Date of publication xxxx 00, 0000, date of current version xxxx 00, 0000.

Digital Object Identifier 10.1109/ACCESS.2023.0322000

The Impact of Asymmetric Transistor Aging on Clock Tree Design Considerations

FIRAS RAMADAN[1], (Graduate Student Member, IEEE), MAJD GANAEIM[1], MAAYAN ELLA[1], AND FREDDY GABBAY[2], (Senior Member, IEEE)

[1]Faculty of Electrical and Computer Engineering, Technion - Israel Institute of Technology, Haifa, 3200003, Israel
[2]Faculty of Sciences, Institute of Applied Physics, The Hebrew University of Jerusalem, Jerusalem, Israel

Corresponding author: Freddy Gabbay (e-mail: freddy.gabbay@mail.huji.ac.il).

ABSTRACT Ensuring integrated circuits (ICs) operate reliably throughout their expected service life is more vital than ever, particularly as they become increasingly central to mission-critical applications. Advances in semiconductor technology have brought to light the vulnerability of ICs to various reliability challenges, notably those stemming from the phenomenon of transistor aging. Transistor aging refers to the progressive degradation of transistor performance over time. This degradation is predominantly due to bias-temperature instability (BTI), which can significantly undermine the reliability of ICs, leading to performance degradation and the potential for critical failures through timing violations. The situation is further complicated by the occurrence of asymmetric transistor aging, where the degradation is not uniformly distributed, thus intensifying timing violations and reliability concerns. Our study delves into the impact of asymmetric transistor aging on clock tree design and underscores the importance of useful skew, clock gating, and the variances between clock buffer delays and net delays in exacerbating timing violations. In response, we introduce extended timing constraints, a clock tree anti-aging circuitry and a extended design flow aimed at alleviating the effects of asymmetric transistor aging on clock trees, thereby enhancing IC reliability. Our simulation analysis investigates the vulnerability of clock trees to asymmetric aging, using General-Purpose Graphics Processing Units (GPGPUs) as a case study, and highlights the resulting timing violations when factoring in asymmetric transistor aging. The anti-aging circuitry and design flow are validated through aging-aware timing analysis, which confirms their effectiveness in eliminating the observed timing violations.

INDEX TERMS Asymmetric Aging, BTI, Clock-tree, Reliability, Transistor aging.

I. INTRODUCTION

OVER the past several decades, the field of Very Large Scale Integration (VLSI) technology has experienced significant advancements in a few key areas. The relentless pursuit of transistor miniaturization has yielded ever-smaller process nodes, shrinking transistors to nanoscale dimensions in accordance with Moore's law. Additionally, the introduction of cutting-edge devices and novel materials has been central to enhancing performance and reducing power consumption. Despite these technological strides, such progress has laid bare the susceptibility of integrated circuits (ICs) to reliability challenges, especially those stemming from transistor aging. This aging process—a gradual decline in transistor performance—is mainly ascribed to bias-temperature instability (BTI), which will be explored thoroughly in Section II. BTI's influence on IC reliability is profound, as it not only deteriorates performance but also leads to critical failures through timing violations. The issue is further compounded by asymmetric aging, which is particularly problematic due to the uneven degradation it causes, thereby intensifying timing violations and heightening concerns over reliability.

As VLSI technologies continue to evolve, semiconductors are becoming integral components of mission-critical systems across various sectors such as autonomous transportation, healthcare devices, financial services, and security infrastructure [1], [2]. These advanced applications demand heightened levels of resilience, reliability, and safety from integrated circuits (ICs), as mandated by regulatory bodies and industry benchmarks [3]. Consequently, the imperative for IC design now encompasses a stringent focus on reliability.

This paper investigates the impact of asymmetric transistor aging on clock tree design considerations. Clock trees are critical circuit resources responsible for distributing a balanced clock signal across the chip die. The reliability of the clock tree is vital to IC reliability because even a single clock tree failure point can lead to the complete failure of the entire clock distribution network. This paper shows that clock trees are highly vulnerable to reliability issues caused by asymmetric transistor aging. While earlier research [4], [5] has concentrated on the effects of gated clocks on asymmetric aging, our previous work [20] has expanded on these findings by:

(1) Uncovering how certain factors, such as useful skew and asymmetry between net and cell delays, can induce significant timing discrepancies within the clock tree, potentially leading to a complete IC failure.
(2) Performing a detailed simulation analysis within a case study that scrutinizes the vulnerability of clock trees to asymmetric aging, specifically within Nvidia Volta V100 General-Purpose Graphics Processing Units (GPGPUs), and identifying the resulting timing violations when asymmetric transistor aging is taken into account.
(3) Introducing enhanced timing constraints and revised design flow strategies that address the challenges of asymmetric transistor aging. These guidelines can be applied during the physical design and timing verification phases to alleviate the reliability impact of asymmetric aging on transistors.

This paper extends our previous work [20] by:

1) Introducing new anti-aging circuitry for mitigating asymmetric transistor aging in clock trees.
2) Broadening the range of GPGPUs under simulation analysis to include not only the Volta V100 but also the Nvidia RTX 2060, thereby assessing the impact of asymmetric transistor aging across various GPGPU architectures and ensuring the robustness of our mitigation techniques.
3) Extending the variety of workloads executed on GPGPUs to encompass the neural network (NN) benchmark, and examining Breadth-First Search (BFS), and the N-Queens Solver (NQS) [17].

The structure of the rest of this paper is organized as follows: Section II provides an overview of the background and reviews related literature. Section III discusses the susceptibility of clock trees to asymmetric transistor aging and proposes extensions to existing timing constraints. Section IV introduces our clock tree anti-aging circuitry. Our simulation results are presented in Section V, while Section 6 offers the concluding remarks of this study.

II. BACKGROUND AND PRIOR WORK

This section provides background information on transistor aging and the BTI and reviews previous studies in the field of transistor aging. It should be noted that ICs may also incur aging of net elements, governed by electromigration [6]–[8], which is beyond the scope of this paper. Throughout our analysis, we operate under the assumption that nets comply with electromigration design rules.

A. TRANSISTOR AGING

Transistor aging refers to the deterioration over time of transistors in digital circuits and is caused by the trapping of charge carriers from the transistor inversion channel at the dielectric insulator of the transistor gate [9], [10]. The BTI is recognized as the primary mechanism governing transistor aging. The BTI activates when a constant voltage is applied to the transistor gate, elevating the transistor's threshold voltage. This increase in threshold voltage leads to a longer transistor switching delay, thereby reducing the transistor's speed. Asymmetric transistor aging, which refers to the uneven distribution of performance degradation among transistors within an IC, can lead to severe timing issues, including setup and hold timing violations.

The aging model used in this study to describe V_{th} degradation relies on the reaction-diffusion model—the most widely acknowledged model for BTI aging within both industry and research communities [11]. The model provides the following equation to describe V_{th} degradation, denoted as ΔV_{th}, resulting from BTI stress:

$$\Delta V_{\mathrm{th}} \propto e^{\frac{E_a}{kT}} (t - t_0)^{1/6} \tag{1}$$

where E_a is a constant, T is the operating temperature, k is Boltzmann's constant, t_0 is the time when the BTI stress starts, and t is the overall time. A key insight from this model is that substantial V_{th} degradation happens early in the IC's lifetime. For example, approximately 70% of the V_{th} degradation within a 10-year time frame occurs within the first year. NBTI, known to affect p-type transistors, poses a greater susceptibility than PBTI, which affects n-type transistors [10]. Hence, logical gates maintaining a persistent idle state of logical 0 are most vulnerable to aging.

The signal probability (SP) is a common technique [4] for assessing the BTI stress profile on logical elements. The SP quantifies the probability of a signal having a logical value of 1 and is defined as the ratio of the time a signal spends in the logical-1 state to the overall time. A decrease in SP intensifies the effect of the NBTI, resulting in performance degradation or potentially causing failures in integrated circuits over time.

B. PRIOR WORK

Common approaches involve incorporating additional timing margins to mitigate the effects of asymmetric aging. However, such approaches often necessitates complex simulation analysis and can lead to overdesign [11]. Other studies [9], [12], [13] have proposed models for predicting aging degradation and have explored various solutions, including reducing clock cycle time, transistor resizing, V_{DD} tuning, and power gating. Agrawal et al. [14] proposed a method to predict circuit failure by using sensors placed at various locations within the silicon die. Additional research [15] has explored techniques to analyze digital circuits and detect the most vulnerable gates affected by NBTI. This involves using an aging model with BTI-aware libraries and conducting aging-aware timing analysis. Abbas et al. [16] proposed executing anti-aging programs instead of idle tasks during periods of low processor use. Gabbay et al. [4] proposed an aging-aware microarchitecture to minimize the effects of asymmetric aging on execution units, register files, and memory hierarchy elements in microprocessors while keeping overhead to a minimum. Arasu et al. [5] analyzed asymmetric aging in the clock tree segments of power-efficient designs by using a 45 nm process node. The authors examined how BTI affects the clock tree as a result of clock gates and built-in clock tree skews. However, they did not consider how useful skew and the asymmetry between net and cell delays affect clock tree design. These factors are further addressed in the present study using a 28 nm process node.

III. IMPACT OF ASYMMETRIC TRANSISTOR AGING ON CLOCK TREES

Clock trees play a pivotal role in distributing the clock signal throughout digital circuits, aiming to achieve minimal insertion delay and to maintain uniformity in clock skew at all endpoints. The clock signal is fundamental for the proper logical functioning of digital circuits, and any malfunction within the clock tree could lead to a complete circuit failure. Consequently, to guarantee the clock signal's reliable performance, setup and hold timing constraints must be rigorously applied.

This paper identifies three principal elements that contribute to asymmetric aging within clock trees: the use of clock gating, the asymmetry in delays between cells and interconnects, and the intentional introduction of skew for performance optimization. Furthermore, it proposes augmented timing constraints that consider the effects of asymmetric transistor aging. These constraints can be integrated into the workflow of physical design engineers and Electronic Design Automation (EDA) tools to maintain timing integrity despite the presence of asymmetric aging phenomena.

A. CLOCK GATING

Clock gating is a prevalent technique utilized to reduce dynamic power usage. This method involves selectively disabling the clock signal to portions of the circuit that are not active, which in turn decreases dynamic power usage. By halting the clock in dormant sections of the circuit, superfluous switching and its related power use are eliminated. Generally, clock gating is implemented with a clock gate cell that includes both a latch and an and gate.

Clock gating exacerbates asymmetric aging by promoting inactivity within the clock network, as illustrated in Figs. 1(a) and 1(b). Specifically, Fig. 1(a) demonstrates that employing a clock gate in the launch path induces greater aging there compared to the capture path, potentially causing setup timing violations. On the other hand, as shown in Fig. 1(b), when a clock gate is utilized in the capture path, it ages more rapidly than the launch path, which may lead to hold timing violations.

B. ASYMMETRY BETWEEN CELL AND NET DELAYS

The phenomenon of asymmetric aging within clock networks may also stem from the variation in the sum of delays across logical cells and nets. Nets, in contrast to logical cells, remain unaffected by BTI. Asymmetry in the total delay attributed to logical cells within launch and capture paths can lead to BTI-driven asymmetric aging, as depicted in Fig. 1(c). A scenario where the cumulative delay from logical cells in the launch path surpasses that in the capture path might precipitate setup timing violations. Inversely, if the cumulative delay from cells in the launch path is less than that in the capture path, this could give rise to hold timing violations. Fig. 1(c) portrays a situation where both launch and capture paths exhibit a balanced clock insertion delay of 170 ps. Yet, the aggregate delay from clock buffers in the capture path is 150 ps, in contrast to a 100 ps total delay in the launch path's clock buffers. Even with a uniform aging rate across all clock buffers, the asymmetry in accumulated delays between cells and nets, when coupled with BTI effects, can induce hold timing violations due to the resultant delay shift in the capture clock.

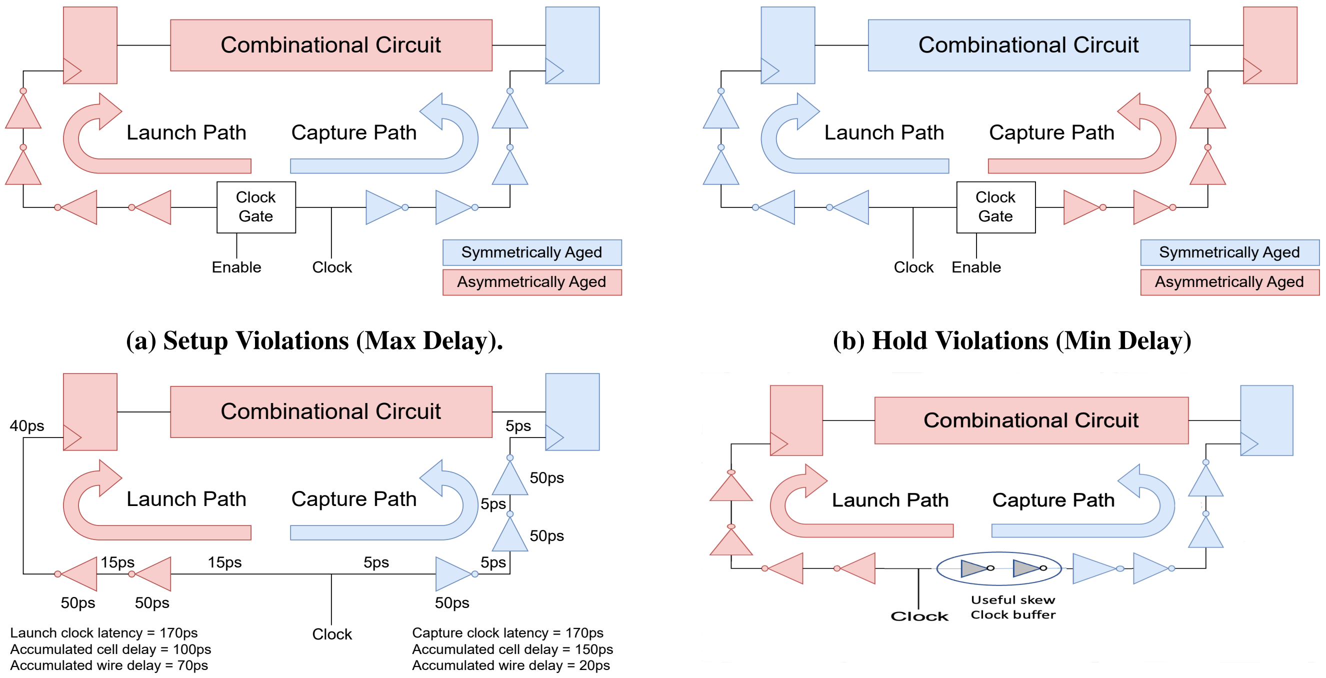
C. USEFUL SKEW

Employing useful skew is a strategic approach in clock tree synthesis, characterized by the deliberate insertion of delays within clock paths to alleviate setup or hold timing violations. Introducing a clock skew to the capture path, as exemplified in Fig. 1(d), facilitates the extension of the design's critical timing path beyond the nominal clock cycle duration. This technique is viable only when there is sufficient positive hold slack present to absorb the additional delay imposed on the capture clock. Consequently, clock skew can serve dual purposes: it can preempt the need to lengthen the clock cycle time or it can rectify timing violations. Similarly, when useful skew is applied to the launch path, it functions analogously, enhancing hold margins while potentially compromising setup times.

The implementation of useful clock skew can heighten the vulnerability of clock trees to asymmetric transistor aging, especially when used alongside clock gating or amidst asymmetry between cell and net delays. Clock gating may exacerbate the asymmetric aging of skew buffers, potentially leading to timing violations in light of transistor aging. Furthermore, clock skew buffers by their nature introduce a fundamental asymmetry between the cumulative delays of nets and cells, amplifying the risk of timing violations due to transistor aging. Consequently, it is prudent to apply useful clock skew with caution, with particular attention to the implications of asymmetric transistor aging.

D. TIMING CONSTRAINTS IN THE PRESENCE OF ASYMMETRIC AGING

The timing constraints for a typical synchronous digital circuit, as depicted in Fig. 2, are governed by the equations referenced as (2) and (3). The specific timing parameters pertinent to the circuit are detailed in Table 1, assuming a clock cycle duration of T. The useful skew buffers, as presented in Fig. 2, are incorporated into the capture path and are characterized by


Combinational Circuit
Launch Path
Capture Path
Clock Gate
Enable
Clock
Symmetrically Aged
Asymmetrically Aged
40ps
5ps
50ps
5ps
50ps
15ps
15ps
5ps
5ps
50ps
50ps
50ps
Launch clock latency = 170ps
Accumulated cell delay = 100ps
Accumulated wire delay = 70ps
Capture clock latency = 170ps
Accumulated cell delay = 150ps
Accumulated wire delay = 20ps
Useful skew
Clock buffer

(a) Setup Violations (Max Delay).

(b) Hold Violations (Min Delay).

(c) Potential Hold/Setup Violations

(d) Useful Clock Skew

FIGURE 1: Possible violation due to asymmetric aging induced by (a) launch path clock gate; (b) capture path clock gate; (c) the asymmetry between the accumulated delay of logical cells and wires; and (d) useful skew

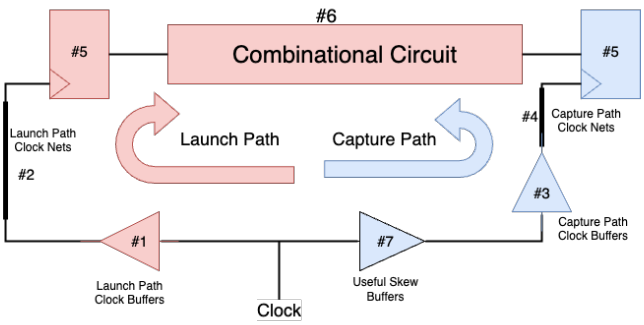


FIGURE 2: Asymmetric aging in clock trees in the presence of useful skew and net delays.

a delay parameter t_{us}. A positive t_{us} value ($t_{\text{us}} > 0$) indicates that the useful skew is being applied to the capture path, whereas a negative t_{us} value ($t_{\text{us}} < 0$) signifies its application to the launch path.

$$\begin{aligned}\Delta slack_{\text{setup}} &= T - t_{\text{pdFF}} - t_{\text{pdC}} - t_{\text{s}} + t_{\text{us}} \\ &\quad + (t_{\text{CC}} + t_{\text{NC}}) - (t_{\text{CL}} + t_{\text{NL}}), &(2)\\ \Delta slack_{\text{hold}} &= t_{\text{cdFF}} + t_{\text{cdC}} + (t_{\text{CL}} + t_{\text{NL}}) \\ &\quad - (t_{\text{CC}} + t_{\text{NC}}) - t_{\text{us}} - t_{\text{h}}. &(3)\end{aligned}$$

TABLE 1: Timing parameters.

Elements	Timing Parameters			
	Propagation delay	***Containment delay***	***Setup time***	***Hold time***
Launch clock buffers #1	t_{CL}	n/a	n/a	n/a
Launch clock nets #2	t_{NL}	n/a	n/a	n/a
Capture clock buffers #3	t_{CC}	n/a	n/a	n/a
Capture clock nets #4	t_{NC}	n/a	n/a	n/a
Flip-flops #5	t_{pdFF}	t_{cdFF}	t_{s}	t_{h}
Combinational circuit #6	t_{pdC}	t_{cdC}	n/a	n/a
Useful skew buffers #7	t_{us}	n/a	n/a	n/a

Given that the launch and capture paths are subject to asymmetric aging, a consequence of different activation of clock gating (or different SPs), the derate factors attributable to BTI can be denoted as $d_{\text{L}} > 1$ for the launch path and $d_{\text{C}} > 1$ for the capture path. When these derate factors are incorporated into the equations as referenced in (2) and (3), the resulting setup and hold slacks for an aged circuit can be articulated using Eqs. (4) and (5) respectively:

$$\begin{aligned}\Delta slack^{\text{aged}}_{\text{setup}} &= T - d_{\text{L}}(t_{\text{pdFF}} + t_{\text{pdC}} + t_{\text{CL}}) - t_{\text{NL}} \\ &\quad + d_{\text{C}}(t_{\text{CC}} + t_{\text{us}}) + t_{\text{NC}} - t_{\text{s}}, &(4)\\ \Delta slack^{\text{aged}}_{\text{hold}} &= d_{\text{L}}(t_{\text{cdFF}} + t_{\text{cdC}} + t_{\text{CL}}) + t_{\text{NL}} \\ &\quad - d_{\text{C}}(t_{\text{CC}} + t_{\text{us}}) - t_{\text{NC}} - t_{\text{h}}. &(5)\end{aligned}$$

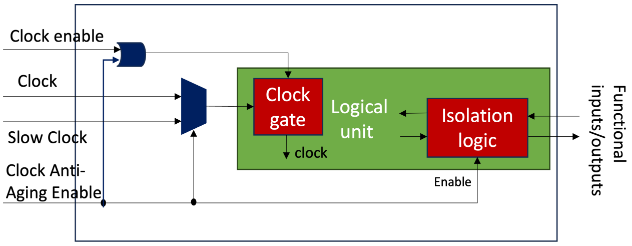


FIGURE 3: Clock tree anti-aging circuitry.

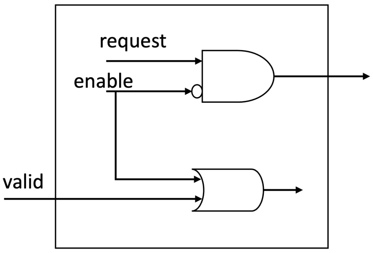


FIGURE 4: Isolation cells.

Denote δ_{L} and δ_{C} as the incremental shift fractions in the launch and capture paths, respectively, where $\delta_{\mathrm{L}} = d_{\mathrm{L}} - 1$, and similarly, $\delta_{\mathrm{C}} = d_{\mathrm{C}} - 1$. By integrating Eq. 2 with Eq. 4 and Eq. 3 with Eq. 5, we can deduce the setup and hold slacks for a circuit that has experienced asymmetric aging, as formulated by Eqs. 6 and 7, respectively.

$$\begin{aligned}\Delta slack^{\mathrm{aged}}_{\mathrm{setup}} &= \Delta slack_{\mathrm{setup}} - [\delta_{\mathrm{L}}(t_{\mathrm{pdFF}} + t_{\mathrm{pdC}} + t_{\mathrm{CL}}) \\ &\quad -\delta_{\mathrm{C}}(t_{\mathrm{CC}} + t_{\mathrm{us}})] \qquad (6) \\ &= \Delta slack_{\mathrm{setup}} - \Delta slack^{\mathrm{degradation}}_{\mathrm{setup}},\end{aligned}$$

$$\begin{aligned}\Delta slack^{\mathrm{aged}}_{\mathrm{hold}} &= \Delta slack_{\mathrm{hold}} - [\delta_{\mathrm{C}}(t_{\mathrm{CC}} + t_{\mathrm{us}}) \\ &\quad -\delta_{\mathrm{L}}(t_{\mathrm{cdFF}} + t_{\mathrm{cdC}} + t_{\mathrm{CL}})] \qquad (7) \\ &= \Delta slack_{\mathrm{hold}} - \Delta slack^{\mathrm{degradation}}_{\mathrm{hold}}.\end{aligned}$$

The degradation in the setup slack ($\Delta slack^{\mathrm{degradation}}_{\mathrm{setup}}$) and hold slack ($\Delta slack^{\mathrm{degradation}}_{\mathrm{hold}}$) due to asymmetric aging can be expressed by Eqs. 8 and 9, respectively:

$$\begin{aligned}\Delta slack^{\mathrm{degradation}}_{\mathrm{setup}} &= \delta_{\mathrm{L}}(t_{\mathrm{pdFF}} + t_{\mathrm{pdC}} + t_{\mathrm{CL}}) \\ &\quad -\delta_{\mathrm{C}}(t_{\mathrm{CC}} + t_{\mathrm{us}}), \qquad (8) \\ \Delta slack^{\mathrm{degradation}}_{\mathrm{hold}} &= \delta_{\mathrm{C}}(t_{\mathrm{CC}} + t_{\mathrm{us}}) \\ &\quad -\delta_{\mathrm{L}}(t_{\mathrm{cdFF}} + t_{\mathrm{cdC}} + t_{\mathrm{CL}}). \qquad (9)\end{aligned}$$

Equations 8 and 9 delineate two contradictory forces that determine the deterioration of slack. Within the context of setup, a greater delay shift in the launch path as compared to the capture path results in the diminution of setup slack. In contrast, the hold slack diminishes when the delay delay shift in the capture path surpasses that in the launch path. Even in cases where $\delta_{\mathrm{L}} = \delta_{\mathrm{C}}$, signifying uniform aging across both the capture and launch paths, a reduction in both types of slack is still possible. Furthermore, setup timing violations may arise if $\Delta slack^{\mathrm{degradation}}_{\mathrm{setup}} > \Delta slack_{\mathrm{setup}}$, while hold timing violations are likely to occur when $\Delta slack^{\mathrm{degradation}}_{\mathrm{hold}} > \Delta slack_{\mathrm{hold}}$.

IV. CLOCK TREE ANTI-AGING CIRCUITRY

This section introduces our clock tree anti-aging circuitry, which mitigates the impact of asymmetric transistor aging on clock trees. The scheme is illustrated in Figure 3 and consists of a typical logical unit where its clock is globally controlled by a clock gate through a **Clock enable** signal from a control unit external to the module (e.g. power management unit).

Our technique utilizes **isolation logic** circuitry [22], typically employed in the top-level hierarchy of various functional blocks or modules for Design for Testability (DFT) [21] purposes, e.g. Automatic Test Pattern Generation (ATPG). Isolation logic consists of cells that are additionally inserted by synthesis tools to isolate buses or wires crossing from the interface of a functional block to its top-level module. Moreover, isolation cells enable the testing of individual blocks or modules while masking the impact of the tested block on circuitry external to the block. The principle of operation of isolation cells is straightforward as illustrated in Figure 4. As illustrated in Figure 4. When an external signal (e.g., a request signal) needs to be masked to a logical 0, an AND gate is inserted into the signal path. When the isolation cell is activated (enable = 1), the AND operation with 0 effectively forces the output to a logical 0. Similarly, for masking to a logical 1 (e.g., valid signal), an OR gate is added. When isolation is activated(enable = 1), the OR operation ensures the output remains at a logical 1.

Inspired by the use of isolation cells in DFT, we employ isolation cells to ensure the clock tree's anti-aging mechanism operate seamlessly, without affecting the surrounding components. As illustrated in Figure 3, our anti-aging circuitry is activated by asserting the **Clock Anti-Aging Enable (CAAE)** signal. When CAAE is set to a logical one, it triggers the enabling of the global clock gate and also switches the source clock of the logical unit through a clock multiplexer. Instead of using the functional clock source, a free-running slow clock is injected into the block. This ensures continuous toggling of the clock tree within the logical unit, thereby preventing a prolonged duration of static logical states which could accelerate BTI-related asymmetric aging. It should be noted that if additional clock gates are used in the internal circuitry of the logical unit, they should be enabled when CAAE=1, similar to the global clock gate illustrated in Figure 3. Lastly, our proposed technique incurs a relatively small overhead in terms of area and power (summarized in Section V), as they involve only an extra OR gate and a clock multiplexer (assuming isolation logic is already part of the logical unit for DFT purposes).

V. SIMULATION ANALYSIS

In this section, we present our simulation results examining the effects of asymmetric transistor aging on the processing elements (PEs) of GPGPUs. Through functional simulations, we experimentally analyze GPGPUs' aging profile, including measurements of signal probability. Next, we conduct a detailed timing analysis that integrates aging models with the aging profile. Our experimental analysis investigates potential timing violations caused by asymmetric aging, considering the impact of:

1) Useful skew optimization.
2) Asymmetry between net and cell delays.
3) Computational workload running on the GPGPU.

In addition, we evaluate the effectiveness of our clock tree anti-aging circuitry in addressing asymmetric transistor aging timing violations. Finally, we extend existing physical design flows with the necessary enhancements to account for the impact of asymmetric transistor aging. Our functional experiments utilized the gpgpu-sim simulator described in [17]. The gpgpu-sim simulation environment provides cycle-level modeling of the NVIDIA Volta V100 and RTX 2060 GPGPUs [18], allowing for the execution of CUDA or OpenCL computing workloads. To align with the objectives of our experiments, we customized the simulation platform and incorporated necessary mechanisms for the required aging profile measurements. For benchmarking purposes, we employed the neural network (NN), Breadth-First Search (BFS), and the N-Queens Solver (NQS) benchmarks from the gpgpu-sim benchmark suite of IPSS [17].

As part of our functional experimental analysis, we measured the SP of the integer execution unit and the single-precision floating-point unit (FPU). Figure 5 illustrates the activity measured in the Volta V100 and RTX2060 Streaming Multiprocessors (SMs) while running NN, BFS, and NQS benchmarks. Activity is quantified as the percentage of time the execution unit remains active relative to the total elapsed time. Our simulation results indicate that for Volta V100 (Figure 5a), the integer execution units within all SMs are idle $80\% - 85\%$, $88\% - 91\%$, and $98\% - 99\%$ of the time for NN, BFS, and NQU, respectively. For the RTX2060 (Figure 5c), the simulation results indicate that the integer execution units in all SMs are idle $0\% - 30\%$, $70\% - 80\%$ and $90\% - 100\%$ of the time for NN, BFS, and NQU, respectively. The lower utilization of the Volta V100 versus the RTX2060 is attributed to the greater number of SMs (80) in Volta V100 compared to the RTX2060 (30). The FPU presents a significantly smaller utilization with respect to the integer execution units. For the Volta V100 (Figure 5a), the FPU is idle nearly $98\% - 100\%$, 100% and 100% of the time for NN, BFS, and NQU, respectively. The RTX2060 FPU presents higher utilization compared to the Volta V100 due to the smaller number of SMs in the RTX2060. Figure 5d shows that the FPU in the RTX2060 is idle $87\% - 91\%$, 100% and 100% of the time for NN, BFS, and NQU, respectively. Our observations suggest that GPGPU processing elements may be vulnerable to transistor aging due to significant idle time. These extended periods without activity can exacerbate asymmetric aging, leading to potential reliability issues.

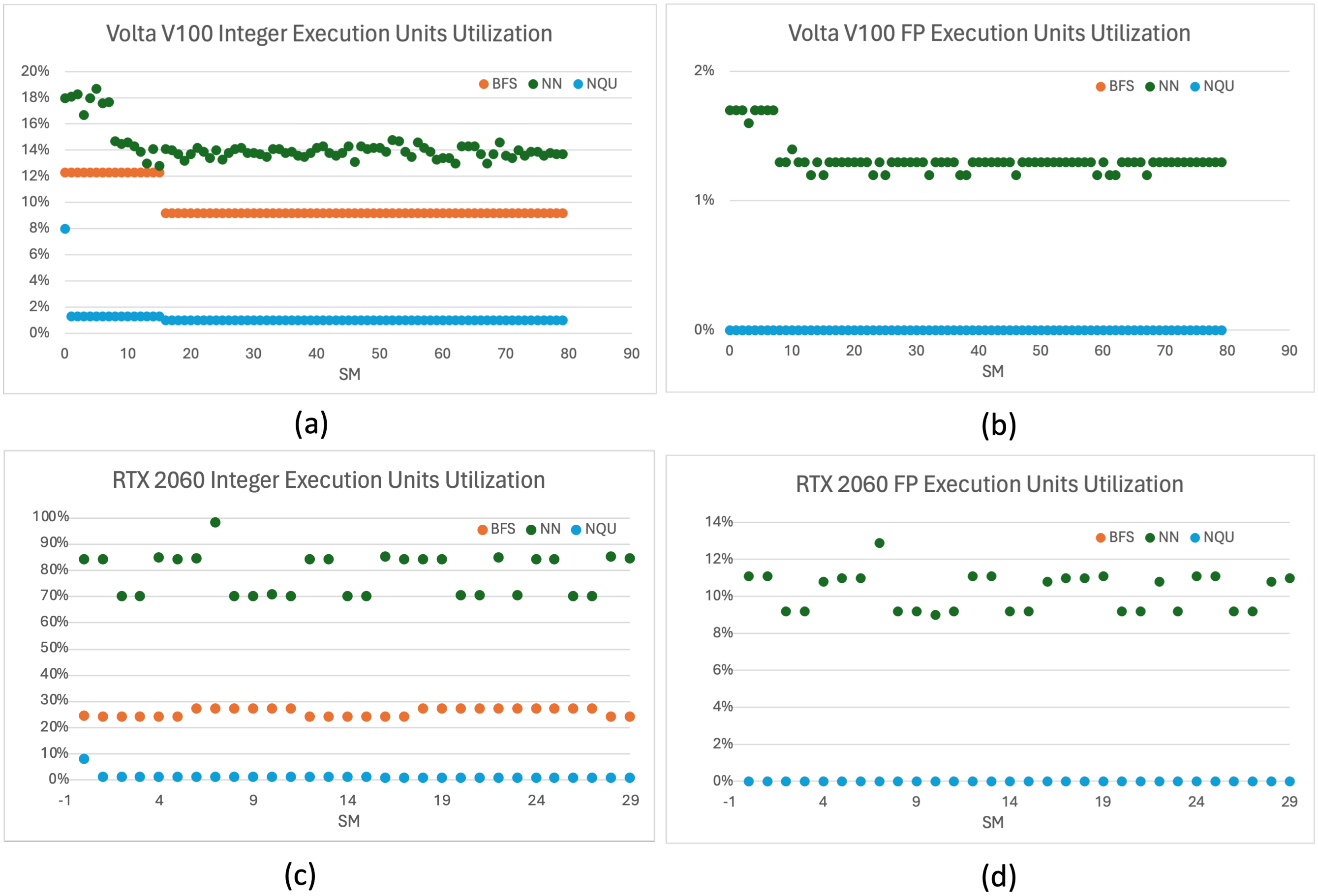
To explore the GPGPU case study, we employ the integer execution unit and Floating Point Unit (FPU) from the open-source Nyuzi Processor GPGPU[1] [19]. This approach allows us to examine the impact of asymmetric aging on timing. We performed synthesis and place-and-route processes on the GPGPU modules using the 28 nm technology node. The synthesis is conducted using Cadence® *Genus*™, and the place-and-route process is carried out with Cadence® *Innovus*™. The clock frequency for the integer execution unit is set at 250 MHz, while the FPU is designed to operate at 167 MHz. For the timing analysis, we utilize the measured aging profiles of the Volta V100 and the RTX2060, in combination with aging-aware library models, as detailed in Ref. [4]. These models account for the impact of Bias Temperature Instability (BTI) by adjusting cell delays based on Negative BTI (NBTI) degradation factors, which are derived from SP values extracted from the functional simulations illustrated in Fig. 5. We employ the reaction-diffusion model [11] described in Section II to model V_{th} degradation. The derate factors for the aged libraries are generated using SPICE simulations, replacing the nominal V_{th} with the aged V_{th} to reflect the aging corresponding to the SP and a 10-year lifetime.

The timing results in Tables 2 and 3 present the worst negative slack (WNS) and the number of timing violations for Volta V100 and RTX2060 case studies, respectively while considering the average SP of our benchmarks. The tables summarize the setup and hold timing analyses in the following cases:

(1) Without considering the impact of asymmetric aging (No Aging);
(2) With the inclusion of asymmetric aging effects in timing analysis (Aging);
(3) Employing useful skew optimization while also considering the impact of asymmetric aging (Aging+U.S.);
(4) Applying useful skew optimization and taking into account both asymmetric aging and the asymmetry between net delay and cell delay (Aging+U.S.+N.D.).
(5) When applying our clock tree anti-aging circuitry (Anti-Age).

The timing analysis results, summarized in Table 2 for the Volta V100, show that setup violations occur in the integer execution unit (I-EXU) and the floating-point unit (FPU) when asymmetric aging is considered without any useful skew optimization. Additionally, this condition introduce slight changes to the hold slack of the I-EXU. However, when useful skew optimization is applied, taking into account the impact of asymmetric aging, there is an improvement in the setup's worst negative slack (WNS), and the number of violations in the I-EXU decreases due to a delayed capture clock edge. Despite these adjustments, the timing constraints within the FPU's logical path prevent the place-and-route tools from applying useful skew to the setup critical path. Consequently, no change is observed in the WNS and the number of violations. Useful skew, however, further degrades the hold slack by nearly 10 ps for the I-EXU and by 1.6 for the FPU. When

[1] https://github.com/jbush001/NyuziProcessor/tree/master/hardware/core



(a)

(b)

(c)

(d)

FIGURE 5: Activity percentages of execution units in Streaming Multiprocessors: (a) V100 GPU Integer execution Unit, (b) V100 GPU single precision FP execution units, (c) RTX2060 GPU Integer execution Unit, and (d) RTX2060 GPU single precision FP execution units.

TABLE 2: Volta V100 Summary of timing analysis in presence of asymmetric aging.

Func, Units	Setup WNS [ps] Number of Violations				
	No Aging	*Aging*	*Aging+U.S.*	*Aging+U.S.+N.D.*	*Anti-Age*
I-EXU	0 / 0	−230 / 34	−200 / 21	−240 / 35	0 / 0
FPU	0 / 0	−270 / 8341	−270 / 8341	−348 / 13006	0 / 0

Func. Units	Hold WNS [ps] Number of Violations				
	No Aging	*Aging*	*Aging+U.S.*	*Aging+U.S.+N.D.*	*Anti-Age*
I-EXU	+13 / 0	+12.4 / 0	+2.5 / 0	−1 / 8	0 / 0
FPU	+4.8 / 0	+4.8 / 0	+3.2 / 0	−1.2 / 7	0 / 0

useful skew optimization is applied, considering asymmetric transistor aging along with the asymmetry between net delay and cell delay, a significant number of violations for both setup and hold are revealed. In the setup analysis, the I-EXU's WNS increases from -200 ps to -240 ps, while the FPU's WNS worsens from -270 ps to -348 ps. Additionally, the number of setup violations in the I-EXU increases from 21 to 35, and in the FPU, from 8,341 to 13,006. In the hold analysis, 8 hold violations are identified for the I-EXU with a WNS of -1. For the FPU, the timing analysis identifies 7 hold violations with a WNS of -1.2. Table 3 presents the timing analysis results for the RTX 2060. It can be observed that the timing analysis of both the I-EXU and FPU in the RTX 2060 exhibit similar behavior and trends to those observed in the Volta V100 when considering asymmetric transistor aging, useful skew, and the asymmetry between net and cell delays. As part of our simulation analysis, we examine the efficiency of our clock tree anti-aging circuitry. Our timing analysis, presented in Tables 2 and 3, indicates that our anti-aging circuitry has been able to fully eliminate all timing violations associated with asymmetric transistor aging. Lastly, the area and power overhead of our proposed anti-aging circuitry is negligible, as can be observed in the summary presented in Table 4 for both the I-EXU and FPU.

TABLE 3: RTX2060 Summary of timing analysis in presence of asymmetric aging.

Func,	Setup WNS [ps] Number of Violations				
Units	*No Aging*	*Aging*	*Aging+U.S.*	*Aging+U.S.+N.D.*	*Anti-Age*
I-EXU	0 / 0	−188 / 20	−159 / 13	−214 / 22	0 / 0
FPU	0 / 0	−240 / 7466	−240 / 7466	−340 / 12811	0 / 0

Func,	Setup WNS [ps] Number of Violations				
Units	*No Aging*	*Aging*	*Aging+U.S.*	*Aging+U.S.+N.D.*	*Anti-Age*
I-EXU	+13 / 0	+13.3 / 0	+3.4 / 0	−0.2 / 2	0 / 0
FPU	+4.8 / 0	+4.8 / 0	+3.3 / 0	−1 / 6	0 / 0

TABLE 4: Anti-aging circuitry area and power overhead

Functional Unit	Area [%]	Power [%]
I-EXU	0.005	0.0002
FPU	0.001	0.00004

The experimental findings presented in Tables 2 and 3 reveal that asymmetric aging can lead to significant timing issues. Moreover, employing useful clock skew optimization alongside the asymmetry between net delay and cell delay can exacerbate these timing issues, further compromising the circuit's reliability. Hold violations are generally viewed as more critical than setup violations because, while reducing the clock frequency can mitigate setup violations, there is no remedy for hold violations. Therefore, in the presence of asymmetric transistor aging, the timing constraints should be extended as indicated by Eqs. (8) and (9).

Finally, Fig. 6 presents our design flow, which extends the conventional flow—comprising synthesis, place and route, extraction, timing report generation, and timing fixes—by incorporating new capabilities to account for the impact of asymmetric aging on timing. The new components added to the flow include:

(4) conducting a functional simulation of the design to assess idleness under specific workloads;

(5) deriving the aging profile as SP;

(6) performing aging-aware timing analysis using libraries derated according to their respective SPs.

(9) Incorporating anti-aging circuitry to mitigate asymmetric aging timing violations.

In summary, integrating these new capabilities into the physical design flow addresses asymmetric aging challenges, thereby enhancing timing analysis and design reliability, as evidenced by improvements in both the Volta V100 and RTX 2060 GPGPU processing elements.

VI. CONCLUSIONS

This research explores the impact of asymmetric aging on clock tree design considerations. Clock trees, crucial for delivering a balanced clock signal throughout the chip die, are essential for the reliable operation of ICs. Our findings indicate that clock trees are significantly vulnerable to reliability challenges caused by asymmetric aging. While earlier research mainly concentrated on the influence of gated clocks on asymmetric transistor aging, this study broadens the investigation. It underscores the importance of elements like useful skew and the asymmetry between net and cell delays, which can lead to significant timing violations in clock trees, potentially resulting in IC failure. Through a case study with the Volta V100 and RTX 2060 GPGPUs, our simulation analysis uncovers timing violations caused by asymmetric transistor aging. To address the issues arising from asymmetric aging, we introduce 1) an aging-aware design flow, which includes new extensions to the timing constraints, and 2) anti-aging circuitry for clock trees. Our simulation analysis shows that the anti-aging circuitry has been able to eliminate the timing violations while introducing very small area and power overhead. Grasping the impact of asymmetric aging on clock tree design is essential for maintaining the reliability and performance of ICs. Implementing anti-aging measures alongside an aging-aware design flow and extended timing constraints can help mitigate potential timing violations, ultimately enhancing the reliability ICs.

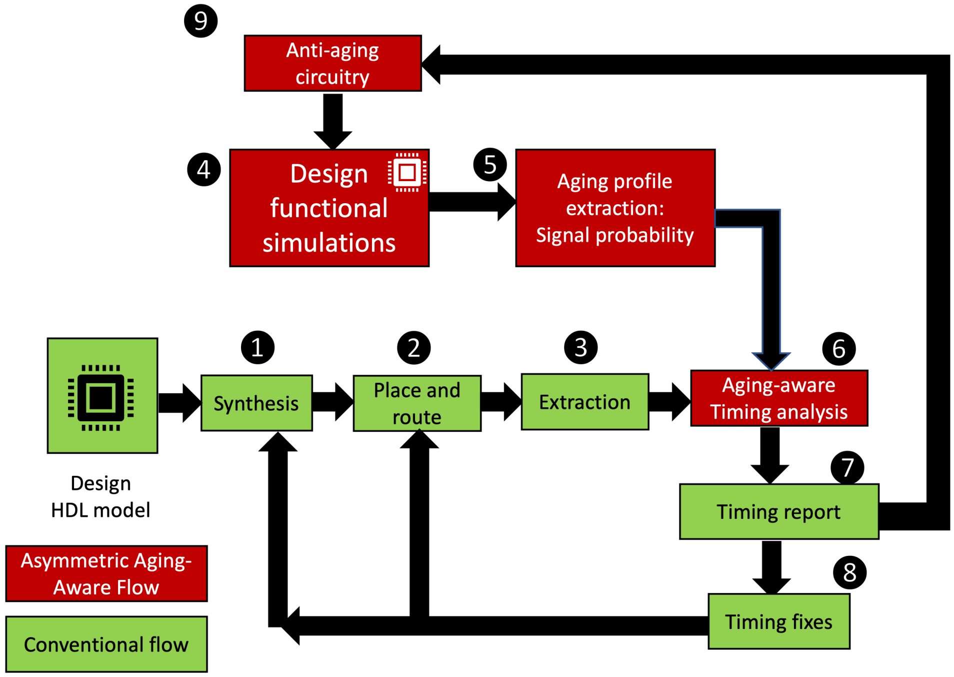

9
Anti-aging circuitry
4
Design functional simulations
5
Aging profile extraction: Signal probability
1
2
3
6
Synthesis
Place and route
Extraction
Aging-aware Timing analysis
Design HDL model
7
Timing report
8
Timing fixes
Asymmetric Aging-Aware Flow
Conventional flow

FIGURE 6: Extended physical design flow in presence of asymmetric transistor aging.

REFERENCES

[1] J. Huang, J. Chai, and S. Cho, “Deep learning in finance and banking: A literature review and classification,” Frontiers of Business Research in China, vol. 14, 2020, https://doi.org/10.1186/s11782-020-00082-6.

[2] M. Macas, C. Wu, and W. Fuertes, “A survey on deep learning for cybersecurity: Progress, challenges, and opportunities,” Computer Networks, vol. 212, 2016, https://doi.org/10.1016/j.comnet.2022.109032.

[3] Component Technical Committee Automotive Electronics Council. Failure mechanism based stress test qualification for integrated circuit. AEC – Q100 – REV-G standard.

[4] F. Gabbay and A. Mendelson, “Asymmetric aging effect on modern microprocessors,” Microelectronics Reliability, vol. 119, pp. 71–81, 2021, https://doi.org/10.1016/j.microrel.2021.114090.

[5] S. Arasu, M. Nourani, F. Cano, J. M. Carulli and V. Reddy, "Asymmetric aging of clock networks in power efficient designs," Fifteenth International Symposium on Quality Electronic Design, Santa Clara, CA, USA, 2014, pp. 484-489, doi: 10.1109/ISQED.2014.6783365.

[6] Gabbay, F., and Mendelson A., "Electromigration-aware Instruction Execution for Modern Microprocessors". Proceedings of the 4th International Conference on Microelectronic Devices and Technologies (MicDAT ’2022), pp. 60-66, 2022.

[7] Gabbay, F., and Mendelson, A., "Electromigration-Aware Architecture for Modern Microprocessors". J. Low Power Electron. Appl. 2023, 13, 7. https://doi.org/10.3390/jlpea13010007

[8] Gabbay, F., and Mendelson, A. , "Electromigration-Aware Memory Hierarchy Architecture". J. Low Power Electron. Appl. 2023, 13, 44. https://doi.org/10.3390/jlpea13030044

[9] S. Bharadwaj, W. Wang, R. Vattikonda, Y. Cao, and S. Vrudhula, “Predictive modeling of the NBTI effect for reliable design,” in Proc. Custom Integrated Circuits Conf., Sep. 2006, pp. 189–192.

[10] A. Ricketts, J. Singh., K. Ramakrishnan, N. Vijaykrishnan, and D. K. Pradhan, “Investigating the impact of NBTI on different power saving cache strategies,” in Proc. DATE, Mar. 2010, pp. 592–597.

[11] S. Ogawa and N. Shiono, Generalized diffusion-reaction model for the low-field charge build up instability at the Si-SiO2 interface, Physical Review, 51(7):4218–4230, Feb. 1995.

[12] M. A. Alam, H. Kufluoglu, D. Varghese, and S. Mahapatra, “A comprehensive model for PMOS NBTI degradation,” Microelectron. Rel., vol. 47, no. 6, pp. 853–862, Jun. 2007. https://doi.org/10.1016/j.microrel.2006.10.012

[13] W. Wang, V. Reddy, A. T. Krishnan, R. Vattikonda, S. Krishnan, and Y. Cao, “Compact modeling and simulation of circuit reliability for 65 nm CMOS technology,” IEEE Trans. Device Mater. Rel., vol. 7, no. 4, pp. 509–517, Dec. 2007.

[14] M. Agarwal, B. C. Paul, Ming Zhang, and S. Mitra, “Circuit failure prediction and its application to transistor aging”, VLSI Test Symposium, pages 277–286, May 2007.

[15] W. Wang, Z. Wei, S. Yang, and Y. Cao, “An efficient method to identify critical gates under circuit aging,” in Proc. Int. Conf. Comput. Aided Des., Nov. 2007, pp. 735–740.

[16] Haider Muhi Abbas, Mark Zwolinski, and Basel Halak. Aging Mitigation Techniques for Microprocessors Using Anti-aging Software. Chapter 3, Ageing of Integrated Circuits - Causes, Effects and Mitigation Techniques, Springer, Cham. ISBN 978-3-030-23781-3.

[17] M. Khairy, Z. Shen, T. M. Aamodt, T. G. Rogers, "Accel-Sim: An Extensible Simulation Framework for Validated GPU Modeling", In proceedings of the 47th IEEE/ACM International Symposium on Computer Architecture (ISCA), May 29 - June 3, 2020.

[18] A.A. Awan, H. Subramoni, and D.K. Panda. An In-depth Performance Characterization of CPU- and GPU-based DNN Training on Modern Architectures. Proceedings of the Machine Learning on HPC Environments, 2017.

[19] Bush, J., TaheriNejad, N., Willegger, E., Wojcik, M., Kessler, M., Blatnik, J., Daktylidis, I., Ferdigg, J. and Haslauer, D., "Nyuzi: An Open Source GPGPU for Graphics, Enhanced with OpenCL Compiler for Calculations." In IEEE Design, Automation & Test in Europe (p. 1). IEEE, 2021.

[20] Gabbay, F., Ramadan, F., and Ganaiem, M., "Clock Tree Design Considerations in The Presence of Asymmetric Transistor Aging" In Proceeding of the 10th Design and Verification Conference (DVCON2023), 2023.

[21] Wang, L. T., Wu, C. W., and Wen, X. (2006). VLSI test principles and architectures: design for testability. Elsevier.

[22] V. D. Agrawal, Kwang-Ting Cheng, D. D. Johnson and T. Sheng Lin, "Designing circuits with partial scan," in IEEE Design and Test of Computers, vol. 5, no. 2, pp. 8-15, April 1988, doi: 10.1109/54.2032.

FIRAS RAMADAN earned his B.Sc in Computer Engineering from the Electrical and Computer Engineering faculty in the Technion - Israel Institue of Technology, Haifa, graduating in 2021. He is currently pursuing his Ph.D. in Electrical and Computer Engineering at the Technion, focusing on reliability issues in VLSI systems and computer architecture under the supervision of Prof. Freddy Gabbay. His research explores Asymmetric Transistor Aging effects and mitigation techniques in machine learning hardware, VLSI circuits, and SoCs. Firas's academic achievements include multiple listings on the Dean's Honorary List, Excellence Awards for teaching assistants, received the Apple's Excellence Award for Academic Achievement and Social Contribution in 2022, and won the Best Research Paper Award at DVCON 2023.

MAJD GANAIEM received his B.Sc. in Computer Electrical Engineering from the Technion, graduating Cum Laude in 2022, with a major in Computer Architecture, Networking, and Intelligent and Autonomous Systems. He is currently pursuing an M.Sc. in Computer Engineering at the Technion's Electrical Engineering faculty in Haifa, Israel, where he is researching software tools and design flows for detecting reliability issues in microprocessors under the supervision of Prof. Freddy Gabbay. In 2020, Majd joined Apple as a student focusing on top-level physical design Place and Route CAD, and he transitioned to a full-time engineer in 2024.

MAAYAN ELLA earned his B.Sc in Computer Engineering from the Electrical and Computer Engineering faculty in the Technion - Israel Institue of Technology, Haifa, graduating in 2023 with honor. He is currently pursuing his M.Sc in Electrical and Computer Engineering at the Technion, focusing on reliability issues in VLSI systems and computer architecture, using ML techniques under the supervision of Prof. Freddy Gabbay. His research focusing on the aging effect on PDK level and in the future his research will be about ML techniques to identify and solve reliability issues.

FREDDY GABBAY (M'19-SM'22) Freddy Gabbay is an associate professor of electrical engineering at the Applied Physics Institute at the Hebrew University of Jerusalem. His main areas of research include VLSI (Very Large-Scale Integration) and chip design, microelectronics, computer architecture, machine learning, and domain-specific accelerators. Gabbay received his B.Sc., M.Sc., and Ph.D. in Electrical Engineering from the Technion – Israel Institute of Technology, Haifa, Israel. In 1998, he worked as a researcher at Intel's Microprocessor Research Lab. In 1999, he joined Mellanox Technologies and held various positions, leading the switch product line architecture and ASIC design. In 2003, he joined Freescale Semiconductor as a senior design manager and led the design of baseband ASIC products. In 2012, he rejoined Mellanox Technologies, where he served as Vice President of Chip Design. He was an associate professor and the Dean of the Engineering Faculty at the Ruppin Academic Center from 2019 to 2024. He also served as a Research Fellow at the Technion, Israel Institute of Technology from 2022 to 2024. Gabbay holds 19 patents and is a senior member of the IEEE.