Computer Organization and Application

Introduction

Xiangzhong FANG
xzfang@sjtu.edu.cn

- Reference Books

■《Computer Organization and Design – the Hardware/Software Interface 4th Edition》（ARM Edition）

■作者：
David A. Patterson, John L. Hennessy

■出版社：
4th英文版（Elsevier出版）2008；
ARM版（机械工业出版社英文影印版）2010.4

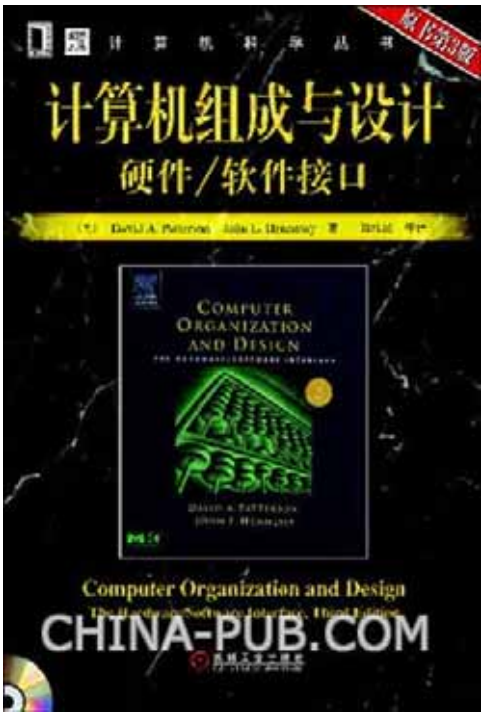
■ 英文原版教材在网络上可下载到扫描电子文档

• Reference Books

■计算机组成与设计—硬件/软件接口
（第三版）
Computer Organization And Design:
The Hardware/Software Interface
Third Edition

作者：David A. Patterson, John L. Hennessy
中译者：郑纬民 等 译
机械工业出版社 2007.4

■ 英文原版教材在网络上可下到 pdf 电子文档



Reference Books

■计算机组成与体系结构-性能设计（第8版）
Computer Organization and Architecture:
Designing for Performance(Eighth Edition)
作者：William Stallings
中译者：彭曼曼 等 译
机械工业出版社 2011.6

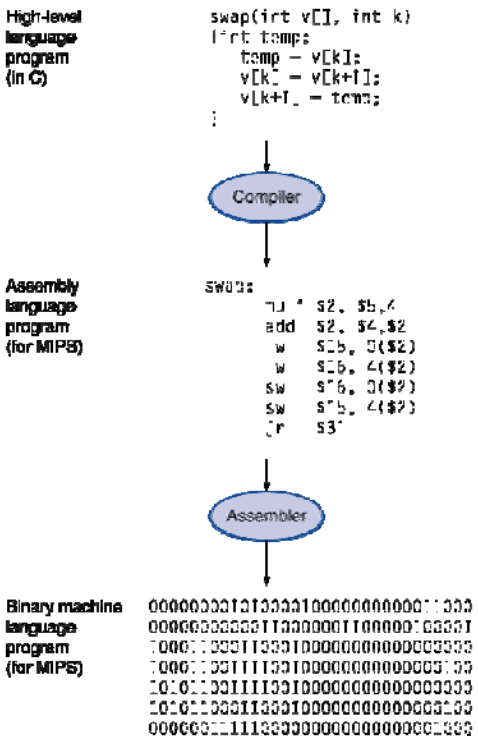
Question

- Is a computer intelligent?

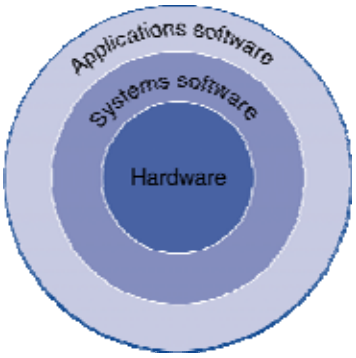
NO! It is you that make a compute to appear to be intelligent

Levels of Program Code

- High-level language
  - —Level of abstraction closer to problem domain
  - —Provides for productivity and portability
- Assembly language
  - —Textual representation of instructions
- Hardware representation
  - —Binary digits (bits)
  - —Encoded instructions and data

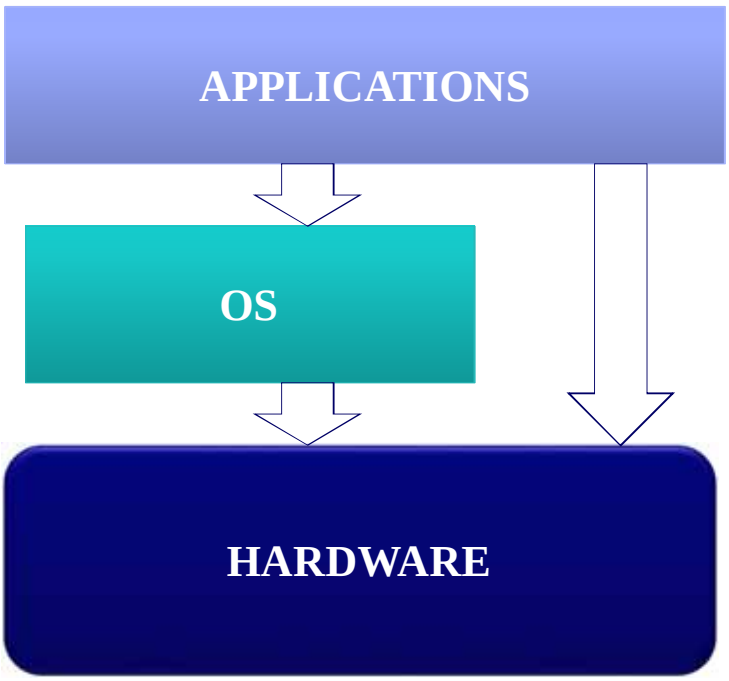


Below Your Program

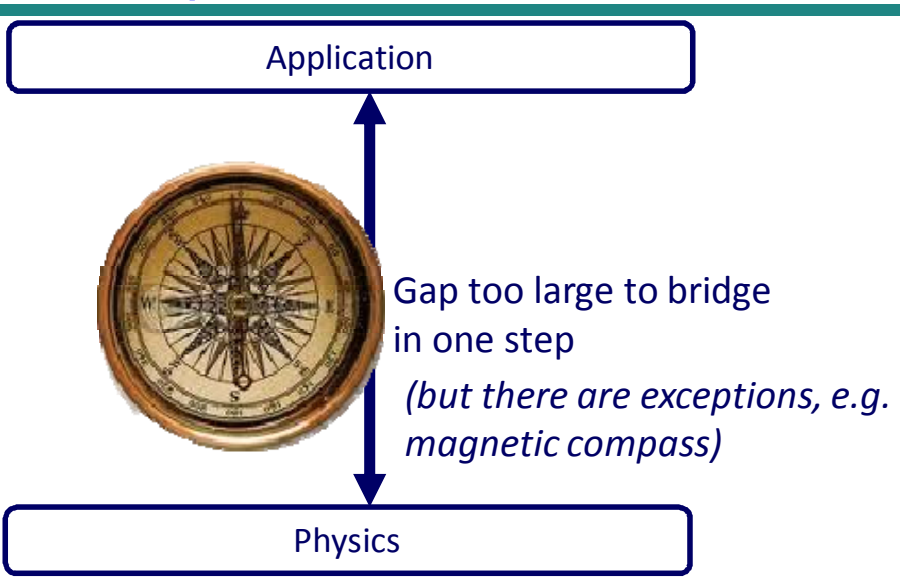


- Application software
  - —Written in high-level language
- System software
  - —Compiler: translates HLL code to machine code
  - —Operating System: service code
    - – Handling input/output
    - – Managing memory and storage
    - – Scheduling tasks & sharing resources
- Hardware
  - —Processor, memory, I/O controllers

Relation



What is Computer Architecture?



In its broadest definition, computer architecture is the *design of the abstraction layers* that allow us to implement information processing applications efficiently using available manufacturing technologies.

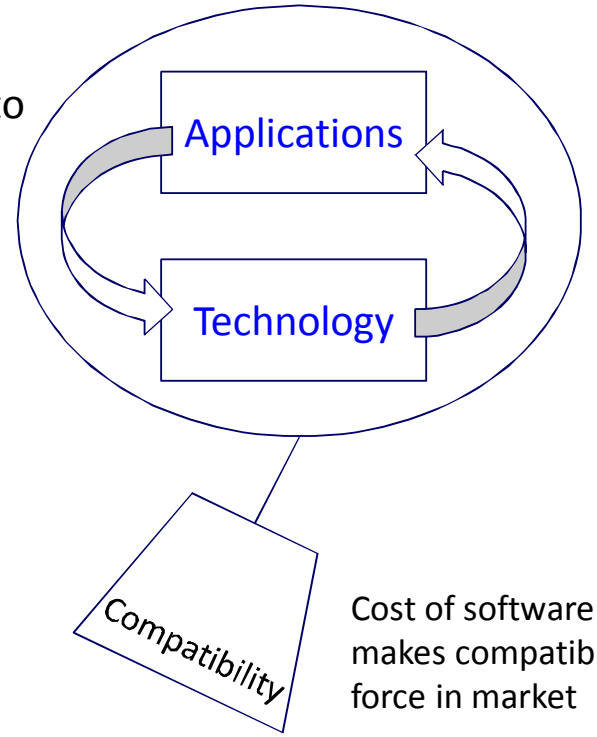
Abstraction Layers in Modern Systems

- Application
- Algorithm
- Programming Language
- Operating System/Virtual Machines
- Instruction Set Architecture (ISA)
- Microarchitecture
- Gates/Register-Transfer Level (RTL)
- Circuits
- Devices
- Physics

Our Course

Architecture continually changing

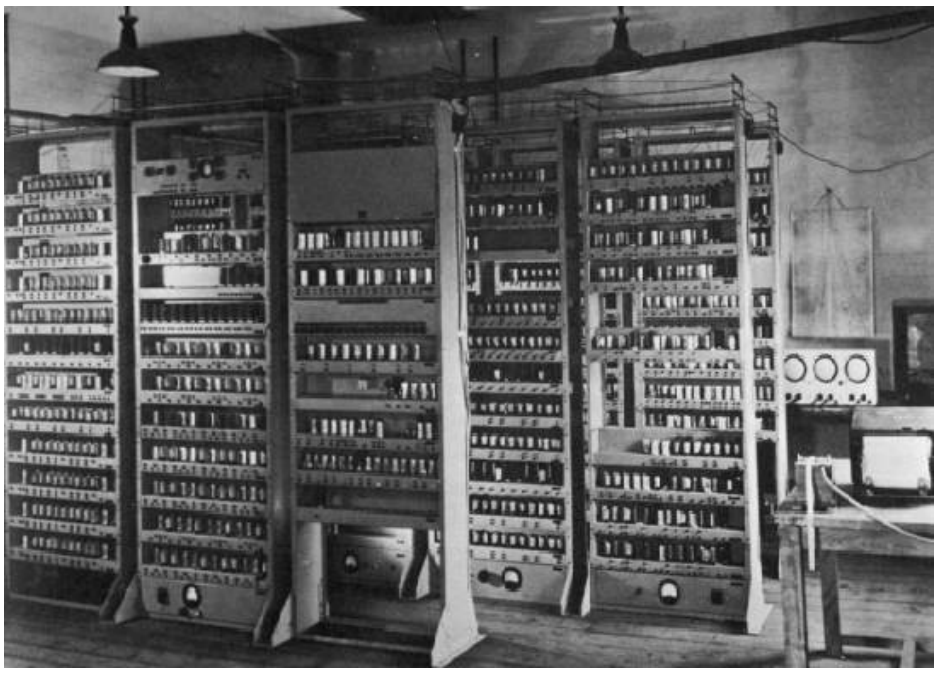
Applications suggest how to improve technology, provide revenue to fund development



Improved technologies make new applications possible

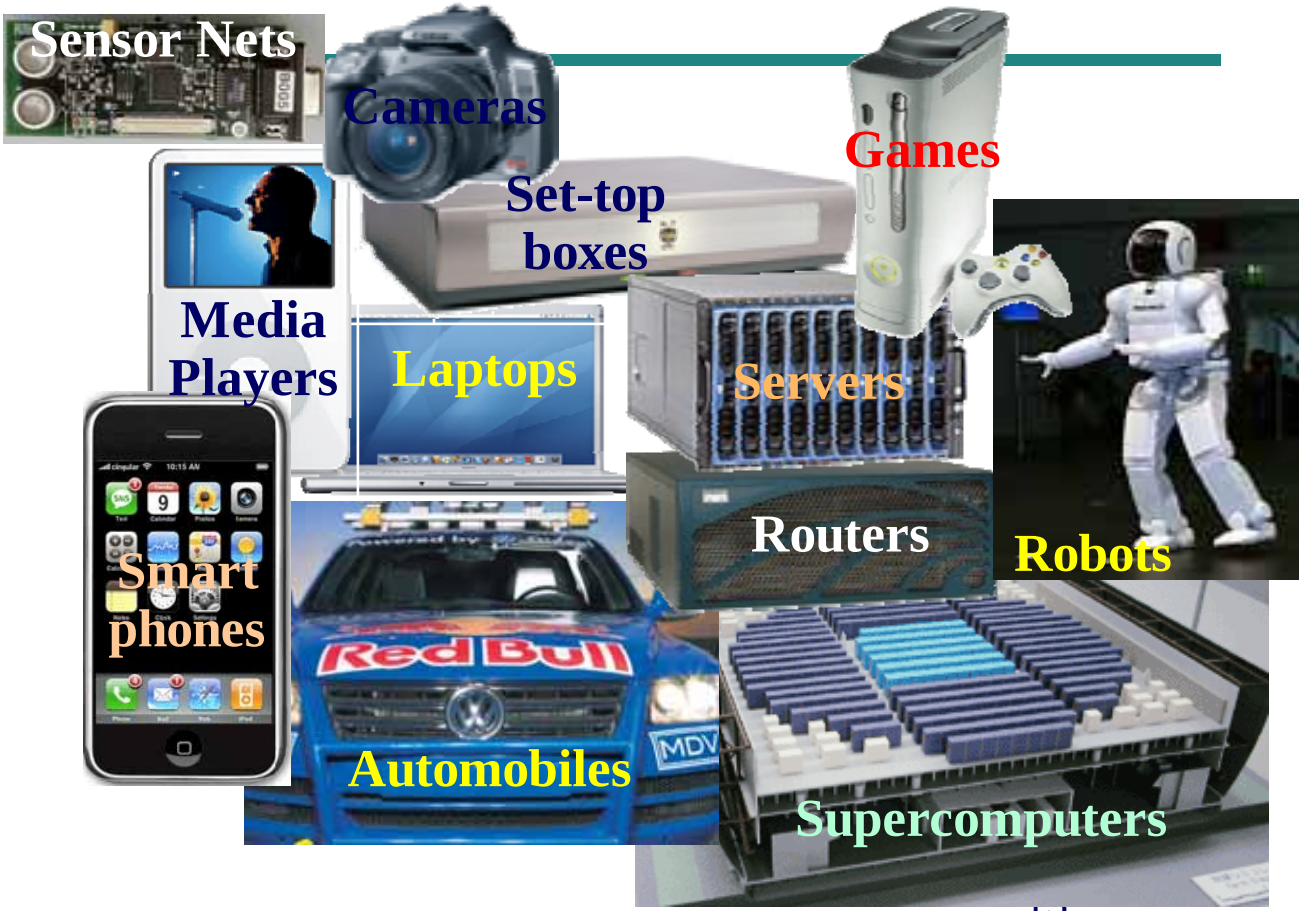
Cost of software development makes compatibility a major force in market

Computing Devices Then...

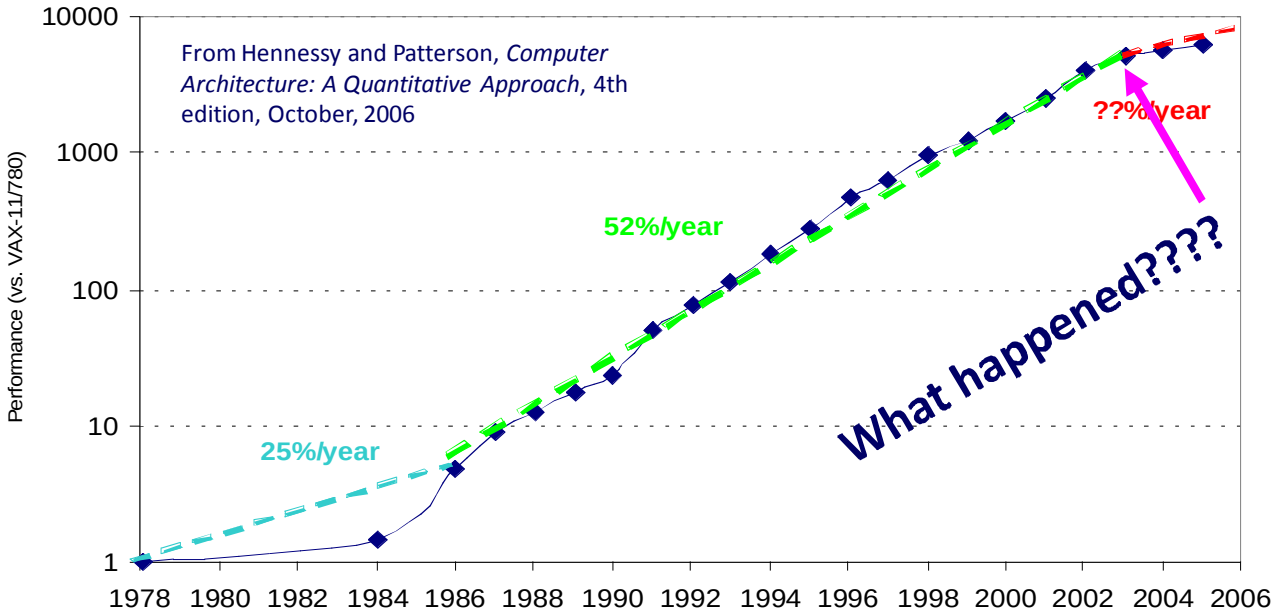


EDSAC, University of Cambridge, UK, 1949

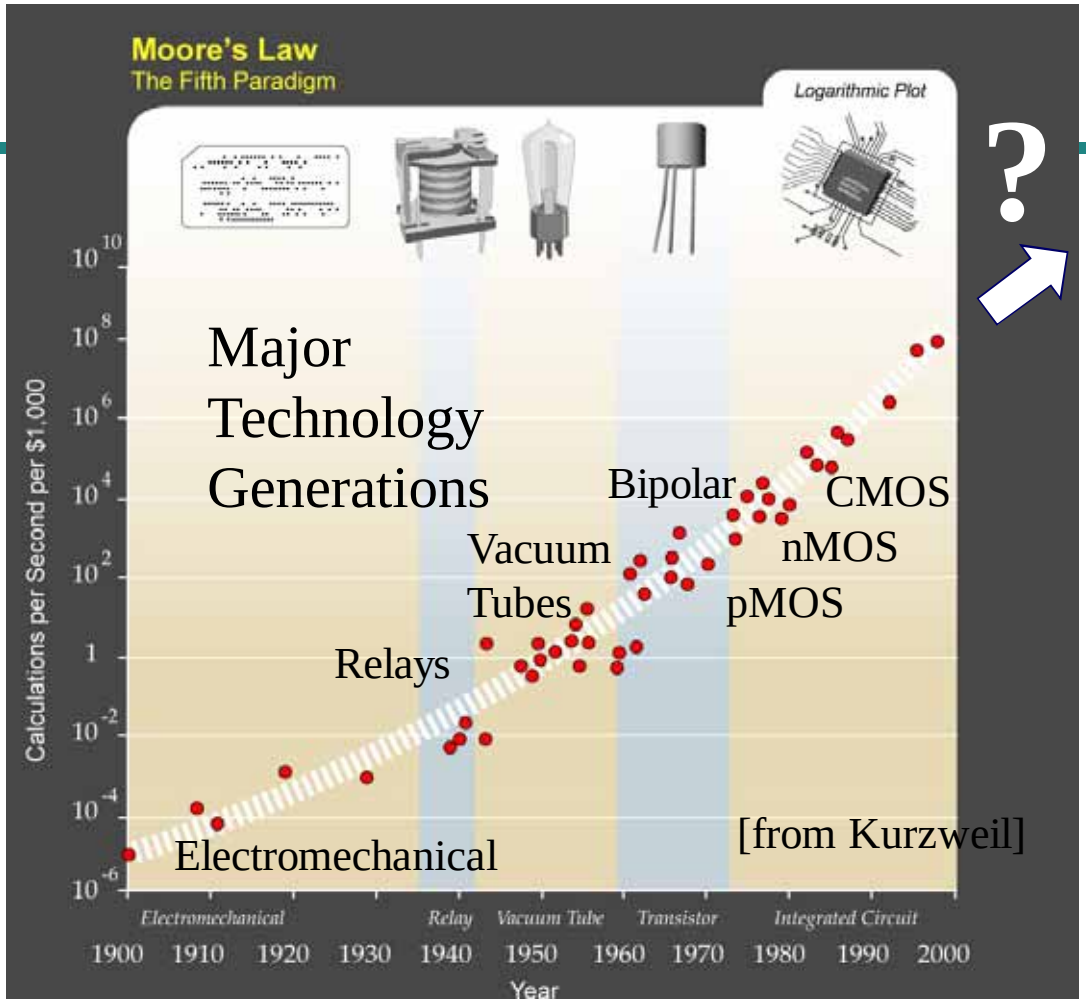
Computing Devices Now



Uniprocessor Performance



- VAX : 25%/year 1978 to 1986
- RISC + x86: 52%/year 1986 to 2002
- RISC + x86: ??%/year 2002 to present



The End of the Uniprocessor Era

Single biggest change in the history of computing systems

Question

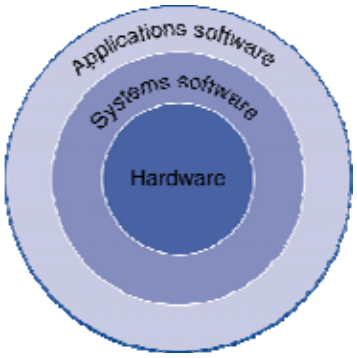
- You develop a program and run it on a computer.
- What factors affect the performance of your program?

Understanding Performance

- Algorithm
  - —Determines number of operations executed
- Programming language, compiler, architecture
  - —Determine number of machine instructions executed per operation
- Processor and memory system
  - —Determine how fast instructions are executed
- I/O system (including OS)
  - —Determines how fast I/O operations are executed

Architecture: Conceptual Design

- Architecture is those attributes visible to the programmer
  - —Instruction set,
  - —number of bits used for data representation
  - —I/O mechanisms
  - —addressing techniques
- e.g. Is there a multiply instruction?



Architecture: Hardware/Software Interface

Organization

- Organization is how features are implemented
  - —Control signals
  - —Interfaces
  - —Memory technology
- e.g. Is there a hardware multiply unit or is it done by repeated addition?

Architecture & Organization

- Contents of computer architecture
  - —Instruction set design, performance evaluation, pipelining, vector machine, SMP, clusters, parallel processing
- Contents of computer organization
  - —Arithmetic operations, ALU
  - —Memory and memory organization
  - —Instruction set and addressing
  - —Central processing unit (CPU)
  - —Bus
  - —I/O

Architecture & Organization

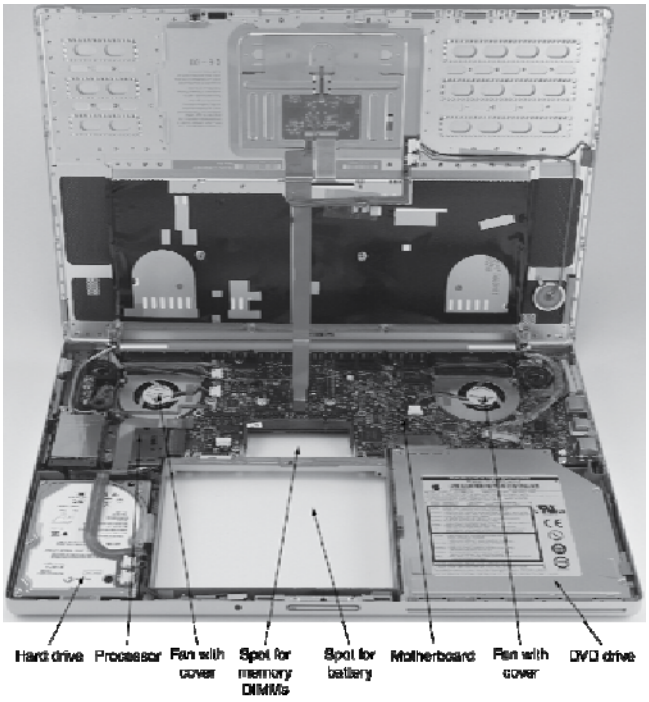
- All Intel x86 family share the same basic architecture
- The IBM System/370 family share the same basic architecture
- This gives code compatibility
  - —At least backwards
- Organization differs between *different versions*

Intel 8086

Intel Core duo

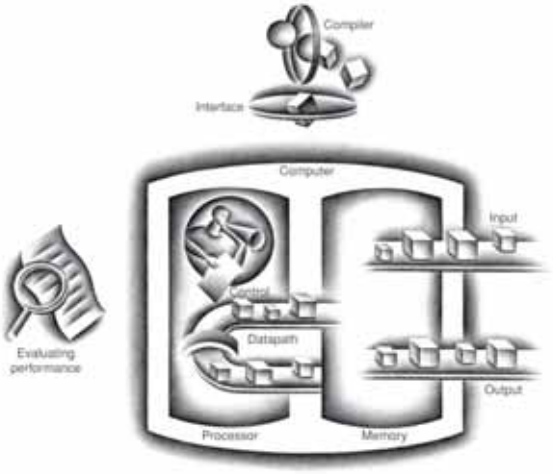
AMD Athlon

Opening the Box



Components of a Computer

The BIG Picture



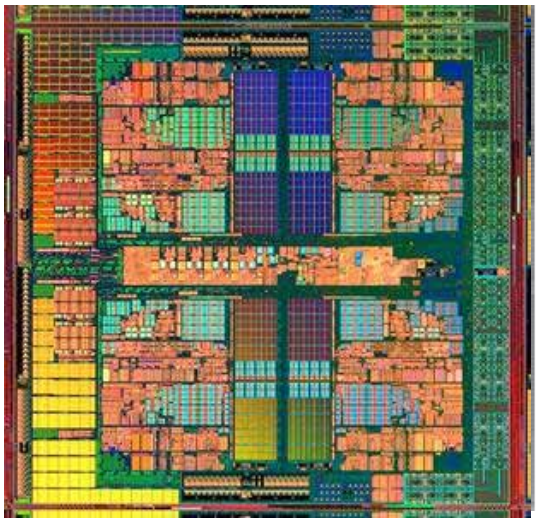
- Same components for all kinds of computer
  - —Desktop, server, embedded
- Input/output includes
  - —User-interface devices
    - – Display, keyboard, mouse
  - —Storage devices
    - – Hard disk, CD/DVD, flash
  - —Network adapters
    - – For communicating with other computers

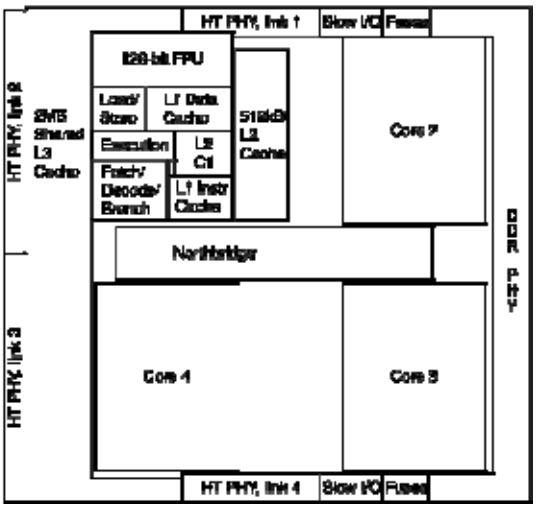
Inside the Processor (CPU)

- Datapath: performs operations on data
- Control: sequences datapath, memory, ...
- Cache memory
  - —Small fast SRAM memory for immediate access to data

Inside the Processor

- AMD Barcelona: 4 processor cores





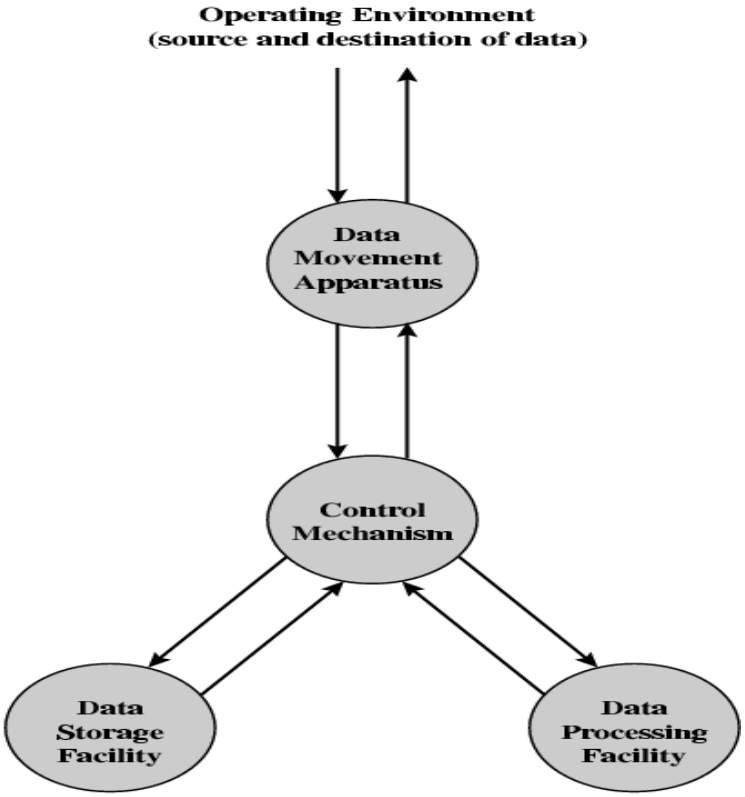
Structure & Function

- Structure is the way in which components relate to each other
  - —How different components, like ALU, control, I/O, memory are connected?
  - —How they interface with each other?
- Function is the operation of individual components as part of the structure
  - —What is the function of a component?

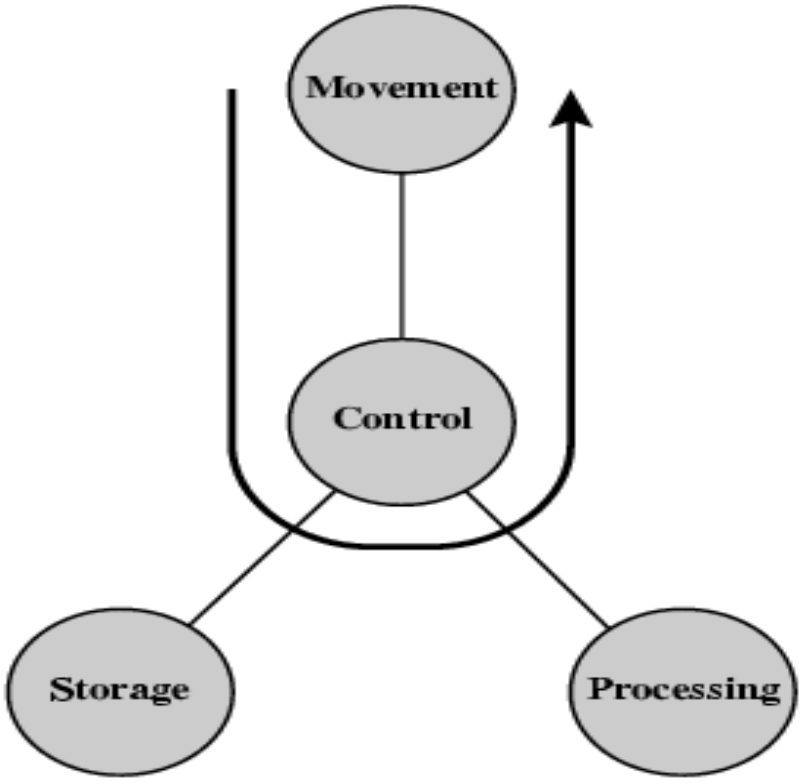
Function

- All computer functions are:
  - —Data processing
  - —Data storage
  - —Data movement
  - —Control

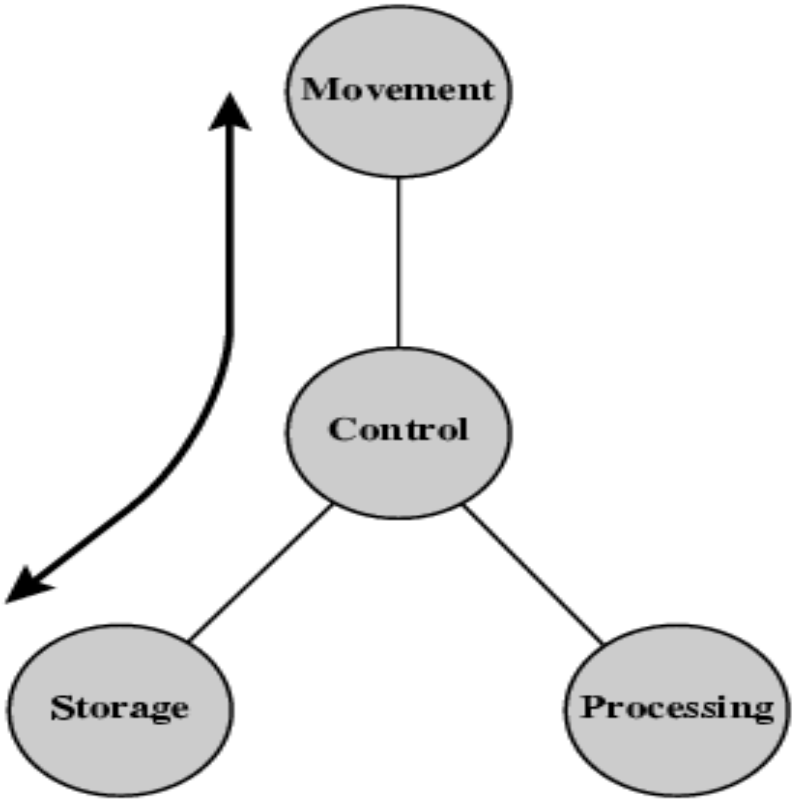
Functional View



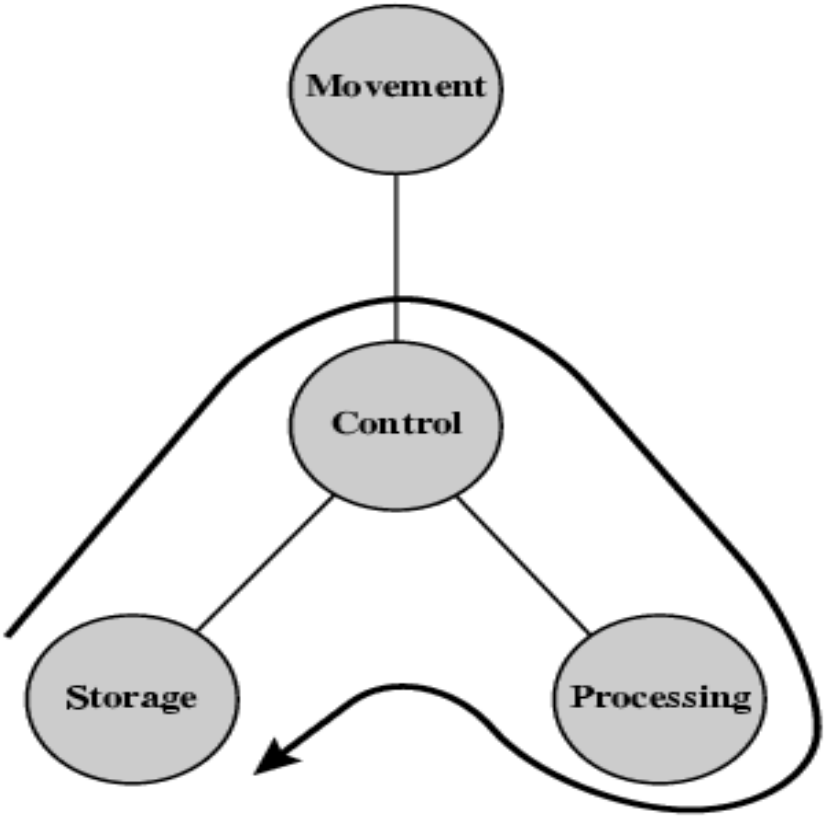
Operations (a) Data movement



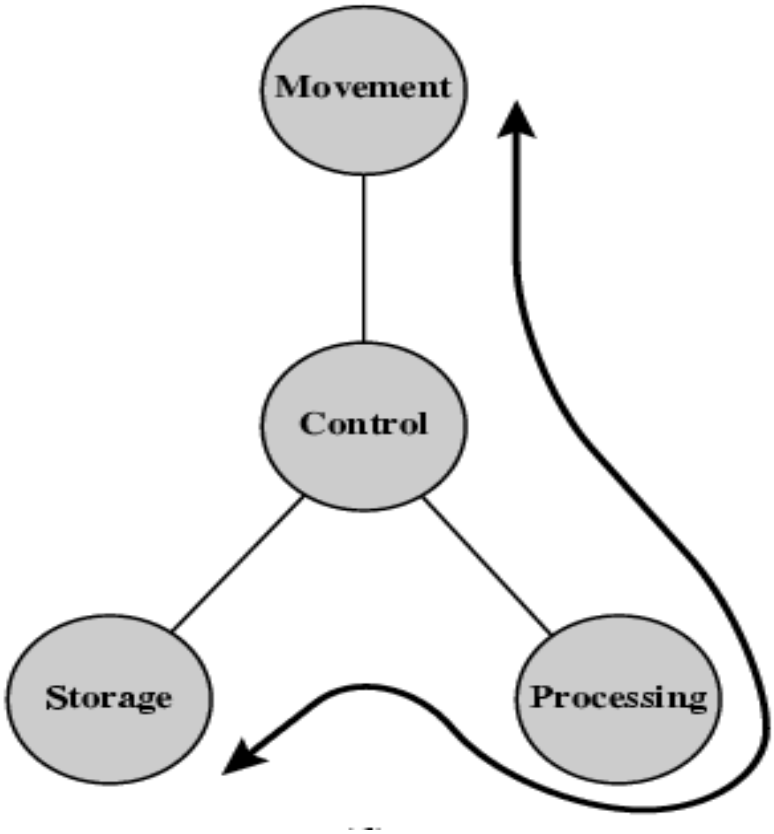
Operations (b) Storage



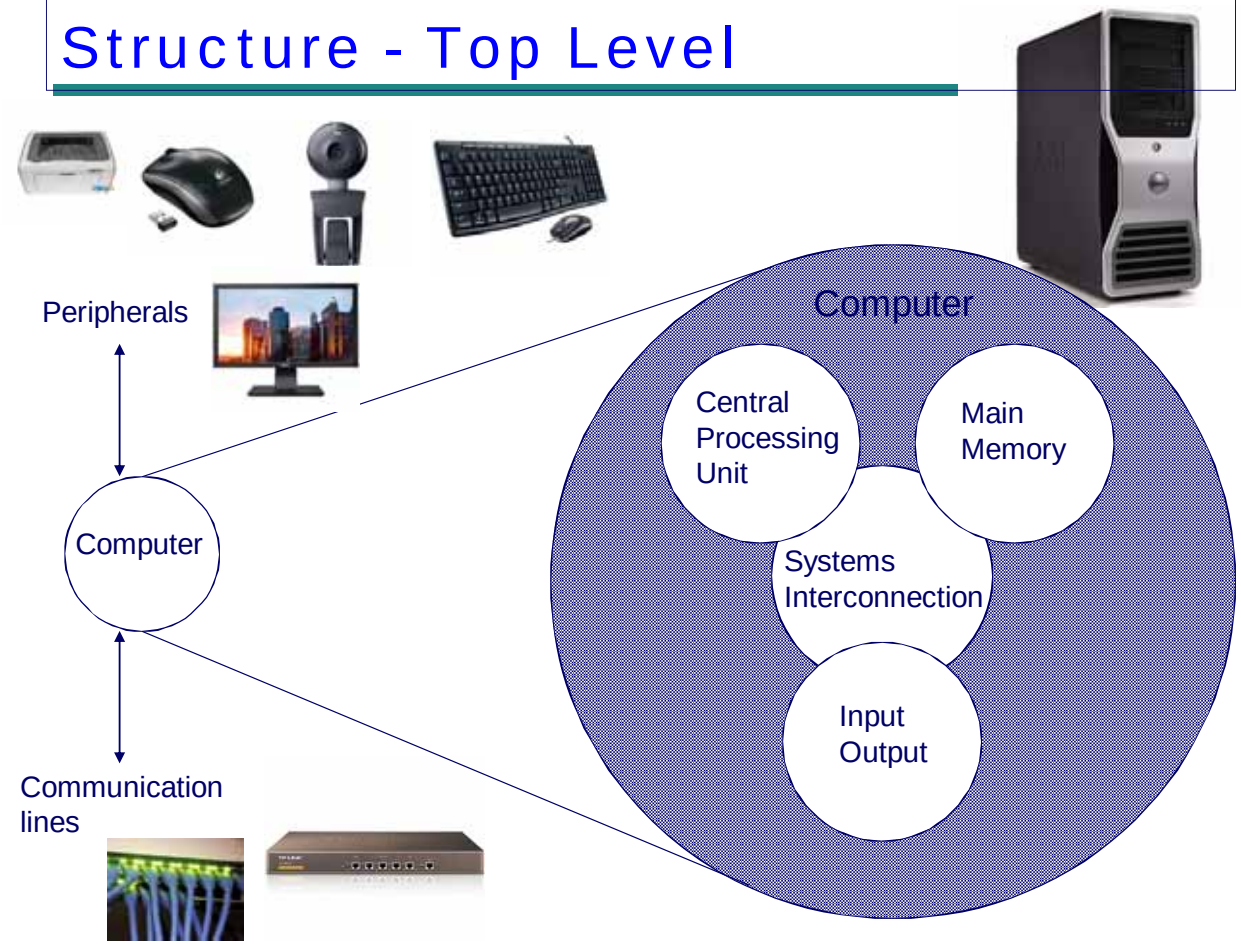
Operation (c) Processing from/to storage



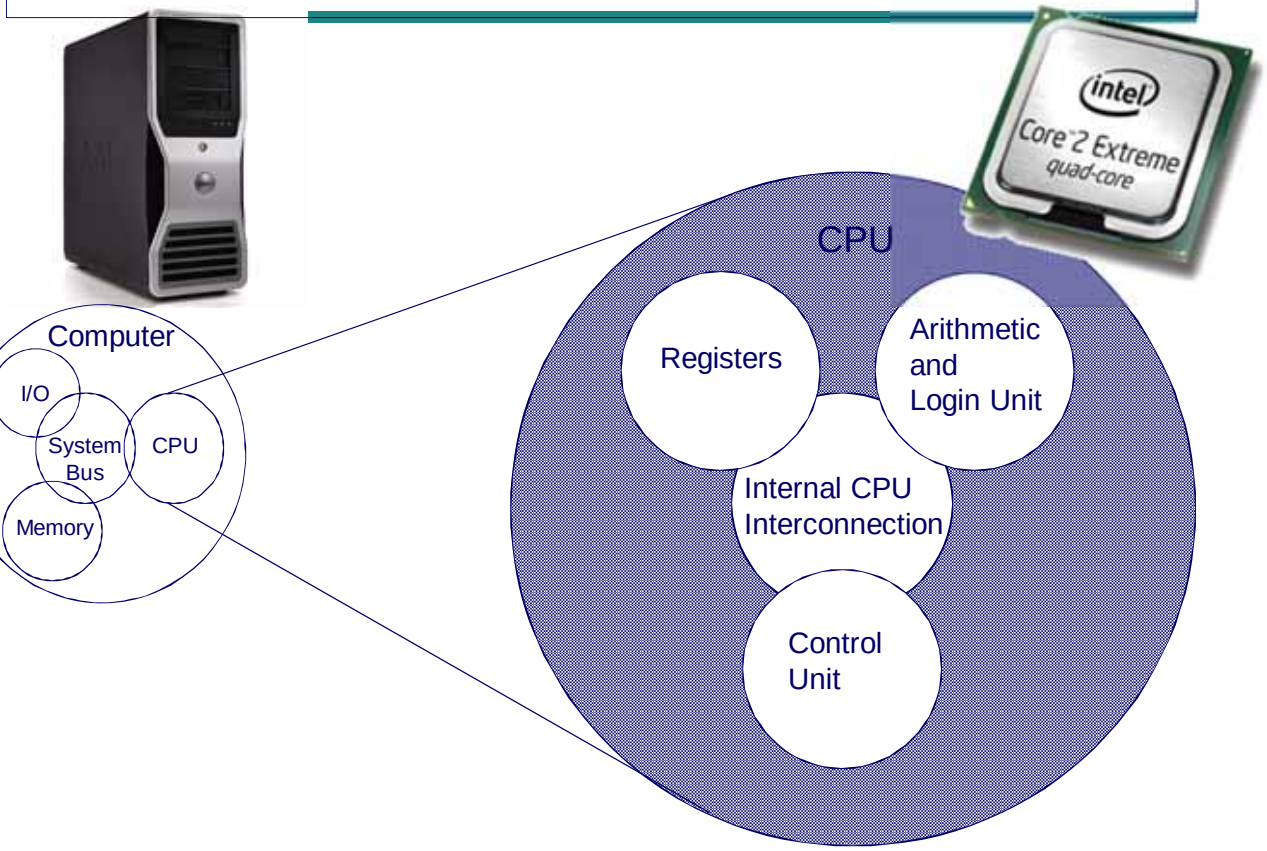
Operation (d)
Processing from storage to I/O



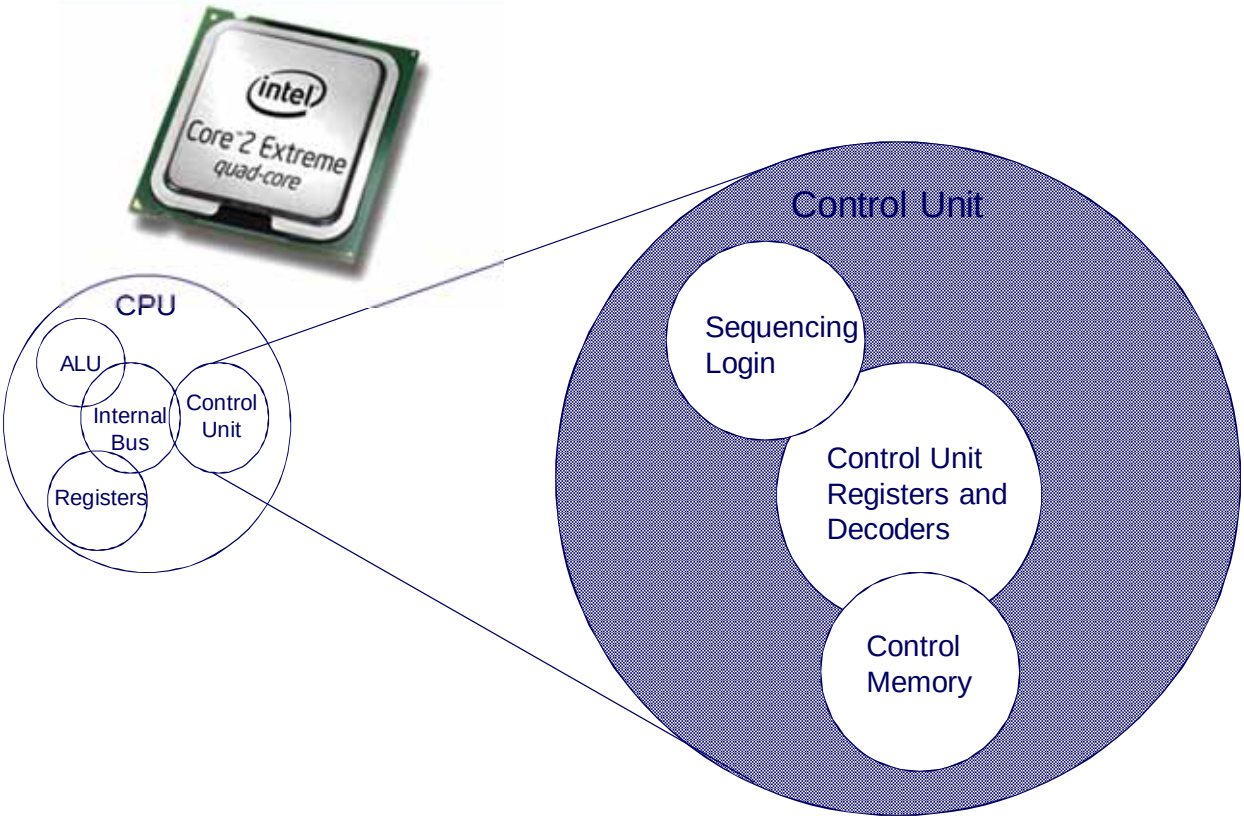
Structure - Top Level



Structure - The CPU



Structure - The Control Unit



Outline of the Course

- Computer Evolution and Performance
- Computer Interconnection Structures
- Internal Memory
- External Memory
- Input/Output
- Operating Systems Support
- Computer Arithmetic
- Instruction Sets

Outline of the Course (2)

- CPU Structure and Function
- Reduced Instruction Set Computers
- Superscalar Processors
- Control Unit Operation
- Microprogrammed Control
- Multiprocessors and Vector Processing

Why Study COA?

- It is a core course for computer science or computer engineering
  - —As recommended by IEEE/ACM Curricula 2008
- To write better programs that run more efficiently on a real machine
- To understand the tradeoff among various components, such as CPU speeds and memory size
  - —Spending more for various alternatives

What You Will Learn

- How programs are translated into the machine language
  - —And how the hardware executes them
- The hardware/software interface
- What determines program performance
  - —And how it can be improved
- How hardware designers improve performance
- What is parallel processing

Acknowledgements

- These slides contain material developed and copyright by:
  - —Arvind (MIT)
  - —Krste Asanovic (MIT/UCB)
  - —John Kubiatowicz (UCB)
  - —David Patterson (UCB)
  - —Geng Wang (SJTU)
  - —Yanmin Zhu (SJTU