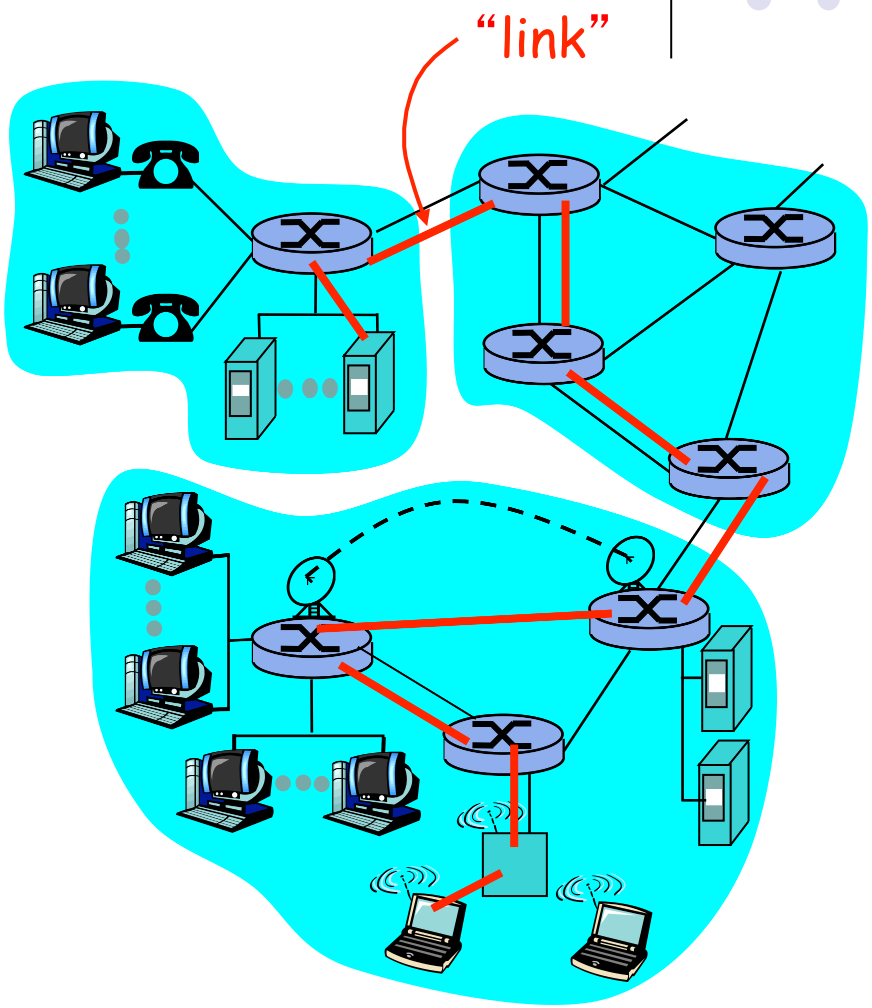
Lecture 4: Datalink layer

- Functionalities:
  - Encapsulation, addressing
  - Error detection and correction
  - Flow control
  - Media access control

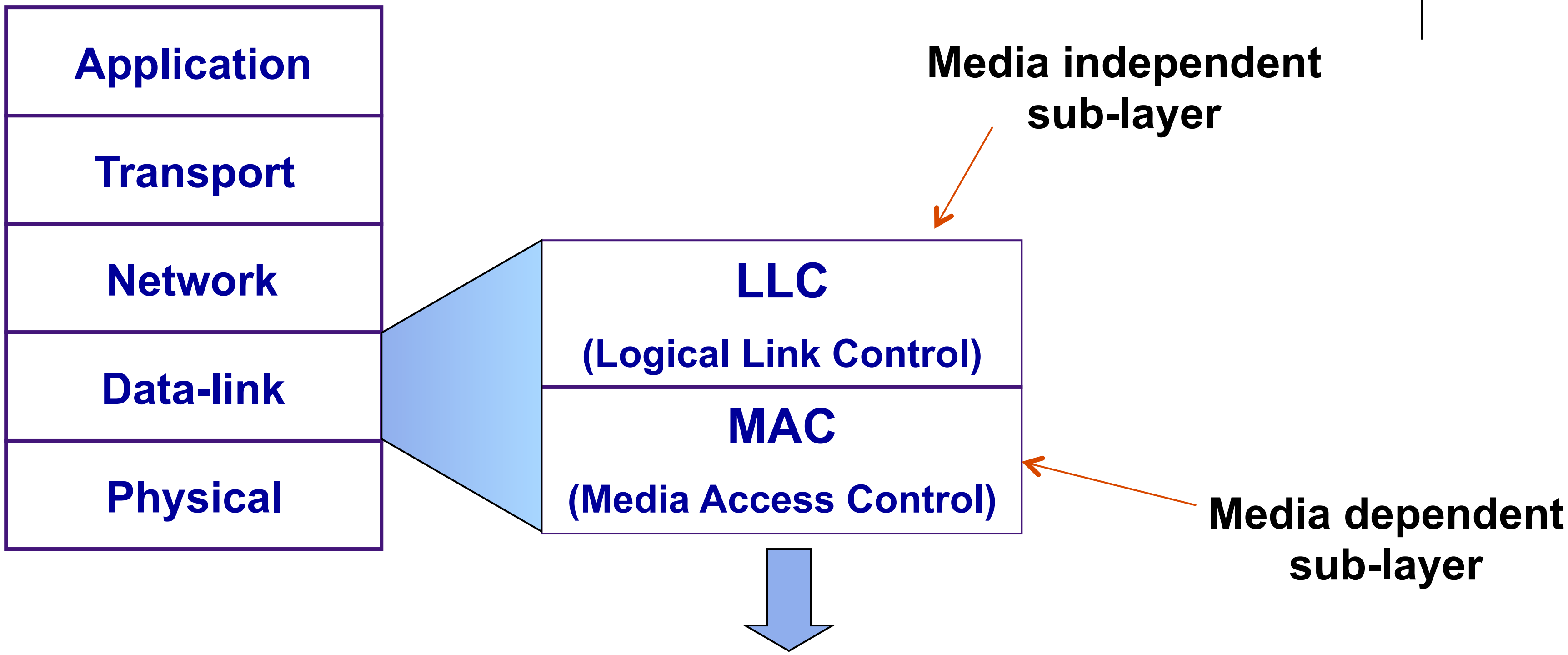
Overview of Data link layer

Network nodes and links

- Network nodes:
  - PCs, Laptop, Routers, Server…
- Links:
  - Communication chanel between **adjacent nodes**
  - Wired link: Ethernet LAN, ADSL, fiber optic…
  - Wireless link: Wi-fi, FSO, Satellite,…
- Datalink layer responsibility:
  - Transmit data between adjacent elements.



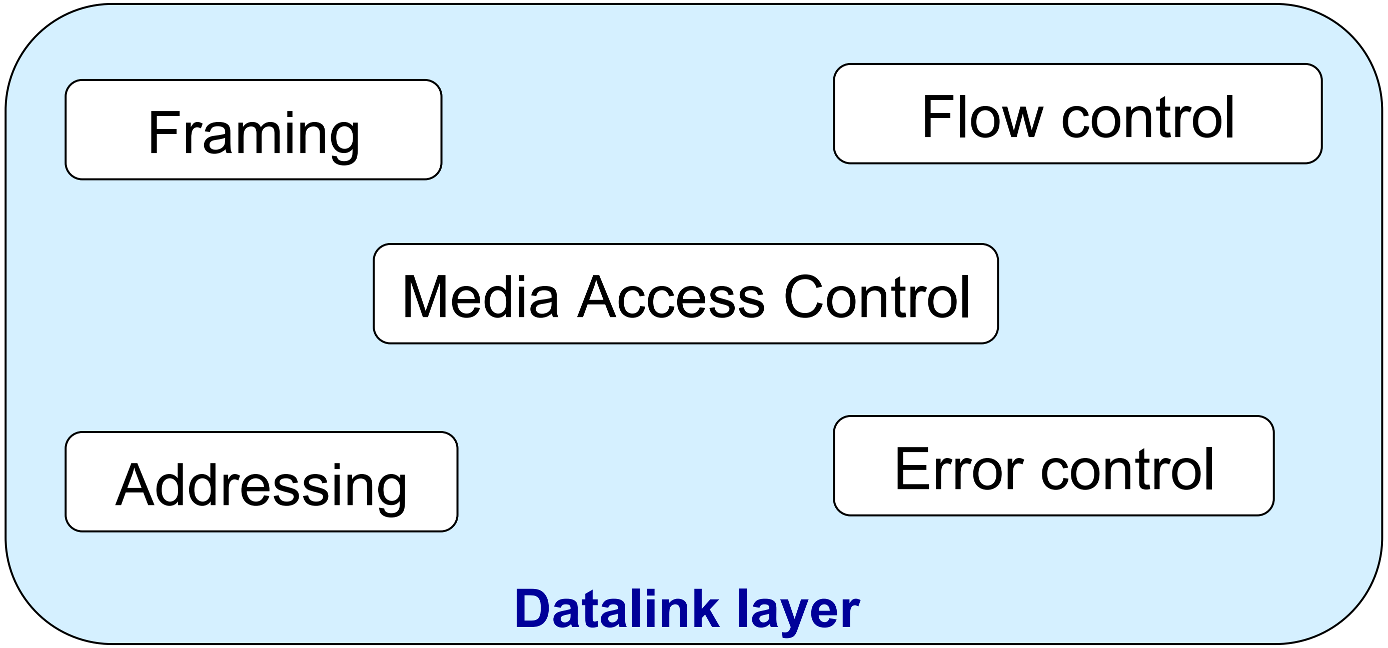
Datalink layer in Layer architecture



802.2 LLC					
802.3 Ethernet	802.4 Token Bus	802.5 Token Ring	802.11 Wi-Fi	…..	802.16 Wi-Max

IEEE 802.x series

Functionalities



Functionalities

- Framing:
  - Sender: place the network layer packet into the frame, add header, tail
  - Receiver: Remove the header, tail for extracting the network packet.
- Addressing:
  - Physical address in the header of the frame for identifying the source and the destination.

Framing-Example of HDLC frame

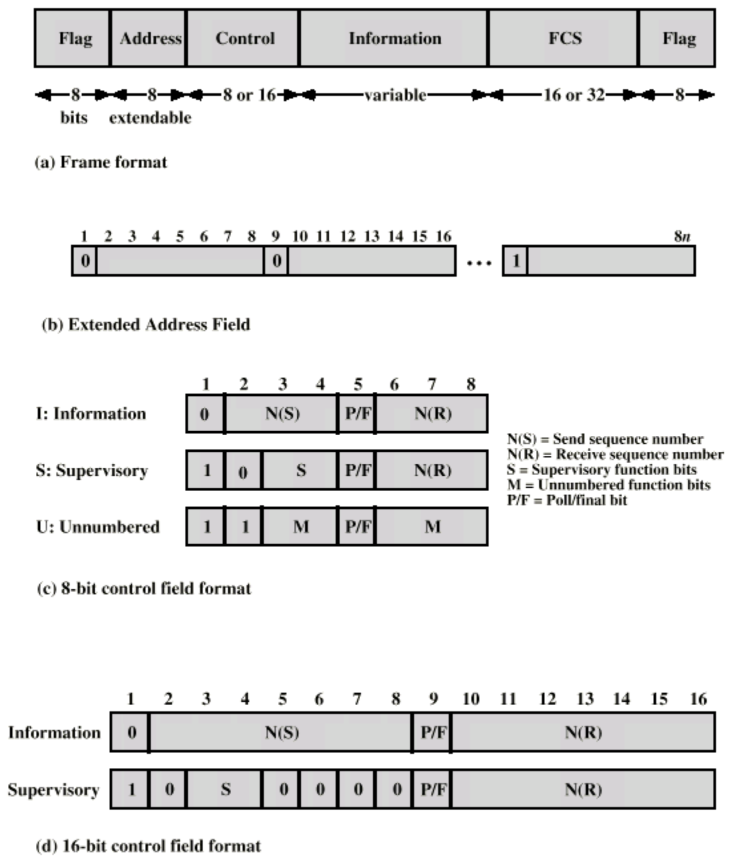


Figure 7.10 HDLC Frame Structure

Functionalities (2)

- Media access control:
  - If the nodes in the network share common media, a Media access control protocol is required.
- Flow control:
  - Control the transmission speed of the sender so that the receiver does not overloaded.
- Error control:
  - Detect and correct errors
  - e.g. parity check, checksum, CRC check

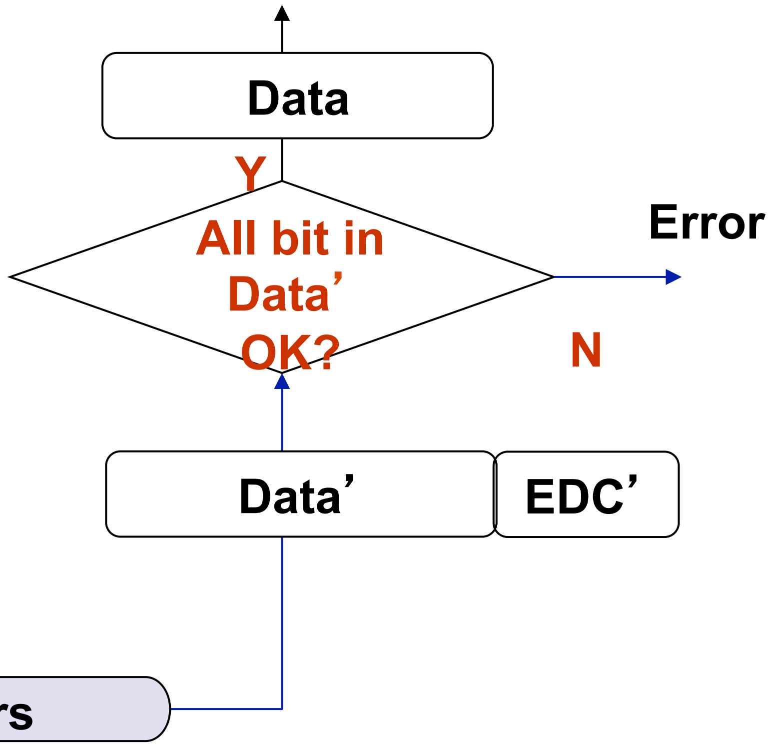
Error control

Error detection

Error correction

Principle of error correction

EDC= Error Detection Code (redundancy)
EDC is added to data before sending to the destination.



Parity code

A check bit is added to the original data to ensure that the total number of bit 1 is even (even parity code) or odd (odd parity code)

- Single code
  - Able to detect single bit error

0111000110101011	0

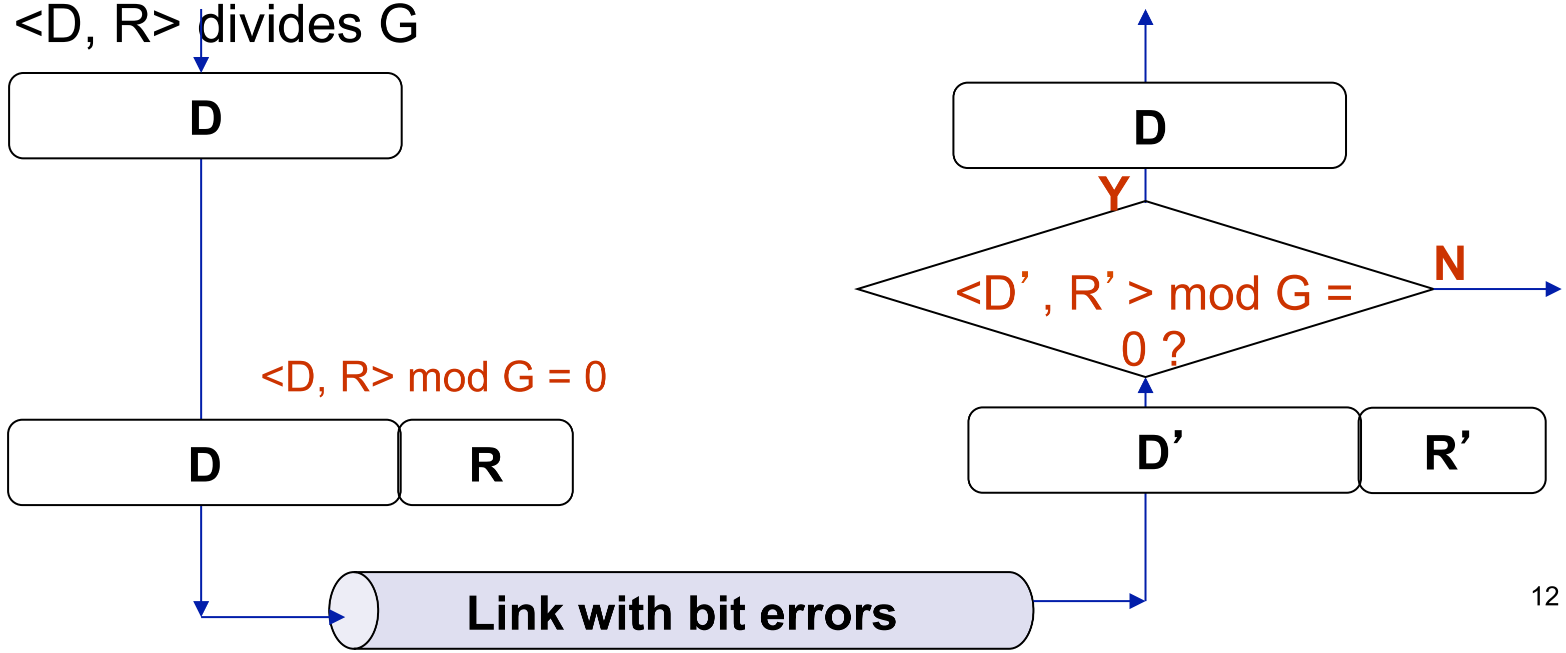
- Two dimension code
  - Detect and correct single bit error

```
10101|1        10101|1
11110|0   ⇒    10110|0
01110|1        01110|1
-------        -------
00101|0        00101|0
```

- Application: mainly on hardware, ex: while sending data on PCI and SCSI bus

CRC: Cyclic Redundancy Check

- Data is considered as a binary string: D
- We wants to generate a error code with length r
- Choose another binary string of (r+1) bit, G (Generator)
- Find a string R with length r bits such that the concatenation of D and R is a binary number that divides G (modulo 2)
  - <D, R> divides G



CRC: How to find R

- <D, R> = D.2^r XOR R
- Since <D, R> divides G then
  - D.2^r XOR R = n.G
  - ➔ D.2^r = n.G XOR R (associativity)
- This means, R is the remainder of the division D.2^r by G (division modulo 2)

$$\boxed{R = D.2^r \bmod G}$$

R=110, the string to send is $\underbrace{10101001}_{D}\underbrace{110}_{R}$

- Ví dụ

```
                 G
 10101001000   |1001
 1001    D      1011110
 ----
   1110
   1001
   ----
    1110
    1001
    ----
     1111
     1001
     ----
      1100
      1001
      ----
       1010
       1001
       ----
         110
          R
```

(The brace labelled D spans 10101001; G labels 1001.)

CRC under polynomial form

- 1011 ←> $x^3 + x + 1$
- Example of some CRC using in the pratice:
  - CRC-8 = $x^8 + x^2 + x + 1$
  - CRC-12 = $x^{12}+x^{11}+x^3+x^2+x$
  - CRC-16-CCITT = $x^{16} + x^{12} + x^5 + 1$
  - CRC-32 = $x^{32} + x^{26} + x^{23} + x^{22} + x^{16} + x^{12} + x^{11} + x^{10} + x^8 + x^7 + x^5 + x^4 + x^2 + x + 1$
- The longer G is, the more possible that CRC detects errors.
- CRC is widely used in the practice
  - Wi-fi, ATM, Ethernet…
  - Operation XOR is implemented in hardware
  - Capable to detect less than r+1 bits errors

Reaction when errors detected

- Objective: assure that data are transmitted correctly even though the chanel is not realiable.
- Condition
  - Data fram must be transmitted correctly
  - Negligible transmission delay.
- Possible errors
  - Whole frame loss
  - Error frame
  - Loss of error warning message
- Popular techniques:
  - Error detection (as we seen)
  - Acknowledgement/ confirmation
  - Retransmis after timeout
  - Retransmis after a clear confirmation that frame is not arrived
- ARQ technique: automatic repeat request). There are 3 versions:
  - Stop and Wait ARQ
  - Go Back N ARQ
  - Selective Reject ARQ
- Similar to techniques used in flow control.

Media access control

Connection types

- Point-to-point
  - ADSL
  - Telephone modem
  - Leased Line….
- Broadcast
  - LAN using bus topology
  - Wireless LAN
  - HFC:
  - …
- Broadcast networks need media access control protocol in order to avoid collision when nodes try to send data.

Classification of MAC protocol

- Chanel division:
  - Resources of the media is divided into small parts (time - TDMA, frequency- FDMA, Code- CDMA)
  - Distribute a part to each nodes
- Random access:
  - Chanel is not divided, all nodes are allowed to access simultaneously with collision possibility
  - Need a mechanism to avoid collision
  - e.g. Pure Aloha, Slotted Aloha, CSMA/CD, CSMA/CA…
- Sequent access:
  - Nodes can send data one after the other.
  - Token Ring, Token Bus….

Channel division

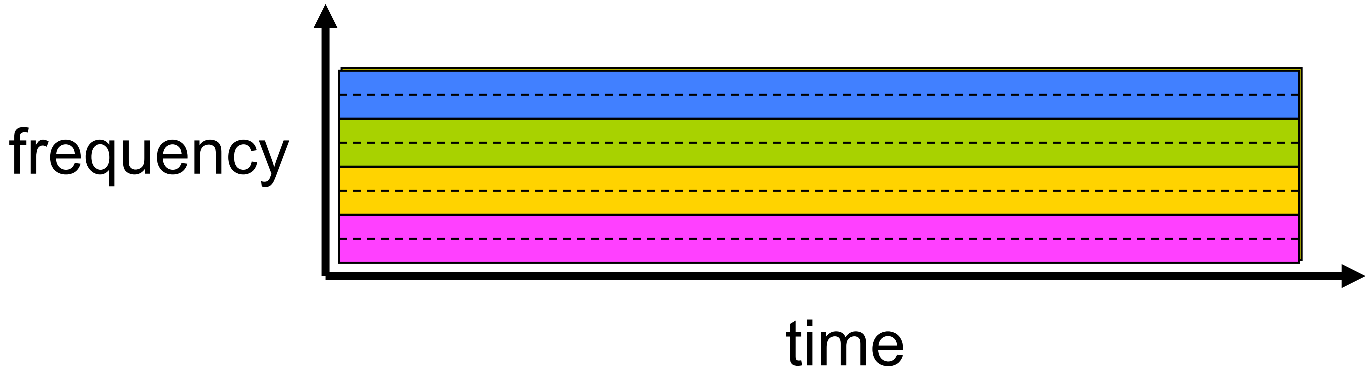
- FDMA: frequency division multiple access
- TDMA: time division multiple access
- CDMA: code division multiple access

TDMA và FDMA

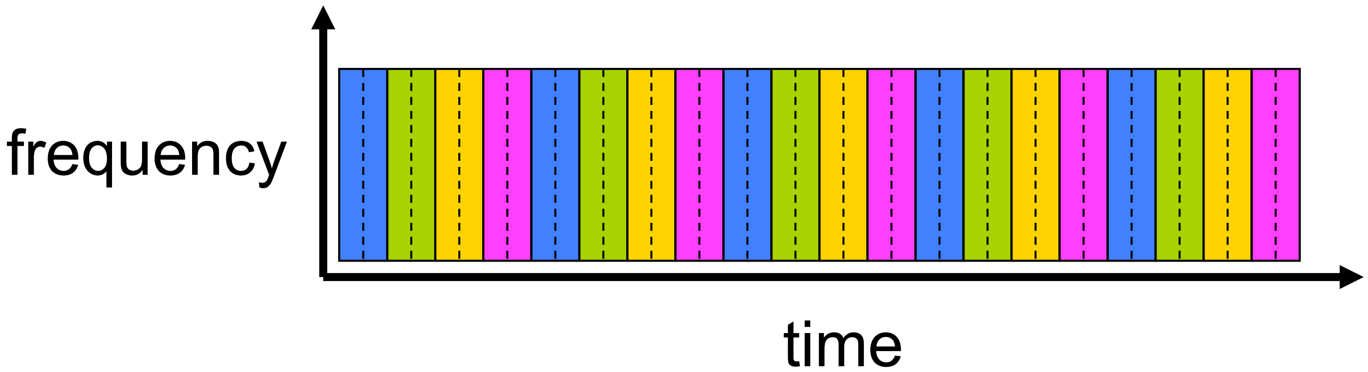
ex

4 stations

FDMA



TDMA:



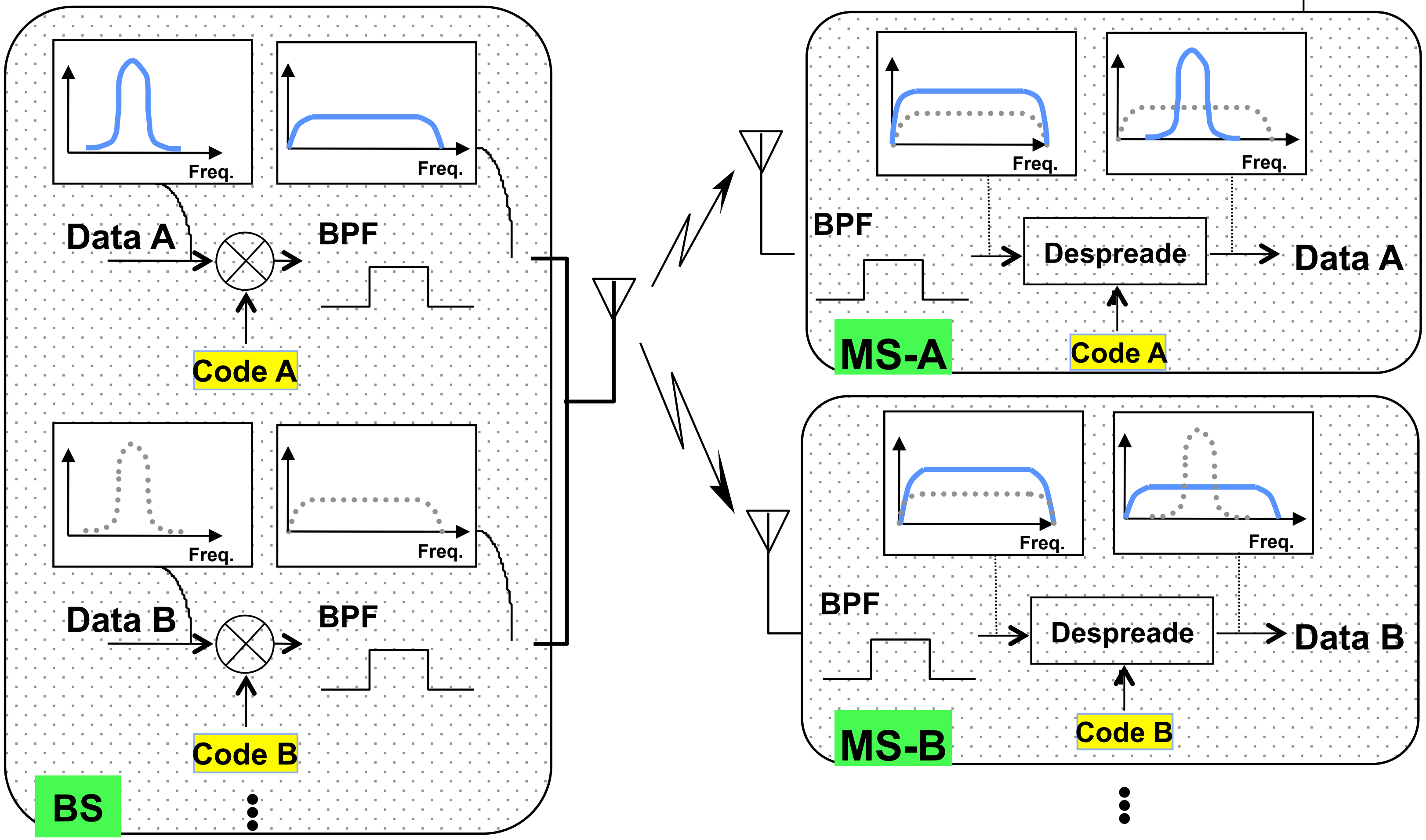
CDMA

- Several senders can share the same frequency on a single physical channel.
- Signals come from different senders are encoded by a different random code
- Encrypted signals are mixed and then transmit on a common frequency.
- The signals are recovered at the receiver by using the same codes as at sender side.
- CDMA use the spread spectrum theory, **CDMA** shows a lot of advantages that other technology cannot achieve.

  http://en.wikipedia.org/wiki/Spread_spectrum

DS-CDMA System Overview (Forward link)

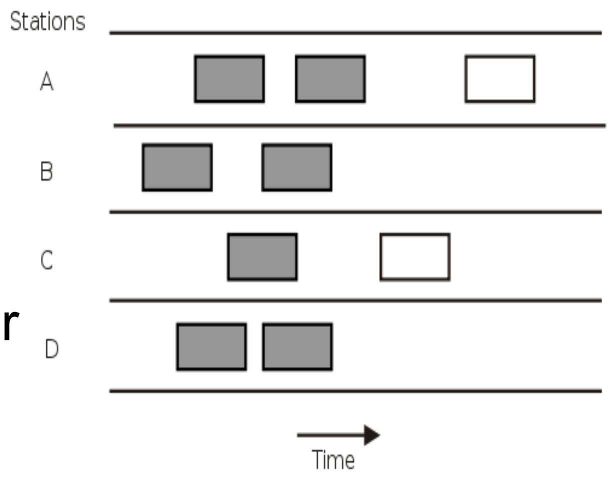
CDMA is a multiple spread spectrum.



Difference between each communication path is only the spreading code

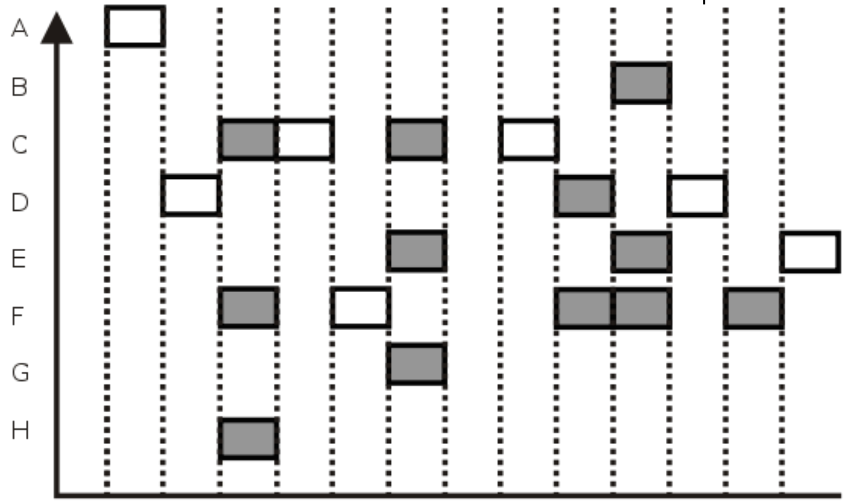
Random access: Pure Aloha

- Aloha is used in mobile network of 1G, 2.5G, 3G using GSM technology .
- Pure Aloha:
  - When one sender has data to send, just sends it
  - If while sending, the senders receive data from other stations → there is collision. All stations need to resend their data.
    - There are possibility to have collision when retransmit.
  - Problem: Sender does not check to see if the chanel is free before sending data
  - Grey package are having overlap in time→ causing collision



Random access: Slotted Aloha

- Times axe is divided into equal slots.
- Each station sends data only at the beginning of a time slot.
- ➔ Collision possibility is reduced
- Still have collision in grey package

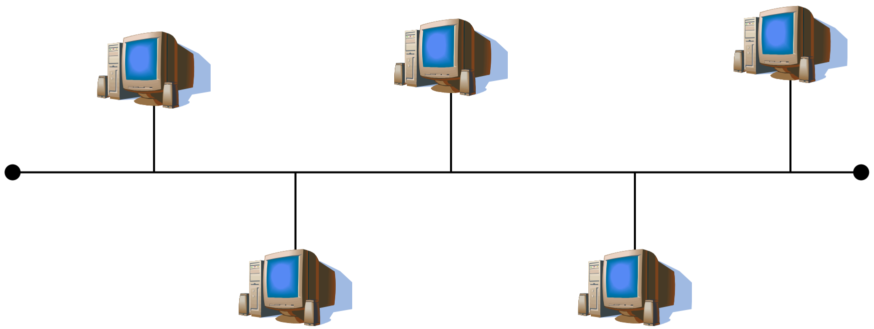


Slotted ALOHA protocol (shaded slots indicate collision)

Random access: CSMA

- CSMA: Carrier Sense Multiple Access
- CSMA idea is similar to what happens in a meeting.
- CSMA:
  - The sender “Listen before talk”
  - If the channel is busy, wait
  - If the chanel is free, transmit

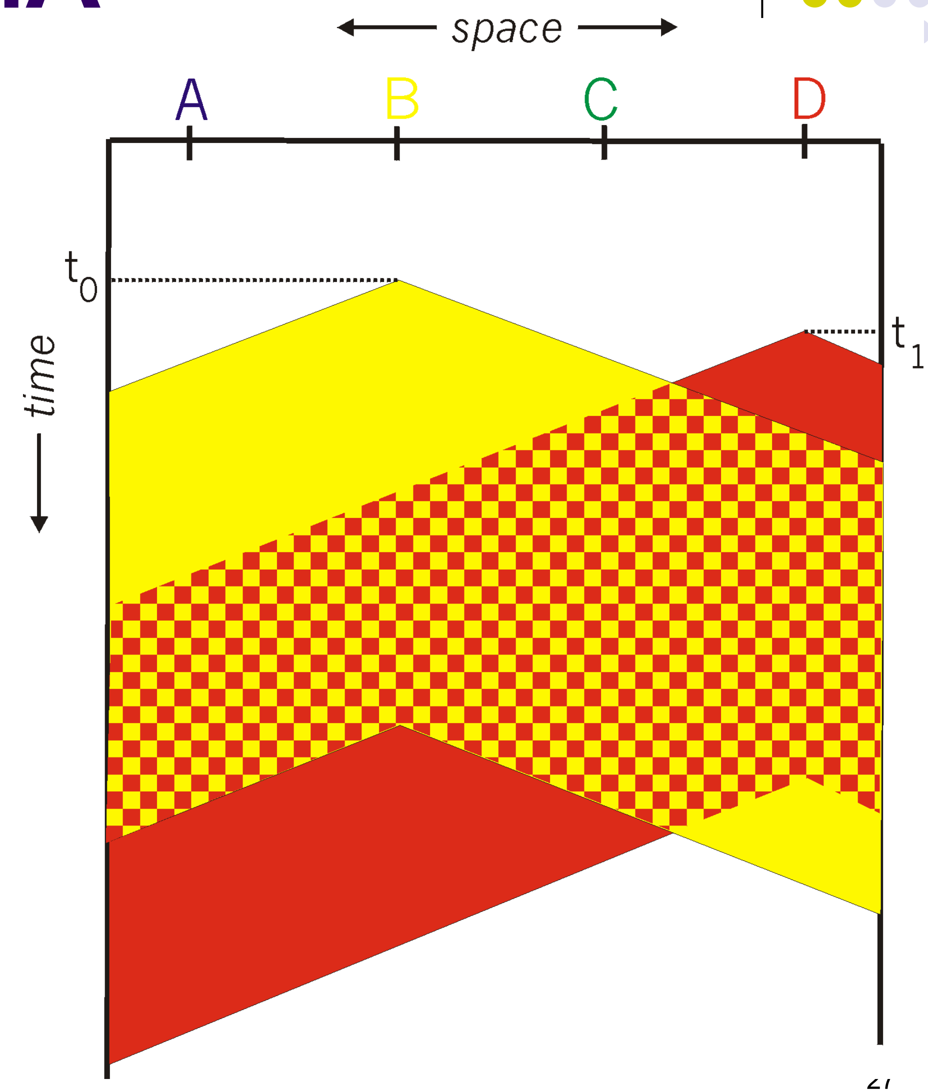
CSMA



- CSMA: Sender listens before transmission:
  - If the channel is free, send all the data
  - If the channel is busy, wait.
- Why there are still collision?
  - Due to propagation delay

Collision in CSMA

- Assume that there are 4 nodes in the channel
- The propagation of the signal from one node to the other requires a certain delay.
- Ex:
  - Transmissions from B and D cause collision



CSMA/CA

- CSMA/CA is used WIFI standard IEEE 802.11
- If two stations discover that the channel is busy, and both wait then it is possible that they will try to resend data in the same time.
  - → collision
- Solution CSMA/CA.
  - Each station wait for a random period → reduce the collision possibility

CSMA/CD

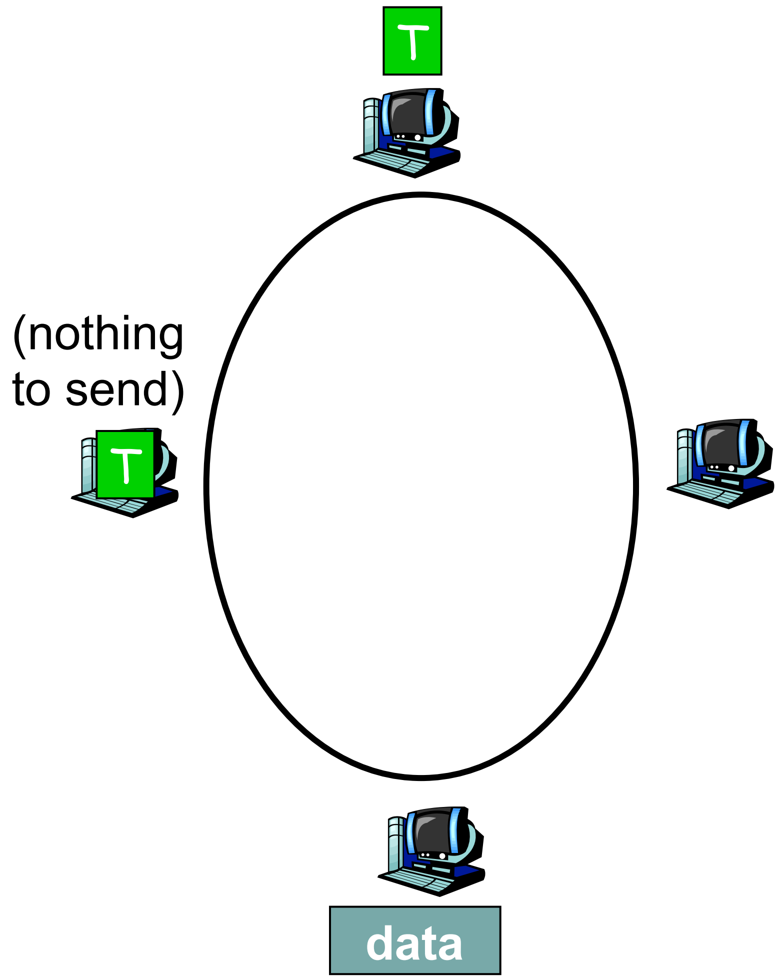
- Used in Ethernet
- CSMA with Collision Detection:
  - “Listen while talk”.
- A sender listen to the channel,
  - If the channel is free then transmit data
    - While a station transmit data, it listens to the channel. If it detects a collision then transmits a short signal warning the collision then stop
    - Do not continue the transmission even in collision as CSMA
  - If the channel is busy, wait then transmit with probability p
- Retransmit after a random waiting time.

Comparison between channel division and random access

- Channel division
  - Efficient, treat stations equally.
  - Waste of resources if one station has much smaller data to send than the others
- Random access
  - When total load is small: Efficient since each station can use the whole chanel
  - When total load is large: Collision possibility increases.
- Token control: compromise between the two above methods.

Token Ring

- A “tocken” is passed from one node to the other in a ring topo
- Only the token holder can transmit data
- After finishing sending data, the token need to be passed to next nodes.
- Some problem
  - Time consuming in passing token
  - Loss of token due to some reasons



Summary on Media access control mechanisms

- Channel division
- Random access
- Token
- What do you thinks about their advantages and weaknesses

Flow control

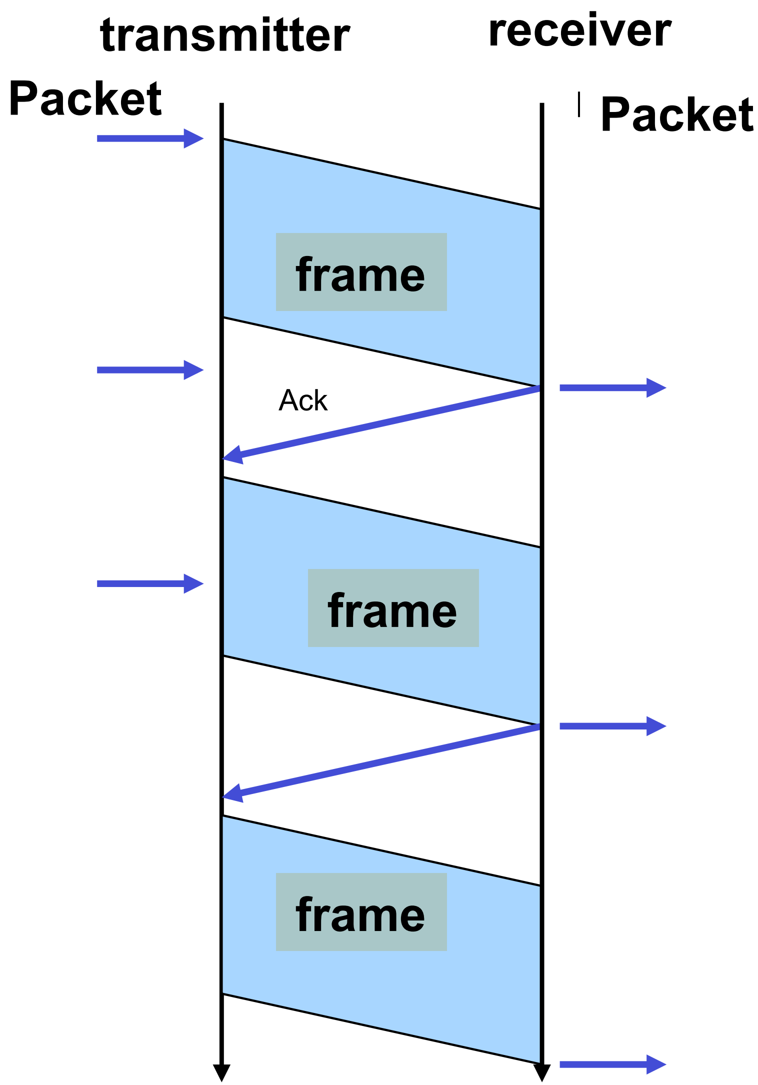
What is flow control

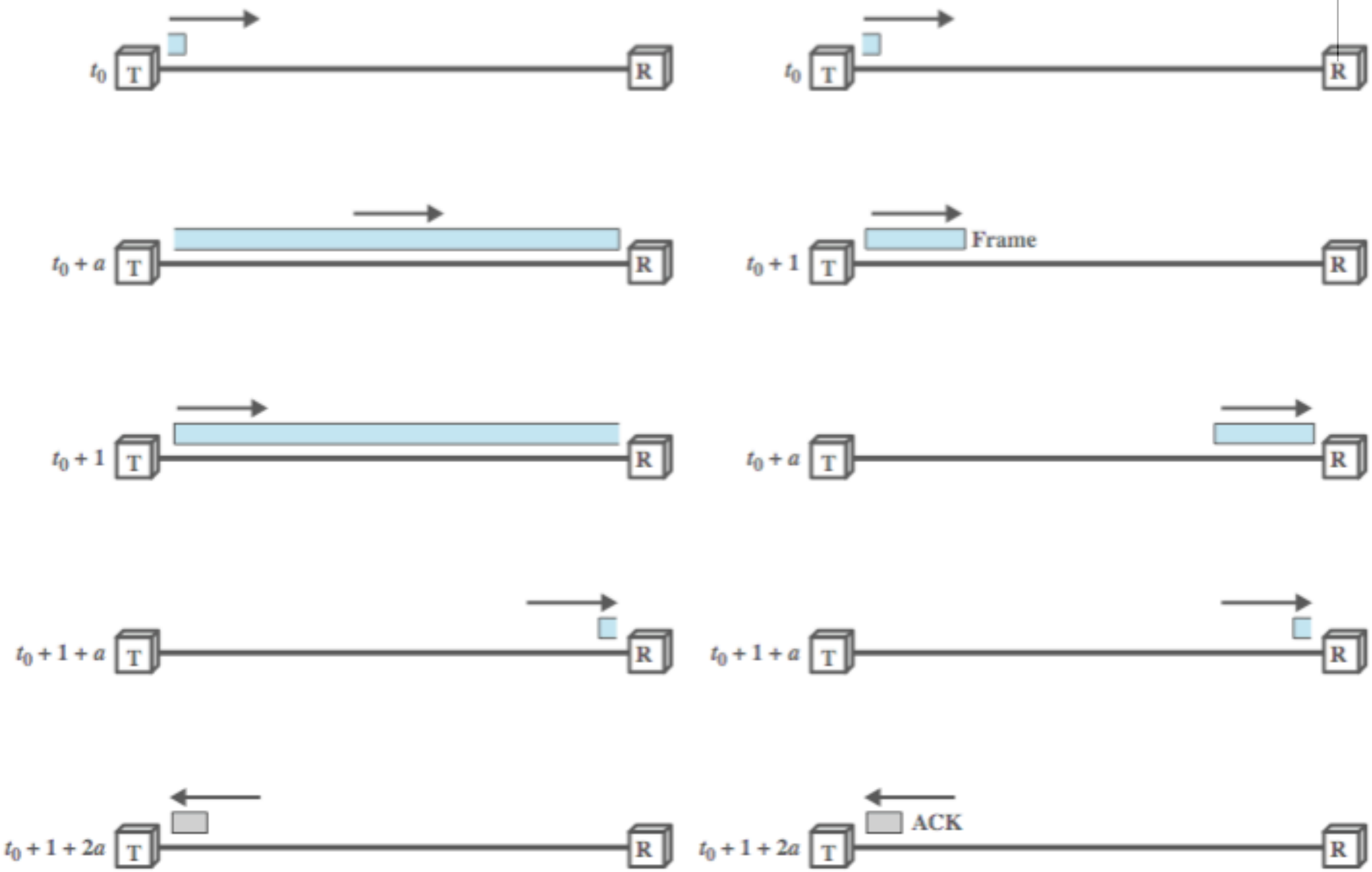
- Goal: Make sure that the sender does not overload the receiver
- Why overloading?
  - The receiver stores data frame in buffer.
  - Receiver performs some processing before deliver data to the upper level.
  - Buffer could be full, leaving no space for receiving more frame → some data fram must be dropped.
- Problem of errors in transmission is excluded
  - All frames are transmitted to correct receiver without error
  - Propagation time is small and could be ignored
- Solution
  - Stop-and-wait mechanism
  - Sliding window mechanism

Stop-and-wait

- Principles
  - Transmitter sends a single frame
  - Receiver receives the fram, process and then informs the transmitter that it is ready to receives next frames by a clear acknowledgement (ACK).
  - Transmitter waits until reception of the ACK before sending next frames.

Stop-and-wait





(a) $a < 1$ (b) $a > 1$

Stop-and-wait

- Advantage
  - Simple, suitable for transmission of big size frames
- Weakness
  - When frames are small, the transmission chanel are not used efficiently.
  - Cannot use often for big size frame due to
    - Limitation in buffer size
    - Big size frame prones to bigger error probability
    - In shared medium, it is not convenient to leave one station using medium for long time

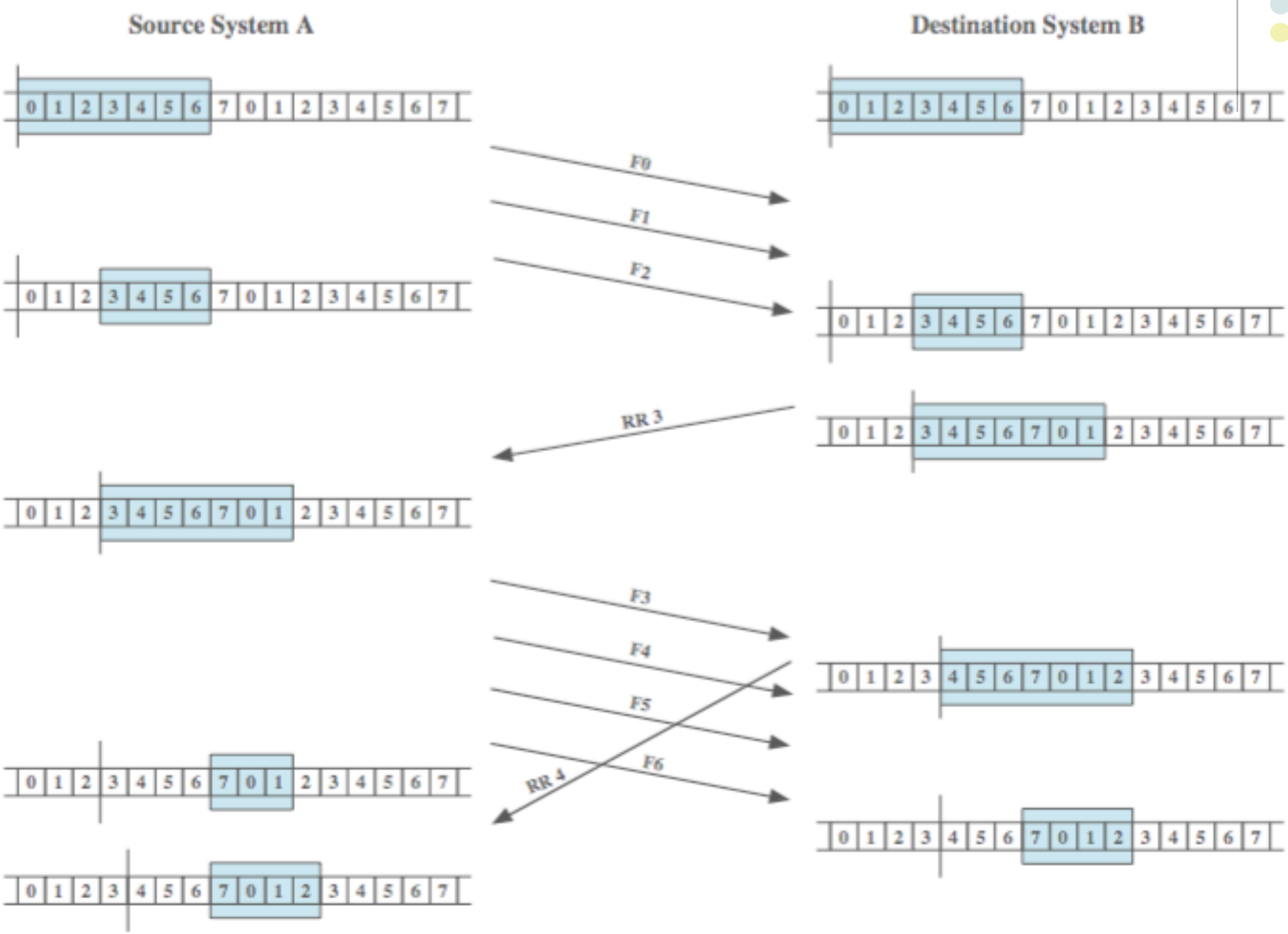
Sliding window: principle

- Transmitter sends more than one frame without waiting in order to reduce waiting time
- Transmitted frame without ACK will still be stored in buffer.
- Number of frame to be transmitted without ACK depends on the size of buffer at transmitter
- When transmitter receives ACK, it realises the succesfully transmitted frame from buffers
- Transmitter continues sending a number of frame equivalent to the number of succesfully trasmitted frames.

Sliding window: principle

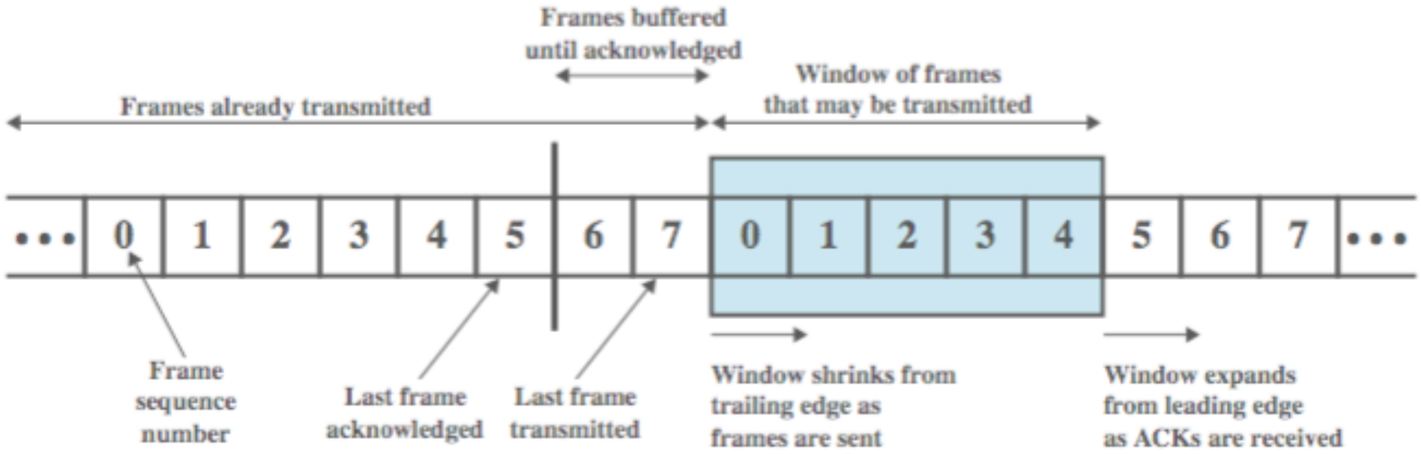
- Assume that A and B are two stations connected by a full duplex media
  - B has a buffer size of n frame.
  - B can receives n frame without sending ACK
- Acknowledgement
  - In order to keep track of ACKed frames. It is neccessary to number frames.
  - B acknowledge a frame by telling A which fram B is waiting for (by number of frame), implicitely saying that B receives well all other frame before that.
  - One ACK frame serves for acknowledes several frames.

Sliding windows: principle

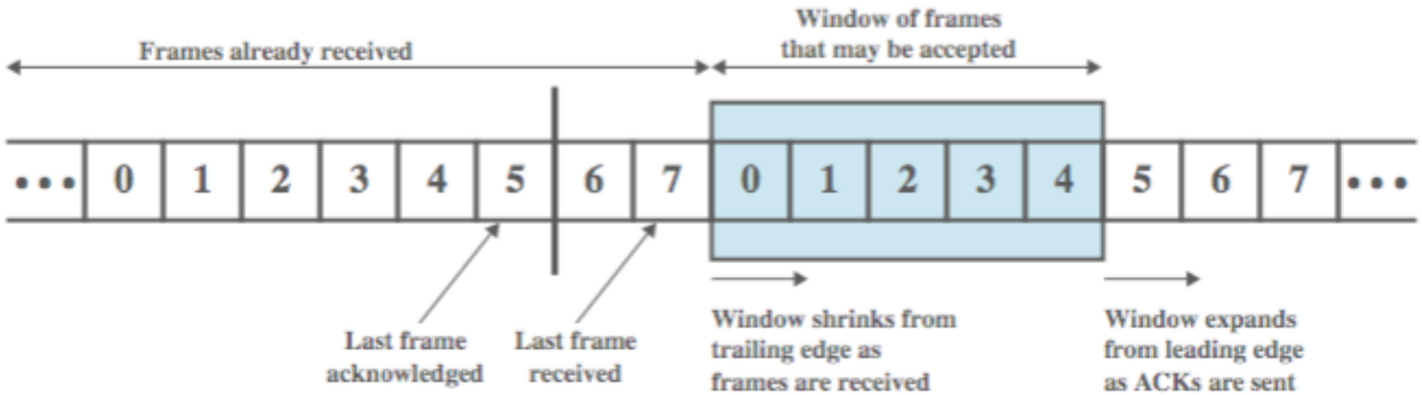


Window list the frames to transmit **Window list the frames in waiting to receive**

Sliding windows



(a) Sender's perspective



(b) Receiver's perspective

Sliding windows

- Frame are numbered. The maximum number must not be smaller than the size of the window.
- Frame are ACKed by another message with number
- Accumulated ACK: If frame 1,2,3,4 are well receive, just send ACK 4
- ACK with number k means all frame k-1, k-2 … already well received.

Sliding windows

- Transmitter needs to manage some information:
  - List of frames transmitted sucessfully
  - List of frames transmitted without ACK
  - List of frames to be sent immediatly
  - List of frames NOT to be sent immediately
- Receiver keep tracks of
  - List of frames well received
  - List of frames expected to receive

Piggy backing

- A and B transmitte data in both sides
  - When B needs to send an ACK while still needs to send data, B attaches the ACK in the Data frame: Piggybacking
  - Otherwise, B can send an ACK frame separatly
  - After ACK, if B sends some other data, it still put the ACK information in data frame.
- Sliding window is much more efficient than Stop-and-Wait
- More complicated in management.

Exercices

- Given a link with rate R=100Mbps
- We need to send a file over data link layer with file size L=100KB
- Assume that the size of a frame is: 1KB, header size is ignored
- Round trip time (RTT) between 2 ends of the link is 3ms
- An ACK message is sent back from receiver whenever a frame is arrived. Size of ACK message is negligible
- What is the transmission time required if using Stop-and-wait mechanism?
- Transmission time with sliding window if the window size is =7?
- Which size of window allow to obtain the fastest transmission?