

네트워크 기본 개념 & 네트워크 인터페이스 계층

목차

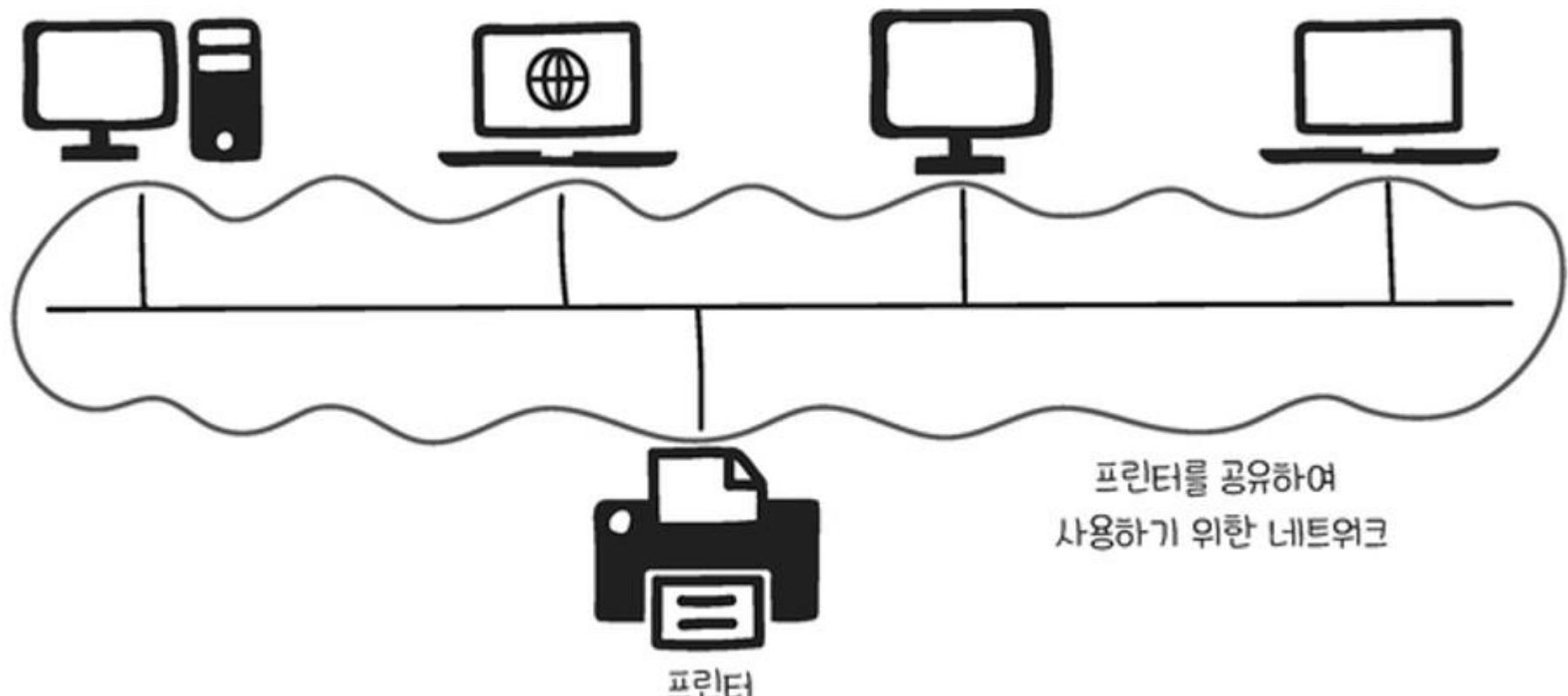
- 네트워크의 필요성
- 네트워크 유형과 종류
- 네트워크의 성능
- 네트워크 구조와 통신
- 네트워크 인터페이스 계층

네트워크의 필요성

- 네트워크가 왜 필요할까?

- 주변 장치 공유

- ✓ 주변 장치: 컴퓨터에 연결해서 사용하는 장치 (예: 프린터, 태블릿, 스마트폰 등)
 - ✓ 네트워크 회선을 공유하여 여러 대의 컴퓨터가 편리하게 인터넷을 사용하거나 프린터를 공동 사용할 수 있음.

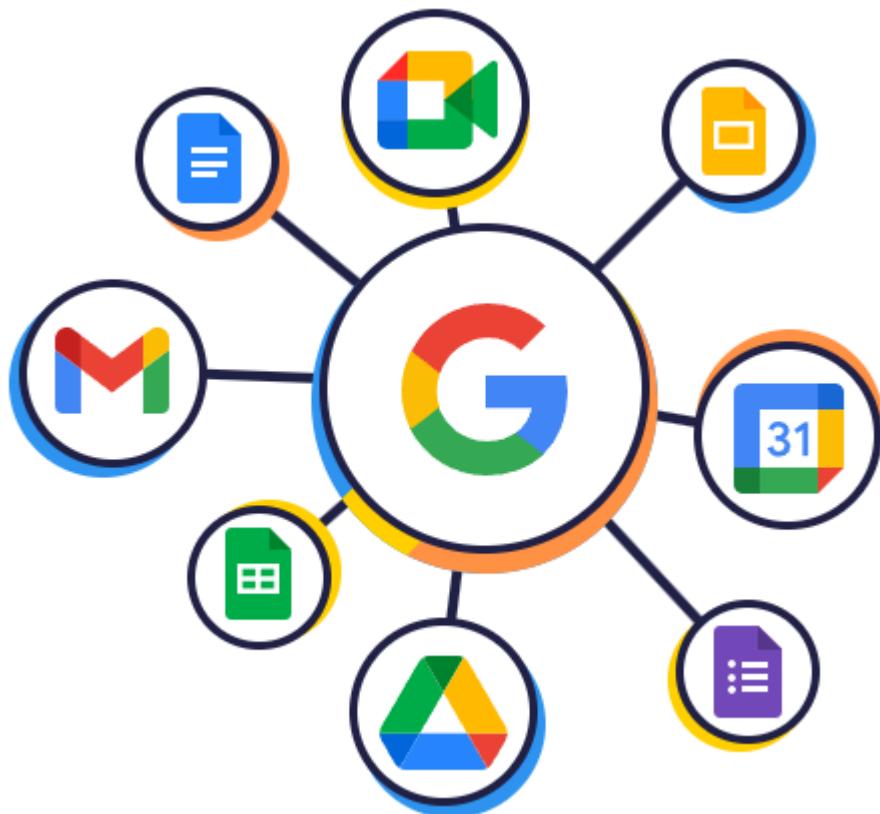


네트워크의 필요성

- 네트워크가 왜 필요할까?

- 데이터 공유

- ✓ 구글 스프레드시트와 문서를 웹으로 공유하여 공동 작업하고 데이터 관리할 수 있음
 - ✓ 클라우드를 공동으로 활용하여 데이터를 백업하거나 관리할 수 있음

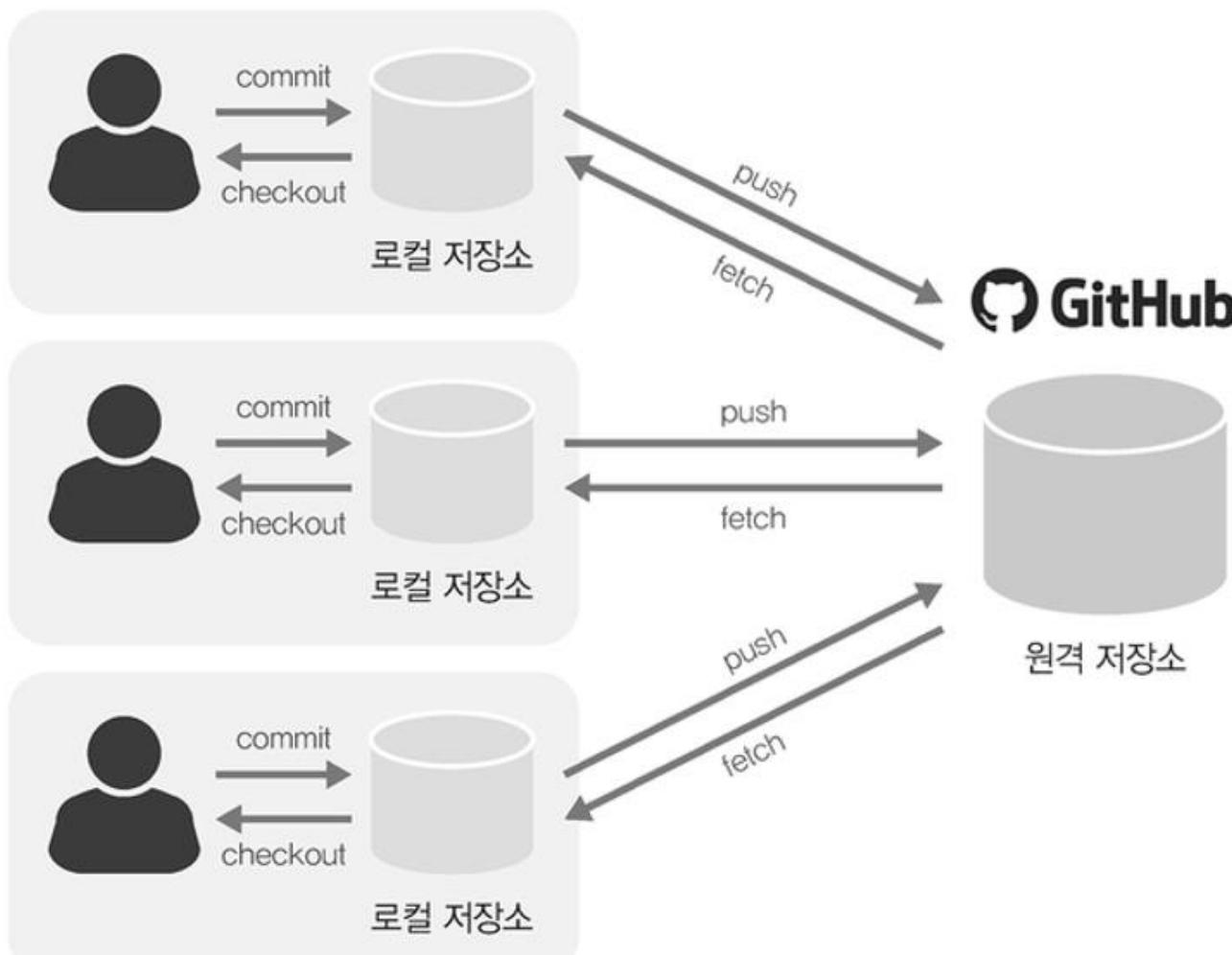


네트워크의 필요성

- 네트워크가 왜 필요할까?

- 공동 작업

- ✓ 원격 저장소와 로컬 저장소를 동기화하여 공동 개발 작업을 할 수 있음



네트워크 유형과 종류

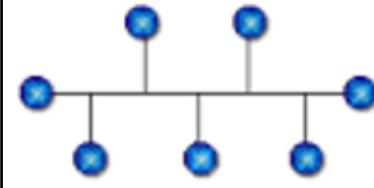
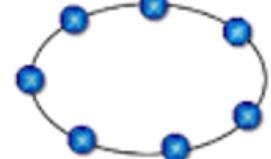
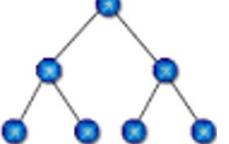
- 전송방식에 따른 네트워크 분류

네트워크 전송방식	특징	개념도
유니캐스팅	두 호스트 사이의 일대일 통신	<p>송신 호스트 수신 호스트</p> A diagram showing two hosts, 'a' and 'b', represented by grey circles. A horizontal line connects them, with a blue arrow pointing from 'a' to 'b'. Below the line, the text '송신 호스트' (transmitter host) is next to 'a', and '수신 호스트' (receiver host) is next to 'b'.
멀티캐스팅	일대다 전송 방식으로 비디오, 오디오, 화상 회의 등 다수의 사용자를 대상으로 하는 데이터 서비스	<p>송신 호스트 수신 호스트</p> A diagram showing host 'a' (grey circle) connected to a central switch or hub (grey circle). The switch is connected to hosts 'b', 'c', 'd', 'e', and 'f' (all blue circles). Blue arrows point from 'a' to each of the hosts. Below the diagram, 'a' is labeled '송신 호스트' (transmitter host) and the hosts b-f are labeled '수신 호스트' (receiver hosts).
멀티포인트 유니캐스팅	일대다 전송 방식으로 수신 호스트마다 다른 데이터를 전송	<p>송신 호스트 수신 호스트</p> A diagram similar to the previous one, but with three distinct sets of blue arrows originating from host 'a' to hosts 'b', 'c', and 'd' respectively, indicating different data streams.
브로드캐스팅	송신호스트가 네트워크의 모든 호스트에게 데이터 전송. 트래픽 양이 매우 큼.	<p>송신 호스트 수신 호스트</p> A diagram showing host 'a' (grey circle) connected to hosts 'b', 'c', 'd', and 'e' (all blue circles). Blue arrows point from 'a' to each of the hosts. Below the diagram, 'a' is labeled '송신 호스트' (transmitter host) and the hosts b-e are labeled '수신 호스트' (receiver hosts).

네트워크 유형과 종류



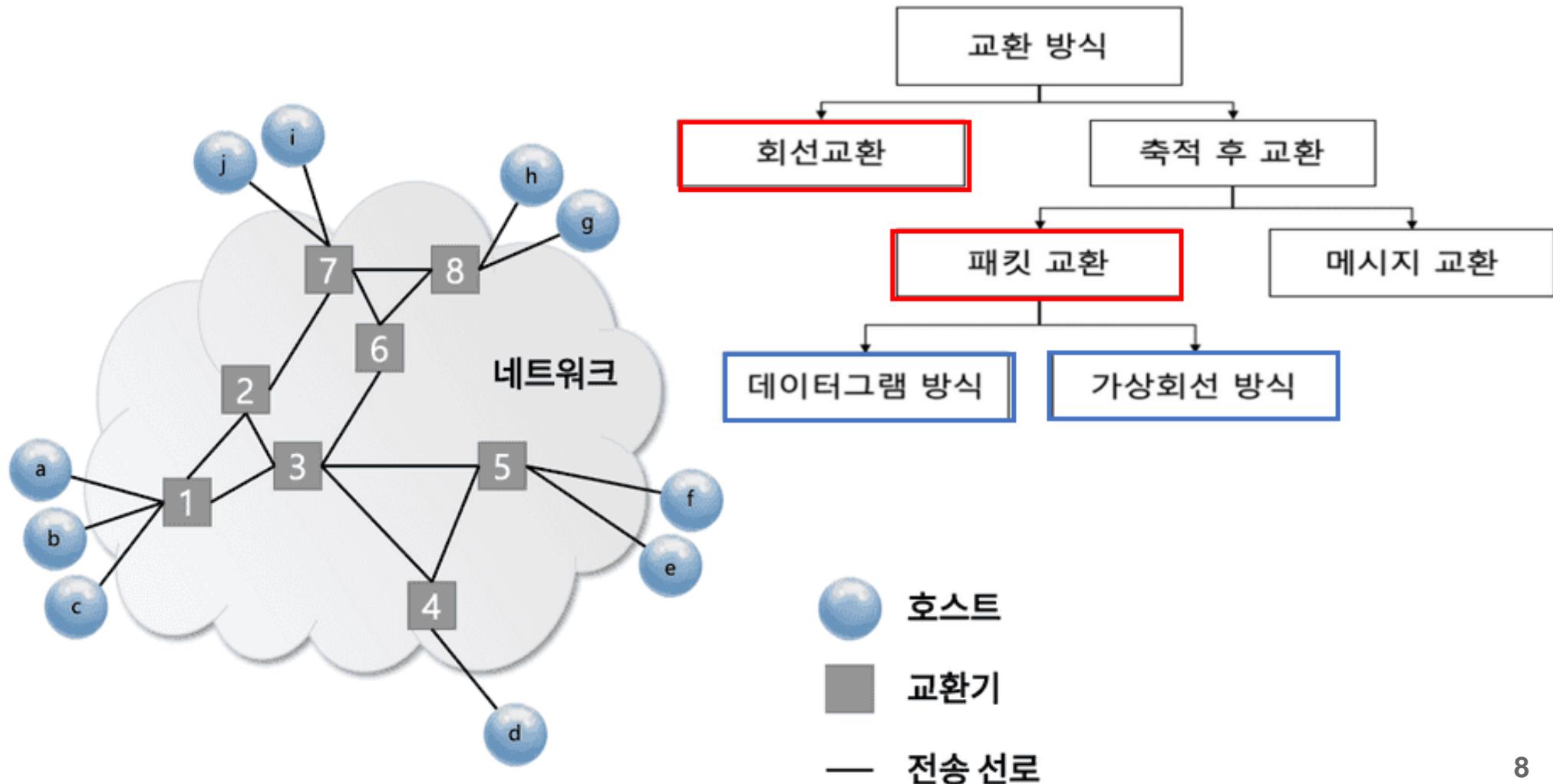
• 토플로지에 따른 네트워크 분류

유형	장점	단점	토플로지
성형	<ul style="list-style-type: none">- 장애 발견이 쉽고, 관리가 쉬움- 한 노드의 장애가 다른 네트워크 장비에 영향을 주지 않음	<ul style="list-style-type: none">- 허브에 장애가 발생하면 전체 네트워크에 영향 줌- 트래픽이 중앙에 집중되어 중앙 노드 부하가 큼	
망형	<ul style="list-style-type: none">- 동시에 다수의 노드 쌍이 통신할 수 있음- 한 노드에 장애가 발생해도 다른 노드에 영향을 주지 않음	<ul style="list-style-type: none">- 선로 구성이 복잡함- 노드 추가할 때 비용이 큼	
버스형	<ul style="list-style-type: none">- 구축이 간단하고 비용이 적게 들- 네트워크 확장이 쉬움- 한 노드에 장애가 발생해도 다른 노드에 영향을 주지 않음	<ul style="list-style-type: none">- 주 라인에 문제가 발생하면 전체 네트워크에 영향- 노드가 많아질 수록 네트워크 전송 속도가 느려짐	
링형	<ul style="list-style-type: none">- 통신 제어가 간단하고 신뢰성이 높음	<ul style="list-style-type: none">- 한 노드의 장애가 전체 네트워크에 영향 줌- 노드 추가나 변경이 어려움	
트리형	<ul style="list-style-type: none">- 성형에 비해 더 많은 노드 증설 가능	<ul style="list-style-type: none">- 성형에 비해 전송 거리 증가	

네트워크 유형과 종류

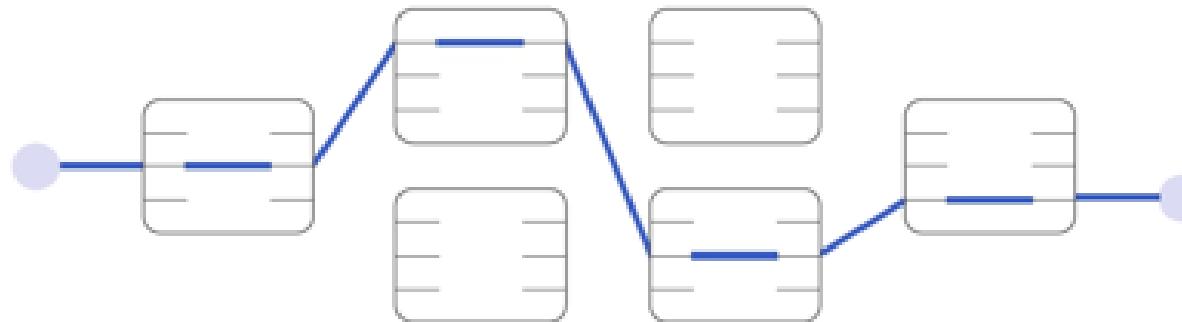
- 메시지 교환 방식에 따른 네트워크 분류

- 교환 시스템 (Switching System): 다수의 호스트들이 네트워크 전송 경로를 활용하여 메시지 전송할 수 있도록 호스트 간의 네트워크 통신로를 관리

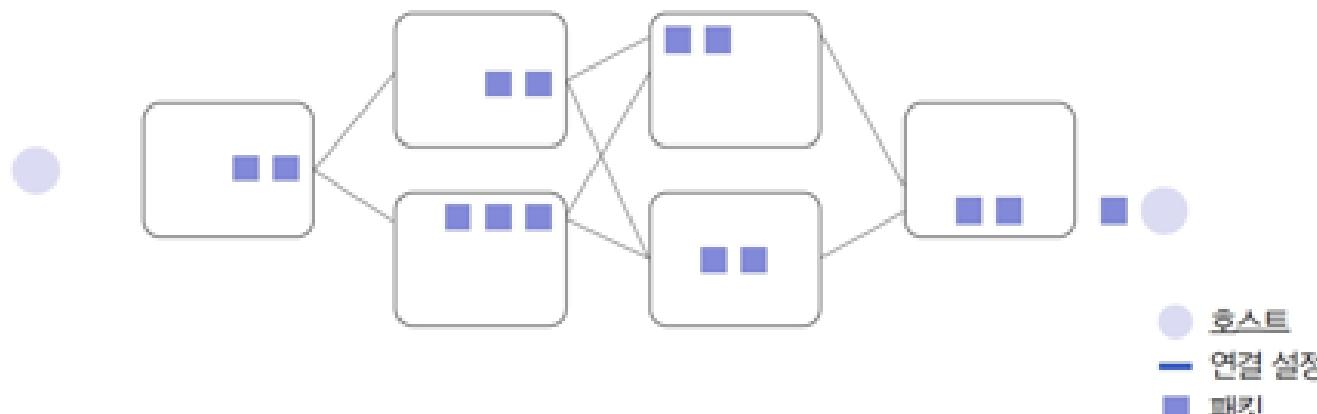


네트워크 유형과 종류

- 메시지 교환 방식에 따른 네트워크 분류
 - 대표적인 교환 시스템 (Switching System)
 - ✓ 유형1: 회선 교환 방식을 사용하는 "회선 교환 네트워크"
 - ✓ 유형2: 패킷 교환 방식을 사용하는 "패킷 교환 네트워크"



(a) 회선 교환



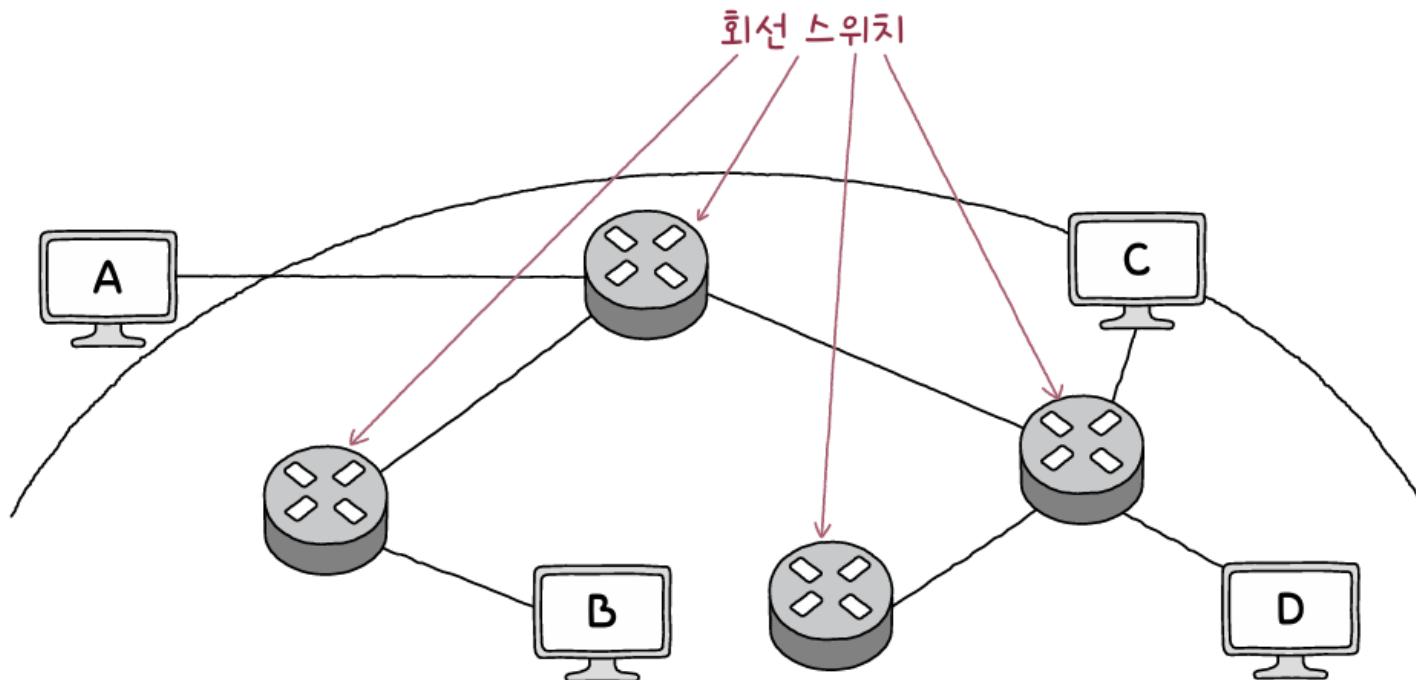
(b) 패킷 교환

네트워크 유형과 종류

- 메시지 교환 방식에 따른 네트워크 분류

- 회선 교환 (Circuit Switching) 방식

- ✓ 방식: 메시지 전송로인 회선 (Circuit) 을 설정하고, 설정된 전송로를 통해 메시지를 주고 받는 메시지 교환 방식
 - ✓ 장점: 두 호스트 사이에 연결을 확보한 후 메시지를 주고받으므로 주어진 시간 동안 전송되는 정보의 양이 비교적 일정하고 안정적임
 - ✓ 단점: 회선 이용 효율이 낮음 (정해진 회선이 항상 이용되는 것이 아니므로)
 - ✓ 대표적인 예: 전화망

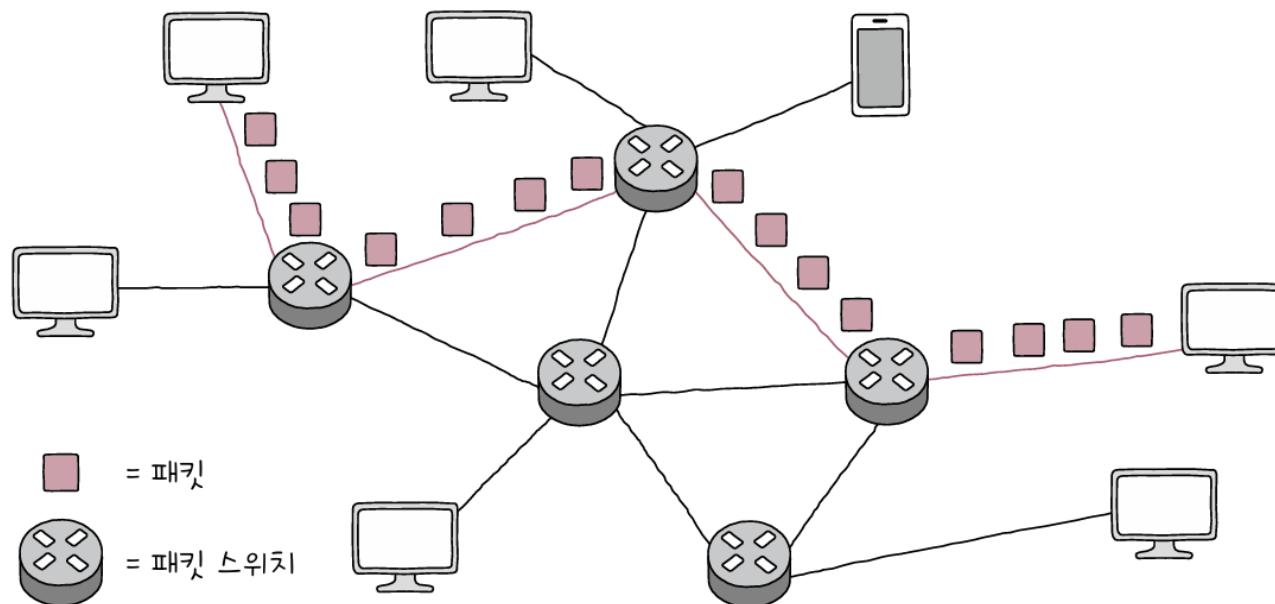


네트워크 유형과 종류

- 메시지 교환 방식에 따른 네트워크 분류

- 패킷 교환 (Packet Switching) 방식

- ✓ 회선 교환 방식의 비효율적인 회선 이용 효율 문제점을 해결.
 - ✓ 메시지를 패킷이라는 작은 단위로 쪼개서 전송 (예: 2GB 영화 파일을 다운로드 받을 때, 패킷 크기만큼 분할되어, 쪼개진 패킷이 컴퓨터에 도달한 후 재조립되어 2GB 영화 파일을 완성한 후 이용)
 - ✓ 패킷 스위치(예: 라우터, 스위치 등)는 패킷이 수신지까지 올바르게 도달할 수 있도록 최적의 경로를 결정하거나 패킷의 송수신지를 식별하고 에러 제어 & 흐름 제어.
 - ✓ 대표적인 예: 인터넷



네트워크 유형과 종류



- 메시지 교환 방식에 따른 네트워크 분류

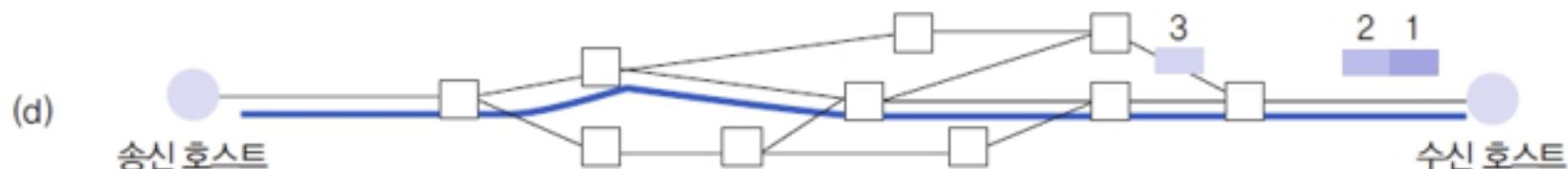
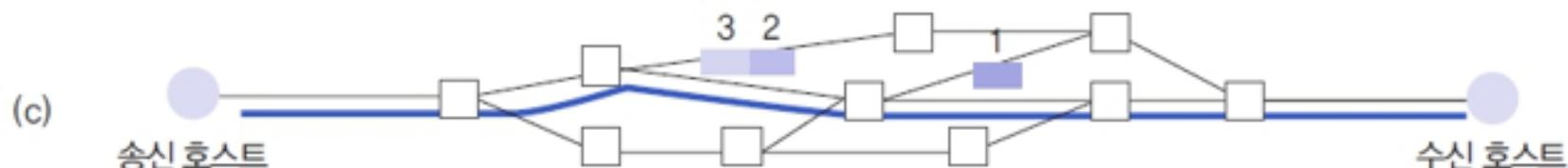
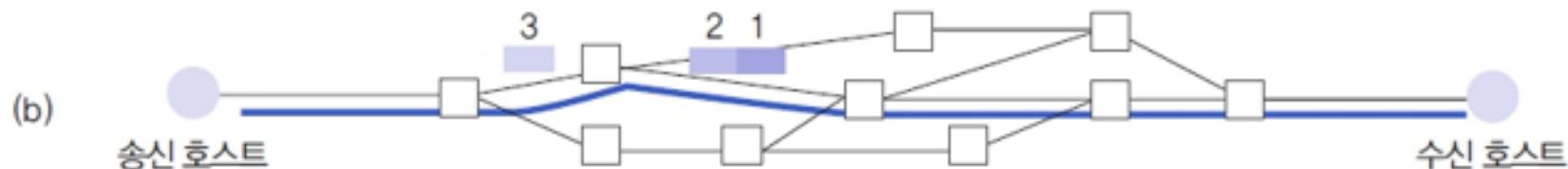
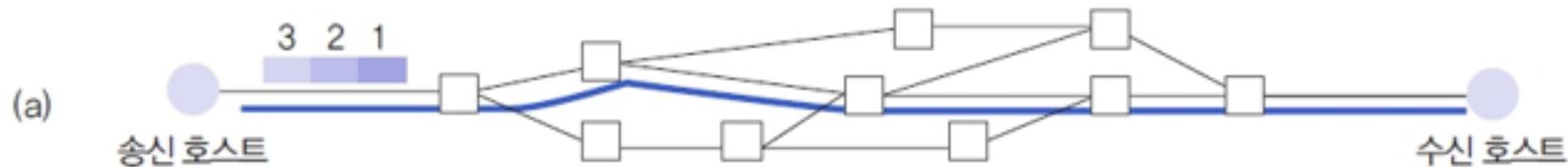
구분	회선 교환	패킷 교환
개념도		
링크의 점유	접속 전에 시그널링을 통해 미리 경로(=링크)를 정하고 통화 종료될 때까지 독점	사전 시그널링 없이 여러 사용자의 패킷들이 혼재되어 동일 링크를 공용
효율성	회선을 사용하지 않아도 회선 점유 (비효율적)	다른 사용자가 자유롭게 사용 (효율적)
서비스 품질(QoS)	음성 등 실시간 서비스, 고신뢰, 보안 강함	비음성 서비스, 신뢰성/보안성이슈
트래픽 폭주할 때	한계를 넘는 접속을 허용하지 않음	병목 현상으로 인한 지연과 정보 손실 발생 가능
라우팅	정적 라우팅	동적 라우팅

네트워크 유형과 종류

- 메시지 교환 방식에 따른 네트워크 분류

- 패킷교환 방식 중 “가상회선 방식”

- ✓ 미리 설정된 논리적 링크를 이용해 연결형 데이터 서비스 지원
 - ✓ 송수신 호스트 사이에 가상의 논리적 전송로를 설정하므로 모든 패킷의 전달 경로가 동일함

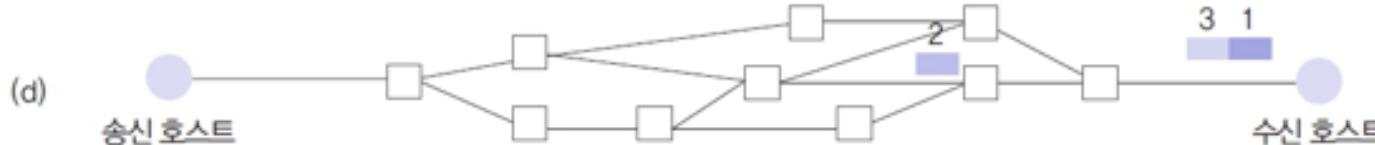
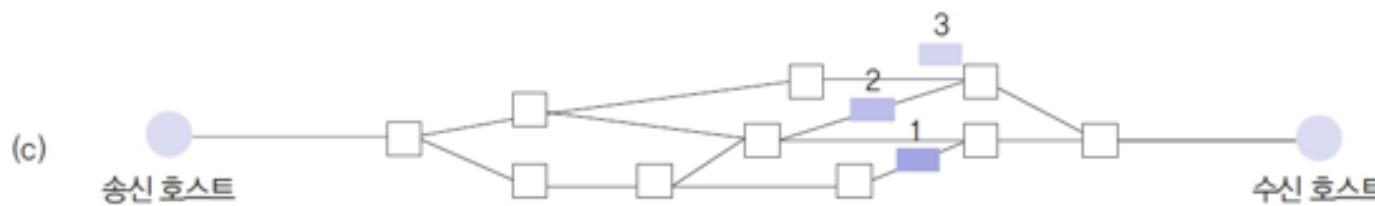


네트워크 유형과 종류

- 메시지 교환 방식에 따른 네트워크 분류

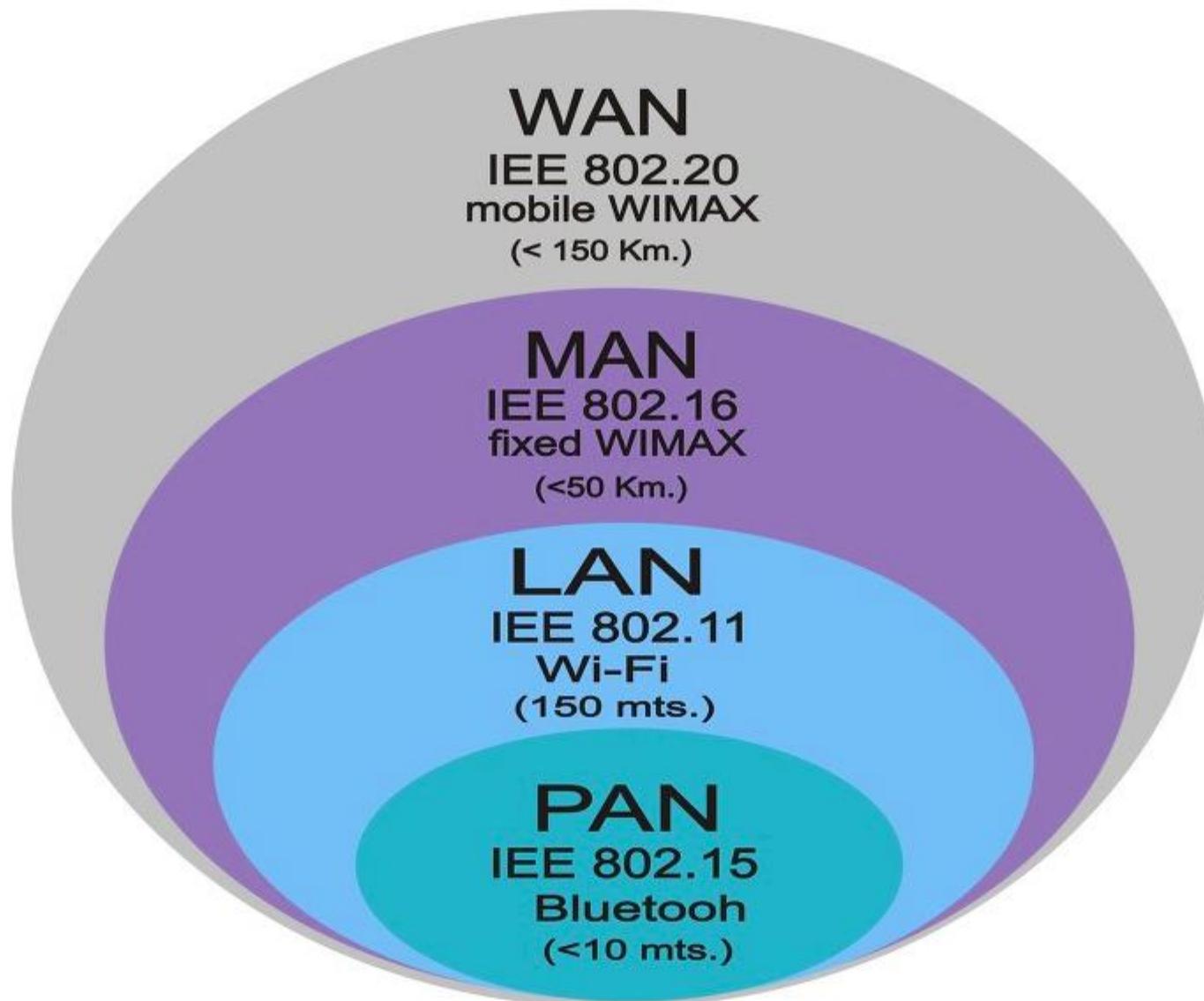
- 패킷교환 방식 중 “데이터그램 방식”

- ✓ 패킷을 독립적으로 전송하여 비연결형 서비스 제공
 - ✓ 전송할 정보의 양이 적거나 상대적으로 신뢰성이 중요하지 않은 환경에서 사용
 - ✓ 각 전송 경로의 속도는 네트워크 혼잡도에 따라 가변적임



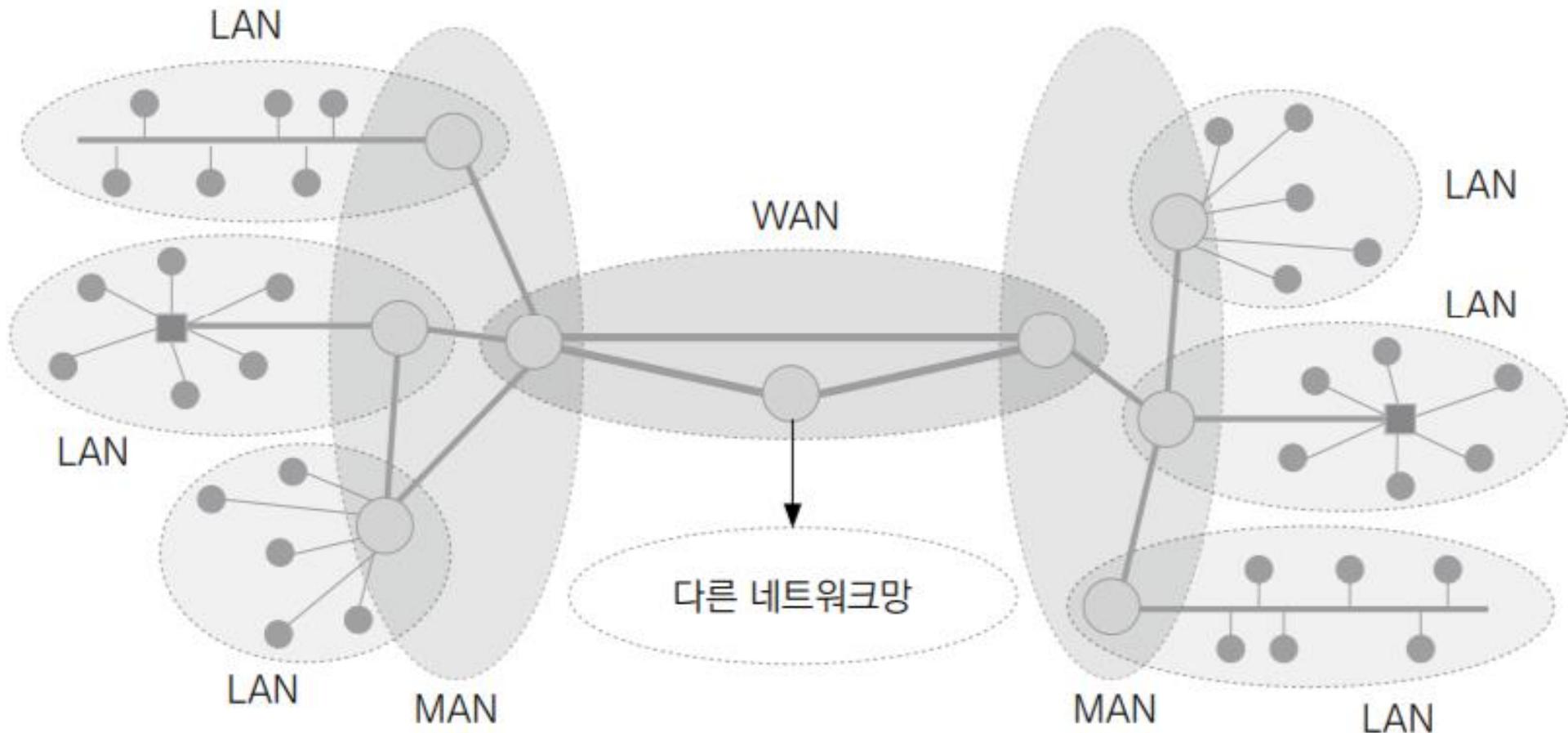
네트워크 유형과 종류

- 네트워크 크기에 따른 분류



네트워크 유형과 종류

- 네트워크 크기에 따른 분류
 - 네트워크 연결도



네트워크 유형과 종류

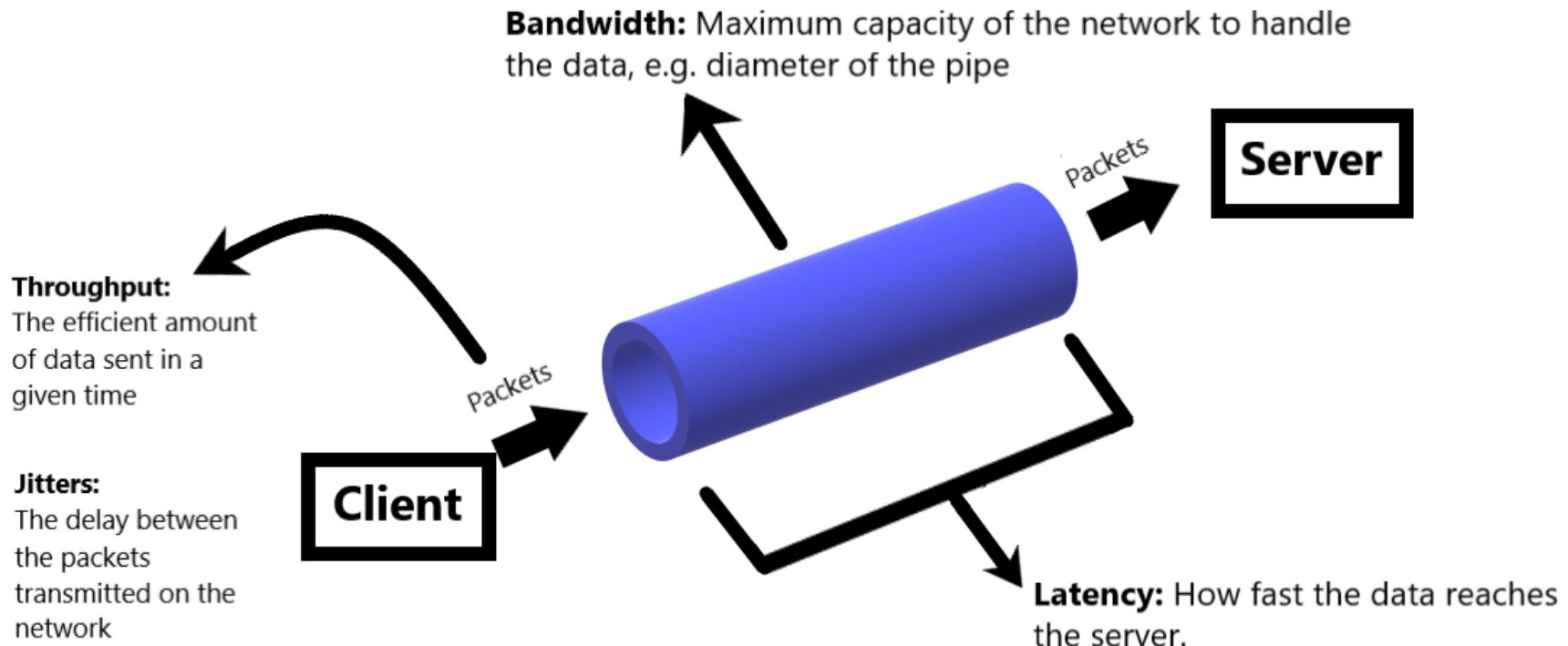


- 네트워크 크기에 따른 분류

분류	LAN	MAN	WAN
약어풀이	Local Area Network	Metropolitan Area Network	Wide Area Network
의미	제한된 지역 (주로 한 건물이나 캠퍼스 내)에서 컴퓨터와 장치들을 연결하는 네트워크	도시 내 여러 LAN을 연결한 네트워크	넓은 지리적 범위 (국가 또는 대륙간)를 커버하는 네트워크
장점	빠른 데이터 전송 속도, 지연 시간 짧음, 설치 및 유지 비용 저렴	넓은 서비스 커버리지, 높은 데이터 전송속도	매우 넓은 서비스 커버리지
단점	거리 제한	설치 및 유지 비용이 비싸고, 관리 복잡	설치 및 유지 비용이 매우 비싸고, 전송 속도 느리며, 지연 시간 큼
서비스 범위 (대략적)	10 km 미만	10~100km	150km 이상
사용처	학교, 병원, 교실, 커피숍	도시	주, 국가, 대륙

네트워크의 성능

- 네트워크의 주요 성능 평가 지표
 - 대역폭 (Bandwidth), 스루풋 (Throughput), 지연시간 (Latency), 지터 (Jitter)



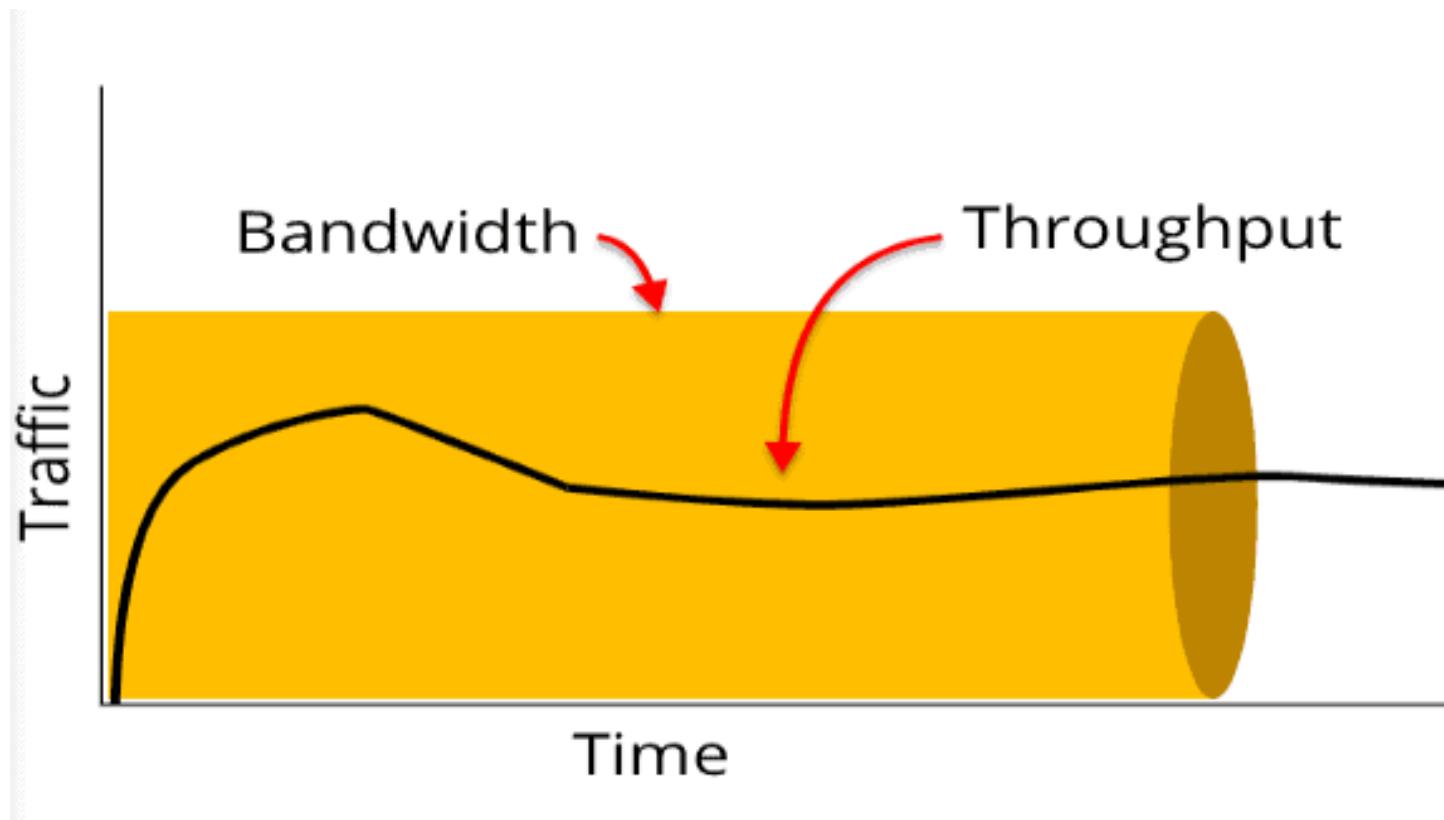
- 패킷에러율 (Packet Error Rate, PER), 비트에러율 (Bit Error Rate, BER), 요구되는 신호대잡음비 (Required Signal to Ratio, Required SNR), 채널용량 (Channel Capacity)

네트워크의 성능

- 네트워크의 주요 성능 평가 지표

- 대역폭 (Bandwidth)

- ✓ 네트워크에서 잠재적으로 동시에 전송될 수 있는 데이터 최대치
 - ✓ 네트워크에 주어진 대역폭 이상의 스루풋은 불가능

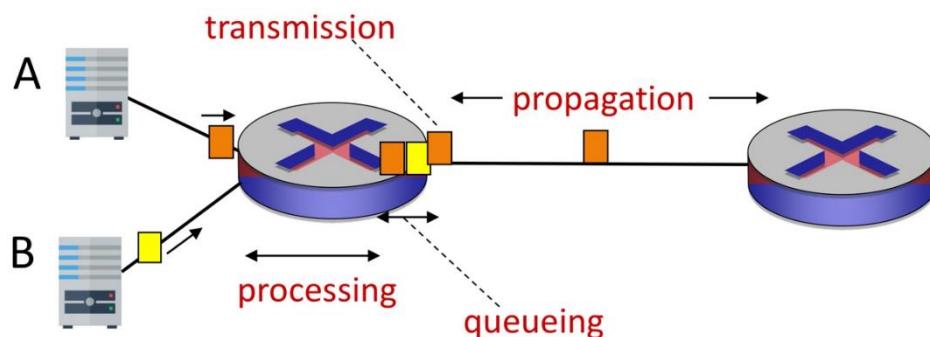
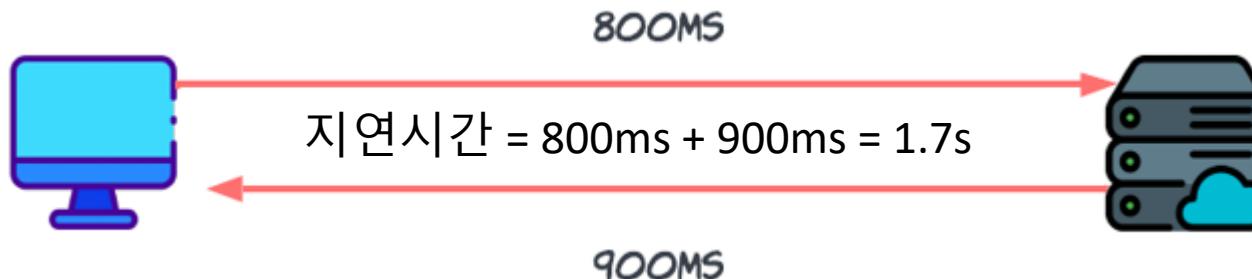


네트워크의 성능

- 네트워크의 주요 성능 평가 지표

- 지연시간 (Latency) = 대기시간

- ✓ 하나의 데이터 패킷이 송신 호스트에서 수신 호스트까지 가는데 걸린 시간
 - ✓ 매체 타입 (무선, 유선), 패킷 크기, 라우터의 패킷 처리 시간에 영향 받음.

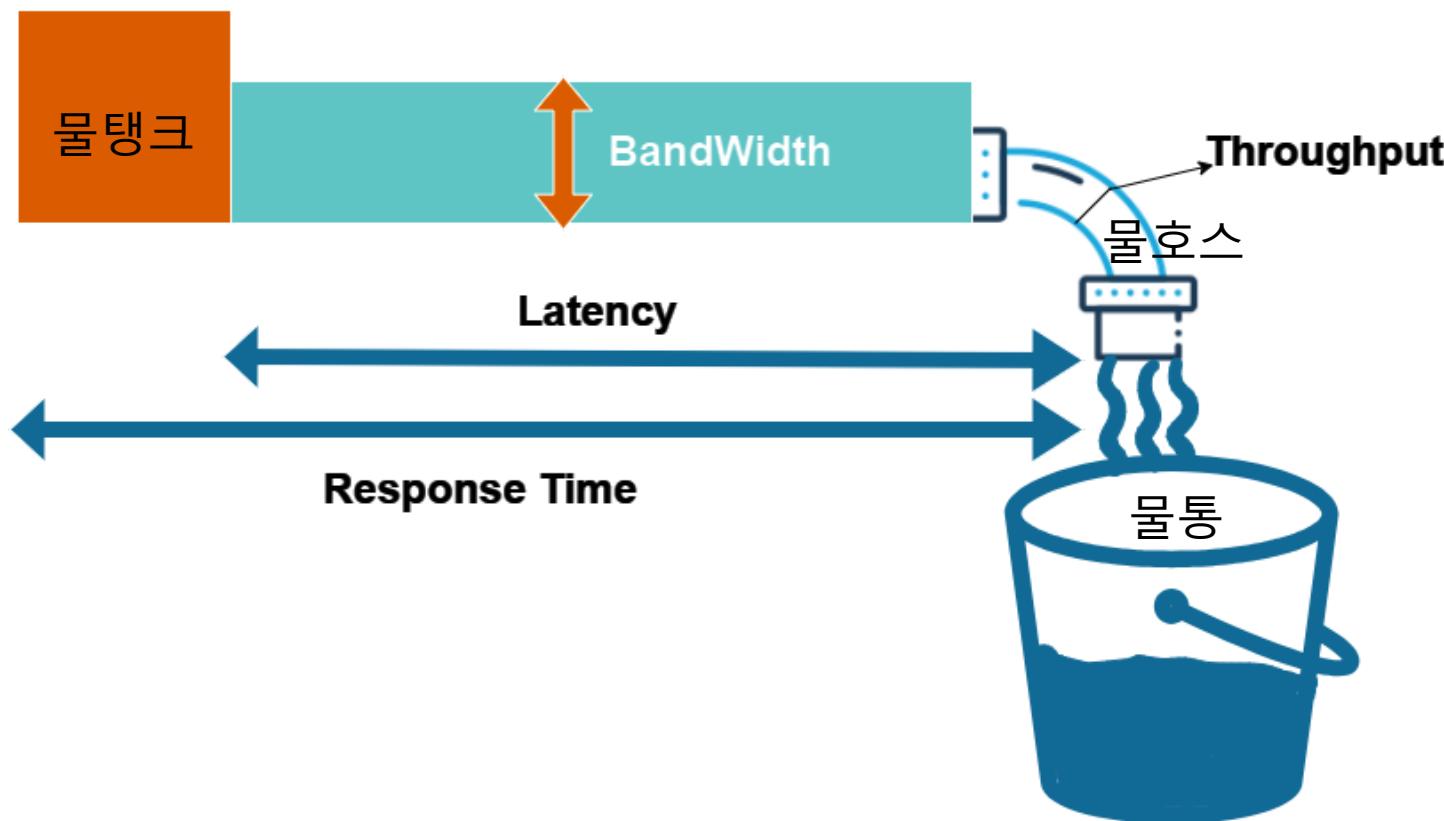


$$\text{End-to-end packet latency} = \sum (d_{\text{proc}} + d_{\text{queue}} + d_{\text{trans}} + d_{\text{prop}})$$



네트워크의 성능

- 네트워크의 주요 성능 평가 지표
 - 대역폭 (Bandwidth), 스루풋 (Throughput), 지연시간 (Latency)
 - ✓ 스루풋: 밸브의 잠긴 정도에 따라 결정되는 호스에서 물이 나오는 속도
 - ✓ 대역폭: 밸브를 최대한 열었을 때 흐르는 물의 양
 - ✓ 지연시간: 물탱크에서 물이 나와서 물통까지 들어가는 시간



네트워크의 성능



- 네트워크의 주요 성능 평가 지표

- 스루풋 (Throughput)

- ✓ 1초 당 성공적으로 처리할 수 있는 데이터의 양 (bit per second) = 성공적으로 전송한 데이터 사이즈 / 처리 시간

- 데이터율 (Data Rate) = 속도 (Speed)

- ✓ 1초당 처리할 수 있는 데이터의 양 (bit per second) = 전송한 데이터 사이즈 / 처리 시간

- 데이터율(속도)와 스루풋의 차이

- ✓ 뉴스기사: <https://www.hankookilbo.com/News/Read/A2021041909250004176>

10GB 가입했는데 실제는 100MB...KT는 실수라지만 내부에선 자
성의 목소리도

상품서비스 > 인터넷 > 요금제

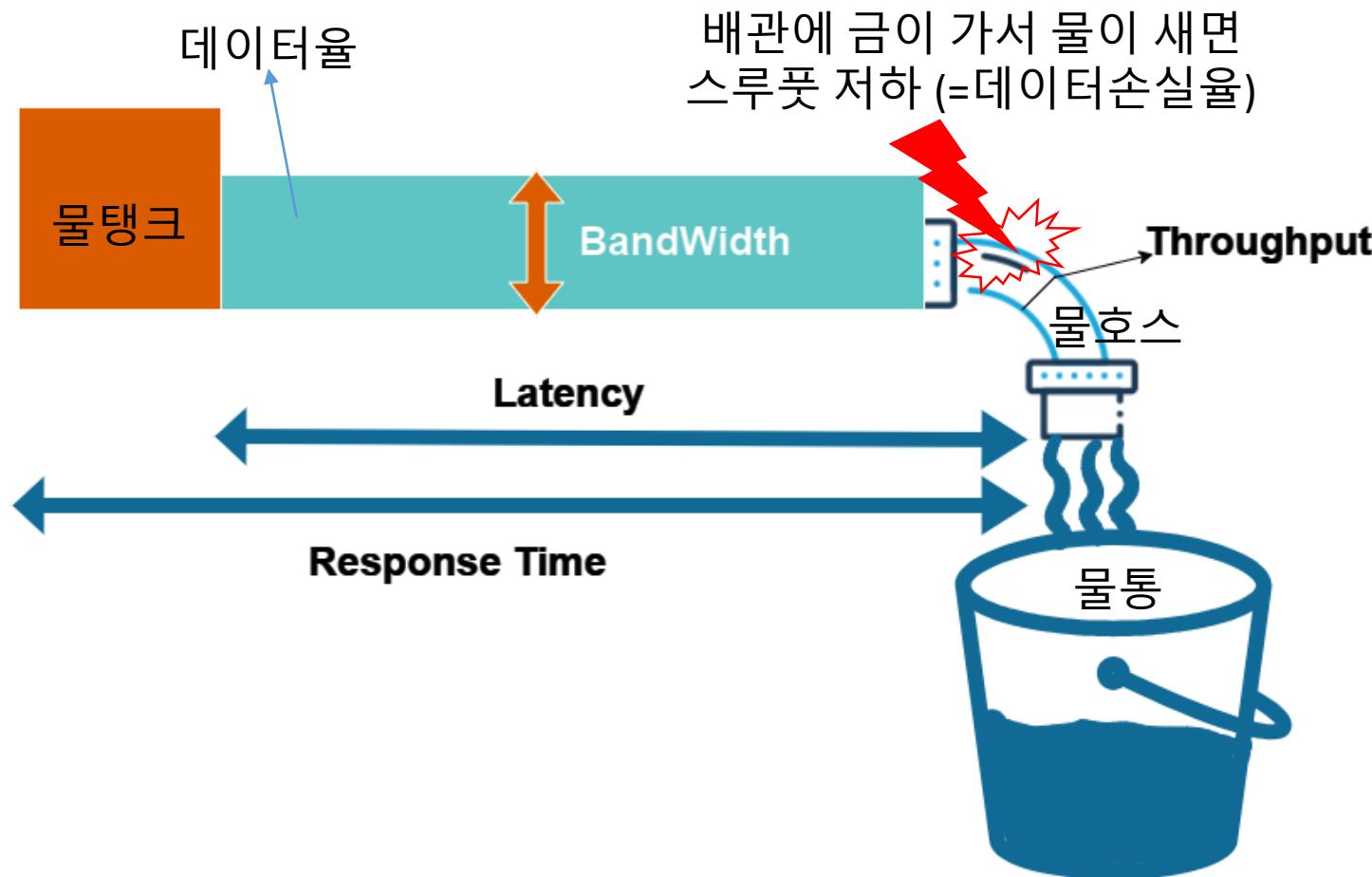
The screenshot shows a news article from Hankook Ilbo about KT's 10GiGA internet plan. The headline reads "10GB 가입했는데 실제는 100MB...KT는 실수라지만 내부에선 자성의 목소리도". Below the headline, there's a snippet of text: "안하늘 기자 [구독 +] 입력 2021.04.20 04:30 수정 2021.04.20 14:46 |". To the right, there's a promotional banner for the "10GiGA 인터넷 최대 10G" plan. The banner includes the price "월 88,000원" (per month), "최대 10Gbps 속도" (maximum speed 10Gbps), "최신 GiGA WiFi 공유기 2대 기본제공" (basic provision of 2 latest GiGA WiFi routers), and "일 사용량 1000GB 제공" (daily usage limit 1000GB). A red button at the bottom right says "전화주문" (Call for inquiry).

네트워크의 성능

- 네트워크의 주요 성능 평가 지표

- 스루풋과 데이터율 (속도)

- ✓ 이론적으로 전송할 수 있는 속도라면, 스루풋은 성공적으로 수신 호스트까지 데이터가 전달되는 속도
 - ✓ 스루풋 = 데이터율 – 데이터손실율



네트워크의 성능

- 네트워크의 주요 성능 평가 지표

- BER (Bit Error Rate)

- ✓ 비트 에러율 (%) = (에러 비트 수 / 전송한 비트 수) * 100

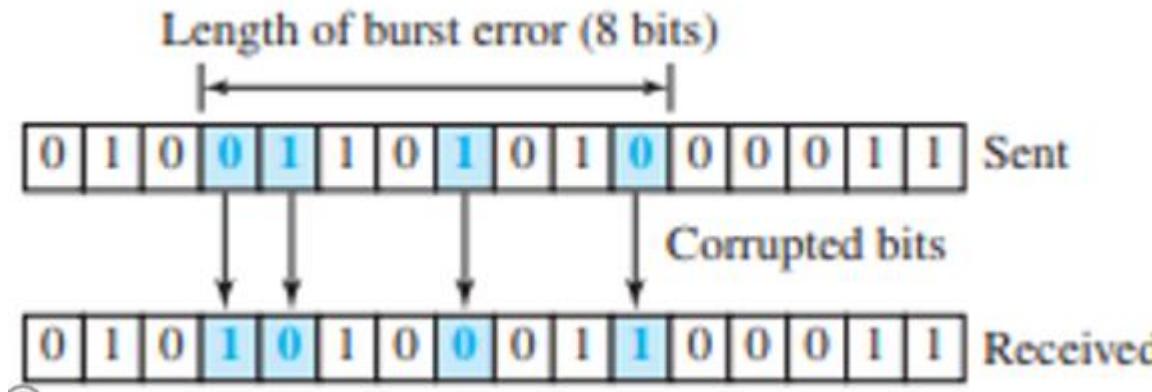
- Single bit error 케이스

- ✓ $\text{BER} = 12.5\% = 1/8 * 100$



- Burst bit error 케이스

- ✓ $\text{BER} = 25\% = 4/16 * 100$



네트워크의 성능

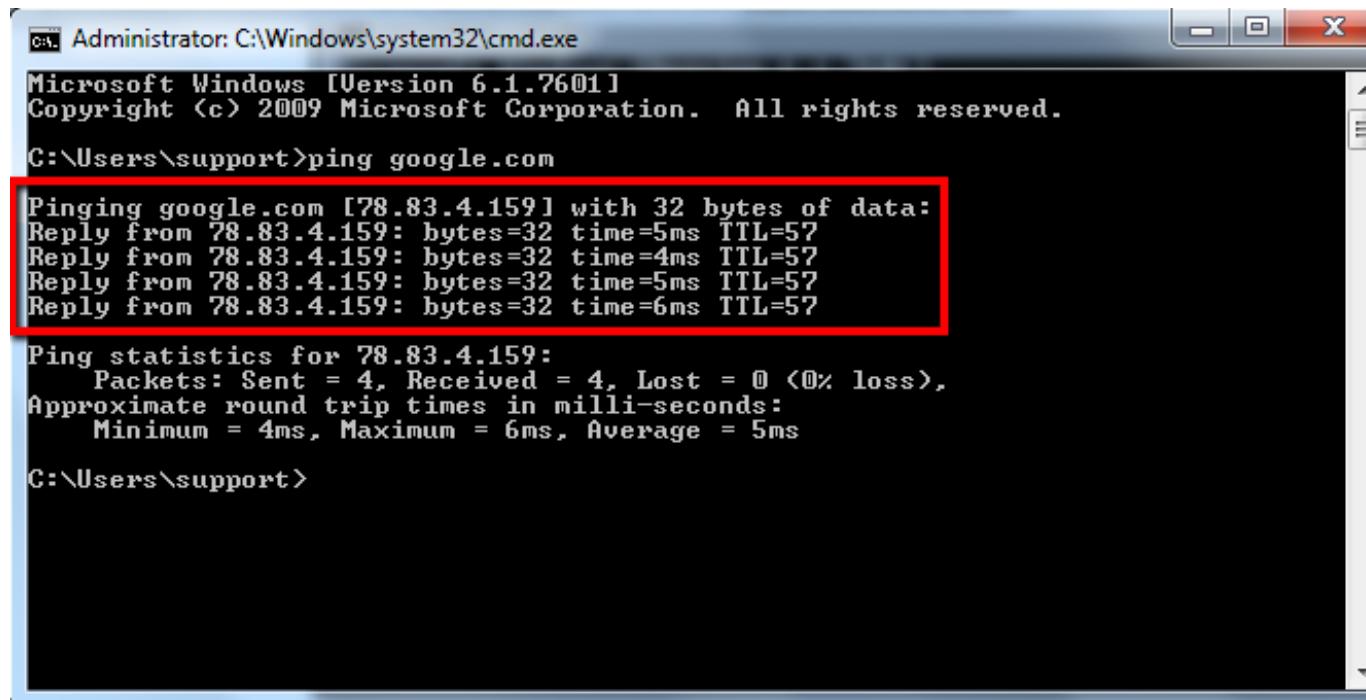
- 네트워크의 주요 성능 평가 지표

- PER (Packet Error Rate)

- ✓ 패킷 에러율(%) = (에러 패킷 수 / 전송한 패킷 수) * 100

- 네트워크 성능 측정 실습1

- Ping 을 이용한 latency 와 PER 측정



A screenshot of a Windows Command Prompt window titled "Administrator: C:\Windows\system32\cmd.exe". The window shows the output of a "ping" command to "google.com". The results are highlighted with a red box:

```
Microsoft Windows [Version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation. All rights reserved.

C:\Users\support>ping google.com

Pinging google.com [78.83.4.159] with 32 bytes of data:
Reply from 78.83.4.159: bytes=32 time=5ms TTL=57
Reply from 78.83.4.159: bytes=32 time=4ms TTL=57
Reply from 78.83.4.159: bytes=32 time=5ms TTL=57
Reply from 78.83.4.159: bytes=32 time=6ms TTL=57

Ping statistics for 78.83.4.159:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 4ms, Maximum = 6ms, Average = 5ms

C:\Users\support>
```

- Latency: 평균 5ms, 최소 4ms, 최대 6ms
 - PER: 0% (=에러 패킷 0개/4개 전송=손실율 0%)

네트워크의 성능

- 네트워크 성능 측정 실습2
 - iperf를 이용한 스루풋 성능 평가

- ✓ Iperf는 네트워크 장치의 스루풋을 측정하는 오픈 소스 도구
 - ✓ 서버 컴퓨터 (데이터 송신 노드)와 클라이언트 컴퓨터 (데이터 수신 노드) 양쪽에서 프로그램 실행해야 함
 - ✓ 지원하는 운영체제: Windows, macOS, Android, Ubuntu/Debian, Fedora/Red Hat/Cent OS, FreeBSD

네트워크의 성능

- 네트워크 성능 측정 실습2
 - iPerf를 이용한 스루풋 성능 평가

- ✓ 원도우용 iPerf 다운로드: <https://iperf.fr/iperf-download.php> → iperf3.1.3.64.zip 다운로드
- ✓ c드라이브의 work 폴더 만들고 그 하위에 저장한 후 압축 풀기

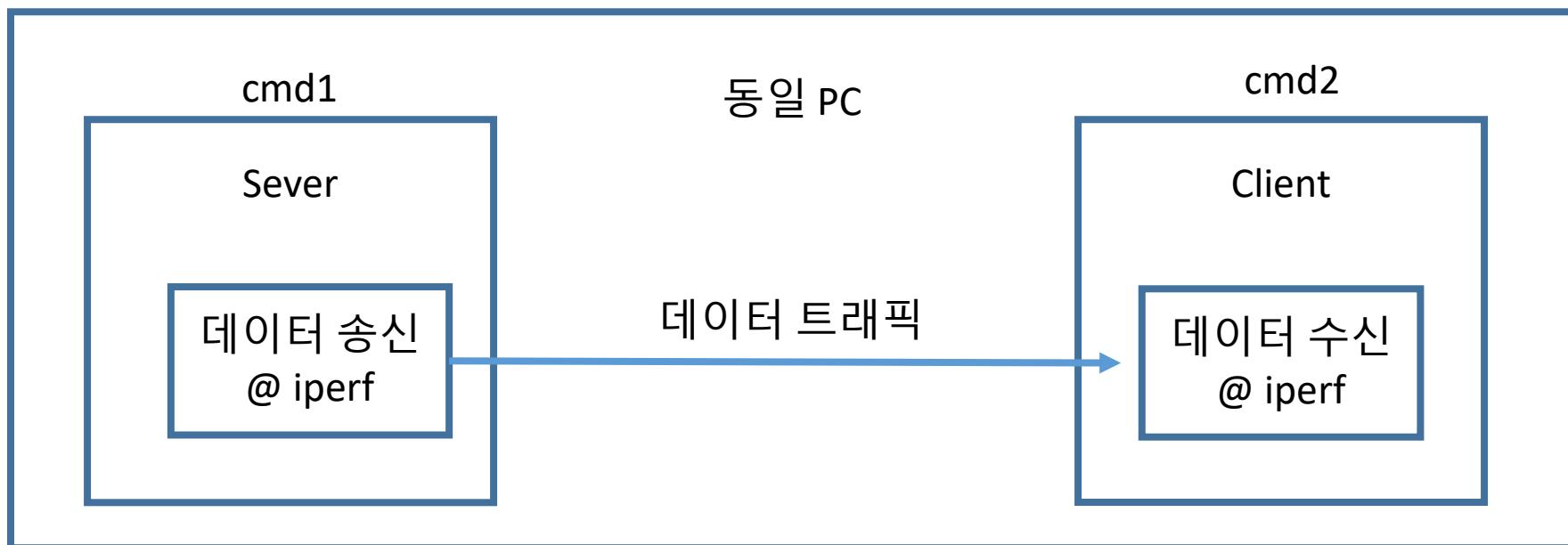
The screenshot shows a web browser with two tabs open. The left tab is a download page for iPerf, with the URL <https://iperf.fr/iperf-download.php> highlighted. It features a network diagram and sections for Windows, macOS, and Android, each with download links. The right tab is a file hosting site, with the URL <https://files.budman.pw> highlighted. It lists several iPerf3 binary files, with the file [iperf3.1.3_64.zip](#) highlighted.

File Name	Size	Date
iperf3.1.1_32.zip	1.29MB	2017-05-23 03:06:...
iperf3.1.1_64.zip	1.23MB	2017-05-23 03:06:...
iperf3.1.2_32.zip	1.29MB	2017-05-23 03:06:...
iperf3.1.2_64.zip	1.24MB	2017-05-23 03:06:...
iperf3.1.3_32.zip	1.34MB	2017-05-23 03:06:...
iperf3.1.3_64.zip	1.28MB	2017-05-23 03:06:...
iperf3.1.4_32.zip	1.29MB	2017-05-23 03:06:...

네트워크의 성능

- 네트워크 성능 측정 실습2
 - iperf를 이용한 스루풋 성능 평가

✓ 동일 PC에서 두 개의 cmd 창을 띄워 놓고, 루프백 (동일 PC에서 송신하고 다시 수신) 테스트 조건



네트워크의 성능

- 네트워크 성능 측정 실습2
 - iPerf를 이용한 스루풋 성능 평가
 - ✓ 서버 실행하기 → iperf3.exe -s

```
C:\ 명령 프롬프트 - iperf3.exe -s
C:\#>cd work
C:\#work>cd iperf3.1.3_64
C:\#work\iperf3.1.3_64>iperf3.exe -s
-----
Server listening on 5201
-----
Accepted connection from 192.168.0.37, port 60426
[ 5] local 192.168.0.37 port 5201 connected to 192.168.0.37 port 60427
[ ID] Interval           Transfer     Bandwidth
[ 5]  0.00-1.00   sec    593 MBytes   4.97 Gbits/sec
[ 5]  1.00-2.00   sec    578 MBytes   4.85 Gbits/sec
[ 5]  2.00-3.00   sec    602 MBytes   5.05 Gbits/sec
[ 5]  3.00-4.00   sec    600 MBytes   5.03 Gbits/sec
[ 5]  4.00-5.00   sec    605 MBytes   5.07 Gbits/sec
[ 5]  5.00-6.00   sec    573 MBytes   4.81 Gbits/sec
[ 5]  6.00-7.00   sec    581 MBytes   4.87 Gbits/sec
[ 5]  7.00-8.00   sec    608 MBytes   5.10 Gbits/sec
[ 5]  8.00-9.00   sec    584 MBytes   4.90 Gbits/sec
[ 5]  9.00-10.00  sec    600 MBytes   5.03 Gbits/sec
[ 5] 10.00-10.00  sec   1.19 MBytes   4.66 Gbits/sec
-----
[ ID] Interval           Transfer     Bandwidth
[ 5]  0.00-10.00  sec   0.00 Bytes  0.00 bits/sec
[ 5]  0.00-10.00  sec  5.79 GBytes  4.97 Gbits/sec
                               sender
                               receiver
-----
Server listening on 5201
-----
```

네트워크의 성능

- 네트워크 성능 측정 실습2
 - iPerf를 이용한 스루풋 성능 평가
- ✓ IP 주소 확인하기 → ipconfig /all

```
C:\#work\iperf3.1.3_64>ipconfig /all

Windows IP 구성

호스트 이름 . . . . . [REDACTED]
주 DNS 접미사 . . . . . [REDACTED]
노드 유형 . . . . . 훈성
IP 라우팅 사용 . . . . . 예
WINS 프록시 사용 . . . . . 아니요

무선 LAN 어댑터 Wi-Fi 2:

연결별 DNS 접미사 . . . . .
설명 . . . . . Realtek 8821CU Wireless LAN 802.11ac USB NIC
물리적 주소 . . . . . [REDACTED]
DHCP 사용 . . . . . 예
자동 구성 사용 . . . . . 예
링크-로컬 IPv6 주소 . . . . . [REDACTED] (기본 설정)
IPv4 주소 . . . . . 192.168.0.37(기본 설정)
서브넷 마스크 . . . . . 255.255.255.0
임대 시작 날짜 . . . . . 2024년 9월 19일 목요일 오후 4:10:17
임대 만료 날짜 . . . . . 2024년 9월 23일 월요일 오후 10:10:11
기본 게이트웨이 . . . . . 192.168.0.1
DHCP 서버 . . . . . 192.168.0.1
DHCPv6 IAID . . . . . 455085008
DHCPv6 클라이언트 DUID . . . . . 00-01-00-01-27-F7-21-F9-3C-7C-ED-3A-00
DNS 서버 . . . . . 210.125.88.1
168.126.63.1
Tcpip를 통한 NetBIOS . . . . . 사용
```

네트워크의 성능

- 네트워크 성능 측정 실습2
 - iPerf를 이용한 스루풋 성능 평가

- ✓ 클라이언트 실행하기 → iperf3.exe -c 서버IP주소

```
C:\work\iperf3.1.3_64> iperf3.exe -c 192.168.0.37
Connecting to host 192.168.0.37, port 5201
[ 4] local 192.168.0.37 port 60427 connected to 192.168.0.37 port 5201
[ ID] Interval           Transfer     Bandwidth
[ 4]  0.00-1.00   sec   594 MBytes  4.97 Gbits/sec
[ 4]  1.00-2.00   sec   578 MBytes  4.85 Gbits/sec
[ 4]  2.00-3.00   sec   602 MBytes  5.05 Gbits/sec
[ 4]  3.00-4.00   sec   600 MBytes  5.08 Gbits/sec
[ 4]  4.00-5.00   sec   605 MBytes  5.07 Gbits/sec
[ 4]  5.00-6.00   sec   573 MBytes  4.81 Gbits/sec
[ 4]  6.00-7.00   sec   581 MBytes  4.87 Gbits/sec
[ 4]  7.00-8.00   sec   609 MBytes  5.10 Gbits/sec
[ 4]  8.00-9.00   sec   583 MBytes  4.90 Gbits/sec
[ 4]  9.00-10.00  sec   600 MBytes  5.03 Gbits/sec
-
[ ID] Interval          Transfer     Bandwidth
[ 4]  0.00-10.00  sec  5.79 GBytes  4.97 Gbits/sec
[ 4]  0.00-10.00  sec  5.79 GBytes  4.97 Gbits/sec
                                         sender
                                         receiver
iperf Done.

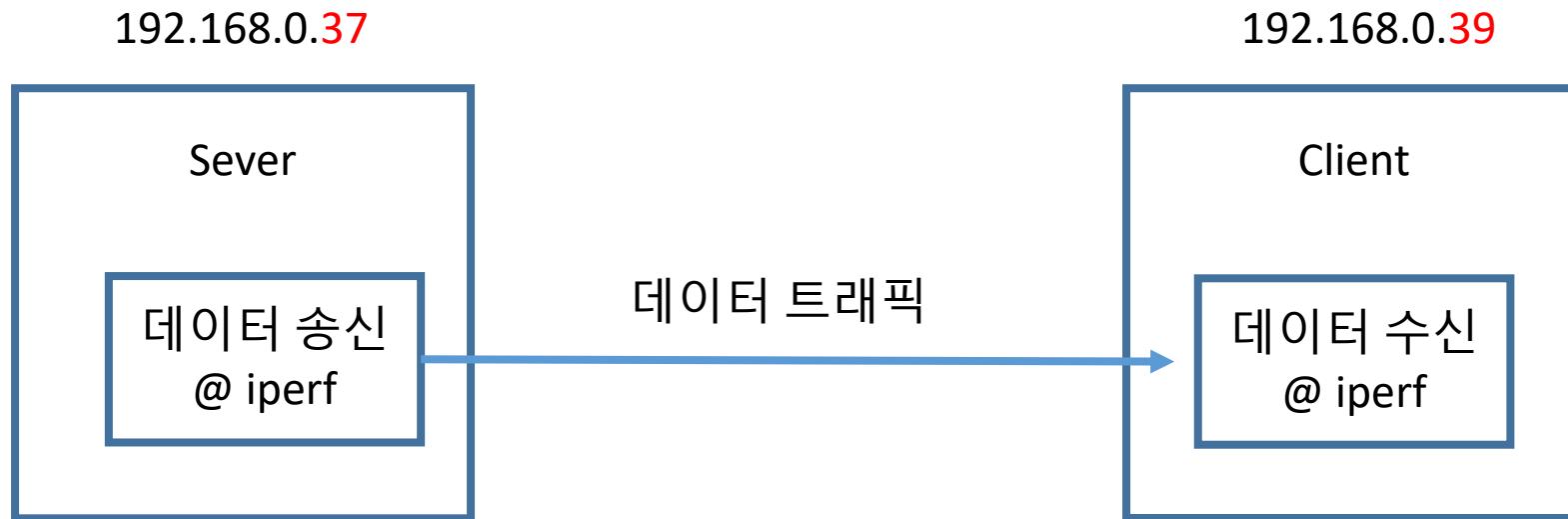
C:\work\iperf3.1.3_64>
```

측정한 서버-클라이언트 네트워크는 약 4.97 Gbps 속도 지원 → 동일 PC 루프백이어서 빠름

네트워크의 성능

- 네트워크 성능 측정 실습2
 - iperf를 이용한 스루풋 성능 평가

- ✓ 두 개의 다른 PC를 사용해서 이더넷을 통해 데이터 트래픽을 전송하는 경우
 - ✓ 서버 (192.168.0.37)가 iperf로 데이터 송신하고, 클라이언트 (192.168.0.39)가 iperf로 데이터 수신.



네트워크의 성능

- 네트워크 성능 측정 실습2
 - iPerf를 이용한 스루풋 성능 평가

```
-----  
Server listening on 5201  
Accepted connection from 192.168.0.39, port 53522  
[ 5] local 192.168.0.37 port 5201 connected to 192.168.0.39 port 53523  
[ ID] Interval Transfer Bandwidth  
[ 5] 0.00-1.00 sec 7.73 MBytes 64.8 Mbits/sec  
[ 5] 1.00-2.00 sec 8.98 MBytes 75.4 Mbits/sec  
[ 5] 2.00-3.00 sec 9.01 MBytes 75.5 Mbits/sec  
[ 5] 3.00-4.00 sec 9.12 MBytes 76.5 Mbits/sec  
[ 5] 4.00-5.00 sec 8.66 MBytes 72.7 Mbits/sec  
[ 5] 5.00-6.00 sec 4.62 MBytes 38.7 Mbits/sec  
[ 5] 6.00-7.00 sec 7.62 MBytes 64.1 Mbits/sec  
[ 5] 7.00-8.00 sec 9.11 MBytes 76.2 Mbits/sec  
[ 5] 8.00-9.00 sec 9.16 MBytes 76.8 Mbits/sec  
[ 5] 9.00-10.00 sec 9.01 MBytes 75.8 Mbits/sec  
[ 5] 10.00-10.06 sec 549 KBytes 77.9 Mbits/sec  
-----  
[ ID] Interval Transfer Bandwidth  
[ 5] 0.00-10.06 sec 0.00 Bytes 0.00 bits/sec  
[ 5] 0.00-10.06 sec 83.6 MBytes 69.7 Mbits/sec  
-----  
sender  
receiver  
-----  
Server listening on 5201  
-----
```

측정한 서버-클라이언트 네트워크는 약 69.7 Mbps 속도 지원

네트워크의 성능

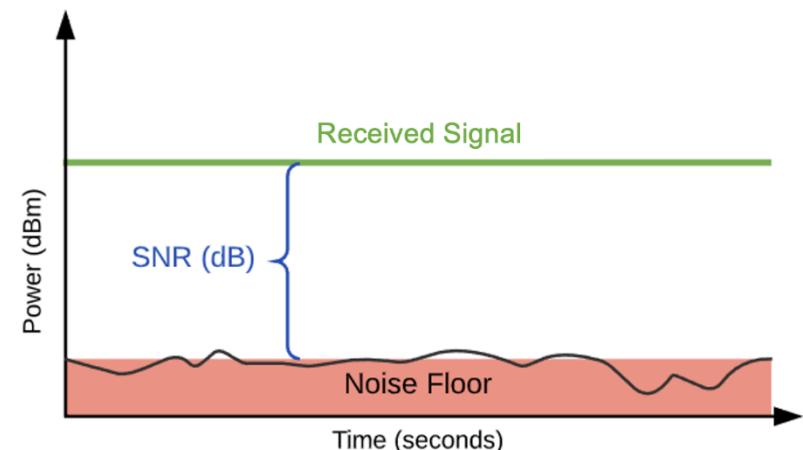


- 네트워크의 주요 성능 평가 지표

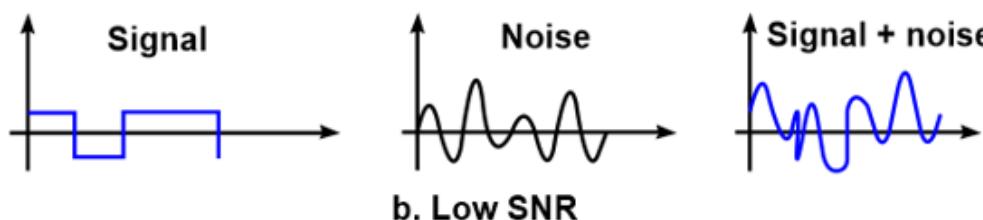
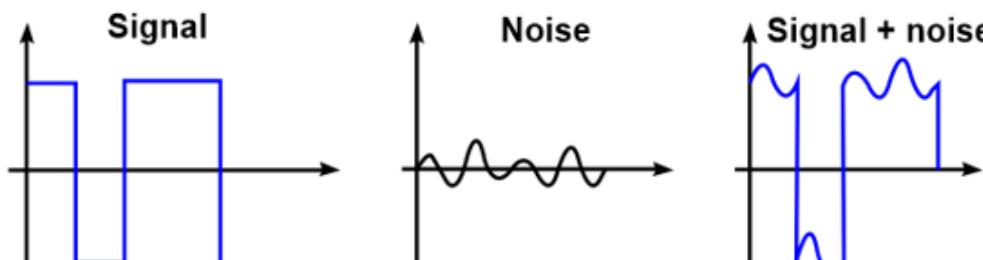
- 신호 대 잡음 비 (=SNR, Signal to Noise Ratio)

$$SNR = P_{\text{signal}} / P_{\text{noise}}$$

$$SNR_{\text{dB}} = 10 \log_{10} SNR$$



- 수신한 신호의 정보 복원 가능 여부는 SNR과 오류 정정 능력에 의해 결정됨

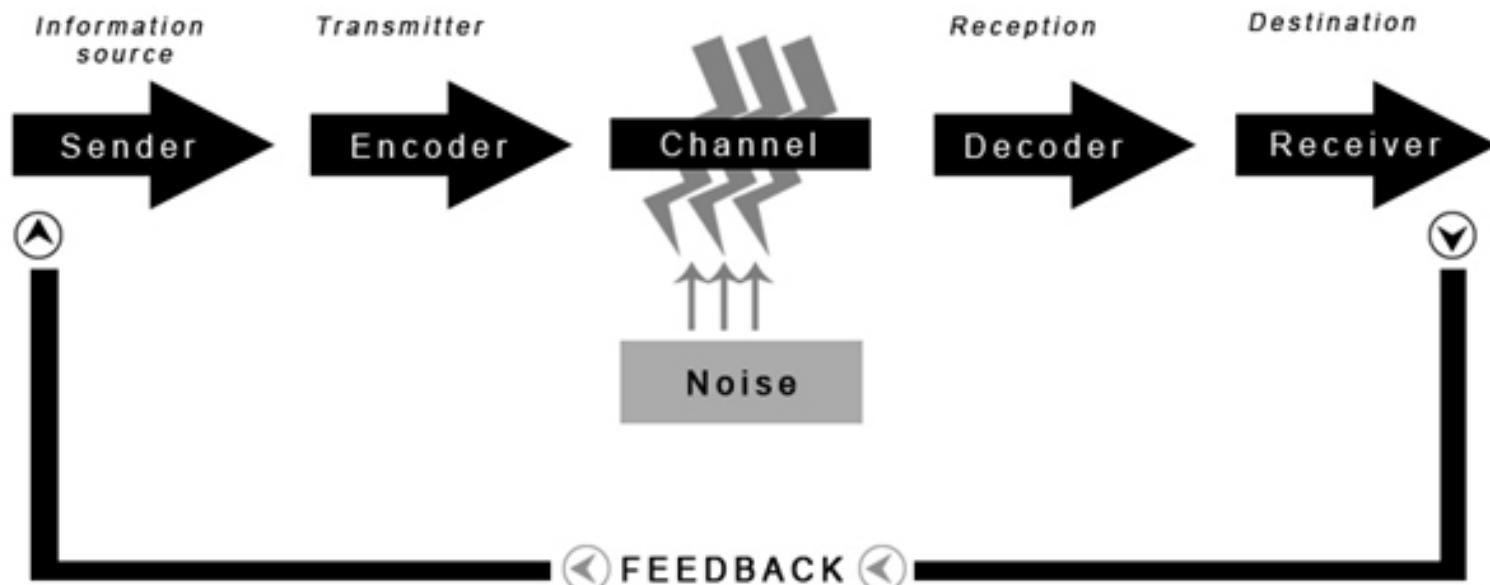
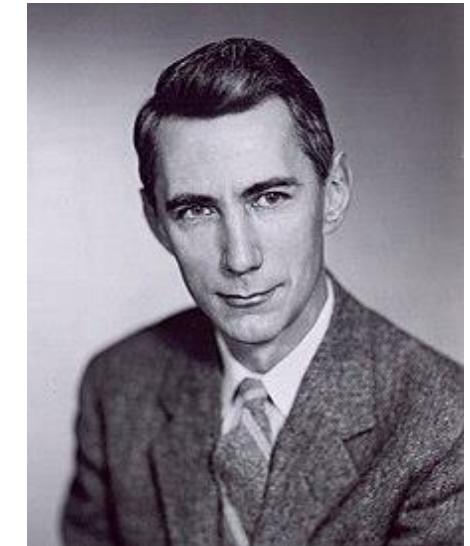


네트워크의 성능

- 네트워크의 주요 성능 평가 지표

- Channel Capacity

- ✓ 클라우드 엘우드 샤논 (Claude Elwood Shannon, 1916~2001) 이 정보 이론 (Information Theory)을 정립하면서 통신 네트워크의 용량을 평가할 수 있는 Channel Capacity 공식을 만듦.
 - ✓ 어떤 곳(source = sender)에서 다른 곳(destination = receiver)으로 정보 (information)를 보내기 위해 사용되는 물리적인 통로인 채널 (channel)에서 정보에 에러를 발생시키지 않고 보낼 수 있는 최대의 속도 (=채널이 수용할 수 있는 정보의 전송 능력)
 - ✓ 채널을 통해 보내지는 데이터의 양은 그 채널의 대역폭 (bandwidth)에 비례.



네트워크의 성능

- 네트워크의 주요 성능 평가 지표

- Channel Capacity

Capacity =

Maximum achievable
data rate (in bits/sec)

$$C = B \cdot \log_2 \left(1 + \frac{S}{N} \right)$$

Diagram illustrating the components of the Channel Capacity formula:

- Radio Channel Bandwidth (in Hz)**: Points to the variable B .
- Signal Power (in Watts)**: Points to the variable S .
- Noise Power (in Watts)**: Points to the variable N .
- SNR (Linear Scale, not in dB)**: A bracket groups S/N , indicating Signal-to-Noise Ratio.

Annotations:

- As this gets larger, C (Capacity) gets larger**: Points to the term $\log_2(1 + SNR)$.
- As this gets larger, C (Capacity) gets larger**: Points to the term $B \cdot \log_2(S/N)$.
- Ex > With Diversity, you can increase this value**: Points to the term $\log_2(1 + SNR)$.

<Channel Capacity 를 계산하는 예제>



1 MHz bandwidth channel 을 사용해서 통신할 때

이 channel 의 SNR 이 63 일 때

최대 도달 가능한 전송 속도는?

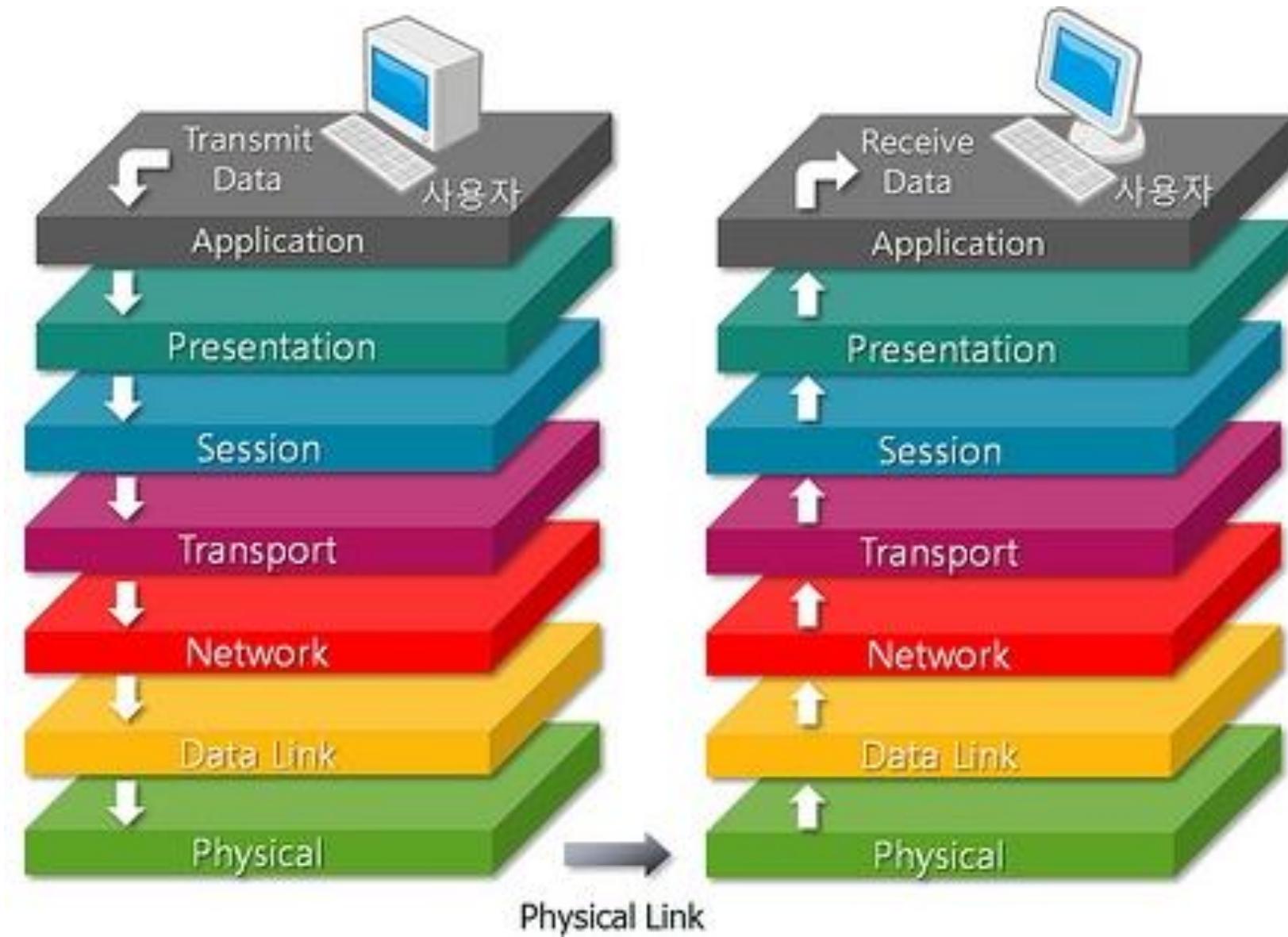
$$C = B \log_2(1+SNR)$$

$$= (10^6) \times \log_2(1+63)$$

$$= (10^6) \times \log_2(64) = 6 \text{ Mbps}$$

네트워크 구조와 통신

- OSI 7 계층 구조와 통신



네트워크 구조와 통신



• OSI 7계층 별 주요 역할

계층 번호	계층 이름	주요 역할	대표 프로토콜
7	응용 계층 (Application Layer)	사용자 인터페이스를 포함한 응용 프로그램	HTTP, FTP, TELNET, DNS, SMTP, NTP
6	표현 계층 (Presentation Layer)	데이터 형식상의 차이를 다루는 코드 간의 번역 제공 인코딩/디코딩, 압축, 암호화	JPG, MPEG, MIME, XDR
5	세션 (Session Layer)	양 끝단의 프로세스가 통신을 관리하는 방법 제공 사용자 동기화, 오류 복구, 세션 확립/유지/중단	NetBIOS, SSH
4	전송 계층 (Transport Layer)	양 끝단의 사용자에게 신뢰성 있는 통신 제공 오류 검출, 흐름 제어, 중복 검사	TCP, UDP, RTP, SCTP
3	네트워크 계층 (Network Layer)	여러 개의 노드를 경유하여 통신할 때 경로를 찾아 주는 라우팅(routing) 서비스 품질(QoS: Quality of Service) 제공 오류 제어, 흐름 제어, 노드의 논리 주소(IP address) 처리	IP, ICMP, IPSec, ARP, RIP, BGP
2	데이터링크 계층 (Data-Link Layer)	두 노드 간(point to point)의 신뢰성 있는 전송 보장. CRC기반의 오류 제어와 흐름 제어, 노드의 물리 주소(MAC: Machine Access Code) 처리	Ethernet, PPP
1	물리 계층 (Physical Layer)	케이블을 통해 데이터를 실제 전송	RS-232, RS-449

네트워크 구조와 통신



- OSI 모델 (7계층 모델)과 TCP/IP 모델 (4계층 모델)

제7계층

Application Layer 응용 계층

제6계층

Presentation Layer 표현 계층

제5계층

Session Layer 세션 계층

제4계층

Transport Layer 전송 계층

제3계층

Network Layer 네트워크 계층

제2계층

Data Link Layer 데이터링크 계층

제1계층

Physical Layer 물리 계층

OSI 모델

Application Layer 응용 계층

Transport Layer 전송 계층

Internet Layer 인터넷 계층

Network Interface Layer
네트워크 인터페이스 계층

제4계층

제3계층

제2계층

제1계층

TCP/IP 모델

네트워크 구조와 통신

• TCP/IP 계층별 기능

TCP/IP 계층

역할

Application Layer
응용 계층

데이터의 내용을 보고 애플리케이션의 목적에
따른 서비스를 사용자에게 제공

서비스 제공



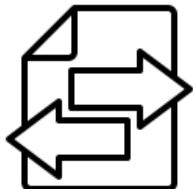
Transport Layer
전송 계층

데이터의 흐름을 제어하여 데이터 전송의 신뢰성과
효율성을 관리하고, 목적지 컴퓨터에 전송된 데이터
를 적절한 애플리케이션에 배분

Internet Layer
인터넷 계층

서로 다른 네트워크에 있는 컴퓨터 간에 데이터
전송

데이터 전송



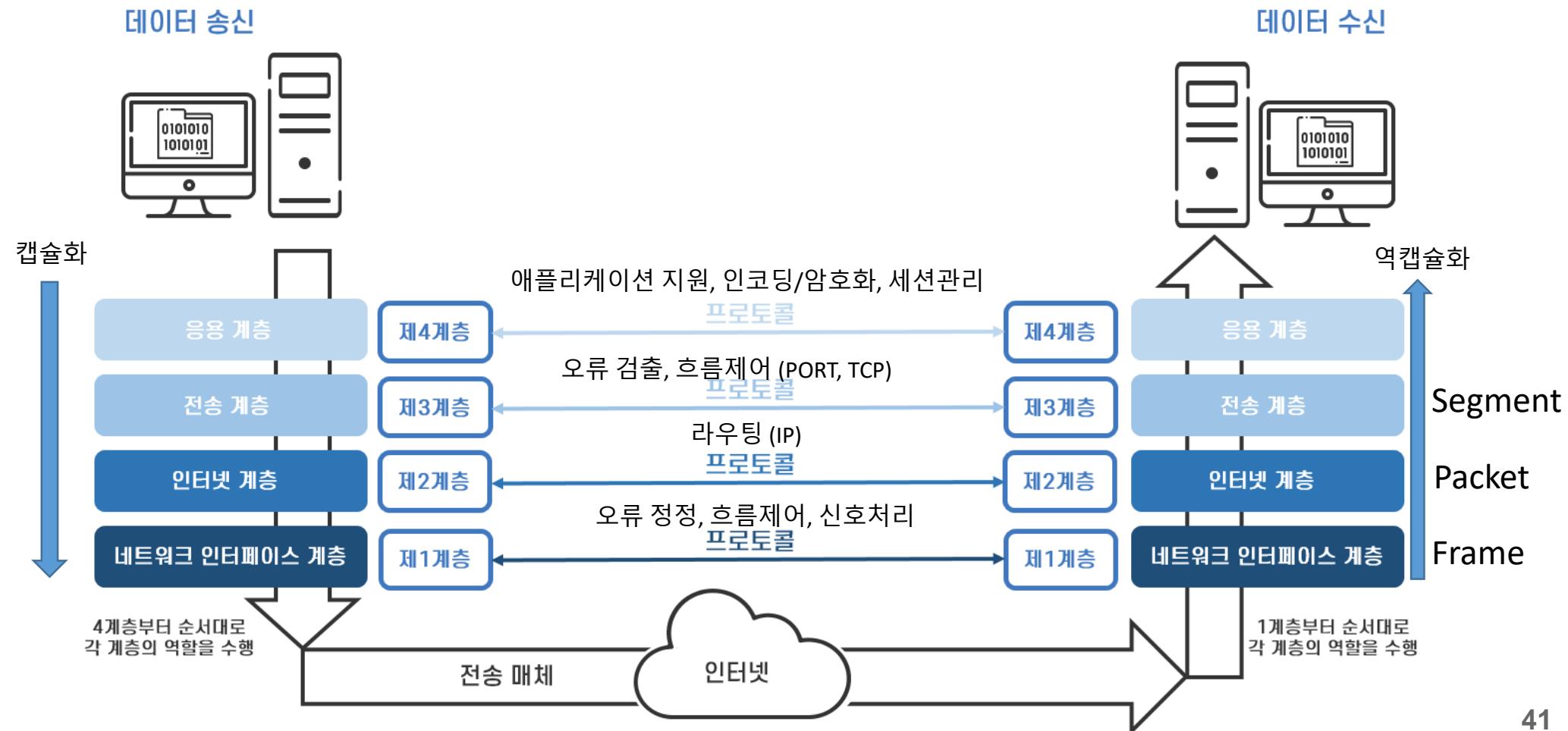
Network Interface Layer
네트워크 인터페이스 계층

같은 네트워크 안에서 물리적으로 인접하여
연결되어 있는 네트워크 장비 간의 데이터 전송

네트워크 구조와 통신

- OSI 7계층의 주요 기능과 데이터 단위

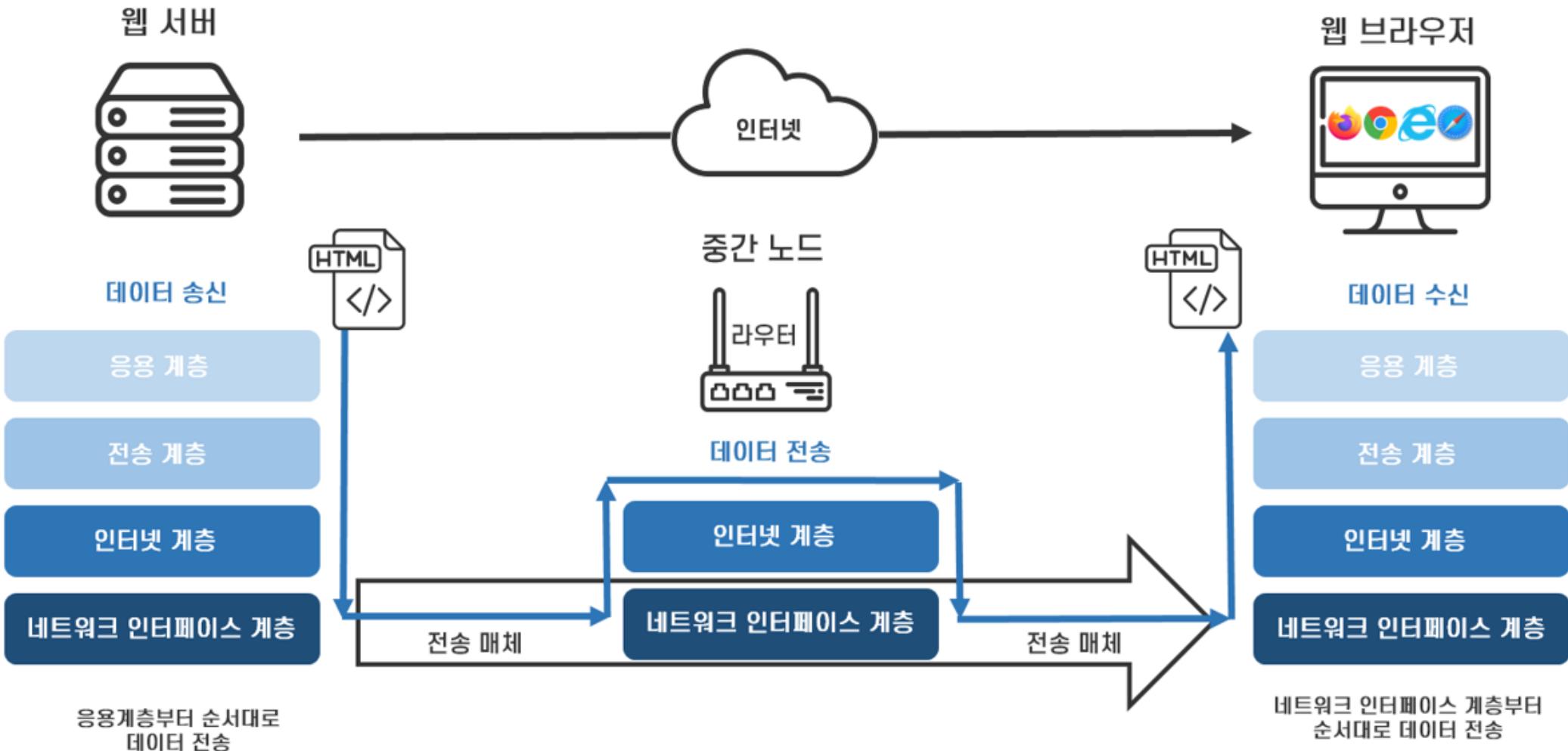
- 전송 매체에서는 스위치와 라우터 (1,2계층 네트워크 장비)를 통해 데이터 전달



네트워크 구조와 통신

- 웹 서버가 웹 브라우저로 웹 서비스 제공하는 과정

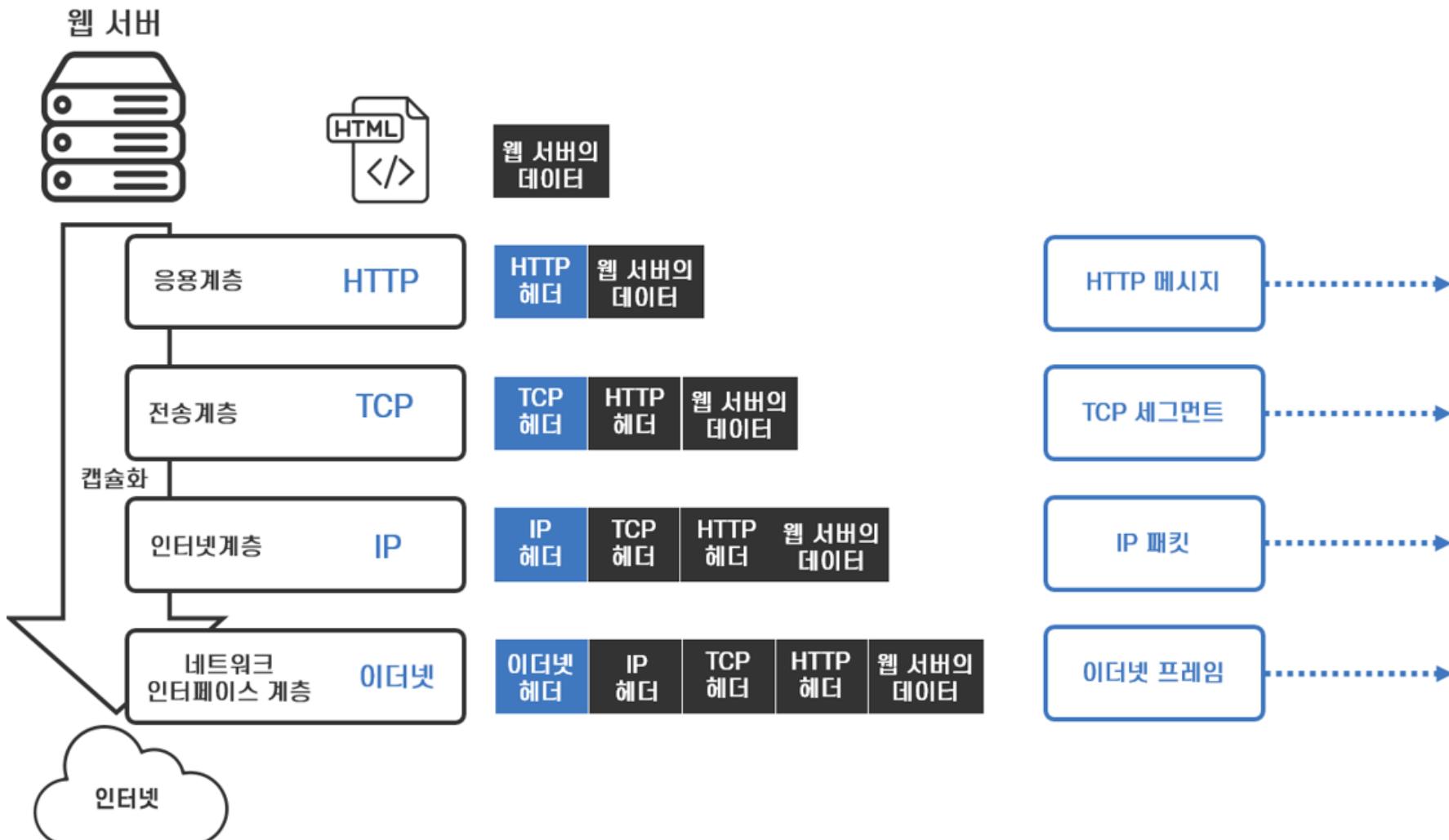
- 응용계층 (HTTP 프로토콜) - 전송계층 (TCP프로토콜) - 인터넷 계층 (IP 프로토콜) - 네트워크 인터페이스 계층 (이더넷 프로토콜)



네트워크 구조와 통신

- 웹서버에서 웹 서비스를 생성하는 과정

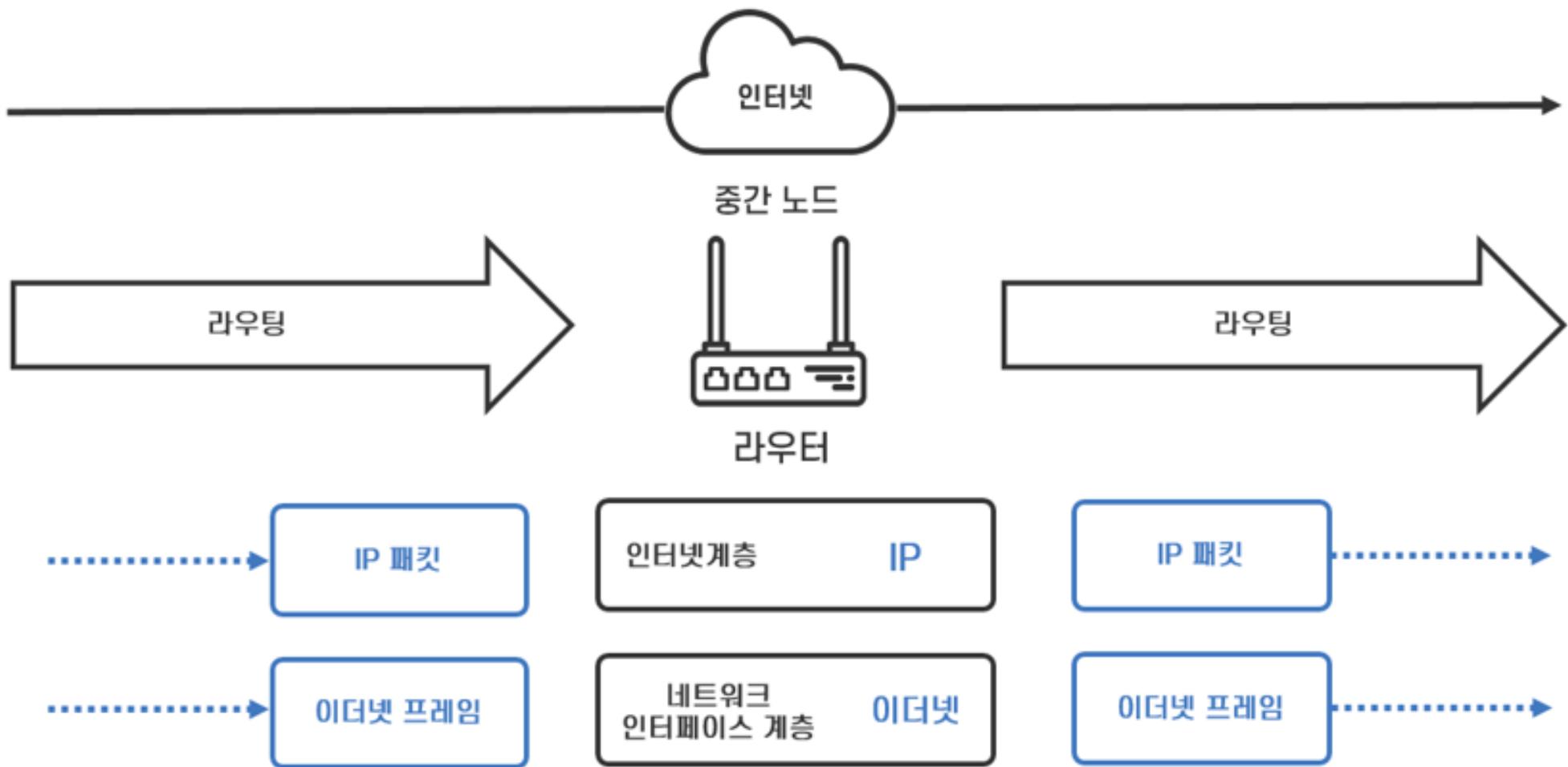
- HTTP 헤더 추가 → TCP 헤더 추가 → IP 헤더 추가 → 이더넷 헤더 추가 → 전기 신호로 변환하여 인터넷 전송



네트워크 구조와 통신

- 인터넷의 여러 중간 노드 (라우터) 동작

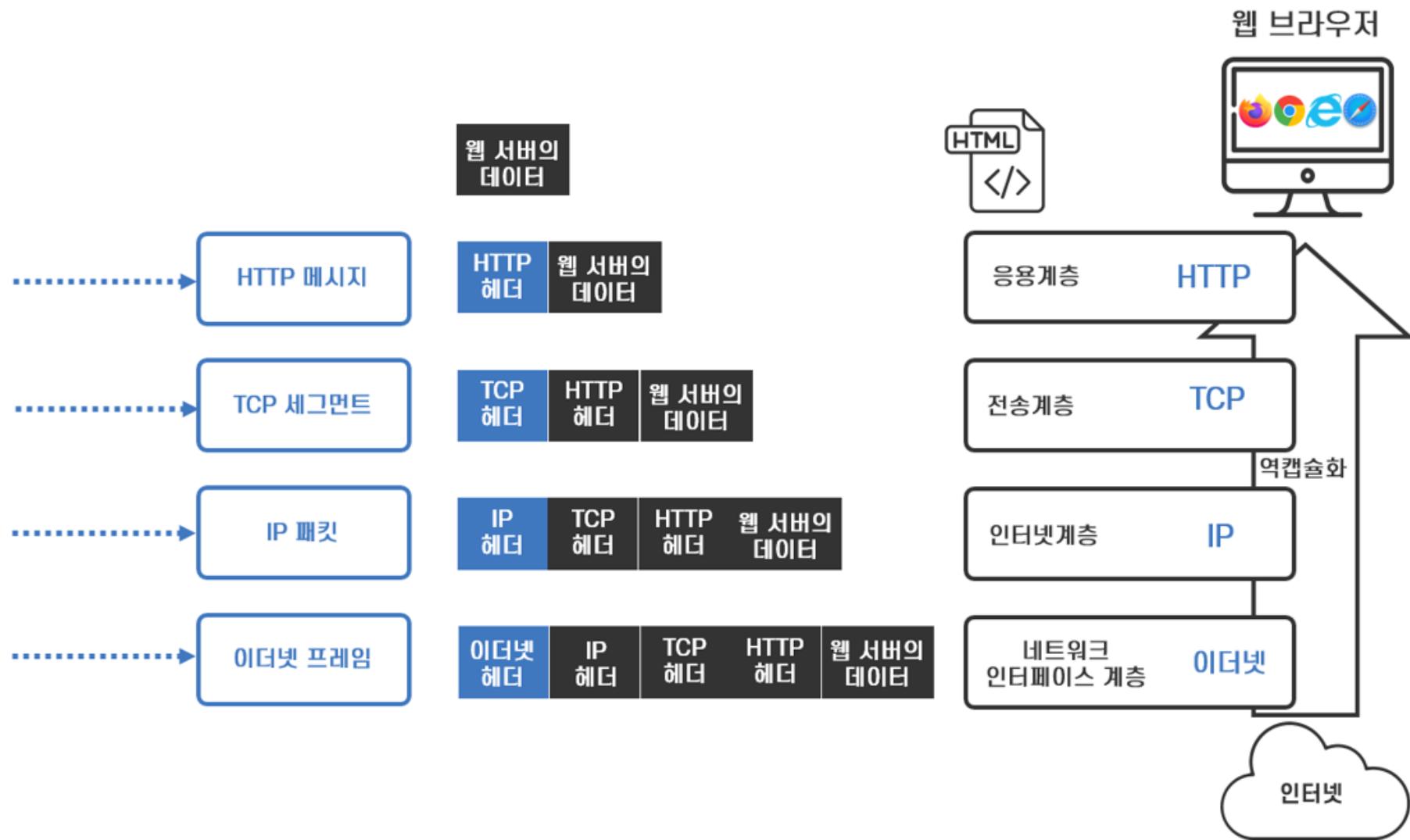
- 라우터는 라우팅 기능을 이용하여 IP 패킷의 IP 주소를 확인하고 경로 정하여 IP 패킷을 다른 라우터에게 전달



네트워크 구조와 통신

- 웹 브라우저의 데이터 수신하는 과정

- (각 계층에서 네트워크 기능 수행) 이더넷 헤더 제거 → IP 헤더 제거 → TCP 헤더 제거 → HTTP 헤더 제거 → 웹 브라우저에서 웹 페이지 출력

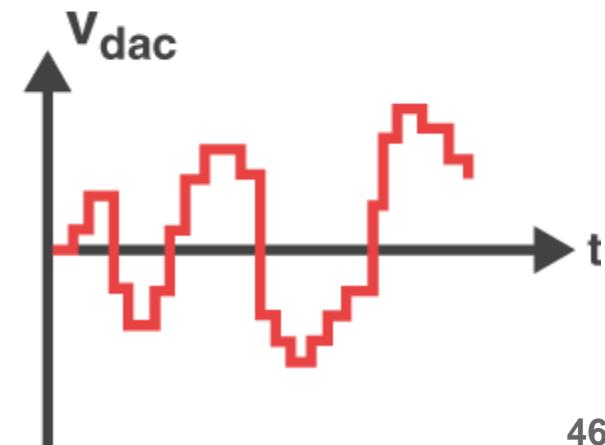
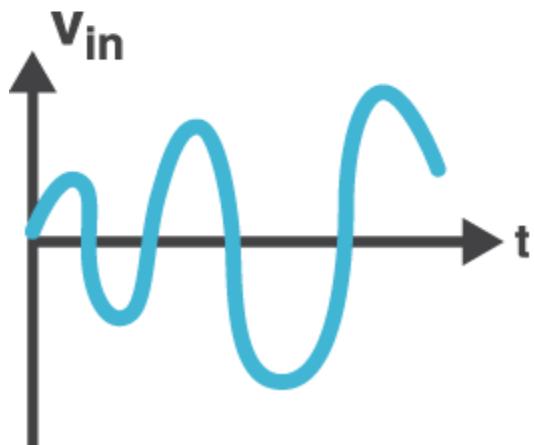
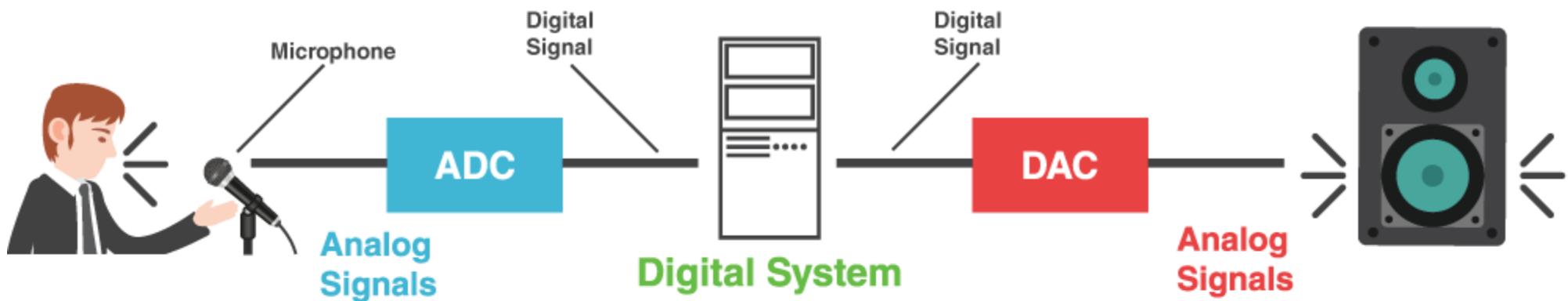


네트워크 인터페이스 계층



- 네트워크 인터페이스 계층의 필요성

- 필요성 1: 아나로그 세상과 디지털 세상의 연결 (=아나로그 데이터와 디지털 데이터 변환)

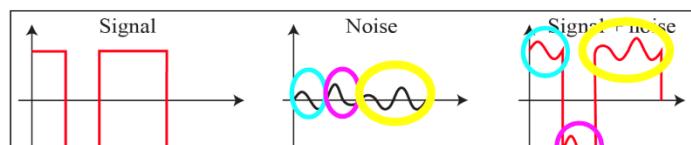
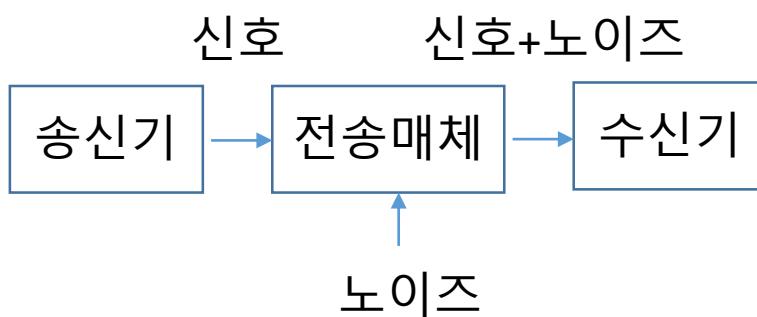


네트워크 인터페이스 계층

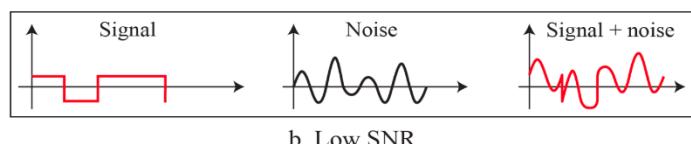


- #### • 네트워크 인터페이스 계층의 필요성

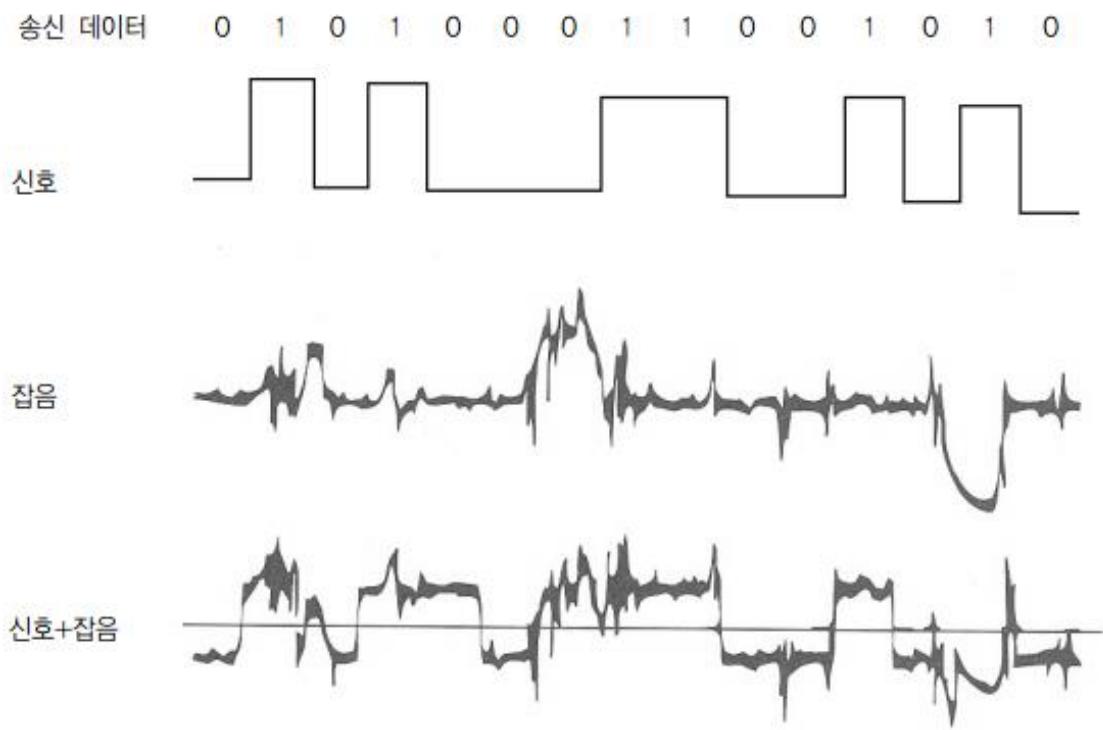
- 필요성 2: 전송 매체 특성, 네트워크 혼잡, 전자기 간섭, 노이즈 등에 의해 발생하는 전송 오류 대응



a. High SNR



b. Low SNR



샘플링 시간

수신 데이터 0 1 0 1 0 0 1 1 1 0 0 1 0 0 0

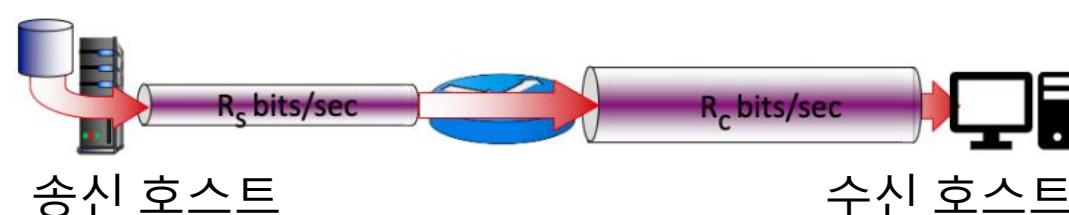
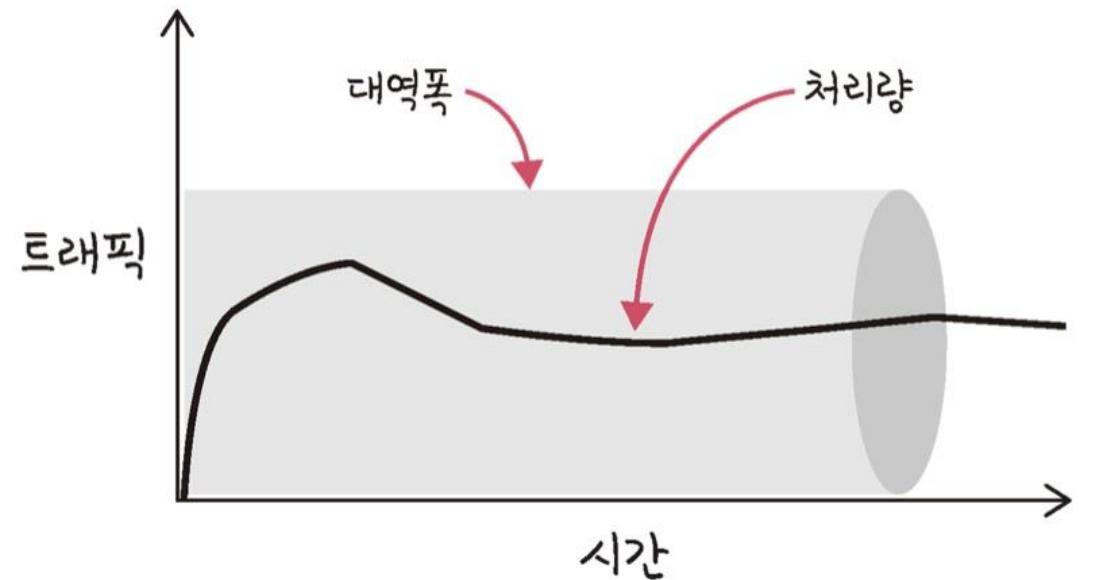
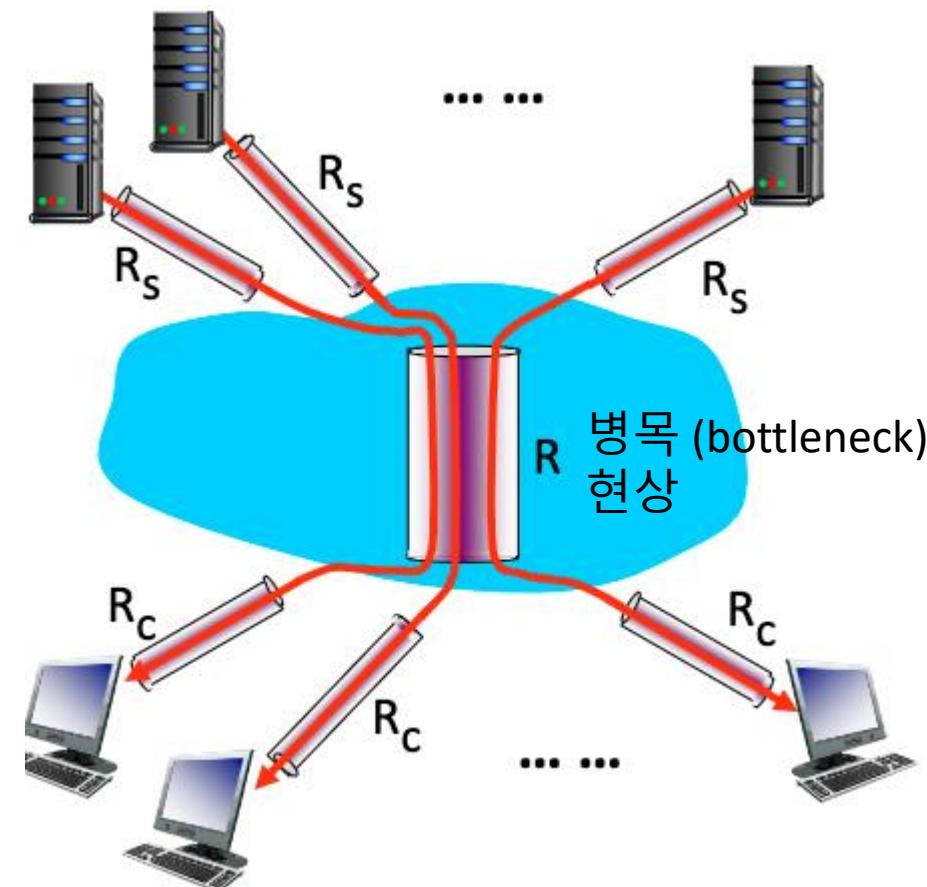
원 데이터 0 1 0 1 0 0 0 1 1 0 0 1 0 1 0

비트오류 발생

네트워크 인터페이스 계층



- 네트워크 인터페이스 계층의 필요성
 - 필요성 3: 송수신 호스트 간의 데이터 처리 능력 차이 극복



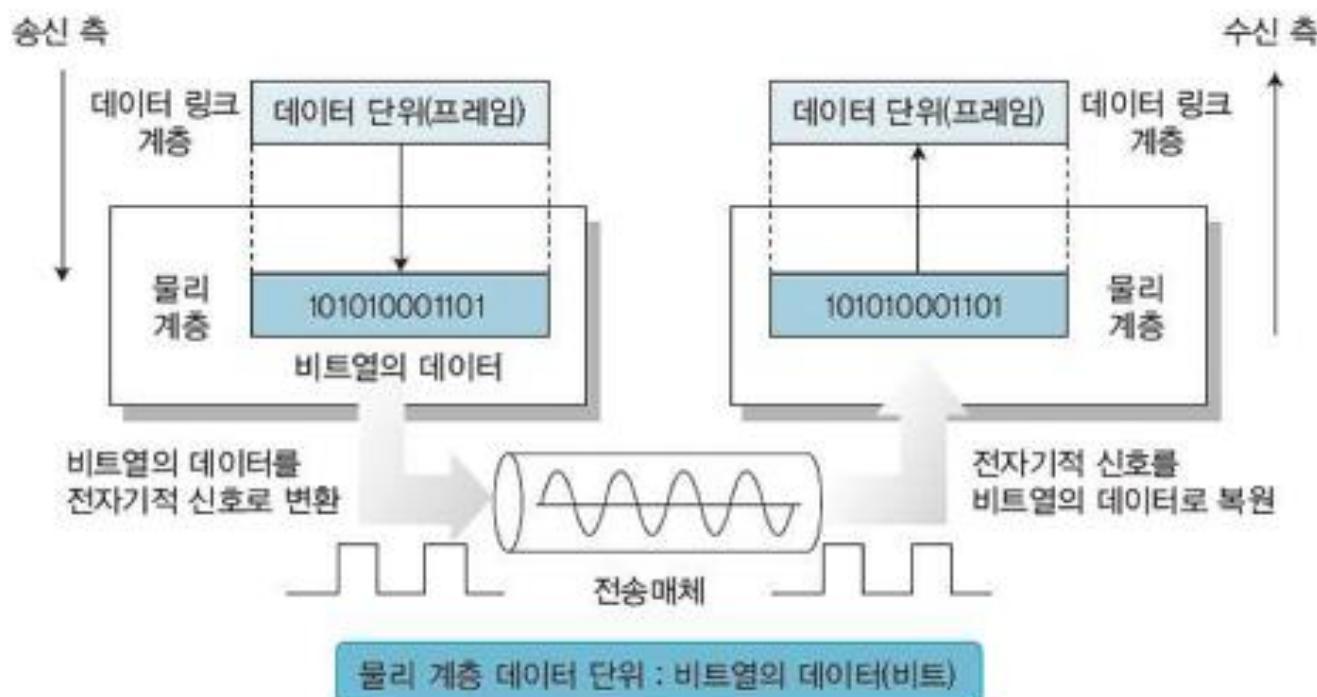
- $R < R_s, R_c$ 인 경우 R 이 bottleneck 이 되고 네트워크 스루풋은 R bps 가 됨.
- $R_s > R_c$ 인 경우에, 송신 호스트는 R_c 로 맞춰서 전송할 수 있어야 함.

네트워크 인터페이스 계층

- 네트워크 인터페이스 계층의 필요성

- 전송 매체를 통해 네트워크 장치 간에 데이터를 전송하고 수신하기 위한 신호 처리와 통신 제어를 하는 역할 (= 물리적 신호를 전송/수신하기 위한 규칙)

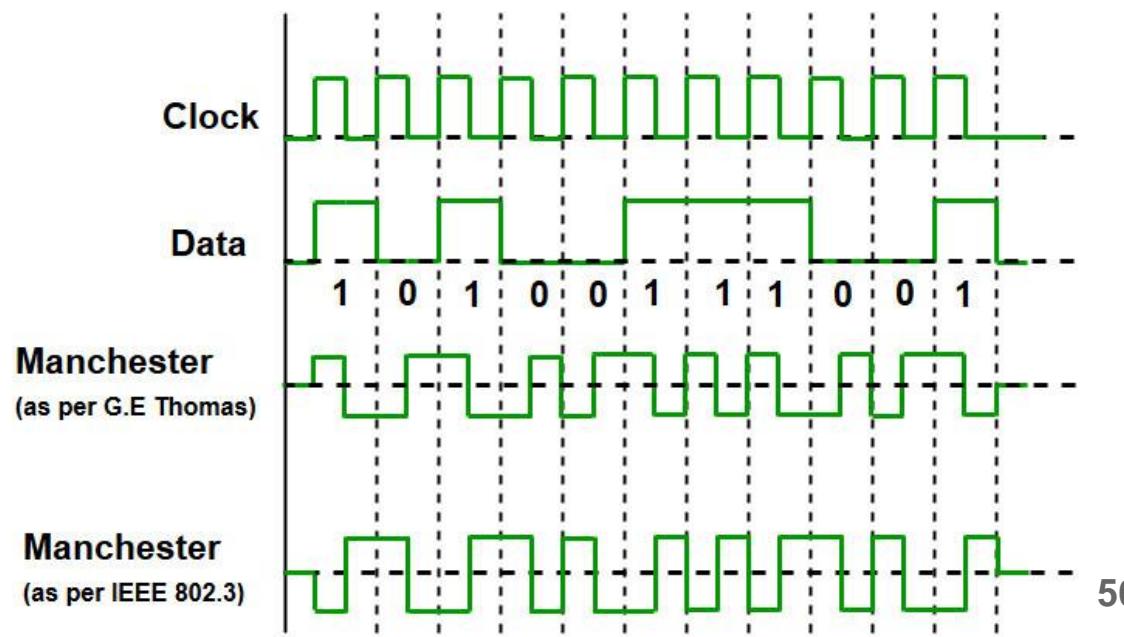
- ✓ 데이터링크 계층: 프레임 단위로 데이터를 송수신하며, 신뢰할 수 있는 통신이 가능하도록 매체 접근 제어할 때, 오류 정정, 순서 제어, 흐름 제어함
- ✓ 물리계층: 비트 단위로 데이터를 송수신하며, 전자기적 신호와 비트열 데이터간 신호 변환/처리함
- ✓ 네트워크 인터페이스 계층에 속한 네트워크 장비: 랜카드, 스위치, 무선 AP



네트워크 인터페이스 계층

- 랜카드 (Network Interface Card, NIC)

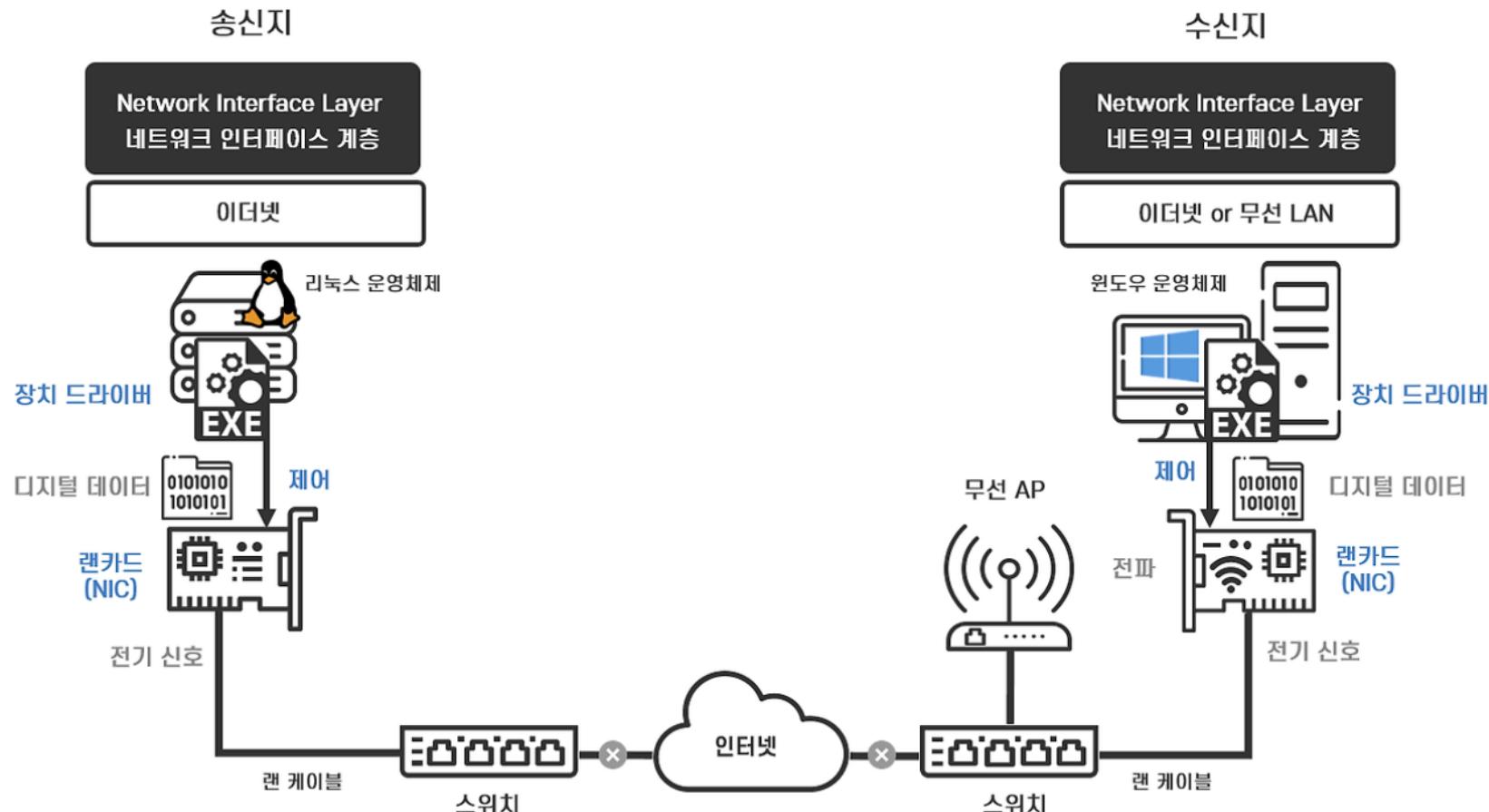
- 컴퓨터 장치를 네트워크에 연결하려면 반드시 하나 이상의 랜카드 (유선 랜카드 or 무선 랜카드)가 있어야 함.
- 컴퓨터가 만든 디지털 데이터를 전기 신호나 전파 같은 물리적 신호로 변환하여 네트워크로 보내거나 (송신 = TX = Transmission), 네트워크로부터 받은 물리적 신호를 디지털 데이터로 변환하여 받는(수신 = RX = Reception) 기능.
- 즉, 랜카드는 컴퓨터 (디지털 장치)와 네트워크 (아나로그 장치) 간에 전송매체를 통해 상호작용이 가능하도록 하는 네트워크 인터페이스임.
- 랜카드는 하드웨어로 구현됨.



네트워크 인터페이스 계층

- 랜카드 (Network Interface Card, NIC)

- 네트워크 하드웨어 장치인 랜카드를 제어하는 소프트웨어가 필요함.
- 랜카드 장치 드라이버 = 랜카드를 제어하는 소프트웨어 (=펌웨어).
- 운영체제 (OS)에 랜카드를 제어하기 위한 장치 드라이버가 설치되어야 컴퓨터가 네트워크 하드웨어 장치를 구동시킬 수 있음 (Plug & Play로 자동 설치).



네트워크 인터페이스 계층

• 네트워크 인터페이스 계층 프로토콜

- 유선랜 표준 : 이더넷 프로토콜 (IEEE 802.3)
- 무선랜 표준 : 와이파이 프로토콜 (IEEE 802.11)
- IEEE (Institute of Electrical and Electronics Engineering, 미국 전기전자학회, 아이 트리플 이): 전기전자공학 및 통신네트워크에 관한 기술 표준화 비영리 단체

표준 작업 그룹	내용
IEEE 802.1	상위 계층 인터페이스 및 MAC 브리지
IEEE 802.2	LLC (Logical Link Control)
IEEE 802.3	CSMA/CD (Carrier Sense Multiple Access / Collision Detect)
IEEE 802.4	토큰버스 (Token Bus)
IEEE 802.5	토큰링 (Token Ring)
IEEE 802.6	MAN (Metropolitan Area Networks)
IEEE 802.7	광대역 LAN
IEEE 802.8	광섬유 LAN
IEEE 802.9	종합 데이터 음성 네트워크 (Integrated Data and Voice Networks)
IEEE 802.10	보안 (Security)
IEEE 802.11	무선 네트워크 (Wireless Networks)

네트워크 인터페이스 계층

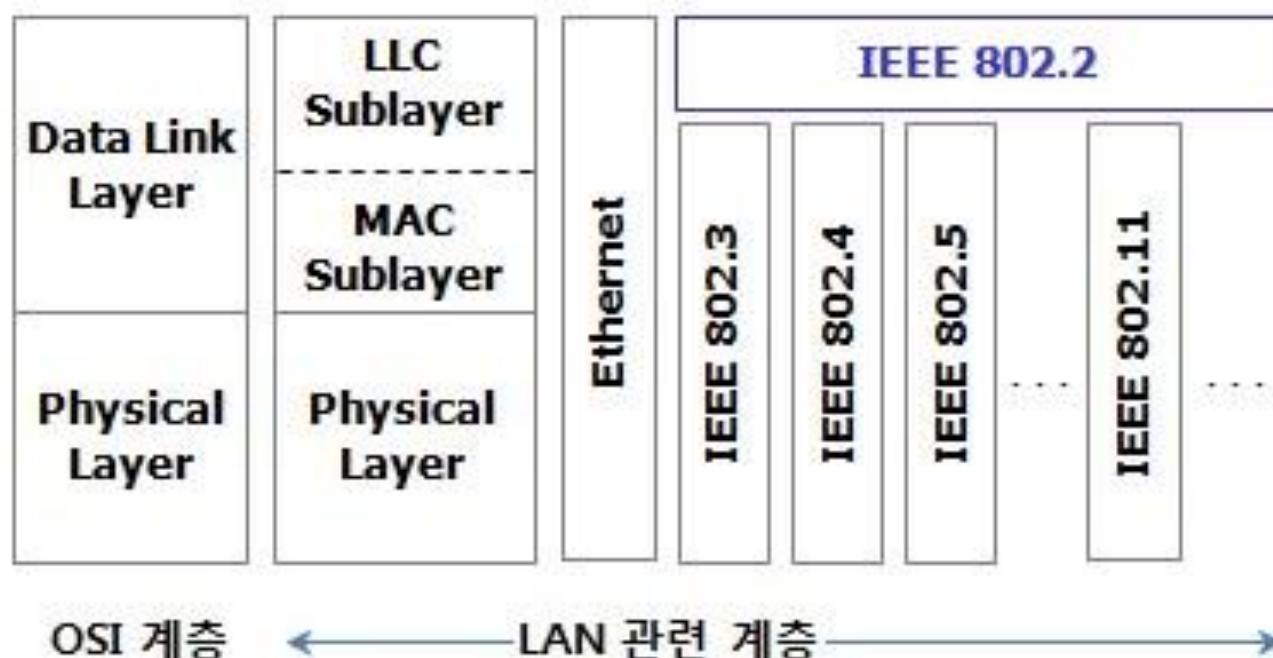
- 네트워크 인터페이스 계층 프로토콜

- LLC (Logical Link Control) 부계층

- ✓ 매체접속제어 방식 간의 차이를 보완하는 역할 (소프트웨어, 프로토콜 스택)
 - ✓ 네트워크 토플로지와 동작 방식과 관계 없이 통신 가능하도록 함

- MAC (Medium Access Control) 부계층

- ✓ 통신 방식에 의존적인 매체접속제어 기능 구현 (하드웨어)
 - ✓ 고속 신호 처리하는 물리 계층 데이터를 하드웨어 기술로 버퍼링하고 데이터 가공하여 상위 계층으로 전달



네트워크 인터페이스 계층

• 네트워크 인터페이스 계층 프로토콜

- LAN 표준: IEEE에서 네트워크 인터페이스 계층과 관련된 다양한 LAN 표준을 IEEE 시리즈로 발표

- ✓ IEEE 802.2는 데이터 링크 계층의 상위 부분인 LLC 프로토콜을 정의함
- ✓ IEEE 802.3, 802.4, 802.5는 물리 계층과 MAC 계층에 관한 기능을 정의함

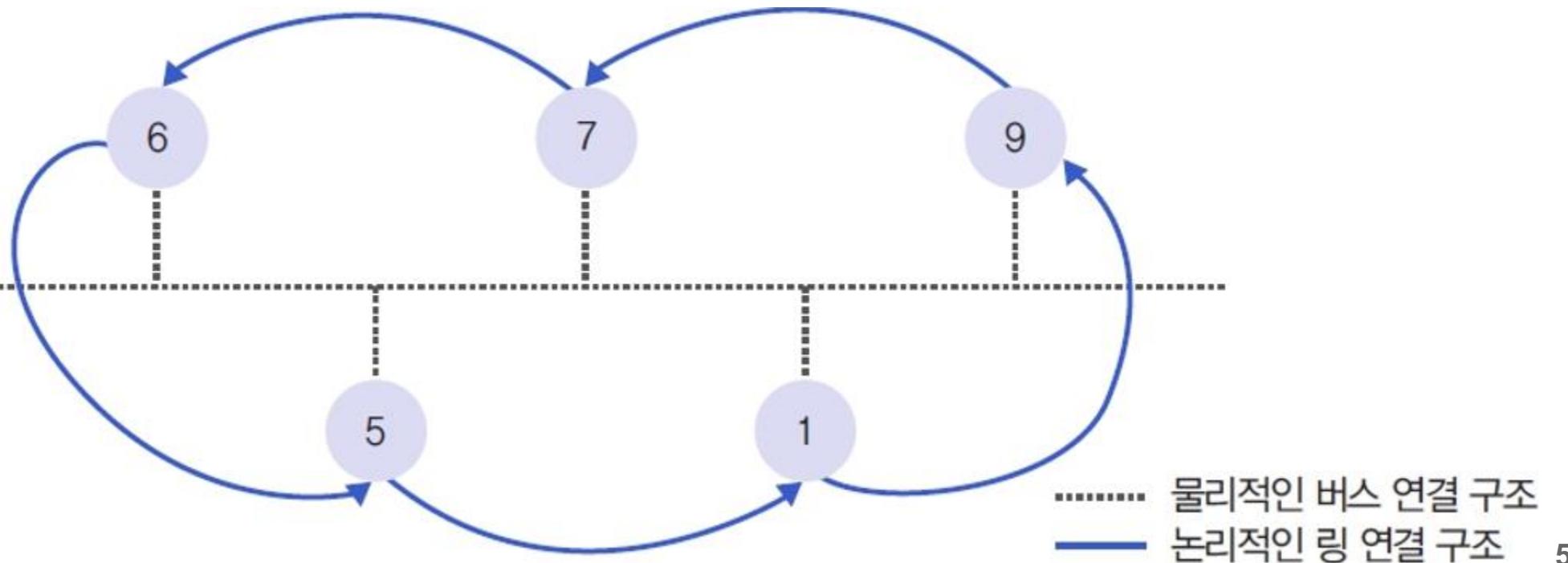
OSI 계층	IEEE 802 LAN 계층	IEEE 802 표준들			
데이터 링크 계층	논리 링크 제어 서브 계층 (LLC sublayer)	IEEE 802.2 논리 링크 제어 (LLC)			
	매체 접근 제어 서브 계층 (MAC sublayer)	IEEE 802.3 CSMA/CD	IEEE 802.4 토큰 버스	IEEE 802.5 토큰 링	FDDI 토큰 링
물리 계층	물리 계층	기저 대역 동축 케이블 10Mbps 광섬유 20Mbps	광대역 동축 케이블 10Mbps 광섬유 20Mbps	차폐 연선 16Mbps	광섬유 100Mbps

네트워크 인터페이스 계층

- 네트워크 인터페이스 계층의 토폴로지와 연결 구조

- 토큰 버스

- ✓ 물리적으로 보면 공유 버스 구조(점선)로 연결되지만, 논리적인 프레임 전달은 링 구조(실선)로 동작
 - ✓ 프레임을 전송하기 전에 버스 신호를 감지하는 CSMA/CD 방식과 다른 방식의 전송 메커니즘 사용
 - ✓ 데이터 프레임 전송이 호스트 사이에 순차적으로 이루어지도록 토큰 기반의 제어 프레임 사용

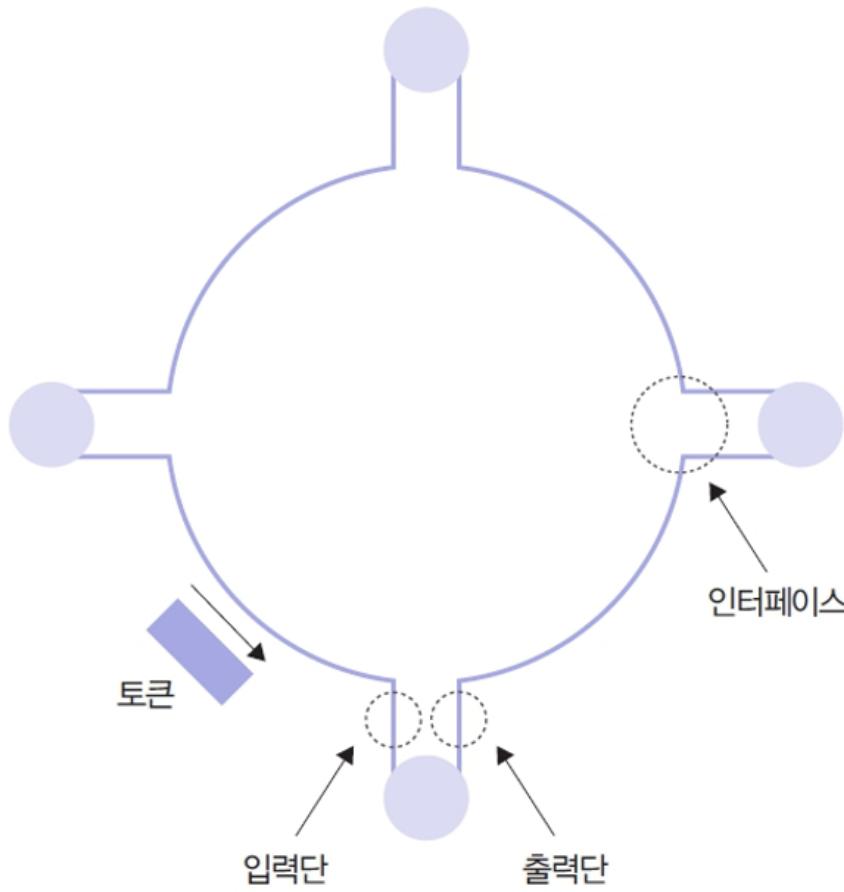


네트워크 인터페이스 계층

- 네트워크 인터페이스 계층의 토폴로지와 연결 구조

- 토큰 링

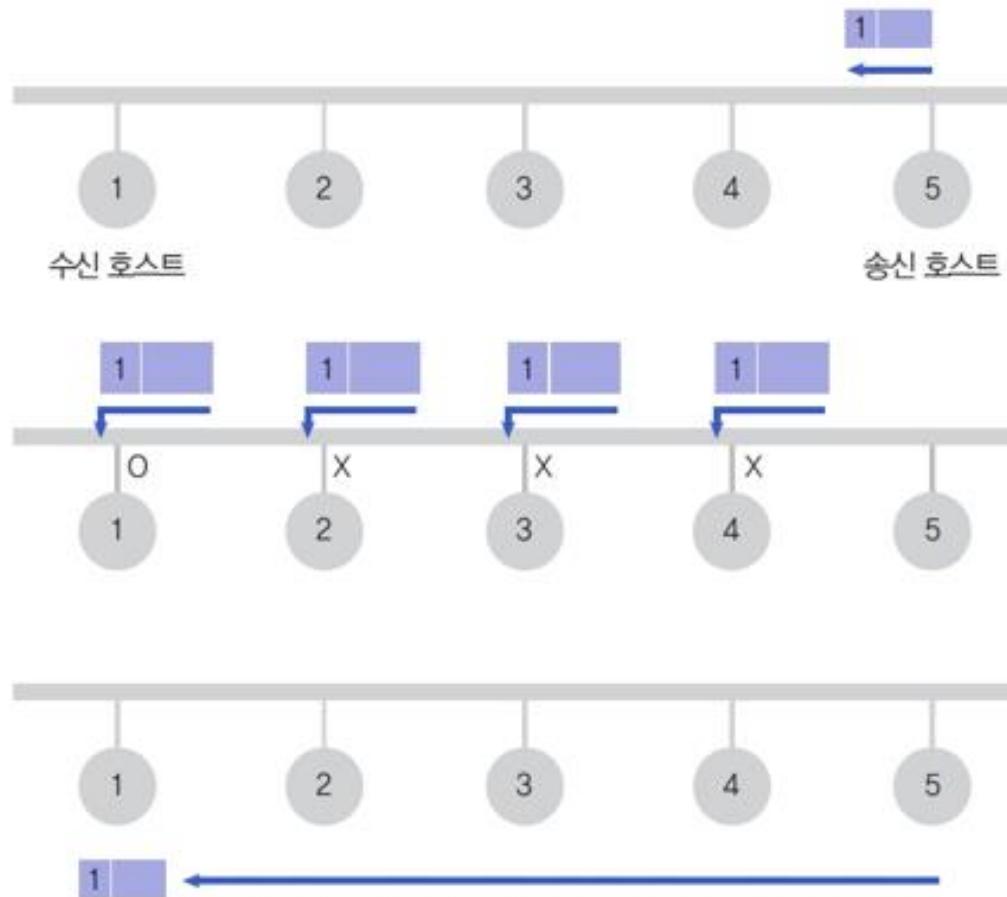
- ✓ 순환 구조의 전송 매체와 점대점 (P2P) 방식으로 연결되어 대기 모드와 전송 모드로 구분되어 동작
 - ✓ 대기 모드: 입력단으로 들어온 신호를 출력단으로 즉시 내보냄
 - ✓ 전송 모드: 호스트가 토큰을 획득하여 데이터 프레임을 전송할 수 있는 권한을 보유한 상태



네트워크 인터페이스 계층

- 네트워크 인터페이스 계층의 매체 접근 제어

- 하나의 전송 매체 (공유 버스)에 호스트 5개가 연결된 LAN 환경 가정
- 5번 호스트가 1번 호스트에게 데이터 전송할 때, 2,3,4번 호스트는 프레임의 헤더의 수신지 호스트 MAC 주소 (=1번)와 자신의 MAC 주소가 일치하는지 확인하고, 불일치하므로 무시. 1번 수신 호스트만 수신.



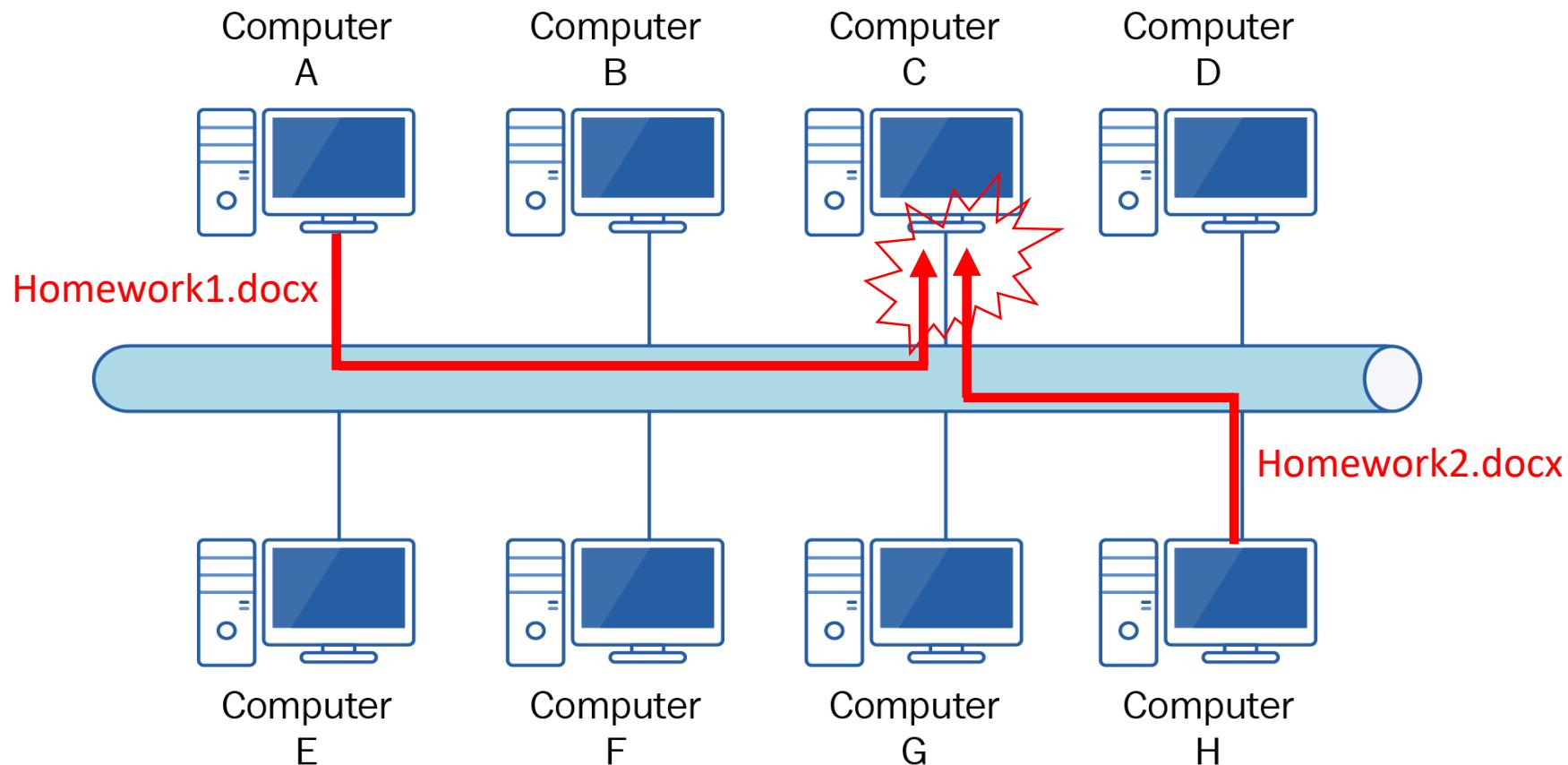
네트워크 인터페이스 계층

- 네트워크 인터페이스 계층의 매체 접근 제어

- 이더넷에서 여러 호스트가 동시에 프레임 전송시 충돌 발생 문제

- ✓ 이더넷: 다수의 컴퓨터, 허브, 스위치 등을 하나의 인터넷 케이블에 연결한 네트워크 구조

- ✓ 컴퓨터 A가 컴퓨터 C에게 homework1.docx 파일 전송할 때, 컴퓨터 H가 동시에 homework2.docx 파일을 전송하면 충돌 발생



네트워크 인터페이스 계층

- 네트워크에서 충돌 문제를 해결하는 방법?

- 방법1: 충돌 허용한 후 문제 해결하는 방법

- ✓ 네트워크 안의 다수의 호스트가 전송 매체를 공유할 때, 충돌을 허용하고, 효과적으로 회피하는 방법

- ✓ 충돌 회피 방법? 각 송신 호스트에서 서로 다른 전송 시간대를 지정하는 방법

- ✓ 대표적인 예: IEEE 802.3 표준에서 사용하는 CSMA/CD (Carrier Sense Multiple Access / Collision Detection)

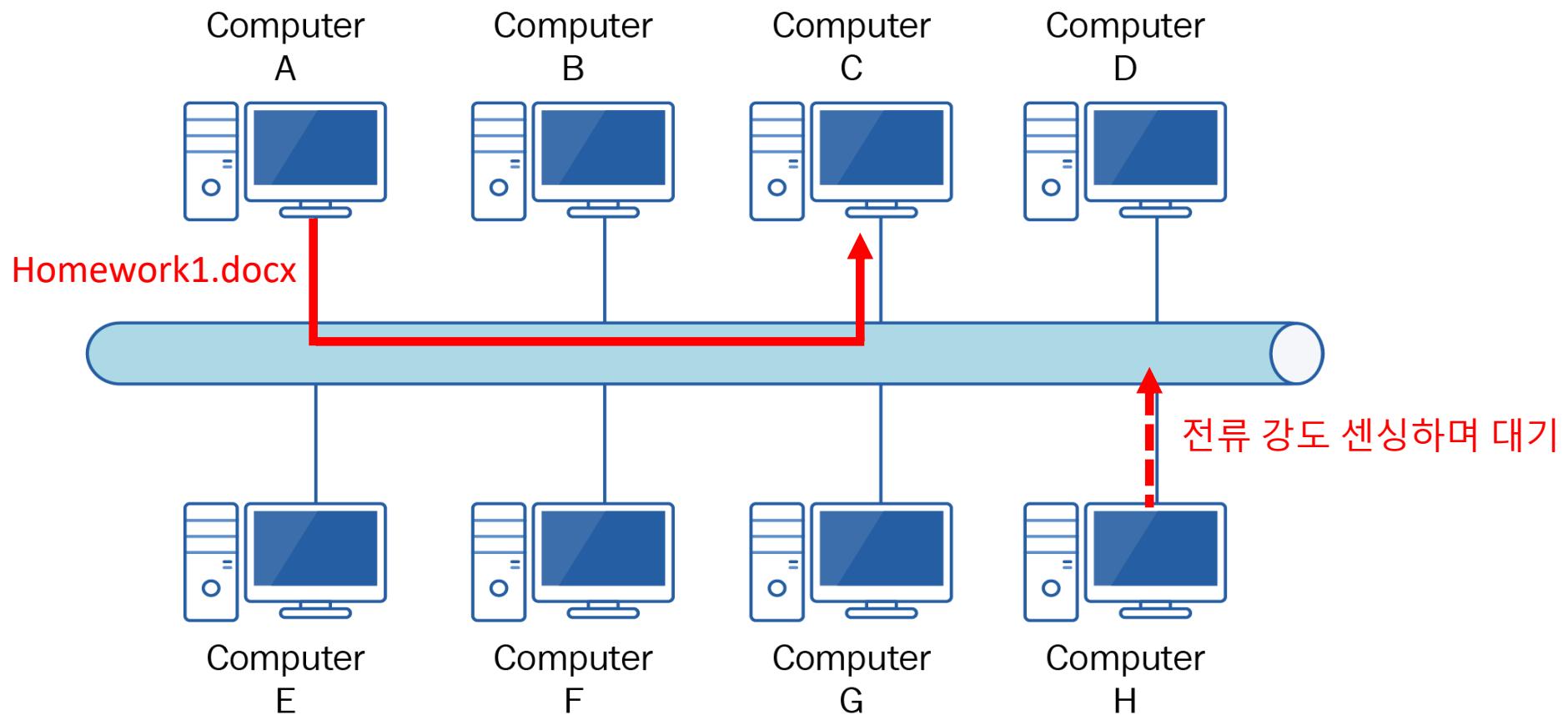
- 방법2: 충돌 발생을 원천적으로 차단하는 방법

- ✓ 토큰을 가진 호스트만 송신할 수 있도록 하는 방법

- ✓ 대표적인 예: IEEE 802.5 표준에서 사용하는 토큰 링

네트워크 인터페이스 계층

- 네트워크 인터페이스 계층의 매체 접근 제어
 - CSMA/CD (Carrier Sense Multiple Access / Collision Detection) = 눈치 게임
 - ✓ 충돌 현상을 방지하기 위해 전류 강도를 확인해 전송 매체가 사용 중인지 확인
 - ✓ 송신 전에 확인 (Carrier Sense)하여 충돌을 탐지 (Collision Detection)하고, 전류 강도가 일정 수준 미만이면 전송 매체를 사용하는 호스트가 없다고 판단하여 데이터를 전송 (Multiple Access)

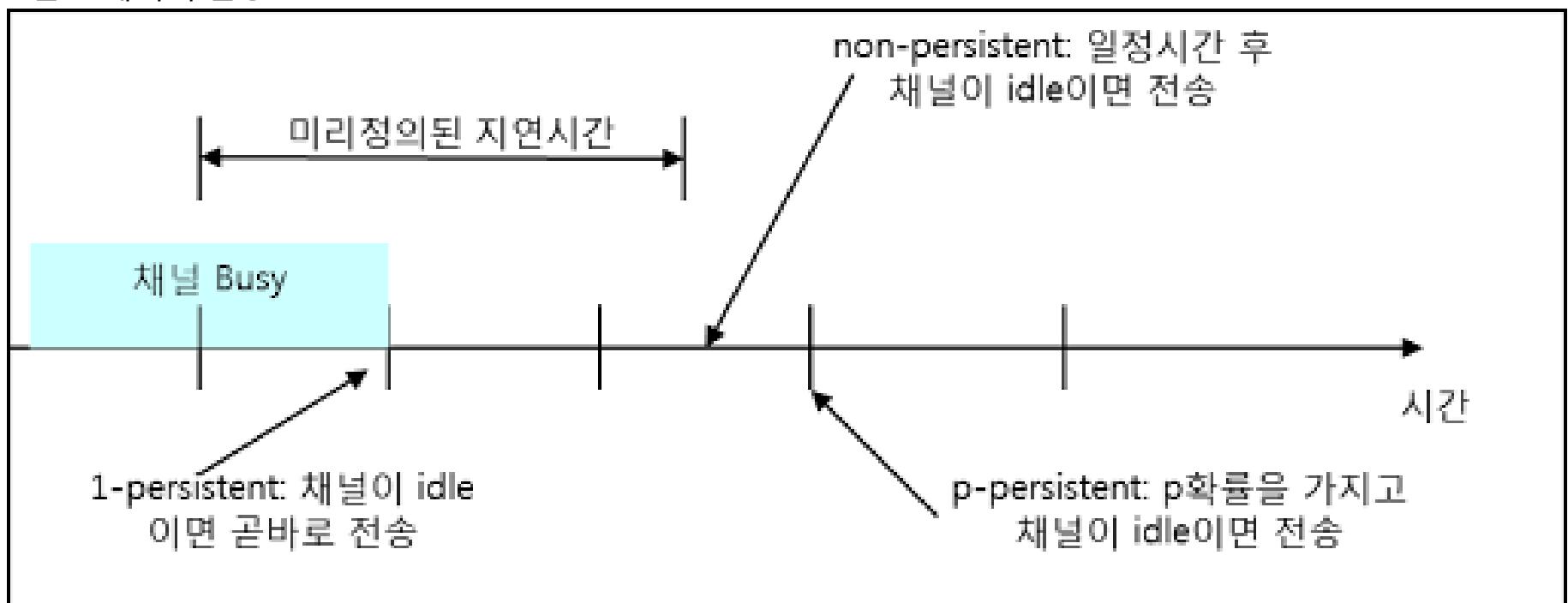


네트워크 인터페이스 계층

- 네트워크 인터페이스 계층의 매체 접근 제어

- 눈치 게임 (CSMA/CD) 방법

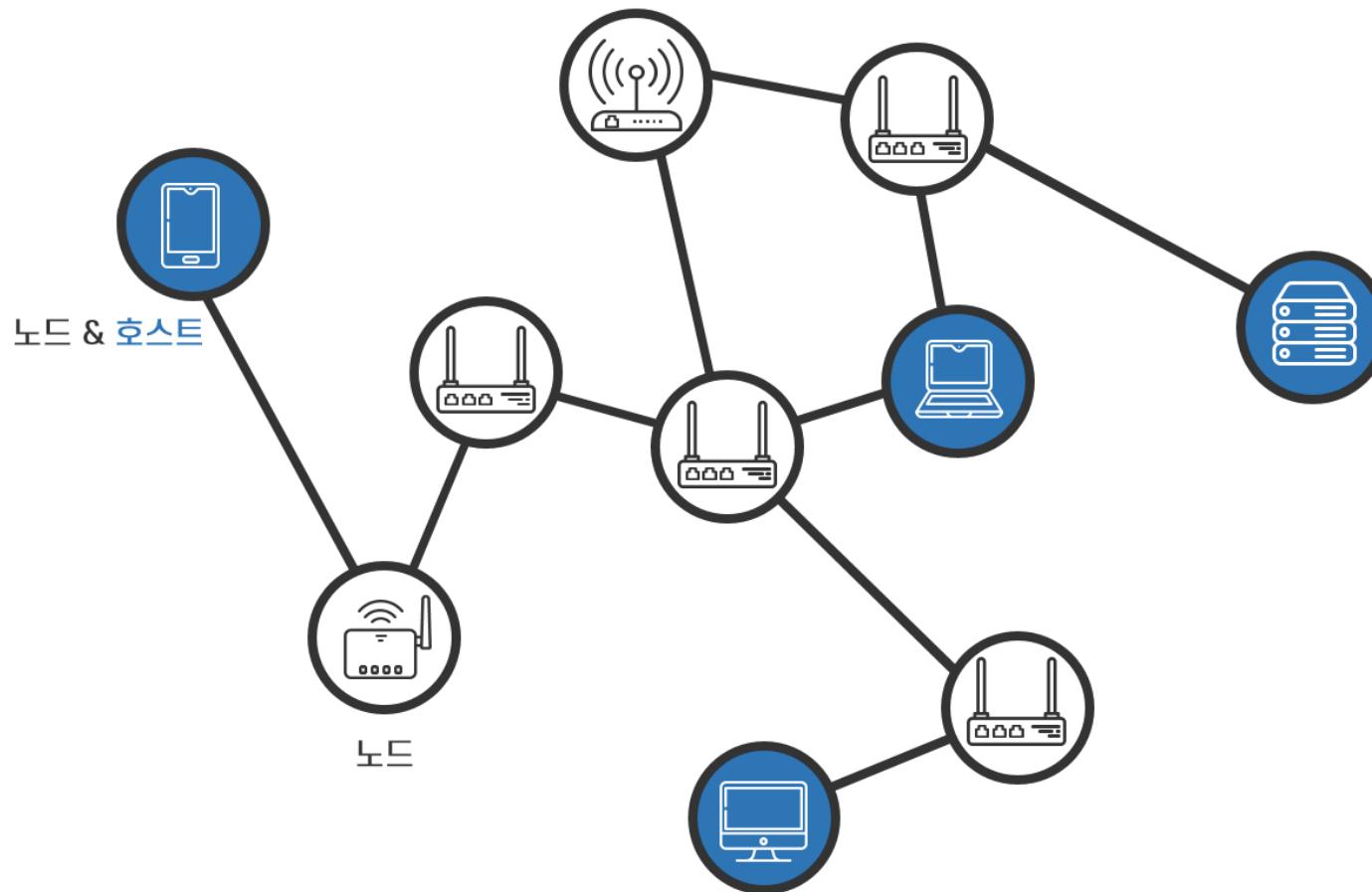
- ✓ 1-persistent: 전송 매체가 사용 중인지, 전송 매체 상태를 계속 확인하다가 아무도 사용하지 않은 상태면 바로 전송
 - ✓ nonpersistent: 누구도 전송 매체를 사용하고 있지 않다면 데이터를 바로 보내고 누군가 사용 중이면 임의의 시간을 기다렸다가 다시 상태 확인
 - ✓ p-persistent: 전송 매체가 사용 중이면 기다렸다가 다시 상태를 확인하고 아무도 사용하고 있지 않다면 p 확률로 데이터 전송



네트워크 인터페이스 계층

- 네트워크 인터페이스 계층의 프레임 전송

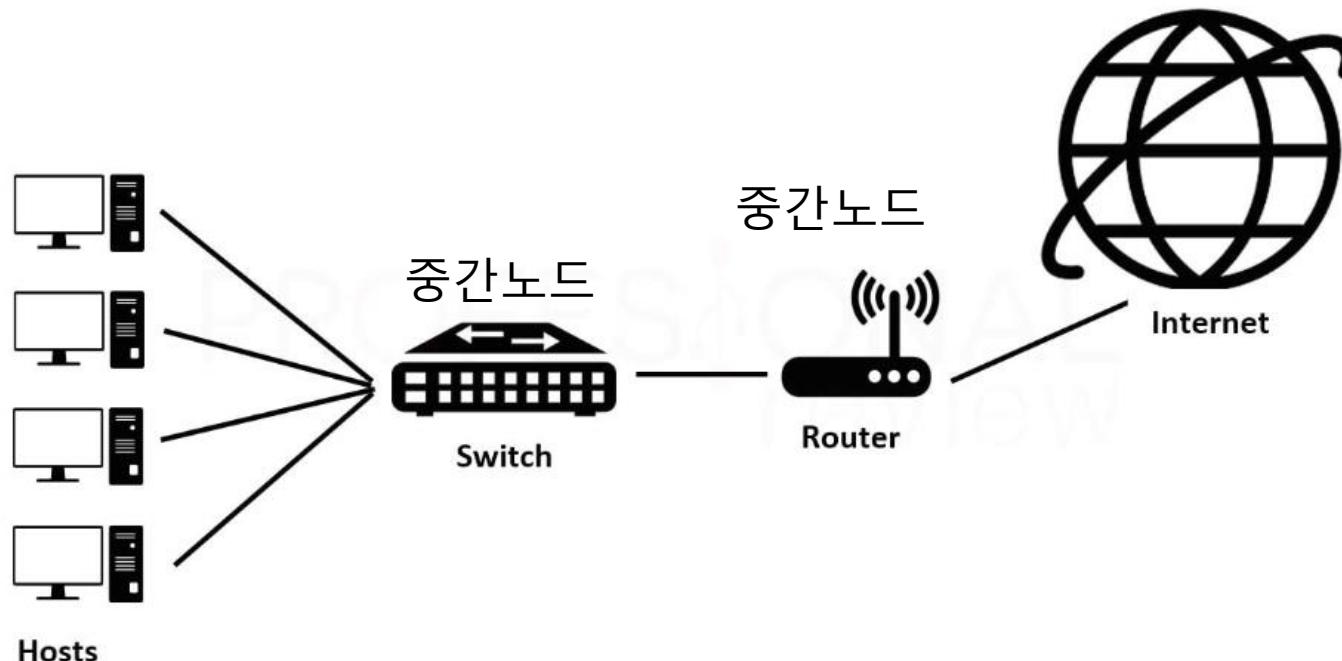
- 노드 (Node): 네트워크 기능을 활용하여 유무선 전송 매체로 데이터를 송수신 할 수 있는 네트워크 주소가 할당된 데이터 송수신 장치 (시스템).
- 노드의 예: PC, 노트북, 스마트폰, 프린터, 라우터, 인터넷 공유기 등.
- 노드의 네트워크 주소 = MAC 주소



네트워크 인터페이스 계층

- 네트워크 인터페이스 계층의 프레임 전송

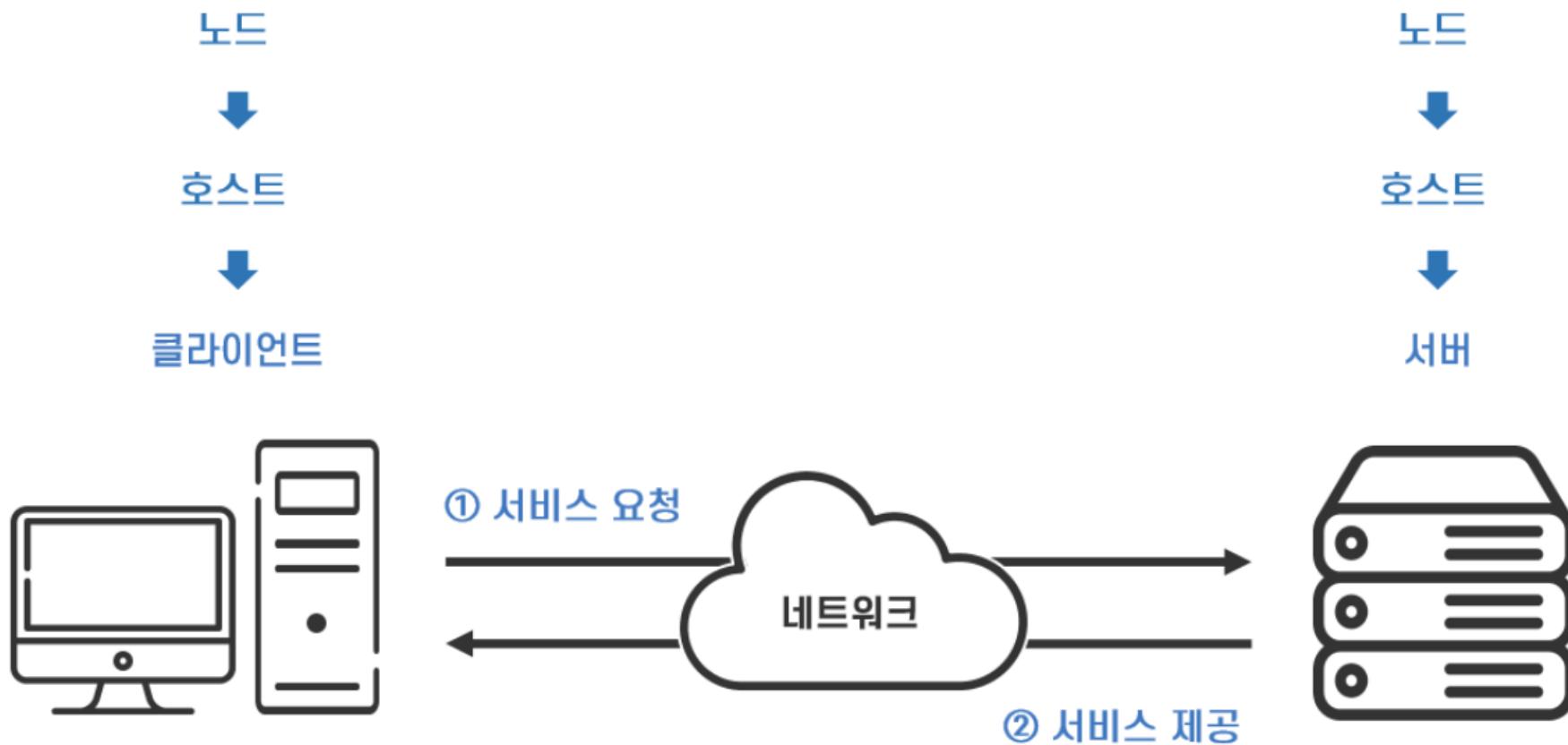
- 호스트 (Host): 노드 중에서 애플리케이션을 실행할 수 있는 네트워크 장치.
- 호스트의 네트워크 주소 = IP 주소 & 포트 번호
- 모든 호스트는 노드이지만, 모든 노드가 호스트는 아님.
- 예: PC, 노트북, 스마트폰은 호스트 & 노드, 그러나 스위치, 라우터는 노드이지만 호스트는 아님
- 중간 노드 (Intermediate Node): 노드 중에 호스트가 아니면서 호스트 중간에서 호스트들의 데이터를 중계하는 노드.



네트워크 인터페이스 계층

- 네트워크 인터페이스 계층의 프레임 전송

- 서버 (Server): 서비스를 제공하는 호스트
- 클라이언트 (Client): 서비스를 요청하고 사용하는 호스트
- 일반적인 컴퓨터나 서버는 서버도 될 수 있고, 클라이언트도 될 수 있는 호스트.



네트워크 인터페이스 계층

- 네트워크 인터페이스 계층의 프레임 전송

- 네트워크 인터페이스 계층 프로토콜 (MAC 프로토콜, Medium Access Control Protocol)

- ✓ 전송 매체 (Media)에 대한 접근 (Access)을 제어 (Control)하는 프로토콜 (규약)
 - ✓ 송수신 노드 간에 데이터가 효율적으로 전송되도록 제어
 - ✓ 전송 도중에 발생한 오류를 정정

- 데이터 전송 제어 (순서 & 흐름)

- ✓ 데이터에 번호를 부여하여 전송 순서를 제어

- ✓ 수신 노드의 수신 성능 (속도와 오류)를 확인하여 데이터 전송 흐름을 제어

- 프레임 (= 네트워크 인터페이스 계층의 데이터 전송 단위) 전송 방법

- ✓ 프레임 헤더에 수신지 MAC 주소와 송신지 MAC 주소가 있음.

- ✓ 수신 노드는 자신의 MAC 주소와 수신한 프레임 헤더의 수신지 MAC 주소가 일치하는지 확인하고, 일치하면 수신하여 데이터 처리하고, 불일치하면 무시함.

- ✓ 수신 노드는 수신한 프레임을 성공적으로 수신하면, 성공적인 수신 결과를 송신지에 알리기 위해, 응답 프레임 (=ACK 프레임)을 생성하여, 응답 프레임의 수신지 MAC 주소에 수신한 프레임의 송신지 MAC 주소를 입력하여 송신(응답)함.

네트워크 인터페이스 계층

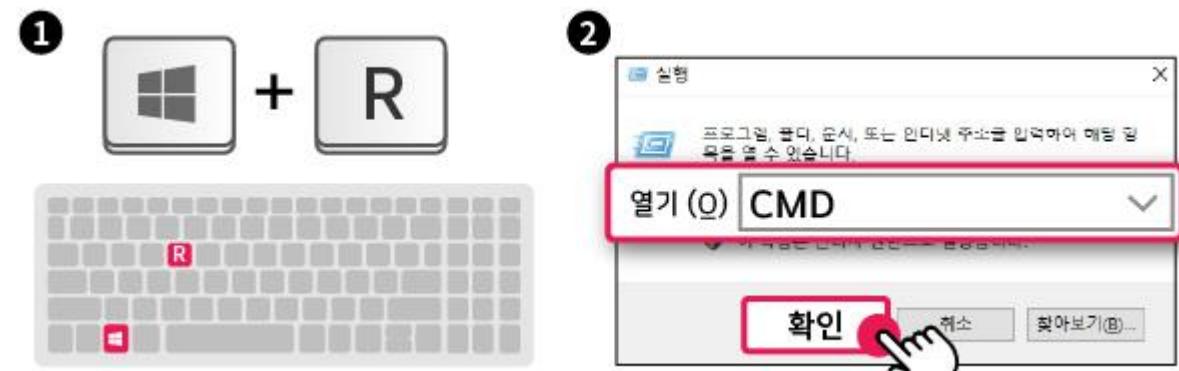


- 네트워크 인터페이스 계층의 프레임 전송
 - MAC 주소 (Medium Access Control Address) = 물리 주소 = 하드웨어 주소
 - ✓ 총 12개의 16진수로 구성
 - ✓ 앞쪽 6개는 네트워크 카드를 만든 회사 식별자 (벤더 코드 = OUI, Organizationally Unique Identifier)
 - ✓ 뒤쪽 6개는 호스트 식별자로서 각 회사에서 임의로 붙이는 시리얼 번호 (호스트 ID)
 - ✓ 하나의 호스트에 여러 개 네트워크 인터페이스가 있으면 여러 개 MAC 주소 할당될 수 있음

```
관리자: C:\Windows\system32\cmd.exe
C:\Users\k_im.hyoungbo> ipconfig/all
Windows IP 구성

호스트 이름 : LGES5R08Y094
주 DNS 접미사 : CLOUDPC.LGE.NET
노드 유형 : 흔성
IP 라우팅 사용 : 아니요
WINS 프록시 사용 : 아니요
DNS 접미사 검색 목록 : CLOUDPC.LGE.NET
LGE.NET

이더넷 어댑터 이더넷 3:
연결된 DNS 접미사 : Microsoft Hyper-V Network Adapter #3
설명 : 물리적 주소 : 00-1D-D8-B9-C5-A1
IPv4 주소 : 10.222.29.137(기본 설정)
서브넷 마스크 : 255.255.255.0
기본 게이트웨이 : 10.222.29.1
DNS 서버 : 10.223.130.200
10.223.130.201
```



XX:XX:XX:XX:XX:XX

OUI(Organization
Unique Identifier)

Host Identifier

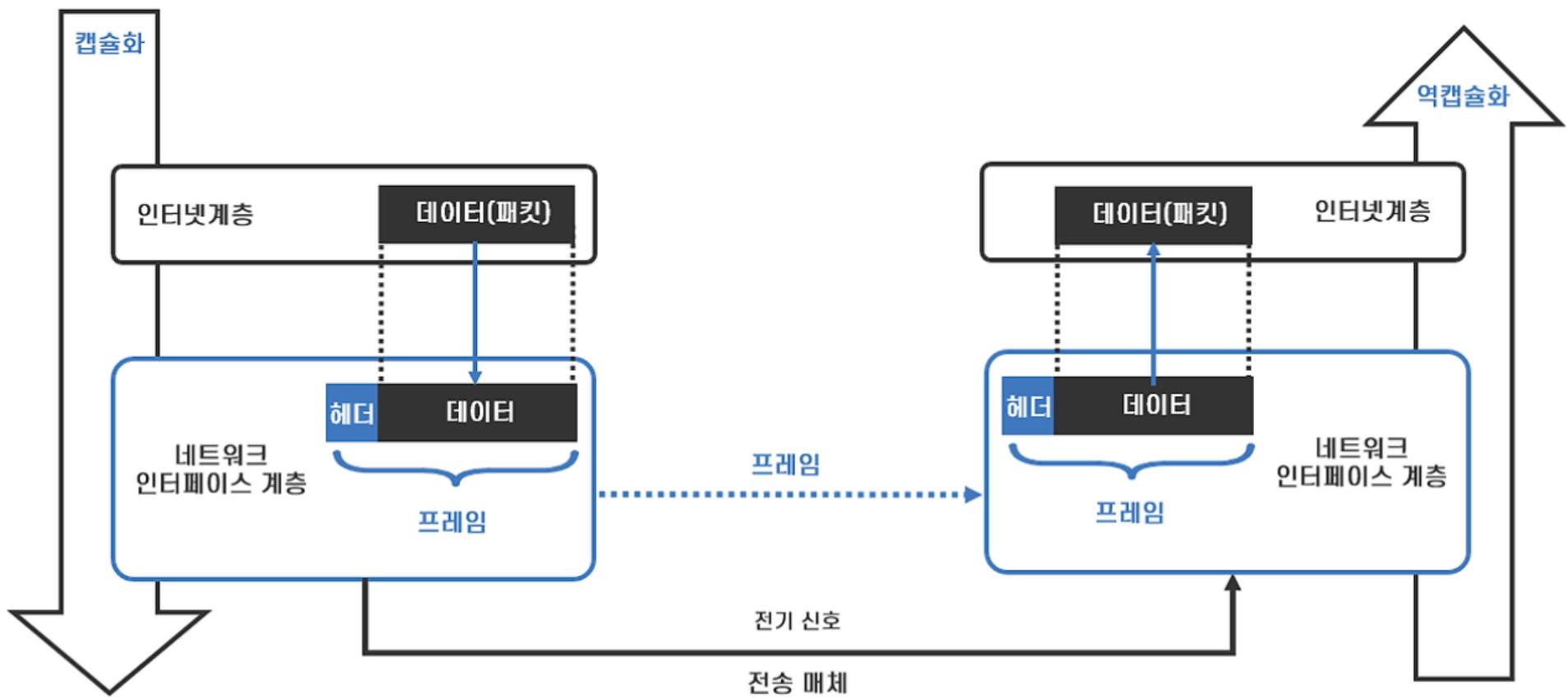
네트워크 인터페이스 계층

- 네트워크 인터페이스 계층의 프레임 전송

- 인터넷 계층이 네트워크 인터페이스 계층으로 보내는 데이터(패킷)의 앞에 헤더를 붙이고, 패킷 뒤에 트레일러 (Trailer)를 붙여서 프레임 완성.
- 트레일러: FCS (Frame Check Sequence) 방법으로 데이터 오류 (=무결성) 확인

송신 호스트

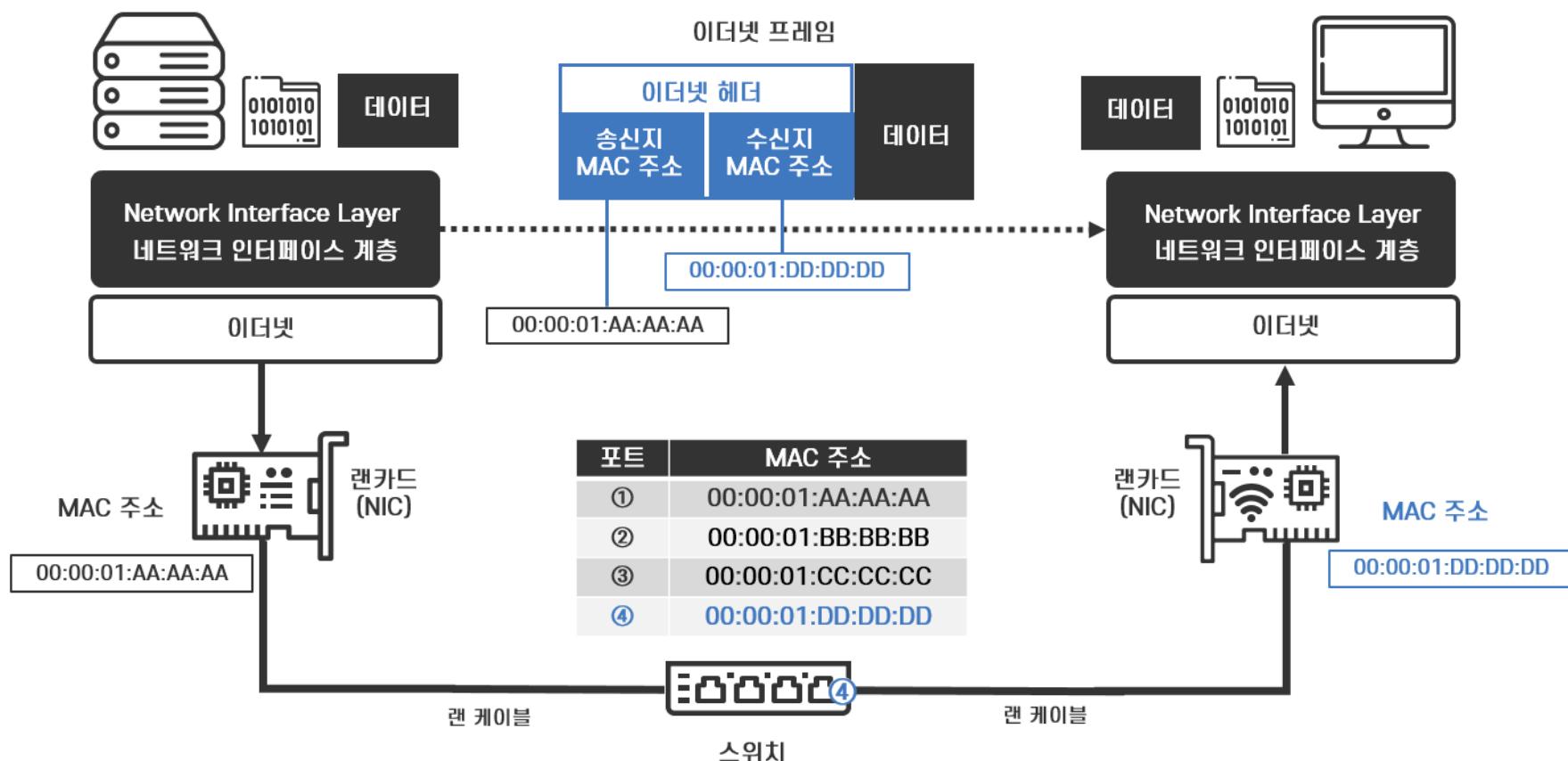
수신 호스트



네트워크 인터페이스 계층

- 네트워크 인터페이스 계층의 프레임 전송

- 유선 전송

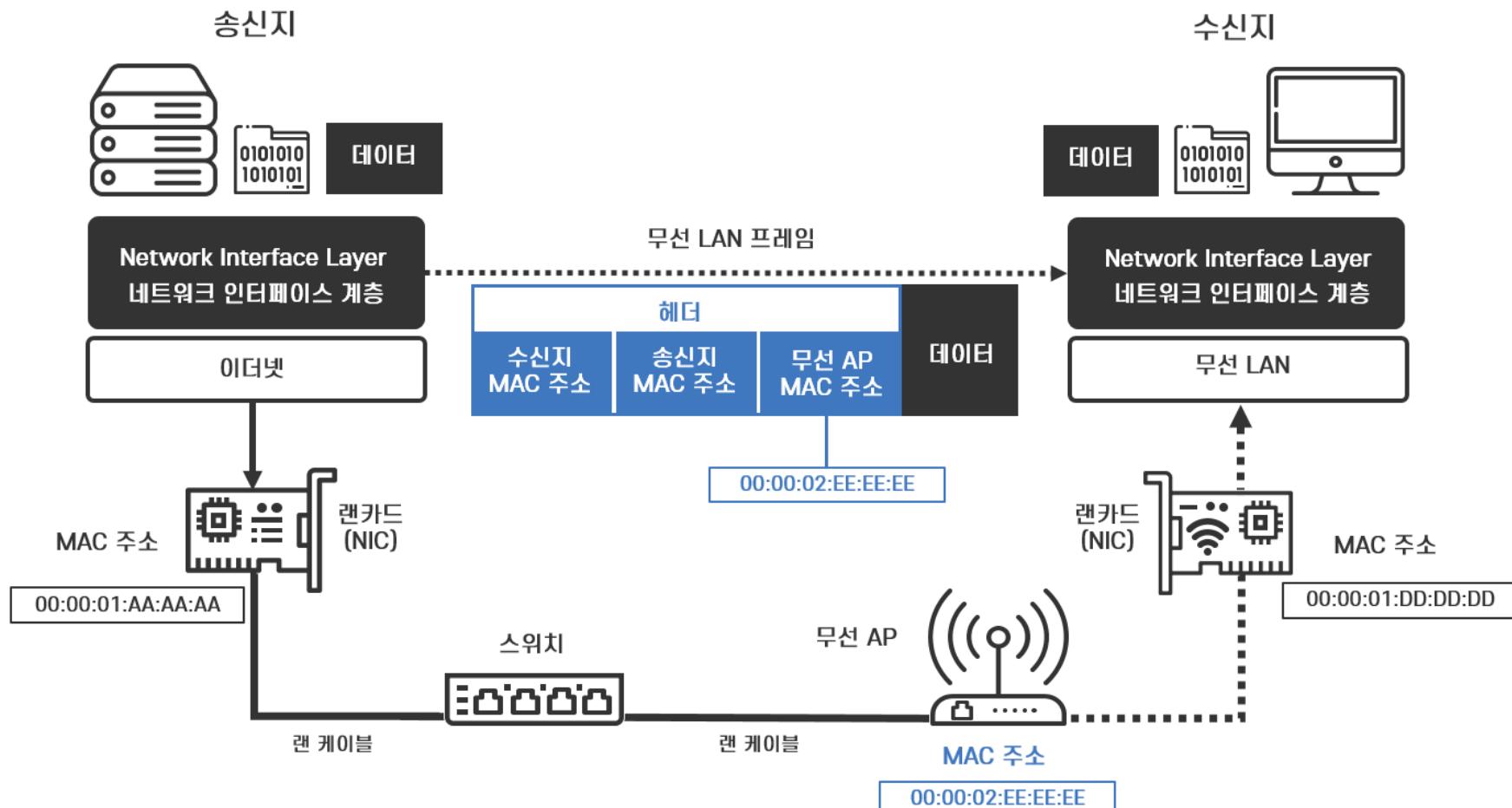


네트워크 인터페이스 계층

- 네트워크 인터페이스 계층의 프레임 전송

- 무선 전송

- ✓ 수신자 MAC 주소와 송신자 MAC 주소 뿐만 아니라 무선 AP의 MAC 주소를 포함한 이더넷 헤더를 붙인 이더넷 프레임을 수신자로 전송 → 수신 호스트가 데이터를 수신 후 응답하기 위해 무선 AP와 송신자 MAC 주소 필요

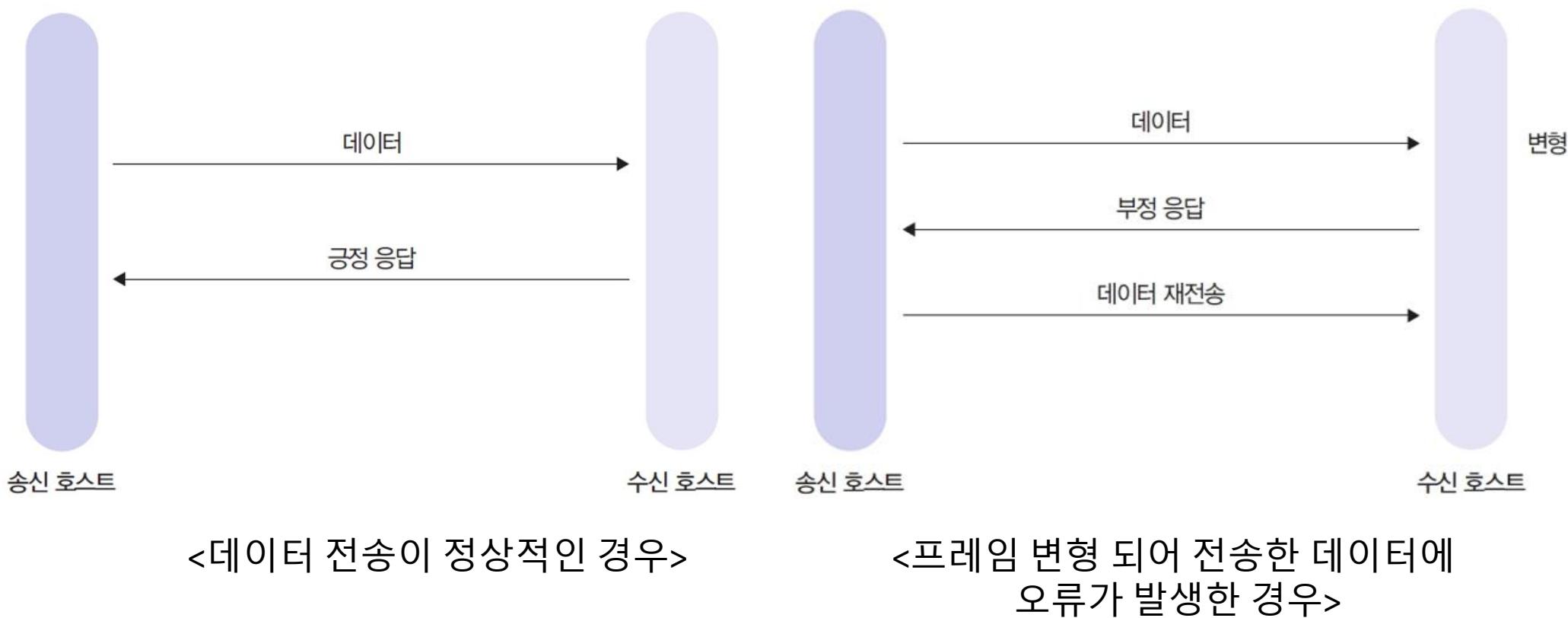


네트워크 인터페이스 계층

- 네트워크 인터페이스 계층의 프레임 전송

- 데이터 전송 오류의 유형

- ✓ 프레임 변형 오류: 데이터 프레임이 수신 호스트에 도착했지만, 전송 중 노이즈나 간섭에 의해 데이터 변형되는 경우 (=비트 오류)
 - ✓ 원인: 전송 매체의 잡음, 신호 감쇠, 간섭 등

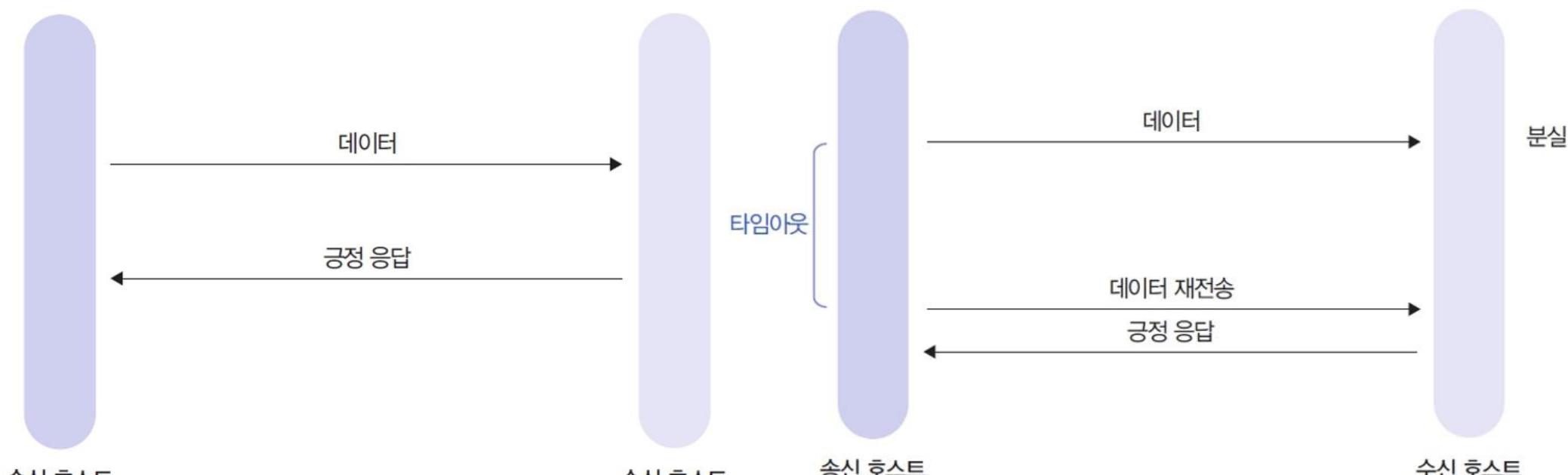


네트워크 인터페이스 계층

- 네트워크 인터페이스 계층의 프레임 전송

- 데이터 전송 오류의 유형

- ✓ 프레임 분실 오류: 송신 호스트가 전송한 데이터 프레임이 전송 과정에서 사라지는 오류
 - ✓ 원인: 네트워크 혼잡, 라우터 과부하, 전송 매체 결함 등



<데이터 전송이 정상적인 경우>

<프레임 분실 되어 전송한 데이터에
오류가 발생한 경우>

네트워크 인터페이스 계층



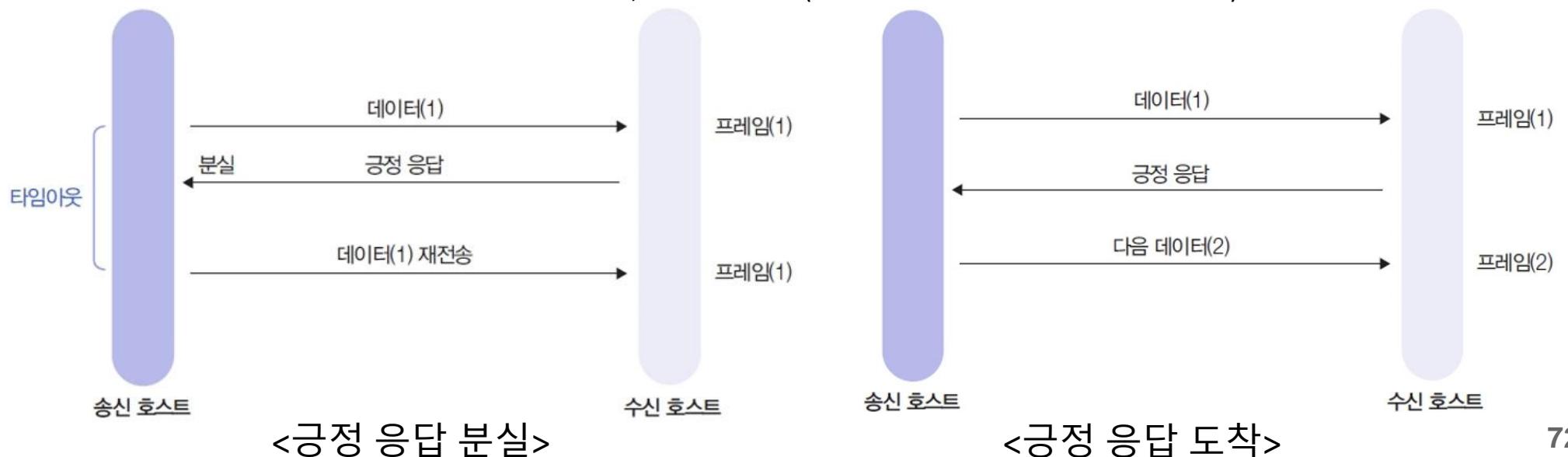
• 네트워크 인터페이스 계층의 프레임 전송

- 순서번호 (=프레임 재전송 판단과 중복 전송 방지하는 방법)

- ✓ 오류 없이 수신된 중복 데이터 프레임을 판별하기 위해 프레임 순서 번호 기록하고 검사
- ✓ 0부터 순차적으로 증가하는 일련번호 (=은행의 대기 번호표 체계와 유사)
- ✓ 임의의 값을 넘어가면 0으로 순환하므로 흐름 제어 기능을 고려해서 최댓값 설계해야 함

- 타임 아웃과 재전송

- ✓ 데이터 프레임의 긍정 응답이 정해진 시간 동안 수신되지 않으면 (=타임아웃), 프레임 분실 판단 후 재전송 (=동일한 순서 번호의 데이터 프레임 전송)
- ✓ 데이터 프레임의 긍정 응답 수신되면, 다음 데이터 (=다음 순서 번호의 데이터 프레임) 전송



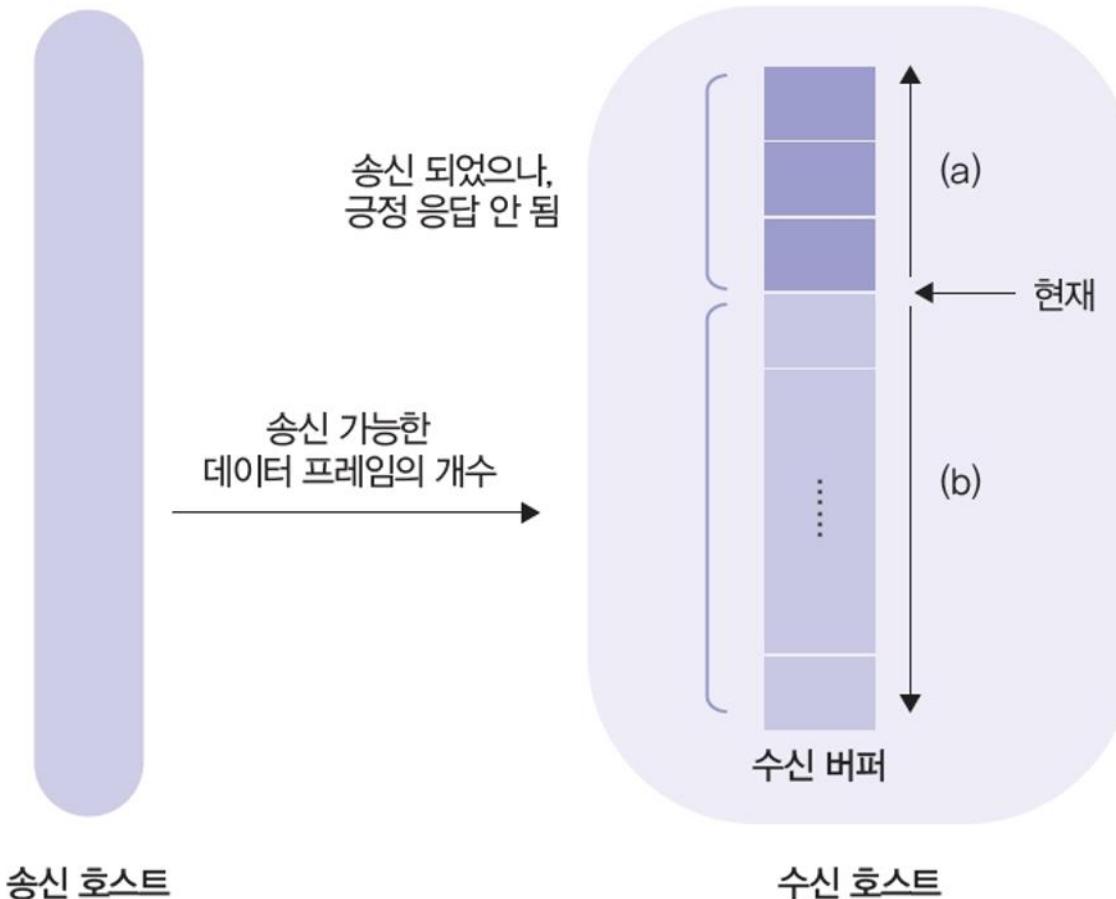
네트워크 인터페이스 계층



- 네트워크 인터페이스 계층의 프레임 전송

- 흐름제어

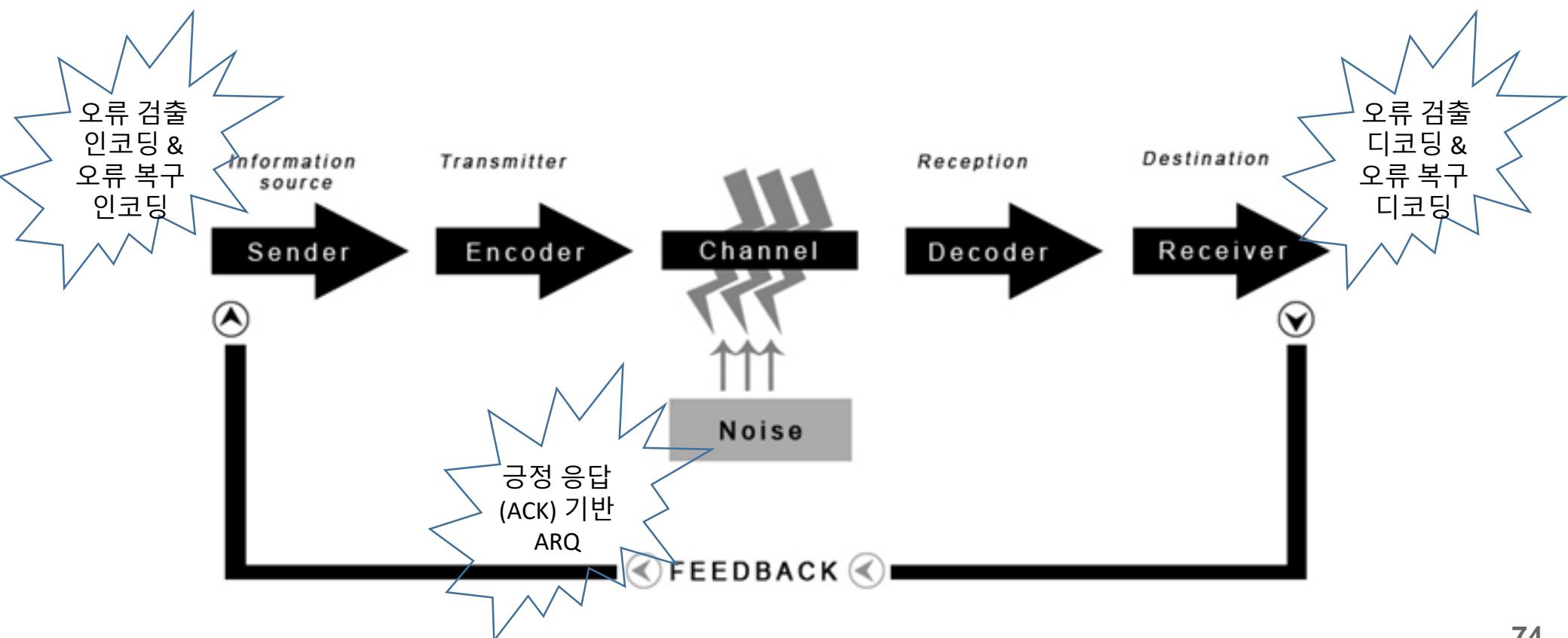
- ✓ 송신 호스트가 수신 호스트에서 수신할 수 있는 전송 속도를 유지하면서 데이터 프레임을 전송하는 기능
 - ✓ 송신 호스트가 수신 호스트보다 매우 빠르게 데이터를 전송하는 경우에 필요
 - ✓ 수신 호스트의 수신 버퍼 상태 정보를 이용한 흐름 제어의 원리



네트워크 인터페이스 계층



- 네트워크 인터페이스 계층의 프레임 전송
 - 프레임 전송 과정 중 데이터 오류 탐지 및 복구하는 방법
 - ✓ 방법 1: 오류 검출 코딩 (Error Detection Coding) 기법
 - ✓ 방법 2: 오류 복구 코딩 (Error Correction Coding) 기법
 - ✓ 방법 3: 재전송 (Retransmission) 기법 (=Automatic Repeat reQuest)



네트워크 인터페이스 계층



- 네트워크 인터페이스 계층의 프레임 전송

- 패리티 비트

- ✓ 데이터 전송 과정에서 오류가 발생했는지 검사하기 위해 추가되는 비트
 - ✓ 1 비트 패리티 = 전송 데이터에 1비트를 추가하여 전송
 - ✓ 송신 노드가 전송하려는 데이터 비트에서 1의 개수를 체크하여 패리티 규칙 (짝수 or 홀수)에 맞춰 패리티 비트 추가하고, 수신 노드가 수신 데이터가 정해진 패리티 규칙에 맞는지 검사하여 데이터 오류 여부 체크
 - ✓ 오류 발생 여부만 체크할 수 있고, 오류 수정 불가능
 - ✓ 다수 비트 오류 발생할 때 오류 검출 불가능

- 패리티 규칙

- ✓ 짝수 패리티 비트: 전체 전송 비트의 1의 개수가 짝수가 되도록 패리티 비트를 정하는 방법
 - ✓ 홀수 패리티 비트: 전체 전송 비트의 1의 개수가 홀수가 되도록 패리티 비트를 정하는 방법

전송되는 데이터		
원본 데이터 (7비트)	짝수 패리티	패리티비트 포함 8비트
1001001	1	10010011
1010011	0	10100110
1111111	1	11111111
1000000	1	10000001

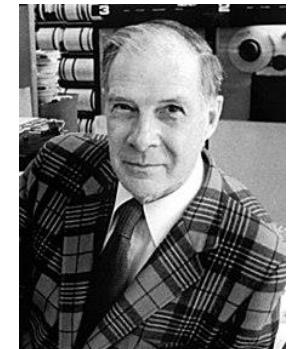
전송되는 데이터		
원본 데이터 (7비트)	홀수 패리티	패리티비트 포함 8비트
1001001	0	10010010
1010011	1	10100111
1111111	0	11111110
1000000	0	10000000

네트워크 인터페이스 계층

- 네트워크 인터페이스 계층의 프레임 전송

- 해밍코드

- ✓ 패리티 비트의 한계점 (= 에러 탐지만 가능하고 복구 불가능)을 해결한 방법
 - ✓ 리처드 웨슬리 해밍 (미국 수학자 & 벨연구소 연구원 & 대학 교수, 1915~1998)이 발명



- 해밍코드는 오류 위치 발견하고 정정할 수 있음

- ✓ 2의 거듭제곱번째 위치의 비트를 패리티로 사용 (1,2,4,8,16 ... 번째 비트는 패리티, 나머지는 데이터)

...	12	11	10	9	P4	7	6	5	P3	3	P2	P1
-----	----	----	----	---	----	---	---	---	----	---	----	----

- ✓ P1 패리티: 1,3,5,7,9,11 ... 번째 비트 오류 검사
 - ✓ P2 패리티: 2,3,6,7,10,11 ... 번째 비트 오류 검사
 - ✓ P3 패리티: 4,5,6,7,12,13,14,15,... 번째 비트 오류 검사
 - ✓ P4 패리티: 8,9,10,11,12,13,14,15,24,25,... 번째 비트 오류 검사

네트워크 인터페이스 계층

- 네트워크 인터페이스 계층의 프레임 전송
 - 패리티 비트 체크 범위

P8 check 범위

D12	D11	D10	D9	P8	D7	D6	D5	P4	D3	P2	P1
비트12	비트11	비트10	비트9	비트8	비트7	비트6	비트5	비트4	비트3	비트2	비트1

P4 check 범위

D12	D11	D10	D9	P8	D7	D6	D5	P4	D3	P2	P1
비트12	비트11	비트10	비트9	비트8	비트7	비트6	비트5	비트4	비트3	비트2	비트1

P2 check 범위

D12	D11	D10	D9	P8	D7	D6	D5	P4	D3	P2	P1
비트12	비트11	비트10	비트9	비트8	비트7	비트6	비트5	비트4	비트3	비트2	비트1

P1 check 범위

D12	D11	D10	D9	P8	D7	D6	D5	P4	D3	P2	P1
비트12	비트11	비트10	비트9	비트8	비트7	비트6	비트5	비트4	비트3	비트2	비트1

네트워크 인터페이스 계층

- 네트워크 인터페이스 계층의 프레임 전송

- 예: 1100100 을 해밍코드를 추가하여 전송해서, 한 비트에러나는 경우 오류 정정하기

해밍코드 수정 과정

1. 1, 2, 4, 8번째 자리를 제외하고 데이터 비트(1100100)를 채운다.

1	1	0	P4	0	1	0	P3	0	P2	P1
---	---	---	----	---	---	---	----	---	----	----

2. P4부터 차례로 짝수 패리티로 패리티 비트 값을 넣어 준다.

1	1	0	0	0	1	0		0		
1	1	0	0	0	1	0	1	0		
1	1	0	0	0	1	0	1	0	1	
1	1	0	0	0	1	0	1	0	1	1

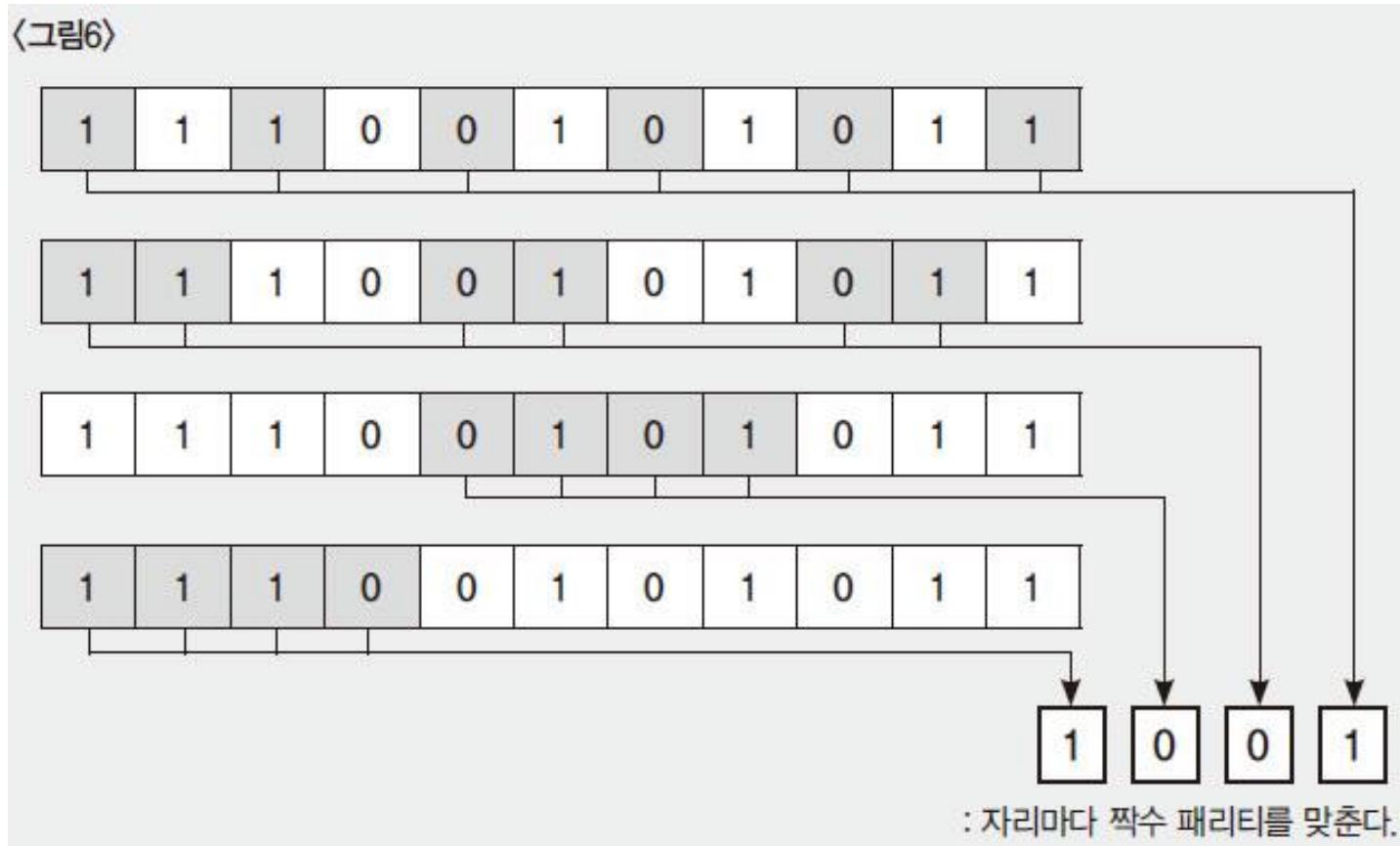
3. 해밍코드를 완성한다.

1	1	0	0	0	1	0	1	0	1	1
---	---	---	---	---	---	---	---	---	---	---

네트워크 인터페이스 계층

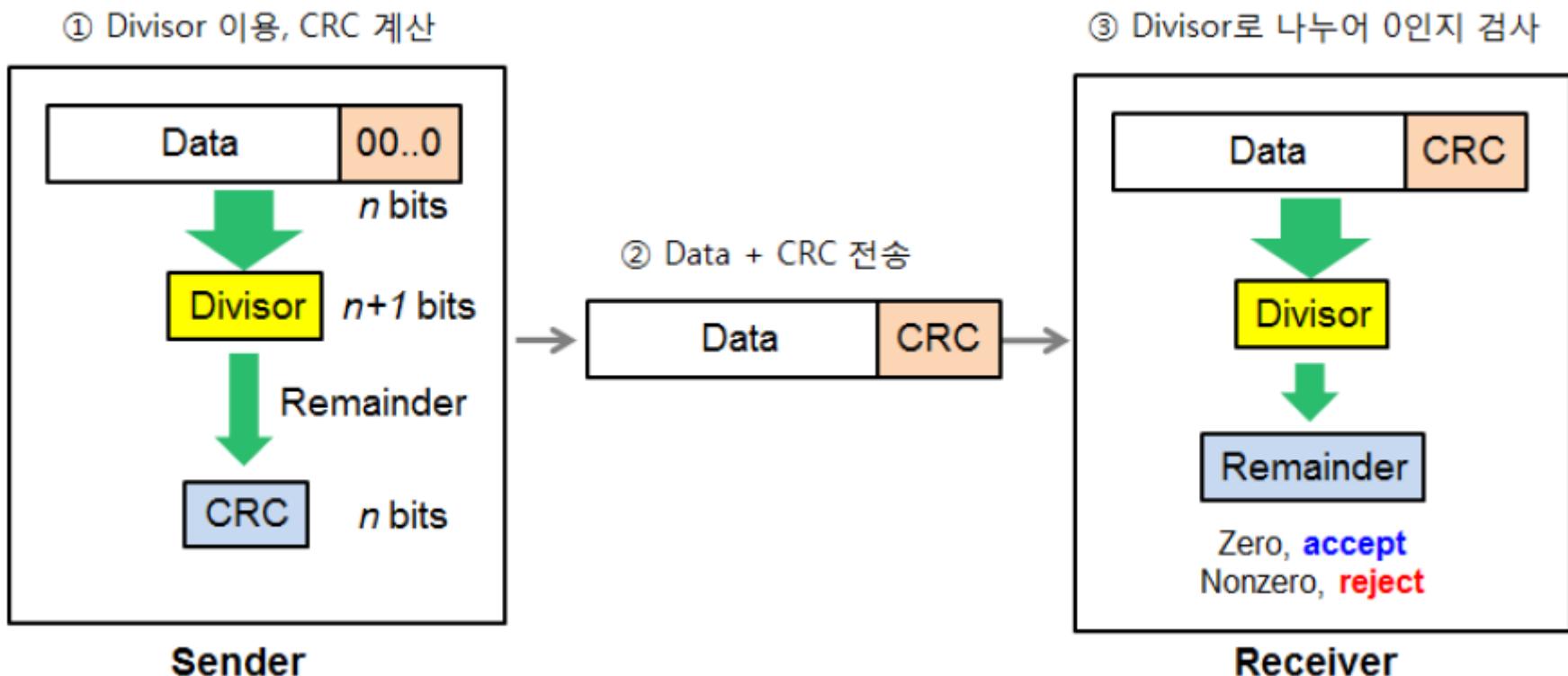
- 네트워크 인터페이스 계층의 프레임 전송
 - 전송한 데이터 (해밍코드 인코딩된 정보): 11000101011
 - 9번째 비트에러난 상태로 수신된 데이터: 11100101011
 - 패리티 계산한 결과 = 1001 → 십진수로 9 이므로 9번째 비트에러.

〈그림6〉



네트워크 인터페이스 계층

- 네트워크 인터페이스 계층의 프레임 전송
 - 다항 코드 기반의 오류 정정 기법 (=순환 중복 검사, CRC, Cyclic Redundancy Check) = 프레임체크시퀀스 (FCS, Frame Check Sequence)
 - ✓ 네트워크를 통해 데이터를 전송할 때 전송 데이터에 오류가 있는지 확인하기 위한 체크섬을 결정하는 방식
 - ✓ 송신 노드에서 CRC 값을 데이터에 붙여서 전송하고, 수신 노드에서 수신한 CRC 값을 검사하여 에러 검출
 - ✓ 수학적 연산 과정은 복잡하지만, 높은 오류 검출율과 간단한 하드웨어 구현 가능



네트워크 인터페이스 계층

• 네트워크 인터페이스 계층의 프레임 전송

- CRC 기법 단계별 동작

단계	기능
송신 & 인코딩 (CRC 추가)	<ul style="list-style-type: none">- 데이터워드 + n 비트로 구성된 데이터를 $(n+1)$ bit divisor로 나눠서 CRC 정보 생성- 데이터워드 + CRC로 이루어진 코드워드 생성- Divisor는 다항식 표현
정보 전송 (DATA + CRC)	<ul style="list-style-type: none">- 코드워드 (데이터워드 + CRC)를 네트워크를 통해 수신측에 전송- 노이즈와 간섭에 의해 에러 발생
수신 & 디코딩 (CRC 검사)	<ul style="list-style-type: none">- 수신된 코드워드를 Divisor 이용하여 나눔- 나머지가 0인 경우에 데이터 정상 수신- 나머지 발생 시 에러 발생한 것으로 판단 → 데이터 재전송

네트워크 인터페이스 계층



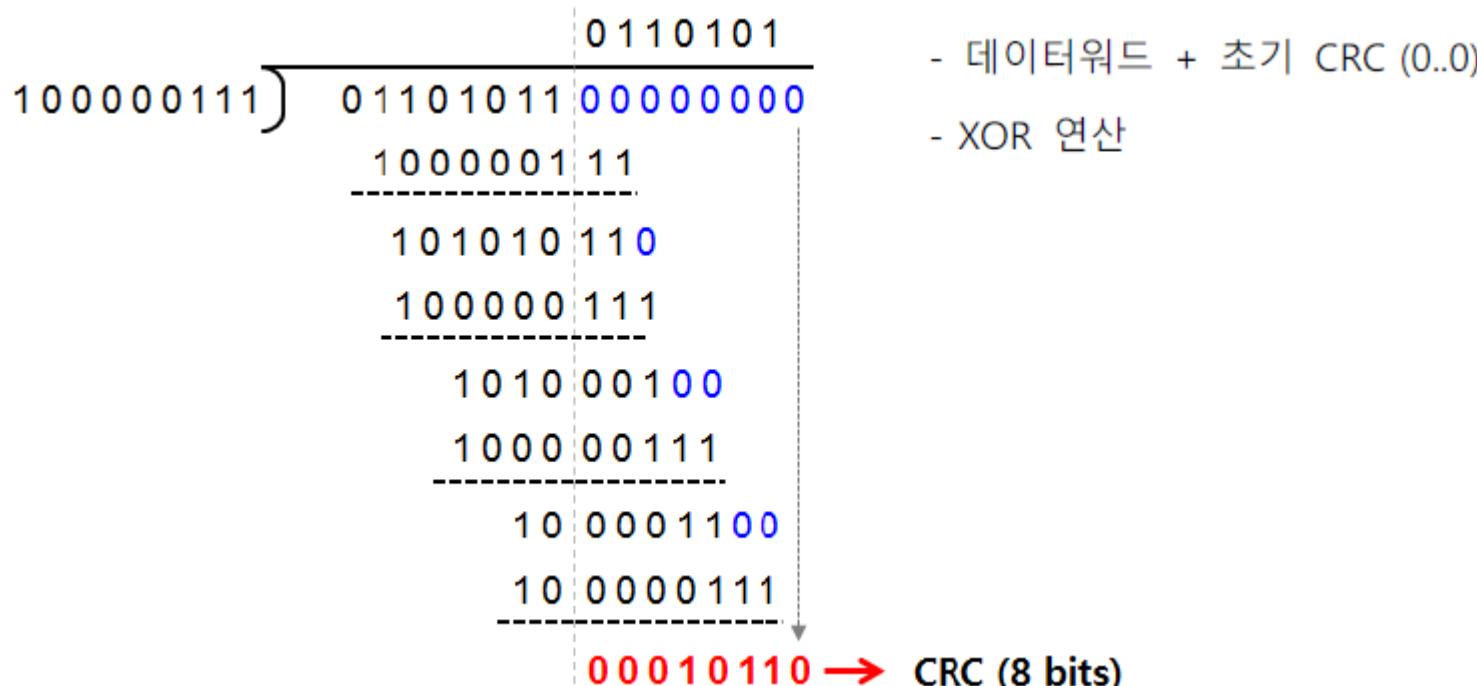
- 네트워크 인터페이스 계층의 프레임 전송

- 송신할 데이터 및 사용하는 CRC

- ✓ 데이터워드: 01101011

- ✓ Divisor: 100000111 (= $x^8 + x^2 + x + 1$ 인 CRC-8)

- 송신 및 CRC 인코딩하기 위한 코드워드 계산



- 전송 데이터 = 01101011 + 00010110 = 0110101100010110

네트워크 인터페이스 계층

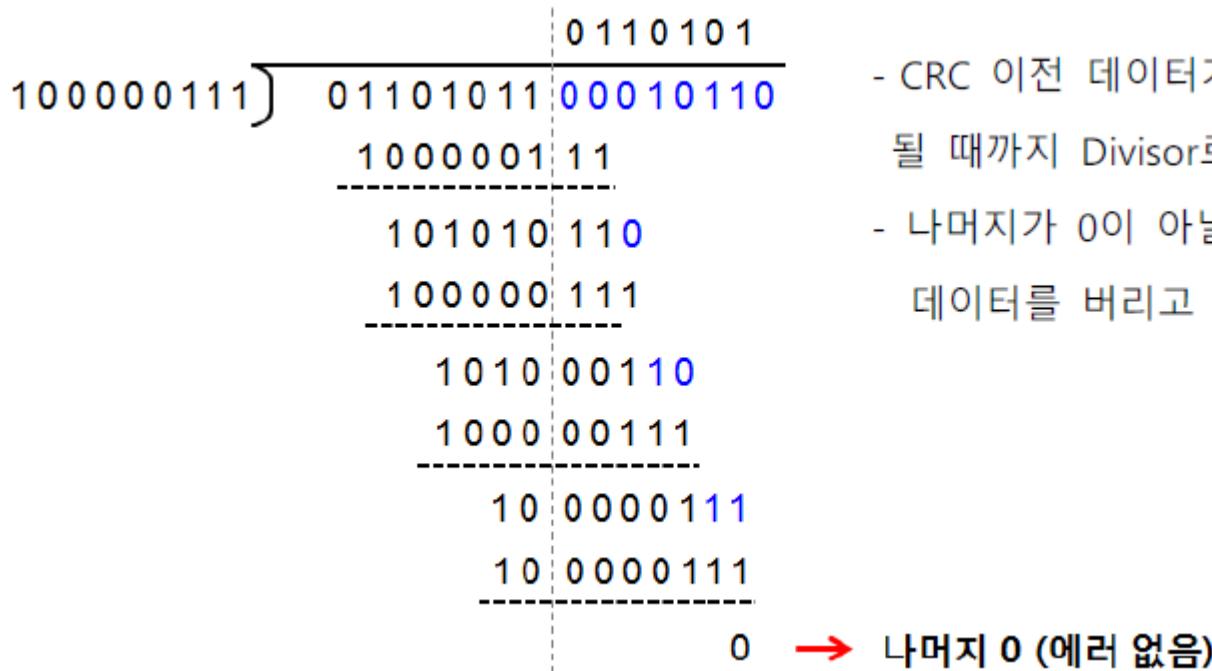
- 네트워크 인터페이스 계층의 프레임 전송

- 수신 & CRC 디코딩 & 에러 검증

- ✓ 수신 데이터 = 011010110**00010110**

- ✓ Divisor: 100000111 (= $x^8 + x^2 + x + 1$ 인 CRC-8)

- CRC 검사를 통한 에러 검증



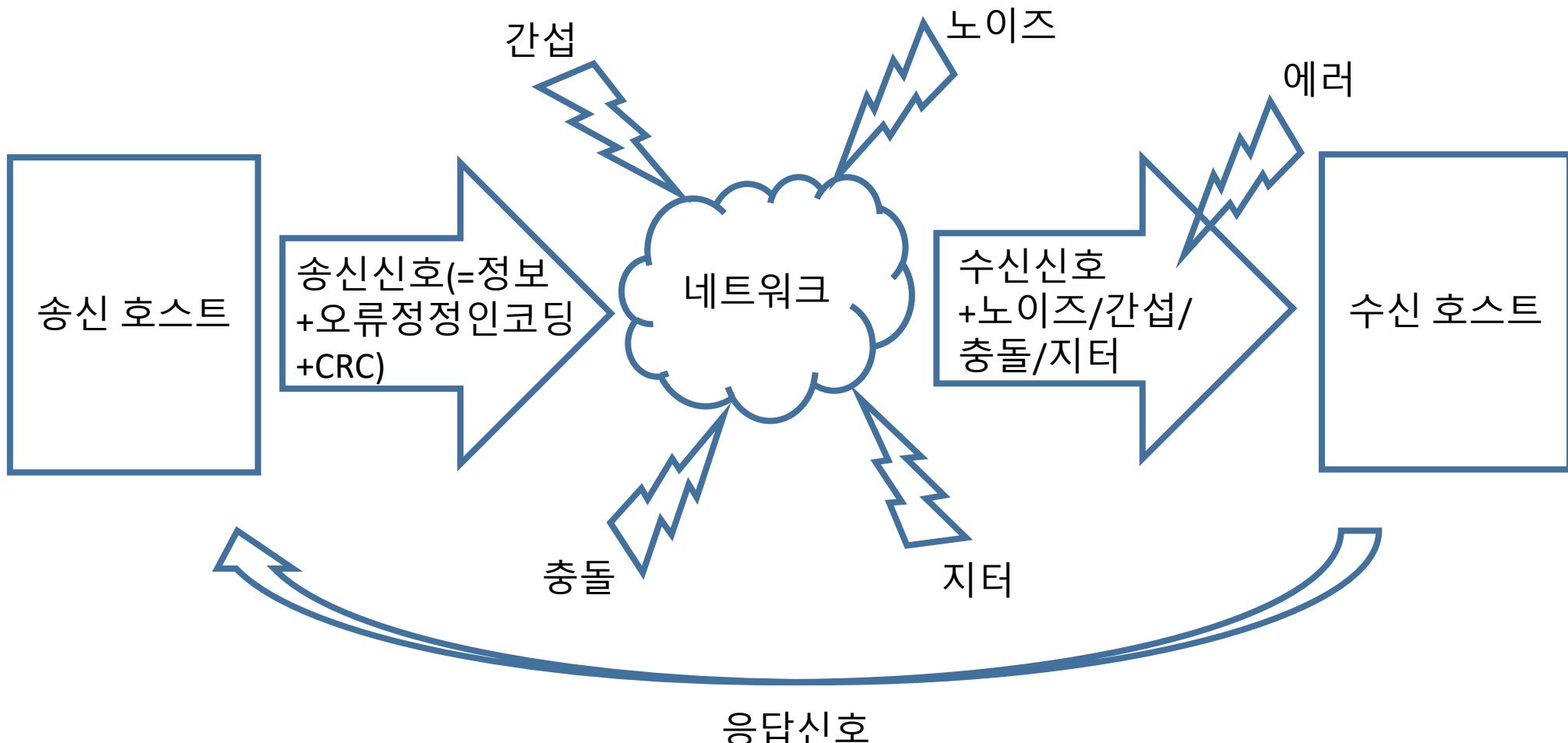
- CRC 이전 데이터가 모두 0이 될 때까지 Divisor로 나눔
- 나머지가 0이 아닐 경우, 데이터를 버리고 재전송 요청

- 검증 결과: Divisor로 나눈 값이 0 이므로 에러 없음.

네트워크 인터페이스 계층

- 네트워크 인터페이스 계층의 프레임 전송

- 네트워크의 신뢰 통신 방법 = 순방향 에러 정정 (Forward Error Correction, FEC)
 - + 역방향 에러 정정 (Backward Error Correction, BEC) + 재전송 기법

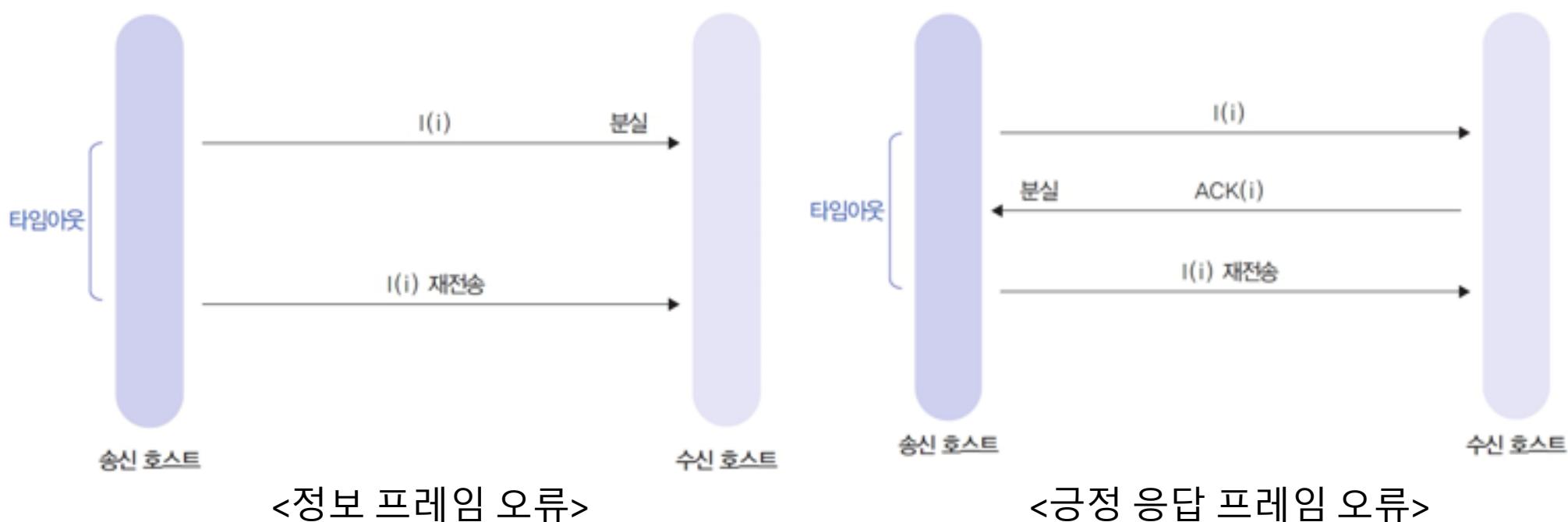


네트워크 전송 데이터

- 네트워크 인터페이스 계층의 프레임 전송

- 긍정 응답 프레임

- ✓ 수신한 정보 프레임 오류가 발생하지 않았으면 (=정상적으로 수신했으면) 송신 호스트에 해당 프레임을 올바르게 수신했다는 의미로 ACK 프레임 (=긍정 응답 프레임)을 회신
 - ✓ 수신한 정보 프레임 오류가 발생하면, 송신 호스트에게 긍정 응답 프레임을 회신하지 않고, 송신 호스트는 일정 시간 동안 긍정 응답 프레임이 회신 받지 못하면 재전송함

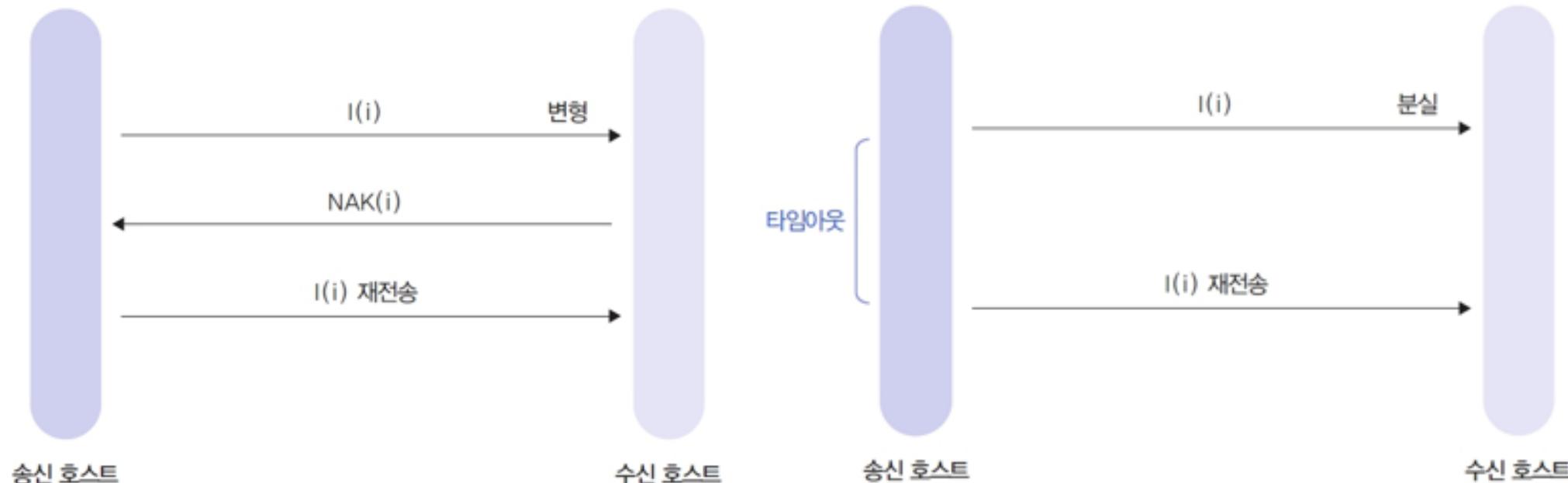


네트워크 전송 데이터

- 네트워크 인터페이스 계층의 프레임 전송

- 부정 응답 프레임

- ✓ 정보 프레임에 오류가 발생하면 수신 호스트는 부정 응답을 송신 호스트에게 회신하여 오류 발생을 인지하고 원래의 정보 프레임을 재전송하도록 요청
 - ✓ 송신 호스트는 부정 응답 프레임을 수신하면 재전송함



<정보 프레임 변형 오류>

<긍정 응답 프레임 분실 오류>

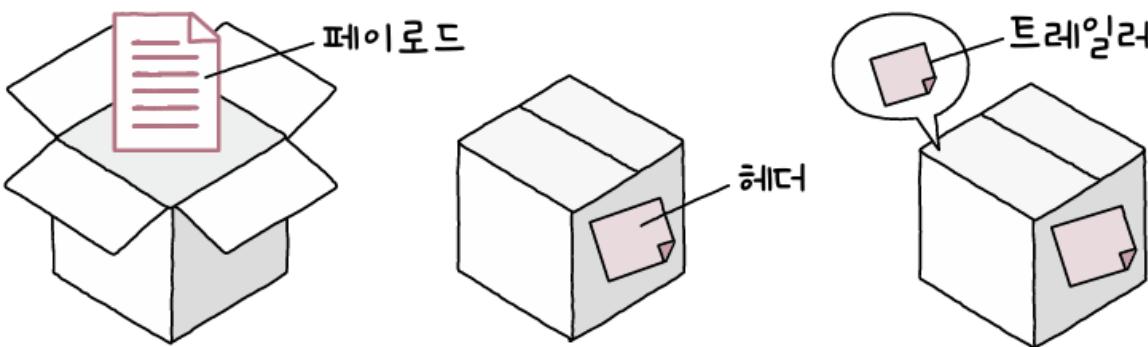
네트워크 전송 데이터

- 네트워크 전송 데이터 구조

- 헤더: 데이터 종류, 데이터 길이, 페이로드를 수신하는 방법 (=전송 기법)에 관한 정보 (헤더 데이터도 자체적인 CRC로 오류 검사)
- 페이로드: 송신 호스트가 전송하는 데이터 (오류 정정 인코딩 된 데이터)
- 트레일러: 페이로드 오류 확인하기 위한 체크섬 (FCS)



Preamble, Dest. Address, Source Address, Type/Length



패킷은 페이로드와 헤더로 구성되고, 때로는 트레일러도 포함됩니다.



Q&A

