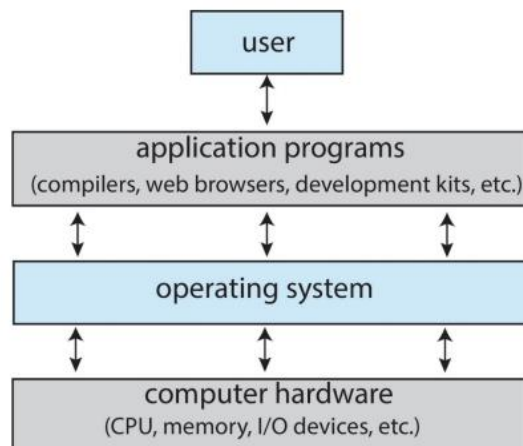


Os Introduction

Computer System Structure

컴퓨터 시스템은 4가지 구성 요소로 나눌 수 있다.

- 하드웨어(Hardware) – 기본 컴퓨팅 리소스 제공
 - CPU, 메모리, storage, I/O 디바이스
- 운영체제(Operating System)
 - 다양한 애플리케이션 및 사용자 간의 하드웨어 사용 제어 및 조정
 - Windows, MacOS, Linux 등
- 응용 프로그램(Application Programs)
 - 시스템 리소스를 사용하여 사용자의 컴퓨팅 문제를 해결하는 방법
 - 워드 프로세서, 컴파일러, 웹 브라우저, 데이터베이스 시스템, 비디오 게임
- 사용자
 - 사람, 기계, 컴퓨터 등



운영체제의 작업

- 사용자는 사용 편의성 및 우수한 성능을 원함 – 리소스 활용에 신경 쓰지 않음
- 운영체제는 하드웨어를 효율적으로 활용하고 사용자 프로그램의 실행을 관리하는 리소스 할당자 및 제어 프로그램.
- 모바일 장치는 사용성과 배터리 수명에 최적화되어 리소스가 부족

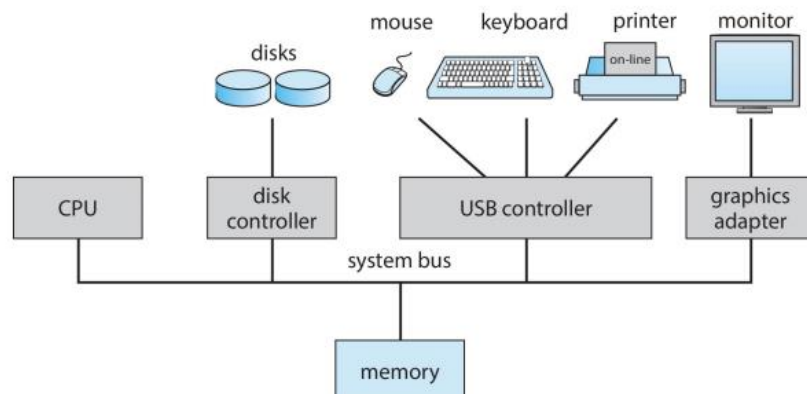
- 모바일 사용자 인터페이스 (터치 스크린, 음성 인식)
- 일부 컴퓨터는 사용자 인터페이스가 없음
- 사용자 개입 없이 실행

운영 체제의 정의

- **Kernel** : 컴퓨터에서 **항상 실행**되는 프로그램
- **Middleware**
 - 데이터베이스, 멀티미디어, 그래픽과 같은 애플리케이션 개발자에게 **추가 서비스를 제공하는 소프트웨어 프레임워크 세트**
 - 오늘날 범용 및 모바일 컴퓨팅용 OS에 포함
 - **iOS, Android**
- **OS**
 - 컴퓨터 **사용자**와 컴퓨터 **하드웨어** 사이의 **중개자 역할**을 하는 프로그램
 - 컴퓨터 **하드웨어, 소프트웨어 리소스를 관리**하고 컴퓨터 프로그램을 위한 **공통 서비스를 제공하는 시스템 소프트웨어**

컴퓨터 시스템 조직

- 컴퓨터 시스템 작동
 - 공유 메모리에 대한 액세스를 제공하는 공통 **버스**를 통해 하나 이상의 CPU, 장치 컨트롤러 연결
 - **메모리 사이클**을 위해 경쟁하는 CPU 및 장치의 **동시 실행**

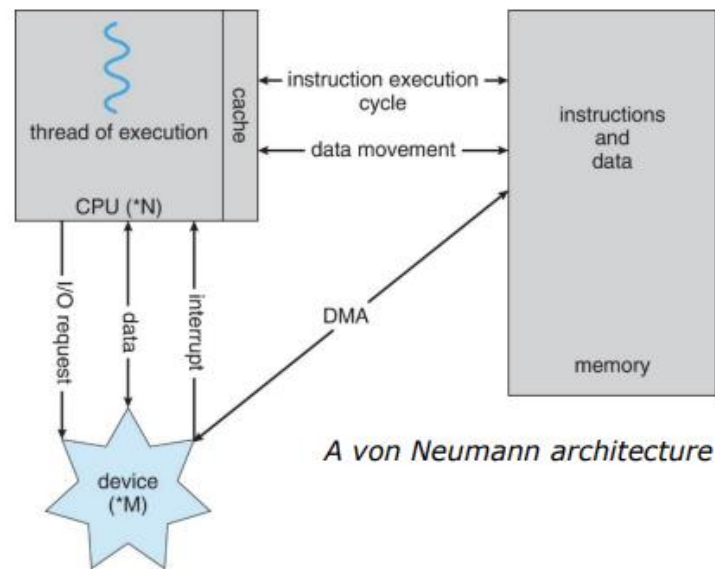


컴퓨터 시작 및 작동

- 시작

- 전원을 켜거나 재부팅할 때 부트스트랩 프로그램이 로드
- 일반적으로 ROM 또는 EPROM에 저장되며, 펌웨어(firmware)라고 한다.
- 시스템의 모든 측면을 초기화
- 운영체제 커널을 로드하고 실행 시작

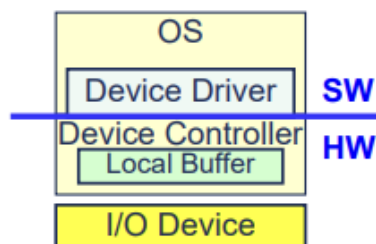
- 작동



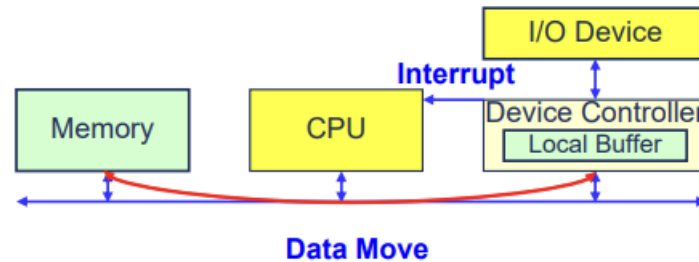
컴퓨터 시스템 동작

- 각 장치 컨트롤러

- 특정 I/O 장치 유형 담당
- I/O 데이터를 저장할 로컬 버퍼가 있음
- 관리할 운영체제 장치 드라이버(device driver)가 있음.
- 장치 드라이버(device driver) : 컨트롤러와 커널 사이에 균일한 인터페이스 제공

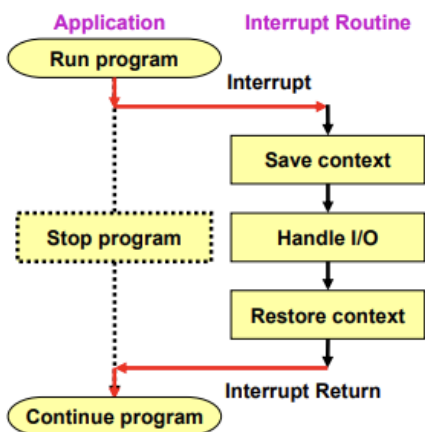


- CPU가 메인 메모리에서 로컬 버퍼로 데이터를 이동하거나 로컬 버퍼에서 데이터를 이동
- I/O가 컨트롤러의 로컬 버퍼에서 디바이스로 송수신됨
- 디바이스 컨트롤러가 인터럽트를 발생시켜 CPU에 작업을 완료했음을 알림



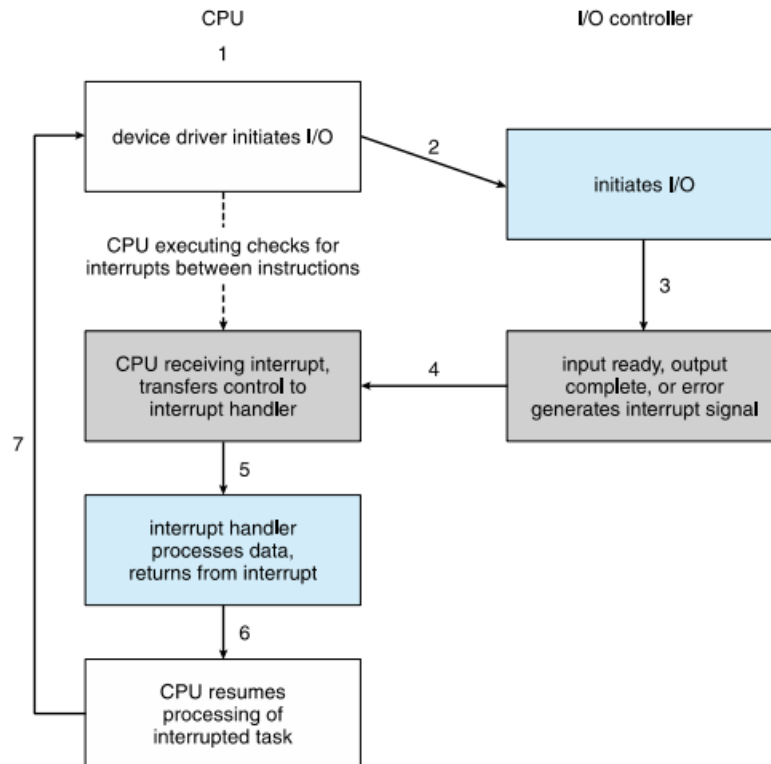
인터럽트 처리 (Interrupt Handling)

- 인터럽트가 ISR(인터럽트 서비스 루틴)에 제어 권한을 전달
- 인터럽트 벡터를 통해 (또는 소프트웨어에 의해) 모든 ISR의 주소를 포함
- 인터럽트 아키텍처는 인터럽트된 명령의 주소를 저장
- OS는 CPU 레지스터와 PC를 저장하여 CPU 상태(context)를 보존
- 코드는 각 인터럽트 유형에 대해 취해야 할 조치를 결정



- trap, exception
- 오류 또는 사용자 요청에 의해 발생하는 소프트웨어 생성 인터럽트
- 운영체제는 interrupt driven

Interrupt-driven I/O Cycle



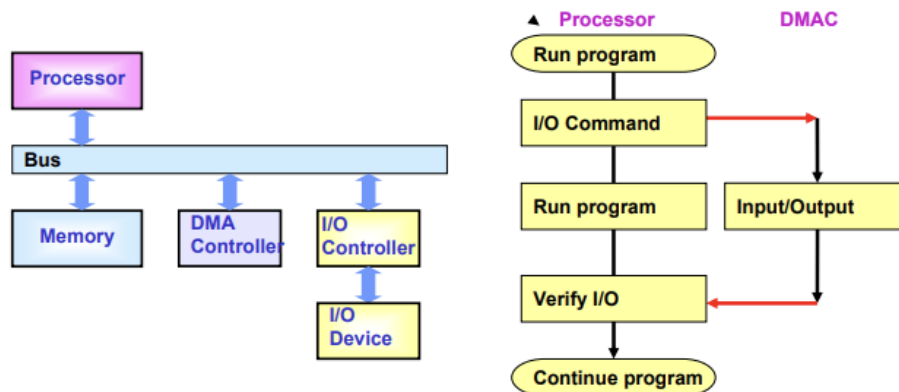
두 개의 I/O 구조

- I/O가 시작된 후 I/O가 완료될 때 사용자 프로그램으로 돌아감
 - 다음 인터럽트가 발생할 때까지 CPU를 유휴 상태로 유지 : Wait instruction
 - Wait loop : 메모리 액세스에 대한 경합
 - 한 번에 최대 한 개의 I/O 요청이 처리되고 동시에 처리되지 않음
- I/O가 시작되면 I/O가 완료될 때까지 기다리지 않고 사용자 프로그램으로 돌아감
 - System call : 사용자가 I/O 완료를 기다릴 수 있도록 OS에 요청
 - Device-status table에는 각 I/O 디바이스의 유형, 주소 및 상태를 나타내는 항목이 포함됨.
 - 장치 상태를 확인하고 인터럽트를 포함하도록 테이블 항목을 수정하기 위해 I/O 장치 테이블로 OS 인덱싱

Direct Memory Access Structure

- 메모리 속도에 가까운 속도로 정보를 전송할 수 있는 고속 I/O 장치에 사용

- 장치 컨트롤러는 CPU 개입 없이 버퍼 스토리지에서 메인 메모리로 데이터 블록을 직접 전송
- 블록당 하나의 인터럽트만 생성



Storage Structure

- Main Memory - CPU가 직접 액세스할 수 있는 대용량 스토리지 미디어
 - Random access
 - 휘발성(volatile)
 - DRAM(Dynamic Random-Access Memory) 형태로 사용
- Secondary storage - 대용량 비휘발성 스토리지 용량을 제공하는 메인 메모리 확장
 - 하드 디스크 드라이브(HDD) - 자기 기록 재료로 덮인 단단한 금속 또는 유리 플래터 (platter)
 - 디스크 표면은 논리적으로 트랙으로 분할되고 섹터로 세분화
 - 디스크 컨트롤러는 장치와 컴퓨터 간의 논리적 상호 작용을 결정
 - 비휘발성 메모리(NVM) 장치 - 하드 디스크보다 빠름
 - SSD(Solid-State Disk)
 - NAND 플래시 메모리
 - 용량 및 성능이 증가함에 따라 가격이 하락
 - 광 디스크
 - CD, DVD, BD

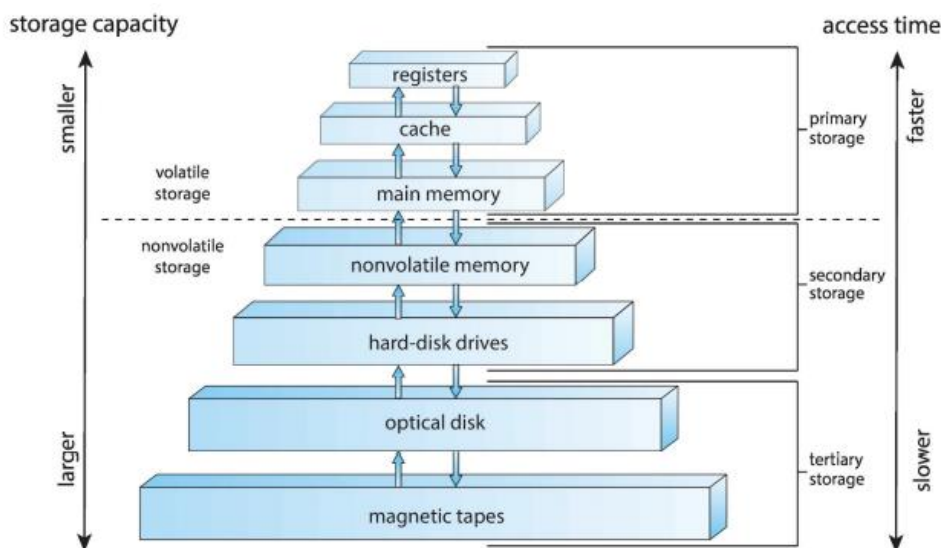
Storage 표기법

- Bit: 컴퓨터 저장소의 기본 단위

- 0과 1의 두 값 중 하나를 포함.
- **Byte**: 8비트, 가장 작은 편리한 스토리지 청크
- **Word**: 주어진 컴퓨터 아키텍처의 기본 데이터 단위
- 1바이트 이상: 64비트 아키텍처 컴퓨터에 64비트(8바이트) 단어가 있음
- 한 번에 바이트가 아닌 네이티브 워드 크기로 많은 작업을 수행

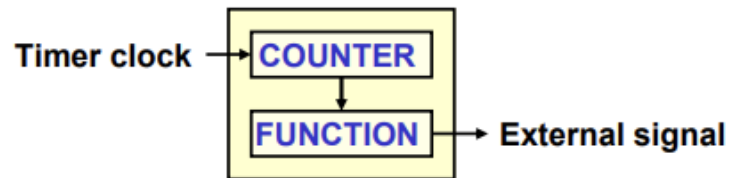
Storage Hierarchy

- 계층 구조로 구성된 스토리지 시스템
- 속도, 비용, 휘발성
- **Caching** – 정보를 더 빠른 스토리지 시스템에 복사, 메인 메모리를 보조 스토리지용 캐시로 볼 수 있음



Timer

- 무한 루프에 빠지거나 시스템 서비스를 호출하지 못하고 운영 체제에 제어 권한을 반환하지 않도록 방지하려면
- **Timer clock**: 수정 발진기(crystal oscillator)에서 생성됨
- **Counter**: 위 또는 아래
- 기능: interrupt, reset, 일회성 또는 주기적, 주파수 분할기
- 외부 신호: interrupt, reset, 프로그래밍된 신호



- OS에 의해 일정 시간 후 컴퓨터를 중단하도록 타이머가 설정됨
 - Linux에서, 일반적인 기간은 4 mS입니다.
 - 카운터가 타이머 클럭에 의해 감소합니다.
 - 카운터 0이 인터럽트를 생성할 때
- 제어권을 되찾거나 할당된 시간을 초과하는 프로그램을 종료하도록 프로세스 예약 전에 설정