

# CSE 141-- Introduction to Computer Architecture

Allan Snavelly

*CSE 141*

*Allan Snavelly*

## What is Computer Architecture? (am I in the right class?)

- Hardware Designer
  - thinks about circuits, components, timing, functionality, ease of debugging
- Computer Architect
  - thinks about high-level components, how they fit together, how they work together to deliver performance.

“construction engineer”

“building architect”

*CSE 141*

*Allan Snavelly*

## Why do I care?

- **You may actually do computer architecture someday**
- **You may actually care about software performance someday**
  - The ability of application programs, compilers, operating systems, etc. to deliver performance depends critically on an understanding of the underlying computer organization.
  - That becomes more true every year.
  - Computer architectures become more difficult to understand every year.

## Which is faster?

```
for (i=0; i<N; i=i+1)
  for (j=0; j<N; j=j+1) {
    r = 0;
    for (k=0; k<N; k=k+1)
      r = r + y[i][k] * z[k][j];
    x[i][j] = r;
  }

for (jj=0; jj<N; jj=jj+B)
  for (kk=0; kk<N; kk=kk+B)
    for (i=0; i<N; i=i+1) {
      for (j=jj; j<min(jj+B-1,N); j=j+1)
        r = 0;
      for (k=kk; k<min(kk+B-1,N); k=k+1)
        r = r + y[i][k] * z[k][j];
      x[i][j] = x[i][j] + r;
    }
```

## Which is faster?

load R1, addr1	→	load R1, addr1
store R1, addr2	→	add R0, R2 -> R3
add R0, R2 -> R3	→	add R0, R6 -> R7
subtract R4, R3 -> R5	→	store R1, addr2
add R0, R6 -> R7	→	subtract R4, R3 -> R5
store R7, addr3	→	store R7, addr3

CSE 141

Allan Snively

## Which is faster?

loop1: add ...	loop1: add ...
load ...	load ...
add ...	add ...
bne R1, loop1	bne R1, loop1
loop2: add ...	nop
load ...	nop
bne R2, loop2	loop2: add ...
	load ...
	bne R2, loop2

CSE 141

Allan Snively

## Administration

- Instructor -- Dr. Allan Snaveley
- Course webpages [www.sdsc.edu/~allans/cse141.html](http://www.sdsc.edu/~allans/cse141.html)
- TAs:
  - Mrunal Kulkarni (teaches section)
  - Vivek Manpuria
  - Rahul Lahoti
- Lab Consultants:
  - Tim Pevzner
- Grading 40% midterm, 10% class participation, 50% final
- Homework
- Relationship of lab and lecture
- The course workload

CSE 141

Allan Snaveley

## What is Computer Architecture?

Computer Architecture =  
Machine Organization +  
Instruction Set Architecture

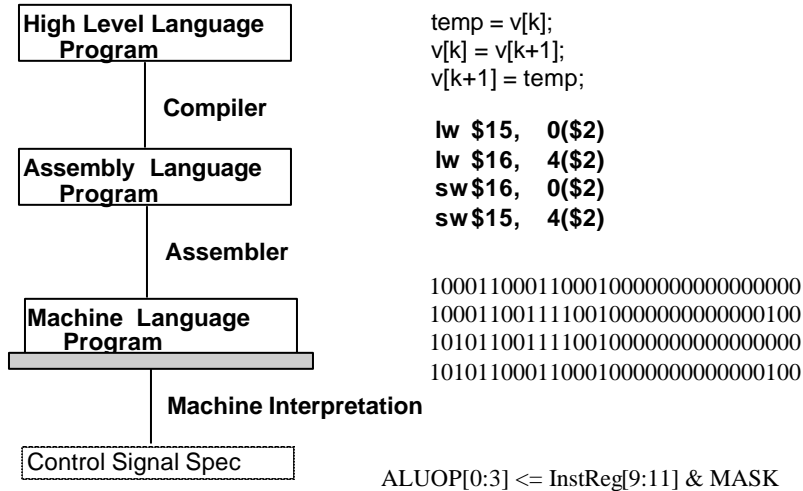
▶ *What the machine looks like*

▶ *How you talk to the machine*

CSE 141

Allan Snaveley

## How to Speak Computer



CSE 141

Allan Snively

## The Instruction Set Architecture

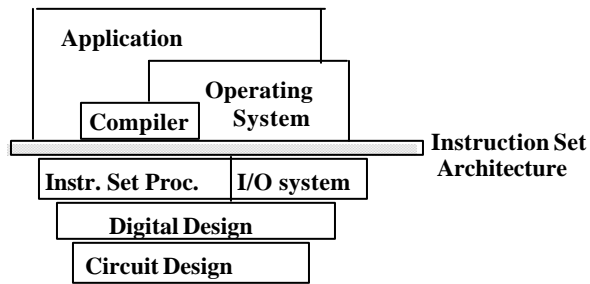
- that part of the architecture that is visible to the programmer
  - opcodes (available instructions)
  - number and types of registers
  - instruction formats
  - storage access, addressing modes
  - exceptional conditions

CSE 141

Allan Snively

## The Instruction Set Architecture

° is the agreed-upon interface between all the software that runs on the machine and the hardware that executes it.



CSE 141

Allan Snively

## Examples of ISAs

- Alpha AXP
- Intel 80x86/pentium
- VAX
- MIPS
- SPARC
- IBM 360
- Intel IA-64 (Itanium)
- PowerPC

CSE 141

Allan Snively

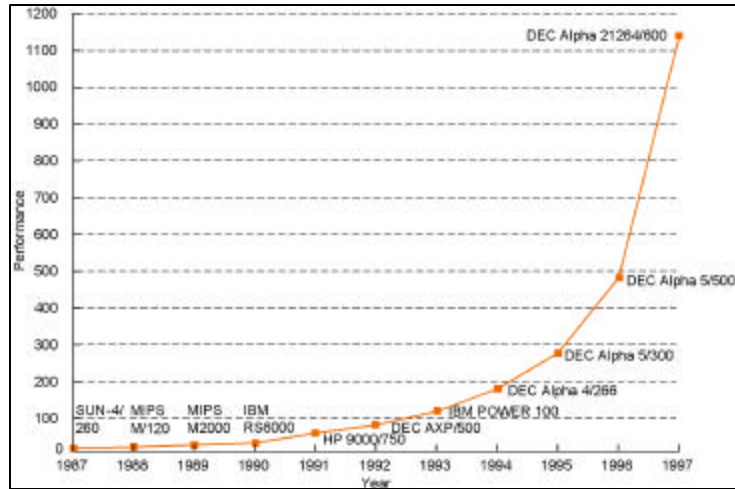
## Computer Organization

- Once you have decided on an ISA, you must decide how to design the hardware to execute those programs written in the ISA as fast as possible.
- This must be done every time a new implementation of the architecture is released, with typically very different technological constraints.

## The Challenge of Computer Architecture

- The industry changes faster than any other.
- The ground rules change every year.
  - new problems
  - new opportunities
  - different tradeoffs
- It's all about making programs run faster than the next guy's machine.

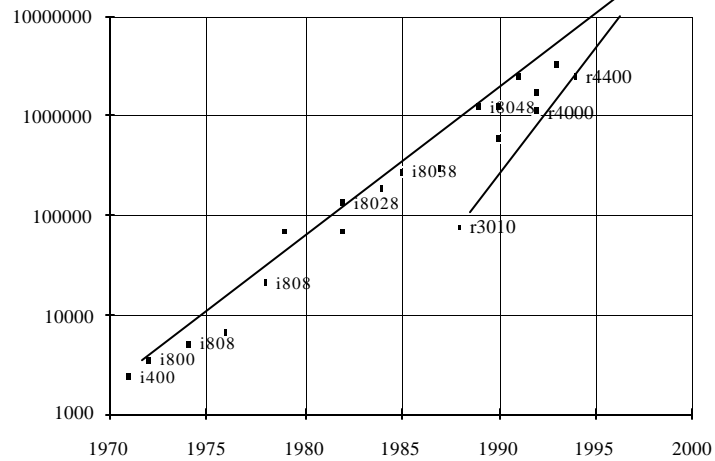
## Performance Trends



CSE 141

Allan Snively

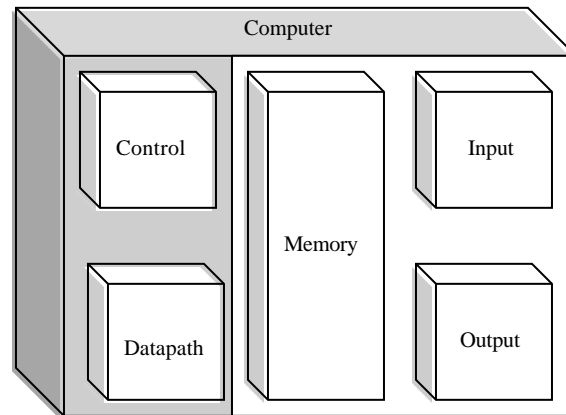
## Technology: Microprocessor Logic Density



CSE 141

Allan Snively

## The five classic components of computers



*CSE 141*

*Allan Snively*

## Course Outline

- I. Instruction Set Architecture
- II. Computer System Performance and Performance Metrics
- III. Computer Arithmetic and Number Systems
- IV. CPU Architecture
- V. Pipelining
- VI. Superscalars
- VII. The Memory/Cache Hierarchy
- VIII. Parallel Machines

*CSE 141*

*Allan Snively*

## What you can expect to get out of this class

- to become conversant with computer architecture terms and concepts.
- to understand fundamental concepts in computer architecture and how they impact computer and application performance.
- to be able to read and evaluate architectural descriptions of even today's most complex processors.
- to gain experience designing a working CPU completely from scratch.
- to learn experimental techniques used to evaluate advanced architectural ideas.

## Key Points

- High-performance software requires a deep understanding of the underlying machine organization.
- The instruction set architecture defines how software is allowed to use the processor. Multiple computers with vastly different organizations and performance can share an ISA.
- Most every component in a computer system falls into one of five categories.