

컴퓨터 네트워크

# 6장. 데이터링크 계층 (1)

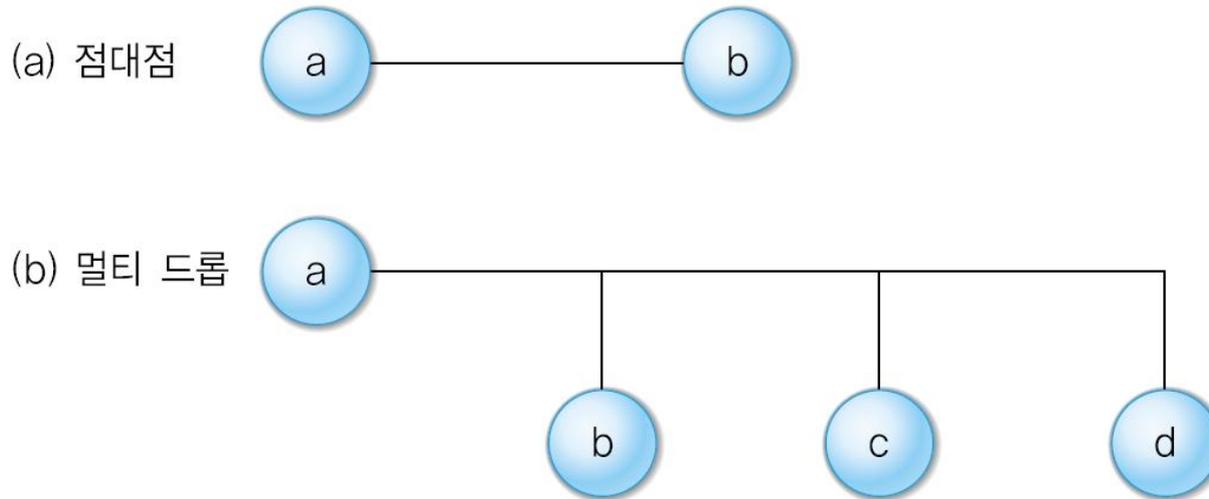
## - 프로토콜의 기초

# 이번 시간의 학습 목표

- ▶ 오류 제어, 흐름 제어의 원리와 동작 방식을 이해한다.

# 연결 구성

- ▶ 점대점(point-to-point)
- ▶ 멀티드롭(multi drop)
  - ▶ 주소(address) 개념 필요



[그림 6-1] 연결 구성도

# 프레임 종류

- ▶ 정보 프레임: I 프레임
  - ▶ 상위 계층이 전송 요구한 데이터를 송신하는 용도
  - ▶ 순서번호, 송수신 호스트 정보 등이 포함됨
- ▶ 긍정 응답 프레임: ACK 프레임
  - ▶ 전송 데이터가 올바르게 도착했음을 회신하는 용도
  - ▶ 데이터를 수신한 호스트가 데이터를 송신한 호스트에게 전송
- ▶ 부정 응답 프레임: NAK 프레임
  - ▶ 전송 데이터가 깨져서 도착했음을 회신하는 용도
  - ▶ 데이터를 수신한 호스트가 데이터를 송신한 호스트에게 전송
  - ▶ 데이터를 송신한 호스트는 원래의 데이터를 재전송하여 오류 복구
- ▶ 긍정 응답, 부정 응답 프레임 모두 회신하고자 하는 I 프레임 순서 번호 포함
  - ▶ 제대로 도착한 프레임과 오류가 발생한 프레임 판단 가능

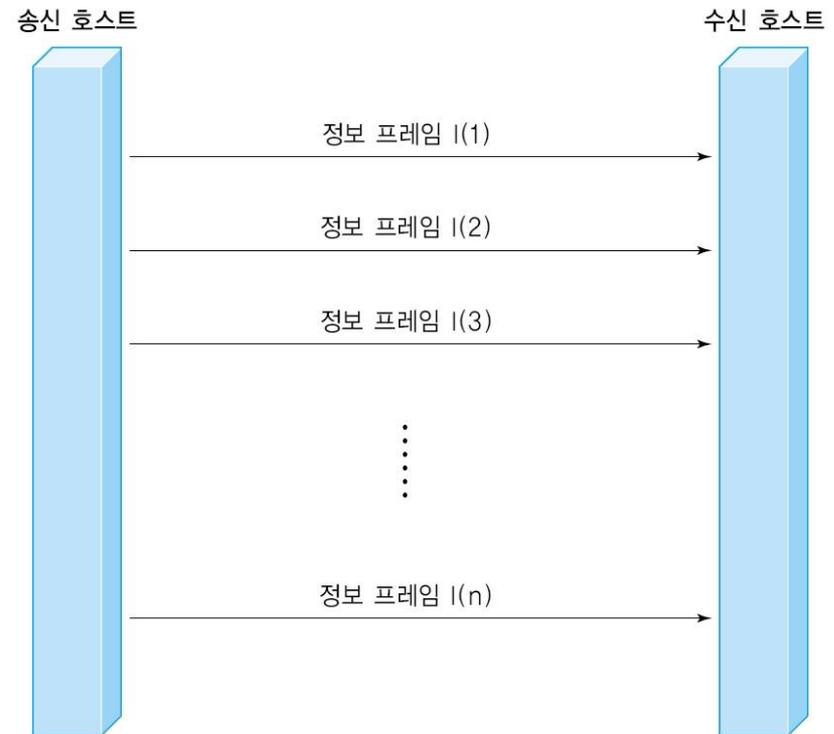
# 오류, 흐름 제어가 없는 프로토콜

## ▶ 가정

- ▶ 단방향 통신
  - ▶ 송신 호스트에서 수신 호스트 한쪽 방향으로만 데이터 전송
- ▶ 전송 오류가 없는 물리 매체
  - ▶ 어떠한 전송 오류도 발생하지 않음
- ▶ 무한 개의 수신 버퍼
  - ▶ 흐름 제어 기능이 필요 없음

## ▶ 단순 프로토콜

- ▶ 송신 호스트는 원하는 만큼 자유롭게 프레임 전송할 수 있음
- ▶ 오류 제어 없음
  - ▶ 프레임 분실/변형 오류가 발생하지 않음
- ▶ 흐름 제어 없음
  - ▶ 수신 버퍼가 무한이므로 분실 오류 없음
- ▶ 순서번호도 불필요



[그림 6-2] 단순 프로토콜

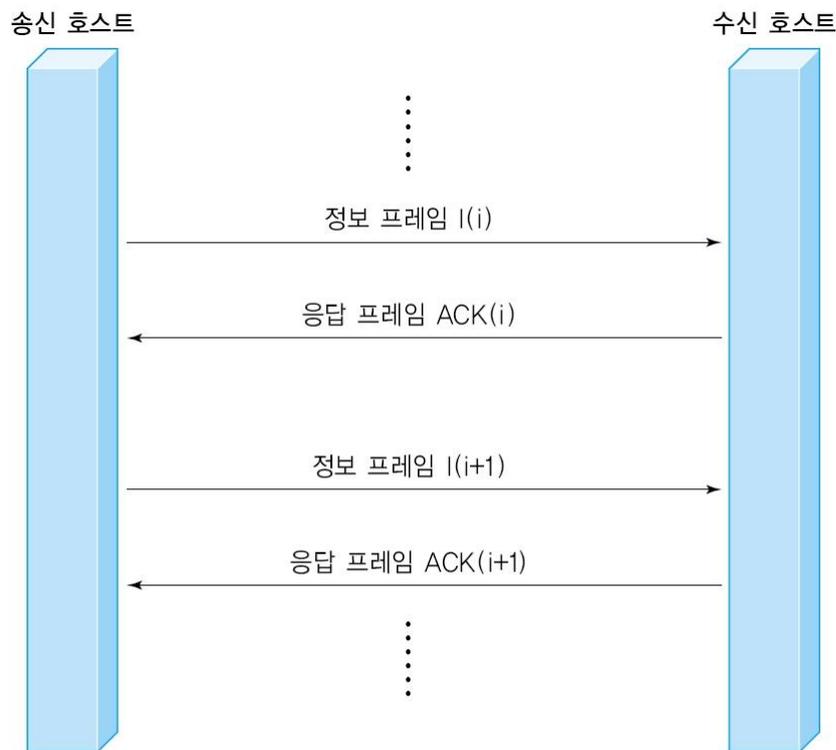
# 오류 제어가 없는 프로토콜

## ▶ 가정

- ▶ 단방향 통신
  - ▶ 송신 호스트에서 수신 호스트 한쪽 방향으로만 데이터 전송
- ▶ 전송 오류가 없는 물리 매체
  - ▶ 어떠한 전송 오류도 발생하지 않음
- ▶ 전송 오류는 없으나 버퍼 부족으로 프레임 분실 가능

## ▶ 정지-대기 프로토콜 1

- ▶ 수신 버퍼의 개수가 유한
  - ▶ 흐름 제어 필요
- ▶ 이전 프레임을 잘 받았다는 긍정 응답 기능과 수신 호스트가 송신 호스트의 전송 시점을 지정하기 위한 ACK 프레임 필요
- ▶ 데이터의 중복 수신 우려가 있으며 순서 번호 기능 필요
- ▶ 정지대기(Stop-and-wait)
  - ▶ 하나의 프레임 전송 후 응답 대기
  - ▶ 전송 효율이 떨어지므로 잘 사용하지 않음



[그림 6-3] 정지-대기 프로토콜 1

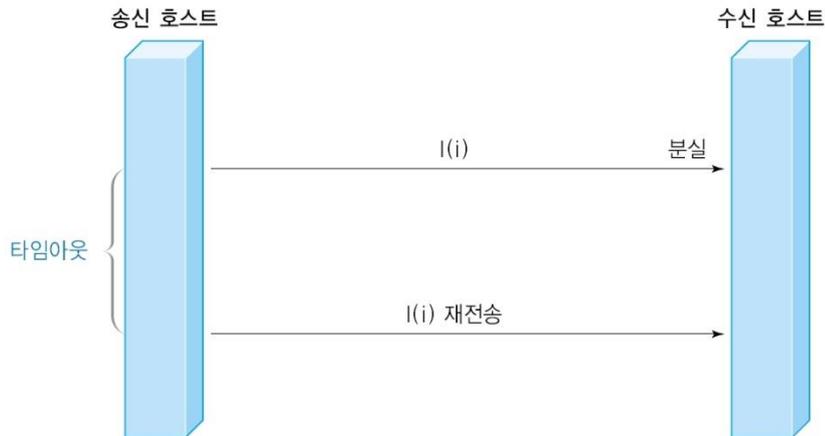
# 단방향 프로토콜

- ▶ 가정
  - ▶ 단방향 통신: 송신 호스트에서 수신 호스트 한쪽 방향으로만 데이터 전송
  - ▶ 전송 매체 오류 발생 가능
  - ▶ 버퍼 유한
- ▶ 오류 제어와 흐름 제어가 모두 필요
- ▶ 프레임 변형 오류를 해결하기 위한 수신 호스트의 NAK 기능 필요
- ▶ 프레임 분실 오류를 해결하기 위한 송신 호스트의 타임아웃 기능 필요

# 단방향 프로토콜 - NAK가 없는 경우 (1)

## ▶ 정보 프레임 분실

- ▶ 송신 호스트의 타임아웃 기능으로 오류 복구



(a) 정보 프레임 분실 오류

## ▶ ACK 프레임 분실

- ▶ 송신 호스트의 타임아웃 기능으로 오류 복구



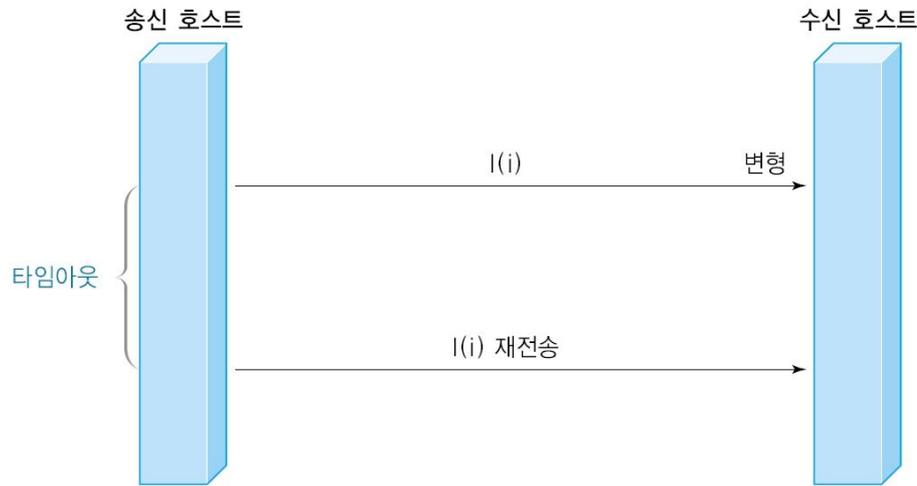
(b) ACK 프레임 분실 오류

[그림 6-4] 정지-대기 프로토콜 2 : NAK가 없는 경우

# 단방향 프로토콜 - NAK가 없는 경우 (2)

## ▶ 정보 프레임 변형

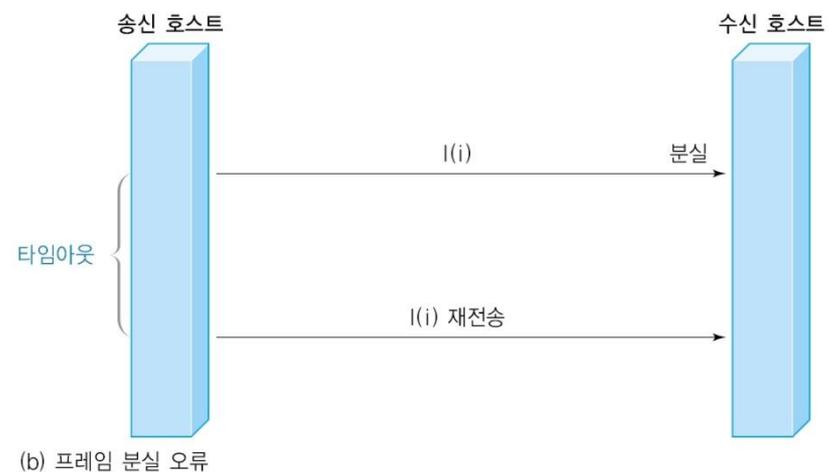
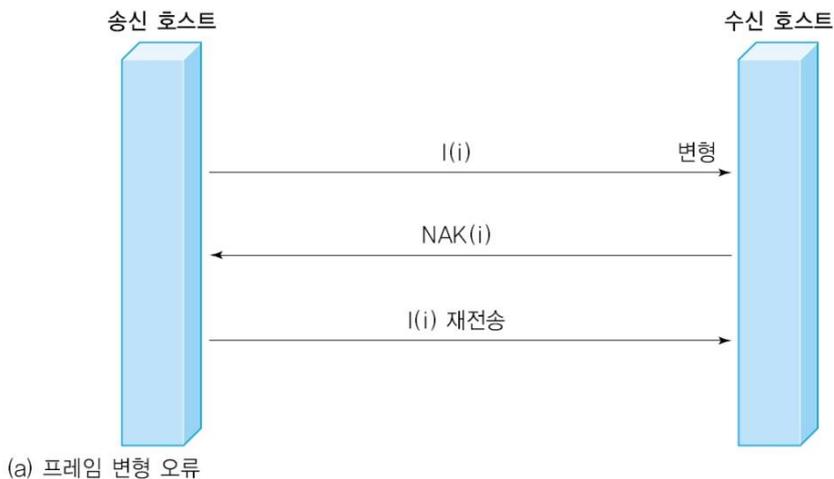
- ▶ 송신 호스트의 타임아웃 기능으로 오류 복구



[그림 6-5] 프레임 변형 오류

# 단방향 프로토콜 - NAK가 있는 경우

- ▶ 정보 프레임 변형 오류
  - ▶ 변형된 프레임 무시
    - ▶ 프레임 분실 오류와 동일하게 처리
  - ▶ NAK를 이용하여 통보
    - ▶ 더 효율적일 수 있음
    - ▶ 다른 요인에 의해 NAK를 사용하지 못하는 경우도 있음



[그림 6-6] 정지-대기 프로토콜 3 : NAK가 있는 경우

# 질의 / 응답