CDA5106 - Advanced Computer Architecture
Final Exam Review

Kobee Raveendran

1 Module 1: High-Performance Microprocessor Architecture

1.1 Module 1.2: Power Wall and Dennard Scaling

1.1.1 Notes

- energy: ability of a physical system to do work on other physical systems (unit: joule)
- power: rate at which energy is transformed (unit: watt; 1 watt = 1 joule delivered per second)
  - power = $V \cdot I$ (V = voltage, I = current)
- for capacitors:
  - energy stored = $0.5 \cdot C \cdot V^2$ (C = capacitance, V = voltage)
  - if a capacitor is drained at a frequency of f per second: power = $\frac{energy}{second} = 2 \cdot 0.5CV^2 = CV^2$
- Power wall problem
  - $P_{dyn} = ACV^2f$
  - A: fraction of gates actively switching
  - C: total capacitance of all gates
  - V: supply voltage
  - f: frequency of switching
- Power wall fundamentals
  - max frequency vs. threshold voltage:
  - $f_{max} = c \cdot \frac{(V-V_{thd})^{1.3}}{V}$
- **Dennard Scaling Example (old)**
  - if gate length (transistor size) scales by $S = 0.7$ (both length and width), then:
  - capacitance scales by $S = 0.7$
  - original area scales by $S^2 = 0.5$
  - number of transistors scales by $\frac{1}{S^2} \approx 2$
  - supply voltage (V) scales by $S = 0.7$
  - frequency (f) scales by $\frac{1}{S} = 1.4$
  - then, **dynamic power** $P_{dyn} = ACV^2f$
  - and **new dynamic power** $P'_{dyn} = A'C'V'^2f'$
  - $P'_{dyn} = (2A)(0.7C)(0.7V)^2(1.4f) \approx 1 \cdot ACV^2f = P_{dyn}$
- **Post Dennard Scaling example (new)**
  - capacitance scales by $S = 0.7$
  - number of transistors scales by $\frac{1}{S^2} = 2$

- supply voltage (V) cannot scale without also scaling threshold voltage (V_{thd}), and doing that increases static power exponentially
- frequency (f) scales by $\frac{1}{S} = 1.4$
- result: dynamic power doubles every generation
- $P_{dyn} = ACV^2f$
- $P'_{dyn} = A'C'V'^2f' = (2A)(0.7C)(1 \cdot V)^2(1.4f) \approx 2 \cdot P_{dyn}$

1.1.2 Exercises

1. Suppose that instead of progressing at a ratio of 0.7, Moore's law slows down and transistor gate length scales at a ratio of 0.8 instead. Find the dynamic power consumption under *unlimited* and *limited* scaling for the next process generation.
   - Unlimited/old scaling rule
     - gate length scales by $S = 0.8$
     - capacitance scales by $S = 0.8$
     - original area scales by $S^2 = 0.64$
     - num transistors thus scales by $\frac{1}{S^2} = 1.56$
     - supply voltage scales by $S = 0.8$
     - frequency scales by $\frac{1}{S} = 1.25$
     - dynamic power stays constant:
       $P_{dyn} = (1.56A)(0.8C)(0.8V)^2(1.25f)$
   - leakage-limited/new scaling
     - capacitance scales by $S = 0.8$
     - num transistors scales by $\frac{1}{S^2} = 1.56$
     - supply voltage does not scale without scaling threshold voltage too, which increases static power exponentially
     - frequency scales by $\frac{1}{S}1.25$
     - dynamic power consumption increases:
       $P'_{dyn} = (1.56A)(0.8C)(V)^2(1.25f) = 1.56 \cdot P_{dyn}$
2. With limited voltage scaling, suppose that we want to keep the dynamic power consumption constant in the next generation by keeping frequency constant and reduce die area. How much should we reduce die area to achieve that?
   - gate length scales by $S = 0.7$
   - capacitance scales by $S = 0.7$
   - original area scales by $S^2 = 0.5$
   - supply voltage and frequency are constant
   - dynamic power consumption must stay constant: $P'_{dyn} = P_{dyn}$
     $ACV^2f = A'(0.7C)V^2f \longrightarrow A = 0.7A'$
   - number of transistors in the next generation: $A' = 1.4A$ (instead of 2A like before; i.e. 70% of 2A)
   - thus die area shrinks by 30%
3. Describe the difference between energy and power.

   Power is the rate of energy consumption.
4. Describe the impact of threshold voltage choice on static and dynamic power consumption as transistors are scaled down.

   If threshold voltage is lowered, dynamic power decreases (nearly linearly) but static power increases exponentially.
5. How has processor design adapted to the power wall problem?

   Stalling frequency growth, multicore, and sophisticated power management (clock gating, voltage and frequency scaling, power gating).

1.1.3 Overview of ILP Techniques

Caches example

- processor with 1-ns clock
- 64KB cache memory with 2-ns read time, 95% hitrate
- 512MB main memory with 150-ns read time
- What is the average access time (AAT) in this memory system?

Answer:

- hits: $95 \cdot 2$ ns, misses: $5 \cdot (2 + 150)$ ns
- total = hit time + miss time = $190 + (10 + 750) = 950ns$
- AAT = $\frac{total}{100} = 9.5ns$

2 Module 2: Performance, Cost, and Reliability of Microprocessors

2.1 Performance Evaluation 1

2.1.1 Amdahl's Law

- performance improvement ("speedup") is limited by the part you can't improve
- (s) $Speedup_{enhanced}$ = best case speedup from gizmo alone
- (f) $Fraction_{enhanced}$ = fraction of task that gizmo can enhance
- $s_{overall} = \frac{1}{(1-f)+\frac{f}{s}}$

Example:

- jet plane wing simulation, where 1 run takes 1 week on your computer
- your program is 80% parallelizable
- new supercomputer has 100,000 processors
- $s = 100,000$
- $f = 0.8$
- overall speedup: $s_{overall} = \frac{1}{(1-f)+\frac{f}{s}} = \frac{1}{(1-0.8)+\frac{0.8}{100000}} \approx \frac{1}{0.2} = 5$
- only about 5 times faster (33 hours instead of 1 week), but not worth the high price tag (using a cheaper computer with only 100 processors instead yields a 4.8X speedup!)

More examples:
Ex 1:

- $f = 0.95$
- $s = 1.10$
- $s_{overall} = \frac{1}{(1-0.95)+\frac{0.95}{1.10}} = 1.094 \approx 1.10$

Ex 2:

- $f = 0.05$
- $s \to \infty$
- $s_{overall} = 1.053$

2.1.2 Run Time

- CPU time = clock cycle count × cycle time
- cycles per instruction (CPI) = $\frac{\texttt{clock cycle count}}{\texttt{instruction count}}$
- CPU time = IC × CPI × CT

2.2 Performance Evaluation 2

-