Computer Architecture
(컴퓨터구조)

2019 Fall

Joonho Kong

Thanks Prof. Sung Woo Chung in Korea University for allowance of using the ppt materials

Contact Information

- Instructor: Joonho Kong (공준호)

- joonho.kong@knu.ac.kr
  - E-mail communication is really encouraged
- Office
  - IT3-309
- Office Hour
  - By appointment

My research interest

- Components
  - SOC (System On Chip)
  - Microprocessor
  - Memory system and hierarchy
- Issues
  - Low power / low energy
  - Reliability
  - Cost-efficient design
  - Data transfer reduction
  - Hardware security

TA

- Youngmin Kim (김영민)
  - catzzang1@naver.com
    - E-mail communication is also encouraged

Overview

- This course focuses on the engineering methodology, design techniques, correctness criteria, technology trends, and evaluation methods involved in the computer architecture. Major concepts discussed in this course include **hardware-software interface**, **processor design**, **pipelining**, **memory hierarchies**, and **emerging issues**.

Goal

- To understand hardware/software co-design
- To understand processor architecture
- To understand how to optimize software based on knowledge on hardware
- To understand memory hierarchy

Textbook & material

- John L. Hennessy and David A. Patterson, Computer Organization and Design : The Hardware/Software Interface, Elsevier/Morgan Kaufmann Publishers,
  - **RISC-V Edition**
- Main material for this course is ppt slide
  - It will be uploaded in LMS bulletin board prior to the class
- I made the lecture slides by referring to the textbook
  - It's up to you whether or not buying the textbook

Quizzes

- Quizzes
  - 4~ quizzes
  - No announcement in advance

Course plan (tentative)

- Week 1: Course intro / performance
- Week 2: Performance / Instruction set architecture 1
- Week 3: Instruction set architecture 2
- Week 4: Processor datapath basics and single-cycle implementation
- Week 5: Processor Pipelining
- Week 6: Processor Pipelining 2
- Week 7: Processor Pipelining 3
- Week 8: Mid-term exam

* course plan is changed a little compared to the syllabus

Course plan (tentative)

- Week 9: Advanced pipelining and design techniques
- Week 10: Memory hierarchy 1
- Week 11: Memory hierarchy 2
- Week 12: Memory hierarchy 3 / virtual memory
- Week 13: Parallel processors
- Week 14: Recent issues in computer architecture
- Week 15: Final exam

  - We will not cover Chapter 3 in the textbook (Arithmetic in Computers)

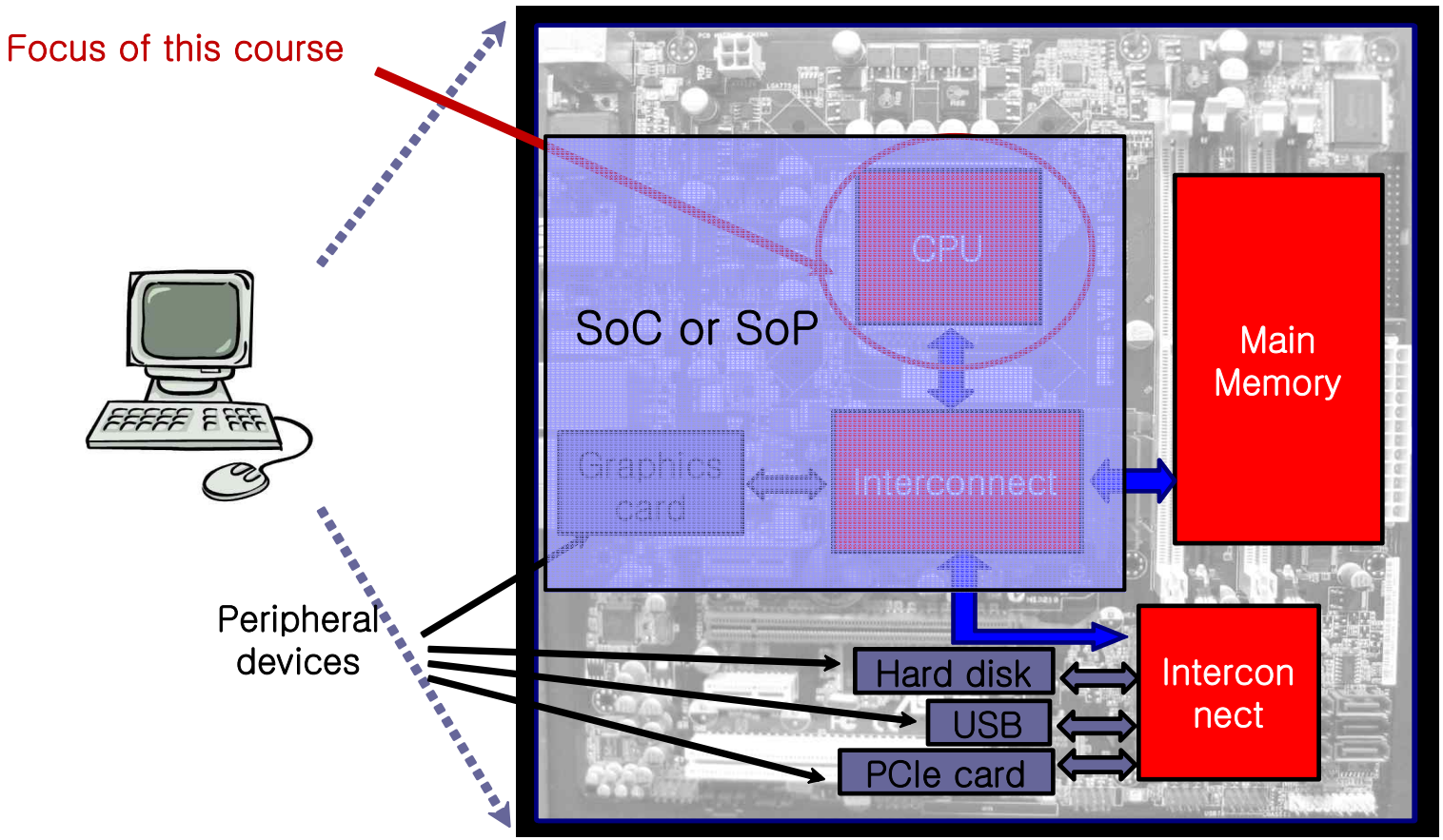
Grading

- Mid Exam (40)
- Final Exam (40)
- Attendance/Participation/Quizzes (20)
  - This portion may be over 20 points / 100 points
  - If you have good participation points and quiz points, you can earn more than 20 points

Details of Grading

- Absence
  - Each absence = -3p
  - In case of more than 8 classes of the absence, F grade will be given
  - Inevitable reasons for absence
    - **Send me an email PRIOR TO the class**
  - TA will check the attendance at 1:25PM on Mon / 4:30PM on Wed
    - Due to a large number of students
- Late
  - Late twice = one absence
  - Please don't be late
- Participation
  - Participation point will be given if you are very active in the class
    - Answers well
    - Good question
- Course retake penalty
  - For the students who retake this course, the highest available grade will be B+

A Computer System



But, don't forget the big picture!

* Modern computer systems typically combines interconnects, GPUs into one chip or one package along with CPU

What is a processor (CPU)?

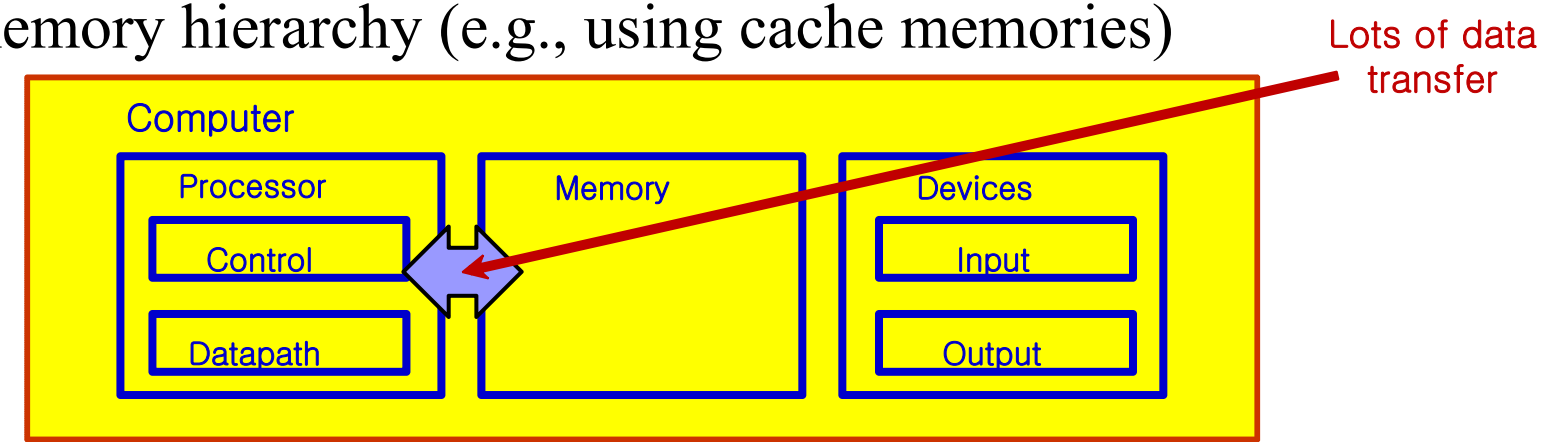
- Traditional definition
  - A computer central processing unit (CPU) built as a single tiny semiconductor chip or as a small number of chips
  - Contains the arithmetic and control logic circuitry necessary to perform the operations of a computer program
  - Intel Coffee Lake, Oracle SPARC, IBM Power9…

What is a processor (CPU)?

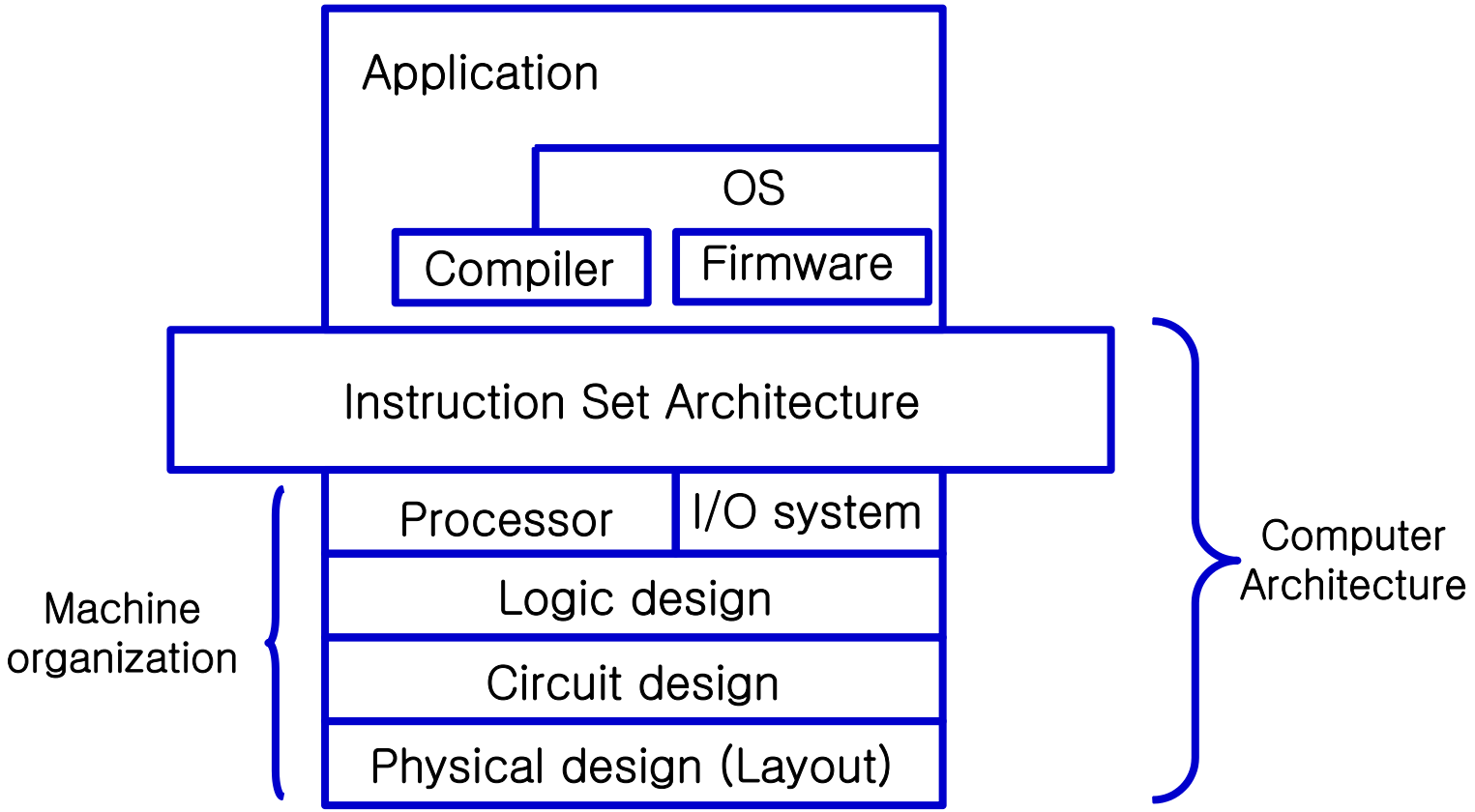
- Recent trend
  - A (micro)processor is now included in a system as a component (SoC: System-on-Chip / SoP: System-on-Package)
  - ARM, X86, RISC-V…
    - Various instruction sets are available
  - More importance on hardware/software co-design
    - Optimizing both hardware and software together for better performance and energy-efficiency
    - We will learn
      - ISA (instruction set architecture) – Software interface, but hugely affects hardware design
      - Processor hardware

What is a memory hierarchy?

- Von Neumann architecture
  - Program codes and data are stored in memory
  - Processing elements (e.g., CPU) loads codes from the memory and execute one by one
  - When doing the calculation, the data is fetched from the memory
  - To compensate for long memory access latency, we use memory hierarchy (e.g., using cache memories)

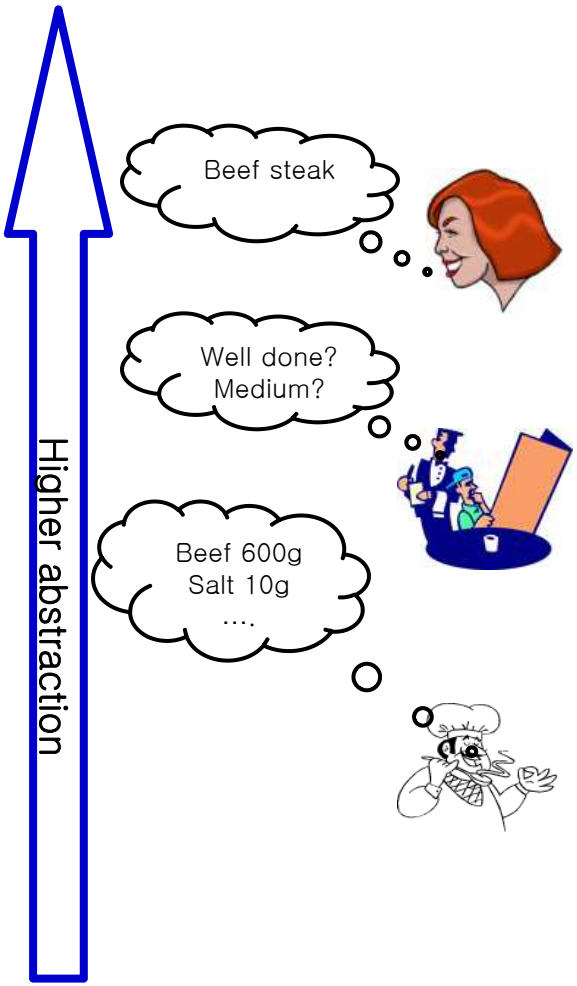


What is computer architecture?



- Computer architecture = Instruction set architecture (ISA) + Machine organization

Representing programs (Abstraction)



High-level language program

Compiler

```
temp =A[i];
A[i]=A[i+1];
A[i+1]=temp;
```

Assembly language program

Assembler

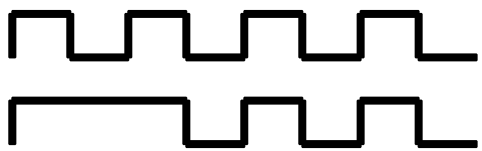
```
lw $12 0($2)
lw $13 4($2)
sw $13 0($2)
sw $12 4($2)
```

Machine language Program based on ISA

Machine interpretation

```
0000 1001 0101 1110 0001 0110 0001 1111
1010 0101 1110 0111 1110 0000 0110 1010
1001 0110 1010 0001 0100 0010 1101 1001
0110 1001 1001 0010 0110 1010 0101 0000
```

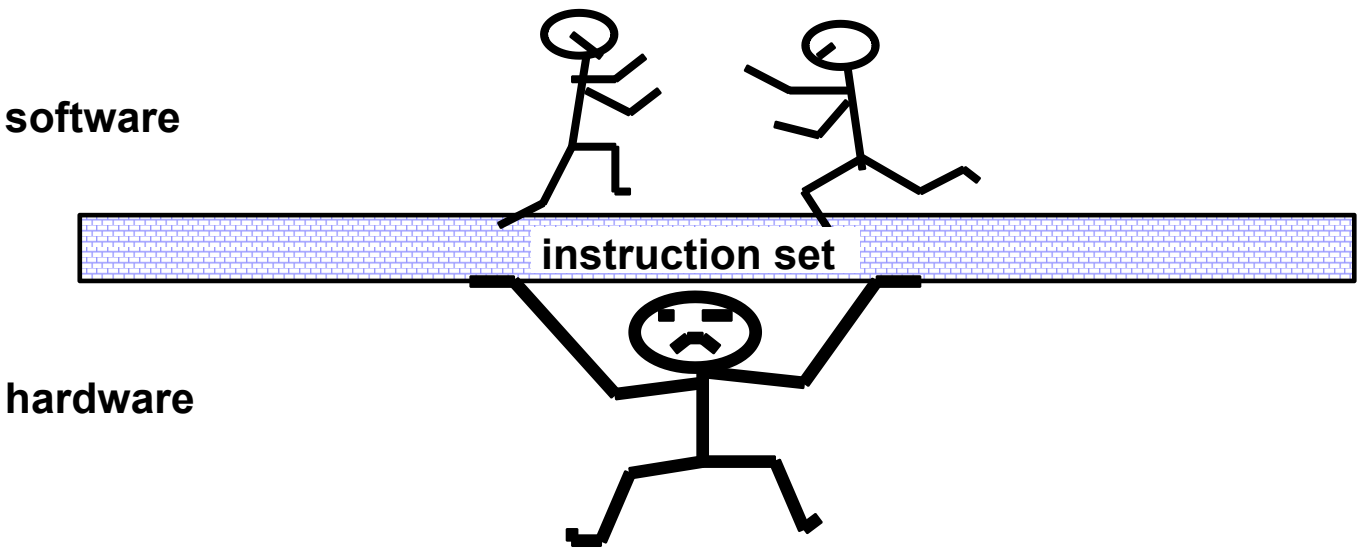
Control signal specification



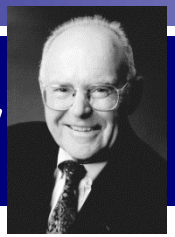
High/Low on control signals

Instruction Set Architecture: ISA

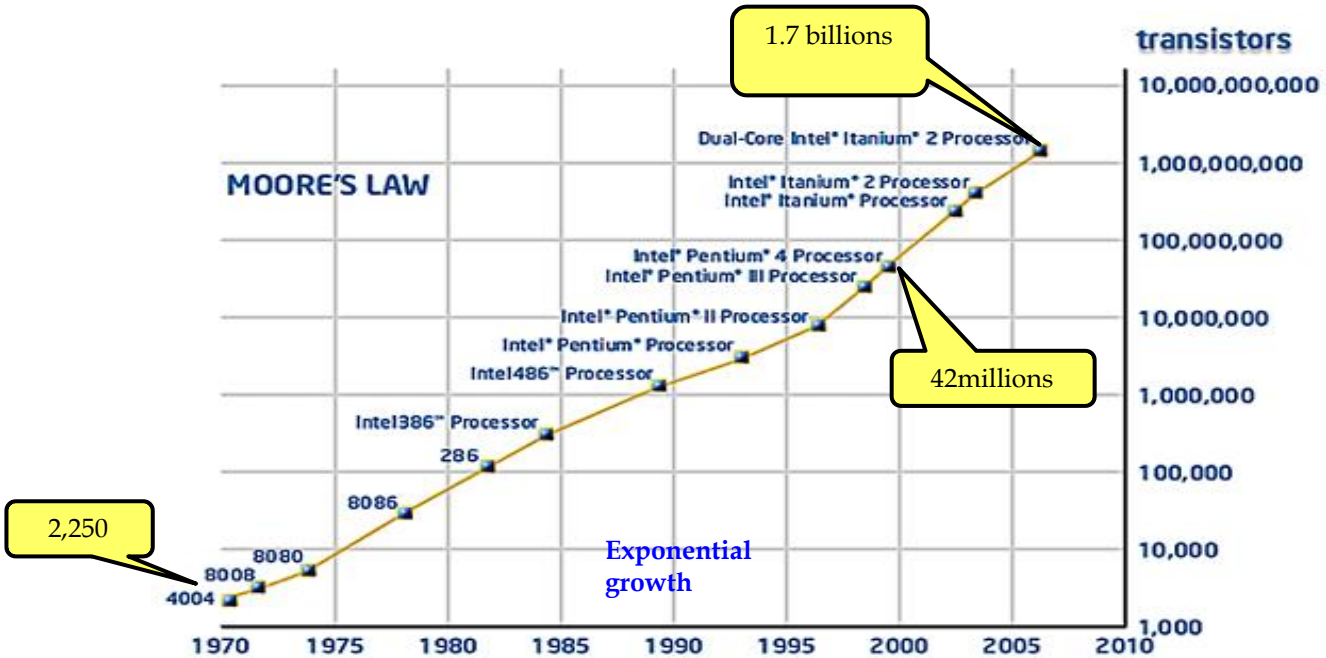
- A very important abstraction
  - Interface between hardware and low-level software
  - Functional interface for assembly-language programming
- Modern instruction set architectures:
  - IA-32, PowerPC, MIPS, SPARC, ARM, RISC-V and others



Moore's Law



- Transistor count will be doubled every 18 months



- Performance will be doubled every 18 months?
  - These days, performance is now hugely limited by power and thermal
  - Dark silicon problem
  - Performance gain is marginal when using general purpose processor
  - Nowadays, domain specific architecture is more important (out-of-the-scope of this course)