

COMPUTER ARCHITECTURE

Jo, Heeseung

Class materials

<http://cslab.cbnu.ac.kr>

- <http://software.cbnu.ac.kr> > 학과소개 > 교수진소개 > 조회승

Instructor

- Jo, Heeseung (heesn@cbnu.ac.kr)
- <http://cslab.cbnu.ac.kr>
- Room#327, S4-1 Bldg.
- Tel: 043-261-3757
- [The best way to contact me is via email](#)

Course description

Computer architectures have been drastically evolved since early 1980. Pipelining, cache, and branch prediction have been major features in modern microprocessor design. They are invented for high performance processors first, and then equipped in PC processors we are using every day.

Topics & Materials

Digital logic design - 데이터 표현

Digital logic design - 논리회로 기초

Digital logic design - 조합논리회로

Digital logic design - 순차논리소자

Computer Abstractions and Technology

Instructions Language of the Computer

Arithmetic for Computers

The Processor1

The Processor2

Memory Hierarchy 1

Memory Hierarchy 2

Storage and Other IO Topics

Other Topics ...

Prerequisites

Prerequisites

- Mathematics in high school
- Introduction to computer systems
- Digital logic design

You should be familiar with the followings:

- Basic computer organization
- Computer hardware
- C programming skills

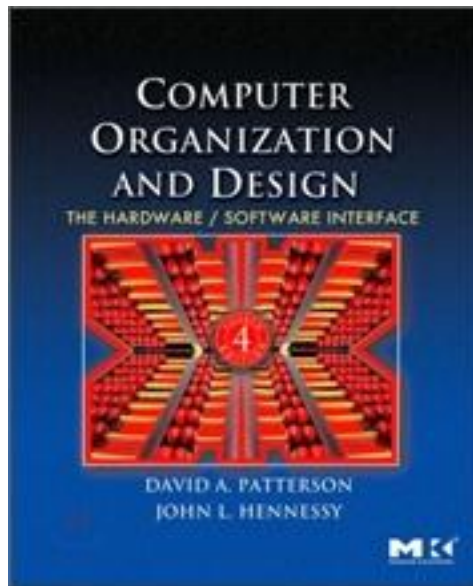
Timetable

| 교시 | 월 | 화 | 수(50) | 목 | 금(50) |
|----------------------------|--|--|---|--|-------|
| 50 | | | | | |
| 0교시 08:00 ~ 08:50 | | | | | |
| 1교시 09:00 ~ 09:50 | 연구윤리 및 연구과제 I 8836996-20 조희승 S4-1-201(21-201) 1교시 | 캡스톤디자인 I 5118047-01 조희승 S4-1-106(21-106) 1교시 | | | |
| 2교시 10:00 ~ 10:50 | 연구윤리 및 연구과제 I 8836996-20 조희승 S4-1-201(21-201) 2교시 | 캡스톤디자인 I 5118047-01 조희승 S4-1-106(21-106) 2교시 | | | |
| 3교시 11:00 ~ 11:50 | 연구윤리 및 연구과제 I 8836996-20 조희승 S4-1-201(21-201) 3교시 | 캡스톤디자인 I 5118047-01 조희승 S4-1-106(21-106) 3교시 | | | |
| 4교시 12:00 ~ 12:50 | | 캡스톤디자인 I 5118047-01 조희승 S4-1-106(21-106) 4교시 | | | |
| 5교시 13:00 ~ 13:50 | | | 운영체제 5118020-02 조희승 S4-1-104(21-104) 5교시 | | |
| 6교시 14:00 ~ 14:50 | | 컴퓨터구조 5118007-01 조희승 S4-1-106(21-106) 6교시 | | | |
| 7교시 15:00 ~ 15:50 | 운영체제 5118020-01 조희승 S4-1-106(21-106) 7교시 | 컴퓨터구조 5118007-01 조희승 S4-1-106(21-106) 7교시 | | 운영체제 5118020-02 조희승 S4-1-104(21-104) 7교시 | |
| 8교시 16:00 ~ 16:50 | 운영체제 5118020-01 조희승 S4-1-106(21-106) 8교시 | | 컴퓨터구조 5118007-01 조희승 S4-1-106(21-106) 8교시 | 운영체제 5118020-02 조희승 S4-1-104(21-104) 8교시 | |
| 9교시 17:00 ~ 17:50 | 운영체제 5118020-01 조희승 S4-1-106(21-106) 9교시 | | | | |

Textbook

Computer Organization and Design – The Hardware/Software Interface

- David A. Patterson and John L. Hennessy, Morgan Kaufmann Publishers
- [MIPS version](#) (Not ARM version)



Reference

동작 원리 중심의 논리회로와 컴퓨터 구조

- 전중남, 배움터

한 눈에 보이는 컴퓨터구조

- 전중남, 생능출판사

Computer Systems: A Programmer's Perspective

- Randal E. Bryant and David R. O'Hallaron,, Prentice Hall Inc.

Computer Architecture - A Quantitative Approach

- John L. Hennessy and David A. Patterson, Morgan Kaufmann Publishers.

Grading

Grading ratio (Subject to change)

- Exams: 75%
- Homework: 15%
- Etc: 10%

No cheating

- Sharing or copying of solutions
- Of course, cooperation on exams, homework, and projects
- Definitely F

Attendance policy

Do not be late!

- You should be present when I take class attendance

You can miss the class up to "three" times without any penalty

- More than 3 times, then D

Reserved seat in class

- We will use reserved seat in this class for this semester
- Your seat will be reserved at the next class

Class policy

Notebook or tablet

- Only for study
- No smartphone

Food, fluid intake

Consecutive lecture (2-hour time)

Before class

What is the computer?

How does a computer work?

How can you build CPU and memory?

Computer architecture

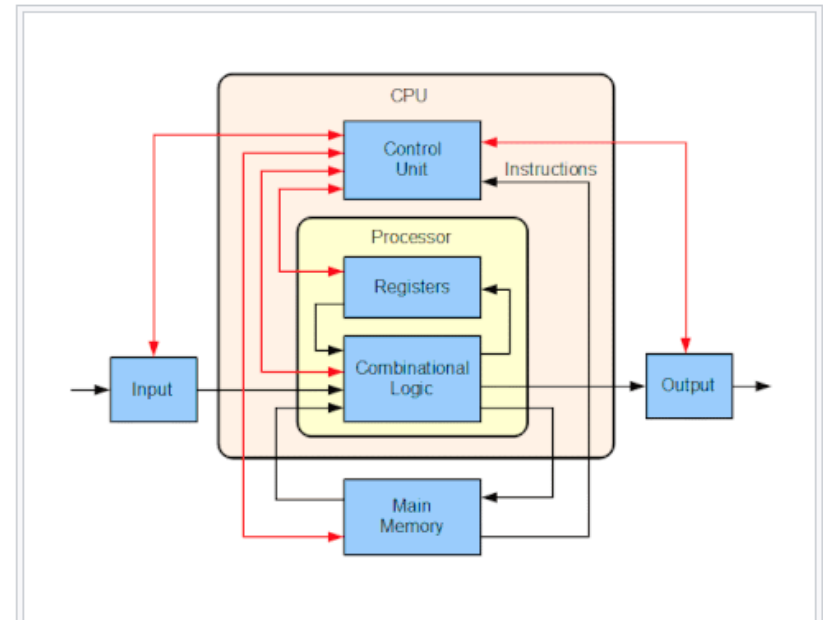
Computer architecture

From Wikipedia, the free encyclopedia

In **computer engineering**, **computer architecture** is a set of rules and methods that describe the functionality, organization, and implementation of computer systems. Some definitions of architecture define it as describing the capabilities and programming model of a computer but not a particular implementation.^[1] In other definitions computer architecture involves **instruction set architecture** design, **microarchitecture** design, **logic design**, and **implementation**.^[2]

Contents [hide]

- History
- Subcategories
- Roles
 - 3.1 Definition
 - 3.2 Instruction set architecture
 - 3.3 Computer organization
 - 3.4 Implementation
- Design goals
 - 4.1 Performance
 - 4.2 Power efficiency
 - 4.3 Shifts in market demand
- See also
- References
- Sources
- External links



Block diagram of a basic computer with uniprocessor CPU. Black lines indicate data flow, whereas red lines indicate control flow. Arrows indicate the direction of flow.

History [edit]

The first documented computer architecture was in the correspondence between **Charles Babbage** and **Ada Lovelace**, describing the **analytical engine**. When building the computer **Z1** in 1936, **Konrad Zuse** described in two patent applications for his future projects that machine instructions could be stored in the same storage used for data, i.e., the **stored-program** concept.^{[3][4]} Two other early and important examples are:

- **John von Neumann's** 1945 paper, **First Draft of a Report on the EDVAC**, which described an organization of logical elements;^[5] and