An Arbiter PUF employing eye-opening oscillation for improved noise suppression

Andreas Herkle, Joachim Becker, Maurits Ortmanns
Institute of Microelectronics, University of Ulm, Ulm, Germany
Email: andreas.herkle@uni-ulm.de

***Abstract*—Like every integrated circuit, Arbiter-PUFs suffer from any ambience variations such as electrical noise, supply voltage variations and temperature fluctuations. In this work, we show that most bit-errors are related to small phase differences, for which noise dominates the readout bit value. We present an approach eliminating this influence by modifying the arbitration circuit part into an eye-opening oscillator. By utilizing the deadzone of two D-Flip-Flops, the decision about the response is delayed until the phase difference becomes significant enough. With this modification, we could increase the PUFs initial bit-error rate of 3.31% to almost zero. We also highlight important design choices for this solution, like the setup-time of the D-Flip-Flops and the minimum number of enforced oscillations.**

I. Introduction

The Internet-Of-Things (IoT) is becoming an increasingly prominent keyword in the last few years, promising fully-connected households and consumer electronics easing everybody's daily life. However, many incidents involving hacked IoT devices have shown that nothing connected to the internet is safe, making hardware embedded security a more relevant topic than ever before. For secure communication and authentication, devices need to hold some kind of secret information to be used in a cryptographic application.

Therefore, the so-called Physical Unclonable Function (PUF) has gained immense interest by the scientific community over the last two decades. A PUF is a storage-free, low-power device used as security primitive, whose functionality is to generate random but stable bit-strings, which is a unique hardware fingerprint of the device. One PUF's individual behavior only depends on small manufacturing variations, making its readouts unpredictable and unclonable.

The Arbiter-PUF, c.f. Fig. 1, is a prominent and well-studied example for the class of "Strong PUFs" which offer a challenge-response-pair (CRP) capability, altering their response based on a given multi-bit challenge input [1]. The working principle of an Arbiter-PUF is a multiplexed signal race, where a rising edge is simultaneously applied to two delay paths, which can be swapped by the different challenge bits. The individual delays of buffers on these paths introduce a phase shift between the two signals. A metastable RS-Latch, the original arbitration element at the PUFs output, then determines the faster edge output and provides either a digital '1' or '0' as the response. In this paper, we propose a technique to deal with noise in the arbiter element by employing an eye-opening comparator for the decision. The paper is organized as follows: Section II introduces some common problems of state-of-the-art arbiter improvements. In Section III, we present the working principle of our solution, which will be analyzed and evaluated in Section IV. Section V concludes the paper.

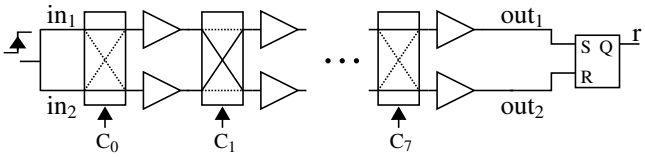

Fig. 1. Standard Arbiter PUF with RS-Latch as arbiter

II. Noise sensitivity of state-of-the-art noise cancelling techniques

A. Drawback of an RS-Latch arbiter

Two system parts dominantly contribute to the quality of the PUFs readouts under environmental influence: the multiplexed delay lines and the sensing unit. Prior work dealt with the reduction of PVT and noise influence on the delay units themselves [2], [3]. Although the quality of these reductions helps in reducing the bit-error rate (BER), the remaining influence of noise has to be eliminated by the arbitration element and improving it will be the focus in this work. To introduce the main problem, Fig. 2 illustrates two possible arbitration curves for a NAND-based RS-Latch.

The dashed lines show transistor level simulations in which the two signals out_1 and out_2 of the arbiter PUF in Fig. 1 arrive by more than 5 ps separated from each other. This large phase difference ends the RS-Latch metastable state quite fast within around 16 ps and ensures a correct decision. In contrast, the solid lines represent the case where the phase difference

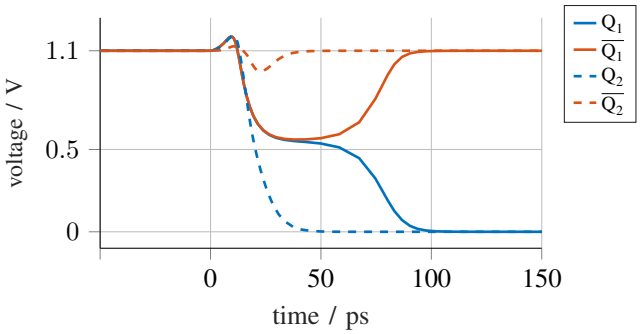

Fig. 2. RS-Latch decision behavior for largely separated input edges (dashed) and close edges (solid)

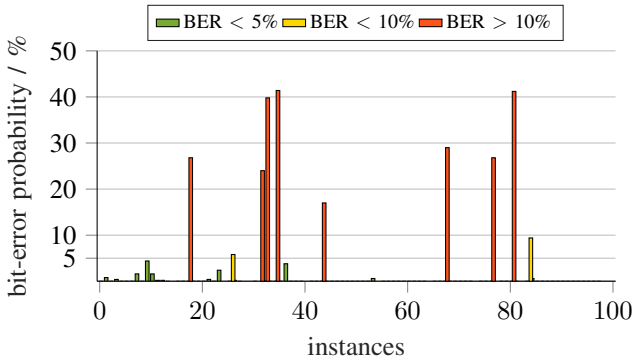

Fig. 3. Noise induced bit-flips of the response for a standard Arbiter-PUF

is a mere 10 fs. In this case, both outputs Q and $\overline{Q}$ drop to near $V_{DD}/2$ and the settling time increases to over 70 ps. In this noise-free simulation, the output Q still settles for the correct decision. But in presence of noise, the response during the metastable phase is highly susceptible to influence of the transient noise and thus may randomly flip. In our simulated example, the resulting bit-error rate increases from 0% for a phase difference of 5 ps to over 25.5% for 0.1 ps in presence of circuit noise.

Fig. 3 shows simulated error bars for an 8-bit Arbiter-PUF with an RS-Latch as arbitration element. A random but fixed challenge was applied and the 100 Monte Carlo instances with each having 500 transient noise runs averaged were simulated in a 40nm CMOS technology. It becomes quite clear that most of the instances are minorly affected by noise and most often tend to the correct value because the applied challenge results in a significant phase difference in the two competing delay lines. Yet, a none negligible portion suffers from the small phase difference. The BER of these instances amounts to 29.8%, which totally dominates the overall averaged exemplary BER of 3.3%.

B. State-of-the-art high-precision arbitration and averaging

To overcome the influence of noise on a PUFs responses, previous work tackled the problem in many different ways. Most often, supplementary or different arbitration circuitry was proposed, e.g. sense amplifiers [4]. Likewise, improved circuit techniques have been utilized like adaptive voltage ramp-up time [5] and custom sized structures [6]. Others employed additional post-processing such as temporal majority voting (TMV) and bit masking [7]. However, those approaches have their own significant drawbacks. E.g. a sense amplifier is able to differentiate a very small voltage difference, which equals the phase difference between two racing signals in this context. Nevertheless, that phase difference might already be "flipped" while racing through the PUF circuit and thus will still produce a bit error. On the other hand, the number of averaged readouts for a correct decision by TMV significantly increases with the required BER and may require hundreds of samples, which either highly increases circuit area or readout time, or reduces the averaging efficiency for less samples.

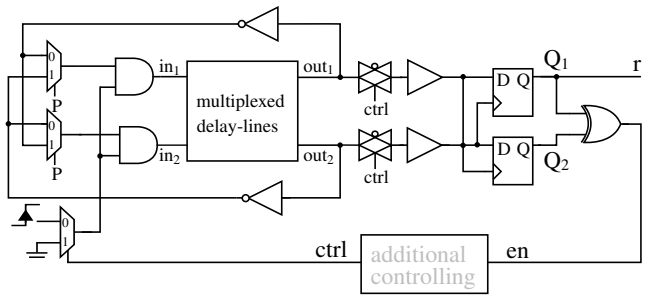

Fig. 4. Proposed eye-opening Arbiter-PUF with deadzoning Flip-Flops, controlled feedback paths and additional controlling (gray)

III. Proposed solution

A. Working principle

The proposed solution utilizes the principle of an eye-opening, oscillation based comparator [8]. In the original work, two ring-oscillators are current starved by two voltage inputs, and the phase difference of the VCOs increases over time, such that at a certain moment a safe decision can be taken. This is because the noise is averaged out, while the signal influence on the phase difference integrates over time. In contrast to [8], in the proposed implementation the phase difference is caused by inherent PUF mismatch instead of a voltage difference between two analog inputs. The principle of the new arbitration is illustrated in Fig. 4: the unaltered delay-lines with multiplexers are illustrated as a black box, which are the circuits source of randomness.

The PUF outputs are connected to two D-Flip-Flops, out_1 to D and out_2 to CLK of the first D-FF and vice versa for the second D-FF. If the racing signals arrive with a significant large temporal difference, the D-FF, where the faster path is connected to, recognizes the signal as data, which will then be sampled to a digital '1'. The second Flip-Flop sees this path as its clock input and thereby samples the other path as data, which in this case is still '0'. However, if both the racing edges arrive within a small time frame, the setup-time condition of each D-FF is violated and both will output a digital '0'. This effectively creates a so-called deadzone [8], which can be used to prevent a premature decision based on signals not separated clearly enough.

The idea is to use this deadzone detector to automatically repeat the measurement. For this, the delay PUF is reconfigured into a feedback loop, where the output signals of both competing delay lines are fed back to their respective inputs, letting them repeatedly race against each other. Thereby, each additional race starts with an increasing phase difference, which then will be summed over multiple oscillations. Similar to the original publication, this oscillation scheme suppresses the influence of accumulated noise, as it is averaged out due to its zero-mean nature. Feeding back the signal effectively closes a loop and, for a static CRP, alters the behavior to one comparable to a ring-oscillator or the Loop PUF [9], depending on the stability of the decision.

In order to save power and accelerate the PUF response, the oscillation is stopped when a clear decision has been

made. Therefore, a control unit is added in Fig. 4, which consists of two parts. The first part is a standard XOR, whose inputs are the outputs Q_1 and Q_2 of the D-FFs. Its sole purpose is to merge these two outputs into a control signal which differentiates if the phase difference between the delay lines is within the deadzone or not. In the latter case, the XORs' output pulls the second input of the PUFs preceding ANDs to a digital '0', effectively stopping the oscillation by breaking the loop. Additional transmission-gate switches after the PUFs output prevent an unfinished signal race from further influencing the D-FFs. The final response is taken as Q_1. It should be noted that by stopping the oscillation very fast, the reconfigured arbiter PUF does not get prone to side-channel attack by tracking the oscillation frequency, as it can be done for common ring-oscillator PUFs [10].

B. Feedback and oscillation considerations

An arbiter PUF switches the race between two competing delay lines. When the feedback is being closed, it must be assured that the output of one configured delay line is fed again to the input of the same. This is because each racing signal should be affected by only one of the two competing paths through the PUF. Additional switch elements are instantiated to correctly feedback the upper output signal to the upper input path and vice versa. These switches are controlled based on a pre-calculated parity P, depending on the number of multiplexer stages.

Another problem of the proposed reconfiguration arises from the feedback loop reconfiguration itself. After propagating an e.g. rising edge through the PUF delay line, the signal needs inversion to be fed back in order to get oscillation; this inversion is implemented with the additional inverter in the feedback. By that, also falling edges are propagated through the paths once the loop is configured, which then experiences a different mismatch variation due to the buffers unequal propagation delay characteristics. For a phase difference, which is large enough to never trigger the oscillation, this is no problem. Further, it is also unproblematic for very small phase differences that will always trigger the oscillation. But the variability of the delay becomes a significant problem for phase differences close to the deadzone, where noise is the main trigger for the oscillation. In this case, the response of consecutive readouts for the same CRP could happen on non-static mismatch parameters, because now the oscillation would sometimes be triggered and sometimes not during the readout. This again would increase the BER.

To tackle this problem, the control logic has been extended by an additional chain of N Flip-Flops connected as shown in Fig. 5. The idea is to let the oscillation run for several cycles until the phase difference is safely outside the deadzone. The implementation works as follows: The previous shutdown signal from the XOR now acts as the asynchronous, active-low reset for the Flip-Flops. The clocking signal can be one of the two PUFs outputs out_1 or out_2 in order to ensure an operation synchronous to the rest of the circuit. The other output of the PUF should then be connected to an additional dummy

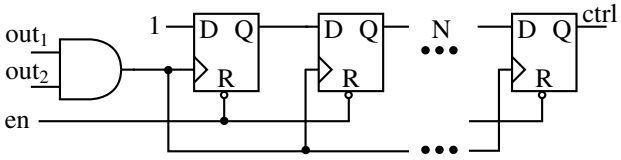

Fig. 5. New control unit employing N shifting D-FFs, shutting down the oscillation only if the phase difference is out of the deadzone for at least N cycles

buffer of the same size in order to avoid the introduction of bias by different loading capacitances. If the XORs output is high, a logical '1' is shifted consecutively through the Flip-Flops after each successful decision of the two D-FFs. Hence, a final decision is not taken before the phase difference was outside the deadzone for at least N cycles. Note that for typical arbiter length of ≥ 128 stages, the additional circuitry for the proposed architecture is very minor.

IV. PUF Simulation results and further design considerations

A. Design of the deadzone

We implemented the proposed design in a 40nm CMOS technology and present the results based on transient noise Monte Carlo-simulations. A PUF delay line as shown in Fig. 1 was used as the source of randomness, with minimum sized buffers and transmission gates switches. In order to design the new arbitration circuit correctly, one has to establish two different characteristic parameters: the deadzone width and the minimum/maximum number of oscillations. The deadzone width depends only on the Flip-Flops capability of recognizing a rising edge on the data input in time. Consequently, D-FFs setup-time defines the positive and negative edge of the deadzone. In our case, we employed standard D-MS-FF with NAND-based RS-Latches triggered on the rising edge. Most of the transistor gate dimensions were left on minimum size with a W/L-ration of 2:1 in order to keep an digital switching threshold of $V_{DD}/2$. With these dimensions, the deadzone edges were simulated to be at $\pm$14ps, which turned out to be unnecessarily large in comparison to a standard RS-Latch as arbiter.

Fig. 6 shows the progress of phase separation of the two competing delay lines for an arbitrary challenge (y-axis) over repeated oscillations (x-axis) for a few exemplary Monte Carlo instances. It is quite obvious that some circuit instances (orange, blue) will take an unreasonably long time to leave such a large deadzone. It can also be seen that for some cases after a few oscillations, the phase shift's polarity flips due to the different propagation delays, as mentioned before. Consequently, the deadzone was adjusted by rescaling the NAND-gates preceding the master RS-Latch in the D-Flip-Flops. These NAND-gates are controlled by the clock input and switch the master latch into the hold condition on a rising edge, a process which mainly specifies the setup-time. More specifically, both inputs of the RS-Latch are initially low and therefore the initial condition of the output is charged. An incoming high signal, either on the data or clock input, has

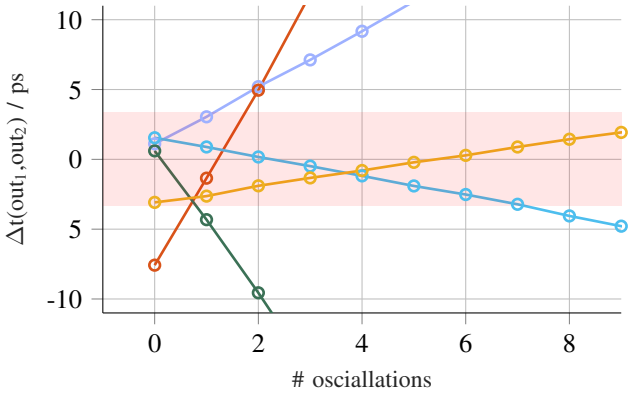

Fig. 6. Drift of the phases difference over multiple oscillations and instances, deadzone after circuit modification highlighted in red

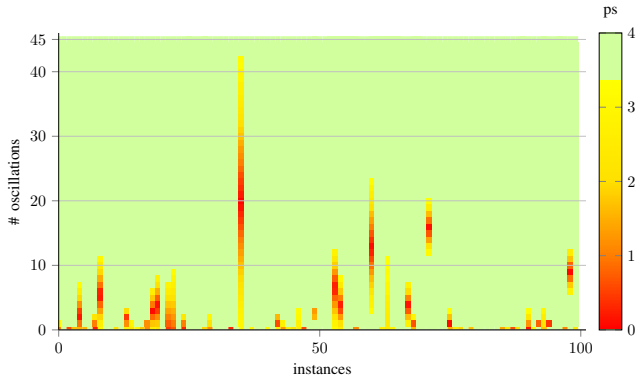

Fig. 7. Trend of entering the deadzone for multiple devices

to discharge the R or S inputs gate capacitance. To speed-up this discharging process, the channel width of the NAND-gates NMOS-transitors was set to 4 times the minimum width, drastically increasing its discharge capability and effectively shortening the D-FFs setup-time. By this modification, the new deadzone width (marked as the red area in Fig. 6) was designed to be $\pm$3.35ps.

B. Simulation results and analysis

Fig. 7 shows the trend of phase differences from 100 simulated PUF instances. These simulations were done initially without noise influence in order to extract the precise trend boundaries. The y-axis lists the number of oscillations, the color gradient from red to yellow indicates the phase difference ”depth” within the deadzone while the green area marks the condition ”out of deadzone”. The trend shows that many devices start in the deadzone, but leave it after few oscillations. However, one instance oscillates 43 times and some others even drop into the deadzone later without originally starting there. These instances resembles extreme variants of the exemplary curves in Fig. 6, but could obviously occur in reality.

These extreme examples disclose two system level questions, which can only be answered roughly in general: what is the minimum number of oscillations which should be enforced and what is the maximum time after a decision needs to be done? In our presented example, N as the number of minimum oscillations has been chosen as 6, at which point all devices either never entered the deadzone or entered it at least once. Although a higher number might give the impression that it will decrease the BER even further, this is a counter-intuitive thought as the phase shift will invert at some point and the faster signal will overtake the other signals previous iteration. The number of maximum oscillations can either be infinity, which means one has to wait an undefined time for a final decision. Or it can be set to a fixed value, thereby omitting undecided responses by marking them as unsolvable, also known as dark bit masking [11]. Both questions have been answered for our implementation but need to be evaluated separately for each different Arbiter-PUF implementation [12], higher numbers of challenge bits and different technologies.

Our presented implementation has been evaluated by noisy simulations over 100 Monte-Carlo sampled device instances and including 500 transient noise runs. A maximum number of 50 oscillations was set to ensure that the mismatch is significant enough to enable a clear decision. The overall BER of all these instances was reduced to a minor 0.009211% with a standard deviation of 0.076516% per device. The statistics of the oscillations without an enforced number was 1.7 oscillations in average with minimum of 0 (direct decision after first race) and a maximum of 17. Including enforced oscillation, these numbers changed to 6.4 in average and a maximum number of 39. This change again highlights our observation that some devices should be intentionally driven into the deadzone as their phase difference after the first race might lead to a false positive decision. The influence on the inter-Hamming distance was evaluated separately and found to be 49.3%, showing no negative influence on the diversity of the PUF responses.

V. Conclusion

In this work, we presented the adaptation of an eye-opening oscillator as arbiter for Arbiter-PUFs. We have shown substantial inherent problems with the phase difference between two racing signals and showed that a sole increase of precision or post-readout averaging is not sufficient to get error-free results. By utilizing the setup-time of two differently connected Flip-Flops as a deadzone, the decision for small phase difference can be postponed. With the application of feedback, the PUF can be modified to oscillate long enough such that the desired deviation by mismatch overtakes any noise induced variance. In our exemplary implementation, simulation results showed that the bit error-rate could be reduced to an insignificant 0.009%, which proves the superiority of the concept. The solution is easy to implement and requires only little custom sizing of transistor devices, while the area overhead remains small due to the remaining minimum-sized device parts.

References

[1] J. W. Lee, D. Lim, B. Gassend, G. E. Suh, M. van Dijk, and S. Devadas, “A technique to build a secret key in integrated circuits for identification and authentication applications,” in *2004 Symposium on VLSI Circuits. Digest of Technical Papers (IEEE Cat. No.04CH37525)*, June 2004, pp. 176–179.

[2] R. Kumar, H. K. Chandrikakutty, and S. Kundu, "On improving reliability of delay based physically unclonable functions under temperature variations," in *2011 IEEE International Symposium on Hardware-Oriented Security and Trust*, June 2011, pp. 142–147.

[3] K. Suzuki, K. Miura, and K. Nakamae, "Nbti/pbti tolerant arbiter puf circuits," in *2017 IEEE 23rd International Symposium on On-Line Testing and Robust System Design (IOLTS)*, July 2017, pp. 80–84.

[4] K. Fruhashi, M. Shiozaki, A. Fukushima, T. Murayama, and T. Fujino, "The arbiter-puf with high uniqueness utilizing novel arbiter circuit with delay-time measurement," in *2011 IEEE International Symposium of Circuits and Systems (ISCAS)*, May 2011, pp. 2325–2328.

[5] M. Cortez, S. Hamdioui, V. van der Leest, R. Maes, and G. J. Schrijen, "Adapting voltage ramp-up time for temperature noise reduction on memory-based pufs," in *2013 IEEE International Symposium on Hardware-Oriented Security and Trust (HOST)*, June 2013, pp. 35–40.

[6] A. Herkle, M. Schuster, J. Becker, and M. Ortmanns, "Enhanced arbiter pufs using custom sized structures for reduced noise sensitivity," in *2016 IEEE International Conference on Electronics, Circuits and Systems (ICECS)*, Dec 2016, pp. 568–571.

[7] A. Alvarez, W. Zhao, and M. Alioto, "14.3 15fj/b static physically unclonable functions for secure chip identification with $<2\%$ native bit instability and $140\times$ inter/intra puf hamming distance separation in 65nm," in *2015 IEEE International Solid-State Circuits Conference - (ISSCC) Digest of Technical Papers*, Feb 2015, pp. 1–3.

[8] K. Yoshioka and H. Ishikuro, "A 13b sar adc with eye-opening vco based comparator," in *ESSCIRC 2014 - 40th European Solid State Circuits Conference (ESSCIRC)*, Sept 2014, pp. 411–414.

[9] Z. Cherif, J. L. Danger, S. Guilley, and L. Bossuet, "An easy-to-design puf based on a single oscillator: The loop puf," in *2012 15th Euromicro Conference on Digital System Design*, Sept 2012, pp. 156–162.

[10] D. Merli, J. Heyszl, B. Heinz, D. Schuster, F. Stumpf, and G. Sigl, "Localized electromagnetic analysis of ro pufs," in *2013 IEEE International Symposium on Hardware-Oriented Security and Trust (HOST)*, June 2013, pp. 19–24.

[11] S. Satpathy, S. K. Mathew, V. Suresh, M. A. Anders, H. Kaul, A. Agarwal, S. K. Hsu, G. Chen, R. K. Krishnamurthy, and V. K. De, "A 4-fj/b delay-hardened physically unclonable function circuit with selective bit destabilization in 14-nm trigate cmos," *IEEE Journal of Solid-State Circuits*, vol. 52, no. 4, pp. 940–949, April 2017.

[12] R. Kumar, V. C. Patil, and S. Kundu, "Design of unique and reliable physically unclonable functions based on current starved inverter chain," in *2011 IEEE Computer Society Annual Symposium on VLSI*, July 2011, pp. 224–229.