Interesting Design Computations

1

COMPUTER ARCHITECTURE: A QUANTITATIVE APPROACH

JOHN L. HENNESSY | DAVID A. PATTERSON

6TH EDITION

Ric-up: Dennard scaling

2

- In 1974: Robert Dennard said that **power density was constant for a given area of a silicon chip even as you increase the number of transistors because of the smaller size of each transistor**. *Dennard scaling* **ended** in 2004 because current and voltage could not keep dropping and still maintain dependability of ICs.

Parallelism

3

- The death of Dennard scaling saw the **birth of parallelism**. The microprocessor industry now started focusing on use of multiple efficient processors or cores instead of a single inefficient processor.
- **In 2004**: **Intel** cancelled it high performance uniprocessor projects and joined others in declaring that the road to higher performance lies in multiple processors per chip rather than a single uniprocessor.

Parallelism: From Instruction Level to Data Level

4

- The switch to multicores saw a shift from relying entirely on instruction level parallelism (ILP) to:
  1. Data Level parallelism (DLP)
  2. Thread level parallelism (TLP)
  3. Request level parallelism (RLP)

ILP, DLP, TLP and RLP

5

- While the compiler and hardware are capable of implementing ILP **implicitly** without the programmer's attention; DLP, TLP and RLP are **explicit**.
- This means in DLP, TLP and RLP the programmer has to restructure the application so that it can exploit explicit parallelism: a task which may be simple in some instances or require considerable effort in others.

Moores Law and Amdahl's Law

6

- **Moores Law**: in 1965 predicted that **the number of transisters per chip would double every year but was amended in 1975 to every two years**.
- **Amdahl's Law**: tried to prescribe the practical limits of the number of useful cores per chip stating that: **if 10% of the task is serial then the maximum performance benefit from parallelism is 10 X no matter how many cores you put on the chip**.

Death of Moores Law

7

- Moores prediction lasted for about 50 years but no longer holds e.g.
  1. In 2010: Intel's Microprocessor had 1,170,000,000 transistors
  2. If Moores law had continued you would expect microprocessors in 2016 to have 18, 720, 000, 000 transistors: but the equivalent had only 1,750, 000, 000.

Interesting Design Computations

Performance: Bandwidth vs Latency

8

- Bandwidth or throughput is the total amount of work done in a given time e.g. megabyte per second when transferring data either on a network or from a disk
- For **microprocessors and networks**: performance is the greatest driving force hence bandwidth has increased 32,000-40,000 X against 50 - 90 X in latency.
- For **memory**: capacity is more important than performance with bandwidth improving 400 – 2400 X and latency 8 – 9 X

Interesting Design Computations

Energy and power metrics

9

- Energy is the biggest challenge facing the computer designer for nearly every class of computer. How;
  1. First power must be brought in and distributed around the chip: if you look at a modern microprocessor it has hundreds of chips and interconnect layers just for power and ground
  2. Second power is dissipated as heat and must be removed

Interesting Design Computations

Design considerations: performance, power and energy

10

1. What is the maximum amount of power a processor ever requires? This is important e.g. if a processor tries to draw more power (current) than a power supply can give, it may lead to a voltage drop hence a malfunction
2. What is the sustained power consumption? This metric is called thermal design power (TDP); it determines the cooling requirement; Maximum power is usually 1.5 times higher than TDP

Interesting Design Computations

TDP

11

- TDP is also not average power which is likely to be lower. Power supply unit has to exceed TDP
- A typical cooling system is designed to match or exceed TDP otherwise junction temperature in the CPU would exceed maximum value resulting in device failure or permanent damage
- Modern processors have the ability to:
  1. Lower clock speed if junction temperature approaches limit
  2. Activate a thermal overload trap to power down the chip

Interesting Design Computations

Energy

12

- Clock rate can be reduced dynamically to limit power consumption
- Energy per task is often a better measurement
- For CMOS chips, energy consumption has been in switching transistors called dynamic energy.
- Dynamic energy:
  - Transistor switch from 0 -> 1 or 1 -> 0; **each switch consumes power**
  - ½ x Capacitive load x Voltage^2
- Dynamic power:
  - ½ x Capacitive load x Voltage^2 x Frequency switched
- Reducing clock rate reduces power, not energy

Interesting Design Computations

Example

13

- Some microprocessors today are designed to have adjustable voltage. A 15% reduction in voltage may result in a 15% reduction in frequency. What would be the impact on dynamic energy and power?
- **Soln**: because the capacitance remains the same, the answer for energy is a ratio of the voltages:
  - Energy_{new} / Energy_{old} = (Voltage X 0.85)^2 / Voltage^2 = 0.85^2 = 0.72
  - This means energy reduces to 72% of the original
  - For power we include the ratio of the frequencies:
  - 0.72 X (Frequency Switched X 0.85) / Frequency Switched = 0.61
  - This means power shrinks to 61% of the original

Energy & Power

14

- As we move from one process to the next, the increase in the number of transistors switching and the frequency with which they change dominate the decrease in load capacitance and voltage, leading to an overall growth in power consumption and energy.
- The first microprocessors consumed less than a watt, then the first 32-bit microprocessor (Intel 80386) consumed 2 watts. Modern microprocessor 4.0GHz Intel Core i7-6700K consumes 95Watts

Cost of ICs

15

- Although the costs of ICs have dropped, the basic process of silicon manufacture remains the same: A wafer is still tested and chopped into dies. The dies are then packaged. Therefore the cost of a packaged IC is:

$$\text{Cost of integrated circuit} = \frac{\text{Cost of die} + \text{Cost of testing die} + \text{Cost of packaging and final test}}{\text{Final test yield}}$$

Where yield = % of good dies on the wafer

$$\text{Die yield} = \text{Wafer yield} \times 1/(1 + \text{Defects per unit area} \times \text{Die area})^N$$

16