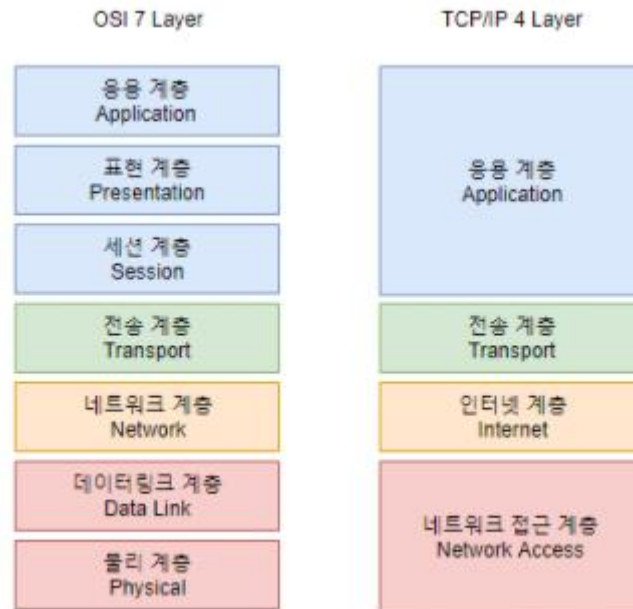


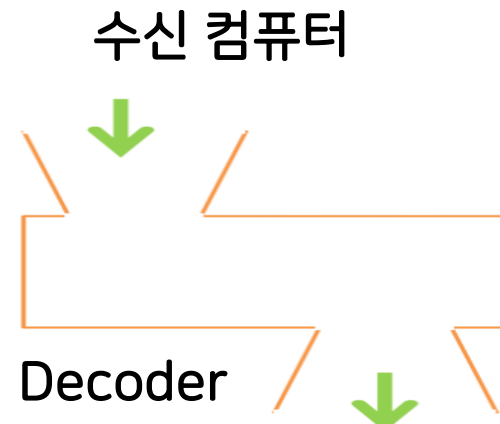
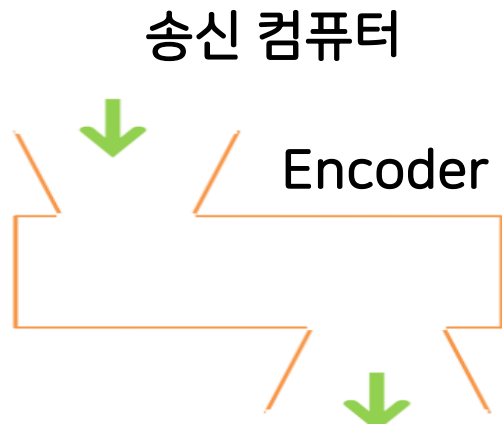
OSI 7계층, TCP/IP 4계층



1계층 Physical Layer

다른 컴퓨터와 통신하기 위해서 +5V를 1, -5V를 0이라고 생각하고 보냈지만 수신이 안 된다.
이유는 전자기파의 값이다.(주파수의 문제)
이 문제를 해결하기 위해서는 인코더로 인코딩한 아날로그 신호를 받을 쪽에서 디코딩해야 한다.

모듈 : 하드웨어적으로 구현되어 있음 (회로)



2계층 Data Link Layer

목적 : 전선 하나로 여러대 컴퓨터 통신하기 위함

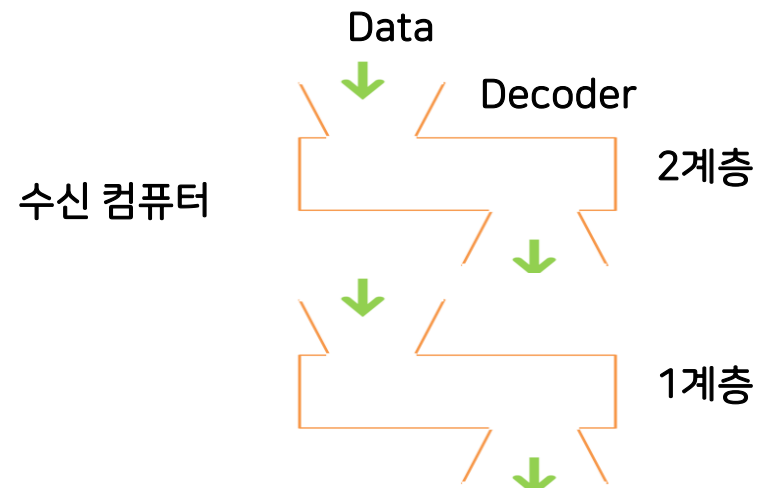
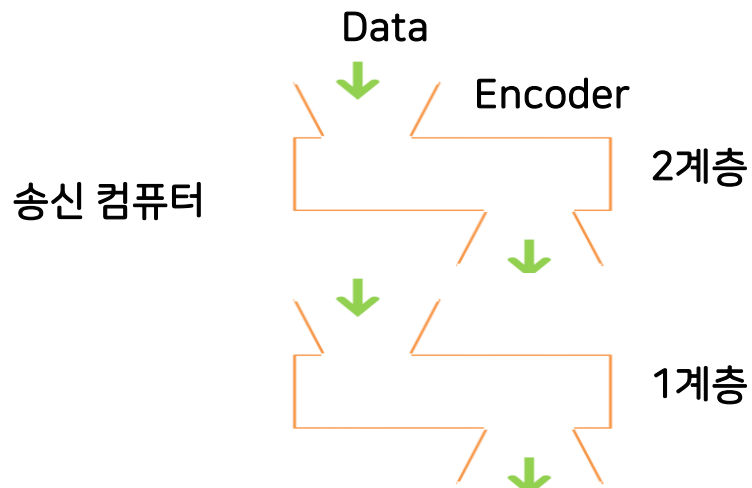
데이터를 특정인원에게 전달하기 위한 스위치에 라우터로 다른 네트워크 간의 연결을 통해 전달

즉 라우터간의 연결이 반복되어 인터넷연결

(예를 들면 일본과의 연결에는 해저케이블이 바다에 설치되어있다.)

동시에 여러개 컴퓨터가 특정 컴퓨터 데이터 송신 시 특정 비트열로 앞 뒤에 붙여 구분한다.

모듈은 하드웨어적 구현 -> 랜카드



3계층 Network Layer

특정 IP주소에 Data 전송 즉 패킷을 전송하기 위해 길을 찾는것(routing)
인접 라우터부터 해당 IP를 찾아 최종 IP에 도달

운영체제 커널에 소프트웨어적 구현.

4계층 Transport Layer

포트번호: 하나의 컴퓨터에서 동시에 실행되는 프로세스들이 서로 겹치지 않게 갖는 정수값.
송신자는 데이터를 보낼 때 수신자 컴퓨터에 프로세스 포트 번호를 붙여 보냄 (미리알고있음)
포트번호는 정해져있기 때문이다. 포트번호를 사용해 도착 컴퓨터의 프로세스까지 데이터가
도달하는 모듈.

예) www.naver.com:80 여기서 80은 포트번호다. 81,82..다 안된다.

3계층과 마찬가지로 운영체제 커널에 소프트웨어적 구현.

7계층 Application Layer

현대 인터넷은 OSI가 아닌 TCP/IP 모델이다.

TCP/IP 소켓 프로그래밍 : 운영체제의 3계층에서 제공한 API를 활용해서 통신 가능한 프로그램을 만드는 것.

누구든 자신만의 Application Layer 프로토콜을 만들수 있다는 뜻.

대표적인 Application Layer Protocol 인 HTTP로 인코딩,디코딩

HTTP이해 -> 클라이언트와 서버 패더다임을 알아야한다. body,header,statuscode

MVC와 마찬가지로 Layerd Architecture가 있는데 그 대표적인 예가 네트워크 시스템 즉 하나의 큰 소프트웨어라 할 수 있다.