Interrupt (계속)

Interrupt

- 외부 인터럽트
  - 인터럽트의 종류
    - ∗ direct interrupt
      1. CPU 에 인터럽트 신호가 직접들어감
      2. CPU 내부의 인터럽트 레지스터 사용 벡터 결정
      3. ISR 수행
      4. 인터럽트 처리가 빠르다 (ack. 싸이클이 필요없음)
    - ∗ indirect interrupt
      1. CPU 에 인터럽트 요청
      2. CPU 가 인터럽트 acknowledge 할 때 해당벡터 제공
      3. 인터럽트 처리가 느리다 (ack. 싸이클 필요)
  - 인터럽트
    - ∗ 외부 장치에서 신호발생 (interrupt request)

Interrupt (계속)

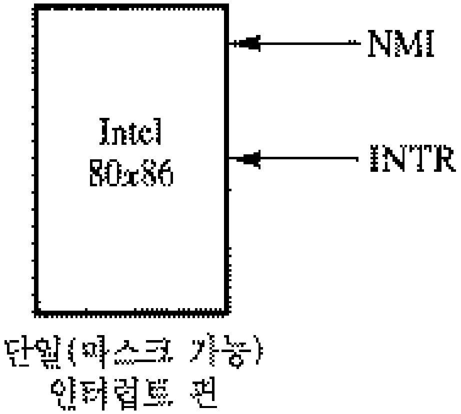
* CPU 인터럽트 인지 및 ACK (interrupt acknowledge)
* 현재 수행중이던 프로그램 정보 저장
* 인터럽트 서비스 루틴 처리
* 수행중이던 프로그램으로 복귀

– 마스크 불가능 인터럽트 (Nonmaskable Interrupt (NMI))

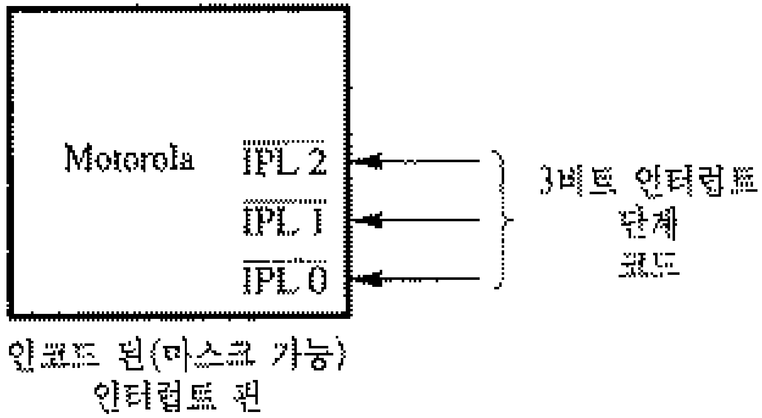
* 소프트웨어로 마스크 불가능한 인터럽트
* 대부분의 MPU에서 INTA 생성필요 없음 (autovectored interrupt)
* 미리 지정된 메모리로 이동
* 이미 진행중이던 인터럽트도 무시될수 있음

– 마스크 가능 혹은 사용자 인터럽트 (그림 7.3)

Interrupt (계속)



(a)Intel 80 x 86의 벡터 인터럽트



(b)Motorola의 벡터/자동벡터 인터럽트

  * Intel 계열 : INTR (범용 인터럽트 입력핀)
  * Motorola 계열: IPL0# ∼ IPL2# 의 인코드된 입력핀 사용 (000: NMI)
- 벡터 인터럽트
  * 대부분의 프로세서가 사용
  * 인터럽트 요청 (IREQ)
  * MPU 가 갖추어야할 기능들
    1. 실행중이던 프로그램으로의 복귀 기능

Interrupt (계속)

2. 여러 장치중 인터럽트 발생장치 확인 기능
3. 여러장치에서 동시에 인터럽트 발생시 우선순위 정하는 기능

∗ 인터럽트 종류

1. 외부 I/O (하드웨어) 인터럽트
2. 소프트웨어 인터럽트나 예외
3. 내부 하드웨어 예외 ex) divide by zero

∗ 우선순위 (그림 7.4)

Interrupt (계속)

1. 클락 사이클의 끝에서:
   - 리세트
   - 주소 오류
   - 버스 오류 (BERR)
2. 버스 사이클의 끝에서:
   - 불법 명령어
   - 구현 안된 명령어
   - 특권 위반
3. 명령어 사이클의 끝에서:
   - 트레이스 예외
   - 인터럽트 예외
4. 명령어 사이클 내에서:
   - TRAP
   - TRAPV
   - CHK
   - 영으로 나누기

Interrupt (계속)

1. RESET, BERR : 가장 높은 우선순위 →clock cycle 의 끝에서 처리
2. 특권명령, 불법명령 : 다음 높은 우선순위 →버스 싸이클의 끝에서 처리
3. 소프트웨어 인터럽트 : 다음 높은 우선순위 →해당 명령어 싸이클 내에서 처리
4. 사용자 I/O 인터럽트 : 가장 낮은 우선순위 →현재 명령어 사이클의 끝에서 처리

∗ 처리 순서

1. 수퍼바이저 모드로 바꾼다
2. 예외의 원인을 확인한다
3. 현재의 프로세서 문맥을 저장
4. 예외처리 루틴으로 분기

∗ 예외의 원인들

1. 내부 하드웨어 예외 : 자동벡터 사용
2. 소프트웨어 예외 : 자동벡터 사용
3. I/O 하드웨어 인터럽트
   (a) 현재의 명령 완료후 인식

Interrupt (계속)

(b) 현재의 프로세서 레벨보다 작은 것은 처리가 연기됨

(c) 자동벡터인가 외부벡터 인터럽트인가를 구분

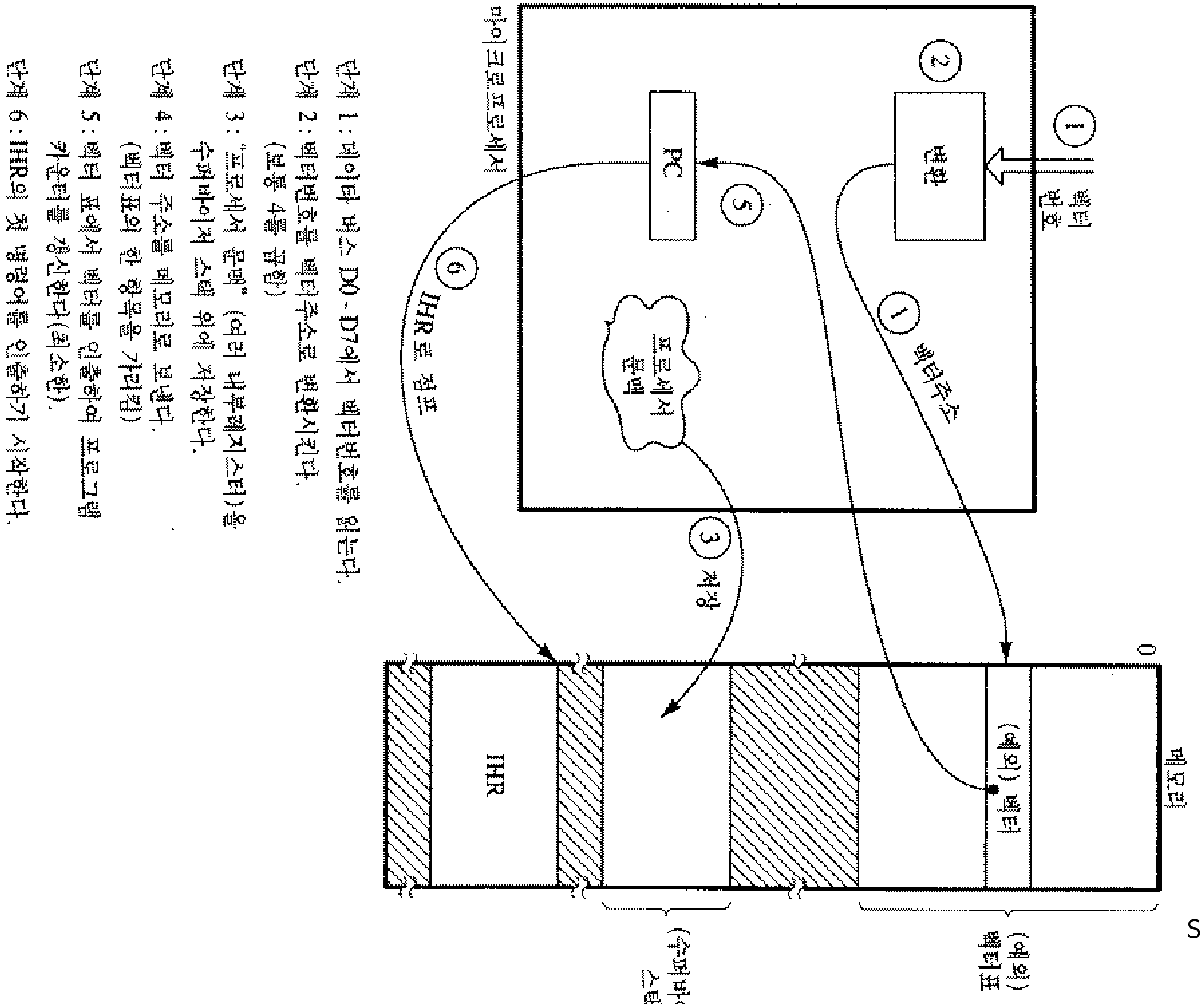
∗ 자동벡터 : 벡터 번호를 생성할수 없는 단순한 장치를 보조

∗ 벡터 인터럽트 : 벡터 번호를 생성할수 있는 장치들

1. 인터럽트 인지싸이클에서 벡터번호를 제공

∗ 인터럽트 서비스 루틴으로 점프 단계 (그림 7.5)

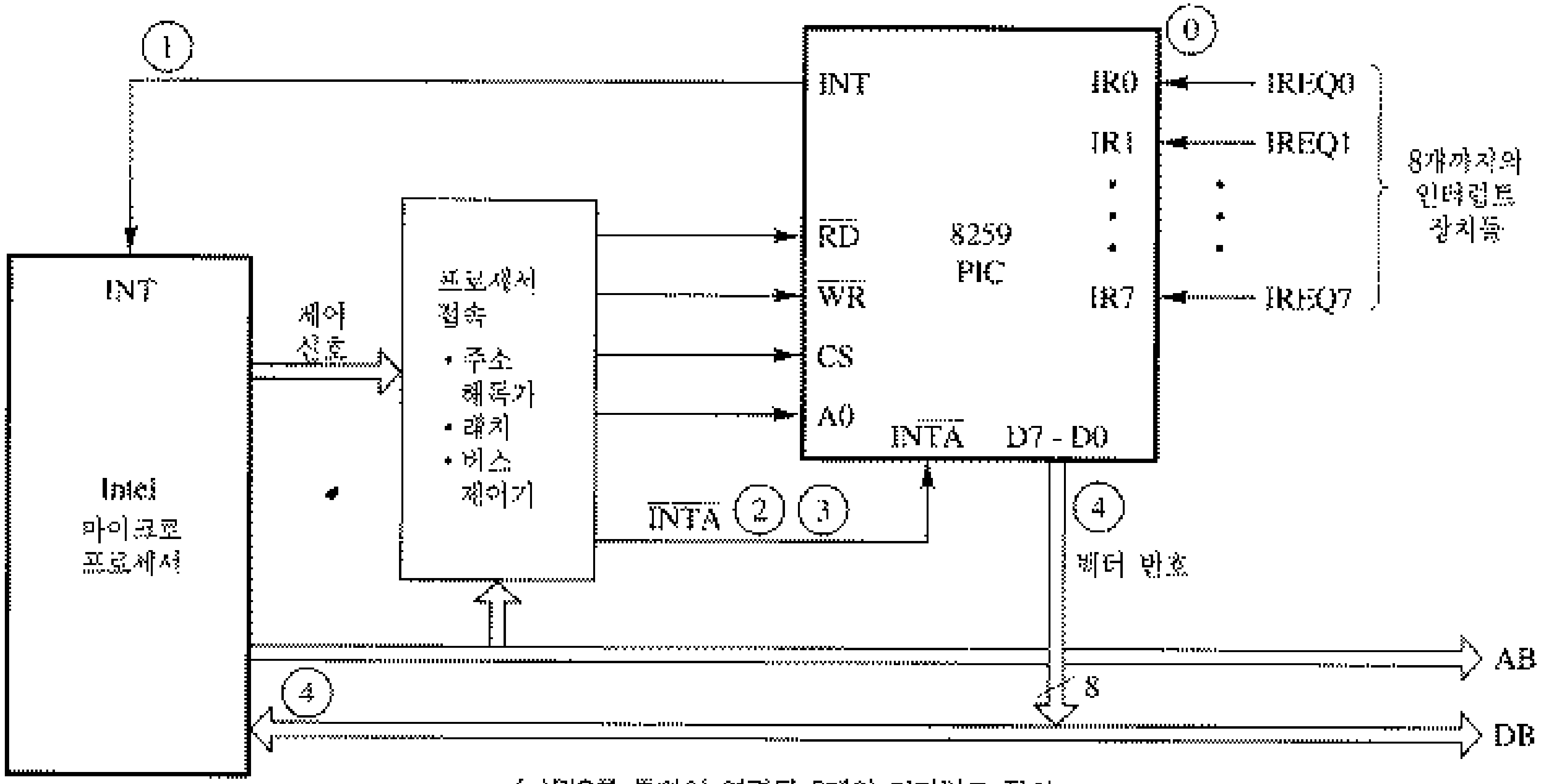
Interrupt (계속)



Interrupt (계속)

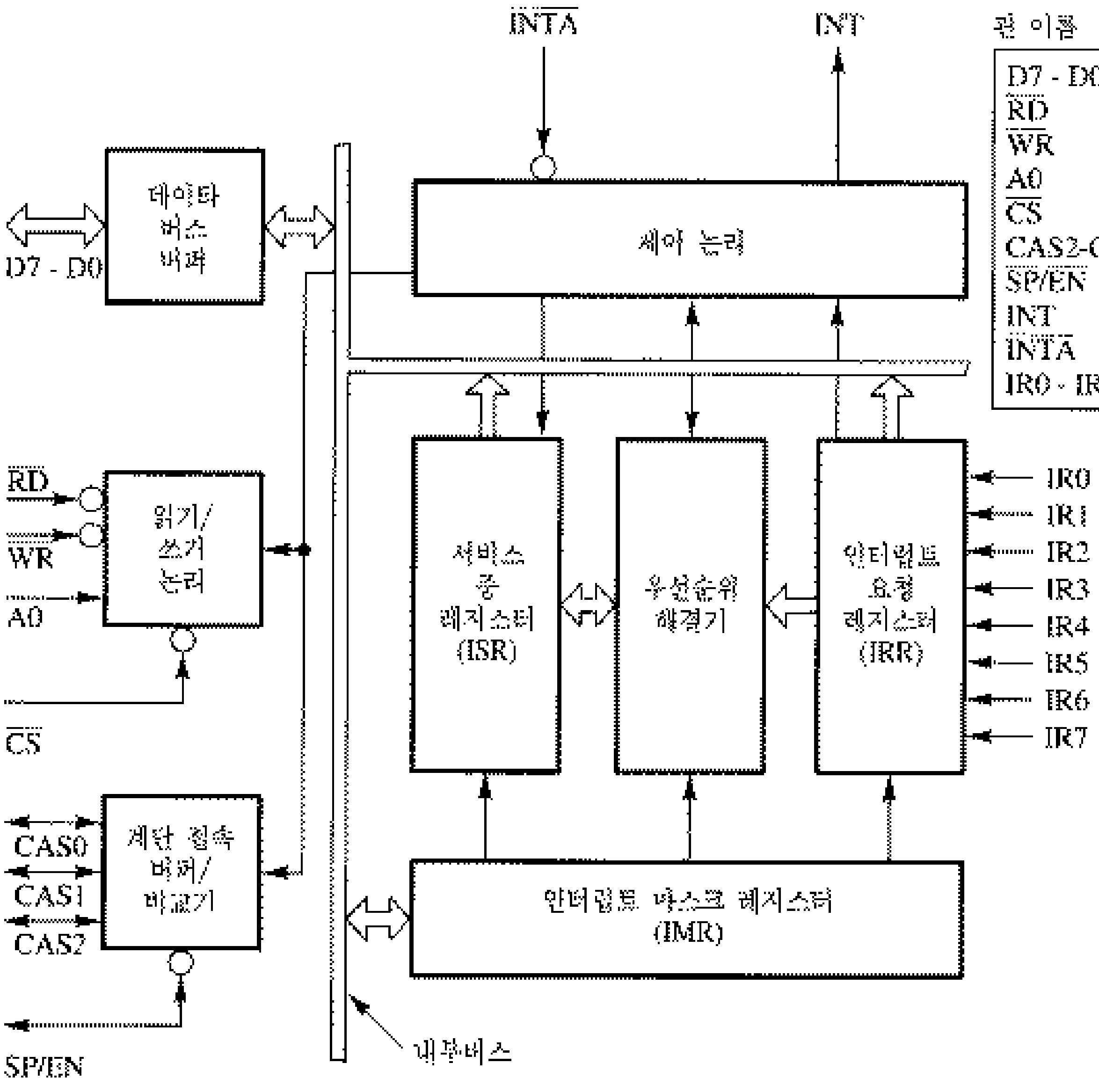
∗ 인텔의 벡터 인터럽트 구조

1. 8259를 사용한 접속 (그림 7.6)



(a)PIC를 통하여 연결된 8개의 인터럽트 장치

Interrupt (계속)



핀 이름

D7 - D0	데이타 버스(양방향)
RD	읽기 입력
WR	쓰기 입력
A0	명령 선택 주소
CS	칩 선택
CAS2-CAS0	계단 주소선
SP/EN	슬레이브 프로그램/인에이블 버스
INT	인터럽트 출력
INTA	인터럽트 인지 입력
IR0 - IR7	인터럽트 요청 입력

S.H. Jung

(b)8259 PIC의 블록도와 핀 [2]. (Intel사의 허가를 얻어 재재함 © 1979 Intel Corp.)

Interrupt (계속)

2. 64개의 I/O 인터럽트를 제공하기 위한 접속 (그림 7.7)

Interrupt (계속)

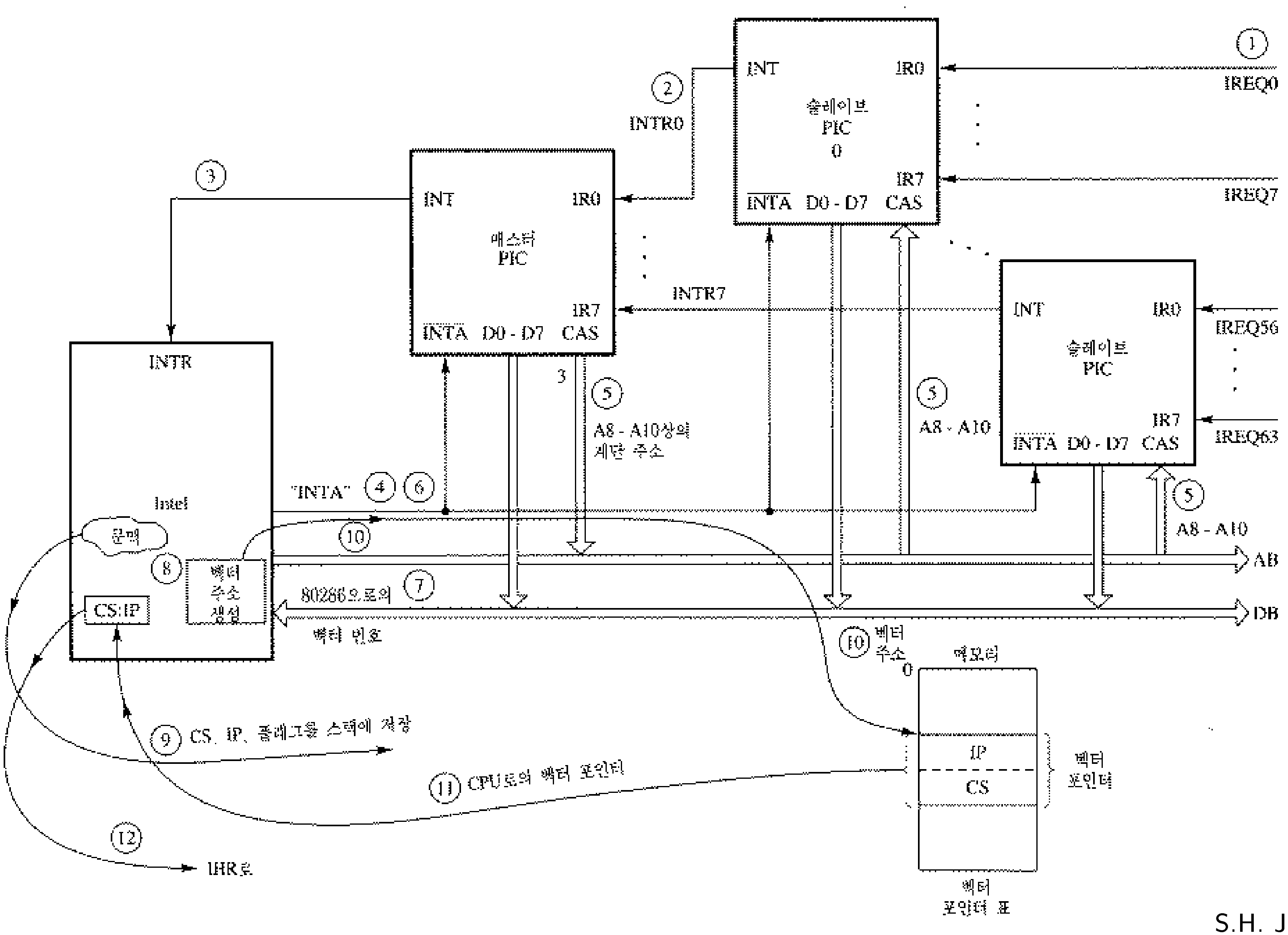


그림 7.7 계단 접속된 PIC로 연결된 Intel 마이크로프로세서 (동그라미친 번호가 매겨진 단계는 절차 7.2의 단계에 해당함).