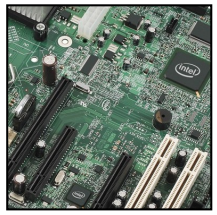
Introduction to Electrical and Computer Engineering

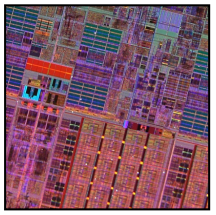
Christopher Batten

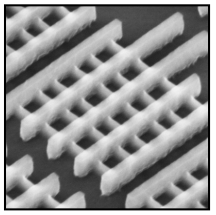
Computer Systems Laboratory
School of Electrical and Computer Engineering
Cornell University

ENGRG 1060 Explorations in Engineering Seminar
Summer 2013


Power Systems


Computer Engineering

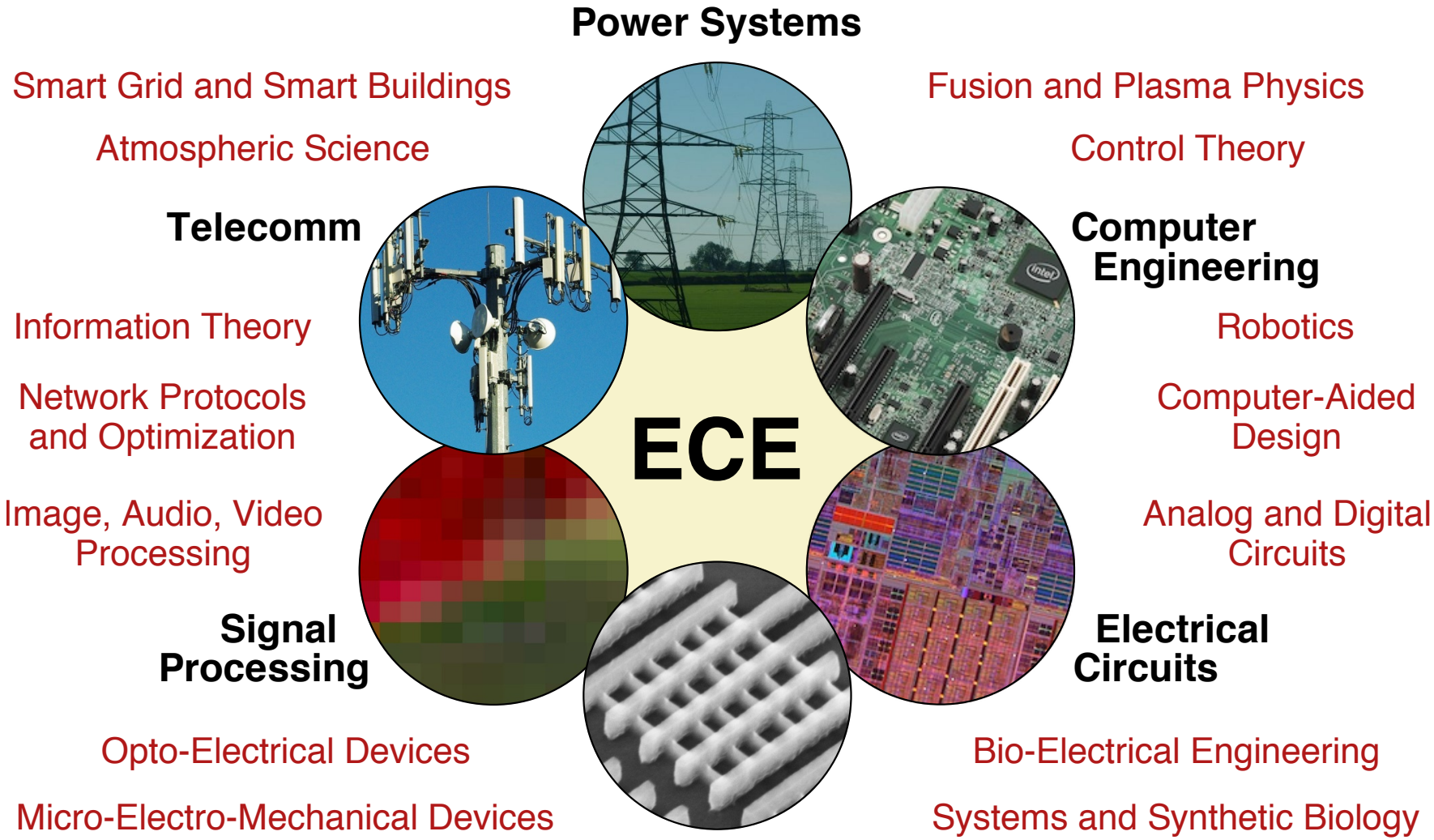

Electrical Circuits


Electrical Devices


Signal Processing


Telecomm

ECE is the Study and Application of Electricity, Micro-Electronics, and Electro-Magnetism



ECE is everywhere!

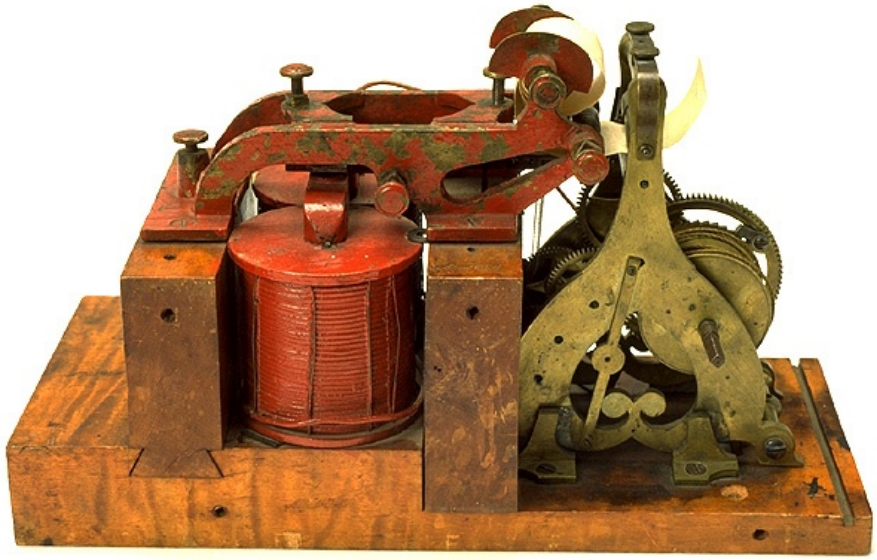
What can one do with a background in ECE?

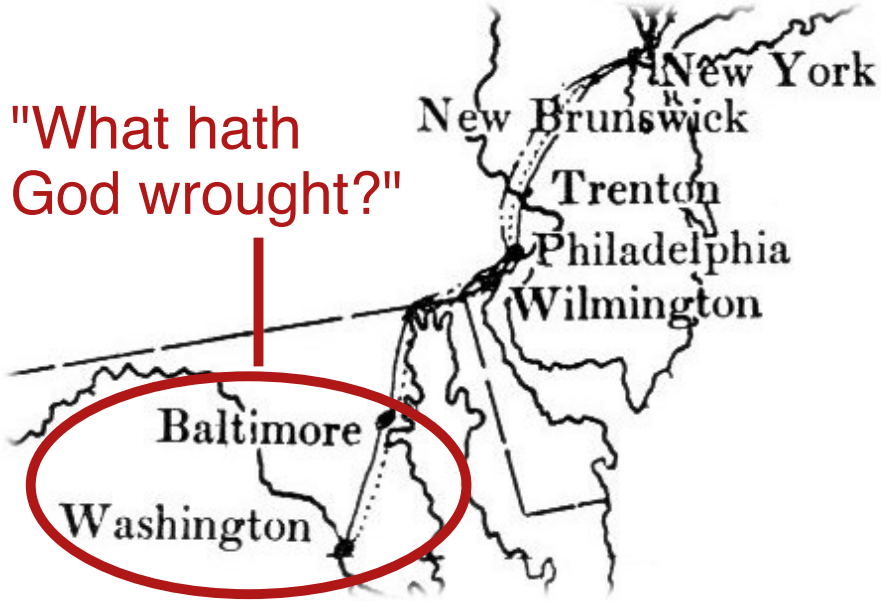
- **ECE Industry**: Intel, AMD, Analog Devices, NVIDIA, HP, Apple
- **General Engineering Industry**: GE, Lockheed Martin, Raytheon
- **Software Industry**: Microsoft, Amazon, Mathworks
- **Join a Startup**: Achronix, Hillcrest Labs
- **Research Lab**: Sandia National Labs, Draper Labs, NASA
- **Consulting**: McKinsey, Accenture, Deloitte, Booz Allen Hamilton
- **Finance**: Deutsche Bank, Capital One, UBS, Bloomberg
- **Graduate School**: Law School, Business School, Med School
- **Found a university!**

Cornell was founded because of ECE!

Samuel Morse invented the telegraph (a digital communication device), but needed help building the network

Ezra Cornell built the first telegraph line (the beginning of telecommunications), and invested in the Western Union Telegraph Co

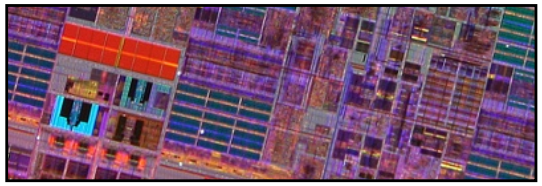
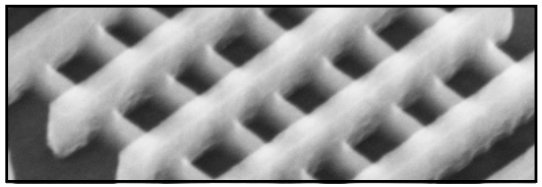
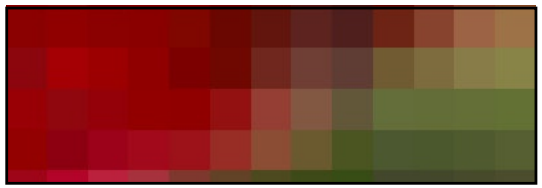




Ezra Cornell's investments created the fortune that eventually enabled the founding of Cornell University

"Optional Homework"

- Visit the statue of Ezra Cornell on the Arts Quad
- Does something on the back of the statue relate to ECE?
- Take a picture with your cellphone and send it to your friend!
  - Power systems
  - Computer engineering
  - Electrical circuits
  - Electrical devices
  - Signal processing
  - Telecommunications


Talk Outline

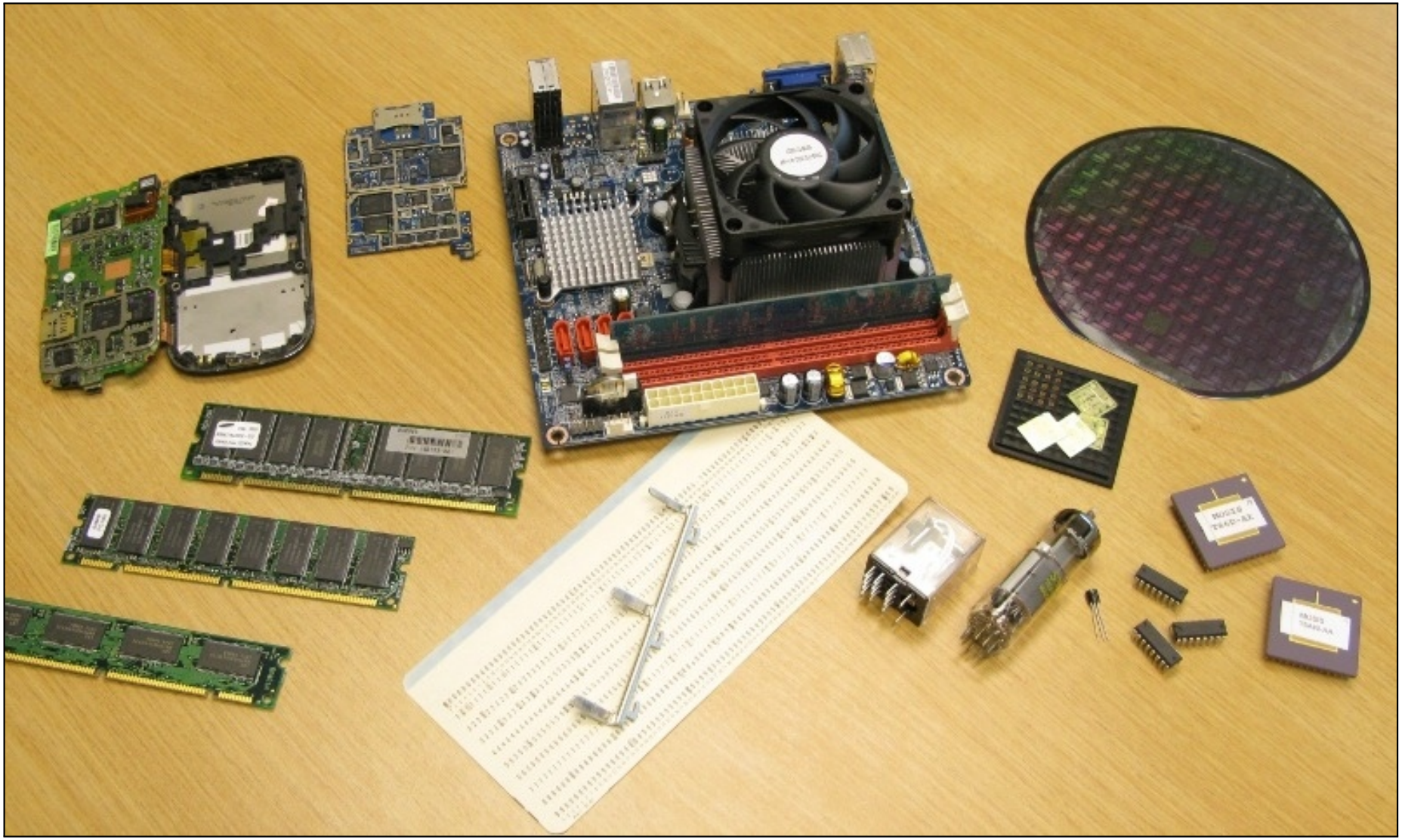
ECE Overview

What is Computer Engineering?

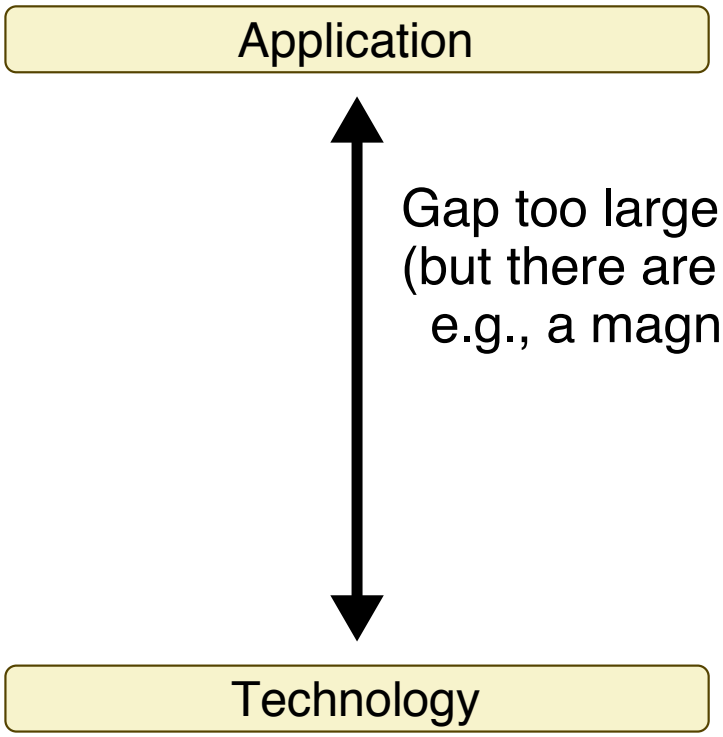
Trends in Computer Engineering

Computer Engineering Design

Computer Engineering Artifacts



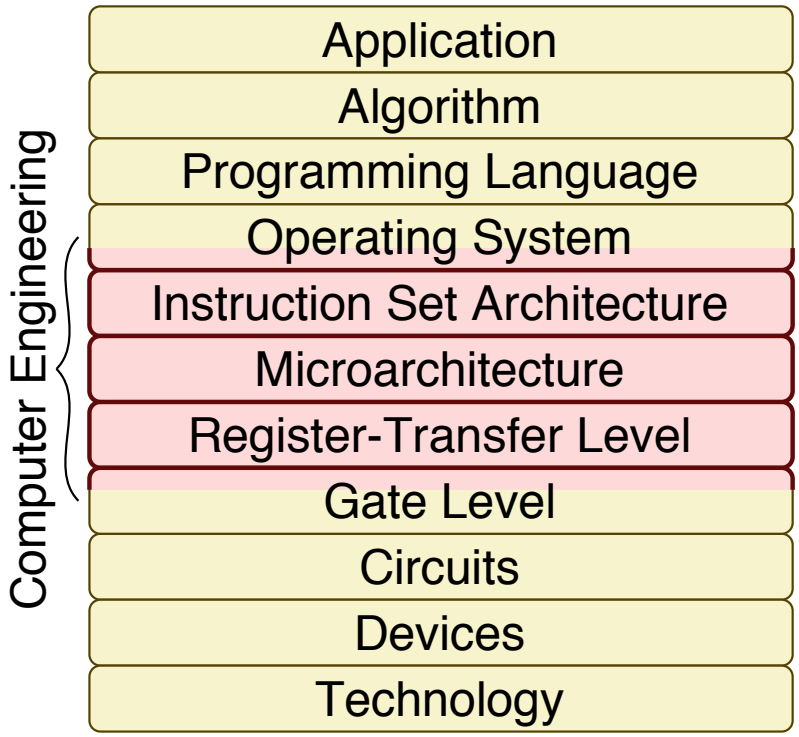
The Computer Systems Stack



Gap too large to bridge in one step
(but there are exceptions,
e.g., a magnetic compass)

In its broadest definition, computer engineering is the development of the abstraction/implementation layers that allow us to execute information processing applications efficiently using available manufacturing technologies

The Computer Systems Stack



Sort an array of numbers

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

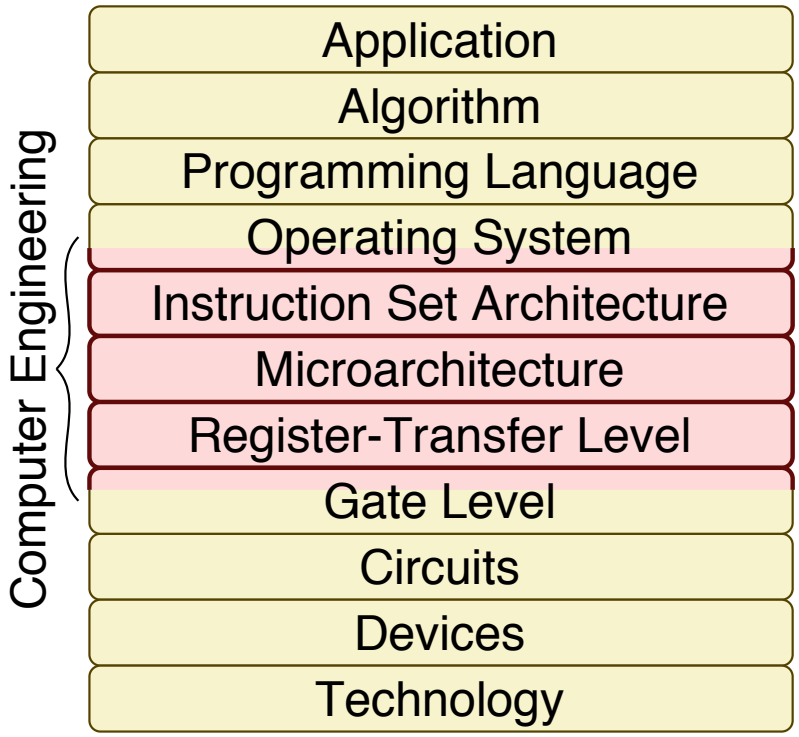
Insertion sort algorithm

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

C implementation of insertion sort

```
void isort( 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;
  }
}
```

The Computer Systems Stack

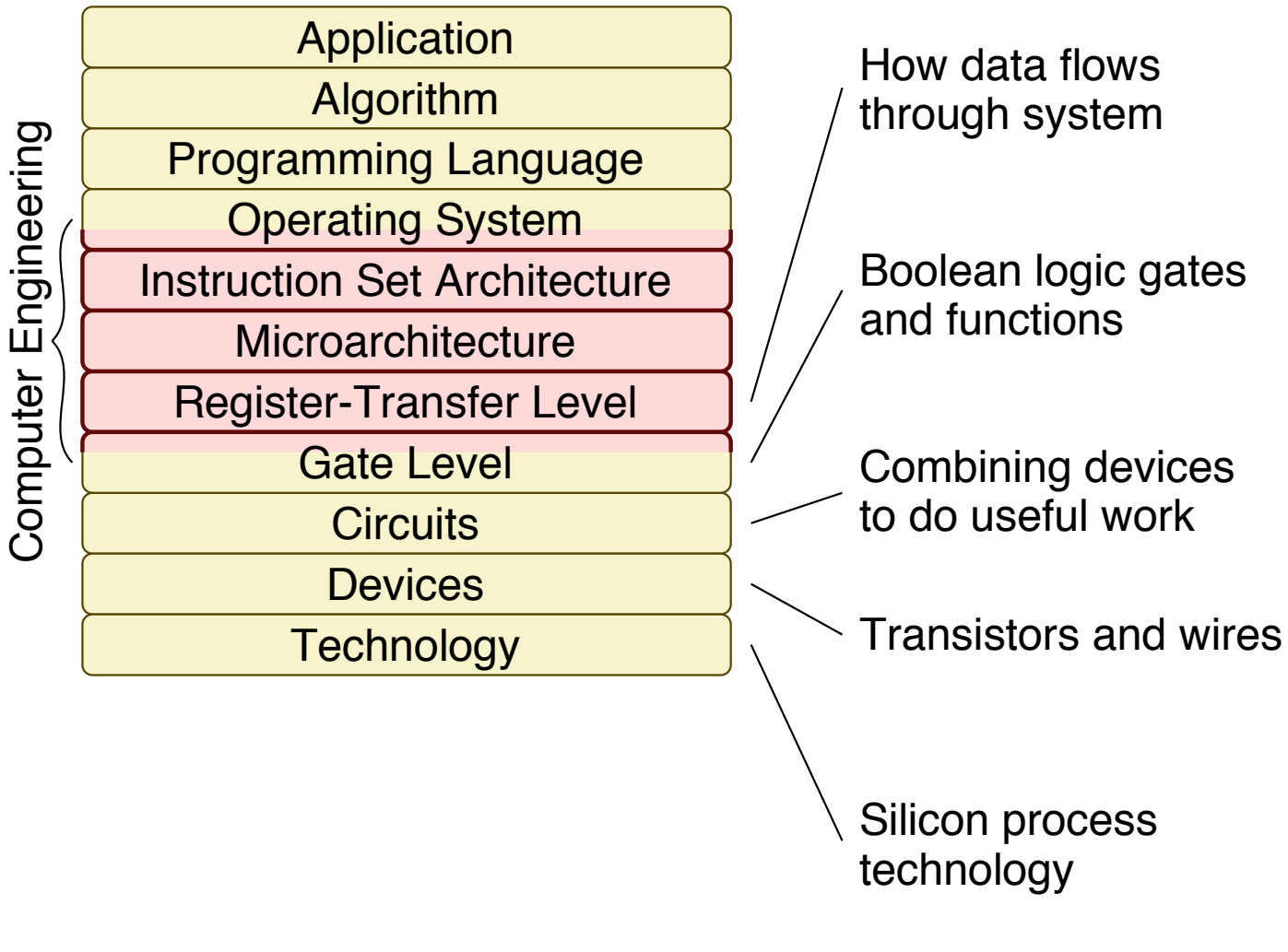


Mac OS X, Windows, Linux
Handles low-level hardware management

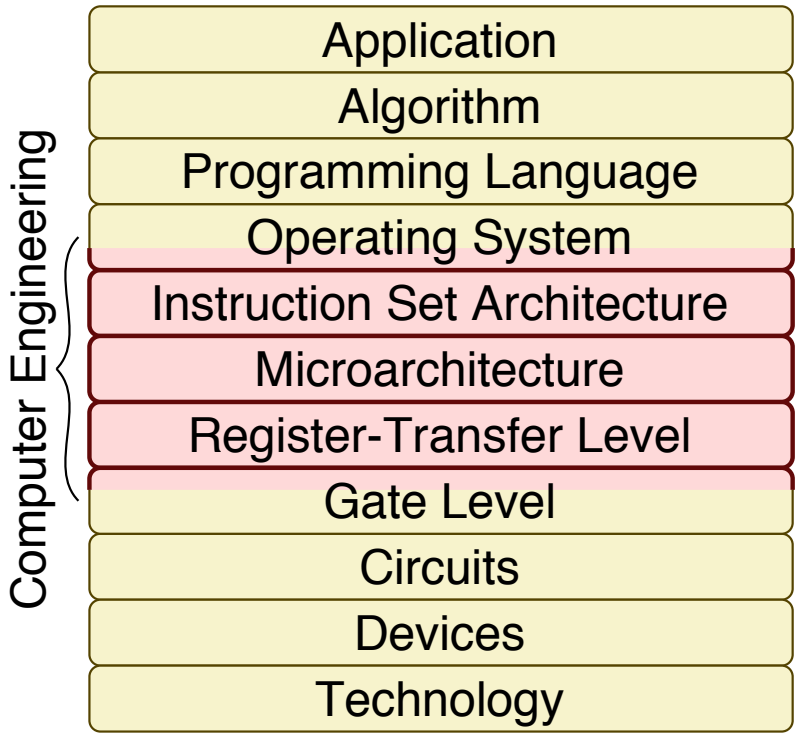
MIPS32 Instruction Set
Instructions that machine executes

```
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
```

The Computer Systems Stack



Computer Systems: CS vs. EE vs. CE

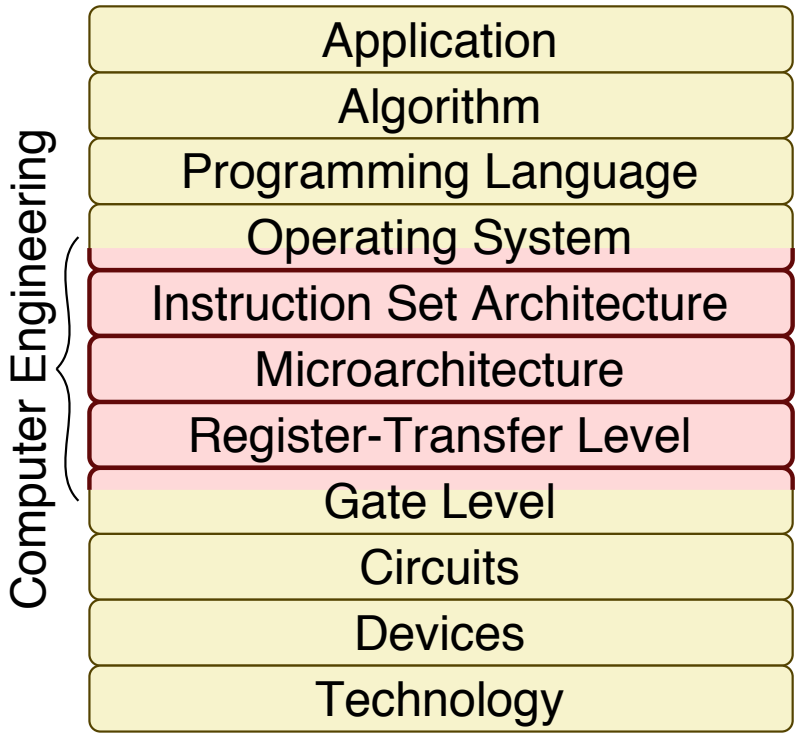


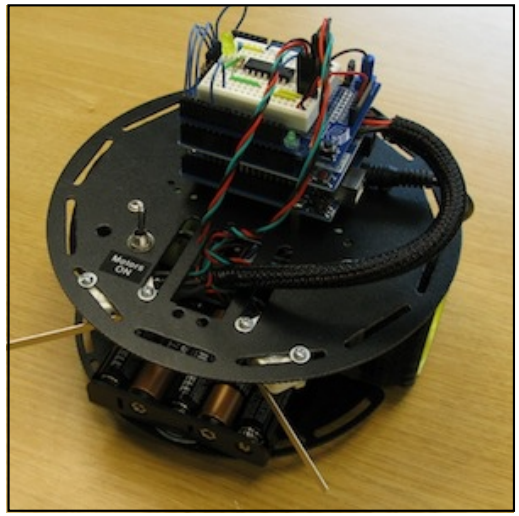
Traditional Computer Science

Computer Engineering is at the interface between hardware and software and considers the entire system

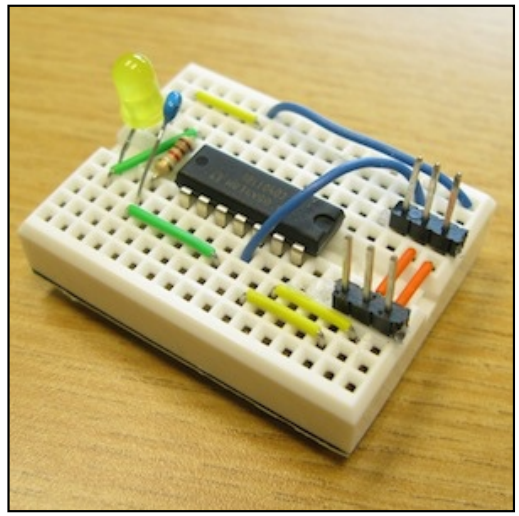
Traditional Electrical Engineering

ENGRG 1060 Computer Systems Labs



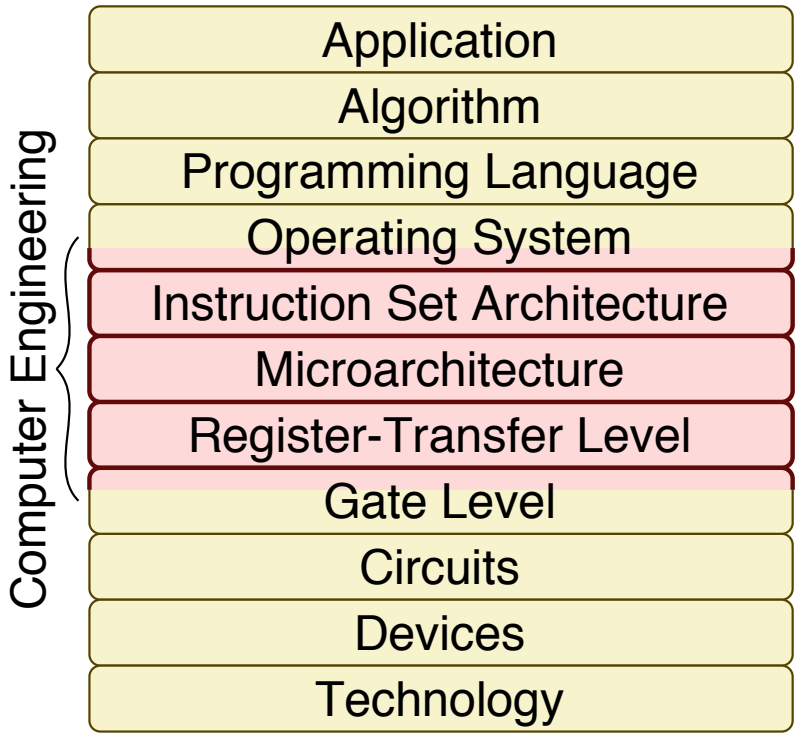


Lab 4
Software pushing towards hardware (CS,CE)



Lab 2
Hardware pushing towards software (EE,CE)

Application Requirements vs. Technology Constraints



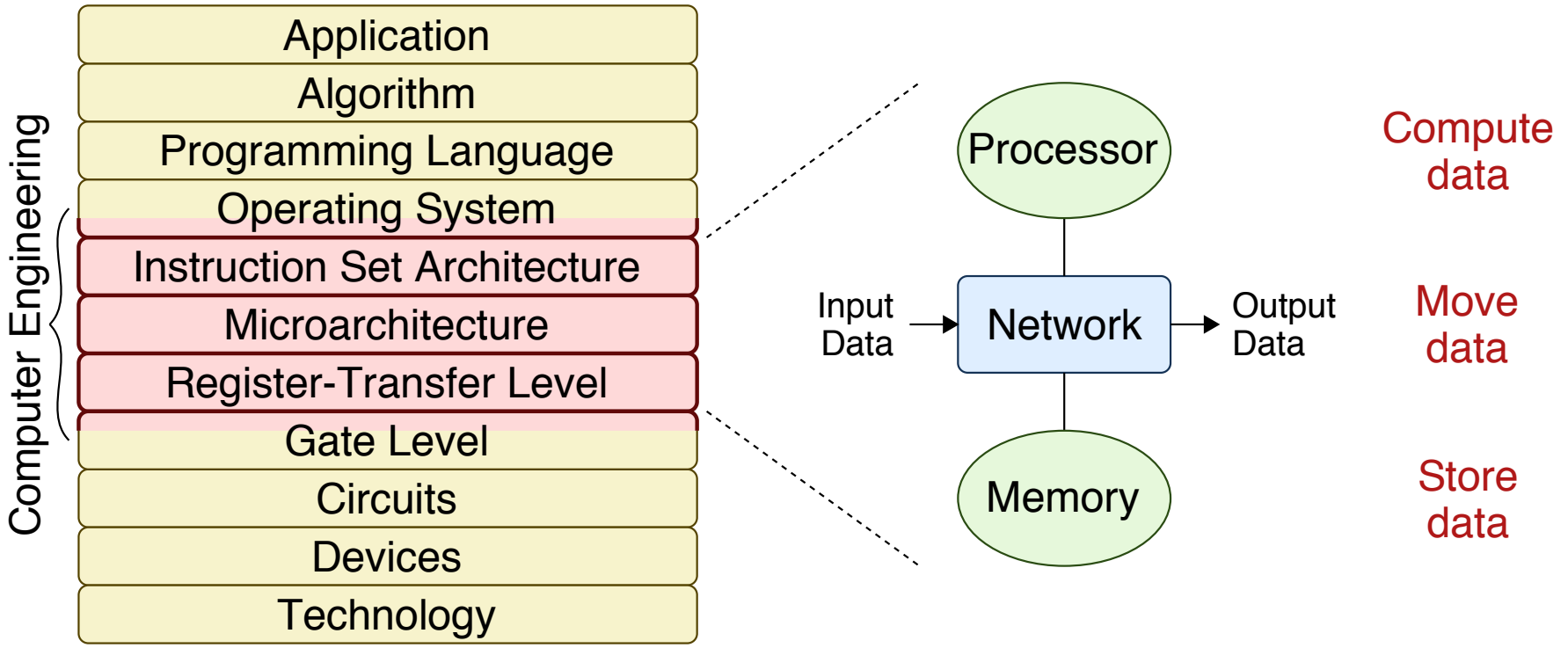
Application Requirements
- Suggest how to improve architecture
- Provide revenue to fund development

Computer engineers provide feedback to guide application and technology research directions

Technology Constraints
- Restrict what can be done efficiently
- New technologies make new arch possible

In its broadest definition, computer engineering is the development of the abstraction/implementation layers that allow us to execute information processing applications efficiently using available manufacturing technologies

Processors, Memories, and Networks

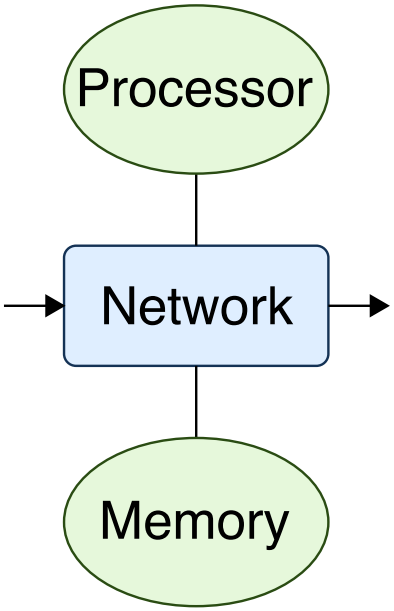


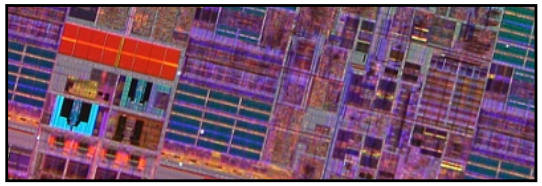
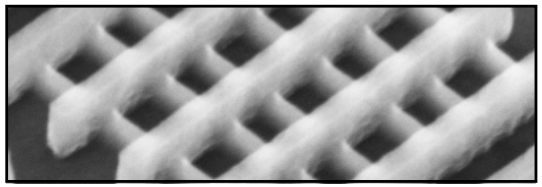
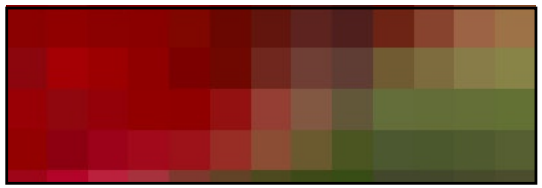
Computer engineering basic building blocks

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

Activity #1: Sorting with a Sequential Processor

- **Application:** Sort 32 numbers
- **Simulated Sequential Computing System**
  - Processor: You!
  - Memory: Worksheet, read input data, write output data
  - Network: Passing/collecting the worksheets
- **Activity Steps**
  1. Discuss strategy with neighbors
  2. When instructor starts timer, flip over worksheet
  3. Sort 32 numbers as fast as possible
  4. Lookup when completed and write time on worksheet
  5. Raise hand
  6. When everyone is finished, then analyze data




Talk Outline

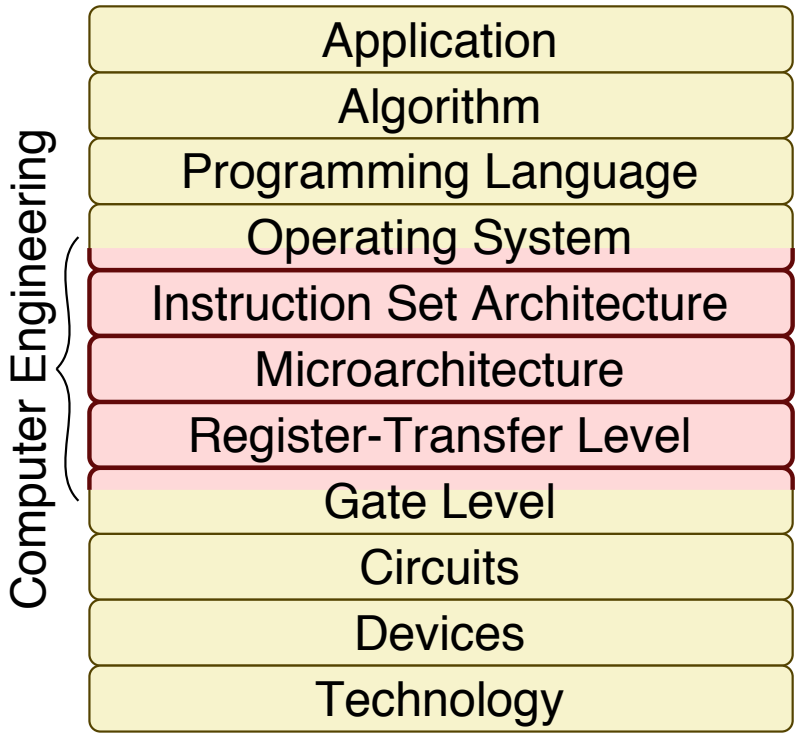
ECE Overview

What is Computer Engineering?

Trends in Computer Engineering

Computer Engineering Design

Application Requirements vs. Technology Constraints



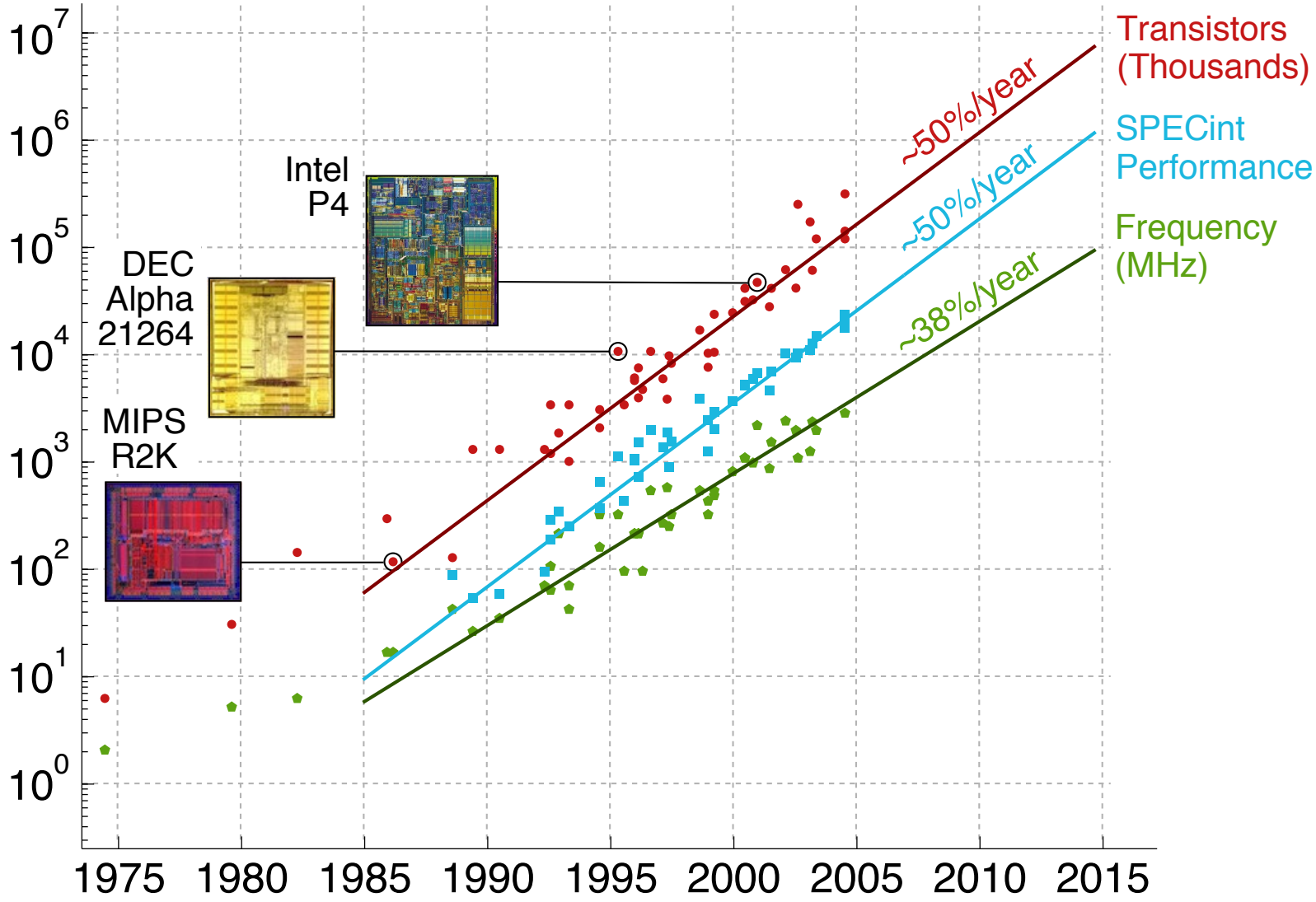
Traditional Application Requirements

- As much processor compute as possible
- As much memory capacity as possible
- As much network bandwidth as possible

Traditional Technology Constraints

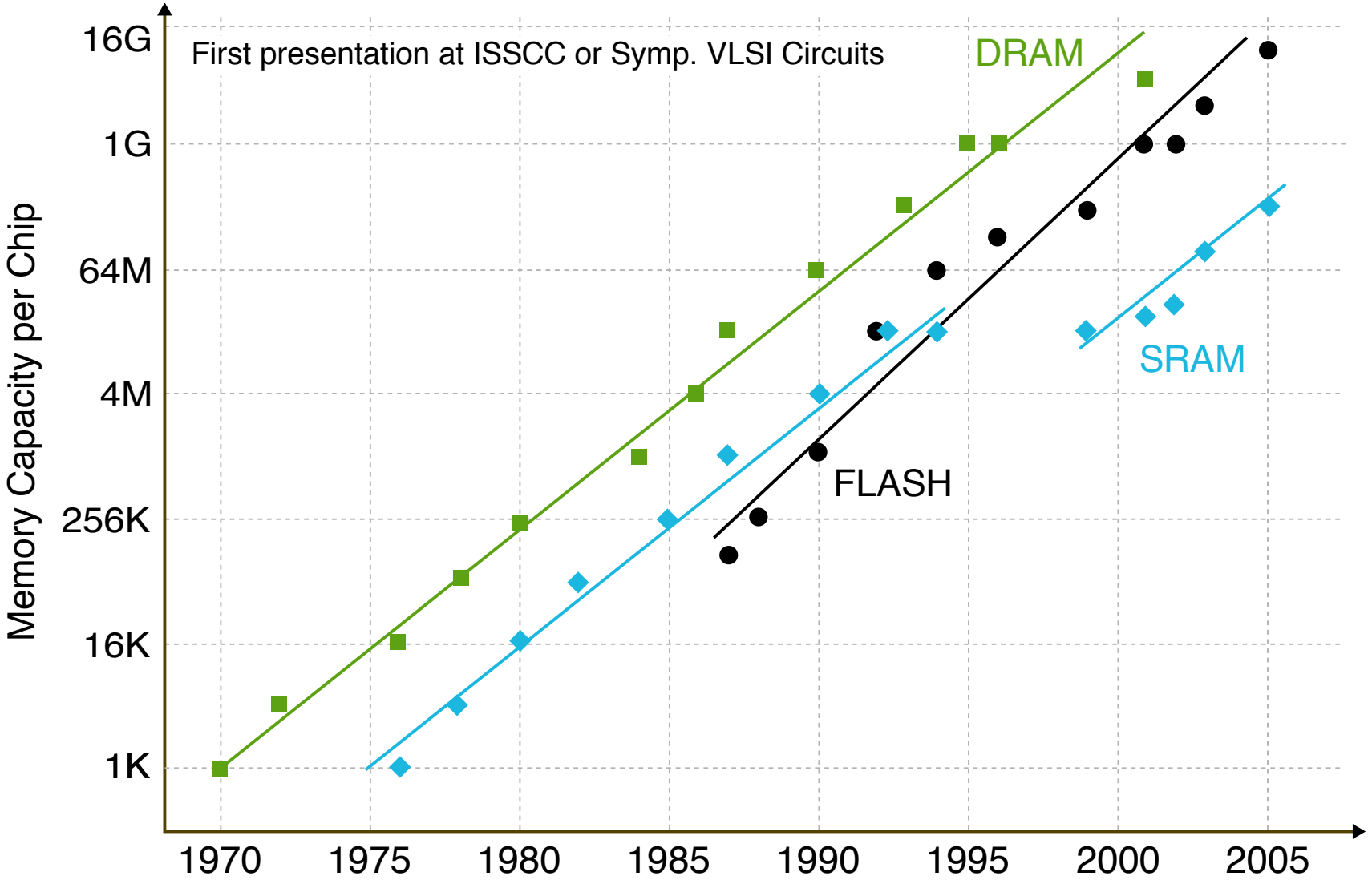
- Exponential scaling of resources

Exponential Scaling for Processor Computation



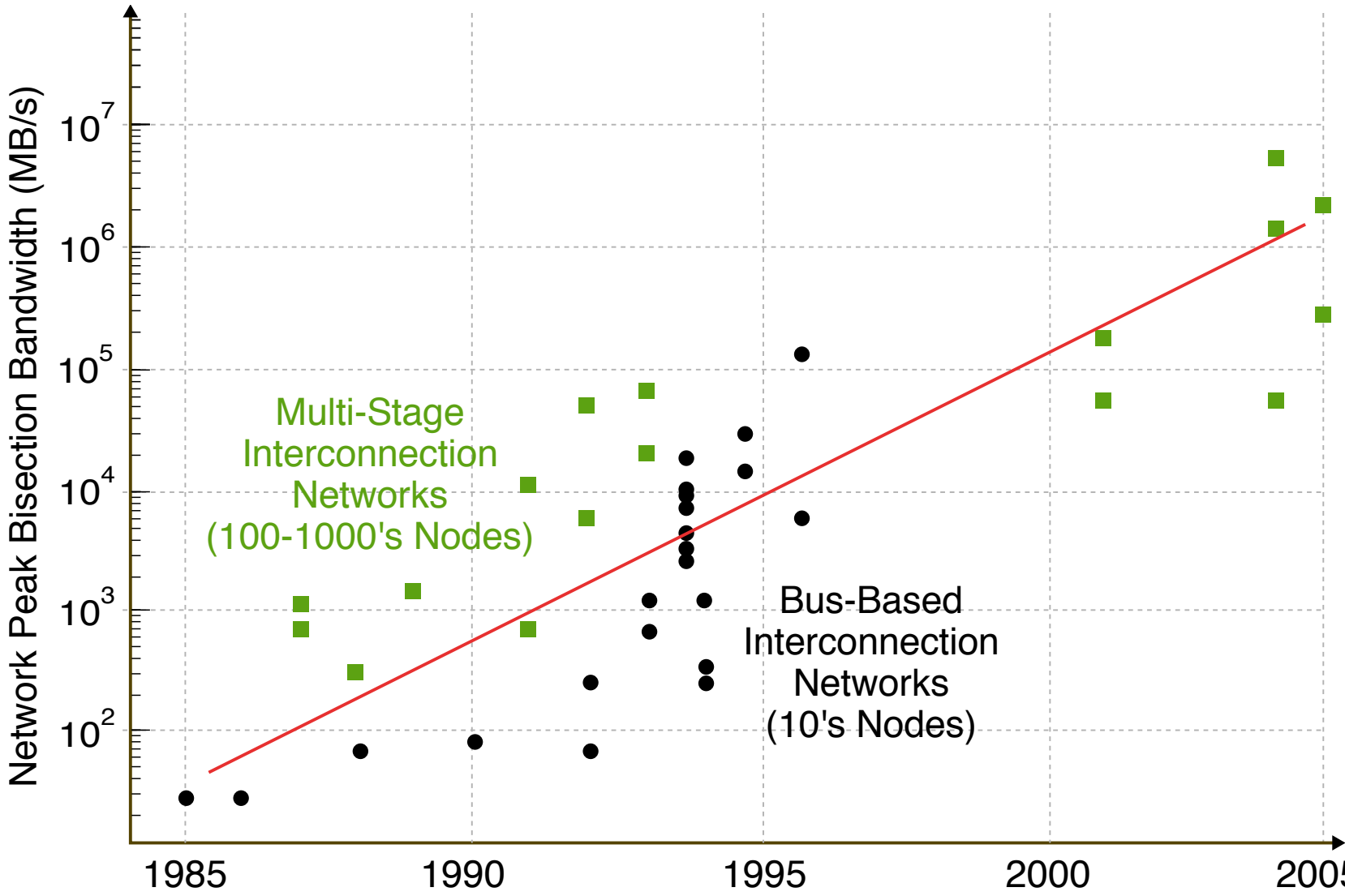
Data collected by M. Horowitz, F. Labonte, O. Shacham, K. Olukotun, L. Hammond, C. Batten

Exponential Scaling for Memory Capacity



Adapted from K. Itoh et al. "Ultra-Low Voltage Nano-Scale Memories." Spring 2007.

Exponential Scaling for Network Bandwidth



Data from Hennessy & Patterson, Morgan Kaufmann, 2nd & 5th eds., 1996 & 2011; D.E. Culler et al., Morgan Kaufmann, 1999.

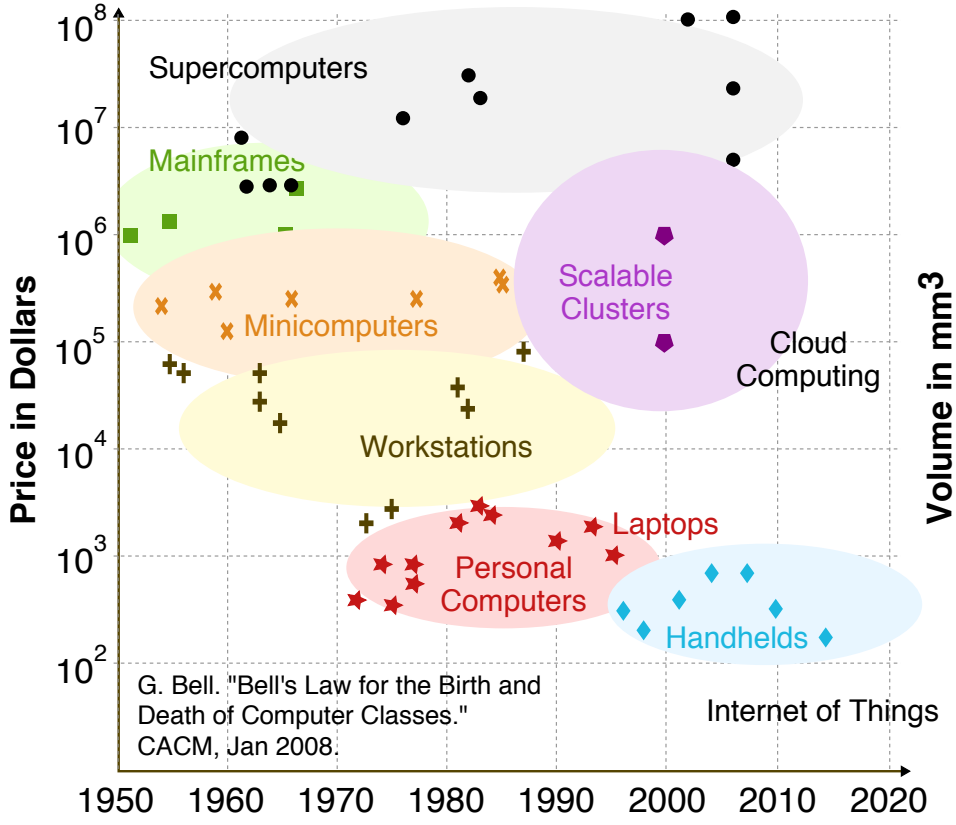
Key trends in application requirements and technology constraints over the past decade have resulted in a radical rethinking of the processors, memories, and networks used in modern computing systems

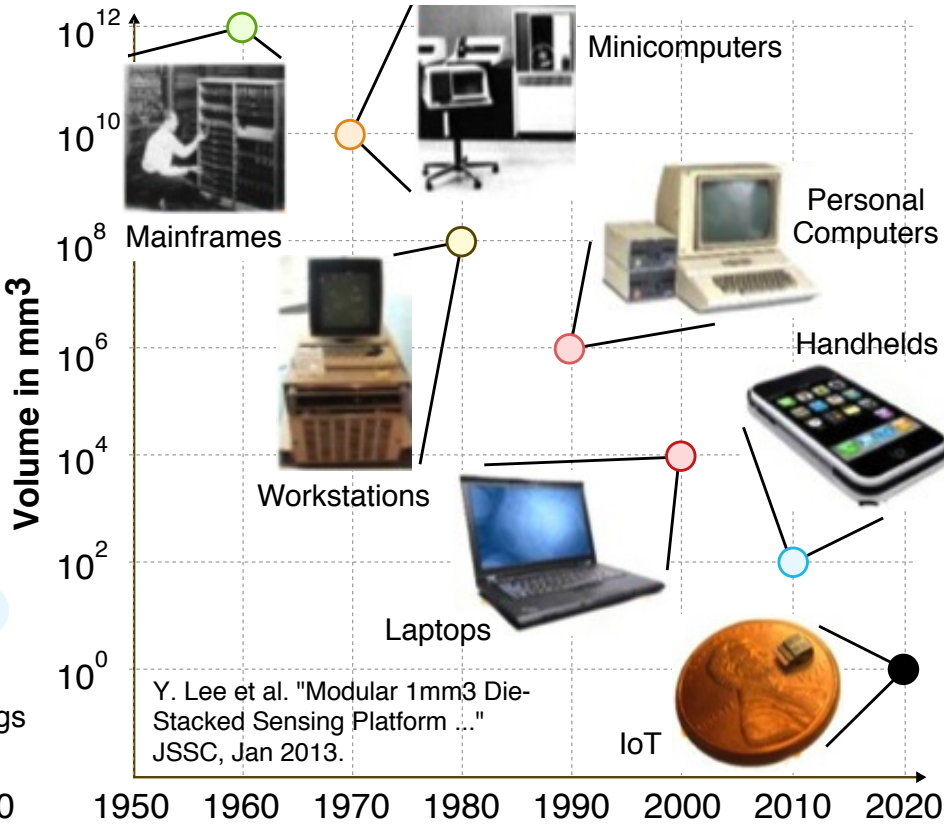
Four Key Trends in Computer Engineering

1. Growing diversity in application requirements motivate growing diversity in computing systems
2. Energy and power constrain systems across the entire computing spectrum
3. Transition to multiple cores integrated onto a single chip
4. Technology scaling challenges motivate new emerging processor, memory, and network 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





Trend 1: Growing Diversity in Apps & Systems

Trend 2: Energy and Power Constraints

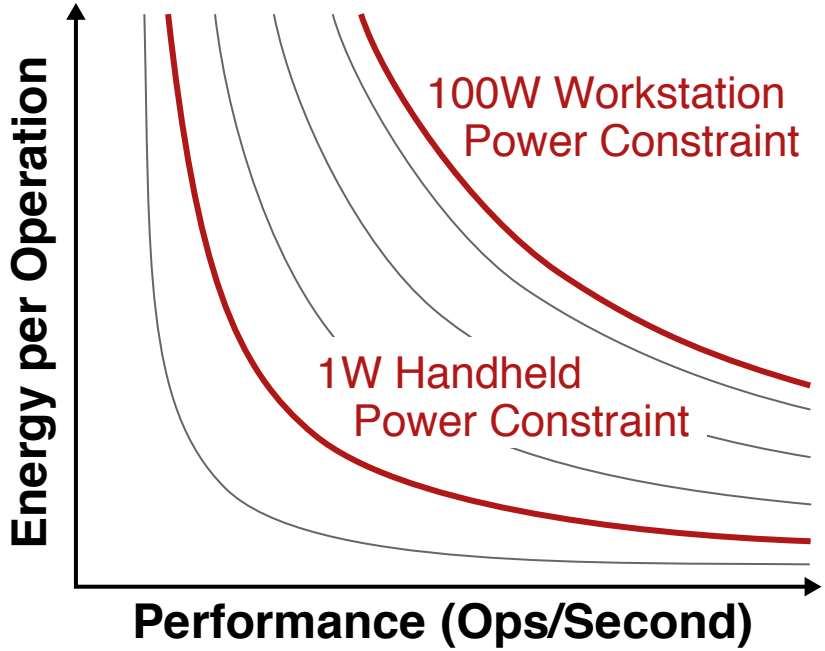
$$\text{Power} = \frac{\text{Energy}}{\text{Second}} = \frac{\text{Energy}}{\text{Op}} \times \frac{\text{Ops}}{\text{Second}}$$

Power

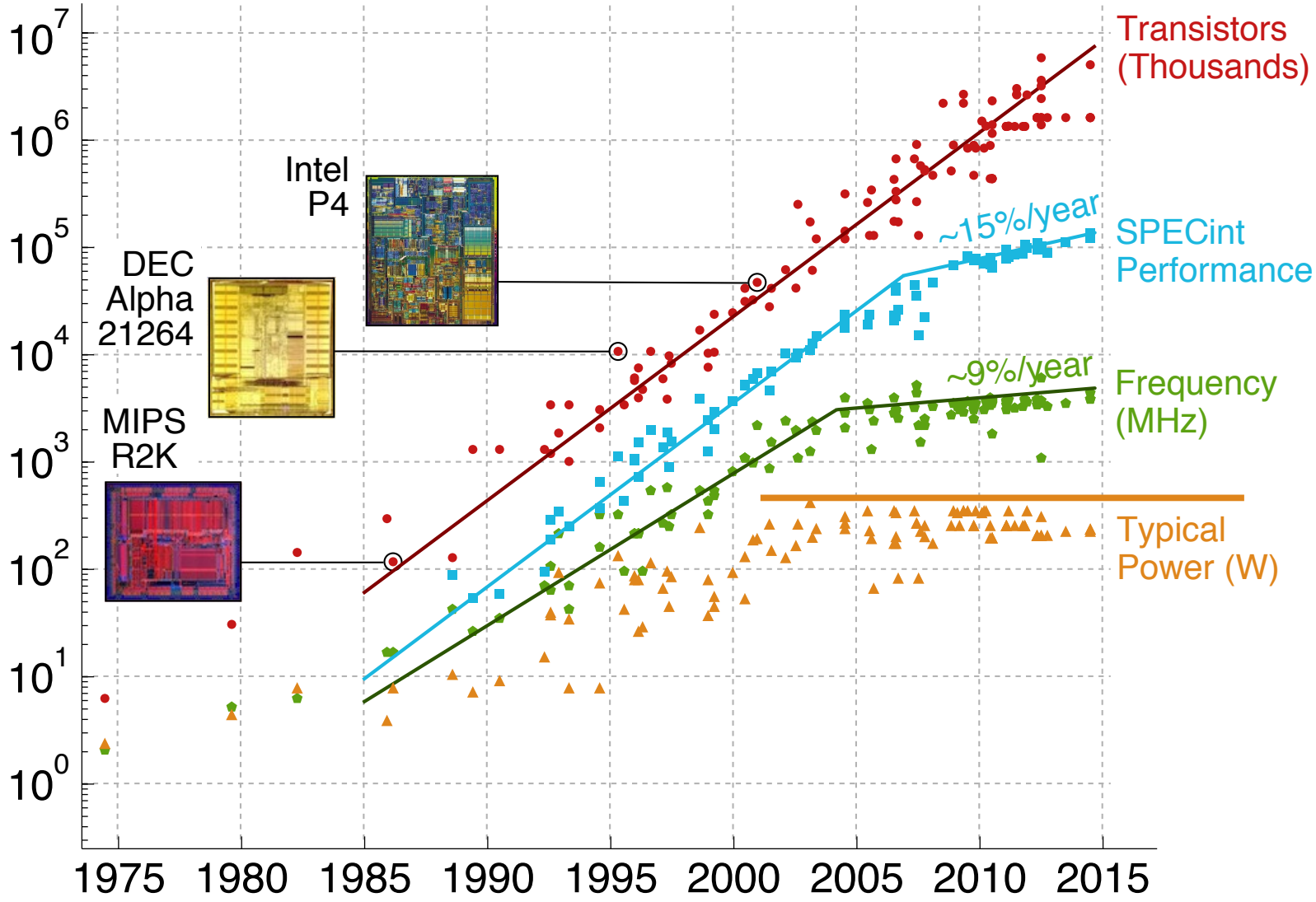
- Chip Packaging
- Chip Cooling
- System Noise
- Case Temperature
- Data-Center Air Conditioning

Energy

- Battery Life
- Electricity Bill
- Mobile Device Weight

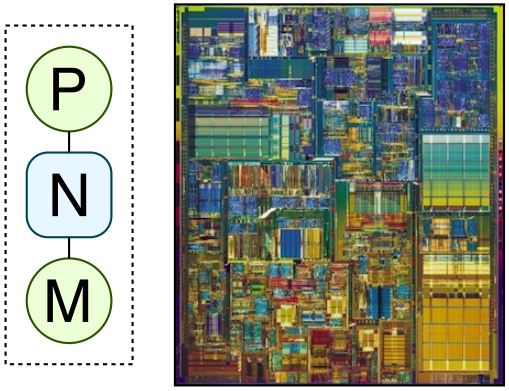


Trend 2: Power Constrains Single-Processor Scaling

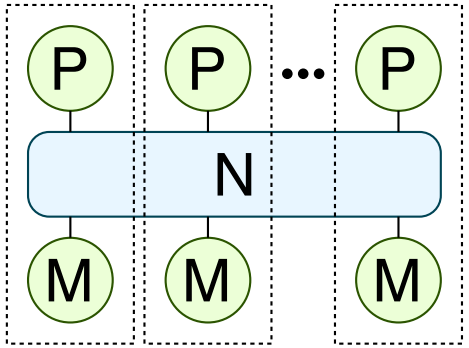


Trend 3: Transition to Multicore Processors

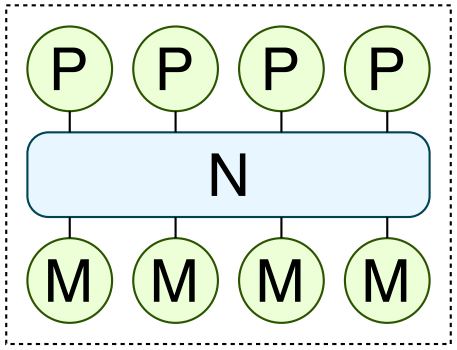
Intel Pentium 4
Single monolithic processor

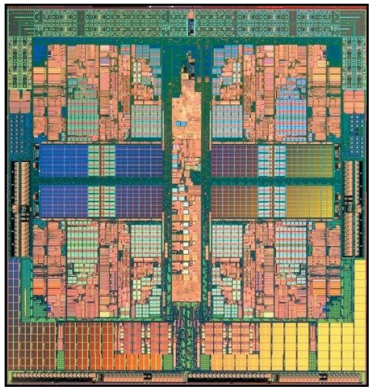


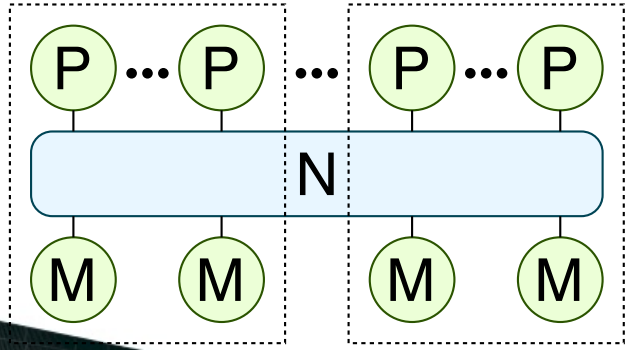
Cray XT3 Supercomputer
1024 single-core processors



AMD Quad-Core Opteron
Four cores on the same die

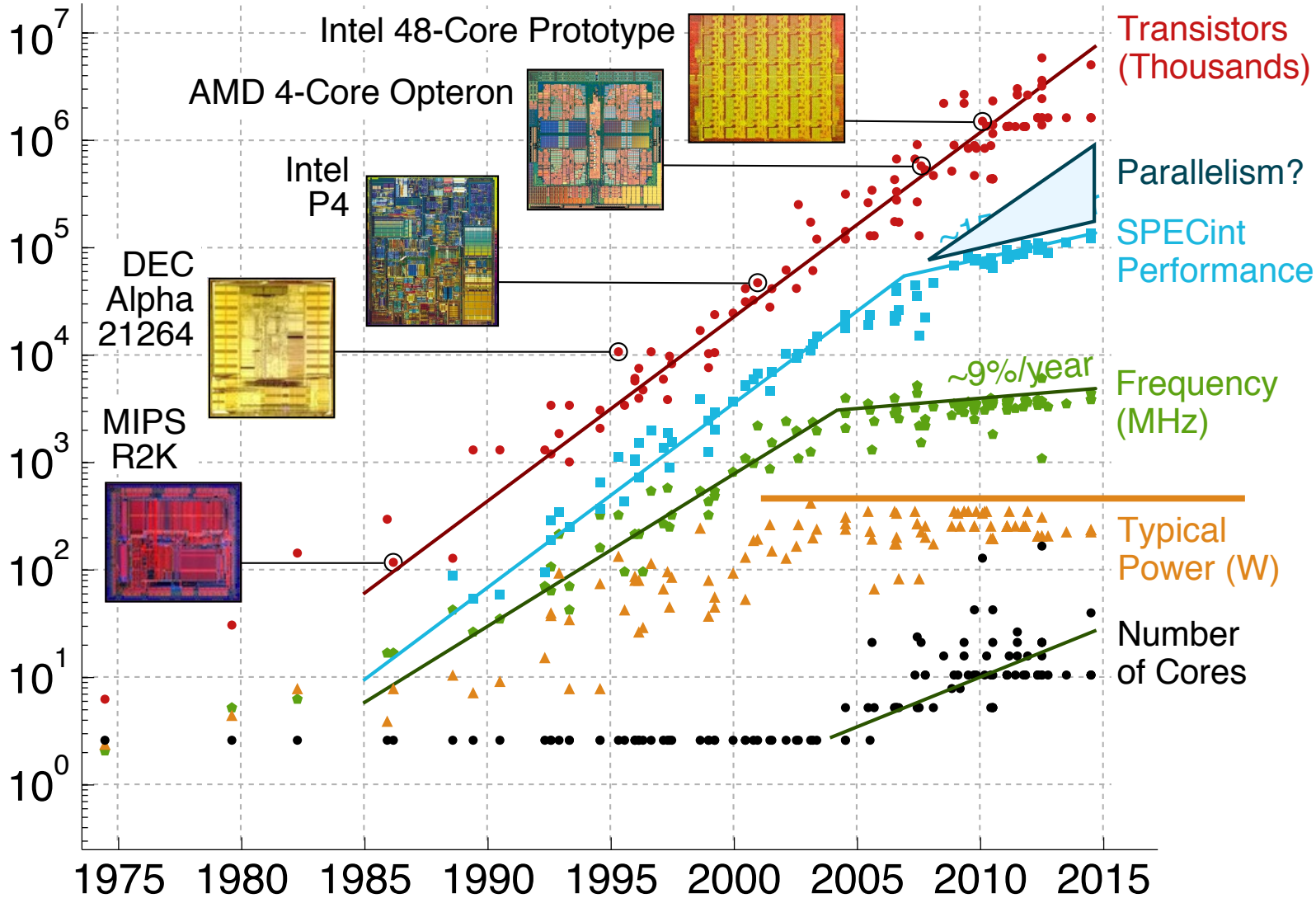




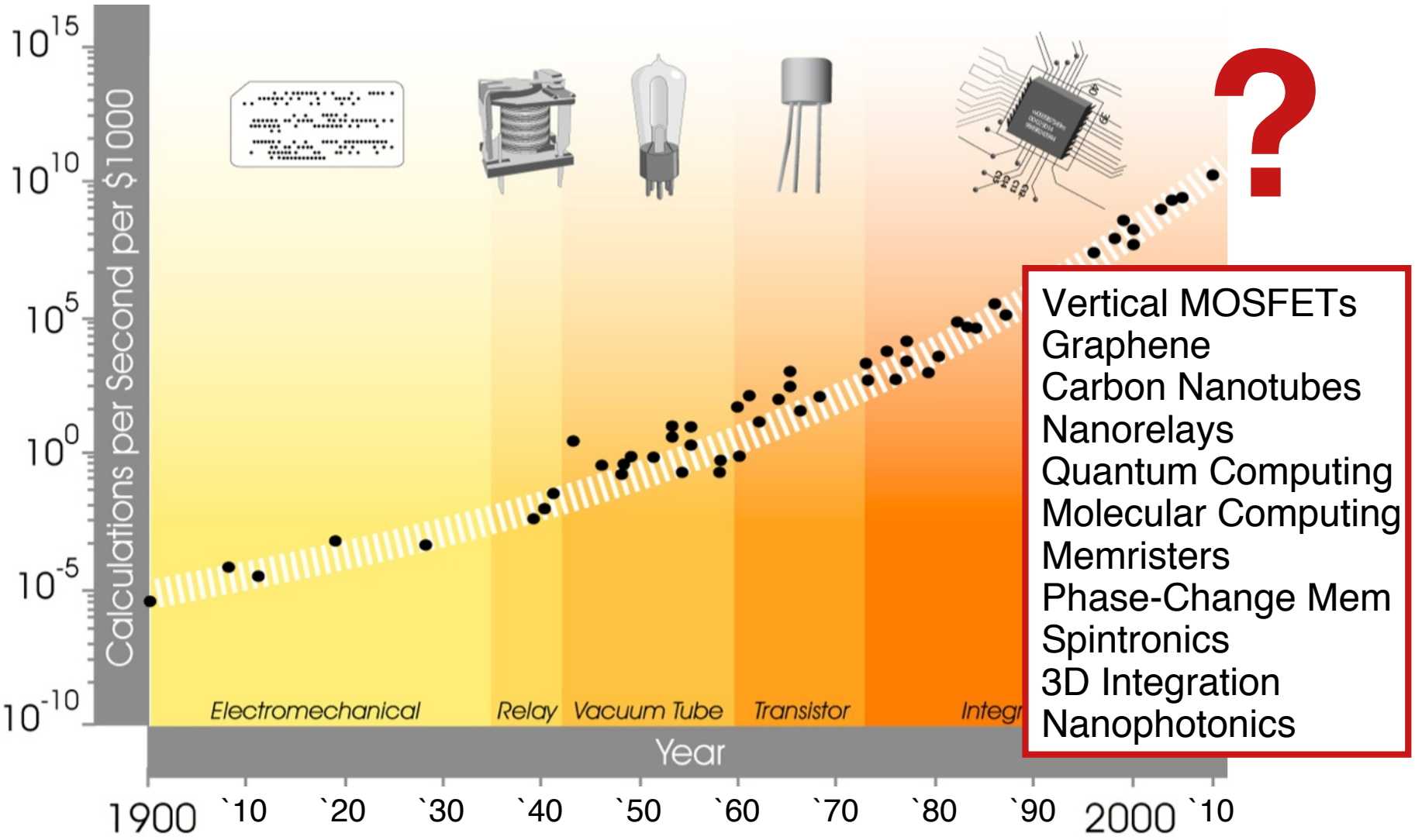


IBM Blue Gene Q Supercomputer
Thousands of
18-core processors

Trend 3: The Multicore "Hail Mary Pass"



Trend 4: Emerging Device Technologies



Adapted from R. Kurzweil. "The Singularity is Near." Penguin Books, 2006.

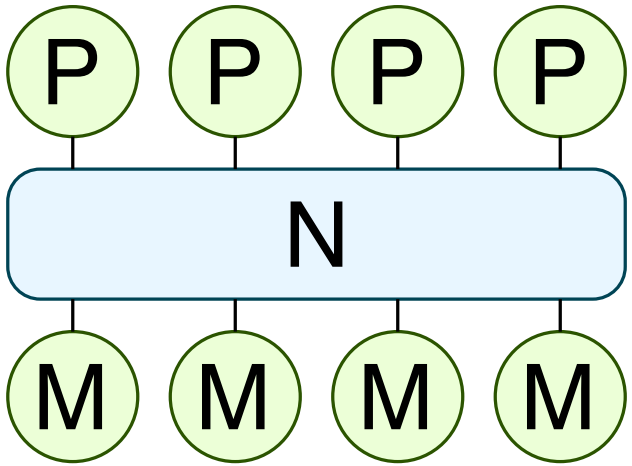
Key trends in application requirements and technology constraints over the past decade have resulted in a radical rethinking of the processors, memories, and networks used in modern computing systems

Four Key Trends in Computer Engineering

1. Growing diversity in application requirements motivate growing diversity in computing systems
2. Energy and power constrain systems across the entire computing spectrum
3. Transition to multiple cores integrated onto a single chip
4. Technology scaling challenges motivate new emerging processor, memory, and network device technologies

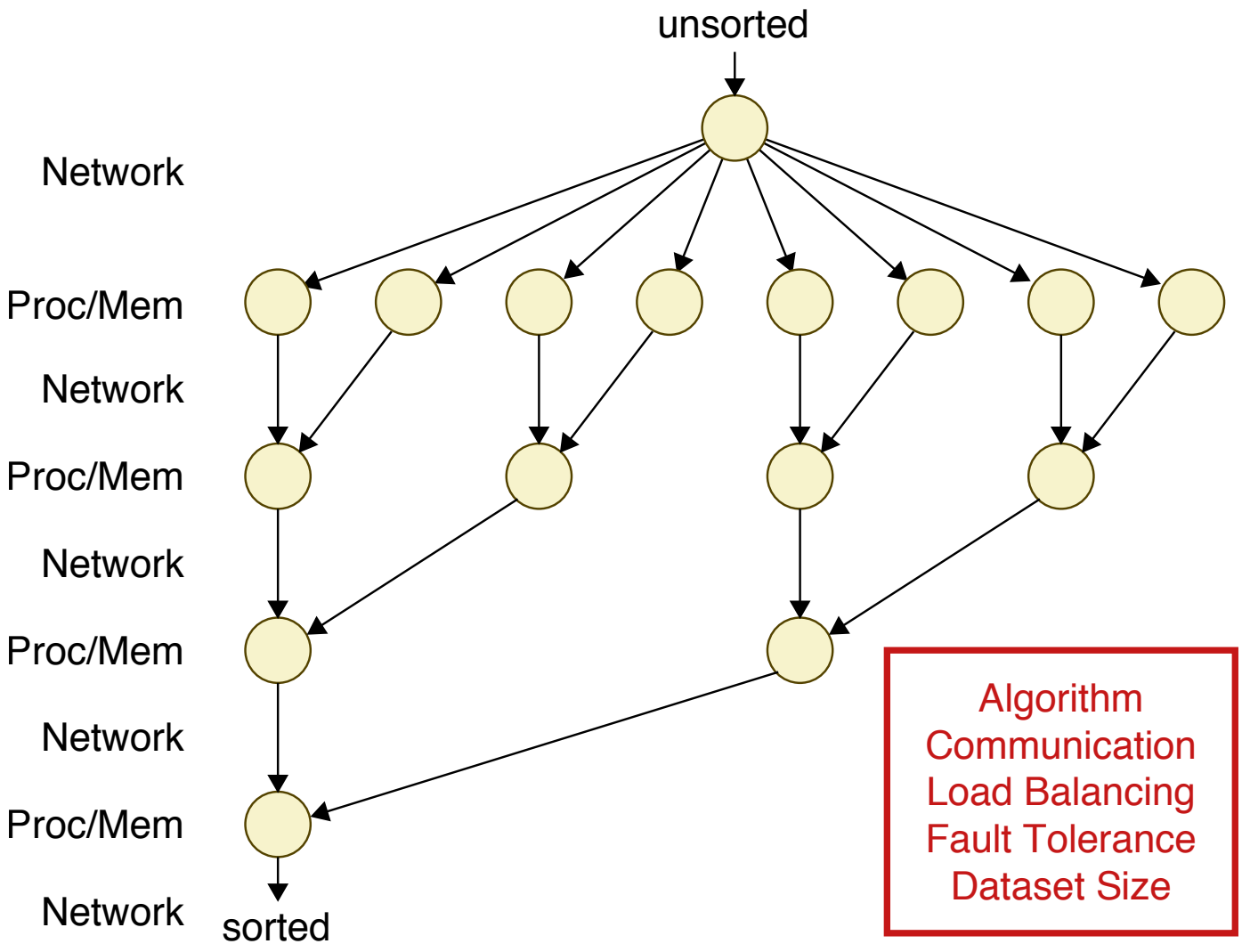
Activity #2: Sorting with a Parallel Processor

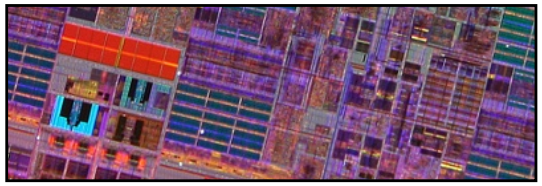
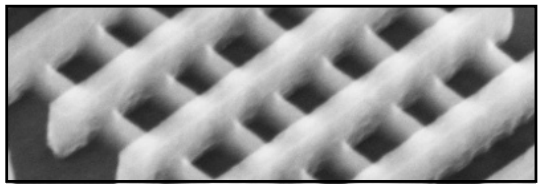
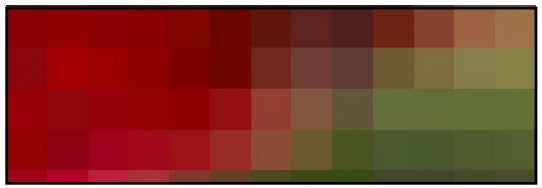
- **Application:** Sort 32 numbers
- **Simulated Parallel Computing System**
  - Processor: Group of 2–8 students
  - Memory: Worksheet, scratch paper
  - Network: Communicating between students



- **Activity Steps**
  1. Discuss strategy with group
  2. When instructor starts timer, master processor flips over worksheet
  3. Sort 32 numbers as fast as possible
  4. Lookup when completed and write time on worksheet
  5. *Master processor only* raises hand
  6. When everyone is finished, then analyze data

Activity #2: Discussion




Talk Outline

ECE Overview

What is Computer Engineering?

Trends in Computer Engineering

Computer Engineering Design

What do computer engineerings actually do?

General Science

Discover truths about nature

Ask question about nature

↓

Construct hypothesis

↓

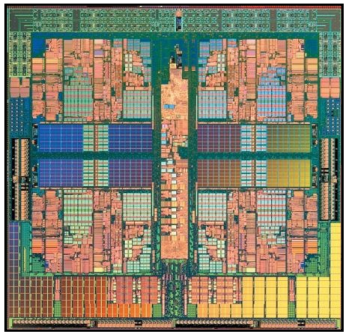
Test with experiment

↓

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

How do we design something so incredibly complex?

Computer Engineering

Explore design space for a new system

Design and model baseline system

↓

Ask question about system

↓

Test with experiment

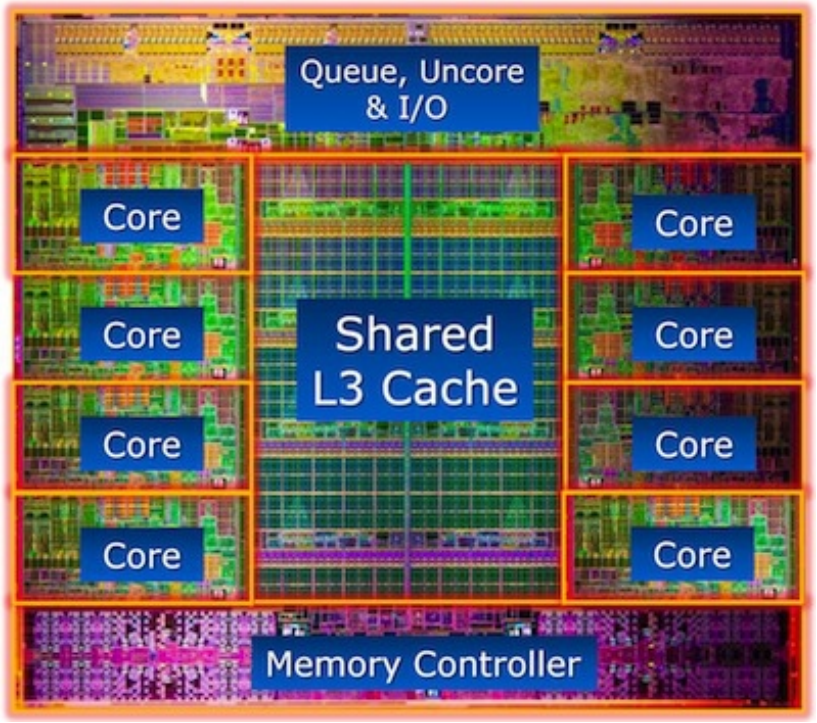
↓

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

Design Principles

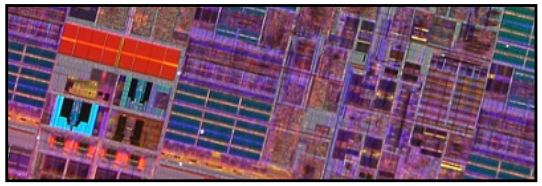
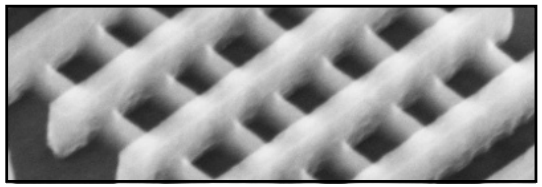
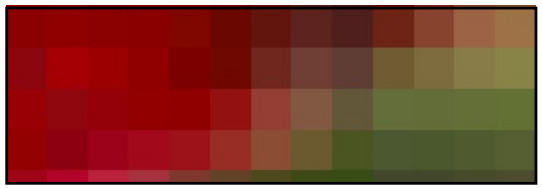
- Modularity – Decompose into components with well-defined interfaces
- Hierarchy – Recursively apply modularity principle
- Encapsulation – Hide implementation details from interfaces
- Regularity – Leverage structure at various levels of abstraction
- Extensibility – Include mechanisms/hooks to simplify future changes

Design Patterns

- Processors, Memories, Networks
- Control/Datapath Split
- Single-Cycle, FSM, Pipelined Control
- Raw Port, Message, Method Interfaces

Design Methodologies

- Agile Hardware Development
- Test-driven Development
- Incremental Development


Take-Away Points

- ECE is a broad field focused on the study and application of electricity, micro-electronics, and electro-magnetism
- Computer engineering is the process of designing abstraction and implementation layers to meet application requirements within physical technology constraints
- We are entering an exciting new era of computer engineering with emerging applications and systems, a remarkable shift towards mainstream parallel processing, and significant technology challenges