Device aging

A reliability and security concern

Kraak, D.; Taouil, M.; Hamdioui, S.; Weckx, P.; Catthoor, F.; Chatterjee, A.; Singh, A.; Wunderlich, H.; Karimi, N.

DOI
10.1109/ETS.2018.8400702

Publication date
2018

Document Version
Final published version

Published in
2018 IEEE 23rd European Test Symposium (ETS)

Device Aging: A Reliability and Security Concern

Daniël Kraak*, Mottaqiallah Taouil*, Said Hamdioui*, Pieter Weckx†, Francky Catthoor†
Abhijit Chatterjee††, Adit Singh‡, Hans-Joachim Wunderlich§ and Naghmeh Karimi**

*Delft University of Technology
†IMEC
††Georgia Institute of Technology
‡Auburn University
§University of Stuttgart
**University of Maryland Baltimore County

***Abstract*—Device aging is an important concern in nanoscale designs. Due to aging the electrical behavior of transistors embedded in an integrated circuit deviates from original intended one. This leads to performance degradation in the underlying device, and the ultimate device failure. This effect is exacerbated in emerging technologies. To be able to tailor effective aging mitigation schemes and improve the reliability of devices realized in cutting edge technologies, there is a need to accurately study the effect of aging in high performance industrial applications. According, this paper targets a high performance SRAM memory realized in 14nm FinFET technology and depicts how aging degrades the individual components of this memory as well as the interaction between them. Aging mitigation is critical not only from device reliability point of view but also regarding device security perspectives. It is essential to assure the security of the sensitive tasks performed by the security-sensitive circuits and to guarantee the security of information stored within these devices in the presence of aging. Accordingly in this paper, we also focus on aging-related security concerns and present the cases in which aging need to considered to preserve security.**

I. Introduction

As the process technology paves its way to nano technologies, time-dependent degradation of the electrical properties in integrated circuits, so-called "*aging*", becomes more severe. In practice, in advanced technologies, electrical behaviors of transistors embedded in a chip deviate from original intended behaviors during the chip lifetime. This deviation degrades the chip performance, and consequently, the chip fails to meet some of the required specifications [1], [2].

Aging mechanisms include Bias Temperature Instability (BTI), Hot-Carrier Injection (HCI), Time-Dependent Dielectric Breakdown (TDDB), and Electromigration (EM) [3]–[5]. Among these aging mechanisms, the first three deal with the gate oxides of transistors while EM occurs in the interconnect metal lines. The effect of aging can range from increased transistor switching delay to permanent faults that cause a transistor or interconnect wire to fail entirely. Environmental and electrical stress can exacerbate aging rate and result in premature vulnerabilities. In practice, performance degradation of an integrated circuit due to aging is influenced by the operating conditions of the circuit including temperature, voltage bias, and current density [1].

BTI includes NBTI (Negative-Bias Temperature Instability) and PBTI (Positive-Bias Temperature Instability) 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. However, PBTI emerged as a reliability concern after the introduction of high-k gate oxides and metal gates transistors [6].

A PMOS transistor experiences two phases of NBTI aging depending on its operating condition. In the "stress" phase, which occurs when the transistor is on, positive interface traps are generated at the Si-SiO_2 interface. This leads to an increase of the threshold-voltage of the transistor over time. The second phase, so-called "recovery", occurs when the transistor is off. In this phase, the threshold-voltage drift occurred during the stress phase will partially recover. Similarly, an NMOS transistor is under PBTI stress when the transistor is on. Otherwise, it is in the recovery phase [7].

To mitigate NBTI-related performance degradation and to increase the reliability of circuits, several methods have been proposed in literature. Guard-banding, gate-sizing, and voltage tuning are among the methods used in industry to reduce aging effects. However, these methods are either insufficient or otherwise over-pessimistic as the rate of aging degradation depends on operating conditions including temperature, voltage bias, and workload [8].

As transistor aging intensifies in emerging technologies, considering their effect from both reliability and security perspectives, and leveraging efficient aging prognosis and mitigation schemes that preserves chips reliable and secure is crucial. Being able to perform aging prognosis and prevent precaution actions before a circuit experiences malfunction is highly beneficial for both reliability and security communities. In practice, with moving to smaller feature size, the need for leveraging aging-aware design schemes and identifying aging-related speed-limiting paths during the design process is exacerbated. Accordingly, in this paper we concentrate on a number of aging-related reliability and security concerns in digital circuits and discuss how these concerns can be addressed.

978-1-5386-3728-9/18/$31.00 ©2018 IEEE

The rest of this paper is organized as follows. Section II deals with the aging-related degradation of an industrial FinFET memory, and analyzes the relative degradation of each memory component as well as the interaction between the components and their impact on the whole memory. Section III focuses on SRAM-based digital fingerprint generators and true random number generators (TRNGs), and presents a methodology to mitigate aging degradation in these circuits. Section IV discusses the need of secure aging monitoring schemes. Section V focusses on the effect of aging from security perspectives and presents the cases where aging positively/negatively affects security. Finally, Section VI concludes the paper.

II. DEGRADATION ANALYSIS OF HIGH PERFORMANCE INDUSTRIAL FINFET SRAMS

It is well known that the smaller the technology nodes, the more severe are the reliability challenges; higher failure rate, reduced lifetime/ accelerated aging [9], [10]. Hence, accurately understanding the impact of degradation on cutting edge technologies in order to effectively mitigate for such impact while optimizing the designs is of great importance [2]. In the rest of this section, the degradation analysis of high performance 14nm FinFET based on a "realistic industrial strength" circuit design will be presented. The analysis uses calibrated aging models [11], and focuses not only on the individual degradation of each memory component but also on the interaction between the components and their impact on the whole memory.

A. Methodology and Experimental performed

A complete memory model is used in order to realize a high accuracy; the core of the model consists of a 1KB 14nm FinFET array constructed from 6T SRAM cells with a word length of 32 bit; it is able to run at 2 GHz under worst-case conditions. The memory model also has input and output buffers, decoders, sense amplifiers, write drivers, precharge circuits and a timing control circuit [12]. To simulate the memory netlist, the PTM 14nm FinFET LSTP library [13] is used. We calibrated the PTM library with commercial 14nm libraries to match the power and delay.

To analyze the impact of memory aging, BTI is selected as it is one of the dominating reliability failure mechanisms [14]. The analysis methodology is based on performing 1000 Monte Carlo simulations (using Spectre) of the updated netlist of the memory; this incorporates both process variations and BTI. During each Monte Carlo iteration the targeted metrics are measured in order to identify the mean and the spread. It is worth noting that for modeling BTI, the atomistic model presented in [11] is used; it takes into account the workload dependency, which is modeled by the duty factors and frequencies of the signals applied to the gates of the transistors. Two major experiments were performed:

- *Individual component degradation*: here, the relative degradation of each component is identified. The following components are considered: Memory cell, Sense Amplifier (SA), Address Decoder, and Timing Circuit. For the memory cell, the delay to discharge the bitline by 10% w.r.t. the precharge voltage is used. For the SA, its sensing delay is used. For the address decoder, the delay to activate the wordline is used. It is measured as the delay between the enabling of the decoder by the timing circuit and the wordline being high. For the timing circuit, the delay between the rising edge of the clock and the sense amplifier activation is used.
- *Combined aging effect*: here, the interaction between different components is explored and the impact on the overall memory aging is measured. The used metric is the memory's read access time. It is measured as the time between the rising edge of the clock and the data appearing at the memory's output.

In order to mimic the dependency of the degradation on the application, four different workload are used; they are described with the following March algorithm notations:

- Low activity Balanced (LB): ⇑(w0, r0, i8, w1, r1, i8)
- Low activity Unbalanced (LU): ⇑(w0, r0, i8)
- High activity Balanced (HB): ⇑(w0, r0, w0, r0, i, w1, r1, w1, r1, i)
- High activity Unbalanced (HU): ⇑(w0, r0, w0, r0, i)

Workloads LB and LU (HB and HU) assume a workload that consists of 20% (80%) memory instructions and 80% (20%) idle (i) time. For example, for LB, 4 cycles are consumed by four memory instructions (2 writes and 2 reads) and 16 by idle cycles. Note that LB and HB both have balanced workloads for the memory cells and read/write circuitry, while LU and HU have unbalanced workloads. All the workloads iterate over all 256 memory addresses. Hence, each address is selected/stressed an equal amount of time.

B. Simulation results

Individual component degradation: Figure 1 shows the relative degradation of the 6σ corner of the individual component delay after three years of aging at 85°C and nominal supply voltage for several workloads. The figure shows that the address decoder has the highest degradation; its delay increases with up to 9.2%. The memory cell shows the second highest degradation; the delay to discharge the bitline by 10% w.r.t. the precharge voltage increases with up to 6.42%. After the memory cell, the SA shows the highest degradation, closely followed by the timing circuit. Their delays increase with up 5.7% and 5.6%, respectively.

When examining the dependency on the workload, we observe that the degradation of all components strongly depends on the workload; workloads with a higher activity (i.e., HB and HU) give a higher degradation for all components. For example, high activity workload HB gives a degradation of ≈9.2% for the address decoder, while low activity workload LB gives a degradation of ≈6.9%. In addition, we observe that the memory cell and SA are also dependent on the balancing of the read/write values of the workload; unbalanced workloads (LU and HU) result typically in a higher degradation. For

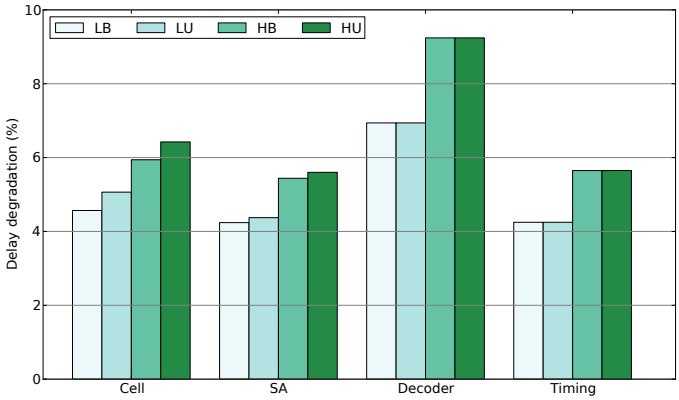

Fig. 1. The relative delay degradation of individual components for three years of aging at 85°C and nominal VDD.

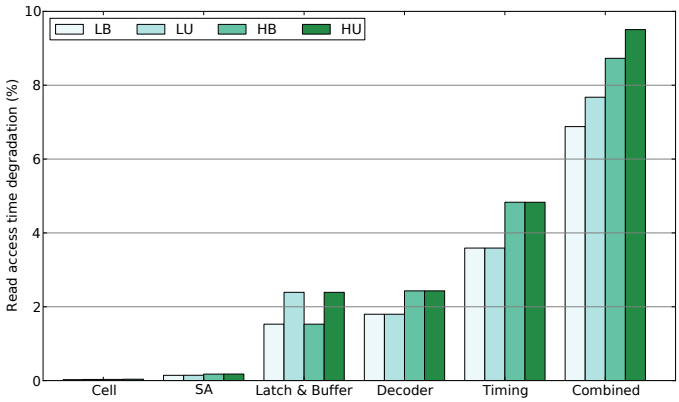

Fig. 2. Memory read access time degradation for three years of aging at 85°C and nominal VDD.

instance, balanced workload LB gives a degradation of ≈4.6% for the memory cell, while unbalanced workload LU gives a degradation of ≈5.1%. Finally, it is worth noting that for all components the activity of the workload has a higher impact than the balancing of the workload. For example, unbalanced, low activity workload LU gives a degradation of ≈5.1% for the memory cell, while balanced, high activity workload HB gives a degradation of ≈5.9

Combined aging effect: Figure 2 shows the degradation of the 6σ corner of the memory read access time after three years aging at 85°C and nominal supply voltage. The graph shows the individual component impact (i.e., assuming only one component is aged) as well as the combined/ overall impact on the read accessed time which considers not only the fact that all components suffer from aging, but also the interaction between the components.

The figure reveals that the cell and sense amplifier have a low impact on the read access degradation, while the output latch and buffer, address decoder, and, especially, the timing circuit have a high impact. The impact of the SA is low, as its delay is only a fraction of the total access time. As a result, the impact of the cell is also low, as it only impacts the SA. The latch and output buffer have a relatively high impact, as these have a relatively long path in this small memory. Hence, the increased cumulative delays along this path result in a high impact. Aging in the address decoder delays the activation of the wordline. Activating the wordline takes a significant portion of the total access time due to the high parasitic capacitance of the wordlines. Hence, degradation of the address decoder has also a high impact. The timing circuit contains the longest paths of the circuit. Hence, it also has the highest absolute degradation, due to the increased cumulative delays along its paths.

Examining the workload dependency shows that workloads with a higher activity result in a higher degradation of the access time. For example, low activity workload LU gives a degradation of ≈7.7% (combined case), while high activity workload HU gives a degradation of ≈9.5%. This happens due to the fact that high activity workloads stress the address decoder and timing circuit more often. In addition, unbalanced workloads result in a higher degradation of the access time. However, the balancing has a lower impact than the activity of the workload, as the degradation of the access time is dominated by the address decoder and timing circuit, which are only impacted by the activity of the workload.

Comparing the results of the individual component degradation (Figure 1) and the combined aging effect (Figure 2) reveals that it is misleading to only investigate the individual degradation of components. Even though some components may have a high degradation, it does not necessarily mean that they have a high impact on the access time. For instance, the individual delay degradation of the cell is the second highest with a degradation of up to 6.4% (Figure 1). However, we observe from Figure 2 that it has a negligible impact on the overall read access time. Moreover, the timing circuit shows the lowest individual delay degradation of all components. However, it has the highest impact on degradation of the memory's access time.

C. Discussion

The reliability of embedded memories is extremely important for the overall system reliability. Based on this work the following observations w.r.t. FinFET SRAM reliability:

Individual component vs combined aging: it is misleading to only investigate the individual degradation of components. For instance, we observed that the timing circuit's individual degradation was the lowest of all components, while it has the highest contribution to the degradation of the overall access time. This shows that it is extremely important to consider the impact of aging on the whole system, rather than analyzing components separately.

Component sensitivity: our case study (1KB, high performance FinFET SRAM) reveals that the degradation of the timing circuit, address decoder, and output latch and buffer have the highest impact on the overall memory access time. For bigger memories, it is expected that the timing circuit becomes the dominant factor for the access time degradation, as the time needed to activate the wordline by the address

decoder and the propagation delays of the output latch and buffer will be a smaller fraction of the whole operation period.

III. Ensuring the Reliability of Digital Fingerprints and Random Numbers Extracted from CMOS SRAMs as the Device Ages

Unique device identifiers and random number generators are key primitives that have been widely used in proposals to ensure security in integrated circuits and systems deployed on the internet of things (IOT), and in numerous other applications [15]. An identifier, or digital fingerprint, is a unique bit pattern associated with each instance of a manufactured integrated circuit. If generated by on-chip circuitry in such a manner that it remains stable in the face of circuit and environmental noise, and device aging over time, the digital fingerprint can be used to reliably differentiate among different instances of integrated circuits and SOCs, even if they are identically manufactured. A true random number generator (TRNG), on the other hand, generates bit patterns that are unstable, random and different each time the generating circuit is activated; furthermore, the bit pattern must satisfy statistical properties reflecting true randomness. Random numbers are essential for security applications such as key generation for data encryption in secure communication. Together with digital fingerprints, TRNGs comprise key primitives needed to ensure reliable identification and secure communication on the internet of things.

Exploiting the power-up state of an SRAM for obtaining both a unique digital fingerprint for the device, as well as true random numbers, has long been proposed as part of low cost security solutions for IOT nodes, especially if the same SRAM also doubles as functional memory. When a conventional SRAM is powered up, individual cells acquire either a 0 or 1 logic state, depending on the inherent bias within the 6-transistor cell circuitry. This bias is in large part determined by the threshold voltage mismatches within the PMOS and NMOS transistor pairs that implement the storage latch within each cell. Although the layout for all SRAM cells is identical, this mismatch is the result of random process variations that cause the threshold voltage of each individual transistor to deviate somewhat from the targeted nominal value.

A lower threshold voltage in the NMOS transistor connected to the output bit line, compared to that for the paired NMOS connected to the bit complement line, tends to discharge the output capacitance faster. This will cause the cell output to acquire a logic 0 at power up, unless the bias is overcome by a stronger differential bias in the opposite direction in the two PMOS pull-up transistors in the cell. If both the NMOS and PMOS transistor pairs bias the cell towards the same logic value, and the bias is strong, then the cell is a ”strong” cell, which will reliably power up to the same logic value every time. If on the other hand, the threshold voltage biases inside a cell are small, or the NMOS and PMOS transistors generate conflicting biases, the cell is a ”weak” cell. The state that such a cell acquires can be unstable and dependent on random circuit and ambient noise. The key idea behind SRAM based security primitives is to exploit weak cells for random number generation, and strong cells for generating a digital fingerprint of the device. The first challenge in practice is to reliably identify the strongest and weakest cell in each instance of a manufactured SRAM.

While using repeated power up cycles have been considered as a means of identifying stable and unstable cells [15], classification using this simple functional approach has not proven sufficiently robust in the face of noise and device degradation due to aging for reliable use. Recently, two new approaches for classifying the strength of SRAM cells have been proposed [16], [17]. The first approach exploits cell remanence to identify strong cells for Physically Unclonable Function (PUF) applications; it does not attempt to identify weak cells. In [17] power up experiments with varying power supply ramp rates are utilized to estimate the strength of the threshold voltage bias in individual pairs of PMOS and the NMOS transistors in each cell. The approach has been shown to be highly effective in calibrating the individual strength of each cell in the SRAM and thereby facilitate an ordering of the SRAM cells by cell strength. This allows the strongest cells in the SRAM to be used to create a unique and robust circuit identifier. Furthermore, it is possible to trade-off the number of the strongest cells from the SRAM chosen to form this digital fingerprint, and the robustness of those cells to noise. If only 2-5% of the cells are used, the digital fingerprint can be made very reliable [17].

Note that a fingerprint constructed from just 50-100 SRAM cells can uniquely identify billions of individual parts, even allowing for some level of random aliasing. Thus the digital fingerprint can be made extremely robust by using a very small fraction of the strongest cells in a large SRAM. Similarly, selecting a handful of the weakest, least stable, cells can provide a true random number generator.

However, while the digital fingerprint and TRNG can be made quite robust to noise at the time of initial test and deployment using the cell calibration approach described above, changes in threshold voltages due to aging remain a challenge. This is particularly a problem for the TRNG because the cells are required to have a near zero bias to ensure true randomness. Even a relatively small threshold bias developing in the cell due to BTI stress and aging can affect the randomness of the generated patterns. The reduction of BTI stress during periods of inactivity can also cause threshold voltage shifts from stress recovery. To address this serious problem, we propose a novel methodology that uses controlled BTI stress to reduce any bias that exists or develops in each TRNG SRAM cell over time.

Consider the SRAM cell that acquires state 0 at power up as shown in Figure 3. This cell must have an inherent bias such that the left NMOS transistor in the NMOS transistor pair has a smaller threshold voltage and/or the right PMOS has a smaller (in magnitude) threshold voltage, such that these transistors are the first to turn ON at power up. Notice that after power up these very same transistors, the left NMOS and the right PMOS are ON and therefore under BTI stress. This tends to increase the magnitude of their threshold voltages,

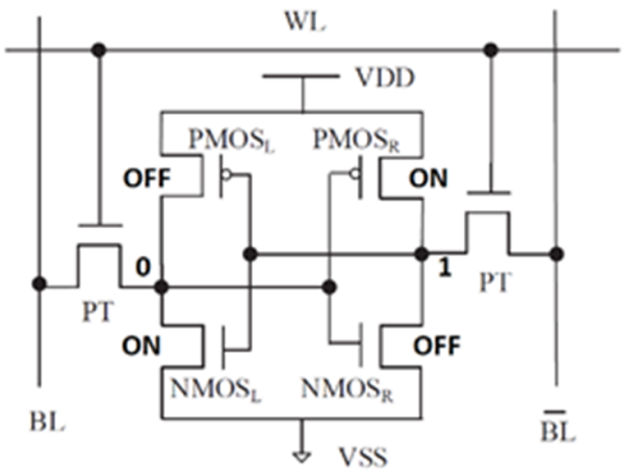

Fig. 3. A six-transistor SRAM Cell.

thereby reducing the bias that they create in the cell over time because the complementary transistors in each pair are OFF and therefore do not degrade with time. Thus, BTI stress has the effect of reducing the inherent bias in a cell if the cell is held in the initial power up state over time. This suggests that for reliable operation, the TRNGs that are constructed using known weak cells in the SRAM should be periodically powered up, and kept in the power up state for a calculated window of time to mitigate any built up bias that develops over time. This period should be small enough to allow only minimum shifts in the threshold voltage, well within the threshold voltage mismatch range acceptable for TRNG operation. This is to ensure that the controlled BTI stress cycle does not itself leave an unacceptable bias in any cell. The power up and hold operation can then be repeated as frequently as needed to ensure proper TRNG operation.

Notice that any SRAM cell, utilized in the TRNG, that for any reason develops an unwanted bias will be stressed back towards neutrality during each controlled BTI stress cycle, with small threshold voltage changes at a time. In case the threshold shift caused by the BTI stress overshoots neutrality and creates an opposite bias in the cell, then during the next controlled stress cycle, the cell will power up in the complementary state creating BTI stress in the complementary set of transistors, thereby again driving the cell bias towards neutrality. The proposed methodology is thus fully self-correcting. It is expected in practice that with very small cell biases in a properly operating TRNG, the power up states will be random. Consequently, on average, the short periodic controlled stress windows will have no net impact on the cell bias over time. The periodic BTI stress that is introduced ensures that any unwanted bias from threshold voltage mismatch that develops in the TRNG SRAM cells due to any reason whatsoever is quickly neutralized.

Executing a sufficient number of such power up and hold stress cycles before deployment can also ensure that any small initial biases in the TRNG cells caused by errors in calibrating SRAM cell strength are also mitigated. Recall that the BTI stress time window must be small enough that it causes threshold voltage shifts that are well within the cell bias tolerance levels for proper TRNG operation. Clearly, if the methodology used to select SRAM cells for use in the TRNG can accurately pick cells with only minimal bias, relatively few such stress cycles will be sufficient to achieve acceptable cell neutrality. On the other hand, if the selected cells display large biases, it may be difficult or even impossible to achieve cell neutrality using BTI stress. Before deployment, controlled BTI stress cycles should be repeated and the TRNG output data collected from every power up cycle, until the data shows acceptable statistical randomness indicating unbiased SRAM cells in the TRNG.

BTI stress can also be used to increase the robustness of the strong SRAM cells used in the identifier bits that form the unique digital fingerprint. (Such cells can also be used as SRAM PUFs.) In this case, since we need to work with strong cells, the cell bias needs to be enhanced rather than neutralized. This requires applying controlled BTI stress that holds the SRAM cells at logic values opposite to what they acquire at initial power up, i.e. complementary bit values compared to those that form the digital fingerprint. The duration of this stress can be much longer, and will depend on the statistics of the data stored in the memory during normal operation, because here stress due to functional operation needs to be mitigated. The aim is to counteract any reduction in the cell biases in the SRAM bits selected for generating the digital fingerprint, thereby ensuring continued reliability and robustness.

In summary, we have shown how controlled BTI stress can be effectively used to mitigate aging degradation in SRAM based true random number generators, PUFs and digital fingerprint generators.

IV. Secure Aging Monitoring

Aging is one of the major reliability threats especially in safety critical applications for instance in the automotive domain. In the year 2014, the average maximum lifetime of a passenger car in Germany was 18 years [18], and the average age of the passenger cars on Germany's roads was 9.3 years in January 2017 [18]. This time is by far longer than the survival times published for the most relevant wear-out mechanisms of semiconductors [19], which strongly depend on the usage pattern and the environmental conditions like temperature and mechanical stress. These environmental conditions are especially harsh in the automotive domain, and at the same time, functional safety requirements are especially high. As mentioned earlier, the main aging mechanisms for semiconductor circuits include BTI, HCI, TDDB and electro migration.

Aging may affect system security under various aspects, authorization and protection schemes may degrade, or confidential information may be leaked due to failures. Moreover,

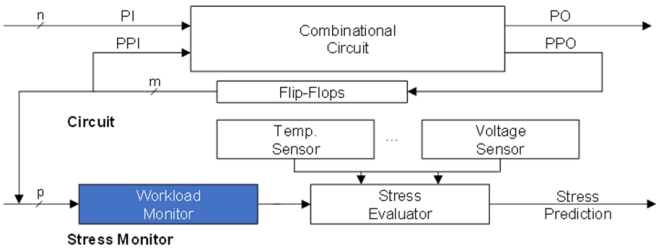

Fig. 4. Stress evaluation by workload monitors.

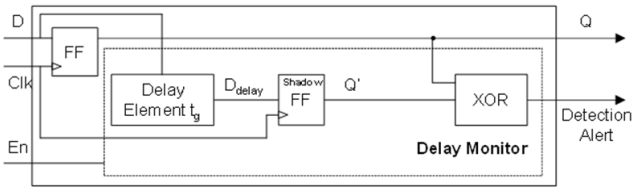

Fig. 5. Structure of a delay detecting flip-flop [27], [32].

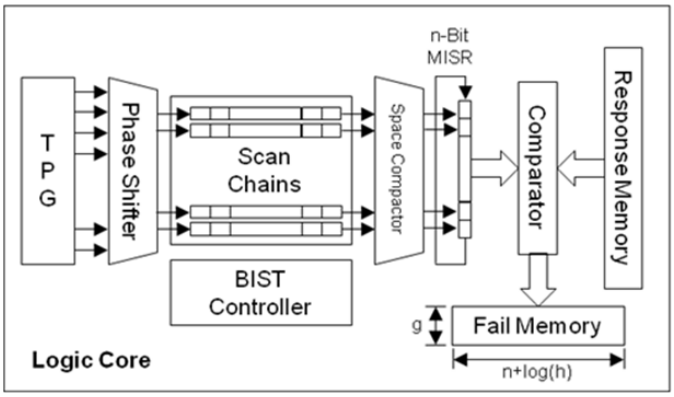

Fig. 6. Scheme for online built-in self-test and self-diagnosis.

aging detection, protection and prevention schemes may introduce additional security threats, and this section is concerned with the connection between reliability infrastructure and security. Mainly three classes of aging detection and prevention schemes are usually applied: workload monitoring, circuit monitoring and periodic testing.

Workload Monitoring: Workload monitors may be implemented by designing and implementing replica circuitry to track local critical path delays [20], which have to be selected in a representative way (RCRP: Representative Critical Reliability Paths [21]). A similarly effective but less costly way to evaluate the workload is found in [22], where a combinational circuit is synthesized which outputs "1" for each critical assignment to the off-path inputs of a reliability path. The assignments are counted and combined with other observables from internal instruments [23], [24] as seen in Figure 4.

Workload monitors have the advantage of minimal interference with the circuit operational speed [25], [26], however, they do not consider the specific properties of an individual circuit which may differ significantly due to variations. Hence, false positives may occur as well as false negatives, if unexpectedly some components become critical in a circuit. Unwanted access to workload monitors will not only exhibit comprehensive circuit information but will also leak information about both processed data and applications executed.

Circuit Monitoring: While workload monitoring is proactive, circuit monitoring and online testing react on changes which already happened due to stress and aging. Performance degradation and small delay faults are indicators of these changes. Delay detecting flip-flops belong to the class of circuit monitors and can detect a violation of pre-defined guard bands by sensing the transitions [27]–[34]. Often, they are placed at the end of critical paths or at intermediate positions of combinational circuits. Figure 5 shows one possible structure of those monitors.

The usual placement of a delay detecting flip-flop at all the pseudo-primary outputs of a combinational circuit may be expensive but may still overlook some critical faults. If a certain path has rather a large slack, degradation may not be detected before a catastrophic fault will occur, e.g. due to HCI. Both problems can be reduced, if the monitors are placed inside the combinational circuitry and cover the initial segments of most of the paths. The monitors are active at a reduced period (e.g. by using the inverted clock signal) and the slack can be controlled [35], [36]. Other circuit monitors can be characterized with respect to the degradation indicator they measure: working current hsieh-07-low,wang-11-multi, threshold voltage [37], frequency [38], or slew rate [39]. Their basic technique can be either the reference cell method or the prognostic cell method. The reference cell method places a replica close to the critical component [20], [21], [37], which is usually switched on, and the difference between the critical, aged component and the replicated, not-aged, one is measured.

The prognostic method implements cells which are permanently stressed, for example a ring oscillator. Its degradation gives a worst-case indication for the entire circuit [37], [38]. Due to large workload differences between applications [25], self-stressed monitors may be pessimistic and underestimate the real lifetime. However, they provide less information about both circuit internals, processed data and applications than the reference cell method does which has to be securely protected.

Concurrent Testing Periodic online testing is a complementary technique to circuit monitoring since the latter may be not effective, if a certain module has not been activated for some time, typical examples are components for accident prevention and protection. Periodic logic built-in self-test (BIST) can be based either on random patterns like the well-known STUMPS scheme, on deterministic test patterns or on both [40] (Figure 6).

Data collection: Both the response memory and the fail memory contain sensitive information which on one hand has to be protected against attacks, but on the other hand, it must be visible at system level to allow aging and fault handling. The fault and state information sampled on chip by

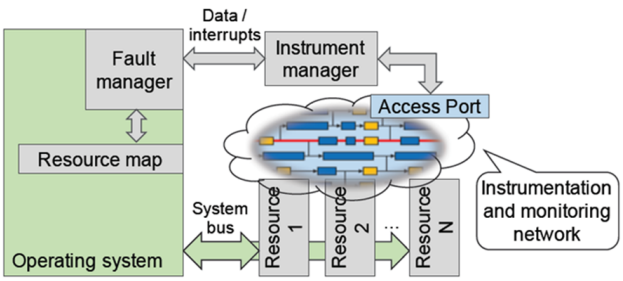

Fig. 7. Fault management architecture according to [42].

any of the above mentioned techniques must be aggregated and evaluated in the system. This requires both access mechanisms to the test infrastructure and a central or distributed resource management. Resource management can be implemented on a processor by firmware or as a hardware unit [41].

The increasing number and diversity of reliability infrastructure lead to the development of more flexible and scalable access mechanisms based on reconfigurable scan networks (RSNs) and their recent standardization in IEEE Std 1149.1-2013 and IEEE Std 1687-2014. In RSNs, the path through which data is shifted can be reconfigured, for instance for minimum access latency to a set of targeted instruments.

These scan-based access mechanisms have been used to construct comprehensive access architectures for system and reliability management [42]–[44]. Such architectures support self-aware systems by well-defined (instead of ad-hoc) interfaces, access procedures, and shared resources for instrument management such as for calibration, start and control of measurements, local response storage, or event-based signaling. Figure 7 shows the architecture proposed in [42] to gather data from instruments spread over the resources in the system. The instrument manager decouples the details of the RSN-based communication to instruments in the hardware resources from the fault manager, which maintains the state of the resources as part of the operating system. The flexible, self-reconfiguring scan network described in [44] provides low-latency error signaling for concurrent checkers and monitors and also an efficient error localization.

Secure reliability infrastructure access must be provided to prevent leakage or manipulation of sensitive instrument data or side-channel based attacks. This level of security is especially important in safety-critical systems, where an attack may cause unsafe system behavior. This requires a design methodology beyond ad-hoc solutions that incorporates access privileges and protection and secure data transmission at infrastructure level [45]–[47]. An attacker may exploit the scan infrastructure as a side-channel to gain access to protected data (secret key or IP), or to alter the system state to perform illegal or unsafe operations [48], [49].

Defenses against attacks that exploit the external JTAG interface (test access port, TAP) include access authorization [50]–[52], scan data encryption [46], and scan chain obfuscation [50], [53]. The goal of these approaches is to assure that only users who know a shared secret (e.g. encryption key, challenge-response pair, or obfuscation principle) can access the scan infrastructure. In RSNs, the scan security problem is further exacerbated due to the distributed control over the access to scan segments [53], [54]. Recently proposed secure RSN architectures control the access to sensitive infrastructure by extending the RSN [47], [55] or the TAP, employing obfuscation, challenge-response authentication, or a filter that restricts the allowed scan accesses. These approaches provide access control and data protection at TAP level, but do not sufficiently protect against attacks from within the chip, such as sniffing or spoofing of shifted data by components in the scan network. Of course, permanently disabling the infrastructure access [48] is not an option if the infrastructure is use for online monitoring. Dedicated encryption for the communication to each attached data register, on the other hand, incurs high costs in area and power and is thus impractical.

In [21], a method is presented which implements access port protection by utilizing a filter at the RSN interface. The filter monitors the scanned in pattern as well as the control signals of the RSN. The structure of the RSN is modeled as a Finite State Machine (FSM), where each state represents a different test register or segment. Using this FSM the filter is able to identify which segment each scanned-in bit will reach after a complete access. Based on the user authentication, the FSM can either terminate and lock or grant access to the active scan path and prevent violations of security properties online.

V. Aging-induced Security Concerns

Device aging can jeopardize the reliability of integrated circuits over time. Thereby, there is a need to monitor and analyze the aging effects at real time to be able to project aging degradation in a circuit in a foreseeable future [56]. In practice, aging prognosis allows to proactively estimate the effect of degradation before it actually occurs, such that preventive actions can be put in place to avoid catastrophic consequences.

As aging-induced degradation depend on different factors such as running workload, operating temperature, voltage source, and physical specification of transistors, an aging prognosis method that considers these factors into account can be highly beneficial in leveraging the reliability of circuits and preventing system wearout before it occurs. Figure 8 depicts the abstract flowchart of our aging prognosis approach that is based on training of non-linear regression models to map various circuit operating conditions to delays of critical and near-critical paths (with delays higher than 80% of the critical-path delay) [57]. To train the model, a comprehensive set of IC operating conditions including workload, usage time, and run-time temperature are used. Then, a calibration technique is leveraged to compensate the effect of process variations on our path delay prediction. Applying the proposed algorithm on ISCAS85 benchmarks shows that that the impact of IC aging on critical path delays can be accurately predicted using our nonlinear regression models (mean of prediction error$\approx 3\%$). Reference [57] discusses the training model in more details.

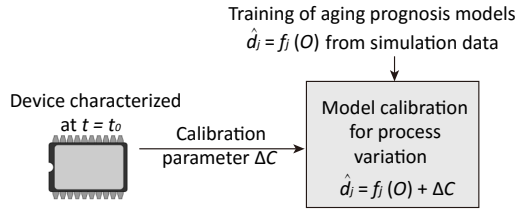

Fig. 8. Flowchart of the proposed aging prognosis approach

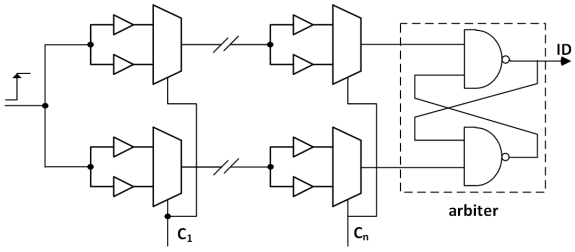

Fig. 9. Arbiter-PUF.

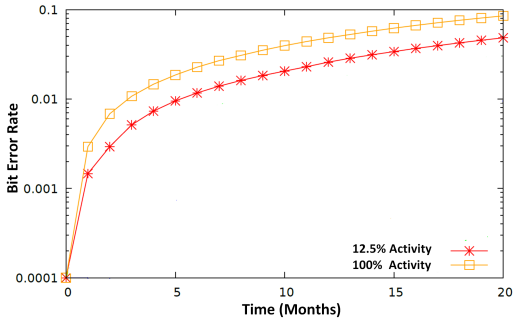

Fig. 10. The effect of aging in the arbiter of the arbiter-PUF for different activity rates.

Although, utilizing aging prognosis schemes is highly beneficial for reliability enhancement, such methods may not be highly useful for the applications with high level of security requirements. As mentioned, for aging prognosis, models are trained based on the typical workload and operating conditions of a device. However, for secure devices such as cryptographic devices the typical workload (data and keys) is not known beforehand. This results in lower precisions when aging prognosis methods are utilized.

In fact, aging-related reliability degradation is also important from security perspectives. One such case is the reliability of PUFs. A PUF signature can be used for device authentication, or for generating secret keys and random variables in cryptographic devices. Deploying PUFs prevents the leakage of secret keys that would be stored in the device memories in lack of embedded PUFs, hence improves the security. However, due to the deployment of PUFs for device authentication and secret key generation purposes, reliability degradation in PUFs can cause significant security concerns.

Delay-PUFs are widely used in industrial applications as in these PUFs signal-to-noise (SNR) can be improved easily via increasing the number of delay elements. However, due to the multiple queries that these PUFs experience (as they only deliver one bit per measurement), they are more prone to aging [58], [59]. Figure 9 shows the schematic of one type of delay-PUFs so-called arbiter-PUFs. In practice, the reliability of this arbiter-PUF is highly affected by aging, as in this PUF, the arbiter itself (composed of an SR latch) experiences high bit-error rate due to aging. Fig. 10 depicts the aging-induced bit-error rate of this arbiter in $45°$C [59]. The results have been extracted using the 45nm NANGATE technology. As shown, in the deployed arbiter, the aging-induced bit-flips reaches to $\approx 10\%$ in 20 months when the PUF is fully active. The aging effects are exacerbated in higher temperatures such that in $80°$C, the effect would be 30% worse compared to $45°$C. As shown in this figure, the aging degradation is lower when the PUF is not always active. Thereby, as PUFs may not be queried frequently, the arbiter-PUF can be re-designed such that its embedded transistors are not aged significantly when the PUF is not active. One option is using sleep transistors to design such aging-resilient PUFs [60].

Aging mitigation schemes have been proposed in literature to address reliability concerns [8], [61]. Although the proposed schemes have been broadly adopted by industry to prolong the lifetime of integrated circuits, these schemes may not be highly beneficial to address security concerns when adversaries intentionally accelerate aging effects. An adversary may accelerate the aging process of an IC and thus shorten the devices lifespan. This type of hardware security threat results in denial of service to the IC user and may cause failure of the system. On the other hand, an adversary may aim at aging acceleration to thwart the protection schemes tailored against leaking secret information, e.g., secret key in cryptographic devices. Due to the deployment of cryptographic devices in applications with high level of security requirements, malicious aging acceleration in crypto devices may result in major catastrophes.

In fact, aging can be exacerbated via controlling external operating conditions such as temperature, power supply, and workload. In practice, an adversary can deliberately generate a workload such that when running, one (or more) targeted path(s) are aged significantly [62]. To generate such workload, the adversary needs to get access to the device gate-level netlist, either from a rogue element in the design house or via reverse engineering the layout. As the distribution of signal values (i.e., probability of holding the value of "0" or "1") as well as number of transitions between these values highly affect the aging process, the malicious workload can be generated such that most (if not all) of the PMOS transistors resided in a target path gets the value of "0" as their inputs, thereby, NBTI effect is accelerated in that path. The effect can be exacerbated if the adversary can also control temperature and power supply.

Figure 11 shows the effect of NBTI in propagation delay of primitive logic gates over time when these gates are fed with "00" continuously ("0" for inverter). As shown, each primitive gate experiences different amount of delay change related to its

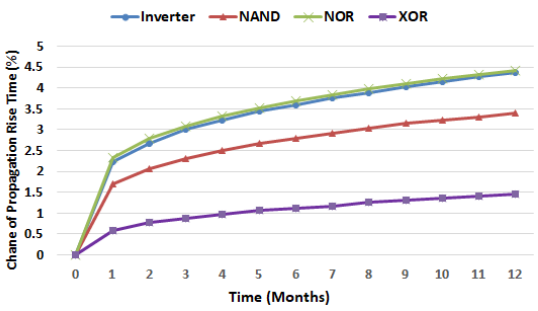

Fig. 11. Delay change of primitive gates due to NBTI aging.

transistor level topology. This observation confirms that each path degrades with a different rate based on the type of its underlying gates, and the input values feeding it. Thereby, an adversary can target specific paths for acceleration depend on his/her intention. For example, in case of aiming at denial of service, critical and near-critical paths can be targeted, while in case of leaking information, the paths whose delay change will result in information leakage can be considered for aging acceleration. Note that, in case of malicious aging acceleration, aging prognosis schemes are not useful as they predict the device wearout based on typical workloads.

Although device aging can jeopardize the security of digital circuits when malicious aging-induced denial of service comes into account or when aging adversely affects the circuitry aimed to be deployed for authentication purposes, in some cases, device aging may have a positive effect, i.e., aging makes the attacks aiming at information leakage more difficult. One such case is when an adversary uses profiling power analysis attacks [63] to retrieve secret keys from cryptographic devices. In profiling attacks, to recover the keys, the attacker gets benefit of another similar device that is under his/her full control. He/She uses the device under control to train a leakage model which is then used for leaking the keys of target devices with the same implementation. In practice, the similarity of the specifications of training and target devices are highly important in leaking sensitive data. Thereby, if the training and target devices have experienced different aging effects, the attack aiming at retrieving sensitive data would be more difficult.

In sum, device aging adversely affects the reliability of digital circuits, and thereby mitigation schemes are leveraged to address aging-induced reliability concerns. However, regarding security perspectives, aging may or may not result in security degradation. For example, aging may help an adversary in generating wrong challenge/response values in PUFs and thereby disqualifying device authentications using PUFs, but on the other hand it may decrease the success of profiling attacks launched to retrieve secret information.

VI. Conclusion

Device aging is a growing concern in emerging technologies. Due to the aging-related deviation of the specification of transistors embedded in a device, both reliability and security of the device can be affected. Current mitigation and prevention schemes may not be highly beneficial to address such concerns in emerging technologies. Thereby, there is a need for thorough investigation of aging effects in cutting edge technologies in order to tailor efficient reliability and security preserving schemes. In this paper, we first presented the effect of aging in a high performance SRAM memory realized in 14nm FinFET technology and showed how aging degrades the individual components of this memory as well as the interaction between them. Then, we investigated the current system level aging monitoring schemes from security perspectives. Finally, we presented a number of cases where aging positively/negatively affects security.

References

[1] 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.

[2] S. Khan, I. Agbo, S. Hamdioui, H. Kukner, B. Kaczer, P. Raghavan, and F. Catthoor, "Bias temperature instability analysis of finfet based sram cells," in *Design, Automation and Test in Europe (DATE)*, 2014, p. 31.

[3] Y. Lu et al., "Statistical reliability analysis under process variation and aging effects," in *DAC*, 2009, pp. 514–519.

[4] 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.

[5] 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.

[6] M. T. Rahman, F. Rahman, D. Forte, and M. Tehranipoor, "An aging-resistant ro-puf for reliable key generation," *IEEE Trans. on Emerging Topics in Computing*, vol. 4, no. 3, pp. 335–348, July 2016.

[7] 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.

[8] Y. Lee and T. Kim, "A fine-grained technique of nbti-aware voltage scaling and body biasing for standard cell based designs," in *ASP-DAC*, 2011, pp. 603–608.

[9] F. Catthoor and G. Groeseneken, "Will chips of the future learn how to feel pain and cure themselves?" *IEEE Design & Test*, vol. 34, no. 5, pp. 80–87, 2017.

[10] S. Hamdioui, M. Nicolaidis, D. Gizopoulos, A. Grasset, G. Guido, and P. Bonnot, "Reliability challenges of real-time systems in forthcoming technology nodes," in *Design, Automation and Test in Europe (DATE)*, 2013, pp. 129–134.

[11] P. Weckx, M. Simicic, K. Nomoto, M. Ono, B. Parvais, B. Kaczer, P. Raghavan, D. Linten, K. Sawada, H. Ammo *et al.*, "Defect-based compact modeling for rtn and bti variability," in *Int'l Symp. on Reliability Physics (IRPS)*, 2017, pp. CR–7.

[12] D. Kraak et al., "Degradation analysis of high performance 14nm finfet sram," in *Design, Automation and Test in Europe (DATE)*, 2018.

[13] "Predictive technology model," "http://ptm.asu.edu/".

[14] S. Bhardwaj, W. Wang, R. Vattikonda, Y. Cao, and S. Vrudhula, "Predictive modeling of the nbti effect for reliable design," in *Custom Integrated Circuits Conference (CICC), pages=189–192, year=2006, organization=*.

[15] 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.

[16] M. Liu, C. Zhou, Q. Tang, K. K. Parhi, and C. H. Kim, "A data remanence based approach to generate 100% stable keys from an sram physical unclonable function," in *Int'l Symp. on Low Power Electronics and Design*. IEEE, 2017, pp. 1–6.

[17] W. Wang, A. D. Singh, U. Guin, , and A. Chatterjee, "Exploiting power supply ramp rate for calibrating cell strength in sram pufs," in *Int'l Latin America Test Symp. (LATS)*, 2018, pp. 1–6.

[18] "Personenkraftwagen im durchschnitt 9,3 jahre alt," "https://www.kba.de/DE/Statistik/Fahrzeuge/Bestand/Fahrzeugalter/fahrzeugalter_node.html".

[19] W. Strong, A. et al., *Reliability wearout mechanisms in advanced CMOS technologies*. John Wiley & Sons, 2009, vol. 12.

[20] J. Tschanz, K. Bowman, S. Walstra, M. Agostinelli, T. Karnik, and V. De, "Tunable replica circuits and adaptive voltage-frequency techniques for dynamic voltage, temperature, and aging variation tolerance," in *Symp. on VLSI Circuits*, 2009, pp. 112–113.

[21] S. Wang, J. Chen, and M. Tehranipoor, "Representative critical reliability paths for low-cost and accurate on-chip aging evaluation," in *Int'l Conf. on Computer-Aided Design (ICCAD)*, 2012, pp. 736–741.

[22] R. Baranowski, A. Cook, M. E. Imhof, C. Liu, and H.-J. Wunderlich, "Synthesis of workload monitors for on-line stress prediction," in *Defect and Fault Tolerance in VLSI and Nanotechnology Systems (DFTS)*, 2013, pp. 137–142.

[23] H. Yi, T. Yoneda, and M. Inoue, "A scan-based on-line aging monitoring scheme," *Journal of Semiconductor Technology and Science*, vol. 14, no. 1, pp. 124–130, 2014.

[24] J. P. Keane, C. H. Kim, Q. Liu, and S. S. Sapatnekar, "Process and reliability sensors for nanoscale cmos," *IEEE Design & Test of Computers*, vol. 29, no. 5, pp. 8–17, 2012.

[25] R. Baranowski, F. Firouzi, S. Kiamehr, C. Liu, M. Tahoori, and H.-J. Wunderlich, "On-line prediction of nbti-induced aging rates," in *Design, Automation and Test in Europe (DATE)*, 2015, pp. 589–592.

[26] A. Vijayan, A. Koneru, S. Kiamehr, K. Chakrabarty, and M. B. Tahoori, "Fine-grained aging-induced delay prediction based on the monitoring of run-time stress," *IEEE Trans. on Computer-Aided Design of Integrated Circuits and Systems (TCAD)*, 2016.

[27] M. Agarwal, B. C. Paul, M. Zhang, and S. Mitra, "Circuit failure prediction and its application to transistor aging," in *VLSI Test Symp. (VTS)*, 2007, pp. 277–286.

[28] M. Agarwal, V. Balakrishnan, A. Bhuyan, K. Kim, B. C. Paul, W. Wang, B. Yang, Y. Cao, and S. Mitra, "Optimized circuit failure prediction for aging: Practicality and promise," in *Int'l Test Conference (ITC)*, 2008, pp. 1–10.

[29] H. Dadgour and K. Banerjee, "Aging-resilient design of pipelined architectures using novel detection and correction circuits," in *Design, Automation and Test in Europe (DATE)*, 2010, pp. 244–249.

[30] S. Das et al., "Razorii: In situ error detection and correction for pvt and ser tolerance," *IEEE Journal of Solid-State Circuits*, vol. 44, no. 1, pp. 32–48, 2009.

[31] M. Omana, D. Rossi, N. Bosio, and C. Metra, "Low cost nbti degradation detection and masking approaches," *IEEE Trans. on Computers*, vol. 62, no. 3, pp. 496–509, 2013.

[32] M. Saliva et al., "Digital circuits reliability with in-situ monitors in 28nm fully depleted SOI," in *Design, Automation, and Test in Europe (DATE)*, 2015, pp. 441–446.

[33] J. Vazquez et al., "Programmable aging sensor for automotive safety-critical applications," in *Design, Automation and Test in Europe (DATE)*, 2010, pp. 618–621.

[34] D. Ernst et al., "Razor: circuit-level correction of timing errors for low-power operation," *IEEE Micro*, vol. 24, no. 6, pp. 10–20, 2004.

[35] C. Liu, M. A. Kochte, and H.-J. Wunderlich, "Efficient observation point selection for aging monitoring," in *Int'l On-Line Testing Symposium (IOLTS)*, 2015, pp. 176–181.

[36] L. Lai, V. Chandra, R. Aitken, and P. Gupta, "Slackprobe: A low overhead in situ on-line timing slack monitoring methodology," in *Design, Automation and Test in Europe (DATE)*, 2013, pp. 282–287.

[37] R. Carlsten, J. Ralston-Good, and D. Goodman, "An approach to detect negative bias temperature instability (nbti) in ultra-deep submicron technologies," in *Int'l Symp. on Circuits and Systems (ISCAS)*, 2007, pp. 1257–1260.

[38] T.-H. Kim, R. Persaud, and C. H. Kim, "Silicon odometer: An on-chip reliability monitor for measuring frequency degradation of digital circuits," *IEEE Journal of Solid-State Circuits*, vol. 43, no. 4, pp. 874–880, 2008.

[39] A. Ghosh, R. B. Brown, R. M. Rao, and C.-T. Chuang, "A precise negative bias temperature instability sensor using slew-rate monitor circuitry," in *Int'l Symp. on Circuits and Systems (ISCAS)*, 2009, pp. 381–384.

[40] A.-W. Hakmi, S. Holst, H.-J. Wunderlich, J. Schlöffel, F. Hapke, and A. Glowatz, "Restrict encoding for mixed-mode bist," in *VLSI Test Symp. (VTS)*, 2009, pp. 179–184.

[41] T. Ter Braak, S. Burgess, H. Hurskainen, H. G. Kerkhoff, B. Vermeulen, and X. Zhang, "On-line dependability enhancement of multiprocessor socs by resource management," in *Int'l Symp. on System on Chip (SoC)*, 2010, pp. 103–110.

[42] A. Jutman, S. Devadze, and K. Shibin, "Effective scalable ieee 1687 instrumentation network for fault management," *IEEE Design & Test*, vol. 30, no. 5, pp. 26–35, 2013.

[43] M. T. He and M. Tehranipoor, "Sam: A comprehensive mechanism for accessing embedded sensors in modern socs," in *Symp. on Defect and Fault Tolerance in VLSI and Nanotechnology Systems (DFT)*, 2014, pp. 240–245.

[44] F. G. Zadegan, D. Nikolov, and E. Larsson, "A self-reconfiguring ieee 1687 network for fault monitoring," in *European Test Symposium (ETS)*, 2016, pp. 1–6.

[45] M. A. Kochte, M. Sauer, L. R. Gomez, P. Raiola, B. Becker, and H.-J. Wunderlich, "Specification and verification of security in reconfigurable scan networks," in *European Test Symp. (ETS)*, 2017, pp. 1–6.

[46] K. Rosenfeld and R. Karri, "Attacks and defenses for jtag," *IEEE Design & Test of Computers*, vol. 27, no. 1, 2010.

[47] R. Baranowski, M. A. Kochte, and H.-J. Wunderlich, "Fine-grained access management in reconfigurable scan networks," *IEEE Trans. on Computer-Aided Design of Integrated Circuits and Systems*, vol. 34, no. 6, pp. 937–946, 2015.

[48] M. Tehranipoor and C. Wang, *Introduction to hardware security and trust*. Springer Science & Business Media, 2011.

[49] B. Yang, K. Wu, and R. Karri, "Scan based side channel attack on dedicated hardware implementations of data encryption standard," in *Int'l Test Conf.(ITC)*, 2004, pp. 339–344.

[50] J. Lee, M. Tehranipoor, C. Patel, and J. Plusquellic, "Securing designs against scan-based side-channel attacks," *IEEE Trans. on dependable and secure computing*, vol. 4, no. 4, pp. 325–336, 2007.

[51] K. Park, S. G. Yoo, T. Kim, and J. Kim, "Jtag security system based on credentials," *Journal of Electronic Testing*, vol. 26, no. 5, pp. 549–557, 2010.

[52] G.-M. Chiu and J. C.-M. Li, "A secure test wrapper design against internal and boundary scan attacks for embedded cores," *IEEE Trans. on Very Large Scale Integration (VLSI) Systems*, vol. 20, no. 1, pp. 126–134, 2012.

[53] Y. Shi, N. Togawa, M. Yanagisawa, and T. Ohtsuki, "Robust secure scan design against scan-based differential cryptanalysis," *IEEE Trans. on Very Large Scale Integration (VLSI) Systems*, vol. 20, no. 1, pp. 176–181, 2012.

[54] J. Dworak and A. Crouch, "A call to action: Securing ieee 1687 and the need for an ieee test security standard," in *VLSI Test Symp. (VTS)*, 2015, pp. 1–4.

[55] M. A. Kochte and H.-J. Wunderlich, "Dependable on-chip infrastructure for dependable mpsocs," in *Latin-American Test Symp. (LATS)*, 2016, pp. 183–188.

[56] A. Koneru, A. Vijayan, K. Chakrabarty, and M. B. Tahoori, "Fine-grained aging prediction based on the monitoring of run-time stress using dft infrastructure," in *Int'l Conf. on Computer-Aided Design (ICCAD)*, 2015, pp. 51–58.

[57] N. Karimi and K. Huang, "Prognosis of nbti aging using a machine learning scheme," in *Defect and Fault Tolerance in VLSI and Nanotechnology Systems (DFTS)*, 2016, pp. 7–10.

[58] N. Karimi, J.-L. Danger, S. Guilley, and F. Lozach, "Predictive aging of reliability of two delay PUFs," in *Security, Privacy, and Applied Cryptography Engineering (SPACE)*, 2016, pp. 213–232.

[59] N. Karimi, J.-L. D. M. Slimani, and S. Guilley, "Impact of the switching activity on the aging of delay-PUFs," in *European Test Symp. (ETS)*, 2017.

[60] A. Calimera, E. Macii, and M. Poncino, "Nbti-aware sleep transistor design for reliable power-gating," in *Great Lakes symposium on VLSI (GLSVLSI)*, 2009, pp. 333–338.

[61] D. R. Bild, G. E. Bok, and R. P. Dick, "Minimization of NBTI performance degradation using internal node control," in *DATE*, 2009, pp. 148–153.

[62] N. Karimi, A. K. Kanuparthi, X. Wang, O. Sinanoglu, and R. Karri, "MAGIC: Malicious aging in circuits/cores," *ACM Trans. on Architecture and Code Optimization (TACO)*, vol. 12, no. 1, p. 5, 2015.

[63] S. Chari, J. R. Rao, and P. Rohatgi, "Template attacks," in *CHES*, 2002, pp. 13–28.