Predictive Aging of Reliability of two Delay PUFs

Naghmeh Karimi[1], Jean-Luc Danger[2,3], Florent Lozac'h[3], and Sylvain Guilley[2,3]

[1] ECE Department, Rutgers University, Piscataway, NJ, USA 08854
naghmeh.karimi@rutgers.edu
[2] LTCI, CNRS, Télécom ParisTech, Université Paris-Saclay, 75013 Paris, France
firstname.lastname@telecom-paristech.fr
[3] Secure-IC SAS, 35510 Cesson-Sévigné, France
firstname.lastname@secure-ic.com

Abstract. To protect integrated circuits against IP piracy, Physically Unclonable Functions (PUFs) are deployed. PUFs provide a specific signature for each integrated circuit. However, environmental variations, (e.g., temperature change), power supply noise and more influential IC aging affect the functionally of PUFs. Thereby, it is important to evaluate aging effects as early as possible, preferentially at design time. In this paper we investigate the effect of aging on the stability of two delay PUFs: arbiter-PUFs and loop-PUFs and analyze the architectural impact of these PUFS on reliability decrease due to aging.
We observe that the reliability of the arbiter-PUF gets worse over time, whereas the reliability of the loop-PUF remains constant. We interpret this phenomenon by the asymmetric aging of the arbiter, because one half is active (hence aging fast) while the other is not (hence aging slow). Besides, we notice that the aging of the delay chain in the arbiter-PUF and in the loop-PUF has no impact on their reliability, since these PUFs operate differentially.

1 Introduction

With the advancement of VLSI technology, people are increasingly relying on electronic devices and in turn integrated circuits (ICs). Therefore, it is essential to assure the security of the sensitive tasks performed by such devices and to guarantee the security of information stored within these devices.

Having a unique identifier for each electronic chip offers many security benefits. If, for example, the chip is in a smartphone, the identifier can be used to associate the device with a specific service. The identifier can also be used to thwart overbuilding since it can be recorded at fabrication and can later be checked against a whitelist (in this way, overproduced or counterfeited chips can be detected). However, for the identifier to be trusted, it must meet some security properties: essentially, it must be *unique* and it must not be *tamperable*. Physically Unclonable Functions (PUFs) are known as technical solutions [1–3] as they can generate volatile secret keys for a system [4].

A PUF signature is used either via a Challenge-Response Pair (CRP) protocol for authentication, or to generate a private key or random variable in a ciphering operation. A PUF can avoid the use of digital memory to store a key imposed by the IC manufacturer or the user. Hence PUFs are well suited in low-cost devices such as the RFIDs or smartcards. [5]. In practice, PUFs have different applications including cryptographic key generation, device authentication, Intellectual Property (IP) protection, etc.

Indeed, PUFs benefit from process variations which occur during the manufacturing of integrated circuits and thereby each PUF generates a unique signature extracted based on physical characteristics of the circuit elements. The unique behavior after fabrication stems from a *static randomness* due to technological dispersion. It is a well known source of mismatch in electronics circuits design and was characterized by Pelgrom [6] to follow a normal distribution.

The PUF responses are also subject to *dynamic randomness* due to measurement noise, which is detrimental to the reliability of the PUF measurement. For this reason, it is important in practice to increase the signal-to-noise ratio (SNR). In a so-called SRAM-PUF [7] which consists in one SRAM memory bit booting up at either value 0 or 1, it seems difficult to improve the SNR except by repeating measurements, which demands a power down between each measurement. In a delay-PUF [8], n elements are chained, and the total delay of the chain is measured. The SNR is then increased by a factor n as the signal power grows linearly with n. Because of this property, we focus in this paper on delay-PUFs, namely loop-PUF [5] and arbiter-PUF [2, §2.2].

PUFs can be deployed as identifiers of electronic chips if the responses (keys) generated by PUFs are reliable and do not change over time. However, similar to other ICs, PUFs are vulnerable to aging mechanisms that jeopardize their reliability over time. In practice, with the advancement of VLSI technology and moving towards nano technologies, run time degradation mechanisms such as Negative-Bias Temperature-Instability (NBTI), Hot-Carrier Injection (HCI), and gate Oxide Breakdown (OB) play a critical role in urging circuits malfunctions [9–13]. In these degradation mechanisms, so-called aging, electrical behavior of transistors eventually deviates from its original intended behavior. This deviation may degrade performance; and consequently, the chip suddenly fails to meet some of the required specifications [14, 15].

In practice, NBTI is the main aging mechanism resulting in circuit malfunctions [16]. While NBTI happens continuously when the circuit is powered on, the HCI only happens provided the circuit has some activity: the more activity, the more HCI effect. In this paper, we investigate the effect of NBTI on the reliability of PUFs.

Problem Statement. Different schemes have been proposed in literature to improve the reliability of PUFs against aging effects. In particular, Error Correction Codes (ECCs) are employed in [17] to recover unwanted bit-flips (erroneous bits) in the output of PUFs. However, using ECC for error recovery is costly and its overhead is not negligible in case of multiple errors [4]. Software techniques have been proposed in [18] 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 [18].

Guajardo et al. investigated the robustness of SRAM-PUF against aging caused by a specific case of continuous writing of ones and zeros in SRAM cells [19]. Maes et al. also showed that SRAM-PUF aging can be inverted by programming them with the opposite value [20]. This reinforces the bias of the PUF, hence a rejuvenation. However, such technique does not apply to delay-PUFs.

In order to combat aging of ring oscillator PUFs (RO-PUFs), an aging resistant RO-PUF has been proposed in [4]. This ring oscillator stops oscillation when the PUF is not used. Such reconfiguration slows down the aging effects, but there is a need of custom design with the use of pass transistors. Maiti et al. conducted an accelerated aging to investigate the effect of aging on the functionality of RO-PUFs. They proposed a reconfigurable RO-PUF to mitigate aging effects [21]. As the RO-PUF consists in comparing two ROs among a set of identical ROs, the proposed anti-aging method is to choose the ROs which have the maximum frequency differences.

The RO-PUF has the specificity of having many ROs in parallel and the aging has a direct impact when two oscillators behave differently with aging. The contribution of this paper is to study the aging of simpler delay-PUF: the Loop-PUF which has a single delay chain, and the arbiter-PUF which uses two controllable delay chains.

Contributions. We notice that the reliability of the arbiter-PUF decreases with aging, whereas the reliability of the loop-PUF remains unchanged with aging. We manage to explain this discrepancy, by noting that the arbiter (seen as a hardware IP) becomes less reliable over time. The reason is the same as for SRAM-PUF aging: half of the PMOS transistors of the arbiter are conducting, while the other half are blocked. Regarding the delay chains, they age similarly: therefore, their timing difference is not impacted by aging. The loop-PUF does not use an arbiter, therefore its reliability is not affected by aging.

To support this interpretation, we use the fact that:

- Synopsys HSPICE MOSRA tool [22] can be used to simulate the aging effects in PUFs, and
- transient noise in the simulations can be used to simulate the reliability change over time of PUFs, in particular delay-PUFs.

The simulations are validated by real-world experiments on a 65 nm ASIC.

Outline. The remainder of this paper is organized as follows. Section 2 presents a preliminary background on aging mechanisms. Section 3 provides a description of the studied PUFs. The steps taken to evaluate the aging effects using Synopsys MOSRA tool are discussed in Section 4. Then, Section 5 presents the simulation results depicting the impact of aging on the reliability of PUFs. Confirmation of results on real silicon is presented in Sec. 6. Conclusions and perspectives are drawn in Section 7.

2 Aging Mechanisms

Digital circuits can be affected by various aging mechanisms including Negative Bias Temperature Instability (NBTI), Hot Carrier Injection (HCI), Time Dependent Dielectric Breakdown (TDDB), and Electro-Migration (EM) resulting in performance degradation and eventually design failure [23]. Among these aging mechanisms, the NBTI impact on PMOS transistors and the HCI impact on NMOS transistors are more prominent in the reliability of digital circuits. BTI, HCI, and TDDB aging all relate to gate oxides of transistors while EM happens in the interconnect metal lines.

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.

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 [4]. 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 [24].

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 [25].

As the PUF is used mainly to get an identifier or a response to a challenge for authentication, we can consider a rather low switching activity. Consequently we can assume less impact caused by HCI compared to NBTI. This is the reason why we mainly investigate the effect of NBTI on the reliability of PUFs, what follows discusses NBTI effects in more detail.

2.1 Background on NBTI aging

NBTI is one of the leading factors in performance degradation of digital circuits. In practice, a PMOS transistor experiences two phases of NBTI depending on its bias condition. The first phase, i.e., the stress phase, occurs when the transistor is on, i.e., when a negative voltage (i.e. $VGS < Vt$, the threshold voltage Vt being negative for a PMOS) is applied to its gate. In the stress phase, positive interface traps are generated at the Si-SiO_2 interface. As a result, the magnitude of the threshold voltage Vt of the transistor is increased. In the second phase, i.e., recovery phase, a ”positive” voltage (i.e. $VGS > Vt$) is applied to the gate of the transistor. In this phase, the threshold voltage drift induced by NBTI during the stress phase can partially “recover”.

Threshold voltage drifts of a PMOS transistor under stress depend on the physical parameters of the transistor, supply voltage, temperature, and stress

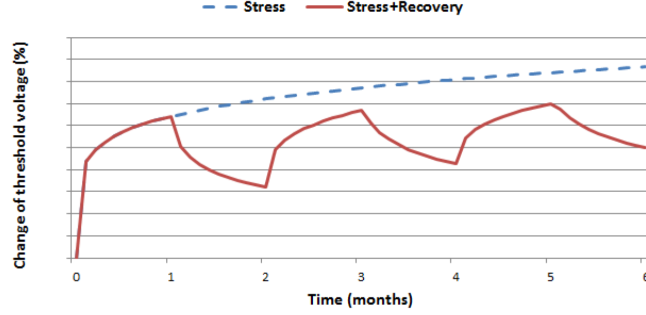


Fig. 1: Change in threshold voltage of a PMOS transistor over time.

time. Figure 1 shows the threshold voltage drift of a sample PMOS transistor that is continuously under stress for 6 months as well as a transistor that is under stress and recovery every other month. As shown, the NBTI effect is high in the first couple of months but the threshold voltage tends to saturate for long stress times. NBTI affect is exacerbated with thinner gate oxide and higher operating temperature [26, 27].

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 [28, 29]. 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 [30].

Wang et al. presented an R-D model to evaluate the NBTI effects [31]. In this model, the change in threshold voltage of a PMOS transistor in stress and recovery modes at time t are evaluated by Equation (1) and Equation (2), respectively.

$$\Delta V_{th} = \left(K_v (t - t_0)^{0.5} + \sqrt[2n]{\Delta V_{th0}} \right)^{2n}, \text{ and} \tag{1}$$

$$\Delta V_{th} = \Delta V_{th1} \left(1 - \frac{2\xi_1 t_e + \sqrt{\xi_2 C (t - t_1)}}{2t_{ox} + \sqrt{Ct}} \right), \tag{2}$$

where t_0 and t_1 denote the time at which the stress and recovery phases begin, t_e denotes the effective oxide thickness, and ξ_1 and ξ_2 are constants. Parameter n is the time exponent parameter, and for H_2 diffusion, it is 1/6. K_v and C are computed by using Equation (3) and Equation (4), where E_{ox} is the electrical field, T is the temperature, and E_a, K_1, T_0, and k are constants. As shown in

Equation (1), the magnitude of the threshold voltage of a PMOS transistor is increased during stress time.

$$K_v = \left(\frac{qt_{ox}}{\epsilon_{ox}}\right)^3 {K_1}^2 C_{ox}(V_{gs} - V_{th})\sqrt{C}\exp\left(\frac{2E_{ox}}{E_{01}}\right) \tag{3}$$

$$\text{where } C = \exp(-E_a/kT)/T_0. \tag{4}$$

In this paper, to evaluate the impact of NBTI on the performance of a circuit under stress, HSPICE MOSRA (MOS Reliability Analysis) [22] that uses an R-D model is deployed.

3 Loop-PUF and Arbiter-PUF

3.1 Loop-PUF

The *loop-PUF* structure [5] consists of a single delay chain which is looped to form a ring oscillator by means of an inverter. The delay can be obtained with high accuracy as many oscillations (N) are measured.

Fig. 2 illustrates the Loop-PUF structure composed of n delay elements and Fig. 3 illustrates the detail of one delay element in the chain of n elements. For each $i = 1, 2, \ldots, n$ element, i can have two delays (theoretically equal at blueprint level), chosen according to one challenge bit $\mathsf{c}_i \in \{0, 1\}$.

Let $d(\mathsf{c}_i)$ be the corresponding delay. As time is an extensive physical quantity, we have

$$d(\mathsf{c}_i) = \begin{cases} d_i^{T_1} + d_i^{B_2} = d_i^{TB} & \text{if } \mathsf{c}_i = 0, \\ d_i^{B_1} + d_i^{T_2} = d_i^{BT} & \text{if } \mathsf{c}_i = 1. \end{cases}$$

The delays d_i^{TB} and d_i^{BT} are modeled as i.i.d. normal random variables selected at fabrication [6]. Actually, variation at fabrication can be explained by many factors, amongst which random dopant fluctuation [32].

The n elements are chained by connecting y_i to x_{i+1}, for $i = 1, \ldots, n-1$. The principle of the loop-PUF is to measure the difference $\Delta_{\mathrm{C}}^{\mathrm{LPUF}}$ of cumulative delays $d(\mathsf{c}) = \sum_{i=1}^{n} d(\mathsf{c}_i)$ for a challenge $\mathsf{c} = (\mathsf{c}_1, \ldots, \mathsf{c}_n)$ and its complementary value $\neg\mathsf{c} = (\neg\mathsf{c}_1, \ldots, \neg\mathsf{c}_n)$:

$$\Delta_{\mathrm{C}}^{\mathrm{LPUF}} = \quad \mathrm{sign}(\lfloor N \sum_{i=1}^{n} d(\mathsf{c}_i) \rfloor - \lfloor N \sum_{i=1}^{n} d(\neg\mathsf{c}_i) \rfloor), \tag{5}$$

where N is the number of loops and the $\lfloor \cdot \rfloor$ symbol expresses the quantization of the number of loops. Thus, the LPUF computes response bits based on a mode of operation, given in Protocol 1.

The N oscillations contribute to diminish the noise impact. For simplification we can consider a single loop ($N = 1$) and a perfect quantization. Thus:

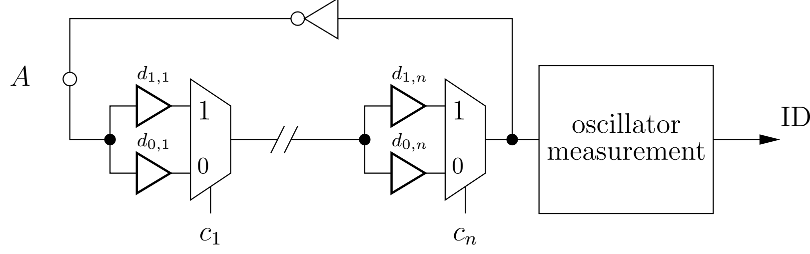


Fig. 2: Loop-PUF structure.

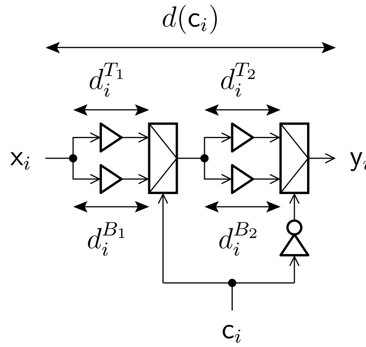


Fig. 3: Delay element i in a Loop-PUF. The output y_i is equal to the input x_i, but occurs after a delay $d(\mathsf{c}_i)$.

$$\Delta_{\mathrm{C}}^{\mathrm{LPUF}} = \sum_{i=1}^{n} d(\mathsf{c}_i) - d(\neg\mathsf{c}_i) = \sum_{i=1}^{n} (-1)^{\mathsf{c}_i}(d_i^{TB} - d_i^{BT}) \tag{6}$$

$$= \sum_{i=1}^{n} (-1)^{\mathsf{c}_i} \Delta_i, \tag{7}$$

where we have used that $\neg\mathsf{c}_i = 1 - \mathsf{c}_i$. Since d_i^{TB} and d_i^{BT} are i.i.d. normal, the random variables

$$\Delta_i = d_i^{TB} - d_i^{BT} \qquad (i = 1, 2, \ldots, n) \tag{8}$$

are themselves i.i.d. normal and have zero mean. Each Δ_i represents the delay difference from x_i to y_i in the path through first top/second bottom and first bottom/second top buffers. One bit of the identifier is the *sign* of the cumulative delay difference $\Delta(\mathsf{c})$:

$$B_{\mathsf{c}}^{\mathrm{LPUF}} = \mathrm{sign}(\Delta_{\mathsf{c}}^{\mathrm{LPUF}}). \tag{9}$$

The overall loop-PUF function is summarized in Fig. 4. A unique identification number can be obtained by querying the PUF for M different challenges c.

Protocol 1: Protocol to get one bit out of an LPUF using challenge c.

input : Challenge c
output : Response B_c

1. Set challenge c
2. Measure $d_1 \leftarrow \lfloor N \sum_{i=1}^{n} d(\mathsf{c}_i) \rfloor$
3. Set challenge $-c$
4. Measure $d_2 \leftarrow \lfloor N \sum_{i=1}^{n} d(\neg \mathsf{c}_i) \rfloor$
5. **return** $B_c = \text{sign}(d_1 - d_2)$

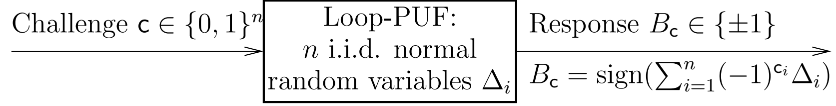


Fig. 4: Operation of a loop-PUF.

3.2 Arbiter-PUF

The arbiter-PUF (APUF) is an architecture [2, §2.2] with a pair of delay chains, so as to obtain one challenge bit per challenge, in one single query. Figure 5 represents the architecture of the Arbiter-PUF. The race of a signal along the top path and the bottom path is grabbed by the arbiter.

The PUF element thus consists in the duplication of the paths: the $\mathsf{x}_i \to \mathsf{y}_i$ path of Fig. 3 is turned into two parallel paths $(\mathsf{x}_i, \mathsf{x}'_i \to \mathsf{y}_i, \mathsf{y}'_i)$. This is depicted in Fig. 6a.

We have

$$d(\mathsf{c}) = \sum_{i=1}^{n} \mathsf{c}_i d_i^T + \neg \mathsf{c}_i d_i^B$$
$$d'(\mathsf{c}) = \sum_{i=1}^{n} \mathsf{c}_i d_i^{T'} + \neg \mathsf{c}_i d_i^{B'} \quad ,$$

and the APUF measures the fastest of the two cumulative paths. Therefore,

$$B_\mathsf{C}^{\text{APUF}} = \text{sign}(\Delta_\mathsf{C}^{\text{APUF}}), \qquad \text{where} \tag{10}$$

$$\Delta_\mathsf{C}^{\text{APUF}} = d(\mathsf{c}) - d'(\mathsf{c}) = \sum_{i=1}^{n} \mathsf{c}_i (d_i^T - d_i^{T'}) + (1 - \mathsf{c}_i)(\mathsf{c}_i d_i^B - \mathsf{c}_i d_i^{B'}). \tag{11}$$

However, contrary to the case of the loop-PUF, this equation does not simplify as in (7).

Indeed, we have that the expected value of Δ is not zero. The reason is that the delays in either input of the multiplexer are not the same. That is, let

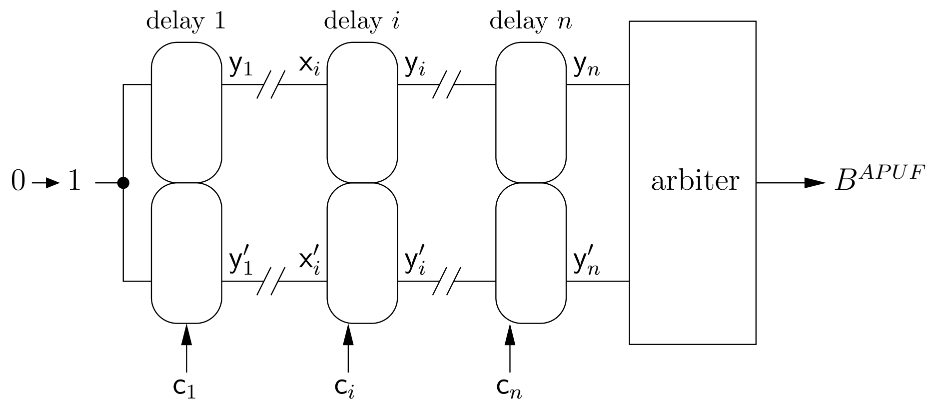
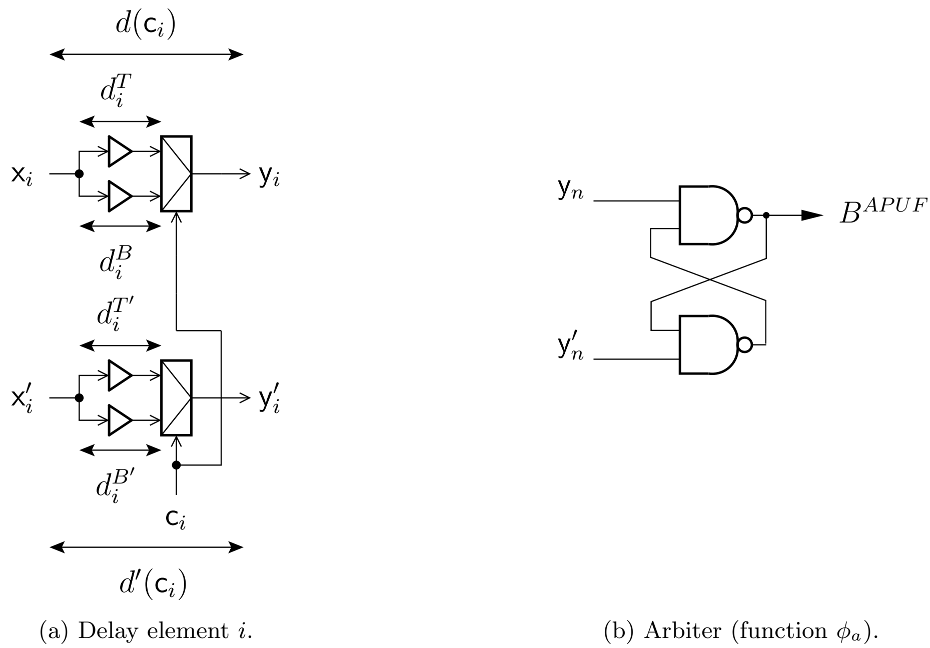


Fig. 5: Arbiter-PUF.



(a) Delay element i.

(b) Arbiter (function ϕ_a).

Fig. 6: Arbiter-PUF element examples.

us denote $\mathbb{E}(d_i^T) = \mathbb{E}(d_i^{T'}) = \mathbb{E}(d^T)$, and $\mathbb{E}(d_i^B) = \mathbb{E}(d_i^{B'}) = \mathbb{E}(d^B)$. We have $\mathbb{E}(d^T) \neq \mathbb{E}(d^B)$. Thus,

$$\mathbb{E}(\Delta_{\mathrm{C}}^{\mathrm{APUF}}) = \sum_{i=1}^{n} (-1)^{\mathsf{c}_i} (\mathbb{E}(d^B) - \mathbb{E}(d^T)) \neq 0.$$

Moreover the sign and subtraction functions of the equation (10) cannot be performed by arithmetic. They use an arbiter function ϕ_a which is able to detect the slight delay differences between two signals. The real equation of the bit generated by a APUF is:

$$B_{\mathrm{C}}^{\mathrm{APUF}} = \phi_a(\sum_{i=1}^{n} d(\mathrm{C}_i), \sum_{i=1}^{n} d(\neg \mathrm{C}_i)), \tag{12}$$

where ϕ_a is a two input function with value in $\{0, 1\}$. Typically, the function ϕ_a is a latch as illustrated in Fig. 6b, which is sensitive to aging and can be unbalanced. Given a threshold th_a close to 0, it computes:

$$\phi_a(d_1, d_2) = \begin{cases} 1 & \text{if } d_1 - d_2 \geq th_a, \\ 0 & \text{otherwise .} \end{cases}$$

4 Aging methodology with MOSRA

Figure 7 shows a flowchart of the steps involved in our aging evaluation scheme. The circuit netlist is defined at transistor level using HSPICE. The technology library is given and the input values and operating temperature are decided.

We first run a HSPICE simulation to capture the outputs of the circuit-under-evaluation (and the required delay parameters) at time zero, i.e., no aging is considered in this phase. Then, we get benefit of HSPICE MOSRA in our simulations and run another simulation (pre-stress simulation) during which we setup MOSRA to evaluate the aging effects for the given circuit running with the given set of inputs under the considered temperature. During the pre-stress simulation phase, the simulator evaluates the electrical stress of user-selected MOSFETs in the circuit, based on the MOSRA models. For example, in this phase, the aging-related change of threshold voltage of the user-selected MOSFETs are evaluated for user-defined aging time intervals and the results are reported. In practice, The calculation depends on the electrical simulation conditions of each targeted device [22].

As the next step, we launch the post-stress simulation phase during which the degradation of device characteristics that was computed in the pre-stress phase is translated to performance degradation at the circuit level.

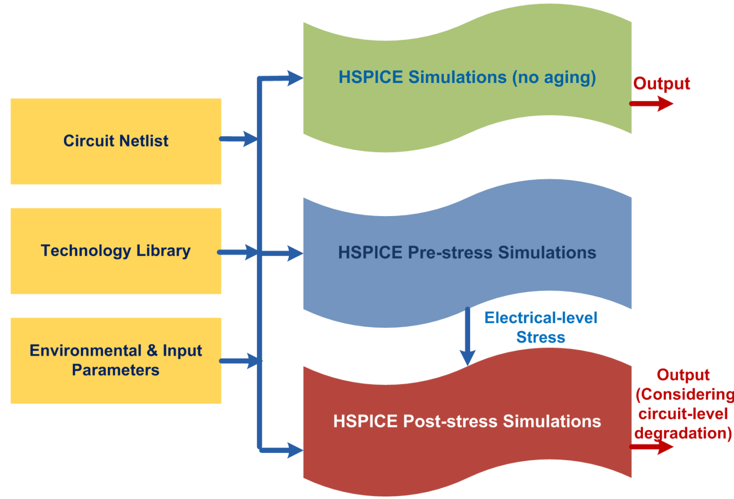


Fig. 7: Flowchart for applying HSPICE MOSRA to evaluate aging effects.

5 Impact of aging on the reliability of PUFs

5.1 Experimental Setup

In this section, we provide the details of the simulation setup used to evaluate the effect of aging on our targeted PUFs.

We first implemented our PUFs in a transistor level using a 45-nm technology extracted from the open-source NANGATE library [33]. We then used Synopsys HSPICE for the transistor-level simulations and employed the HSPICE built-in MOSRA Level 1 model to capture NBTI effects in MOSFETs [22].

We ran Monte Carlo (MC) simulations for 8192 instances of loop-PUF and arbiter-PUF each including one PUF element. We then extracted the NBTI effects to extrapolate the effect of aging on 512 loop-PUFs and arbiter-PUFs each including 16 delay elements using our in-house tool. 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}: $3\sigma = 3\%$.

Using HSPICE MOSRA, the effect of aging was evaluated for 20 months of PUF operation in time steps of one month. The operating temperature was considered as 45°C.

5.2 Experimental Results of the Loop-PUF aging

The Loop-PUF aging has been simulated by considering 512 delay chains of 16 elements taken from the 8192 instances of one element. To be able to evaluate the effect of aging in the functionality of Loop-PUFs in different cases, we cut the closed loop of the Loop -PUF at point A in Fig. 2 and injected periodic

pulses to the delay chain at point A. In order to measure the oscillation period of the loop-PUF which is now in an open loop, it is necessary to consider the delays from both rising-edge and falling edge from the SPICE simulation with the challenge bit then its complementary. The following delay is thus obtained:

$$\Delta_{\mathrm{C}}^{\mathrm{LPUF}} = \sum_{i=1}^{n} d(\mathsf{c}_i) - d(\neg \mathsf{c}_i). \tag{13}$$

As three extreme case, we considered that the duty cycle of the pulse can be either, 1%, 50% or 99%. A duty cycle of X% means that the pulse is at level '0' X% of the time. Note that when a PMOS transistor gets a pulse at level '1' in its gate input, the NBTI impact is mitigated. Thereby, different PMOS transistors in a Loop-PUF may behave differently regarding the value of X. The simulation is performed during the equivalence of 20 months of aging at 45°C and 1.2 V instead of 1.0 V. The chosen challenges are selected amongst those given the maximum PUF entropy. The study in [34] has shown that the best challenges correspond to Hadamard codes, the way to construct them is explained in [35]. For the 16-element delay-PUF, there are 32 Hadamard codewords, giving 16 pairs of complementary challenges necessary for the Loop-PUF.

The results in terms of evolution of mean and standard deviation during 20 months for 3 challenge pairs are given in Fig. 8, 9, and 10.

These results provide many pieces of information:

1. The mean is not always a monotonous function, thus there is no direct relation between the aging and the mean. This can be explained by the independence of the delay elements. Hence when a delay element has a delay increase with aging, the other one decreases. All in one, there is not a unique tendency. As a straightforward consequence, a positively (resp. negatively) biased delay element remains positively (resp. negatively) biased over time.
2. The standard deviation is always increasing with aging. This behavior is intuitively that of the standard deviation of a random walk.
3. The aging impact is stronger at the beginning of the circuit life. This is a specificity of NBTI (recall Fig. 1).
4. The duty cycle of the pulse has a small impact on aging. The standard deviation slope is slightly smaller when the duty cycle is 1% (than when it is 50% or 99%). The offset at time 0 of the standard deviation is not relevant, only the slope matters. The difference between the duty cycle value is not very significant. This means that the anti-NBTI aging strategy to force the PMOS to be most of the time "off" is not so efficient.
5. The challenges do not impact the observed behavior w.r.t. aging.

5.3 Experimental Results of the Arbiter-PUF aging

The aging on the arbiter-PUF can be studied on separate parts which are the parallel delay chains and the arbiter.

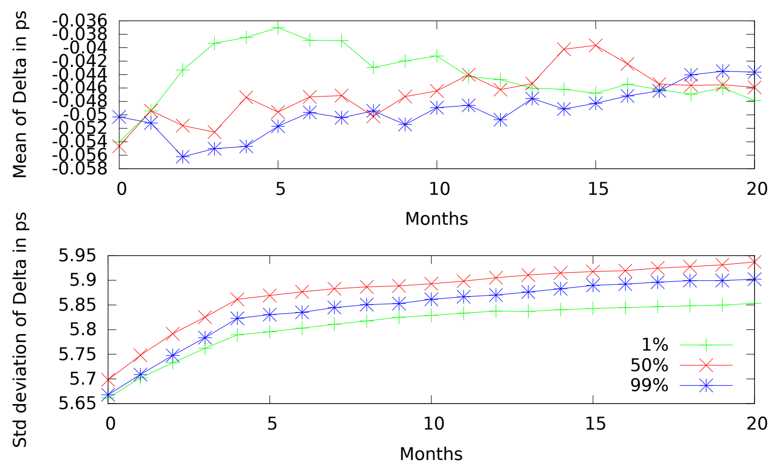


Fig. 8: Mean and variance evolution for challenge pair = 0x00FF/0xFF00.

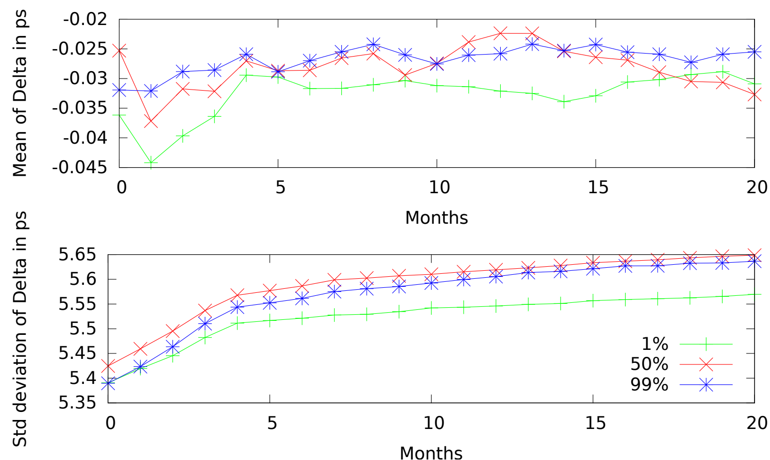


Fig. 9: Mean and variance evolution for challenge pair = 0x33CC/0xCC33.

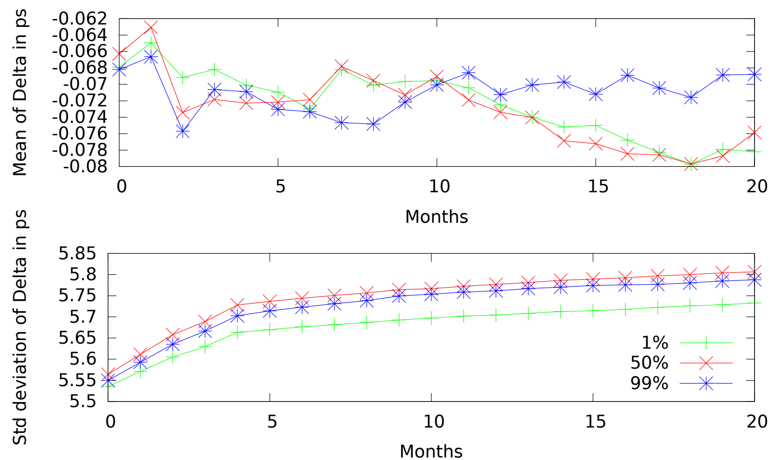


Fig. 10: Mean and variance evolution for challenge pair = 0x6996/0x9669.

Results of the delay chain part. The delay chain corresponds to Eq. (11) where the two paths are configured with complementary challenges. The delay chains have been configured with 16 elements, thus requiring 16-bit challenges. The results are very similar to the Loop-PUF. Figure 11 represents the mean and standard deviation among the 512 arbiter-PUFs for the challenge=`0x5A5A`.

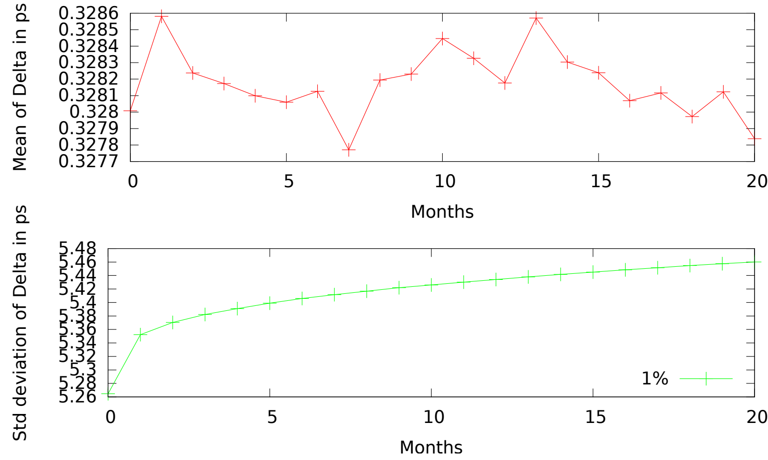


Fig. 11: Mean and variance evolution for challenge = `0x5A5A`.

The same conclusions as the LPUF can be drawn from these results:

1. The mean is not always a monotonous function, thus there is no direct relation between the aging and the mean. Thus, if one chain is faster than the other initially, it will remain so despite aging.
2. The standard deviation is always increasing with aging.
3. The aging impact is stronger at the beginning of the circuit life.

Results of the arbiter part. The test design of arbiter uses a latch composed of two NAND gates as shown in Fig. 12.

The aging impact is assessed by counting the number of bit flips in the APUF response. Fig. 13 illustrates the results obtained with 16384 APUF of one element.

It clearly shows that the number of bit flips increases with aging significantly as 1% of bit flips occurred after one year at 45°C. In order to make sure that these bit flips do not come from the delay chain, the bottom figure of Fig. 13 represents the dependence between the bit flip and the Δ value. It is expressed in probability to get a bit flip vs the sign of Δ. As it remains around 0.5, it indicates the delay chain has no noticeable impact.

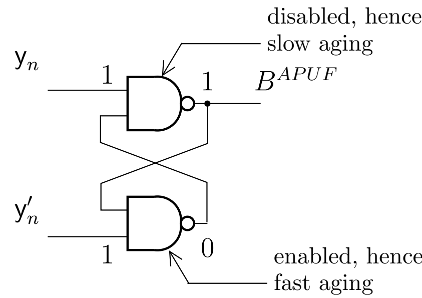


Fig. 12: Latch with two NAND gates.

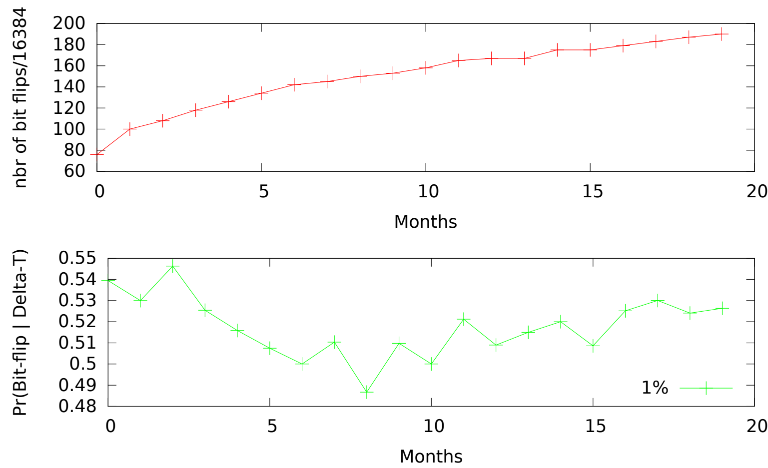


Fig. 13: Bit flips at the arbiter output.

5.4 Discussion

The fact that the arbiter reliability decreases over time can be accounted by the fact the steady state of the latch is asymmetrical. For example, if the arbiter evaluates to 0, then the logic states in the arbiter are represented in Fig. 12. One can see that one NAND gate is active, while the other one is not. This "asymmetric" state is similar to that of an SRAM memory point in the SRAM-PUF. Therefore, the reliability of the latch arbiter of Fig. 12 is decreasing over time.

Now, in a delay chain (recall Fig. 3 and 6a), each element ages independently. But, as the final measurement is a difference, the aging has no impact on the reliability.

6 Aging acceleration on real silicon

6.1 Aging acceleration setup

An ASIC with 49 LPUF has been implemented in 65 nm technology. Figure 14 shows the layout with a 7×7 LPUF matrix which makes up the largest part in the upper right-hand corner of the layout.

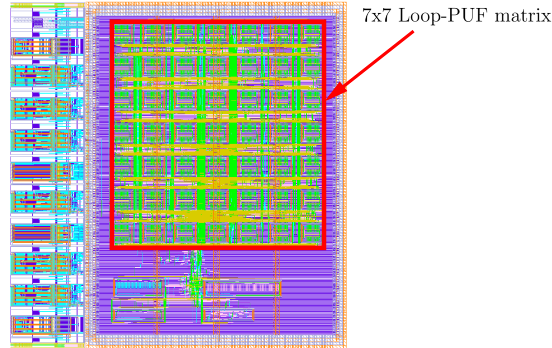


Fig. 14: Layout of the test chip embedding 49 LPUFs.

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. The test procedure is described in Protocol 2 which corresponds to cycles of 24 hours.

Protocol 2: Aging acceleration Protocol.

Input: Non aged device
Output: Aged device

1 **STEP 1: Stress during 23 hours** ....
2 $V_{dd} \leftarrow 2.0$ V, T°C $\leftarrow$ 85°C
3 Challenge $C_i \leftarrow$ `0x00000000FFFFFFFF`
4 Always measure PUF_i, for $i \in \{0, \ldots, 7\}$
5 Measure PUF_j every 1/8 time, for $j \in \{8, \ldots, 15\}$
6 Measure PUF_k every 1/64 time, for $k \in \{16, \ldots, 31\}$
7 **STEP 2: Evaluation during 1 hour** ....
8 $V_{dd} \leftarrow 1.2$ V, T°C $\leftarrow$ 20°C
9 Measurement of the 49 LPUFs with the Hadamard Challenges
10 Go to **STEP 1**

In this protocol the devices are placed in a high temperature, high voltage environment which should accelerate the NBTI and HCI effects [36, §5.3]. The

first 8 PUFs PUF_0 to PUF_7 are always measured, whereas PUF_8 to PUF_{15} are measured 1/8 of the time, and PUF_{16} to PUF_{31} are measured 1/64 of the time. PUF_{32} to PUF_{48} are never measured. This differences in switching activity (X%) allows us to test the switching activity impact on the 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.

The results in Fig. 15 represent the evolution of the mean delay $N \sum_{i=1}^{n} d(\mathsf{c}_i)$, not the differential delay, for the challenge `0x00000000FFFFFFFF`. This delay is measured when the device is back in its typical condition (**STEP 2** of the protocol).

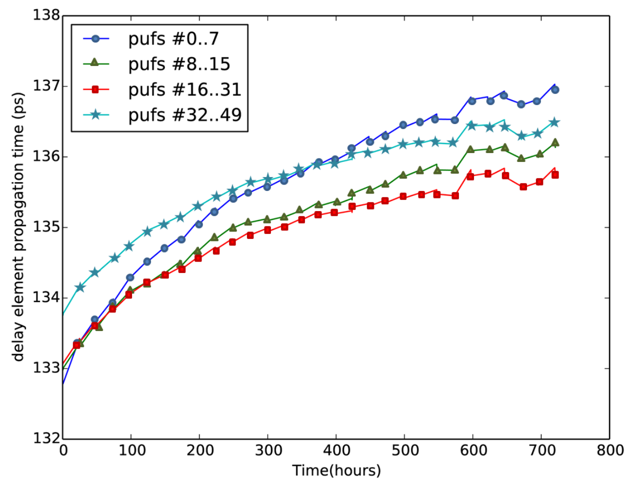


Fig. 15: Evolution of the mean delay with aging. Recall that switching rate X for $\text{PUF}_{[0,7]}$ is 100%, for $\text{PUF}_{[8,15]}$ is 12.5%, for $\text{PUF}_{[16,31]}$ is 1.6%, and for $\text{PUF}_{[32,49]}$ is 0%.

This figure brings a lot of information concerning the impact of aging:

1. The mean delay is always increasing with age.
2. Every 24 hours, we can notice a small recovery phenomenon, as expected for NBTI.
3. The slopes satisfy $PUF_{[7:0]} > PUF_{[15:8]} > PUF_{[31:16]} > PUF_{[48:32]}$. This highlights the importance of the switching activity on the aging, as also observed in simulations (Fig. 8, 9, and 10) when X increases."

Now, considering the differential delay $\text{sign}(\lfloor N \sum_{i=1}^{n} d(\mathsf{c}_i) \rfloor - \lfloor N \sum_{i=1}^{n} d(\neg \mathsf{c}_i) \rfloor)$, we obtain the results shown in Fig. 16 (*left*). These results represent the mean of the differential delay for one delay element. As the evolution is very small we can notice the strong impact of the noise.

As it was observed for the simulation of the delay chain, the evolution of the differential delay is not monotonous. Hence we can conclude that the aging has

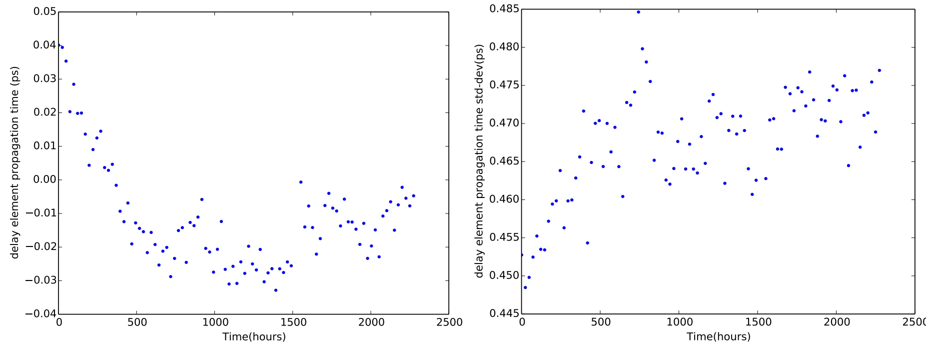


Fig. 16: Evolution of the mean (*left*) and of the standard deviation (*right*) of the differential delay with aging.

a slight and non monotonous impact on the delay chain of the Loop-PUF. The evolution of the standard deviation is illustrated in Fig. 16 (*right*). The results are very noisy as the differential delay is very small. However it is possible to observe that the standard deviation is always increasing with a greater increase during the first hours. This confirms the simulation results.

7 Conclusions and perspectives

In this paper the aging on delay-PUFs has been evaluated by simulation and aging acceleration on a real silicon. Two types of PUF taking advantage of a delay chain have been considered: the Loop-PUF and the arbiter-PUF. It has been shown that the aging has a very small impact on delay chains as each element ages independently. However the memory point as the latch of the arbiter is much more sensitive to aging, due to the asymmetry of its dual structure. Hence the aging of element is different from the aging of its dual element, and the difference is always increasing. This also highlights the interest of using simple delay-PUFs as the Loop-PUF, to avoid the imbalance of the arbiter or SRAM memory points. It has also been noticed with the experiments on a real device that the NBTI impact is dominant and that the HCI is significant only with a high switching rate.

More generally, from a user perspective, it makes sense for low-power applications to switch off completely the PUF as the aging is mainly due to having it on. This does not apply for Loop-PUF which is naturally resilient against aging. A solution to counter the aging for arbiter-PUF would be to complement its state (as the SRAM anti-aging proposed in Maes et al. [20]) or use an arbiter based on RS latch based on NOR and forces the output at '0' to mitigate the NBTI impact.

Acknowledgments

This work was supported by the KeyHAS project, the R&D program of IITP/MSIP (Study on secure key hiding technology for IoT devices).

References

1. B. Gassend, D. Clarke, M. van Dijk, and S. Devadas, "Controlled physical random functions," in *Computer Security Applications Conference*, 2002, pp. 149 – 160.
2. G. E. Suh and S. Devadas, "Physical unclonable functions for device authentication and secret key generation," in *Design Automation Conf. (DAC)*, 2007, pp. 9–14.
3. J. Guajardo, S. S. Kumar, G. J. Schrijen, and P. Tuyls, "FPGA Intrinsic PUFs and Their Use for IP Protection," in *Workshop on Cryptographic Hardware and Embedded Systems (CHES)*, 2007, pp. 63–80.
4. M. T. Rahman, D. Forte, J. Fahrny, and M. Tehranipoor, "ARO-PUF: An aging-resistant ring oscillator PUF design," in *Design, Automation Test in Europe Conference (DATE)*, 2014, pp. 1–6.
5. Z. Cherif, J. Danger, S. Guilley, and L. Bossuet, "An easy-to-design PUF based on a single oscillator: The loop PUF," in *Digital System Design (DSD)*, 2012, pp. 156–162.
6. M. J. Pelgrom, A. C. Duinmaijer, and A. P. Welbers, "Matching properties of MOS transistors," *IEEE Journal of Solid State Circuits*, vol. 24, no. 5, pp. 1433–1439, 1989.
7. 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. on Computers*, vol. 58, no. 9, pp. 1198–1210, 2009.
8. S. Morozov, A. Maiti, and P. Schaumont, "An analysis of delay based PUF implementations on FPGA," in *Reconfigurable Computing: Architectures, Tools and Applications (ARC)*, 2010, pp. 382–387.
9. H. Kufluoglu and M. A. Alam, "A generalized reaction-diffusion model with explicit h-h2 dynamics for negative-bias temperature-instability (nbti) degradation," *IEEE Trans. on Electron Devices*, vol. 54, no. 5, pp. 1101–1107, May 2007.
10. Y. Lu, L. Shang, H. Zhou, H. Zhu, F. Yang, and X. Zeng, "Statistical reliability analysis under process variation and aging effects," in *Design Automation Conference (DAC)*, july 2009, pp. 514–519.
11. S. Chakravarthi, A. Krishnan, V. Reddy, C. F. Machala, and S. Krishnan, "A comprehensive framework for predictive modeling of negative bias temperature instability," in *Reliability Physics Symp.*, 2004, pp. 273–282.
12. D. Saha, D. Varghese, and S. Mahapatra, "Role of anode hole injection and valence band hole tunneling on interface trap generation during hot carrier injection stress," *IEEE Electron Device Letters*, vol. 27, no. 7, pp. 585–587, 2006.
13. R. Rodriguez, J. Stathis, and B. Linder, "Modeling and experimental verification of the effect of gate oxide breakdown on CMOS inverters," in *IEEE Int'l Reliability Physics Symposium*, 2003, pp. 11–16.
14. O. Sinanoglu, N. Karimi, J. Rajendran, R. Karri, Y. Jin, K. Huang, and Y. Makris, "Reconciling the IC test and security dichotomy," in *European Test Symp. (ETS)*, 2013, pp. 1–6.
15. S. Khan, N. Z. Haron, S. Hamdioui, and F. Catthoor, "NBTI monitoring and design for reliability in nanoscale circuits," in *Int'l Symp. on Defect and Fault Tolerance in VLSI and Nanotechnology Systems (DFT)*, 2011, pp. 68–76.

16. J.-S. Yuan, Y. W.-K., S. Chen, and H. C.-W., "NBTI reliability on high-k metal-gate SiGe transistor and circuit performances," *Microelectronics Reliability*, vol. 51, no. 5, pp. 914–918, 2011.
17. M. Yu and S. Devadas, "Secure and robust error correction for physical unclonable functions," *Design & Test of Computers*, vol. 27, no. 1, pp. 48–65, 2010.
18. M. S. Kirkpatrick and E. Bertino, "Software techniques to combat drift in PUF-based authentication systems," in *Workshop on Secure Component and System Identification (SECSI)*, 2010, p. 9.
19. J. Guajardo, S. S. Kumar, G.-J. Schrijen, and P. Tuyls, "FPGA intrinsic PUFs and their use for IP protection," in *Cryptographic Hardware and Embedded Systems (CHES)*, 2007, pp. 63–80.
20. R. Maes and V. van der Leest, "Countering the effects of silicon aging on SRAM PUFs," in *Hardware-Oriented Security and Trust (HOST)*, 2014, pp. 148–153.
21. A. Maiti and P. Schaumont, "The impact of aging on a physical unclonable function," *IEEE Trans. on Very Large Scale Integration Systems*, vol. 22, no. 9, pp. 1854–1864, Sept 2014.
22. Synopsys, "HSPICE User Guide: Basic Simulation and Analysis," 2016.
23. K. K. Kim, "On-chip delay degradation measurement for aging compensation," *Indian Journal of Science and Technology*, vol. 8, no. 8, 2015.
24. C. Nunes, P. F. Butzen, A. I. Reis, and R. P. Ribas, "BTI, HCI and TDDB aging impact in flip-flops," *Microelectronics Reliability*, vol. 53, no. 9-11, pp. 1355–1359, 2013.
25. E. Mizan, "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, 2008.
26. M. A. Alam, H. Kufluoglu, D. Varghese, and S. Mahapatra, "A comprehensive model for PMOS NBTI degradation: Recent progress," *Microelectronics Reliability*, vol. 47, no. 6, pp. 853–862, 2007.
27. S. Mahapatra, D. Saha, D. Varghese, and P. Kumar, "On the generation and recovery of interface traps in MOSFETs subjected to NBTI, FN, and HCI stress," *IEEE Trans. on Electron Devices*, vol. 53, no. 7, pp. 1583–1592, 2006.
28. D. K. Schroder, "Negative bias temperature instability: What do we understand?" *Microelectron. Reliability*, vol. 47, no. 6, pp. 841–852, 2007.
29. S. Cha, C.-C. Chen, T. Liu, and L. S. Milor, "Extraction of threshold voltage degradation modeling due to negative bias temperature instability in circuits with I/O measurements," in *VLSI Test Symp. (VTS)*, 2014, pp. 1–6.
30. K. B. Sutaria, J. B. Velamala, A. Ramkumar, and Y. Cao, "Compact modeling of BTI for circuit reliability analysis," in *Circuit Design for Reliability*, 2015, pp. 93–119.
31. W. Wang, S. Yang, S. Bhardwaj, S. Vrudhula, F. Liu, and Y. Cao, "The Impact of NBTI Effect on Combinational Circuit: Modeling, Simulation, and Analysis," *IEEE Trans. on Very Large Scale Integration Systems*, vol. 18, no. 2, pp. 173–183, 2010.
32. Y. Ye, F. Liu, M. Chen, S. Nassif, and Y. Cao, "Statistical modeling and simulation of threshold variation under random dopant fluctuations and line-edge roughness," *IEEE Trans. VLSI Syst.*, vol. 19, no. 6, pp. 987–996, 2011.
33. "Nangate 45nm open cell library," "http://www.nangate.com" (last accessed 1 May 2016).
34. O. Rioul, P. Solé, S. Guilley, and J.-L. Danger, "On the Entropy of Physically Unclonable Functions," in *IEEE Int'l Symp. on Information Theory (ISIT)*, July 2016, Barcelona, Spain.

35. A. S. Hedayat and W. D. Wallis, “Hadamard matrices and their applications,” *Ann. Statist.*, vol. 6, no. 6, pp. 1184–1238, 11 1978. [Online]. Available: http://dx.doi.org/10.1214/aos/1176344370
36. JEDEC, “JEP122G: Failure mechanisms and models for semiconductor devices,” October 2011, http://www.jedec.org/standards-documents/docs/jep-122e.