Lecture 8
Computer Architecture Perspective

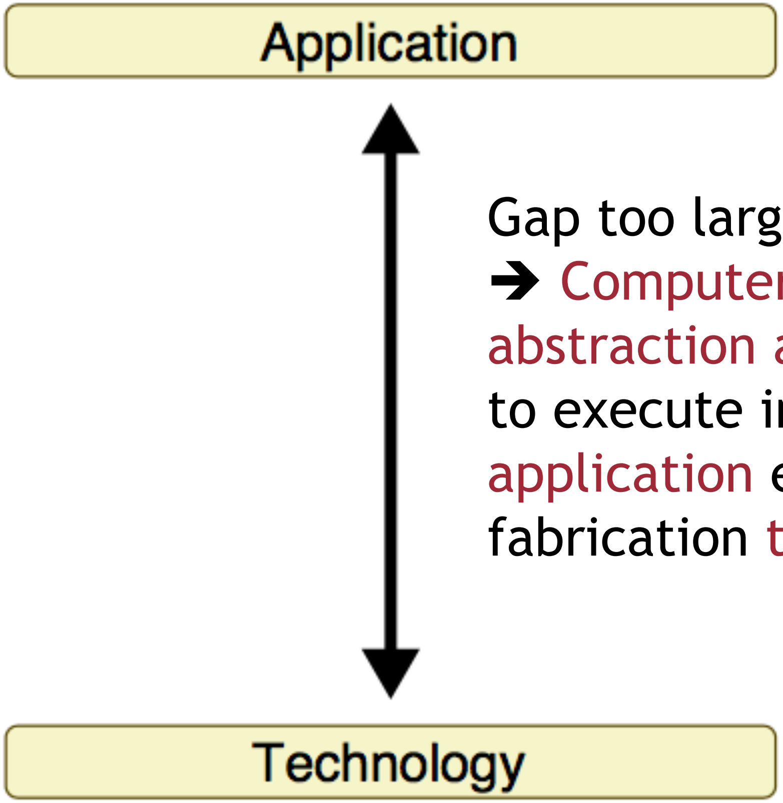
Xuan ‘Silvia’ Zhang

Washington University in St. Louis

http://classes.engineering.wustl.edu/ese566/

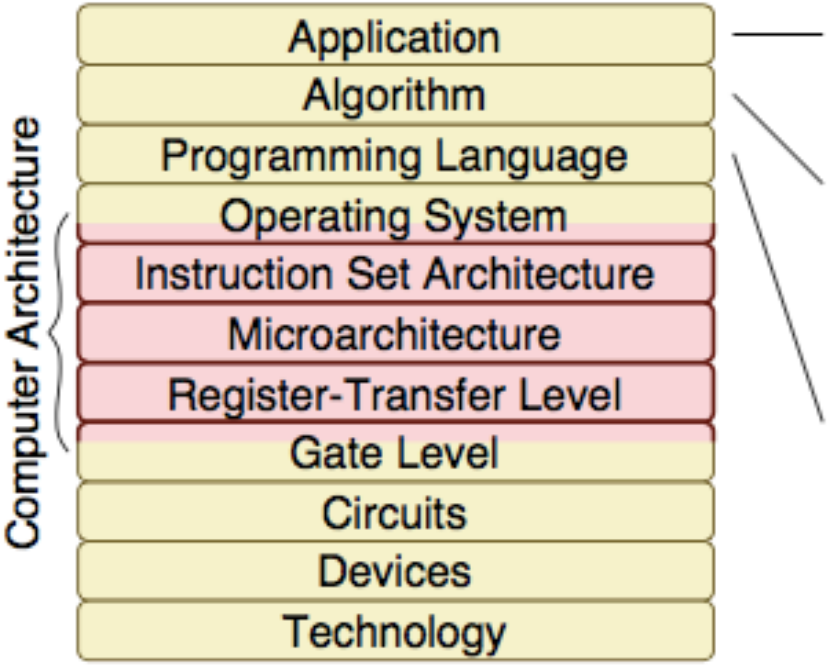
Computer Architecture Definition

- Bridge application and technology
  - examples
  - app: navigation (North), tech: magnetic compass
  - app: face recognition, tech: charge-coupled device (CCD)



Gap too large to bridge in one step
➔ Computer Architecture: develop abstraction and implementation layers to execute information processing application efficiently using available fabrication technology

Computer System Stack



Computer Architecture

Application

Algorithm

Programming Language

Operating System

Instruction Set Architecture

Microarchitecture

Register-Transfer Level

Gate Level

Circuits

Devices

Technology

Sort an array of numbers

2,6,3,8,4,5 -> 2,3,4,5,6,8

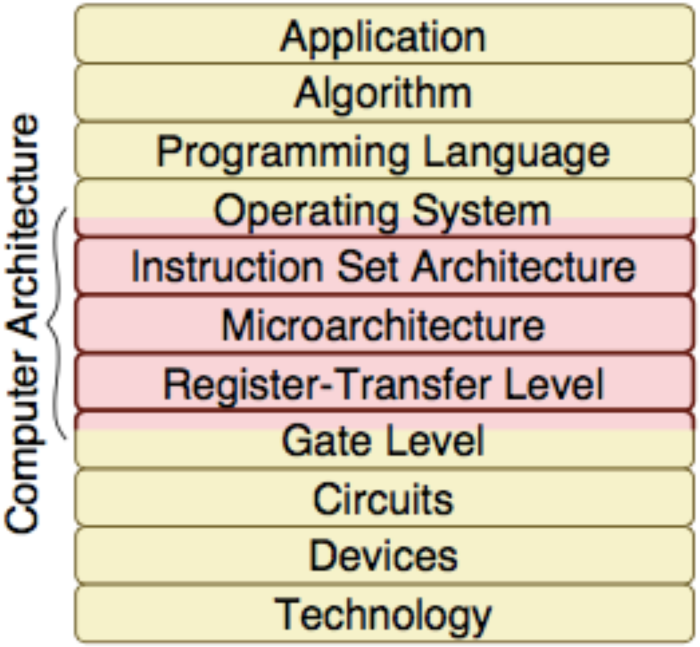
Out-of-place selection sort algorithm

1. Find minimum number in array
2. Move minimum number into output array
3. Repeat steps 1 and 2 until finished

C implementation of selection sort

```
void sort( int b[], int a[], int n ) {
  for ( int idx, k = 0; k < n; k++ ) {
    int min = 100;
    for ( int i = 0; i < n; i++ ) {
      if ( a[i] < min ) {
        min = a[i];
        idx = i;
      }
    }
    b[k]   = min;
    a[idx] = 100;
  }
}
```

Computer System Stack



Mac OS X, Windows, Linux

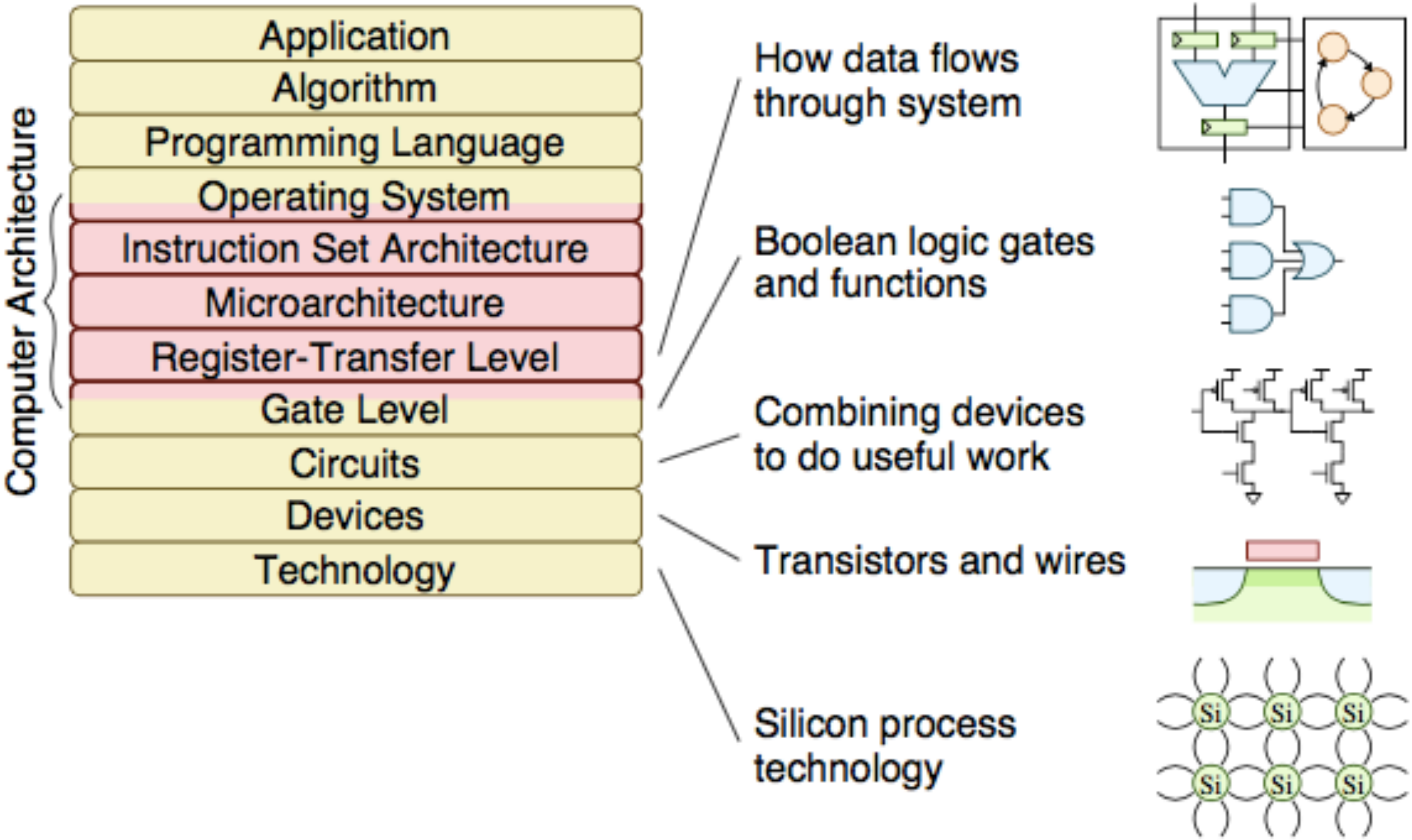
Handles low-level hardware management

MIPS32 Instruction Set

Instructions that machine executes

```
blez  $a2, done
move  $a7, $zero
li    $t4, 99
move  $a4, $a1
move  $v1, $zero
li    $a3, 99
lw    $a5, 0($a4)
addiu $a4, $a4, 4
slt   $a6, $a5, $a3
movn  $v0, $v1, $a6
addiu $v1, $v1, 1
movn  $a3, $a5, $a6
```

Computer System Stack



Application

Algorithm

Programming Language

Operating System

Instruction Set Architecture

Microarchitecture

Register-Transfer Level

Gate Level

Circuits

Devices

Technology

Computer Architecture

How data flows through system

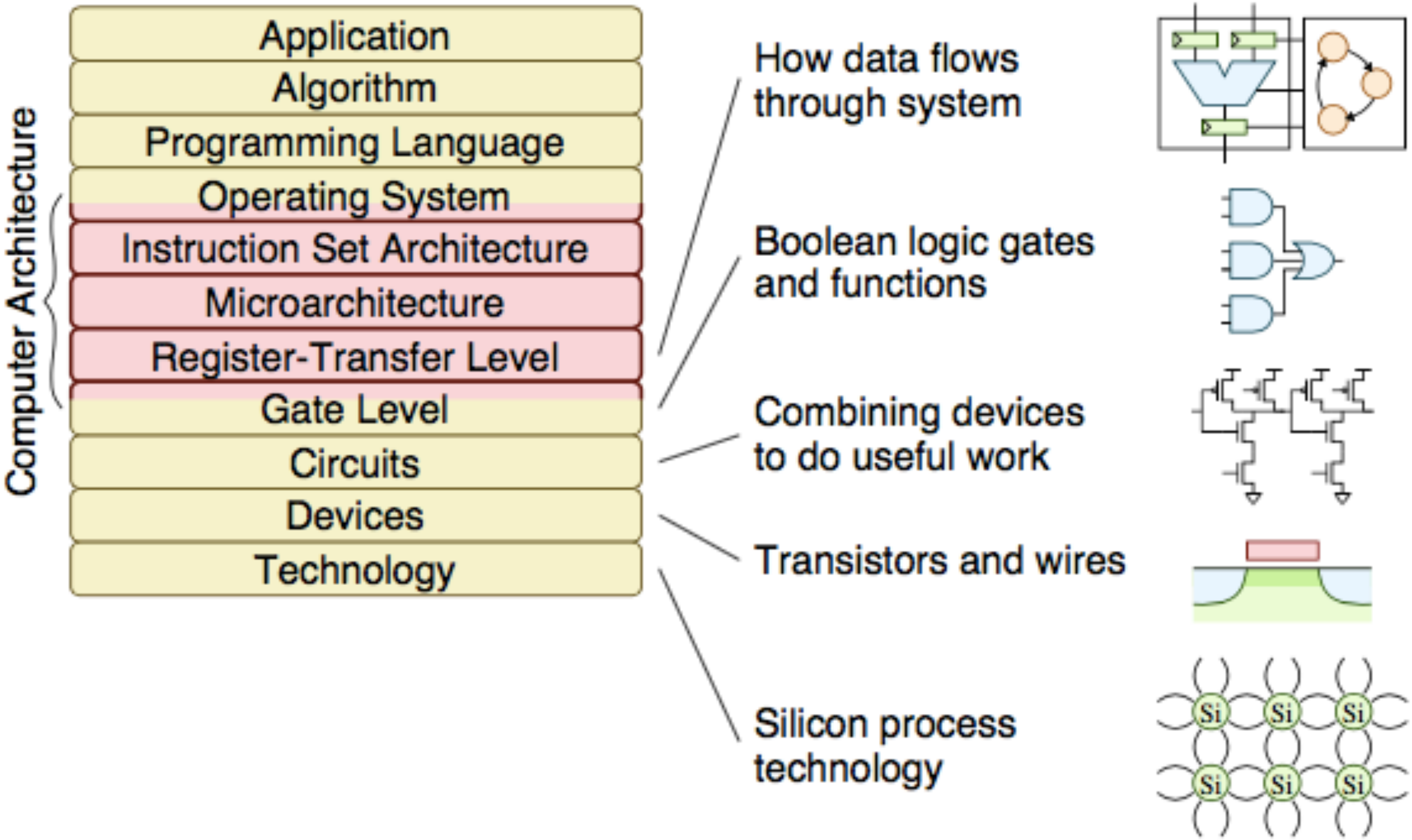
Boolean logic gates and functions

Combining devices to do useful work

Transistors and wires

Silicon process technology

Computer System Stack



Application

Algorithm

Programming Language

Operating System

Instruction Set Architecture

Microarchitecture

Register-Transfer Level

Gate Level

Circuits

Devices

Technology

Computer Architecture

How data flows through system

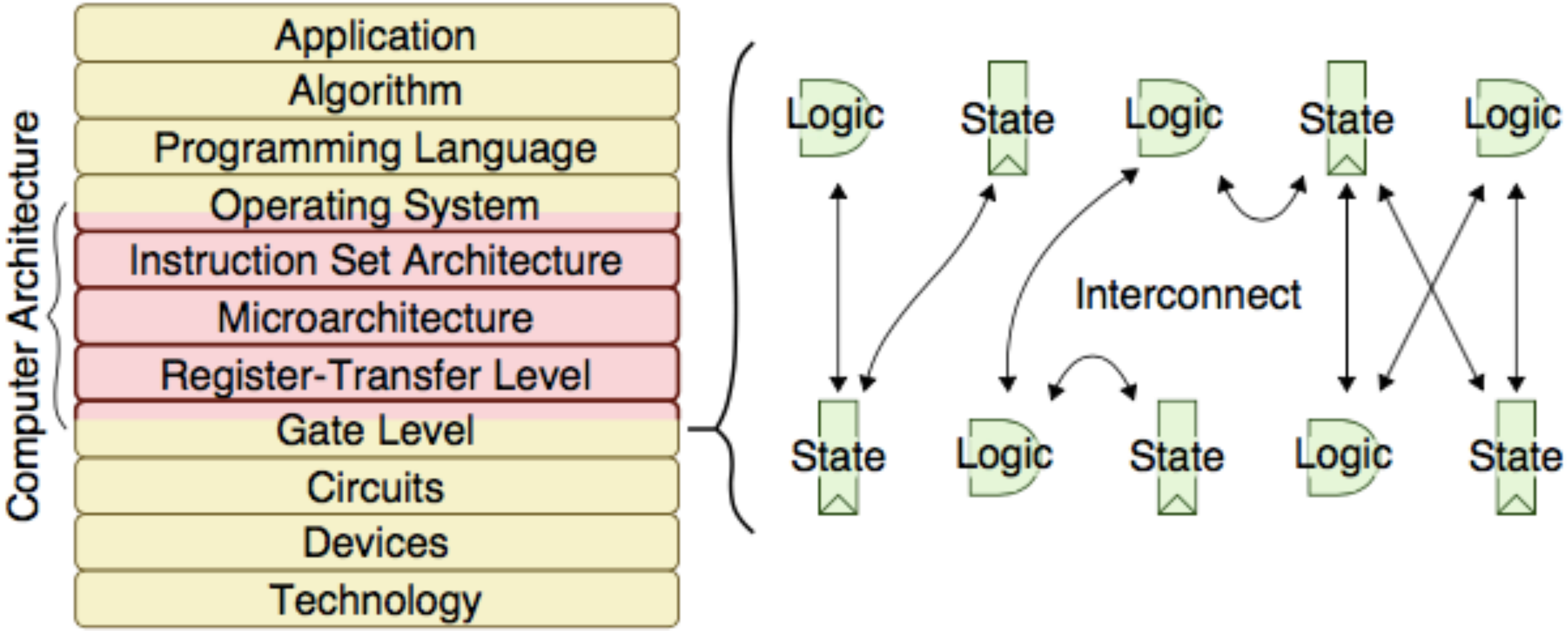
Boolean logic gates and functions

Combining devices to do useful work

Transistors and wires

Silicon process technology

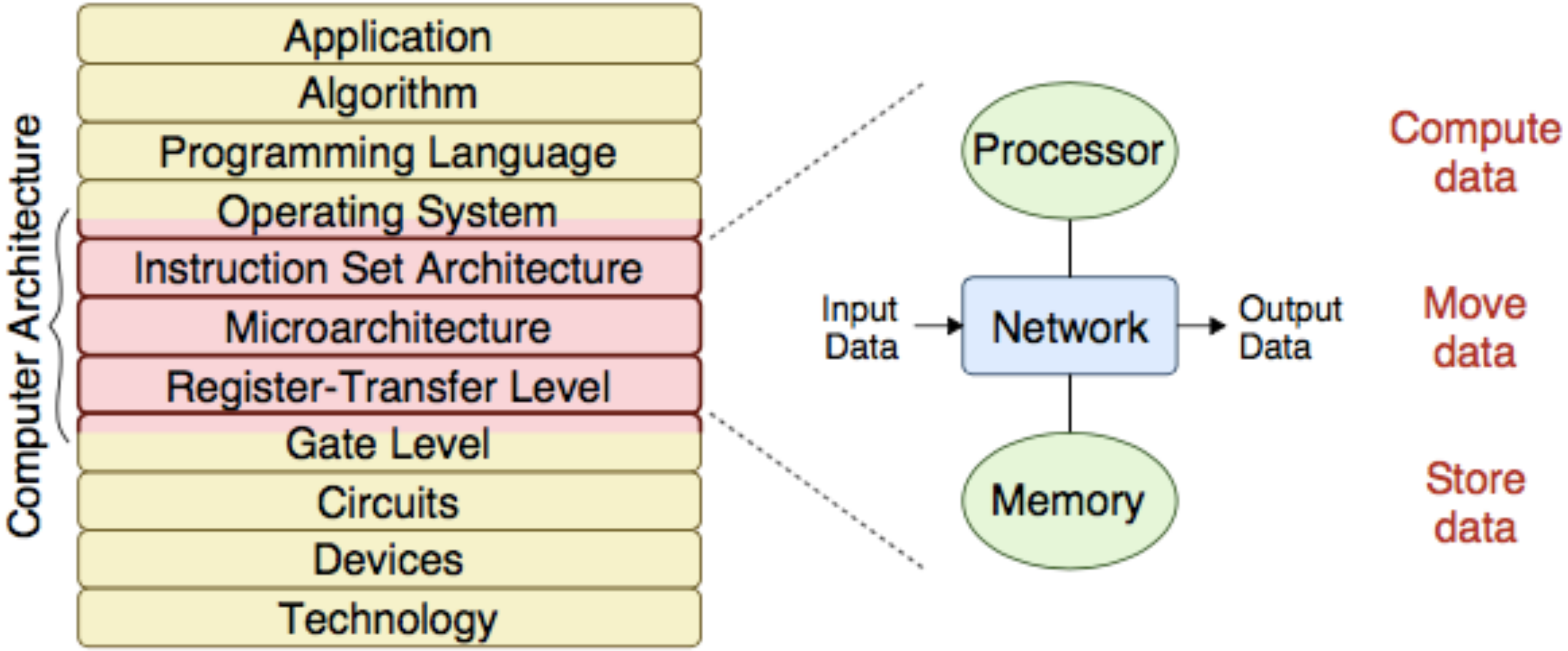
Logic, State, and Interconnect



Digital systems are implemented with three basic building blocks

- Logic to process data
- State to store data
- Interconnect to move data

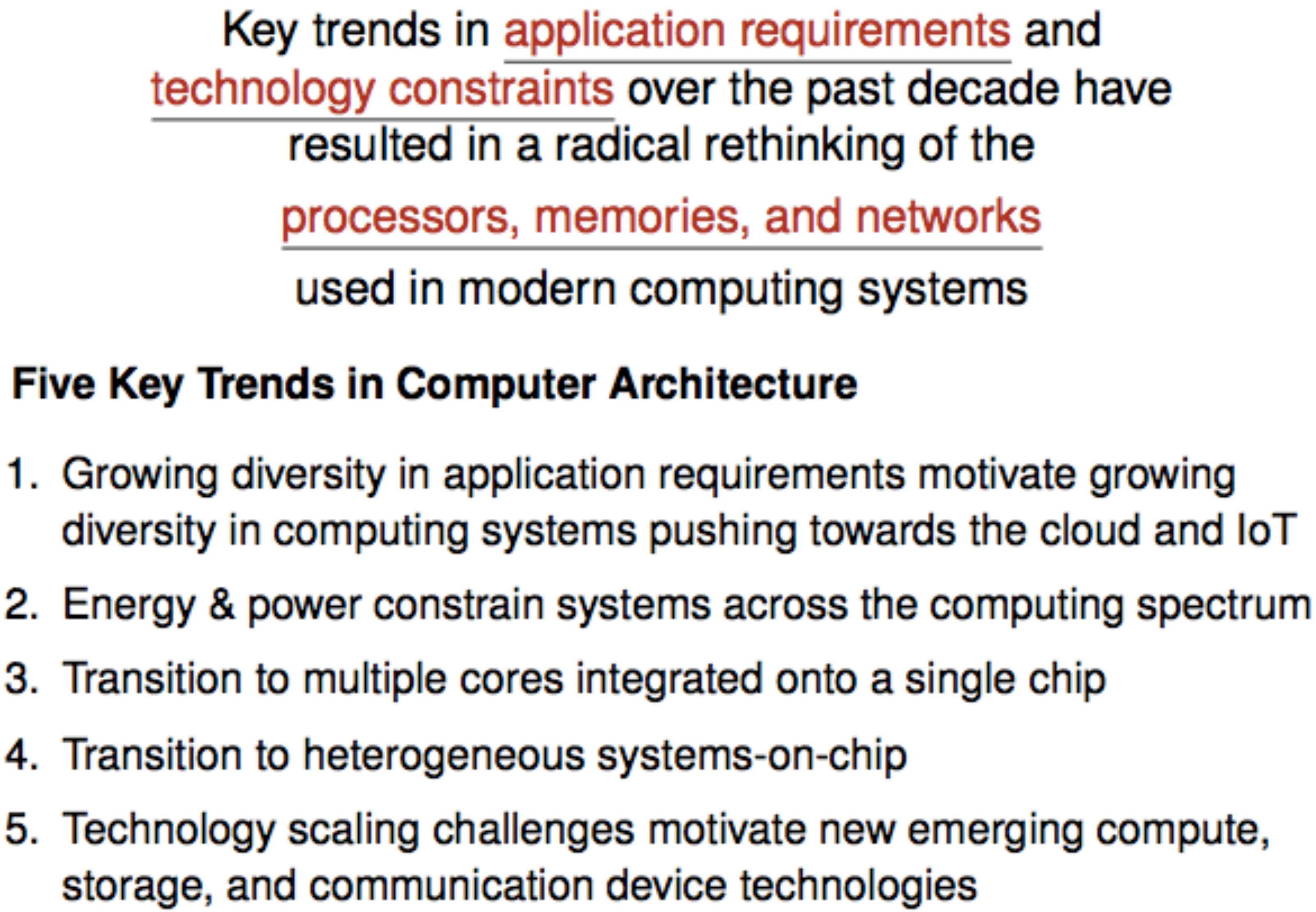
General-Purpose Computing: Processors, Memories, and Networks



Computer engineering basic building blocks

- Processors for computation
- Memories for storage
- Networks for communication

Key Trends in Computer Architecture



Key trends in application requirements and technology constraints over the past decade have resulted in a radical rethinking of the

processors, memories, and networks

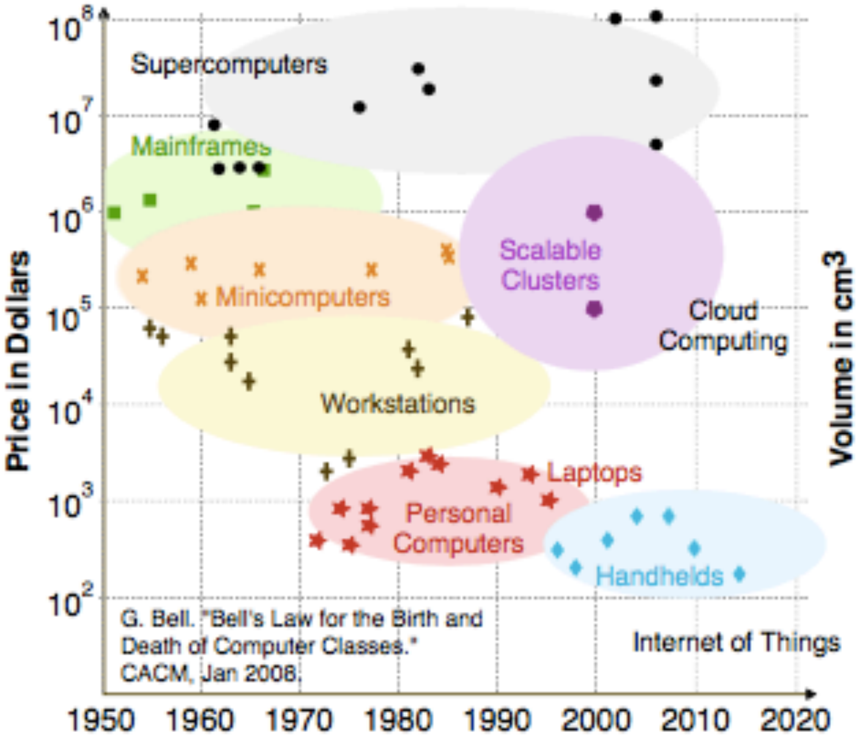
used in modern computing systems

Five Key Trends in Computer Architecture

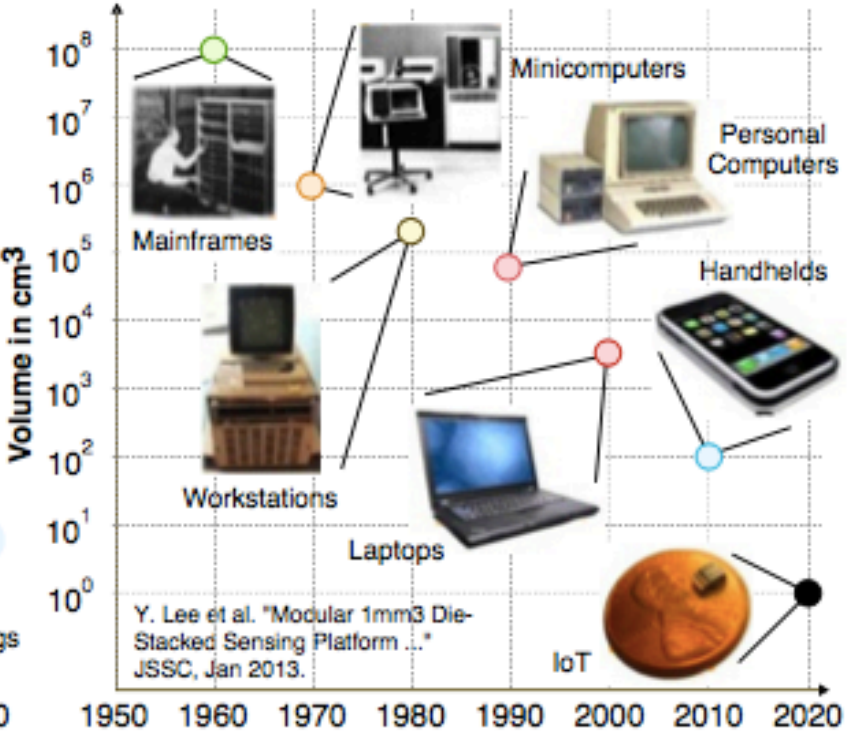
1. Growing diversity in application requirements motivate growing diversity in computing systems pushing towards the cloud and IoT
2. Energy & power constrain systems across the computing spectrum
3. Transition to multiple cores integrated onto a single chip
4. Transition to heterogeneous systems-on-chip
5. Technology scaling challenges motivate new emerging compute, storage, and communication device technologies

Trend 1: Bell’s Law

Roughly every decade a new, smaller, lower priced computer class forms based on a new programming platform resulting in entire new industries



G. Bell. "Bell's Law for the Birth and Death of Computer Classes." CACM, Jan 2008.



Y. Lee et al. "Modular 1mm3 Die-Stacked Sensing Platform ..." JSSC, Jan 2013.

Trend 2: Energy and Power Efficiency in Computing

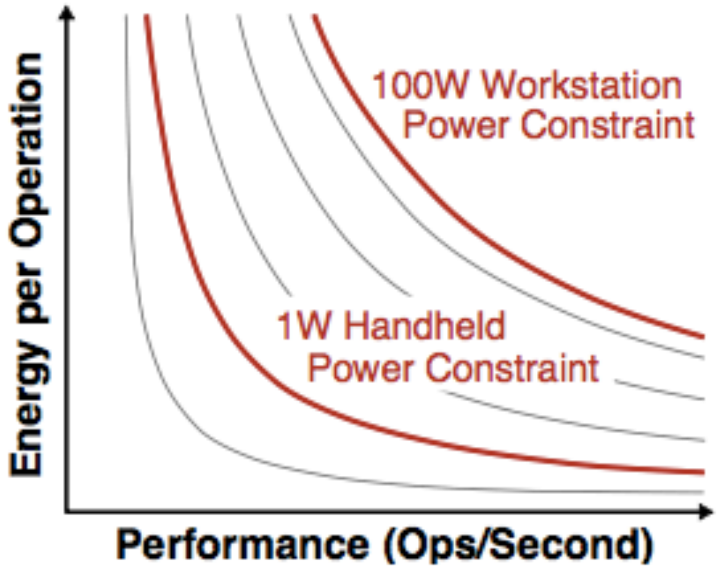
$$\text{Power} = \frac{\text{Energy}}{\text{Second}} = \frac{\text{Energy}}{\text{Op}} \times \frac{\text{Ops}}{\text{Second}}$$

Power

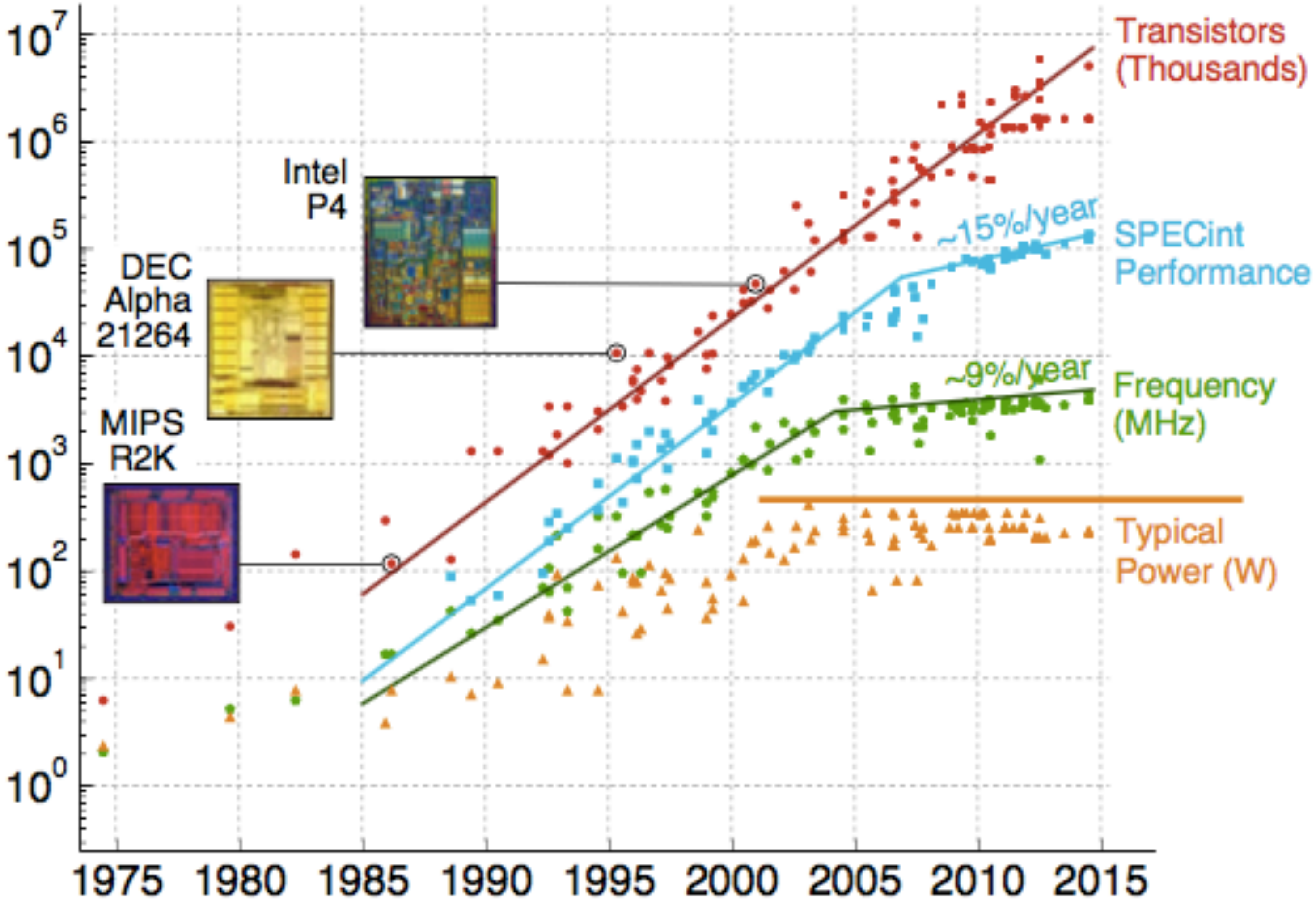
Chip Packaging
Chip Cooling
System Noise
Case Temperature
Data-Center Air Conditioning

Energy

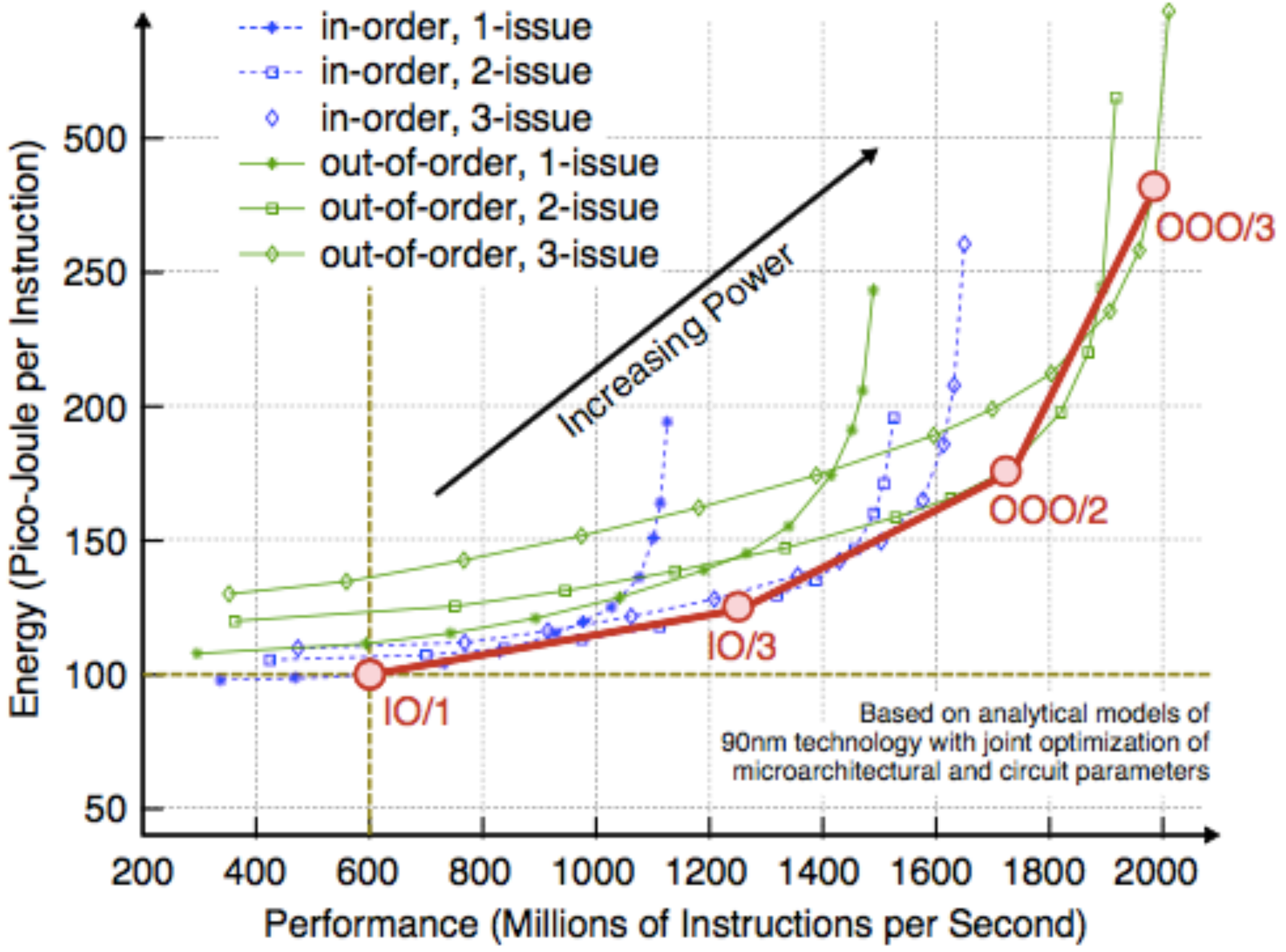
Battery Life
Electricity Bill
Mobile Device Weight



Trend 2: Energy and Power Efficiency in Computing



Trend 2: Energy and Power Efficiency in Computing

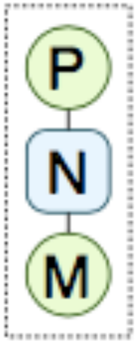


Based on analytical models of 90nm technology with joint optimization of microarchitectural and circuit parameters

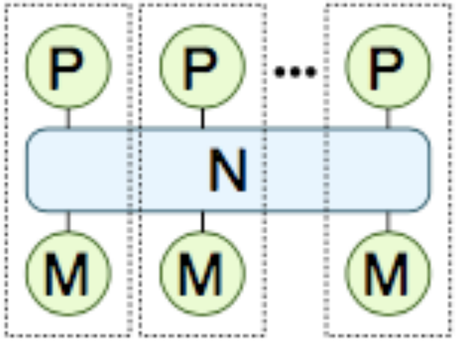
Adpated from O. Azizi et al. "Energy-Performance Tradeoffs ..." ISCA, 2010.

Trend 3: Manycore Processor Architecture

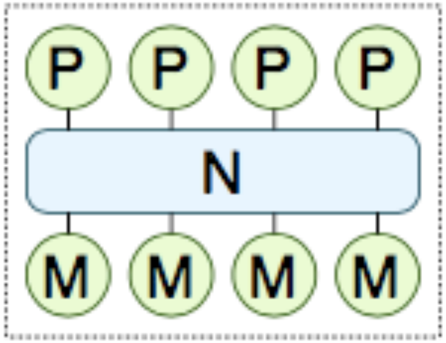
Intel Pentium 4
Single monolithic processor

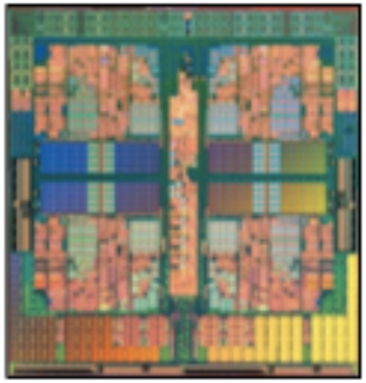


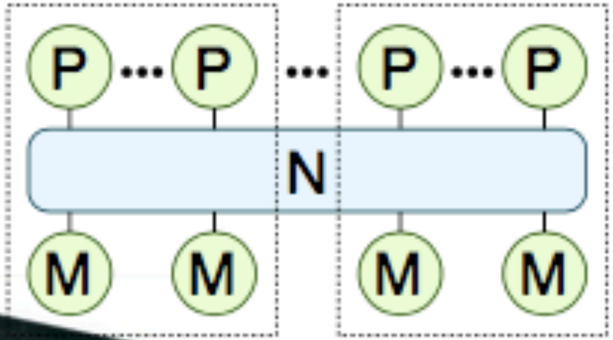
Cray XT3 Supercomputer
1024 single-core processors



AMD Quad-Core Opteron
Four cores on the same die

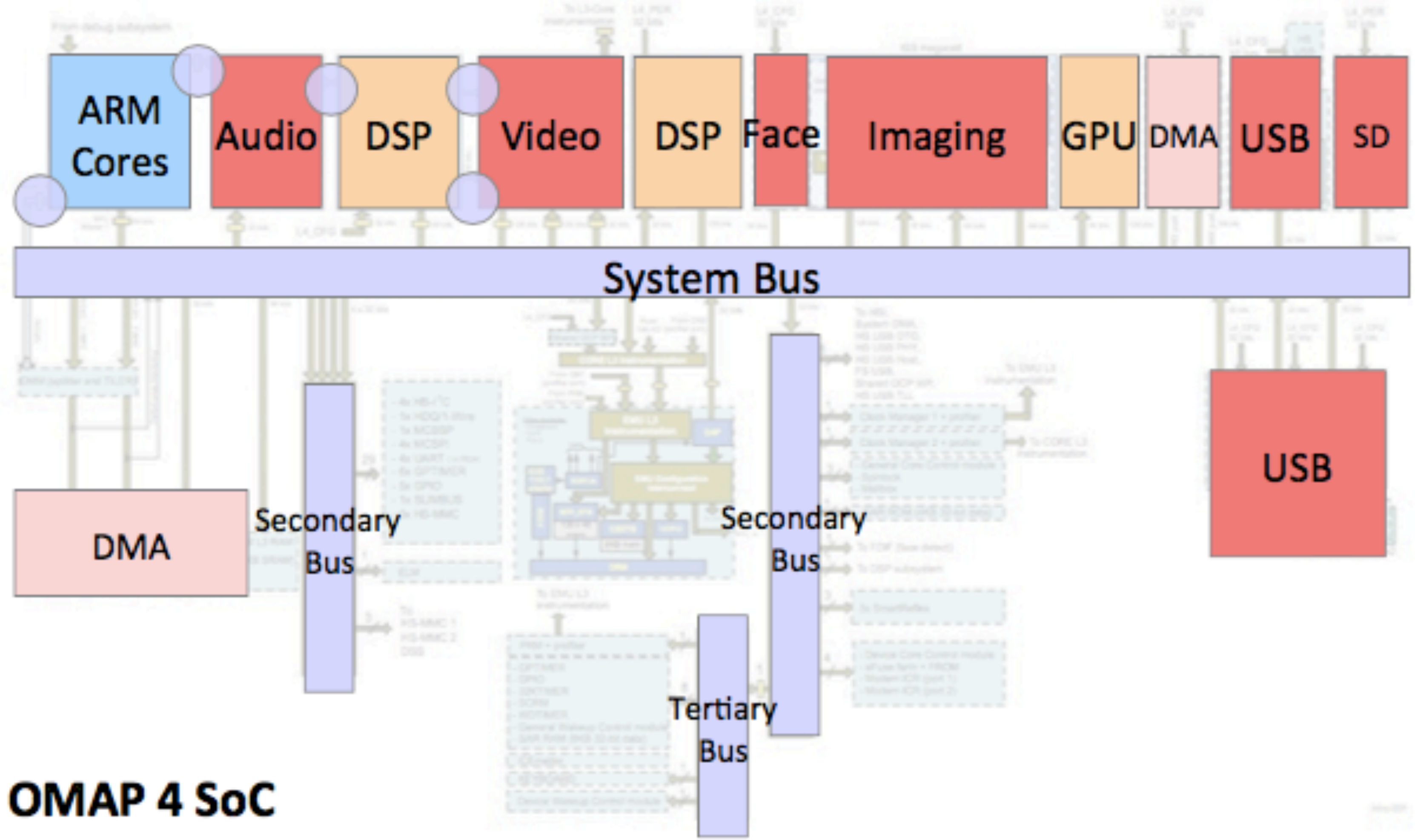






IBM Blue Gene Q Supercomputer
Thousands of
18-core processors

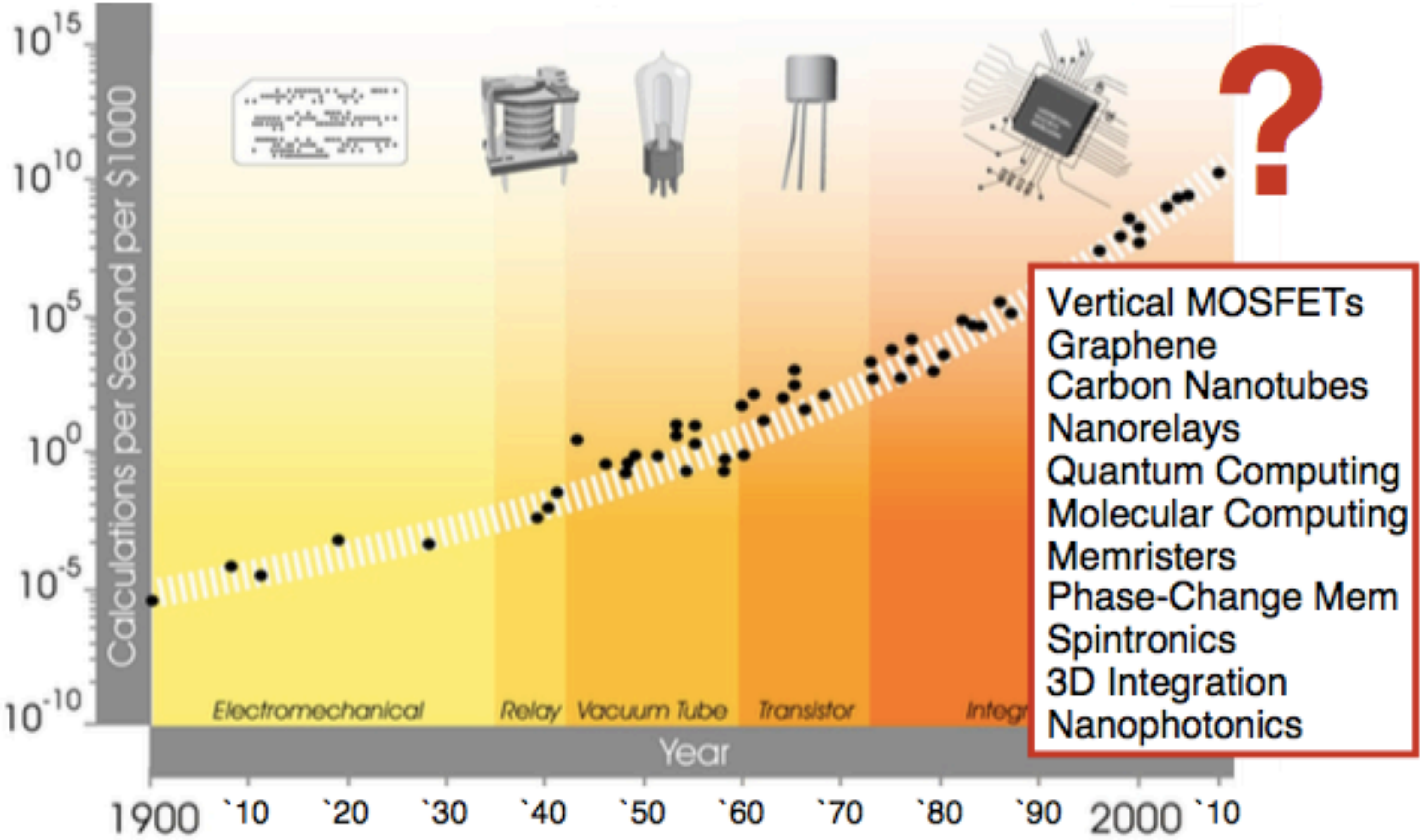
Trend 4: Heterogeneous System-on-Chip (SoC)



OMAP 4 SoC

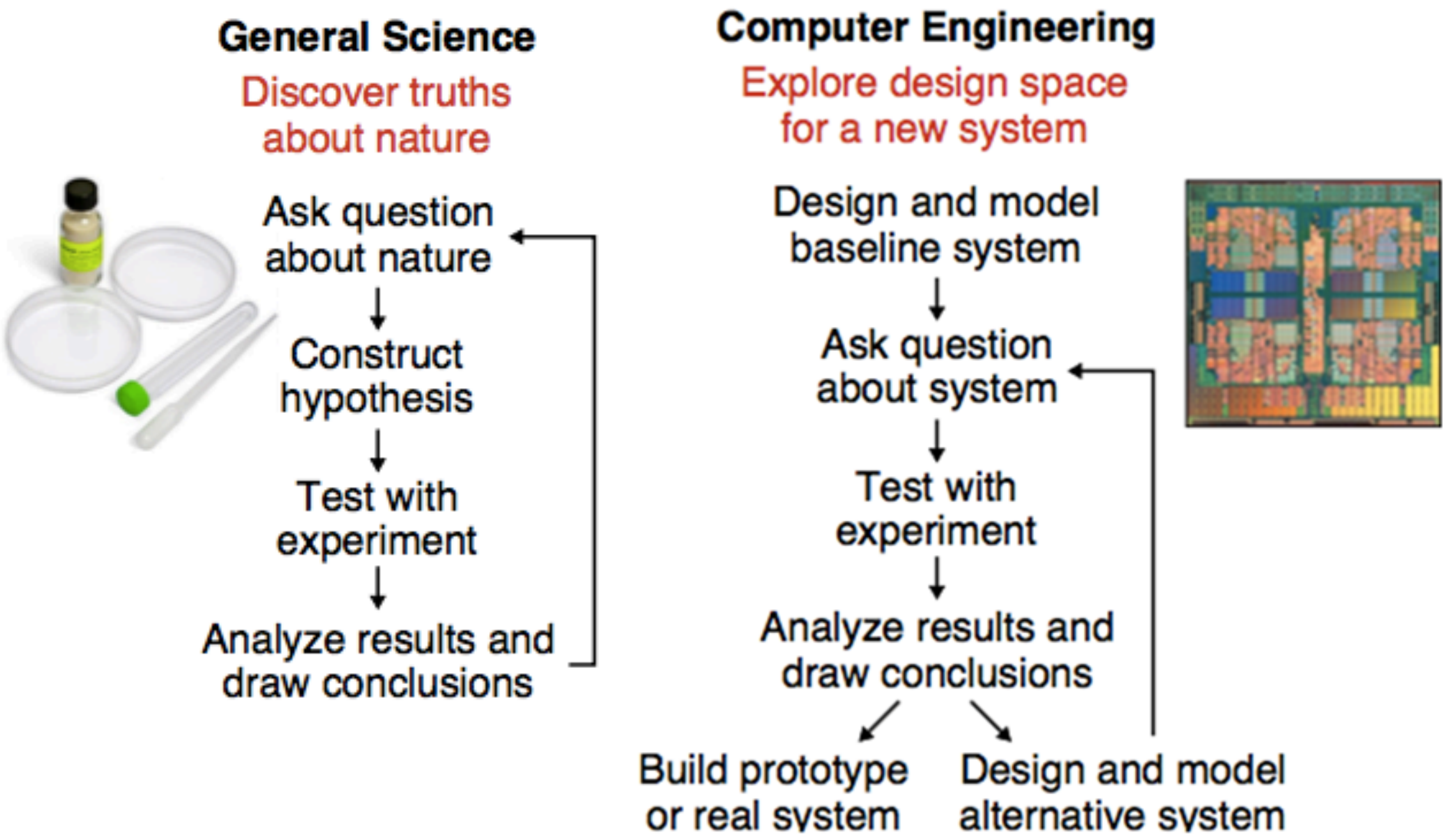
Adapted from D. Brooks Keynote at NSF XPS Workshop, May 2015.

Trend 5: Emerging Device Technologies



Vertical MOSFETs
Graphene
Carbon Nanotubes
Nanorelays
Quantum Computing
Molecular Computing
Memristers
Phase-Change Mem
Spintronics
3D Integration
Nanophotonics

System Research as a Scientific Approach



General Science

Discover truths about nature

Ask question about nature

Construct hypothesis

Test with experiment

Analyze results and draw conclusions

Computer Engineering

Explore design space for a new system

Design and model baseline system

Ask question about system

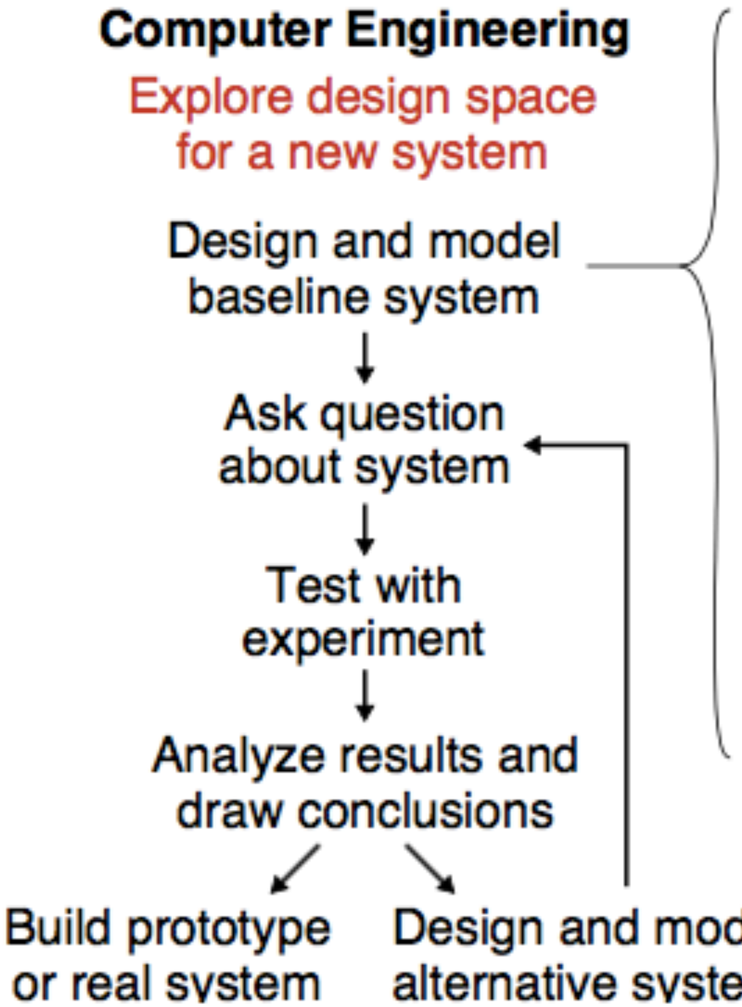
Test with experiment

Analyze results and draw conclusions

Build prototype or real system

Design and model alternative system

Modeling in Computer Architecture


Computer Engineering
Explore design space for a new system
Design and model baseline system
Ask question about system
Test with experiment
Analyze results and draw conclusions
Build prototype or real system
Design and model alternative system

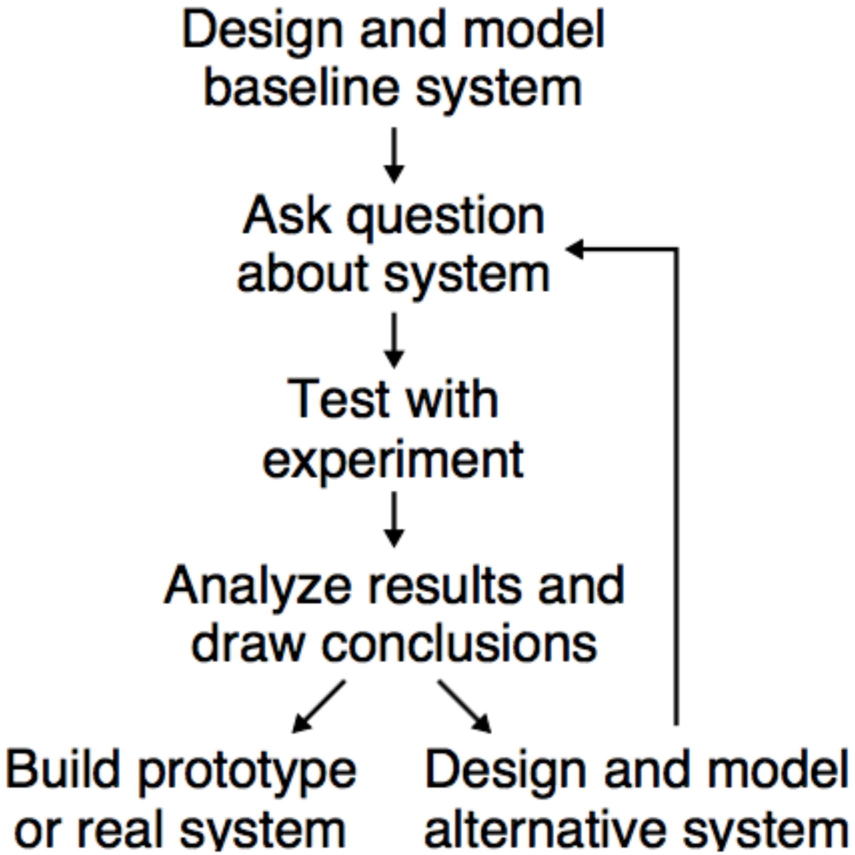
```
// rdy is OR of the AND of reqs and grants
assign in_rdy = | (reqs & grants);

reg [2:0] reqs;
always @(*) begin
  if ( in_val ) begin

    // eject packet if it is for this tile
    if ( dest == p_router_id )
      reqs = 3'b010;

    // otherwise, just pass it along ring
    else
      reqs = 3'b001;

  end else begin
    // if !val, don't request any ports
    reqs = 3'b000;
  end
end
```

Verilog • SystemVerilog • VHDL
C++ • SystemC
Bluespec • Chisel • Python

Dealing with Complexity

Computer Engineering

Explore design space for a new system



Design and model baseline system

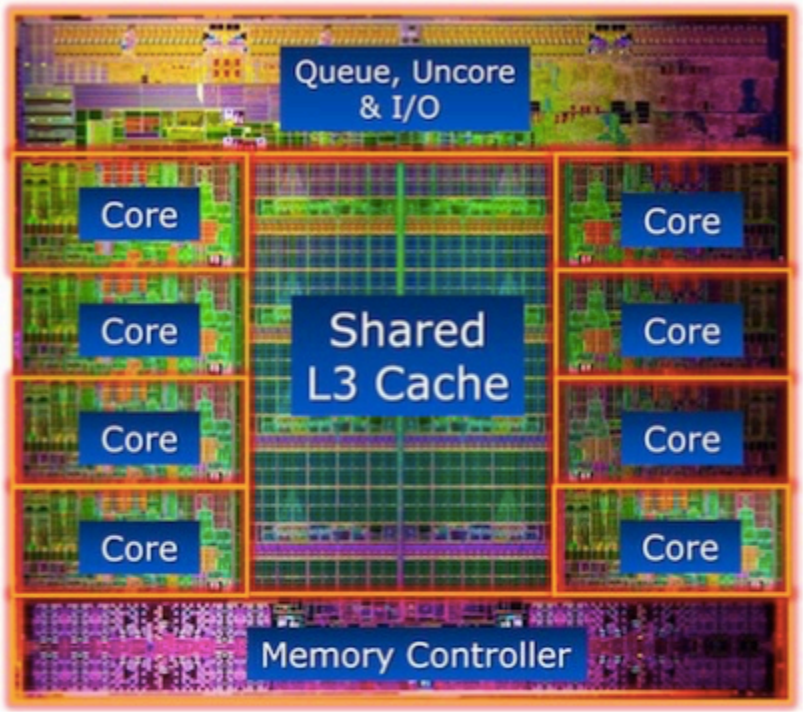
Ask question about system

Test with experiment

Analyze results and draw conclusions

Build prototype or real system

Design and model alternative system



Fighter Airplane: ~100,000 parts

Intel Sandy Bridge E:
2.27 Billion transistors

Dealing with Complexity

- Design Principles
  - modularity
  - hierarchy
  - encapsulation
  - regularity
  - extensibility
- Design Patterns
  - processors, memories, networks
  - control/datapath split
  - single-cycle, FSM, pipelined control
  - raw port, message, method interfaces

Lessons from the general-purpose processor architecture and design can be extended to SoC and serve as the baseline for comparison.

Next Lecture Preview

- Instruction Set Architecture (ISA)
  - contract between software and hardware
- Example
  - MIPS32
  - how is data represented?
  - where can data be stored?
  - how can data be accessed?
  - what operations can be done on data?
  - how are instructions encoded?
- Reading Assignment
  - PARC ISA
  - http://www.csl.cornell.edu/courses/ece4750/2015f/handouts/ece4750-parc-isa.txt

Questions?

Comments?

Discussion?

Acknowledgement

Cornell University, ECE 4750