



4 데이터 전송의 기초

쉽게 배우는 데이터 통신과 컴퓨터 네트워크

학습목표

- ✓ 전송과 교환 시스템의 구조와 원리
- ✓ 프레임 전송 과정에서 발생하는 오류의 유형
- ✓ 문자 프레임과 비트 프레임의 구조
- ✓ 오류 검출 코드의 종류와 원리를 이해
- ✓ 생성 다항식을 이용한 오류 검출 방식

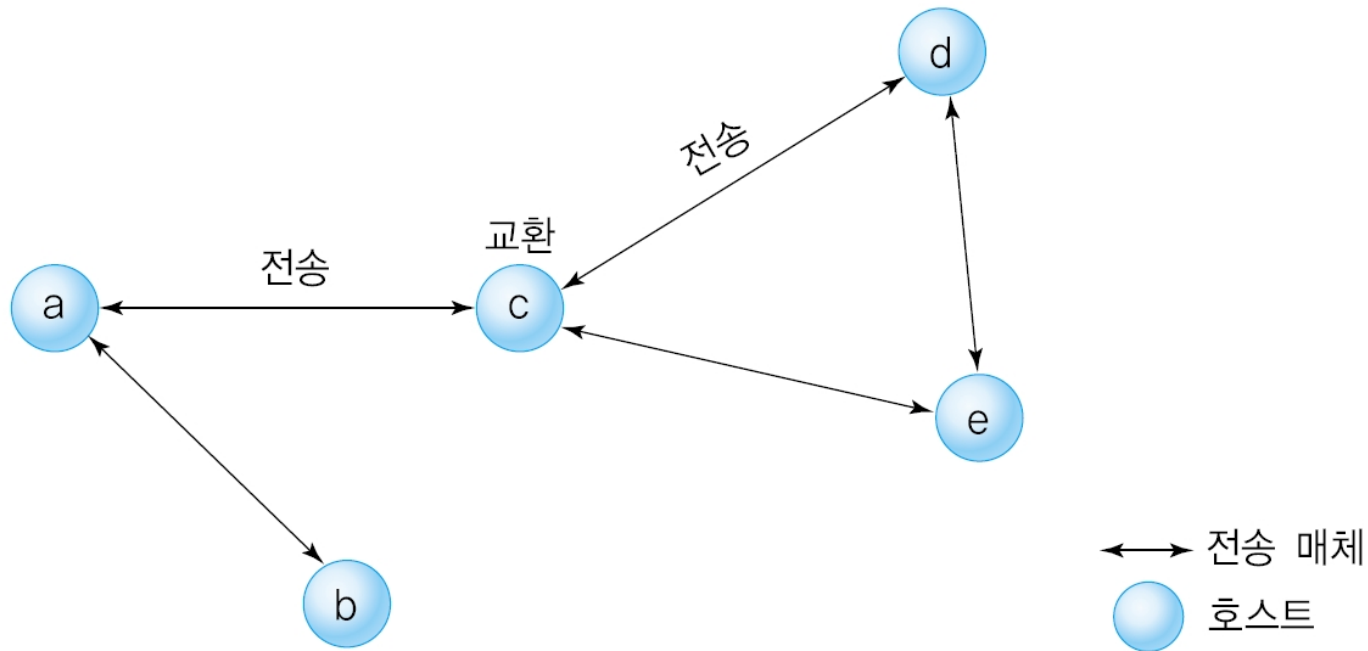


1절. 데이터 전송 방식

□ 전송과 교환(전달 : transfer)

■ 전송과 교환 개요 [그림 4-1]

- 교환(Switching): 갈림길에서 데이터가 전송되어야 할 경로를 선택
- 전송(Transmission): 물리적으로 1:1 연결된 시스템 사이의 데이터 전송
라우팅 개념이 없는 전달



[그림 4-1] 전송과 교환



1절. 데이터 전송 방식

□ 전송과 교환

■ 전송 방식의 종류

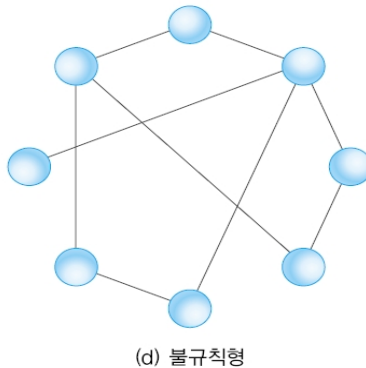
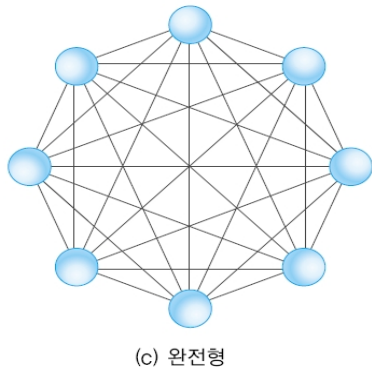
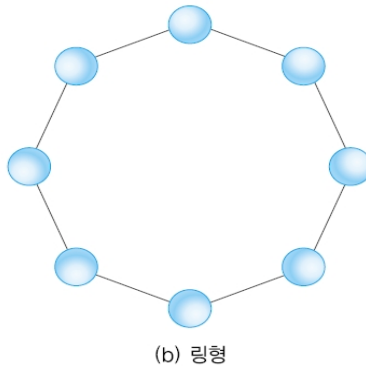
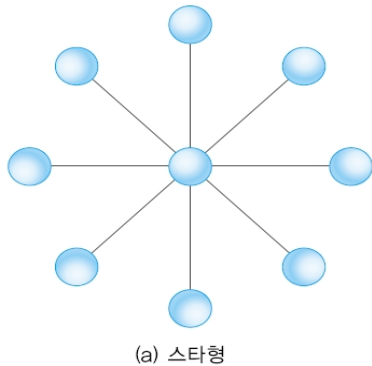
- **점대점(Point-to-Point)방식(스타형, 트리형, 링형...)**
 - 호스트들이 물리적으로 1:1 형식으로 연결
 - 교환 기능이 필수적이며 단계적으로 반복
 - 주로 **WAN과 같이 원거리 통신 환경에서 사용**
- **브로드캐스팅(Broadcasting) 방식(버스형, 링형...)**
 - 호스트들이 공유 전송 매체에 연결
 - 교환 기능이 불필요(자신에게 오는 데이터 중 불필요한 것은 버림)
 - 주로 **LAN과 같이 근거리의 통신 환경에서 사용**



1절. 데이터 전송 방식

□ 점대점 방식

- 통신하고자 하는 두 호스트를 중계 과정을 거쳐 연결
- 종류: 스타형, 링형, 완전형, 불규칙형 [그림 4-2]
- 연결 수가 증가하면 성능적인 면은 유리하지만 비용이 증가됨



[그림 4-2] 점대점 방식

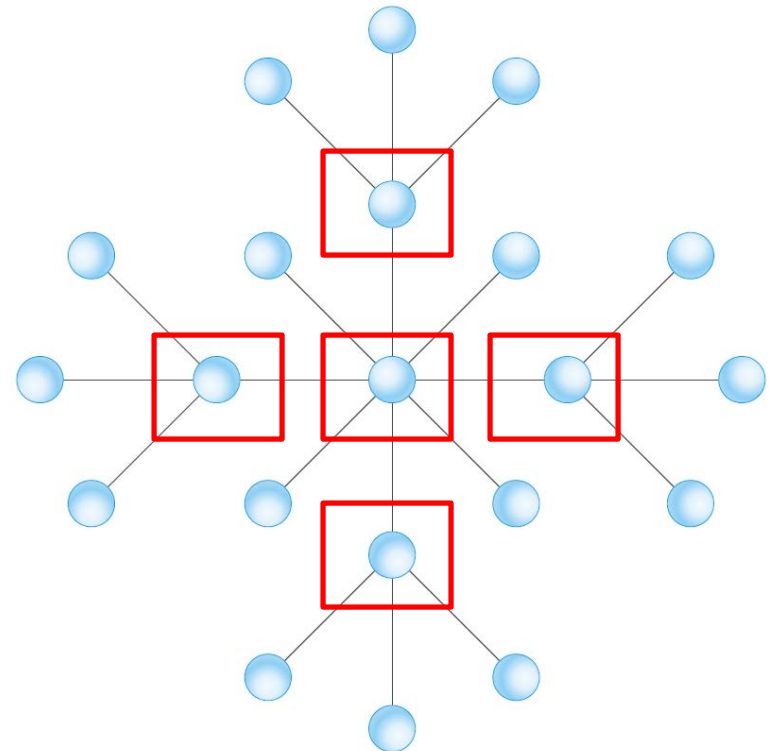
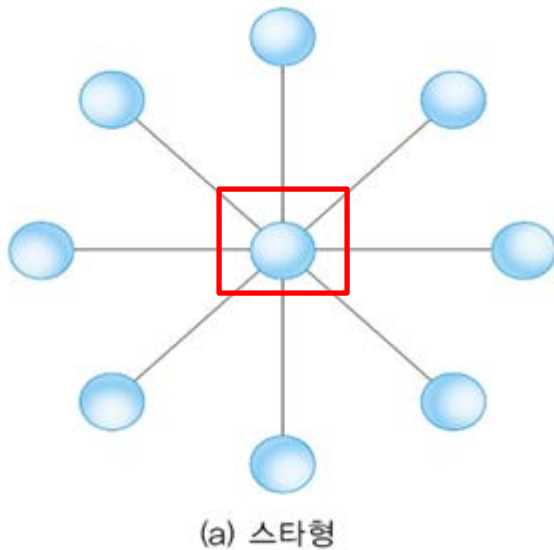


1절. 데이터 전송 방식

□ 점대점 방식

■ 1) 스타형

- 중앙의 중계 호스트 주위로 여러 호스트를 1:1 연결
- 중앙 호스트의 성능과 신뢰성이 중요
- 트리형: 스타형을 다단계로 확장 [그림 4-3]



1절. 데이터 전송 방식

□ 점대점 방식

■ 2) 링형

- 호스트의 연결이 순환 구조를 이룸
- 모든 호스트가 전송과 교환 기능을 수행
- 토큰
 - 데이터를 전송할 수 있는 권리
 - 데이터 전송을 원하는 호스트는 미리 토큰을 확보해야 함
 - 데이터 전송이 완료되면 호스트는 토큰을 반납해야 함
- 데이터 전송 원리
 - 먼저, 토큰을 링에서 회수하여 확보한 후, 데이터를 링에 전송함
 - 데이터는 링을 한 바퀴 순환한 후, 다시 송신 호스트에게 돌아옴
 - 이 과정에서 링에 연결된 모든 호스트가 데이터를 수신함
(자신을 목적지로 하는 호스트만 데이터를 보관하고, 다른 호스트는 버림)
 - 마지막으로, 송신 호스트는 데이터를 회수한 후에 토큰을 링에 돌려줌

■ 3) 완전형

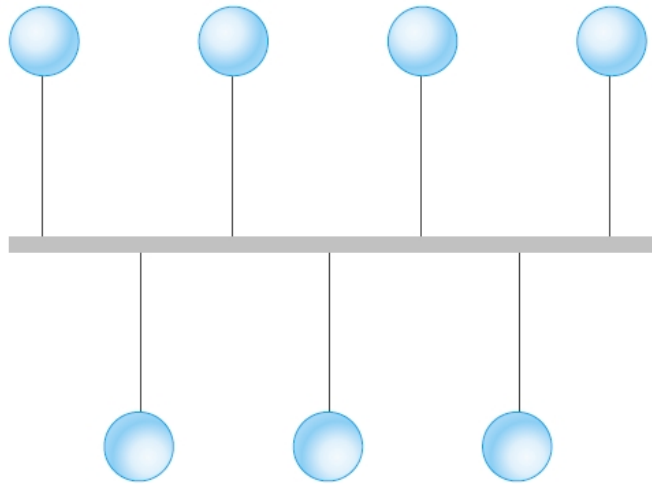
■ 4) 불규칙형



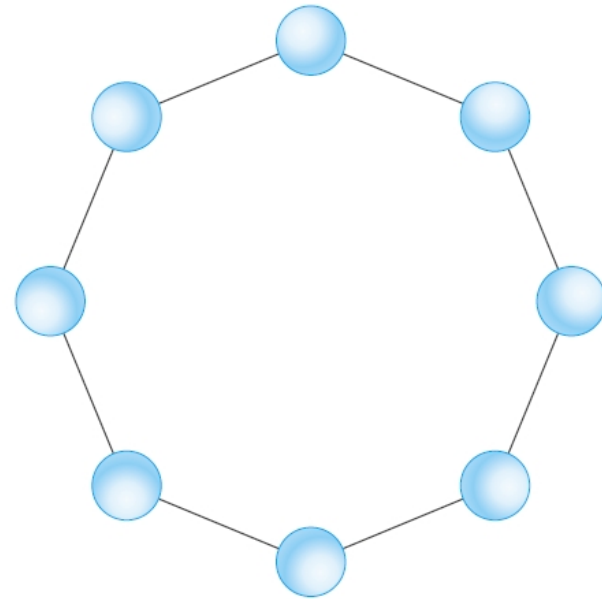
1절. 데이터 전송 방식

□ 브로드캐스팅 방식

- 네트워크에 연결된 모든 호스트에게 데이터를 전달하는 방식
- 주로 통신 거리가 짧은 LAN 환경에서 사용
- **버스형과 링형(모든 호스트에 전달)이 존재 [그림 4-4]**



(a) 버스형



(b) 링형

[그림 4-4] 브로드캐스팅 방식



1절. 데이터 전송 방식

□ 브로드캐스팅 방식

■ 1) 버스형

- 공유 버스에 모든 호스트를 연결
- 둘 이상의 호스트가 데이터를 전송하면 충돌 발생
- 충돌 문제의 해결 방법
 - 사전 예방: 전송 시간대를 다르게 하는 방법과 토큰 제어 방식이 가능
 - 사후 해결: 충돌을 감지하는 기능이 필요 (예: 이더넷)

■ 2) 링형

- 호스트를 순환 구조로 연결
- 송신 호스트가 전송한 데이터는 링을 한 바퀴 순환한 후 송신 호스트에 되돌아옴
- 종간의 호스트 중에서 수신 호스트로 지정된 호스트만 데이터를 내부에 저장
- 데이터를 전송하기 위해서는 토큰 확보가 필수



1절. 데이터 전송 방식

□ 멀티포인트 통신

■ 송신 호스트가 **한번의 전송으로**

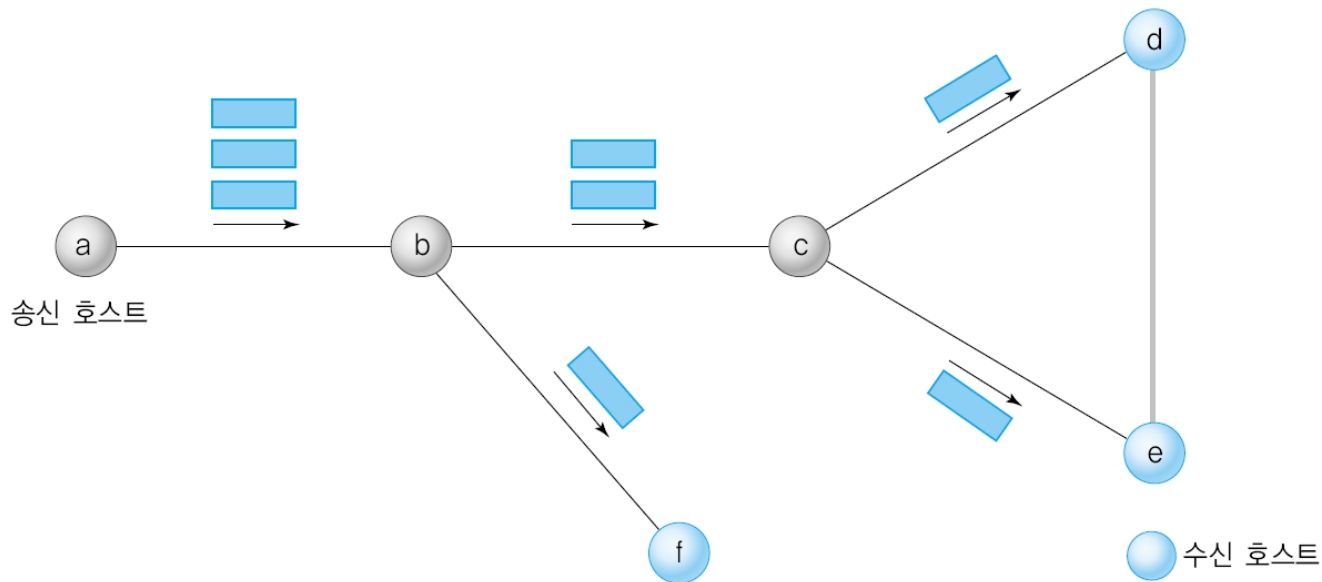
- 유니캐스팅(Unicasting): 하나의 수신 호스트에 데이터를 전송
→ 텔넷, FTP, 웹 검색 등 대부분의 서비스
- 멀티캐스팅(Multicasting) : **다수의 수신 호스트에 데이터를 전송**
→ 화상회의, 원격교육, 인터넷 채팅
- 브로드캐스팅(Broadcasting) : **네트워크의 모든 수신 호스트에 데이터를 전송**
→ 방송형으로서 수신 여부는 수신 호스트의 선택



1절. 데이터 전송 방식

□ 멀티포인트 통신

- **(멀티포인트) 유니캐스팅 [그림 4-5]**
 - 유니캐스팅 방식을 이용하여 일대다 통신을 지원
 - **호스트 a가 호스트 d, e, f에게 데이터를 전송하려면 3번의 송신 절차가 필요**
 - 수신 호스트의 수가 증가하면 성능에 문제점 발생



[그림 4-5] 멀티포인트 유니캐스팅

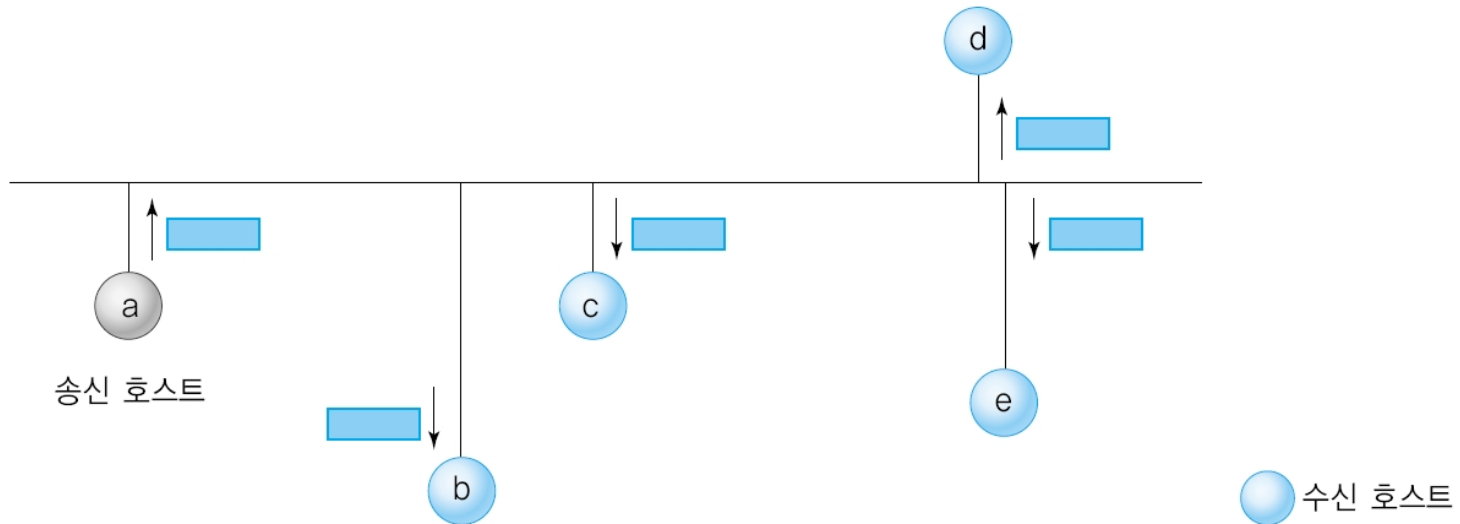


1절. 데이터 전송 방식

□ 멀티포인트 통신

■ 브로드캐스팅 [그림 4-6]

- 네트워크에 연결된 **모든 호스트에게 데이터 전송**
- 자신을 목적지로 하는 호스트만 데이터를 내부에 저장하고, 다른 호스트는 데이터를 무시함



[그림 4-6] 브로드캐스팅

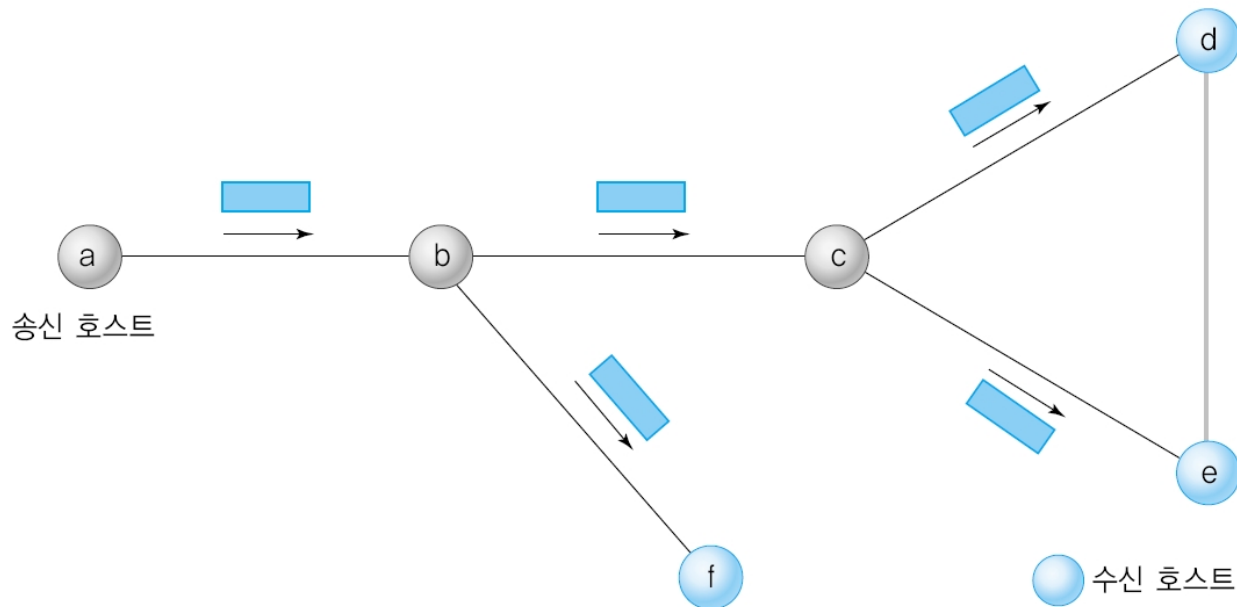


1절. 데이터 전송 방식

□ 멀티포인트 통신

■ 멀티캐스팅 [그림 4-7]

- 1:다 전송 기능을 지원, 멀티 캐스트 그룹을 구성해야 함
- 송신 호스트는 한번의 데이터 전송으로 여러 호스트에게 데이터를 전송할 수 있음
- 예: 송신 호스트 a, 수신 호스트 d, e, f
- 오디오 비디오 서비스, 화상회의 등에 활용



[그림 4-7] 멀티캐스팅



2절. 오류 제어

□ 전송 오류의 유형 : 프레임 변형 및 분실 → 재전송으로 극복

□ 신뢰성 있는 전송 서비스 조건

- 수신 호스트의 응답 프레임(변형시 대비)

- 긍정 응답 프레임(ACK): 데이터가 정상적으로 도착했을 때, 수신 호스트가 송신 호스트에게 회신
- 부정 응답 프레임(NAK): 데이터가 깨져서 도착했을 때, 수신 호스트가 송신 호스트에게 회신
- 부정 응답 프레임을 받은 송신 호스트는 재전송 기능으로 오류 복구 시도

- 송신 호스트의 타이머 기능(분실시 대비)

- 송신 호스트가 전송한 데이터가 네트워크에서 사라지는 문제를 해결
- 데이터 분실 시 수신 호스트로부터 어떠한 응답 프레임도 발생하지 않음
- 송신 호스트는 일정 시간 동안 응답 프레임이 없으면 타임아웃 기능으로 재전송 시도

- 순서 번호(Sequence Number) 기능(응답 분실 후 데이터 중복 전송시 대비)

- 송신자가 중복해서 프레임을 보낼 경우에 대비
- 수신 호스트가 중복 프레임을 구분할 수 있도록 지원
- 데이터 프레임 내에 프레임 구분을 위한 일련 번호 부여

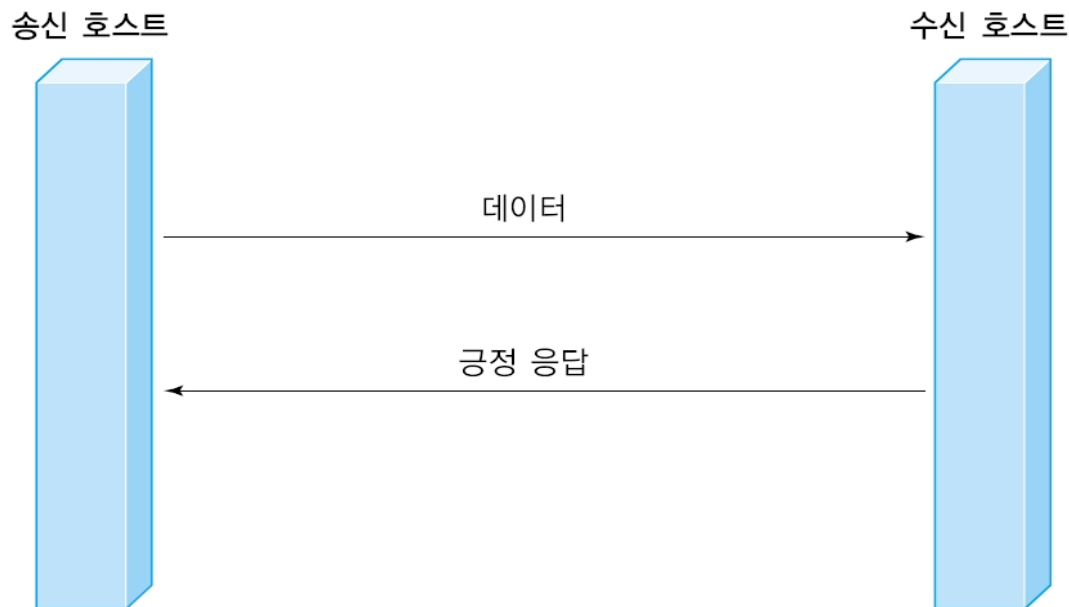


2절. 오류 제어

□ 전송 오류의 유형

■ 1) 정상적인 전송 [그림 4-8]

- 송신 호스트가 전송한 데이터 프레임이 수신 호스트에 오류 없이 도착
- 수신 호스트는 송신 호스트에게 **긍정 응답 프레임**을 회신
(Positive Acknowledgement, ACK)



[그림 4-8] 정상적인 데이터 전송



2절. 오류 제어

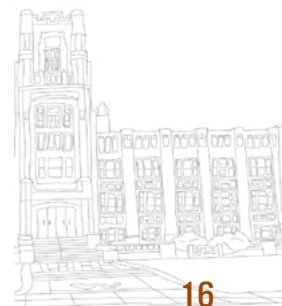
□ 전송 오류의 유형

■ 2) 프레임 변형 [그림 4-9]

- 송신 호스트가 전송한 데이터 프레임이 깨져서 도착
- 수신 호스트는 송신 호스트에게 **부정 응답 프레임을 회신** (Negative Acknowledgement, NAK)
- 송신 호스트는 원래의 데이터 프레임을 **재전송하여** 오류 복구



[그림 4-9] 프레임 변형 오류

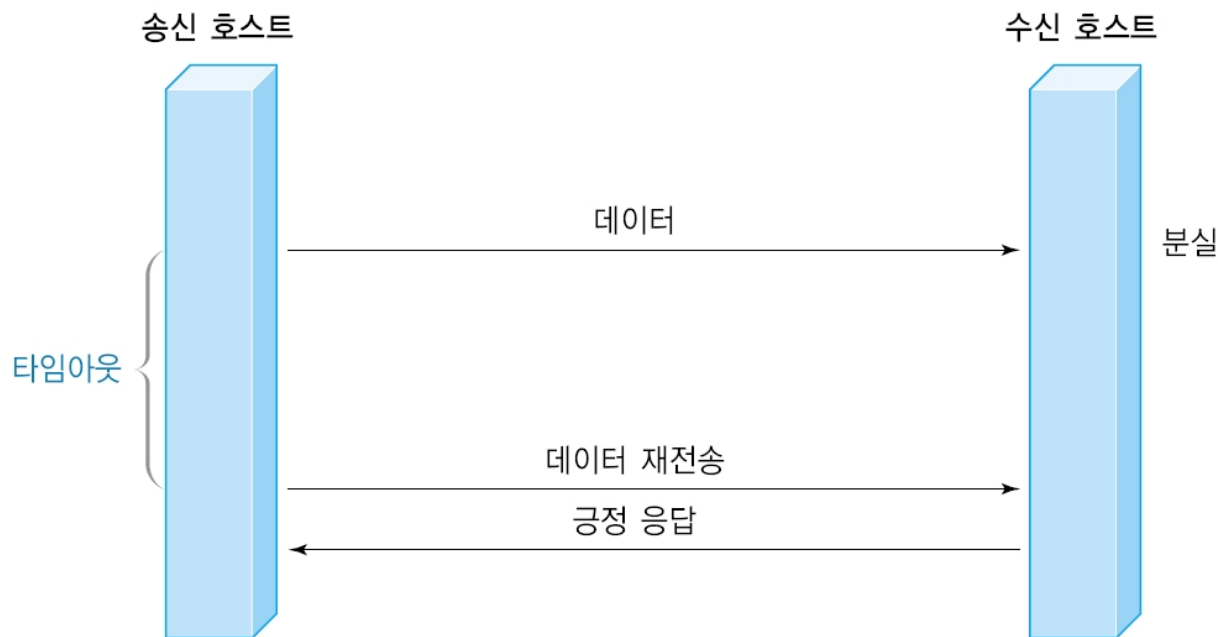


2절. 오류 제어

□ 전송 오류의 유형

■ 3) 프레임 분실 [그림 4-10]

- 송신 호스트가 전송한 **데이터 프레임이 네트워크에서 사라짐**
- 수신 호스트는 어떠한 데이터도 받지 않았으므로 송신 호스트에게 응답하지 않음
- 송신 호스트는 **타임아웃 기능으로** 원래의 데이터 프레임을 **재전송하여 오류 복구**



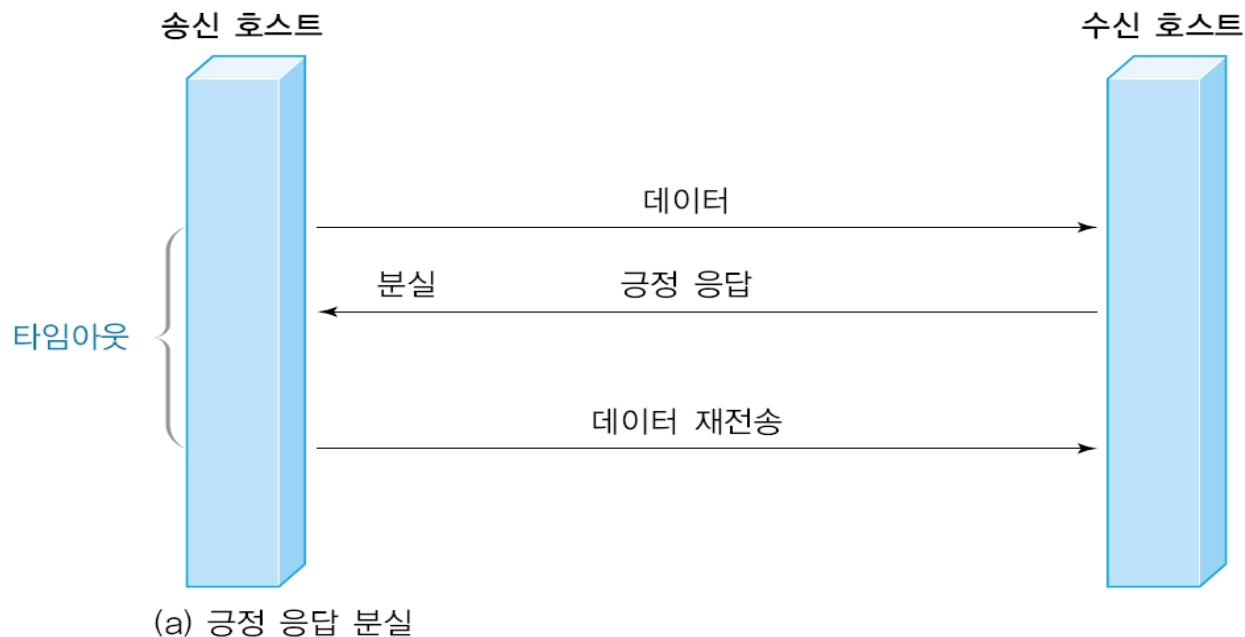
[그림 4-10] 프레임 분실 오류



2절. 오류 제어

□ 순서 번호

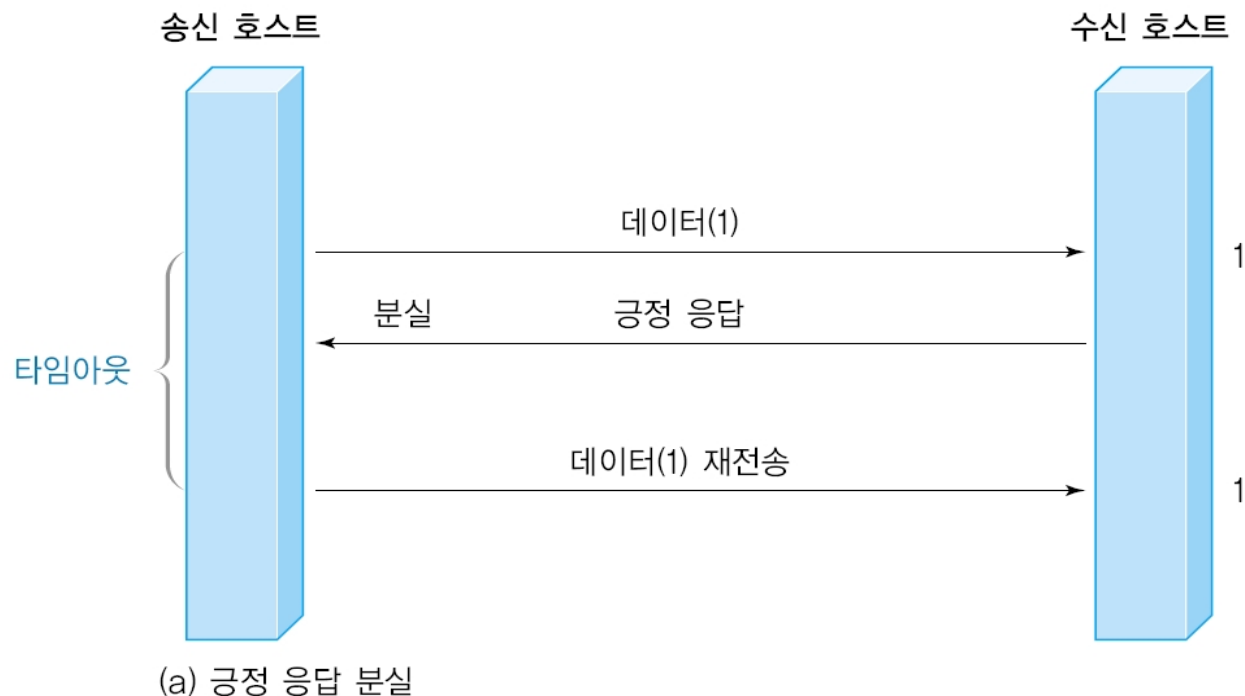
- 중복 수신 문제를 해결하기 위하여 데이터 프레임에게 부여되는 고유 번호
- 순서 번호의 필요성(순서 번호가 없을 경우)
 - 긍정 응답 프레임 분실에 따른 원래 데이터 재전송 [그림 4-11(a)]
 - 수신 호스트는 동일한 데이터 프레임을 중복 수신



2절. 오류 제어

□ 순서 번호

- 순서 번호에 의한 프레임 구분(순서 번호가 있는 경우)
 - 긍정 응답 프레임 분실에 따른 원래 데이터 재전송 [그림 4-12(a)]
 - 수신 호스트는 **순서 번호를 보고 같은 프레임이라고 판단할 수 있음**



2절. 오류 제어

□ 흐름 제어

- 수신 호스트가 감당할 수 있는 속도로 송신 호스트가 데이터를 전송하도록 제어
- 너무 빨리 전송하는 경우
 - 수신 호스트가 내부 버퍼에 보관하지 못할 수 있음
 - 이는 프레임 분실과 동일한 효과를 야기함
- 기본 원리
 - 기본 개념: 수신 호스트가 송신 호스트의 전송 시점을 통지함으로써 제어하는 방법
 - 대표적인 예: 슬라이딩 윈도우 프로토콜

