore, it facilitates model-guided hardware design, allowing RO architectures (e.g., stage count, routing) to be optimized for parameters for which the model predicts desired entropy values.

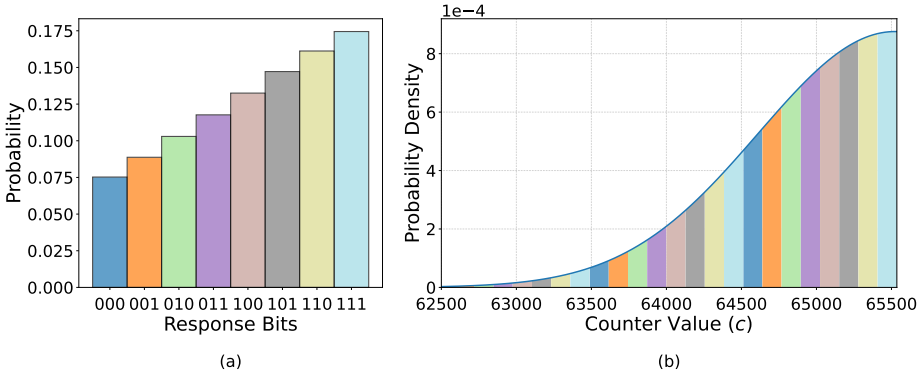


Fig. 7. Counter-based bit derivation from bit positions 7–9 visualized. (a) The discrete probability distribution $\mathcal{X}$ of the derived 3-bit response. (b) The continuous PDF of C ($\mu_x = \mu_y = 2.194\,\text{ns}$, $\sigma_x = \sigma_y = 21.571\,\text{ps}$), partitioned into periodic intervals corresponding to the resulting bit sequences. $H(\mathcal{X}) = 2.949$ bits, corresponding to 0.983 bits per response bit. $H_{\min}(\mathcal{X}) = 2.517$ bits, corresponding to 0.839 bits per response bit.

An example of this theoretical entropy calculation is illustrated in Fig. 7, which demonstrates the derivation of a 3-bit response from positions 7 to 9 of the counter value c.

Selecting specific bit positions partitions the domain of the counter variable C into a series of periodic intervals. As shown in Fig. 7b, each of the $2^3 = 8$ possible resulting bit sequences corresponds to a set of non-contiguous regions along the x-axis, represented by distinct colors overlaid on the theoretical PDF of C. To evaluate the probability of generating a particular bit sequence, the PDF is integrated across all disjoint intervals that map to this specific sequence.

This integration transforms the continuous distribution of C into the discrete probability distribution of the PUF response bits, depicted in Fig. 7a. Finally, treating this discrete distribution as a random variable $\mathcal{X}$, the theoretical Shannon entropy $H(\mathcal{X})$ and min-entropy $H_{\min}(\mathcal{X})$ of the extracted bits can be calculated as:

$$H(\mathcal{X}) = -\sum_{x\in\mathcal{X}} p(x)\log_2 p(x) \tag{25}$$

$$H_{\min}(\mathcal{X}) = -\log_2\left(\max_{x\in\mathcal{X}} p(x)\right) \tag{26}$$

Selecting bit positions closer to the LSB of the counter value c inherently narrows the periodic regions mapped to each bit sequence. While this narrower integration window yields a more uniform discrete distribution (increasing its entropy), it simultaneously degrades reliability, as the denser decision boundaries make the derivation more susceptible to errors in the measurement of c. Conversely, shifting the selection towards the MSB improves stability at the cost

of entropy. Similarly, extracting a wider block of bits from c increases the total bit yield of the PUF for a given number of ROs, while decreasing the entropy per extracted bit and vice versa.

6.2 Equiprobable Quantization Scheme

Explicitly defining the distribution of C also unlocks other bit derivation schemes previously inapplicable to the ratio-based PUF.

A prime example is the Equiprobable Quantization scheme, which was introduced in Section 2.2 (illustrated for a difference-based PUF in Fig. 1c). To apply it to the ratio-based architecture from [5], the continuous distribution of C is partitioned into n intervals of equal probability, with each assigned a unique symbol. During evaluation, the measured counter value c is mapped to its corresponding symbol. Because these intervals are equiprobable, the resulting discrete symbol distribution is uniform, yielding full theoretical entropy. Typically, n is chosen as a power of two ($n = 2^k$), allowing symbols to be encoded using a k-bit Gray code to minimize bit errors.[2]

Compared to the original counter-based bit derivation, this scheme achieves theoretical full entropy while increasing reliability, as the number of interval boundaries, where bit errors may occur, is lower. However, this approach introduces minor hardware overhead, as the evaluated c must be compared against the $n - 1$ decision thresholds to determine the output symbol. Furthermore, if a global model (a single C distribution for all devices) cannot be used, these threshold values must be calculated individually and stored as configurable parameters during per-device enrollment.

6.3 Two-Metric Helper Data Scheme

The explicit knowledge of the C distribution also enables the adaptation of the Two-Metric Helper Data (TMHD) scheme (introduced in Section 2.2 and illustrated in Fig. 1b for a difference-based PUF). Although originally designed for a normal distribution [2], this scheme can be applied to the ratio-based PUF using our proposed model.

Rather than deriving the eight necessary octiles from a standard normal distribution, they are calculated directly from the C distribution. The definitions of the metrics M_1 and M_2, as well as the metric selection criteria for each RO pair, remain identical to the original scheme.

Compared to the original counter-based bit derivation, this scheme theoretically achieves significantly higher reliability by using only two interval boundaries that are intentionally moved away from the expected value of c. However, it only extracts one bit per RO pair and requires additional hardware to store and compare c against the decision thresholds (similar to the Equiprobable Quantization scheme), as well as to calculate and store the per-device helper data in a non-volatile memory during PUF enrollment.

[2] Gray code can also be applied to the counter-based derivation scheme to improve reliability, as successfully demonstrated in [6].

Table 1. Comparison of different bit derivation schemes (the newly unlocked model-based ones in bold) using common PUF quality metrics. The PUF response bits used to calculate the PUF metrics were derived from the large-scale dataset of 218 FPGAs [3]. Uniformity, uniqueness, and reliability are given in % and min-entropy is in bits per response bit. Ideal values are 50 % (uniformity, uniqueness), 100 % (reliability), and 1 (min-entropy). Bit position 0 is the LSB and position 15 is the MSB.

		PUF response bits-based			Model-based
Bits/pair	Bit derivation scheme	Uniformity [%]	Uniqueness [%]	Reliability [%]	$H_{\min}$/bit
1	Counter-based, bit 7	52.70	49.84	91.54	0.921
	Counter-based, bit 8	56.02	49.24	96.00	0.846
	Equiprobable[3], $n = 2$	**51.32**	**48.77**	**99.17**	**1**
	TMHD	**50.09**	**49.75**	**99.992**	**1**
2	Counter-based, bits 7–8	54.39	49.56	93.76	0.887
	Counter-based, bits 8–9	59.28	47.97	97.09	0.787
	Equiprobable, $n = 4$	**50.82**	**49.09**	**98.44**	**1**
3	Counter-based, bits 6–8	53.41	49.68	90.06	0.913
	Counter-based, bits 7–9	57.03	48.64	95.22	0.839
	Equiprobable, $n = 8$	**50.64**	**49.33**	**97.17**	**1**
4	Counter-based, bits 6–9	55.66	48.92	92.09	0.873
	Counter-based, bits 7–10	62.58	44.55	96.27	0.768
	Equiprobable, $n = 16$	**50.70**	**49.49**	**95.02**	**1**

6.4 Empirical Comparison of Bit Derivation Schemes

To demonstrate the practical benefit of the model, we empirically compare the original counter-based bit derivation against the two newly unlocked schemes (Equiprobable Quantization and TMHD) on real-world hardware data. Table 1 reports common PUF quality metrics calculated from the PUF response bits as well as min-entropy predicted by the model for each scheme across several bit yields (number of bits produced per RO pair).

First, the counter values c were obtained in the same way as in Section 5: computed from the RO periods of the large-scale dataset of 218 FPGAs [3] using Equation (1), forming 800 RO pairs per device for $l = 16$. Each pair was then mapped to its response bits using the respective bit derivation scheme. Because each RO was measured 100 times in the dataset, 100 counter values per device per RO pair were obtained. The PUF response is a concatenation of all response bits for each device. The reference response of each device was taken as the bitwise majority vote across all 100 device responses.

The metrics were calculated following the methodology of Maiti et al. [8] from the PUF responses. Uniformity is the mean Hamming weight of the response bits, uniqueness is the mean inter-device Hamming distance, and reliability is $100\% - \mathrm{HD}_{\mathrm{intra}}$, where $\mathrm{HD}_{\mathrm{intra}}$ is the mean intra-device Hamming distance between the

[3] Equiprobable Quantization with $n = 2$ is equivalent to a sign-based bit derivation, see Fig. 1a.

reference response and the individual measurements. The min-entropy $H_{\min}$ per bit is calculated directly by the model (as described earlier in this section, Fig. 7) for the counter-based scheme, whereas the Equiprobable and TMHD schemes produce a uniform response distribution by construction and therefore reach the theoretical maximum of 1 bit of min-entropy per response bit.

The results confirm the entropy–reliability trade-off predicted by the model. For the counter-based scheme, selecting bits closer to the LSB increases min-entropy but lowers reliability, whereas shifting towards the MSB improves reliability at the cost of entropy, uniformity and uniqueness. The uniformity of the counter-based scheme in particular degrades substantially for higher bit positions, reaching 62.58 % for bits 7–10. In contrast, both model-based schemes maintain uniformity and uniqueness close to the ideal 50 % and full min-entropy at every bit yield. At an equal bit yield, the Equiprobable Quantization scheme consistently matches or exceeds the reliability of the counter-based scheme, while the TMHD scheme achieves a near-perfect reliability of 99.992 % for a single bit per RO pair.

These results show that the schemes unlocked by the model produce measurably better PUF responses, offering theoretically full entropy and higher reliability without reducing bit yield. Notably, although the model itself does not capture evaluation-time noise and thus cannot predict reliability (Section 4), the bit derivation schemes it unlocks improve it. The Equiprobable Quantization scheme owes its full entropy to its equiprobable intervals and its improved reliability to the reduced number of decision boundaries, while the TMHD scheme maximizes reliability by placing its two equiprobable boundaries far from the expected counter value.

The scope of this evaluation is, however, limited in several respects. Reliability was evaluated from repeated measurements taken under nominal operating conditions, rather than across temperature and voltage variations. Together with the dataset limitations already noted in Section 5, a rigorous hardware evaluation of the two schemes remains important future work. Despite these limitations, the presented comparison provides strong evidence that the bit derivation schemes newly enabled by the model improve both the reliability and the randomness of the PUF responses.

7 Conclusion and Future Work

In this paper, we introduced a novel theoretical statistical model characterizing the behavior of a ratio-based ROPUF [5]. By explicitly defining the closed-form distribution of the PUF counter value, we formally describe the PUF behavior and can transition counter-based bit derivation from a purely empirical exercise into a more rigorous, model-guided process.

We verify the model and its assumptions on a real-world FPGA dataset of RO measurements while identifying the limitations of our analysis. We also show that the model unlocks the use of advanced bit derivation schemes. Specifically, the Two-Metric Helper Data (TMHD) and the Equiprobable Quantization

schemes, which were previously inapplicable to this architecture, offer potentially significant improvements in reliability, randomness, or PUF response size. Both schemes reach the theoretical full entropy by construction, and an empirical comparison on the real-world FPGA dataset confirms that they also provide higher reliability and better PUF quality metrics than the original counter-based bit derivation.

While this work establishes a theoretical foundation, important research directions remain to bridge the gap between the theoretical model and the hardware reality. The model captures the entropy of the noise-free PUF response but does not model reliability — the bit errors that occur when the response is re-evaluated. Nevertheless, the empirical comparison shows that the bit derivation schemes it unlocks improve reliability in practice. RO intra-die spatial correlations, which might break the independence assumption, must also be investigated and potentially taken into account, and the effects of parallel versus sequential RO measurements need to be evaluated.

Because the model naturally supports asymmetric ROs, future work will study their behavior compared to standard symmetric designs. Finally, subsequent research will focus on the physical implementation of the newly unlocked TMHD and Equiprobable Quantization schemes in hardware to rigorously assess their real-world benefits and overhead.

Acknowledgments. This work was supported by the Grant Agency of the Czech Technical University in Prague, grant No. SGS26/187/OHK3/3T/18 funded by the MEYS of the Czech Republic.

Disclosure of Interests. The authors have no competing interests to declare that are relevant to the content of this article.

References

1. Cherif, Z., Danger, J.L., Guilley, S., Bossuet, L.: An easy-to-design PUF based on a single oscillator: The Loop PUF. In: Euromicro Conference on Digital System Design (DSD). pp. 156–162 (2012). https://doi.org/10.1109/DSD.2012.22
2. Danger, J.L., Guilley, S., Schaub, A.: Two-metric helper data for highly robust and secure delay PUFs. In: IEEE International Workshop on Advances in Sensors and Interfaces (IWASI). pp. 184–188 (2019). https://doi.org/10.1109/IWASI.2019.8791249
3. Hesselbarth, R., Wilde, F., Gu, C., Hanley, N.: Large scale RO PUF analysis over slice type, evaluation time and temperature on 28nm Xilinx FPGAs. In: IEEE International Symposium on Hardware Oriented Security and Trust (HOST). pp. 126–133 (2018). https://doi.org/10.1109/HST.2018.8383900
4. Hinkley, D.V.: On the ratio of two correlated normal random variables. Biometrika **56**(3), 635–639 (1969). https://doi.org/10.1093/biomet/56.3.635
5. Kodýtek, F., Lórencz, R.: A design of ring oscillator based PUF on FPGA. In: IEEE International Symposium on Design and Diagnostics of Electronic Circuits & Systems (DDECS). pp. 37–42 (2015). https://doi.org/10.1109/DDECS.2015.21

6. Kodýtek, F., Lórencz, R., Buček, J.: Improved ring oscillator PUF on FPGA and its properties. Microprocessors and Microsystems **47**, 55–63 (2016). https://doi.org/10.1016/j.micpro.2016.02.005
7. Kodýtek, F., Lórencz, R., Buček, J.: Three counter value based ROPUFs on FPGA and their properties. Microprocessors and Microsystems **88**, 104375 (2022). https://doi.org/10.1016/j.micpro.2021.104375
8. Maiti, A., Gunreddy, V., Schaumont, P.: A systematic method to evaluate and compare the performance of physical unclonable functions. In: Embedded Systems Design with FPGAs, pp. 245–267. Springer (2013). https://doi.org/10.1007/978-1-4614-1362-2_11
9. Okada, K., Yamaoka, K., Onodera, H.: A statistical gate delay model for intra-chip and inter-chip variabilities. In: Asia and South Pacific Design Automation Conference (ASP-DAC). pp. 31–36 (2003). https://doi.org/10.1109/ASPDAC.2003.1194989
10. Pelgrom, M.J., Duinmaijer, A.C.: Matching properties of MOS transistors. In: European Solid-State Circuits Conference (ESSCIRC). pp. 327–330 (1988). https://doi.org/10.1109/ESSCIRC.1988.5468276
11. Rioul, O., Solé, P., Guilley, S., Danger, J.L.: On the entropy of physically unclonable functions. In: IEEE International Symposium on Information Theory (ISIT). pp. 2928–2932 (2016). https://doi.org/10.1109/ISIT.2016.7541835
12. Schaub, A., Danger, J.L., Guilley, S., Rioul, O.: An improved analysis of reliability and entropy for delay PUFs. In: Euromicro Conference on Digital System Design (DSD). pp. 553–560 (2018). https://doi.org/10.1109/DSD.2018.00096
13. Staníček, O., Kodýtek, F., Lórencz, R.: Counter power leakage for frequency extraction of ring oscillators in ROPUF. In: Euromicro Conference on Digital System Design (DSD). pp. 26–32 (2024). https://doi.org/10.1109/DSD64264.2024.00013
14. Suh, G.E., Devadas, S.: Physical unclonable functions for device authentication and secret key generation. In: Design Automation Conference (DAC). pp. 9–14 (2007). https://doi.org/10.1145/1278480.1278484
15. Tebelmann, L., Danger, J.L., Pehl, M.: Interleaved challenge loop PUF: A highly side-channel protected oscillator-based PUF. IEEE Transactions on Circuits and Systems I: Regular Papers **69**(12), 5121–5134 (2022). https://doi.org/10.1109/TCSI.2022.3208325
16. Tuyls, P., Schrijen, G.J., Škorić, B., van Geloven, J., Verhaegh, N., Wolters, R.: Read-proof hardware from protective coatings. In: Cryptographic Hardware and Embedded Systems - CHES 2006. pp. 369–383 (2006). https://doi.org/10.1007/11894063_29
17. Weste, N.H.E., Harris, D.: CMOS VLSI Design: A circuits and systems perspective (4th Edition). Pearson India (2011)