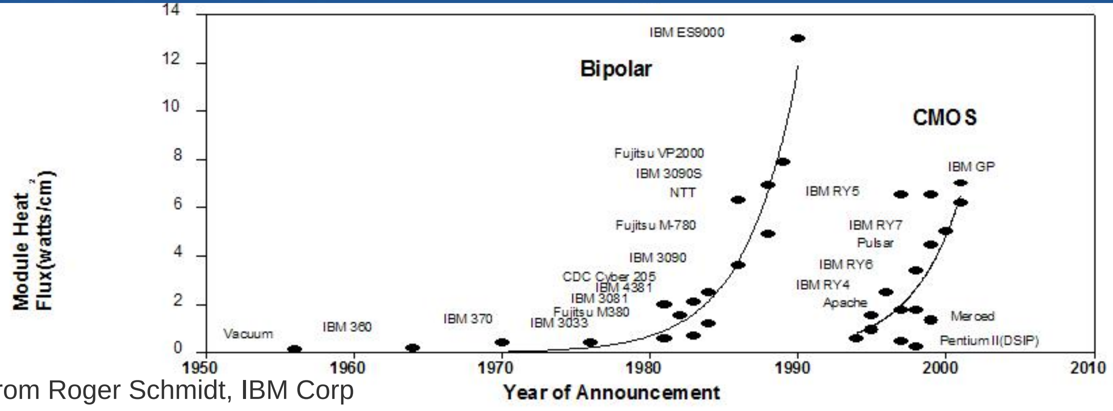
Scaling, Power
and the Future of CMOS

Mark Horowitz, Stanford

The 80's Power Problem

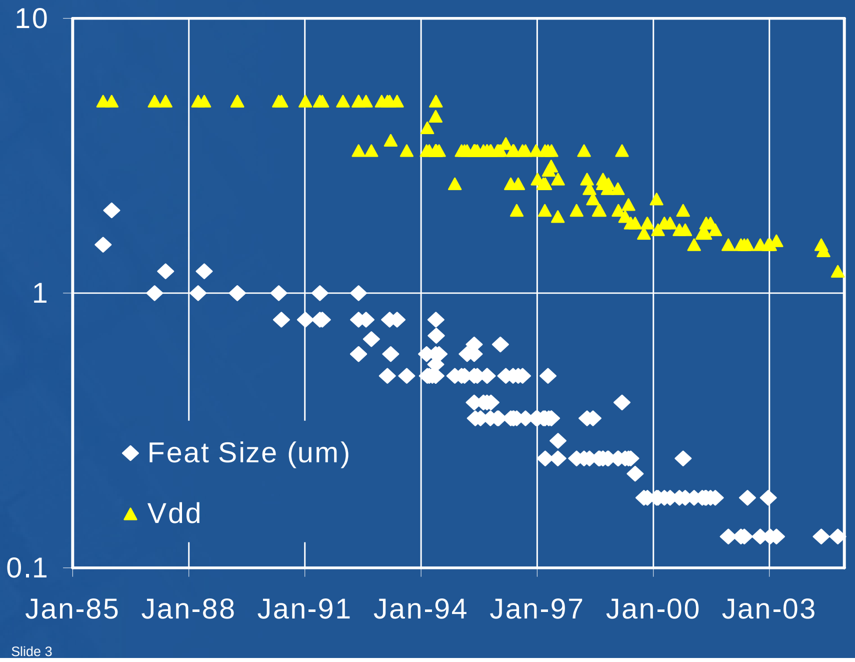
- Until mid 80s technology was mixed
  - nMOS, bipolar, some CMOS
- Supply voltage was not scaling / power was rising
  - nMOS, bipolar gates dissipate static power



From Roger Schmidt, IBM Corp

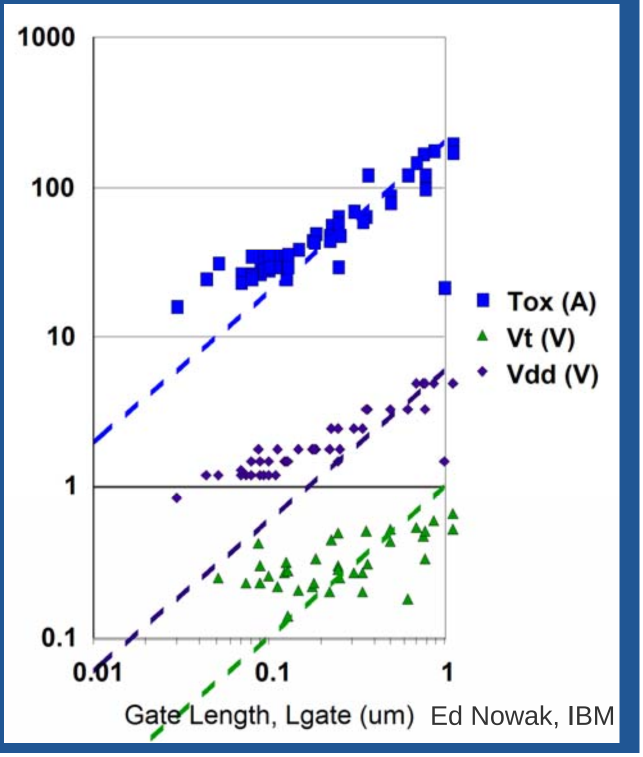
Solution: Move to CMOS

- And then scale Vdd



Bad News

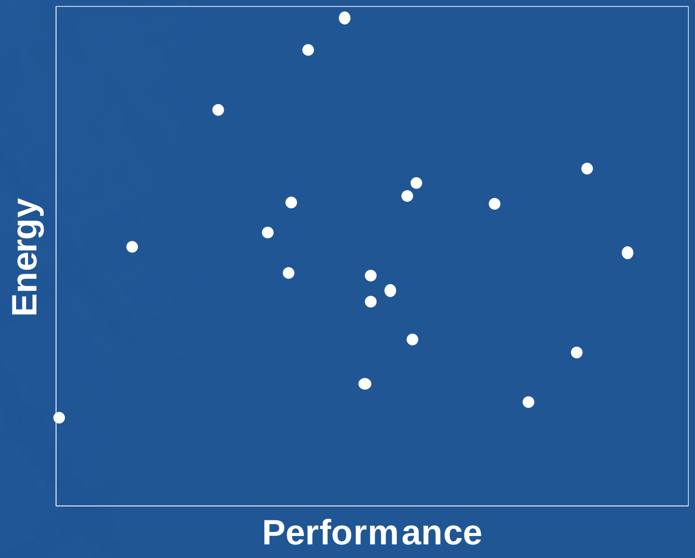
- Voltage scaling has stopped as well
  - kT/q does not scale
  - Vth scaling has power consequences
- If Vdd does not scale
  - Energy scales slowly



Ed Nowak, IBM

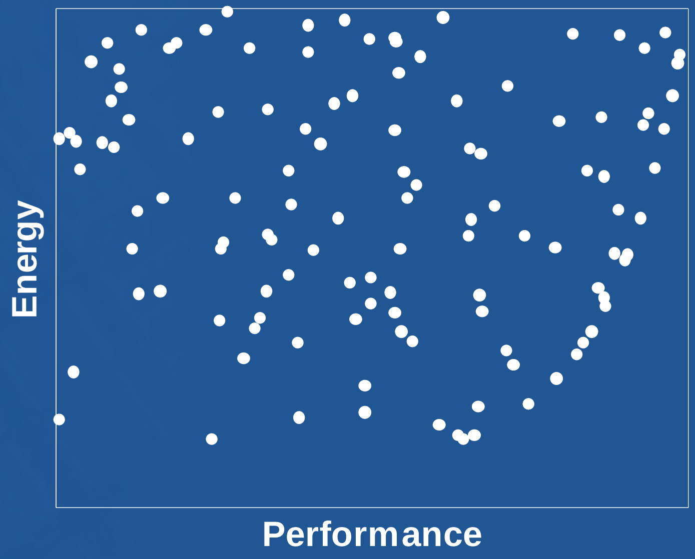
Energy – Performance Space

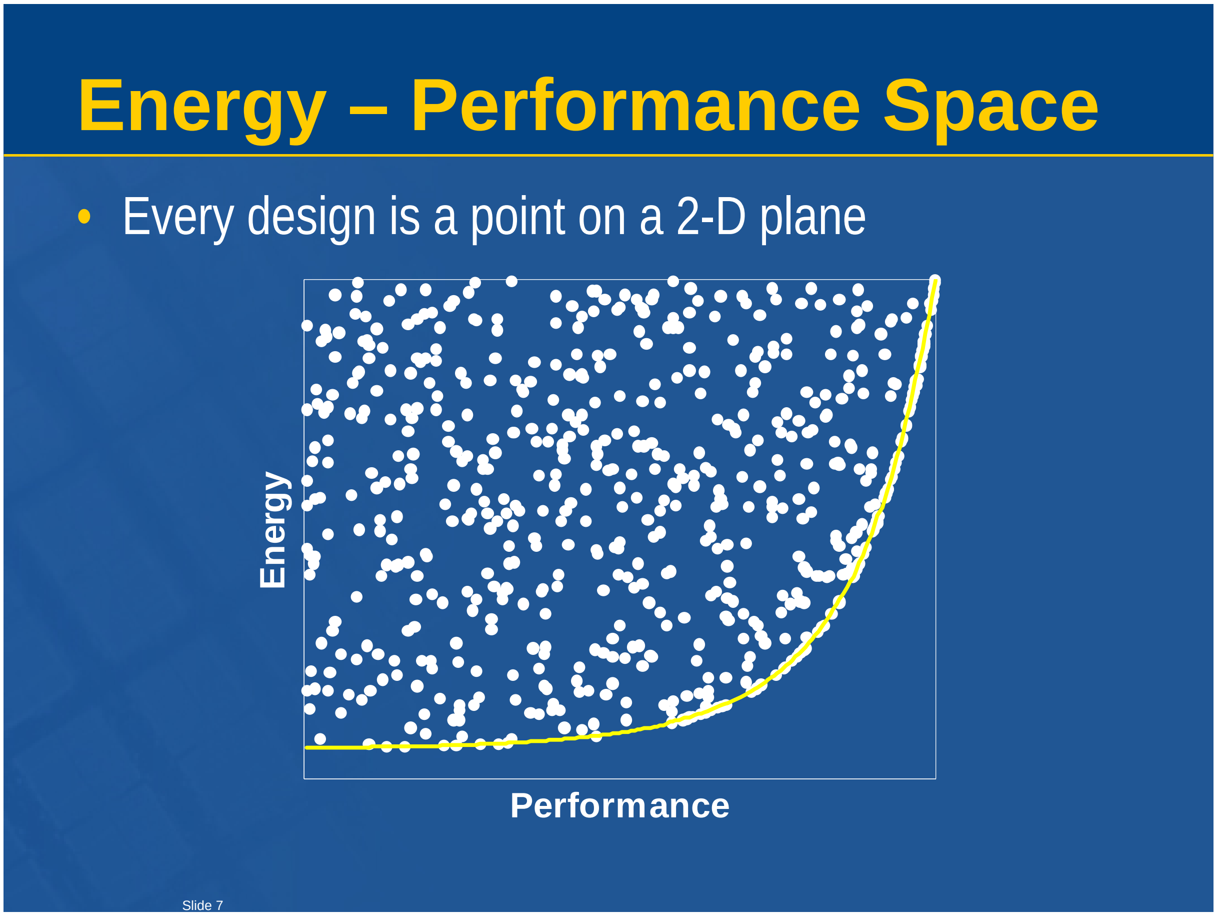
- Every design is a point on a 2-D plane



Energy – Performance Space

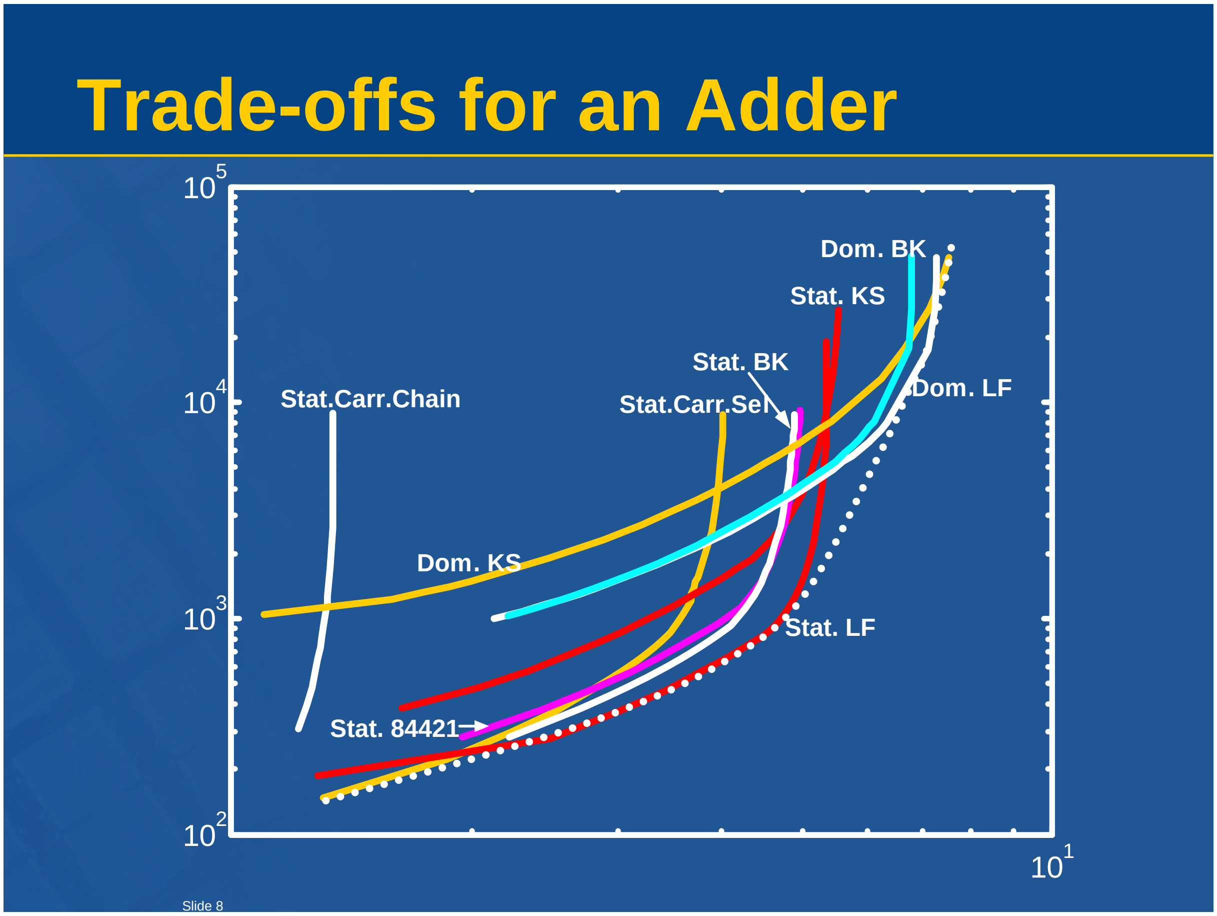
- Every design is a point on a 2-D plane





Energy – Performance Space

- Every design is a point on a 2-D plane



Trade-offs for an Adder

Key Observation:

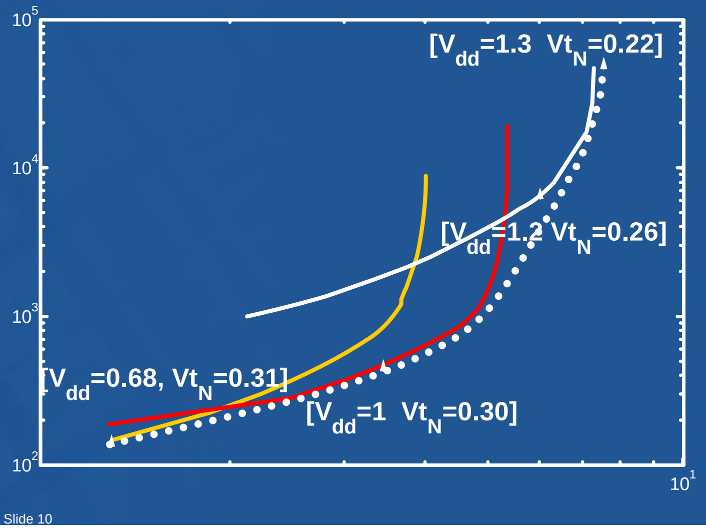
- Define the Energy/Delay sensitivity of parameter
  - For example Vdd:

$$Sens\left(V_{dd}\right) = -\frac{\partial E \big/ \partial V_{dd}}{\partial D \big/ \partial V_{dd}}\Bigg|_{V_{dd}=V_{dd}*}$$

- At optimal point, all sensitivities should be the same
  - Must equal the slope of the Pareto optimal curve

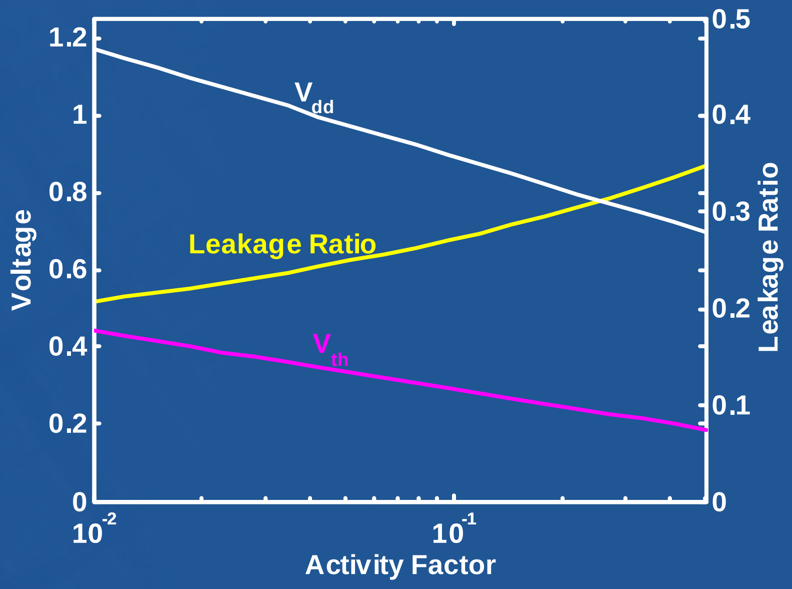
What This Means

- Vdd and Vth are not directly set by scaling
  - Instead set by slope of Pareto optimal curve
  - Leakage rose to lower total system power!



Leakage Energy

- Matching marginal costs for Vdd and Vth



Low Power Design Techniques

Three main classes of methods to reduce energy:

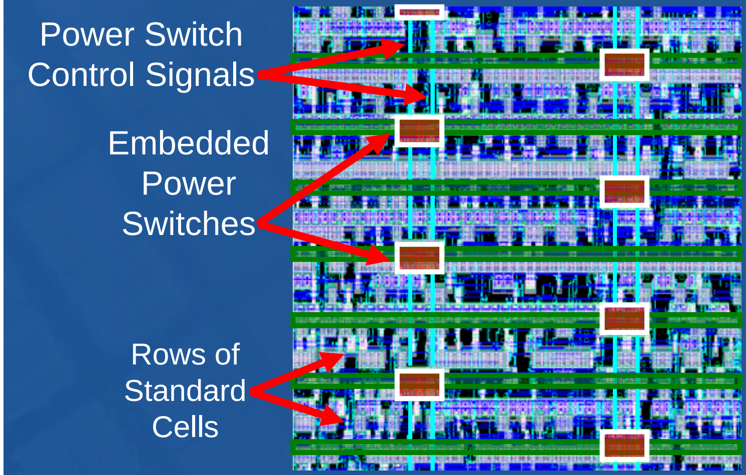
- Cheating
  - Reducing the performance of the design
- Reducing waste
  - Stop using energy for stuff that does not produce results
  - Stop waiting for stuff that you don’t need (parallelism)
- Problem reformulation
  - Reduce work (less energy and less delay)

Reducing Energy Waste

- Clock gating
  - If a section is idle, remove clock
  - Removes clock power
  - Prevents any internal node from transitioning
- Create system power states
  - Turn on subsystems only when they are needed
  - Can have different “off” states
    - Power vs. wakeup time
    - Disk (do you stop it from spinning?)

Embedded Power Gating

- Since transistors still leak when power is off



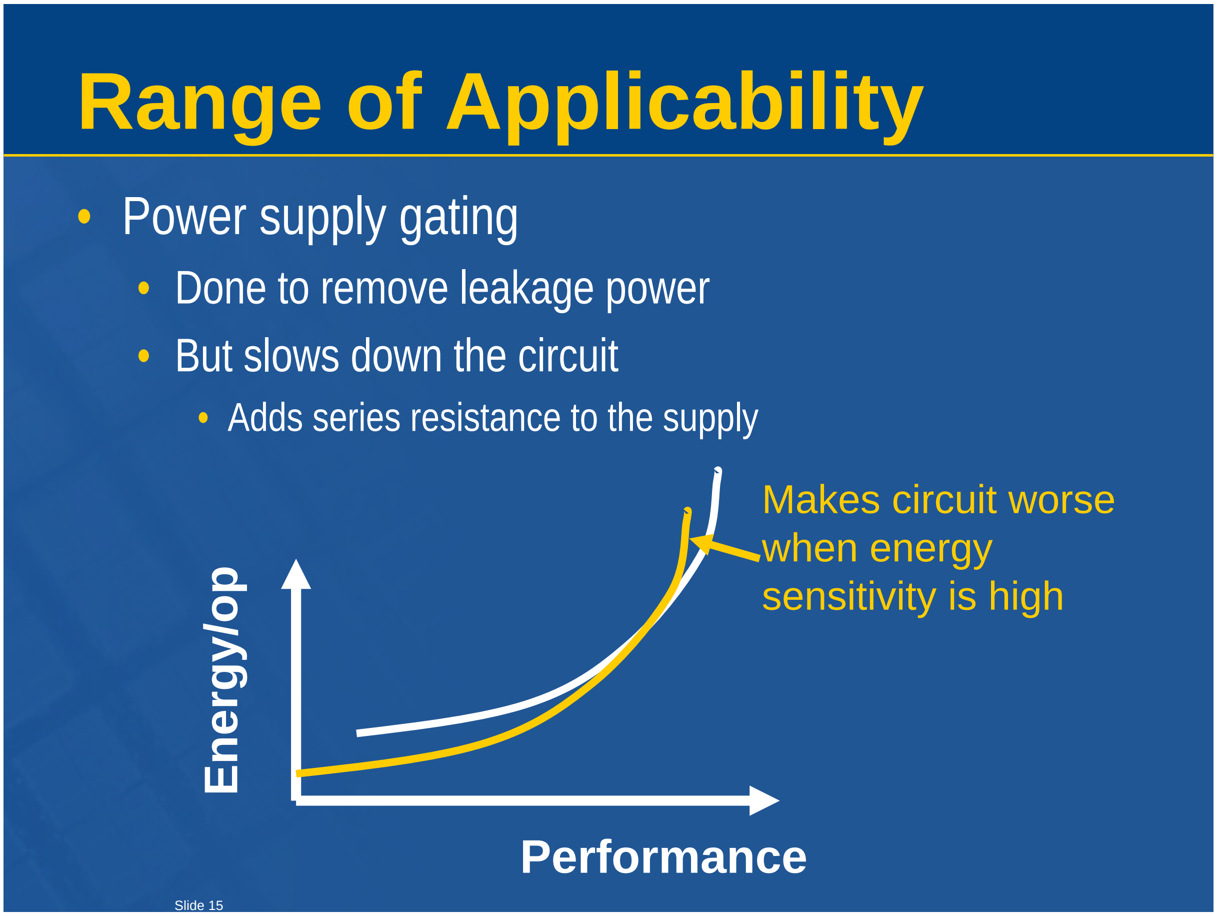
Power Switch Control Signals

Embedded Power Switches

Rows of Standard Cells

- Can reduce leakage
  - 250x reported
- But costs
  - Performance
  - Drop in Vdd, Gnd

Royannez, et al, 90nm Low Leakage SoC Design Techniques for Wireless Applications, ISSCC 2005

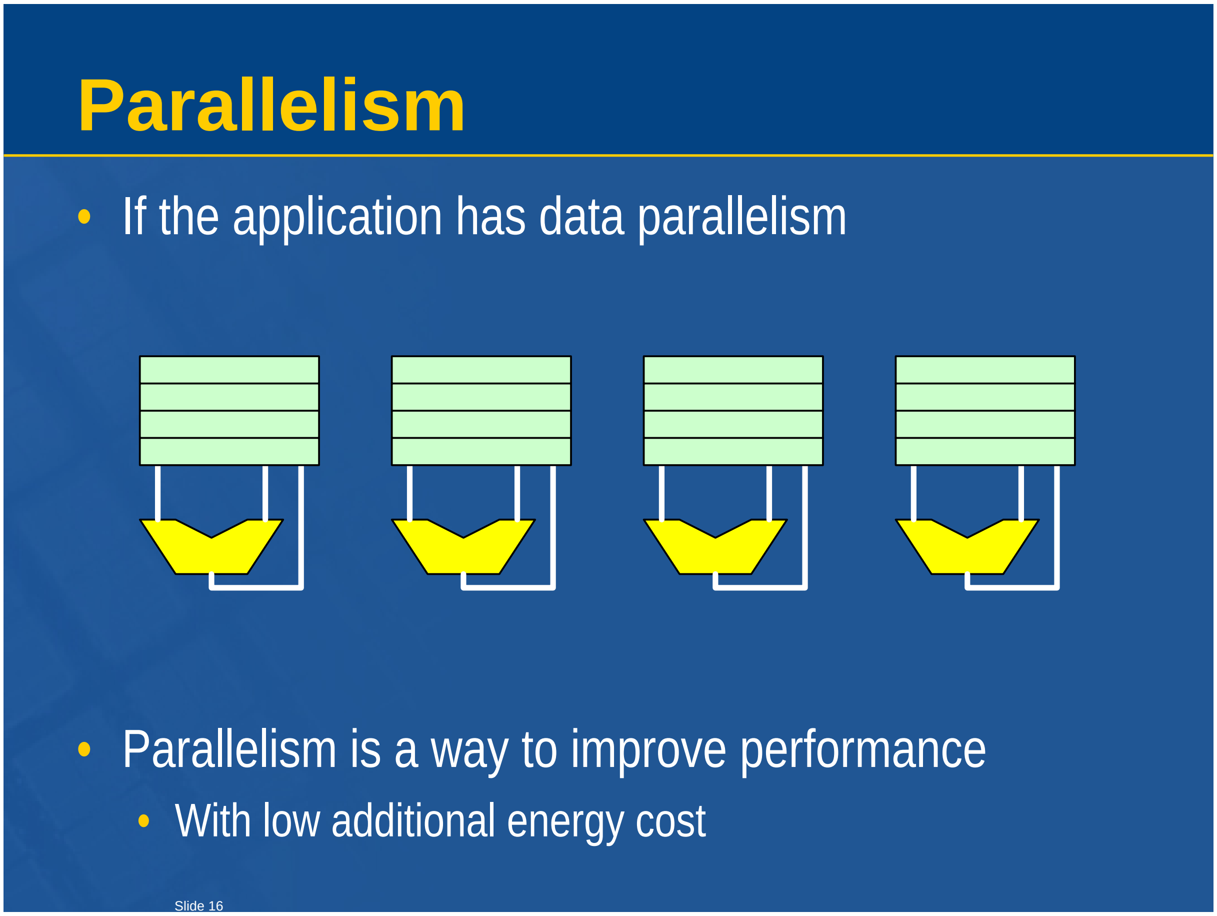


Range of Applicability

- Power supply gating
  - Done to remove leakage power
  - But slows down the circuit
    - Adds series resistance to the supply

Energy/op
Performance

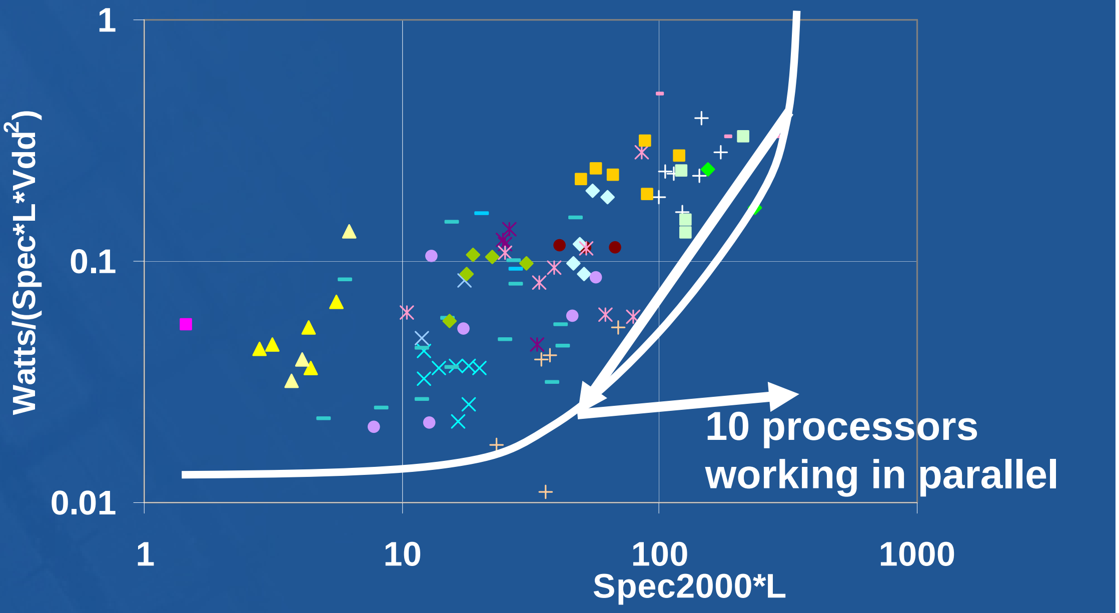
Makes circuit worse when energy sensitivity is high



Parallelism

- If the application has data parallelism
- Parallelism is a way to improve performance
  - With low additional energy cost

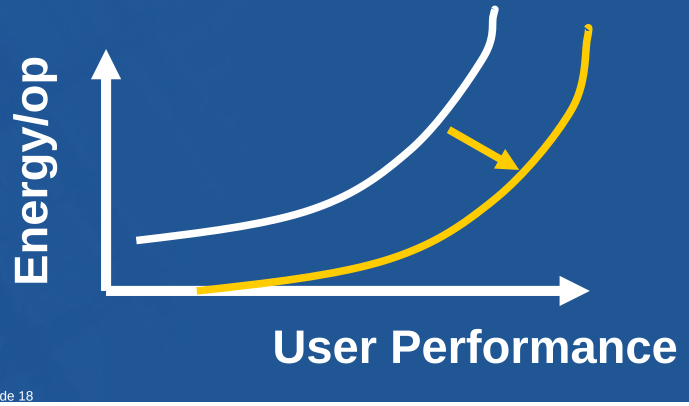
Existing Processors



10 processors working in parallel

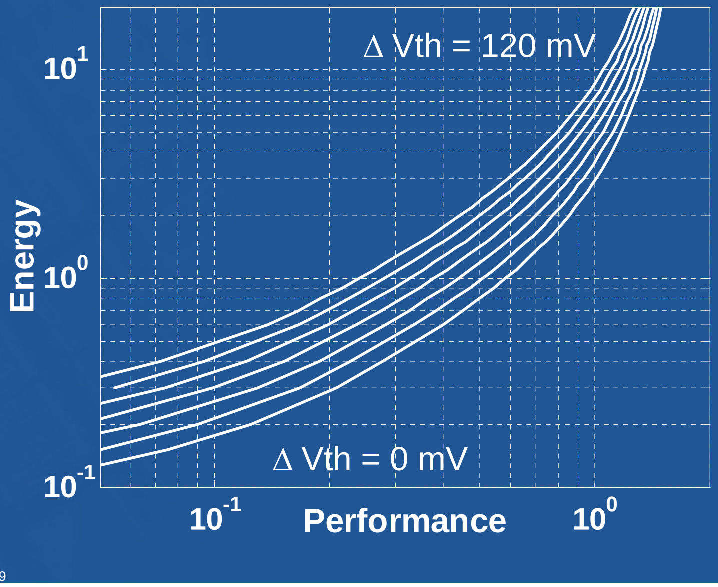
Problem Reformulation

- Best way to save energy is to do less work
  - Energy directly reduced by the reduction in work
  - But required time for the function decreases as well
    - Convert this into extra power gains
  - Shifts the optimal curve down and to the right



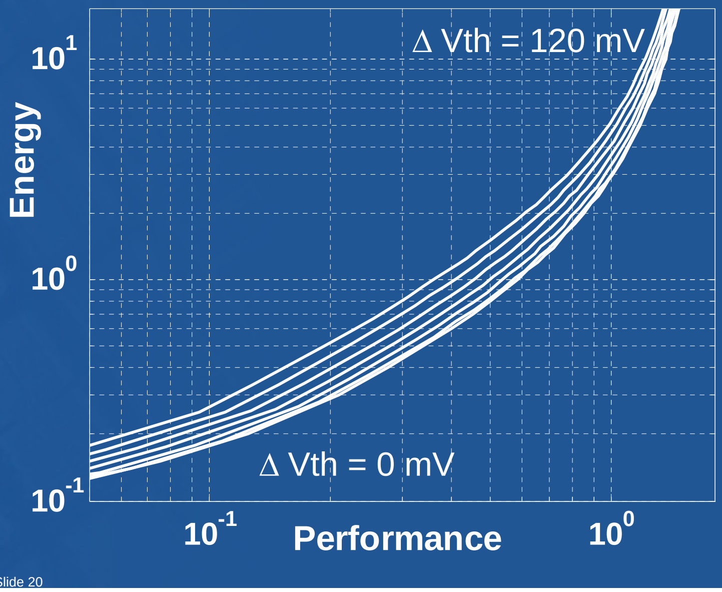
Cost of Variation

- Variability changes position of the optimal curves
  - Need to margin Vth, Vdd to ensure circuit always works



Partial Compensation

- Adjust Vdd after you get part back
  - Compensates very well for small deviations in Vth

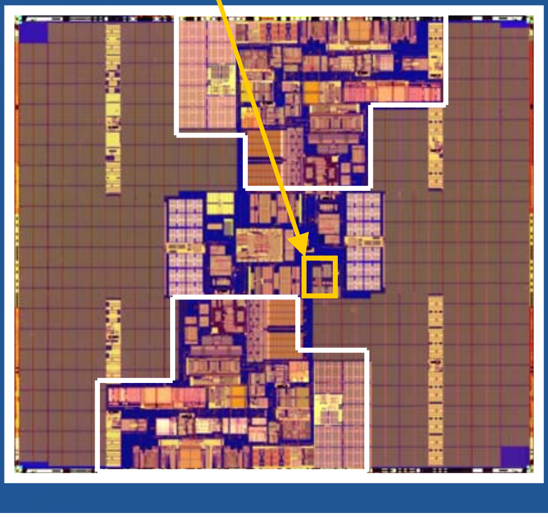


Variable Application Demands

- Try to provide a couple of operating points
  - Application can control speed and energy
  - Hard question is what are valid Vdd, F pairs
    - Usually determined during test
- Dynamic voltage scaling
  - Intel Speed Step in laptop processors
    - 2 performance/power points
  - Transmeta Long Run Technology
    - Many operating points. Test data + formula

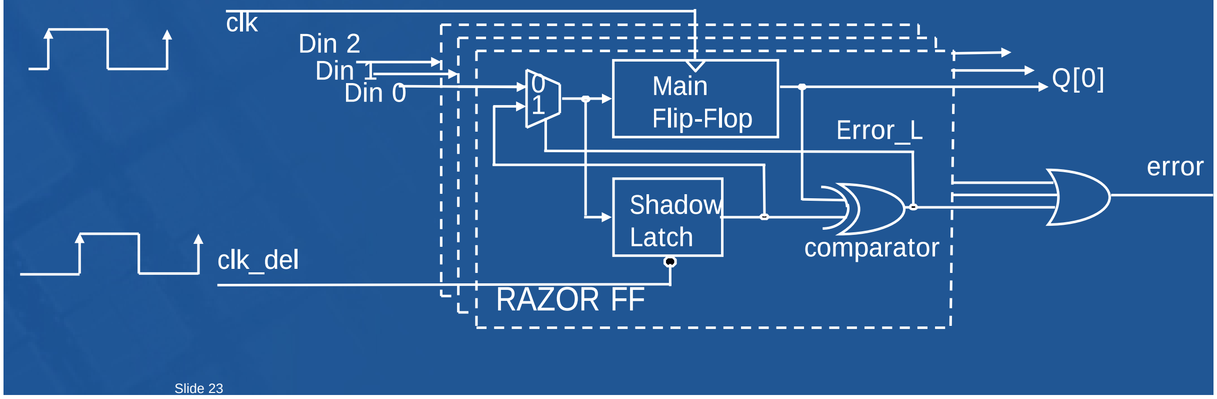
Constant Power Scaling

- Foxton controller on next-gen. Itanium II
  - Raises Vdd/boosts F when most units idle
  - Lowers Vdd for parallel code to stay in budget



Self Checking Hardware

- Razor (Austin/Blaauw, U of Mich)
  - Use the actual hardware to check for errors
  - Latch the input data twice
    - Once on the clock edge, and then a little later
    - If the data is not the same, you are going too fast

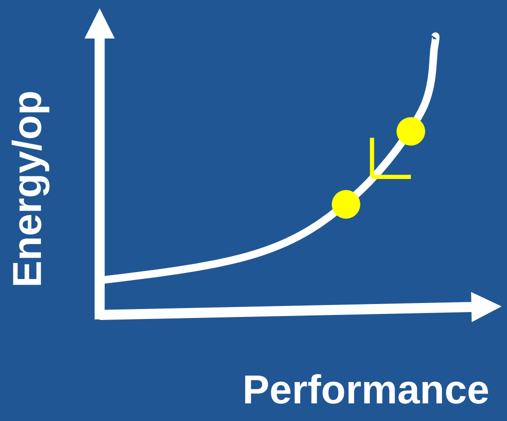


Future Systems

- Some simple math
  - Assume scaling continues
  - Dies don’t shrink in size
  - Average power/gate must decrease by 2x / generation
- Since gates are shrinking in size
  - Get 1.4x from capacitive reduction
- Where is the other factor of 1.4x ?

Exploit Parallelism / Scale Vdd

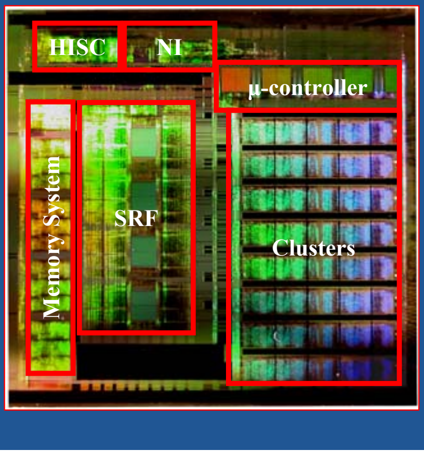
- If you have parallelism
  - Add more function units
    - Fill up new die (2x)
  - Lower energy/op
    - ΔE/ΔP will decrease
    - Vdd, sizes, etc will reduce
    - Build simpler architectures



- Works well when ΔE/ΔP is large
  - Per unit performance decrease is small

Exploit Specialization

- Optimize execution units for different applications
  - Reformulate the hardware to reduce needed work
  - Can improve energy efficiency for a class of applications
- Stream / Vector processing is a current example
  - Exploit locality, reuse
  - High compute density

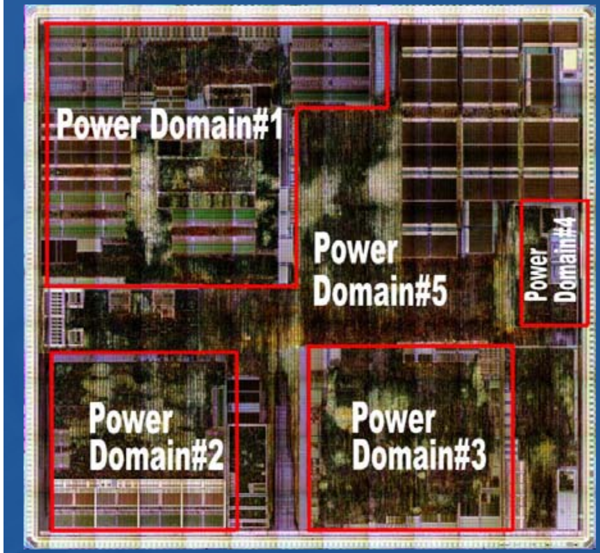


Bill Dally et al, Stanford
Imagine

Exploit Integration

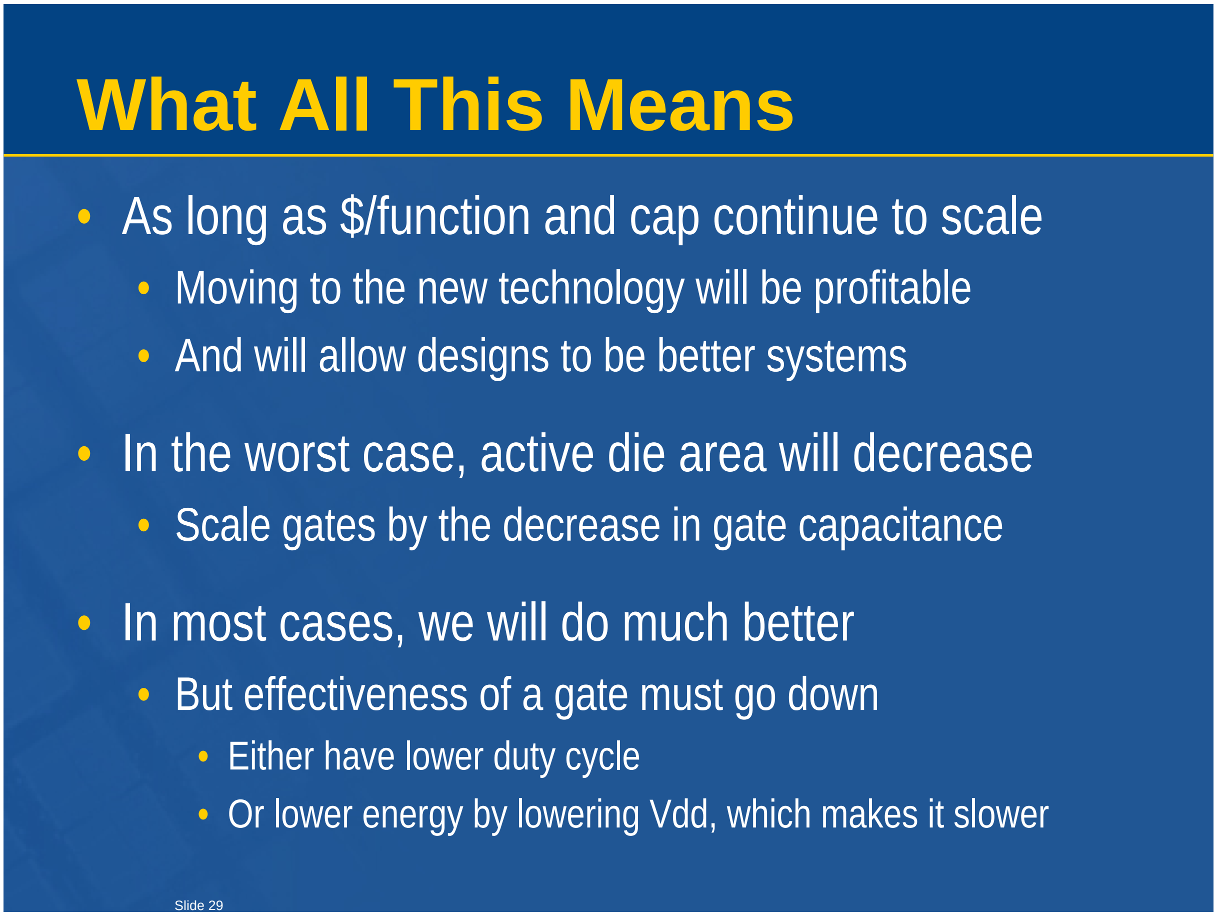
- If both those techniques don’t work
  - Still can increase integration by at least 1.4x
- Moving units onto one chip
  - Reduces the number of I/Os on system
  - I/O can take significant power today
  - Allows even larger integration

TI - OMAP2420



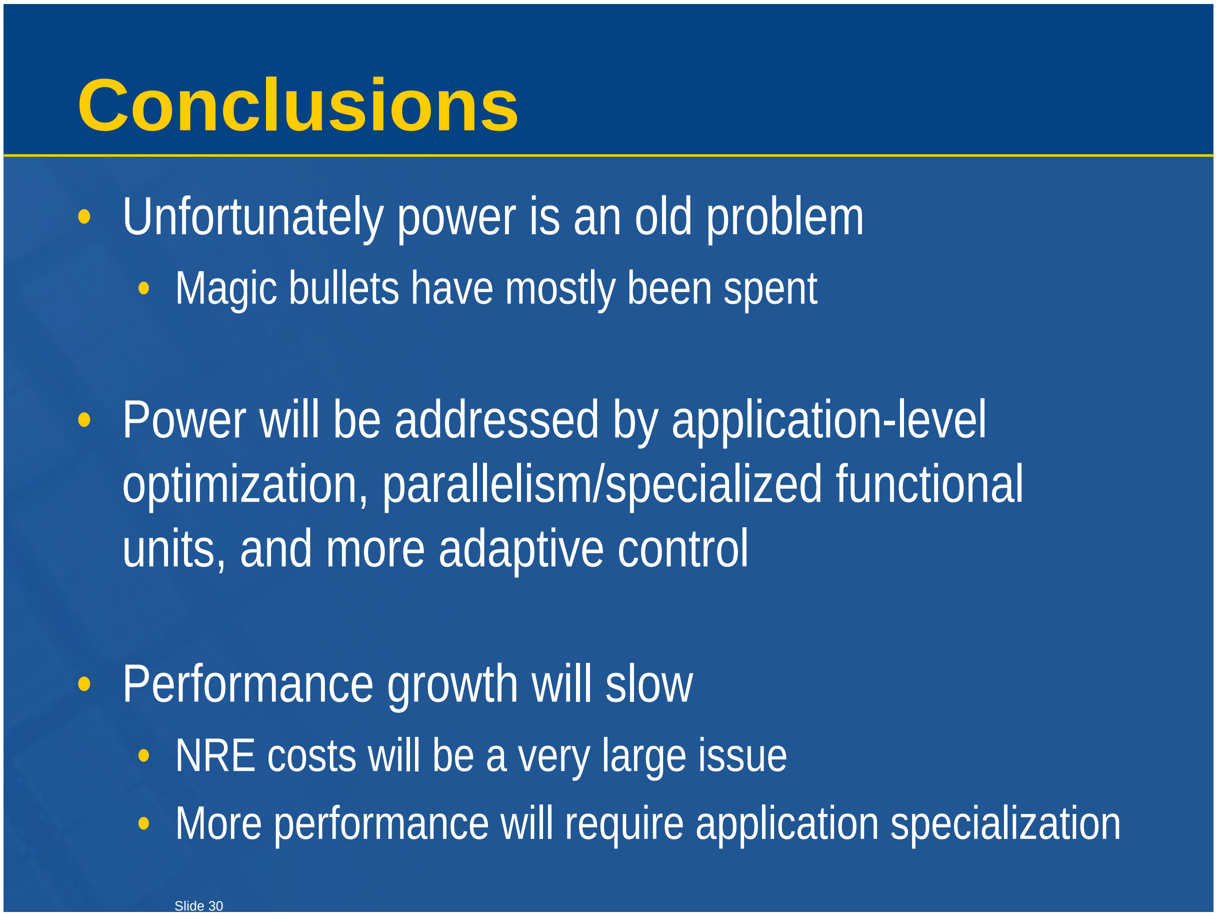
- Specialization
- And power domains
  - Most units are off
- OMAP 2420
  - 5 Power Domains
  - #1: MCU Core
  - #2: DSP Core
  - #3: Graphic Accelerator
  - #4: Core + Periph.
  - #5: Always On logic

Royannez, et al, 90nm Low Leakage SoC Design Techniques for Wireless Applications, ISSCC 2005



What All This Means

- As long as $/function and cap continue to scale
  - Moving to the new technology will be profitable
  - And will allow designs to be better systems
- In the worst case, active die area will decrease
  - Scale gates by the decrease in gate capacitance
- In most cases, we will do much better
  - But effectiveness of a gate must go down
    - Either have lower duty cycle
    - Or lower energy by lowering Vdd, which makes it slower



Conclusions

- Unfortunately power is an old problem
  - Magic bullets have mostly been spent
- Power will be addressed by application-level optimization, parallelism/specialized functional units, and more adaptive control
- Performance growth will slow
  - NRE costs will be a very large issue
  - More performance will require application specialization