An Architectural Perspective on Soft Errors from Cosmic Radiation

Shubu Mukherjee

shubu.mukherjee@intel.com, VSSAD, MMDC, Intel Corporation

FACT (Fault Aware Computing Technology) Project

Joel Emer, Shubu Mukherjee, Steve Reinhardt, and Chris Weaver

Other Contributors

Mike Kontz, currently in HP

44th IFIP WG10.4 Workshop on Hardware Design and Dependability, June 28, 2003

"If a problem has no solution, it may not be a problem, but a *FACT*, not to be solved, but to be coped with over time," Shimon Peres, Nobel Laureate 1994.

Evidence of Cosmic Ray Strikes

- Documented strikes in large servers found in error logs
  - Normand, “Single Event Upset at Ground Level,” IEEE Transactions on Nuclear Science, Vol. 43, No. 6, December 1996.
- Sun Microsystems, 2000 (R. Baumann, 2002 IRPS Workshop talk)
  - Cosmic ray strikes on L2 cache with no error detection or correction
    - caused Sun,s flagship servers to suddenly and mysteriously crash!
  - Companies affected
    - Baby Bell (Atlanta), America Online, Ebay, & dozens of other corporations
    - Verisign moved to IBM Unix servers (for the most part)

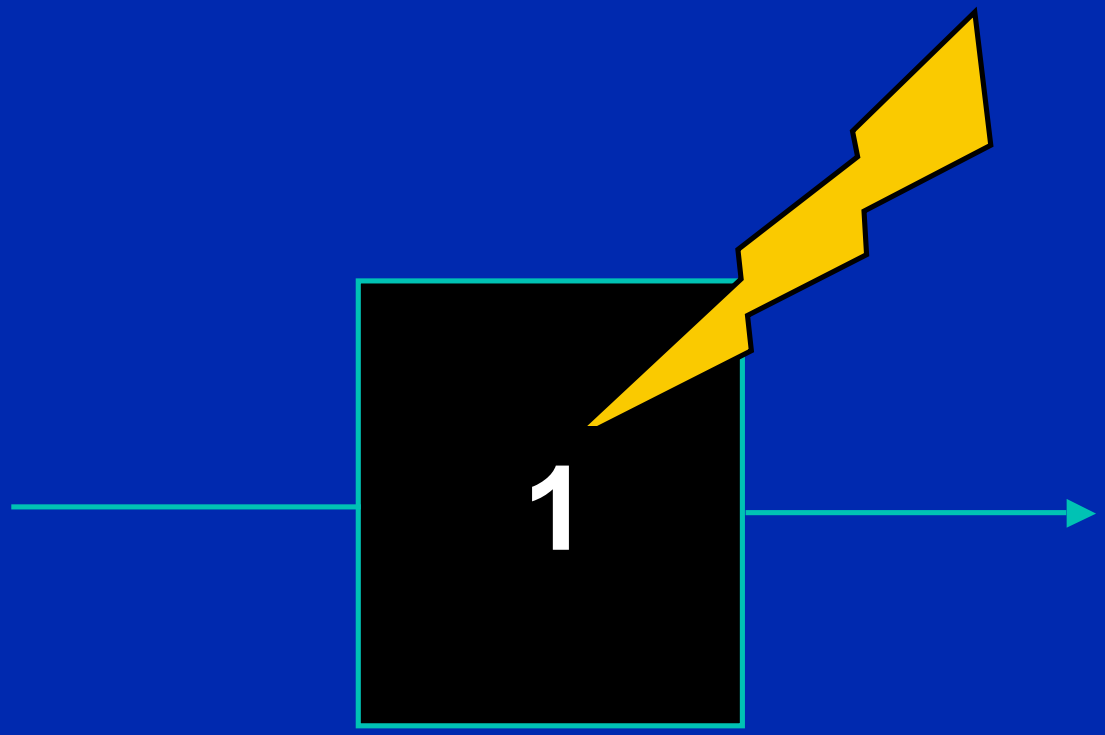
Reactions from Companies

- Fujitsu SPARC in 130 nm technology
  - 80% of 200k latches protected with parity
  - compare with very few latches protected in Mckinley
  - ISSCC, 2003
- IBM declared 1000 years system MTBF as product goal
  - for Power4 line
  - very hard to achieve this goal in a cost-effective way
  - Bossen, 2002 IRPS Workshop Talk

Outline

- Faults from Cosmic Rays
- Terminology
- Computing a chip,s Soft Error Rate
- Redundant Multithreading
- Summary

Strike Changes State of a Single Bit



Impact of Neutron Strike on a Si Device

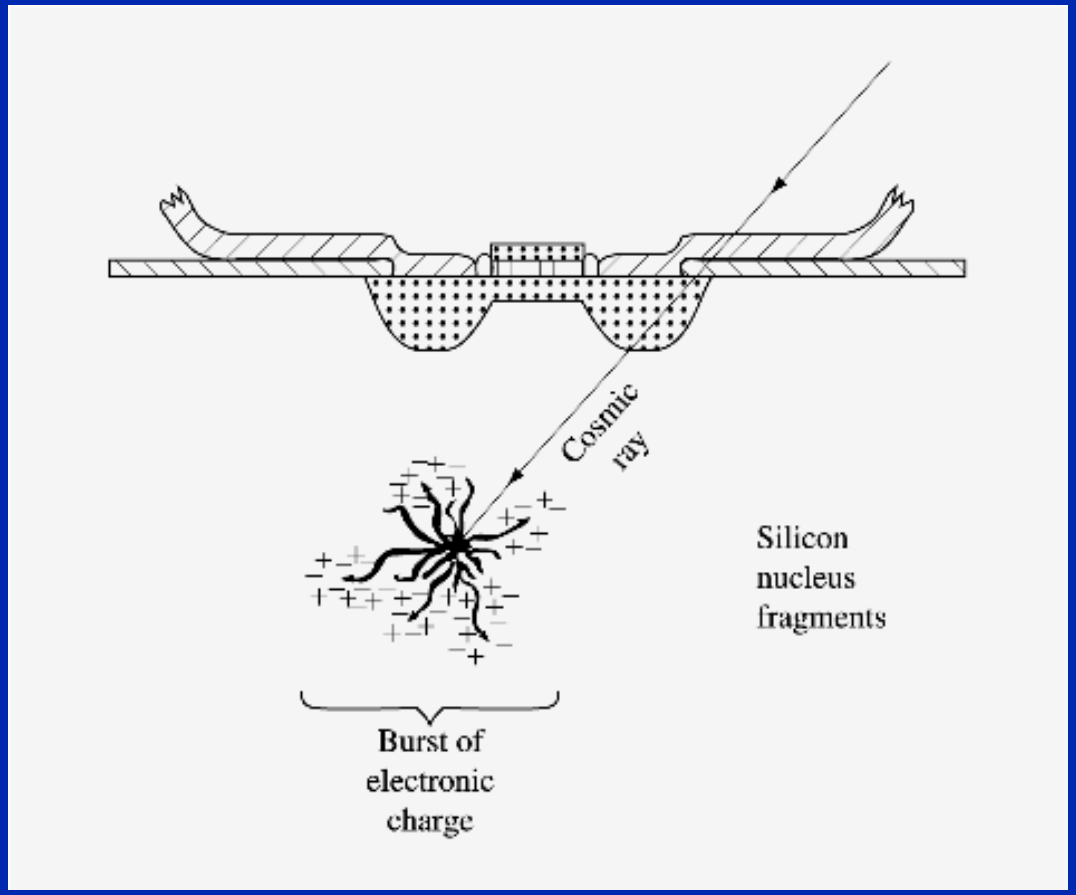


Figure 3, Ziegler, et al., "IBM experiments in soft fails in computer electronics (1978 - 1994)," IBM J. of R. & D., Vol. 40, No. 1, Jan. 1996.

- Strike creates electron-hole pairs that can be absorbed by source/diffusion areas to change state of device

Origin of Cosmic Rays

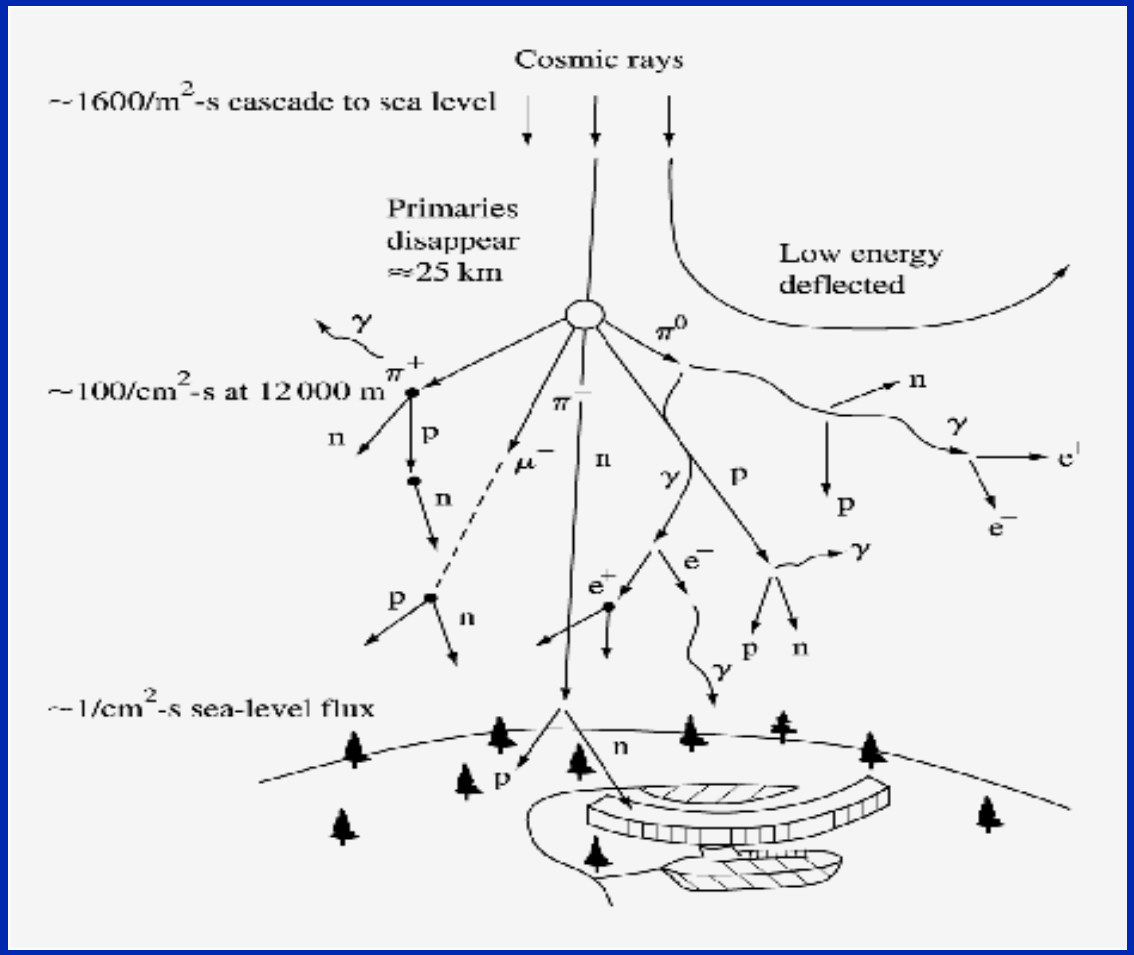


Figure 2, Ziegler, et al., "IBM experiments in soft fails in computer electronics (1978 - 1994)," IBM J. of R. & D., Vol. 40, No. 1, Jan. 1996.

- Cosmic rays come from deep space

Impact of Elevation

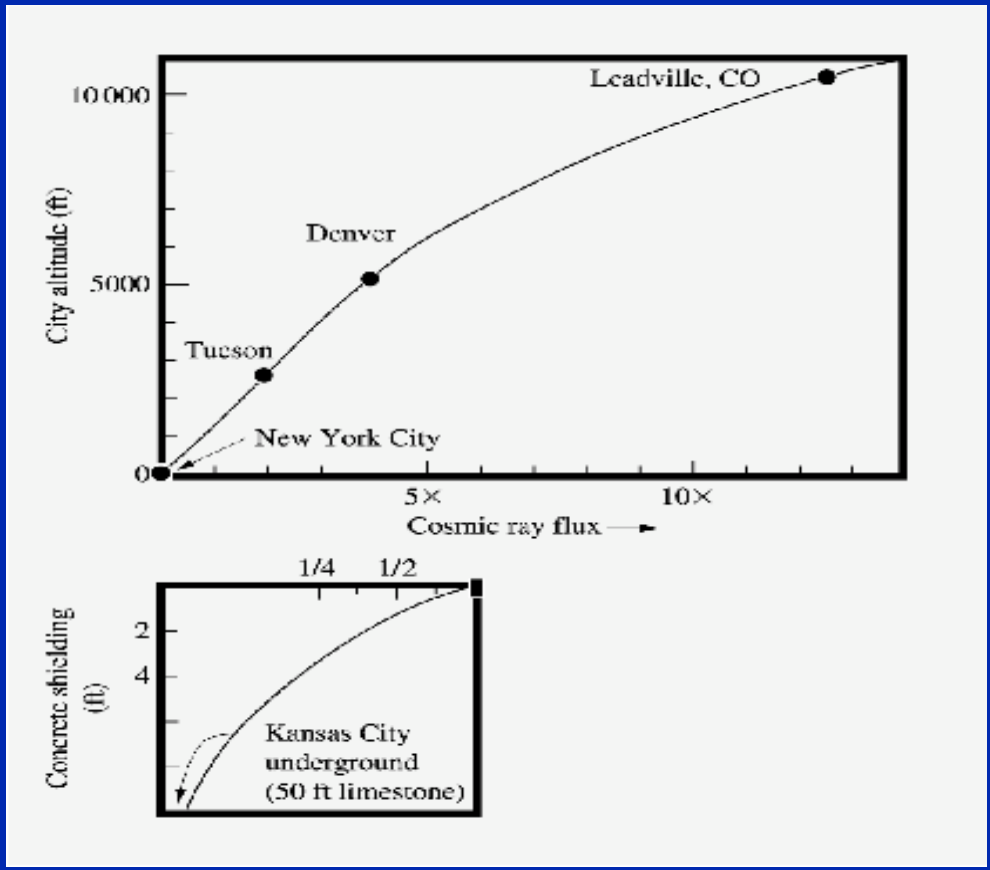


Figure 8, Ziegler, et al., "IBM experiments in soft fails in computer electronics (1978 - 1994)," IBM J. of R. & D., Vol. 40, No. 1, Jan. 1996.

- 3x - 5x increase in Denver at 5,000 feet
- 100x increase in airplanes at 30,000+ feet

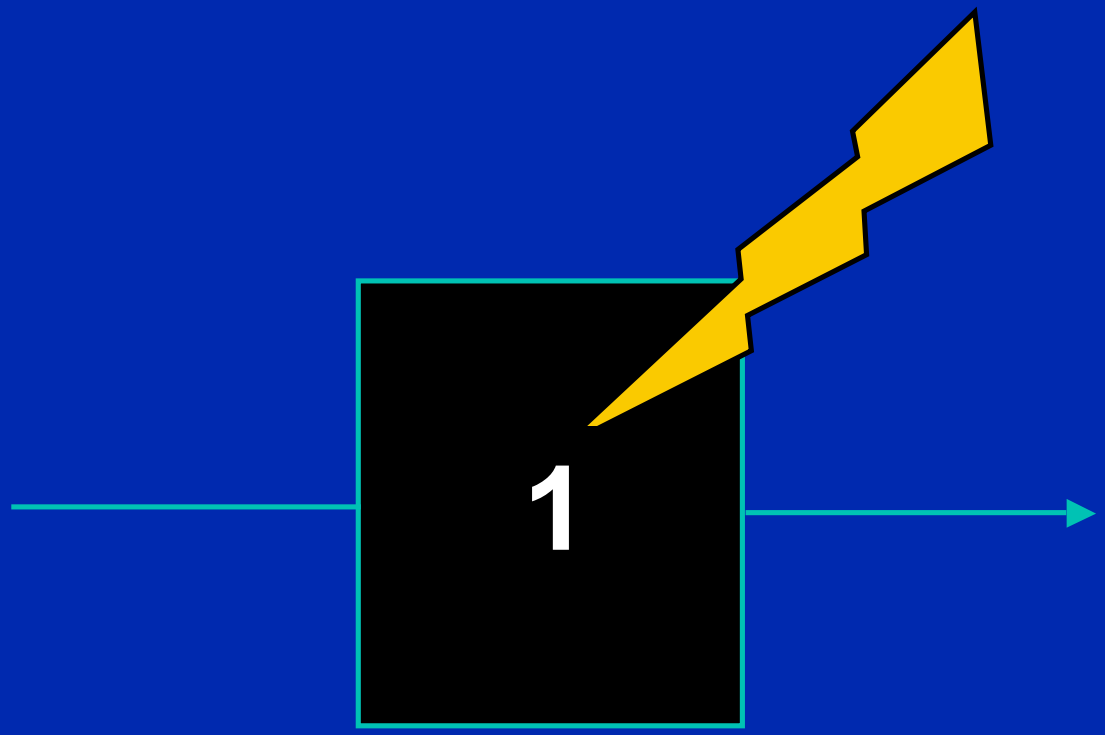
Physical Solutions are hard

- Shielding?
  - No practical absorbent (e.g., approximately > 10 ft of concrete)
  - unlike Alpha particles
- Technology solution: SOI?
  - SOI probably no help in 250 nm and beyond
- Radiation-hardened cells?
  - 10x improvement possible with significant penalty in performance, area, cost
  - 2-4x improvement may be possible with less penalty
- We think some of these techniques will help alleviate the impact of Soft Errors, but not completely remove it

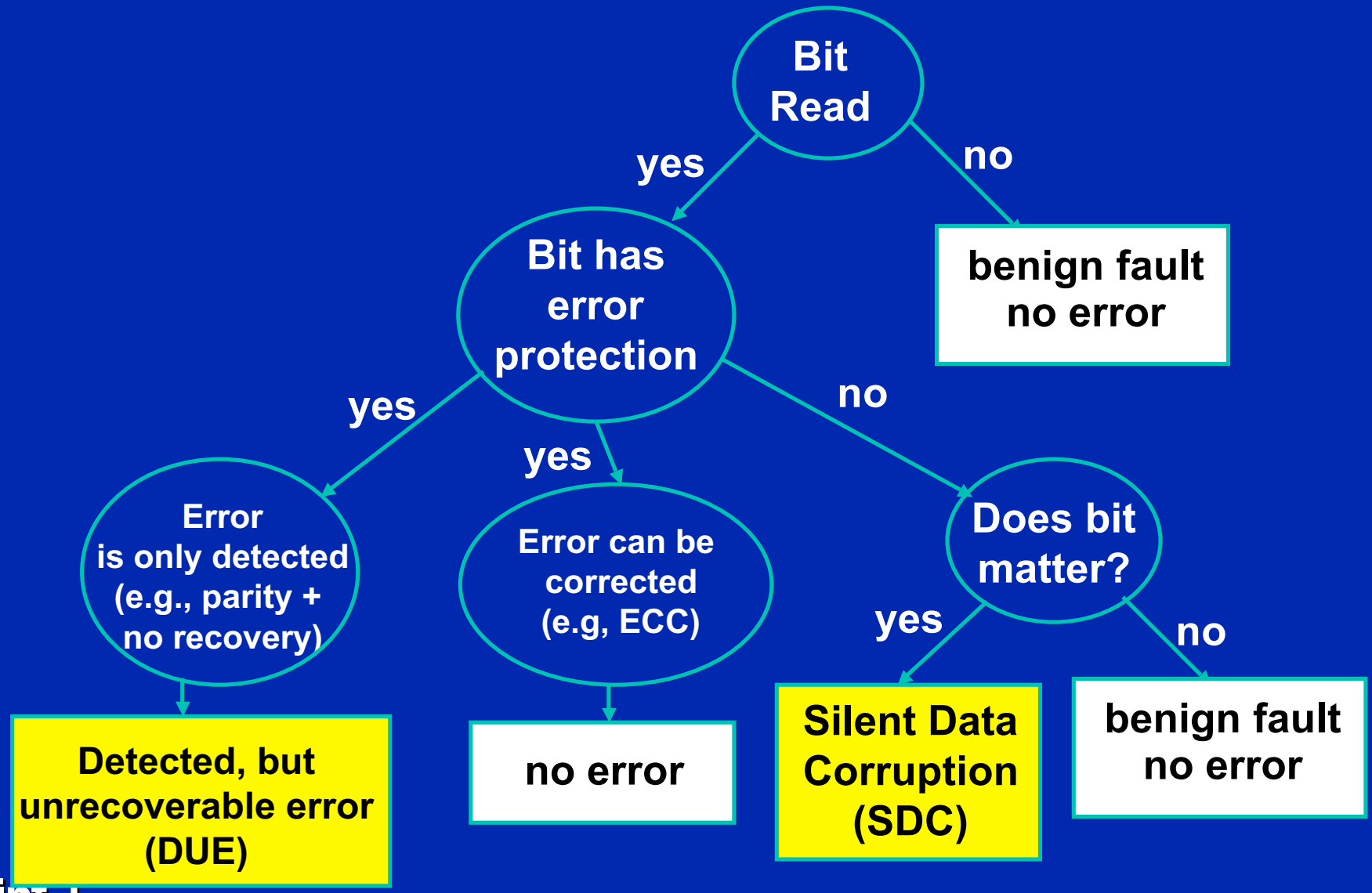
Outline

- Faults from Cosmic Rays
- Terminology
- Computing a chip,s Soft Error Rate
- Redundant Multithreading
- Summary

Strike Changes State of a Single Bit



Strike on state bit (e.g., in register file)



Definitions 1

- SDC = Silent Data Corruption
- DUE = Detected & unrecoverable error
- SER = Soft Error Rate = Total of SDC & DUE

Definitions 2

- **Interval-based**
  - MTTF = Mean Time to Failure
  - MTTR = Mean Time to Repair
  - MTBF = Mean Time Between Failures = MTTF + MTTR
  - Availability = MTTF / MTBF
- **Rate-based**
  - FIT = Failure in Time = 1 failure in a billion hours
  - 1 year MTTF = 10^9 / (24 * 365) FIT = 114,155 FIT
  - SER FIT = SDC FIT + DUE FIT

IBM,s Soft Error Goals for Power4
(D.C.Bossen, 2002 IRPS Tutorial Reliability Notes)

Error Type	IBM System MTBF Target at 300 meters (ref. D. Bossen)
SDC (Silent Data Corruption)	1000 years (114 FIT)
DUE for system crash	25 years
DUE for application crash	10 years

Outline

- Faults from Cosmic Rays
- Terminology
- Computing a chip,s Soft Error Rate
- Redundant Multithreading
- Summary

Measuring a Chip,s FIT

Chip	Physically bombard with neutrons in neutron accelerators
Circuit Models + RTL	Obtain raw error rate Statistical fault injection
Circuit Models + Performance Model	Obtain raw error rate Work in progress in FACT group

- Like performance measurement

Computing FIT rate of a Chip

- **FIT Rate Law:** FIT rate of a system is the sum of the FIT rates of its individual components
- **Vulnerable Bit Law:** FIT rate of a chip is the sum of the FIT rate of *vulnerable* bits in that chip!
- **Total FIT =**

  $\sum_{\text{(for each vulnerable device i)}} (\text{raw soft error rate}_i * \text{vulnerability factor}_i)$

  - Vulnerability Factor = fraction of faults that become errors
  - Vulnerability Factor is also known as "derating factor" and "soft error sensitivity (SES)."

FIT Equation: Raw Soft Error Rate

FIT = $\sum_{\text{(for each vulnerable device i)}}$ (raw soft error rate$_i$ * vulnerability factor$_i$)

- SRAM cells
  - FIT/bit decreasing slightly across generations w/ usu. voltage scaling
  - FIT/chip increasing overall
- Latch cells
  - FIT/bit constant across generations w/ usu. voltage scaling
- Static Logic Gates
  - ignored, see later
- Dynamic Logic
  - similar to latches

FIT Equation: Vulnerability Factors

FIT = $\sum_{\text{(for each vulnerable device i)}}$ (raw soft error rate$_i$ * vulnerability factor$_i$)

Vulnerability Factor =

Timing Vulnerability Factor * Architectural Vulnerability Factor

- Timing Vulnerability Factor
  - fraction of time bit is vulnerable
- Architectural Vulnerability Factor (AVF)
  - fraction of time bit matters for final output of a program

Timing Vulnerability Factor

- SRAM cells
  - 100%
- Latch cells
  - ~ 50%
- Static Logic Gates
  - Shivakumar, et al. (DSN 2002) predict near zero today
  - signal attenuation and latch window masking
  - may be a problem in future
- Dynamic Logic: reference Rachid Rayess
  - $1 / 2^{N+1}$, where N = # pulldowns
  - 2 pulldowns: ~13%
  - 8 pulldowns: ~2%

Architectural Vulnerability Factor

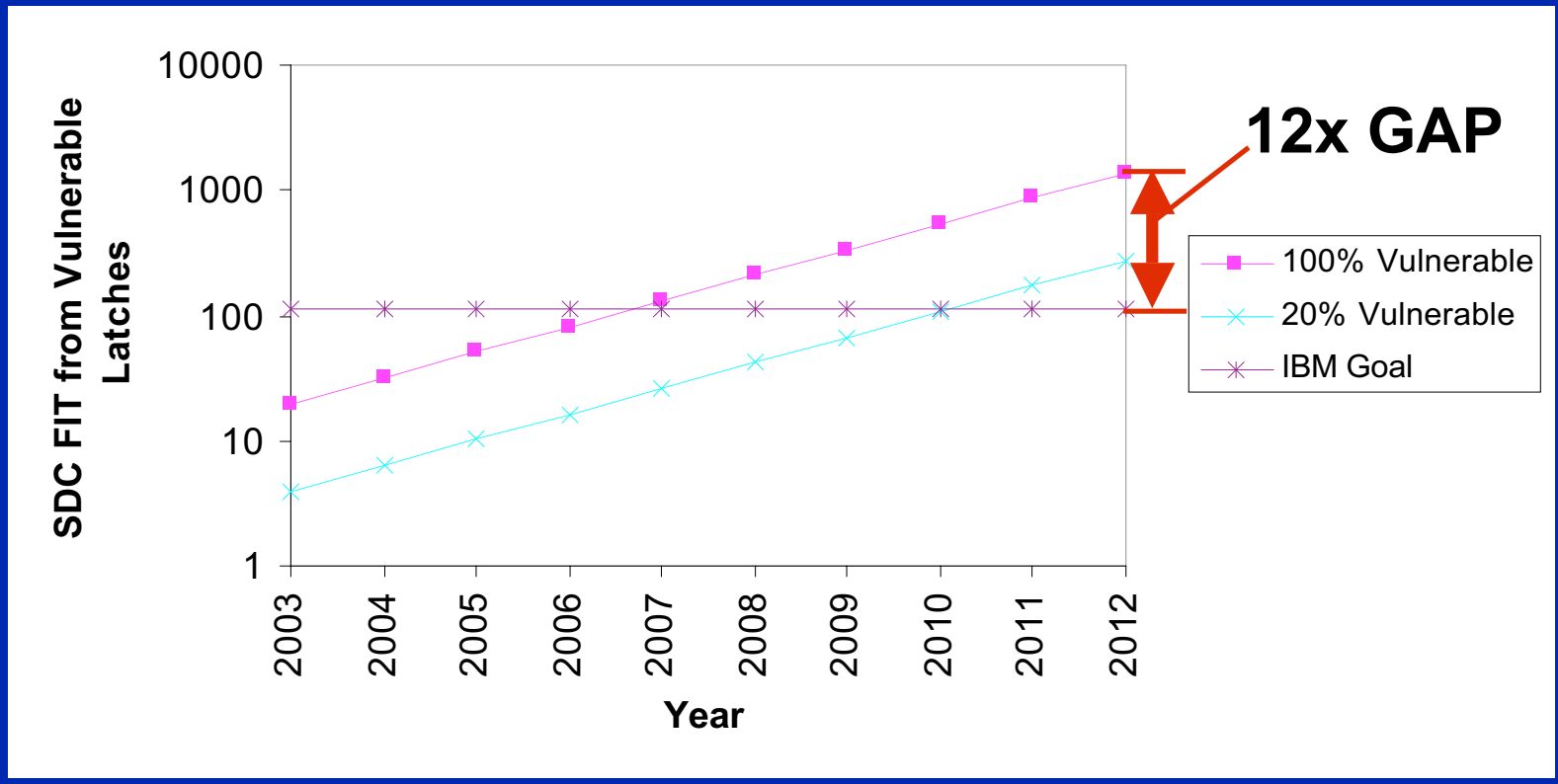
- SRAM cells
  - hold state, varies across structures
- Latches
  - hold state, varies across structures
- Static Logic Gates
  - no clear answer, depends on circuit
- Dynamic Logic
  - similar to latches

- ongoing work in architecture community

Punchline: Simple Conceptual Model

- FIT rate = sum of FIT rate of “vulnerable” bits
- Vulnerable bits (RAM & latch cells)
  - for SDC, this means unprotected bits
- Rule of thumb: vulnerability factor
  - architectural vulnerability factor ~= 20%
  - timing vulnerability factor = 50% for latches & 13% dynamic
- Rule of thumb: raw FIT rate
  - 0.001 – 0.010 FIT/bit (Normand 1996, Tosaka 1996)

Vulnerable Bits Growing with Moore,s Law

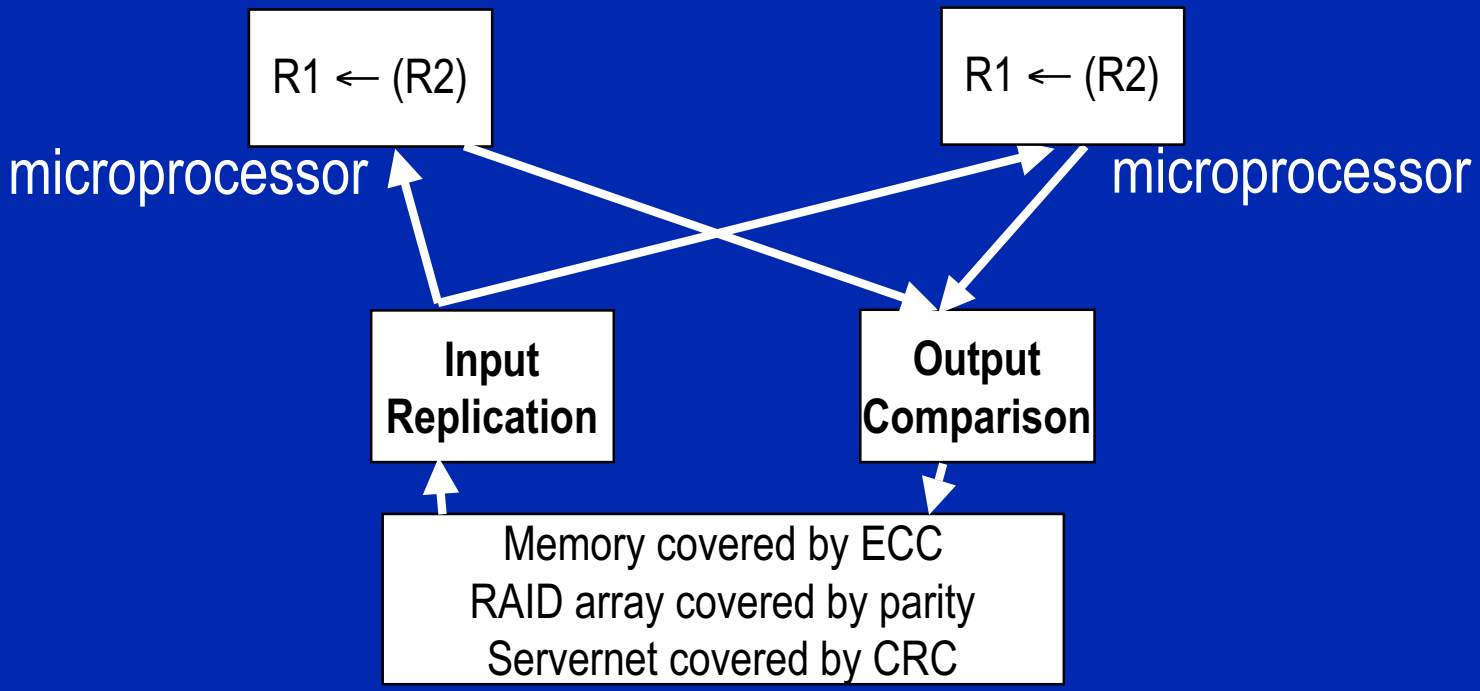


- Fujitsu SPARC has 20% of 200k latches vulnerable in 2003
- Higher SDC FIT from RAM cells, static logic, & dynamic logic
- Higher SDC FIT in multiprocessor systems
  - Gap ~= 100x for 8 processor system!

Outline

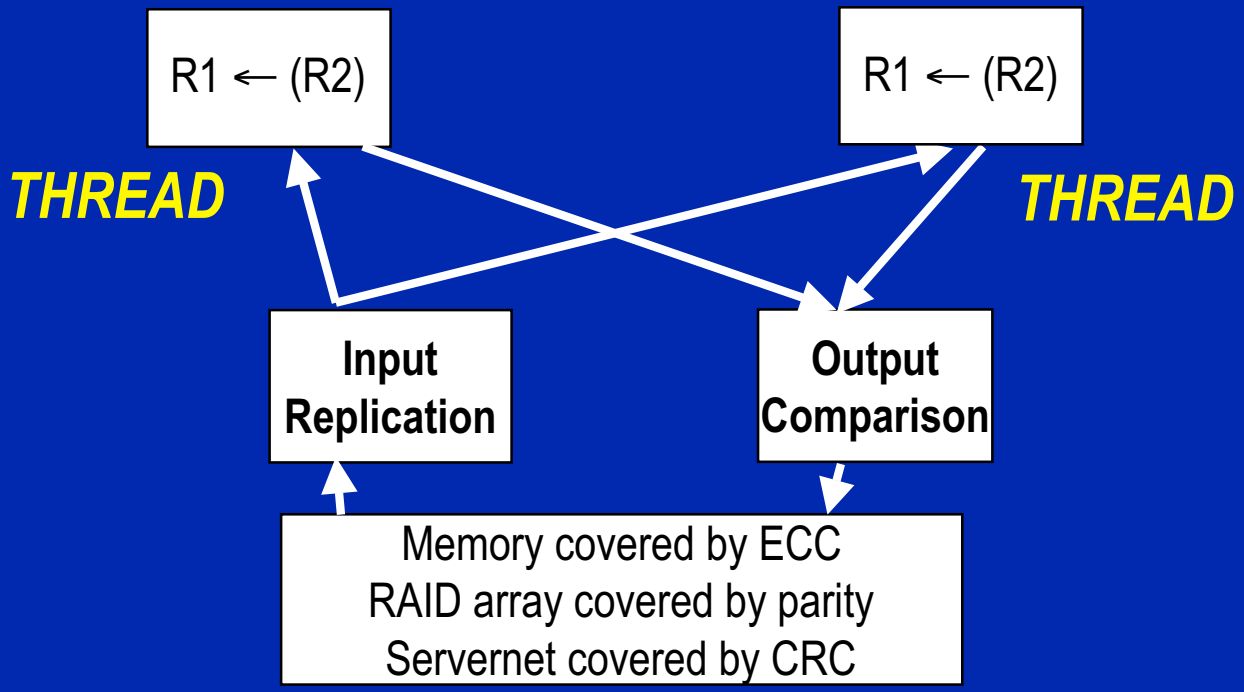
- Faults from Cosmic Rays
- Terminology
- Computing a chip,s Soft Error Rate
- Redundant Multithreading
- Summary

Fault Detection via Lockstepping
(HP Himalaya)



Replicated Microprocessors + Cycle-by-Cycle Lockstepping

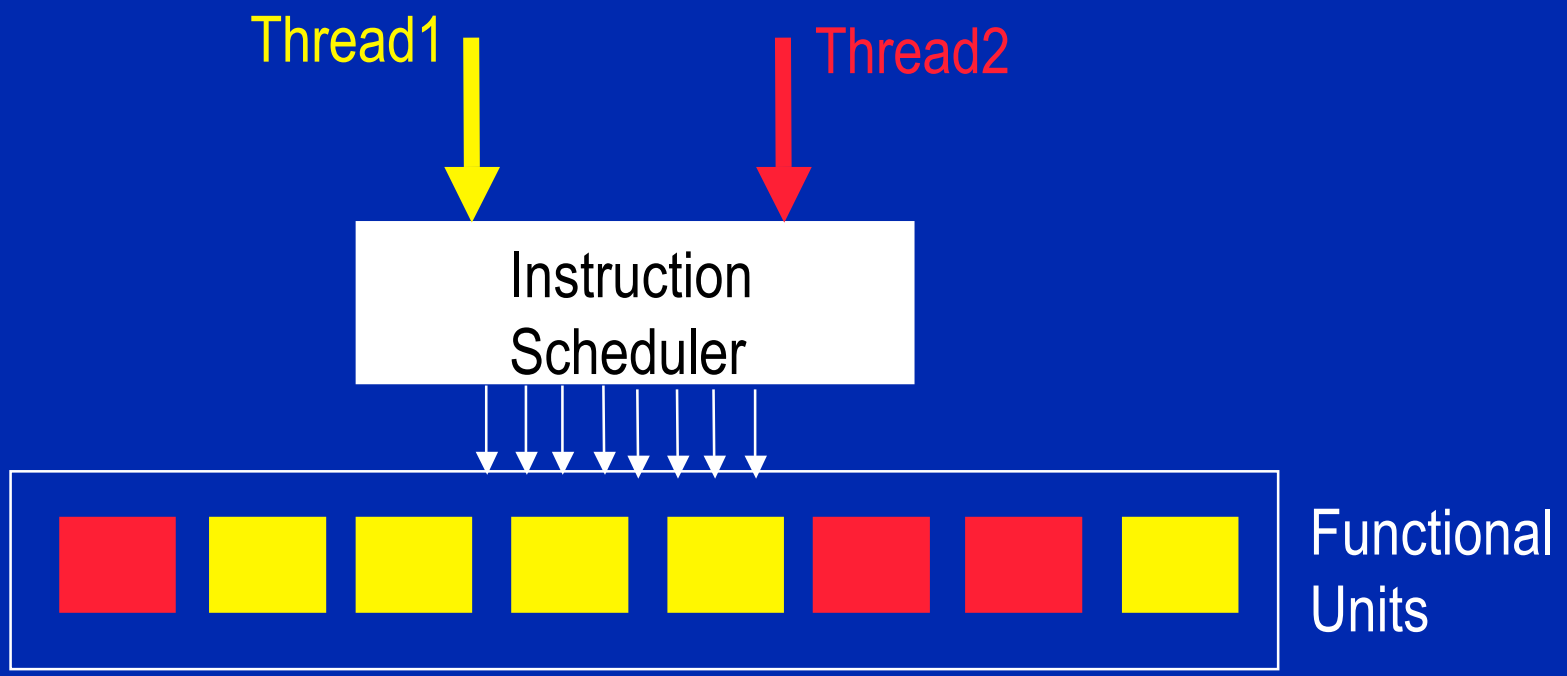
Fault Detection via Simultaneous Multithreading



Threads

Replicated ~~Microprocessors~~ + ~~Cycle-by-Cycle Lockstepping~~ ?

Simultaneous Multithreading (SMT)



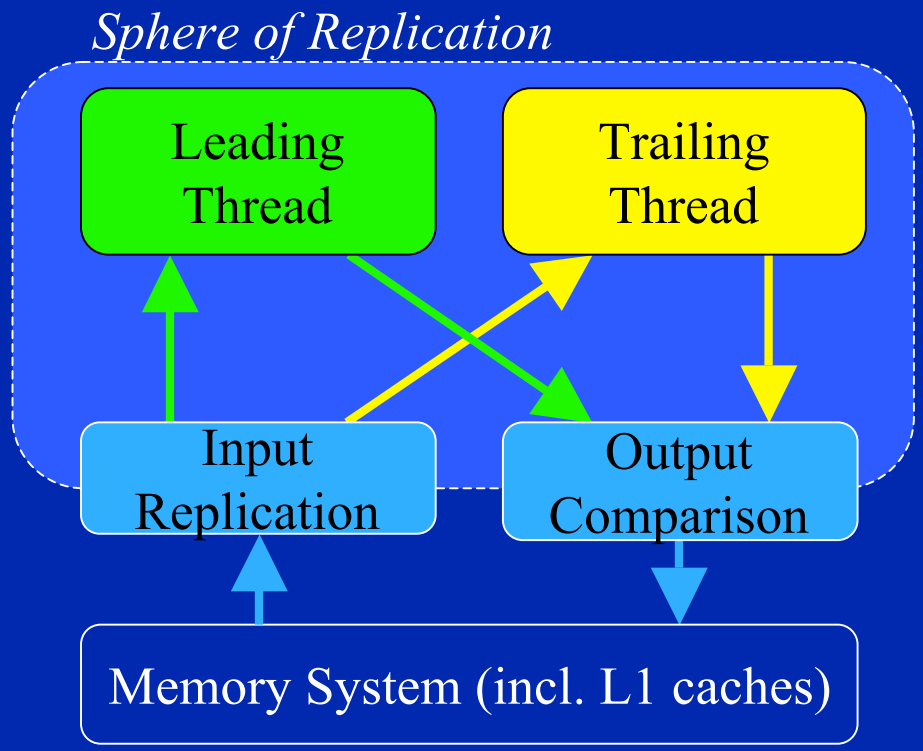
Example: Alpha 21464, Intel Northwood

Redundant Multithreading (RMT)

RMT = Multithreading + Fault Detection (& Recovery)

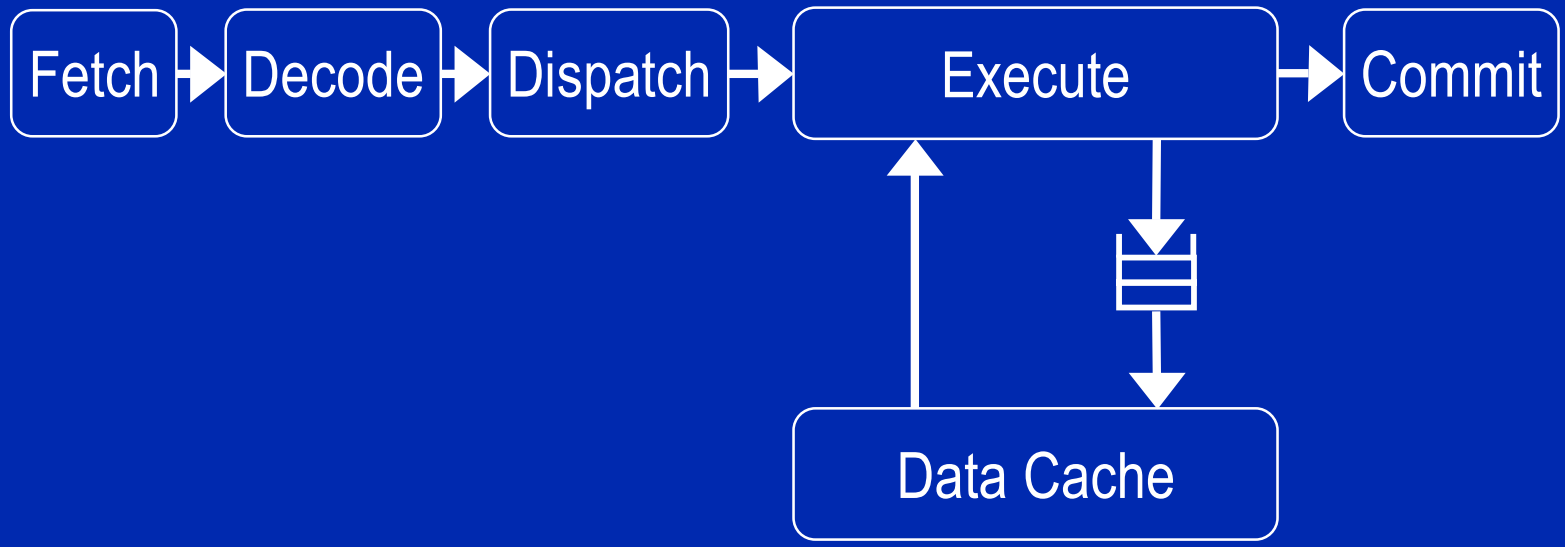
	Multithreading (MT)	Redundant Multithreading (RMT)
Multithreaded Uniprocessor	Simultaneous Multithreading (SMT)	Simultaneous & Redundant Threading (SRT)
Chip Multiprocessor (CMP)	Multiple Threads running on CMP	Chip-Level Redundant Threading (CRT)

Sphere of Replication



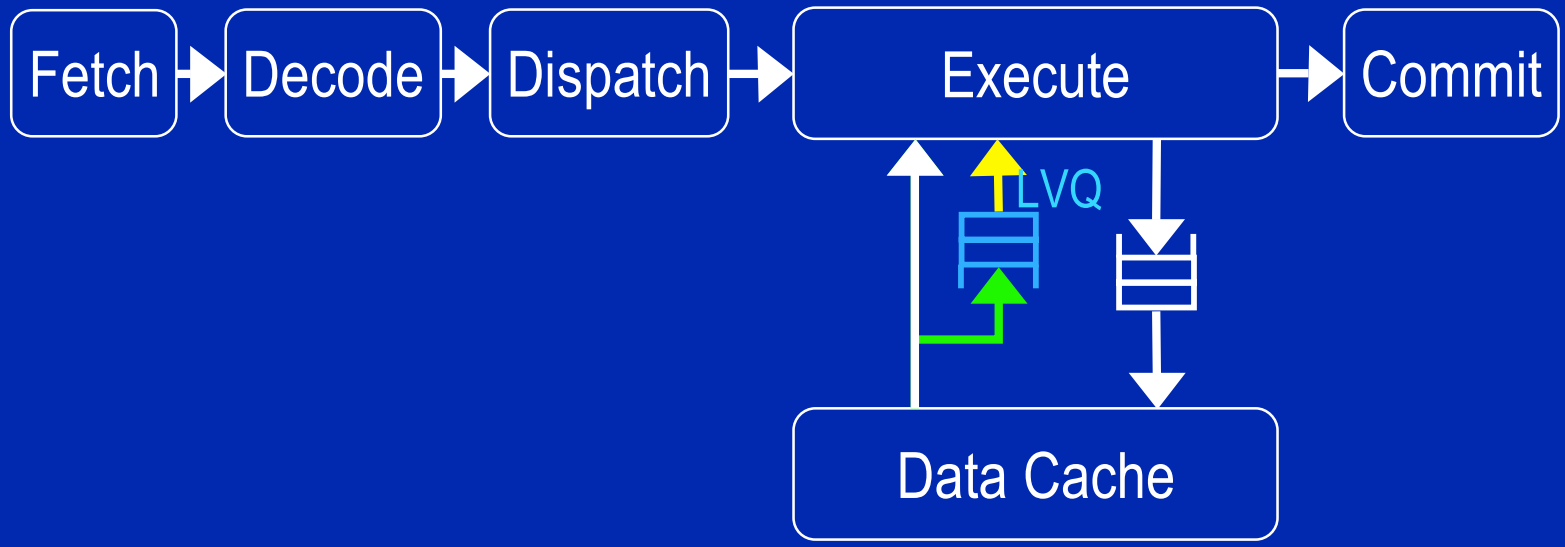
- Two copies of each architecturally visible thread
  - Co-scheduled on SMT core
- Compare results: signal fault if different

Basic Pipeline



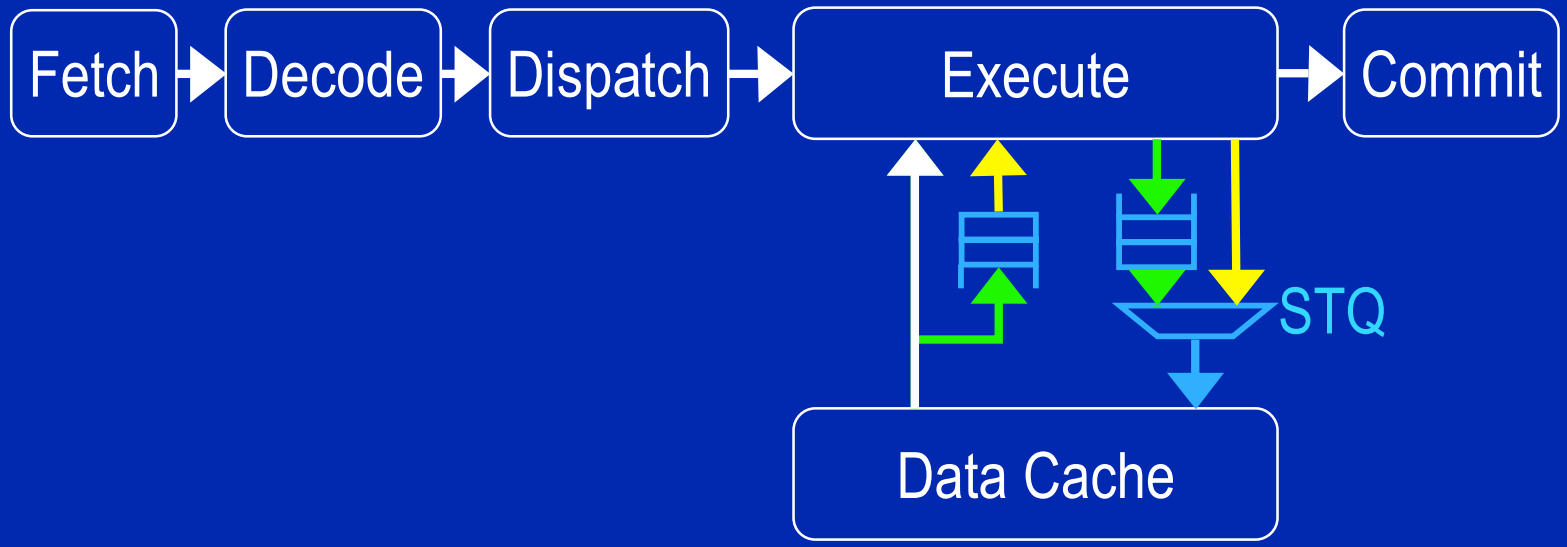
Both leading & trailing threads would go through this pipeline

Load Value Queue (LVQ)



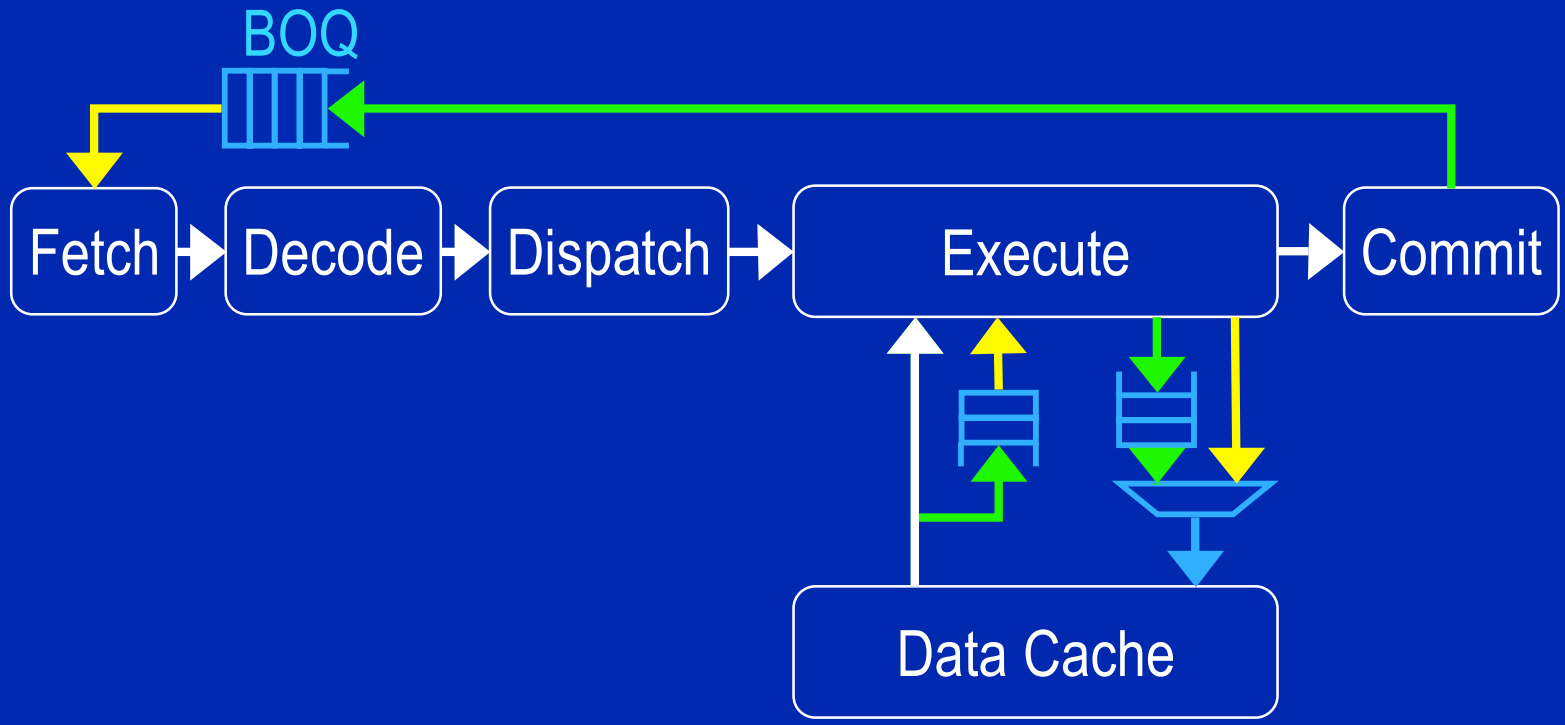
- Load Value Queue (LVQ)
  - Keep threads on same path despite I/O or MP writes
  - Out-of-order load issue possible

Store Queue Comparator (STQ)



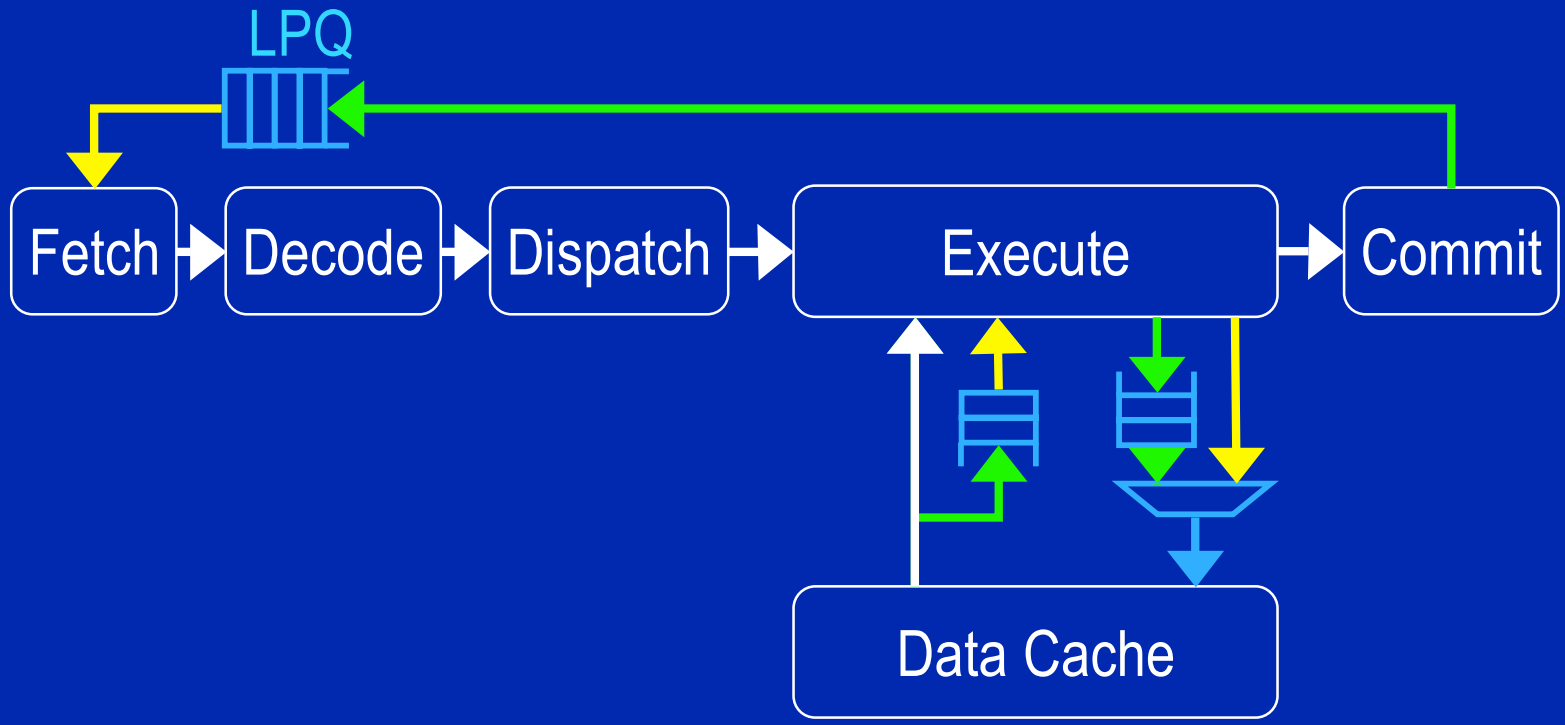
- **Store Queue Comparator**
  - **Compares outputs to data cache**
  - **Catch faults before propagating to rest of system**

Branch Outcome Queue (BOQ)



- **Branch Outcome Queue**
  - **Forward leading-thread branch targets to trailing fetch**
  - **100% prediction accuracy in absence of faults**

Line Prediction Queue (LPQ)

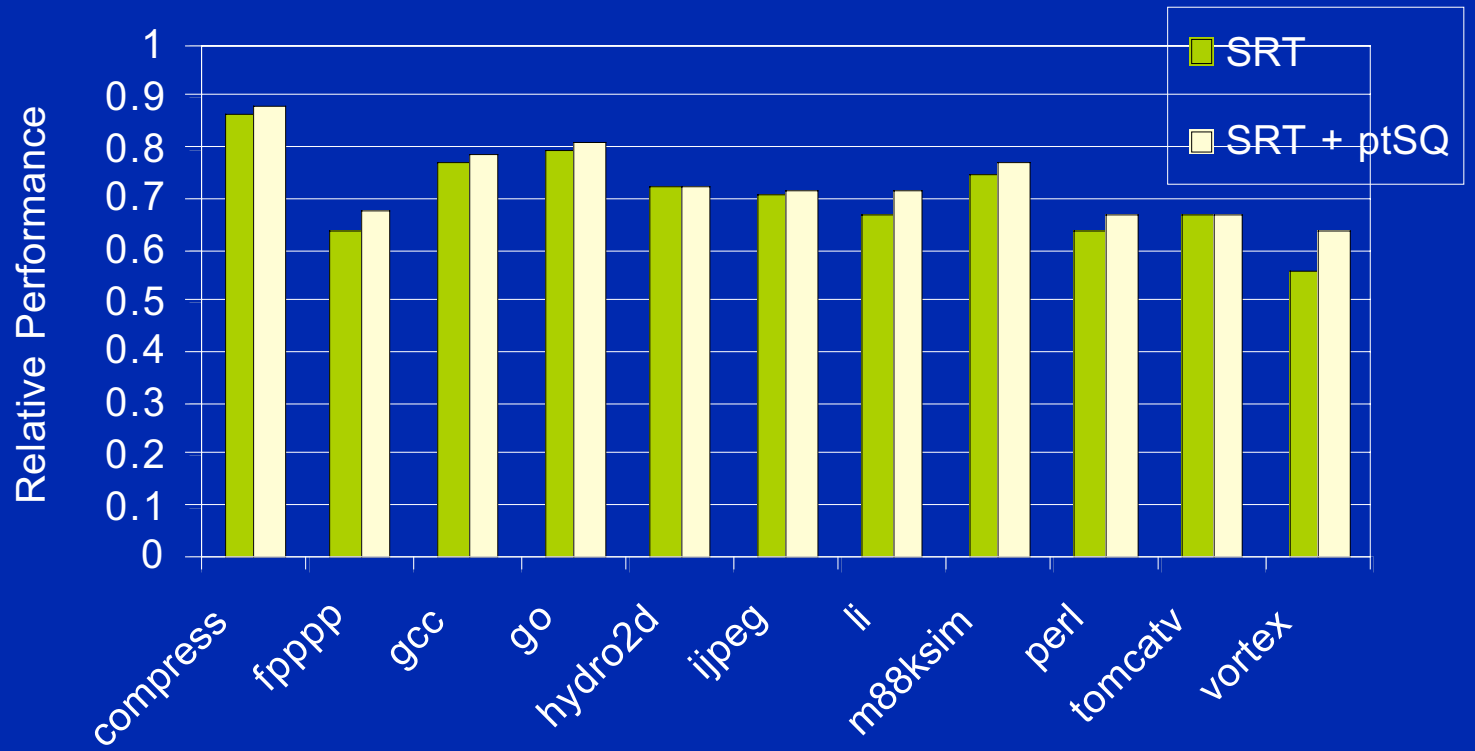


- **Line Prediction Queue**
  - **Alpha 21464 fetches *chunks* using *line predictions***
  - **Chunk = contiguous block of 8 instructions**

SRT Evaluation

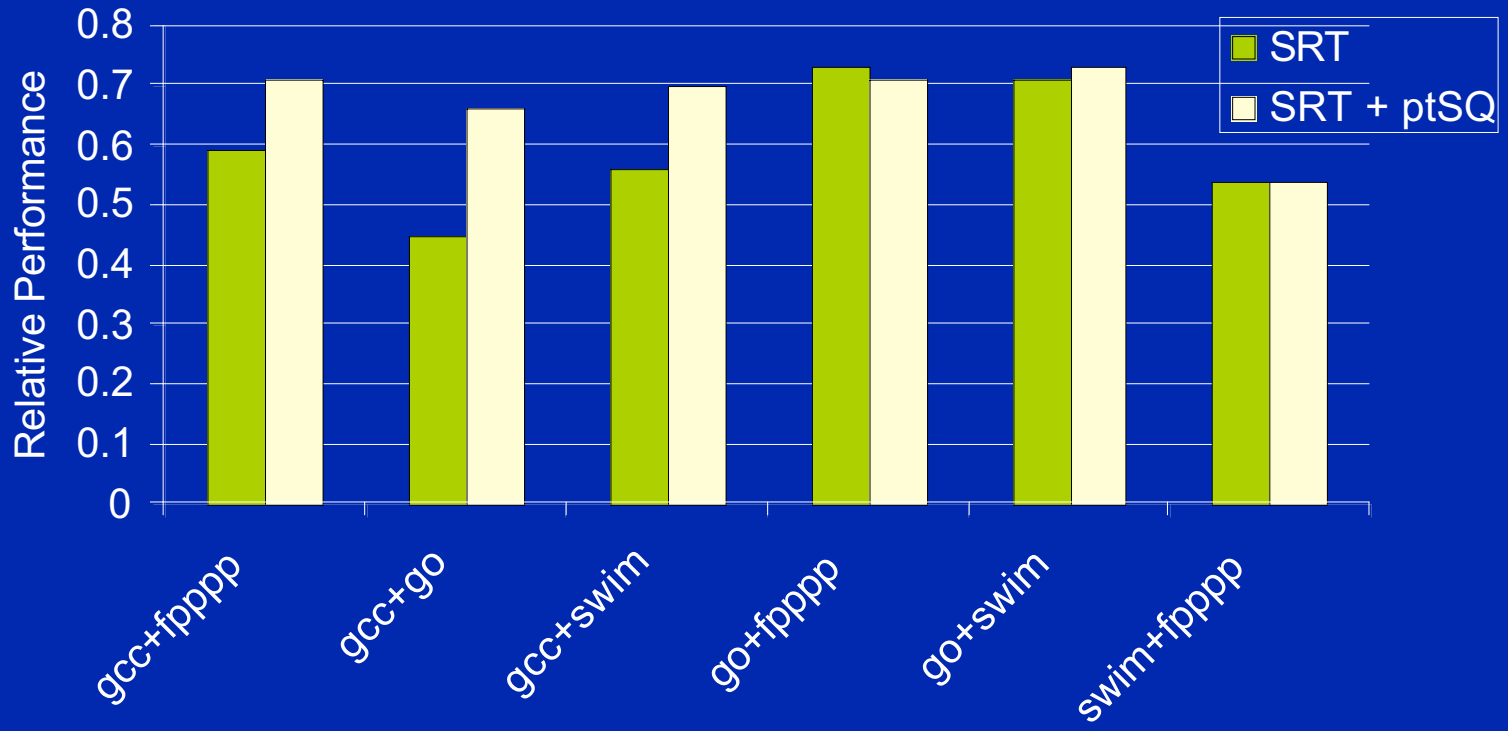
- Used SPEC CPU95, 15M instrs/thread
  - Constrained by simulation environment
  - → 120M instrs for 4 redundant thread pairs
- Eight-issue, four-context SMT CPU
  - 128-entry instruction queue
  - 64-entry load and store queues
    - Default: statically partitioned among active threads
  - 22-stage pipeline
  - 64KB 2-way assoc. L1 caches
  - 3 MB 8-way assoc L2

SRT Performance: One Thread
(Using Alpha 21464-like processor simulator)



- One logical thread → two hardware contexts
- Performance degradation = 30%
- Per-thread store queue buys extra 4%

SRT Performance: Two Threads
(Using Alpha 21464-like processor simulator)



- Two logical threads → four hardware contexts
- Average slowdown increases to 40%
- Only 32% with per-thread store queues

Redundant Multithreading (RMT)

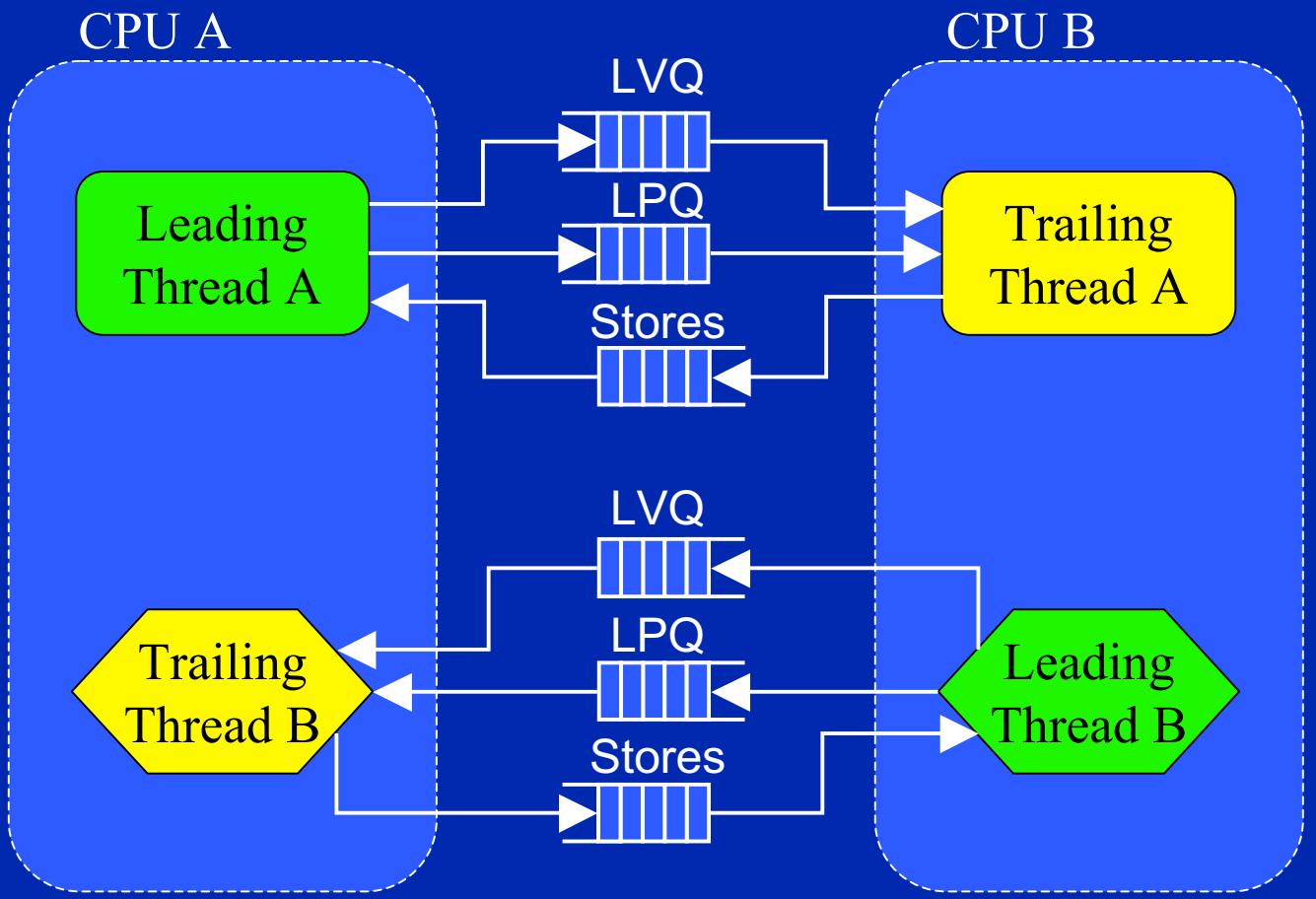
RMT = Multithreading + Fault Detection (& Recovery)

	Multithreading (MT)	Redundant Multithreading (RMT)
Multithreaded Uniprocessor	Simultaneous Multithreading (SMT)	Simultaneous & Redundant Threading (SRT)
Chip Multiprocessor (CMP)	Multiple Threads running on CMP	Chip-Level Redundant Threading (CRT)

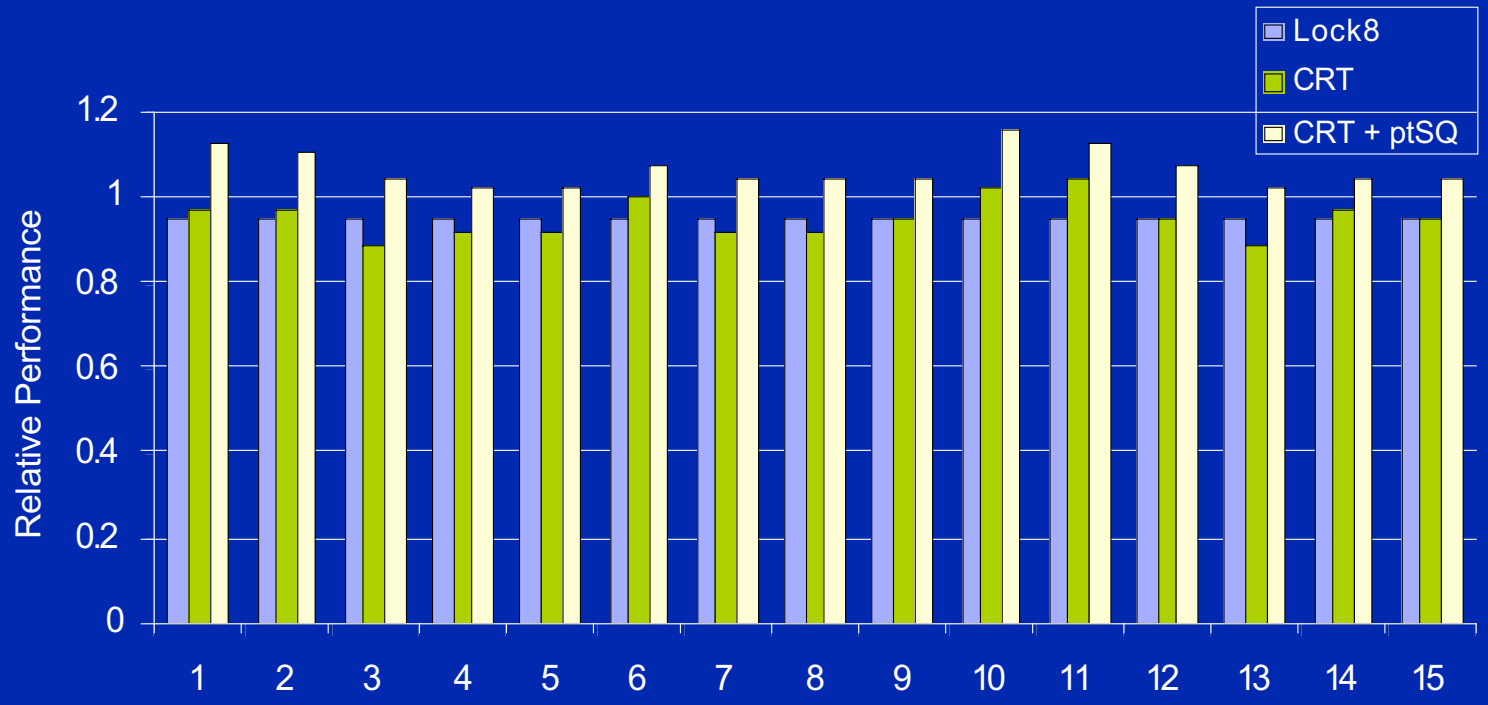
Chip-Level Redundant Threading

- SRT typically more efficient than splitting one processor into two half-size CPUs
- What if you already have two CPUs?
  - IBM Power4, HP PA-8800 (Mako)
- Conceptually easy to run these in lock-step
  - Benefit: full physical redundancy
  - Costs:
    - Latency through centralized checker logic
    - Overheads (misspeculation etc.) incurred twice
- CRT combines best of SRT & lockstepping
  - requires multithreaded CMP cores

Chip-Level Redundant Threading



CRT Performance



- With per-thread store queues, ~13% improvement over lockstepping with 8-cycle checker latency

More Information

- **Publications**
  - S.K. Reinhardt & S.S.Mukherjee, “Transient Fault Detection via Simultaneous Multithreading,” International Symposium on Computer Architecture (ISCA), 2000
  - S.S.Mukherjee, M.Kontz, & S.K.Reinhardt, “Detailed Design and Evaluation of Redundant Multithreading Alternatives,” International Symposium on Computer Architecture (ISCA), 2002
  - Papers available from:
    - http://www.cs.wisc.edu/~shubu
    - http://www.eecs.umich.edu/~stever
- **Patents**
  - Compaq/HP filed eight patent applications on SRT
  - Several more to be filed by Intel in the coming years

Summary

- Soft Errors: real problem today
  - industry seeing this now
- MAJOR problem in next few technology generations
  - problem scales with # chips and Moore,s Law
  - industry will have a hard time making chips reliable
- FACT project
  - working on various aspects of fault measurement, detection, and recovery
  - Redundant Multithreading: example of a cost-effective solution
    - explored implementations in multithreaded processors & CMPs

BACKUPS FOLLOW

Faults, Errors, Failures
(From Pradhan, "Fault-Tolerant Computer System Design")

- **Fault**
  - defect in hardware or software component
  - defect for cosmic ray = upset from high-energy neutron strike
- **Error**
  - manifestation of a fault, resulting in deviation from accuracy
  - faults cause errors (but, not vice versa)
  - a masked fault is not an error!
  - vulnerability factor = fraction of faults that cause errors (Intel term)
- **Failure**
  - non-performance of expected action
  - errors cause failures (but not vice versa)
  - a corrected error doesn,t cause a failure

Three Views of Soft Errors

- **The Architect,s View**
  - *"(Soft) Errors are the crab grass in the lawn of computer design,"* Itanium Architect, Feb. 2003.
  - ➢ Architects don,t want to deal with soft errors
- **The Physicist,s View**
  - *"You can deny physics only for so long,"* Ted Equi, Hewlett-Packard, early 2003.
  - ¬ Technology has no practical solution to completely eliminate soft errors
- **The Pragmatist,s View**
  - *"If a problem has no solution, it may not be a problem, but a **FACT**, not to be solved, but to be coped with over time,"* Shimon Peres, Nobel Laureate 1994.
  - ¬ Inspired the birth of the *FACT (Fault Aware Computing Technology)* project in VSSAD.

References

- **Documented Strikes**
  - (Sun Microsystems) R. Baumann, "Soft Errors in Commercial Semiconductor Technology," 2002 IRPS Tutorial Notes
  - Normand, "Single Event Upset at Ground Level," IEEE Transactions on Nuclear Science, Vol. 43, No. 6, December 1996.
- **Raw soft error rate: 0.001 – 0.010 FIT/bit**
  - Y.Tosaka, S.Satoh, K.Suzuki, T.Suguii, H.Ehara, G.A.Woffinden, and S.A.Wender, "Impact of Cosmic Ray Neutron Induced Soft Errors, on Advanced Submicron CMOS circuits," *VLSI Symposium on VLSI Technology Digest of Technical Papers*, 1996.
  - Normand, "Single Event Upset at Ground Level," IEEE Transactions on Nuclear Science, Vol. 43, No. 6, December 1996.
- **IBM Goals**
  - D.C.Bossen, "CMOS Soft Errors and Server Design," *IEEE 2002 Reliability Physics Tutorial Notes, Reliability Fundamentals*, pp. 121_07.1 – 121_07.6, April 7, 2002.

FIT/bit for SRAM Cells decreasing

- Shivakumar, et al., “Modeling the Effect of Technology Trends on the Soft Error Rate of Combinatorial Logic,” DSN, 2002.
  - FIT/bit decreasing, FIT/chip increasing
- Hareland, et al., “Impact of CMOS Process Scaling and SOI on the soft error rates of logic processes,” 2001 Symposium on VLSI Technlogy Digest of Technical papers
  - FIT/bit decreasing
- R.Baumann, 2002 IRPS Tutorial Notes
  - FIT/bit decreasing because of voltage saturation
  - FIT/bit increasing in products with B10

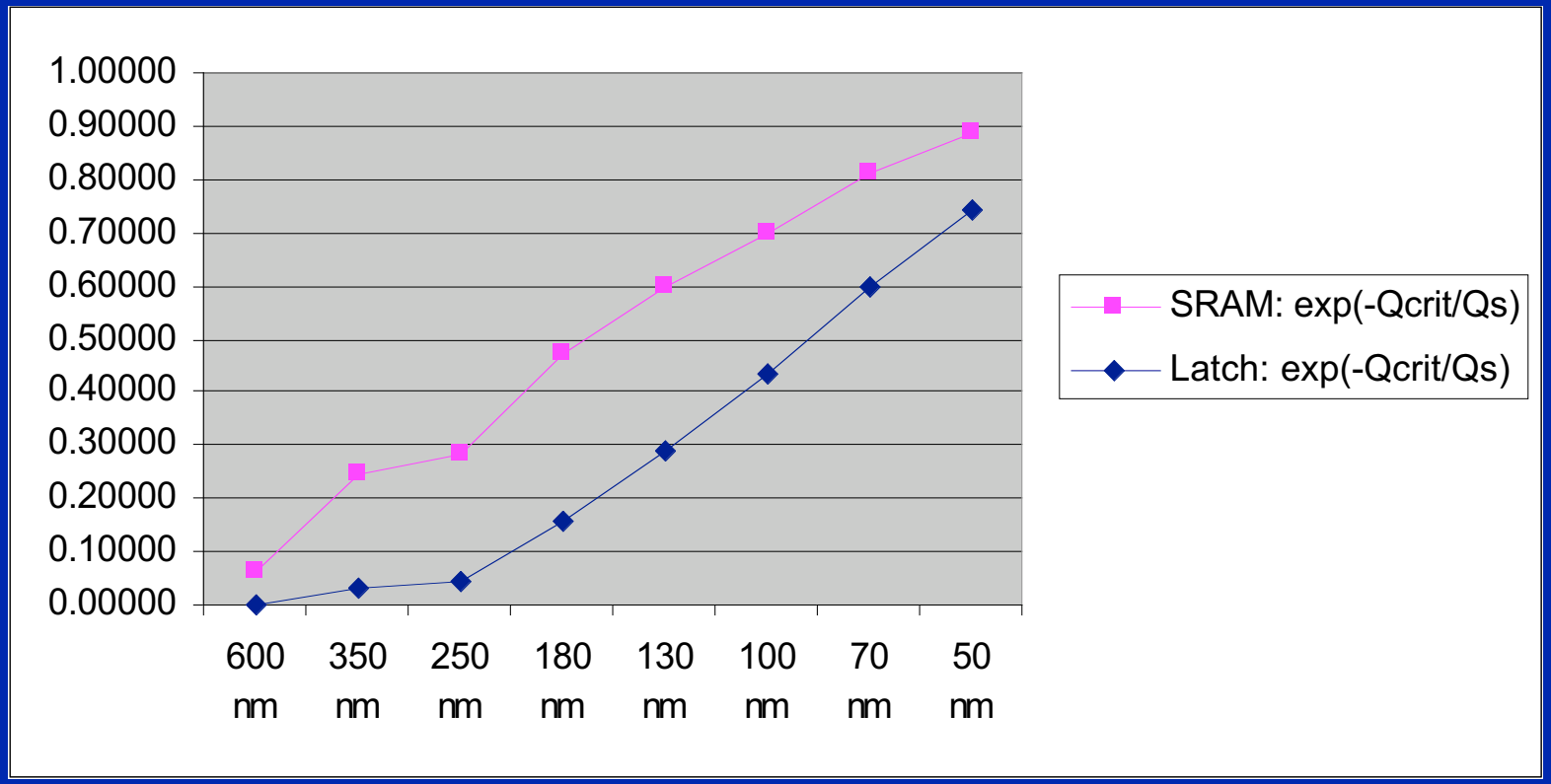
FIT/bit for Latches Constant

- Shivakumar, et al., "Modeling the Effect of Technology Trends on the Soft Error Rate of Combinatorial Logic," DSN, 2002.
  - prediction using models
  - FIT/bit constant (within 2x error range)
- Karnik, et al., "Scaling Trends of Cosmic Rays induced Soft Errors in Static Latches beyond 0.18μ," 2001 Symposium on VLSI Circuits Digest of Technical Papers
  - Neutron beam experiment
  - FIT/bit constant
- λ Internal Intel experiments/data
  - projects that FIT/bit will remain constant (within 2x error bar)

Raw FIT Equation

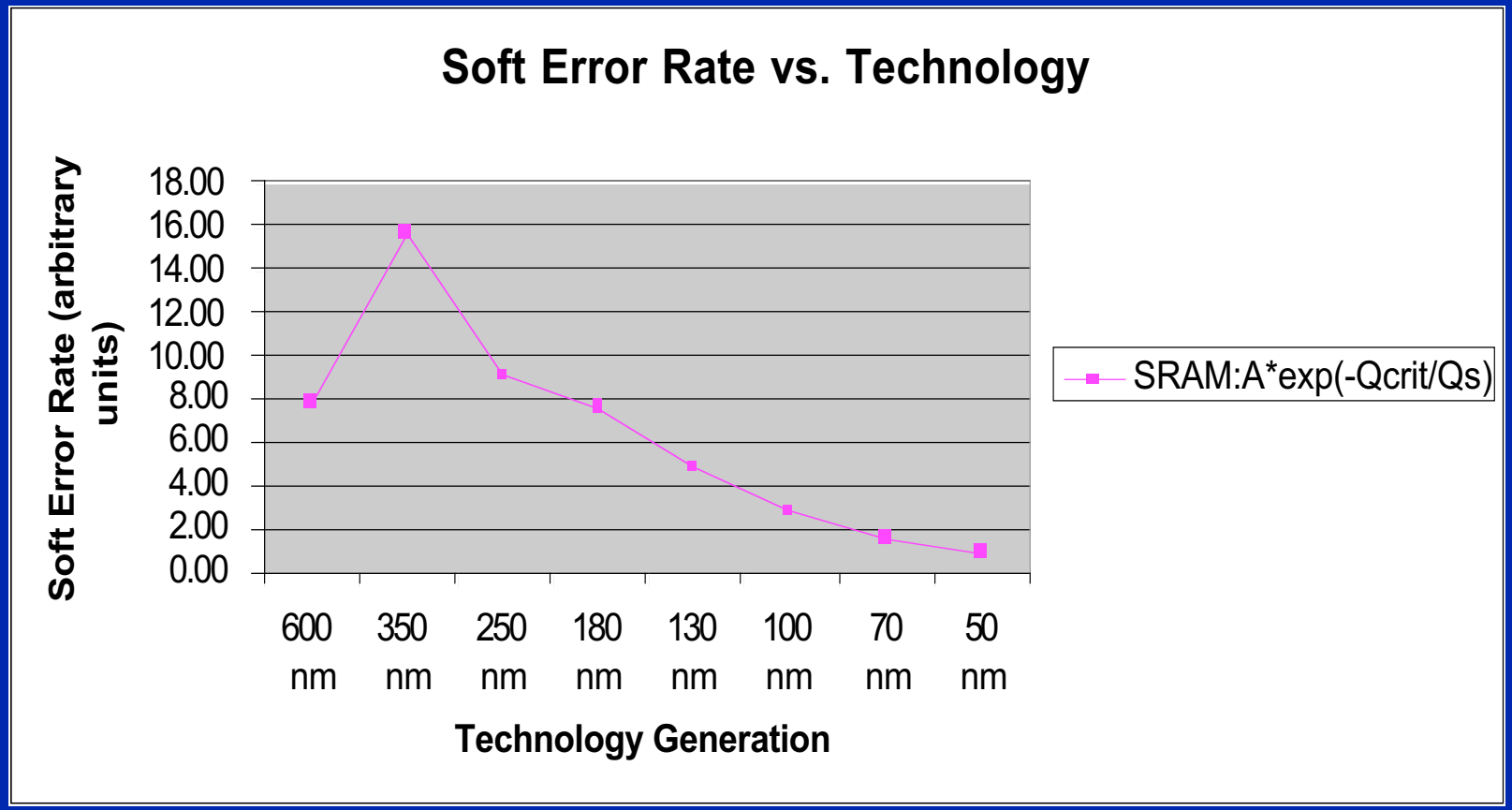
- Raw Neutron FIT rate
  - ∝ Neutron Flux * Area * e $^{-(Qcrit/Qs)}$
- When Qcrit >> Qs
  - exponential dominates
  - we are still in this region
- When Qcrit <= Qs
  - reached saturation
  - area dominates, so FIT/bit will continue to decrease with area

$e^{-Qcrit/Qs}$ trends (Shivakumar et al., DSN 2002)



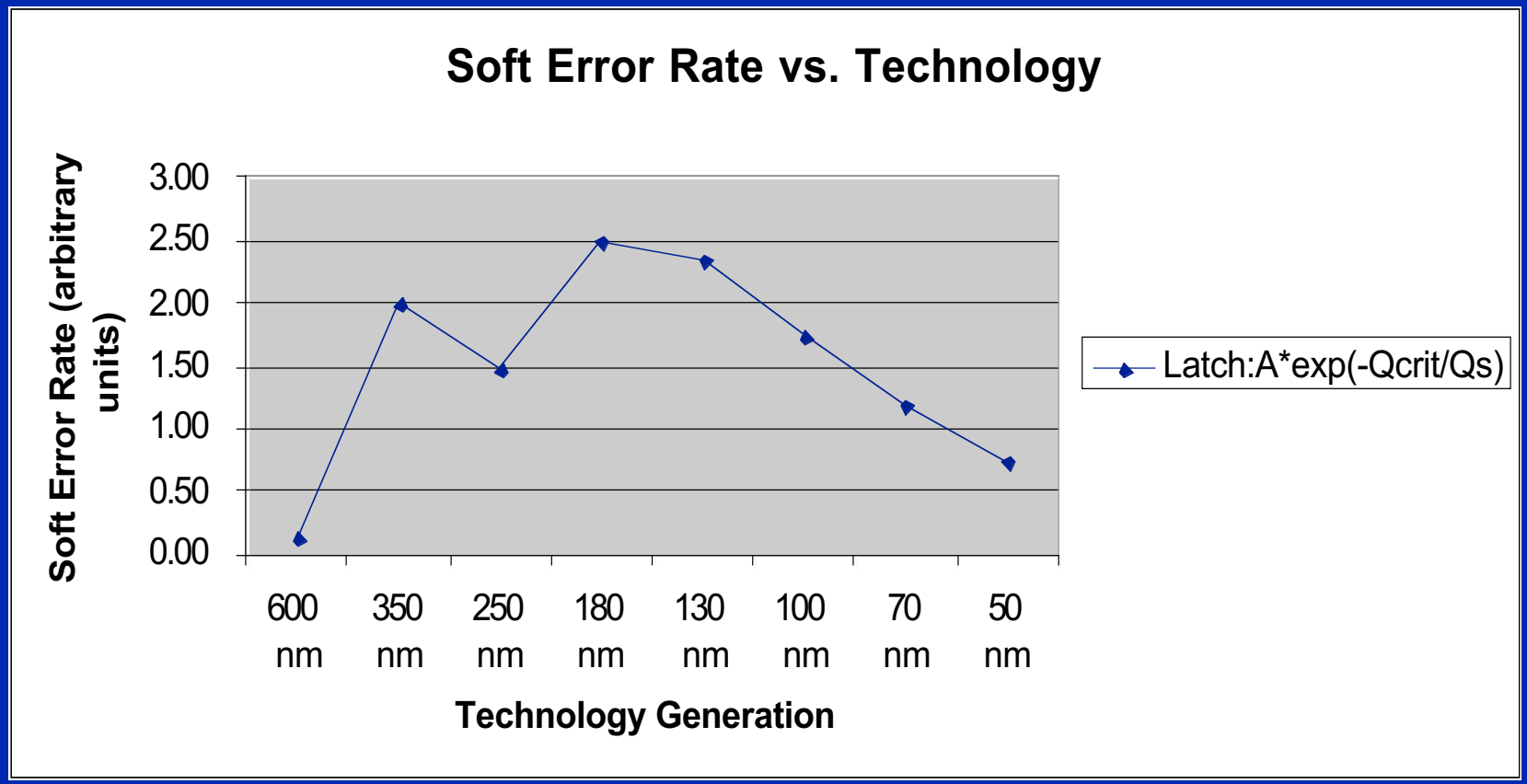
- exp(-Qcrit/Qs) increasing
- area decreasing quadratically

SRAM: FIT/bit decreasing



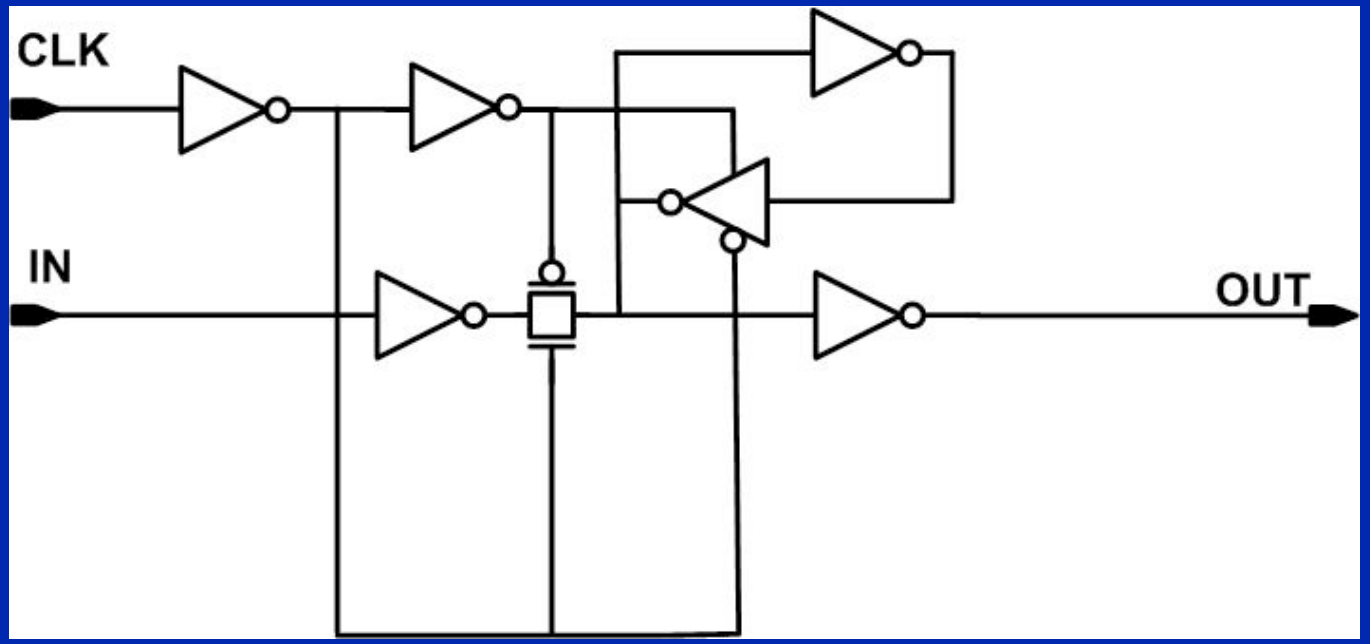
- Source: Shivakumar, et al., DSN 2002

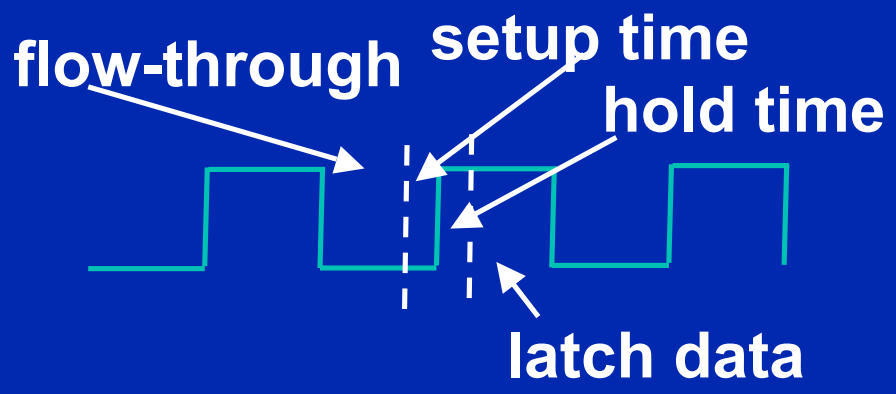
Latch: FIT/bit roughly constant



- Source: Shivakumar, et al., DSN 2002

Timing vulnerability Factor for latches





- Timing vulnerability factor = latch time / clock time ~= 50%

Soft Error Issues

1. **Why is soft error a problem today?**
   - Industry is at the cross-over point
   - Future is worse, IF we don,t do anything
2. **What about system FIT contribution?**
   - System FIT decreased dramatically (e.g., RAID, ECC on DRAM)
   - Large part of system moving on-chip (e.g., memory controller)
3. **Is this a server problem or a desktop problem?**
   - Definitely a server (e.g., data center) problem
   - Desktop problem from IT manager,s point of view
4. **How do software bugs compare to soft error rates?**
   - Limited # of bugs in mature software (e.g., servers, company environment)
   - If we don,t do anything, soft errors will be your dominant failure rate

Energy Spectrum of Cosmic Ray Particles

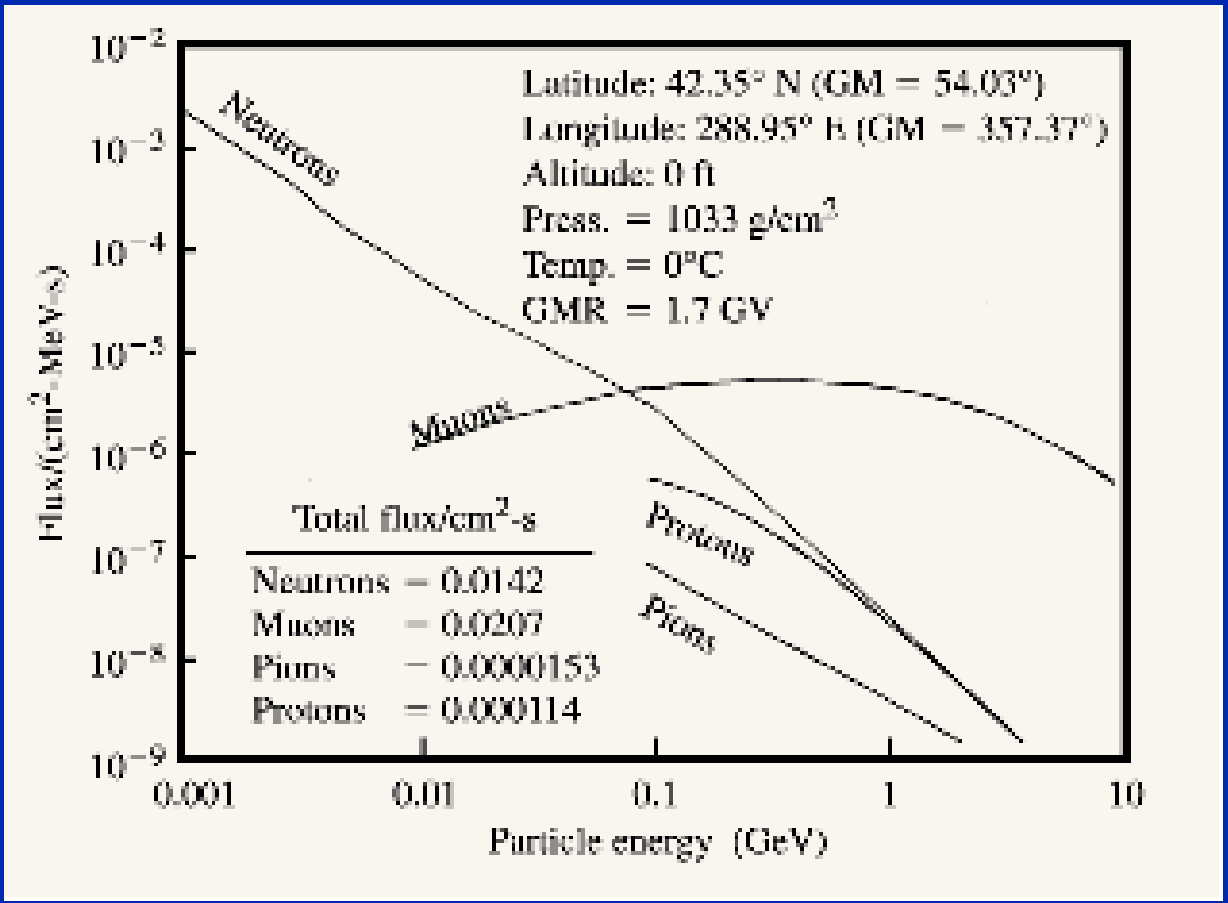


Figure 4, Ziegler, et al., "Terrestrial Cosmic Rays," IBM J. of R. & D., Vol. 40, No. 1, Jan. 1996.

- Neutrons constitute > 96% of cosmic ray particles at sea level
- Higher # of lower energy particles (significant)

Ted Equi, "You can deny physics only for so long!"