

# 운영체제

2019.6.25



컴퓨터공학과 이병문



# 강의일정

6/24 과목, 강의소개

25 운영체제 개요

26 Process, Thread

27 Concurrent Process

28 Concurrent Process

7/1 중간고사1



내용

컴퓨터 역할과 목적  
운영체제 유형별 특징  
운영체제 발전과정  
운영체제 기능  
운영체제 서비스  
운영체제 구조

# 운영체제 역할과 목적

## ■ 운영체제 역할

- ☑ 컴퓨터시스템 = 사용자 + 하드웨어 + 소프트웨어
- ☑ 운영체제는 하드웨어와 사용자(소프트웨어)간의
  - 1)조정자 역할, 2)자원관리자 역할, 3)입출력장치와 사용자프로그램 제어 역할
- ☑ 사용자:
  - 사람,
  - 프로그램
  - 장치,...
- ☑ 소프트웨어:
  - 시스템SW, -응용SW
- ☑ 하드웨어: CPU, Memory, I/O Devices



# 운영체제 역할과 목적

## ■ 운영체제 목적

☑ 운영체제의 목적 : Convenience (편리성) + Efficiency (효율성, 성능향상) + Control

☑ How to get

- Graphic User Interfaces
- Process & Threads Management
- Processor (CPU) Management
- Memory Management
- I/O Management
- File Management
- Security Management

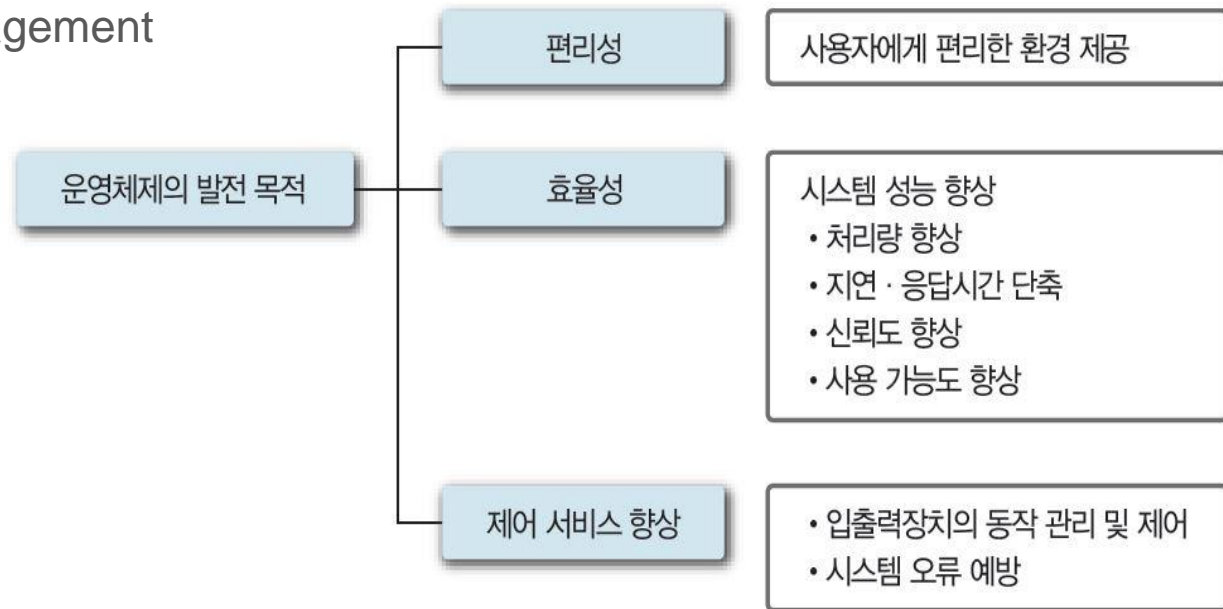
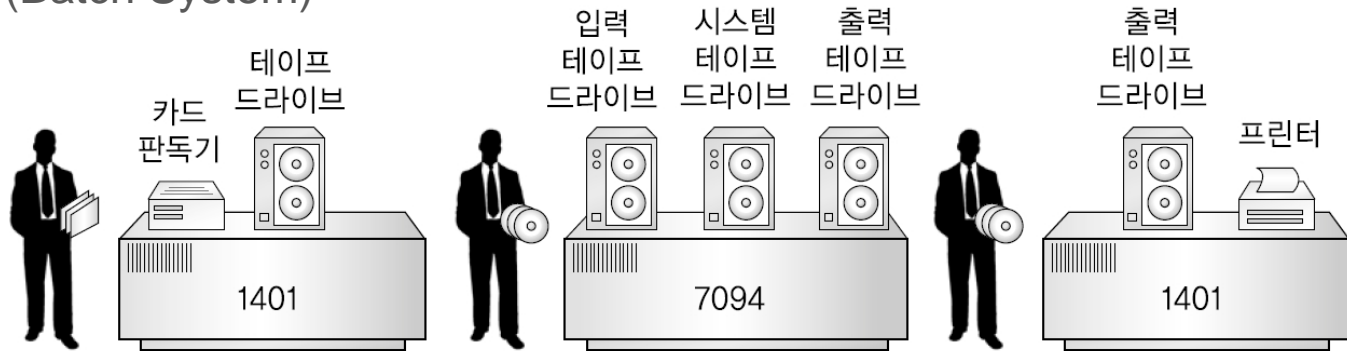


그림 2-3 운영체제의 발전 목적

# 운영체제 역할과 목적

## ■ 운영체제의 발전과정 (운영처리과정의 발달)

- ✓ 작업 별 처리 (수작업 -> 처리)
- ✓ 일괄처리(Batch System)



초기의 일괄 처리 시스템

- ✓ 오프라인(Offline) System
- ✓ Buffering
- ✓ Spooling

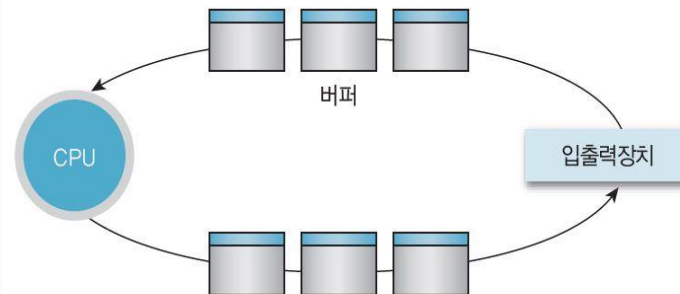
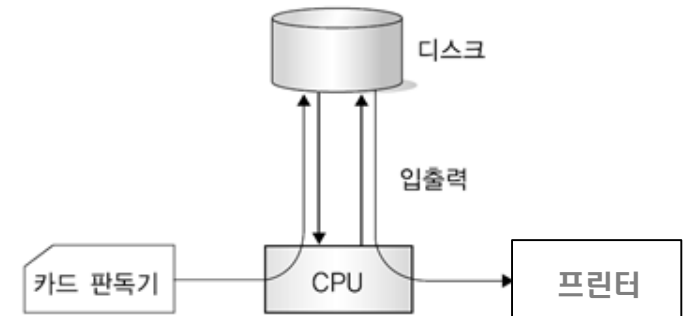


그림 2-7 버퍼링

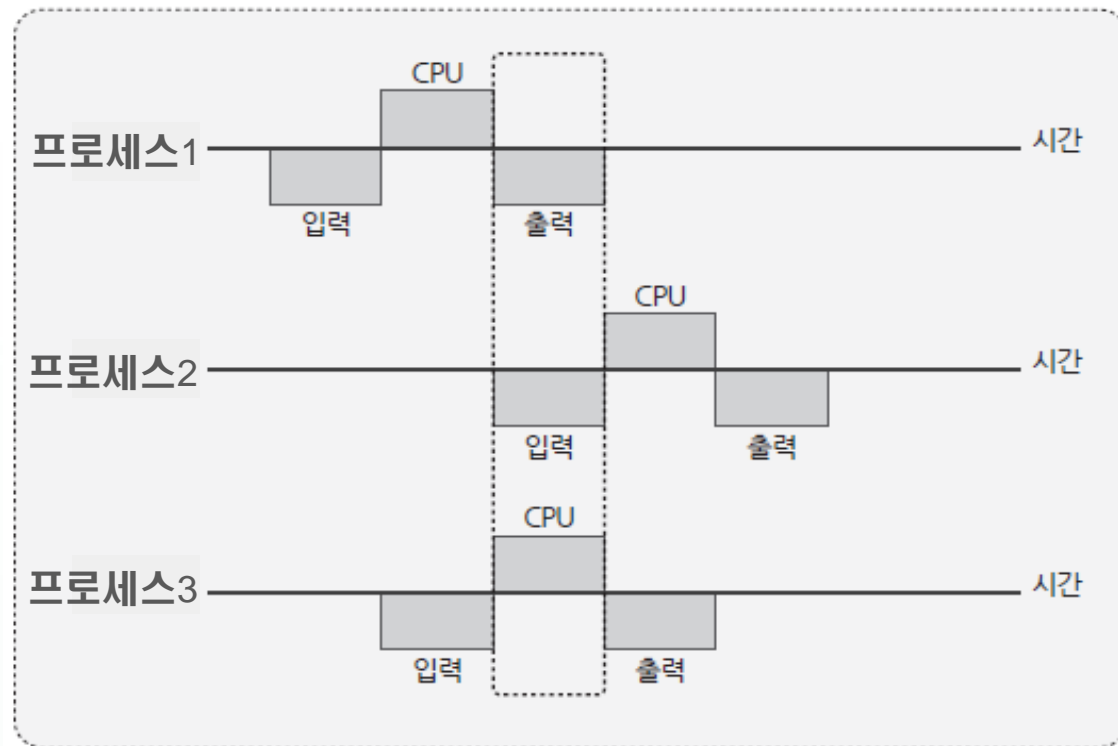




# 운영체제 유형별 특징

## ■ 유형(Computer System)

### 1. Multi Programming System ( = Multi tasks = Multi Processes )



다중 프로그래밍에서의 프로그램과 시간 관계

# 운영체제 유형별 특징

## ■ 유형(Computer System)

### 2. Time Sharing System (시분할시스템, TSS)

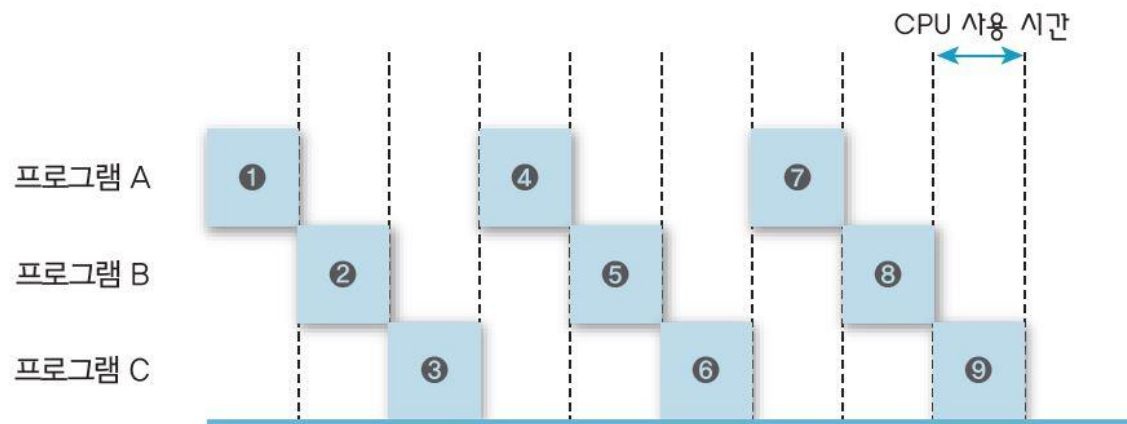
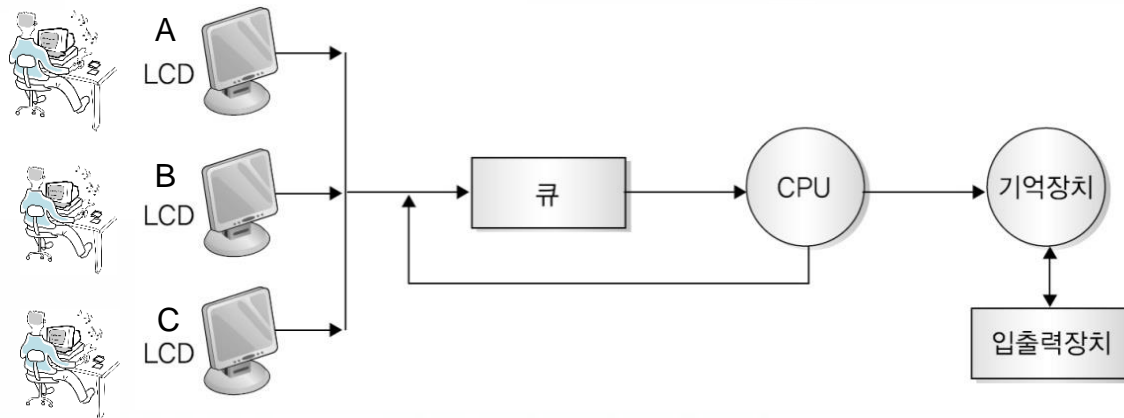


그림 2-14 시분할 시스템의 처리 방법 예

# 운영체제 유형별 특징

## ■ 유형(Computer System)

### 3. Distributed Processing System

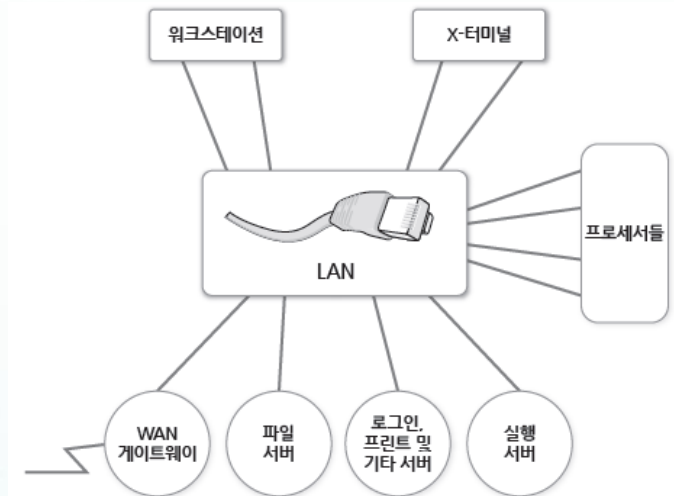
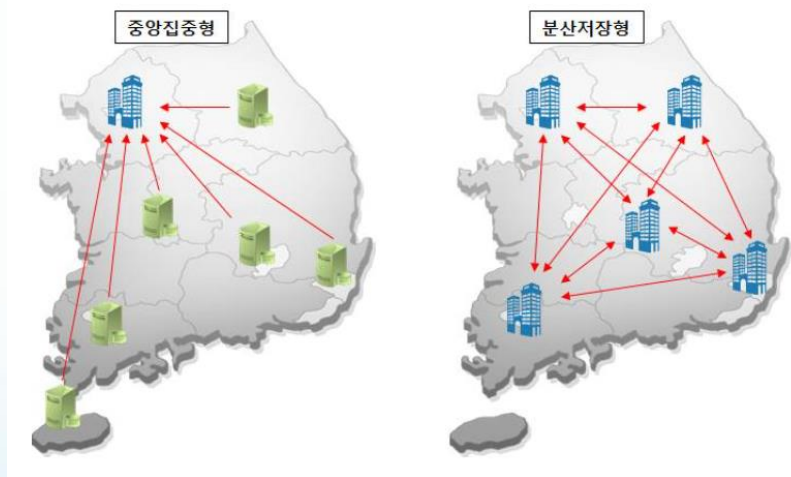
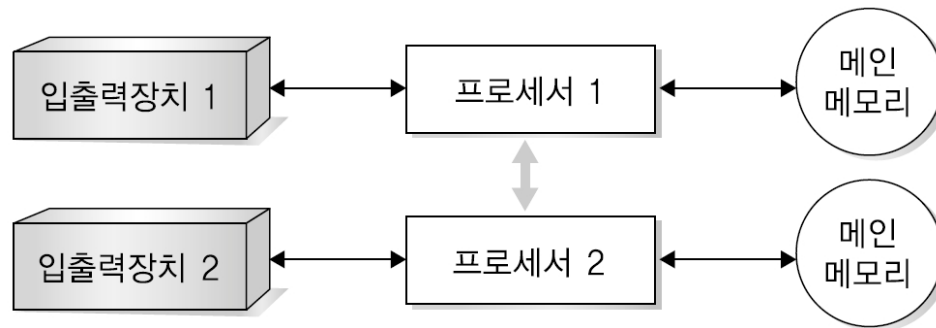


그림 2-20 분산 처리 시스템



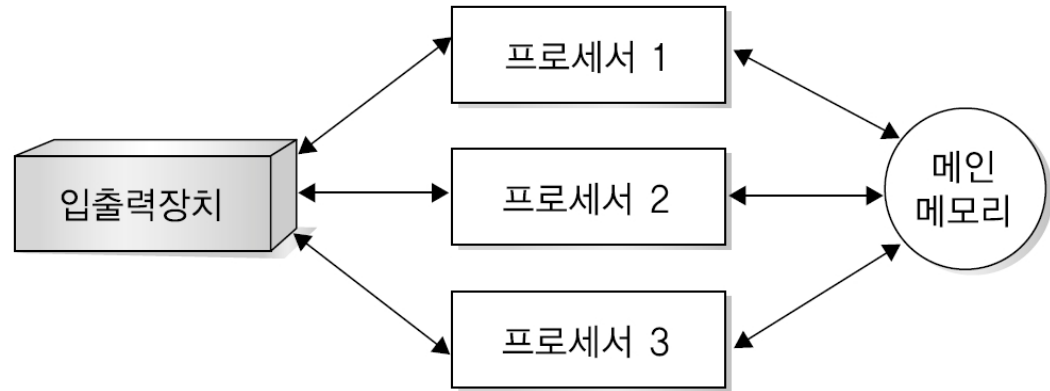
# 운영체제 유형별 특징

## ■ 유형(Computer System)

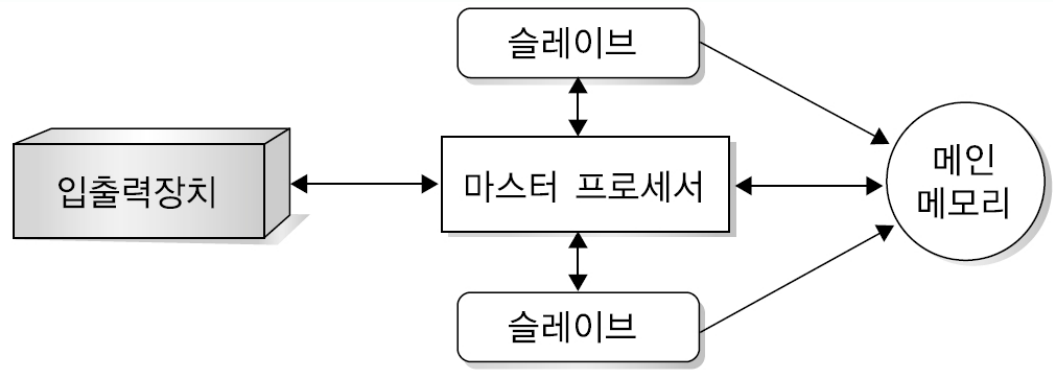
### 4. Multi Processor System

- 장점: 1) 처리율 증가  
2) 비용절감  
3) 신뢰성 증가

- Symmetric Multiprocessing(**SMP**)
- Asymmetric Multiprocessing(**AMP**)



대칭적 구성 다중 처리 시스템(**Symmetric**)

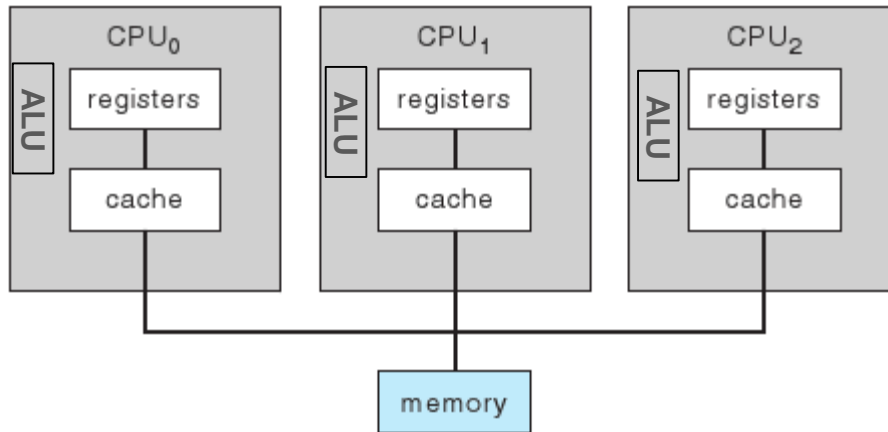


주/종 다중 처리 시스템 구성(**Asymmetric**)

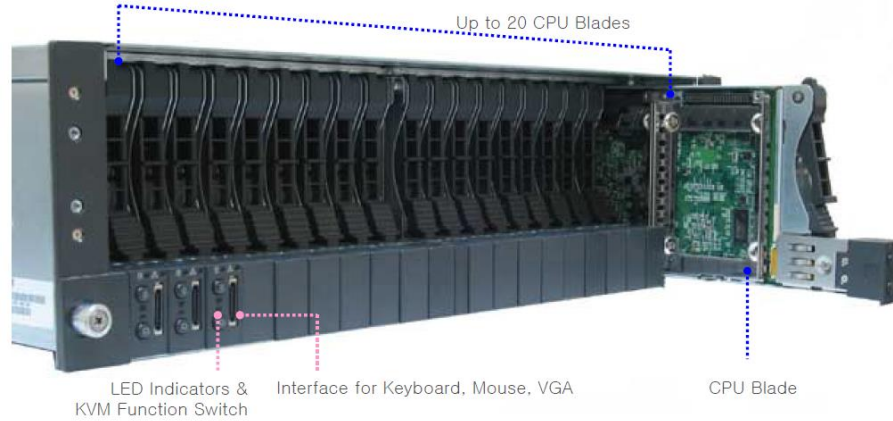
# 운영체제 유형별 특징

## ■ 유형(Computer System)

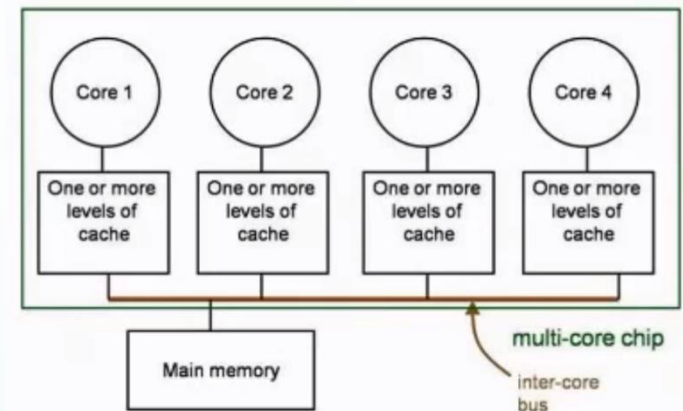
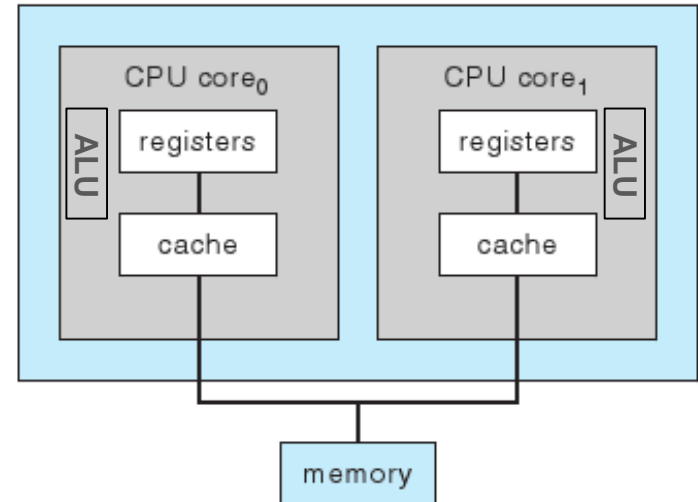
- Symmetric Multiprocessing(SMP)



## Blade Server ▼



## Dual Core CPU - Design ▼



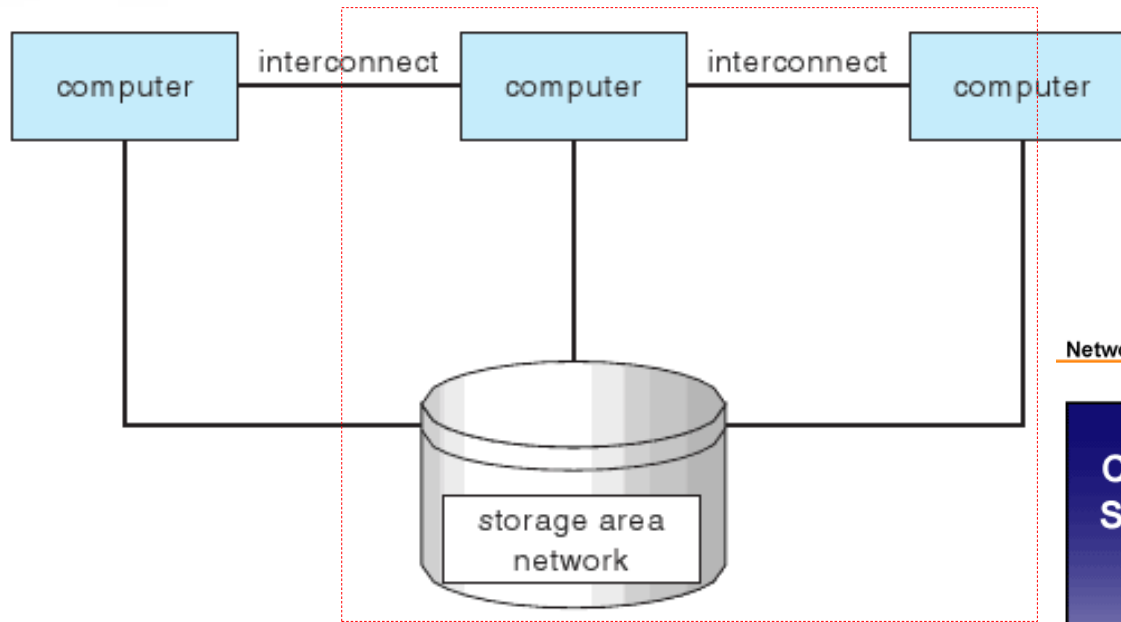
## Quad Core CPU

# 운영체제 유형별 특징

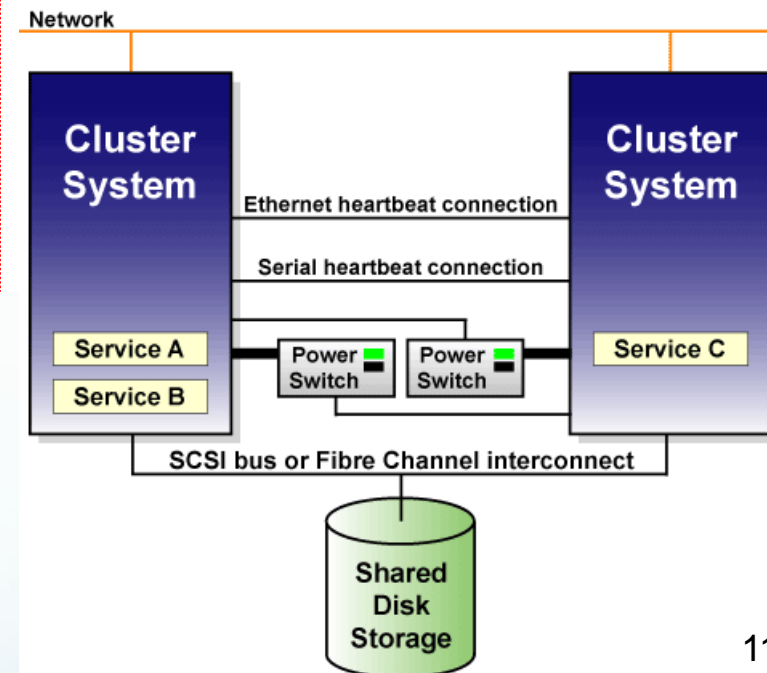
## ■ 유형(Computer System)

5.Desktop System (단일사용자시스템)

6.Cluster System



Clustered Systems



# 운영체제 유형별 특징

## ■ 유형(Computer System)

### 7. Real-Time System

- 데이터가 발생할 때 마다 즉시 처리
- 로봇제어, 화재감시,  
각종 센서/Actuator 처리 기능
- 시급한 순서(우선순위)기준으로 처리

처리시간 단축, 처리비용 절감  
데이터발생지점에서 직접 입출력  
특정상태의 재현불가



실시간처리시스템 예) 차선이탈 경보시스템

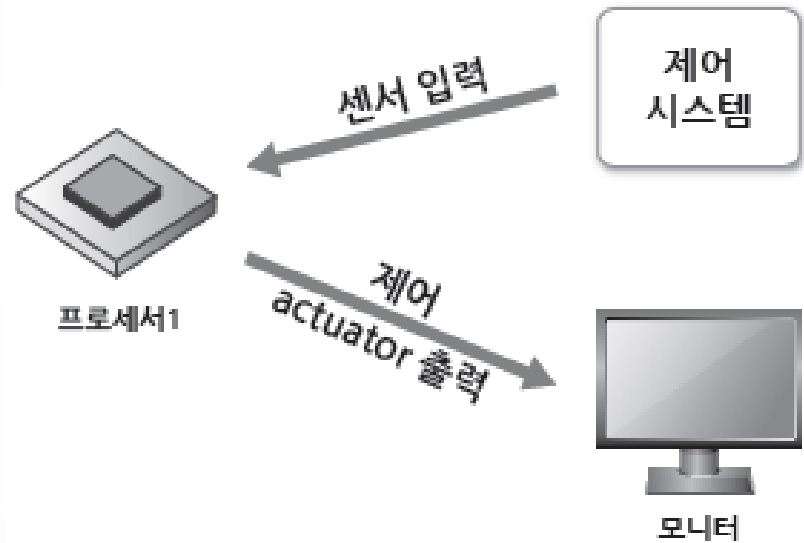


그림 2-19 실시간 시스템

차선이탈 경보 시스템 LDWS  
사고를 미연에 방지해주는 안전운전시스템



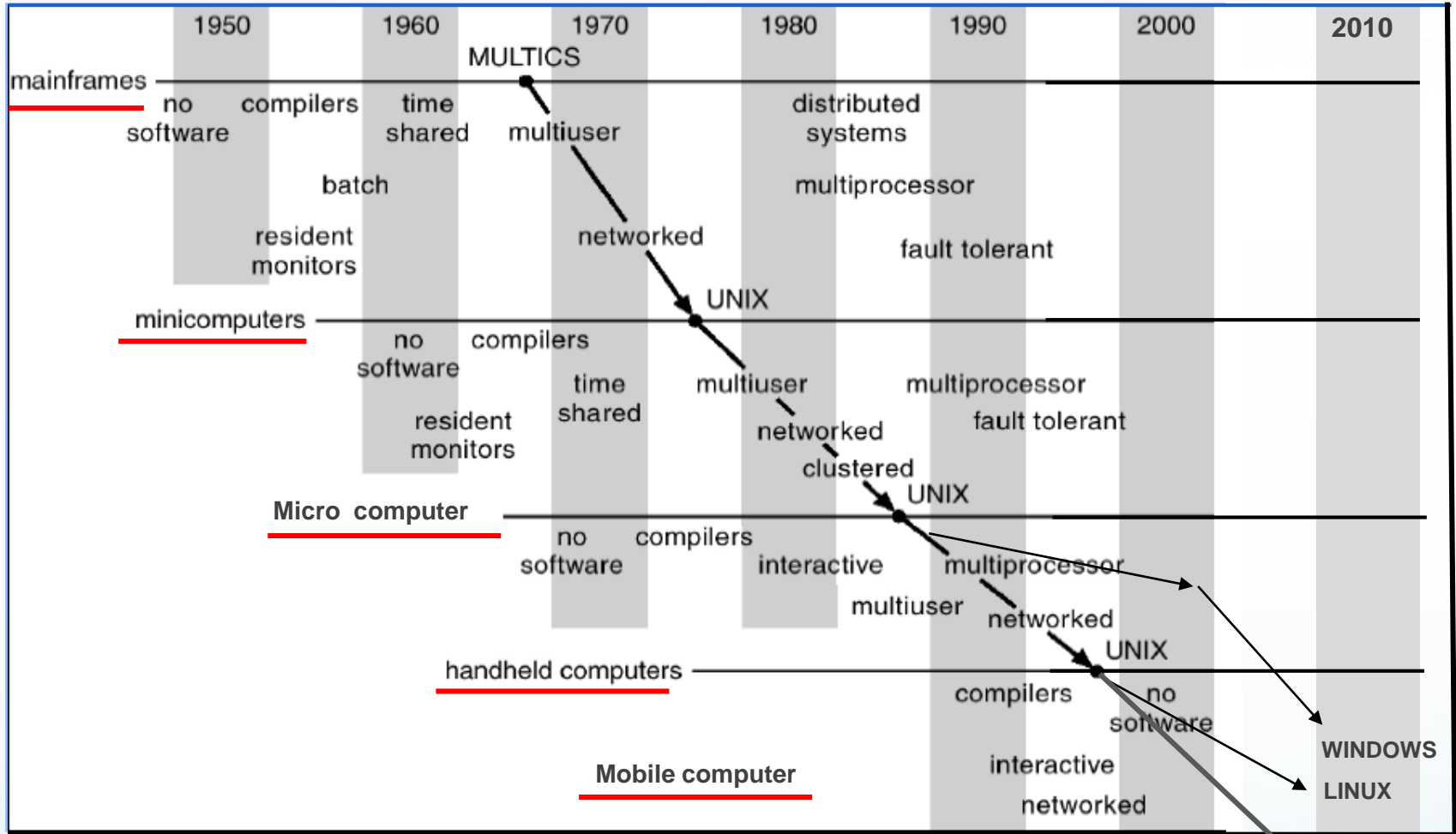
# 운영체제 발전과정

## ■ 세대별 발전과정

구분	시대	특징
제0세대	1940년대	<ul style="list-style-type: none"> <li>• 운영체제 없음</li> <li>• 기계어 사용</li> </ul> <b>진공관</b>
제1세대	1950년대	<ul style="list-style-type: none"> <li>• IBM 701(운영체제 효시)</li> <li>• 작업간의 원활한 변환</li> <li>• <u>버퍼링, 스푼링, 일괄처리 시스템</u></li> </ul>
제2세대	1960년대 초기	<ul style="list-style-type: none"> <li>• 고급언어로 운영체제 작성</li> <li>• 장치 독립성</li> <li>• 다중 프로그래밍, <u>다중 처리, 시분할 시스템</u></li> </ul>
제3세대	1960년대 중반 ~ 1970년대 중반	<ul style="list-style-type: none"> <li>• IBM 360 시리즈, <u>유닉스</u></li> <li>• 범용 시스템</li> <li>• 다중 모드 시스템</li> </ul>
제4세대	1970년대 중반 ~ 1990년대	<ul style="list-style-type: none"> <li>• 네트워크시스템</li> <li>• 가상기계, <u>분산 데이터 처리</u></li> <li>• 개인 컴퓨터</li> </ul>
	2000년대~	<ul style="list-style-type: none"> <li>▪ 모바일, 임베디드 시스템</li> <li>▪ 가상화, 클라우드 컴퓨팅</li> <li>▪ 사물인터넷, 웨어러블컴퓨터</li> </ul>

# 운영체제 발전과정

## ■ 운영체제의 역사 (Timeline of OS)



-> [https://en.wikipedia.org/wiki/Timeline\\_of\\_operating\\_systems](https://en.wikipedia.org/wiki/Timeline_of_operating_systems)

UNIX



# 운영체제의 기능

## ■ 운영체제의 기능

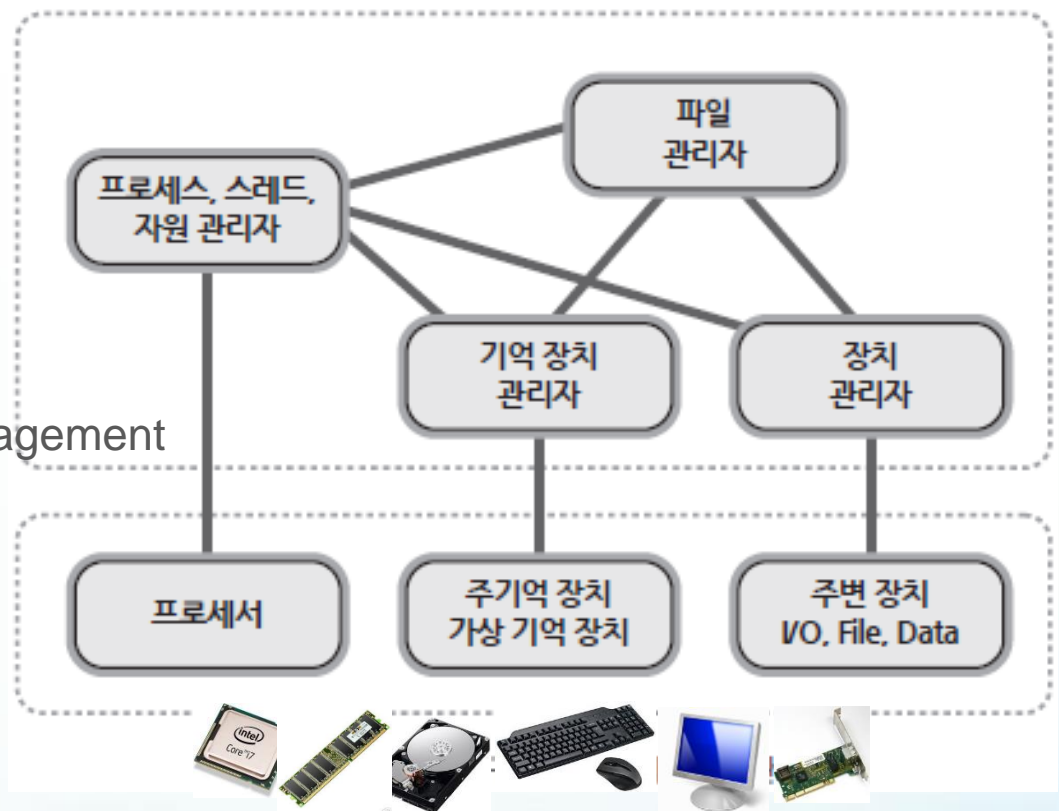
### ☑ 자원관리 기능

- Memory Management
- Process Management
- I/O Device Management
- File Management

### ☑ 기타 기능

- Security Management
- Networking & Distributed Management
- Command Interpreter (=Shell)

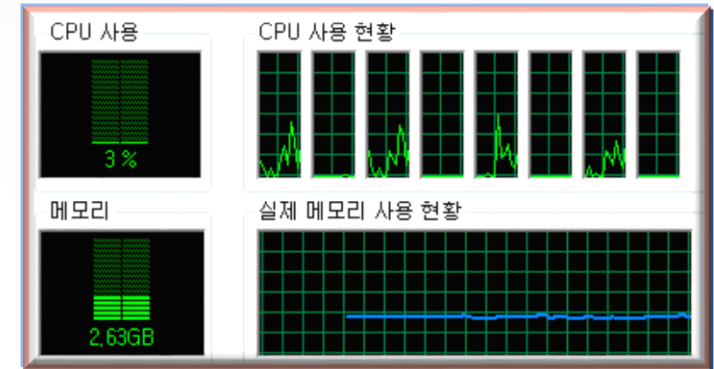
자원할당 -> 자원회수 -> 자원할당 -> 자원회수 -> ...



# 운영체제의 기능

## ■ 메모리(Memory) 관리기능

- ☑ Main Memory ;  
Usage Monitoring, Address Mapping,  
Allocation & Free
- ☑ Secondary Memory ;  
Memory <- Program & Data -> Disk  
Disk Scheduling & Allocation



## ■ 프로세스(Process) 관리기능

- ☑ Process vs Program ;  
Process : 실행중인 프로그램.  
즉, 실행을 위해 자원이 할당된 프로그램  
자원 = 프로세서 점유, 메모리, 파일, 입출력장치등.
- ☑ 운영체제가 수행할 프로세스 관리사항  
프로세스의 생성과 제거, 프로세스의 중지와 재수행  
프로세스 동기화기법, 프로세스간의 상호통신  
교착상태(Deadlock)방지를 위한 기능

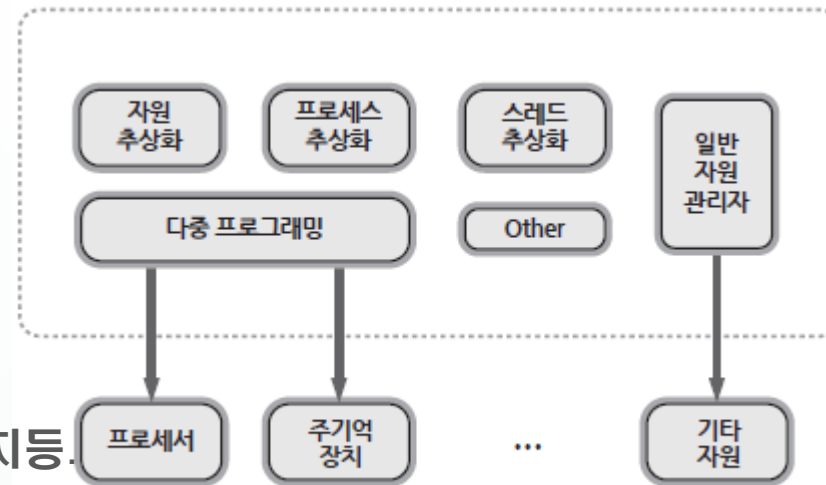


그림 2-4 프로세스 관리자

# 운영체제의 기능

## ■ 입출력(I/O) 장치관리기능

- ☑ 사용자가 I/O Devices 의 자세한 하드웨어특성을 알지 못해도 자유롭게 활용할 수 있도록
- ☑ 프로세서와 I/O Devices 간의 처리속도 차이해결 (버퍼링)
- ☑ Device Driver(장치 구동기 또는 장치 관리자)



## ■ 파일(File) 관리기능

- ☑ 다양한 저장장치를 동일한 방식으로 사용할 수 있도록 기능을 제공  
Disk, CD, DVD, Tape, USB Memory, Memory Stick,
- ☑ 파일의 생성/제거, 디렉터리(폴더)의 생성/제거등
- ☑ 파일시스템 : FAT, FAT32, NTFS, ext2, ext3, nfs, ...

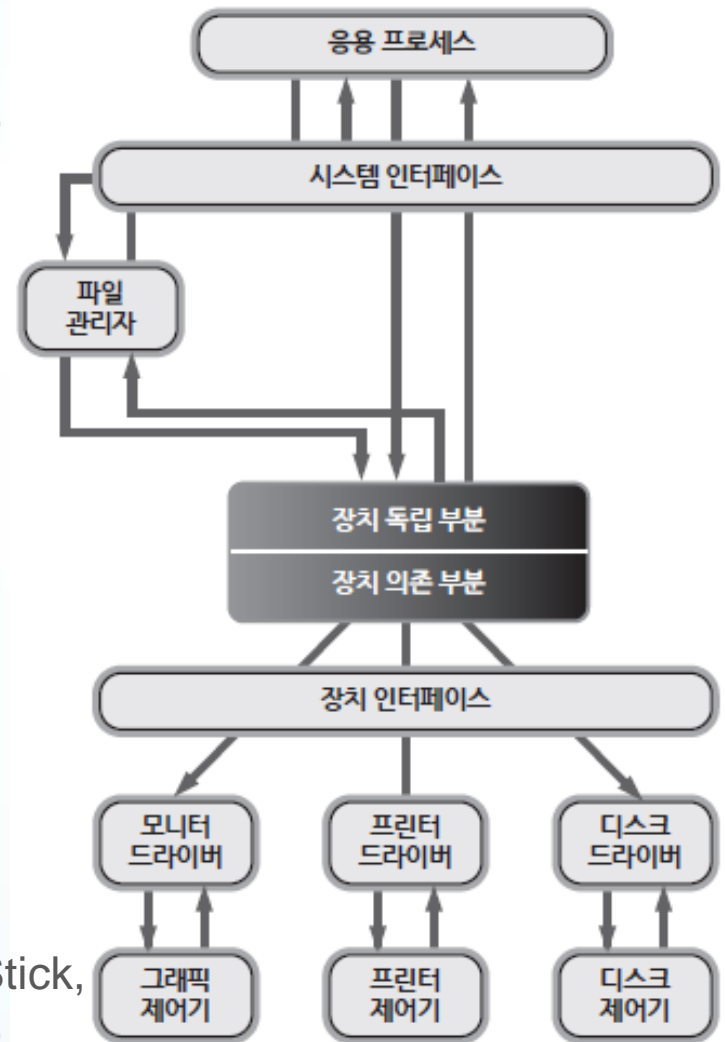


그림 2-6 장치 관리자

# 운영체제 서비스

## ■ Bootstrapping(=Booting) 서비스

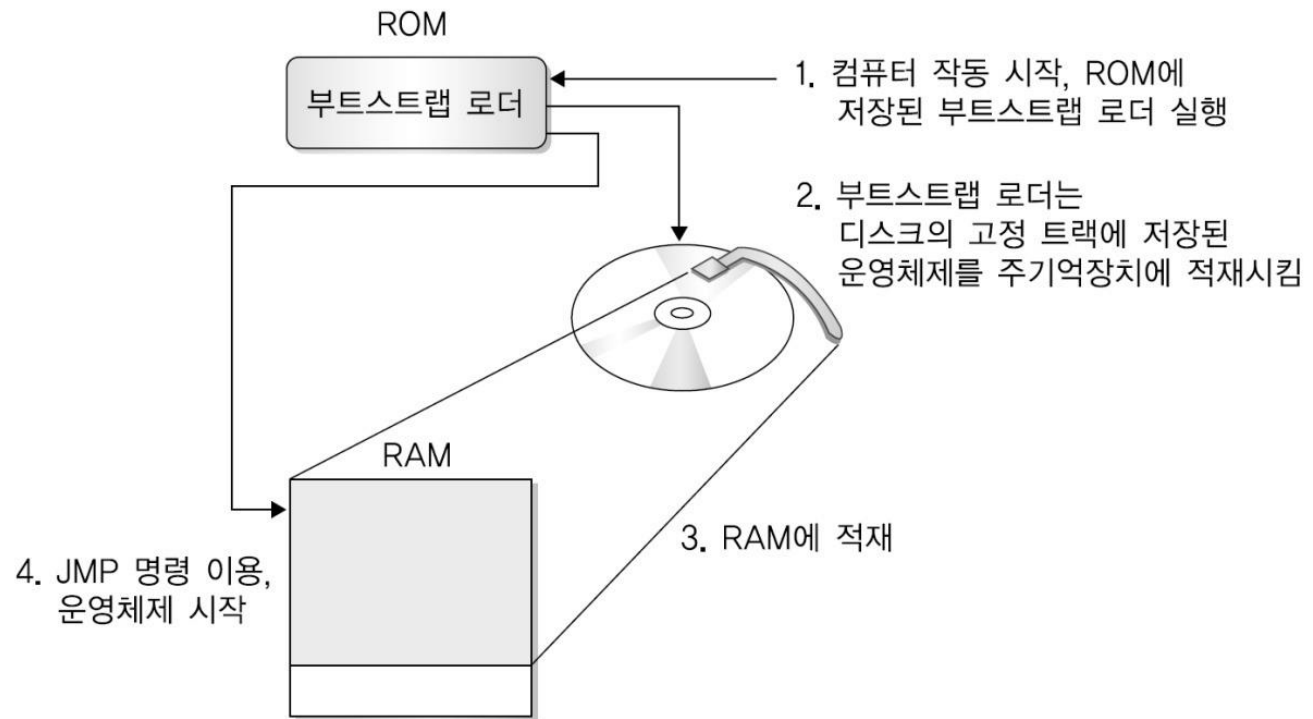
☑ Booting : 운영체제가 메모리로 로딩(적재)되는 과정을 부팅

☑ Boot Strap Loader (적재기) : (디스크 0트랙에 위치)

운영체제를 메모리로 로딩하는 프로그램

☑ User service

☑ System service

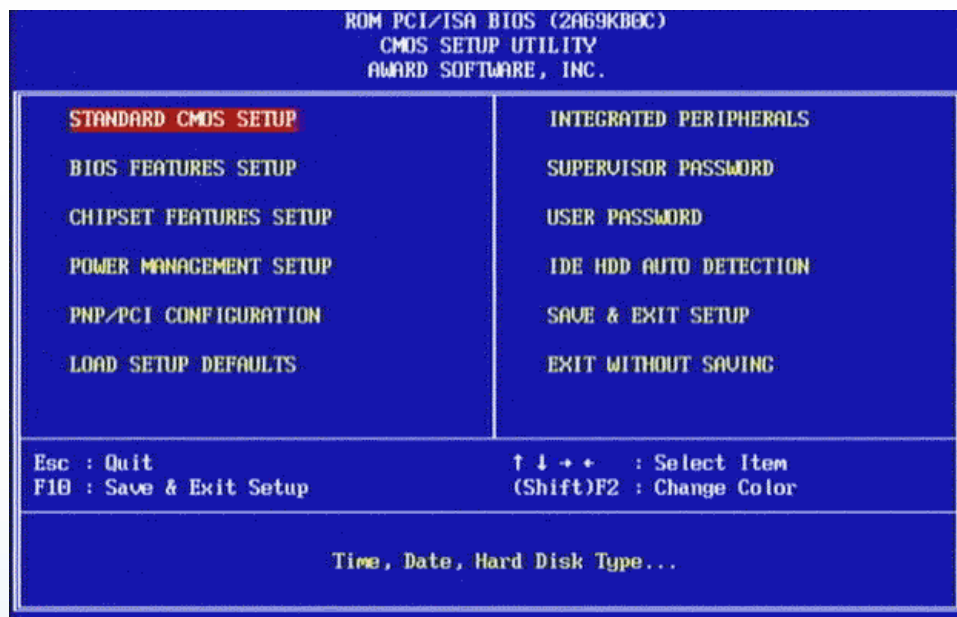


# 운영체제 서비스

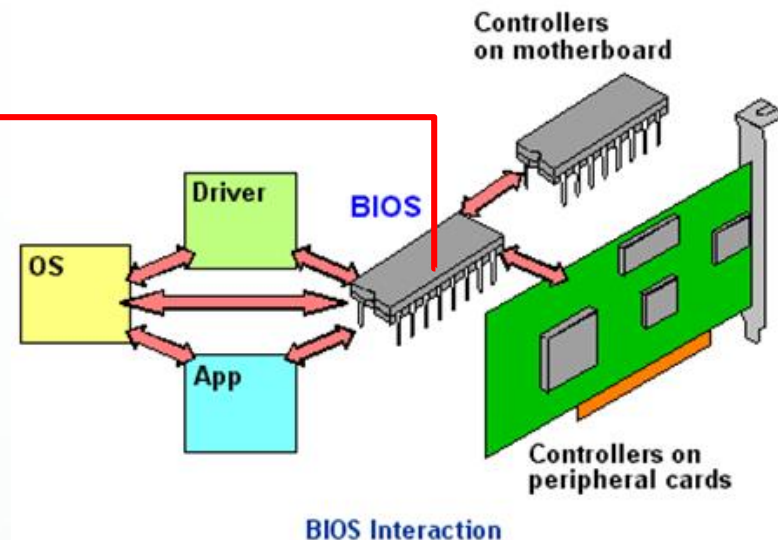
## ■ Bootstrapping(=Booting) 서비스

### ☑ Booting 절차

- 1) Power up
- 2) BIOS boot strap loader execution
- 3) HW check up
- 4) loading kernel OS into memory
- 5) execution OS



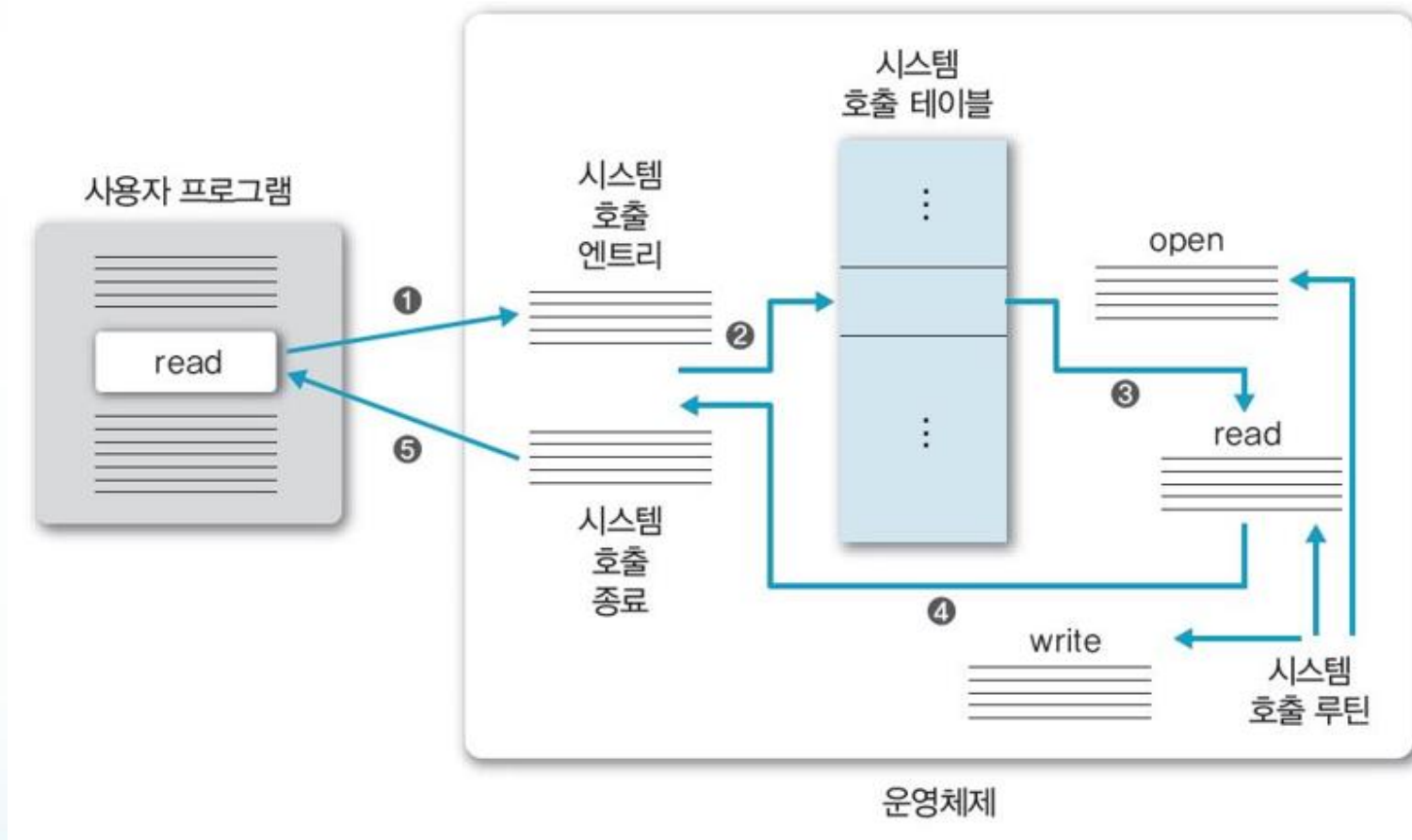
▲ CMOS BIOS setup 화면



# 운영체제 서비스

## ■ System Call 서비스

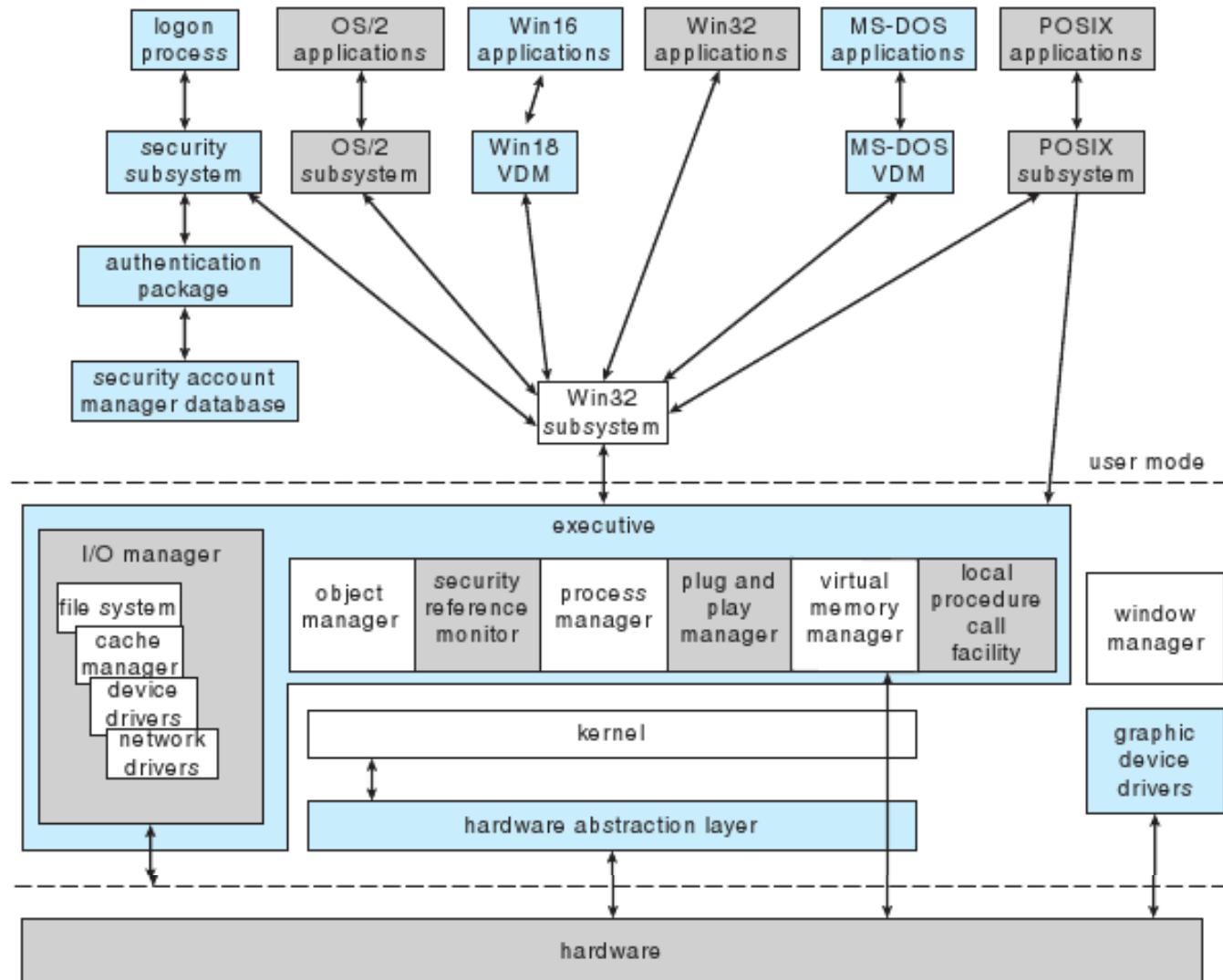
- ☑ 사용자 프로그램에서 자원(디스크, 메모리 등)을 사용하기 위한 API





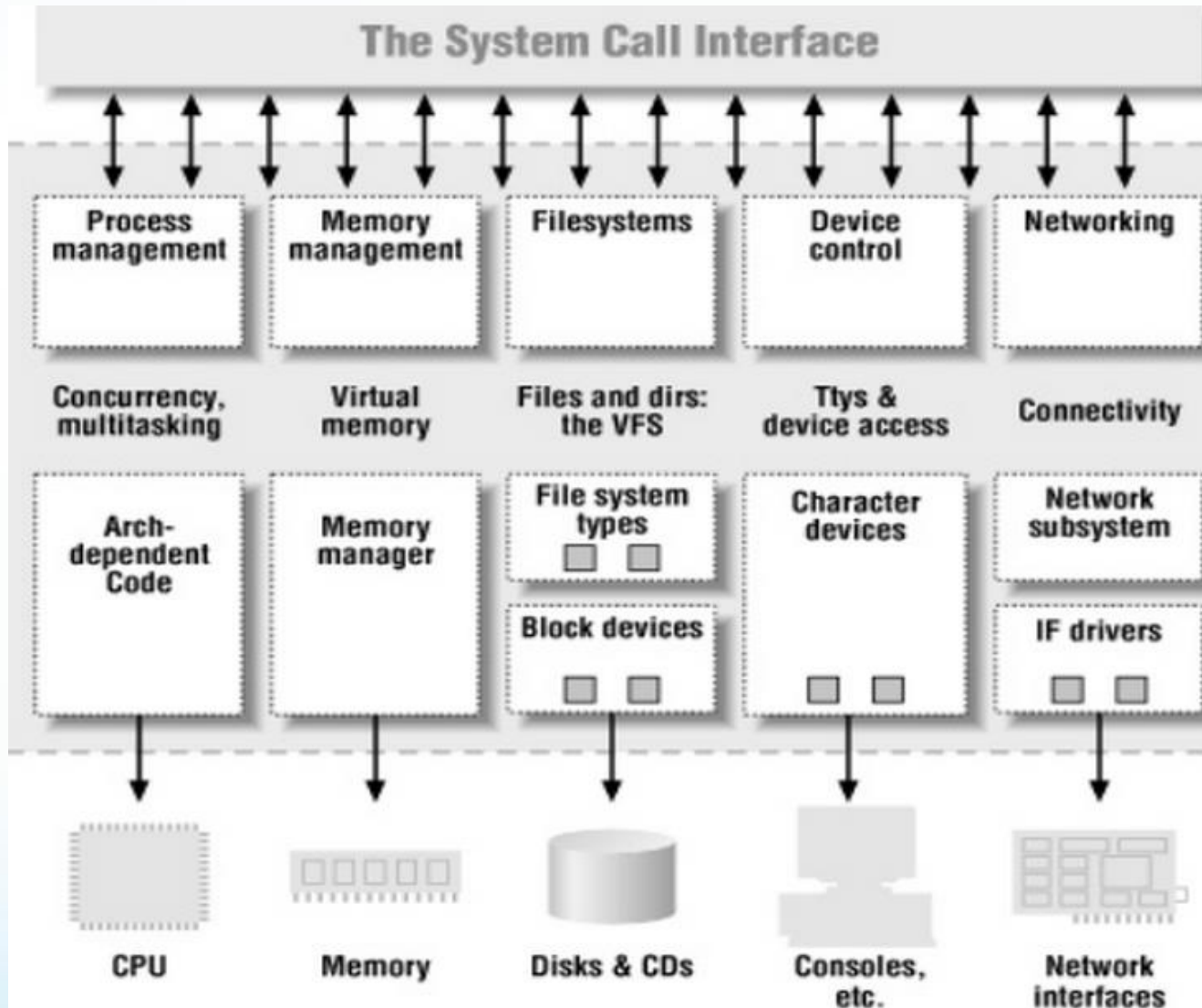
# 운영체제 구조

## ■ Windows 7/10 구조



# 운영체제 구조

## ■ Linux 구조



# 운영체제 구조

## ■ UNIX 구조



그림 2-20 단일 구조 운영체제 예 2 : 초기 유닉스

# 운영체제 구조

## ■ 계층적 구조(Hierarchical Architecture)

- ✓ 유사한 기능끼리 그룹으로 묶음 -> 계층화
- ✓ 계층은 모듈화, 계층간 독립성  
검증과 오류수정이 용이
- ✓ 계층이 많을 수록,  
성능이 저하가 예상

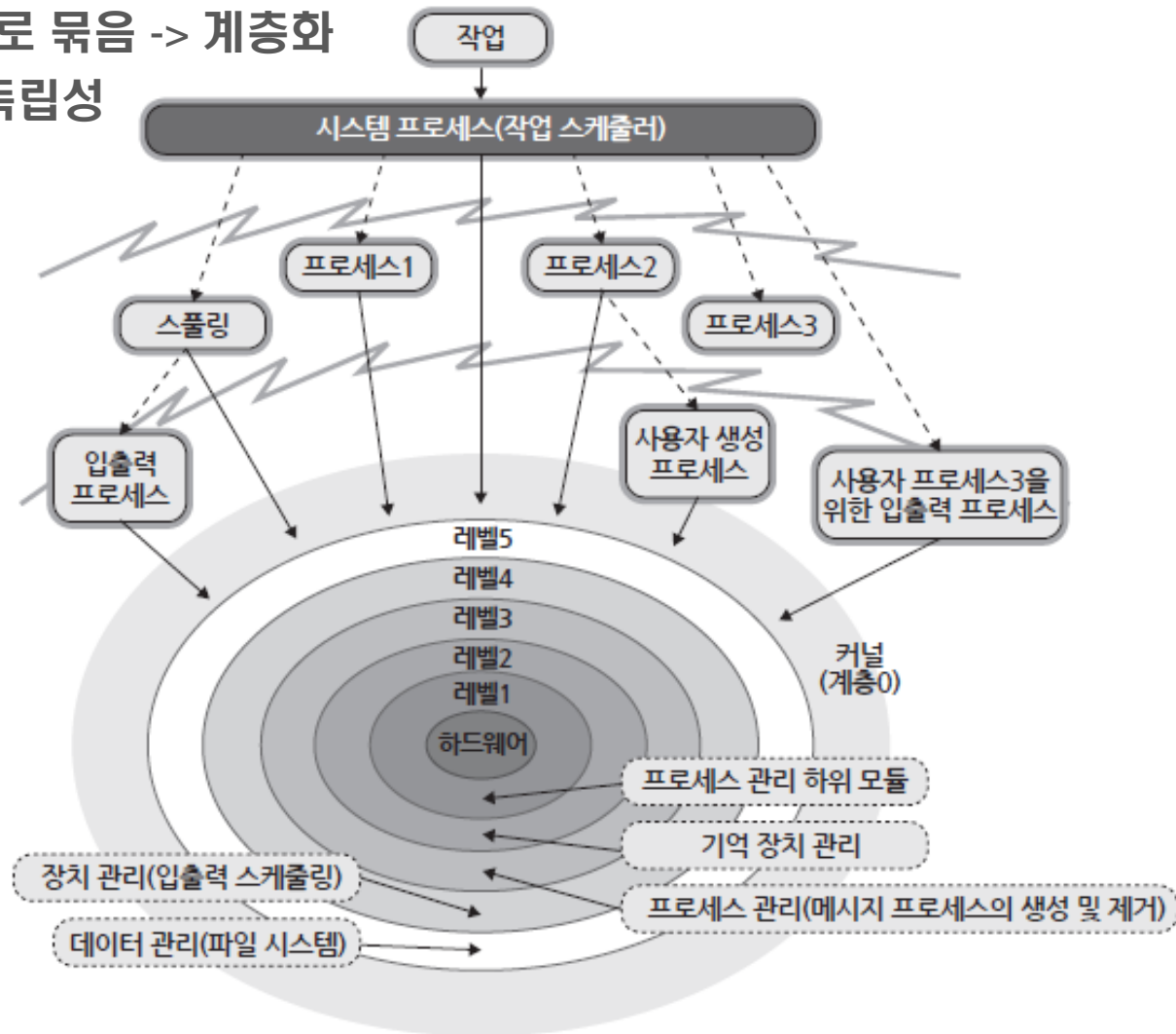
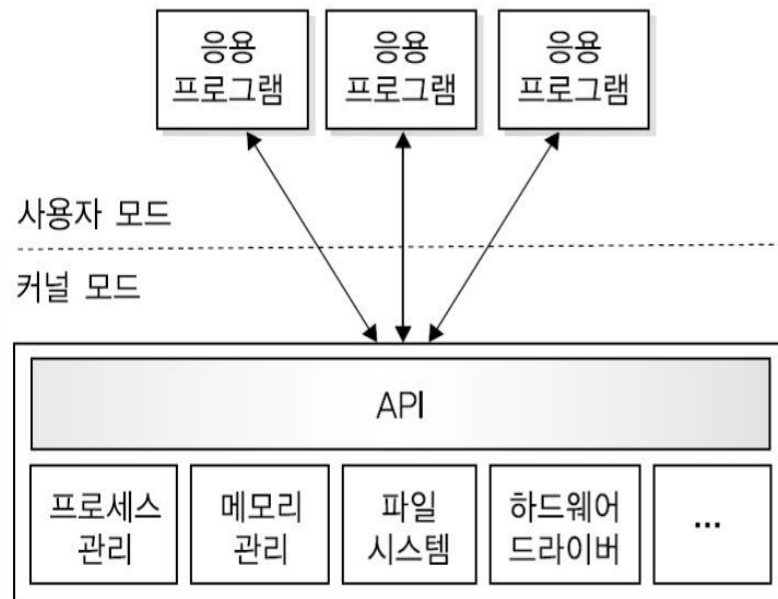


그림 2-8 계층 구조 관점에서의 운영 체제

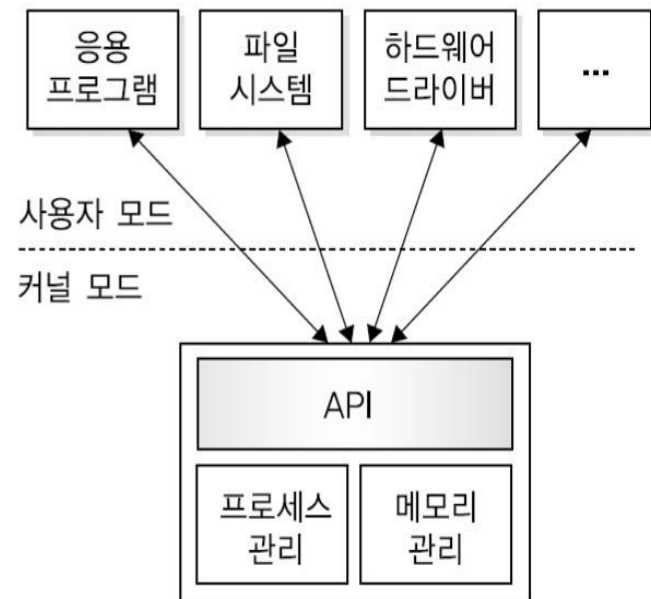
# 운영체제 구조

## ■ 커널구조에 따른 분류

- ☑ 커널(Kernel) : 메모리에 상주하는 운영체제의 핵심모듈(부분)
- ☑ 마이크로 커널(Micro Kernel)
- ☑ 단일 커널(Monolithic Kernel)



(a) 단일 커널



(b) 마이크로 커널

단일 커널과 마이크로 커널

# 운영체제 구조

## ■ 커널구조에 따른 분류

### ☑ 마이크로(Micro) 커널(Kernel)

#### 장점

1. 커널이 가볍다. (커널에서 필수 기능만 제공)
2. 한 부분에서 발생한 문제가 시스템 전체에 영향을 주지 않는다.
3. 서버의 개발이 용이하고 운영체제 기능의 변경이 쉽다.  
(많은 기능이 사용자 영역으로)
4. 실시간 시스템에 활용.  
(커널 내부에서 발생 지연이 작고 예측이 가능)

#### 단점

1. 프로세스 간 통신 발생을 최소화하여야
2. 속도가 느리다.  
(Context Switch time)

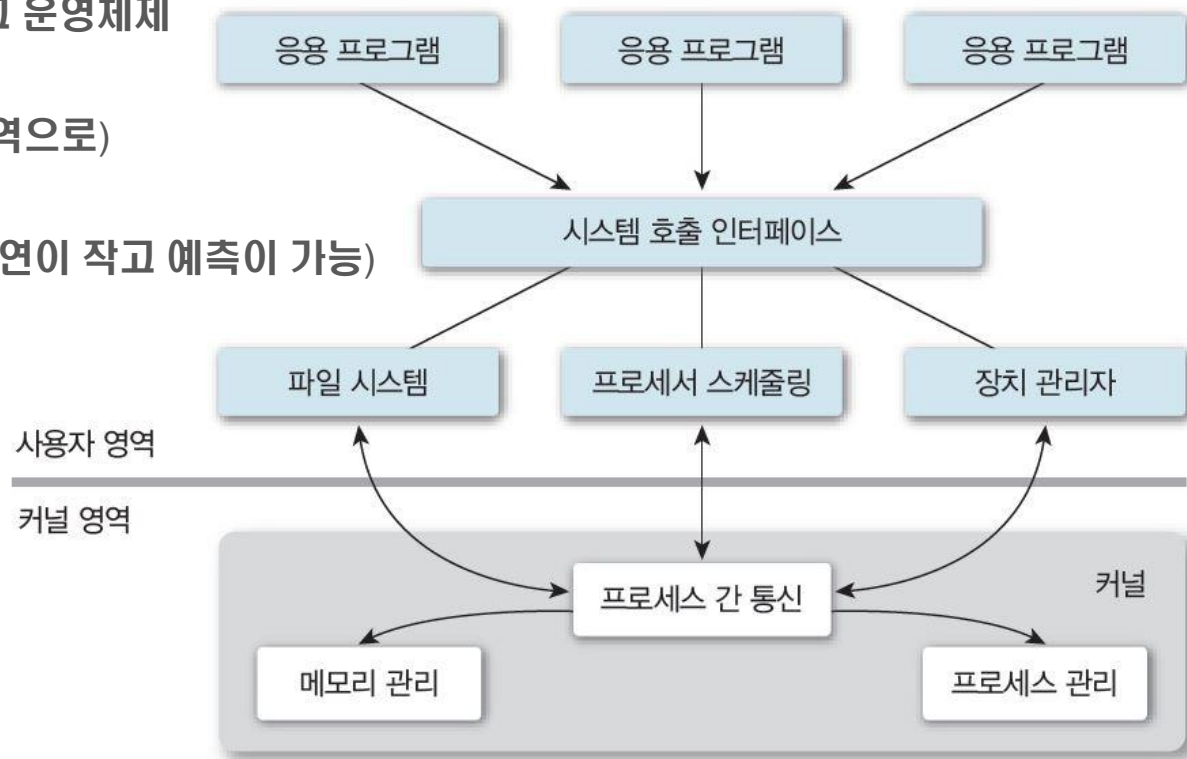


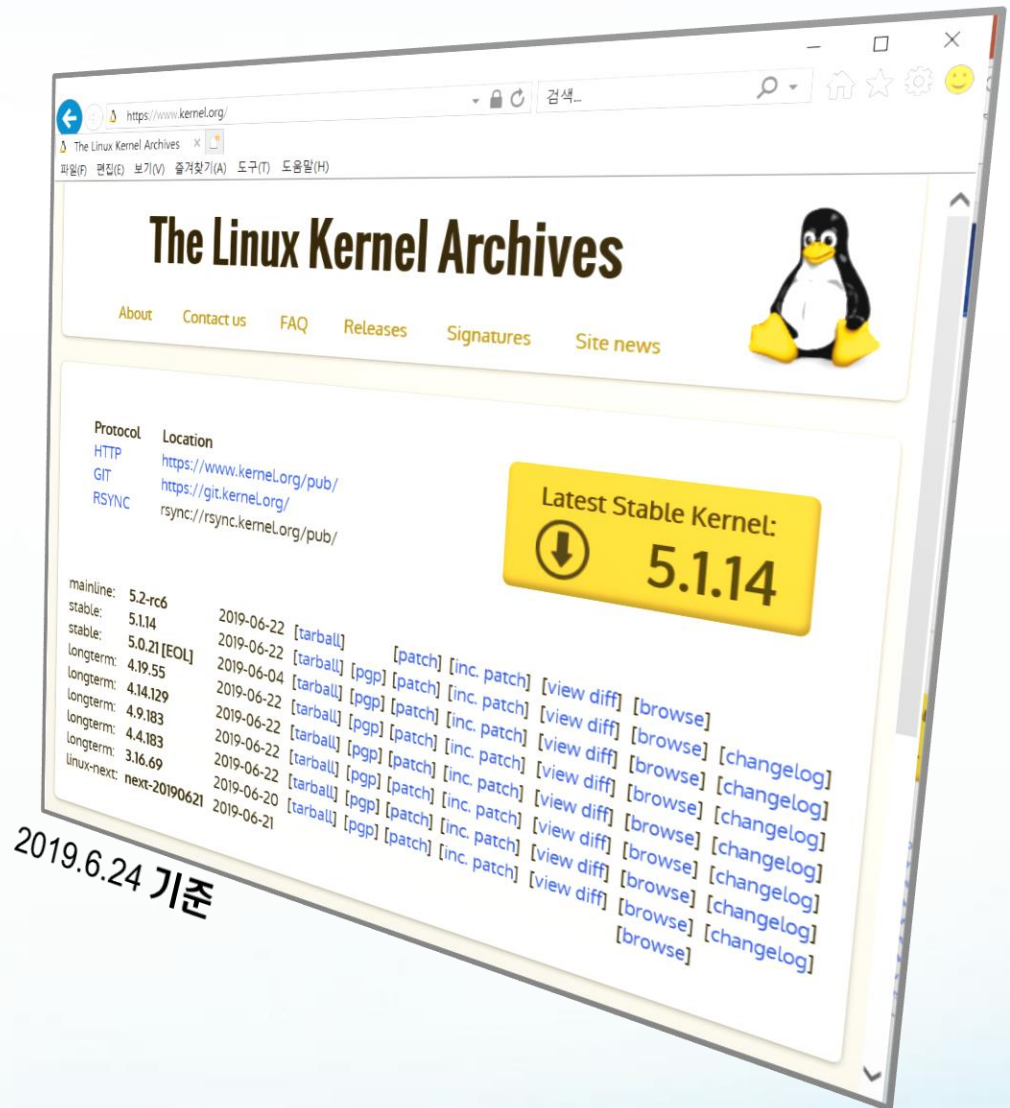
그림 2-23 마이크로 커널 구조 예



# 운영체제 구조

## ■ 리눅스 커널 (오픈소스)

☑ [www.kernel.org](https://www.kernel.org)

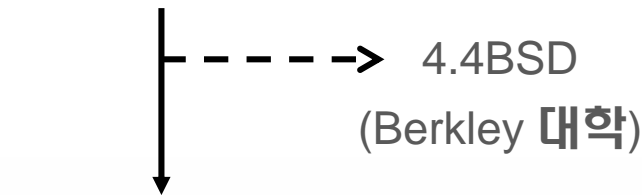


2019.6.24 기준

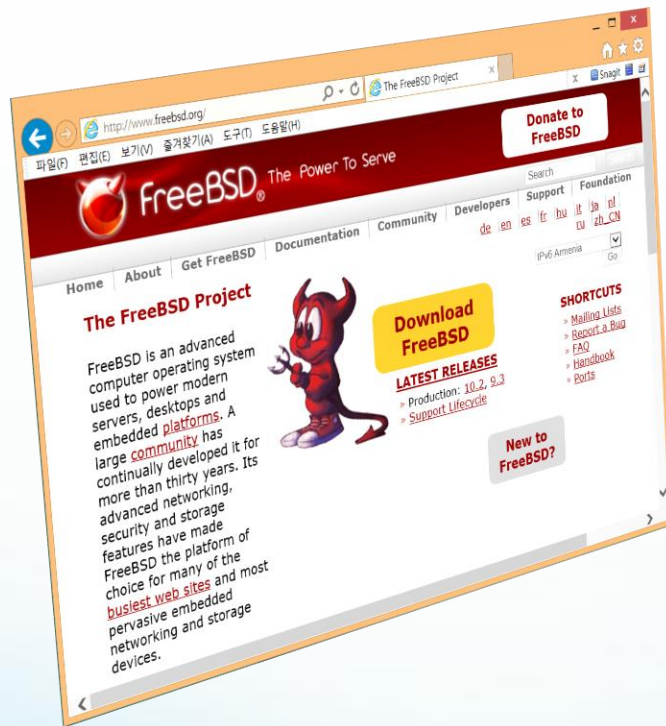
# 운영체제 구조

## ■ 유닉스(UNIX)

UNIX  
(AT&T)



- FreeBSD [www.freebsd.org](http://www.freebsd.org)
- OpenBSD [www.openbsd.org](http://www.openbsd.org)
- NetBSD [www.netbsd.org](http://www.netbsd.org)



# Q&A

---