6.893: Advanced VLSI Computer Architecture

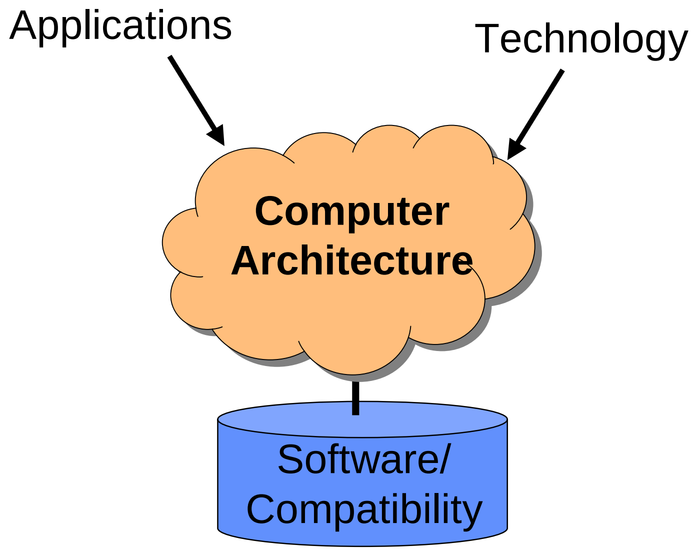
Krste Asanovic

`krste@lcs.mit.edu`

`http://www.cag.lcs.mit.edu/6.893-f2000/`

The goal of this course is to help prepare you for research in computer architecture and related areas, including compilers and VLSI design

The Defining Forces in Computer Architecture



Applications

Technology

Computer Architecture

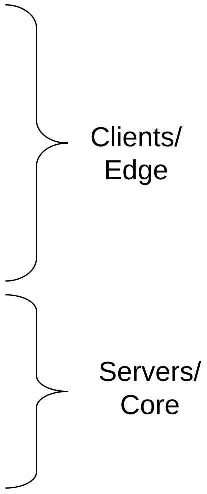
Software/ Compatibility

International Technology Roadmap for Semiconductors ‘99

Year	2005	2008	2011	2014
Technology (nm)	100	70	50	35
DRAM chip area (mm^2)	526	603	691	792
DRAM capacity (Gb)	8		64	
MPU chip area (mm^2)	622	713	817	937
MPU transistors (x10^9)	0.9	2.5	7.0	20.0
MPU Clock Rate (GHz)	3.5	6.0	10.0	13.5

New Computing Applications and Infrastructure

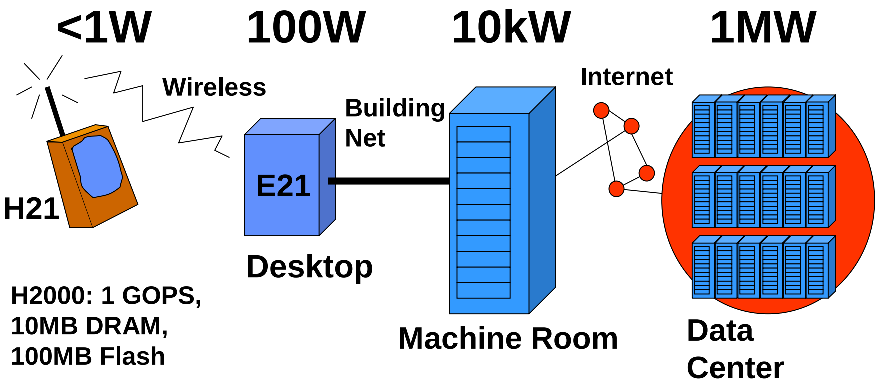
- **Real-time real-world data processing**
  - □ video
  - □ audio
  - □ sensor data
  - □ wireless
- **Human-Machine interfaces**
  - □ speech recognition
  - □ gesture recognition
  - □ language understanding
- **Global-scale servers**
  - □ non-stop service
  - □ secure data storage
- **Networking**
  - □ intelligent routers



Clients/ Edge

Servers/ Core

Computers Defined by Watts not MIPS



H2000: 1 GOPS,
10MB DRAM,
100MB Flash

H2010: 100 GOPS,
1GB DRAM,
10GB Flash/Ferro

(Electricity is 25% of running costs)

The Importance of Volume

- Non-Recurring Engineering (NRE) costs are increasing rapidly for new processor designs
  - □ >$1M for masks to spin a new design
  - □ Engineers cost ~$200K/year (salary+benefits+overhead)
  - □ Pentium Pro design verification took around 350 engineer years or ~$70M

=> Tremendous economies of scale

(Can’t sell <1,000,000 parts for <$100 each)

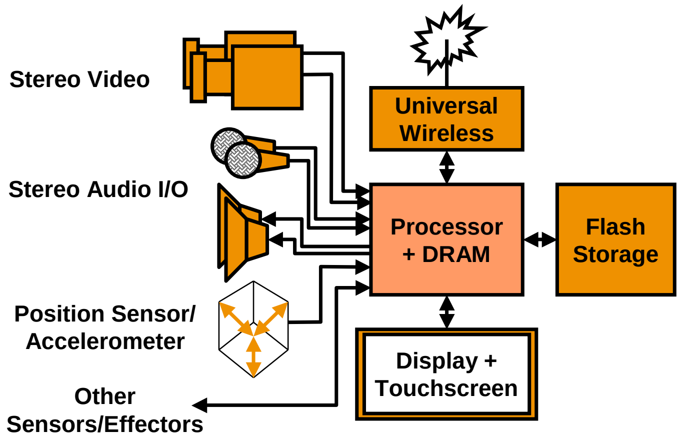
- CMOS following Moore’s Law until 2011-2014
  - □ ITRS’99* roadmap 2011, 50nm technology
    - 64 Gb DRAMs (8 GB/chip)
    - 7 billion transistor CPUs
    - 10 GHz clocks (100 ps cycle time)

=> Smallest viable chips have huge capacity

(~10 million transistors/mm^2)

*[*International Technology Roadmap for Semiconductors]*

Universal Client Devices



Stereo Video

Stereo Audio I/O

Position Sensor/ Accelerometer

Other Sensors/Effectors

Universal Wireless

Processor + DRAM

Flash Storage

Display + Touchscreen

Software-configurable processor array replaces ASICs or collections of DSPs, microprocessors and glue logic

Our Meta-Project This Term

Assume humanity can’t afford to make more than one kind of client chip (plus one kind of memory chip)

Architect ***the*** 10 billion transistor processor chip

(This premise is a little exaggerated but captures the challenge behind future general-purpose computer architecture)

Course Content

Approximately:

1/3 Review and critique previous real machines with novel architectures (some overlap with Spring's 6.911)

1/3 Review and critique of research papers

1/3 Generation and discussion and of new ideas

i.e., your course projects and presentations

Course grading policy:

- 60% Course Project
- 20% Assigned Paper Presentation
- 20% Class Participation

Course Project

- **Work in groups of 2 or 3 (can go solo with permission)**
- **Preferably in an area related to your research interests**
- **Final result: 10 page conference paper + 20 minute presentation**
- **Staged project deadlines:**
  - □ **September 26 (19 days time): Project proposal + presentation**
  - □ **October 19: First project checkpoint + presentation**
  - □ **November 9: Second project checkpoint + presentation**
  - □ **December 5/7: Final project presentations**
  - □ **December 12, 5pm: Final project writeup due in NE43-617**
- **Each student in group must give at least one project presentation**
- **Your work will be made publicly available through class web site**

First Project Deadline: September 26 (19 days time)

- Find topic (check class web page for project ideas)
- Find group partners (preferably with complementary expertise)
- Prepare one page written proposal
  - □ identify topic
  - □ identify tools and research infrastructure
  - □ give plan of work
- Prepare 5-minute, 2-slide presentation for class

Readings

- Each student will lead a 30 minute discussion session for at least one assigned paper
- Prepare ~10 minute (5 slide) overview and critique of each paper
- Other students must read paper and bring comments and questions to class (you will be asked for comments!)
- Papers available online or in NE43-624 one week before class
- First reading sessions: *Alpha 21264 case study*
  - □ September 14 a) microarchitecture b) performance
  - □ September 19 c) overall VLSI design d) out-of-order circuitry
- First four volunteers?

How do we compare two designs?

Cost of Processor

- **Design cost (Non-recurring Engineering Costs, NRE)**
  - □ **dominated by engineer-years (~$200K per engineer year)**
  - □ **also mask costs (approaching $1M per spin)**
- **Cost of die**
  - □ **die area**
  - □ **die yield (maturity of manufacturing process, redundancy features)**
  - □ **cost/size of wafers**
  - □ **die cost ~= f(die area^4) with no redundancy**
- **Cost of packaging**
  - □ **number of pins (signal + power/ground pins)**
  - □ **power dissipation**
- **Cost of testing**
  - □ **built-in test features?**
  - □ **logical complexity of design**
  - □ **choice of circuits (minimum clock rates, leakage currents, I/O drivers)**

Architect affects all of these

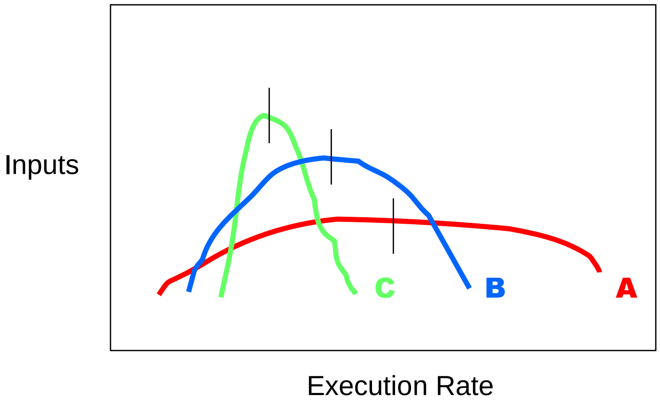
System-Level Cost Impacts

- Power supply and cooling
- Support chipset
- Off-chip SRAM/DRAM/ROM
- Off-chip peripherals

What is Performance?

- Latency (or response time or execution time)
  - time to complete one task
- Bandwidth (or throughput)
  - tasks completed per unit time

Performance Guarantees



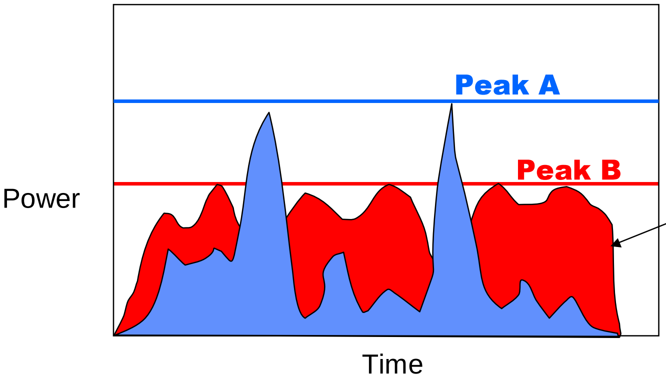
Average Rate: **A** > **B** > **C**

Worst-case Rate: **A** < **B** < **C**

Power and Energy

- **Energy to complete operation (Joules)**
  - □ **Corresponds approximately to battery life**
  - □ **(Battery energy capacity actually depends on rate of discharge)**
- **Peak power dissipation (Watts = Joules/second)**
  - □ **Affects packaging (power and ground pins, thermal design)**
- **di/dt, peak change in supply current (Amps/second)**
  - □ **Affects power supply noise (power and ground pins, decoupling capacitors)**

Peak Power versus Lower Energy



Integrate power curve to get energy

- System **A** has higher peak power, but lower total energy
- System **B** has lower peak power, but higher total energy

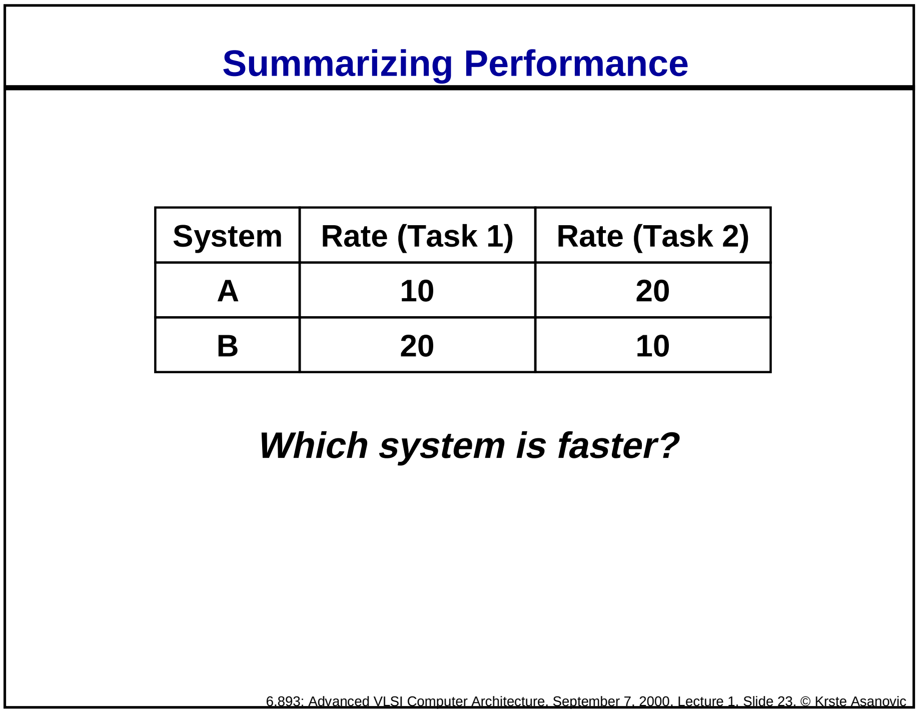
Metrics Summary

- **Cost**
  - □ **Die cost and system cost**
- **Execution Time**
  - □ **average and worst-case**
- **Energy**
  - □ **Also peak power and peak switching current**
- **Reliability**
  - □ **Electrical noise**
  - □ **Robustness to bad software**
- **Maintainability**
  - □ **System administration costs**
- **Compatibility**
  - □ **Software costs dominate**

What is a “General-Purpose” Machine?

Types of Benchmark

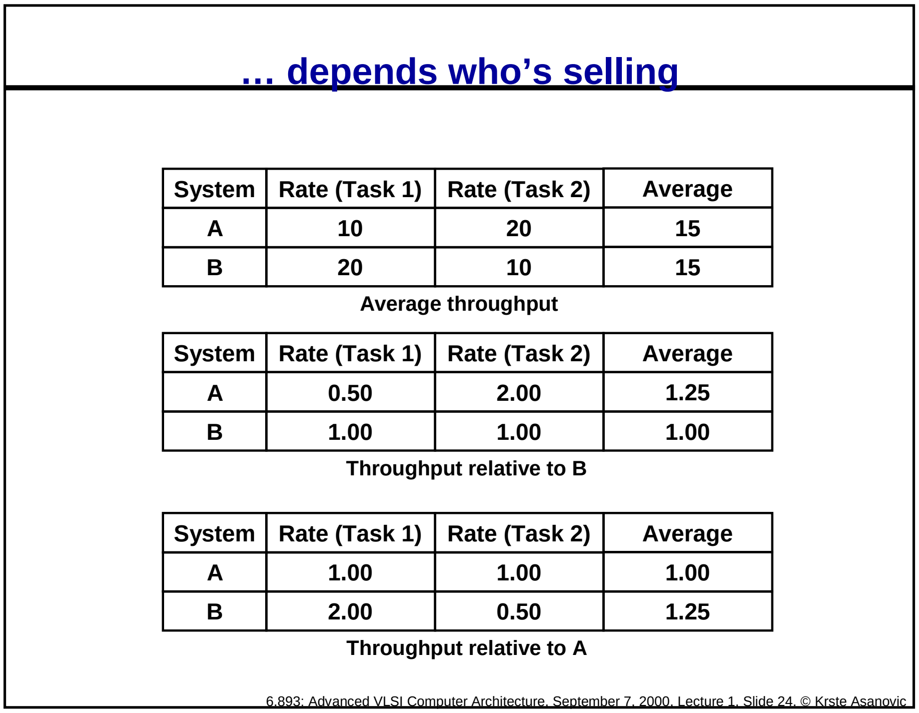
- Synthetic Benchmarks
  - Designed to have same mix of operations as real workloads, e.g., Dhrystone, Whetstone
- Toy Programs
  - Small, easy to port. Output often known before program is run, e.g., Nqueens, Bubblesort, Towers of Hanoi
- Kernels
  - Common subroutines in real programs, e.g., matrix multiply, FFT, sorting, Livermore Loops, Linpack
- Simplified Applications
  - Extract main computational skeleton of real application to simplify porting, e.g., NAS parallel benchmarks, TPC
- Real Applications
  - Things people actually use their computers for, e.g., car crash simulations, relational databases, Photoshop, Quake



Summarizing Performance

System	Rate (Task 1)	Rate (Task 2)
A	10	20
B	20	10

Which system is faster?



… depends who's selling

System	Rate (Task 1)	Rate (Task 2)	Average
A	10	20	15
B	20	10	15

Average throughput

System	Rate (Task 1)	Rate (Task 2)	Average
A	0.50	2.00	1.25
B	1.00	1.00	1.00

Throughput relative to B

System	Rate (Task 1)	Rate (Task 2)	Average
A	1.00	1.00	1.00
B	2.00	0.50	1.25

Throughput relative to A

Summarizing Performance over Set of Benchmark Programs

Arithmetic mean of execution times t_i (in seconds)

$$1/n \ \Sigma_i \ t_i$$

Harmonic mean of execution rates r_i (MIPS/MFLOPS)

$$n/ \ [\Sigma_i \ (1/r_i)]$$

- Both equivalent to workload where each program is run the same number of times
- Can add weighting factors to model other workload distributions

Normalized Execution Time and Geometric Mean

- Measure speedup up relative to reference machine

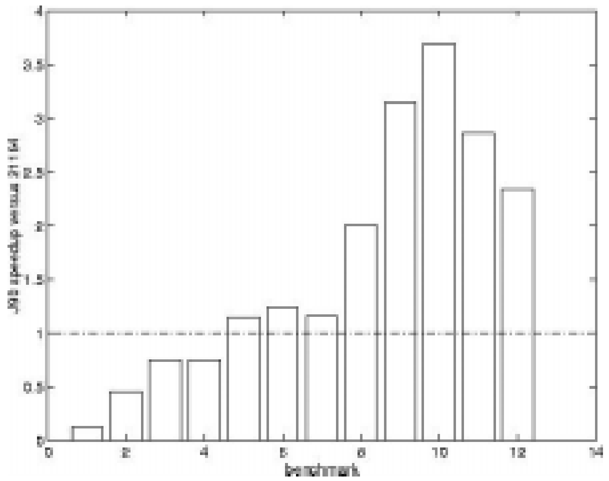
$$\text{ratio} = t_{Ref}/t_A$$

- Average time ratios using geometric mean

$$\sqrt[n]{(\prod_I \text{ratio}_i \,)}$$

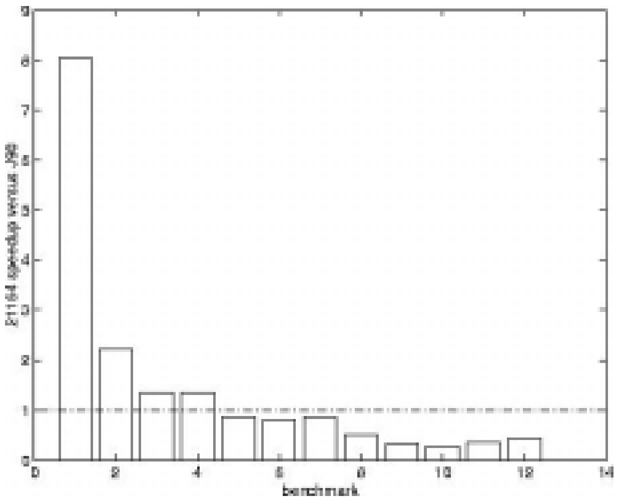
- Insensitive to machine chosen as reference
- Insensitive to run time of individual benchmarks
- Used by SPEC89, SPEC92, SPEC95, ...

Vector/Superscalar Speedup



- 100 MHz Cray J90 vector machine versus 300MHz Alpha 21164
- LANL Computational Physics Codes, Wasserman, ICS’96

Superscalar/Vector Speedup



- 100 MHz Cray J90 vector machine versus 300 MHz Alpha 21164
- LANL Computational Physics Codes, Wasserman, ICS’96

How to Mislead with Performance Reports

- Select pieces of workload that work well on your design, ignore others
- Use unrealistic data set sizes for application (too big or too small)
- Report throughput numbers for a latency benchmark
- Report latency numbers for a throughput benchmark
- Report performance on a kernel and claim it represents an entire application
- Use 16-bit fixed-point arithmetic (because it’s fastest on your system) even though application requires 64-bit floating-point arithmetic
- Use a less efficient algorithm on the competing machine
- Report speedup for an inefficient algorithm (bubblesort)
- Compare hand-optimized assembly code with unoptimized C code
- Compare your design using next year’s technology against competitor’s year old design (1% performance improvement per week)
- Ignore the relative cost of the systems being compared
- Report averages and not individual results
- Report speedup over unspecified base system, not absolute times
- Report efficiency not absolute times
- Report MFLOPS not absolute times (use inefficient algorithm)

[David Bailey “Twelve ways to fool the masses when giving performance results for parallel supercomputers”]