

Billion-Transistor Architectures: There and Back Again

pp. 22-28

Doug Burger and James R. Goodman

In September 1997, *Computer* published a special issue on billion-transistor microprocessor architectures. Comparing that issue's predictions about the trends that would drive architectural development with the factors that subsequently emerged shows a greater-than-predicted emphasis on clock speed and an unforeseen importance of power constraints.

Of seven architectural visions proposed in 1997, none has yet emerged as dominant. However, as we approach a microarchitectural bound on clock speed, the primary source of improved performance must come from increased concurrency. Future billion-transistor architectures will be judged by how efficiently they support distributed hardware without placing intractable demands on programmers.

Metamorphic Programming: Unconventional High Performance

pp. 30-38

Peter M. Maurer

A programming methodology that violates most of the rules of good programming has shown spectacular reductions in simulation times on several benchmarks. Applying this technique in logic-level VLSI circuit simulation also improved simulation performance. For a new VLSI circuit, faster simulation translates into faster time to market, so even the most peculiar programming type is worth exploring if the carrot is increased performance.

Discovering efficient and effective metamorphic programming techniques across a range of problems outside simulation will require a concerted effort across the software community. The most important problem is the lack of metamorphic constructs in mainstream high-level languages.

Reliable and Efficient System-on-Chip Design

pp. 42-50

Naresh R. Shanbhag

To increase processor performance, the microprocessor industry is scaling feature sizes into the deep sub-micron and sub-100-nanometer regime. The recent emergence of noise and the dramatic increase in process variations have raised serious questions about using nanometer process technologies to design reliable, low-power, high-performance computing systems.

The design and electronic design automation communities must work closely with the process engineering community to address these problems. Specifically, researchers must explore the tradeoffs between reliability and energy efficiency at the device, circuit, architectural, algorithmic, and system levels.

Going Beyond Worst-Case Specs with TEAtime

pp. 51-56

Augustus K. Uht

Virtually all engineers use worst-case component specifications for new system designs, thereby ensuring that the resulting product will operate under worst-case conditions. However, given that most systems operate under *typical* operating conditions that rarely approach the demands of worst-case conditions, building such robust systems incurs a significant performance cost. Further, classic worst-case designs do not adapt to variations in either manufacturing or operating conditions.

A timing-error-avoidance prototype provides a circuit and system solution to these problems for synchronous digital systems. TEAtime has demonstrated much better performance than classically designed systems and also adapts well to varying temperature and supply-voltage conditions.

Making Typical Silicon Matter with Razor

pp. 57-65

Todd Austin, David Blaauw, Trevor Mudge, and Krisztián Flautner

Voltage scaling has emerged as a powerful technology for addressing the power challenges that current on-chip densities pose. Razor is a voltage-scaling technology based on dynamic, in-situ detection and correction of circuit-timing errors. Razor permits design optimizations that tune the energy in a microprocessor pipeline to typical circuit-operational levels. This eliminates the voltage margins that traditional worst-case design methodologies require and lets digital systems run correctly and robustly at the edge of minimum power consumption.

Occasional heavyweight computations may fail and require additional time and energy for recovery, but the optimized pipeline requires significantly less energy overall than traditional designs.

Speeding Up Processing with Approximation Circuits

pp. 67-73

Shih-Lien Lu

Current microprocessors employ a global timing reference to synchronize data transfer. A synchronous system must know the maximum time needed to compute a function, but a circuit usually finishes computation earlier than the worst-case delay. The system nevertheless waits for the maximum time bound to guarantee a correct result.

As a first step in achieving variable pipeline delays based on data values, approximation circuits can increase clock frequency by reducing the number of cycles a function requires. Instead of implementing the complete logic function, a simplified circuit mimics it using rough calculations to predict results. The results are correct most of the time, and simulations show improvements in overall performance in spite of the overhead needed to recover from mistakes.