|  |  |
| --- | --- |
| In-Person: Oral / Poster / The same  Virtual: Pre-recorded video or Zoom  *<please select the type of your presentation>* | Topic: *<Microelectronics reliability and qualification>* |

The Effect of Asymmetric Transistor Aging on GPGPUs

**F. Gabbay 1, F. Ramadan2, M. Ganaiem 2, Ofrie Rosenthal2 and Lior Bashari2**  
1 Engineering Faculty, Ruppin Academic Center, 4025000, Emek Hefer, Israel

2 Electrical and Computer Engineering Faculty, Technion – Israel Institute of Technology, Technion City, 3200000 Haifa, Israel  
E-mail: [{freddyg@ruppin.ac.il](mailto:%7bfreddyg@ruppin.ac.il), [firasramadan@campus.technion.ac.il](mailto:firasramadan@campus.technion.ac.il), [majd.ga@campus.technion.ac.il](mailto:majd.ga@campus.technion.ac.il), [ofrie.r@campus.technion.ac.il](mailto:ofrie.r@campus.technion.ac.il), [li-orbashari@campus.technion.ac.il](mailto:li-orbashari@campus.technion.ac.il)}

**Summary:** General-Purpose Graphics Processing Units (GPGPUs) are specialized hardware devices designed for parallel computing tasks in various domains. Their integration into critical systems like autonomous vehicles, security systems, and medical devices has raised the need for enhanced reliability and resilience, meeting industry and regulatory standards. However, GPGPUs face reliability concerns due to transistor aging caused by bias temperature instability (BTI). This progressive degradation of transistor performance can lead to performance degradation and critical circuit failures, affecting timing violations. This paper investigates the impact of transistor aging on GPGPU execution units, highlighting their vulnerability to BTI. Experimental analysis reveals the significant influence of BTI on computational elements within GPGPUs. To address this issue, we propose an effective mitigation technique that specifically targets the challenges of asymmetric aging in GPGPU execution units, effectively mitigating timing violations.

**Keywords:** GPGPU, BTI, Asymmetric Transistor Aging.

**1. Introduction**

General-Purpose Graphics Processing Units (GPGPUs) are specialized hardware devices designed to perform highly parallel computations ([1, 2]) and are designed to accelerate a wide range of computational tasks, including machine learning, high performance computing, scientific simulations, data analytics, and more. Furthermore, there has been a recent utilization of GPGPUs in critical systems such as autonomous vehicles, security systems and medical devices [3, 4]. These emerging applications establish stringent requirements for the resiliency and reliability of GPGPUs, as mandated by industry and regulatory standards.

In recent decades, VLSI technologies have witnessed remarkable advancements, characterized by several significant trends. First, the continuous development of new process nodes has ensured the consistent miniaturization of transistors to nanometric dimensions, in line with the principles of Moore's law. Second, revolutionary devices and materials have played a pivotal role in driving advancements, resulting in improved performance and reduced energy consumption. However, these advancements have also brought to light the vulnerability of integrated circuits (ICs) to reliability issues, particularly those caused by transistor aging [5,6]. Transistor aging refers to the declining process in a transistor's performance over time, primarily attributed to bias temperature instability (BTI), which will be further described in Section 2. The impact of BTI on IC reliability is significant, causing performance degradation and critical circuit failures due to timing violations. Moreover, asymmetric aging exacerbates timing violations and amplifies concerns regarding reliability, as it results from unevenly distributed degradation.

This paper investigates the effect of transistor aging on GPGPUs execution units. Our experimental analysis includes both functional and physical simulations indicates that computational elements units within GPGPUs can be highly susceptible to BTI. Additionally, our analysis indicates that the various execution units within GPGPU processing elements (PEs) may experience asymmetric aging, resulting in even more serious timing violations. As part of this paper, we present a mitigation technique that effectively addresses concerns regarding asymmetric aging in GPGPU execution units and mitigates timing violations associated with it. Our proposed solution employs a pseudo random bit sequence (PRBS) generator which is activated on idle slots of GPGPU execution units. The PRBS circuitry generates dynamic random patterns of data, which are injected into the GPGPU execution units, thereby avoiding a constant idle state that contributes to asymmetric aging.

The remainder of this paper is structured as follows: Section 2 provides background and discusses prior works. Section 3 investigates the vulnerability of GPGPUs to transistor aging and presents our mitigation approaches and experimental results. Finally, Section 4 concludes our work.

**2. Background and Prior Works**

Transistor aging refers to the deterioration process of transistors in digital circuits ([5, 6]), which is caused by the trapping of charge carriers from the transistor inversion channel at the dielectric insulator of the transistor gate. BTI is recognized as the primary mechanism governing transistor aging. BTI is activated when a constant voltage is applied to the transistor gate, resulting in an elevation of the transistor's threshold voltage. Consequently, this increase in threshold voltage leads to a rise in transistor switching delay and a reduction in transistor speed. Practically, logical gates that remain in a constant idle state of logical 0 are particularly susceptible to aging because p-type transistors are more prone to BTI than n-type transistors. Asymmetric aging, which denotes the uneven distribution of performance degradation among transistors within an IC, can lead to severe timing issues, including setup and hold timing violations.

Common approaches propose incorporating additional timing margins to mitigate the effects of asymmetric aging. However, this approach often necessitates complex simulation analyses and can lead to overdesign [7]. Other studies ([8-10]) have proposed models for predicting aging degradation and have explored various solutions, including clock cycle time reduction, transistor resizing, VDD tuning and power gating. Agrawal et al. ([11]) proposed a method to predict circuit failure by using sensors placed at various locations within the silicon die. Additional research ([12]) has explored techniques to analyze digital circuits and detect the most vulnerable gates affected by NBTI stress. This involves utilizing an aging model with BTI-aware libraries and conducting aging-aware timing analysis. Abbas et al. ([13]) proposed executing anti-aging programs during periods of low processor utilization instead of idle tasks. In [6], an aging-aware microarchitecture was proposed to minimize the effects of asymmetric aging on execution units, register files, and memory hierarchy in microprocessors, while keeping overhead to a minimum.

**3. Analyzing the Impact of Asymmetric Transistor Aging on GPGPU Processing Elements**

In this section, we delve into the impact of transistor aging on GPGPU processing elements (PEs). We conduct an experimental analysis that extracts the aging profile of GPGPUs using functional simulations. Subsequently, we perform a comprehensive timing analysis using aging models derived from the aging profile. Finally, we propose an asymmetric aging avoidance circuitry aimed at mitigating timing violations that arise from asymmetric transistor aging, and we analyze the effectiveness of this scheme.

Our experiments are conducted using the GPGPU simulator [14]. The simulation environment incorporates cycle-level modeling of the RTX 2060 [15] GPGPUs, enabling the execution of computing workloads written in CUDA or OpenCL. To cater to the specific requirements of our experiments, we made modifications to the simulation platform and implemented the necessary mechanisms for accurate measurements. For benchmarking purposes, we utilized simulation benchmarks employed in the gpgpu-sim ispass 2009 paper [16]. These benchmarks encompass a diverse range of applications, including neural networks, graph algorithms, and complex mathematical calculations.

The signal probability (SP) ([6]) is a widely used technique to assess the BTI stress profile on logical elements. It quantifies the probability of a signal having a logical value of 1. Specifically, it is determined by the ratio of the time during which a signal remains in the logical 1 state to the total elapsed time. A smaller SP corresponds to a more pronounced impact of BTI, resulting in performance deterioration and potential failures of integrated circuits as time progresses. Figure 1 depicts the activity levels observed in RTX2060 Streaming Multi-Processors (SMs) for two specific benchmarks: BFS search algorithm and a neural network (inference). Activity is measured as the percentage of time the execution unit remains active relative to the total elapsed time. Our experimental findings indicate that, in the case of the BFS benchmark, approximately 70% of the time the integer execution units within all SMs are idle. On the other hand, for the NN benchmark, more than 85% of the single precision floating-point units remain idle. These observations suggest that the GPGPU processing elements (PEs) may be vulnerable to transistor aging due to their prolonged periods of idle state. The extended duration of maintaining an idle state increases the exposure to aging effects, potentially impacting the reliability and performance of the PEs.

For our case study, we use an integer ALU, and a single precision floating point unit (FPU) taken from OpenCore[[1]](#footnote-1)1. We perform full synthesis, place and route and timing analysis on these modules in 28nm process node. The clock frequency for timing signoff for FPU and ALU is 164MHz and 240MHz respectively. In our timing analysis, we employ aging-aware library models as described in [6]. These models account for the impact of BTI by derating cell delays using NBTI degradation factors derived from the SP values extracted from the functional simulations illustrated in Figure 1. The results of our timing analysis, presented in Table 1, reveal that the presence of BTI can lead to significant timing violations in both GPGPU ALUs and FPUs. It can be observed that when aging is not considered, as in the case of a fresh design, there are no timing violations. However, when aging effects are considered, both the FPU and ALU exhibit setup and hold timing violations. It is worth noticing that although setup violations can be alleviated by reducing the clock frequency, hold violations do not have an effective mitigation strategy at present.



(a)



(b)

**Fig. 1.** The Activity measured in RTX2060 (a) integer execution unit and (b) Floating point unit for BFS search algorithm and NN (Neural Network) benchmarks respectively.

**Table 1.** Worst Negative Slack (WNS) and the Number of Violated Timing Paths (NVP) for FPU and ALUs for fresh design, aged design, and a design with asymmetric aging avoidance.

|  |  |  |  |
| --- | --- | --- | --- |
| Setup WNS [ps] /NVP | | | |
|  | Fresh | Aged | Asymmetric Aging Avoidance |
| FPU | 0/0 | -115/469 | 0/0 |
| ALU | 0/0 | -23/1 | 0/0 |
| Hold WNS [ps] /NVP | | | |
|  | Fresh | Aged | Asymmetric Aging Avoidance |
| FPU | +4.5/0 | -2/7 | 0/0 |
| ALU | +13/0 | -1/10 | 0/0 |

To address the impact of BTI on GPGPU execution units, we propose the adoption of a pseudorandom sequence bit (PRBS) generator, activated by a low-frequency clock as illustrated in Figure 2. This approach is inspired by the technique suggested in [6] for general-purpose microprocessors. As illustrated in Figure 2, the PRBS data patterns are multiplexed with the functional data path inputs through a designated multiplexer. By employing a PRBS generator, pseudorandom patterns are generated and fed into the GPGPU FPU and ALU, effectively mitigating extended periods of constant stress. The PRBS circuitry is timed using a slow-frequency clock during idle time slots of the FPU and ALU. When the PRBS circuitry is enabled, the input multiplexer selects the PRBS data, which is then injected into the data path of the execution units. The clock frequency for this PRBS generator can be set to a few megahertz (MHz) or even lower to minimize any potential dynamic power overhead. This technique can provide a practical solution for reducing the vulnerability of GPGPU execution units to BTI, enhancing their resilience and prolonging their operational lifetime. Our timing analysis for the FPU and ALU employing the PRBS asymmetric aging avoidance circuitry, demonstrates the successful elimination of all timing violations as shown in Table 1.



**Fig. 2.** Asymmetric aging avoidance circuitry based on PRBS generator.

**4. Conclusions**

The emerging deployment of GPGPUs in mission-critical systems establishes stringent requirements for the resilience and reliability of GPGPUs. Nonetheless, the advent of new advanced process nodes has also illuminated the susceptibility of ICs to reliability concerns, with a specific focus on those stemming from transistor aging.

In this paper, we examined the impact of asymmetric transistor aging on GPGPU execution units. Our experimental analysis has shown that execution units within GPGPUs can be highly susceptible to BTI due to prolonged periods of idle stress. As a case study, we investigated the NVIDIA RTX 2060 GPGPU using BFS and NN benchmarks, indicating that execution units such as the integer execution unit and the FPU can remain idle for around 70% and 85% of the total time, respectively. These observations suggest that such execution units are highly susceptible to asymmetric transistor aging.

These concerns were further demonstrated through a detailed timing analysis that combines an aging library model with the aging profile derived from functional simulations. The observed timing violations suggest that GPGPU computational elements can experience asymmetric transistor aging, resulting in setup and hold timing violations. As part of this study, we have also introduced an asymmetric aging avoidance circuitry based on a PRBS generator to mitigate asymmetric transistor aging in GPGPU execution units. Our detailed timing analysis indicates that the asymmetric aging avoidance circuitry has been successful in eliminating the timing violations caused by asymmetric transistor aging.

Moving forward, further research is warranted in the domain of asymmetric transistor aging in GPGPUs and other computational elements. Firstly, exploring adaptive techniques to dynamically adjust clock frequencies, clock latencies or resource allocation could offer promising avenues for mitigating aging-induced timing violations. Secondly, investigating novel design methodologies that integrate fine-grained aging-aware optimizations into the microarchitecture could provide more comprehensive solutions. Additionally, extending the study to encompass a wider array of benchmarks and real-world workloads will be crucial to establishing the robustness and generalizability of the proposed solution.

**References**

[1]. Kirk, D., and W. Hwu. "GPGPU: General-Purpose Computation on Graphics Hardware." ACM SIGGRAPH, vol. 25, no. 3, 2006, pp. 657-666.

[2] Harris, M., et al. "GPGPUs for General-Purpose Scientific Computing: A Survey." Journal of Supercomputing, vol. 73, no. 1, 2017, pp. 3-50.A. Author, Book title, Editor, *Publisher*, 1990.

[3] Campmany, V., Silva, S., Espinosa, A., Moure, J. C., Vázquez, D., & López, A. M. (2016). GPU-based pedestrian detection for autonomous driving. Procedia Computer Science, 80, 2377-2381.

[4] Yang, M. (2018, July). Avoiding pitfalls when using NVIDIA GPUs for real-time tasks in autonomous systems. In Proceedings of the 30th Euromicro Conference on Real-Time Systems.

[5]. 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>

[6] F. Gabbay, A. Mendelson, Asymmetric aging effect on modern microprocessors, Microelectronics Reliability, Volume 119, 2021, 14090, SSN 0026-2714, <https://doi.org/10.1016/j.microrel.2021.114090>.

[7] 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.

[8] 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>

[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] 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.

[11] 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.

[12] 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.

[13] H. M. Abbas, M. Zwolinski, and B. 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.

[14] 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.

[15] https://www.nvidia.com/content/dam/en-zz/Solutions/design-visualization/technologies/turing-architecture/NVIDIA-Turing-Architecture-Whitepaper.pdf

[16] P. Harish and P. J. Narayanan. Accelerating Large Graph Algorithms on the GPU Using CUDA. In HiPC, pages 197–208, 2007.

1. 1 [www.opencores.org](http://www.opencores.org) [↑](#footnote-ref-1)