A Reliability Odometer - Lemon Check Your Processor! *

Jayanth Srinivasan†, Sarita V. Adve†, Pradip Bose‡, Jude A. Rivers‡
†Department of Computer Science, University of Illinois, Urbana-Champaign
‡IBM T.J. Watson Research Center, Yorktown Heights, NY
{srinivsn,sadve@cs.uiuc.edu},{pbose,jarivers@us.ibm.com}

The Problem

Guaranteeing long processor lifetimes by limiting the on-set of wear-out or hard failures is a critical requirement for all processor manufacturers [3, 4]. In addition, *lifetime predictability* is also crucial from the point of view of service replacements, avoiding down-time, etc. This is particularly true for vendors of high-end servers where predictable reliability is an implicit requirement. However, processors running different applications exhibit a large range of lifetimes based on application characteristics like power consumption and temperature. More importantly, this range is rapidly increasing as we scale to deep sub-micron technologies [4]. Hence, due to the variety of applications run on a typical processor, and the inherently unpredictable nature of these applications, guaranteeing processor lifetimes is becoming increasingly difficult, if not impossible. One current approach to handling this unpredictability is to over-design systems from a reliability standpoint. However, this is expensive, and curtails performance and yield, making design more expensive and eating into profits.

There is a clear need to dynamically track processor wear-out and predict hard error outages. In high-end server class processors, this can be used to preempt service failures minimizing expensive periods of down-time. Even in lower-end systems, tracking reliability statistics dynamically can be a useful feature. For example, in large processor farms, by monitoring wear-out, we can better balance lifetimes across many processors by intelligently distributing applications. Additionally, hardware adaptation techniques like Dynamic Reliability Management (DRM) [3] have been proposed to allow the microarchitecture to trade-off performance and reliability on the fly. Such techniques also require dynamic reliability monitoring. Finally, there is a large market for used hardware, particularly among network processors and high-end server processors [1, 2]. Knowledge of pre-existing wear-out and usage information for such hardware would be immensely useful in pricing and warranty certification.

The Solution

To address these problems, we propose a **reliability odometer**. Much like the odometer found in automobiles, the reliability odometer would dynamically monitor processor usage and store wear-out information. Every event in the processor's history would increment the odometer appropriately. The odometer can be implemented either in terms of Failures-in-Time (FITS) or MTTF.

In [3], we describe RAMP, a microarchitectural model for processor reliability measurement, that models the impact of application characteristics on different processor failure mechanisms (electromigration, stress migration, gate-oxide breakdown, and negative-bias temperature inversion). RAMP is highly modular in design - as a result, other failure mechanisms, if important to the hardware being monitored, can easily be included. Additionally, RAMP also models application-independent wear-out events like thermal cycles which arise from the processor being turned on and off.

RAMP can be directly applied in the use of the reliability odometer. Most of the hardware required by RAMP (or the reliability odometer), like temperature sensors and activity counters, already exist in current processors. The only additional hardware requirements would be some simple combinatorial logic and a little storage.

Even if precise failure mechanism models are not available, a simpler version of the reliability odometer can be implemented using thermal sensors. Most wear-out failure mechanisms follow an exponential dependence on processor temperature based on the Arrhenius relationship [4]. A simple counter which incorporates this exponential effect of temperature would give a reasonable estimate of prior processor usage.

Such a reliability odometer would not only be of use to lemon check used hardware, but also in adaptive control which can budget reliability based on applications, performance requirements, and cost considerations.

References

[1] Datalease Systems. http://www.datalease.com/.

[2] Network Hardware Resale. http://www.networkhardware.com/.

[3] J. Srinivasan et al. The Case for Lifetime Reliability-Aware Microprocessors. In *Proc. of the 31st Annual Intl. Symp. on Comp. Architecture*, June 2004.

[4] J. Srinivasan et al. The Impact of Technology Scaling on Lifetime Reliability. In *Proceedings of the 2004 International Conference on Dependable Systems and Networks*, June 2004.

[*] This work was performed while Jayanth Srinivasan was a summer intern at IBM T.J. Watson Research Center.