



6 데이터 링크 계층

쉽게 배우는 데이터 통신과 컴퓨터 네트워크

학습목표

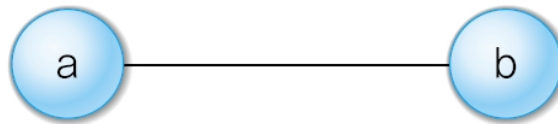
- ✓ 오류 제어, 흐름 제어의 원리와 동작 방식을 이해
- ✓ 통신 프로토콜에서 윈도우의 개념과 동작 방식을 이해
- ✓ 양방향 통신을 지원하는 슬라이딩 윈도우 프로토콜
- ✓ HDLC 프로토콜을 통해 프로토콜을 구현하는 원리를 이해



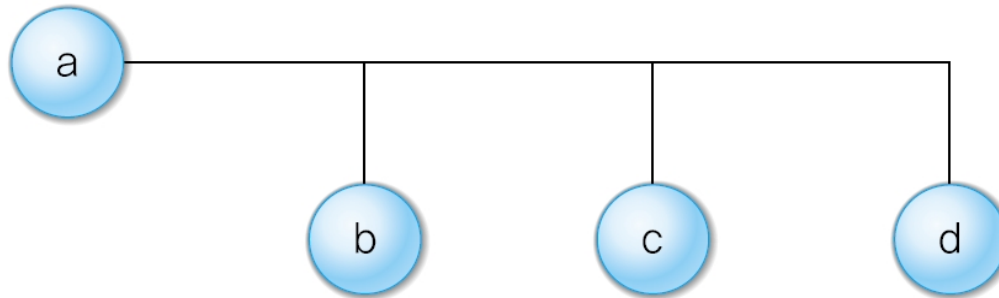
1절. 데이터 링크 계층 프로토콜의 기초

- 데이터 링크 계층은 두 호스트를 1:1로 연결하기 위한 **점대점 방식을 지원해야 함**
(중간에 중계 과정은 없음)
- 연결 구성도
 - 점대점: [그림 6-1(a)]
 - 멀티 드롭(multi drop) : 구분을 위한 주소(Address) 개념 필요 [그림 6-1(b)]
: 하나의 호스트가 다수의 호스트와 연결된 비대칭 구조
- **주요 기능 : 물리 계층의 전송 오류를 재전송(Retransmission)기법으로 복구**

(a) 점대점



(b) 멀티 드롭



[그림 6-1] 연결 구성도



1절. 데이터 링크 계층 프로토콜의 기초

□ 1) 프레임의 종류

- 정보 프레임: I 프레임
 - 상위 계층이 전송 요구한 데이터를 송신하는 용도
 - 순서 번호, 송수신 호스트 정보 등이 포함됨
- 긍정 응답 프레임: ACK 프레임(Positive Acknowledge)
 - 전송 데이터가 올바르게 도착했음을 회신하는 용도
 - 데이터를 수신한 호스트가 데이터를 송신한 호스트에게 전송
- 부정 응답 프레임: NAK 프레임(Negative Acknowledge)
 - 전송 데이터가 깨져서 도착했음을 회신하는 용도
 - 데이터를 수신한 호스트가 데이터를 송신한 호스트에게 전송
 - 데이터를 송신한 호스트는 원래의 데이터를 재전송하여 오류 복구
- → 3가지 프레임을 사용해서 데이터 링크의 전송 프로토콜을 작성함



1절. 데이터 링크 계층 프로토콜의 기초

□ 2) 오류, 흐름 제어가 없는 프로토콜

■ 가정

- (1) 단방향 통신: 데이터는 송신 호스트에서 수신 호스트 한쪽 방향으로만 전송
- (2) 전송 오류가 없는 물리 매체: 어떠한 전송 오류도 발생하지 않음
- (3) 무한 개의 수신 버퍼: 흐름 제어 기능이 필요 없음

■ 단순 프로토콜

- 송신 호스트는 원하는 만큼 자유롭게 프레임을 전송할 수 있음
- 그러나 단순히 정보 프레임만 전송 가능함
- 오류 제어: 프레임 분실/변형 오류가 발생하지 않음
- 흐름 제어: 수신 버퍼가 무한개이므로 분실 오류 없음

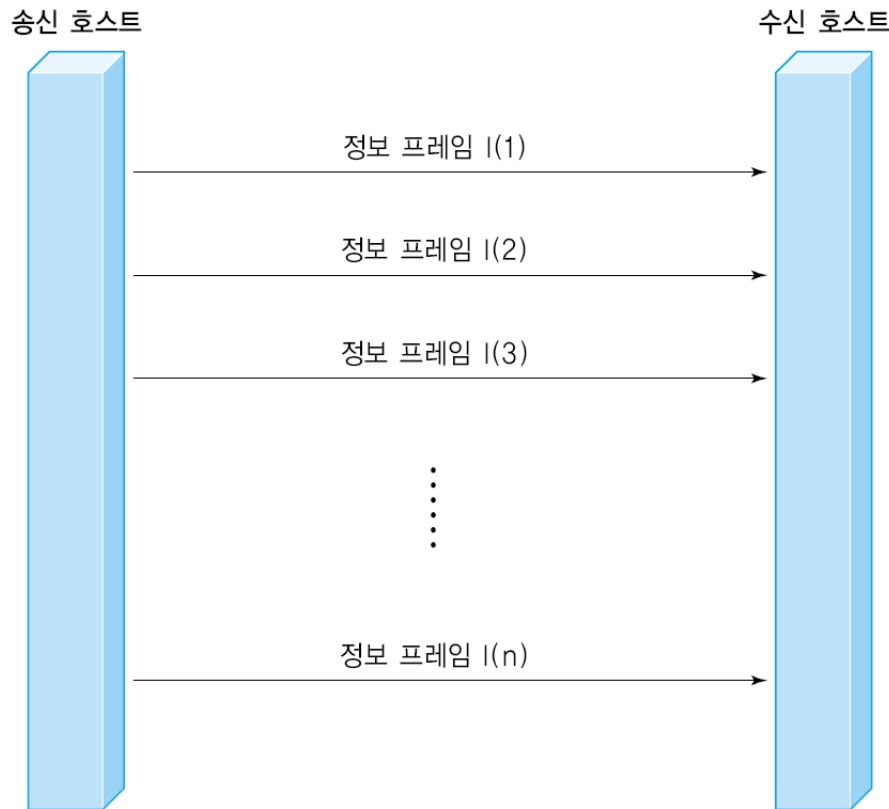


1절. 데이터 링크 계층 프로토콜의 기초

□ 2) 오류, 흐름 제어가 없는 프로토콜

■ 단순 프로토콜 [그림 6-2]

- 버퍼가 무한대이기 때문에 오류 없이 도착하므로 **순서 번호 기능 불필요**
(그림에서는 이해 편의를 위해 순서 번호 표기)



[그림 6-2] 단순 프로토콜



1절. 데이터 링크 계층 프로토콜의 기초

□ 3) 오류 제어가 없는 프로토콜(단, 흐름 제어가 필요)

■ 가정

- (1) 단방향 통신: 데이터는 송신 호스트에서 수신 호스트 한쪽 방향으로만 전송
(제어 신호는 반대 방향으로 전송 가능)
- (2) 전송 오류가 없는 물리 매체: 어떠한 전송 오류도 발생하지 않음
- 수신 버퍼의 개수가 유한하므로 흐름 제어 필요

■ 정지-대기 프로토콜 -1

- 수신 호스트가 송신 호스트의 전송 시점을 지정하기 위한 ACK 프레임 필요
- 데이터의 중복 수신 우려가 없으며 순서 번호 기능이 불필요
- → 정지-대기 프로토콜(Stop-and-Wait Protocol)은 기본적으로 흐름제어를 수행하지만

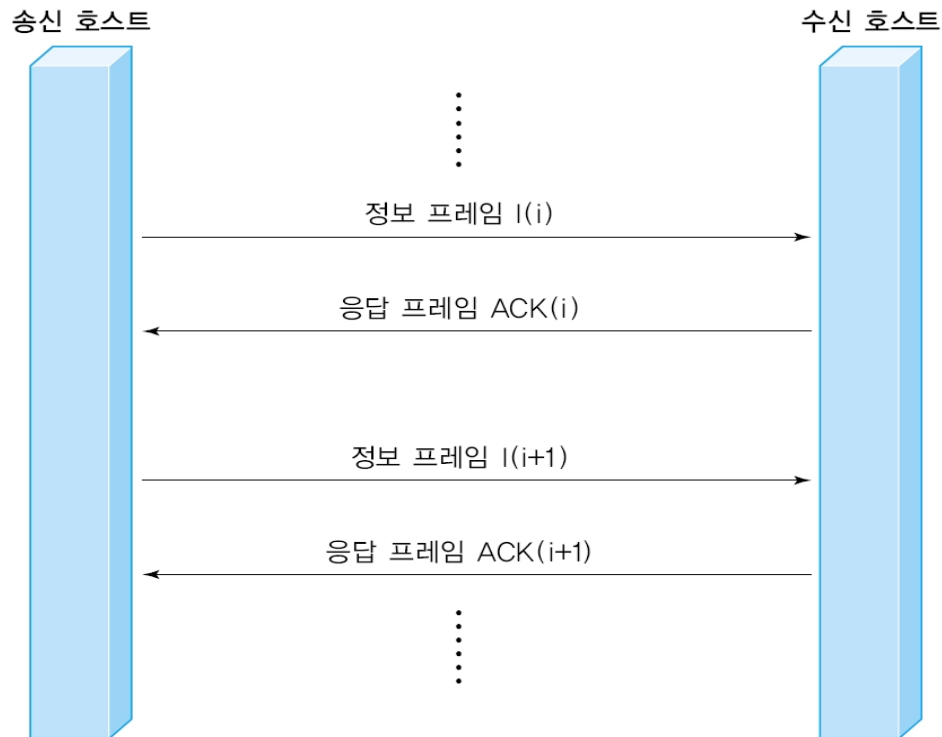
NAK 신호와 타임아웃 기능을 이용한다면 오류제어도 수행함



1절. 데이터 링크 계층 프로토콜의 기초

□ 3) 오류 제어가 없는 프로토콜(단, 흐름 제어가 필요)

- 정지-대기 프로토콜(Stop-and-Wait)-1 [그림 6-3] – 흐름제어
 - 송신 호스트는 ACK 프레임을 회신 받은 경우에만 다음 데이터 전송 가능
 - 효율성이 낮아 일반 네트워크에서는 사용하지 않음



[그림 6-3] 정지-대기 프로토콜 1



1절. 데이터 링크 계층 프로토콜의 기초

□ 4) 단방향 프로토콜 : 정지-대기 프로토콜(Stop-and-Wait)-2 혹은 3)

■ 가정

- 단방향 통신: 데이터는 송신 호스트에서 수신 호스트 한쪽 방향으로만 전송
(제어 신호는 반대 방향으로 전송 가능)
- 오류 제어와 흐름 제어가 모두 필요
- 프레임 분실 오류를 해결하기 위한 송신 호스트의 타임아웃 기능 필요
- 프레임 변형 오류를 해결하기 위한 수신 호스트의 NAK 기능 필요
- 일반적으로 NAK과 타임아웃 기능을 모두 사용하여 오류 제어를 수행

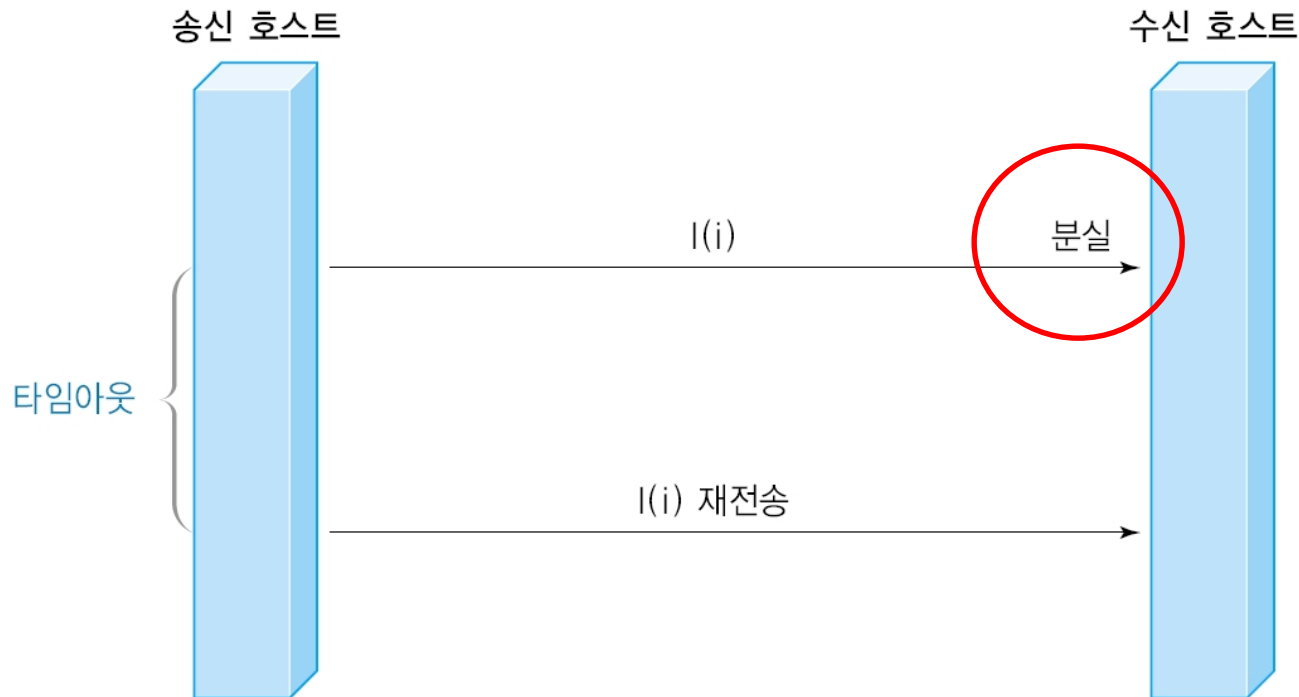


1절. 데이터 링크 계층 프로토콜의 기초

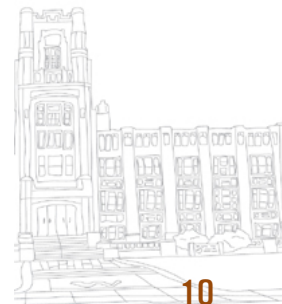
□ 4) 단방향 프로토콜 :정지-대기 프로토콜(Stop-and-Wait)-2

■ NAK가 없는 경우(NAK가 정의되지 않은 경우)

- 정보 프레임 분실: 송신 호스트의 타임아웃 기능으로 오류 복구 [그림 6-4(a)]



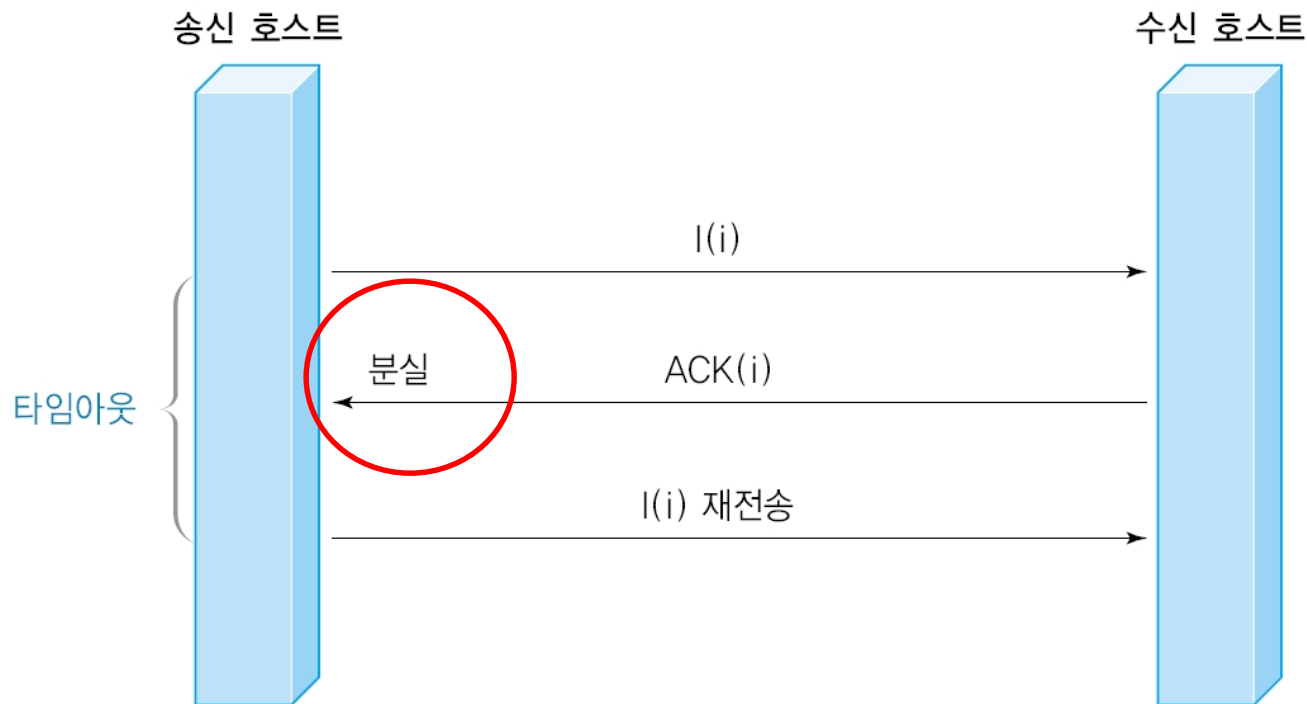
(a) 정보 프레임 분실 오류



1절. 데이터 링크 계층 프로토콜의 기초

□ 4) 단방향 프로토콜 :정지-대기 프로토콜(Stop-and-Wait)-2

- NAK가 없는 경우 (NAK가 정의되지 않은 경우)
 - ACK 프레임 분실: 송신 호스트의 타임아웃 기능으로 오류 복구 [그림 6-4(b)]



(b) ACK 프레임 분실 오류

[그림 6-4] 정지-대기 프로토콜 2 : NAK가 없는 경우

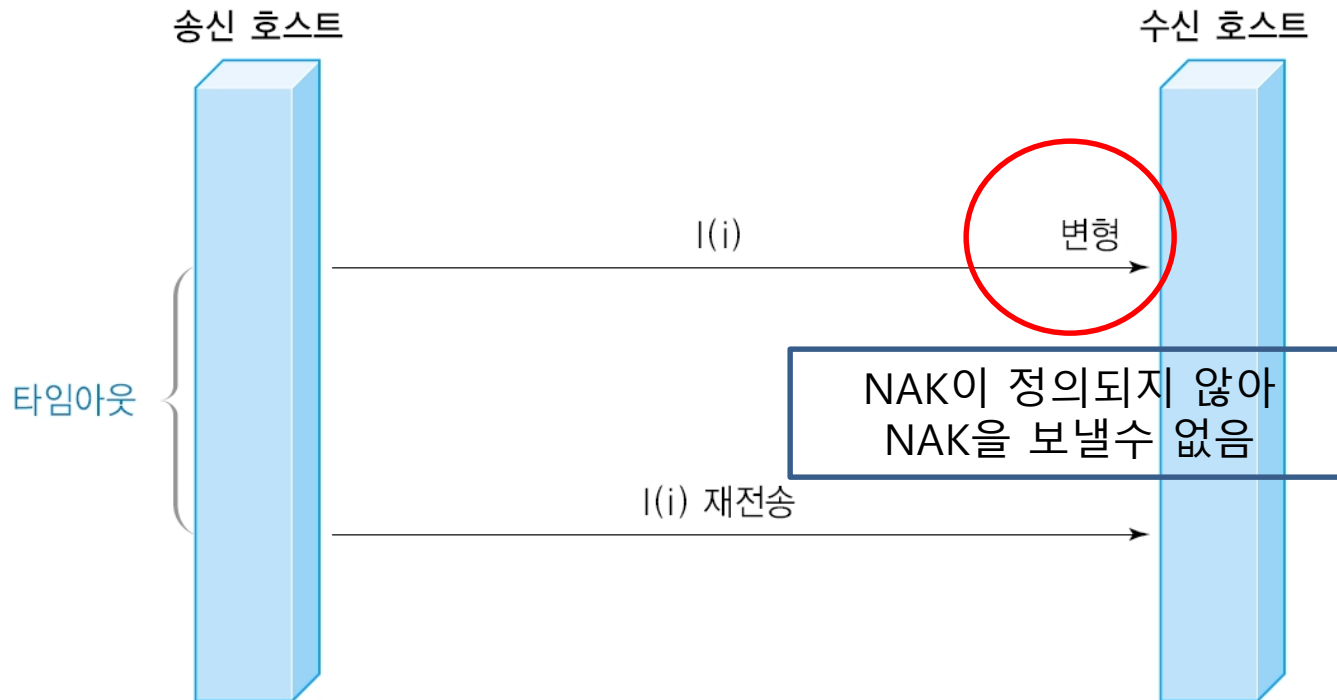


1절. 데이터 링크 계층 프로토콜의 기초

□ 4) 단방향 프로토콜 :정지-대기 프로토콜(Stop-and-Wait)-2

■ NAK가 없는 경우 (NAK가 정의되지 않은 경우)

- 정보 프레임 변형: 송신 호스트의 타임아웃 기능으로 오류 복구 [그림 6-5]



[그림 6-5] 프레임 변형 오류

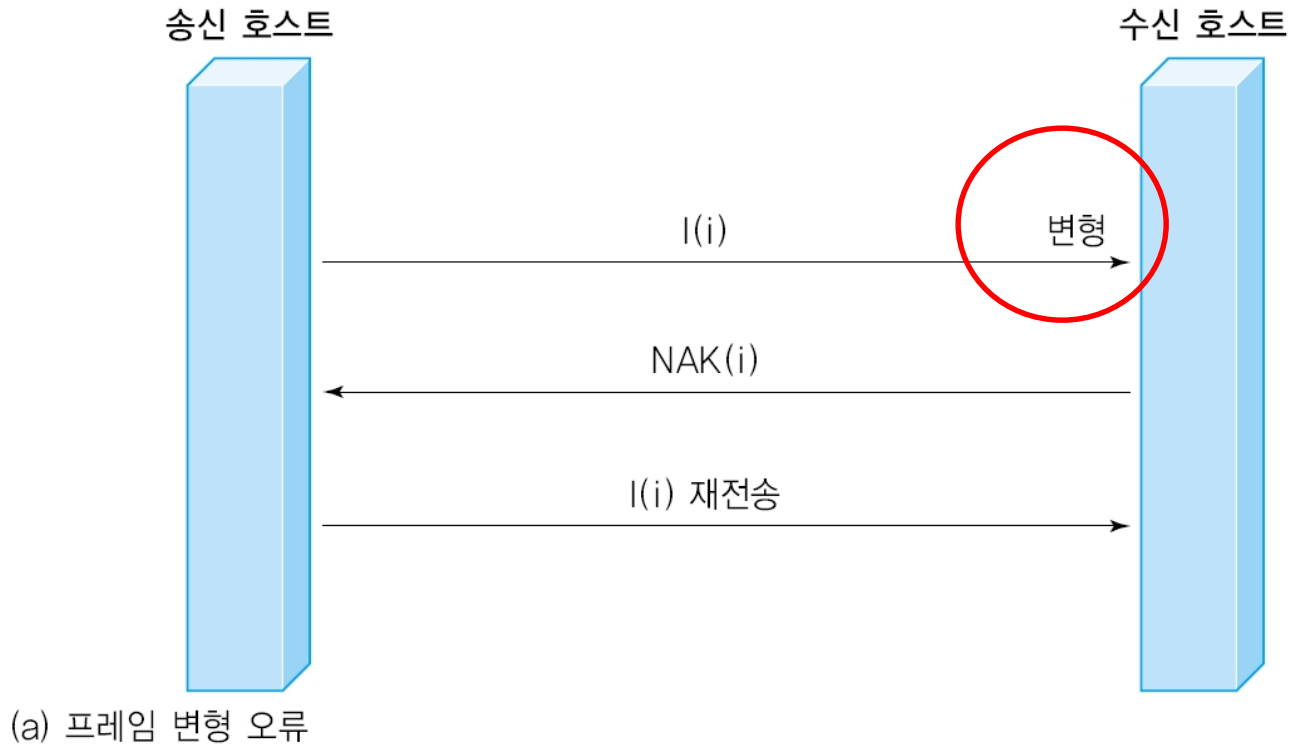


1절. 데이터 링크 계층 프로토콜의 기초

□ 4) 단방향 프로토콜 :정지-대기 프로토콜(Stop-and-Wait)-3

■ NAK가 있는 경우

- **정보 프레임 변형**: 수신 호스트의 NAK 프레임 응답으로 오류 복구 [그림 6-6(a)]

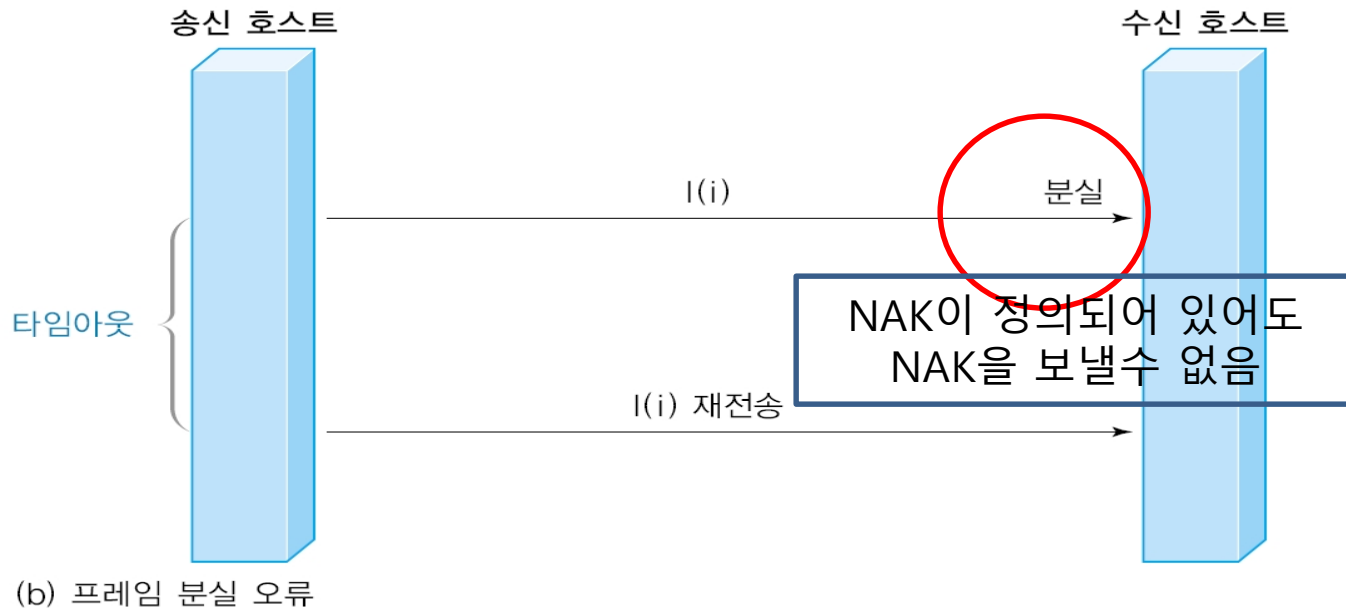


1절. 데이터 링크 계층 프로토콜의 기초

□ 4) 단방향 프로토콜 :정지-대기 프로토콜(Stop-and-Wait)-3

■ NAK가 있는 경우

- 정보 프레임 분실: 송신 호스트의 타임아웃 기능으로 오류 복구 [그림 6-6(b)]



[그림 6-6] 정지-대기 프로토콜 3 : NAK가 있는 경우

➔ 정지-대기 프로토콜은 기본적으로 흐름제어를 수행하며
NAK 신호와 타임아웃 기능을 이용하여 오류제어도 수행함

