

운영체제와 컴퓨터

운영체제 (OS, Operating System)

사용자가 컴퓨터를 쉽게 다루게 해주는 인터페이스

한정된 메모리나 시스템 자원을 효율적으로 분배

운영체제의 역할과 구조

운영체제의 역할

크게 네 가지

1. CPU 스케줄링과 프로세스 관리

CPU 소유권을 어떤 프로세스에 할당할지, 프로세스의 생성과 삭제, 자원 할당 및 반환을 관리

2. 메모리 관리

한정된 메모리를 어떤 프로세스에 얼마나 할당해야하는지 관리

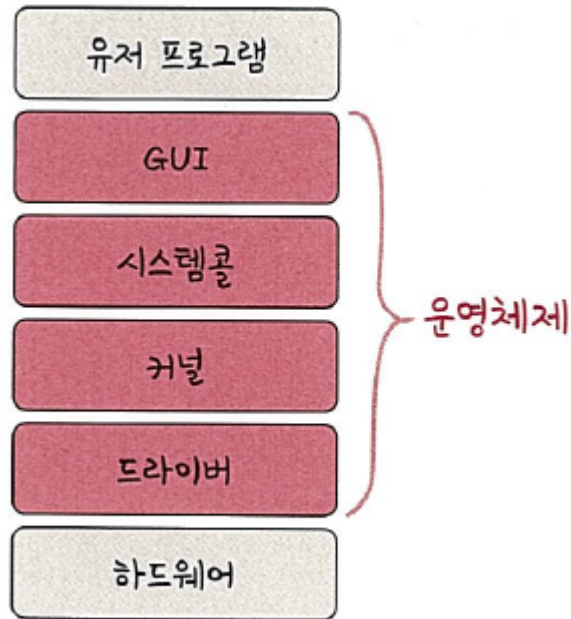
3. 디스크 파일 관리

디스크 파일을 어떤 방법으로 보관할지 관리

4. I/O 디바이스 관리

마우스, 키보드와 컴퓨터 간에 데이터를 주고받는 것을 관리

운영체제의 구조



커널(Kernel)

운영체제의 핵심부이자 시스템콜 인터페이스를 제공

보안, 메모리, 프로세스, 파일시스템, I/O 디바이스, I/O 요청관리 등 운영체제의 중추적인 역할을 함

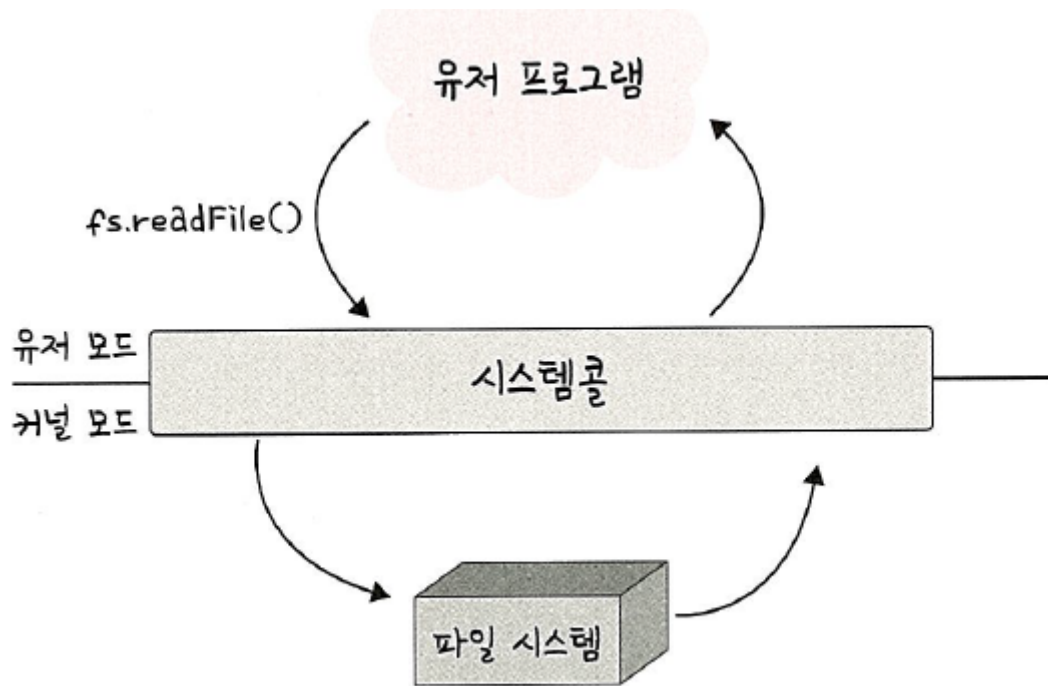
시스템콜 (System Call)

운영체제가 커널에 접근하기 위한 인터페이스

유저 프로그램이 운영체제의 서비스를 받기 위해 커널 함수를 호출할 때 씀

유저 프로그램이 I/O 요청으로 트랩(trap)을 발동하면 올바른 요청인지 확인한 후 유저 모드가 시스템콜을 통해 커널모드로 변환되어 실행됨

ex) `fs.readFile()` 이라는 파일 시스템의 파일을 읽는 함수 실행



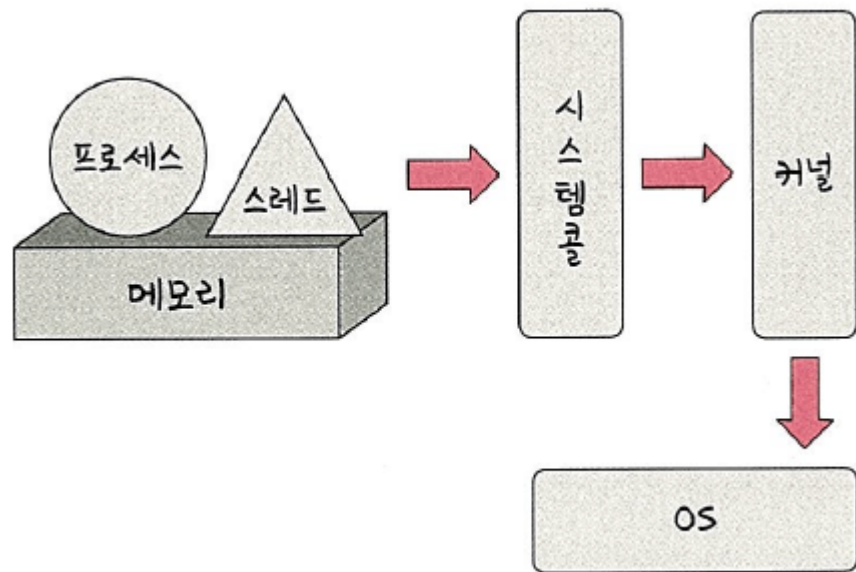
유저모드에서 파일을 읽지 않고 커널 모드로 들어가 파일을 읽음

- 유저모드: 유저가 접근할 수 있는 영역. 컴퓨터 자원에 함부로 침범하지 못하는 모드
- 커널모드: 모든 컴퓨터 자원에 접근할 수 있는 모드

다시 유저모드로 돌아가 그 뒤에 있는 유저 프로그램의 로직을 수행함

컴퓨터 자원에 대한 직접 접근을 차단할 수 있음

프로그램을 다른 프로그램으로부터 보호할 수 있음



프로세스나 스레드에서 운영체제로 어떠한 요청을 할 때 시스템콜이라는 인터페이스와 커널을 거쳐 운영체제에 전달됨

시스템콜은 하나의 추상화 계층으로, 이를 통해 네트워크 통신이나 DB와 같은 낮은 단계의 영역 처리에 대한 부분을 크게 신경 쓰지 않고 프로그램을 구현할 수 있는 장점이 있음

modebit

1 또는 0의 값을 가지는 플래그 변수

시스템콜 작동 시 modebit 을 참고해서 유저 모드와 커널 모드를 구분함

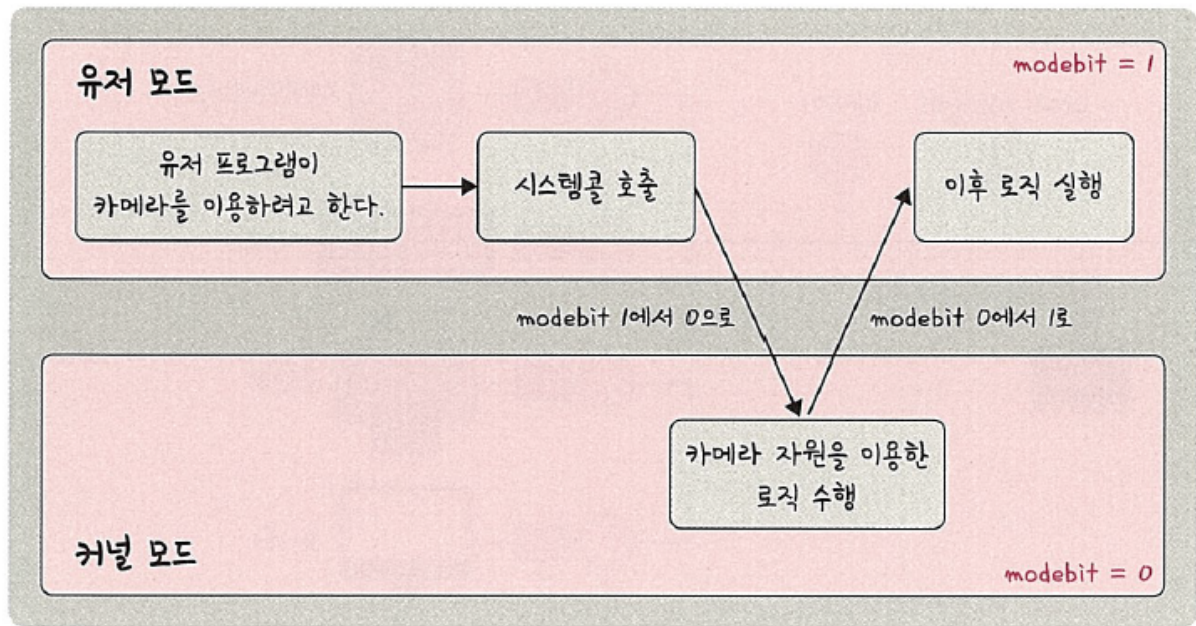
카메라, 키보드 등의 I/O 디바이스는 운영체제를 통해서만 작동해야함

만약 유저 모드를 기반으로 켜진다면, 사용자의 의도와는 상관없이 장치가 작동하는 등의 공격이 행해질 수 있음

커널모드를 거쳐 운영체제를 통해 작동한다고 해도 완전히 막을 수는 없으나, 운영체제를 통해 작동하도록 해야 이러한 공격을 막기 쉬움. 이를 위한 장치가 modebit

modebit 의 0은 커널모드, 1은 유저모드로 설정됨

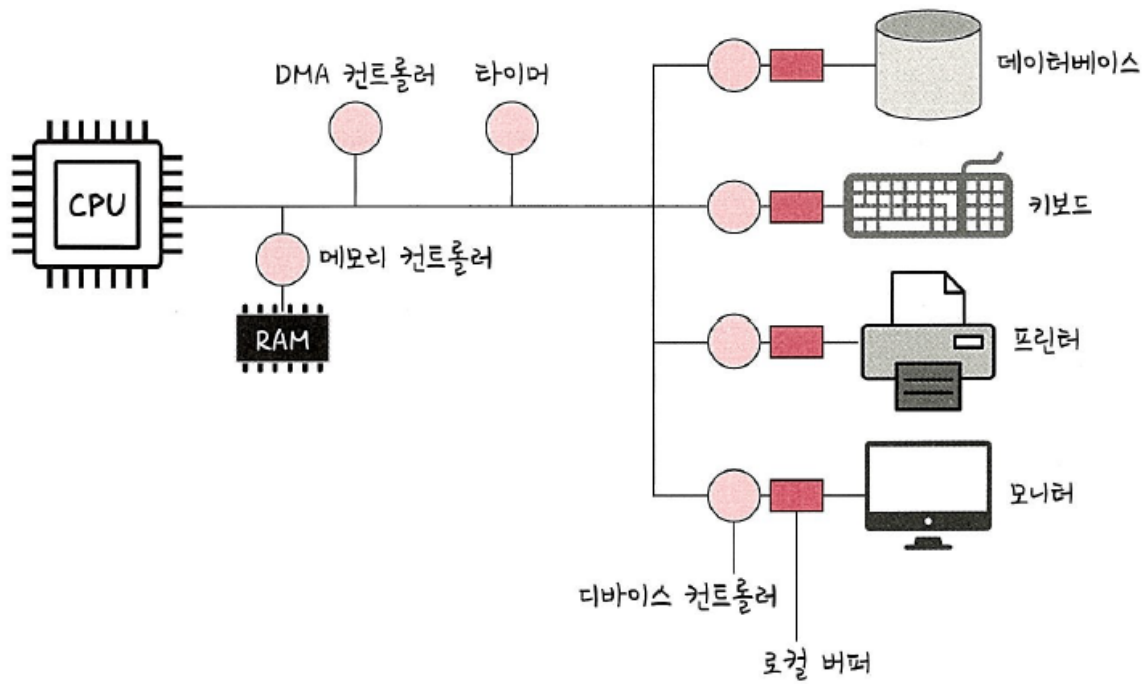
유저모드일 경우 시스템콜을 못하게 막아 한정된 일만 가능하게함



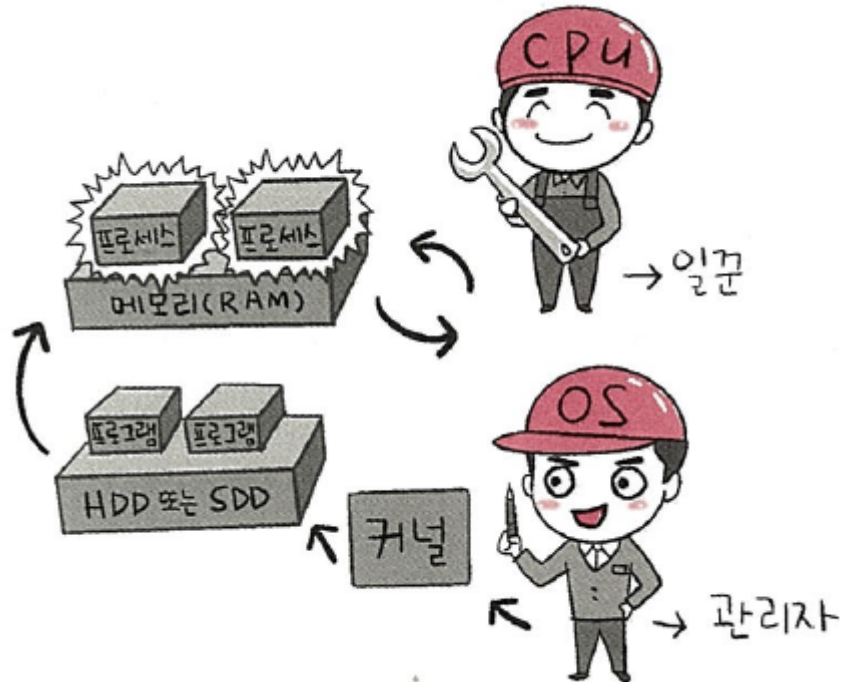
위와 같이 유저프로그램이 카메라를 이용하려고 할 때, 시스템콜을 호출
modebit 을 1에서 0으로 바꿔 커널모드로 변경
카메라 자원을 이용한 로직 수행
이후 modebit 을 다시 1로 바꿔 유저모드로 변경하고 이후 로직을 수행

컴퓨터의 요소

CPU, DMA, 컨트롤러, 메모리, 타이머, 디바이스 컨트롤러 등으로 이루어져 있음



CPU (Central Processing Unit)



산술논리연산장치, 제어장치, 레지스터로 구성되어있는 컴퓨터 장치
인터럽트에 의해 단순히 메모리에 존재하는 명령어를 해석해서 실행

위 그림과 같이 관리자 역할을 하는 운영체제의 커널이 프로그램을 메모리에 올려 프로세스로 만들면 일꾼인 CPU가 이를 처리

제어장치 (CU, Control Unit)

프로세스 조작을 지시하는 CPU의 한 부품

입출력 장치 간 통신을 제어함

명령어들을 읽고 해석함

데이터 처리를 위한 순서를 결정함

레지스터 (Register)

CPU 안에 있는 매우 빠른 임시기억장치

CPU와 직접 연결되어 있으므로 연산 속도가 메모리보다 빠름

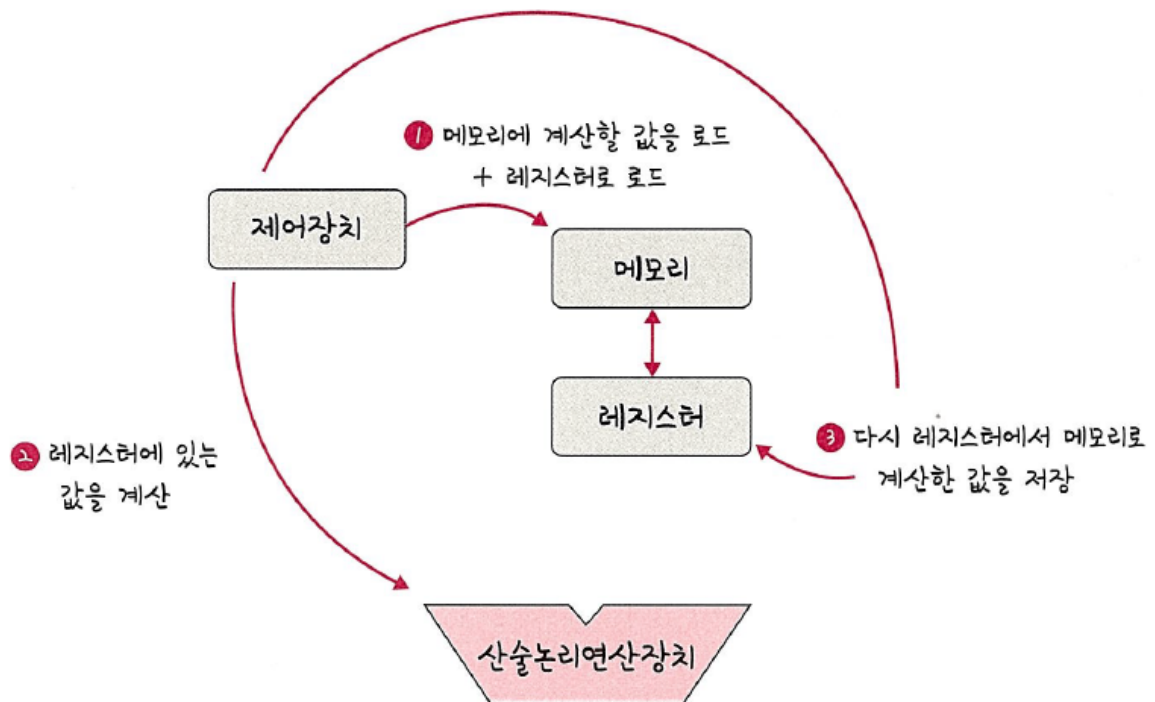
CPU는 자체적으로 데이터를 저장할 방법이 없음 → 레지스터를 거쳐 데이터를 전달함

산술논리연산장치 (ALU, Arithmetic Logic Unit)

산술연산과 배타적 논리합, 논리곱과 같은 논리연산을 계산하는 디지털 회로

CPU의 연산처리

CPU에서 제어장치, 레지스터, 산술논리연산자를 통해 연산하는 예는 아래와 같음



1. 제어장치가 메모리와 레지스터에 계산할 값을 로드함
2. 제어장치가 레지스터에 있는 값을 계산하라고 ALU에 명령함
3. 제어장치가 계산된 값을 다시 레지스터에서 메모리로 계산한 값을 저장함

인터럽트 (Interrupt)

어떤 신호가 들어왔을 때 CPU를 잠깐 정지시키는 것

IO 디바이스로 인한 인터럽트, 0으로 나누는 산술 연산에서의 인터럽트, 프로세스 오류 등으로 발생

인터럽트 발생 시 인터럽트 핸들러 함수가 모여 있는 인터럽트 벡터로 가서 인터럽트 핸들러 함수가 실행됨

• 인터럽트 핸들러 함수

인터럽트 발생 시 이를 핸들링하기 위한 함수

커널 내부의 IRQ를 통해 호출됨

regist_irq() 를 통해 등록할 수 있음

인터럽트 간에는 우선순위가 있으며, 우선순위에 따라 실행됨

인터럽트는 하드웨어 인터럽트, 소프트웨어 인터럽트의 두가지로 나뉨

- **하드웨어 인터럽트**

IO 디바이스에서 발생하는 인터럽트를 말함

인터럽트 라인이 설계된 이후 순차적인 인터럽트 실행을 중지하고 운영체제에 시스템콜을 요청해서 원하는 디바이스로 향해 디바이스에 있는 작은 로컬 버퍼에 접근하여 일을 수행함

- **소프트웨어 인터럽트**

트랩(trap) 이라고도 함

프로세스 오류 등으로 프로세스가 시스템콜을 호출할 때 발동

DMA 컨트롤러

IO 디바이스가 메모리에 직접 접근할 수 있도록 하는 하드웨어 장치

CPU에만 너무 많은 요청이 들어옴에 따라 생기는 CPU 부하를 막아주며 CPU의 일을 부담하는 보조 일꾼

하나의 작업을 CPU와 DMA 컨트롤러가 동시에 하는 것을 방지

메모리 (Memory)

전자회로에서 데이터나 상태, 명령어 등을 기록하는 장치

보통 RAM (Random Access Memory)을 일컬어 메모리라고도함

CPU 는 계산을 담당하고, 메모리는 기억을 담당함

메모리가 크면 클수록 많은 일을 동시에 할 수 있음

- **공장에 비유**

CPU는 일꾼이고, 메모리는 작업장

작업장이 크면 → 물건을 많이 가져다놓고 많은 일을 할 수 있다.

타이머 (Timer)

몇 초 안에는 작업이 끝나야 한다는 것을 정하고 특정 프로그램에 시간 제한을 두는 역할
시간이 많이 걸리는 프로그램 작동 시 제한을 걸기 위해 존재함

디바이스 컨트롤러 (Device Controller)

컴퓨터와 연결되어 있는 작은 IO 디바이스들의 작은 CPU를 말함