Journal of Electronic Testing (2018) 34:571–586
https://doi.org/10.1007/s10836-018-5745-6

Impact of Aging on the Reliability of Delay PUFs

Naghmeh Karimi[1] · Jean-Luc Danger[2,3] · Sylvain Guilley[2,3,4]

Received: 14 April 2018 / Accepted: 2 August 2018 / Published online: 20 August 2018

Abstract
Physically Unclonable Functions (PUFs) are mainly used for generating unique keys to identify electronic devices. These entities mainly benefit from the process variations occurring during the device manufacturing. To be able to use PUFs to identify electronic devices or to utilize them in cryptographic applications, the reliability of PUFs needs to be assured under a wide variety of environmental conditions and aging mechanisms, including the switching activity of the PUFs' internal signals. In practice, it is important to evaluate aging effects as early as possible, preferentially at design time. In this paper, we evaluate the impact of aging on two types of delay-PUFs (arbiter-PUFs and loop-PUFs) with different switching activities. This work takes advantage of both simulation tool and silicon tests on a 65nm ASIC implementation. To expedite the simulation process and get rid of conducting simulations of multiple delay-element PUFs, we propose an extrapolation method to evaluate the effect of BTI (Bias Temperature-Instability) and HCI (Hot Carrier Injection) aging under different switching activities on PUFs with multiple delay elements using the aging effects on single delay-element PUFs. The results show that switching activity (expressed in terms of transitions/time) has a limited impact on delay chains of considered delay-PUFs, while it has a greater impact on the arbiter (RS latch) of the arbiter-PUF. The simulation results show that the aging-related Bit Error Rate in an arbiter-PUF with high switching activity can be 11 times worse than the Bit Error Rate in the same PUF when there is no activity in 20 months.

Keywords Hardware security · Physically Unclonable Functions (PUFs) · Delay-PUFs · Device aging · Reliability

1 Introduction

With the increasing concern about the security of integrated circuits, Physically Unclonable Functions (PUFs) are broadly deployed to provide a unique signature for each integrated circuit. A PUF signature can be used for device authentication, or for generating secret keys and random variables in cryptographic devices. Indeed, using PUFs avoids storing secret keys in digital memory, thereby enhances the security of the systems in which they are deployed. Due to their small size and their high resiliency against physical attacks, PUFs are well suited for low-cost devices such as radio-frequency identifiers (RFIDs) or smart cards [5].

PUFs generate secret keys by exploiting the inherent physical variations of devices during the manufacturing process and thereby each PUF generates a unique signature extracted based on physical characteristics of its elements. The unique behavior after fabrication stems from a *static randomness* due to technological dispersion, a well known source of mismatch in electronic circuits that follows a normal distribution [34].

To cope with the measurements' noise and increase the reliability of PUFs, the signal-to-noise ratio (SNR) should increase, which in turn needs a power down between measurements. This solution may not be practical for SRAM-PUFs which rely on the value that each SRAM memory bit gets during the power boot up [12]. Indeed, in these cases

Responsible Editor: P. Mishra

✉ Naghmeh Karimi
nkarimi@umbc.edu

Jean-Luc Danger
jean-luc.danger@telecom-paristech.fr

Sylvain Guilley
sylvain.guilley@secure-ic.com

[1] CSEE Department, University of Maryland Baltimore County, Baltimore, MD 21250, USA

[2] Institut Mines-Telecom, Telecom ParisTech, Paris 75634, France

[3] Secure-IC S.A.S., 35510 Cesson-Sévigné, France

[4] Ecole Normale Supérieure (ENS), Département d'Informatique, Paris, France

measurements cannot be repeated, and alternatively error correcting codes are used to compensate the noisy environment. In contrast, SNR can improve easily (by a factor of n) in so-called delay-PUFs [30] where n elements are chained, and the total delay of the chain is measured. In this paper, we focus on two different types of delay-PUFs: arbiter-PUFs [9] and loop-PUFs [5].

Delay-PUFs suffer from the drawback that they are only able to deliver one bit per measurement. Therefore, building a 128 bit key requires (at least) 128 measurements. In practice, more measurements are needed if some are repeated for the sake of enhancing the SNR through averaging/voting. *In fact, due to the extensive number of queries in delay-PUFs, they encounter an innate problem of unsteadiness stemming from noise and thermal effect, i.e., hundreds or thousands of required successive queries locally heat up the silicon, thereby modifying delays* [3]. *This issue can be mitigated at firmware level as follows. The delay-PUF is initially queried randomly, simply to heat up its logic aiming at increasing the effective calls* [3]. *In the second step, the PUF is queried several times and its response is found based on the average of all query responses in order to increase its reliability. Accordingly, in practical use-cases of delay-PUFs, the number of calls is much higher than the strictly necessary ones. Therefore, delay-PUFs are more prone to aging due to this phenomena.*

To be able to utilize a PUF in practical security applications, the key generated by the PUF should be stable over time and resilient against the aging mechanisms that affect integrated circuits [23, 24, 35, 38]. However, in practice, with the advance of VLSI technology and moving towards smaller feature sizes, the effect of aging mechanisms such as Bias Temperature-Instability (BTI), Hot-Carrier Injection (HCI), and Time Dependent Dielectric Breakdown (TDDB) has increased. Aging-related degradation may result in transistors' parameters shift during the operation time and eventually performance degradation and/or functional failures of the PUF devices. Thereby, characterizing the impact of aging degradation on PUFs and developing aging mitigation methods is crucial.

Since normally a PUF is not solicited very often (e.g., solicitation occurs at each boot, or on each authentication), PUF aging may be considered as a secondary issue. However, for the following reasons, in industrial products, a PUF is used more frequently:

- *There is usually a BIST (Built-In Self-Test) at each system power-up;*
- *For reliability improvement reasons, a PUF is typically called several times to vote which bit is the most stable, hence the most likely (despite the noise).*

In addition, since PUFs are security related components, they are potential targets of different attacks. Note that *a malicious user compromises a PUF by requesting continuous authentications. Therefore, it is important for PUFs to withstand such attacks, by tolerating aging.*

Although in industrial products PUFs are queried more frequently, to have a thorough investigation of the effect of aging on the embedded PUFs, the study need to be performed for different PUF activity rates. Accordingly in this paper, we extract the aging-related reliability degradation of the considered PUFs under different switching activities. Note that *even in cases when there is no switching activity, the transistors experience static BTI aging and thereby their reliability is mitigated.*

Using Error Correction Codes (ECCs) [2] to improve the reliability of PUFs is costly, and the code correction capability has to be extended to anticipate the aging impact. Software techniques have been proposed in [21] to combat the aging effects in PUFs. The proposed protocol-level solutions can either detect drifts in PUFs and update the affected challenge/response pairs or prevent such drifts by shortening the lifespan of challenge/response pairs [21]. Rahman et al. [35] studied the impact of aging on ring-oscillator delay-PUFs (RO-PUFs) and presented an aging resistant RO-PUF that stops oscillation when the PUF is not used. Maiti et al. [27] conducted an accelerated aging process to investigate the effect of aging on the functionality of RO-PUFs. They proposed a reconfigurable RO-PUF to mitigate aging effects. The proposed anti-aging architecture selects the ROs with maximum frequency differences as the RO pair to compare [27].

To the best of the authors' knowledge, previous research conducted on the reliability of PUFs against aging mainly focuses on SRAM-PUFs, RO-PUFs and the PUFs based on sense amplifiers [2, 11, 14, 25, 27, 35, 36, 46] and there is little research in open literature on the impact of aging on other delay-PUFs including loop-PUFs and arbiter-PUFs [16, 28]. Mispan et al. [28] focus on arbiter-PUFs but they only consider the aging of delay chains and ignore the aging of the arbiter itself. However, as our results (presented in Section 4) show, the effect of aging in arbiter itself is significant and it may jeopardize the reliability of the arbiter-PUFs. In addition, [28] assumes that the threshold voltage and other specifications of the transistors in the delay chain of the arbiter-PUF are similar and based on this assumption, they discuss that both paths feeding the arbiter of an arbiter PUF have similar specifications and so they experience equal aging rate which makes the PUF reliable. However, this assumption is not valid due to the process variations occur during manufacturing (which naturally is the basis of PUFs). Therefore, each path of the delay chain in an arbiter-PUF may experience different aging rate (related to its specifications) and therefore a thorough analysis on the impact of aging in arbiter-PUFs is required.

A study of aging on arbiter and Loop PUFs has been done in [16] but it only considers the NBTI (Negative-Bias Temperature-Instability) impact. In [17] the impact of switching activity has been simulated on the arbiter and Loop PUFs but it is only based on simulation, i.e., it does not include the aging results of real silicon arbiter-PUFs.

In this paper, we extend the study of aging effects to other types of delay-PUFs, mainly arbiter-PUFs and loop-PUFs as due to their high entropy, these PUFs are very popular [37]. Using extensive simulation results, we investigate the reliability of these PUFs against NBTI and HCI aging mechanisms, and study their sensitivity to aging degradation by evaluating the aging-related Bit Error Rate (BER) in these PUFs. We also present real silicon results demonstrating the effect of aging degradation in these PUFs. The contributions of this paper are as follows:

- Detailed HSpice MOSRA simulations investigating the effect of NBTI and HCI degradations on delay-PUFs;
- A generic extrapolation-based approach to assess the effect of aging in different type of delay-PUFs;
- Analysis of the effect of the switching activity on the degradation of delay-PUFs;
- Demonstrating the effect of aging on the BER of delay-PUFs manufactured in a 65nm process technology.

The rest of this paper is organized as follows. Section 2 presents the preliminary backgrounds on aging mechanisms and describes the two delay-PUFs we consider in this paper. Section 3 first presents our aging evaluation methodology. We use this methodology to evaluate the impact of aging on a delay-PUF with single delay-element. This section, then discusses the proposed extrapolation methodology utilized to assess the aging effects in a delay-PUF with an arbitrary size. Experimental results including simulation results and real silicon data are presented and discussed in Section 4. Section 5 presents an aging-resilient arbiter to increase the reliability of arbiter-PUFs. Finally, Section 6 concludes the paper.

2 Preliminaries

2.1 Background on Aging Mechanisms

Aging mechanisms including Negative-Bias Temperature-Instability (NBTI), Positive-Bias Temperature-Instability (PBTI), Hot Carrier Injection (HCI), Time Dependent Dielectric Breakdown (TDDB), and Electro-Migration (EM) result in performance degradation and eventual failure of digital circuits over time [20]. In practice, NBTI, PBTI, HCI, and TDDB all deal with the gate oxides of transistors while EM occurs in the interconnect metal lines.

BTI includes NBTI and PBTI effects, and is one of the major causes of threshold-voltage increase in transistors during their lifetime. NBTI and PBTI occur in PMOS and NMOS transistors, respectively. In practice, the impact of NBTI is more dominant than PBTI beyond 45nm technology nodes. With the introduction of high-k gate dielectrics and metal gate transistors, PBTI effects have also received significant attention [8, 48]. NBTI occurs in a PMOS transistor when a negative voltage is applied to its gate. In this mechanism, positive interface traps are generated at the Si-SiO$_2$ interface. As a result, the threshold voltage increases and the PMOS transistor becomes slower and fails to meet timing constraints. In contrast to NBTI, PBTI occurs when a positive voltage is applied to the gate of an NMOS transistor. This results in generating traps at the interface of gate oxide and channel, and in turn increases the transistor threshold-voltage.

HCI occurs when hot carriers are injected into the gate dielectric during transistor switching and remain there. HCI is a function of switching activity and degrades the circuit by shifting the threshold voltage and the drain current of transistors under stress [35]. HCI mainly affects NMOS transistors.

TDDB relates to the creation of an electrical current conduction path through the gate oxide in the device. It degrades the isolation properties of gate dielectric, increasing the tunneling current across the transistor gate terminal. Ultimately, TDDB results in device breakdown [32].

High density currents result in EM aging. The currents create electron winds that cause metal atoms to migrate over time, gradually removing metal atoms from wires, thereby increasing interconnect resistance. EM eventually results in an open circuit, creating a permanent error [29].

Among all aging mechanisms, BTI and HCI are two leading factors in performance degradation of digital circuits [33]. Both mechanisms result in increasing switching and path delays in the circuit under stress [18, 19]. This will eventually lead to timing violations and finally to faster wearout of the system. Thereby, both mechanisms jeopardize the reliability of digital circuits. *Accordingly, due to the dominant effect of NBTI and HCI in the reliability of digital circuits, compared to other aging mechanisms, in this paper we focus on NBTI and HCI aging and investigate their effects on the reliability of the considered delay-PUFs*. What follows discusses these aging mechanisms in more detail.

NBTI Aging NBTI affects a PMOS transistor when a negative voltage (i.e. $V_{gs} < V_t$) is applied to its gate. In pulsed signals, a PMOS transistor experiences two phases of NBTI depending on its operating condition. The first phase, so-called stress phase, occurs when the transistor is on, i.e., when a negative voltage ($V_{gs} < V_t$) is applied to its gate. In this case, positive interface traps are generated at the Si-SiO$_2$

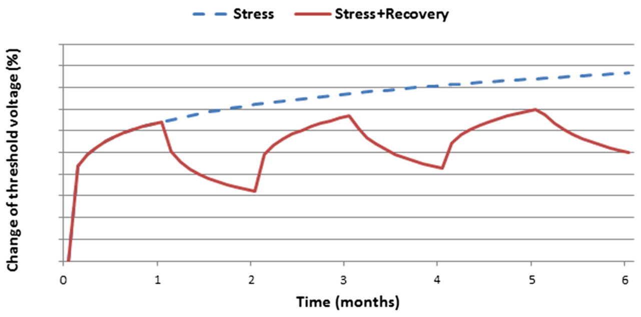


Fig. 1 Threshold voltage shift of a PMOS transistor under NBTI effect

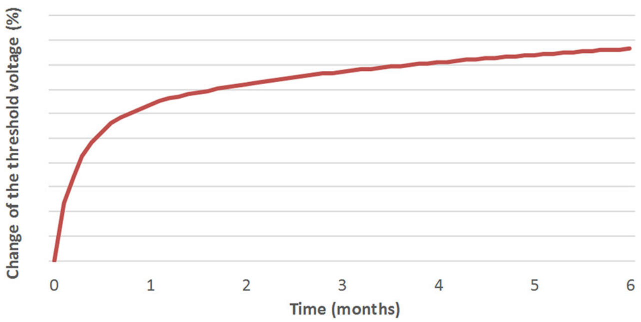


Fig. 2 Threshold voltage shift of an NMOS transistor under HCI effect

interface which lead to an increase of the threshold voltage of the transistor. The second phase, so-called recovery phase, occurs when a positive voltage ($V_{gs} > V_t$) is applied to the gate of the PMOS transistor. As a result, the threshold voltage drift that occurred during the stress phase will partially recover.

Threshold voltage drifts of a PMOS transistor under stress depend on the physical parameters of the transistor, supply voltage, temperature, and stress time [1, 22]. The last three parameters (so-called external parameters) are generally used as acceleration factors of aging process. Figure 1[1] shows the threshold voltage drift of a PMOS transistor that is continuously under stress for 6 months and a PMOS transistor that alternates stress/recovery phases every other month. As shown, NBTI effect is high in the first couple of months but the threshold voltage tends to saturate for long stress times. The impact is exacerbated with thinner gate oxide and higher operating temperature [1, 26].

Two prevalent theories, Reaction-Diffusion (R-D) and Trapping-Detrapping (T-D), have been proposed in literature to explain NBTI. The R-D model explains the NBTI phenomenon as the breaking and rebonding of hydrogen-silicon bonds at the silicon-gate dielectric interface of PMOS devices [4, 40]. The T-D model considers a number of defect states with different energy levels, and capture and emission time constants. In the T-D model, the threshold voltage increases when a trap captures a charge carrier from the channel of a PMOS device [42].

According to the R-D model proposed in [45], the NBTI-related increase in the threshold voltage of a PMOS transistor in the stress phase is evaluated by Eq. 1 [44].

$$\Delta V_{th_{stress}} = A_{NBTI} \times t_{ox} \times \sqrt{C_{ox}(V_{dd} - V_{th})} \times e^{(\frac{V_{dd}-V_{th}}{t_{ox} \times E_0} - \frac{E_a}{k \times T})} \times t_{stress}^{0.25} \tag{1}$$

where t_{ox} is the oxide thickness, and C_{ox} is the gate capacitance per unit area. E_0 and E_a are device dependent parameters and constant. A_{NBTI} is a technology dependent constant and k is the Boltzmann constant. T is the temperature and t_{stress} is the stress time.

As discussed, the threshold-voltage drift of a PMOS transistor is partially recovered if the transistor is placed in the recovery phase (i.e., a positive voltage is applied to its gate). Equation 2, shows the final change in the threshold voltage of a PMOS transistor [44].

$$\Delta V_{th_{NBTI}} = \Delta V_{th_{stress}} \times (1 - \sqrt{\eta \frac{t_{recovery}}{t_{recovery} + t_{stress}}}) \tag{2}$$

In this equation, η is equal to 0.35 and t_{stress} and $t_{recovery}$ are the stress and recovery time durations, respectively.

HCI Aging Hot carriers refer to the electrons or holes in the substrate that attain energies above the average [36]. These high energetic carriers, which are the result of high electric fields in the drain region of a transistor are injected into the gate oxide and form interface states and eventually result in performance degradation in the transistor under stress. HCI mainly affects NMOS transistors and has become more severe as the transistor features continue to shrink [7].

HCI results in the change of the threshold voltage of the device under stress. Besides increasing the threshold voltage, HCI reduces the mobility of a device, which leads to a decrease in drain current. Figure 2[2] shows the threshold voltage drift of an NMOS transistor that is continuously under HCI stress for 6 months. Unlike NBTI, there is no recovery for HCI effects.

HCI effect is due to the switching between '0' and '1' on an NMOS transistor. Thereby, HCI is highly sensitive to the number of transitions occur in the gate input of the transistor under stress. In fact, the threshold voltage changes sublinearly with the number of transitions occur in the input of an NMOS transistor. In practice, HCI has a sublinear dependency on the clock frequency, usage time, and the activity factor of the transistor under stress, where activity factor represents the ratio of the cycles the transistor is doing transitions and the total number of cycles the device

[1] The Y axis has not been shown to make the graph generic for different technologies.

[2] The Y axis has not been shown to make the graph generic for different technologies.

is utilized. In addition, HCI effects depend on the operating temperature [33]. Equation 3 evaluates the HCI-induced threshold voltage shift [44, 45].

$$\Delta V_{th_{HCI}} = A_{HCI} \times \alpha \times f \times e^{\frac{V_{dd}-V_{th}}{t_{ox}\times E_1}} \times t^{0.5} \quad (3)$$

where t is time, and α and f are the activity factor and the frequency, respectively. In addition, t_{ox} is the oxide thickness, and E_1 depends of device specifications as well as temperature and V_{dd}. Here, A_{HCI} is a technology dependent constant.

In this research, to evaluate the impact of NBTI and HCI on the performance of a circuit under stress, HSpice MOSRA (MOS Reliability Analysis) [43] is deployed. MOSRA uses the Reaction-Diffusion (R-D) model discussed in [45].

2.2 Background on Delay-PUFs

Silicon PUFs can be divided into two main categories: the PUFs operating based on delay comparisons of different paths and the PUFs exploiting the initial state of memory blocks [6]. In this paper, we focus on the first group of PUFs. What follows briefly reviews the structure of the two delay-PUFs we consider in this study: loop-PUFs and arbiter-PUFs.

Arbiter-PUF Figure 3 depicts the architecture of an arbiter-PUF [41]. It includes a pair of delay chains and generates one response bit per challenge, in one single query. Indeed, this PUF operates based on the race between two identical paths (top and bottom paths shown in Fig. 3).

The race corresponds to the difference of the delay of these two paths. Indeed, only the sign of this difference is important (not the exact amount). The sign is extracted by the arbiter and corresponds to the PUF identifier. The downstream arbiter is a simple RS latch implemented by using two NAND (NOR) gates. NAND gates are used when the race takes place between the rising edge of the signals that feed the arbiter. Otherwise (in case of falling edge race), a NOR-based RS latch is deployed.

Loop-PUF Figure 4 shows the structure of a loop-PUF [5]. It includes a loop realized by n delay elements and one inverter. Each element $i \in \{1, 2, \ldots, n\}$ can have two delays (theoretically equal at blueprint level), chosen according to its challenge bit $\mathsf{c}_i \in \{0, 1\}$. The principle of the loop-PUF is to measure the difference of cumulative delays for a challenge and its complementary value. The sign of this delay difference corresponds to the PUF identifier. The structure of a loop-PUF is very similar to a ring-oscillator PUF (RO-PUF) [41], except that in a loop-PUF the delays are controllable and there is only one oscillator. Due to this similarity, we could think the aging study on loop-PUFs may be applicable to RO-PUFs too. This is not really the case as the Loop PUF uses only one ring oscillator which runs twice with a different challenge, whereas the RO-PUF uses two different ring oscillators selected by a challenge. *As loop-PUFs include only one oscillator (to generate each response bit), in these PUFs, the fact of having a differential measurement filters the common aging-related delay offset. Thereby, as the experiential results also confirm Loop-PUFs are little sensitive to aging while reference* [35] *shows, RO-PUFs are highly prone to aging.*

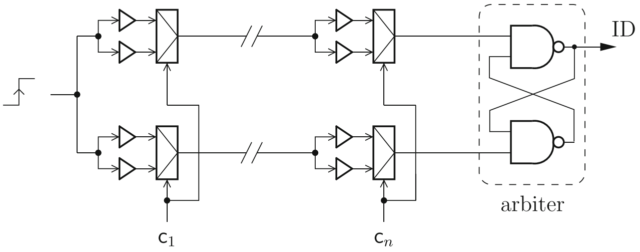


Fig. 3 Arbiter-PUF

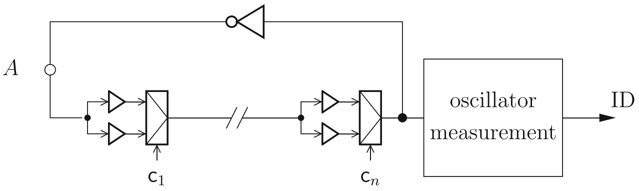


Fig. 4 Loop-PUF

3 Aging Methodology and Evaluation

To investigate the effect of aging on the reliability of PUFs, we conduct transistor-level HSpice simulations. However, HSpice simulations are highly time consuming. Thereby, conducting simulations for a large number of PUFs with arbitrary sizes are not timely efficient when thousands of Monte Carlo simulations are conducted. Accordingly, in this paper, we present an extrapolation scheme that extracts the timing specifications of PUFs with arbitrary sizes based on the specifications of single delay-element PUFs. In this Section, we first present how we use HSpice MOSRA to measure the aging effects of a PUF with single delay-element. Then, we discuss the proposed extrapolation methodology. Figure 5 shows the process.

3.1 Measuring the Aging Effects on a Single Delay Element PUF

To evaluate the effect of BTI and HCI aging on a delay-PUF with one delay-element, HSpice tool is deployed. The left dotted box in Fig. 5 shows this process. First, the transistor-level PUF is simulated in time zero (pre-aging mode) and fresh (pre-aging) rising and falling delays of the simulated

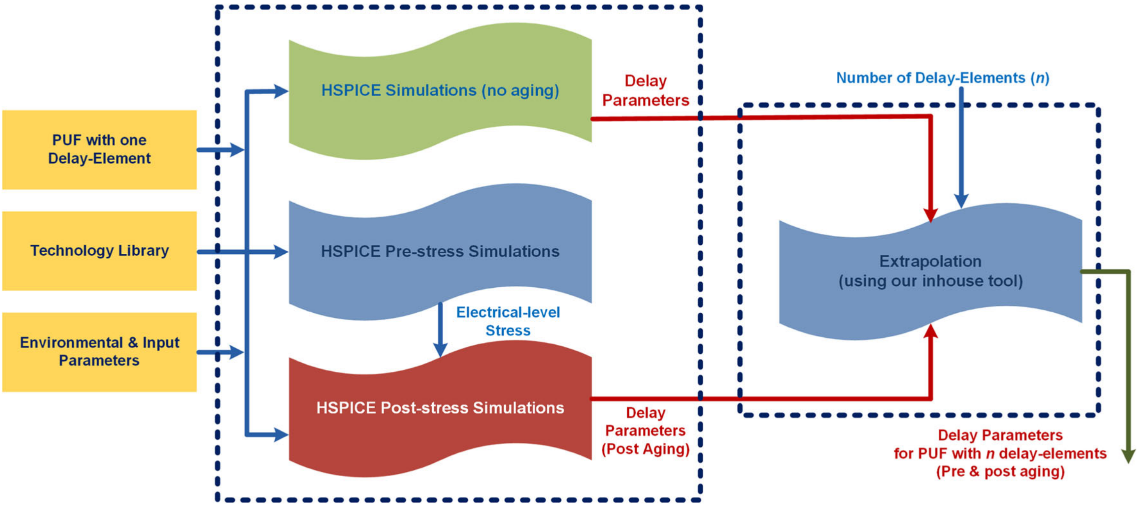

PUF with one Delay-Element
Technology Library
Environmental & Input Parameters
HSPICE Simulations (no aging)
HSPICE Pre-stress Simulations
Electrical-level Stress
HSPICE Post-stress Simulations
Delay Parameters
Delay Parameters (Post Aging)
Number of Delay-Elements (n)
Extrapolation (using our inhouse tool)
Delay Parameters for PUF with n delay-elements (Pre & post aging)

Fig. 5 Flowchart for extracting the delay parameters of a delay-PUF with an arbitrary size

PUF are extracted. Then, using HSpice MOSRA simulator, the drift of transistors' electrical parameters under both BTI and HCI aging and with different switching activities (for the internal signals of the PUF) are evaluated [43].

In the next step, using HSpice MOSRA, the degradation of the transistor-level device characteristics, computed previously, is translated to performance degradation at the circuit level. The delay parameters extracted in this phase are discussed in the following section.

3.2 Proposed Extrapolation Methodology

As discussed in Section 3.1, the aging evaluation process is first carried out on a single delay-element shown in Fig. 6. This element can be used to build both types of delay-PUFs. Then, an extrapolation methodology is conducted to evaluate the impact of aging in delay-PUFs with arbitrary sizes using the results extracted for single delay-element PUFs (the right dotted box in Fig. 5). Since a delay-PUF mainly relies on comparing two delay chains, the study focuses on the trend of delay change in the two considered delay paths, and more specifically for the arbiter-PUF, the probability to have bit-flips at the arbiter stage.

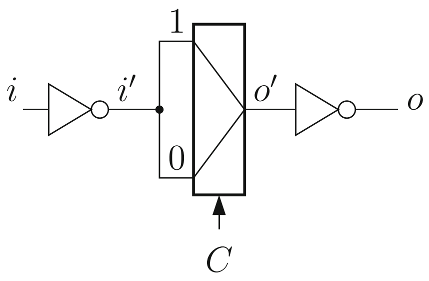


Fig. 6 Delay element considered for aging

The multiplexer shown in Fig. 6 is driven by the challenge bit C. In our method, we first extract four delay parameters for the delay element shown in this figure (for different aging durations). These parameters represent the propagation time (to send a rising/falling edge) between the input i' and the output o' of the multiplexer and are shown in Table 1.

To extract the delay parameters of each targeted delay-PUF (arbiter-PUF and loop-PUF), the following steps are taken:

1. A set of instances of single delay-element is modeled by using Monte Carlo (MC) simulations in Hspice. Each instance has its own delay.
2. The four timing parameters discussed above are extracted using HSpice MOSRA for each instance of delay-element considered in stage 1. The timing parameters are extracted for different aging durations.
3. For each type of delay-PUF, the timing characteristics are deduced from the set of four base timings ①,②,③,④ extracted in stage 2. The deduction scheme is discussed below for each PUF type.

- **Loop-PUF:** Only one delay element is needed and the following protocol is used:

    1. The challenge C is applied and the oscillation period is measured.

Table 1 Propagation delay time between input i' and output o' of the multiplexer in Fig. 6

Time ①	Rising edge propagation time when challenge is 1
Time ②	Falling edge propagation time when challenge is 1
Time ③	Rising edge propagation time when challenge is 0
Time ④	Falling edge propagation time when challenge is 0

2. The complementary challenge $\overline{C}$ is applied and the oscillation period is measured.
3. The difference of the oscillation periods extracted above (when C and $\overline{C}$ were applied) represents T_{LPUF}. As discussed in Section 2.2, the PUF identifier is the sign of T_{LPUF}. If $C = 1$, T_{LPUF} is computed by the difference of rising and falling propagation delays as (① + ②) - (③ + ④). The value is negated for $C = 0$. Note that these parameters were defined in Table 1. In sum, T_{LPUF} is evaluated using Eq. 4.

$$T_{LPUF} = C((\textcircled{1}+\textcircled{2})-(\textcircled{3}+\textcircled{4}))+ \overline{C}((\textcircled{3}+\textcircled{4})-(\textcircled{1}+\textcircled{2})) \tag{4}$$

- **Arbiter-PUF:** As shown in Fig. 3, in this PUF two elements are in parallel per challenge bit C. If $C = 1$, the race between the two elements corresponds to the delay difference between time ① of the first element (upper element fed by c_1 in Fig. 3) and time ① of the second element (lower element fed by c_1 in Fig. 3). However, if $C = 0$, time ③ is considered instead of time ①. Thereby, the delay difference (T_{APUF}) is evaluated using Eq. 4.

$$T_{APUF} = C(\textcircled{1}_1 - \textcircled{1}_2) + \overline{C}(\textcircled{3}_1 - \textcircled{3}_2) \tag{5}$$

  As shown in Fig. 3, the arbiter used in our arbiter-PUF is an RS latch realized by two NAND gates. Therefore, in the computations, we considered the rising edge propagation times (① or ③). Similarly, for NOR-based RS latch arbiters, falling edge times (② or ④) should be considered based on the challenge value.

After calculating T_{LPUF} for loop-PUFs (T_{APUF} for arbiter-PUFs) with one delay element, we evaluate the timing specifications of each PUF with arbitrary number of elements. In practice, for loop-PUFs with n elements (arbiter-PUFs with $2n$ elements), we add the T_{LPUF} (T_{APUF} for arbiter-PUFs) of all elements extracted using the above scheme. The sign (not the absolute value) of this addition determines the outcome of the PUF. Note that each element is being driven by its own challenge bit. This addition is seamless as the delay elements are separated by buffers.

We verified the correctness of the extrapolation scheme for both arbiter-PUFs and loop-PUFs. First, we simulated a number of 16-element PUFs of each type with different challenges and extracted the response/challege of each PUF. Then, we separately simulated the delay elements included in each of these PUFs, and used our extrapolation scheme to extract the result of the each 16-element PUF. The results of the first and second experiments matched thoroughly. This confirms the correctness of our extrapolation scheme. We repeated our experiments using 32- and 64-element PUFs and the results confirmed the correctness of the extrapolation scheme for each PUF.

4 Experimental Results and Discussions

In this section, we first provide the details of the simulation setup used to evaluate the effect of aging on the reliability of the considered delay-PUFs. Then, we present simulation results and discuss our observations. Finally, we present the real silicon results demonstrating the aging degradation effects in these PUFs.

4.1 Experimental Setup

In this research, we first implemented our single delay-element PUFs in the transistor level using a 45 nm technology extracted from the open-source NANGATE library [31] (as we didn't have access to the same technology library we used in the experiments with real silicon). We then used Synopsys HSpice for the transistor-level simulations and deployed the HSpice built-in MOSRA Level 3 model to capture aging effects in MOSFETs [43]. We evaluated the effect of both BTI and HCI effects. To assess the effect of switching activity of the internal signals of a PUF in aging-related degradation, we considered four different cases. In the first case, the PUF is always inactive (switching activity=0). Second and third cases consider the situations in which the PUF is active 1/64 or 1/8 of the time, respectively. In the fourth case, the PUF is always active with pulses injected every 2 ns with a duty cycle of 50%.

We ran Monte Carlo simulations for 8192 instances of basic delay elements. We then extracted the delay parameters discussed in Section 3.2 to extrapolate the effect of aging on 512 loop-PUFs, each including 16 delay elements using our in-house tool discussed above. We repeated the same scheme for the arbiter-PUFs. Simulations were carried out using the following process-variation parameters for a Gaussian distribution: transistor gate length L: $3\sigma = 10\%$; threshold voltage V_{TH}: $3\sigma = 30\%$, and gate-oxide thickness t_{OX}: $3\sigma = 3\%$. The process variation data reflects a 45-nm process in commercial use today [47].

Using HSpice MOSRA, the effect of aging was evaluated for 20 months of PUF operation in time steps of one month. As mentioned earlier, aging effects will be exacerbated in higher temperatures. To show the effect of temperature, in our experiments, we extracted the aging results in two different temperatures: 45 °C and 80 °C. Note that these temperatures are the internal chip temperatures.

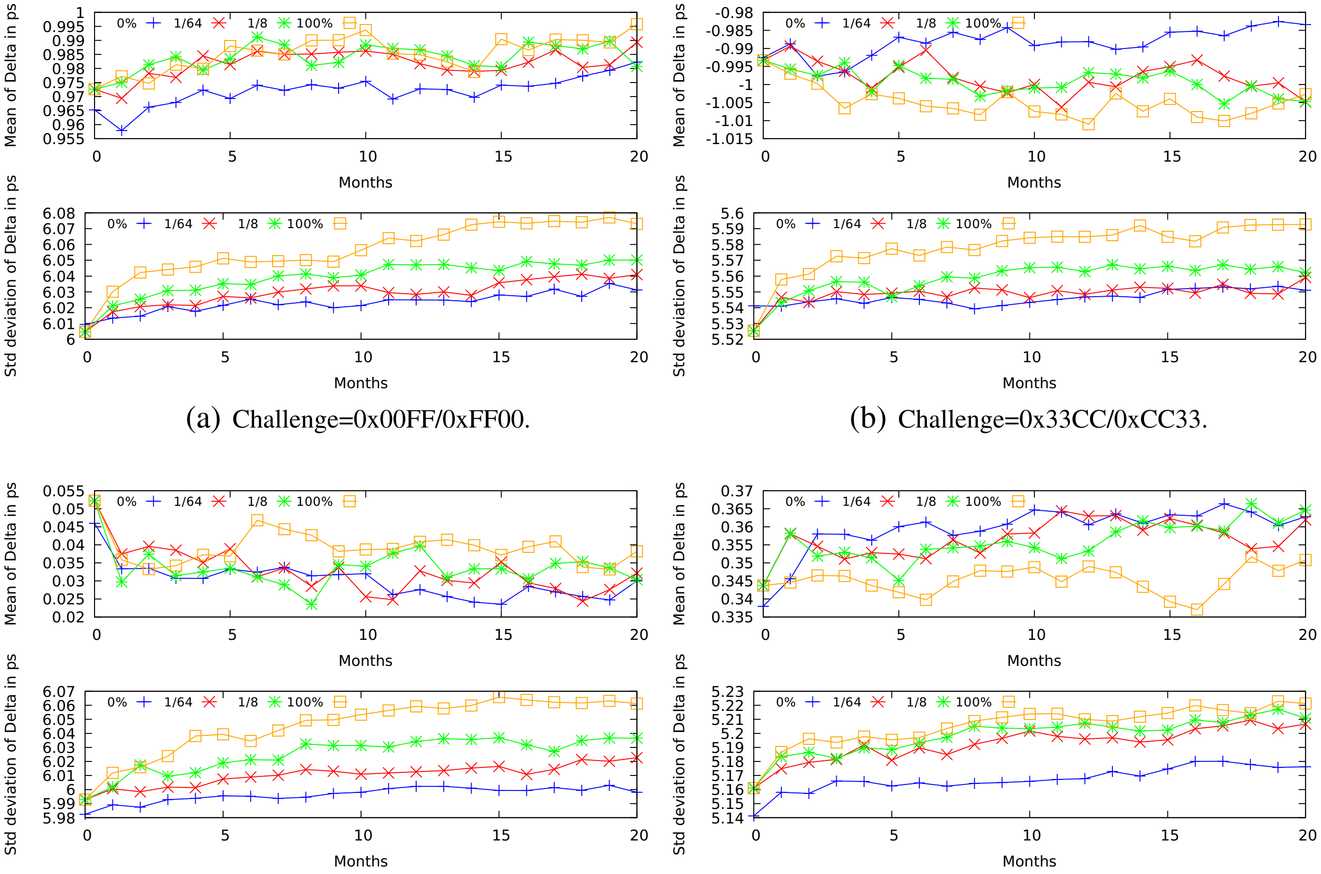
4.2 Simulation Results

Effect of Aging on Delay Parameters of Loop-PUFs The first set of results shows how aging affects the delay specifications of loop-PUFs. We show the effect of aging on the reliability of PUFs using different parameters. The first set of parameters include the mean and variance of the aging-related delay degradation of loop-PUFs over time. As shown in Fig. 7, we considered four different challenges for the loop-PUF. These challenges are Hadamard codewords and have been chosen as they are among the best challenges result in maximum entropy, as explained in Rioul et al. [37]. For each case, using the method discussed in Section 3, we evaluated the magnitude of delay change over time for all 512 loop-PUFs fed by each of these challenges. Note that *in loop-PUFs, a challenge pair always include a challenge and its complements* [37]. Each loop-PUF includes 16 delay elements. Then, we calculated the mean and variance of delay change of all PUFs over time for each challenge. As shown in Fig. 7 the mean evolves randomly and on a very tiny scale with aging. This random change can be related to the insufficient accuracy of the HSpice simulation.

The simulation result shows that aging affects the delay variance in a monotonic manner. As expected, due to the NBTI effects, a higher increase of variance is noticeable at the early stages of the PUF lifetime. Moreover, the higher switching activity, the more increase rate of the variance. In practice, switching activity results in HCI effects.

Effect of Aging on Reliability of Loop-PUFs The second set of results investigates the effect of aging on the reliability of loop-PUFs. Here, we evaluate the aging-related bit error rate (BER) for a loop-PUF running under different operating conditions as below.

As discussed in Section 2.2, in a delay chain, the measurement for a given challenge (or a challenge pair for the Loop-PUF) is the difference of the delay of the considered paths. This time (represented as t) has a normal distribution $t \sim \mathcal{N}(\hat{t}, \sigma^2)$, where $\hat{t}$ is the mean value of t and σ^2 is the variance of the noise at transistor level. In addition, $\hat{t}$ itself is normally distributed and centered: $\hat{t} \sim \mathcal{N}(0, \sigma_{PUF}^2)$, where σ_{PUF}^2 represents the PUF variance, which comes from the process mismatch of the technology.



(a) Challenge=0x00FF/0xFF00.

(b) Challenge=0x33CC/0xCC33.

(c) Challenge=0x6996/0x9669.

(d) Challenge=0x5A5A/0xA5A5.

Fig. 7 Mean and variance of delay-change for different challenge pairs in loop-PUFs

The Signal to Noise Ratio (SNR) is defined as:

$$SNR = \frac{\sigma_{PUF}^2}{\sigma^2}$$

The BER for a given challenge depends on the probability of the change of the sign of t as explained in Section 2.2. When the PUF has not been aged, the initial bit error rate, BER^0, for a given challenge is as below:

$$BER^0 = \mathbb{P}(x > \hat{t}) = \frac{1}{2}(1 - \text{erf}(\frac{\hat{t}}{\sigma\sqrt{2}}))$$

The absolute value of BER is unknown as it depends on the noise variance which is also unknown. However, it is possible to analyze its change due to aging, as σ_{PUF}^2 and $\hat{t}$ can be measured. Indeed, the evolution of σ_{PUF}^2 is equivalent to an increase of the noise variance Δ_{σ^2} assuming that SNR is invariant with aging [10]. Thereby, the BER change can be assessed by considering the ratio between the BER at time i (i represents the aging time) and at time 0 for M challenges.

$$\frac{BER^i}{BER^0} = \frac{1}{M}\sum_M \frac{\frac{1}{2}(1 - \text{erf}(\frac{\hat{t}-\Delta_{\hat{t}}^i}{\frac{(\sigma_{PUF}+\Delta_{\sigma_{PUF}}^i)}{\sqrt{SNR}}\sqrt{2}}))}{\frac{1}{2}(1 - \text{erf}(\frac{\hat{t}}{\frac{\sigma_{PUF}}{\sqrt{SNR}}\sqrt{2}}))} \tag{6}$$

Note that the value of $\hat{t}$ is found by averaging the delay differences in different paths of the delay chain. In addition, σ_{PUF} is obtained by computing the variance of $\hat{t}$ considering the set of challenges which are Hadamard Code words. In this equation, $\Delta_{\hat{t}}^i$ represents the change of $\hat{t}$ due to aging at time i (compared to time 0). Moreover, $\Delta_{\sigma_{PUF}}^i$ denotes the aging-induced change of σ_{PUF} at time i (compared to time 0).

Figure 8 represents the impact of aging due to both HCI and NBTI on the delay chain of a loop-PUF with 16 elements during 20 months of usage. The results correspond to the mean from 16 challenges, and are expressed in terms of BER with $BER^0 = 10^{-2}$, which corresponds to the measurement from the real silicon, presented in Section 4.3, for a PUF without any activity and PUFs with 1.56% (= 1/64), 12.5% (= 1/8), and 100% activities in 45 °C. The SNR was considered as 25, as we adopted the SNR value from our real-silicon results. We can see that the BER increase is very limited for this SNR.

As shown in this figure, the BER is $\approx$ 1.3 × 10^{-2} for a loop-PUF with 100% activity after 20 months of operation. Although in this situation, the BER increased $\approx$ 30% in 20 months, its value after 20 months of aging is not considerable. This confirms that loop-PUFs are little sensitive to aging. Another observation taken from this figure is that the reliability degradation under 100% of switching activity is more than the one under 0% of switching activity. In particular, as shown in Fig. 8, the BER increases around 17% over a time period of 20 months for a device with no activity while changes 30% for the same device with 100% activity in 20 months. This can be explained by the fact that under dynamic conditions, the BER change is due to the combined effect of the two aging mechanisms (NBTI and HCI). While, under static conditions just NBTI degradation is observed. As expected, the degradation increase is more significant under the dynamic conditions (100% of activity). However, as mentioned BER value is not considerable, and thereby, the conclusion is that loop-PUFs are not considerably affected by aging.

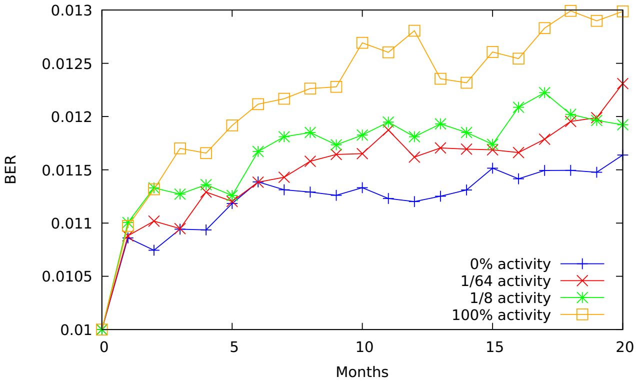


Fig. 8 Simulated mean of BER for 16 challenges and different activity rates (Temperature=45 °C)

Effect of Aging on Reliability of Arbiter-PUFs This set of results deals with the reliability degradation of arbiter-PUFs due to aging. The BER formula presented earlier for the loop-PUFs, also stands for the delay-chain of the arbiter-PUFs. However, to assess the complete BER for an arbiter-PUF, the aging of the arbiter should be considered as well.

We consider the arbiter realized by a NAND-based RS-latch, as represented in Fig. 3. The BER/BER^0 formula in Eq. 6 could apply by considering the evolution of the timing difference between the RESET and SET inputs of the latch to get the metastable state. However, the BER can be directly accessible as it corresponds to the bit-flips of the latch. This BER metrics is well suited to notice that the aging has more impact on the arbiter than the delay chain. Indeed, as shown in Fig. 9, the arbiter deployed in an arbiter-PUF is highly sensitive to aging as the bit-flips seriously increase after 20 months, especially during the first month. These bit-flips are not related to the delay chain, i.e., they are exclusively initiated by the aging of the RS-latch. An offset of the BER curve has been inserted to consider the real BER measured from the real silicon.

The results confirm the high impact of aging in an RS-latch. As shown, the RS-latch experiences the BER $\approx$ 0.1 after 20 months of aging at high switching rate. As shown,

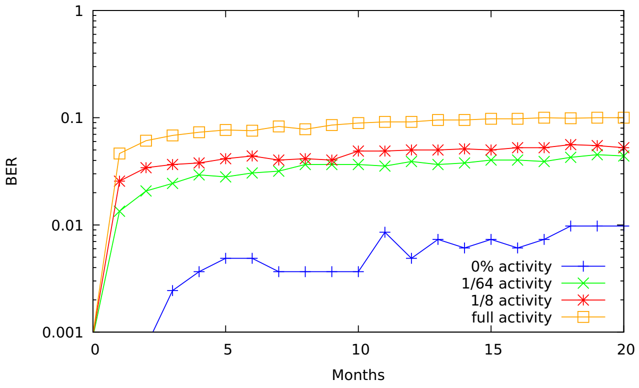


Fig. 9 BER evolution of the arbiter in the arbiter-PUF with different activity rates (Temperature = 45 °C)

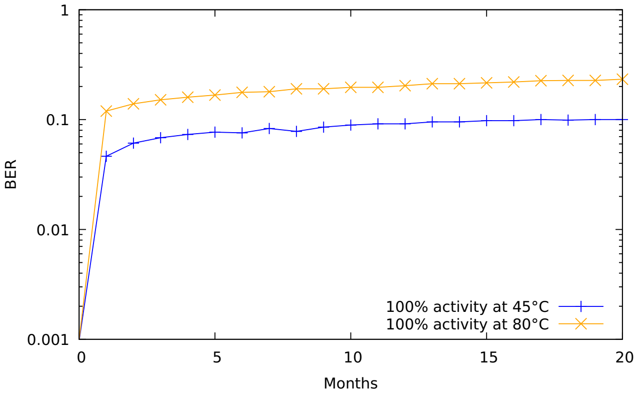


Fig. 11 The effect of temperature in aging of arbiter

in the case of high activity the BER experiences 100 times increase after 20 months (changing from 10^{-3} to 10^{-1}). As shown in Fig. 9, the BER in the case of high switching activity can be 11 times worse than the case without activity after 20 months. Note that in this figure values are shown in logarithmic scale. A potential cause is the imbalance of the states between the two NAND gates residing in the arbiter. Indeed, if the output of one of these NAND gates gets the value of ‘1’, it experiences more NBTI aging than the other one since in the former, the PMOS network conducts. This should also affect the set-up time at the sampling time, such that the state changes at the next sampling time when the difference of time between the two paths is very small.

Robustness against Temperature Variations The fourth set of results represents how the increase of temperature affects aging-related degradation in the considered PUFs. As shown in Figs. 10 and 11, the aging in both the delay chain deployed in arbiter-PUFs (as well as loop-PUFs) and the arbiter itself increases in a higher temperature. As depicted, the BER increase is around 30% for the delay chain over 20 months of usage in 45 °C while it is ≈ 50% for 20 months usage in 80 °C. However, due to the magnitude of BER, the effect is still not considerable. For the arbiter itself, as shown in Fig. 11, the degradation can be 2.3 times worse in 80 °C compared to 45 °C for full activity. Due to the magnitude of the BER, as mentioned, arbiter-PUFs are highly sensitive to aging.

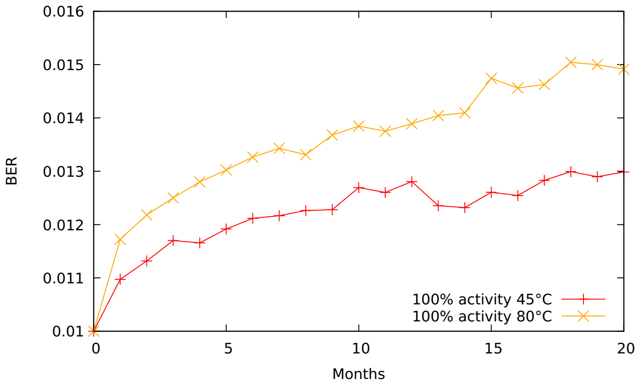


Fig. 10 The effect of temperature in aging of delay chain

Extrapolation Speedup As discussed in Section 3.2, to avoid conducting thousands of Monte Carlo simulations for PUFs with large number of delay elements, we extract the aging results of single delay-element PUFs and utilize our proposed extrapolation scheme to evaluate the effect of aging in PUFs with large number of elements. Table 2 shows the speedup of using our extrapolation method over using traditional Monte Carlo simulations for PUFs with large number of delay elements. In this table, the second row provides the time required for conducting HSpice simulation of a PUF with one delay element. The third, forth and fifth rows present the time required for conducting HSpice simulation of a PUF with 16, 32 and 64 delay elements, respectively.

The sixth row of Table 2 presents the time needed using our *extrapolation scheme* for simulating a PUF with 8192 elements. The 7^{th}, 8^{th} and 9^{th} rows show the time required for HSpice simulations of a PUF with 8192 elements too. In particular, 7^{th} row depicts the time needed for Monte Carlo simulation of 512 PUFs each included 16 delay elements (so-called traditional scheme as no extrapolation is used here and all delay parameters are extracted only by simulations). Similarly, the 8^{th} row shows the time required for Monte Carlo simulations of 256 32-element PUFs and the 9^{th} row depicts the time required for Monte Carlo simulations of 128 64-element PUFs. Finally, the last three rows show the speedup of using the extrapolation scheme over traditional scheme (using Monte Carlo simulations

Table 2 HSpice Monte Carlo(MC) simulations speed comparison

	Arbiter-PUF	Loop-PUF
Time[1] per MC simulation of one 1-element PUF	6	1
Time per MC simulation of one 16-element PUF	131	58
Time per MC simulation of one 32-element PUF	512	198
Time per MC simulation of one 64-element PUF	2080	773
Total time of MC simulations using *extrapolation* for a PUF with 8192 elements	49152	8192
Total time of MC simulations using 512 16-element PUFs (no extrapolation)	67072	29696
Total time of MC simulations using 256 32-element PUFs (no extrapolation)	131072	50688
Total time of MC simulations using 128 64-element PUFs (no extrapolation)	266240	98944
Speedup of extrapolation scheme vs. traditional		
Monte-Carlo simulations using 16-element PUFs	1.36	3.63
Speedup of extrapolation scheme vs. traditional		
Monte-Carlo simulations using 32-element PUFs	2.67	6.19
Speedup of extrapolation scheme vs. traditional		
Monte-Carlo simulations using 64-element PUFs	5.42	12.08

[1] In this table, all time values are represented in Second

with different PUF elements). As depicted, for PUFs with 64-delay elements, the speedup is more than 12 for loop-PUFs (5 for arbiter-PUFs). The results confirms that the speed up of our extrapolation scheme is more significant in loop-PUFs than arbiter-PUFs. This is related to the arbiter (RS latch) residing in the arbiter PUFs as well as the twice number of delay-elements available in these PUFs compared to loop-PUFs. The results confirm the high efficiency of the proposed extrapolation scheme. In this paper, all experiments were performed on an 8-processor quad-core Intel server running at 2.80 GHz with 32 GB of memory.

Change of PUF Metrics Due to Aging In practice, the quality of a PUF is assessed using three metrics: reliability, uniqueness, and randomness. These metrics have been discussed in details in [13] for PUFs. The randomness of a PUF deals with the unpredictability of PUF responses. The uniqueness shows how well a single PUF is differentiated from other PUFs based on its challenge/response pairs. Finally, as mentioned earlier, reliability of a PUF depicts how stable a PUF response is over time and in different environmental conditions (e.g., change of temperature).

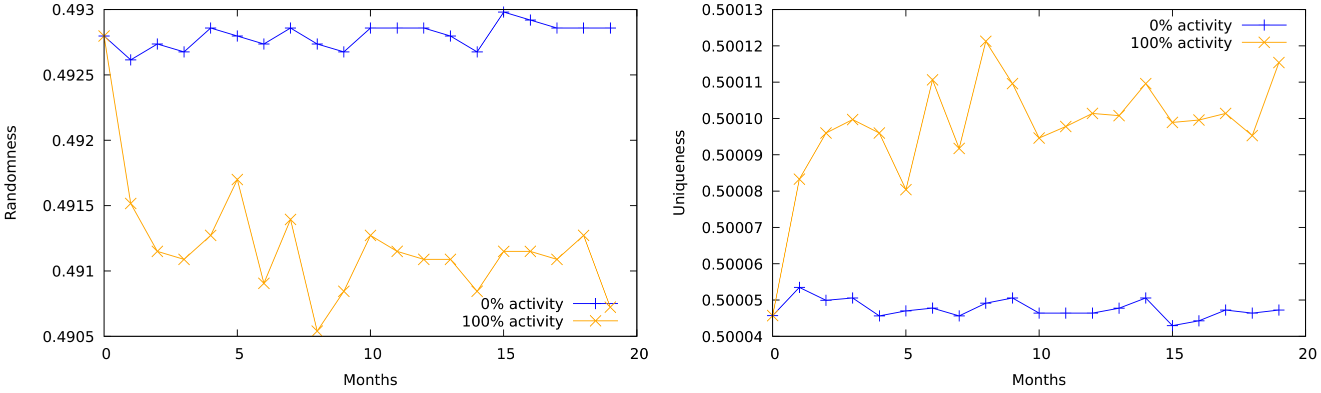
Previous results extensively investigated the reliability of loop-PUFs and arbiter-PUFs w.r.t aging and in different temperatures. As the results depicted, loop-PUFs are reliable against aging while in arbiter-PUFs challenge/response pairs change over time. As loop-PUFs are reliable and their challenge/response pairs do not change over time, we can deduce that their randomness and uniqueness of loop-PUFs will not change over time either, i.e., the randomness and uniqueness of a used (e.g., aged) loop-PUF will be similar to the randomness and uniqueness of a new (not-used) Loop-PUF. For the arbiter-PUF, the challenge-response pairs change over time. Therefore, the randomness and uniqueness may (or may not) change. The amount of change can be extracted easily by finding Hamming distances of responses over time to evaluate the uniqueness and by evaluating the avalanche effect using statistical tests to evaluate the randomness [39].

Figure 12 shows the change of the randomness and uniqueness of an arbiter-PUF in 80 °C over time. The results have been extracted for different PUF activities. As shown in this figure, the effect of aging on the randomness and uniqueness of arbiter-PUFs is not considerable (i.e., before and after aging these values are around 50%). In addition, as depicted, switching activity has a slight impact on the and uniqueness of these PUFs. The takeaway point from this observation is that the aging induced bit-flips in arbiter-PUFs similarly affect the state ‘0’ and state ‘1’ of the arbiter, i.e., the number of bits whose value changed from ‘0’ to ‘1’ due to aging is approximately similar to the number of bits whose state changed from ‘1’ to ‘0’. Thereby, although aging resulted in reliability degradation, it didn’t change the randomness and uniqueness metrics.

4.3 Real Silicon Results

In this section, we present the aging results extracted from real-silicon. We first present the observations for loop-PUFs and then we discuss the arbiter-PUFs.

Loop-PUFs An ASIC with 49 loop-PUFs composed of 64 delay elements was implemented in 65 nm technology.



(a) randomness

(b) uniqueness

Fig. 12 Effect of aging on the randomness and uniqueness of arbiter-PUFs over time

Figure 13 shows the layout with a 7 × 7 loop-PUF matrix which makes up the largest part in the upper right-hand corner of the layout.

The circuit has been placed on a PCB and put in a laboratory oven adjusted at 85 °C. The power supply has been set to 2.0 V instead of the nominal voltage of 1.2 V. In practice, these operating conditions accelerate the NBTI and HCI effects [15, §5.3]. The test procedure is described in Protocol 1 which corresponds to cycles of 24 hours. The challenges used for the test are Hadamard codewords, as they allow to obtain the maximum entropy of the loop-PUF, as explained in [37].

The first 8 PUFs (PUF_0 to PUF_7) are always running (100% activity), whereas PUF_8 to PUF_{15} run 1/8 of the time, and PUF_{16} to PUF_{31} run 1/64 of the time. PUF_{32} to PUF_{48} never run. These differences in switching activity (X%) allows us to investigate the impact of switching activity on PUF aging. Every 24 hours and during one hour, the device is back in its typical environment and all the challenges are used to measure the PUF values. Figure 14 represents the aging impact during 100 days in terms of the ratio of BER for the Loop-PUF with the 4 different groups of PUFs each having different activities as mentioned above. The BER^0 is 10^{-2} and the SNR was assessed as 25.

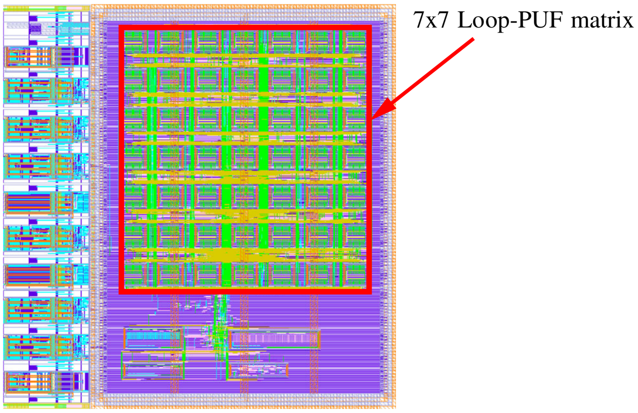


Fig. 13 Layout of the test chip embedding 49 loop-PUFs

Protocol 1 Aging acceleration protocol.

```
Input: Non aged device
Output: Aged device
STEP 1: Stress during 23 hours ..................
$V_{dd} \leftarrow 2.0$ V, $T$°C $\leftarrow$ 85 °C
Challenge $C_i \leftarrow$ 0x00000000FFFFFFFF
Always measure $PUF_i$, for $i \in \{0, \dots, 7\}$
Measure $PUF_j$ every 1/8 time, for $j \in \{8, \dots, 15\}$
Measure $PUF_k$ every 1/64 time, for $k \in \{16, \dots, 31\}$
STEP 2: Evaluation during 1 hour ................
$V_{dd} \leftarrow 1.2$ V, $T$°C $\leftarrow$ 20 °C
Measurement of the 49 Loop-PUFs with 63 Hadamard
    Challenges [35]
Go to STEP 1
```

The real silicon results shown in Fig. 14 confirms the simulation results in showing that the effect of aging in delay chains is not considerable. In addition, even with the high level of noise, and similar to the simulation results extracted for delay chains in previous section, we observe that for real silicon, aging degradation in case of 100% switching activity is more than other cases. In particular, reliability degradation is around 1.8 times worse in the delay chains when there is full activity compared to zero activity. The degradation under 1/8 and 1/64 of switching activity is less than the one under static condition.

Arbiter-PUFs The Arbiter-PUF and Loop-PUF share the same delay elements. The common PUF cell is configured as either two Arbiter-PUFs of 16 elements, or a Loop PUF of 64 elements (totally 64 elements in each case). The

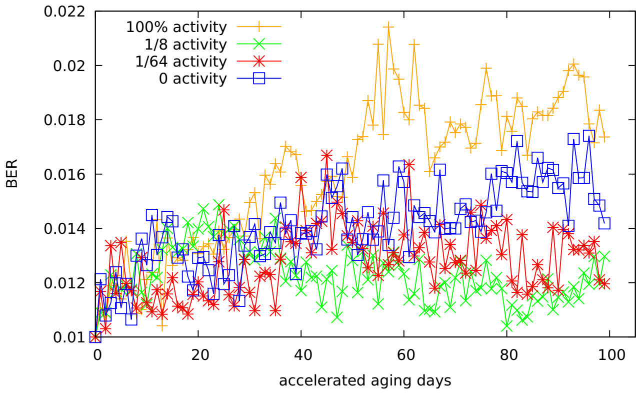


Fig. 14 BER evolution of the delay chain for different activity rates

results on real silicon for the arbiter-PUFs are illustrated in Fig. 15. As shown, the BER change is very fast during the first days of aging and then stabilizes rapidly. This shows that the NBTI is the main aging factor affecting the arbiter residing in the arbiter-PUF. Both simulation results shown in Fig. 9 and the real silicon results shown in Fig. 15 depict the high senility of arbiter to aging. However, the results for real-silicon arbiter-PUF do not thoroughly follow what we expected from simulations, i.e., in the simulation results shown in Fig. 9, BER experiences its highest value when the PUF is fully active. Similarly for a non-active PUF, the BER has its lowest value. While, the real-silicon results shown in Fig. 15 do not follow the same trend. In practice, in the results shown in Fig. 15, the BER of PUFs with different activities are very similar. This can be explained by the high level of noise in real silicon experiments. Moreover, in our experiment, the PUFs with no activity have been placed near other IPs in the ASIC chip, and thereby, experienced more measurement noise. In the layout shown in Fig. 13, the PUF with the highest index (lowest activity as shown in Protocol 1) has been located at the bottom of the chip and close to the control circuitry and other blocks.

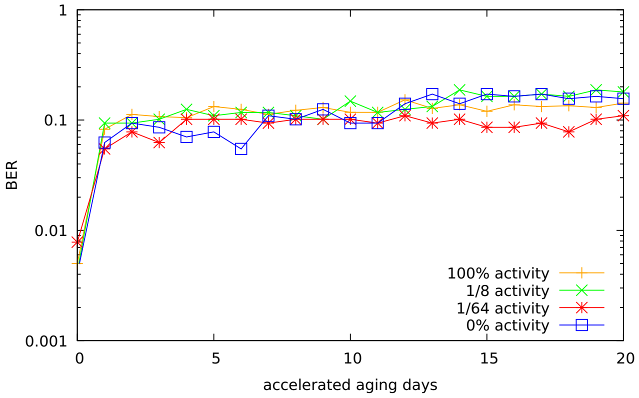


Fig. 15 BER evolution of the arbiter for different activity rates

5 Mitigating Aging Effects

As the experimental results presented in Section 4 showed, in the considered PUFs, delay chains are not sensitive to aging. However, the arbiter resided in an arbiter-PUF (Fig. 3) is highly affected by aging. Thereby, to increase the reliability of arbiter-PUFs against aging, we need to deactivate the arbiters embedded in these PUFs when the PUFs are not queried. Accordingly, we can replace the arbiter shown in Fig. 3 with the aging-resilient arbiter shown in Fig. 16. As depicted, for aging-resilient arbiter, to generate each response bit, we only need to add eight transistors to the corresponding delay chains (four transistors to each chain). Thereby, the area overhead is highly negligible compared to the total size of the PUF. A semi-similar aging-resilient structure proposed in [36] for mitigating the aging effects in ring-oscillator PUFs. However, due to the architecture of ring-oscillator PUFs, in those PUFs two transistors need to be inserted for each delay elements in each delay chain, i.e., totally $2 \times n \times m$ extra transistors needed for a PUF with n chain each including m delay elements. While in our aging-resilient arbiter PUF, we only need $4 \times n$ extra transistors to make the PUF resilient against aging. As Fig. 16 depicts, when the PUF is queried, signal EN gets the value of ‘1’ and therefore, the arbiter operates as a simple RS-latch (similar to the arbiter shown in Fig. 3). However, when the PUF is not queried, EN is ‘0’ and both NAND gates are fed with “11” (i.e., each input of the NAND gate is fed with $VDD - V_{th}$ where V_{th} is the threshold voltage of the NMOS transistor connected between VDD and that input.). This turns off the PMOS transistors resided in the arbiter, and thereby mitigates the NBTI and HCI effects in the arbiter.

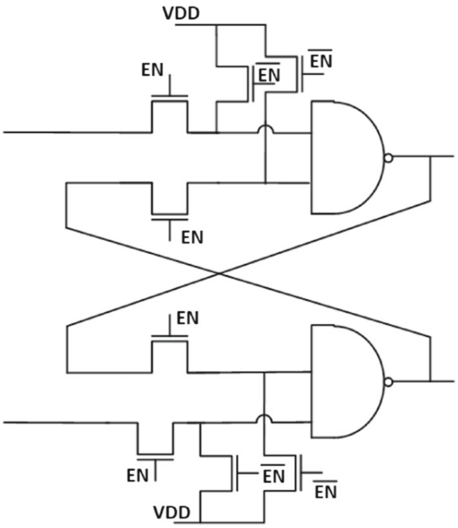


Fig. 16 An aging-resilient arbiter

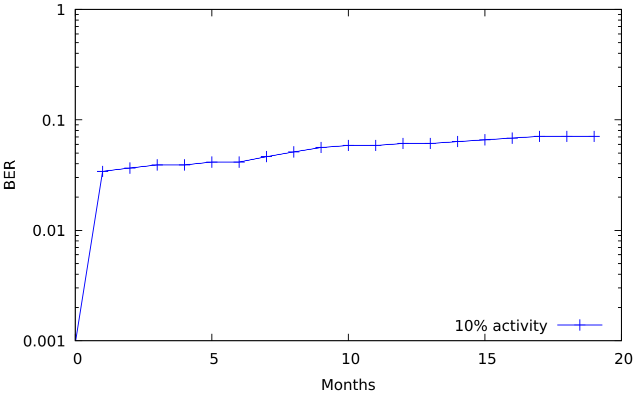


Fig. 17 BER evolution of the proposed aging-resilient arbiter

Figure 17 represents the simulation results demonstrating the impact of aging due to both HCI and NBTI on the proposed aging-resilient arbiter in 80 °C during 20 months of usage. In this experiment, the arbiter is activated (i.e., EN gets the value of '1') in 10% of the PUF usage time, while the PUF itself (the chain) was fully active (100% activity). Comparing this result with the results shown in Fig. 11 confirms the aging resiliency of the proposed arbiter. As shown, the aging-resilient arbiter experiences the BER $\approx$ 0.07 after 20 months of aging while in the original arbiter BER reaches to $\approx$ 0.24 after 20 months of aging. Thereby, by using the proposed arbiter, BER is decreased significantly.

6 Conclusion

In this paper we investigate the impact of aging on two types of delay-PUFs, namely arbiter-PUFs and loop-PUFs. In particular, we study the impact of switching activity and temperature on the reliability of these targeted PUFs. The simulation results show that the delay chains of the these PUFs are slightly impacted by aging, while aging has a high impact on the arbiters resided in the arbiter-PUFs. Aging effect is exacerbated with high switching activity. Simulation results show that the BER in arbiters can increase up to 10^{-1} after 20 months of aging with high switching activity whereas it reaches to 10^{-2} without activity. The results confirms that loop-PUFs, and more generally the PUFs with only combinatorial logics, are less sensitive to aging. However, sequential PUFs such as arbiter-PUFs, which use a memory cell, e.g. an RS-latch, are much more impacted by aging, and in particular, are more sensitive to the temperature. This confirms the necessity of developing anti-aging mechanisms for arbiter-PUFs. Accordingly, we designed an aging-resilient arbiter. The experimental results shows the high resiliency of the proposed arbiter against aging comparing to the original arbiter used in this study. This paper extended the experiments to real silicon. The real silicon observations validate the simulation results in confirming the sensitivity of arbiter-PUFs to aging and the resistance of loop-PUFs against aging.

We will extend the scope of this paper in our future research and investigate the impact of aging on other type of delay-PUFs including RO-SUM PUFs.

Acknowledgments This work was partly supported by Institute for Information & communications Technology Promotion (IITP) grant funded by the Korea government (MSIT) (No.2016-0-00399, Study on secure key hiding technology for IoT devices [KeyHAS Project]). It was also partly supported by a Faculty Fellowship Award from University of Maryland Baltimore County.

References

1. Alam MA, Kufluoglu H, Varghese D, Mahapatra S (2007) A comprehensive model for PMOS NBTI degradation: recent progress. Microelectron Reliab 47(6):853–862
2. Bhargava M, Mai K (2014) An efficient reliable PUF-based cryptographic key generator in 65nm CMOS. In: Design, automation & test in Europe (DATE), pp 70:1–70:6
3. Cao Y, Zhang L, Chang C, Chen S (2015) A low-power hybrid RO PUF with improved thermal stability for lightweight applications. IEEE Trans CAD Integrated Circ Syst 34(7):1143–1147
4. Cha S, Chen C-C, Liu T, Milor LS (2014) Extraction of threshold voltage degradation modeling due to negative bias temperature instability in circuits with I/O measurements. In: VLSI test symposium (VTS), pp 1–6
5. Cherif Z, Danger J-L, Guilley S, Bossuet L (2012) An easy-to-design PUF based on a single oscillator: the Loop PUF in DSD
6. Cherif Z, Danger J-L, Lozac'h F, Mathieu Y, Bossuet L (2013) Evaluation of delay PUFs on cmos 65 nm technology: ASIC vs FPGA. In: International workshop on hardware and architectural support for security and privacy (HASP), pp 4:1–4:8
7. Ching SP, Ping CT, Sun YH (2008) Studies of the critical LDD area for HCI improvement. In: International conference on semiconductor electronics, pp 622–625
8. Crupi F, Pace C, Cocorullo G, Groeseneken G, Aoulaiche M, Houssa M (2005) Positive bias temperature instability in nMOSFETs with ultra-thin hf-silicate gate dielectrics. Microelectron Eng 80:130–133
9. Gassend B, Clarke DE, van Dijk M, Devadas S (2002) Silicon physical random functions. In: ACM conference on computer and communications security, CCS 2002, pp 148–160
10. Gerrer L, Ding J, Amoroso SM, Adamu-Lema F, Hussin R, Reid D, Millar C, Asenov A (2014) Modelling RTN and BTI in nanoscale MOSFETs from device to circuit: a review. Microelectron Reliab 54(4):682–697
11. Guajardo J, Kumar SS, Schrijen G-J, Tuyls P (2007) FPGA intrinsic PUFs and their use for IP protection. In: Cryptographic hardware and embedded systems (CHES), pp 63–80
12. Holcomb DE, Burleson WP, Fu K (2009) Power-up SRAM state as an identifying fingerprint and source of true random numbers. IEEE Trans Comput 58(9):1198–1210
13. Hori Y, Yoshida T, Katashita T, Satoh A (2010) Quantitative and statistical performance evaluation of arbiter physical unclonable functions on FPGAs. In: International conference on reconfigurable computing and FPGAs, pp 298–303

14. Hosey A, Rahman MT, Xiao K, Forte D, Tehranipoor M (2014) Advanced analysis of cell stability for reliable SRAM PUFs. In: Asian test symposium (ATS), pp 348–353
15. JEDEC (2011) JEP122G: Failure mechanisms and models for semiconductor devices, http://www.jedec.org/standards-documents/docs/jep-122e
16. Karimi N, Danger J-L, Guilley S, Lozach F (2016) Predictive aging of reliability of two delay PUFs. In: Security, privacy, and applied cryptography engineering (SPACE), pp 213–232
17. Karimi N, Slimani J-LDM, Guilley S (2017) Impact of the switching activity on the aging of delay-PUFs. In: European Test Symp. (ETS)
18. Karimi N, Guilley S, Danger J-L (2018) Impact of aging on template attacks. In: Proceedings of the ACM great lakes symposium on VLSI (GlSVLSI), pp 455–458
19. Karimi N, Danger J-L, Guilley S (2018) On the effect of aging in detecting hardware trojan horses with template analysis. In: Proceedings of the international symposium on on-line testing and robust system design (IOLTS)
20. Kim KK (2015) On-chip delay degradation measurement for aging compensation. Indian J Sci Technol 8:8
21. Kirkpatrick MS, Bertino E (2010) Software techniques to combat drift in PUF-based authentication systems. In: Workshop on secure component and system identification (SECSI), p 9
22. Krishnan AT, Chancellor C, Chakravarthi S, Nicollian PE, Reddy V, Varghese A, Khamankar R, Krishnan S (2005) Material dependence of hydrogen diffusion: implications for NBTI degradation. In: International electron devices meeting (IEDM), pp 688–691
23. Kufluoglu H, Alam MA (2007) A generalized reaction-diffusion model with explicit h-h2 dynamics for negative-bias temperature-instability (NBTI) degradation. IEEE Trans Electron Dev 54(5):1101–1107
24. Lu Y, Shang L, Zhou H, Zhu H, Yang F, Zeng X (2009) Statistical reliability analysis under process variation and aging effects. In: Design automation conference (DAC), pp 514–519
25. Maes R, van der Leest V (2014) Countering the effects of silicon aging on SRAM PUFs. In: International symposium hardware-oriented security and trust (HOST), pp 148–153
26. Mahapatra S, Saha D, Varghese D, Kumar P (2006) On the generation and recovery of interface traps in MOSFETs subjected to NBTI, FN, and HCI stress. IEEE Trans Electron Dev 53(7):1583–1592
27. Maiti A, Schaumont P (Sep 2014) The impact of aging on a physical unclonable function. IEEE Trans Very Large Scale Integrated Syst (TVLSI) 22(9):1854–1864
28. Mispan MS, Halak B, Zwolinski M (2016) NBTI aging evaluation of PUF-based differential architectures. In: International symposium on on-line testing and robust system design (IOLTS), pp 103–108
29. Mizan E (2008) Efficient fault tolerance for pipelined structures and its application to superscalar and dataflow machines. Ph.D. thesis, Electrical and Computer Engineering Dept. University of Texas At Austin
30. Morozov S, Maiti A, Schaumont P (2010) An analysis of delay based PUF implementations on FPGA. In: Reconfigurable computing: architectures, tools and applications (ARC), pp 382–387
31. Nangate 45nm open cell library, http://www.nangate.com. (Last Accessed 1 May 2016)
32. Nunes C, Butzen PF, Reis AI, Ribas RP (2013) BTI, HCI and TDDB aging impact in flip-flops. Microelectron Reliab 53(6-11):1355–1359
33. Oboril F, Tahoori MB (2012) Extratime: modeling and analysis of wearout due to transistor aging at microarchitecture-level. In: Dependable systems and networks (DSN), pp 1–12
34. Pelgrom MJ, Duinmaijer AC, Welbers AP (1989) Matching properties of MOS transistors. IEEE J Solid State Circuits 24(5):1433–1439
35. Rahman MT, Forte D, Fahrny J, Tehranipoor M (2014) ARO-PUF: an aging-resistant ring oscillator PUF design. In: Design, automation test in Europe conference (DATE), pp 1–6
36. Rahman MT, Rahman F, Forte D, Tehranipoor M (July 2016) An aging-resistant RO-PUF for reliable key generation. IEEE Trans Emerg Topics Comput 4(3):335–348
37. Rioul O, Solé P, Guilley S, Danger J-L (2016) On the entropy of physically unclonable functions. In: IEEE international symposium on information theory (ISIT). Barcelona
38. Rodriguez R, Stathis J, Linder B (2003) Modeling and experimental verification of the effect of gate oxide breakdown on CMOS inverters. In: IEEE international reliability physics symposium, pp 11–16
39. Rukhin A et al (2001) A statistical test suite for random and pseudorandom number generators for cryptographic applications. National Institute of Standards and Technology (NIST)
40. Schroder DK (2007) Negative bias temperature instability: what do we understand? Microelectron Reliab 47(6):841–852
41. Suh GE, Devadas S (2007) Physical unclonable functions for device authentication and secret key generation. In: Design automation conference (DAC), pp 9–14
42. Sutaria KB, Velamala JB, Ramkumar A, Cao Y (2015) Compact modeling of BTI for circuit reliability analysis. In: Circuit design for reliability, pp 93–119
43. Synopsys (2016) HSPICE user guide: basic simulation and analysis
44. Tiwari A, Torrellas J (2008) Facelift: hiding and slowing down aging in multicores. In: International symposium on microarchitecture, pp 129–140
45. Wang W, Yang S, Bhardwaj S, Vrudhula S, Liu F, Cao Y (2010) The impact of NBTI effect on combinational circuit: modeling, simulation, and analysis. IEEE Trans Very Large Scale Integr Syst 18(2):173–183
46. Xiao K, Rahman MT, Forte D, Huang Y, Su M, Tehranipoor M (2014) Bit selection algorithm suitable for high-volume production of SRAM-PUF. In: International symposium on hardware-oriented security and trust (HOST), pp 101–106
47. Yilmaz M, Chakrabarty K, Tehranipoor M (2008) Test-pattern grading and pattern selection for small-delay defects. In: VTS, pp 233–239
48. Zafar S, Kim Y, Narayanan V, Cabral C, Paruchuri V, Doris B, Stathis J, Callegari A, Chudzik M (2006) A comparative study of NBTI and PBTI (charge trapping) in SiO2/HfO2 stacks with FUSI, TiN, Re gates. In: Symposium on VLSI technology, pp 23–25

Naghmeh Karimi received the B.Sc., M.Sc., and Ph.D. degrees in Computer Engineering from the University of Tehran, Iran in 1997, 2002, and 2010, respectively. She was a visiting researcher at Yale University, USA between 2007 and 2009, and a post-doctoral researcher at Duke University, USA during 2011-2012. She has been a visiting assistant professor at New York University and Rutgers University between 2012 and 2016. She joined University of Maryland Baltimore County as an assistant professor in 2017 where she leads the SECure, REliable and Trusted Systems (SECRETS) research lab. She has published three book chapters and authored/co-authored more than 40 papers in referred conference proceedings and journal manuscripts. Her current research interests include hardware security, design-for-trust, design-for-testability, and design-for-reliability.

Jean-Luc Danger is a full Professor at TELECOM ParisTech (www.telecom-paristech.fr/). He is the head of the digital electronic system research team involved in Research in security/safety of embedded systems, configurable architectures, and implementation of complex algorithms in ASICs or FPGAs. He authored more than 200 scientific publications and 20 patents in architectures of embedded systems and security, and is the co-founder of the Secure-IC company (www.secureic.com). He received his engineering degree in Electrical Engineering from Ecole Supérieure d'Electricité in 1981. After 12 years in industrial laboratories (PHILIPS,NOKIA), he joined TELECOM ParisTech in 1993 where he became full professor in 2002. His personal research interests are trusted computing, cyber-security, random number generation, and protected implementations in novel technologies.

Sylvain Guilley is CTO at Secure-IC, a company offering security for embedded systems. Within Secure-IC, He is also director of "Threat Analysis" and "Think Ahead" business lines, which develop respectively security evaluation tools and advanced research. Sylvain is also professor at TELECOM-ParisTech, associate research at Ecole Normale Supérieure (ENS), and adjunct professor at the Chinese Academy of Sciences (CAS). His research interests are trusted computing, cyber-physical security, secure prototyping in FPGA and ASIC, and formal/mathematical methods. Since 2012, he organizes the PROOFS workshop, which brings together researchers whose objective is to increase the trust in the security of embedded systems. He is also lead editor of international standards, such as ISO/IEC 20897 (Physically Unclonable Functions) and ISO/IEC 20085 (Calibration of non-invasive testing tools). Sylvain has co-authored 200+ research papers and filed 30+ patents. He is member of the IACR, the IEEE and senior member of the CryptArchi club. He is an alumni of Ecole Polytechnique and TELECOM-ParisTech.