ESE534
Computer Organization

Day 19: March 28, 2012
Minimizing Energy

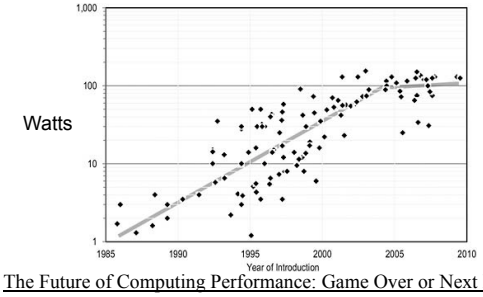
Today

- Discuss broad set of architectural options to reduce energy
- Relevant to upcoming project

At Issue

- Many now argue **energy** will be the ultimate scaling limit
  - (not lithography, costs, …)
- Proliferation of portable and handheld devices
  - …battery size and life biggest issues
- Cooling, energy costs may dominate cost of electronics
  - Even server room applications

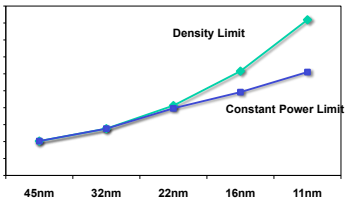
Microprocessor Power Density



The Future of Computing Performance: Game Over or Next Level?
National Academy Press, 2011
http://www.nap.edu/catalog.php?record_id=12980

Impact

Power Limits Integration



Source: Carter/Intel

Watch

- Energy optimized along with area, delay
- Energy optimized with little or no effect on area/delay
- Energy trades off with area, delay

Energy

$$E = \sum_i \left(\frac{1}{2} a_i C_i V_i^2 \right)$$

Energy

$$E = \sum_i \left(\frac{1}{2} a_i C_i V_i^2 \right)$$

How scale down V?

Energy and Delay

$$E = \frac{1}{2} CV^2$$

$$\tau_{gd} = Q/I = (CV)/I$$

$$I_{d,sat} = (\mu C_{OX}/2)(W/L)(V_{gs} - V_{TH})^2$$

Energy/Delay Tradeoff

- $E \approx V^2$
- $\tau_{gd} \approx 1/V$

$$E = \frac{1}{2} CV^2$$

$$\tau_{gd} = (CV)/I$$

$$I_{d,sat} \propto (V_{gs} - V_{TH})^2$$

- We can trade speed for energy
- $E \times (\tau_{gd})^2 \approx$ constant

Martin *et al*. *Power-Aware Computing*, Kluwer 2001
http://caltechcstr.library.caltech.edu/308/

Area/Time Tradeoff

- Also have Area-Time tradeoffs
  - HW2 spatial vs temporal multipliers
  - See more next week
- Compensate slowdown with additional parallelism
- …trade Area for Energy ➔ Architectural Option
  - HW3.3

Reduce V

- Lower voltage and run in parallel
  - HW3.3
- Lower voltage when you can get away with it (as much as can get away with)
  - Dynamic Voltage Scaling
- Lower voltage where don't need it
  - Multiple Vdd

Energy

$$E = \sum_i \left(\frac{1}{2} a_i C_i V_i^2 \right)$$

How reduce C?

Reduce C

- Shallow memory banks
- Reduce overhead programmable
  - Switches/stubs
- Reduce wire lengths
  - Maybe from reducing area
- Size gates
- Exploit technology
- Specialize computation or compute element

Energy

$$E = \sum_i \left(\frac{1}{2} a_i C_i V_i^2 \right)$$

How reduce a?

Reduce a

- Avoid/reduce glitches
- Code data
  - Example FSM
  - Example data encoding
  - Example bus-invert
  - One hot
- Compress data (send less)
- Exploit correlations
- Power Gate unused regions

Energy

$$E = \sum_i \left(\frac{1}{2} a_i C_i V_i^2 \right)$$

How reduce CV^2?

Reduce CV coupling

- Low swing highly capacitive signals
  - Bit-lines
  - Interconnect?

Energy

$$E = \sum_i \left(\frac{1}{2} a_i C_i V_i^2 \right)$$

How reduce aC?

Reduce aC

- Put high a with low C
- Coding/compression at high C points

Total Energy

- $E_{total} = E_{switch} + E_{leak}$

$$E = \sum_i \left(\frac{1}{2} a_i C_i V_i^2 \right)$$

$$E_{leak} = T_{cycle} \times V \times I_{leak}$$

Total Energy

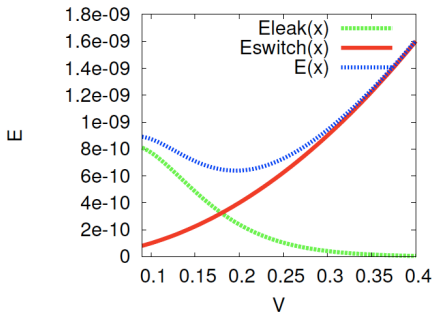
- $E_{total} = E_{switch} + E_{leak}$

$$E = \sum_i \left(\frac{1}{2} a_i C_i V_i^2 \right)$$

$$E_{leak} = T_{cycle} \times V \times I_{leak}$$

What if a_i is small? T_{cycle} large?

Graph for In Class (Day 6)



Leakage

$$E_{leak} = T_{cycle} \times V \times I_{leak}$$

How reduce T_{cycle}?

Reduce T_{cycle}

- Run as fast as can
- Optimize for delay
- Pipeline

Leakage

$$E_{leak} = T_{cycle} \times V \times I_{leak}$$

How reduce V?

Reduce V?

- Power gate → power down when not in use

Leakage

$$E_{leak} = T_{cycle} \times V \times I_{leak}$$

How reduce I_{leak}?

Reduce Leakage Current

- Power gating
- Increase V_{th}

Admin

- HW9 – note parts to run on computer
  - Don't wait until Sunday to run them
- Reading for Monday on web

Big Ideas

- Power major limiter going forward
  - Can put more transistors on a chip than can switch
- Some optimizations good for energy as well as (area, delay)
- Some optimizations just for energy
  - No effect on area, delay
- Changes tradeoffs will make