

기억장치

학습목표

- 기억 장치의 의미와 읽기·쓰기 과정과 기억 장치 관련 용어·특징을 이해할 수 있다.
- RAM과 ROM의 특성을 이해하고 기억 장치 모듈을 설계할 수 있다.
- 캐시 기억 장치의 세 가지 매핑 방법과 교체 알고리즘을 이해하고 설명할 수 있다.
- 가상 기억 장치의 필요성과 매핑 방법을 이해하고 설명할 수 있다.
- 연관 기억 장치의 동작 원리를 이해하고 설명할 수 있다.
- SDRAM, DDR SDRAM, 플래시 메모리의 동작 원리를 이해하고 설명할 수 있다.

내용

- 01 기억 장치 시스템의 개요
- 02 주기억 장치
- 03 캐시 기억 장치
- 04 가상 기억 장치
- 05 연관 기억 장치
- 06 최신 기억 장치 기술

01 기억 장치 시스템의 개요

1 기억 장치의 종류와 특성

표 6-1 컴퓨터 기억 장치 시스템의 주요 특성 분류

기준	설명
위치	CPU, 컴퓨터의 내부와 외부
용량	워드 크기, 워드 개수
전송 단위	워드, 블록
성능	액세스 시간, 사이클 시간, 전송률
액세스 방법	순차, 직접, 임의, 연관
물리적인 유형	반도체, 자기, 광, 자기-광
물리적인 특성	휘발성과 비휘발성, 파괴적과 비파괴적

□ 위치에 따른 분류

- 위치는 컴퓨터 내부와 외부를 구분하는 기준이다.
- CPU 내부의 레지스터와 주기억 장치는 내부 기억 장치고, 자기 디스크와 자기 테이프는 외부 기억 장치다.

01 기억 장치 시스템의 개요

□ 용량에 따른 분류

- 용량(capacity)은 기억 장치가 저장할 수 있는 데이터의 총량으로 바이트나 워드로 나타낸다.
- 워드(word)는 시스템에 따라 8, 16, 32, 64비트로 길이가 다양한데 이는 CPU가 처리하는 명령어 길이나 내부에서 한 번에 연산할 수 있는 데이터 비트 수와 같다.
- 일반적으로 외부 기억 장치는 용량을 바이트로 표시한다.

□ 전송 단위에 따른 분류

- 내부 기억 장치에서 전송 단위는 기억 장치로 들어가고 나오는 데이터선의 수로, 워드 길이와 같거나 다를 수도 있다.
- 외부 기억 장치에서 전송 단위는 워드보다 큰 **블록**이다. 예를 들어 외부 기억 장치인 하드 디스크는 블록 크기가 512바이트 또는 1024바이트다.
- 외부 기억 장치에 연결되는 데이터 버스의 폭은 8비트, 16비트, 32비트이므로 블록 하나를 전송하려면 전송 동작이 여러 번 연속으로 이루어져야 한다.

01 기억 장치 시스템의 개요

□ 성능에 따른 분류

- **액세스 시간**(access time) : 주소나 제어(읽기/쓰기) 신호가 기억 장치에 도착한 순간부터 데이터가 저장되거나 읽히는 순간까지이다.
- **사이클 시간**(cycle time) : 액세스 시간과 다음 액세스를 시작하기 위해 필요한 동작에 걸리는 시간을 더한 것이다.
- **전송률**(transfer rate) : 전송률은 데이터가 기억 장치로 들어가거나 나오는 초당 비트 수로, 기억 장치의 전송 속도를 측정하는 기준이다. 대역폭(bandwidth)이라고도 한다. 데이터 버스의 폭이 크면 더 많은 비트가 한 번에 전송되므로 전송률이 높아진다.

01 기억 장치 시스템의 개요

□ 액세스 방법에 따른 분류

- **순차 액세스**(sequential access) : 임의 위치의 데이터를 처음부터 순서대로 읽으므로 데이터의 위치에 따라 액세스 시간이 크게 달라진다. (자기 테이프, 자기 드럼)
- **직접 액세스**(direct access) : 기억 장치의 각 FAT(또는 레코드, 블록) 근처로 먼저 이동한 위치부터 순서대로 읽으므로 데이터의 저장 위치에 따라 액세스 시간이 달라진다. (자기 디스크, CD-ROM, DVD)
- **임의 액세스**(random access) : 기억 장치의 장소마다 고유의 주소가 있어 어떤 위치든 임의로 액세스할 수 있다. 데이터의 위치에 상관없이 액세스 시간이 같다. (주 기억 장치를 구성하는 반도체 기억 장치)
- **연관 액세스**(associative access) : 임의 액세스 방식의 일종으로, 주소 대신 내용의 일부를 이용한다. 하드웨어로 구현한다. (캐시 기억 장치)

01 기억 장치 시스템의 개요

□ 물리적 유형에 따른 분류

- 기억 장치를 제작에 사용하는 물리적인 재료에 의해 분류할 수 있다. 반도체 기억 장치, 자기-표면 기억 장치, 광 저장 장치 등이 있다.

□ 물리적 특성에 따른 분류

- 전원이 끊기면 데이터의 소멸 여부에 따라 **휘발성 기억 장치**와 **비휘발성 기억 장치**로 분류
- RAM은 휘발성 기억 장치고, ROM, 플래시 메모리, 자기-표면 기억 장치, 광 저장 장치 등은 비휘발성 기억 장치다.
- 데이터를 읽으면 데이터 파괴 여부에 따라 **파괴적 기억 장치**와 **비파괴적 기억 장치**로 분류
- 자기 코어나 FRAM은 파괴적 기억 장치고, 반도체 기억 장치, 자기-표면 기억 장치, 광 저장 등은 비파괴적 기억 장치다.

01 기억 장치 시스템의 개요

2 계층적 기억 장치 시스템

❖ 기억 장치 시스템에서 액세스 속도, 가격, 용량의 관계

- 액세스 속도가 빨라질수록 비트당 가격은 높아진다.
- 용량이 증가할수록 비트당 가격은 감소하고, 액세스 속도도 낮아진다.

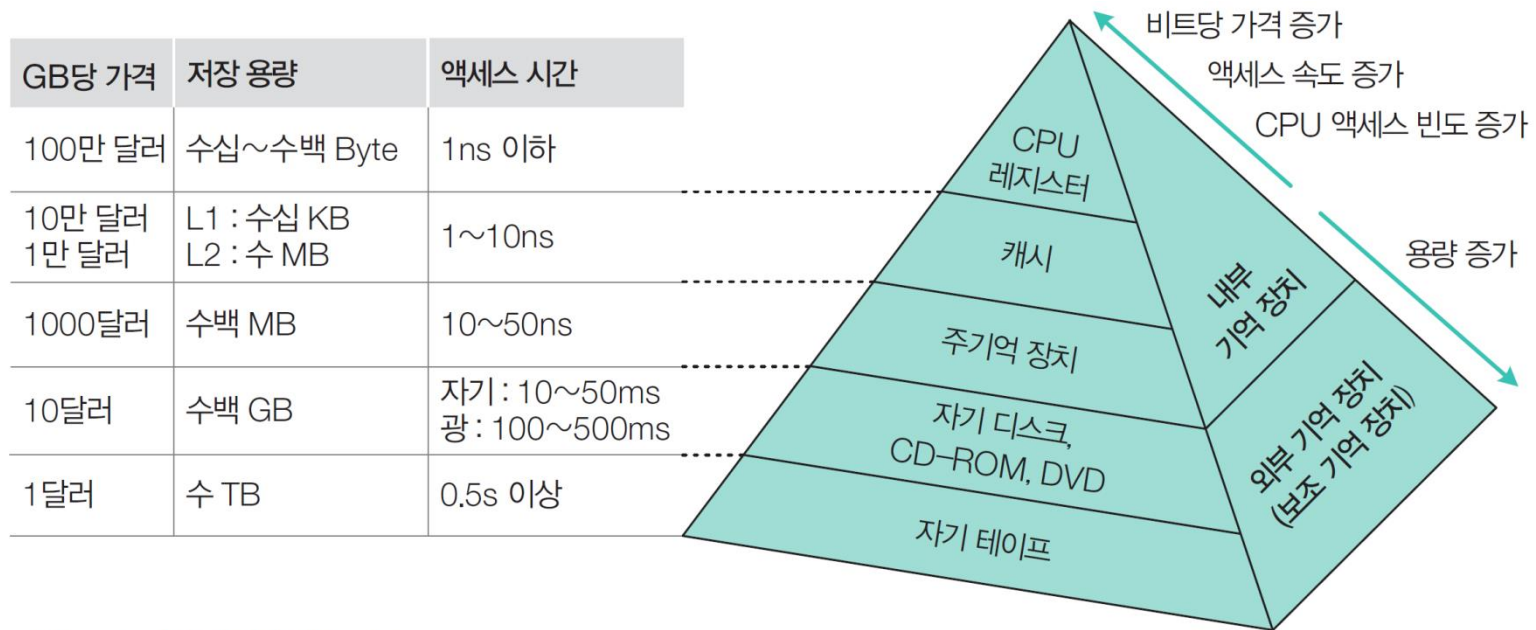


그림 6-1 기억 장치의 계층 구조

평균 액세스 속도를 빠르게 하면서
가격 대비 성능도 적절히 유지하는 방법이 계층적 구조다.

01 기억 장치 시스템의 개요

❖ 기억 장치의 계층 구조

❶ 레지스터	<ul style="list-style-type: none">• 액세스 속도가 가장 빠르지만 비트당 가격도 가장 높아 CPU에 소량 존재한다.• RISC 계열의 CPU는 레지스터가 200개 이상이지만 보통은 수십 개 정도다.
❷ 캐시	<ul style="list-style-type: none">• 캐시는 주기억 장치와 CPU의 속도 차를 줄이기 위해 레지스터와 주기억 장치 사이에서 CPU가 자주 사용하는 명령어나 데이터를 일시 저장하는 버퍼 역할을 한다.• 캐시는 SRAM으로 구성되는데, CPU 내부에 있으면 L1 캐시, 외부에 있으면 L2 캐시라고 하는데 요즘은 L2 캐시도 CPU 내부에 집적한다.
❸ 주기억 장치	<ul style="list-style-type: none">• DRAM으로 구성했다가 요즘은 SDR SDRAM이나 DDR SDRAM으로 구성한다.• 프로그램을 수행하려면 먼저 해당 프로그램의 명령어와 데이터를 주기억 장치에 적재해야 한다.• 내부 기억 장치는 고속 동작을 위해 보통 하나의 CPU 보드상에 위치한다.
❹ 외부 기억 장치	<ul style="list-style-type: none">• 대규모 데이터를 영구 저장하기 위해 사용된다. CPU가 직접 액세스할 수 없고 제어기를 통해 액세스할 수 있다.• 보조 기억 장치라고도 한다.• 고속 액세스가 가능한 것은 자기 디스크이며, 요즘에는 플래시 메모리도 보편적으로 사용된다

1 주기억 장치의 동작

- CPU와 주기억 장치 사이의 데이터 전송은 CPU 내부에 있는 레지스터 2개(MAR, MBR)와 제어 신호 3개(읽기, 쓰기, 칩 선택)를 통해 이루어진다.
- 메모리 주소 레지스터(MAR) : 메모리 액세스 시 특정 워드의 주소가 MAR에 전송
- 메모리 버퍼 레지스터(MBR) : 레지스터와 외부 장치 사이에서 전송되는 데이터의 통로

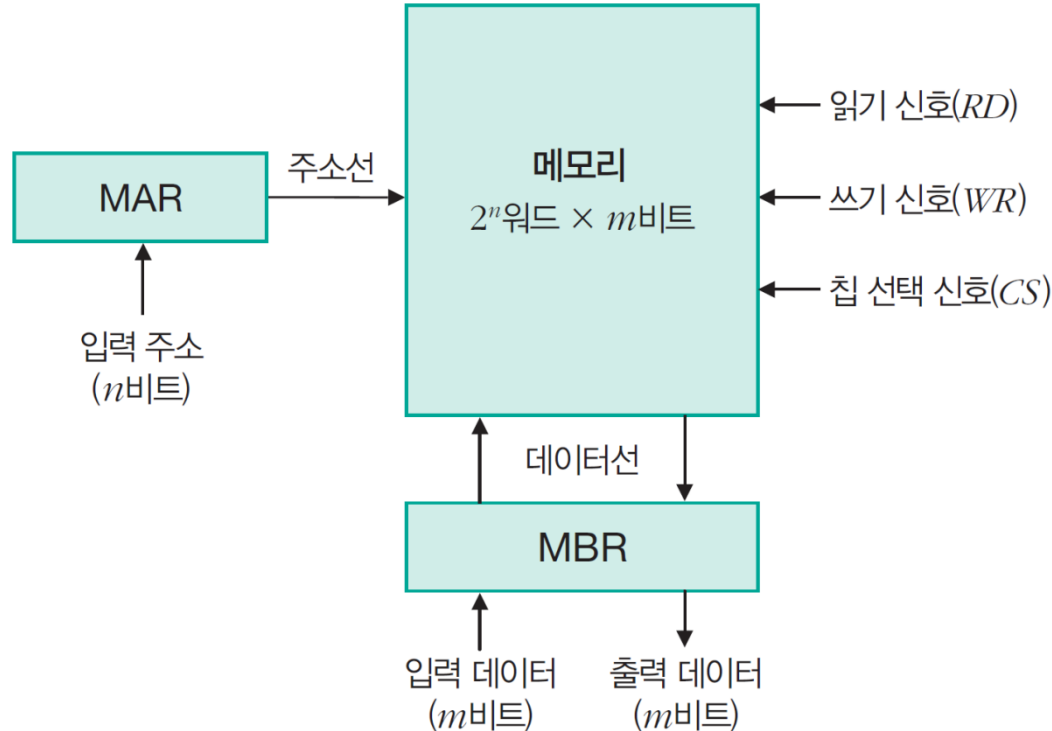


그림 6-2 주기억 장치의 동작 블록도

□ 메모리 읽기 동작

- ❶ 읽으려는 메모리의 주소를 MAR로 전송한다.
- ❷ 칩 선택 신호와 읽기 신호를 활성화시키면 지정된 메모리의 워드가 MBR로 들어온다.

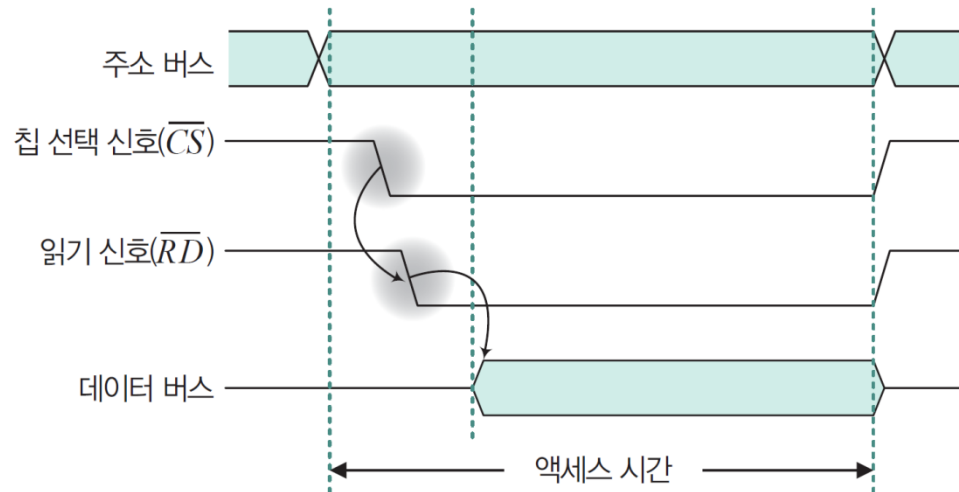


그림 6-3 메모리 읽기 동작의 타이밍도

□ 메모리 쓰기 동작

- ❶ 지정된 메모리의 주소를 MAR로 전송하는 동시에 저장하려는 데이터의 워드를 MBR에 전송한다.
- ❷ 칩 선택 신호와 쓰기 신호를 활성화시킨다.

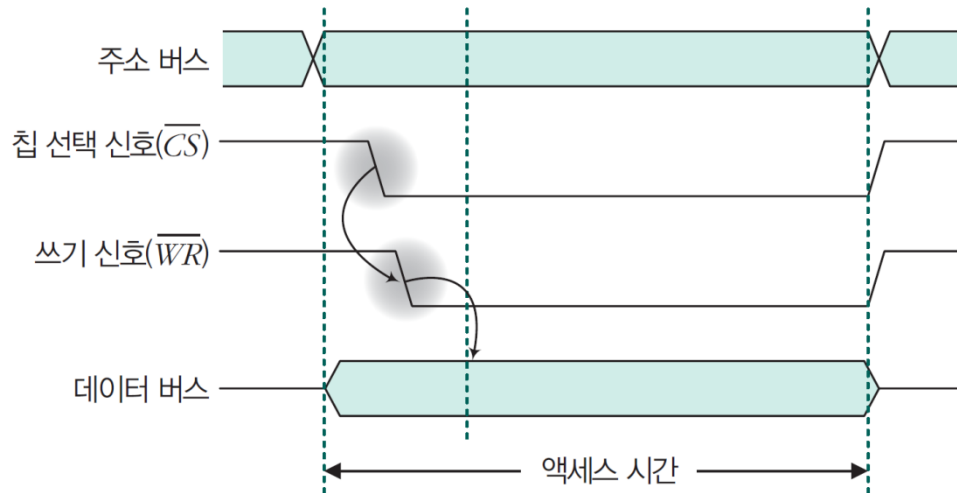


그림 6-4 메모리 쓰기 동작의 타이밍도

02 주기억 장치

□ 기억 장치의 용량 표현

- 기억 장치의 용량은 주소 버스의 길이와 지정된 주소에 들어 있는 데이터의 길이로 나타낸다.
- 주소 버스의 길이가 n 비트고, 워드당 비트 수가 m 일 때 용량은 다음과 같다.

$$\text{용량} = 2^n \times m$$

- MAR의 비트 수는 주소 버스의 길이 n 과 같으며, 워드 개수는 2^n 이다.
- MBR의 비트 수는 워드당 비트 수이므로 데이터 버스의 길이 m 과 같다.

예제 6-1 기억 장치의 용량이 1024×8 이라면 MAR과 MBR은 각각 몇 비트인가?

풀이

$1024 \times 8 = 2^{10} \times 8$ 이므로 MAR=10비트, MBR=8비트이다.

참고로 MAR = address line = address bus, MBR = data line = data bus = word의 길이이다.

MAR = address line

$2^{10} \times 8$

MBR = data line = word의 길이

End of Example

02 주기억 장치

□ 워드의 저장 방법

- 기억 장치에 저장되는 데이터를 구별하려면 주소와 데이터 단위를 정의해야 한다.
- 주소에는 0번지부터 고유의 일련번호를 부여한다.
- 각 주소에 데이터가 1바이트나 워드 단위로 저장되므로 바이트 주소와 워드 주소로 분류할 수 있다.
- 기억 장치에 바이트를 배열하는 방법을 **엔디안**(endian)이라고 한다.



(a) 바이트 주소



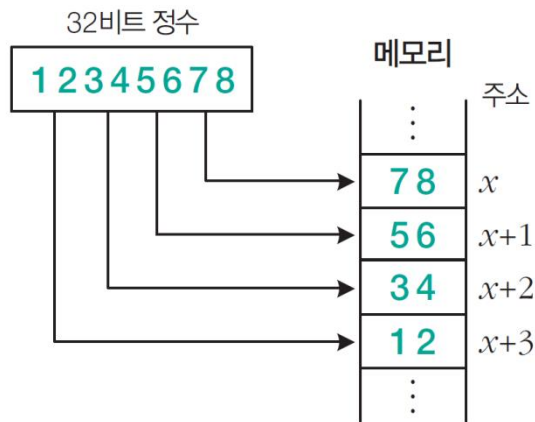
(b) 워드 주소

그림 6-5 바이트 주소와 워드 주소

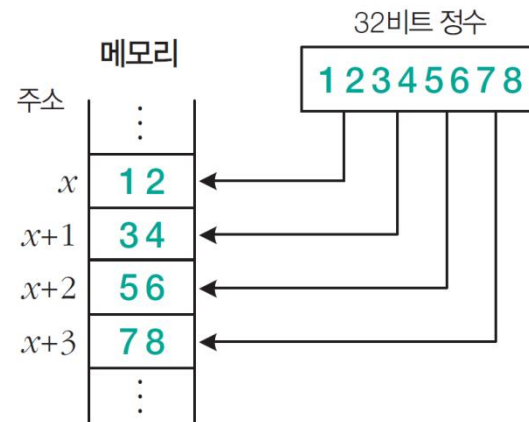
02 주기억 장치

❖ 리틀 엔디안(little endian)

- 하위 바이트를 낮은 주소에 저장하는 방법이다. 오른쪽에서 왼쪽으로 저장한다.
- 이는 산술 연산이 주소가 낮은 쪽에서 높은 쪽으로 처리되는 순서와 같다.
- 홀수와 짝수를 검사할 때도 첫 바이트만 확인하면 되므로 빠르다.
- 리눅스, 인텔 계열의 CPU, AMD 계열의 CPU에서 사용하는 방식이다.



(a) 리틀 엔디안



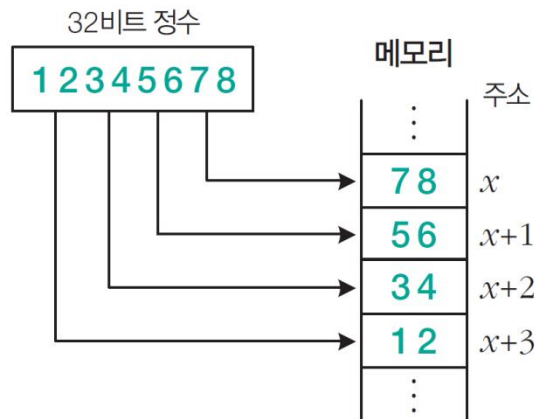
(b) 빅 엔디안

그림 6-6 워드의 저장 방법(바이트 주소인 경우)

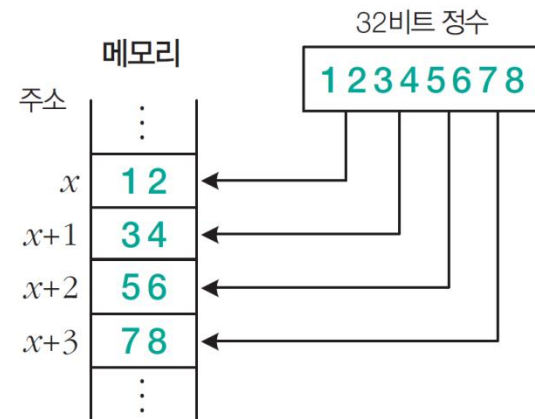
02 주기억 장치

❖ 빅 엔디안(big endian)

- 상위 바이트를 낮은 주소에 저장하는 방법으로 왼쪽에서 오른쪽으로 저장한다.
- 숫자를 읽고 쓰는 일반적인 방식과 같아 사람이 읽기 편하다.
- IBM이나 모토로라의 CPU가 사용하는 방식이다. 또 TCP/IP 전송도 이 방식을 사용한다.



(a) 리틀 엔디안



(b) 빅 엔디안

그림 6-6 워드의 저장 방법(바이트 주소인 경우)

❖ 바이 엔디안(biendian)

- 엔디안을 선택할 수 있도록 설계된 방법으로 ARM, PowerPC, DEC Alpha, MIPS 등이 있다.



Thank You
