

22강_데이터 저장장치 (Memory+SSD)

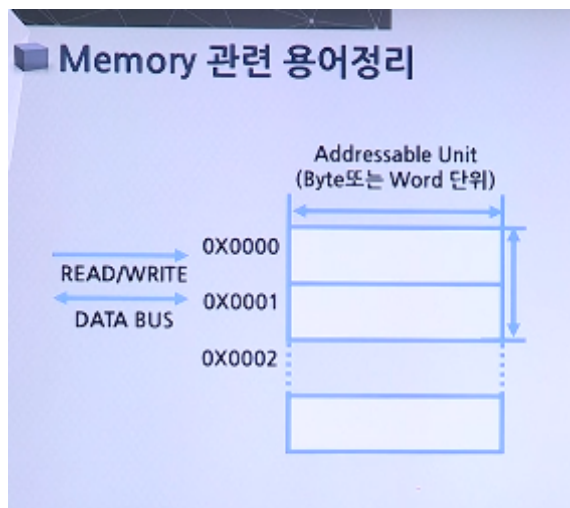
🕒 Created	@Aug 13, 2020 12:04 PM
🏷 Tags	

학습목표

» 기억장치의 성능과 비용을 고려한 계층적 구조에서부터 반도체 메모리의 기본 구조 및 모듈설계에 관한 내용을 설명할 수 있다.

학습내용

» 데이터 저장장치(Memory+SSD)
» Semiconductor Memory
» Main Memory Module Design
» Quiz, PBL, 탐구주제

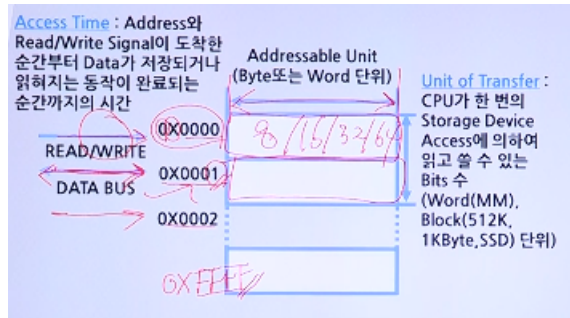


0X : 16진수

폭 : addressable unit

Word : 컴퓨터가 한 번에 처리할 수 있는 Bit의 길이. 따라서 컴퓨터 기종에 따라 Word는 그 크기가 다 달라질 수 있음

SSD는 블록단위로 데이터가 전송된다
SSD의 Unit of Transfer은 512k 한 블록

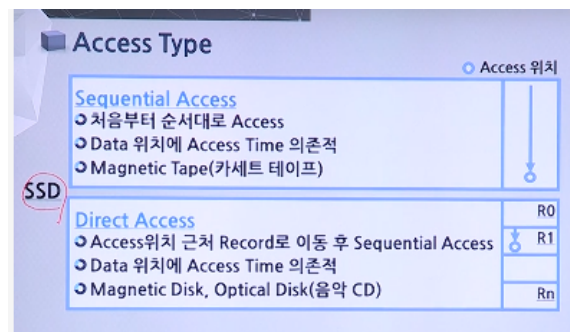


이 512KBYTE 나 1KBYTE 등 상당히 큰 양이 한 번에 이동함

Memory Capacity = $2^{(\text{No. of Address Bits})} \times \text{Addressable Unit}$
Data Transfer Rate = $(1/\text{Access 시간} = \text{초당 Access되는 횟수}) \times (\text{한 번에 읽혀지는 Data Bits 수})$
 [Ex] Access Time=100ns이고 Access 단위=32Bits인 경우
 Data 전송률=(1/100ns)x32=320Mbits/sec

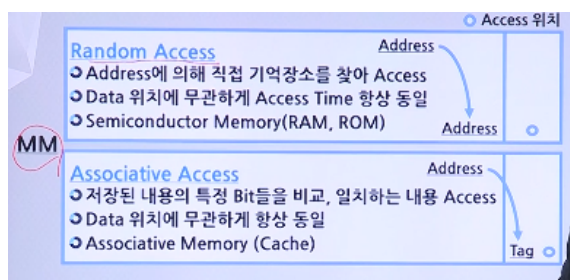
Data Transfer Rate

Data 전송률 : 1초에 몇 Bit를 이동시킬 수 있는가? : $1/\text{Access 시간}$

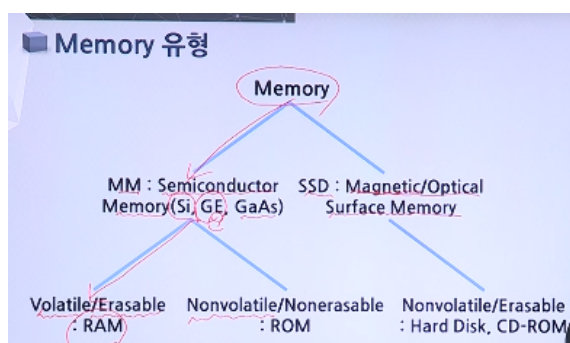


Sequential Access - Data 위치

Access Time 의존적 : 어느 위치에 데이터가 있냐에 따라서 접속시간이 다 달라짐



메인 메모리

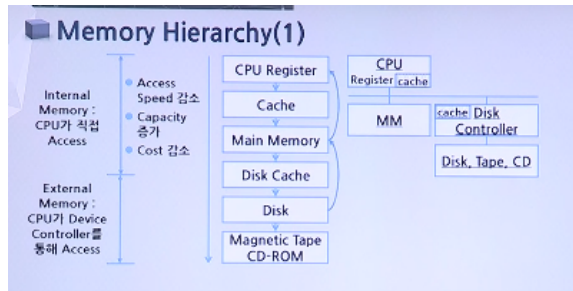


Semiconductor 반도체를 이용해서 메모리를 만들게됨. 대표적인 디바이스가 RAM.

Main Memory는 RAM이라고 생각하면 됨

SSD 보조기억장치. 마그네틱 디스크 혹은 옵티컬 디스크~

RAM Volatile 휘발성(전원차단되면 내용물 소멸) / ROM Nonvolatile 비휘발성 Nonerasable 지웠다 다시 쓸수 없음 최근에 지웠다 다시 쓸 수 있는 ROM들 ex) USB 메모리



디스크보다 캐시가 비쌈...저장 비트당 가격 캐시가 훨씬 빠름 == 훨씬 빠르다 하드디스크는 반도체를 이용한 저장장치가 아니라 속도가 굉장히 느리고 저렴한 레지스터가 여기서 가장 빠름 CD-ROM 이렇게 제일 느리고 싼

캐시가 어디에 있느냐가 문제 : 보통 디스크 안에 있음. 컨트롤러에 붙어있게끔 만들 수도 있다

Principle of Locality

- Memory의 Access가 몇몇 특정 영역에 집중되는 현상
- 짧은 시간을 기준으로 보면 Processor가 Memory의 한정된 위치들만을 Access하면서 작업을 수행
- 이 원리가 적용되는 데이터에 대해 Hierarchical Memory를 사용할 경우 최저 비용으로 최고 성능을 얻을 수 있음

Memory Hierarchy(2)

Hierarchical Memory System: Memory의 성능 대 가격비(Performance/Cost Ratio)를 향상시키기 위하여 한 System 내에 다양한 종류의 Memory들을 사용하는 방식

- CPU에 가까운 Memory : 속도가 빠르나, 가격은 높은 Memory 사용
- CPU에서 먼 Memory : 속도는 느리지만, 가격은 낮은 Memory 사용

Memory Hierarchy(3)

Hierarchical Memory System의 효과: Average Access Time이 빨라짐

- Main Memory의 Access 시간 = 50ns
- Secondary Storage Device의 Access 시간 = 500ns
- Access할 정보가 Main Memory에 있을 확률 = 50%
- Average Memory Access Time = $(0.5 \times 50ns) + (0.5 \times 500ns) = 275ns$
- Data가 Main Memory에 있는 비율에 따른 Average Access Time의 변화