The microprocessor chip design industry has enjoyed a remarkably good record in terms of user reliability since the dawn of the digital computer revolution. Arguably, improved technological integration over the years only added to the hardware reliability of microprocessor-based systems as the industry crammed more and more of the chipset constituting the “CPU board” into the central electronics complex embodied by the microprocessor chip itself.

Semiconductor electronics was a significant advance over vacuum tubes to begin with and progressive advances in semiconductor technology (first bipolar and then CMOS) just made things progressively better. Of course, as chips became more and more complex in their design and functionality, verification became a major problem and occasionally that did cause a stir. (Recall the infamous Pentium division hardware bug in the early to mid 1990s, for example?) And, design bugs at the chip- (and more frequently at the system-) level continue to haunt vendors; often, they must resort to remedial action through firmware patches or software workarounds to take care of rare corner case bugs that escaped premarket testing and verification. But, the industry took the basic lifetime reliability of the microprocessor virtually for granted all these years, which is to say that the robustness of the basic transistor devices (computing and storage) and interconnects was known to be high enough.

Transient (or soft) errors caused by single-event upsets arising from high-energy particle strikes have been a source of concern for a while, especially in SRAM arrays; but most low-end system designs achieved adequate protection through relatively low-overhead parity and/or error-correcting code (ECC) protection. High-end server and mainframe systems have often invested in higher levels of protection (that is, dual-core, lock-step processing, sometimes with on-chip ECC-protected hardware recovery support) in their processor chips to provide the levels of hardware reliability and systems availability required in those markets.

The problems of power dissipation and process or parametric variability have added new dimensions to the problem of protecting the hardware from transient and hard failure mechanisms in the deep-submicron era. Despite Moore’s law, silicon real estate is at a premium, because of power—especially of the passive (or leakage) variety. That’s why area- and power-efficient design is the key mantra of the modern architect and designer. So, affording extra transistors and area for error detection, recovery, redundant computing, or spare units is not something that is easy any more.

Yet, with CMOS scaling, as on-chip temperatures rise and device geometries shrink, both hard (or permanent) and soft (or transient) failure modes are on the rise. In addition, because of high levels of temperature-sensitive leakage power, traditional means of weeding out weak chips through a burn-in process are difficult to use in the newer technologies. So, while designers are under pressure to reduce overhead for reliability support, the chips taking advantage of newer technology are becoming fundamentally less reliable. The process and parametric variability problem only adds to the challenge.

Herein lies a key challenge for current and future microprocessor and system designers: In addition to innovative new power and area-efficient mechanisms to provide on-chip reliability support, system designers must increasingly look to solutions in the software stack to recover from detected errors and failures and thereby maintain expected levels of system availability. And all this must happen without compromising market-driven expectations of robust system performance. (This is a tall order, especially because systems software is complex, poorly tuned, and buggy enough already in many environments!) For these reasons, this theme issue covers the topic of reliability-aware microarchitectures. The guest editors’ introduction (by Sarita Adve and Pia Sanda) provides an excellent roadmap, guiding readers through particular problems and how each paper addresses one or more of those. I am grateful to these guest editors for doing such an excellent job of review and selection to bring us a very valuable set of timely articles on this emerging area of interest to the microarchitecture R&D community.

Pradip Bose
Editor-in-Chief
IEEE Micro