The Bistable Ring PUF: A New Architecture for Strong Physical Unclonable Functions

Qingqing Chen[*‡]
[*]Institute for Electronic Design Automation
Technische Universität München
Munich, Germany
qingqing.chen@tum.de

György Csaba[†‡]
[†]Department of Electrical Engineering
University of Notre Dame
Notre Dame, IN, United States
gcsaba@nd.edu

Paolo Lugli
[‡]Institute for Nanoelectronics
Technische Universität München
Munich, Germany
lugli@tum.de

Ulf Schlichtmann
Institute for Electronic Design Automation
Technische Universität München
Munich, Germany
ulf.schlichtmann@tum.de

Ulrich Rührmair
Institute for Security in Information Technology
Technische Universität München
Garching, Germany
ruehrmai@in.tum.de

Abstract—**This paper introduces a new architecture for circuit-based Physical Unclonable Functions (PUFs) which we call the Bistable Ring PUF (BR-PUF). Based on experimental results obtained from FPGA-based implementations of the BR-PUF, the quality of this new design is discussed in different aspects, including uniqueness and reliability. On the basis of the observed complexity in the challenge-response behavior of BR-PUFs, we argue that this new PUF could be a promising candidate for Strong PUFs. Our design shows noticeable temperature sensitivity, but we discuss how this problem can be addressed by additional hardware and protocol measures.**

Keywords-physical cryptography; field-programmable gate array; physical unclonable function; bistable ring PUF; identification; authentication; process variations; strong PUF

I. Introduction

The idea of using objects for identification that bear manufacturing variations which even the manufacturer cannot control has existed for a long time [1, 2]. However, the formalization of this concept has just been introduced some ten years ago [3, 4]. Once called Physical One-Way Functions [3, 4], and later Physical Random Functions [5], the concept is now known as Physical(ly) Unclonable Functions (PUFs). Despite the fact that the discussion of an exact definition is still ongoing [6–8], it can generally be stated that a PUF is a physical function that maps challenges (inputs) to responses (outputs) based on the complex physical phenomena taking place in the PUF structure. It can easily be evaluated but it should be hard to physically clone or characterize a PUF. With these properties, PUFs are able to get rid of some inherent vulnerabilities of conventional cryptography with regard to invasive and non-invasive side-channel attacks [9], and are able to serve in various security applications, e.g., key obfuscation, device identification, and challenge-response authentication, based on their challenge-response pairs (CRPs).

Since the introduction of the PUF concept, a large number of studies on PUFs have been reported in the literature. To name a few, [6, 7, 10] discussed the theoretical foundations of the PUF concept, [11–13] described applications of PUFs, [14, 15] introduced algorithmic measures to combat the unreliability of PUFs, and [16–22] discussed public-key variants of PUFs which extend the application areas of conventional PUFs. Meanwhile, a large number of PUF designs [3, 23–31] have been proposed. Among them, a type of PUFs which we call the electrical intrinsic PUFs has drawn great attention.

Electrical intrinsic PUFs are based on integrated circuits (ICs) and their conventional manufacturing technologies, which provides them with some natural advantages over other types of PUFs, e.g., the optical PUF (non-electrical) [3] and the coating PUF (electrical, but non-intrinsic) [23]. Firstly, electrical intrinsic PUFs do not require the development of special technologies to manufacture and to introduce randomness in their challenge-response behaviors, since they directly use existing IC manufacturing technologies; secondly, they do not require extra measurement devices to convert responses into electrical form which is the most often used data form in security applications; and thirdly, they are small in size, benefiting from the ever-shrinking IC sizes.

Some most often discussed electrical intrinsic PUFs are the Arbiter PUF [24] and its variants [24, 30], the Ring Oscillator PUF (RO-PUF) [25], the SRAM PUF [12], the Butterfly PUF [26], the Flip-flop PUF [27], etc. Arbiter PUFs are the first proposed electrical intrinsic PUFs which possess an exponential (relative to the size of the PUF) number of CRPs. For an Arbiter PUF with a relatively small number of stages, e.g., 64 stages, a complete measurement of all its CRPs within a limited time frame (such as several days to even weeks) is already infeasible. However, since their challenge-response behaviors can be described with a simple additive delay model [24], its CRPs can be predicted with low error rate through machine learning algorithms [6] using just a relatively small proportion of its CRPs that can be measured in a short time. Certain countermeasures (e.g., adding feed-forward stages to the original structure, or XORing multiple arbiter outputs) are vulnerable to improved machine-learning attacks, too, albeit only up to a certain level of complexity of the PUF (e.g., only up to six XORs) [6]. A RO-PUF provides only a quadratic number of CRPs, which probably makes an exhaustive read-out of all its CRPs in a limited time frame feasible. Memory-based PUFs, namely, the SRAM PUF, the Butterfly PUF, and the Flip-flop PUF, possess an even smaller number of CRPs which is proportional to their size. They can be used as so-called Weak PUFs or Physically Obfuscated Keys (POKs) in diverse applications. But they are not usable as so-called Strong PUFs [6], since they do not possess a sufficiently large number of CRPs. This means that the search for new PUF architectures that fulfill the criteria of Strong PUFs is both interesting and relevant.

In this paper we introduce a new PUF architecture which we call the Bistable Ring PUF (BR-PUF). We explain its special properties that may make it a good candidate for Strong PUFs, and present experimental results showing the quality of the BR-PUF based on implementations on Field Programmable Gate Arrays (FPGAs).

The rest of the paper is organized as follows: Sec. II describes the structure and the working principle of our BR-PUF. Sec. III gives a brief introduction of PUF quality measurement and explains the FPGA-based system we have used to implement and characterize the BR-PUFs. Sec. IV presents the experimental results and discusses some special properties discovered from the results. Sec. V summarizes the results and concludes the paper.

II. The Bistable Ring PUF

A. *The Basic Idea: Bistable Rings*

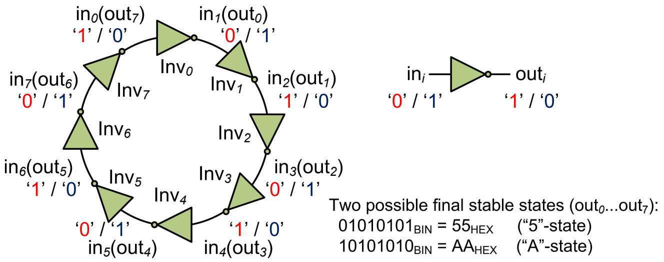


Figure 1. Two possible stable states of an eight-stage bistable ring.

The basic idea of the BR-PUF is based on the fact that an inverter ring consisting of an even number of inverters has two possible stable states. Similar to a static random access memory (SRAM) cell which is based on a pair of cross-coupled inverters, a ring with even number of inverters falls into one of its two possible stable states when powered up or, more generally, when it is released from some unstable state. We call such kind of an inverter ring a bistable ring.

For illustration, Fig. 1 shows a bistable ring with eight inverters Inv_0, Inv_1, …, Inv_7, each of them having an input in_i and an output out_i. When powered up, each of the eight inverters tries to force its output voltage to rise—from an initial logic "0" to a logic "1". However, in a stable state not all input and output signals of the inverters can take the "1" state, since this would contradict the functionality of inverters. Because of the functionality of inverters, the ring can only stabilize in a "01010101" state (clockwise enumeration of inverter outputs starting from Inv_0) or a "10101010" state, in which a "0"/"1" is always followed by a "1"/"0". Therefore, the following more complex process occurs. For any inverter Inv_i, its input in_i keeps changing since its previous stage of inverter forces it. As the in_i voltage increases, the tendency to drive up out_i weakens. If in_i goes beyond a metastable point, where Inv_i shows no preference of driving a "0" or "1" output, Inv_i starts to force out_i to drop, and this tendency gets stronger as in_i increases. Since all eight inverters form a feedback loop, none of them can behave independently—any voltage change in one node would transfer through the whole loop and affect itself again. A simplest situation could be that, e.g., due to the mismatch of the inverters and/or noise, out_0, out_2, out_4 and out_6 rise over their metastable points at the same time (or at very close times), while out_1, out_3, out_5 and out_7 do not reach their metastable points. This creates a positive-feedback situation, in which all inverters help to make the bistable ring converge to the "10101010" state. However, this is often not the case. At some time point, a node voltage may have already crossed the metastable point and may even have been very close to the maximum voltage. A wave coming from its previous stages may still change the situation totally, forcing it to drop to below the metastable point, although it may rise again. In this case a complex feedback situation causes oscillations that may take a long time until the whole ring converges to a stable state. Thus, the bistable ring may show a complex transition process when powered up. Whether it stabilizes in a "01010101" ("5"[1]) or a "10101010" ("A") state or does not stabilize ("X" state) in a reasonable time depends on the process variation mismatch of, e.g., the threshold voltage and carrier mobility of transistors, and noise.

However, up to this point, bistable rings as described do not provide more attractive properties compared to SRAM PUFs, since one bistable ring merely produces a 1-bit output, representing two possible stable states. The next subsection discusses how a more complex behavior can be induced in a bistable ring like structure, leading to our BR-PUF design.

B. *Structure of the Bistable Ring PUF*

To turn a bistable ring as described above into a Bistable Ring PUF that is able to generate an exponential number of CRPs and easy to use, we make the following architectural changes: 1) For each stage, the inverter is duplicated, and a pair of multiplexor (MUX) and demultiplexor (DEMUX) is added to select either of the inverters to be connected in the inverter loop; 2) All the inverters are replaced with 2-input NOR or NAND gates, and the second inputs of the gates are connected to a reset signal that replaces the power up and power off operations.

[1] Since, in practice, the number of stages is usually an integer multiple of 4, the hexadecimal form of the stable state is often a series of "5"s or a series of "A"s. We will refer to these states as a "5" and an "A" state, respectively, in the rest of the paper, and we call an unstable state an "X" state.

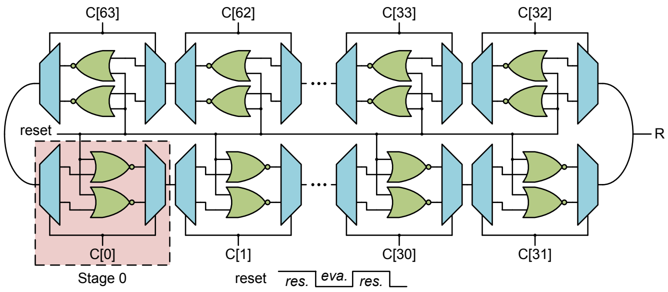


Figure 2. A 64-stage Bistable Ring PUF.

From a bistable ring with 64 inverters, a 64-stage BR-PUF can be created. Fig. 2 shows an example of such a BR-PUF with 64 stages of duplicated NOR gates. In this PUF, each basic stage consists of two NOR gates, a MUX at the gate outputs, and a DEMUX at the gate inputs. For clarity, "Stage 0" is marked with dashed lines in Fig. 2. In each stage, the MUX and the DEMUX share a select signal which is one bit out of a 64-bit challenge signal *C*. This select signal selects either the upper or the lower paths of the MUX and the DEMUX, and therefore connects one of the NOR gates between them (through one of the inputs and the output of the NOR gate) to the previous and the next stages. Since the 64 stages are connected in a ring, when applied with a 64-bit challenge, all the NOR gates selected form a NOR-gate ring. By applying different challenges, different combinations of NOR gates are selected. For the design shown in Fig. 2, a number of 2^{64} different rings can be created.

The reset signal connected to the second input of each NOR gate makes it easier to bring the ring into an all-"0" state, without the need to power off the whole circuit or to cut off the power supply of the PUF circuit before a new measurement. This is done by simply pulling the reset signal high. A high-to-low transition of the reset signal starts an evaluation phase, similar to a power up process of a bistable ring. During the evaluation phase, each NOR gate operates like an inverter, since the reset input is tied to "0". Thus, an inverter ring is formed and is released from an all-"0" state. As is previously discussed, this leads to a stabilization process, and whether it settles at a "5" state or an "A" state or does not settle down within a given period of time depends on process variation mismatch and noise. Since a number of 2^{64} different inverter rings can be created in the PUF shown in Fig. 2, each having a different set of process variation mismatch, 2^{64} different converging processes can be generated, resulting in responses dependent on the mismatch combinations. Applying a rectangular wave as the reset signal makes it possible to generate many CRPs periodically and to evaluate with the same challenge for multiple times.

To output a response (a 1-bit response representing the final state), any node between two basic stages can be used as a readout port. In Fig. 2, the node between Stage 31 and Stage 32 is used as the response output *R*. To measure a CRP, a challenge signal *C* should be given before the evaluation phase and kept stable until the end of the evaluation.

The sequence of operations is listed in the following:

0. $i = 0$;
1. Pull high *reset*;
2. Apply a 64-bit challenge signal C_i;
3. Wait for the bistable ring to be in the all-"0" state;
4. Pull low *reset*;
5. Wait for some time to let the bistable ring stabilize;
6. Read out the 1-bit response R_i;
7. $i = i + 1$; repeat from Step 1 to measure more CRP(s).

C. Physical Layout

Generally, the BR-PUF would require a symmetric layout (similar to Arbiter PUFs) to produce "ideal" performance (discussed in Sec. III), which could become a drawback compared to the RO-PUF that has a relatively loose requirement on the layout. However, Sec. IV will show how quasi-"ideal" performance can be achieved on BR-PUFs without special care of the layout.

III. PUF QUALITY AND EXPERIMENTAL SETUP

A. Quality of PUFs

The measurement of PUF quality has been discussed by a large number of publications [7, 31, 32]. The two most important figures of merit are uniqueness (also known as inter-chip/die Hamming distance) and reliability (also known as intra-chip/die Hamming distance).

Uniqueness measures how unique the CRPs are that a PUF can generate from different chips (dies). It is usually defined by the average of normalized inter-die Hamming distances (HDs; by a Hamming distance, we always mean a normalized one) of the responses to the same set of challenges, measured pair-wise between a set of chips implementing the same PUF design. For a set of *m* chips implementing the same PUF design *P*, with each pair of chips, *u* and *v*, having *n*-bit responses R_u and R_v, respectively, the measure of uniqueness for *P* is therefore defined as

$$\frac{2}{m(m-1)}\sum_{u=1}^{m-1}\sum_{v=u+1}^{m}\frac{HD\left(R_u,R_v\right)}{n}\times 100\% \qquad (1)$$

Reliability measures how reliable/reproducible the CRPs are that a PUF can generate under possibly varying operating conditions. Here the definition of varying operating condition may depend on the application of the PUF. Typical varying operating conditions include temperature, supply voltage and ambient noises. The reliability of PUFs is usually defined by the average of intra-die HDs between reference responses and the responses measured under varying operating conditions, in which the same set of challenges is used. Following the assumptions for the definition of uniqueness, each chip *i* is then measured at the normal operating condition to generate an *n*-bit reference response R_i *to* a set of challenges, and it is then measured for another *k* times with the same set of challenges, generating $R'_{i,j}$ (j=1,2,…,k) for different operating conditions. The measure of reliability for chip *i* is therefore defined as

$$\frac{1}{k}\sum_{j=1}^{k}\frac{HD\left(R_i,R'_{i,j}\right)}{n}\times 100\% \qquad (2)$$

In an ideal case, a set of PUFs with the identical design should give 50% in the uniqueness measurement, meaning that the PUF can provide the highest identifiability. Meanwhile, the ideal value for the reliability measure should be 0%, meaning that the chips can reproduce CRPs without making errors. However, in reality, the ideal values are hard to achieve. PUFs showing uniqueness close to 50% and reliability close to 0% are considered to be PUFs of high quality.

B. *Experimental Setup*

To characterize the BR-PUF, two possibilities exist for a proof-of-concept: one is SPICE simulation, and the other is FPGA-based implementation. However, since the BR-PUF is basically a loop structure and often requires a relatively long time to stabilize, which may cause many more iterations than a normal circuit of the same size, SPICE simulations have been very slow, making it infeasible to generate enough data for statistical analyses. The use of hardware can generate CRPs much faster and the results are also more accurate, whence we implemented and evaluated the BR-PUFs on FPGA resources.

We employed nine (in some experiments, eight) Xilinx Virtex-II Pro FPGA boards for the experiments. While this is a relatively small population compared to industrial scale applications, it can produce sufficient results for a first verification of the proposed idea. For further studies and more accurate results, a larger population will be used in future work, as is done, e.g., in [32].

In order to compare the performance of BR-PUFs with different numbers of stages, we implemented BR-PUFs with 32, 64, and 128 stages.

Since the detailed layout of the FPGAs is proprietary information and is unknown to users, it is hard to predict whether an implementation produces highly layout-biased responses or not. Usually we only realize this when very low inter-chip HDs are observed. Therefore, we used different methods to implement the PUF structure, e.g., use of different on-chip resources (Look Up Tables (LUTs), latches, etc.) to implement the DEMUX. This gives us better chances to obtain a design that is not highly layout-biased.

In the following experiments, the experimental setup varies between experiments, depending on the specific question that is being analyzed. For example, in the experiment that created Fig. 4, each of eight FPGA chips implementing the same 64-stage design was applied with 50,000 random challenges, and each CRP was measured 12 times. During each evaluation phase, the temporary ring state was examined 28 times at different time points, allowing us to observe the stabilization process and obtain an approximate settling time for each measurement. The temporary ring states were transferred to a PC through a transmitter module. To save time, the transmitter only tells if the ring is in a “5” state, an “A” state, or an “X” state. A Finite State Machine (FSM) controls the whole on-board processes described above. On the PC side, the data from FPGA boards are stored in text files and processed by MATLAB programs.

IV. Experimental Results

This section presents the experimental results obtained from the experimental system described in Sec. III.

A. *Settling Time*

To use a PUF, after applying a challenge, one always needs to know the time it requires until the response can safely be measured. For a BR-PUF, this time can be chosen by measuring the settling time of responses. Based on the experimental system described in Sec. III, we were able to measure the approximate settling time of each response with the resolution defined by the cycle time (1.698μs) of the reset signal. For each challenge, we take the time point (out of the 28 time points) as the estimated stabilized time point, where the bistable ring first enters a “5” or an “A” state. By checking all the responses we obtained, it was found that as long as a bistable ring enters a “5” or an “A” state, it does not leave this state again, which proved our previous assumption.

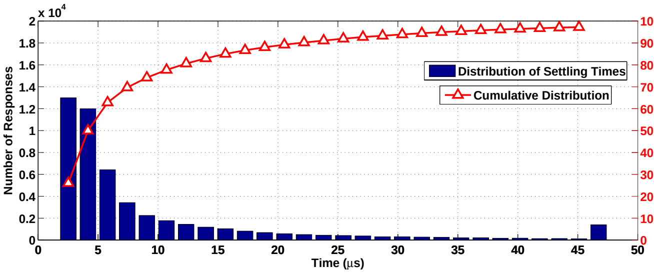


Figure 3. Settling time of a 64-stage BR-PUF.

Fig. 3 shows the distribution of the settling times of 50,000 responses measured from one chip implementing a 64-stage BR-PUF design using LUTs. The rightmost bar represents all the responses that are still unstabilized at the last time point. It can be seen from the figure that the responses of even a single BR-PUF have a broad range of possible settling times. This makes the choice of the evaluation time different from that of an Arbiter PUF, say, which takes a relatively constant time to produce responses. For BR-PUFs, one solution is to read out each response until it stabilizes (an upper time limit may be applied at the same time)—however, this would require some extra circuit to observe the stabilization process; another option is to set a common evaluation time for all CRPs—this may increase the average measuring time, but it does not require an extra circuit to do the observation of the stabilization process. For both solutions, we can choose an upper time limit or a common evaluation time according to the percentage of stabilized responses we want to obtain. In the case shown in Fig. 3, taking the common evaluation time method, to obtain over 90.21% of stabilized responses, 22.17μs is required as the minimum evaluation time for every response.

TABLE I. Settling Time[a] of Bistable Ring PUFs

Nr. of Stages	*32*	*64*	*128*
Max. Settling Time[b] (μs)	8.44	22.25	37.20
Average Settling Time[b] (μs)	5.26	10.78	23.09

a. In this table the settling time is defined as the evaluation time that is required for 90% out of 50,000 responses on a single chip to stabilize. b. The maximum/average settling time is the maximum/average of the settling times measured from eight chips, each with four different designs.

The study of settling times can also help estimating the risks that a fraudster manages to measure a large proportion of the CRPs, if he gains access to a BR-PUF for a short limited period, such as days or even weeks. Tab. 1 shows the maximum and the average settling times (90% of the responses stabilize) of the BR-PUFs with different numbers of stages we implemented on eight chips, each with four different designs that use different FPGA resources. Assuming the average settling time in Tab. 1 is taken as the evaluation time for each challenge, a 32-stage BR-PUF can be fully measured in 6.28 hours (reset phase not counted), which makes it a risky PUF, while a 64-stage BR-PUF requires already 6.31×10^4 years to measure just 1% of all its CRPs, and the measurement of a significant fraction of CRPs of a 128-stage implementation in reasonable timescales is clearly infeasible.

B. Uniqueness and Reliability against Ambient Noises

Since bistable rings do not always stabilize in a predefined period, we may need to deal with those cases, in which the responses do not stabilize, for the estimation of the uniqueness and reliability attributes of the BR-PUFs.

The simplest solution is to ignore the existence of unstabilized states, i.e., to read out the 1-bit state of an arbitrary node in the ring, and use it as the response. By doing this, unstabilized states will not be regarded as special states (although the responses they produce may seem random). Using this method, we calculated the inter- and intra-die HDs, according to the definitions presented in Sec. III. In this experiment, nine FPGA chips were employed, with each implementing a 64-stage BR-PUF. Each chip was applied with 10,000 challenges generated by a random number generator, and each CRP was measured 20 times (each time on a different day at room temperature[2]), taking the responses of the first measurement as the reference responses. The relatively low average inter-die HD of 14.8% showed that the design has been layout-biased. However, the average intra-die HD of 0.8% is also quite low. Considering that the minimum inter-die HD of 6.0% is still clearly larger than the maximum intra-die HD of 1.2%, this result suggested a relatively workable and operational PUF design.

The other solution to the question we brought up at the beginning of this subsection would be to take into account the broad range of the PUFs' settling times and deal with responses differently according to their settling times. As we looked into the relationship between settling times and CRP qualities (uniqueness and reliability), we found that CRPs with very short settling times are often very reliable but are also constant between different chips, while CRPs with longer settling times show more uniqueness of chips but may lose some reliability. This is probably because some systematically biased (e.g., through biased layout) CRPs have relatively strong preference of one of the stable states, making them quickly converge into the preferable state and hard to be affected by process variation or noise; while CRPs that are less biased have weaker preference of a stable state, making them wander between the two stable states. This implies a simple way to separate useless (low inter-die HD) CRPs from useful (high inter-die HD and possibly not very high intra-die HD) CRPs by measuring their settling times.

To verify this, we separately calculated the inter- and intra-die HDs of PUF responses with different settling times based on the experimental setup described in Sec. III. The calculation was based on eight chips implementing a 64-stage BR-PUF design, each applied with 50,000 challenges, and each CRP was measured 12 times. The settling time of each response was measured by observing the ring state at 28 check points during the evaluation phase. In this special computation, each pair of responses needs to agree on a settling time, so that their HD can be included into a group defined by the same settling time. The rule is simple: the slower one (with longer settling time) between them defines the common settling time of the pair. And then the HDs in each group are averaged to produce an inter- or intra-die HD for responses with different settling times. This definition must not have a direct practical meaning. However, we will be able to see the clear trend of the relationship between settling times and CRP qualities upon this definition.

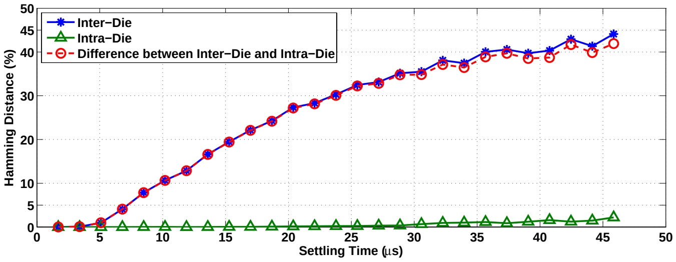


Figure 4. Inter- and intra-die Hamming distances of CRPs that are separated according to their settling times.

Fig. 4 shows the inter- and intra-die HDs of PUF responses that are separated according to their settling times based on the rules presented above. It can be seen that the CRPs with longer settling times show larger inter-die HDs that approach 50%, while the intra-die HD increases from 0% to 2.19% as the settling time gets longer. However, since the distance between the two curves grows from 0% to the significant 41.91%, it is clearly seen that the quality of the CRPs with long settling times is significantly better than that of the CRPs with short settling times. Therefore, to achieve more efficient identification and authentication, choosing CRPs that have longer settling times (e.g., in this case, we suggest CRPs with settling times between 35µs and 47µs) would be a simple solution. This selection process can be included in the PUF enrollment phase, where CRPs are measured and recorded for later use in, e.g., challenge-response authentication. Besides, this would ease a designer's work when building a BR-PUF in a chip, since it would not be so necessary to take great care of the symmetry of the layout or even the very detailed characteristics of process variations, e.g., the position-on-the-wafer related variations or mismatch. And especially for FPGA-based designs, in which designers are less free on the layout, this special property of BR-PUFs becomes an important advantage.

[2] Room temperature in this paper means the temperature in a normal office room without any special control. It may vary from time to time.

Due to the limited evaluation time we have applied in our experiment, we were not able to verify whether CRPs with even longer settling times would show even better uniqueness, and how much the unreliability increases. However, from the trend shown in Fig. 4, we estimate that the increasing of the distance between the inter-die and the intra-die curves has a turning point, where the distance starts to decrease, since the inter-die curve is already close to 50%, which is the ideal case, and the intra-die curve should continue to rise as the settling time gets longer. We argue that another useful situation would occur at the far-end of the curve, where both inter- and intra-die HDs are close to 50%, making the BR-PUF no longer a PUF, but a True Random Number Generator [33]. This may open up a new application area for the BR-PUF, although it should not be called a PUF in that situation.

C. *Reliability against Temperature Variation*

The reliability of the challenge-response behaviors of PUFs against temperature variations is an important measure of quality for PUFs that may be applied in a temperature-varying environment. To evaluate the reliability of the BR-PUF against temperature variations, we measured the intra-die HDs of a 64-stage implementation on an FPGA chip under different temperatures, taking a set of 10,000 CRPs measured at room temperature as the reference. The result is shown in Fig. 5.

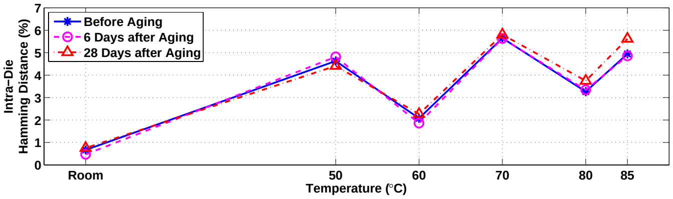
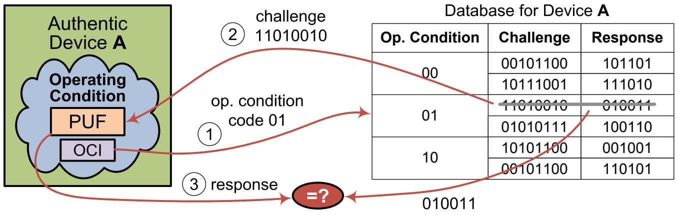


Figure 5. Reliability of the BR-PUF against temperature variations.

From the curves obtained at three different times with regard to an aging test that will be discussed in Subsec. D, it can be seen that an intra-die HD of up to 5.81% can be caused by the change in environmental temperature. Compared to the minimum inter-die HD of 6.0% measured from the same PUF design, the result makes this PUF design hard to be used in the conventional way in a temperature-varying environment. However, this figure also suggests that the reliability at each specific temperature is quite high, with a maximum distance of 0.76% (occurred at 85°C) between every two curves. This combination of reliability characteristics with regard to temperature conditions suggests a new way of using the BR-PUFs in, e.g., challenge-response authentications, if we can incorporate an extra circuit to identify different environmental temperatures.



Database for Device **A**

Op. Condition	Challenge	Response
00	00101100	101101
	10111001	111010
01	~~11010010~~	~~010011~~
	01010111	100110
10	10101100	001001
	00101100	110101

Figure 6. Authentication with operating-condition-sensitive PUFs.

Fig. 6 illustrates the basic idea of using such PUFs that are operating-condition-sensitive but are relatively reliable under specific operating conditions. In the PUF enrollment phase, a trusted party creates a database for the authentic device **A**—The PUF built in **A** is measured for CRPs under some different discrete operating conditions (e.g., different temperatures and/or different supply voltages) that the PUF may be exposed to when it is later used. Meanwhile, an operating condition identifier (OCI) identifies the operating condition at the time when a CRP is measured, and represents the operating condition in binary form, called the operating condition code (OCC). Different from a conventional PUF-based authentication [25], not only the CRPs are stored in the database, but also the OCCs that can distinguish the CRPs measured in one operating condition from those measured in another operating condition. To check the authenticity of a device later, the trusted party selects from his previously recorded database an unused challenge having the OCC that is sent by the OCI. The challenge is sent to the device and a response is then obtained. If the response matches (or is close enough to) the previously recorded one, the device is authentic. To protect against man-in-the-middle attacks, the CRP just used is removed from the database or is labeled as used.

Now the question left is whether an OCI can be realized. A most direct solution is to build sensors to report the exact operating conditions, which could be costly. We propose a simple solution that is especially suitable for electrical PUFs. Since the most common varying operating conditions that may affect electrical PUFs are just the temperature and the supply voltage, and the frequency of a ring oscillator is known to have a reliable linear relationship to the temperature and the supply voltage [34], together with a counter that measures the frequency, a ring oscillator can be used as a temperature sensor (when the power supply is constant) or a voltage sensor (when the temperature is constant). But if neither of the two conditions is constant, a ring oscillator would not be able to serve. To solve this problem, we propose to use two ring oscillators with distinct characteristic curves (lines), or to be exact, surfaces (planes). Theoretically, as long as the intersection of their characteristic planes forms a line that is neither parallel to the temperature-voltage plane, nor to the frequency axis, the combination of these two ring oscillators can be used as an OCI to explicitly identify operating conditions with varying temperatures and varying supply voltages. A special advantage of using this OCI design for operating-condition-sensitive PUFs is that the process variations of the ring oscillators need not to be considered, since an OCI just needs to identify the operating conditions applied to the PUF it is attached to, and the OCCs do not need to be compared between different OCIs, even if two OCIs generate different OCCs under the identical operating condition, it does not sabotage the authentication process based on it.

With the extended protocol and the OCI design described above, the BR-PUF is able to serve in varying temperature conditions even though it is a temperature-sensitive PUF. At the same time, this has suggested even more complicated challenge-response behaviors of the BR-PUF, making it even harder to carry out machine learning and modeling-based attacks.

D. *Reliability against Aging*

The characteristics of electronic circuits are possible to drift during their service life, which is called the aging of circuits. Electrical intrinsic PUFs including the BR-PUFs also cannot escape from it. However, we would like to know how much their characteristics drift, and whether they spoil the protocols, in which the BR-PUFs are used. To verify this, we employed a 64-stage BR-PUF implemented on FPGA in an aging test, in which the circuit was kept running, generating CRPs continuously for 29 days, at the environmental temperature of 85°C. This experimental setup accelerates the aging process of the BR-PUF. During the aging process, the PUF responses to the same set of 10,000 challenges were measured every other one to four days at room temperature. Taking the CRPs measured before the aging test at room temperature as the reference, we calculated the intra-die HDs for each working day, and the result is shown in Fig. 7. The intra-die HDs during the aging process lie between 0.41% and 1.96%, which is still much lower than the minimum inter-die HD of 6.0% measured from the same PUF design. Compared to the average intra-die HD of 0.8% measured before the aging test, the aging effect does not show great impact on the performance of the BR-PUFs. This can also be seen in Fig. 5, in which the temperature effect has been compared over three different times before or during the aging process. The maximum variation of 0.76% in the intra-die HD exhibited at 85°C is quite small, even if we exclude the fact that the variation must have partially been brought by the inaccuracy of temperature measurement and ambient noises.

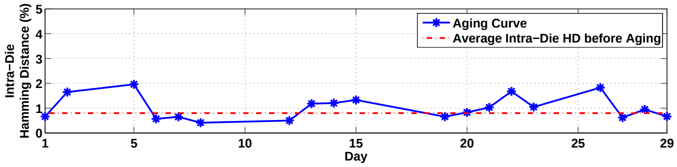


Figure 7. Aging effect on the BR-PUF.

V. Summary and Conclusion

We have introduced a new PUF design which we call the Bistable Ring PUF. Based on the experimental results we obtained from implementations on FPGAs, we studied the quality of this new PUF in different aspects and discussed some special characteristics of the BR-PUFs which make them easier to provide highly efficient identification and authentication. Although the BR-PUF is found to be a temperature-sensitive PUF, since it is still quite reliable under specific temperatures, an extended authentication protocol that incorporates an operating condition identifier is introduced for BR-PUFs when used under varying temperatures. Besides, an aging test shows that the BR-PUFs are relatively reliable against aging.

From the design principles we used, and from the properties we observed in our experiments, we believe that the BR-PUF potentially offers the following advantages:

- Compared to PUFs that are based on cross-coupled structures (SRAM PUF, Butterfly PUF, and Flip-flop PUF), the BR-PUF is able to generate an exponential number of CRPs. This makes it usable both as a Weak PUF and a Strong PUF [8], which broadens its application area.
- Compared to standard delay-based PUFs (Arbiter PUF and RO-PUF), we suspect that its behavior could be more complex, since it exhibits strong and inherent nonlinearities. Even though this needs to be confirmed by further analysis, we have yet been unable to find a simple model of the BR-PUF that would be necessary to mount machine learning and modeling-based attacks.
- For certain challenges, the BR-PUF would oscillate for a relatively significant period before it stabilizes, leading to long read-out times. While this is undesirable in certain applications (where the use of such CRPs should be avoided), it can be useful in other scenarios: It could prevent the time-efficient collection of large amounts of CRPs, which would probably be needed for efficient modeling attacks. BR-PUFs naturally exhibit this slow-readout attribute, which has also been exploited in [35].
- The non-uniformity of response settling times of the BR-PUF makes it possible to separate CRPs by quality very easily, and to select CRPs with different characteristics for possibly different purposes. This has been discussed in greater detail in Sec. IV.

Our future work will concentrate on the evaluation of the hardness of the BR-PUF and on further investigations of their properties by simulations and exploring on a larger scale of hardware.

Acknowledgment

The authors are grateful to H. Habal, who came up with the name “Bistable Ring PUF” for the new PUF design, M. Becherer, N. Chen, T. Zimmer, B. Li, P. Wu, and Y. Wang. This work has been supported by the International Graduate School of Science and Engineering (IGSSE) of the Technische Universität München. This work has been conducted within the Physical Cryptography Project of the Technische Universität München.

References

[1] D. Bauer, “An anti-counterfeiting concept for currency systems,” Technical Report, Sandia National Labs, 1983.

[2] K. Lofstrom, W. R. Daasch, and D. Taylor, “IC identification circuit using device mismatch,” IEEE Intl. Solid-State Circuits Conf. (ISSCC 00), Digest of Technical Papers, Feb. 2000, pp. 372–373.

[3] R. S. Pappu, Physical One-Way Functions, Ph.D. Thesis, MIT, 2001.

[4] R. S. Pappu, B. Recht, J. Taylor, and N. Gershenfeld, “Physical one-way functions,” Science, vol. 297, Sep. 2002, pp. 2026–2030.

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

[6] U. Rührmair, J. Sölter, and F. Sehnke, “On the foundations of physical unclonable functions,” Cryptology ePrint Archive, Report 277, 2009.

[7] R. Maes and I. Verbauwhede, “Physically unclonable functions: a study on the state of the art and future research directions,” in Towards Hardware-Intrinsic Security, Springer-Verlag, 2010, pp. 3–37.

[8] U. Rührmair, H. Busch, and S. Katzenbeisser, “Strong PUFs: models, constructions, and security proofs,” in Towards Hardware-Intrinsic Security, Springer-Verlag, 2010, pp. 79–96.

[9] Side Channel Attacks Database, http://www.sidechannelattacks.com/.

[10] U. Rührmair, S. Devadas, and F. Koushanfar, “Security based on physical unclonability and disorder,” in Introduction to Hardware Security and Trust, Springer-Verlag, in press.

[11] G. E. Suh, C. W. O'Donnell, and S. Devadas, “AEGIS: a single-chip secure processor,” Information Security, vol. 10, 2005, pp. 63–73.

[12] J. Guajardo, S. S. Kumar, G. Schrijen, and P. Tuyls, “FPGA intrinsic PUFs and their use for IP protection,” Proc. Intl. Workshop on Cryptographic Hardware and Embedded Systems (CHES 07), 2007, pp. 63–80.

[13] K. Rosenfeld, E. Gavas, and R. Karri, “Sensor physical unclonable functions,” IEEE Intl. Symp. on Hardware-Oriented Security and Trust (HOST 2010), IEEE Press, Jun. 2010, pp. 112–117.

[14] R. Maes, P. Tuyls, and I. Verbauwhede, “A soft decision helper data algorithm for SRAM PUFs,” IEEE Intl. Symp. on Information Theory (ISIT 09), IEEE Press, Jun. 2009, pp. 2101–2105.

[15] M. Yu and S. Devadas, “Secure and robust error correction for physical unclonable functions,” IEEE Des. Test. Comput., Jan. 2010, pp. 48–64.

[16] U. Rührmair, “SIMPL systems: on a public key variant of physical unclonable functions,” Cryptology ePrint Archive, Report 255, 2009.

[17] Q. Chen, G. Csaba, X. Ju, S. B. Natarajan, P. Lugli, M. Stutzmann, et al., “Analog circuits for physical cryptography,” Proc. IEEE Intl. Symp. of Integrated Circuits (ISIC 09), Dec. 2009, pp. 121–124.

[18] Q. Chen, G. Csaba, P. Lugli, U. Schlichtmann, M. Stutzmann, and U. Rührmair, “Circuit-based approaches to SIMPL systems,” Journal of Circuits, Systems, and Computers, vol. 20, 2011, pp. 107–123.

[19] G. Csaba, X. Ju, Z. Ma, Q. Chen, W. Porod, J. Schmidhuber, et al., “Application of mismatched cellular nonlinear networks for physical cryptography,” Proc. IEEE Intl. Workshop on Cellular Nanoscale Networks and Their Applications (CNNA 2010), Feb. 2010, pp. 1–6.

[20] U. Rührmair, “SIMPL systems, or: can we design cryptographic hardware without secret key information?” Proc. 37th Conf. on Current Trends in Theory and Practice of Computer Science (SOFSEM 2011), Springer-Verlag, Jan. 2011, pp. 26–45.

[21] N. Beckmann and M. Potkonjak, “Hardware-based public-key cryptography with public physically unclonable functions,” Intl. Workshop on Information Hiding (IH 09), Jun. 2009, pp. 206–220.

[22] M. Majzoobi, A. Elnably, and F. Koushanfar, “FPGA time-bounded unclonable authentication,” Intl. Conf. on Information Hiding (IH 2010), Jun. 2010, pp. 1–16.

[23] P. Tuyls, G. Schrijen, B. Škorić, J. van Geloven, N. Verhaegh, and R. Wolters, “Read-proof hardware from protective coatings,” Proc. Intl. Workshop on Cryptographic Hardware and Embedded Systems (CHES 06), 2006, pp. 369–383.

[24] D. Lim, Extracting Secret Keys from Integrated Circuits, Master’s Thesis, MIT, 2004.

[25] G. E. Suh and S. Devadas, “Physical unclonable functions for device authentication and secure key generation,” Proc. ACM/IEEE Design Automation Conf. (DAC 07), ACM Press, Jun. 2007, pp. 9–14.

[26] S. Kumar, J. Guajardo, R. Maes, G. J. Schrijen, and P. Tuyls, “The butterfly PUF protecting IP on every FPGA,” Proc. IEEE Intl. Workshop on Hardware-Oriented Security and Trust (CHES 08), IEEE Press, 2008, pp. 67–70.

[27] R. Maes, P. Tuyls, and I. Verbauwhede, “Intrinsic PUFs from flip-flops on reconfigurable devices,” Proc. Benelux Workshop on Information and System Security (WISSec 08), Nov. 2008.

[28] M. Majzoobi, F. Koushanfar, and M. Potkonjak, “Lightweight secure PUFs,” Proc. Intl. Conf. on Computer-Aided Design, IEEE Press, 2008, pp. 670–673.

[29] R. Helinski, D. Acharyya, and J. Plusquellic, “A physical unclonable function defined using power distribution system equivalent resistance variations,” Proc. ACM/IEEE Design Automation Conf. (DAC 09), ACM Press, Jul. 2009, pp. 676–681.

[30] E. Ozturk, G. Hammouri, and B. Sunar, “Physical unclonable function with tristate buffers,” Proc. IEEE Intl. Symp. on Circuits and Systems (ISCAS 08), IEEE Press, May 2008, pp. 3194–3197.

[31] M. Majzoobi, F. Koushanfar, and M. Potkonjak, “Testing techniques for hardware security,” IEEE Intl. Test Conf. (ITC 08), Oct. 2008, pp. 1–10.

[32] A. Maiti, J. Casarona, L. McHale, and P. Schaumont, “A large scale characterization of RO-PUF,” IEEE Intl. Symp. on Hardware-Oriented Security and Trust (HOST 2010), Jun. 2010, pp. 94–99.

[33] D. E. Holcomb, W. P. Burleson, and K. Fu, “Power-up SRAM state as an identifying fingerprint and source of true random numbers,” IEEE Trans. Comput., vol. 58, 2009, pp. 1198–1210.

[34] B. Datta and D. Kumar, “Analysis of a ring oscillator based on chip thermal sensor in 65nm technology,” Report, UMass Amherst, 2005.

[35] U. Rührmair, C. Jäger, M. Bator, M. Stutzmann, P. Lugli, and G. Csaba, “Applications of High-Capacity Crossbar Memories in Cryptography,” IEEE Trans. Nanotechnol., vol. 9, 2010.