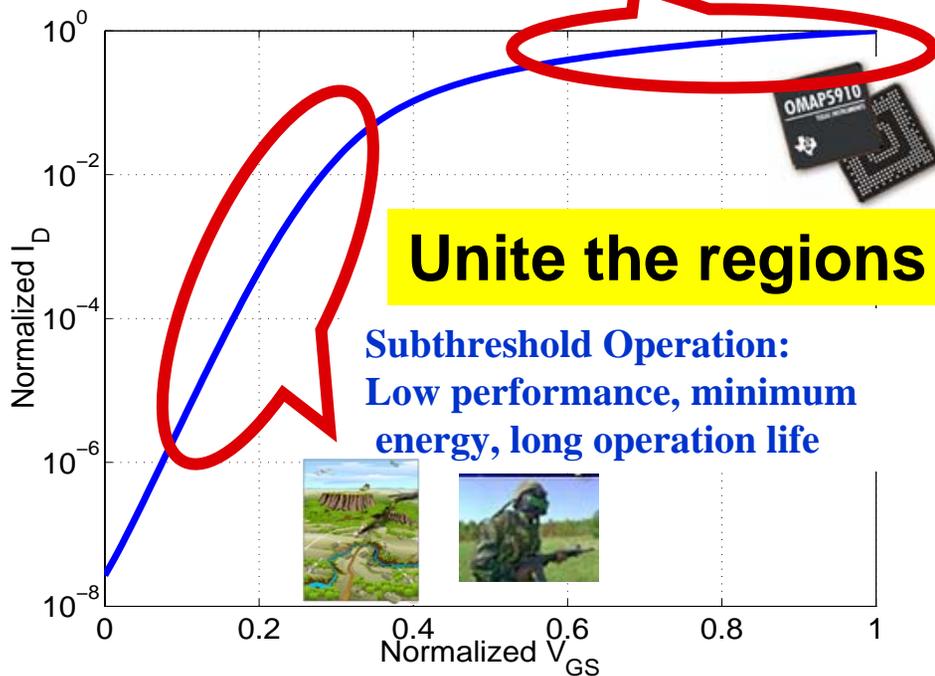


Energy Starved Electronics (ESE)



Dr. Robert Reuss
DARPA/MTO

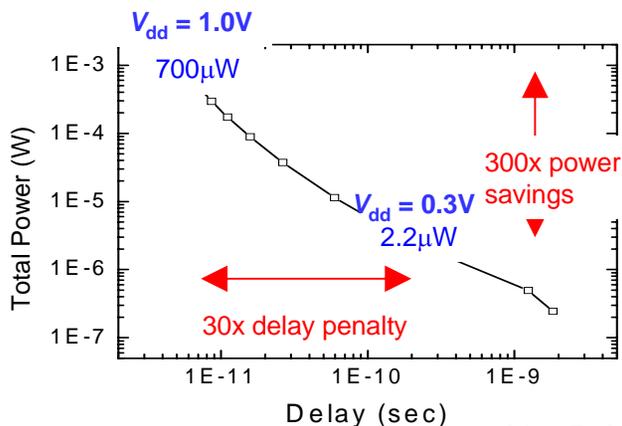
Normal Operation, Strong Inversion : fast, high-energy, high performance



From EETimes Feb. 2005

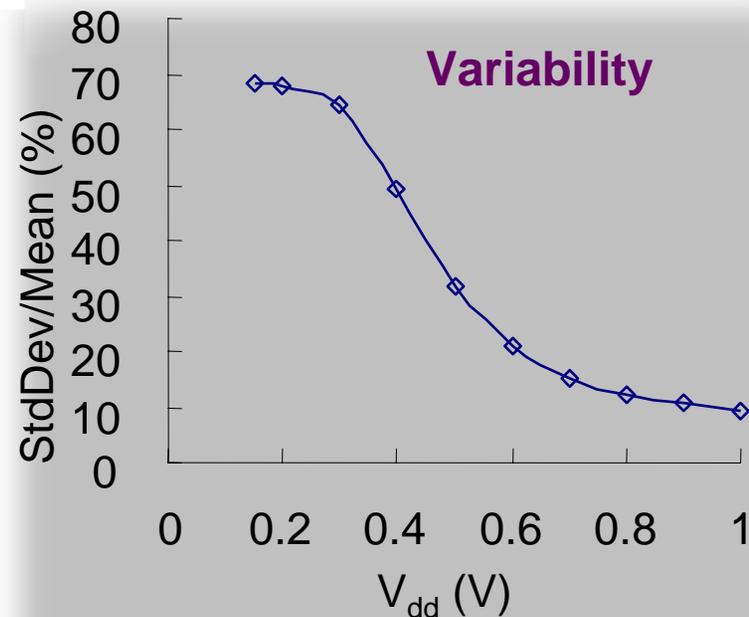
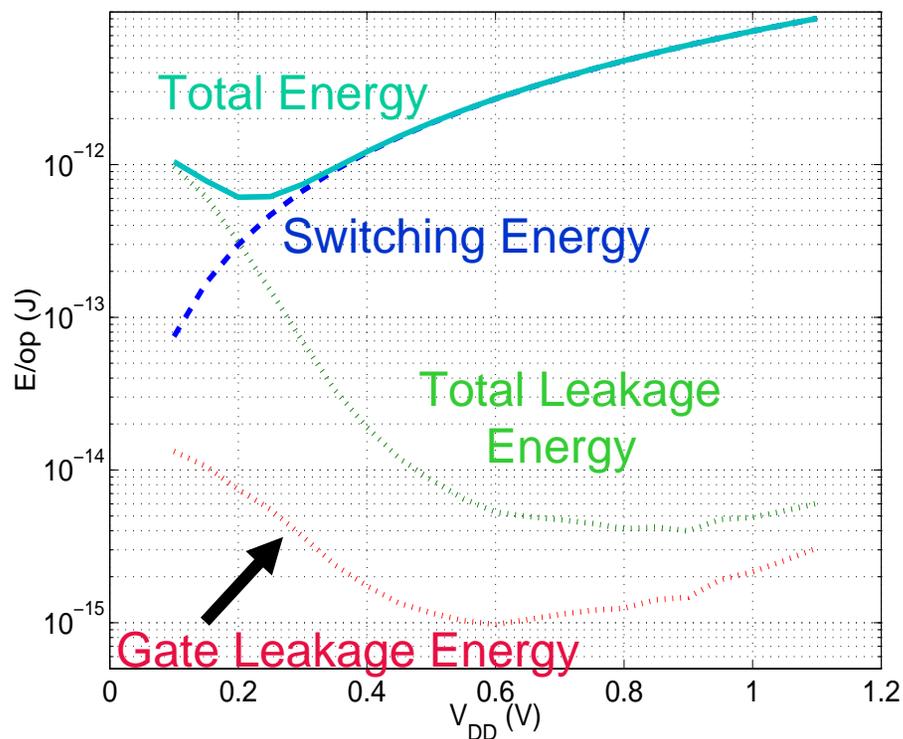
“A paper from MIT may introduce a whole new metric: lowest operating voltage. By aggressive use of voltage-frequency scaling, subthreshold circuit operation and supply voltage dithering, the team was able to keep an adder circuit operating over the full range from 1.1 V to under 300 mV.

This appears to be the lowest reported operating voltage for a digital circuit at the conference.”

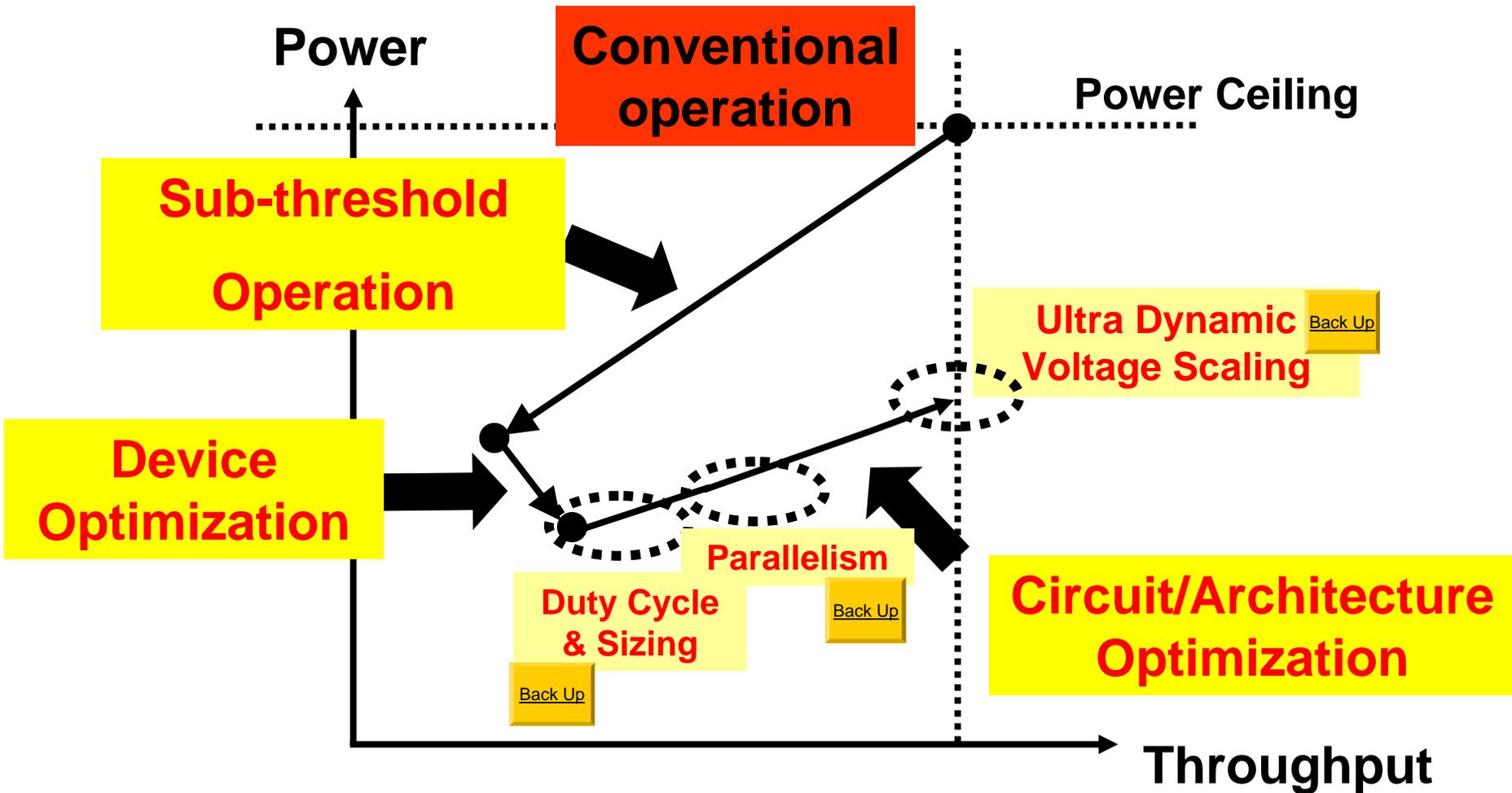


Requirements for Successful Ultra Low-voltage Operation:

- Architectures, circuits and devices to reduce power consumption by >100X yet mitigate throughput loss in deep sub-threshold operation
- Reliable operation with highly variable device components
- Wide-dynamic voltage range capability to allow high performance operation when needed

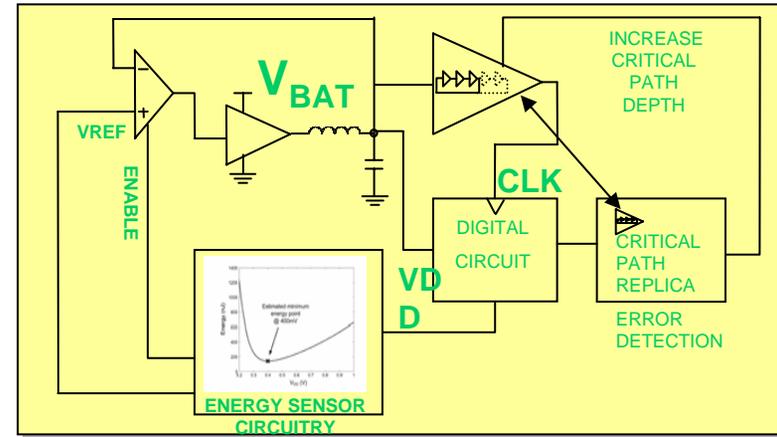
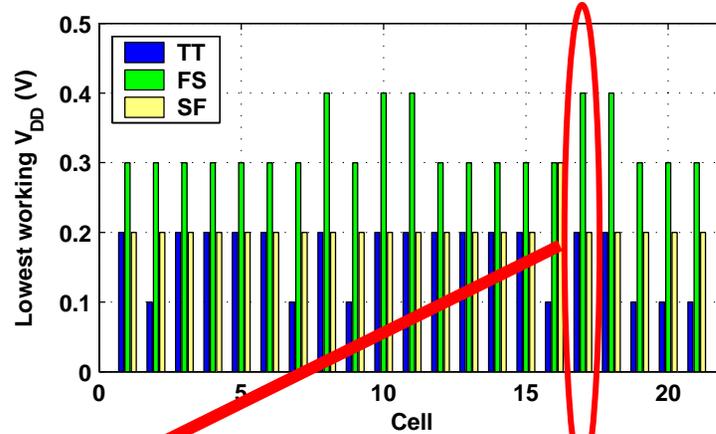


- Delay of logic increases, which must be compensated for through architecture optimization (e.g., parallelism and voltage dithering)
- Leakage energy increases at extremely low voltages
- Increased variability requires wider margins, reduced performance

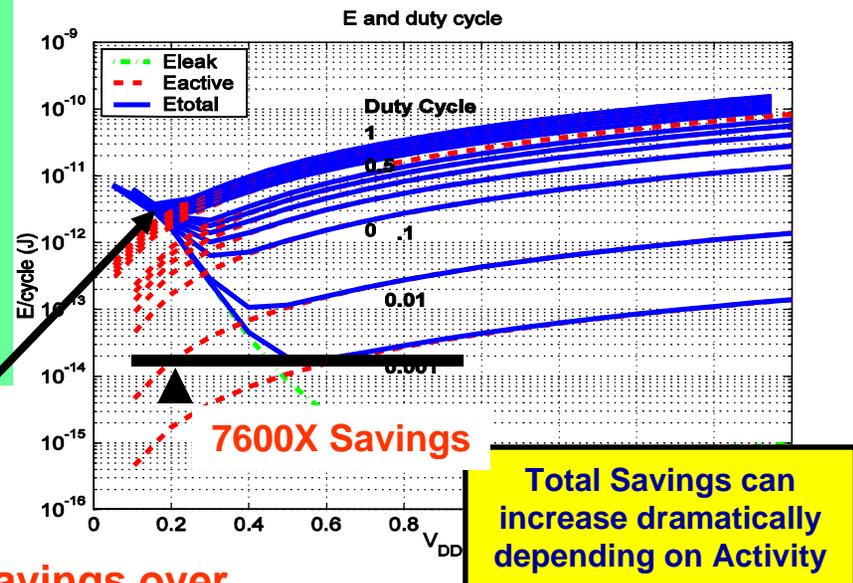
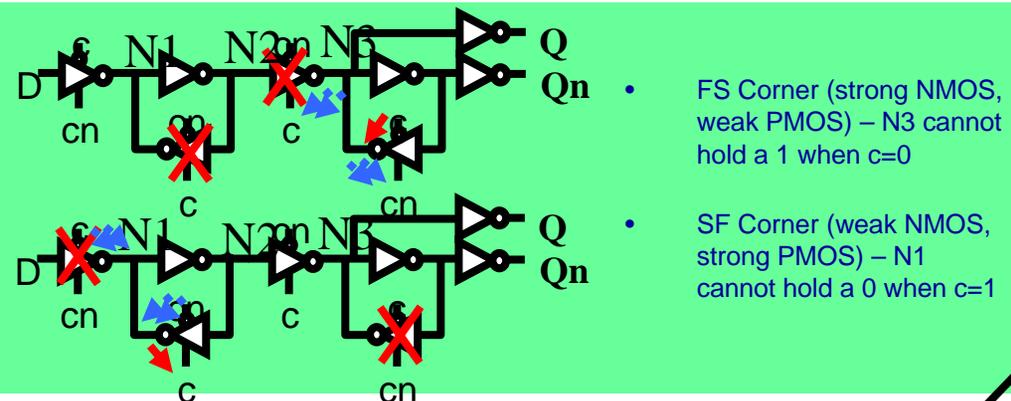


Dev/Ckt/Arch Optimization is needed to achieve target throughput using sub-threshold logic

- Some cells fail above the minimum energy point
- “Fixing” them by changing the sizes increases switched capacitance, but allows lower VDD operation
- What cells are optimal for a certain design?



Continuous feedback of energy/ power used to set minimum energy point.

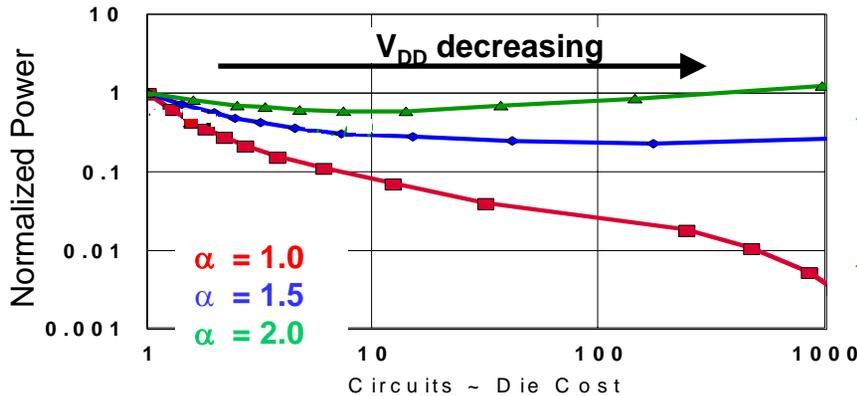


Min Energy Point 40X Savings over nominal case

[Back](#)

[Back](#)

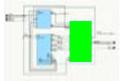
- Ultra-low power circuits will require parallelism to make up for longer switching delays.
- How many extra devices are needed?



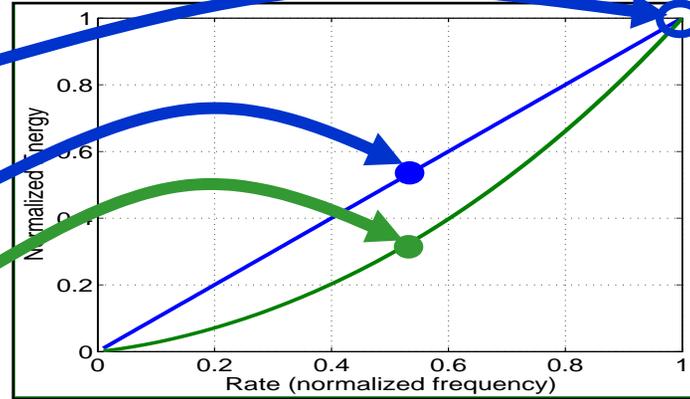
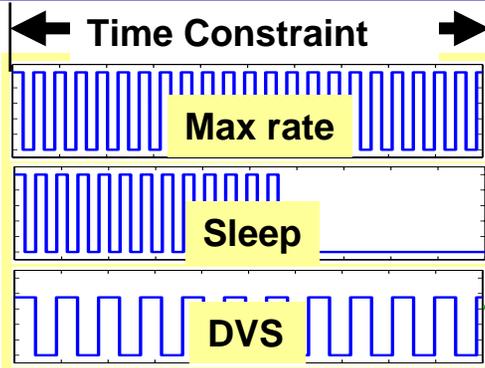
Model using power law function:
 Performance = (Device count) ^{α} / delay

- If $\alpha = 2$, then too much parallelism makes things worse.
- If $\alpha = 1$, then parallelism works effectively.

- *What types of computations/tasks can best utilize parallelism?*

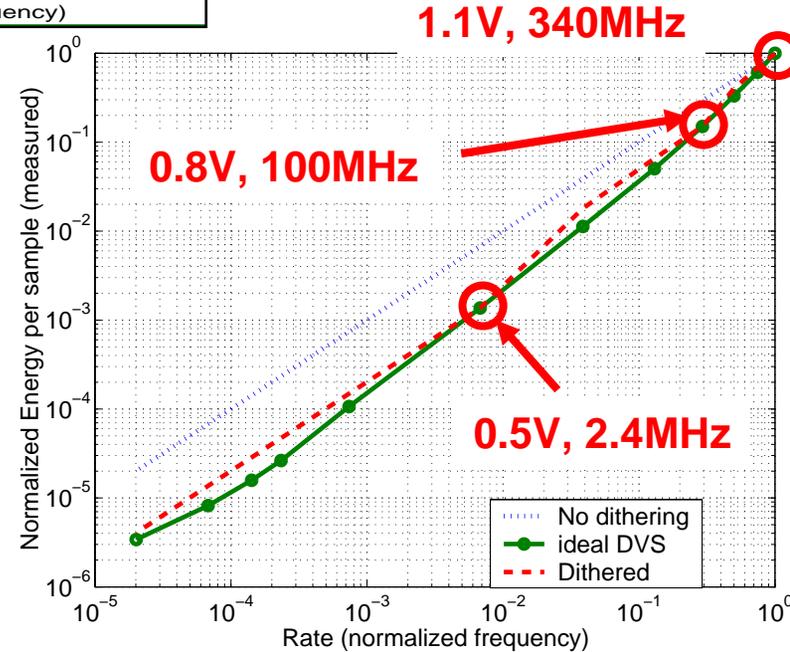
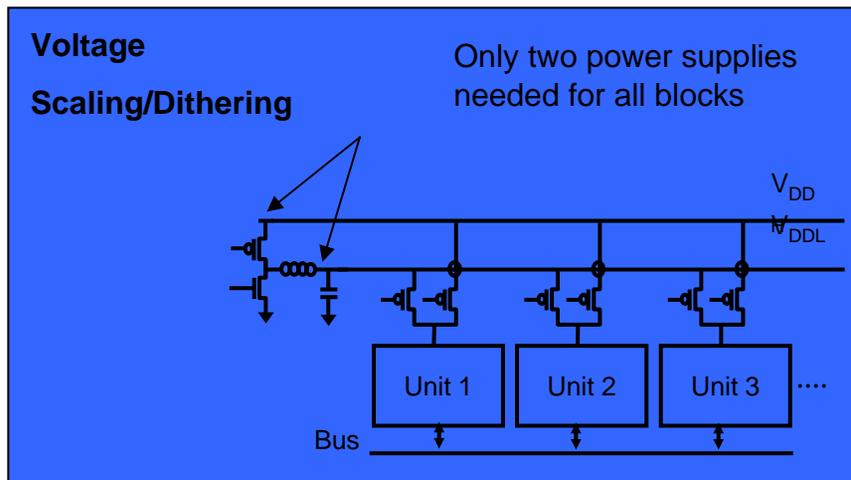
Implementation	Clock frequency	Vdd	Energy /Operation	# of Transistors
 + Sub-CMOS	748 kHz	650 mV	19.1 nJ	31k
 + Sub-CMOS	22 kHz	450 mV	2.47 nJ	111k
 + Sub-Pseudo NMOS	22 kHz	400 mV	1.77 nJ	86k

- Parallel architecture lowers the clock rate, reduces power dissipation by 87%
- Pseudo NMOS logic styles provides another 28% reduction



[Back](#)

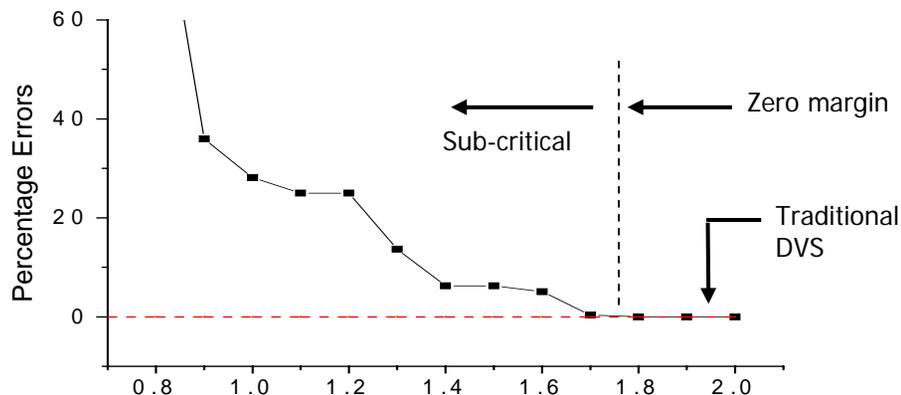
Dynamic Voltage Scaling with **infinite levels** saves energy per sample when the workload varies



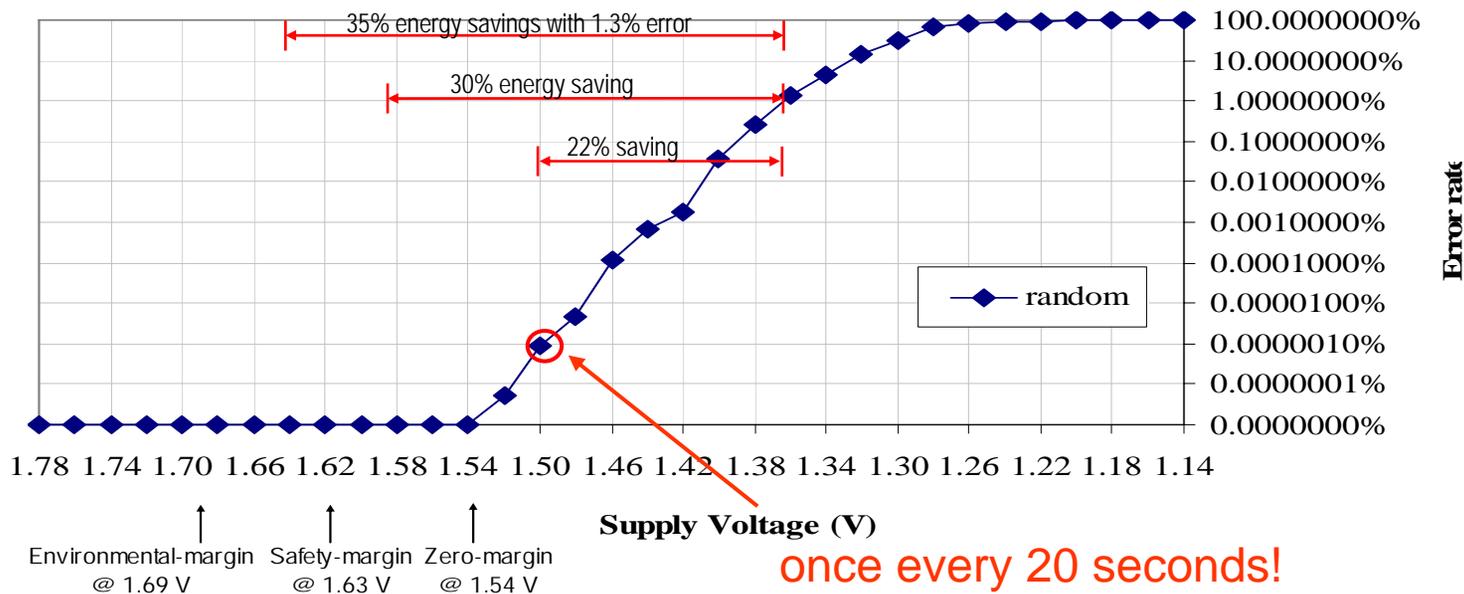
Closer to ideal over full range

Choose dithered voltages to match application of interest

- 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



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



[Back](#)