PERFORMANCE METRICS

Mahdi Nazm Bojnordi

Assistant Professor

School of Computing

University of Utah

Overview

- Announcement
  - Aug. 28th: Homework 1 release (due on Sept. 4th)
    - Verify your uploaded files before deadline

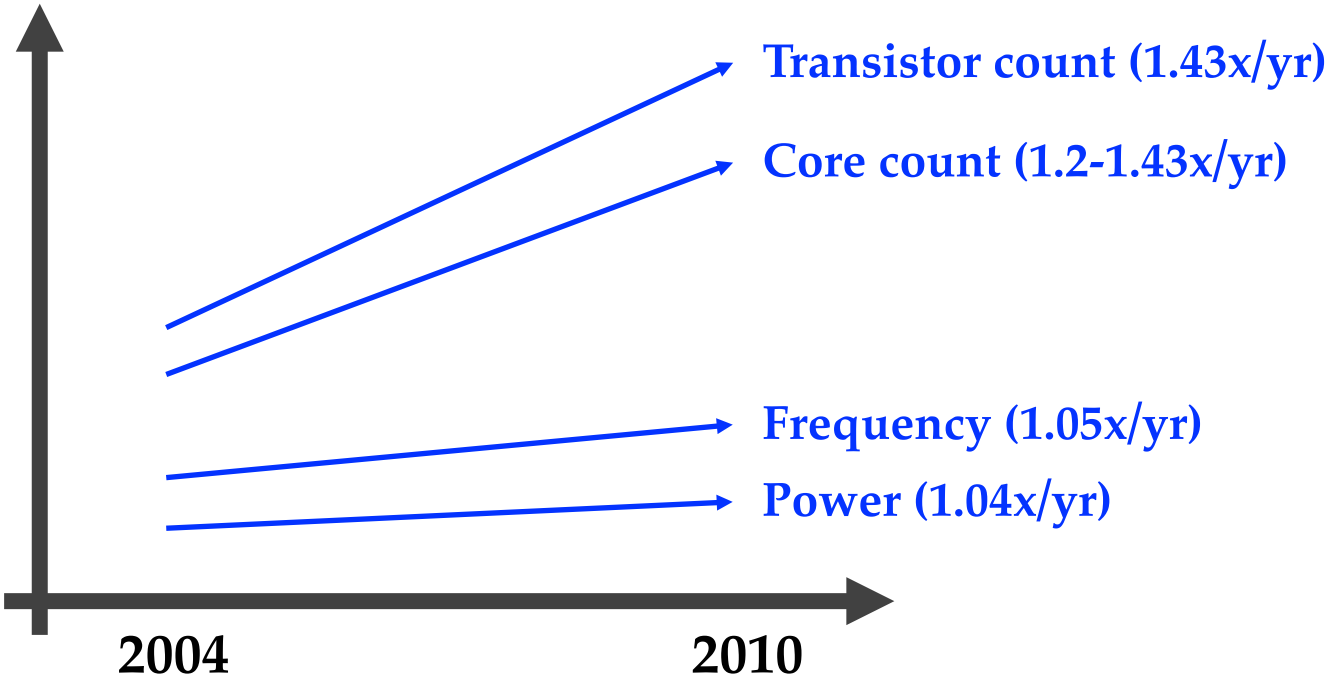
- This lecture
  - Technology trends
  - Measuring performance
  - Principles of computer design
  - Power and energy
  - Cost and reliability

Technology Trends (Historical Data)

- IC logic Technology: on-chip transistor count doubles every 18-24 months (Moore's Law)
  - Transistor density increases by 35% per year
  - Die size increases 10-20% per year
- DRAM Technology
  - Chip capacity increases 25-40% per year
- Flash Storage
  - Chip capacity increases 50-60% per year

Technology Trends (Historical Data)

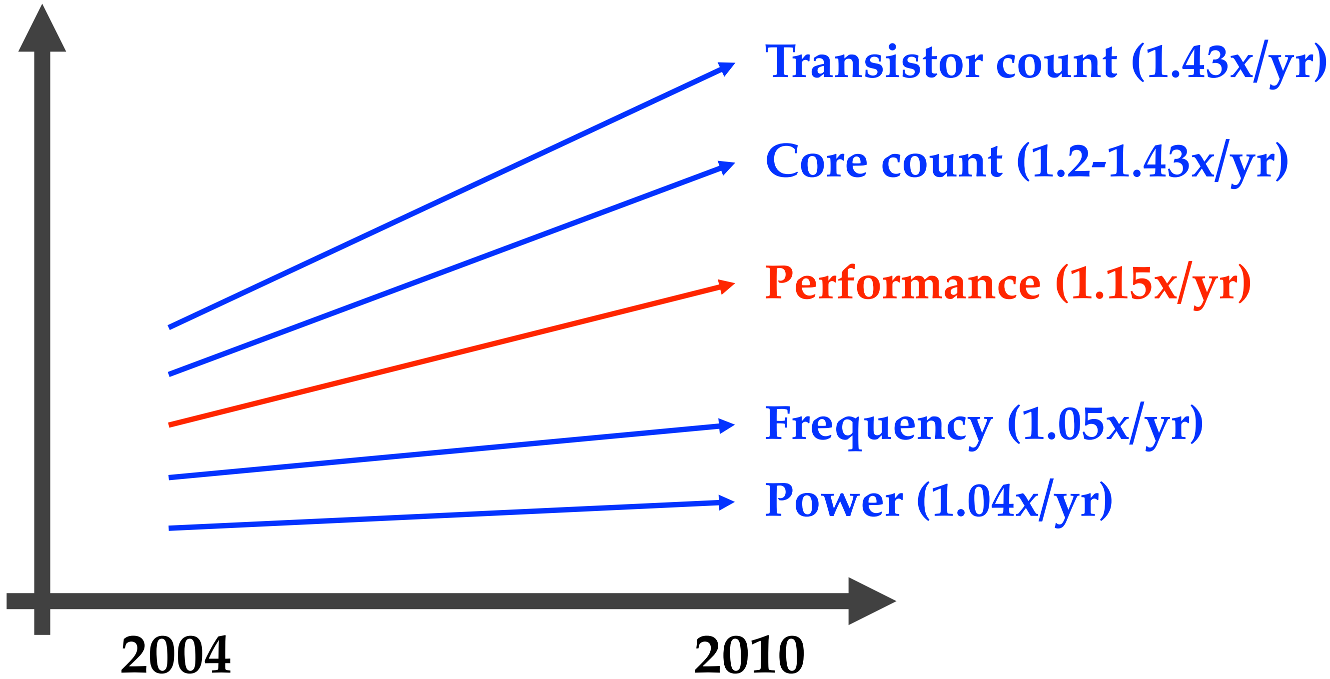
- Recent Microprocessor Trends



Source: Micron University Symposium

Technology Trends (Historical Data)

- Recent Microprocessor Trends



Source: Micron University Symposium

Measuring Performance

- How to measure performance?

Measuring Performance

- How to measure performance?
  - Latency or response time
    - The time between start and completion of an event (e.g., milliseconds for disk access)
  - Bandwidth or throughput
    - The total amount of work done in a given time (e.g., megabytes per second for disk transfer)

Measuring Performance

- How to measure performance?
  - Latency or response time
    - The time between start and completion of an event (e.g., milliseconds for disk access)
  - Bandwidth or throughput
    - The total amount of work done in a given time (e.g., megabytes per second for disk transfer)
- Which one is better? latency or throughput?

Measuring Performance

- Which one is better (faster)?

Car	Bus
Delay=10m	Delay=30m
Capacity=4p	Capacity=30p

Measuring Performance

- Which one is better (faster)?

Car

- Delay=10m
- Capacity=4p
- Throughput=0.4PPM

Bus

- Delay=30m
- Capacity=30p
- Throughput=1PPM

It really depends on your needs (goals).

Measuring Performance

- What program to use for measuring performance?
- Benchmarks Suites
  - A set of representative programs that are likely relevant to the user
  - Examples:
    - SPEC CPU 2017: CPU-oriented programs (for desktops)
    - SPECweb: throughput-oriented (for servers)
    - EEMBC: embedded processors/workloads

Summarizing Performance Numbers

- How to capture the behavior of multiple programs with a single number

	Comp-A	Comp-B	Comp-C
Prog-1	10	5	25
Prog-2	5	10	20
Prog-3	25	10	25

Summarizing Performance Numbers

- How to capture the behavior of multiple programs with a single number

	Comp-A	Comp-B	Comp-C
Prog-1	10	5	25
Prog-2	5	10	20
Prog-3	25	10	25

- AM: Arithmetic Mean (good for times and latencies)

$$\frac{1}{n}\sum_{i=1}^{n} x_i$$

Summarizing Performance Numbers

- How to capture the behavior of multiple programs with a single number

	Comp-A	Comp-B	Comp-C
Prog-1	1/10	1/5	1/25
Prog-2	1/5	1/10	1/20
Prog-3	1/25	1/10	1/25

Summarizing Performance Numbers

- How to capture the behavior of multiple programs with a single number

	Comp-A	Comp-B	Comp-C
Prog-1	1/10	1/5	1/25
Prog-2	1/5	1/10	1/20
Prog-3	1/25	1/10	1/25

- HM: Harmonic Mean (good for rates and throughput)

$$\frac{n}{\sum_{i=1}^{n} \frac{1}{x_i}}$$

Summarizing Performance Numbers

- How to capture the behavior of multiple programs with a single number

	Comp-A	Comp-B	Comp-C
Prog-1	10/10	10/5	10/25
Prog-2	5/5	5/10	5/20
Prog-3	25/25	25/10	25/25

Summarizing Performance Numbers

- How to capture the behavior of multiple programs with a single number

	Comp-A	Comp-B	Comp-C
Prog-1	10/10	10/5	10/25
Prog-2	5/5	5/10	5/20
Prog-3	25/25	25/10	25/25

- GM: Geometric Mean (good for speedups)

$$\left(\prod_{i=1}^{n} x_i\right)^{1/n}$$

Processor Performance

- Clock cycle time (CT = 1/clock frequency)
  - Influenced by technology and pipeline
- Cycles per instruction (CPI)
  - Influenced by architecture
  - IPC may be used instead (IPC = 1/CPI)
- Instruction count (IC)
  - Influenced by ISA and compiler
- CPU time = IC x CPI x CT

Example Problem

- Find the average CPI of a load/store machine when running an application that results in the following statistics

Instruction Type	Frequency	Cycles
Load	20%	2
Store	20%	2
Branch	20%	2
ALU	40%	1

Example Problem

- Find the average CPI of a load/store machine when running an application that results in the following statistics

Instruction Type	Frequency	Cycles
Load	20%	2
Store	20%	2
Branch	20%	2
ALU	40%	1

$$CPI = 0.2x2 + 0.2x2 + 0.2x2 + 0.4x1 = 1.6$$

Example Problem

- Find the average CPI of a load/store machine when running an application that results in the following statistics

Instruction Type	Frequency	Cycles
Load	20%	2
Store	20%	2
Branch	20%	2
ALU	40%	1

- 50% of the branches can be combined with ALU instructions and executed as Branch-ALU fused in 2 cycles. What is the new average CPI?

Example Problem

- Find the average CPI of a load/store machine when running an application that results in the following statistics

Instruction Type	Frequency	Cycles
Load	~22%	2
Store	~22%	2
Branch	~11%	2
ALU	~33%	1
Branch-ALU	~12%	2

- 50% of the branches can be combined with ALU instructions and executed as Branch-ALU fused in 2 cycles. What is the new average CPI? CPI = 1.67

Processor Performance

- Points to note
  - Performance = 1 / execution time
  - AM(IPCs) = 1 / HM(CPIs)
  - GM(IPCs) = 1 / GM(CPIs)

$$\frac{1}{n}\sum_{i=1}^{n} x_i \qquad \frac{n}{\sum_{i=1}^{n} \frac{1}{x_i}} \qquad \left(\prod_{i=1}^{n} x_i\right)^{1/n}$$

Speedup vs. Percentage

- Speedup = old execution time / new execution time
- Improvement = (new performance - old performance)/old performance
- My old and new computers run a particular program in 80 and 60 seconds; compute the followings
  - speedup
  - percentage increase in performance
  - percentage reduction in execution time

Speedup vs. Percentage

- Speedup = old execution time / new execution time
- Improvement = (new performance - old performance)/old performance
- My old and new computers run a particular program in 80 and 60 seconds; compute the followings
  - speedup = 80/60 = ~1.33
  - percentage increase in performance = 33%
  - percentage reduction in execution time = 20/80 = 25%

Example Problem

- The IPC of a new computer is 20% worse than the old one. Its clock speed is 30% higher than the old one. If running the same binaries on both machines. What speedup is the new computer providing?

Example Problem

- The IPC of a new computer is 20% worse than the old one. Its clock speed is 30% higher than the old one. If running the same binaries on both machines. What speedup is the new computer providing?

	OLD	NEW
IPC	1	0.8
Frequency	1	1.3
IC	1	1
CPI	?	?
CT	?	?
CPU Time	?	?

Example Problem

- The IPC of a new computer is 20% worse than the old one. Its clock speed is 30% higher than the old one. If running the same binaries on both machines. What speedup is the new computer providing?

Speedup = 1/0.96 = 1.04

	OLD	NEW
IPC	1	0.8
Frequency	1	1.3
IC	1	1
CPI	1/1	1/0.8 = 1.25
CT	1/1	1/1.3 = ~0.77
CPU Time	1	~0.96

Principles of Computer Design

- Designing better computer systems requires better utilization of resources
  - Parallelism
    - Multiple units for executing partial or complete tasks
  - Principle of locality (temporal and spatial)
    - Reuse data and functional units
  - Common Case
    - Use additional resources to improve the common case