

Soft Error Effects on Arm Microprocessors: Early Estimations Versus Chip Measurements

Pablo R. Bodmann^{ID}, Federal University of Rio Grande do Sul

George Papadimitriou^{ID}, National and Kapodistrian University of Athens

Rubens L. Rech Junior^{ID}, SAP

Dimitris Gizopoulos^{ID}, National and Kapodistrian University of Athens

Paolo Rech^{ID}, University of Trento

This installment of Computer's series highlighting the work published in IEEE Computer Society journals comes from IEEE Transactions on Computers.

Early microprocessor soft error predictions are essential to guide effective protection techniques at design time. We show, for the first time, how (presilicon) microarchitectural fault injection

error rate estimation can be very close to physical (postsilicon) neutron beam experiments. The prediction accuracy holds even when the microprocessor is integrated in a system-on-chip and an operating system is deployed.

The correctness of microprocessor operation when executing workloads in any domain, from Internet of Things (IoT), through edge computing all the way to large-scale cloud datacenters and supercomputers, can be jeopardized by multiple factors: transient faults due to radiation, permanent faults due to latent defects or aging, timing faults due to chip variability, and design bugs.

During the operational lifetime of a microprocessor, the prevailing factor that determines the failure rate of microprocessor chips (number of times it will fail to deliver the expected computation) are transient faults, also known as *soft errors*. These errors are extremely challenging to detect and mitigate since the transient corruption cannot be distinguished from a correct value processing.



The early (predesign) estimation of microprocessor soft error rates or their actual (in-field) measurement is a cumbersome process which depends on the radiation flux and energy (location of the system), the chip implementation technology (probability for a transistor to be disturbed), the microprocessor instruction set architecture and microarchitecture (probability for the fault to modify data or computation), as well as the executed workloads and systems software stack (probability for the error to propagate to the output). Understanding the reliability of a microprocessor is of paramount importance and is guiding both industry and research efforts.

The error rate of physically implemented chips and systems (postsilicon) are evaluated either with excessively long (years) collection of failure data from large fleets of microprocessors (datacenters or supercomputers) or with beam testing, where chips are exposed to accelerated flux of neutrons (or other particles) that in a few hours mimic the impact of million-years natural radiation. While both approaches are highly accurate, they can be performed only when silicon is available. If error rates are found to be excessive, the redesign cost is extremely high. An alternative that has been explored lately is to estimate the device and system error rate at design time (presilicon) through fault injection at different abstraction levels: software or architecture levels (fast but wrong due to missing hardware information), gate level (hardware-accurate but with unaffordable simulation times and no system software), and microarchitecture level. Microarchitecture is the only level that brings the best balance between simulation throughput (so that an entire computing system hardware and software stack can be analyzed) and accuracy of the modeled

microprocessor and the phenomena that affect it.

Although extensive studies using fault injection exist in the literature, a direct comparison of the reported soft error rates between presilicon microarchitecture level fault injection

software domain, with the OS required to take full advantage of the available resources.

In our article, "Soft Error Effects on Arm Microprocessors: Early Estimations versus Chip Measurements," *IEEE Transactions on Computers*, vol. 71,

A direct comparison of the reported soft error rates between presilicon microarchitecture level fault injection and postsilicon accelerated neutron beaming has never been conducted.

and postsilicon accelerated neutron beaming has never been conducted. Thus, it has been unclear to which extent microarchitecture level fault injection can provide an accurate error rate estimation at early stages to guide hardware and software design decisions and whether an early estimation really matches the physical measurements from beam experiments. The importance and challenges associated with a timely, yet realistic, evaluation grow with the increase of complexity in both the hardware domain, with the integration of different types of cores in a system on chip (SoC), and the

no. 10, pp. 2358–2369, 1 2022 October, we combine and analyze data gathered with extensive beam experiments (on the final physical CPU hardware) and microarchitectural fault injections (on early microarchitectural CPU models in the state-of-the-art gem5 simulator). We target both a standalone Arm Cortex-A5 CPU and a Cortex-A9 CPU integrated into an SoC and evaluate their reliability in both bare-metal and Linux-based configurations. Our objective, as depicted in Figure 1, is to demonstrate that presilicon evaluation can provide an accurate estimation of the final system

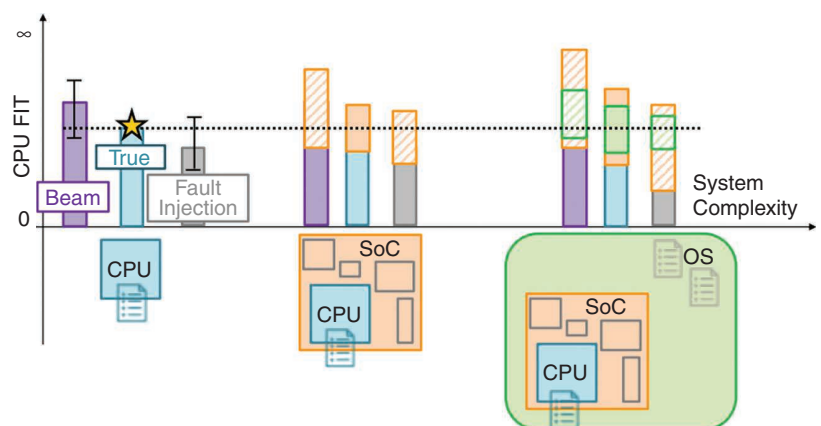


FIGURE 1. An abstract view of the failure-in-time (FIT) rates evaluation uncertainties and possible impact of SoC integration and operating system on the CPU error rate.

error rate and to understand if—and how—the integration of the micro-processor in a SoC and the use of an

data corruptions (SDCs) rate which is solely attributed to the CPU core. Our reliability analysis demonstrates

guide effective soft error protection decisions at the hardware or the software level at very early design stages of the computing system, significantly reducing the implementation costs and increasing their efficiency. **■**

We find that both the SoC integration and the presence of the OS increase the system detected unrecoverable error rate (for different reasons) but do not significantly impact the silent data corruptions rate.

operating system impact the reliability. The results of the complete evaluation are summarized in Figure 2.

Combining and comparing experimental data that covers more than 18 million years of device time with the result of more than 176,000 injections, we find that both the SoC integration and the presence of the OS increase the system detected unrecoverable error (DUE) rate (for different reasons) but do not significantly impact the silent

that even considering SoC integration and OS inclusion, early, presilicon microarchitecture-level fault injection delivers accurate SDC rates estimations and lower bounds for the DUE rates.

Our evaluation also demystifies the accuracy gap between presilicon and postsilicon reliability evaluations, providing fundamental indications on how to ensure an early, yet realistic, error rate estimation at design time. The investigation we performed can

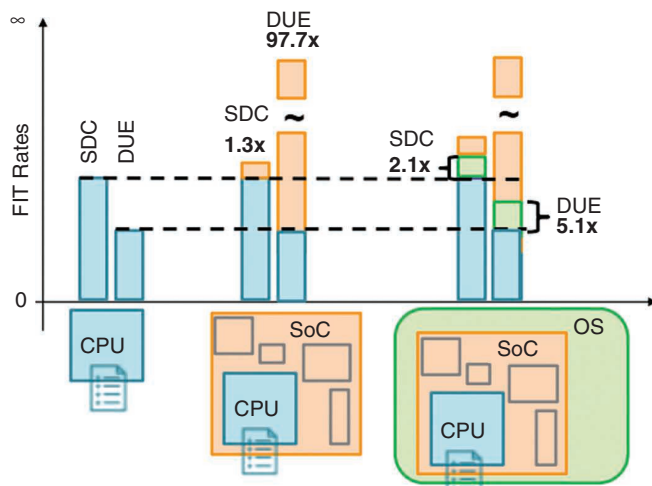


FIGURE 2. The overall impact of SoC integration and OS on the CPU error rate as resulting from our data (vertical axis is in log scale). On average, the core integration barely increases the SDC rates (about 1.3x) but significantly increases the DUE rates (97.7x). The OS has a smaller impact on SDCs (about 2.1x) and increases DUE by about 5.1x.

PABLO R. BODMANN is a Ph.D. student at the Universidade Federal do Rio Grande do Sul, 91509-900, Porto Alegre, Brazil. Contact him at prjbodmann@inf.ufrgs.br.

GEORGE PAPADIMITRIOU is a postdoctoral researcher in the Department of Informatics and Telecommunications, National and Kapodistrian University of Athens, GR 157 84, Athens, Greece. He is a Member of IEEE. Contact him at georgepap@di.uoa.gr.

RUBENS L. RECH JUNIOR is a software developer with SAP, 90480-000, Porto Alegre, Brazil. Contact him at rubensrechjr@gmail.com.

DIMITRIS GIZOPOULOS is a professor of computer architecture in the Department of Informatics and Telecommunications, National and Kapodistrian University of Athens, GR 157 84, Athens, Greece. He is a Fellow of IEEE. Contact him at dgizop@di.uoa.gr.

PAOLO RECH is an associate professor at the Università di Trento, Italy, and an associate professor at the Universidade Federal do Rio Grande do Sul, 90480-000, Porto Alegre, Brazil. He is a Senior Member of IEEE. Contact him at paolo.rech@unitn.it.