

16강. 보조기억장치와 입출력장치 2) 입출력 기법

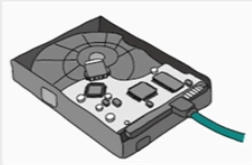
1.25

Chapter 02-5 보조기억장치와 입출력장치(13)

입출력 기법

장치 컨트롤러와 장치 드라이버

- 장치 컨트롤러(device controller)
 - CPU와 입출력장치 사이의 통신을 중개하는 중개자 역할의 하드웨어
 - 모든 입출력장치는 각자의 장치 컨트롤러를 통해 컴퓨터 내부와 연결되어 정보를 교환
- 장치 드라이버(device driver)
 - 장치 컨트롤러의 동작을 알고, 장치 컨트롤러가 컴퓨터 내부와 정보를 주고받을 수 있도록 하는 프로그램



Chapter 02-5 보조기억장치와 입출력장치(14)

프로그램 입출력

- 프로그램 입출력(Programmed I/O) - 프로그램 속 명령어로 입출력 작업을 수행하는 방법

주소
0
:
:
1023

주소 공간

메모리를 위한
주소 공간

주소
0
:
:
1023

주소 공간

입출력장치를 위한
주소 공간

주소
0
:
511
512
:
1023

주소 공간

메모리를 위한
주소 공간

입출력장치를 위한
주소 공간

고립형 입출력

메모리 맵 입출력

>> 이것이 취업을 위한 컴퓨터 과학이다

15

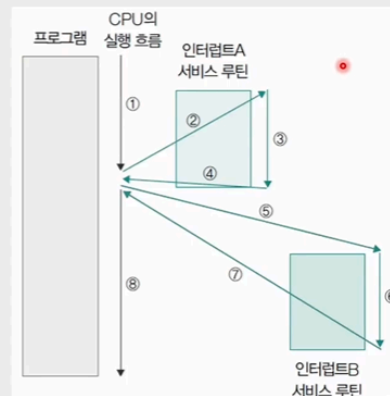
고립형은 별도로 분리하는 방식. 메모리 맵형은 별도로 분리하지 않는 방식

메모리에 접근하는 명령어랑 입출력장치에 접근하기 위한 명령어가 다름. 근데 메모리 맵형은 메모리에 접근하는 명령어를 쓰고, 입출력장치에 접근하기 위한 명령어가 필요 없음. 그냥 메모리에 접근하는 것과 똑같이 접근하면 되기 때문임.

Chapter 02-5 보조기억장치와 입출력장치(16)

인터럽트 기반 입출력: 다중 인터럽트

- 인터럽트가 여러 입출력장치로부터 동시다발적으로 발생하는 경우의 인터럽트 처리
 - 키보드, 마우스, 모니터, 스피커 등의 입출력장치를 동시에 사용하는 상황
 - CPU가 플래그 레지스터 속 인터럽트 비트를 비활성화한 채 인터럽트를 처리할 경우
 - 다른 하드웨어 인터럽트를 받아들이지 않기 때문에 CPU는 기본적으로 다음과 같이 인터럽트가 발생한 ①~⑧의 순서대로 인터럽트 서비스 루틴을 순차적으로 실행



>> 이것이 취업을 위한 컴퓨터 과학이다

17

입출력은 보통 동시다발적임.

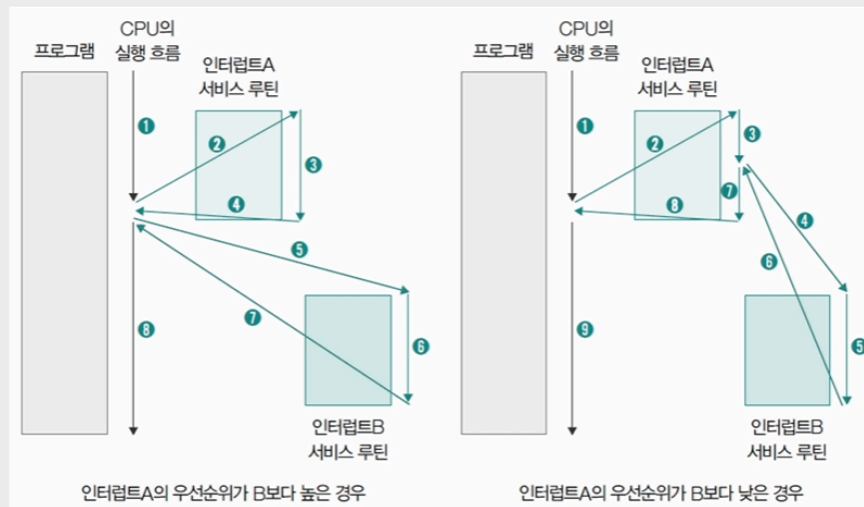
Chapter 02-5 보조기억장치와 입출력장치(17)

- 인터럽트 중에서도 우선순위가 더 높은 인터럽트가 우선적으로 처리되는 경우가 일반적
 - 예) 현재 CPU가 인터럽트A를 처리하는 도중에 또 다른 인터럽트B가 들어왔다고 가정
 - 지금 처리 중인 인터럽트A보다 인터럽트B의 우선순위가 낮다면, CPU는 A를 모두 처리한 뒤에 B를 처리
 - 인터럽트A보다 B의 우선순위가 높다면, CPU는 인터럽트A의 실행을 잠시 멈추고 B를 처리한 뒤에 다시 A를 처리
 - CPU는 플래그 레지스터 속 인터럽트 비트가 활성화되어 있는 경우, 혹은 인터럽트 비트를 비활성화해도 무시할 수 없는 인터럽트인 NMI(Non-Maskable Interrupt)가 발생한 경우, 우선순위가 높은 인터럽트부터 처리

>> 이것이 취업을 위한 컴퓨터 과학이다

18

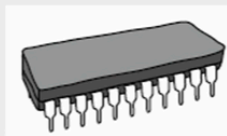
Chapter 02-5 보조기억장치와 입출력장치(18)



우선순위에 따름.

Chapter 02-5 보조기억장치와 입출력장치(19)

- 프로그래머블 인터럽트 컨트롤러(PIC, Programmable Interrupt Controller)
 - 다중 인터럽트를 처리하기 위해 사용되는 하드웨어
 - PIC는 여러 장치 컨트롤러에 연결되어 있어 장치 컨트롤러에서 보낸 하드웨어 인터럽트 요청들의 우선순위를 판별한 뒤, CPU에게 지금 처리해야 할 하드웨어 인터럽트가 무엇인지를 알려 주는 장치
- PIC 각각의 핀에는 CPU에게 하드웨어 인터럽트 요청을 보낼 수 있도록 약속된 하드웨어가 연결
 - 예) 첫 번째 핀은 타이머 인터럽트를 받아들이는 핀이고, 두 번째 핀은 키보드 인터럽트를 받아들이는 핀 등



[NOTE]
PIC가 무시할 수 없는 인터럽트인 NMI까지 우선순위를 판별하지는 않음

>> 이것이 취업을 위한 컴퓨터 과학이다

20

우선순위는 이 PIC를 통해 판별하게 됨. 저 그림에서 다리로부터 인터럽트 요청 신호를 받아들이게 됨. CPU에게 우선순위를 알려주게 됨.

Chapter 02-5 보조기억장치와 입출력장치(20)

- PIC는 많은 하드웨어 인터럽트를 관리하기 위해 다음과 같이 2개 이상의 계층으로 구성

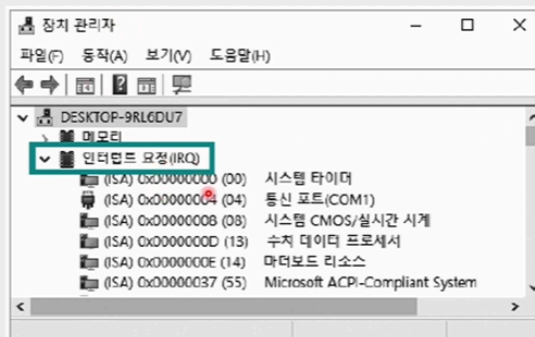


>> 이것이 취업을 위한 컴퓨터 과학이다

21

Chapter 02-5 보조기억장치와 입출력장치(21)

- PIC와 연결되어 하드웨어 인터럽트를 보낼 수 있는 장치들을 운영체제에서 조회
 - 윈도우의 경우 [장치 관리자]의 [인터럽트 요청(IRQ)]에서 확인



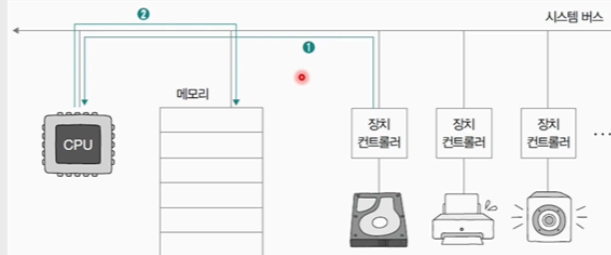
>> 이것이 취업을 위한 컴퓨터 과학이다

22

Chapter 02-5 보조기억장치와 입출력장치(22)

DMA 입출력

- 프로그램 기반의 입출력과 인터럽트 기반의 입출력에 공통점
 - CPU가 입출력장치와 메모리 간의 데이터 이동을 주도해야 하며, 이동하는 데이터들도 반드시 CPU를 거침
 - 예1) 입출력장치의 데이터를 메모리에 저장하는 경우
 - CPU는 ①장치 컨트롤러로부터 데이터를 하나씩 읽어 레지스터에 적재, ②적재한 데이터를 하나씩 메모리에 저장
 - 예2) 메모리 속 데이터를 입출력장치에 내보내는 경우
 - CPU는 ①메모리로부터 데이터를 하나씩 읽어 레지스터에 적재, ②적재한 데이터를 하나씩 입출력장치에 내보냄.



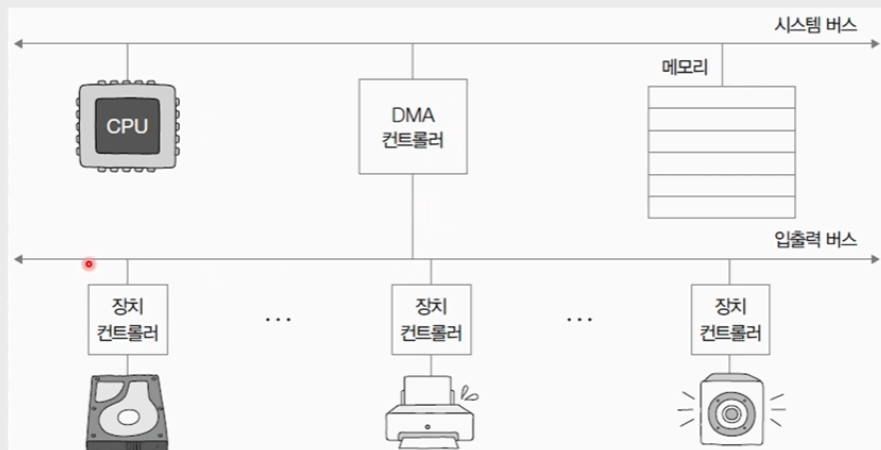
>> 이것이 취업을 위한 컴퓨터 과학이다

23

지금까지 입출력 기법들은 반드시 CPU를 거쳤어야만 했음.

근데 입출력 작업이 많아지면 CPU에 과부하가 생길 수 있음. 그래서 직접 메모리에 접근할 수 있는 입출력 기법이 바로 DMA임.

Chapter 02-5 보조기억장치와 입출력장치(24)



>> 이것이 취업을 위한 컴퓨터 과학이다

25

Chapter 02-5 보조기억장치와 입출력장치(25)

- DMA 입출력 과정

1. CPU가 DMA 컨트롤러에게 입출력장치의 주소, 수행할 연산, 연산할 메모리 주소 등의 정보와 함께 **입출력 작업을 명령**
2. DMA 컨트롤러가 CPU 대신 장치 컨트롤러와 상호작용하며 **입출력 작업을 수행**
이때 DMA 컨트롤러는 필요한 경우 메모리에 직접 접근하여 정보를 읽거나 씀
입출력장치와 메모리 사이에 주고 받을 데이터가 CPU를 거치지 않음
3. DMA 컨트롤러는 입출력 작업이 끝나면 CPU에게 **인터럽트를 통해 작업이 끝났음을 알림**

- CPU 입장에서는 DMA 컨트롤러에게 입출력 작업 명령을 내리고, 인터럽트만 받으면 되기 때문에 입출력 부담을 크게 줄일 수 있음

>> 이것이 취업을 위한 컴퓨터 과학이다

26

Chapter 02-5 보조기억장치와 입출력장치(26)

여기서 잠깐

사이클 스틸링

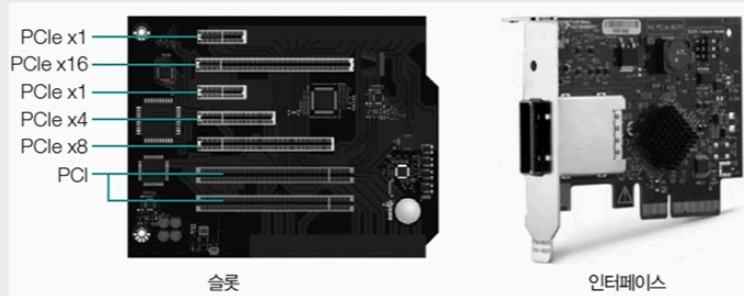
- 버스는 공유 자원 - 두 장치가 동시에 하나의 버스를 이용할 수 없다는 의미
- 시스템 버스도 마찬가지
 - DMA 컨트롤러는 시스템 버스를 통해 메모리에 직접 접근이 가능하지만, 시스템 버스는 동시 사용이 불가능
 - CPU가 시스템 버스를 사용할 때는 DMA 컨트롤러가 시스템 버스를 사용할 수 없고, DMA 컨트롤러가 시스템 버스를 사용할 때는 CPU가 시스템 버스를 사용할 수 없음
- 따라서 DMA 컨트롤러는 CPU가 시스템 버스를 사용하지 않을 때마다 조금씩 사용하거나, CPU가 시스템버스 사용을 양보
- 이런 의미로 DMA의 시스템 버스 사용을 사이클 스틸링(cycle stealing)이라고 함

>> 이것이 취업을 위한 컴퓨터 과학이다

27

Chapter 02-5 보조기억장치와 입출력장치(27)

- PCIe(Peripheral Component Interconnect express)
 - PCIe 버스는 PCI라는 입출력 버스의 발전된 형태
 - 오늘날의 메인 보드에서 가장 대중적으로 볼 수 있는 입출력 버스 중 하나
 - SSD, GPU, 또 5장에서 학습할 네트워크 인터페이스 카드 등 다양한 입출력장치를 연결



Chapter 02-5 보조기억장치와 입출력장치(28)

- PCIe와 연결된 장치의 성능
 - 1) PCIe 버전에 따라 최대 속도가 달라질 수 있음
 - PCIe 버스는 지속적으로 발전
 - PCIe 버스의 버전은 PCIe 3.0, PCIe 4.0, PCIe 5.0과 같이 PCIe 뒤에 버전을 나타내는 숫자를 덧붙이며, 버전에 따라 지원되는 최대 속도가 다름
 - 2) PCIe 버스는 여러 레인을 이용해 정보를 주고받음
 - 레인(lane) - PCIe 버스를 통해 정보를 송수신하는 단위
 - 'x레인 수'와 같이 표기 - PCIe 4.0 x4는 4개의 레인을 활용하는 PCIe 4.0을 의미

버전	레인당 최대 속도
PCIe 1.0	250MB/s
PCIe 2.0	500MB/s
PCIe 3.0	985MB/s
PCIe 4.0	1,969MB/s
PCIe 5.0	3,94GB/s
PCIe 6.0	7.56GB/s
PCIe 7.0	15.13GB/s