IEICE Electronics Express, Vol.17, No.24, 1–6

LETTER

An efficient and stable composed entropy extraction method for FPGA-based RO PUF

Jin Li[1], Lei Li[1, a)], Ji Yang[2], Yuanhang He[3], Wanting Zhou[1], and Shiwei Yuan[1]

Abstract The physical unclonable function (PUF) of the ring oscillator (RO) is applied to key generation and other fields due to its excellent physical characteristics and simple implementation on FPGA. However, the traditional frequency comparison method uses the sign bit of two ROs' frequency difference which can only extract one bit of entropy, and the bit error rate (BER) of the response always exceeds 1% without error correction schemes. In this paper, we designed RO based on FPGA LUT unit in a single CLB, and proposed a method of getting difference after summing the first-order frequency difference, which we called Difference on Summed Difference (DSD) method. According to experimental measurement results, the DSD method can achieve the BER of 0.39%, and the uniqueness of 50.30%. In order to obtain more entropy, we proposed a composed entropy extraction method which combined the DSD method with the existing higher-order difference method. Experimental results demonstrated that the composed method totally obtained 32-bit response with the BER of 0.84% and the uniqueness of 49.15% while the BER of the existing higher-order difference method is 1.85%.

Keywords: PUF, ring oscillator, entropy, FPGA
Classification: Integrated circuits (memory, logic, analog, RF, sensor)

1. Introduction

The physical unclonable function (PUF) refers to the conversion of the physical characteristics introduced by the process deviation during the chip manufacturing process into a specific challenge-response function. Since the process deviation in different chips is completely random, the generated challenge-response relationship is also unique. PUF has multiple characteristics such as unclonability, uniqueness, unpredictability, light weight and tamper resistance, so it is used in multiple hardware security fields such as identity authentication [1, 2], implementation of property rights protection [3, 4], key generation [5, 6], and device authentication [7]. Since the concept of physical unclonable function was first proposed in 2002 [8], delay-based PUF such as arbiter PUF [9], ring oscillator(RO) PUF [10, 11, 12], glitch PUF [13] and storage-based PUF such as SRAM PUF [3], butterfly PUF [14], latch PUF [15] and flip-flop PUF [16] have also been proposed. Compared with other types of PUFs, RO PUF has better cryptographic characteristics and does not require high symmetry which makes it easier to implement on FPGA. Therefore, RO PUF has attracted the attention of many researchers [17, 18, 19].

Fig. 1 shows the basic principle of RO PUF. The ROs are composed of an odd number of inverters with the same structure and connected to an AND gate. Due to the difference in manufacturing processes, each inverter unit has some delay difference, hence two ROs with the same structure will generate different oscillation frequencies. A one-bit response output can be generated by comparing their frequencies.

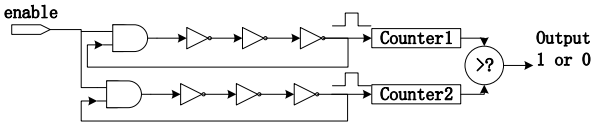


Fig. 1 The basic structure of RO PUF [10]

Maiti [12] pointed out that RO will be affected by environmental factors such as temperature and voltage, which will result in unstable response output. On the other hand, system differences caused by the internal position of FPGA will lead to a fixed position distribution of frequency, which weakens the randomness of PUF. In fact, the frequency deviation caused by environmental factors such as temperature is roughly the same for different PUFs [12], hence the pairwise comparison frequency structure proposed by Suh and Devadas [10] can largely eliminate the influence of environmental factors. In order to further reducing the error rate of the output bits, Suh and Devadas proposed a 1-of-k scheme, selecting the pair with the largest frequency difference from k ROs for comparison to ensure the stability. Mati [11] proposed a configurable RO method in 2009, adding a multiplexer between the two inverters. The multiplexer was configured to select different delay paths, and more challenge-response pairs were generated. At the same time, Mati reduced the system error by comparing adjacent RO units. Amsaad [20] and Xin [21] further optimized the structure of configurable RO PUF, which greatly improved the utilization of hardware resources and produced more output bits. Dodis [22] et al. proposed a fuzzy extractor structure based on an error correction algorithm to reduce the impact of noise. In addition, Cui [23] proposed a structure in which multiple MUXs were connected to an inverter, and at the same time, a method of frequency self-comparison

[1] Research Institute of Electronic Science and Technology, University of Electronic Science and Technology, Chengdu 611731, China
[2] Unit 78102 of the Chinese People's Liberation Army, Chengdu 610036, China
[3] Science and Technology on Communication Security Laboratory, Institute of Southwest Communication, Chengdu 610041, China
[a)] lilei_uestc@uestc.edu.cn

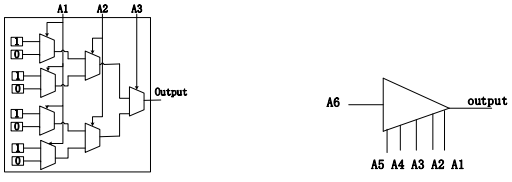
DOI: 10.1587/elex.17.20200350
Received October 19, 2020
Accepted November 5, 2020
Publicized December 4, 2020
Copyedited December 25, 2020

under different configurations of the same RO was used to eliminate system errors. Gan [24] improved this structure and enhanced the stability of the response by adjusting the counter's integration time. The above methods have made a lot of improvement and trade-off work in improving stability, reducing system error and increasing response bit width, but they all bound to increase hardware overhead in varying degrees. In order to obtain more reliable response output with limited hardware overhead, we combined existing higher-order difference method with our DSD method and proposed a composed entropy extraction method. Moreover, we improved the structure of RO, which made full use of CLB resources in FPGA. Based on the method of adjusting the LUT of FPGA [25] to produce a variety of frequency output proposed by Majzoobi [26] et al. and the second-order difference method of frequency used by Zhang [27], the structure of RO PUF is improved in this paper, and RO circuit is constructed in a single CLB. The RO frequencies were subtracted under different configurations to produce a first-order difference result, which eliminated the system difference caused by position factors. Then, the higher-order difference and the proposed DSD method were performed on the first-order difference result to eliminate the deviation caused by environmental noise and got the response. The composed entropy extraction method proposed in this paper uses two CLB resources to generate 32-bit output, and the experimental test on Kintex-7 FPGA proved that it can achieve the BER of 0.84% and the uniqueness of 49.15% at 27°C.

2. The proposed RO PUF structure and entropy extraction method

2.1 RO structure based on LUT and its delay model

Inside the FPGA, LUTs can be used to construct inverters. As shown in Fig. 2 (a), it is a schematic diagram of a three-input LUT, which is composed of a preset storage unit and multiple selectors. After the $A_3A_2A_1$ input configuration, the stored value can be output through different paths. Fig. 2 (b) plots a six-input LUT composed of two LUT5. In Kintex-7 FPGA chip, each slice unit contains 4 six-input look-up tables (LUT6), 3 data selectors (MUX), 1 carry chain and 8 flip-flops. One input A_6 of the LUT was used as the signal input terminal, and A_5~A_1 was used as the configuration port. When $A_5A_4A_3A_2A_1$=00000~11111, the preset value of the storage unit was configured so that the output of the LUT was the inverse of the input. As shown in Fig. 3, eight LUTs in the two CLBs were configured as an AND gate and seven inverters, two D flip-flops are used to divide the frequency by four to prevent measurement errors



(a) (b)

Fig. 2 The component of an inverter cell, (a) LUT3; (b) LUT5

caused by excessive frequency. Although the logic functions of different configurations were the same, the signal transmission path was different, and the transmission delay was therefore different, hence a total of 32 frequency outputs can be generated.

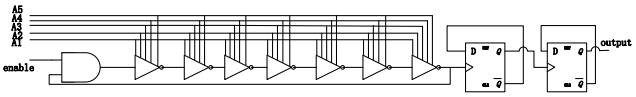


Fig. 3 The structure of a 7-stage RO.

In [24], Maiti established the following delay model of the oscillation loop circuit:

$$d_{LOOP} = d_{AVG} + d_{RAND} + d_{SYST} \tag{1}$$

In the above formula, d_{AVG} represents the average delay of ROs, which is the same for every RO. d_{RAND} represents random manufacturing differences, which is determined by process deviations during chip manufacturing, and d_{SYST} represents system differences caused by different positions of the ROs on FPGA [28].

In order to describe the oscillation loop delay more accurately, we introduced the noise delay d_{NOISE} caused by environmental factors.

$$d_{LOOP} = d_{AVG} + d_{RAND} + d_{SYST} + d_{NOISE} \tag{2}$$

For the same RO under different input configurations, since the same group of LUTs are used, the system difference d_{SYST} can be considered the same. The difference between the j+1th configuration and the jth configuration of the lth RO is:

$$\Delta d_{LOOP}(\mathrm{l}, j) = d_{LOOP}(l, j+1) - d_{LOOP}(l, j) \tag{3}$$

It can be seen from the Eq. (2) and Eq. (3) that the first-order difference for the same RO under different configurations eliminates the influence of system differences, and the delay is only determined by process deviation and environmental noise. Zhang [27] pointed out that the process deviation parameters of n ROs under the same LUT input d_{RAND}(1,j),d_{RAND}(2,j)...d_{RAND}(n,j) obeyed Gaussian distribution. Assuming that the LUT input is j, the standard deviation of d_{RAND} is σ_j:

$$d_{RAND}(l, j) \sim N(\mu_{\mathrm{j}}, \sigma_j^2) \tag{4}$$

As shown in Fig. 4, an RO was tested 200 times under the same LUT input, and the distribution of its noise parameters obey the Gaussian distribution with a mean value of 0.

Then distribution of manufacturing variance Δd_{RAND} and noise variance Δd_{NOISE} can be obtained as:

$$\Delta d_{RAND}(l, j) \sim N(\mu_{RANDj}, \sigma^2_{RANDj}) \tag{5}$$

$$\begin{aligned} \Delta d_{\mathrm{NOISE}}(l, j) &= d_{\mathrm{NOISE}}(l, j+1) - d_{\mathrm{NOISE}}(l, j) \\ &\sim N\left(0, 2\sigma^2_{\mathrm{NOISE}}\right) \end{aligned} \tag{6}$$

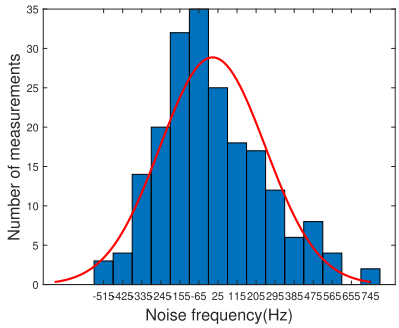


Fig. 4 Noise frequency distribution diagram of 200 mesurements

2.2 Entropy extraction method

2.2.1 Higher-order difference method

Fig. 5 shows the higher-order differential entropy extraction method proposed by Zhang [27] and Li [29]. Taking RO-1 and RO-2 as an example, the 31-bit response can be extracted by this method.

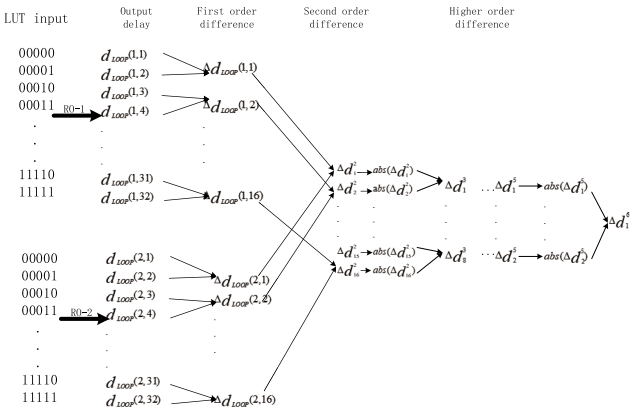


Fig. 5 Higher order difference method [29]

The first-order difference results of RO-1 and RO-2 $\Delta d_{LOOP}(1,1)$ and $\Delta d_{LOOP}(2,1)$ contain the random manufacturing difference Δd_{RAND} and the noise difference Δd_{NOISE} which obey the Gaussian distribution.

$$\Delta d_1^2 = \Delta d_{LOOP}(2,1) - \Delta d_{LOOP}(1,1) \tag{7}$$

Since both $\Delta d_{RAND}(2,1)$ and $\Delta d_{RAND}(1,1)$ obey the same Gaussian distribution, $\Delta d_{RAND}(2,1) - \Delta d_{RAND}(1,1)$ obeys a Gaussian distribution with a mean of 0:

$$\Delta d_{RAND}(2,1) - \Delta d_{RAND}(1,1) \sim N\left(0, 2\sigma_{RAND1}^2\right) \tag{8}$$

As [12] pointed out that in a certain measurement, the impact of environmental factors on each RO is roughly the same, therefore the method of making the difference between ROs can effectively eliminate the influence of environmental noise, that is, $\Delta d_{NOISE}(i+1,j) - \Delta d_{NOISE}(i,j) \approx 0$.

At this point, 16-bit response has been obtained, and it can be concluded that the second-order difference method obeys a Gaussian distribution with a mean of 0, which makes the probability of occurrence $d_i^2 > 0$ and $d_i^2 < 0$ equal. To extract more entropy, Li [29] extended the difference method to higher orders. In order to avoid the low-order comparison result from leaking the higher-order entropy, the absolute value of the low-order difference was calculated first and then the higher difference was made. For two ROs with 32 configurations, six-order difference can be calculated, and a total of 31 differential outputs can be generated.

2.2.2 Proposed method

The above analysis is based on the assumption that different ROs in the same measurement are affected identically by environmental noise, but the actual situation is that the impact of noise on each RO is still slightly different. The higher-order difference result is rather small which makes the result vulnerable to environmental influence, hence the output will be awfully unstable. In order to further extract more stable entropy, this paper proposed a method of summing and then subtracting based on the first-order difference results, which we called DSD method. We can extract additional eight bits of information from the original data.

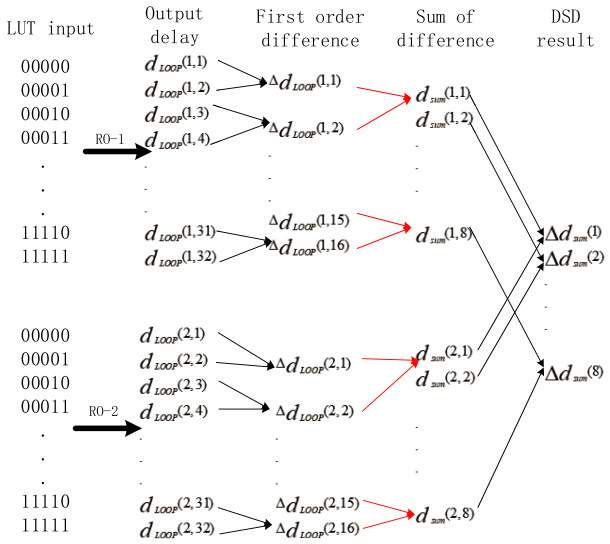


Fig. 6 Proposed DSD method

As shown in the Fig. 6, we first made summation rather than second-order difference on the first order calculation.After summation, the base of frequency was enlarged, which weakened the environmental influence of noise. We then made subtraction.

$$d_{\mathrm{SUM}}(1,1) = \Delta d_{\mathrm{LOOP}}(1,2) + \Delta d_{\mathrm{LOOP}}(1,1) \tag{9}$$

$$\Delta d_{\mathrm{SUM}}(1) = d_{\mathrm{SUM}}(2,1) - d_{\mathrm{SUM}}(1,1) \tag{10}$$

From Eq. (4), it can be deduced that both $d_{SUM}(1,1)$ and $d_{SUM}(2,1)$ obey the Gaussian distribution with the mean of $\mu_{RAND1} + \mu_{RAND2}$, assuming that the variance was δ_{SUM1}^2, namely:

$$d_{\mathrm{SUM}}(l,1) \sim N\left(\mu_{\mathrm{RAND1}} + \mu_{\mathrm{RAND2}}, \sigma_{\mathrm{SUM1}}^2\right) \tag{11}$$

$$\Delta d_{\mathrm{SUM}}(1) \sim N\left(0, 2\sigma_{\mathrm{SUM}}^2\right) \tag{12}$$

As conducted by Eq. (11) and Eq. (12), it is easy to deduce that the DSD result $\Delta d_{SUM}(i)$ (i=1,2,3,...,8) obeys the Gaussian distribution with mean 0. Same as the second-order difference method [27], the proposed DSD method also eliminates the influence of noise and system variation. Besides, the proposed DSD method get better stability.

Based on the above two entropy extraction methods, 31 higher-order differential frequencies and 8 DSD frequencies can be obtained respectively. By Eq. (13), these frequency differences can be converted into 32-bit and 8-bit responses. Besides, the correlation between response bits was eliminated. For the stability requirements, the 32-bit response produced by the high-order differential method is not completely reliable. Through experimental analysis, this paper only used the second-order(16 bits) and the third-order(16 bits) difference result as part of final response so that the output had a lower BER.

In total, after combining the above two methods, 32-bit effective responses can finally be extracted.

$$r = \begin{cases} 1 & if\ \Delta d \geq 0 \\ 0 & else \end{cases} \tag{13}$$

3. Experimental program and result analysis

3.1 Introduction to the experimental program

The scheme designed in this paper is shown in Fig. 7. The control module realizes the functions of measurement state conversion, module start and stop. RO array includes 28 RO units, which are the source of frequency signals, and the frequency calculation module uses FPGA on-chip clock (200MHz) to integrate the frequency signal. The UART module is responsible for sending and receiving data from the PC. Matlab software can directly receive or send data from the serial port. We controlled the PUF to generate measurement data through TX passage and to stored it as a txt file through RX passage. After the measurement was completed, we used the Matlab program to read the data file for further analysis and processing.

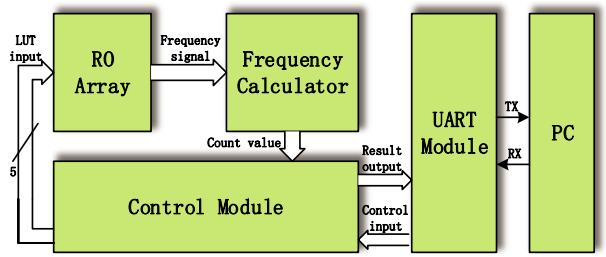


Fig. 7 Experimental architecture scheme

In order to verify the effectiveness of our method, we deployed RO PUFs on six Xilinx Kintex-7 series FPGA development boards. We used the Hard Macro technology of the Vivado development tool to transplant the layout of one RO to the remaining ROs, ensuring that each RO had the same layout. We instantiated 28 ROs on FPGA, and each RO was constrained in a CLB which was composed of two Slice L units. Every two ROs were laid in the same Clock Region and generated a set of outputs. Totally, we generated 14 RO PUF entities on each FPGA, and then measured 200 times at 27°C. All the frequency data was transmitted to PC through the UART interface, the Matlab receiver program captured the data. We performed data processing of our proposed method by Matlab program.

3.2 Experimental results and analysis

3.2.1 Stability

Stability means that the measurement results of a PUF entity under the same challenge input should be consistent in different environments. The stability performance can be measured by average intra-Hamming distance [27], the Hamming distance between two strings of equal length is the number of positions at which the corresponding symbols are different. The formula is as follows:

$$\mu_{\text{intra}} = \frac{1}{m}\sum_{j=1}^{m}\frac{HD\left(R_i, R_{i,j}\right)}{n} \times 100\% \tag{14}$$

Where m represents the number of measurements, n represents the response bit width, HD represents the Hamming distance function, R_i represents the standard response output in a noise-free environment, and $R_{i,j}$ represents the response output of the jth measurement under actual conditions. The concept of average intra-chip Hamming distance is equivalent to the response BER, and its ideal value is 0%.

Fig. 8 shows the relationship between the BER and the difference order of the higher-order difference method. It can be seen that the BER increases as the order of the difference increases, and the result of the fourth-order difference has even exceeded 4%. This performance obviously does not meet the requirements for higher stability applications, so we only used the second and third order differential results to ensure higher stability.

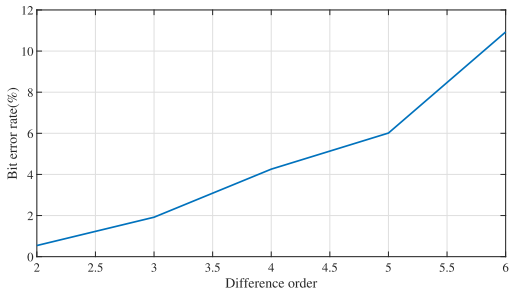


Fig. 8 The relationship between stability and difference order

Table I The intra-HD comparison

Method	BER (intra-HD)
Higher order difference [29]	2.06%
Proposed DSD method	0.39%
Composed entropy extraction method	0.84%

Table I lists the BER of the entropy extraction methods. The BER of DSD method is only 0.39%. And combined with the higher-order difference method, the BER is only 0.84%, which is much better than the existing schemes. Fig. 9 shows the experimentally measured intra-HD distribution of the composed entropy extraction method.

3.2.2 Uniqueness

Uniqueness means that the responses between different PUF entities should be independent, which can be measured by the iner-Hamming distance [27] as Eq. (15). k represents the number of RO PUFs, and the ideal value of inter-Hamming

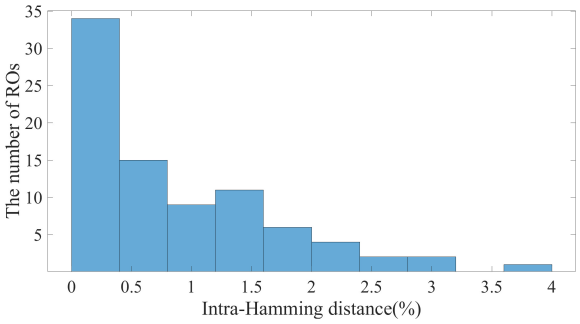


Fig. 9 The intra-Hamming distance distribution of experiment

distance is 50%. Table II lists the iner-Hamming distance comparison of different result. The combined method has a inter-Hamming distance of 49.15% that close to the ideal value. Fig. 10 shows the experimentally measured distribution of inter-Hamming distance of the composed entropy extraction method.

$$\mu_{inter} = \frac{2}{k(k-1)} \sum_{i=1}^{k-1} \sum_{j=i+1}^{k} \frac{HD\left(R_i, R_j\right)}{n} \times 100\% \quad (15)$$

Table II The inter-HD comparison

Method	inter-HD
Higher order difference [29]	48.71%
Proposed DSD method	50.30%
Composed entropy extraction method	49.15%

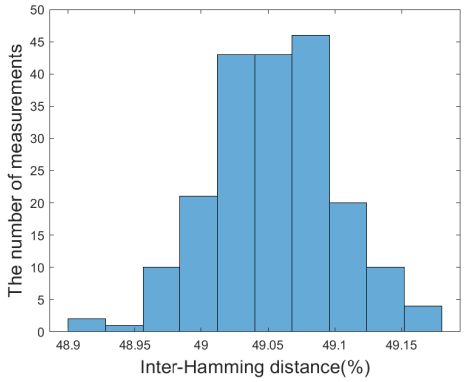


Fig. 10 The inter-Hamming distance distribution of experiment

3.3 Comparison and summary of experimental results

In Table III, we have summarized several important parameters of RO PUF and compared the composed entropy extraction method with some other existing methods. It can be seen that the BER of previous work exceeded 1% which is not reliable enough in applications such as key generation. Compared with Zhang's [27] result, our proposed method not only eliminates the correlation between adjacent bits, but also achieves a lower BER (only 0.84%) which can meet higher-reliability requirement. Meanwhile, 16-bit effective response can be extracted from per RO on average and each RO only consumes 8 LUTs while other work [11, 21, 27, 30] get less than 16 response bits from per RO which means lower entropy extraction efficiency. Therefore, we went further based on Zhang's work and got better performance than others in terms of resource consumption.

Table III The quality factors comparisons

quality factors	Zhang [27]	Pei [30]	Xin [21]	Maiti [11]	Proposed
Uniqueness	49.32	50.01	40	45.9	49.15
BER	1.85	1.12	1.02	3.15	0.84
LUT	16	8	16	16	8
bit per RO	15.5	1	1	4	16

4. Conclusion

In this paper, we proposed an RO PUF structure based on LUT with low hardware overhead. In order to extract more entropy from the frequency information, we combined the higher-order difference method with the proposed DSD method to extract a total of 32 bits of effective information. Through experimental testing on the Kintex-7 FPGA development board, the proposed composed entropy extraction method can achieve the BER of 0.84% and the uniqueness of 49.15%, and we can extract 16-bit response from each RO on average.

Acknowledgments

This work is supported by the National Defense Science and Technology Key Laboratory Fund (No. 6142103190310) and National Natural Science Foundation of China NSAF Joint Fund (No. U1630133).

References

[1] A. Mostafa, *et al.*: "Physical unclonable function and hashing are all you need to mutually authenticate IoT devices," Sensors **20** (2020) 4361 (DOI: 10.3390/s20164361).
[2] P. Gope, *et al.*: "Lightweight and physically secure anonymous mutual authentication protocol for real-time data access in industrial wireless sensor networks," IEEE Trans. Ind. Informat. **15** (2019) 4957 (DOI: 10.1109/TII.2019.2895030).
[3] J. Guajardo, *et al.*: "FPGA intrinsic PUFs and their use for IP protection," CHES 2007 **4727** (2007) 63 (DOI: 10.1007/978-3-540-74735-2_5).
[4] X.T. Ngo, *et al.*: "Cryptographically secure shield for security IPs protection," IEEE Trans. Comput. **66** (2017) 354 (DOI: 10.1109/TC.2016.2584041).
[5] S.K. Cherupally, *et al.*: "A smart hardware security engine combining entropy sources of ECG, HRV, and SRAM PUF for authentication and secret key generation," IEEE J. Solid-State Circuits **55** (2020) 2680 (DOI: 10.1109/JSSC.2020.3010705).
[6] Z. He, *et al.*: "Reliable and efficient PUF-based cryptographic key generator using bit self-tests," Electronics Letters **56** (2020) 803 (DOI: 10.1049/el.2020.0344).
[7] S. Buchovecká, *et al.*: "Lightweight authentication and secure communication suitable for IoT devices," ICISSP (2020) 75 (DOI: 10.5220/0008959600750083).
[8] R. Pappu, *et al.*: "Physical one-way functions," Science **297** (2002) 2026 (DOI: 10.1126/science.1074376).
[9] J.W. Lee, *et al.*: "A technique to build a secret key in integrated circuits for identification and authentication applications," 2004 Symposium on VLSI Circuits. Digest of Technical Papers (IEEE Cat. No.04CH37525) (2004) 176 (DOI: 10.1109/VLSIC.2004.1346548).
[10] G.E. Suh and S. Devadas: "Physical unclonable functions for device

authentication and secret key generation," 44th ACM/IEEE Design Automation Conference (2007) 9 (DOI: 10.1109/DAC.2007. 375043).
[11] A. Maiti and P. Schaumont: "Improving the quality of a physical unclonable function using configurable ring oscillators," 19th International Conference on Field Programmable Logic and Applications (2009) 703 (DOI: 10.1109/FPL.2009.5272361).
[12] A. Maiti and P. Schaumont: "Improved ring oscillator PUF: an FPGA-friendly secure primitive," J. Cryptology **24** (2011) 375 (DOI: 10.1007/s00145-010-9088-4).
[13] D. Suzuki and K. Shimizu: "The glitch PUF: a new delay-PUF architecture exploiting glitch shapes," CHES 2010 **6225** (2019) 366 (DOI: 10.1007/978-3-642-15031-9_25).
[14] S.S. Kumar, *et al.*: "Extended abstract: the butterfly PUF protecting IP on every FPGA," 2008 IEEE International Workshop on Hardware-Oriented Security and Trust (2008) 67 (DOI: 10.1109/HST.2008.4559053).
[15] Y. Su, *et al.*: "A 1.6pJ/bit 96% stable chip-ID generating circuit using process variations," ISSCC 2007 (2007) 406 (DOI: 10.1109/ISSCC.2007.373466).
[16] R. Maes, *et al.*: "Intrinsic PUFs from flip-flops on reconfigurable devices," WISSec 2008 **17** (2008) 2008.
[17] N. Tianming, *et al.*: "Research on physical unclonable functions circuit based on three dimensional integrated circuit," IEICE Electron. Express **15** (2018) 20180782 (DOI: 10.1587/elex.15.20180782).
[18] A.S. Chauhan, *et al.*: "Novel randomized placement for FPGA based robust RO PUF with improved uniqueness," Journal of Electronic Testing-Theory and Applications **35** (2019) 581 (DOI: 10.1007/s10836-019-05829-5).
[19] E. Avaroğlu: "The implementation of ring oscillator based PUF designs in field programmable gate arrays using of different challenge," Physica A: Statistical Mechanics and its Applications **546** (2020) (DOI: 10.1016/j.physa.2020.124291).
[20] F. Amsaad, *et al.*: "A novel security technique to generate truly random and highly reliable reconfigurable ROPUF-based cryptographic keys," HOST 2016 (2016) 185 (DOI: 10.1109/HST.2016.7495580).
[21] X. Xin, *et al.*: "A configurable ring-oscillator-based PUF for Xilinx FPGAs," 2011 14th Euromicro Conference on Digital System Design (2011) 651 (DOI: 10.1109/DSD.2011.88).
[22] Y. Dodis, *et al.*: "Fuzzy extractors: how to generate strong keys from biometrics and other noisy data," International Conference on the Theory and Applications of Cryptographic Techniques (2004) 523 (DOI: 10.1007/978-3-540-24676-3-31).
[23] Y. Cui, *et al.*: "Low-cost configurable ring oscillator PUF with improved uniqueness," ISCAS (2016) 558 (DOI: 10.1109/ISCAS.2016.7527301).
[24] J. Gan, *et al.*: "A FPGA-based RO PUF with LUT-based self-compare structure and adaptive counter time period tuning," ISCAS (2018) 1 (DOI: 10.1109/ISCAS.2018.8351014).
[25] J.-L. Zhang, *et al.*: "Techniques for design and implementation of an FPGA-specific physical unclonable function," J. Comput. Sci. Technol. **31** (2016) 124 (DOI: 10.1007/s11390-016-1616-8).
[26] M. Majzoobi, *et al.*: "FPGA-based true random number generation using circuit metastability with adaptive feedback control," CHES 2011 (2011) 17 (DOI: 10.1007/978-3-642-23951-9-2).
[27] Q. Zhang, *et al.*: "FRO PUF: how to extract more entropy from two ring oscillators in FPGA-based PUFs," International Conference on Security and Privacy in Communication Systems (2016) 675 (DOI: 10.1007/978-3-319-59608-2-37).
[28] Y. Yu, *et al.*: "Improving RO PUF design using frequency distribution characteristics," IEICE Electron. Express **12** (2015) 20141043 (DOI: 10.1587/elex.12.20141043).
[29] C. Li *et al.*: "FRO PUF: extract more entropy from FPGA-based oscillating ring PUF," Journal of Cyber Security **1** (2018) 16 (DOI: 10.19363/j.cnki.cn10-1380/tn.2018.01.002).
[30] S. Pei, *et al.*: "A low-overhead RO PUF design for Xilinx FPGAs," IEICE Electron. Express **15** (2018) (DOI: 10.1587/elex.15.20180093).