Introduction to Memory - Part#2

Yoonjin Kim

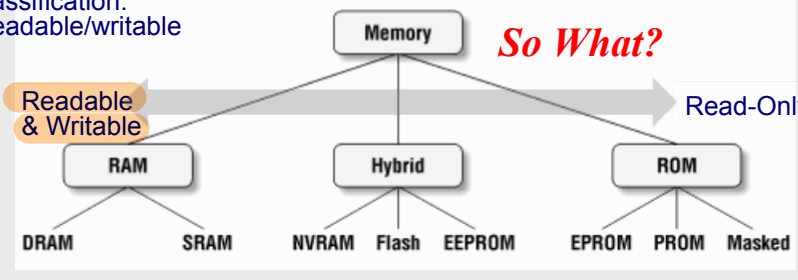
Full Professor

Division of Computer Science
Sookmyung Women's University

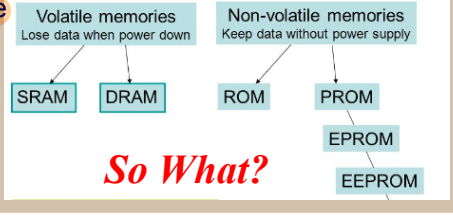
Memory Classification (메모리 분류)

Unsuitable Memory Classification for Computer Science/Engineering Major

• Criterion of classification: Readable/writable



• Criterion of classification: Volatile/nonvolatile



Why Most of Memory Classification Unsuitable for CSE ?

Because they do not show the relationship between processor and memory.

It's necessary to consider following 5 factors (based on two criteria) for the suitable classification.

- [Criterion#1] Is it feasible for main/local memory directly coupled with processor?
  - [Factor#1] Byte-level read & write operation: readable & writable by 1-byte
  - [Factor#2] Fast read & write operation: enough to communicate with processor
- [Criterion#2] Is it feasible for secondary memory?: if it does not qualify for main/local memory
  - [Factor#3] Nonvolatile: saving bits without electric power
  - [Factor#4] Both readable & writable
  - [Factor#5] Large capacity

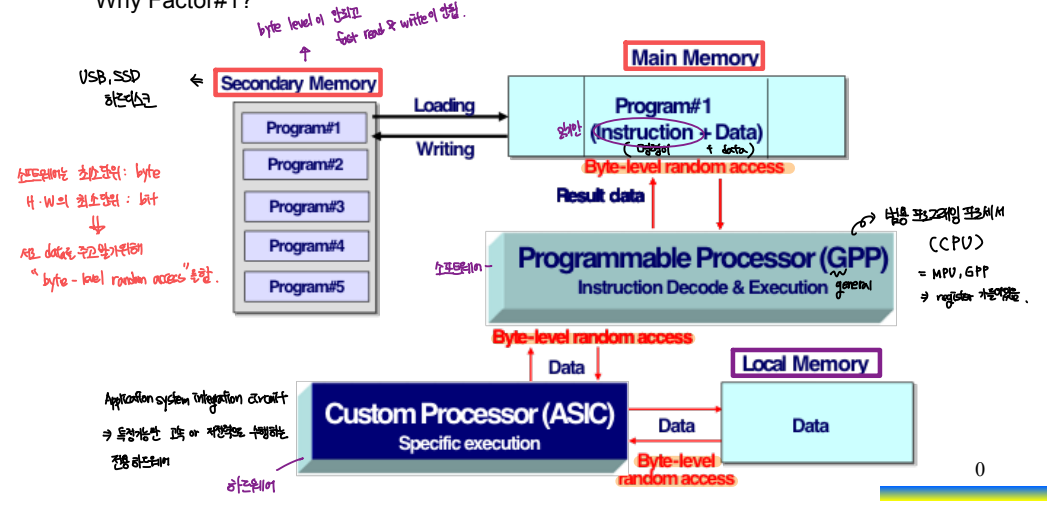
Suitable Memory Classification for CS/E Major

[Criterion#1] Is it feasible for main/local memory directly coupled with processor?

[Factor#1] Byte-level read & write operation: readable & writable by 1-byte

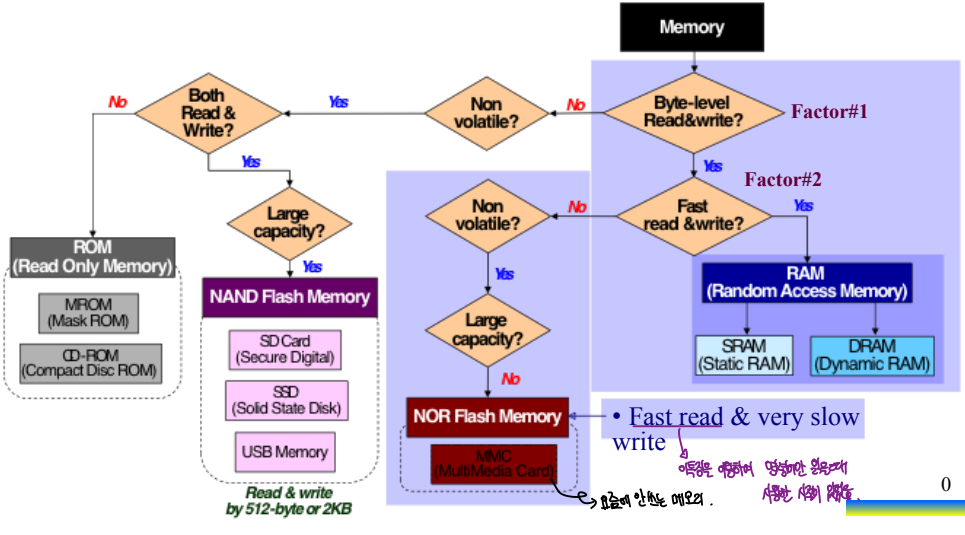
[Factor#2] Fast read & write operation: enough to communicate with processor

Why Factor#1?



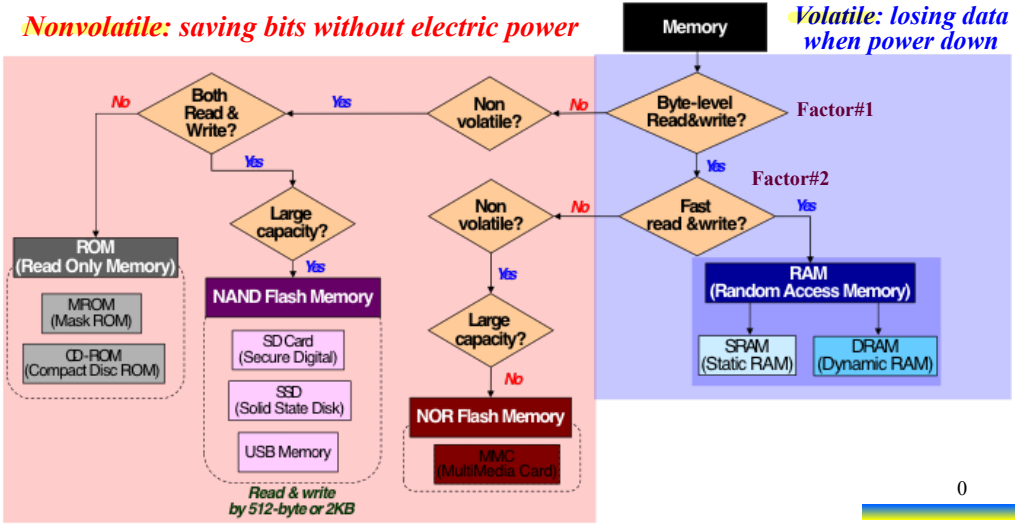
Suitable Memory Classification for CS/E Major

- [Criterion#1] Is it feasible for main/local memory directly coupled with processor?
  - [Factor#1] Byte-level read & write operation: readable & writable by 1-byte
  - [Factor#2] Fast read & write operation: enough to communicate with processor



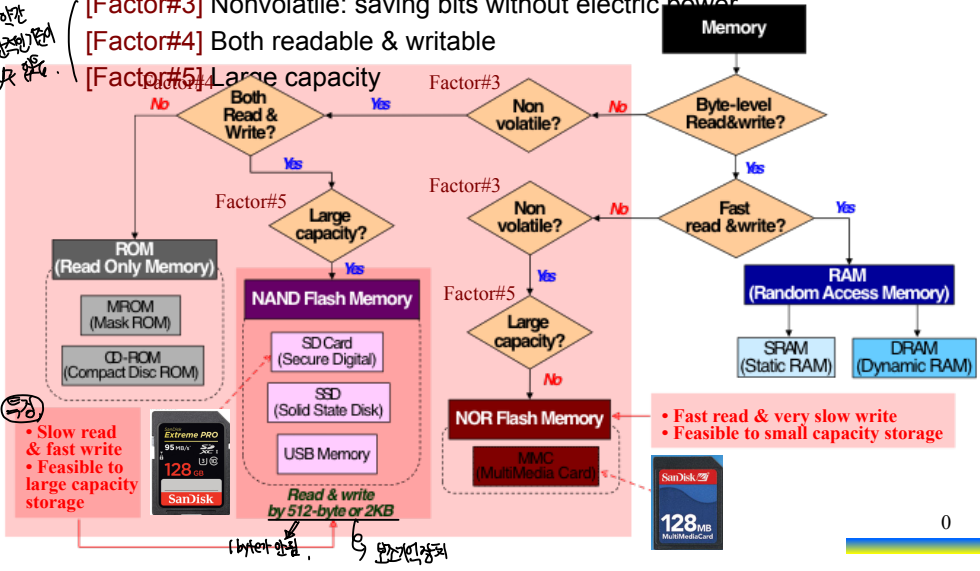
Suitable Memory Classification for CS/E Major

- [Criterion#1] Is it feasible for main/local memory directly coupled with processor?
  - Characteristics of RAM: Volatile
  - 'Volatile' is not a factor for 'Criterion#1' but only a characteristic of RAM.



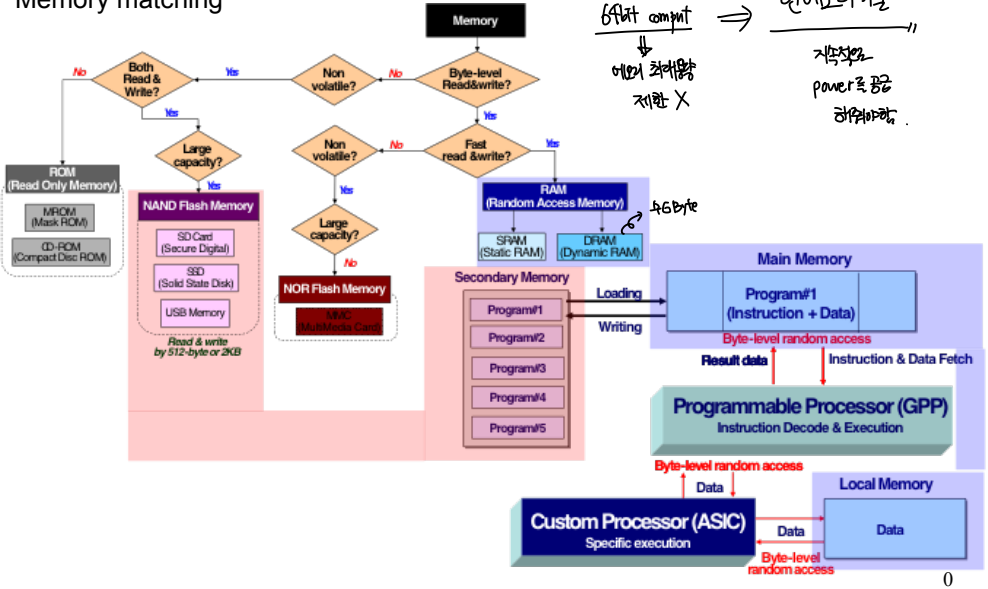
Suitable Memory Classification for CS/E Major

- [Criterian#2] Is it feasible for secondary memory?: if it does not qualify for main/local memory
  - [Factor#3] Nonvolatile: saving bits without electric power
  - [Factor#4] Both readable & writable
  - [Factor#5] Large capacity



Suitable Memory Classification for CS/E Major

Memory matching



Random Access Memory (RAM)

↳ address를 랜덤하게 접근

random address + R/W 동시지원

RAM (Random Access Memory)

Meaning of random access

↳ address를 random 하게 취해 접근하고 쓰기도 함.

means reading/writing data from/to random address immediately.
created several decades ago to contrast with sequentially-accessed storage like tape drives. → 이전에는 순차적으로 저장장치에 접근했음.

The term 'Random' caused an misunderstanding

It means both readable & writable because RAM was the only one to support both read & write operation as well as the only one to support 'random access' at that time.

Today, Strange name

↳ 모든메모리 대부분이 메모리도 random access 기능을 제공함.

Nowadays, most of memories support random access.

Therefore, today, it means the random access supporting byte-level and fast read&write that are key factors of main memory/local memory.

(교수님께서 정의한)

random access ⇒ byte level로 빠르게 read & write가 가능한 random access이다.

Random Access Memory (RAM)

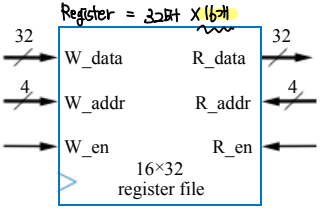
RAM (Random Access Memory)

Logically same as register file – Memory with address inputs, data inputs/outputs, and control.

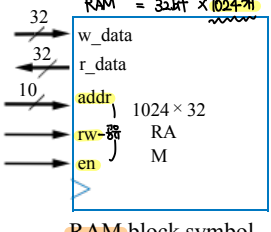
- RAM usually just one address/en port
- Register file usually two address/en ports to support read and write operation together.

RAM vs. register file

- RAM typically larger than *roughly* 512 or 1024 words
- RAM typically stores bits using a bit storage approach that is more efficient than a flip flop.

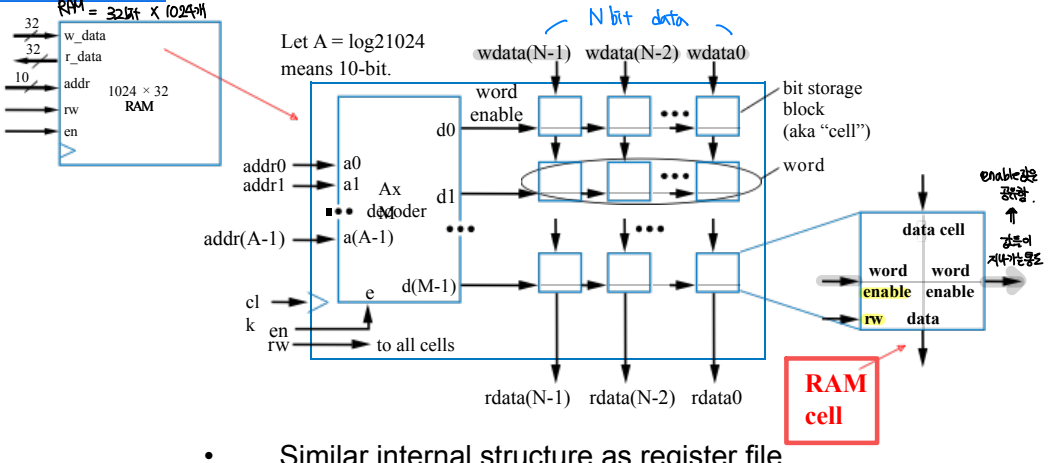


Register file from Ch. 4



RAM block symbol

RAM Internal Structure



- Similar internal structure as register file
  - Decoder enables appropriate word based on address inputs.
  - rw controls whether cell is written or read.
  - Let's see what's inside each RAM cell.

집적도 SRAM > DRAM

Static RAM (SRAM)

⊕ inverter에 2개 MOS필요.

참고 SRAM은 구성하는 트랜지스터의 개수가 적다. (register (D-flipflop) 에서만 씀)

"Static" RAM cell

SRAM cell = 6 transistors (recall inverter is 2 transistors)
whereas a D flip-flop is composed of 38 transistors.
2 x D latch (18 transistors) + inverter (2 transistors)
= 38 transistors
크기가 1/6 차이

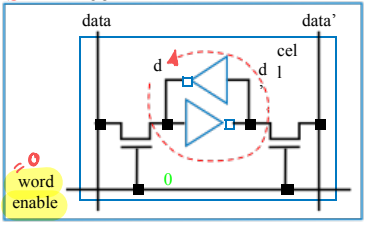
Writing this cell

word enable input comes from decoder
When 0, value *d* loops around inverters
That loop is where a bit stays stored.
When 1, the *data* bit value enters the loop
data is the bit to be stored in this cell.
data' enters on other side
Example shows a "1" being written into cell

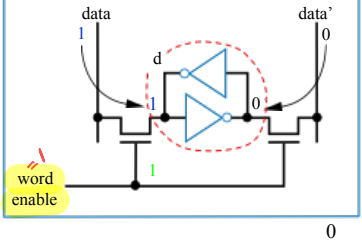
Reading this cell

Somewhat trickier.
The electrical description of SRAM is necessary to explain read-operation – it's really beyond our scope.

SRAM cell



SRAM cell → write



Static의 의미는 무엇인가?

⇒ DRAM의 Dynamic을 알아야 알 수 있음.

Asynchronous
Synchronous

개념, 차이점 공부하기

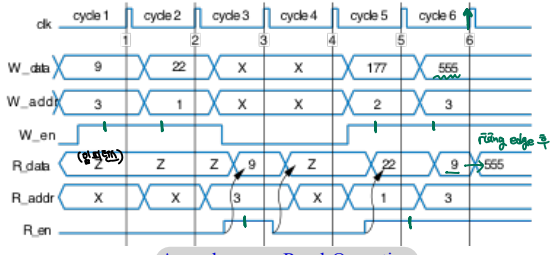
↳ 캐시 메모리에 사용됨. ⇒ 예를들어 사용하냐에 따라 가격이 상당히 결정됨.

SRAM Timing Diagram

⇒ Register는 clk 보다 w_enable , r_enable 의 영향을받음

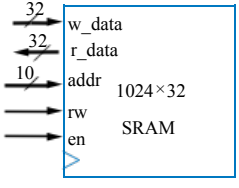
Comparison between Register File and SRAM

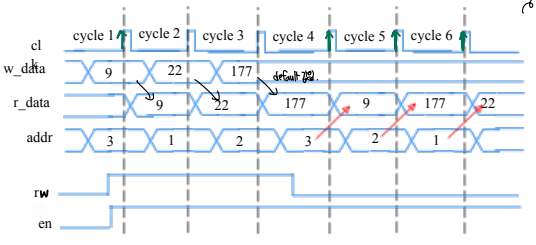
Register



Asynchronous Read-Operation

SRAM





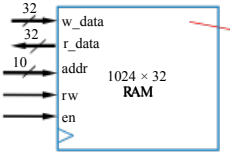
메모리는 구현 때 library로 부터 읽어 구현하므로 Z 표시하지 않았음. 참조.

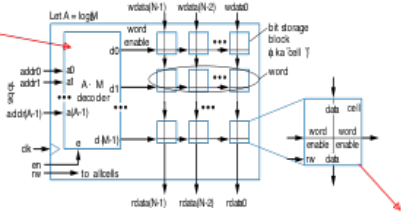
rw 0 en 1
⇒ read mode
rw 1 en 1
⇒ write mode.

Synchronous Read-Operation

Dynamic RAM (DRAM)

↳ SRAM Cell과 차이점 보기





캐패시턴에 값을 저장하고 있음.

"Dynamic" RAM cell

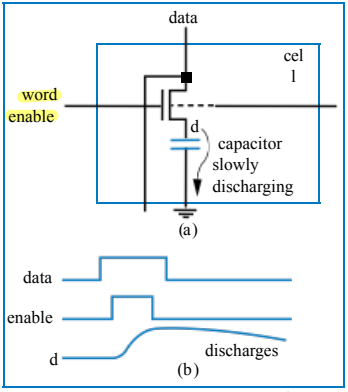
SRAM의 ⇒ 1/6이됨.

- 1 transistor (rather than 6)
- Relies on *large* capacitor to store bit.
  - Write: When word enable is 1, data voltage level gets stored on top plate of capacitor.
  - Read: Just look at value of *d*

- Problem: Capacitor discharges over time.
  - Must "refresh" regularly, by reading *d* and then writing it right back. ⇒ 주기적으로 recharge를 계속 해줘야 함.
  - This is the reason why its name includes 'dynamic'.

↳ 값이 사라지는 것을 방지하기 위해 주기적으로 refresh 시켜줌 (=recharge)

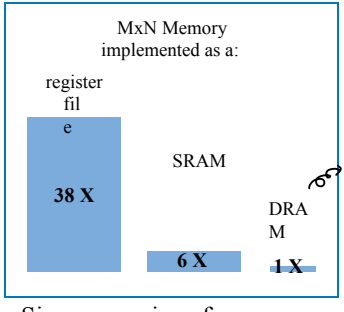
DRAM cell



Static은 dynamic 반대말이므로 ⇒ recharge 해줄 필요 X

Comparing Memory Types

- Register file – 38 transistors for 1-bit
  - Fastest
  - But biggest size
  - Very expensive
- SRAM – 6 transistors for 1-bit
  - Fast
  - More compact than register file
  - Expensive
- DRAM – 1 transistor for 1-bit
  - Slowest
    - And refreshing takes time
  - But very compact
  - Cheap
- Use register file for small items, SRAM for large items, and DRAM for huge items
  - Note: DRAM's big capacitor requires a special chip design process, so DRAM is often a separate chip.



Size comparison for same number of bits