Self-Test and Adaptation for Random Variations in Reliability

Kenneth M. Zick and John P. Hayes
University of Michigan, Ann Arbor, MI USA

August 31, 2010

Motivation

- Physical variation is increasing dramatically

  $\Delta P = \Delta P_{D2D} + \Delta P_{sc} + \boldsymbol{\Delta P_{rand}}$

- Transient faults also increasing in prominence
- What about random variations in transient fault reliability? Very little has been published

Danger ahead

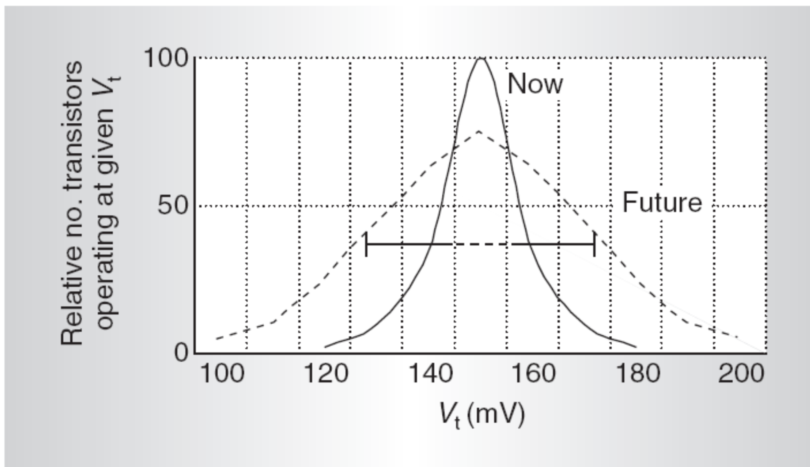


Figure 4. Extreme device variations will become more typical in the future.

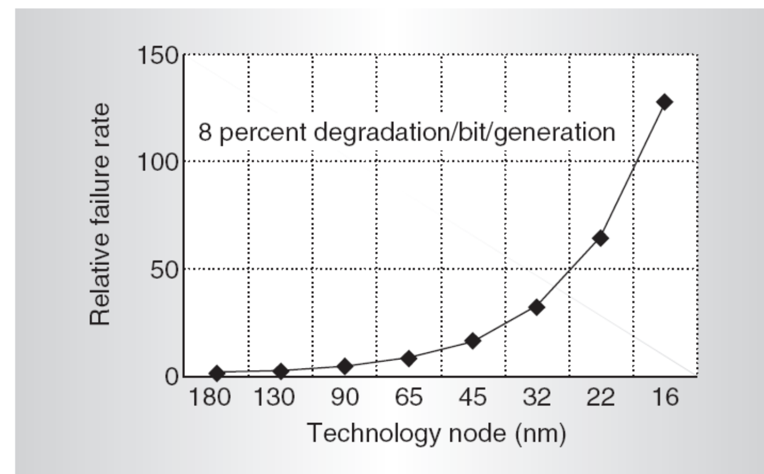


Figure 5. Soft-error failure-in-time of a chip (logic and memory).

Source: Borkar, *IEEE Computer,* 2005

Transient faults

- Caused by radiation and other noise sources
- Random variations in vulnerability have become significant
  - Monte Carlo sim shows Q_{CRIT} variations for four flip-flop designs:

    3σ variation in $Q_{CRIT,1\rightarrow 0}$ = 44 to 115% [Mostafa'09]

    3σ variation in $Q_{CRIT,0\rightarrow 1}$ = 20% to 72%
  - Huge variation in Fault rate $\propto$ Flux × Area × e $^{-QCRIT/QCOLL}$
- Upsetability of individual cells in a chip? Nobody knows!
- Conventional wisdom: nothing you can do anyway

Vision

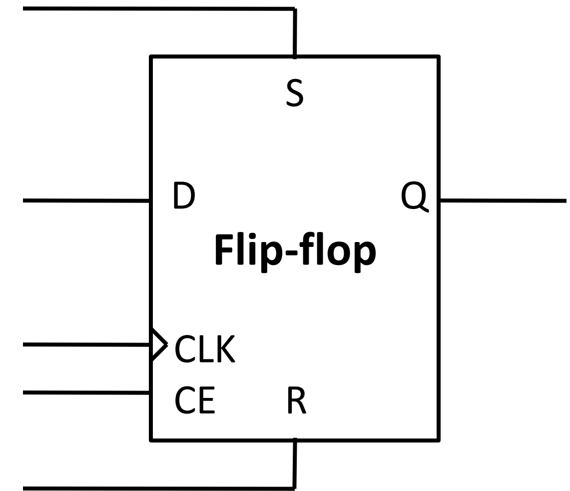
- Need low-cost, fine-grained methods of introspection and self-optimization. Physically-adaptive computing
- Goals: better parametric yield, fewer soft errors, improved power & energy efficiency, longer system lifetimes
- Applicable to FPGA-based systems as well as reconfigurable nanoarchitectures

Proposed approach to self-test

- Introspection: systems probe their own components to uncover random variations
- Generate synthetic noise on-chip
- Inject noise into components during self-test
- Use data to infer variations in reliability

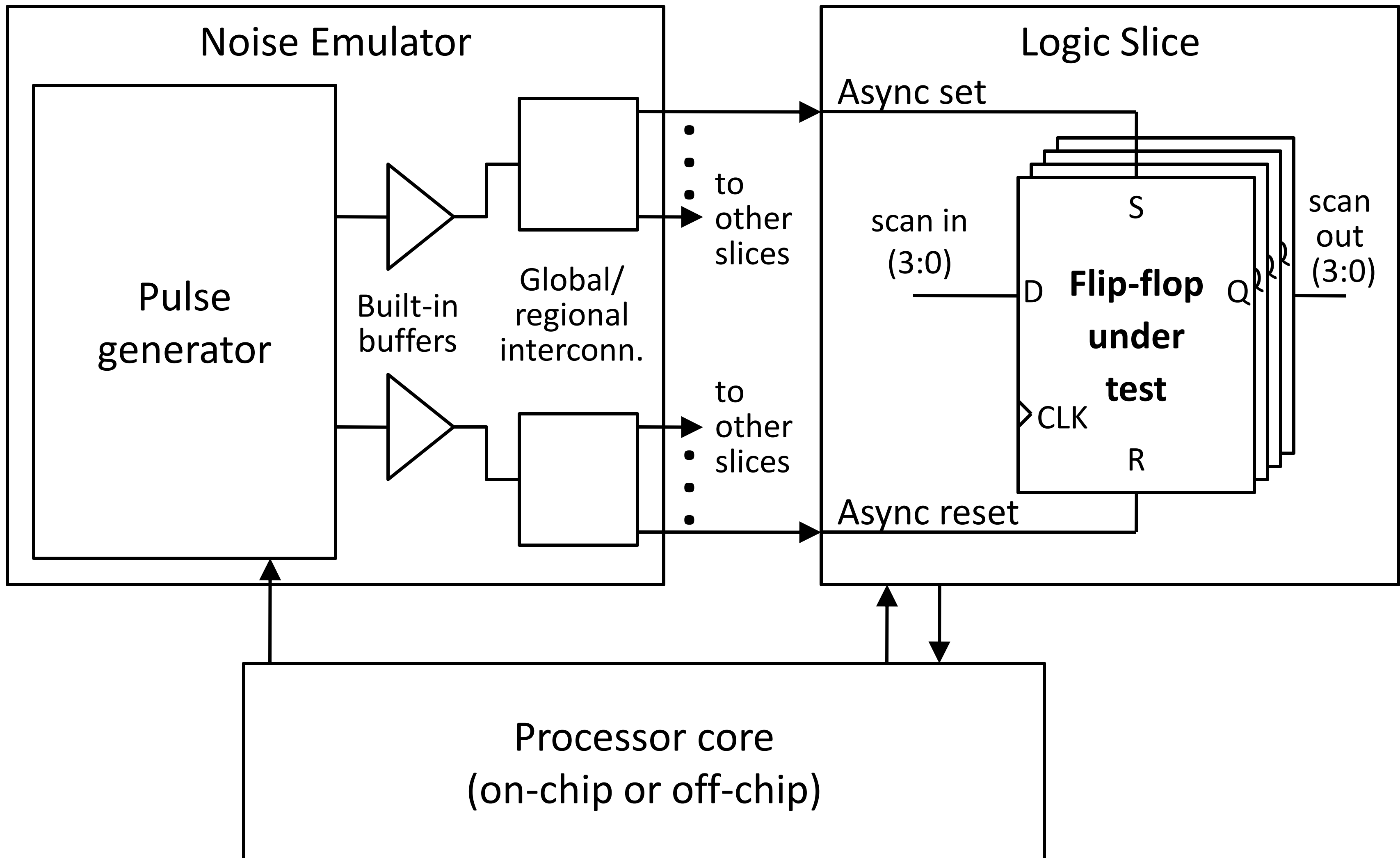
Self-test for latch reliability

- Flip-flops hold temporary state information needed for computation
- Prone to single event transients (SETs) and single event upsets (SEUs)
- Can't be protected by ECC. Need TMR, or extra circuitry



- Virtex-5QV will include SET filters (up to 800ps) and SEU-resistant latches, but most systems don't have them
- Proposal: inject synthetic noise via asynchronous set/reset
- Look for *intra-slice* variations in upsetability (SETs, SEUs)

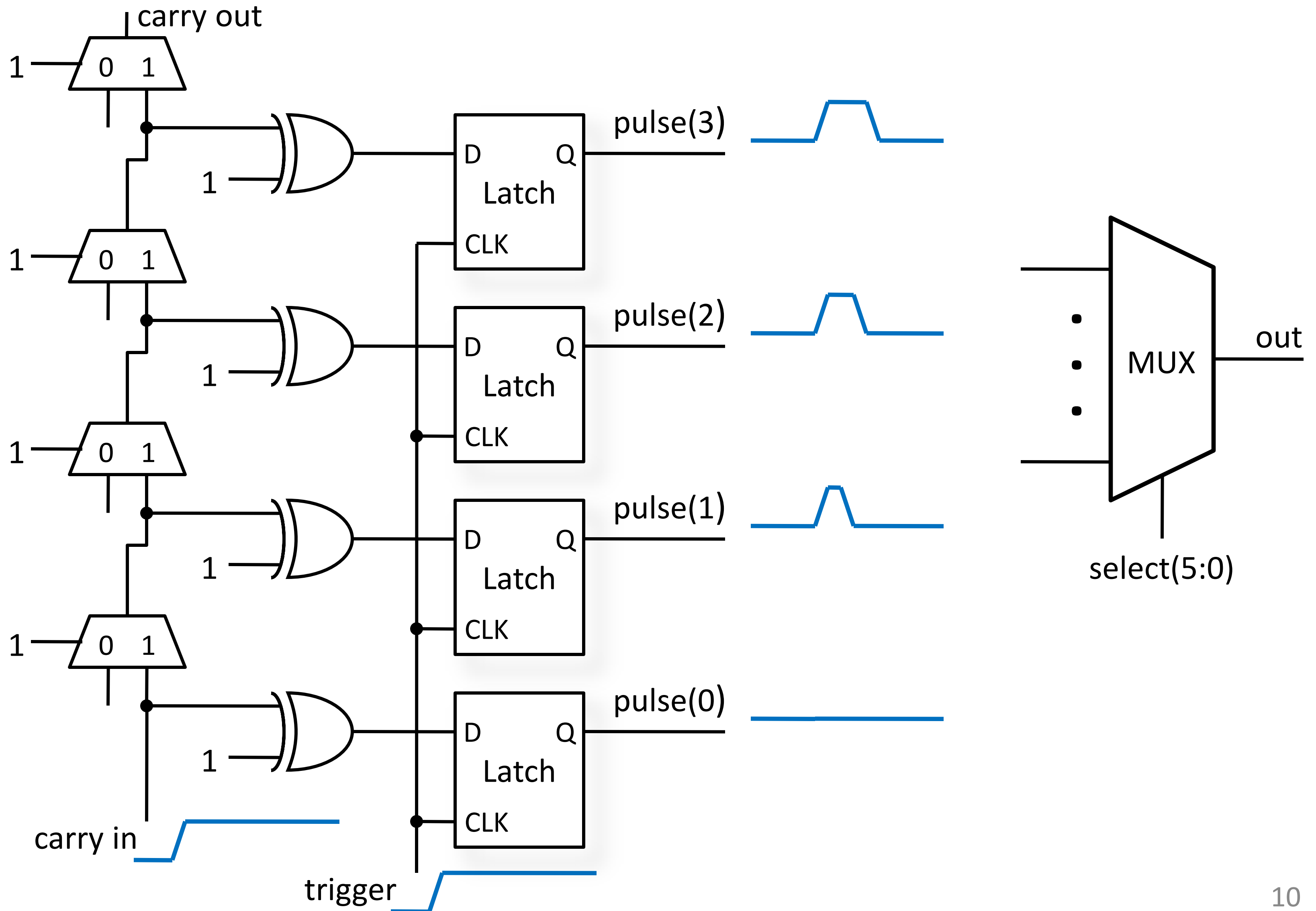
Proposed self-test configuration



Pulse generator

- Similar to digital-to-time converter (DTC)
- Desire high resolution. Linearity less important here.
- Some options:
  - Dual PLLs using the Vernier principle. 35ps but overhead
  - Delay line such IODELAY. 78ps
  - Carry chain. ~50ps

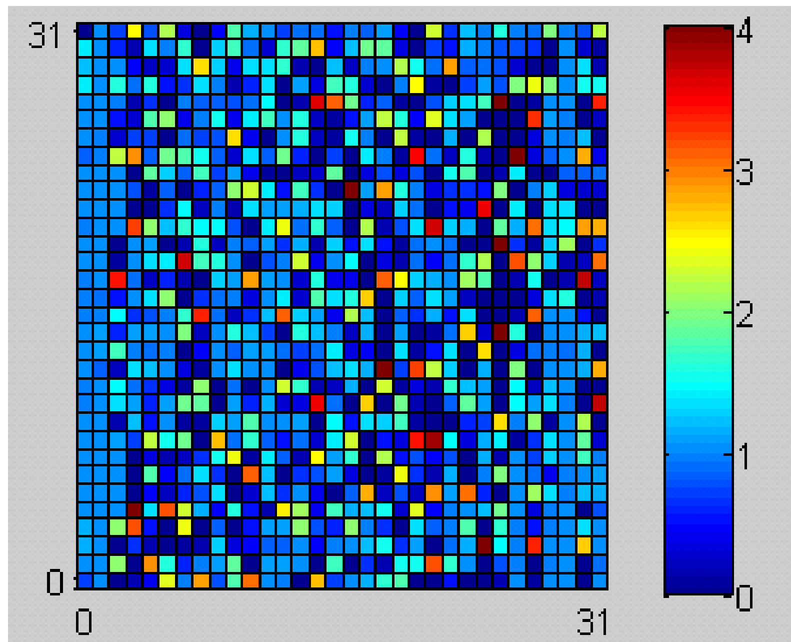
Portion of pulse generator



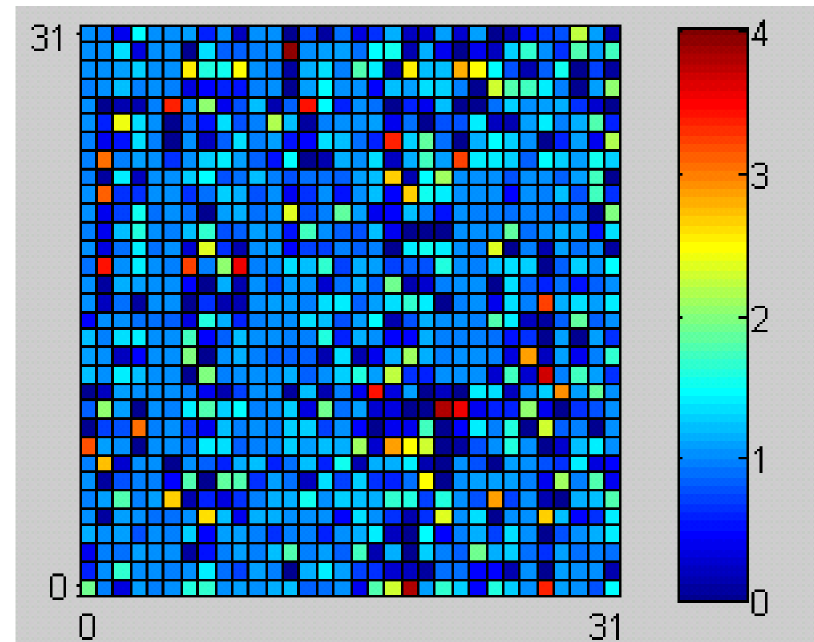
Experimental setup

- Two Virtex-5 LX110T FPGAs
- 1,024 flip-flops under test
- Pulse widths ~600ps (calibrated to each slice)
- MicroBlaze processor
- Overhead:
  - 64KB of MicroBlaze memory in total
  - Calibration: 4B of data per slice. Execution time = 3 min.
  - Characterization time: 5 seconds

Results - upsetability maps



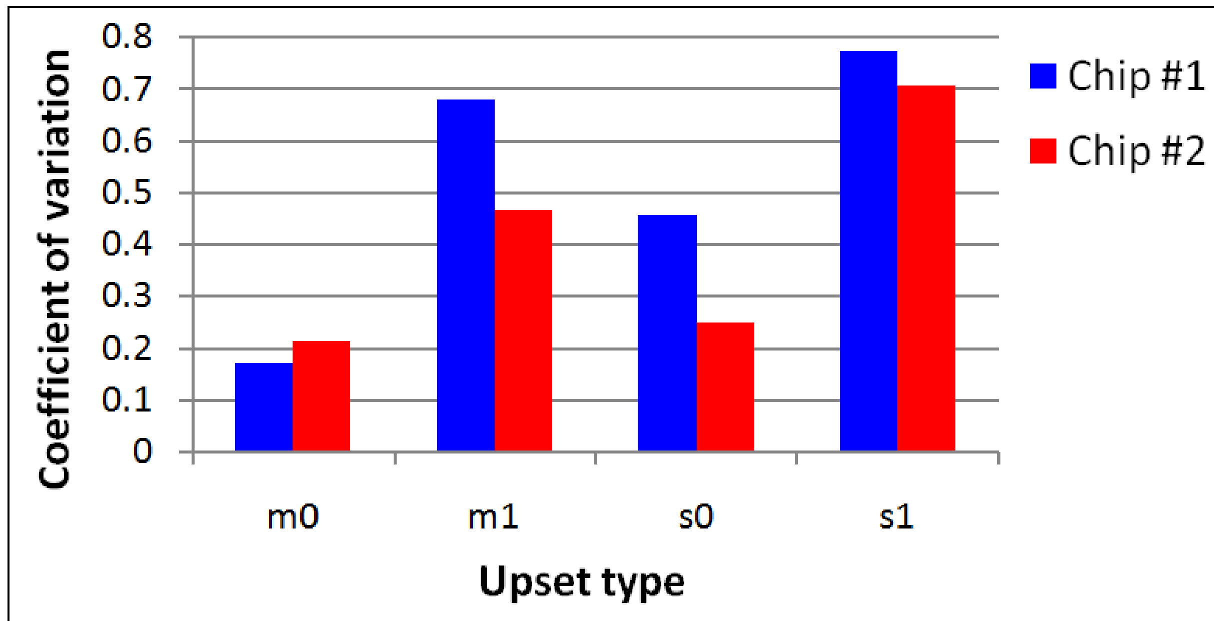
Chip #1



Chip #2

32x32 array of flip-flops. Values shown are the ratios of latch upsets to the mean number of latch upsets for the associated slice, over 255 trials. The test case is master latches in the 1 state. Slices span four cells vertically.

Quantifying intra-slice variation



Coefficient of variation (σ/μ) for latch upsets within a slice in the tested noise environment, averaged over 256 slices.

Upsets vs. location within slice

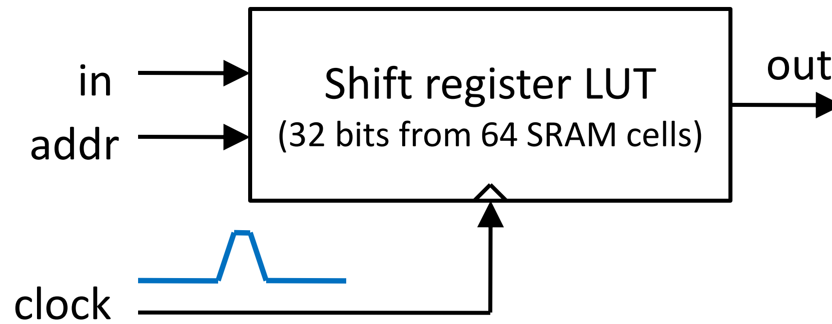
Distribution of 500,000 upsets

Location in slice	Chip #1 upset distribution	Chip #2 upset distribution
D	24.7%	24.5%
C	25.3%	25.2%
B	24.8%	24.9%
A	25.2%	25.3%

- Any systematic bias is negligible compared to the large intra-slice variations
- Results are consistent with variations that are random by latch

What about LUT cells?

- Idea: inject noise into shift register LUTs



- Found a strong bias toward upsets in 1 bits.
- Possible extension: search for marginal SRAM cells. Validate against radiation data.

Adapting to latch variations

- Define the cost function to be the total raw upset rate
  - Latch upsetability: *m*0, *m*1, *s*0, *s*1. Characterized via self-test
  - Signal probabilities *SP*. Can be characterized via logic simulation or "capture & readback"
  - Cost of a state bit *i* placed at flip-flop *j*:

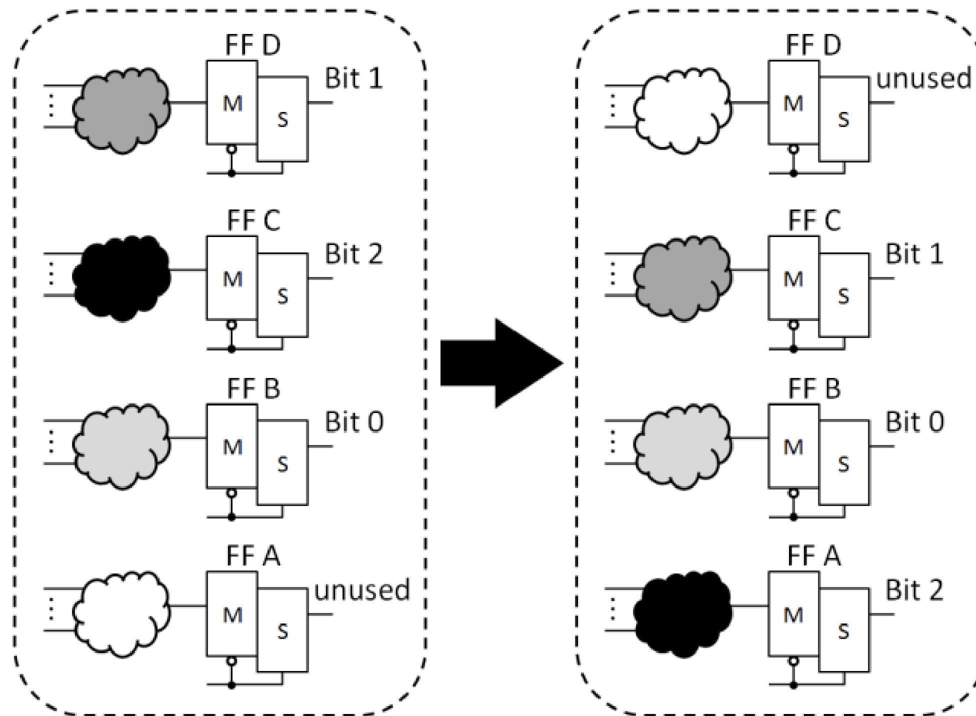
$$cost_{ij} = (m1_j + s1_j)\, SP_i + (m0_j + s0_j)\,(1 - SP_i) \qquad (1)$$

  - Estimated MTBF:

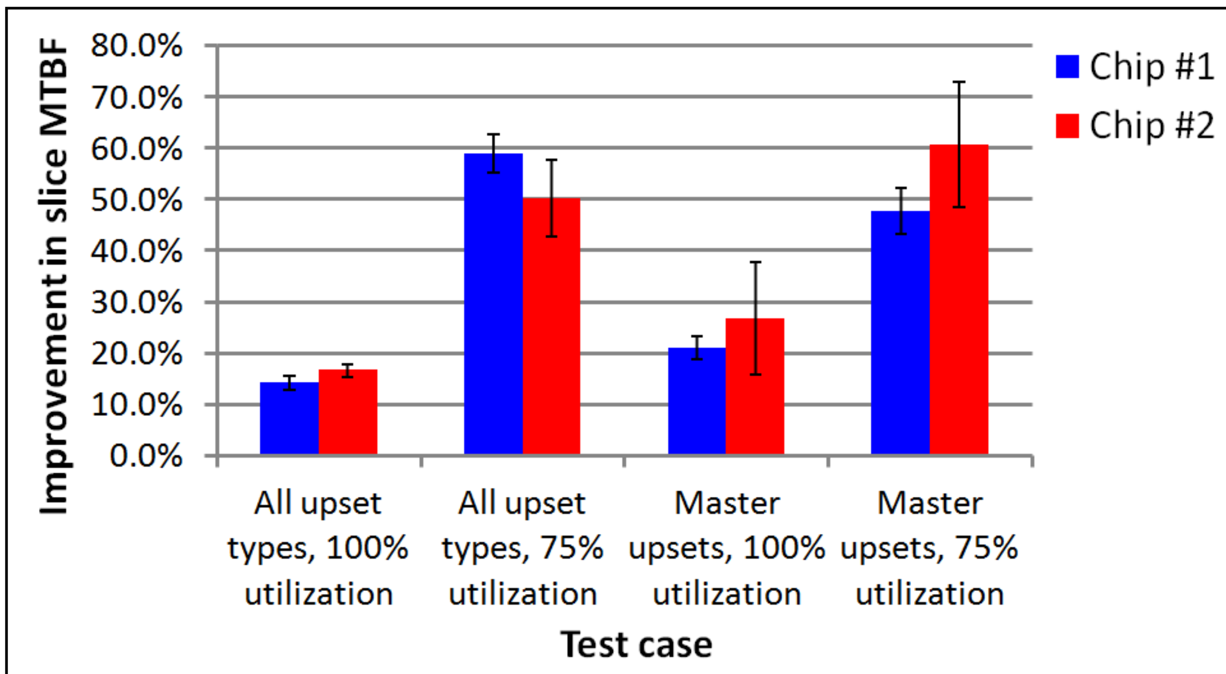
$$\text{MTBF} = 1/ \sum_i \sum_j cost_{ij}\, x_{ij} \qquad (2)$$

- Find configurations that maximize the MTBF
- *Fault avoidance*. Complement to error mitigation.

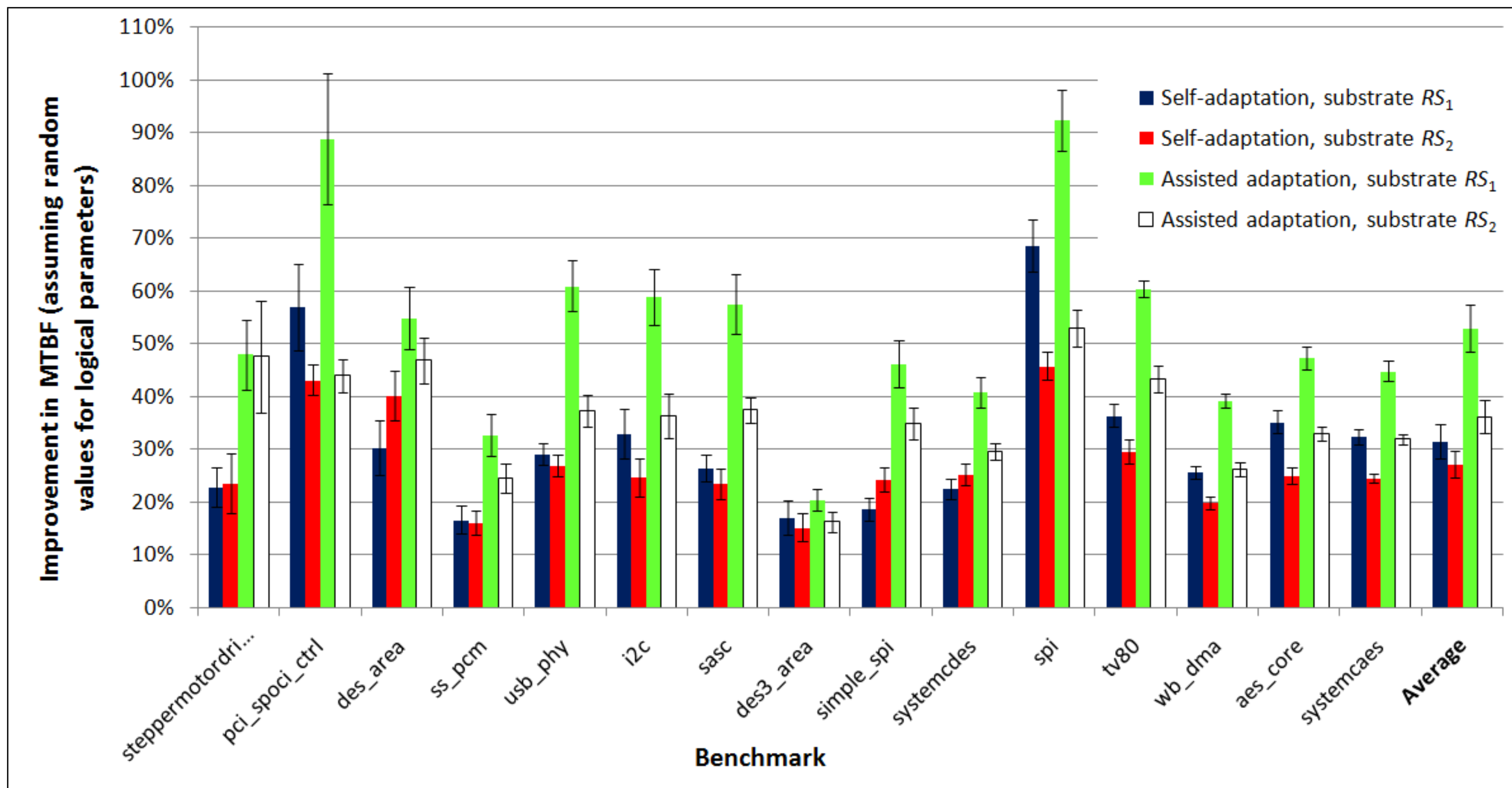
Intra-slice optimization



Potential for reconfiguration



Recent benchmark results



Improvements in MTBF for self-adaptation and assisted adaptation relative to the non-adaptive case, assuming uniformly random signal probabilities. Error bars show the standard deviation across 10 trials.

Conclusions

- Wealth of physical information is out there waiting to be discovered and put to use
- Random variations can be significant and can be estimated via self-test
- Field programmable systems have unique potential for self-test and self-optimization
- Much interesting research ahead!

Acknowledgments

- NASA GSRP Fellowship & NASA Langley Research Center
- National Science Foundation grant CCF-0702276
- Xilinx Inc. and Sun Microsystems
- Adaptive Hardware & Systems group at U. Michigan

Thank you!

Backup slides

Example of CMOS latch