Circuit Authentication based on Ring-Oscillator PUFs

Susana Eiroa and Iluminada Baturone
Dept. Electronics and Electromagnetism (University of Seville)
Microelectronics Institute of Seville (IMSE-CNM-CSIC)
Seville, Spain
{eiroa, lumi}@imse-cnm.csic.es

***Abstract*—The use of Ring Oscillator PUFs to provide circuit authentication is analyzed in this paper. The limitations of the previously reported approach in terms of false rejection (due to high intra-die variations) and false acceptance (due to small inter-die variations) are discussed. These limitations are overcome by a new proposal that does not increase considerably hardware complexity and, besides, provides lower power consumption and/or higher speed to achieve high security requirements. All these issues are illustrated with experimental results obtained with FPGAs from Xilinx.**

I. Introduction

Nowadays, it is needed to authenticate not only the person that uses a device (using secret keys, passwords, etc.) but also the device itself. This means that both the software and hardware of the device (its circuitry) should be authentic. Circuit authentication consists in verifying the trustworthiness of the hardware. This is becoming so important that design-for-trust challenges are being defined (similarly to design-for-test challenges) [1]. The concept of authentication means a 1 to 1 validation, that is, a verifier validates that the circuit is which it claims to be. Similarly, circuit identification is a 1 to N validation process because, in this case, a verifier identifies which, out of the N possible circuits, the circuit is. Both procedures require a pre-verification stage usually called “enrollment” in which the unique and distinctive features (also called “template”) of the circuit have to be stored. In the case of authentication, only the template of the circuit to be authenticated must be stored while in the case of identification the N templates must be recorded. In any case, the key point is to obtain a good template that allows distinguishing an authentic circuit from a fake one (impostor). A usual solution is to employ a template consisting in a digital number, also known as identification number or ID number [2]-[3].

Silicon Physical Unclonable Functions (PUFs) have been proposed as a cost-effective way to produce identifiers that exploit the random variability of the circuit fabrication process [5]. Exploiting the power consumption variability in different realizations of the same circuit, leakage-based PUFs have been proposed [2]. The different leakage current consumption of each circuit, which is an analog number, is translated into a digital ID number. However, such translation is complex and requires a cost both in hardware as in power consumption, which may modify the ID itself. Memory-based PUFs such as SRAM and butterfly PUFs are based in the different start-up values of cross-coupled circuits [4]. The use of memory-based (cross-coupled NOR gates) PUFs for ID creation is described in [3]. This solution is very efficient in terms of readout speed and power consumption, but suffers from the lack of reliability, which can be improved by increasing the length of the ID, making a long number of readouts to reduce the noise, or increasing the signal to noise ratio of the circuit evaluation process. The delay variability in different physical realizations of the same circuit is exploited by arbiter and ring oscillator (RO) PUFs [6]. The basis for using these delay-based PUFs for circuit authentication is presented in [6]. In particular, the use of ring oscillator (RO) PUFs is receiving attention for its simplicity, reliability, and uniqueness [7]. Its main drawbacks are power consumption and dependability on, mainly, power supply variations [3], [8].

This paper proposes a new method to provide circuit authentication based on RO PUFs reducing the problems of power consumption and power supply variations. The paper is organized as follows. Section II reviews the basic RO PUF scheme employed for hardware authentication and illustrates its limitations with experimental results obtained with Spartan 3 FPGAs from Xilinx. Section III presents a new approach that provides better performance for authentication. This is also illustrated with experimental results. Finally, conclusions are given in Section IV.

II. Using RO PUFs For Authentication

A. Basic RO PUF structure

Fig. 1 illustrates the delay PUF based on Ring Oscillators firstly proposed by Su et al. in [6]. The structure is composed of a group of identically laid-out ring oscillators. This way, the slight difference frequency of each ring oscillator is due to manufacturing variation of the physical device where the PUF is included. The procedure to obtain bit strings from these elements is to compare the frequencies between pairs of ring oscillators. The output bits from the same sequence of

This work was partially funded by Junta de Andalucía under the Project P08-TIC-03674 (with support from the PO FEDER-FSE de Andalucía 2007-2013), by the European Community through the MOBY-DIC Project FP7-INFSO-ICT-248858 (www.mobydic-project.eu), and by Spanish Ministerio de Ciencia y Tecnología under the Project TEC2008-04920.

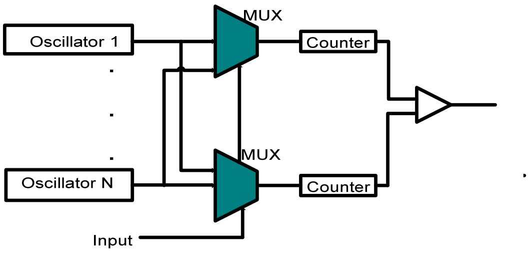

Figure 1. Ring Oscillator PUF presented in [6].

oscillator pair comparisons will vary from chip to chip. The circuit determines the difference between frequencies by using two counters that measure the number of oscillations along some fixed amount of time. The counter that counts higher amount of periods corresponds to the ring oscillator with higher frequency. In order to determine the output of the PUF structure, a comparator is placed after the counters, in such a way that the resultant bit is '1' if the upper oscillator in the floorplan has higher frequency than the lower one, and '0' in other case. If *n* pairs are compared the bit string (ID number) obtained has *n* bits.

The ID number generated in the authentication stage is compared with the template stored in the enrollment stage. Since they are digital numbers, such comparison is performed by calculating the Hamming Distance (HD). In order to allow certain tolerance, a threshold is usually defined, so that, if the difference is below or equal to the threshold, the circuit is authenticated while otherwise it is considered as impostor. A measure to determine the quality of an authentication process is to evaluate the values of the False Acceptation Rate (FAR) and False Rejection Rate (FRR). The ideal situation is to find a threshold so that both FAR and FRR are zero. In the one side, the ideal situation is that the ID number generated by the PUF is always the same (the PUF is completely reliable). Hence, the threshold can be selected as zero and a circuit is rejected if the HD with the template is bigger than zero. However, the PUF is not completely reliable. Such reliability is measured by the average of intra-die HD, as follows (being *x* the number of samples):

$$Intradie\_dist = \frac{1}{x}\sum_{j=1}^{x}\frac{dist(R_i, R'_{i,j})}{n}\times 100 \qquad (1)$$

Due to noise, temperature or power supply variations, the same pair of oscillators in the same device may output the opposite value in the authentication stage to that registered in the enrollment stage ('1' instead of '0' or vice versa). This problem, known as *bit flipping*, causes that the intra-die distance increases, so that a threshold of zero cannot be selected because the FRR could be large.

In the other side, the ideal situation is that the ID numbers generated by different circuits are very much different (the PUF achieves uniqueness). Such uniqueness is measured by the average of inter-die HD, as follows (being *m* the number of circuits compared):

$$Interdie\_dist = \frac{2}{m(m-1)}\sum_{i=1}^{m-1}\sum_{j=i+1}^{m}\frac{dist(R_i, R_j)}{n}\times 100 \qquad (2)$$

The model used in [7] to analyze the total delay in a ring oscillator (which determines the behavior of the ring oscillator PUF) is the following:

$$d_{RO} = d_{AVG} + d_{PV} + d_{noise} \qquad (3)$$

The delay d_{AVG} is the nominal delay of the ring oscillator, which depends on its components (more or less inverters, gates, etc.) and how they are distributed and interconnected. It is the same for all identically laid-out oscillators. The delay d_{PV} is due to process variation. It may vary from one oscillator to other but it is static, that is, it is assumed to be constant over time in a given physical realization (neglecting possible ageing effects). The delay d_{noise} represents a noisy and dynamic component that changes over time. The effect of noise can be removed by counting a big number of intervals because the media of the noise is assumed to be zero. Hence, if noise is removed and identical oscillators are compared, the differences in frequencies only depend on the fabrication process of the chip that includes the PUF. Assuming that variations in the fabrication process are random, this means that the average inter-die HD is ideally 50%.

However, the response of the Ring Oscillator PUF depends on the rest of the circuitry in the system, as described in [8]. This is mainly caused by the sensitivity of oscillation frequency to power supply variations within the chip. Hence the model in (3) should be refined as follows (neglecting noise):

$$d_{RO} = d_{AVG} + d_{PV} + d_{SYSTEM} \qquad (4)$$

Due to the influence of the system, it is quite common that responses of the PUF in different devices show *bit aliasing*, that is, there are bits in the ID numbers of different devices that always take the same value. The consequence of this phenomenon is a decrement in the value of the average inter-die distance. The problem is that if the inter-die distance decreases and the intra-die distance increases, the authentication process can fail, as illustrated in the following.

B. *Experimental Results*

In order to obtain the FAR and FRR curves and the distribution of the genuine and fake population for the basic scheme of RO-based authentication described above, a sort of measurements have been performed. A PUF made with a matrix of 32 identical Ring Oscillators has been implemented into a sort of XC3S200 Spartan 3 FPGAs from Xilinx. The matrix has been placed in the center of the device, with the oscillators placed as close as possible from each other in order to avoid deterministic gradients in the variations of the fabrication process. Each ring oscillator has 4 inverters and 1 NAND gate that serves to enable it. It occupies one CLB (Configurable Logic Block), with the same occupation of slices (the placement is controlled by the synthesis) so as to ensure identical oscillators. The frequency of the enabled oscillator is measured by comparing the results of its associated counter with the count of a reference counter working at the board frequency of 50 MHz. The counter associated to the oscillators has 32 bits. It is stopped when the reference counter counts 2^{15} system clock cycles, what makes a total of 65.53 ms. The reason for using this long count time is to ensure noise removal so as to evaluate frequency changes due only to variations in the fabrications process.

Since the PUF behavior depends on the system where it is included, the PUF has not been studied alone (as reported in other works) but has been analyzed inside a whole security system, similar to that described in [9]. The system contains the PUF structure, a pseudo-random number generator of 32 bits based on non-linear feedback shift registers, and a short version of Keccak [b=400] sponge hash function with line width of 8 bits and 18 rounds [10]. In order to avoid the possible influences of changes in the system, a unique bit stream has been generated from the VHDL code of the whole system. This has been done using ISE environment provided by Xilinx. The objective is to measure only the possible variations due to the fabrication process. The same bit stream has been loaded into 8 different FPGAs and the output values of the counters that measure the frequencies of the 32 oscillators have been recorded 8 times for each FPGA.

From each matrix of 32 oscillators, an ID number with 28 bits is obtained because 28 comparisons are performed. This ID is compared with the rest of IDs obtained from other measurements of the same FPGA (intra-die data) and from other FPGAs (inter-die data). The comparisons performed have been the following:

- Intra-die data: The amount of intra-die comparisons for each device is 28. This is the number of different pairs that can be formed with 8 measurements of each device. As they are 8 devices, the genuine population of the authentication process is formed by 8*28=224 samples.

- Inter-die data: Each measurement of each device is compared with the 8 samples of the rest of the devices (64 comparisons between a pair of devices). Since 28 different pairs can be considered, the resulting comparisons form a population of 64*28=1792 samples. This forms the impostor population.

Fig. 2 shows the Hamming Distance with their templates of the genuine and impostor populations. The average intra-die HD is 2.5% instead of 0% while the average inter-die HD is 30.62% instead of 50%. The maximum value of the intra-die HD is 14.29% while the minimum value of the inter-die HD is 10.71 %. Hence, as there is overlapping between both populations, there would be always a small probability of error in the authentication whatever the threshold is chosen. This effect appears clearly represented in Fig. 3. The FAR and FRR overlap showing the same value (EER, equal error rate) of 0.45% for a threshold of 14.2%. Such EER value can vary from 0.45% to 2% easily if there are variations in the power supply, temperature, etc.

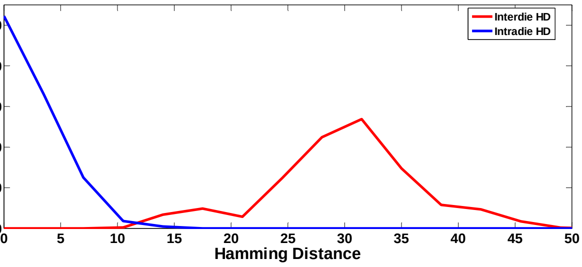

Figure 3. Genuine (in blue) and impostor (in red) populations distributed versus the Hamming Distance with their templates

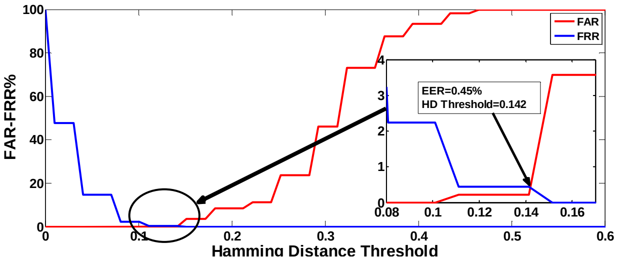

Figure 2. FAR and FRR traditional approach

The usual practice to avoid this problem is to employ more ring oscillators to obtain more bits (64, 128, and 256 ring oscillators are employed in [7]). The problem of such solution is the increase in power consumption. Using the Xilinx Xpower tool provided in the ISE environment, each oscillator shows a power consumption of around 20 µW. Another drawback can be the time needed to generate the ID number (which increases with the number of oscillators if bits are generated serially). Hence, the generation of a large ID number could be power hungry, noisy, slow, and possibly more vulnerable to side channel attacks. To avoid such problems, the novel proposal discussed in the following is not to employ more oscillator pairs but more bits per pair.

III. A Novel RO PUF-Based Authentication

The approach described above codifies the frequency difference between two oscillators with just one bit, because it is only codified the sign of such difference. A drawback of such coarse quantization is, for example, bit aliasing: variations in the fabrication process can be masked by the influence of the surrounding elements to the PUF. Such influence produces variations in the V_{dd} that feeds the oscillators in such a way that one of the oscillators in the pair can be always faster than the other in all the devices [8]. The novel scheme proposed herein is to use the complete resolution of the frequency difference measurement. By using more bits of the counter that counts oscillations, better discrimination between devices can be obtained. In other words, the authentication is improved by using an adder-subtractor instead of a comparator to measure the frequency difference between oscillator pairs.

If the counter associated with a ring oscillator counts *osc_count* oscillations during the 2^{15} clock cycles of the

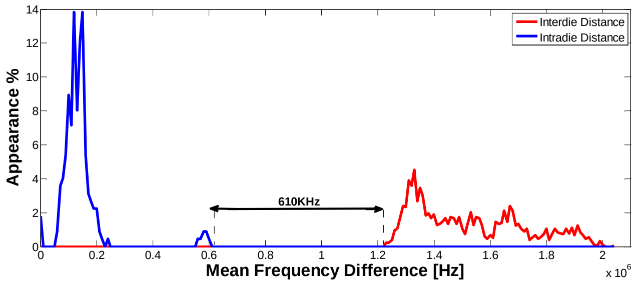

Figure 4. Genuine and impostor distribution in the proposed approach

reference counter, the frequency of the ring oscillator, *frec_RO*, can be calculated as follows:

$$frec\_RO = osc\_count * \frac{50MHz}{2^{15}} \qquad (5)$$

Subtracting two counts (*osc_count_A* and *osc_count_B*), the frequency difference between two oscillators (*diff*) can be measured as follows:

$$diff = (osc\_count\_A - osc\_count\_B) * \frac{50MHz}{2^{15}} \qquad (6)$$

These measurements have been evaluated with the same FPGA samples described in the previous section. The maximum value of *diff_frec* obtained for all the pairs has been 15.6 MHz. This means that:

$$(osc\_count\_A - osc\_count\_B)_{\max} = \frac{15.6 * 2^{15}}{50} \qquad (7)$$

Hence, 14 bits are enough to codify each *diff*. Considering, as in the previous section, 28 oscillator pairs, the ID number now has 28*14=392 bits (which means a template size of 49 bytes). The distance between the ID number generated in the authentication and that must be stored in the enrollment stage is calculated as:

$$mean\_diff = \frac{1}{28}\sum_{i=1}^{28}\left|diff_i - diff\_template_i\right| \qquad (8)$$

Fig. 4 shows the distribution of the 224 samples of the genuine population and the 1792 samples of the impostor population versus such distance. The FAR and FRR are represented in Fig. 5. The dispersion of the genuine population is smaller than in the previous approach. Only several samples force to use a threshold of 0.61 MHz to achieve a FRR of zero. Even though, the FAR is zero. The proposed approach presents a security area of 0.61 MHz, among the genuine and false population. This represents a margin of 25.5% of the universe of discourse of the possible distances. Hence, it is possible not only to obtain a system with a FAR and FRR equal to zero by choosing a threshold in the range of 0.61 MHz to 1.22 MHz, but also if the threshold is selected with a value in the middle of this interval, it is possible to allow a variance of 0.305 MHz in both the genuine and the false populations while the FAR and FRR are still zero. This makes the authentication proposal robust against noise, temperature, and V_{dd} variations.

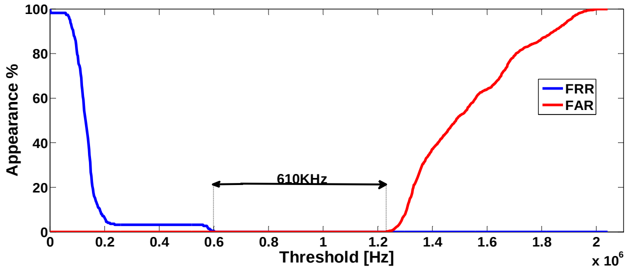

Figure 5. FAR and FRR in the proposed approach

IV. CONCLUSIONS

The new approach for hardware authentication based on RO PUFs improves the results of previously reported approaches because it achieves not only false acceptance rate (FAR) and false rejection rate (FRR) of zero per cent but also enough error margin to be robust against noise, temperature, and V_{dd} variations. In addition, it requires a smaller number of ring oscillators, which means that authentication is performed with less power consumption and higher speed. All these advantages are obtained with no substantial increment in the size and complexity of the hardware required. This has been illustrated with experimental results from FPGAs from Xilinx.

REFERENCES

[1] M. Tehranipoor et al., “Trustworthy hardware: trojan detection solutions and design-for-trust challenges”, D.O.I. 10.1109/MC.2010.369, Computer, 2011.

[2] Y. Alkabani, F. Koushanfar, N. Kiyavash, and M. Potkonjak. “Trusted integrated circuits: A nondestructive hidden characteristics extraction approach”, Lecture Notes in Computer Science, Springer-Berlin, vol. 5284, pp. 102-117, 2008.

[3] Y. Su, J. Holleman, and B. Otis, “A digital 1.6 pJ/bit chip identification circuit using process variations,” IEEE Journal of Solid-State Circuits, vol. 43, no. 1, pp. 69–77, Jan. 2008.

[4] S. S. Kumar, J. Guajardo, R. Maes, G.-J. Schrijenand, P. Tuyls, ”The butterfly PUF protecting IP on every FPGA", in Proc. 1st IEEE Int. Workshop on Hardware-Oriented Security and Trust, HOST, 2008.

[5] I B. Gassend, D. E. Clarke, M. van Dijk, and S. Devadas. “Silicon physical unknown functions”, in Proc. ACM Conf. on Computer and Communications Security, CCS 2002, pp. 148–160.

[6] G.E. Suh and S. Devadas,”Physical unclonable functions for device authentication and secret key generation", in Proc. Design Automation Conference, DAC 2007.

[7] A. Maiti, and P. Schaumont, “Improved ring oscillator PUF: an FPGAfriendly secure primitive”, Journ. of Cryptology, Vol. 4 (2), pp. 375-397, April 2011.

[8] S. Eiroa, I. Baturone,” An Analysis of Ring Oscillator PUF Behavior on FPGAs”, submitted to Int. Conf. on Field Programmable Technology, FPT 2011.

[9] S. Eiroa, I. Baturone, “Hardware authentication based on PUFs and SHA-3 2nd round candidates” International Congress of Microelectronics, ICM 2011, Cairo.

[10] The Keccak sponge function family website: http://keccak.noekeon.org/