

Razor: Dynamic Voltage Scaling Based on Timing Speculation

Todd Austin, David Blaauw, Trevor Mudge
Advanced Computer Architecture Laboratory, University of
Michigan
University of Michigan

Lead PhD Students: Dan Ernst, Nam Sung Kim
Prototype Design Team: Shidhartha Das, Sanjay Pant, Toan Pham, Rajeev Rao
Razor Demo Team: Chris Drake, Seokwoo Lee

Krisztián Flautner
ARM Ltd.



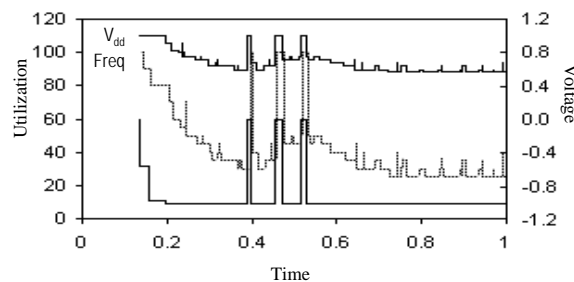
Advanced Computer Architecture Lab
University of Michigan

September 2003

Razor DVS
Austin / Blaauw / Mudge

Voltage Scaling under Dynamic Workloads

- Adapt frequency/voltage to performance demands of workload
 - Software controlled processor speed
 - Lower processor voltage during periods of low operating frequency



- Quadratic reduction in dynamic power and energy
- Super-quadratic reduction in leakage

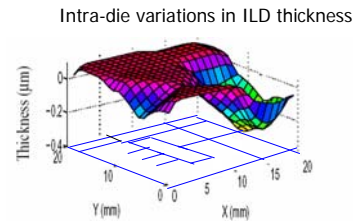


Advanced Computer Architecture Lab
University of Michigan

Razor DVS
Austin / Blaauw / Mudge

Impact of Process Scaling on Power

- Increasing uncertainty with process scaling
 - Inter- and intra-die process variations
 - Temperature variation
 - Power supply drop
 - Capacitive and inductive noise
- Impact on traditional design:
 - Addressing worst-case variation in design requires *large safety margins*
 - Higher energy / lower performance
 - Reduced yield
 - Difficulty in design closure
- Key Observation: worst-case conditions also highly improbable
 - Significant gain for circuits optimized for common case
 - Efficiency mechanisms needed to tolerate infrequent worst-case scenarios

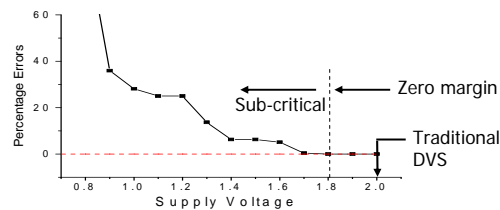


Advanced Computer Architecture Lab
University of Michigan

Razor DVS
Austin / Blaauw / Mudge

Shaving Voltage Margins with Razor

- *Goal*: reduce voltage margins with *in-situ* error detection and correction for delay failures



- Proposed Approach:
 - Tune processor voltage based on error rate
 - Eliminate safety margins, purposely run *below* critical voltage
 - Data-dependent latency margins
 - Trade-off: voltage power savings vs. overhead of correction
- Analogous to wireless power modulation

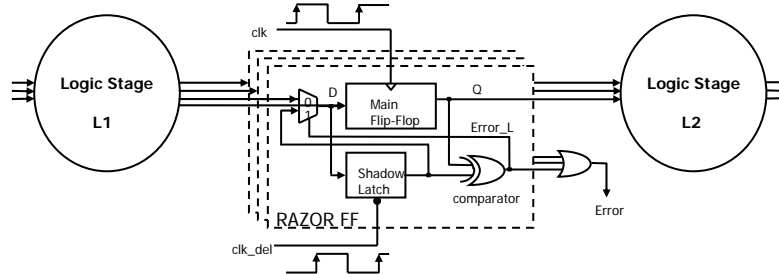


Advanced Computer Architecture Lab
University of Michigan

Razor DVS
Austin / Blaauw / Mudge

Razor Flip-Flop Implementation

- Compare latched data with *shadow-latch* on delayed clock



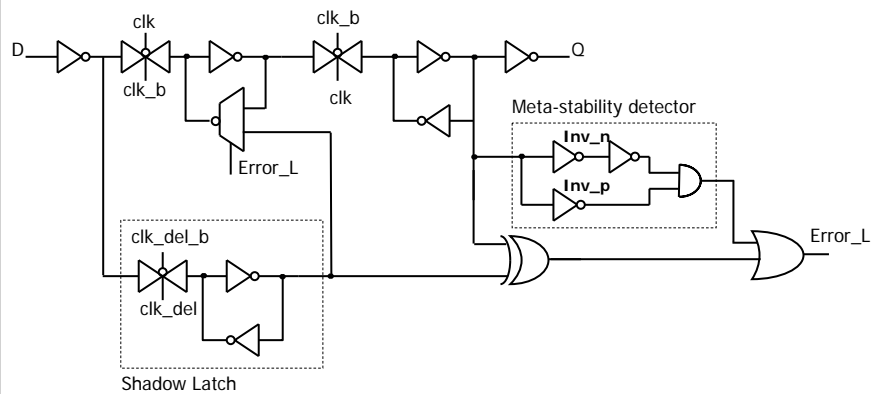
- Upon failure: place data from shadow-latch in main latch
 - Ensure shadow latch always correct using conservative design techniques
- Key design issues:
 - Maintaining pipeline forward progress
 - Short path impact on shadow-latch
 - Power overhead of error detection and correction
 - Recovering pipeline state after errors
 - Meta-stable results in main flip-flop



Advanced Computer Architecture Lab
University of Michigan

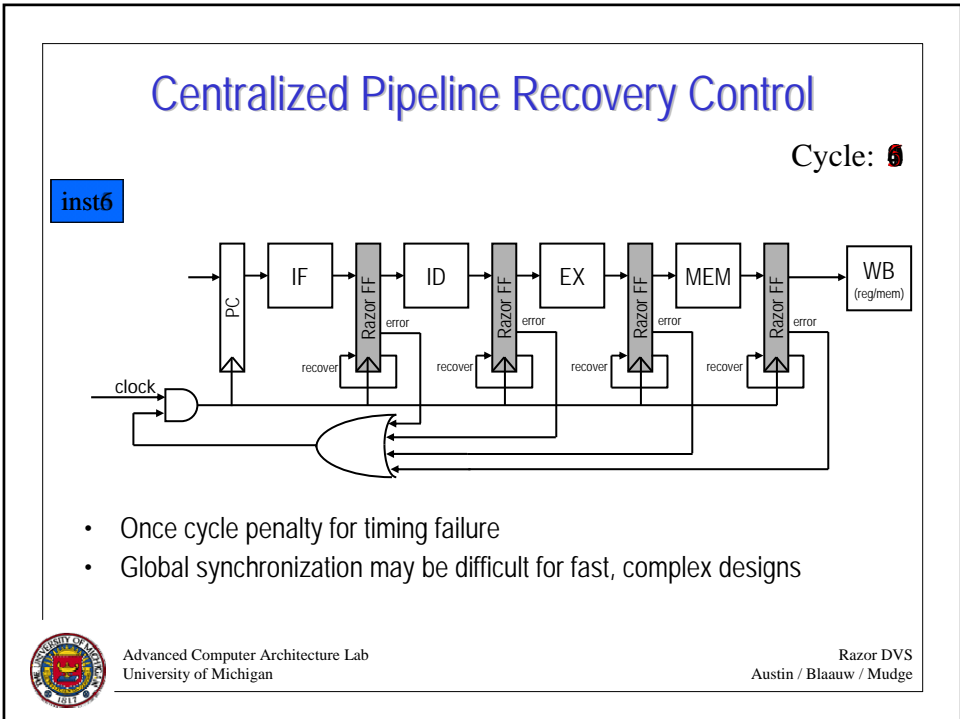
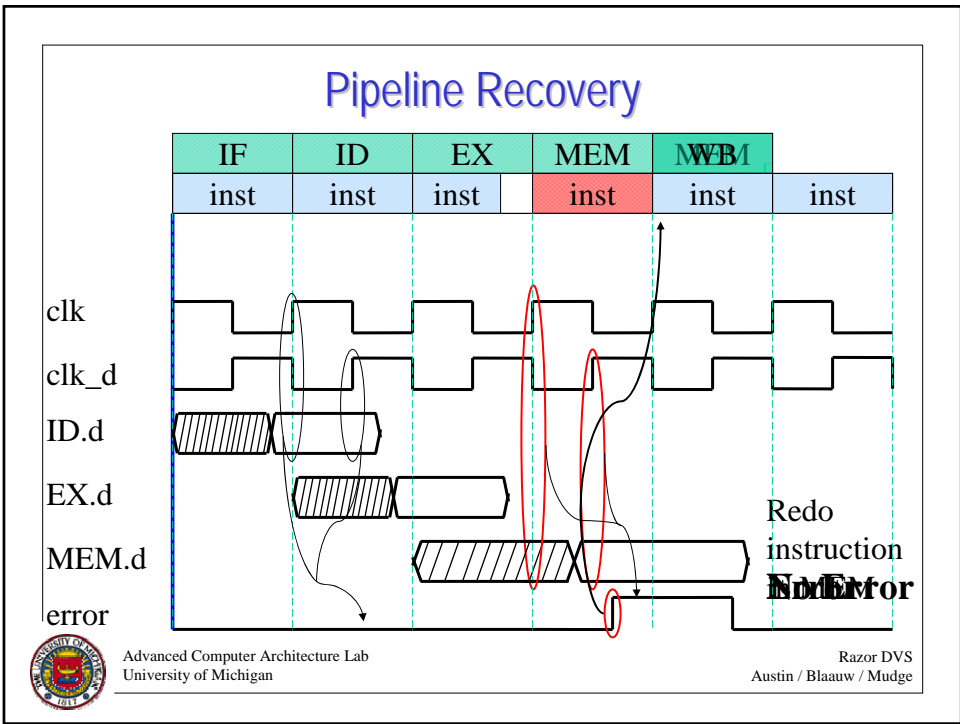
Razor DVS
Austin / Blaauw / Mudge

Razor Flip-Flop Implementation



Advanced Computer Architecture Lab
University of Michigan

Razor DVS
Austin / Blaauw / Mudge

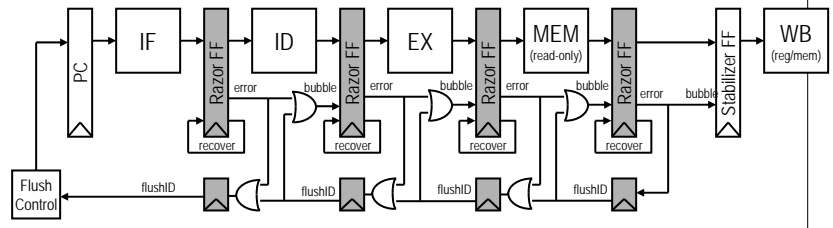


Distributed Pipeline Recovery Control

Cycle: 5

inst4

inst2



- Builds on existing branch / data speculation recovery framework
- Multiple cycle penalty for timing failure
- Scalable design since all recovery communication is local

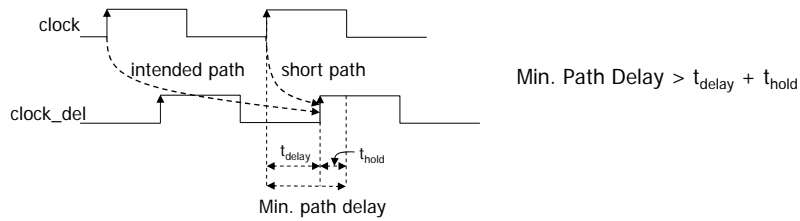


Advanced Computer Architecture Lab
University of Michigan

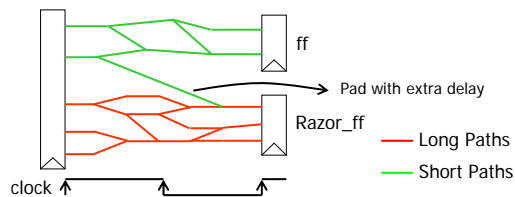
Razor DVS
Austin / Blaauw / Mudge

Short Paths Constraints

- Delayed clock imposes a short-path constraint



- Razor necessary only for latches on slow paths
- Pad fast path for latches with mixed path delays
- Trade-off between DVS headroom and short path constraints

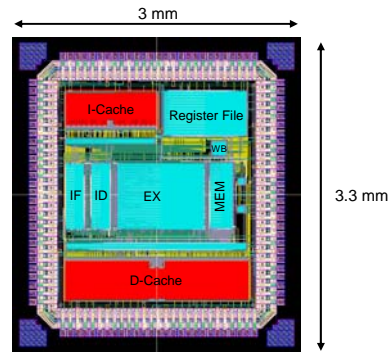


Advanced Computer Architecture Lab
University of Michigan

Razor DVS
Austin / Blaauw / Mudge

Razor I - Prototype Razor Implementation

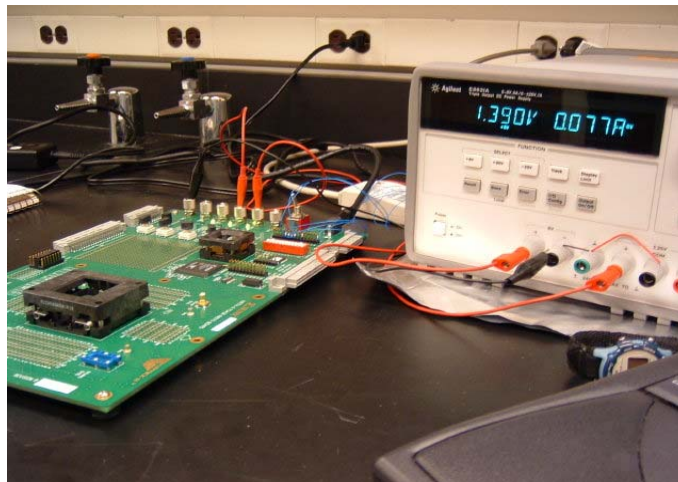
- 4 stage 64-bit Alpha pipeline
 - 200MHz expected operation in 0.18 μ m technology, 1.8V, ~500mW
 - Tunable via software from 200-50MHz, 1.8-1.1V
- Razor overhead:
 - Total of 192 Razor flip-flops out of 2408 total (9%)
 - Error-free power overhead:
 - Razor flip-flops: < 1%
 - Short path buffer: 2.1%
 - Recovery power overhead:
 - Razor latch power overhead: 2% at 10% error rate
 - Additional power overhead due to re-execution of instructions



Advanced Computer Architecture Lab
University of Michigan

Razor DVS
Austin / Blaauw / Mudge

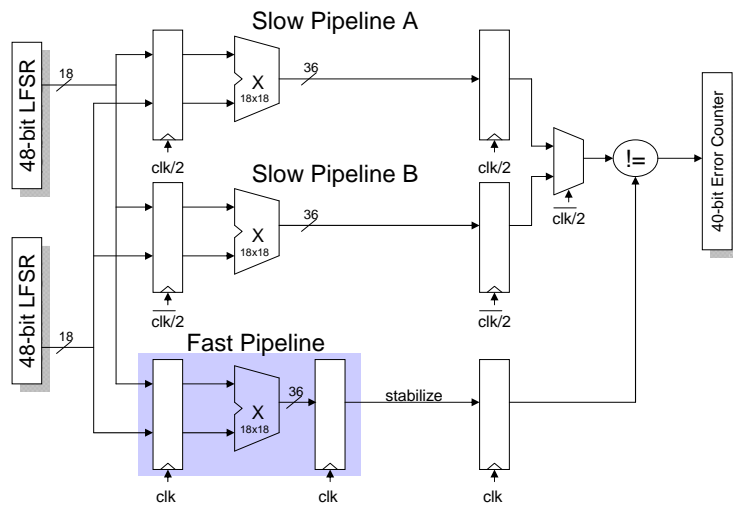
Error-Rate Studies – Hardware Measurement



Advanced Computer Architecture Lab
University of Michigan

Razor DVS
Austin / Blaauw / Mudge

Hardware Measurement Setup



Advanced Computer Architecture Lab
University of Michigan

Razor DVS
Austin / Blaauw / Mudge

Razor Demo

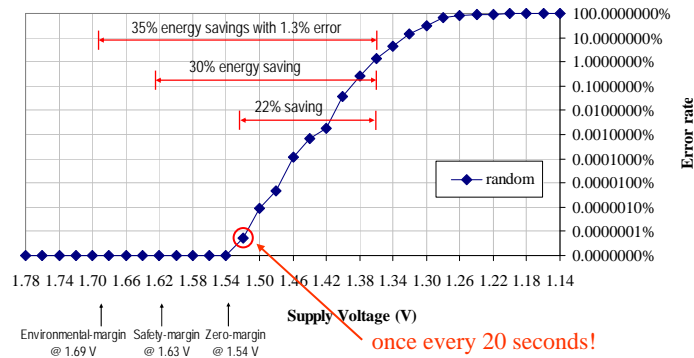


Advanced Computer Architecture Lab
University of Michigan

Razor DVS
Austin / Blaauw / Mudge

Error Rate Studies – Empirical Results

18x18-bit Multiplier Block at 90 MHz and 27 C



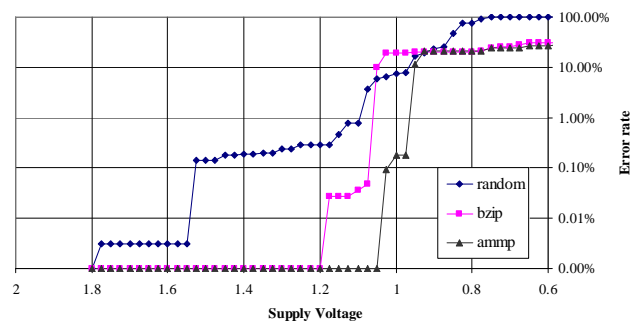
Advanced Computer Architecture Lab
University of Michigan

Razor DVS
Austin / Blaauw / Mudge

Error Rate Studies – SPICE-Level Simulations

- Based on a SPICE-level simulations of a Kogge-Stone adder

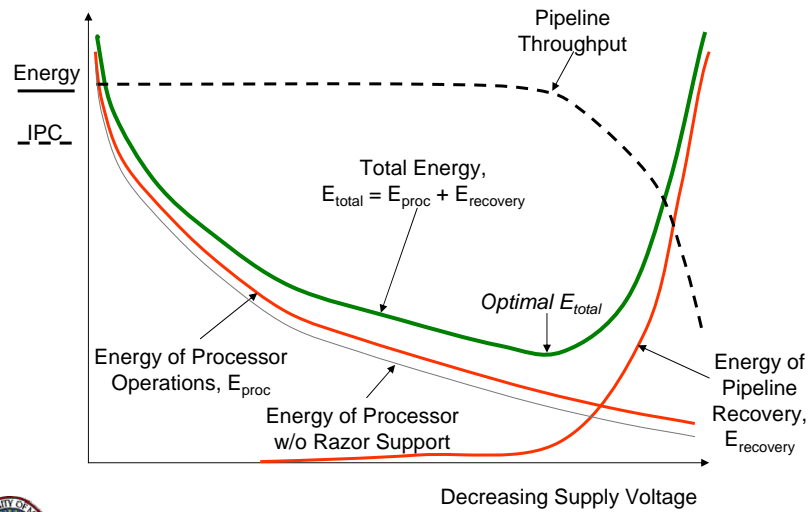
Kogge-Stone Adder at 870 MHz and 27 C



Advanced Computer Architecture Lab
University of Michigan

Razor DVS
Austin / Blaauw / Mudge

Effects of Razor DVS



Advanced Computer Architecture Lab
University of Michigan

Razor DVS
Austin / Blaauw / Mudge

Simulation Methodology

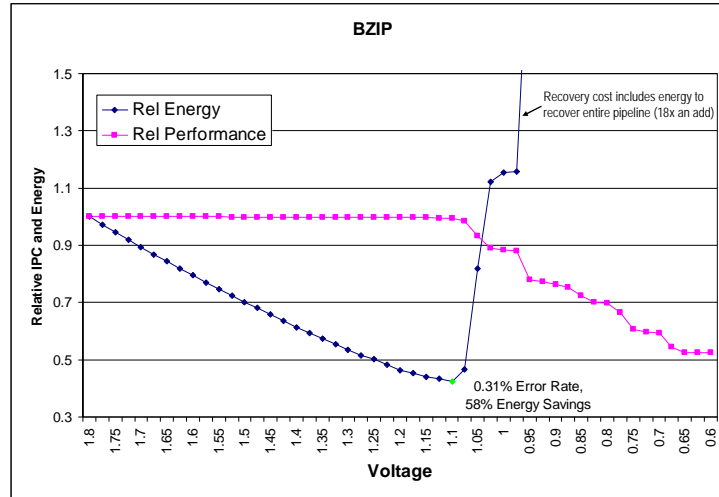
- Challenge: instruction latency depends circuit evaluation latency
 - May vary with changes in stage inputs, stage logic, voltage, temperature...
- *Dynamic timing simulation* combines architectural/circuit simulation
 - SimpleScalar/Alpha architectural-level simulation
 - Gate-level simulation of per-stage logic blocks
 - Logic block model describes cells, local and global interconnect
 - Cells characterized with SPICE at varied slew/cap-load/voltage
 - Each cycle, circuit simulator evaluates delay of each stages' logic block
 - Based on actual instruction inputs from architectural simulator
- Initial implementation utilized a hand-generated EX-stage circuit model
 - Effort ongoing to automate extraction/decomposition/integration into SimpleScalar



Advanced Computer Architecture Lab
University of Michigan

Razor DVS
Austin / Blaauw / Mudge

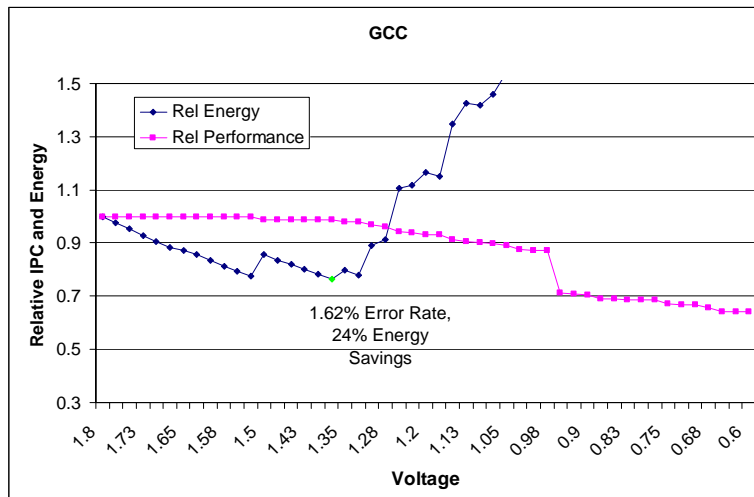
EX-Stage Analysis – Optimal Voltage Sweep



Advanced Computer Architecture Lab
University of Michigan

Razor DVS
Austin / Blaauw / Mudge

EX-Stage Analysis – Optimal Voltage Sweep



Advanced Computer Architecture Lab
University of Michigan

Razor DVS
Austin / Blaauw / Mudge

Simulation Analysis – Energy-Optimal Voltage

Program	Optimal V_{dd}	Error Rate	% Energy Reduced	% IPC Reduced
bzip	1.1	0.31%	57.6%	0.70%
crafty	1.175	0.41%	50.5%	0.60%
eon	1.3	1.21%	34.4%	1.24%
gap	1.275	1.15%	30.1%	2.49%
gcc	1.375	1.62%	23.7%	1.47%
gzip	1.3	1.03%	35.6%	0.41%
mcf	1.175	0.67%	48.7%	0.00%
parser	1.2	0.61%	47.9%	0.29%
twolf	1.275	2.67%	30.7%	0.31%
vortex	1.3	0.53%	42.8%	0.14%
vpr	1.075	0.01%	64.2%	0.00%
<i>Average</i>			42.4%	

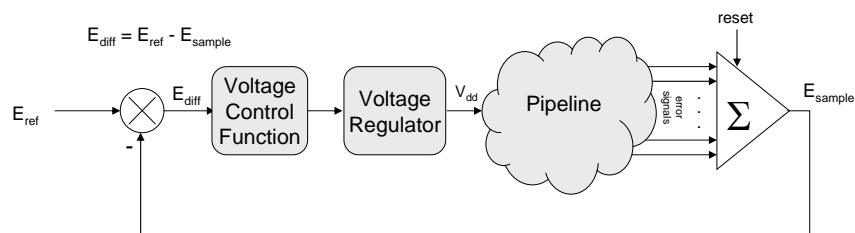
- Simulator only models ALU in EX stage of pipeline



Advanced Computer Architecture Lab
University of Michigan

Razor DVS
Austin / Blaauw / Mudge

Supply Voltage Control System



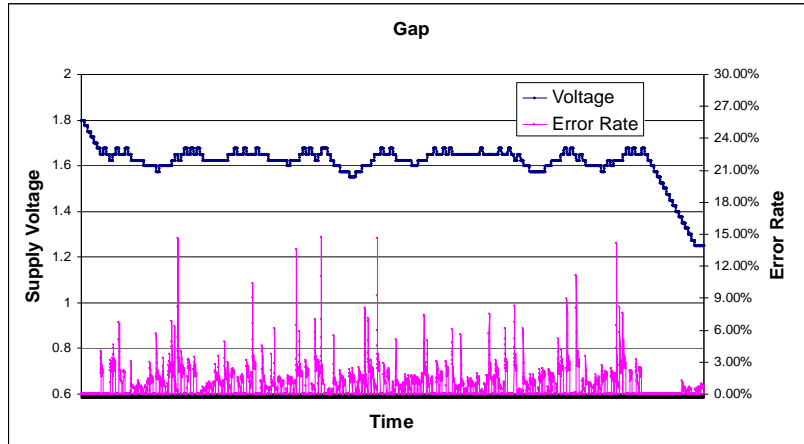
- Current design utilizes a very simple *proportional* control function
 - Control algorithm implemented in software



Advanced Computer Architecture Lab
University of Michigan

Razor DVS
Austin / Blaauw / Mudge

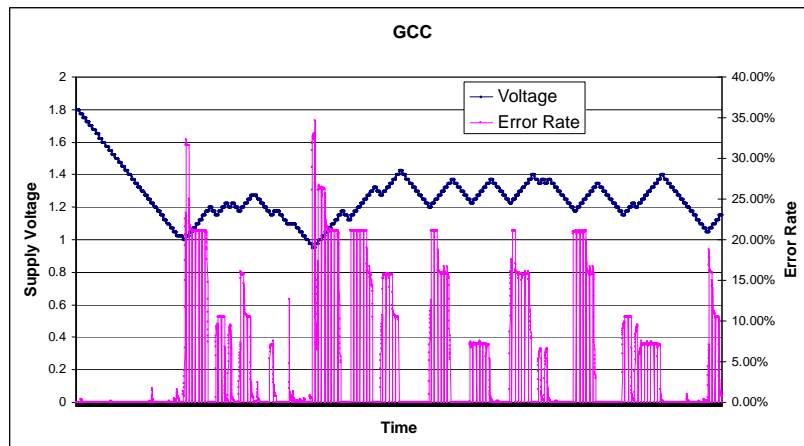
Simulation Analysis – Razor DVS Execution



Advanced Computer Architecture Lab
University of Michigan

Razor DVS
Austin / Blaauw / Mudge

Simulation Analysis – Razor DVS Execution



Advanced Computer Architecture Lab
University of Michigan

Razor DVS
Austin / Blaauw / Mudge

Simulation Analysis – Razor DVS Performance

Program	DVS		Fixed
	% Energy Reduced	% IPC Reduced	% Energy Reduced
bzip	54.5%	4.13%	57.6%
crafty	54.8%	1.78%	50.5%
eon	30.4%	0.78%	34.4%
gap	12.9%	2.14%	30.1%
gcc	31.3%	5.88%	23.7%
gzip	44.6%	1.27%	35.6%
mcf	36.9%	0.47%	48.7%
parser	53.0%	1.94%	47.9%
twolf	20.4%	0.06%	30.7%
vortex	49.1%	1.07%	42.8%
vpr	63.6%	1.66%	64.2%
Average	41.0%		42.4%

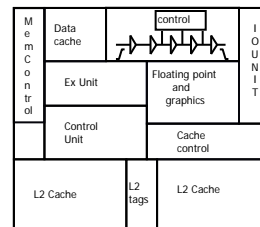


Advanced Computer Architecture Lab
University of Michigan

Razor DVS
Austin / Blaauw / Mudge

Other Approaches to Dynamic Voltage Scaling

- Traditional DVS
 - Valid voltage / delay combinations “blessed” at design time
 - Approach leaves a significant amount of energy “on the table”
 - Temperature, process, data, and safety margins placed on voltage
- Slack detector – automatic tuning
 - National/ARM's *Intelligent Energy Management* (IEM)
 - Processor voltage automatically tuned to external ambient conditions
 - Inverter chain designed to track most restrictive critical path, margin still required
- DSP power / reliability / QoS trade-offs
 - Shanbhag @ UIUC
 - Voltage overscaling approach that limits noise impacts



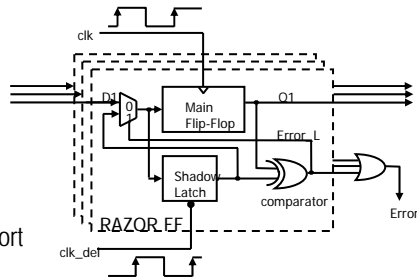
Advanced Computer Architecture Lab
University of Michigan

Razor DVS
Austin / Blaauw / Mudge

Conclusions

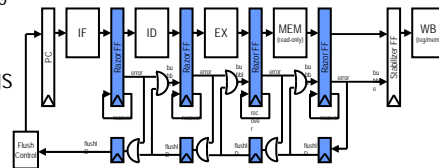
- *In-situ* detection/correction of timing errors

- Tune processor voltage based on error rate
- Eliminate process, temperature, and safety margins (tune for near-zero error rate)
- Purposely run *below* critical voltage to capture *data-dependent latency margins*



- Implemented with architecture/circuit support

- Double-sampling metastability-tolerant Razor flip-flops validate pipeline results
- Pipeline initiates recovery after circuit timing errors, no voltage/clock re-tuning needed



- Trade-off: supply voltage power savings vs. overhead of correction



Advanced Computer Architecture Lab
University of Michigan

Razor DVS
Austin / Blaauw / Mudge

Future Directions

- Research opportunities

- Razor for caches/memory and control logic
- Voltage control algorithms, especially per-stage tuning
- Typical-case energy optimized designs (instead of worse-case latency optimized)
- Turnkey application of Razor technology

- Prototype design, fabrication, evaluation

- Razor I – Q4 2003 – Razor'ized combinational logic, global tuning
- Razor II – Q3 2004 – Razor'ized caches and control logic, per-stage tuning

- Other applications

- Single-event upset (SEU) protection using Razor error detection/re-execution
- Over-clocking for performance improvement (2x shown among hobbyists)

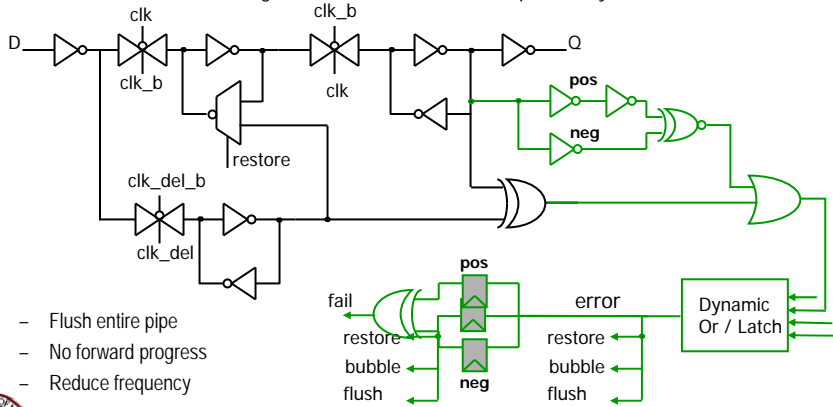


Advanced Computer Architecture Lab
University of Michigan

Razor DVS
Austin / Blaauw / Mudge

More Details on Meta-Stability

- Sub-critical operation invites meta-stability
 - Meta-stability detector itself can become meta-stable
 - double latch error signal to obtain sufficient small probability



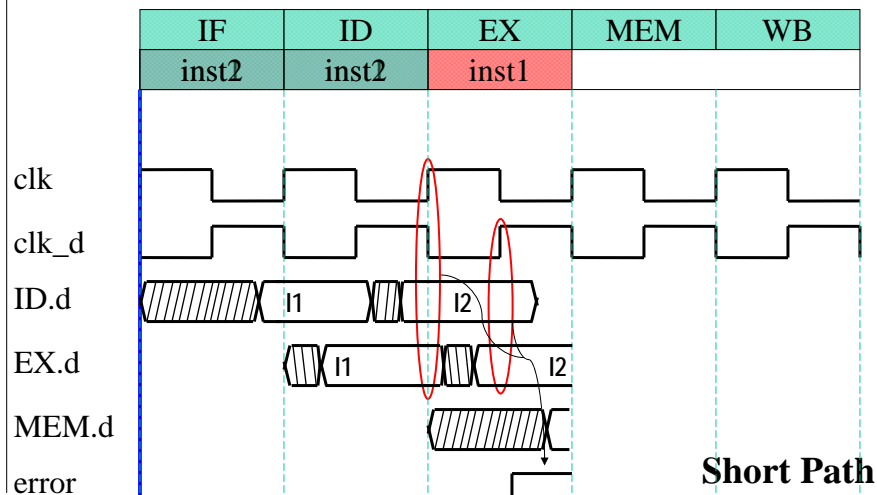
- Flush entire pipe
- No forward progress
- Reduce frequency



Advanced Computer Architecture Lab
University of Michigan

Razor DVS
Austin / Blaauw / Mudge

Short Path Failure



Short Path



Advanced Computer Architecture Lab
University of Michigan

Razor DVS
Austin / Blaauw / Mudge