



인하공업전문대학
INHA TECHNICAL COLLEGE

사물인터넷 7주차

인하공업전문대학 컴퓨터 정보과
김한결

목차

- 01. 통신의 이해
- 02. 통신의 종류
- 03. 통신의 과정
- 04. 통신의 확장, 네트워크

01. 통신의 이해

■ 통신 시스템

■ 구성요소

- 송신기, 수신기, 메시지 또는 데이터, 전송매체, 프로토콜의 다섯가지 요소로 구성됨

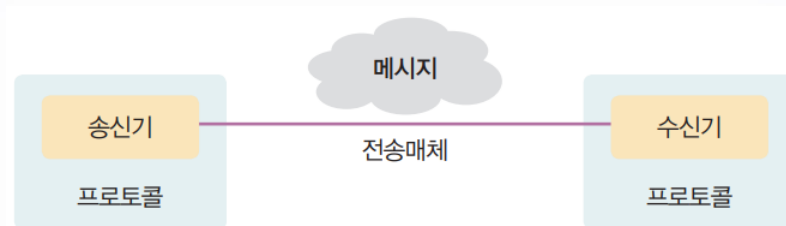


그림 3-1 통신 시스템의 구성요소

• 송신기와 수신기

- 송신기와 수신기는 통신 부품을 포함한 전체 제품을 의미
- 오늘날 송신기와 수신기는 데이터 처리를 담당하는 부분과 데이터 전송을 담당하는 부분으로 나뉨
- 통신 시스템이 네트워크로 확장되면 송신기와 수신기는 각각 단말, 단말장치, 노드, 호스트 등의 이름으로 불림
- 역할에 따라서 송신 단말/수신 단말, 송신 노드/수신 노드, 송신 호스트/수신 호스트로 구분

• 메시지

- 패킷의 가장 큰 특징은 쉽게 구분이 된다는 점
- 인터넷의 탄생과 함께 본격적으로 사용되기 시작된 패킷의 개념은 오늘날 대부분의 통신 네트워크에 사용
- 메시지를 잘게 분해하면 비트가 되므로 메시지를 비트의 흐름인 비트열로 볼 수 있음 (데이터 비트열)
- 메시지를 신호의 높낮이나 진폭의 길이로만 보는 관점도 있음

01. 통신의 이해

■ 통신 시스템

■ 전송속도

• 비트레이트

- 초당 전송 가능한 비트 수이며 단위는 bps 이며 b는 반드시 소문자로 표기
- 초창기 컴퓨터의 키보드는 대략 2,000bps 속도로 입력된 문자를 본체로 전송했고 최초의 인터넷 메시지는 54kbps 속도로 UCLA에서 스탠포드 대학교로 전송.
- 오늘날 LTE는 20MHz 대역에서 최대 전송속도 75Mbps이며 USB 및 이더넷 등의 유선통신은 Gbps전송속도를 가짐

• 보 레이트

- 초당 전송 가능한 펄스 또는 심볼의 수를 나타내며 단위는 baud
- 하나의 펄스가 하나의 비트와 매핑될 때 비트레이트와 보 레이트는 의미하는 바가 같음
- 하나의 펄스에 여러 개의 비트가 매핑되거나 반대로 여러 개의 펄스에 하나의 비트가 매핑되면 bps와 baud의 값은 달라짐

■ 대역폭

- 대역폭은 주어진 데이터를 전송하는 데 필요한 주파수폭을 말함
- 대역폭이 넓다는 것은 그만큼 데이터를 빨리 전송할 수 있다는 의미
- 유선통신에서는 주파 수폭이 의미가 없으므로 대역폭을 데이터 전송속도와 동일한 의미로 사용

01. 통신의 이해

■ 전송매체 - 유선통신과 무선통신

■ 유선통신 케이블

- 유선통신: 전류가 흐르는 물리적인 연결선을 이용하는 방식
- 신호를 전송하는 선로의 재질에 따라 구리를 사용하는 케이블과 광섬유를 사용하는 케이블로 구분
- 인터넷 접속을 위해서는 광케이블을 반드시 거쳐야 함

• 동축 케이블

- 내부도체(구리선)부터 외부의 플라스틱 외피(재킷)까지 모두 중심축이 같다는 의미
- 중심에 있는 내부도체가 신호를 잘 전송할 수 있도록 몇 겹의 보호 장치를 사용
- 초창기 해저 케이블과 가정용 케이블 TV의 연결매체로 사용
- 최근에는 바깥 지름이 0.2~0.38mm 정도로 매우 가는 마이크로 동축 케이블이 개발되어 다양한 분야에 사용



(a) 단면



(b) 케이블 TV용 동축 케이블

그림 3-2 동축 케이블

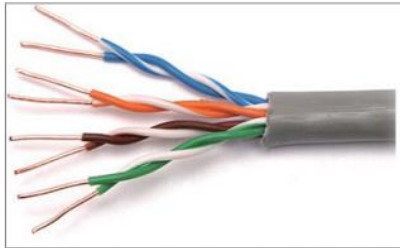
01. 통신의 이해

■ 전송매체 - 유선통신과 무선통신

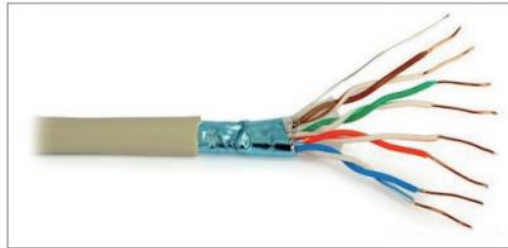
■ 유선통신 케이블

• 이중와선 케이블

- 통신 케이블이 길어져 외부의 잡음 신호나 전자기파의 영향을 받는 것을 막기 위해 내부에 두 개의 통신선이 꼬아져 있음
- 전화선과 근거리 통신망(LAN)에서 사용



(a) UTP 케이블



(b) STP 케이블

그림 3-3 이중와선 케이블

- (a): 근거리 통신망에서 일반적으로 사용되는 UTP 케이블
- (b): 알루미늄 호일과 같은 재료로 연선을 추가로 보호해 신호 전송 품질이 우수한 STP 케이블
- 주변에 잡음 신호가 많은 환경에서 사용되며 주로 저렴한 UTP 케이블이 사용

01. 통신의 이해

■ 전송매체 - 유선통신과 무선통신

■ 유선통신 케이블

• 광케이블

- 빛을 전송하는 코어, 코어를 둘러싸서 목적지로 빛이 전반사될 수 있도록 하는 클래딩, 외부를 감싸는 피복으로 구성
- 코어는 플라스틱이나 유리로 만드는데, 짧은 거리에는 플라스틱, 중장거리에는 유리를 이용
- 광케이블의 굵기는 클래딩을 포함해 대략 $125\mu\text{m}$ 가 표준, 코어는 목적에 따라 최소 $8\sim 10\mu\text{m}$ 까지 사용
- 싱글 모드와 멀티 모드로 구분
- 싱글 모드: 하나의 빛이 광케이블을 따라 목적지까지 신호를 전송, 중장거리 용도
- 멀티 모드: LED를 사용하여 빛이 확산됨, 싱글 모드보다 신호의 전송거리가 짧음, 주로 건물안에서 사용



(a) 구조



(b) 오디오용 광케이블의 예

그림 3-4 광케이블

01. 통신의 이해

■ 전송매체 - 유선통신과 무선통신

■ 무선통신

- 무선통신에 이용되는 전파는 진동수 3,000Hz부터 3,000GHz까지의 전자기파를 의미
- 전파의 파장은 주기적으로 나오는 파동의 마루와 마루 사이, 골과 골 사이의 거리, 단위는 미터
- 전파의 주파수는 마루에서 마루까지, 골에서 골까지 1초에 몇 번 전파가 반복되는지를 나타내는 진동 횟수, 단위는 Hz

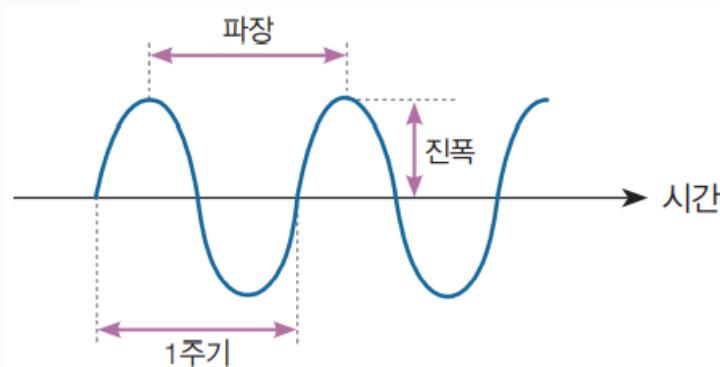


그림 3-5 주기, 파장, 진폭

- 주파수가 높으면 단위 시간당 더 많은 정보를 전송할 수 있으나 장애물을 만나면 반사되거나 굴절되어 목표물에 도달
- 주파수가 낮은 전파는 직진성은 약하지만 장애물을 뛰어넘는 특성이 있어 넓은 지역으로 송신하는 데 유리하지만 전송되는 정보량은 적음

02. 통신의 종류

■ 직렬 통신과 병렬 통신

- 직렬 통신

- 데이터가 차례대로 일렬로 늘어서서 전송되는 방식 / 1차선 도로
- 8개의 데이터 비트가 하나의 통신선로를 따라 순차적으로 전송
- 키보드나 마우스

- 병렬 통신

- 여러 데이터가 나란히 동시에 전송되는 방식 / 8차선 도로
- 8개의 데이터 비트가 8개의 통신선로를 따라 나란히 전송
- 대용량의 데이터를 고속으로 전송할 필요가 있는 주변기기 (프린터, 스캐너, 하드디스크)

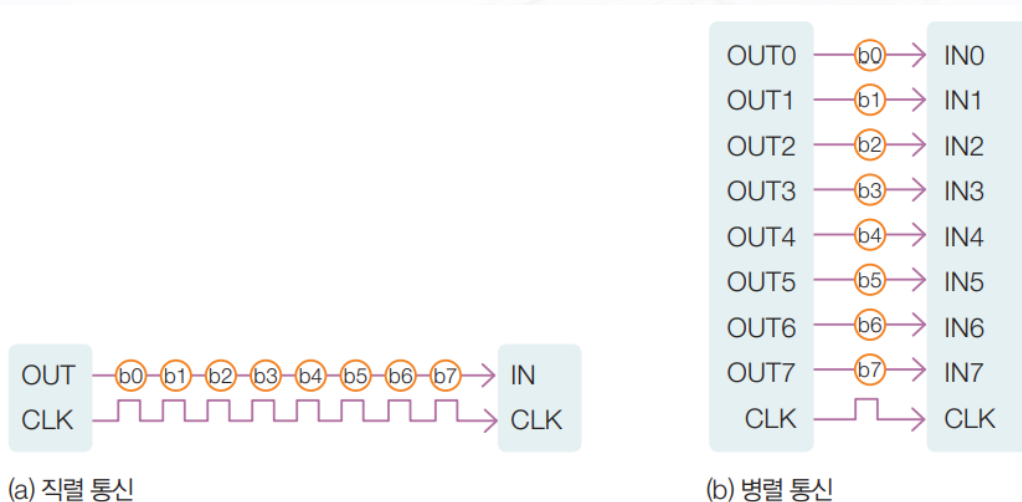


그림 3-6 직렬 통신과 병렬 통신

02. 통신의 종류

■ 직렬 통신과 병렬 통신

■ 병렬 통신의 퇴조와 직렬 통신의 발전

• 타이밍 틀어짐(클록 틀어짐)

- 송신기가 여러 개의 비트 정보를 동시에 전송하더라도 이들 신호가 수신기에 도달하는 시점이 제각각이 되는 현상
- 병렬통신은 개별 통신선로의 데이터 전송속도가 빨라질수록 나란히 전송되는 개별 데이터 비트를 정렬하기 어려워짐

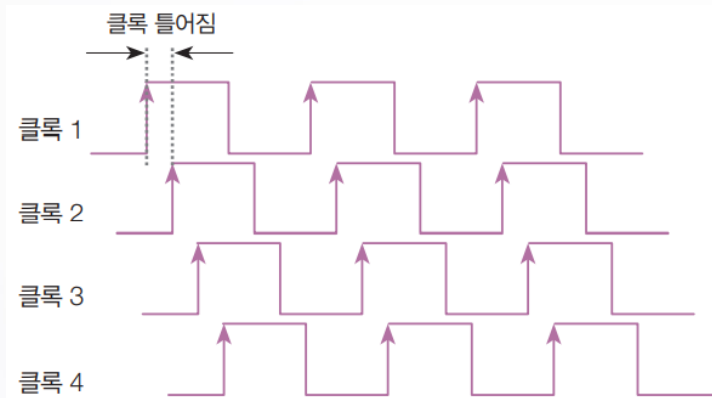


그림 3-7 병렬 통신에서의 클록 틀어짐

• 저전압 차분 신호 방식

- 두 개의 전선을 사용하여 동일한 신호가 양쪽 전선을 통해 전송되지만 신호 방향은 반대
- 한 개의 전선을 사용할 때보다 낮은 전압으로 신호를 보낼 수 있어 그만큼 데이터 전송속도가 빨라짐
- 직렬 통신이지만 두 개의 상반된 신호를 사용하는 방식으로 고속의 신호 전송이 가능

02. 통신의 종류

■ 단방향 통신과 양방향 통신

■ 단방향 통신

- 신호가 한쪽 방향으로만 전송 (예: 리모컨으로 텔레비전으로 신호 전송은 가능하지만 그 반대는 불가능)

■ 양방향 통신

- 신호가 양쪽 방향으로 전송 (예: 전화 통화에서 음성 신호는 양방향으로 이동)
- 전이중 통신과 반이중 통신으로 구분

• 전이중 통신

- 언제든지 송신과 수신이 동시에 가능
- 송신 채널과 수신 채널이 따로 있기 때문에 오가는 신호가 충돌하지 않음
- 예: LTE를 비롯한 이동통신

• 반이중 통신

- 송신과 수신이 한 번에 하나씩만 가능
- 송신 채널과 수신 채널의 구분이 없고 하나의 채널만 존재
- 예: 무전기로 통화를 하는 경우

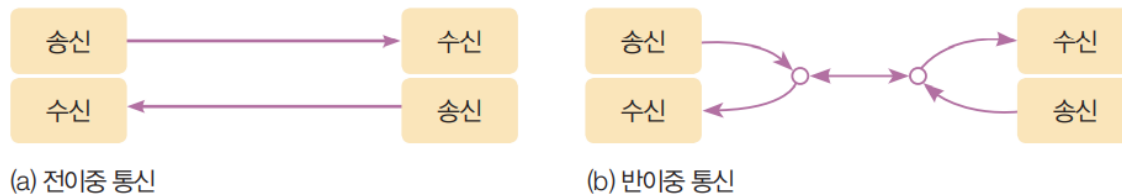


그림 3-8 전이중 통신과 반이중 통신

02. 통신의 종류

■ 동기식 통신과 비동기식 통신

■ 동기식 통신

- 송신기와 수신기가 하나의 신호에 맞추어서 동작하는 것 / 클럭: 동기 신호
- 송신기는 클럭 신호가 상승할 때 데이터 신호를 보내고 수신기는 클럭 신호가 상승할 때 데이터 신호를 읽음
- 더블 데이터 레이트: 데이터 전송속도를 빠르게 하기 위해 상승할 때와 하강할 때 모두 데이터를 전송할 수 있음

■ 비동기식 통신

- 미리 데이터 전송속도와 전송 방법에 대해 합의를 하여 송신기와 수신기를 동기화하는 별도의 신호없이 데이터를 전송
- 클럭 신호가 없기 때문에 적은 양의 신호를 일정한 단위로 구분해서 전송
- 일단 들어오는 신호를 일시적으로 저장한 후 서로 약속한 방법에 따라서 데이터를 해석 (7~8비트씩 끊어서 전송)

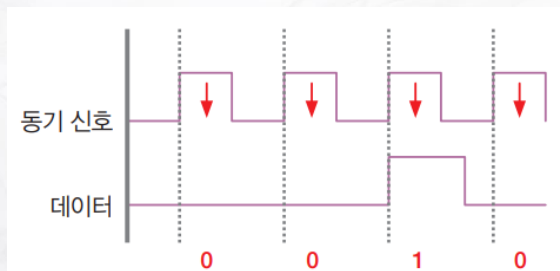


그림 3-9 동기식 통신

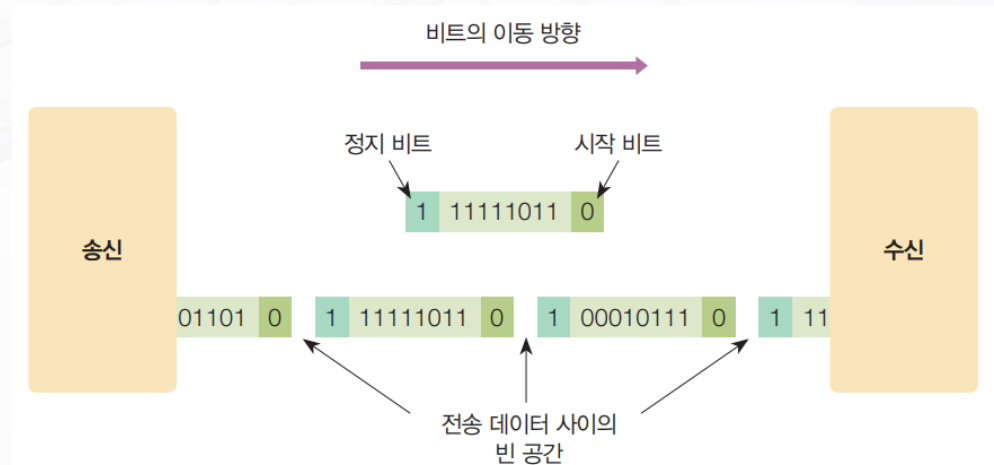


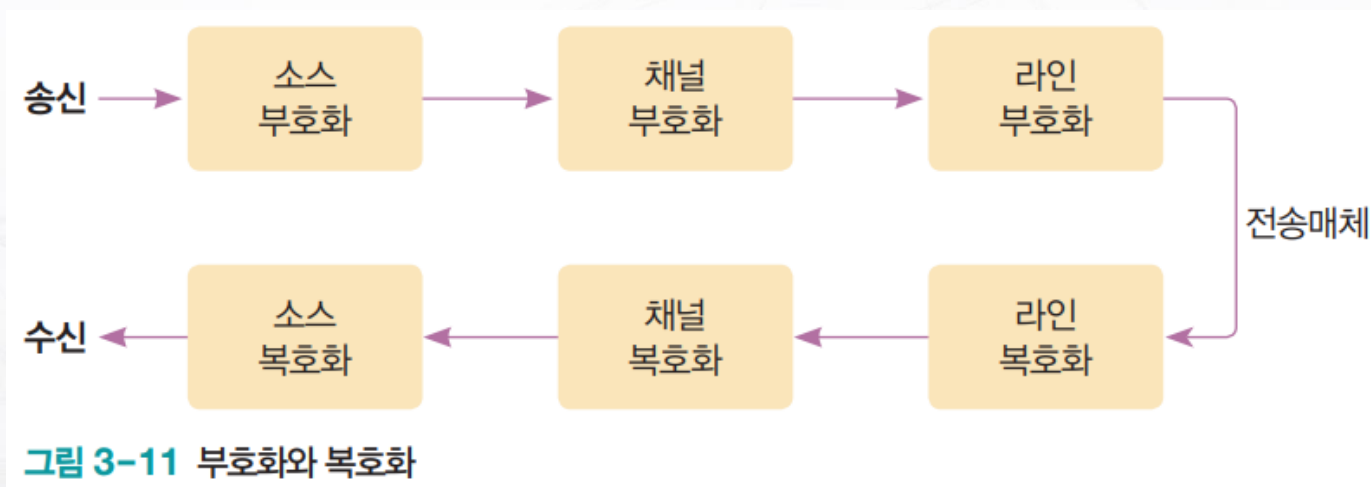
그림 3-10 비동기식 통신

03. 통신의 과정

■ 부호화

■ 부호화의 순서

- 부호화: 원격에 있는 컴퓨터로 이동하기 위해 데이터의 표현 방식을 변경, 데이터를 더 빠르고 안전하게 전송하기 위함
- 소스 부호화, 채널 부호화, 라인 부호화로 구분 (데이터 처리 순서도 동일)
- 소스 부호화: 전송하고자 하는 데이터의 양을 줄이는 압축 과정
- 채널 부호화: 전송이 완료된 데이터에서 오류를 검출하거나 정정할 수 있도록 원본 데이터에 추가 정보를 포함
- 라인 부호화: 통신매체의 특성에 맞도록 원본 데이터 배열을 통신에 적합한 데이터 배열로 변환
- 데이터를 수신한 쪽에서는 부호화의 반대 순서로 복호화 (라인-채널-소스 순서)
- 부호화 중 압축 과정에서 손실 압축 방식이 적용되었다면 압축 과정에서 사라진 정보는 복호화를 통해서도 복원할 수 없음



03. 통신의 과정

■ 부호화

■ 소스 부호화

- 원본 데이터를 압축해 전송 데이터의 양을 줄이는 과정, 데이터의 양이 줄어들면 전체 데이터를 더 빨리 전송할 수 있음
- 압축은 손실 압축과 무손실 압축으로 구분
- 손실 압축: 효과를 극대화하기 위해 인간의 시각이나 청각으로 인지하기 어려운 데이터를 삭제, 일반적인 멀티미디어 압축
- 무손실 압축: 복원하면 원본 파일과 같아짐, FLAC 오디오 파일이나 PNG 이미지 파일 등

■ 채널 부호화

- 전송 데이터에서 에러를 검출하거나 정정할 수 있도록 코드를 추가하는 과정
- 디지털 통신은 0과 1 신호만 사용 하므로 잡음의 영향을 어느 정도 분석할 수 있음
- 잡음의 영향을 판단할 수 있다면 원래 데이터로 복원할 수도 있고 데이터의 재전송을 요청할 수 있음
- 단위 길이의 데이터에 잘 계산된 코드를 부가하면 전송 중 신호에 문제가 생겼는지 수신지에서 확인이 가능

03. 통신의 과정

■ 부호화

■ 채널 부호화

• 패리티 체크

- 패리티 비트를 추가하는 가장 간단한 형태의 채널 부호화 방식
- 전송되는 비트의 1의 개수 에 따라 홀수 패리티와 짝수 패리티가 있음
- 만약 홀수 패리티로 통신하는 경우인데 1111 1001이 수신되었다면 전송 도중에 잡음이 추가된 것이므로 재전송을 요청

표 3-1 패리티 비트의 생성

종류	원본 데이터	패리티 비트	전송되는 데이터
홀수 패리티	1111 000	1	1111 0001
짝수 패리티	1111 000	0	1111 0000

• CRC

- 패리티 체크는 단순하여 쉽게 적용할 수 있지만 신뢰하기는 어려움
- 에러 발생 여부를 더 정확하게 판단하기 위해서 CRC 방식을 사용
- 데이터를 전송하기 전에 주어진 데이터의 값에 따라 CRC 값을 계산해 데이터에 붙여 전송
- 데이터 전송이 끝난 후 받은 데이터의 값으로 다시 CRC 값을 계산
- 두 값을 비교해서 다르면 데이터 전송 과정에서 잡음 등에 의해 오류가 덧붙인 것을 확인 가능
- 패리티 체크나 CRC는 전송 중 에러 발생 여부를 판단할 수 있을 뿐 에러를 정정할 수는 없음
- 에러 발견 후 정정이 가능한 채널 부호화 방법도 존재 (해밍 코드, 리드-솔로몬 코드 등)

03. 통신의 과정

■ 부호화

■ 라인 부호화

- 직류 성분: 신호가 변화 없이 일정한 높이의 전압을 가진 채 통신선로를 따라 흐르는 것
- 직류 성분의 신호는 대부분의 시스템이 정상적으로 처리하지 못하고 왜곡된 값으로 해석
- 이를 해결하기 위해서는 라인 부호화가 필요
- 대표적인 방식은 일정한 길이 단위로 비트열을 구분한 후 완전히 새로운 구조의 비트열로 대체

표 3-2 4B5B 부호화

16진수	이진수	부호화	16진수	이진수	부호화
0	0000	11110	8	1000	10010
1	0001	01001	9	1001	10011
2	0010	10100	A	1010	10110
3	0011	10101	B	1011	10111
4	0100	01010	C	1100	11010
5	0101	01011	D	1101	11011
6	0110	01110	E	1110	11100
7	0111	01111	F	1111	11101

03. 통신의 과정

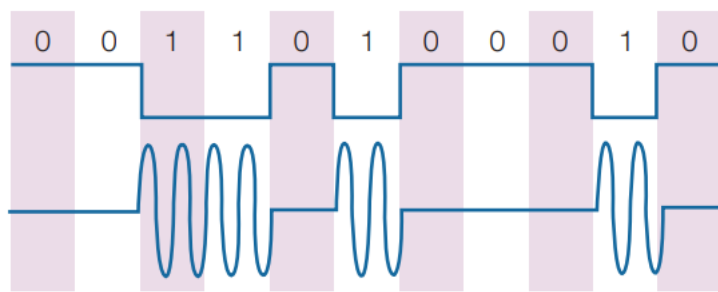
■ 변조와 복조

■ 디지털 변조

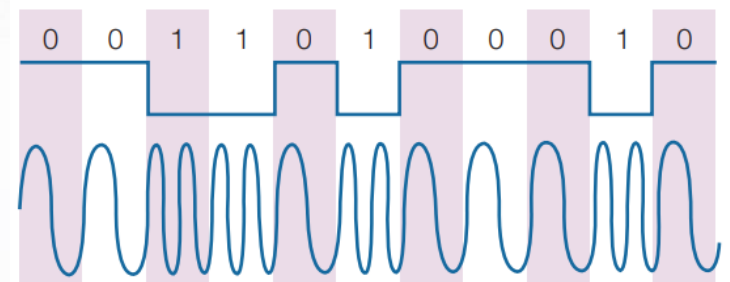
- 변조: 음성 신호를 전기 신호로 변경하는 과정
- 복조: 변조된 신호로부터 원래의 정보 신호를 추출하는 과정
- 오늘날에는 디지털 변조 방식이 대다수를 차지

• 진폭 편이 변조와 주파수 편이 변조

- 진폭 편이 변조 (ASK): 신호 진폭의 크기를 조절해 디지털 신호의 0과 1로 전송하는 방식
- 진폭이 큰 상태의 신호가 일정 시간 지속되면 1, 진폭이 0인 상태로 일정 시간 지속되면 0이 됨
- 주파수 편이 변조 (FSK): 두 개의 다른 주파수를 이용해 0과 1로 나타내는 방식
- 예를 들어 높은 주파수가 일정 시간 지속되면 1, 낮은 주파수가 일정 시간 지속되면 0으로 나타냄



(a) 진폭 편이 변조



(b) 주파수 편이 변조

그림 3-12 진폭 편이 변조와 주파수 편이 변조

03. 통신의 과정

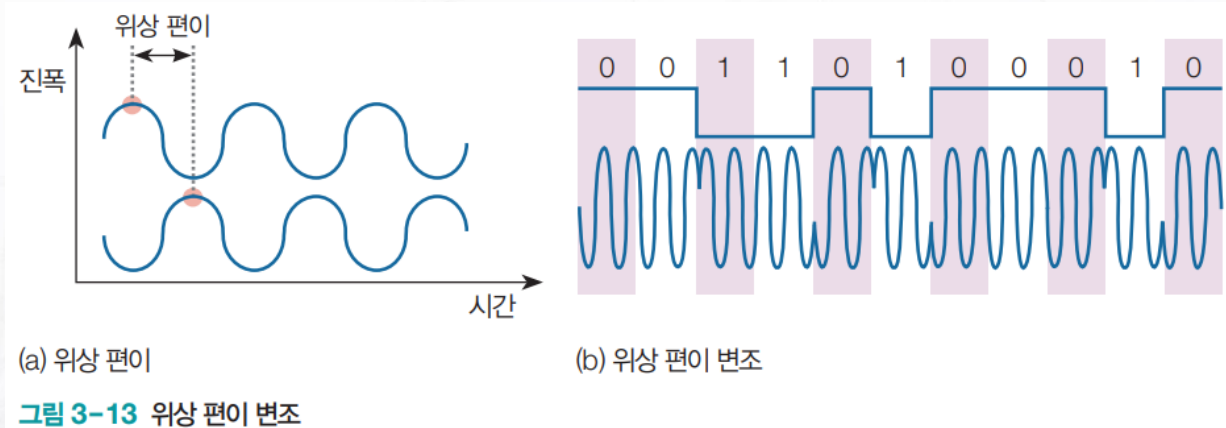
■ 변조와 복조

■ 위상 편이 변조

- 위상 편이 변조 (PSK): 신호의 위상을 바꿈으로써 0과 1로 신호를 표시하는 방식
- 전송하는 데이터가 0에서 1로 바뀌거나 1에서 0으로 바뀔 때 신호의 위상을 180도 바꾸어서 나타냄

■ 직교 진폭 변조

- 직교 진폭 변조 (QAM): 위상 편이 변조와 진폭 편이 변조를 결합한 방식으로 반송파의 크기와 위상을 동시에 바꿔서 정보를 표현
- 신호의 크기와 위상을 얼마나 세세하게 나누는가에 따라 QAM의 차수가 달라짐
- QAM의 차수가 높아질수록 속도가 더 빠름



03. 통신의 과정

■ 다중화와 다중접속

■ 다중화

- 여러 신호를 합쳐서 하나의 신호로 만든 뒤 하나의 통신매체를 통해서 전송하는 방식
- 다중화는 지역을 잇는 통신망뿐만 아니라 방송에도 사용

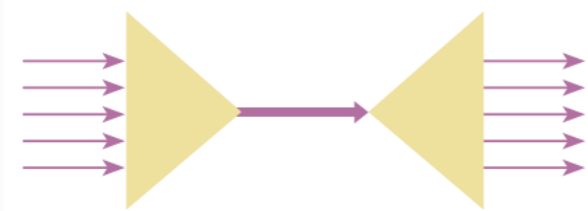


그림 3-14 다중화

• 주파수 분할 다중화 (FDM)

- 가장 쉽게 생각할 수 있는 다중화 방식 / 도로 위의 차선
- TV나 라디오 방송처럼 통신에서도 하나의 통신매체에 사용되는 주파수를 여러 개로 구분해 여러 신호를 동시에 전송

• 시분할 다중화 (TDM)

- 시간을 나누어 여러 사람의 신호를 전송
- 엄밀히 말해 여러 신호를 동시에 전송하는 것이 아니라 순차적으로 전송
- 빠른 속도로 시분할 처리를 해서 여러 신호를 전송할 수 있는 통신장비가 있어야 하기 때문에 기술적 난이도가 높음

03. 통신의 과정

■ 다중화와 다중접속

■ 다중접속

- 여러 통신기기가 공유하는 통신매체를 누가 언제 사용하도록 할 것인지 결정하는 알고리즘
- 다중접속 기법에는 랜덤 액세스, 컨트롤 액세스, 채널화 프로토콜이 있음

• 컨트롤 액세스

- 네트워크에 연결된 기기들은 전송 중인 데이터 간의 충돌을 피하기 위해 데이터 전송 전에 컨트롤 액세스를 사용
- 예약: 데이터 전송이 가능한 일정 구간의 시간을 의미하는 타임슬롯을 이용하는 방식
- 폴링: 네트워크 주판자가 네트워크 기기들에게 전송할 데이터가 있는지 물어보고, 만약 있다면 데이터를 전송할 기회를 주는 방식
- 토큰 돌리기: 네트워크 기기들이 순차적으로 토큰을 넘겨받으면서 전송할 데이터가 있으면 전송하고, 그렇지 않으면 이웃한 다음 네트워크 기기에 토큰을 넘기는 방식

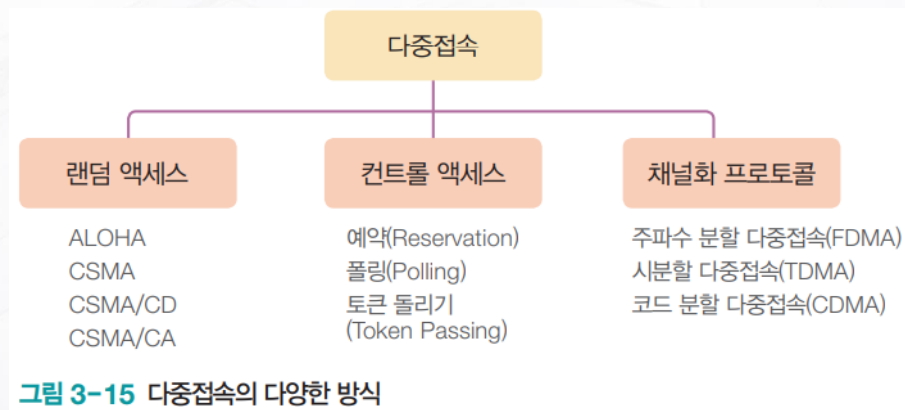


그림 3-15 다중접속의 다양한 방식

03. 통신의 과정

■ 다중화와 다중접속

■ 다중접속

• 채널화 프로토콜

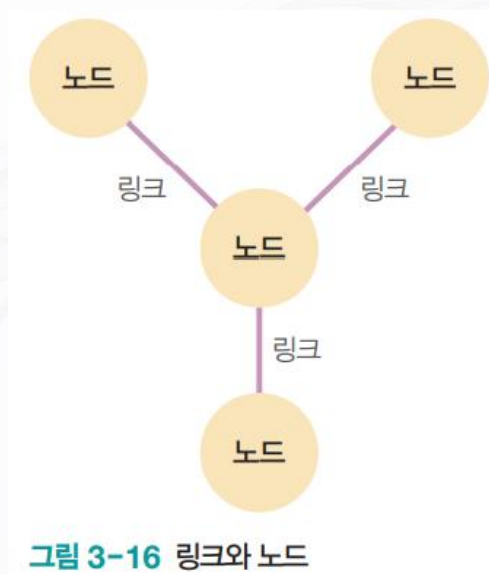
- 다중화 기법인 주파수 분할 다중화와 시분할 다중화는 다중접속에서 주파수 분할 다중접속과 시분할 다중접속으로 사용
- 주파수 분할 다중접속 (FDMA) 1세대 방식 / 시분할 다중접속 (TDMA) 2세대 방식
- 코드 분할 다중접속 (CDMA)는 3세대 방식
- 주파수, 타임슬롯과 무관하게 모든 신호가 혼합되어 모든 신호를 수신
- 그 후 전체 신호에서 내가 받아야 할 신호만 골라서 뽑아냄
- 직교 주파수 분할 다중접속 (OFDMA)은 4세대부터 표준으로 채용
- 특정한 상황에서 여러 개의 신호를 겹쳐 놓더라도 서로 간의 신호 에 간섭 현상이 생기지 않는 것을 활용
- 여러 개의 신호를 동시에 사용함으로써 신호 간의 충돌 없이도 데이터 전송속도를 높일 수 있음

04. 통신의 확장, 네트워크

■ 네트워크란

■ 네트워크의 개념

- 무엇인가를 연결하는것 / '무엇'=노드, '연결'=링크
- 물리적 관점: 노드는 통신 관련 기기를 의미 / 링크는 노드 간의 연결, 즉 통신매체를 의미
- 논리적 관점: 노드는 연결의 대상이 될 수 있는 각종 애플리케이션을 포함 / 링크는 다양한 프로토콜에 기반한 연결을 포함
- 최초의 네트워크는 노드와 노드가 일대일로 연결되는 모습
- 네트워크는 통신회선을 공유하여 서로 연결되므로 우편에서 사용하는 것과 같은 주소가 필요



04. 통신의 확장, 네트워크

■ 네트워크란

■ 데이터통신 방식에 따른 네트워크 분류

• 유니캐스트

- 송신 노드와 수신 노드가 각각 하나
- 송신 노드는 수신 노드의 주소를 지정하여 다른 노드가 메시지를 받는 일이 없음

• 멀티캐스트

- 송신 노드는 하나이고 수신 노드는 여러이며 이때 수신 노드는 특정한 그룹에 속해 있음

• 브로드캐스트

- 송신 노드는 하나이고 수신 노드는 네트워크에 있는 모든 노드
- 라디오나 TV방송처럼 신호를 수신할 수 있으면 누구나 메시지를 받을 수 있음

04. 통신의 확장, 네트워크

■ 네트워크란

■ 영역에 따른 네트워크 분류

• 개인 통신망(PAN)

- 개인이 작업 중인 컴퓨터 또는 스마트폰과 일대일로 연결되는 기기로 구성되는 네트워크
- 컴퓨터는 USB 통신으로 거의 모든 주변기기 연결이 통일되었으며 가끔 블루투스가 보조적으로 사용
- 스마트폰은 블루투스를 통해 주변기기와 연결되며 가끔 USB가 보조적으로 사용

• 홈 네트워크(HAN)

- 가정 내 다양한 정보기기 간에 네트워크를 구축
- 와이파이기능이 포함된 가전기기, 사물인터넷, 각종 센서기기가 홈 네트워크에 포함
- 센서기기는 저전력 무선통신을 사용하기 때문에 지그비, 지웨이브등의 통신 방식이 주로 사용

• 근거리 통신망(LAN)

- 집, 사무실, 학교 등의 건물 내에서 인접한 컴퓨터를 연결하는 네트워크 / 대표적인 것이 와이파이 네트워크
- 과거 와이파이가 보편화되기 전에는 이더넷이 근거리 통신망의 주요 구축 방법

04. 통신의 확장, 네트워크

■ 네트워크란

■ 영역에 따른 네트워크 분류

• 도시권 통신망(MAN)

- 도시에 있는 컴퓨터를 연결하는 네트워크
- 실제로는 컴퓨터를 직접 연결하는 것이라기보다는 이미 구축된 근거리 통신망을 서로 연결 하는 것
- 기본적으로 사용하는 하드웨어와 소프트웨어는 근거리 통신망과 비슷

• 광역 통신망 / 원거리 통신망(WAN)

- 근거리 통신망이나 도시권 통신망을 서로 연결
- 아주 먼 거리에 초고속으로 신호를 전송해야 하므로 고가의 특별한 장비를 사용 (예: 대륙을 잇는 해저 케이블)
- 인터넷이 대표적인 예

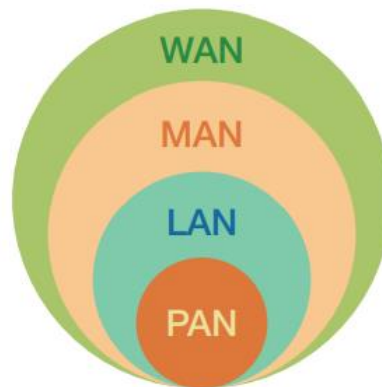


그림 3-17 네트워크의 범위

04. 통신의 확장, 네트워크

■ 네트워크란

■ 회선 교환망과 패킷 교환망

• 회선 교환망

- 주로 전화망에서 사용
- 개별 전화기는 교환기를 통해서 전화망에 접속하고 여러 개의 교환기를 거쳐서 상대방 전화기와 연결
- 예) 서울에 사는 A가 부산에 사는 B에게 전화를 하는 경우
 - » A가 B의 전화번호를 누르는 순간 A의 전화기에서 출발한 신호는 교환기와 교환기 간의 연결 통로를 만듦
 - » B의 "여보세요"라는 음성은 직전에 완성된 연결 통로를 동일하게 사용하면서 A에게 전달
 - » 이 연결 통로는 둘 사이의 통화가 끝날 때까지 유지
 - » 마치 A와 B 사이에 전용선이 구축되어 있는 것처럼 독점적으로 사용

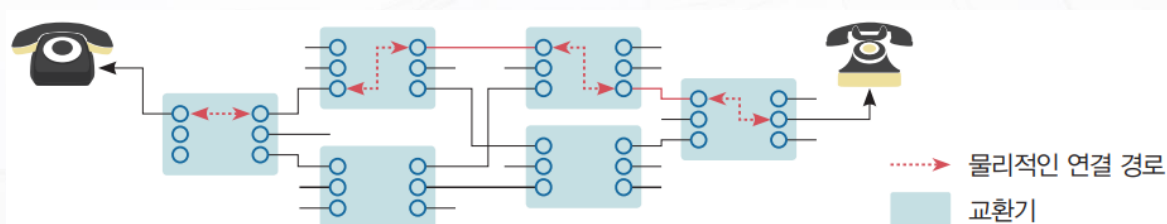


그림 3-18 회선 교환망

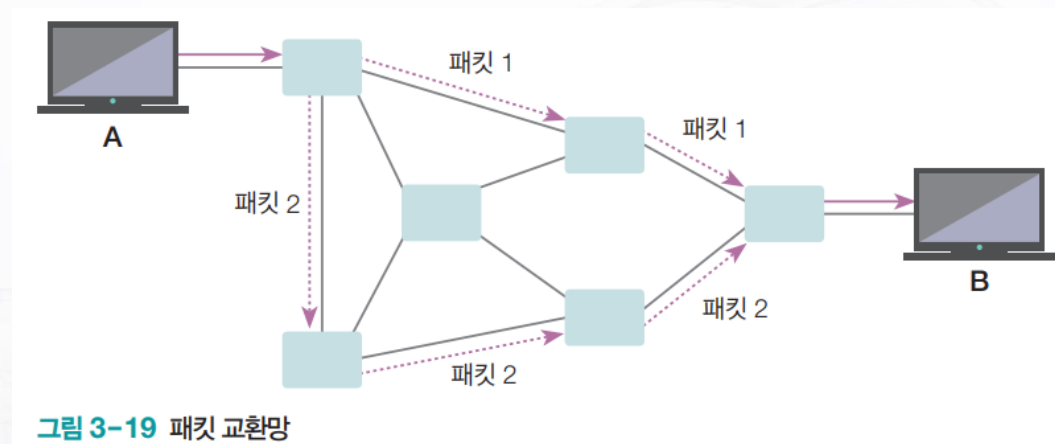
04. 통신의 확장, 네트워크

■ 네트워크란

■ 회선 교환망과 패킷 교환망

• 패킷 교환망

- 주로 인터넷망에서 사용
- 회선 교환망과 달리 연결 통로가 생성되지 않으며 패킷은 라우터가 그 때의 최적의 경로를 찾아서 전송
- 첫 번째 패킷의 경로와 두 번째 패킷의 경로가 다를 수 있음
- A가 보낸 패킷에 대해 B가 회신한 패킷은 A가 보낸 패킷이 거친 길과 동일하다는 보장이 없음



04. 통신의 확장, 네트워크

■ 네트워크의 구성 방식

■ 토폴로지

- 네트워크의 요소들을 물리적으로 연결한 것, 또는 그 연결 방식
- 컴퓨터가 네트워크를 형성하기 시작하면서 부각된 개념

■ 토폴로지의 종류

• 스타 토폴로지

- 중앙에 교환기 역할을 하는 장비가 있고 그 주위로 컴퓨터가 연결되는 방식
- 컴퓨터는 교환기를 통해서 이웃한 기기와 통신
- 유·무선 공유기가 형성하는 네트워크

• 트리 토폴로지

- 최하단에 컴퓨터, 그 위에 교환기 역할을 하는 장비, 그 상위에는 교환기가 위치 (계층구조)
- 전화망

• 버스 토폴로지

- 교환기 역할의 장비가 없고 컴퓨터들이 하나의 케이블로 연결
- 간선 역할의 케이블과 지선 역할의 케이블로 이루어지며, 컴퓨터는 지선의 끝에 위치

04. 통신의 확장, 네트워크

■ 네트워크의 구성 방식

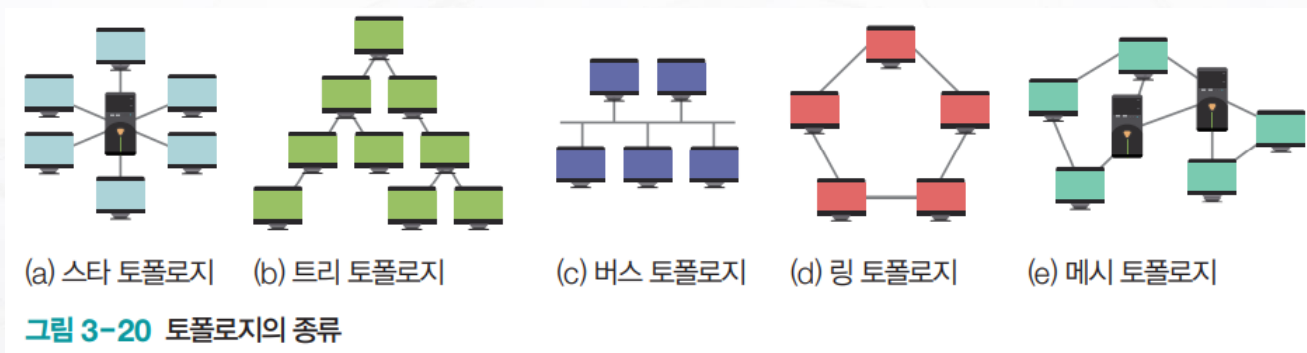
■ 토폴로지의 종류

• 링 토폴로지

- 컴퓨터들이 케이블로 연결돼 전체 모습이 원형을 이룸
- 네트워크 초창기에 사용됐으나 이더넷의 버스 토폴로지에 밀려 거의 사라짐

• 메시 토폴로지

- 메시 토폴로지는 일정한 형태가 없고 통신기기들이 그물망처럼 서로 연결된 형태
- 유선 네트워크에서는 케이블 연결이 제한적이라 구축이 어렵지만, 무선 네트워크에서는 비교적 수월
- 사물인터넷에 사용되는 지그비, 지웨이브와 같은 무선 네트워크는 메시 토폴로지를 지원



- 01. 인터넷의 구조
- 02. 인터넷 주소
- 03. 인터넷 프로토콜(IP)과 TCP
- 04. 인터넷 서비스와 WWW

01. 인터넷의 구조

■ 인터넷의 개념

- 인터넷의 정의

- 한마디로 정의하기 어려움 / 'Inter'와 'net'의 의미를 분석해보면 여러 종류의 네트워크가 서로 연결된 것
- 패킷 네트워크와 TCP/IP라는 두 가지 중요한 요소로 정의

- 패킷 네트워크

- 전송하고자 하는 메시지를 작은 조각으로 나누고 조각마다 송신지와 수신지 주소를 기입한 후 조각들이 독립적으로 네트워크를 이동하도록 하는 방식

- TCP/IP 프로토콜 구현

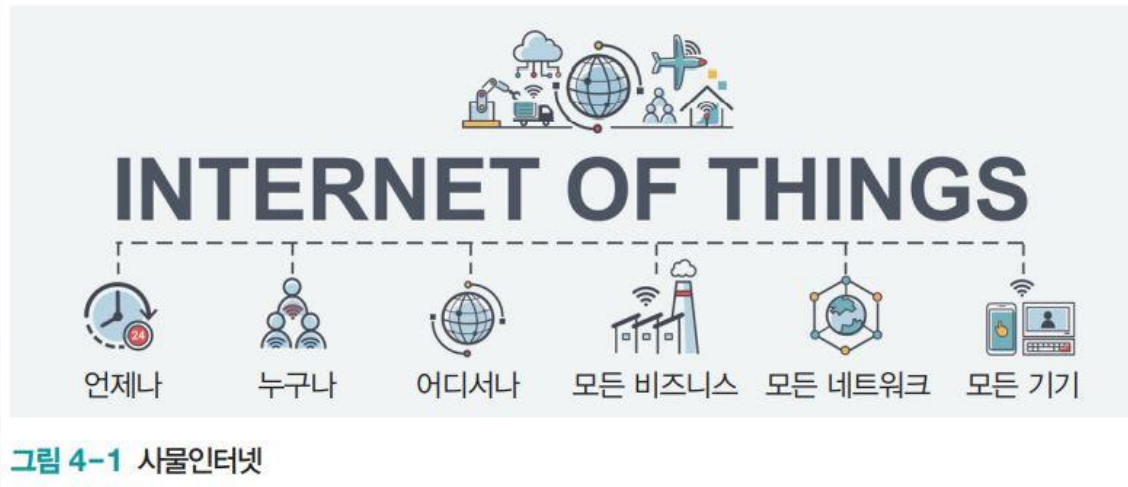
- 인터넷에 연결하려면 필수적 / 이 프로토콜이 없다면 물리적으로 연결돼 있더라도 통신이 불가능
- TCP/IP만 구현하고 있다면 하드웨어나 운영체제의 종류에 상관없이 어떤 통신 방식이라도 인터넷에 연결될 수 있음

01. 인터넷의 구조

■ 인터넷의 개념

■ 사물인터넷

- 어떤 제품에 통신 기능을 추가하는 것은 간단한 일이 아님
- 인터넷 초창기와 성숙기에 통신 기능을 갖출 수 있는 제품은 컴퓨터가 거의 유일
- 통신 부품의 가격이 낮아졌을 뿐만 아니라 네트워크 연결 방법이 쉬워짐에 따라 여러 기기가 인터넷에 연결
- 사물인터넷(IOT): 컴퓨터로만 구성되던 과거의 인터넷과 구별하기 위해 오늘날의 인터넷을 지칭하는 말



01. 인터넷의 구조

■ 인터넷의 개념

■ 서버와 클라이언트

• 서버

- 웹, 이메일, FTP 등 인터넷 서비스를 제공하는 컴퓨터
- 통상 서버룸 또는 서버팜이라는 곳에 위치하여 일반인이 쉽게 접근할 수 없는 곳에서 보호
- 클라이언트의 요청을 받고 필요한 서비스를 제공

• 클라이언트

- 사용자들이 키보드, 터치스크린 등의 입력장치를 이용해서 서버에 해당 인터넷 서비스를 요청하는 컴퓨터
- 개인용 컴퓨터, 스마트 폰, 스마트 안경, 자동차, 센서가 부착된 신발 등등...
- 서버나 클라이언트는 컴퓨터의 하드웨어 종류나 보관 위치보다는 소프트웨어에 의해 구별
- 예: 스마트폰에 웹 서버 소프트웨어를 설치하면 해당 스마트폰은 서버
- 이 경우 스마트폰은 서버와 클라이언트의 역할을 동시에 수행

■ 피어-투-피어(P2P)

- 연결된 모든 컴퓨터가 서버이자 클라이언트 역할을 동시에 수행
- 예: 비트토렌트, 블록체인
- 효율적인 동작을 위해 중간에 서버를 두는 경우도 있음

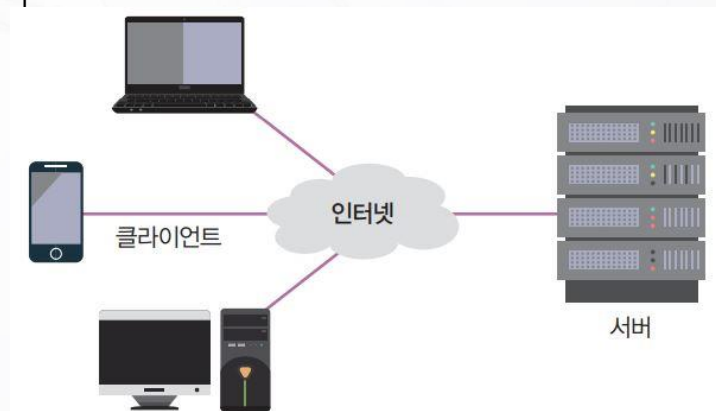


그림 4-2 서버-클라이언트 모델

01. 인터넷의 구조

■ 인터넷 인프라

■ 네트워크와 네트워크의 연결

- 외부로 나가는 데이터든 내부로 들어오는 데이터든 반드시 스위치를 경유
- 스위치는 내부 네트워크에 연결된 컴퓨터로부터 패킷이 오면 패킷의 헤더에서 주소를 읽음
- 패킷의 전송이 네트워크 내부의 컴퓨터, 외부의 컴퓨터 중 어디로 전송되는지 판단하고 결과에 따라 패킷을 전송

■ 인터넷 교환 노드

- 네트워크는 스위치를 통해 계층 구조를 이루면서 피라미드 구조가 됨
- 피라미드 하부에는 지역 기반의 네트워크 또는 기업이나 연구소 단위의 네트워크가 위치
- 피라미드 상부의 네트워크 연결은 서로 다른 조직에 속하므로 일관성 있는 설계가 어려움
- 초기에는 각 기관들이 필요에 의해 일대일로 네트워크를 연결
- 인터넷 교환 노드 (IX): 무수히 많은 스위치와 라우터가 모여 있어서 네트워크 간의 연결이 용이해짐

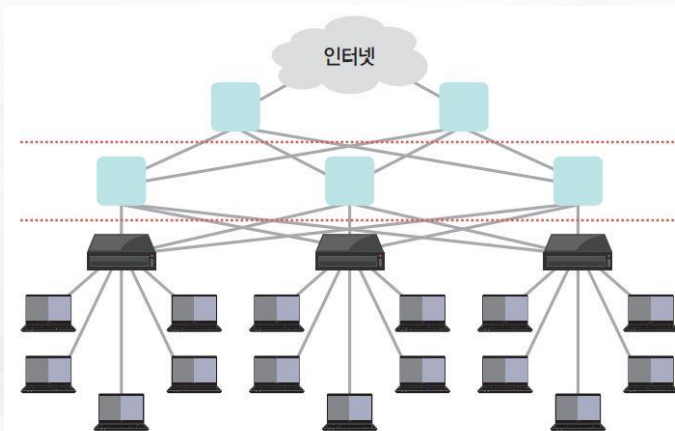


그림 4-3 네트워크의 계층 구조

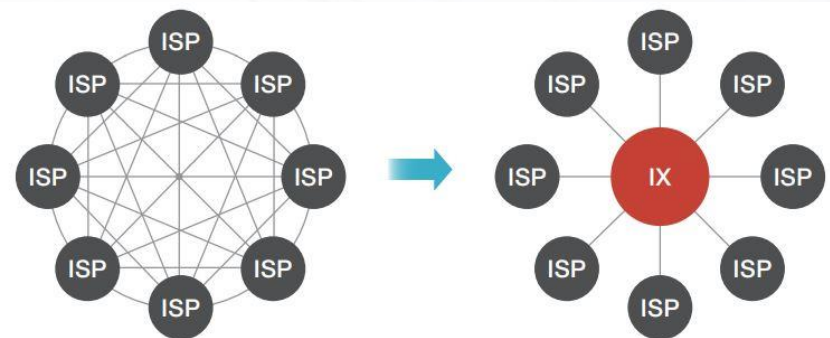


그림 4-4 인터넷 교환 노드 사용의 효과

01. 인터넷의 구조

■ 인터넷 인프라

■ 국내외의 인터넷 교환 노드

- 현재 국내에서 운영 중인 인터넷 교환 노드에는 KTX(KT), DIX(유플러스), SKBIX(SK 브로드밴드), KINX
- 차세대 인터넷 주소인 IPv6 주소 기반 연동망인 6NGIX가 비영리로 운영
- 미국에서는 1990년대에 최초의 상업용 인터넷 교환 노드가 캘리포니아 지역에 구축
- 현재는 미국 최대 IX 사업자인 에퀴닉스가 세계에 19개의 인터넷 교환 노드를 운영

• 인터넷 백본망

- 인터넷 백본망: 인터넷 서비스업체를 서로 연결하는 초고속 네트워크
- KT나 U+, SKB 처럼 전국망을 구축한 통신업체들은 내부에 백본망이 있어 지역과 지역을 연결
- 인터넷 사용자 입장에서는 이더넷이나 와이파이 또는 LTE의 존재는 알 수 있어도 백본망의 존재는 알기 어려움

• 인터넷 데이터센터

- 인터넷 데이터센터: 서버 컴퓨터가 엄청난 규모의 데이터 저장장치, 초고속의 대용량 통신선과 연결된 시설
- 데이터센터는 전 세계 수십억 명의 인터넷 사용자 요청에 대해 서비스를 제공



그림 4-5 데이터센터의 내부 모습

01. 인터넷의 구조

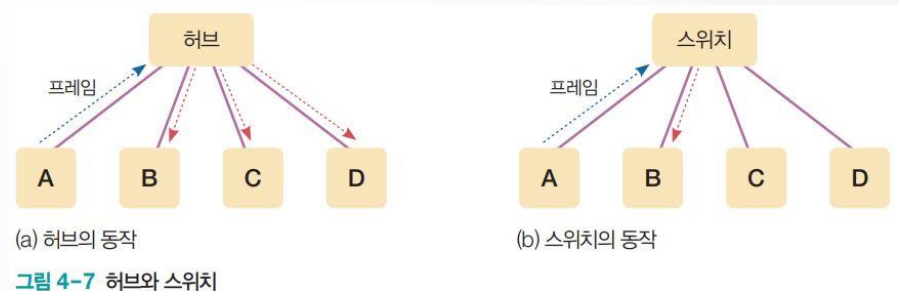
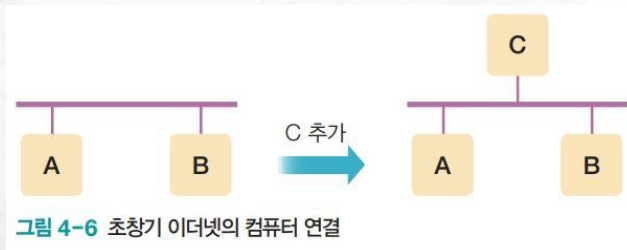
■ 인터넷 연결 장비

■ 허브

- 여러 대의 컴퓨터를 간단하게 연결하는 장비 / 허브 발명 이전 컴퓨터를 연결하는 방법은 케이블로 직접 연결하는 것
- 포트 개수가 많을수록 많은 컴퓨터를 네트워크에 연결할 수 있지만 연결된 컴퓨터가 많을수록 네트워크 성능은 낮음
- 허브는 리피터(신호 증폭기) 역할을 담당

■ 스위치(스위칭 허브)

- 스위치는 자신에게 연결된 컴퓨터의 물리적 네트워크 주소와 포트 번호를 테이블로 가지고 있음
- 이더넷 프레임이 오면 프레임의 헤더에서 목적지 컴퓨터의 물리적 네트워크 주소를 읽음
- 물리적 포트 번호를 찾고 그 포트에 연결된 케이블을 통해 신호를 내보냄
- 이더넷 프레임을 보내는 컴퓨터의 케이블과 이를 받는 컴퓨터의 케이블에만 신호가 흐르기 때문에 다른 컴퓨터는 언제라도 다른 통신을 시작할 수 있음
- 허브는 전송된 프레임의 헤더를 읽지 않고 연결된 모든 컴퓨터에 그대로 재전송



01. 인터넷의 구조

■ 인터넷 연결 장비

■ 스위치(스위칭 허브)

• 주소 변환 프로토콜(ARP)

- 컴퓨터가 이더넷 프레임을 주고받으려면 상대방 컴퓨터의 이더넷 주소를 알아야 하지만 최초의 통신에서는 인터넷 주소만 알 수 있음
- 송신하고자 하는 컴퓨터는 상대방 컴퓨터의 인터넷 주소를 담은 패킷을 만듦
- 수신해야 할 컴퓨터의 이더넷 주소 자리에 FF:FF:FF:FF:FF:FF를 기입해서 스위치에 보냄
- 수신한 스위치는 연결된 모든 컴퓨터에 이 이더넷 프레임을 전송
- 이 패킷을 받은 컴퓨터 중에서 자신의 인터넷 주소를 발견한 컴퓨터는 자신의 이더넷 주소를 적어 되돌려보냄
- 최초로 통신을 시작한 컴퓨터의 이더넷 주소가 송신 컴퓨터의 주소란에 기입되어 있으므로 그 주소를 이용

■ 라우터

- 둘 혹은 그 이상의 네트워크와 네트워크를 연결할 때 사용
- 인터넷이 네트워크의 네트워크이므로 라우터는 인터넷의 가장 핵심적인 장비
- 자신에게 전송된 인터넷 패킷의 헤더에서 목적지 컴퓨터의 인터넷 주소를 읽고 어느 물리적 포트로 패킷을 내보내야 할지 결정
- 라우터에 연결된 컴퓨터나 네트워크가 최종 목적지일 수도 있지만 다른 라우터로 전송해야 하는 경우도 있음
- 라우터는 갖고 있는 경로 정보를 서로 교환하며 최적의 패킷 경로를 결정

01. 인터넷의 구조

■ 인터넷 연결 장비

■ 공유기

- 공식 기관에 의해 할당된 인터넷 주소를 공유
- 공유기의 가장 핵심적인 기능은 네트워크 주소 변환(NAT)
- 인터넷에서는 오로지 공인 인터넷 주소만 통용되지만 공유기 내부의 컴퓨터는 사설 인터넷 주소를 할당받음
- 사설 인터넷 주소를 공인 인터넷 주소로 바꾸어주는 네트워크 주소 변환 기능을 공유기가 담당



네트워크 주소 변환		
사설 IP 주소: 포트	공인 IP 주소: 포트	서버 공인 IP 주소: 포트
192.168.1.1:4567	101.89.101.12:6754	68.1.31.1:80
192.168.1.2:5934	101.89.101.12:3568	68.1.31.1:80
192.168.1.3:9867	101.89.101.12:7745	68.1.31.1:80

그림 4-8 네트워크 주소 변환

01. 인터넷의 구조

■ 이더넷

■ 이더넷과 인터넷

- 최초의 인터넷이라고 할 수 있는 아파넷은 1969년에 연결되었고 이더넷 첫 규격은 1983년에서 등장
- 이더넷은 인터넷에서 사용하는 패킷 네트워크 기술을 함께 사용
- 이더넷은 오늘날 프린터와 복사기로 유명한 제록스 파크 연구소에서 발명
- 알파넷은 레이저 프린터를 다수의 컴퓨터와 연결하기 위해 패킷 네트워크 기술을 연구한 결과물
- 가정 및 일반 사무실에서는 기가비트 이더넷이 보편적으로 사용

■ 이더넷 통신 방식 - CSMA/CD

- 동시에 통신을 할 때 발생하는 충돌을 막기 위해 CSMA/CD를 사용
- 이더넷과 같은 버스 시스템에서 데이터를 전송하기 전에 버스 시스템 상에 전송 중인 신호가 있는지 확인하는 CSMA
- 버스 시스템 상에서 신호의 충돌을 감지하는 CD
- 와이파이의 충돌 감지가 어렵기 때문에 CD 대신 신호의 충돌을 피하기 위한 대책인 CA를 규정

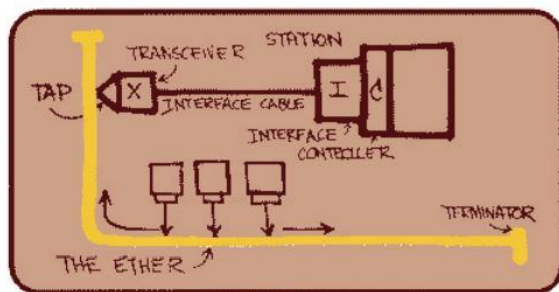


그림 4-9 이더넷을 설명하는 로버트 맥칼프의 스케치

01. 인터넷의 구조

■ 이더넷

■ 이더넷 통신 방식 - CSMA/CD

• CSMA

- CSMA는 데이터를 보내기 전에 네트워크에 전송 중인 신호를 감지하고 신호가 없다면 바로 데이터 전송을 시작
- 만약 전송 중인 신호가 있다면 무작위의 시간 동안 기다리도록 규정
- 여러 단말이 데이터를 전송하기 위해 네트워크 상태를 감지하고 있을 수 있기 때문
- 무작위의 시간 동안 대기함으로써 자연적으로 단말들 간에 순서가 정해짐
- 무작위에 의해 가장 짧은 시간을 할당받은 단말이 데이터를 전송할 확률이 제일 높음

• CD

- CSMA만으로는 데이터의 충돌을 100% 피할 수 없음
- 트랜시버만 믿고 데이터를 전송하면 네트워크 중간에서 신호의 충돌이 발생
- 트랜시버가 충돌을 감지하면 단말은 데이터 충돌이 발생했음을 네트워크 상의 모든 컴퓨터에 알리기 위해 재밍신호를 전송하고 무작위 시간만큼 데이터 전송을 중지
- 그 후 네트워크 상태를 확인하고 전송 중인 데이터가 없는 것을 확인하면 자신의 데이터를 전송
- 만약 이때도 충돌이 발생한다면 이번에는 지난번 기다렸던 시간의 두 배를 기다림(이진 지수 백오프)
- 16회까지만 반복하고 16회 때도 데이터 충돌이 감지되면 최종적으로 데이터 전송 실패 처리를 실시(CD)

02. 인터넷 주소

■ 인터넷 프로토콜(IP) 주소

- 인터넷 프로토콜 주소의 개념

- 컴퓨터의 고유한 주소
- 전 세계 IP 주소 자원의 총괄 관리 기관인 IANA에서 관리
- 각 대륙별로 인터넷 주소 자원 관리 기관인 RIR를 두어 세부 주소를 분배하고 관리
- IANA는 일정한 범위의 IP 주소를 기업이나 정부, 인터넷 서비스 제공업체에 할당
- 1993년에 클래스의 제한을 두지 않고 필요한 호스트의 수에 따라 적당한 크기의 IP 주소를 할당하는 CIDR방식 도입

■ 공인 IP 주소와 사설 IP 주소

- IP 주소는 A, B, C 클래스 단위로 할당되기 때문에 주소가 부족하거나 남아도는 문제가 발생
- 문제를 해결하기 위해 6바이트로 구성되는 인터넷 주소체계인 IPv6가 만들어져 사용 중
- IP 주소가 부족함에도 여전히 인터넷이 동작하는 이유 중의 하나는 공인 IP 주소와 사설 IP 주소를 분리해 사용하기 때문
- 공인 IP 주소는 전 세계에서 유일한 인터넷 주소로 라우터는 공인 IP 주소를 참조해 패킷을 어디로 보낼지 결정
- 사설 IP 주소는 기업이나 단체에서 내부용으로 사용하는 주소
- IANA에서는 사설 IP 주소로 사용할 수 있는 주소 범위를 정함
- 사설 IP 주소를 사용하는 대표적인 경우는 공유기로 구성되는 네트워크

02. 인터넷 주소

■ 도메인 네임

- 도메인 네임(도메인 주소)

- 사람들이 이해하고 기억하기 쉽게 만든 주소 / 구글 웹서버에 접속 할 경우 172.217.27.67 대신 www.google.co.kr 입력

- ❶ 최상위 도메인(TLD) / 일반 최상위 도메인 (gTLD)와 국가코드 최상위 도메인(ccTLD)이 있음
공공 조직: .org / 영리 조직: .com / 교육 기관: .edu / 정부: .gov / 네트워크 관리용: .net
- ❷ 명확히 규정되어 있지는 않지만 대부분의 경우 조직의 이름
- ❸ 호스트 이름을 나타내기도 하지만 일정하지는 않음

최상단의 '.'은 en.wikipedia.org가 절대 주소임을 의미(완전 표기 도메인 네임)

인터넷 사용자가 웹 브라우저에 주소를 입력할 때 맨 뒤의 '.'를 붙이지 않더라도 웹 브라우저가 자동으로 이를 처리

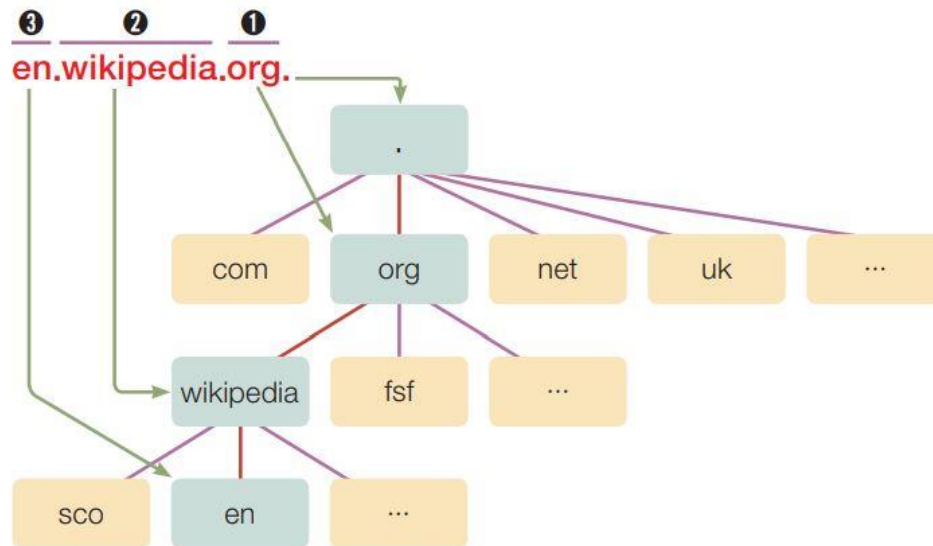


그림 4-10 도메인 네임의 계층 구조

02. 인터넷 주소

■ 도메인 네임

■ 도메인 네임 서버(DNS)

- 문자로 구성된 도메인 네임과 숫자로 구성된 IP 주소를 연결해주는 소프트웨어
- 인터넷에 항상 연결되어 동 작하면서 웹 브라우저로부터 주소 변환 요청이 있으면 응답
- 웹 브라우저와 같은 인터넷 클라이언트 소프트웨어는 도메인 네임 서버의 주소를 사전에 알고 있어야 함
- 기기들이 공유기에 연결되는 동안 자신의 IP 주소를 자동으로 할당받을 때 도메인 네임 서버의 주소 정보를 함께 전송받음

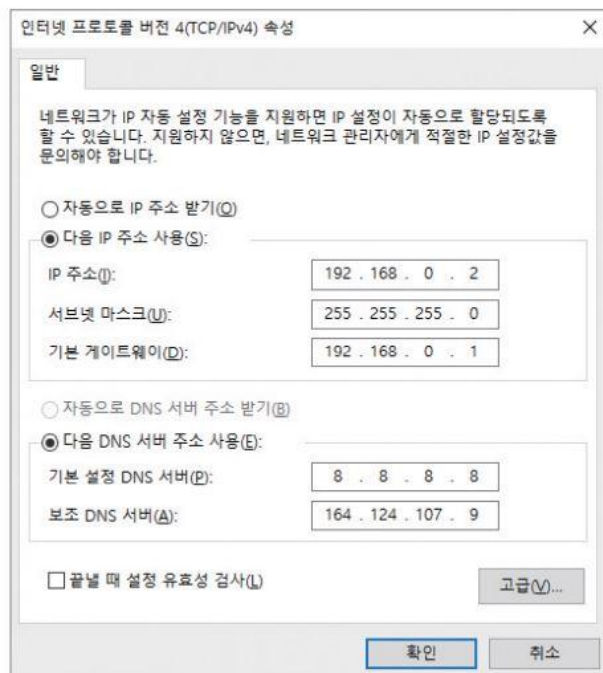


그림 4-11 DNS 주소의 수동 설정

03. 인터넷 프로토콜(IP)과 TCP

■ 인터넷 프로토콜(IP)

■ 인터넷 프로토콜

- 인터넷 초창기에는 컴퓨터가 여러 사용자의 터미널을 지원했기 때문에 컴퓨터를 호스트라고 부름
- 원격에서 호스트 컴퓨터를 사용하기 위해서는 통신의 발전이 필요
- 인터넷은 컴퓨터와 통신의 발전으로 인해 탄생했고 패킷 네트워크가 동작에 성공하면서 완성

■ 패킷

- 프로토콜: 통신에 참여하는 모든 기기가 준수해야 하는 약속
- 인터넷을 돌아다니는 패킷은 인터넷 프로토콜을 준수하기 때문에 구조가 모두 동일
- 헤더와 데이터로 구성되며 헤더에 들어가는 정보는 엄격하게 위치와 길이가 규정
- 인터넷 패킷의 헤더는 통상 20바이트로 구성되고, 이어서 데이터가 전송
- 데이터는 인터넷 프로토콜 계층의 상위에 있는 프로토콜 계층에서 작성
- 예: 전송 제어 프로토콜 (TCP), 사용자 데이터그램 프로토콜 (UDP)

03. 인터넷 프로토콜(IP)과 TCP

■ TCP

■ TCP의 개념

- 신뢰성 높은 통신을 위해 만들어진 전송 제어 프로토콜
- 애플리케이션이 사용하는 프로토콜은 인터넷 프로토콜과 달리 신뢰성이 높아야 함
- 전송 도중에 사라지면 재전송을 요청할 수 있거나 패킷의 순서가 뒤바뀌어 도착한다면 순서를 바로 잡아야하기 때문

■ 세그먼트

- 전송 제어 프로토콜에 따라 작성된 메시지 / 인터넷 프로토콜에 따라 작성된 메시지를 부르는 패킷과 구분
- 세그먼트도 패킷과 동일하게 헤더와 데이터로 구성
- 세그먼트의 헤더에 기입되는 포트 번호는 동일한 컴퓨터 내에 실행 중인 애플리케이션들을 구분하기 위함
- 패킷을 수신한 서버 컴퓨터에 웹 서버(HTTP), 파일 전송 서버(FTP), 이메일 서버 가 실행 중인 경우 어느 애플리케이션이 세그먼트를 받아야 할지 결정하기위해 각 애플리케이션에 포트 번호를 할당
- 서비스를 요청하는 원격 컴퓨터는 서버의 IP 주소와 함께 포트 번호를 지정
- TCP를 기반으로 다양한 애플리케이션이 개발되었고 각 애플리케이션을 위한 별도의 프로토콜이 정의
- 암호화되지 않은 텍스트 통신을 위한 텔넷, 암호화 통신 프로토콜인 SSH, 메일 전송을 위한 SMTP 등

03. 인터넷 프로토콜(IP)과 TCP

■ TCP

■ 캡슐화

- TCP 세그먼트는 앞에 인터넷 프로토콜의 헤더가 붙어 있고 그 앞에는 이더넷의 헤더가 붙어있음
- 캡슐화: 프로토콜 계층 구조의 상위에서 구성된 데이터 패킷이 하위 계층으로 내려가 새로운 헤더가 붙고 다시 하위 계층으로 내려가 새로운 헤더가 붙는 구조
- 메시지를 보내는 컴퓨터에서는 계층 구조를 따라 물리 계층을 향해 내려가면서 헤더가 덧붙여짐
- 메시지를 받는 컴퓨터에서는 물리 계층에 해당하는 헤더부터 순차적으로 분리하며 상위의 응용 계층으로 데이터를 전송
- 최상위 계층에 있는 애플리케이션은 헤더가 모두 제거된 데이터만 받음

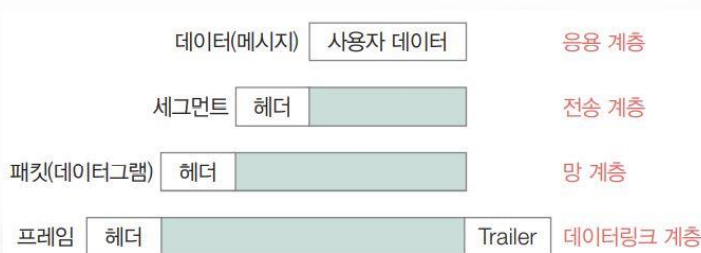


그림 4-12 프로토콜의 캡슐화

■ UDP

- 애플리케이션 중에서 데이터 전송의 신뢰성이 중요하지 않거나 애플리케이션 간 연결 상태를 유지하지 않아도 되는 경우에는 UDP를 사용
- TCP와 같이 신뢰성을 위한 절차나 구조가 없어서 메시지를 보내는 컴퓨터는 상대방이 잘 받았는지 확인할 수 없음
- 송수신 포트 번호, 길이, 체크섬만으로 간단하게 구성
- 도메인 네임 서버(DNS), IPTV, 음성 인터넷 프로토콜(VoIP), TFTP, IP 터널, 그리고 많은 온라인 게임 등에 사용

04. 인터넷 서비스와 WWW

■ 인터넷 서비스

■ 이메일

- 인터넷이 본격적으로 등장하기 이전부터 사용되었으나 컴퓨터에서 컴퓨터로 전송되는 것은 아니었음
- 1971년에는 미국의 컴퓨터 엔지니어인 레이 톰린슨이 최초로 컴퓨터 간 이메일 전송에 성공
- 초기의 이메일 소프트웨어는 텍스트 기반이어서 키보드만 사용할 수 있었음
- 윈도우 환경의 개인용 컴퓨터가 보급되면서 그래픽 사용자 인터페이스를 갖춘 이메일 소프트웨어가 개발
- 윈도우 운영체제는 아웃룩익스프레스를 무료로 제공했고 파이어폭스를 제작하는 모질라에서는 썬더버드를 배포
- 웹의 사용 편의성이 지속적으로 발전하면서 소프트웨어를 컴퓨터에 설치 하는 방식보다 웹 브라우저를 이용해 이메일을 관리하는 방향으로 발전

■ FTP

- 파일 전송을 위해 탄생한 프로토콜
- 초기에는 텍스트로 명령어를 입력해 전송하는 방식이었고, 이후에 그래픽 사용자 인터페이스를 갖춘 소프트웨어가 개발
- 오늘날에는 파일 공유를 위해 FTP를 사용하는 경우가 거의 없음 / 대신 HTTP나 SMB를 사용
- 그러나 기가바이트 정도로 용량이 큰 파일을 공유하거나 파일의 내용이 보안상 중요하다면 여전히 FTP를 사용

04. 인터넷 서비스와 WWW

■ 월드와이드웹(WWW)

■ 월드와이드웹의 탄생

- 1989년 CERN에서 근무하던 팀 버너스리가 제안한 방식
- CERN에 근무하는 과학자들이 서로 정보를 간편하게 공유할 수 있는 체계를 연구하던 중 창안
- 개발자들은 무료 웹 브라우저를 개발하여 배포했는데 이로써 웹 사용자가 크게 늘어남

■ 웹 언어

• HTML

- 하이퍼텍스트 마크업 언어
- 태그를 이용해 제목, 단락, 목록 등과 같은 본문을 위한 구조적 의미를 나타낼 뿐만 아니라 링크, 인용과 그 밖의 항목으로 구조적 문서를 만들 수 있는 방법을 제공하는 언어
- 하이퍼텍스트: 하이퍼링크(참조)를 통해 한 문서에서 다른 문서로 즉시 접근할 수 있는 텍스트
- 마크업 언어: 태그 등을 이용해 문서나 데이터의 구조를 표현하는 언어 / HTML 외에도 SGML, XML 등
- 태그: 원래 텍스트와는 별도로 원고의 교정 부호와 주석을 표현하기 위한 것이었으나 용도가 점차 확장되어 문서의 구조를 표현하는 역할을 수행

04. 인터넷 서비스와 WWW

■ 월드와이드웹(WWW)

■ 웹 언어

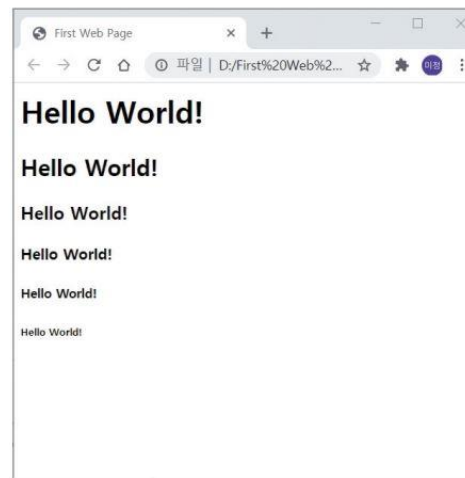
• HTML

- HTML은 간단한 문서 편집기로도 작성할 수 있음
- HTML 문서의 최종적인 모습은 웹 브라우저로 확인
- MS워드나 한글과 같은 문서 편집기는 일반 문서로 작성한 것을 HTML 문서로 변환하는 기능을 제공
- 웹 브라우저는 HTML 태그가 표시된 웹 문서를 읽어서 태그가 지시하는 형태로 문서를 만들어 사용자에게 보여줌

```
<html>
<head>
  <title>
    First Web Page
  </title>
</head>

<body>
  <h1>Hello World!</h1>
  <h2>Hello World!</h2>
  <h3>Hello World!</h3>
  <h4>Hello World!</h4>
  <h5>Hello World!</h5>
  <h6>Hello World!</h6>
</body>
</html>
```

(a) HTML 문서



(b) HTML 문서를 웹 브라우저로 확인한 결과

그림 4-13 웹 브라우저의 기본동작

04. 인터넷 서비스와 WWW

■ 월드와이드웹(WWW)

■ 웹 언어

• 자바스크립트

- 사용자 입장에서 수동적 일 수 있는 초기의 HTML에 비해 자바스크립트는 사용자와 상호 작용이 가능
- 사용자의 입력을 받아 서버와 통신할 수 있고 검색 기능도 사용할 수 있음

[q1.html]

```
<html>
<body>
<script src="https://code.jquery.com/jquery-3.1.0.min.js"></script>
<script src="app.js"></script>
</body>
</html>
```

[app.js]

```
$(function( ){
    alert('Hello World');
});
```



그림 4-14 자바 스크립트의 동작

04. 인터넷 서비스와 WWW

■ 월드와이드웹(WWW)

■ 웹 언어

• HTML5

- 동영상 재생은 웹 문서로는 지원이 어려웠는데 이를 구현하기 위해 여러 기술이 추가로 개발
- 어도비의 플래시, 마이 크로소프트의 실버라이트, 액티브X 등
- 특정한 소프트웨어를 설치하도록 함으로써 웹 브라우저가 지원하지 힘든 기능을 대신 수행
- HTML5가 발표되면서 동영상 재생을 외부 도움 없이 웹 브라우저 자체에서 해결 할 수 있게 됨
- 태그를 이용해서 간단히 동영상 재생을 구현
- HTML5는 플러그인이 필요 없는 스크립트 API를 20종 이상 지원

```
<!DOCTYPE html>
<html >
<body>
<video width="320" height="240" controls>
<source src="dance.mp4" type="video/mp4">
</video>
</body>
</html>
```

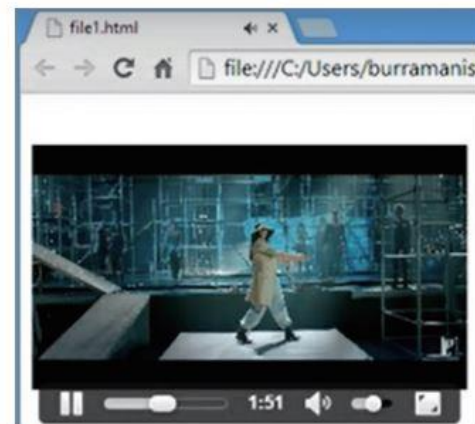


그림 4-15 HTML5로 동영상 재생하기

04. 인터넷 서비스와 WWW

■ 월드와이드웹(WWW)

■ HTTP

- 클라이언트가 서버에 HTML 문서를 요청하는 방법, 서버가 클라이언트에 HTML 문서를 전달하는 방법을 정의
- 오늘날엔 HTML 문서뿐만 아니라 멀티미디어 데이터를 전송하거나 클라이언트에서 서버로 데이터를 전송할 때도 사용
- 웹 브라우저는 어떤 HTML 문서를 전송받고자 하는지 URL 주소에 적어 웹 서버에 서비스 요청을 함

• 클라이언트 요청

- HTTP 버전이 1.1로 표기
- 클라이언트의 HTTP 헤더에는 웹 브라우저의 종류 표시
- 웹 브라우저가 처리할 수 있는 파일과 언어의 종류 표시

• 서버 응답

- 클라이언트 요청 정상 처리 알림을 위해 헤더의 데이터에 '200 OK'가 표시
- 헤더 정보 확인이 끝나면 클라이언트가 요청한 index.html 파일을 전송
- 클라이언트가 HTML 파일을 해석한 후 콘텐츠를 화면에 표시

[클라이언트 요청]

```
GET /index.html HTTP/1.1
User-Agent: Mozilla/5.0
Accept: text/html
Accept-Language: en-US,en;q=0.5
```

[서버 응답]

```
HTTP/1.1 200 OK
Date: Mon, 23 May 2005 22:38:34 GMT
Content-Type: text/html; charset=UTF-8
Content-Encoding: UTF-8
Content-Length: 138
Last-Modified: Wed, 08 Jan 2003 23:11:55 GMT
Server: Apache/1.3.3.7 (Unix) (Red-Hat/Linux)
ETag: "3f80f-1b6-3e1cb03b"
Accept-Ranges: bytes
Connection: close

<html>
<head>
  <title>An Example Page</title>
</head>
<body>
  Hello World, this is a very simple HTML document.
</body>
</html>
```

**7주차 강의가 끝났습니다,
모두 고생하셨습니다.**

