

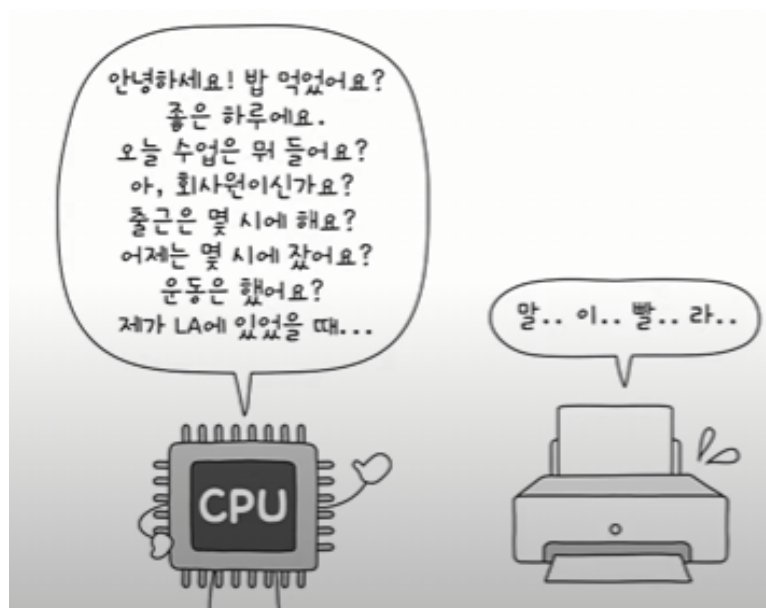
20강. 장치 컨트롤러와 장치 드라이버

CPU 입출력장치 정보 주고 받기



컴퓨터에는 매우 다양한 종류의 입출력장치가 연결되어 있습니다. 키보드, 마우스, 모니터, 프린터, SSD, 네트워크 카드 등 그 종류만 해도 수백 가지에 이릅니다.

문제는 이 장치들이 모두 동작 방식과 데이터 전송 속도가 다르다는 점입니다.



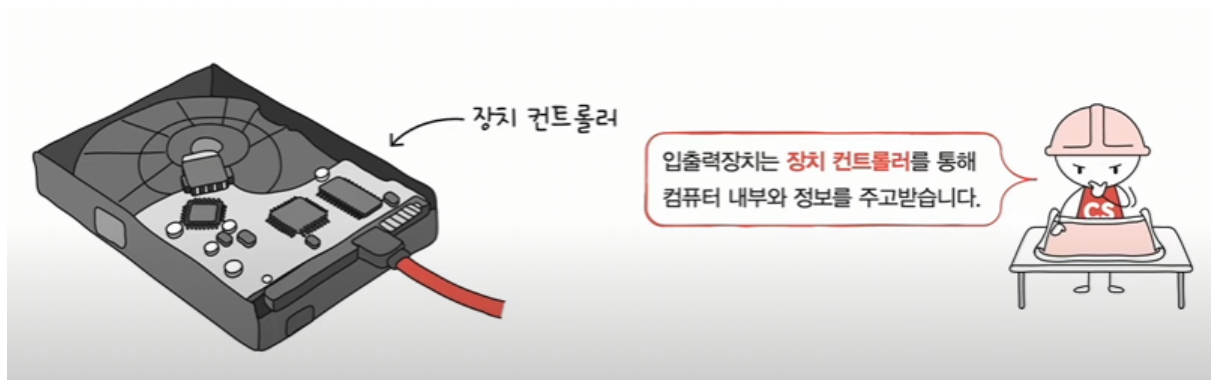
어떤 장치는 초당 수 기가바이트의 데이터를 전송할 수 있는 반면, 어떤 장치는 초당 수 킬로바이트 수준의 속도를 가집니다. 또한 각 장치는 데이터 형식과 통신 프로토콜이 서로 달라,

CPU가 모든 장치의 특성을 일일이 이해하고 직접 제어하는 것은 현실적으로 불가능합니다. 일반적으로 CPU와 메모리의 데이터 전송률은 높지만 입출력 장치의 데이터 전송률은 낮습니다.

이러한 이유로 컴퓨터는 장치 컨트롤러(Device Controller)를 사용합니다.

- 전송률 : 데이터를 얼마나 빨리 교환할 수 있는지를 나타내는 지표

장치 컨트롤러



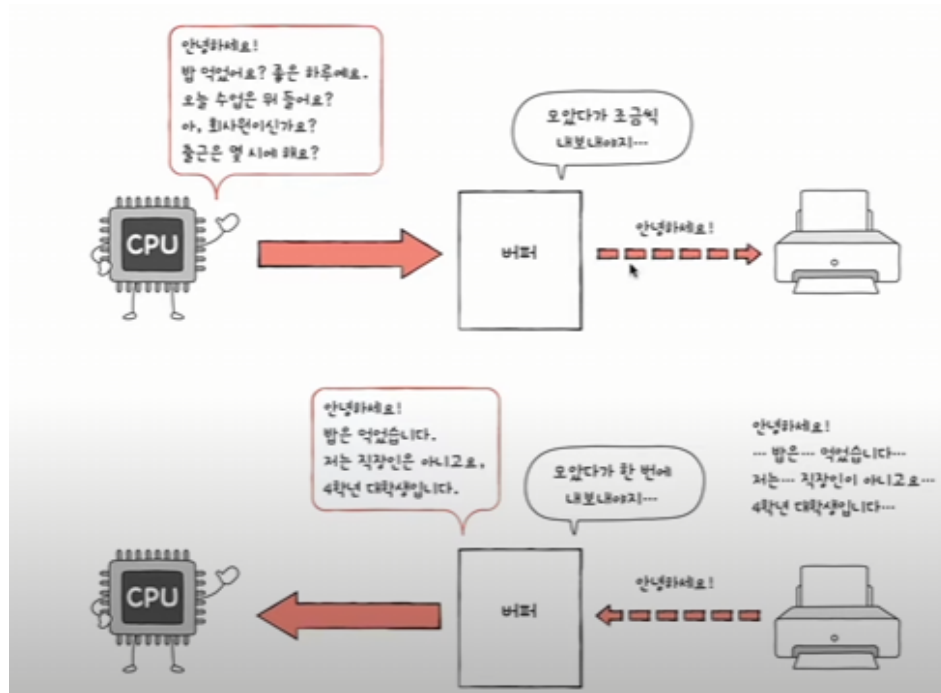
입출력장치는 CPU와 직접 연결되지 않고, 장치 컨트롤러라는 하드웨어를 통해 컴퓨터와 연결됩니다. 장치 컨트롤러는 입출력 제어기(I/O Controller) 또는 입출력 모듈(I/O Module)이라고도 부르며, CPU와 입출력장치 사이에서 통신 중개자 역할을 합니다.

CPU가 입출력장치에 직접 명령을 내리는 대신, 장치 컨트롤러가 그 명령을 받아 장치가 이해할 수 있는 신호로 변환합니다. 반대로, 장치에서 발생한 데이터를 CPU가 이해할 수 있도록 변환해 전달하기도 합니다.

즉, 장치 컨트롤러는 CPU와 입출력장치 간의 언어를 번역하고, 속도를 조정하며, 오류를 관리하는 하드웨어 중재자라고 할 수 있습니다.

장치 컨트롤러 역할

장치 컨트롤러는 다음과 같은 역할을 수행합니다.



1. CPU와 입출력장치 간의 통신을 중개

CPU의 제어 신호나 데이터가 장치로 전달될 수 있도록 하며, 반대로 장치의 응답을 CPU가 이해할 수 있도록 변환합니다.

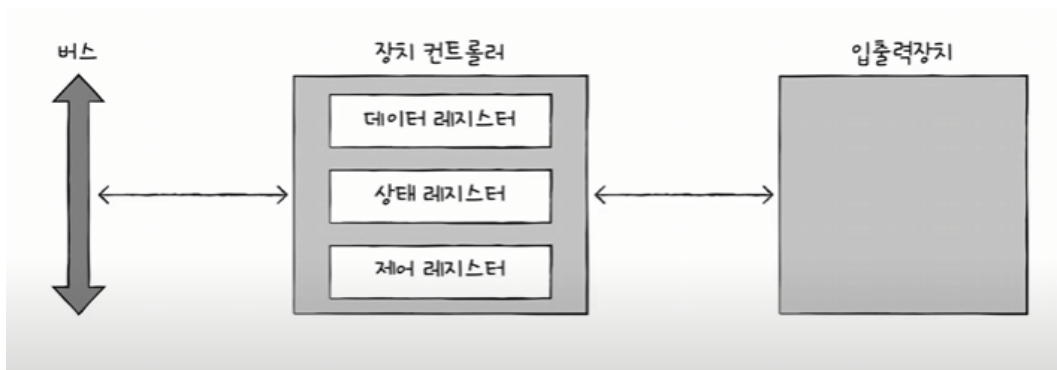
2. 오류 검출 (Error Detection)

데이터 전송 도중 발생할 수 있는 손상이나 전송 실패를 감지해 CPU에 알리고, 안정적인 데이터 처리를 가능하게 합니다.

3. 데이터 버퍼링 (Buffering)

버퍼링이란 전송률이 서로 다른 두 장치 사이에 데이터를 임시로 저장해 전송 속도의 차이를 완화하는 기술입니다. 빠른 장치가 느린 장치를 기다리지 않고 효율적으로 동작할 수 있도록 돕는 완충 역할을 합니다.

장치 컨트롤러의 구조



장치 컨트롤러는 보통 버스(Bus)와 연결되어 있으며, 내부적으로는 세 가지 주요 구성 요소를 가집니다.

- 데이터 레지스터
- 상태 레지스터
- 제어 레지스터

데이터 레지스터 (Data Register)

CPU와 입출력장치 사이에서 주고받는 실제 데이터가 임시로 저장되는 공간입니다.

이는 일종의 버퍼 역할을 하며, 때로는 RAM이 사용되기도 합니다.

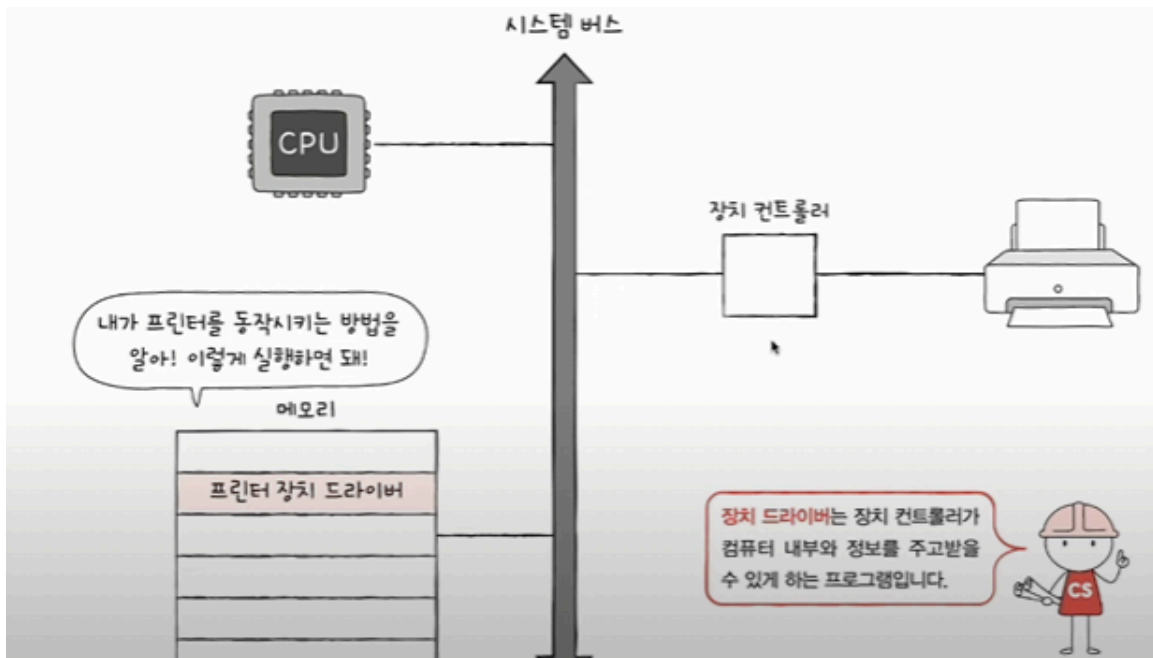
상태 레지스터 (Status Register)

상태 레지스터는 장치의 현재 상태 정보를 저장합니다. 예를 들어 장치가 작업을 수행할 준비가 되었는지, 입출력 작업이 완료되었는지, 오류가 발생했는지 등의 상태가 기록됩니다.

제어 레지스터 (Control Register)

CPU가 장치에 내리는 명령이나 제어 신호를 저장합니다. 어떤 작업을 수행해야 하는지, 언제 시작할지 등의 제어 정보가 담깁니다. 일부 컨트롤러에서는 상태 레지스터와 제어 레지스터를 하나로 통합하여 사용하기도 합니다.

장치 드라이버 (Device Driver)



장치 컨트롤러가 하드웨어적인 통로라면, 장치 드라이버는 그 위에 동작하는 소프트웨어적인 통로입니다.

장치 드라이버는 운영체제(OS)에 포함된 프로그램으로, 장치 컨트롤러를 제어하고 감시합니다. 즉, 운영체제가 하드웨어 장치와 정보를 주고받을 수 있도록 돕는 소프트웨어 인터페이스 역할을 합니다.

운영체제가 특정 장치를 인식하고 사용하기 위해서는 반드시 해당 장치의 드라이버가 설치되어 있어야 합니다. 드라이버가 없다면 운영체제는 장치의 기능이나 데이터 전송 방식을 이해하지 못하므로, 아무리 하드웨어적으로 연결되어 있어도 장치는 동작하지 않습니다.

예를 들어, 새 프린터나 그래픽카드를 연결했을 때 "드라이버를 설치해야 합니다." 라는 메시지가 표시되는 이유가 바로 이 때문입니다. 드라이버가 설치되면 운영체제는 장치 컨트롤러를 통해 장치와 정상적으로 통신할 수 있습니다.

결국 장치 컨트롤러와 장치 드라이버는 CPU와 장치를 이어주는 하드웨어적, 소프트웨어적 다리라고 할 수 있습니다. 컨트롤러가 물리적 신호를 주고받는 하드웨어라면, 드라이버는 운영체제와 장치를 연결해주는 논리적 소프트웨어 계층입니다.