In-field Detection of Small Delay Defects and Runtime Degradation using On-Chip Sensors

S. Maryam Ghasemi, Sergej Meschkov, Jonas Krautter, Dennis R. E. Gnad, Mehdi B. Tahoori
Institute of Computer Engineering, Karlsruhe Institute of Technology (KIT), Karlsruhe, Germany
{seyedehmaryam.ghasemi, sergej.meschkov, jonas.krautter, dennis.gnad, mehdi.tahoori}@kit.edu

***Abstract*—The increasing safety requirements for modern complex systems mandate Silicon Lifecycle Management (SLM) using various sensors for in-field test. In this work, we evaluate so-called Path Transient Monitors (PTMs), which are based on delay lines, to detect path delay increase caused by manufacturing defects or runtime degradation. These sensors are integrated into a RISC-V SoC on an FPGA, allowing software-controlled measurements and calibration. Additionally, we introduce means to emulate delay defects and degradations by injecting additional delay elements into a custom add instruction. Furthermore, by using power wasters, we provoke runtime voltage variations. Our evaluation in different temperatures shows the dependencies between different sources of delay variations and how the sensors can help in better detection of delay defects.**

I. Introduction

The complexity of the most-advanced Systems on Chip (SoCs) and High Performance Computing (HPC) systems at both the hardware and software side steadily increases. When coupled with nanoscale effects in advanced technology nodes, detecting all manufacturing defects and runtime failures becomes extremely challenging.

Silicon Lifecycle Management (SLM) is a comprehensive approach that addresses the full lifecycle of silicon-based hardware components. SLM is geared towards optimizing device performance, reliability, and longevity by implementing effective strategies across stages like design, manufacturing, deployment, operation, maintenance, and end-of-life management. Unlike traditional approaches that focus on individual lifecycle stages, SLM takes a holistic view, considering the interdependencies and opportunities for optimization across all stages.

Previous works have introduced means for measuring different indicators of chip health degradation, the so-called chip telemetry [1], [2], [3]. In this manner, empowered by in-situ collection of measurements from on-chip sensors and monitors, the system will be primed for investigating chip health and performance monitoring [4]. Moreover, sensors have been integrated into a system-level approach, to perform parametric testing in the field from software in a RISC-V SoC [2], [3].

In this paper, we investigate the relationship between runtime effects, such as temperature and voltage fluctuations, as well as sensor readouts in detecting delay defects and degradations at the software level. To do so, we introduce means to emulate delay increase in a custom adder instruction from software. To also account for runtime load variation, we add controllable power wasters which emulate the effects of different workloads on the supply voltage. Our proposed architecture offers a simple way to evaluate both the dependencies between different sources of degradation.

II. Proposed Approach

The overall RISC-V-based system architecture with the added delay monitoring on-chip telemetry, custom adder instruction, and configurable power wasters, is shown in Fig. 1. For delay monitoring we employ a Path Transient Monitor (PTM) as presented in [2], [3], which is based on a Time-to-Digital Converter (TDC), implemented as a delay line, to capture signal transitions at fine granularity. The PTM can be calibrated, triggered, and read through custom RISC-V instructions.

In order to investigate the effectiveness of the proposed on-chip sensors in detecting small delay defects and runtime degradation, we insert controllable delay elements in the specific critical paths of the circuit under test. The length of the added delay chain can be configured at the software level, allowing to perform various experiments on the relationship between the sensor measurements, and functional failures.

In the case investigated in this paper, the delay elements are injected into the carry chain of a Carry Ripple Adder (CRA), that is supported via a custom instruction defined for this module. There is an additional custom instruction for controlling the amount of delay.

In the implementation on the FPGA, the chains of LUT elements were used to model the delay faults added in the path. The delay elements were added in three sections, in the beginning, middle, and ending part of the CRA carry chain. By increasing the amount of injected delay, the more delayed output on the combinational path endpoint is expected. This will be interpreted as a delay fault on the path. The monitoring schemes will be evaluated at the same time with delay fault emulation in this system.

Finally, we add an array of power wasters based on AES encryption modules as used in [5], which can also be activated through custom instructions. The activated power wasters cause supply voltage fluctuations, emulating the effect of different workloads on the entire system.

The entire approach is implemented on an Field Programmable Gate Array (FPGA) as a full RISC-V SoC that is generated with the help of LiteX [6] and deployed on an AMD/Xilinx Kintex 7 KC705 FPGA Evaluation Kit. The

979-8-3503-4859-0/DATE24/© 2024 EDAA

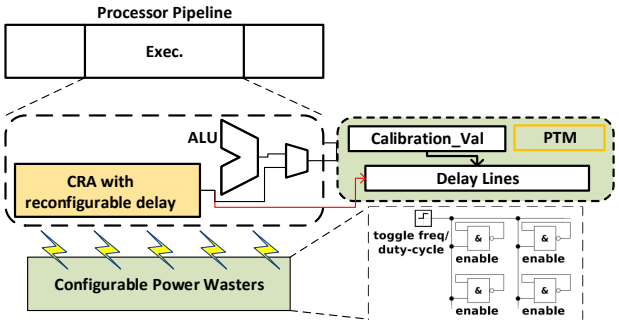


Fig. 1: Overall sensor-augmented architecture.

evaluation board is enclosed in a electromagnetically shielded climate chamber (Weisstechnik LabEvent T/210/40/EMC), to additionally evaluate the effect of different temperatures.

III. Results

To evaluate the effect of injected delays into the custom adder instruction onto the collected PTM measurements, we exemplary show a specific transition in Fig. 2. This figure shows the transition on the combinational endpoint, here MSB of CRA, captured by PTM, and all belong to a specific transition in the presence of different amount of injected delays on the path. The green lines show the cases where the injected delay caused no actual failure in the computation, whereas the red lines are captured when there is incorrect output from the custom CRA instruction. By increasing the number of injected delay, the same transition is observed just with a bit of shift to the right, meaning the transition happens later. If the injected delay exceeds a certain amount, the ALU output register fails to capture the correct result at the next rising clock edge and a timing fault occurs. The exact amount of delay to provoke failures depends on the input data.

In Fig. 3 we show the amount of functional faults in the custom CRA instruction w.r.t. different sources of runtime stress. First, we observe in all cases that additional runtime stress, no matter if thermal or voltage related, always leads to an earlier manifestation of functional faults from the injected delay, i.e. fewer delay elements need to be injected in order to cause incorrect outputs. From comparing Fig. 3a with Fig. 3b, we conclude that the effect of power wasters has a slightly bigger impact than that of temperature in our range of experiments. However, we find that a combination of both temperature and voltage fluctuations, is most effective to detect the emulated delay degradation as a functional failure, as can be seen in Fig. 3c.

IV. Conclusion

In this paper we presented how software-controlled high-resolution delay sensors can be employed to detect path delay degradation in the field. Moreover, we evaluated the effects of different sources of runtime stress, namely temperature and voltage fluctuations, on delay-based functional failures.

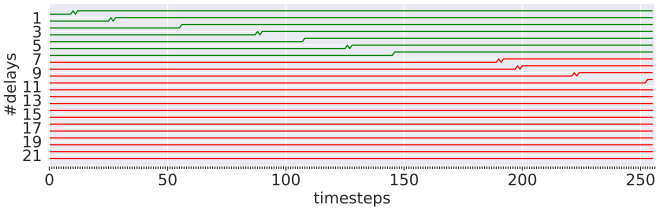
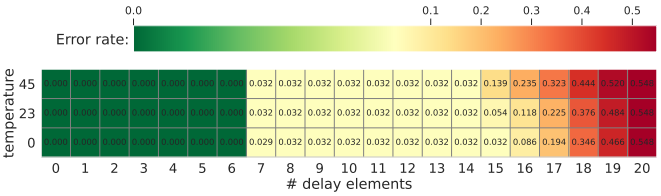
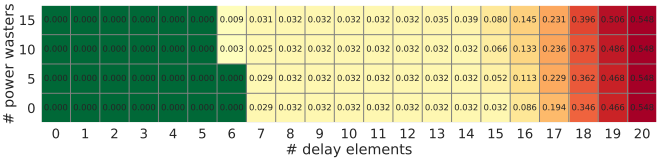


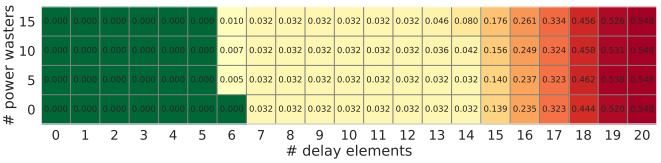
Fig. 2: PTM sensor output for one transition with different number of injected delays in the path.



(a) No power wasters, effect of temperature



(b) Temperature 0°C, effect of power wasters



(c) Temperature 45°C, effect of power wasters

Fig. 3: Amount of functional failures in the custom CRA for different amounts of injected delay and different sources of runtime variations.

The platform is implemented on a RISC-V FPGA SoC, where we emulate the delay degradation through artificially injected delay elements controlled by software with a custom instruction. Our evaluation shows, that the increased delay can be observed in the transitions captured by the PTM. Moreover, we find that a combination of increased temperature and voltage fluctuations, provoked by power wasters, can improve the detection of delay defects or degradation.

References

[1] H. A. Balef et al., "Chip health tracking using dynamic in-situ delay monitoring," in Design, Automation & Test in Europe Conference & Exhibition (DATE), 2019, pp. 304–307.

[2] S. M. Ghasemi et al., "Enabling in-field parametric testing for risc-v cores," in 2023 IEEE International Test Conference (ITC), 2023.

[3] ——, "SLM ISA and hardware extensions for RISC-V processors," in 2023 IEEE 29th International Symposium on On-Line Testing and Robust System Design (IOLTS), 2023, pp. 1–5.

[4] A. E. Landman et al., "Applying universal chip telemetry to detect latent defects and aging in advanced electronics," in 2022 IEEE International Reliability Physics Symposium (IRPS), 2022, pp. P38–1–P38–4.

[5] J. Krautter et al., "Remote and stealthy fault attacks on virtualized FPGAs," in 2021 Design, Automation & Test in Europe Conference & Exhibition (DATE), 2021, pp. 1632–1637.

[6] Enjoy-Digital & LiteX developers. (2023) Build your hardware, easily! [Online]. Available: https://github.com/enjoy-digital/litex