



11. 입출력 구조

목표

- 입출력 구조를 이해하고, 입출력 동작 방법을 이해한다.



입출력 인터페이스



입출력 서브 시스템

- 외부 세계와의 효율적인 통신 방법 제공

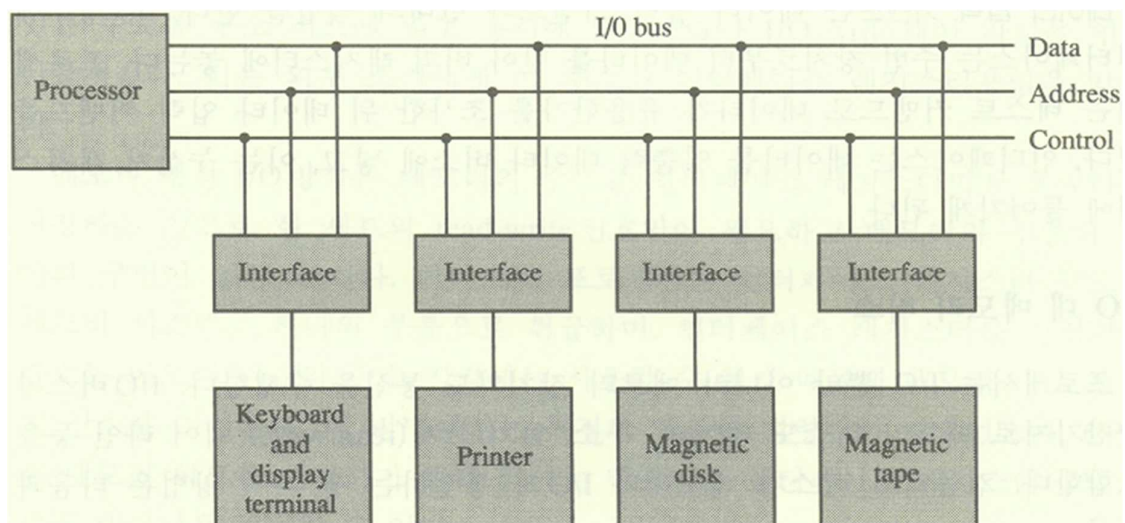


입출력 인터페이스

- Interface - 프로세서와 주변 장치 연결
- CPU-입출력장치간 이진 정보 전송 방법 제공
- CPU-입출력장치간의 차이점 해소
 - 동작 방식
 - 전송 속도
 - 데이터 코드와 워드 형식 등...



버스와 인터페이스



입출력 동작

- 1 : CPU는 IO 장치 주소를 버스에 적재
- 2 : CPU는 IO 기능코드를 제어 버스에 적재
- 3 : 인터페이스는 주소 검출
- 4 : IO 장치는 IO 명령에 따라 IO 동작 실행



IO 장치 주소 지정 방식

- Isolated IO
- Memory-Mapped IO

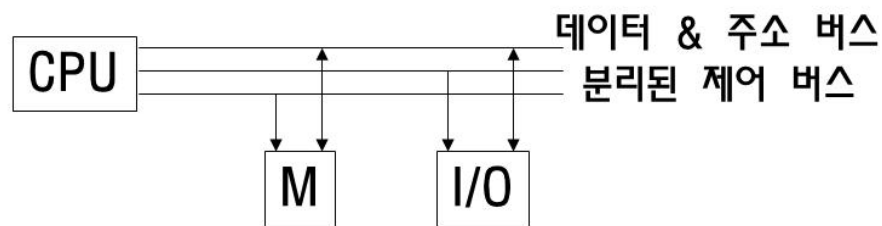
입출력 동작

- 1 : CPU는 IO 장치 주소를 버스에 적재
- 2 : CPU는 IO 기능코드를 제어 버스에 적재
- 3 : 인터페이스는 주소 검출
- 4 : IO 장치는 IO 명령에 따라 IO 동작 실행



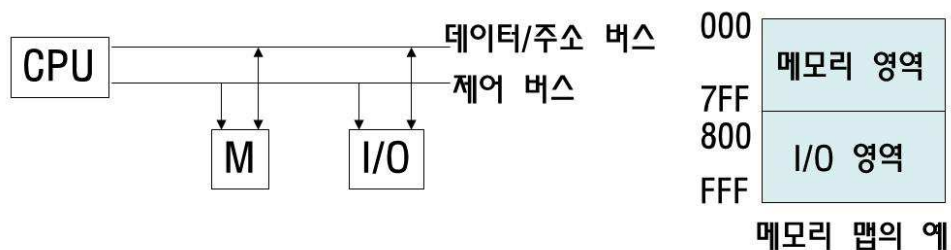
Isolated IO

- 격리형 IO
- 메모리와 IO 장치 제어 라인 분리
- 메모리와 IO 장치의 주소 공간 분리
- 각각 접근 명령어 별도 존재

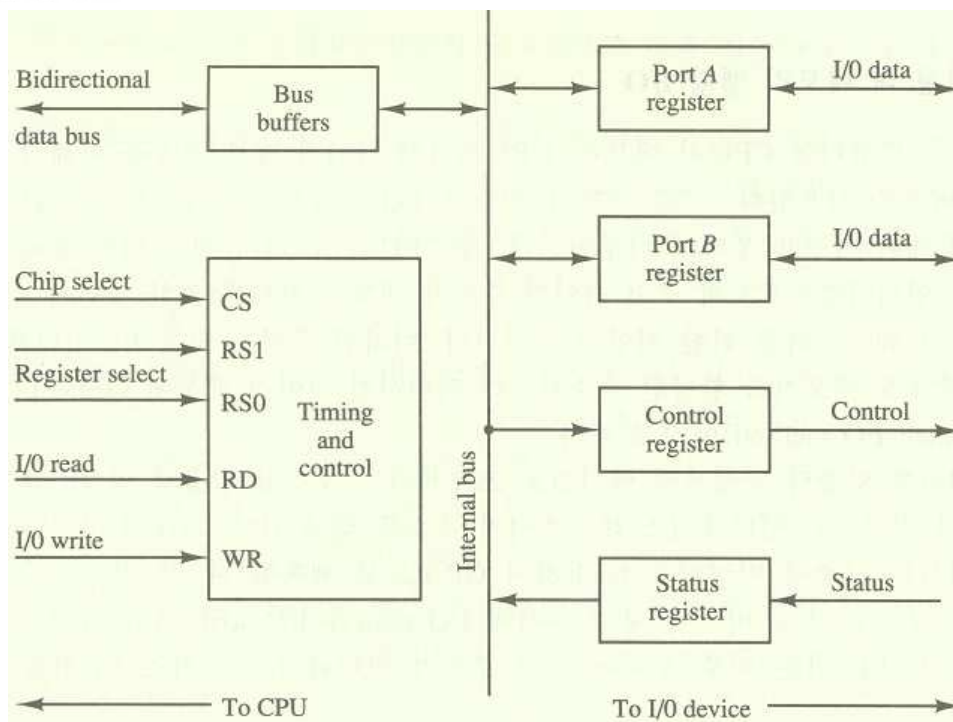


Memory-Mapped IO

- 기억장치 사상형 IO
- 메모리와 IO 장치가 주소 공간 공유
- 메모리와 IO 장치 접근 명령어 분리 불필요



인터페이스 구조



인터페이스 구조

- 주소 버스를 통해 사용할 레지스터 선택
- 선택된 레지스터는 데이터 버스를 통해 데이터 및 IO 명령 전송
- IO 명령은 제어 레지스터에 전송

전송 모드



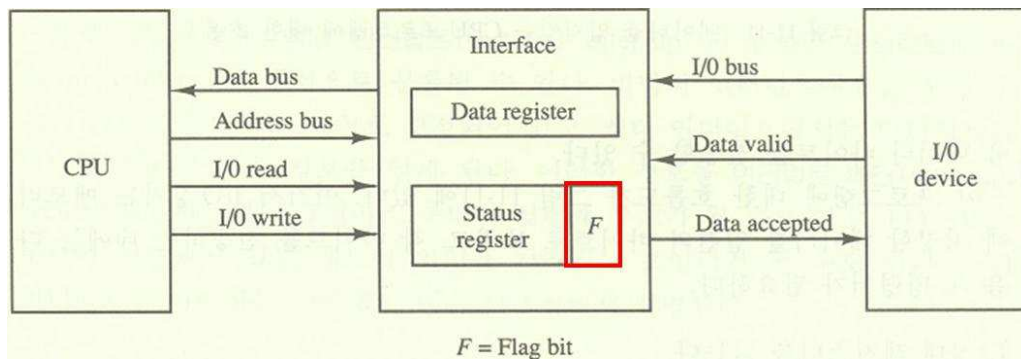
전송 방법

- 프로그램된 IO
- 인터럽트 IO
- DMA
- IOP

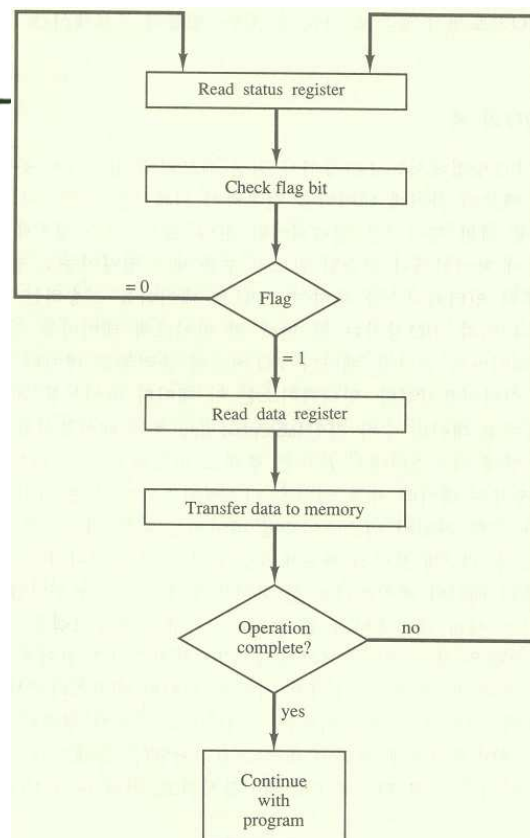


프로그래밍된 IO

- CPU가 IO 장치 상태를 **지속적으로 CHECK**
- IO 장치가 사용 가능할 때 데이터 전송



프로그래밍된 IO



프로그램된 IO

- IO 제어 간단
- 비효율적 자원 이용
 - CPU가 IO 장치를 기다림



인터럽트 IO

- Interrupt IO
- CPU가 IO 장치의 상태를 계속 CHECK X
- IO ready된 장치는CPU에 인터럽트 신호 전송



인터럽트 처리

- CPU의 현재 상태 저장
 - 모든 레지스터
 - 보통 스택에 저장
- 인터럽트 처리 루틴(ISR) 수행
- CPU의 이전 상태 복구



인터럽트 처리 루틴 결정

- 비벡터 인터럽트
 - 모든 인터럽트 처리는 하나의 루틴
- 벡터 인터럽트, vector interrupt
 - 발생한 인터럽트 종류에 따라 ISR 결정
 - I/O 장치는 인터럽트 벡터(또는 관련 정보) 전송함



우선 순위 인터럽트



인터럽트 IO 장치 결정

- 폴링(polling)
- 데이지 체인
- 병렬 우선순위

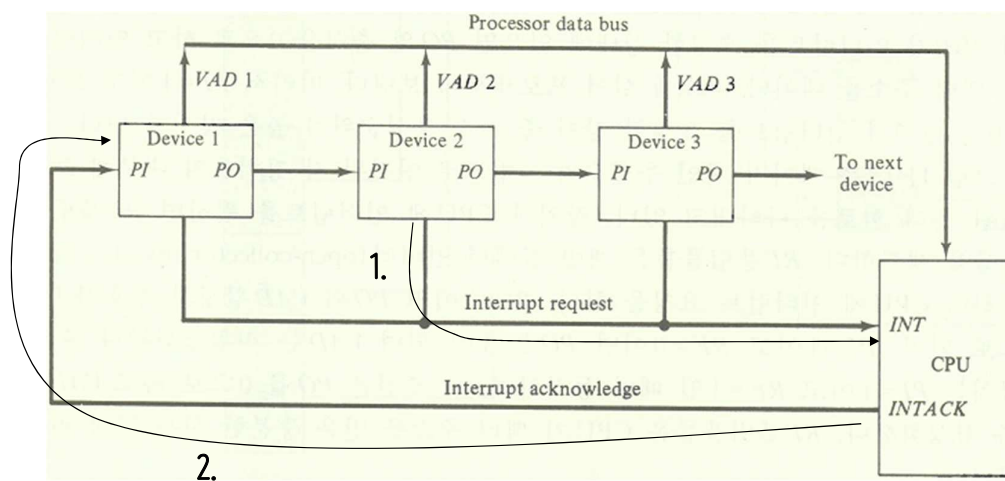


폴링

- 설정된 순서에 따라 장치에 인터럽트 질의
- SW적 방법
- 설정된 순서 = 장치 우선순위
- 간단
- 많은 인터럽트 상황에서 적절한 처리 어려움

데이지 체인, Daisy Chain

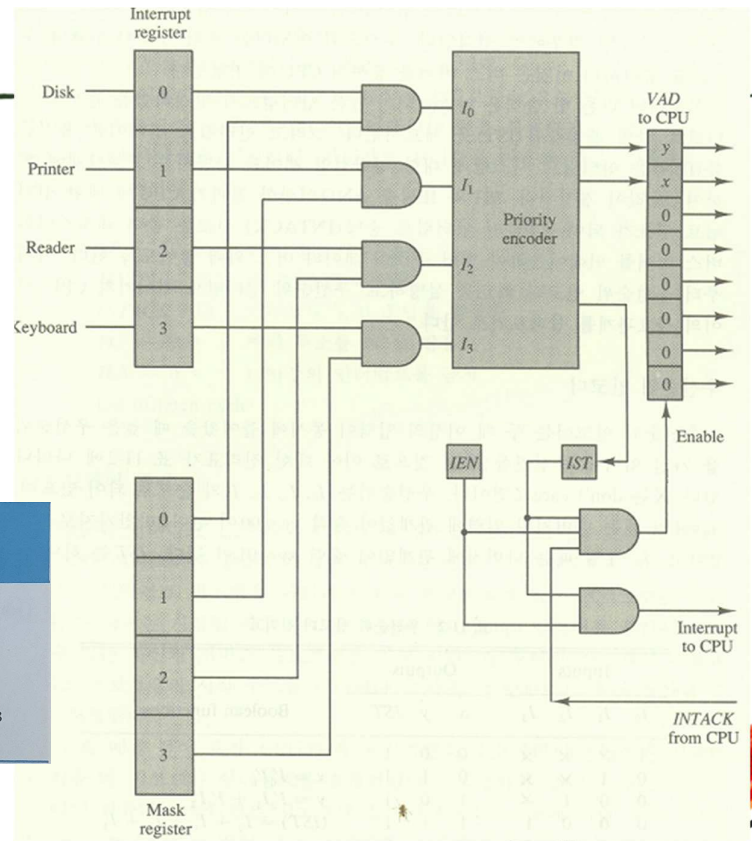
- 장치 우선순위에 따라 직렬로 연결



병렬 우선 순위

- 각 장치마다 1 비트 할당
- 인코더의 출력은 VAD(인터럽트 벡터 주소)의 2 비트

Inputs				Outputs			Boolean functions
I_0	I_1	I_2	I_3	x	y	IST	
1	x	x	x	0	0	1	$x = I_0' I_1'$
0	1	x	x	0	1	1	
0	0	1	x	1	0	1	$y = I_0' I_1 + I_0' I_2'$
0	0	0	1	1	1	1	
0	0	0	0	x	x	0	$(IST) = I_0 + I_1 + I_2 + I_3$



직접메모리 접근



인터럽트 IO의 문제점

- 한 워드 IO마다 인터럽트 발생
- 대량의 데이터 IO의 경우 과한 인터럽트 발생

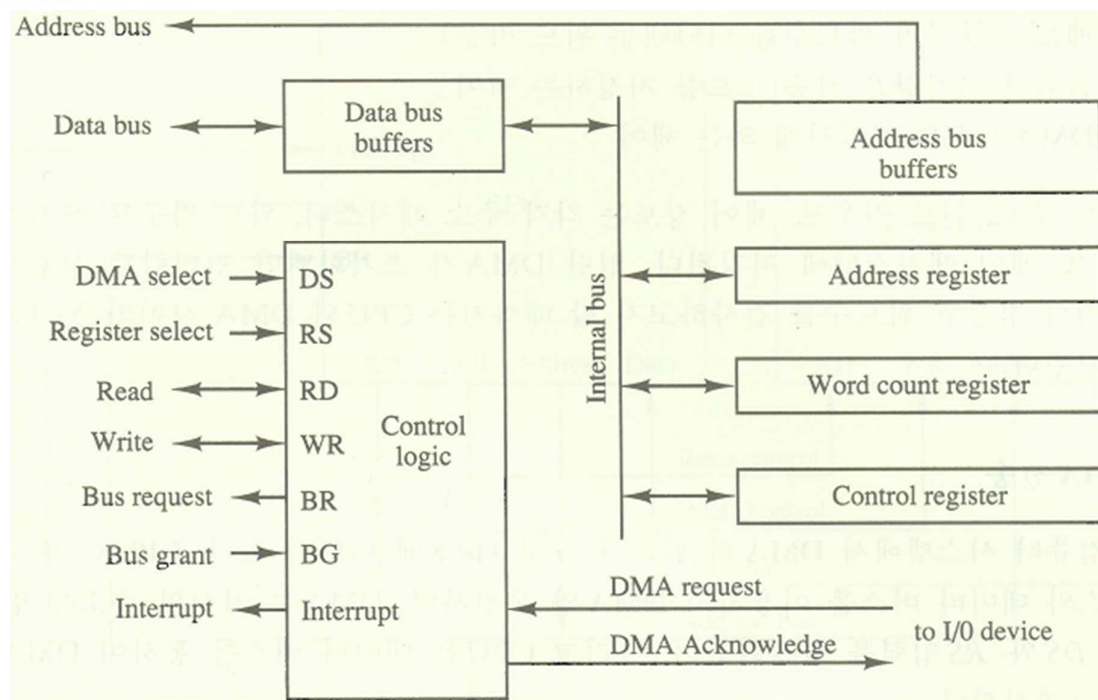


DMA, Direct Memory Access

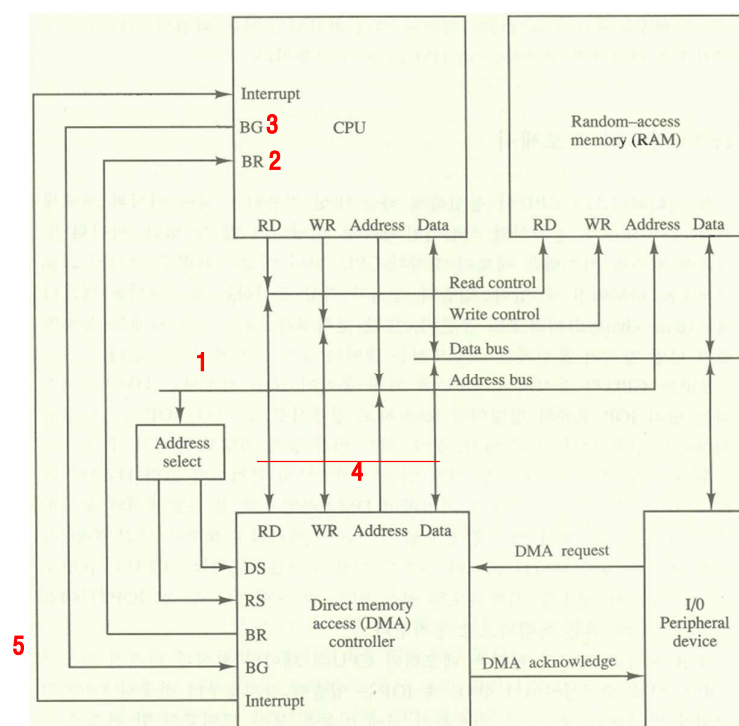
- IO 장치(또는 인터페이스)가 직접 메모리 접근
- CPU 간섭없이 데이터 IO 가능
- 메모리-IO 장치간 데이터 전송



DMA 제어기



DMA-CPU-Memory 연결



입출력 프로세서

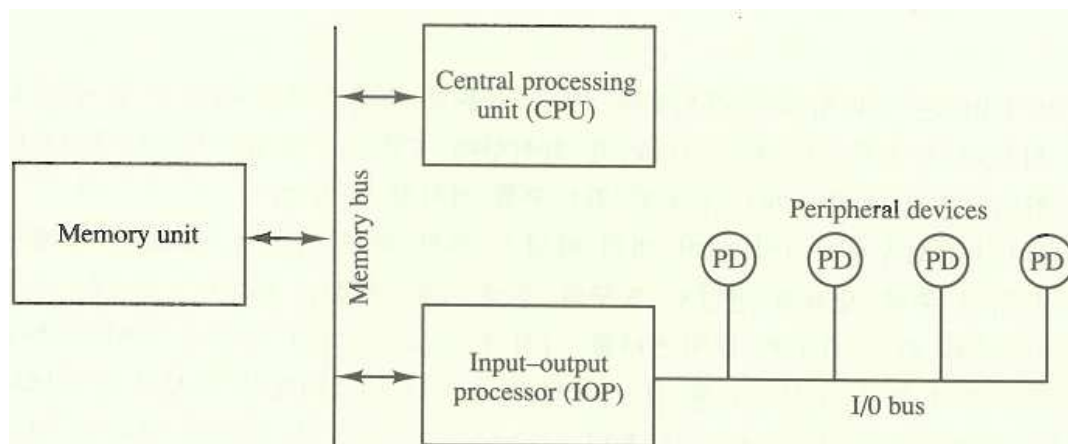


IOP, 입출력 프로세서

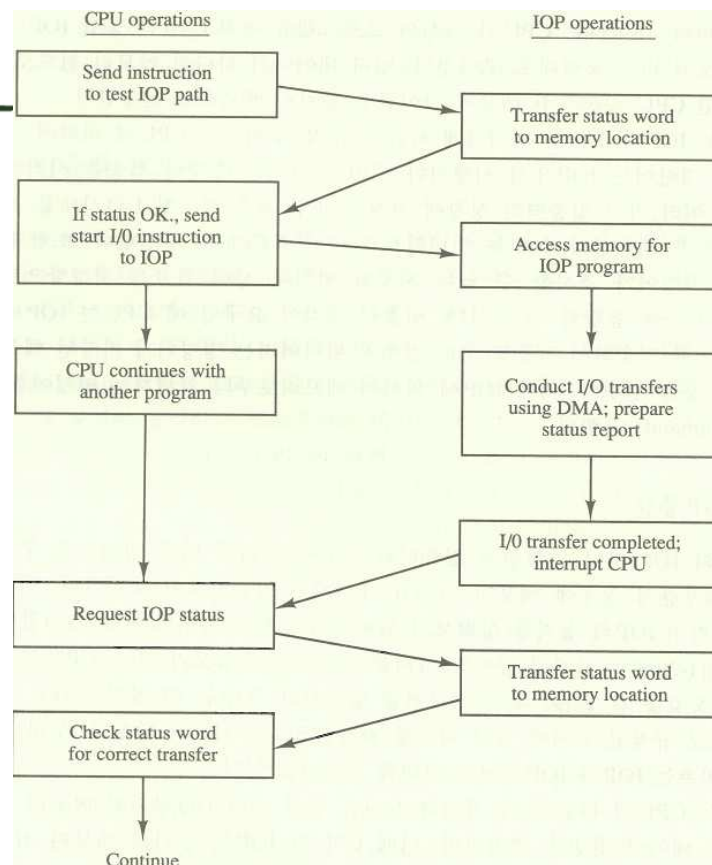
- IO를 전담하는 프로세서
- 데이터 전송 시 DMA 기능
- IO 프로그램 실행
 - 데이터 변환 등...



IOP-CPU



IOP-CPU 통신



Q&A

