Summary of Second Cross-Layer Reliability Meeting (July 8-9, 2009, Los Alamos, NM)

Context

The first workshop reinforced the need for new reliability approaches and solutions as we scaled to smaller device feature sizes and larger system sizes. We further identified a host of promising directions to improve reliability above the device level while containing or reducing reliability overhead. It was clear that we had a problem and research vectors to address them.

While there was broad agreement that problems were coming or were already here, there was less agreement on the exact nature of the problem, how it would impact our systems, and how fast it was coming. In the meeting and the aftermath of the meeting, it became clear that we lacked a common roadmap and agreement on the key metrics and composibility. Further, it was clear that we needed to more carefully discuss the complex multi-dimensional design space (area-delay-energy-reliability-thermal...) in which we operated as well as the various environments (sea-level, atmosphere, space) and system sizes (hand helds to supercomputers). It was not yet clear if the reliability problems encountered in harsh environments like space, the reliability problems of large-scale systems like supercomputers, and the reliability challenges associated with voltage scaling to reduce power in consumer electronics had common underlying causes that would benefit from the same kinds of solutions. Alternately, it might turn out that these different domains demand distinct solutions because they operate in different pieces of this complex state with different primary limitations.

Workshop Target

From this experience, it was clear we needed to divide the group into narrower constituencies to get a better characterization of the problems in each domain. The hope was, and is, that with a clear view of the problems in each domain, we could then look for underlying commonality as well as identifying necessary points of divergence. This led us to form a set of constituency groups organized around key systems types, including:

- consumer electronics (e.g. PCs, cell phones, mp3 payers, etc.)
- space/avionics (e.g. satellites, airplanes, space crafts)
- large-scale systems (e.g. supercomputers, large clusters, data centers)
- life-critical systems (e.g. medical, automotive)
- infrastucture (e.g. networking, telephony, power grid, building control)

The idea was for each of these groups to identify their own key challenge problems without getting bogged down, at this point, with the needs of the other areas. We could then compare the challenges and identify common problems and opportunities for common solutions if they arise. These constituency groups were to help quantify the challenge and help identify the impact of not solving these problems (or the beneficial impact of solving).

We succeeded in forming three of these challenge groups (space/avionics, large-scale systems, and consumer electronics) between the meetings, and they were able to meet and develop their thoughts to varying degrees before the meeting. The meeting served as an opportunity to compare notes among these groups, clarify the target for the groups, and meet face-to-face to further the effort. All the groups presented works in progress and will continue working toward crisper, quantified, and prioritized description of their challenges.

We failed to find critical mass in life-critical systems and infrastructure. Forming these groups continues to be an ongoing effort and the workshop helped identify more targets to contact. The hope is that these groups can be formed soon, work during August and September, and be ready to brief out results at the final workshop in October.

In addition to the constituency groups, we also created a group on metrics to help identify the proper way to define the challenge problems and a group on roadmapping to provide a common reference for how bad silicon devices could become and how fast. The metrics group expanded its scope to thinking about composition of metrics and which metrics would enable and characterize cross-layer cooperation and optimization. These groups also presented their status and plans at the workshop and compared notes with the constiuency challenge groups.

As a result, this workshop was very much a work-in-progress status report meeting for all involved, giving everyone glimpses of the key issues for the study groups and possible study outcomes and helping keep the groups converging toward the overall goals of the visioning effort. The workshop was roughly organized as:

- set of brief ins
- breakout working time
- set of brief outs
- group discussion on next steps

Presentations and Status

Workshop Opener

Workshop opened with a brief on the goal and status of the study by the organizers (slides). This served both as a introduction and reminder of the goals of the study for the participants and as an opportunity to try out a refined version of the study vision. The briefing highlighted some of the areas of need (e.g. quantified challenge problems) that the focus groups would be providing in order to provide a complete and solid story. This was also a chance to review the timeline for the study itself for all the participants so everyone could see how it should come together.

Metrics

The metrics group has formed a healthy team that was represented by a smaller subset at the meeting itself. Through pre-meeting telectonferences they had scoped their effort and identified subgroups within their effort. (slides)

- There is evidence, even on the commercial side, that there are reliability challenges to address today (not just for future technologies).
- The need to build reliable systems out of components from a wide variety of suppliers adds to the challenge---especially lacking common languages and standards for quantifying the reliability of components.
- Noted need and desire for in-system reliability adjustment (differential reliability of tasks or systems).
- Noted old models of simply summing hardware error rates (e.g. FIT rates) are not adequate---this ignores the impact of higher level mitigation techniques, one of the things this study advocates.
- Separately, it was noted that MIL-STD-217B was the old answer to calculating reliability but was not adequate or appropriate for dealing with these kinds of systems or future technology. One capture of this, in part, shows up in: *State of the Art Semiconductor Devices in Future Aerospace Systems* paper from Joint Council on Aircraft Aging, 2007.
- Presented a sketch for a model for composition of metrics in order to calculate system-level metrics for components.
- Noted why it was important to address hard errors and aging as well as soft errors. Discussion agreed that we needed to, at least, be able to divide errors and error rates into these three classes. Questions remained as to whether further subdivision was necessary and productive.
- Reminded us of the importance of using statistics on the rate of detected but recoverable errors as an indication of the safety of the system (e.g. does the system retain the safety margins we expect? is the environment or system changing from our design expectations?).
- Reminded us, and showed evidence that, FIT rates are a strong function of age, showing that many old rules-of-thumbs about aging and wear-out are not accurate for many components today.

Roadmap

Roadmapping group was just being constituted at the time of the meeting and has since begun to meet via regular teleconferences. The workshop provided a jumping-off point to define and refine the goals for the roadmap effort. (slides)

- Roadmap group is targeting inclusion of some initial results on variability induced reliability trends for the ITRS; this demands completion of this piece by late August.
- Roadmap would ideally include variability, aging, and soft-error effects and separate internal effects (e.g. thermal and shot noise) from external effects (e.g. ionizing particles).
- Discussed a study on variability and reliability effects previously done for IBM that roadmap group would extend using public PTM models and ITRS technology numbers. Study looked at how many sigma various circuit elements (SRAM bits, buffers, latches) could tolerate before various notions of failure (vdd and slowdown limits, complete failure), showing all of these decrease as technology continues to scale.

Space/Avionics

The space and avionics group formed earliest, was able to meet in person at other conferences, and had the most in-depth discussions prior to the workshop. (brief-in slides brief-out slides)

- Space as canary: one idea that arose previously and was articulated here is that space systems, due to their harsh environments, sees many of the problems well in advance of other groups. So, while they may have extreme requirements, their problems are often an early warning for challenges we will see in other situations. A potentially oversimplified hierarchy of where things shows up is: space-->avionics-->large-scale systems-->consumer electronics
- Lifetime of space systems is long. Many satellites are designed for 12 year lifetimes. The aforementioned *State of the Art Semiconductor Devices in Future Aerospace Systems* notes the trend in commodity silicon to reduced lifetimes that will be problematic for space and aircraft systems.
- Space agrees that characterization of the problem must be more than just soft-error FIT rates; space talks about tens--hundreds of error categories, but agree many of these categories could be unified.
- Is Total Ionizing Dose (TID) similar to other wear mechanisms that are being seen on the ground just different time constants? Does that help us unify phenomena even if the constants involved here are significantly different than other mechanisms? What variation is there (transistors to transistor) in accumulation? (i.e. are they likely to all fail at once, or distributed over a long period of time?)
- Design time for systems can easily be 5--10 years.
- Challenges
  - Satellites have fixed mission science capabilities, but world events and needs change faster than satellite time-to-design.
  - Systems demand multidimensional energy-delay-area-thermal-reliability optimization, but the design space is navigated sub-optimally by hand.

    - There is a widening gap between Mil/Aero and commercial---currently the gap is around 16 years (e.g., they hope to have 2006 part capabilities in space by 2022).
    - Space systems are designed for the worst-case scenario and the variability of environment in space is very high with the actual worst-case scenarios occurring infrequently.
    - Qualification of components for space is bottlenecked on testing; at least in part, this is because for some devices they must essentially reverse-engineer designs from external experimentation, and testing is the only source of "ground truth" about components.
    - Assessing system reliability is not considered tractable, forcing a focus on part reliability.
    - Many space systems (including satellites) are essentially unique systems, meaning system debug is not amortized across volume units.
    - Community is highly conservative and risk averse.
- Potential Solutions
    - Modeling and tools---this addresses system-level reliability, multi-dimensional optimization, and test challenges; with increased confidence from modeling, it should be possible to reduce sole reliance on test.
    - Agile [adaptable] satellite---this addresses flexible to mission and needs changes and worst-case overdesign by adapting to the current environment. Owners/agencies are, appropriately, most concerned about the control system, so it makes sense to start by using this just for the payload and build confidence for that usage.
    - Multicore---this might address the Mil/Aero vs. commercial gap by exploiting the ability to gang or spare cores for reliability enhancement.
    - Commercial Electronics with modes---this addresses the Mil/Aero vs. commercial gap. Here the hope is to have ways to tune commercial, volume parts that allow use in space. The big benefit is if the same tuning knobs can serve as yield enhancer for commercial market so the features pay for themselves in a commercial setting, but also make them viable for space.
    - Supercomputers+Satellites on same platform---this may partially reduce the uniqueness challenge. It also eases the transition of science and analysis software codes from ground to space.
    - Demonstrate/Validate solutions in commercial use first---this addresses the need to build confidence with the solution to get it accepted by a conservative community. Note that FPGAs, which are now (perhaps grudgingly) being qualified for space, had years of success in ground and commercial signal processing before being considered for space.

  - Synthesizeable cores---this addresses the Mil/Aero vs. commercial gap by allowing designs to be tuned appropriately for space needs (e.g. add extra ECC/parity, replace latches with DICE latches).

Large-Scale Systems

The large-scale systems group was able to initiate some discussions in advance of the meeting. (brief-in slides brief-out slides)

- Challenges:
  - Shared memory is dead. Abstraction is not scalable to large-scale systems with acceptable overhead.
  - System stabilization -- all the largest supercomputer systems are unique. This is really an issue of development and debugging. Here it is not amortized across large number of units as in volume products. It is not clear if this is in scope of our effort. However it comes up in part because of non-scalable reliability solutions. That is, these systems are often larger than previous systems straining reliability in new ways because of their size.
  - Tolerate persistent failures.
  - Tolerate loss of entire data center.
  - "Standard approaches of the last 25-30 years of Fault Tolerance won't address today's challenges" -- Ravi Iyer. These one-size-fits-all approaches lead to solutions that are too expensive, too much overhead for addressing the large-scale reliability problems we now face.
- Potential Solutions:
  - Abstractions across machines---this addresses loss of nodes and data centers and scaling.
  - Application-level abstractions that capture both functionality and operation---this promises light-weight fault tolerance. Note that the key innovations in Map-Reduce is ideas about managing locality and resilience that abstract these away from programmer, but allows the system to manage these issues.
  - Algorithms more robust (prepared for) failures and spatial and temporal locality of distributed machines---this addresses the lack of shared memory and lighter-weight fault tolerance.
  - Components with reliability modes and adaptable resilience could contribute to broader scaling that would help with stablization.
  - Cross-layer solutions may help extend scalability and provide options for stabilization. Note that BlueGene/L used in the 100,000 processor supercomputer at LLNL had L1 data error detection but not correction. This was a reasonable solution for smaller machines where L1 errors were sufficiently rare, but provided a significant limit to machines of this size. LLNL developers were able to reduce the impact of this error by allowing the recovery to be handled by

    the application (*Extending Stability Beyond CPU Millennium* in Supercomputing 2007). This illustrates both how the commodity solution didn't scale to the large system sizes and how cross-layer application-level assistance could make up for the hardware parts which lacked the absolute reliability to address the problem at the component level.

- Other points:
    - Dilation factors replace MTTF. The design is to make things work. With failure recover, the question is not when does it fail, but how much overhead in area, energy, and time does it cost to handle the errors? This is certainly true of the detectable errors. Mean-Time-To-Undetectable-Errors (Unrecoverable-Errors) may still be of interest.

Consumer Electronics

The consumer electronics group used the workshop to kickoff their discussions. (brief-in slides brief-out slides) The commercial sector has been seeing increased need to address failures. So far, solutions have been ad hoc fixes here and there as they trip over acceptable failure rates (e.g. apply ECC here and there to reduce failure rate). Challenges:

- Conflict between variation/reliability and power threatens scaling. We've moved into a power-limited region where we can produce more transistors than we can afford to use. (Example shown suggests that, even if we could use all of our devices at 45nm, by 11nm, we will be able to place almost twice as many devices on the chip as we can afford to power.) So, reducing energy (and hence voltages) are essential to being able to use a good fraction of the new transistors provided by scaling. However, high variation and noise are preventing the voltage from scaling down. Handling these reliability problems without spending excessive energy is thus essential to continuing to derive benefits from scaled technologies. From this perspective the question come up, how much overhead are we willing to pay for reliability? But the real question is, what is the net benefit to achieving an acceptable level of reliability after scaling and mitigation overhead? How much of our traditional capacity doubling per generation do we effectively lose? or can we change the game so we net recover capacity we've had to pay as overhead in the past?
- We do not have the CAD tools necessary for resilience analysis. Much of the data needed to drive tools is empirical and must come back from runtime (e.g. activity factors).
- We're always in the scenario where we don't fully understanding aging. Uncertainty of process effects will be with us as we move forward in technology.

Potential Solutions:

- Resilience can be a tool to allow exploitation of small feature sizes. Efficient techniques for supporting reliability have a second effect of allowing use of more aggressive feature sizes, performance, and voltage scaling.
- Adaptation so run at average case rather than worst case---recover the energy overhead spent for worst-case scenaios by detecting failure and recovering in those cases. In-system adaptation potentially provides the missing runtime data needed to find good solutions by pushing some optimization into the system lifetime. Note that it only takes 5 seconds to run full processor diagnostics, suggesting it is viable to run self-diagnostic and assessment at least once per day with negligible cost.
- Information margins instead of energy margins---more reliability mitigation from the now-expensive mode of energy into cheaper modes.

Common themes identifiable so far...

Groups continue to work and refine their challenges. From the discussions at the workshop, a few themes that are beginning to emerge include:

- Varying demands, workloads, environment (and uncertainty about the environment) means worst-case design is overdesign for most uses. This motivates adaptive solutions.
- Worst-case design independent of the application and its needs is too expensive. Similarly, worst-case design for uncommon, but potentially avoidable, worst-case scenarios is also a large, unnecessary cost. These motivate cross-layer, application-aware solutions and/or models/middleware that support management of operational aspect of application.
- Fully custom/unique construction of all components is not viable (costs, manpower) for anyone. Some domains see more acute versions of this, but no domain is really able to do everything custom themselves these days. This motivate: interfaces/metrics/tools to perform composition/analysis/optimization/validation of separately sourced (sub)components.
- Across the board, there is considerable conservative overdesign. This motivates system assessment methodology, tools support energy-delay-area-reliability-thermal-mechanical space.
- Modes and configuration options that allow the component to tune what it spends on reliability. This could allow commercial devices to enhance yield or operate at extremely low energy levels while also making the same parts more usable in larger scale systems or harsher environments.

Next Steps

All of these groups need further work to refine their story and make their challenges more quantitative. They will continue to work during the coming month to draft more complete answers for input into the full study report.

Our working schedule is:

- August 24: AssignmentsFromJuly meeting due to group leaders. The Roadmap group already has a deadline in August for their contribution to ITRS, and will be continuing their work afterwards. New groups (infrastructure, life critical) should have a status report/plan for this date.
- September 15: Drafts of vision/consensus summaries due from each working group
- October 7: 1st draft of final report outline done (core group), circulated to workshop participants
- 7 days before 3rd workshop: Comments on first draft due back to core group
- Late October: 3rd workshop
- 12/1: 1st draft of final report released to workshop participants
- 12/15: Feedback on 1st report draft due

We still need to constitute the infrastructure and life-critical groups. We took further contact suggestions from the workshop and are pursuing this further. Additional suggestions are still appreciated, especially in the life-critical area.

The organizers are planning to try to talk with individual NSF Program Mangers in August to help them see where we are and get input on what else may be needed to provide a compelling, convincing, and useful story to them. The organizers are open to talking to program managers in other agencies that may be appropriate. Suggestions and contacts are welcomed.

Final workshop will be at the IBM conference cneter in Austin, TX in October. We are taking input on the dates and are particularly trying to select a date that will allow NSF program managers to attend. Preliminary polling is leaning toward late October (29/30th), but we have not, yet, set a final date.