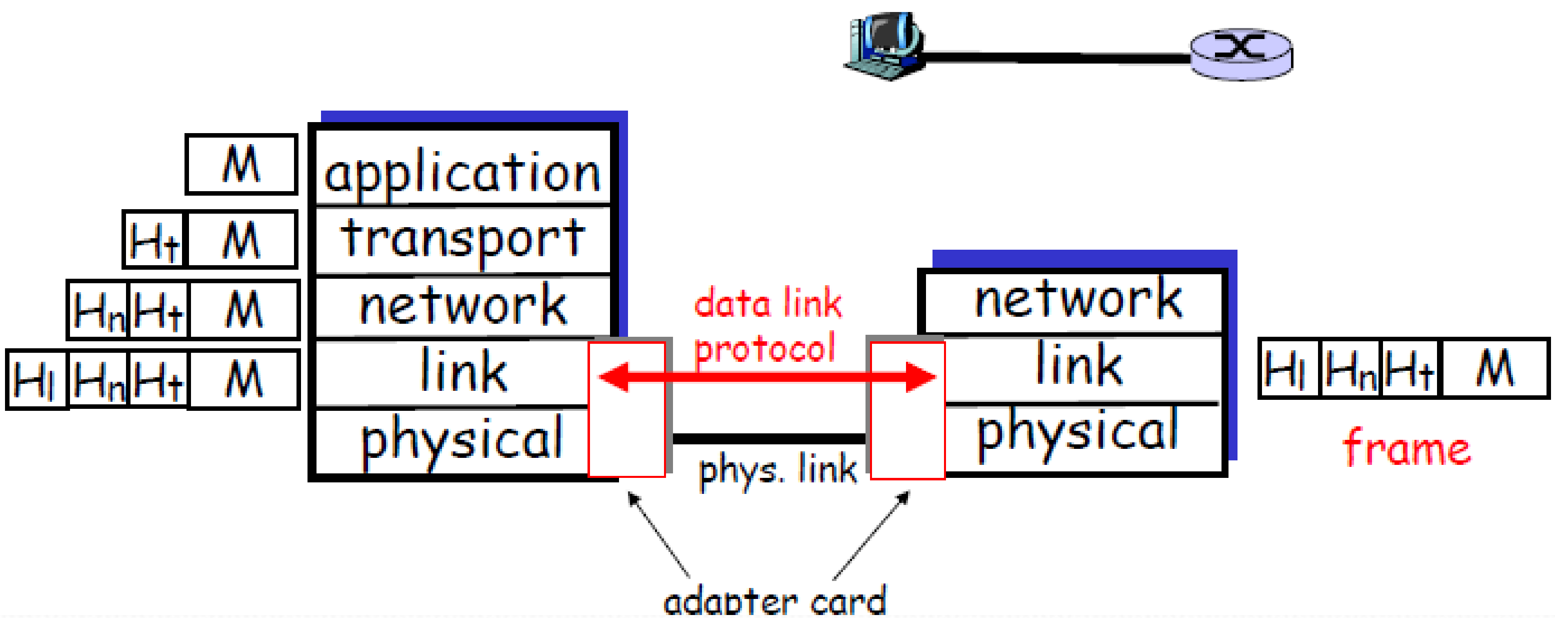
Data Link Layer

Data Link Layer

- Two *physically connected* devices:
  - Host-Router, Router-Router, Router-Host
- Unit of data: *frame*



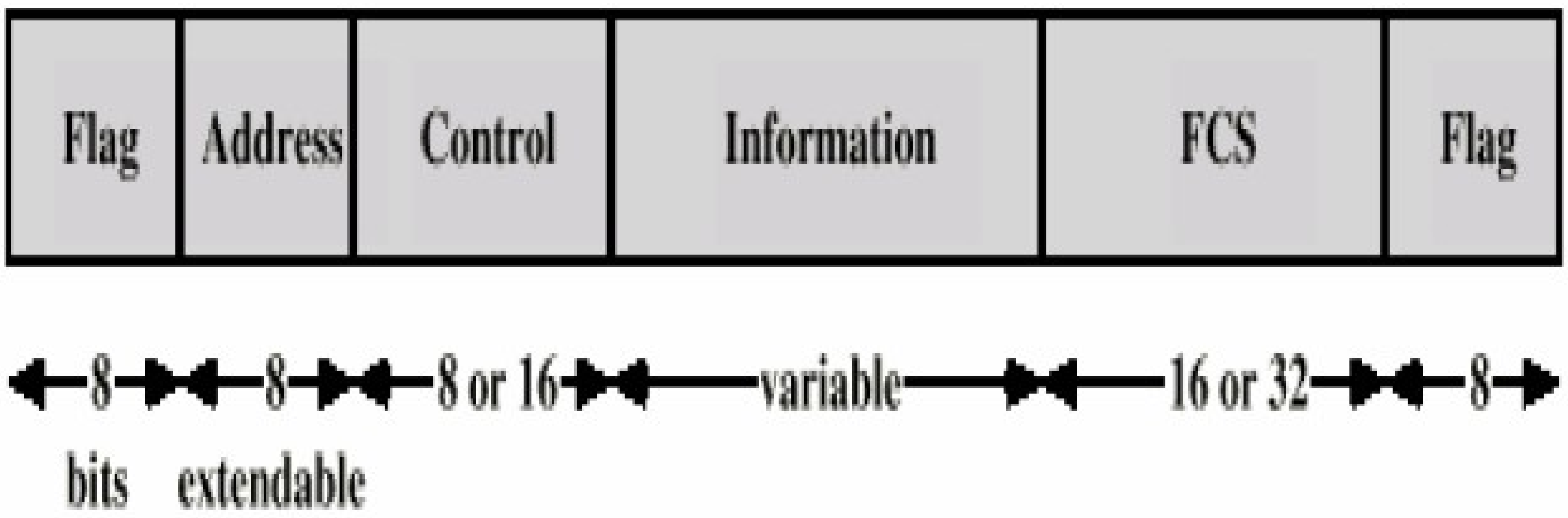
Data Link Control Protocols

- need layer of logic above Physical
- to manage exchange of data over a link
  - frame synchronization
  - flow control
  - error control
  - addressing
  - control and data
  - link management

DLC Service Layer

- ***Framing***
  - Encapsulated packets into frames
  - Establishes frame synchronization
- ***Error Detection and Control***
  - Errors caused by signal attenuation and noise
  - Receiver detects presence of errors
    - Receiver drops frame
    - Receiver requests retransmission (ARQ)
    - Receiver corrects error
- ***Flow Control***
  - Ensuring that the sender does not overwhelm the receiver

Typical Frame Structure



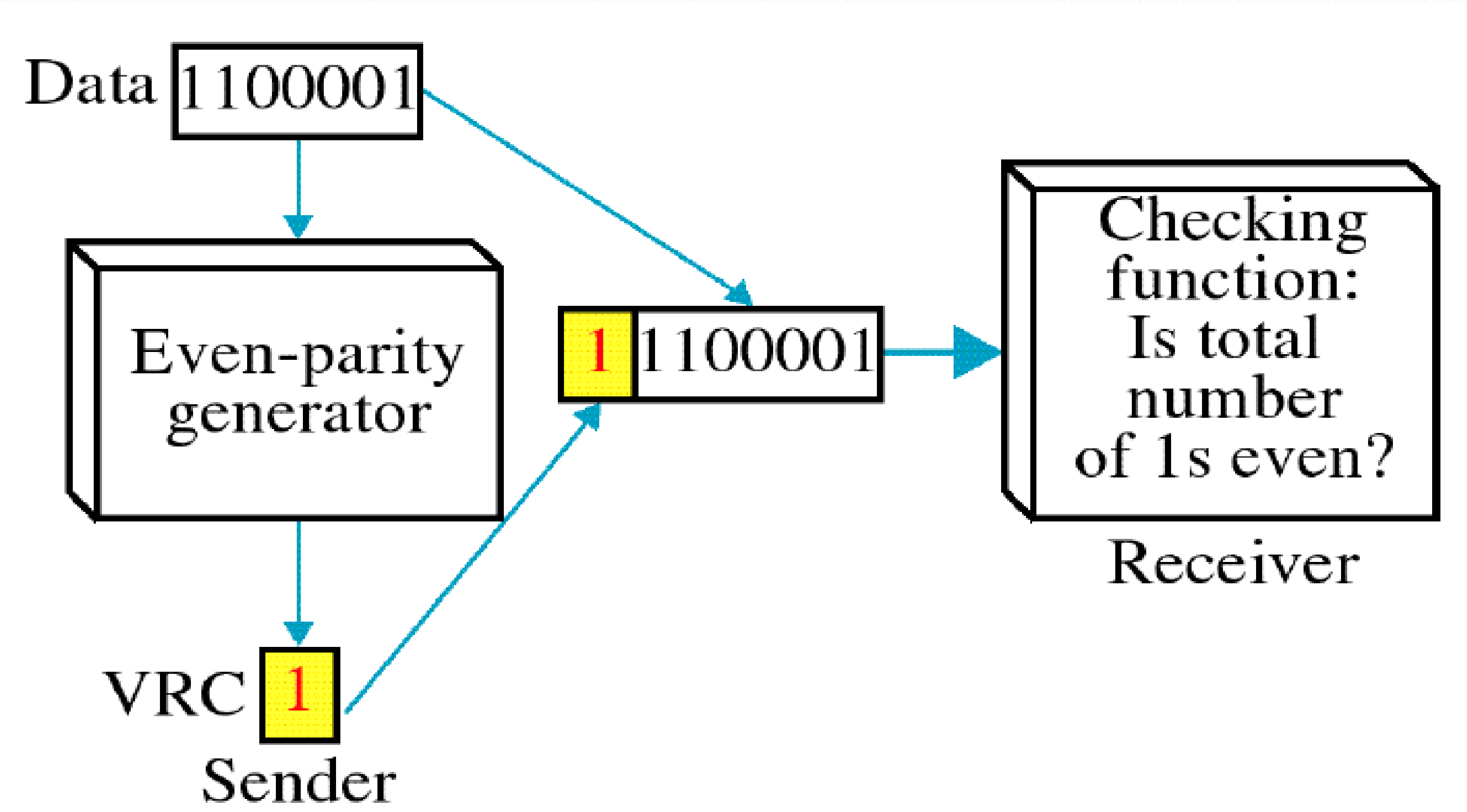
Frame Synchronization

- A special pattern, called a Flag (01111110) appears at the beginning and the end of the frame
- Receiver hunts for flag sequence to synchronize
- Bit stuffing used to avoid confusion with data containing 01111110
  - 0 inserted after every sequence of five 1s
  - If receiver detects five 1s it checks next bit
  - If 0, it is deleted
  - If 1 and seventh bit is 0, accept as flag
  - If sixth and seventh bits 1, sender is indicating abort

Error Detection

- Additional bits added by transmitter for error detection code
- Parity
  - Value of parity bit is such that character has even (even parity) or odd (odd parity) number of ones
  - Even number of bit errors goes undetected

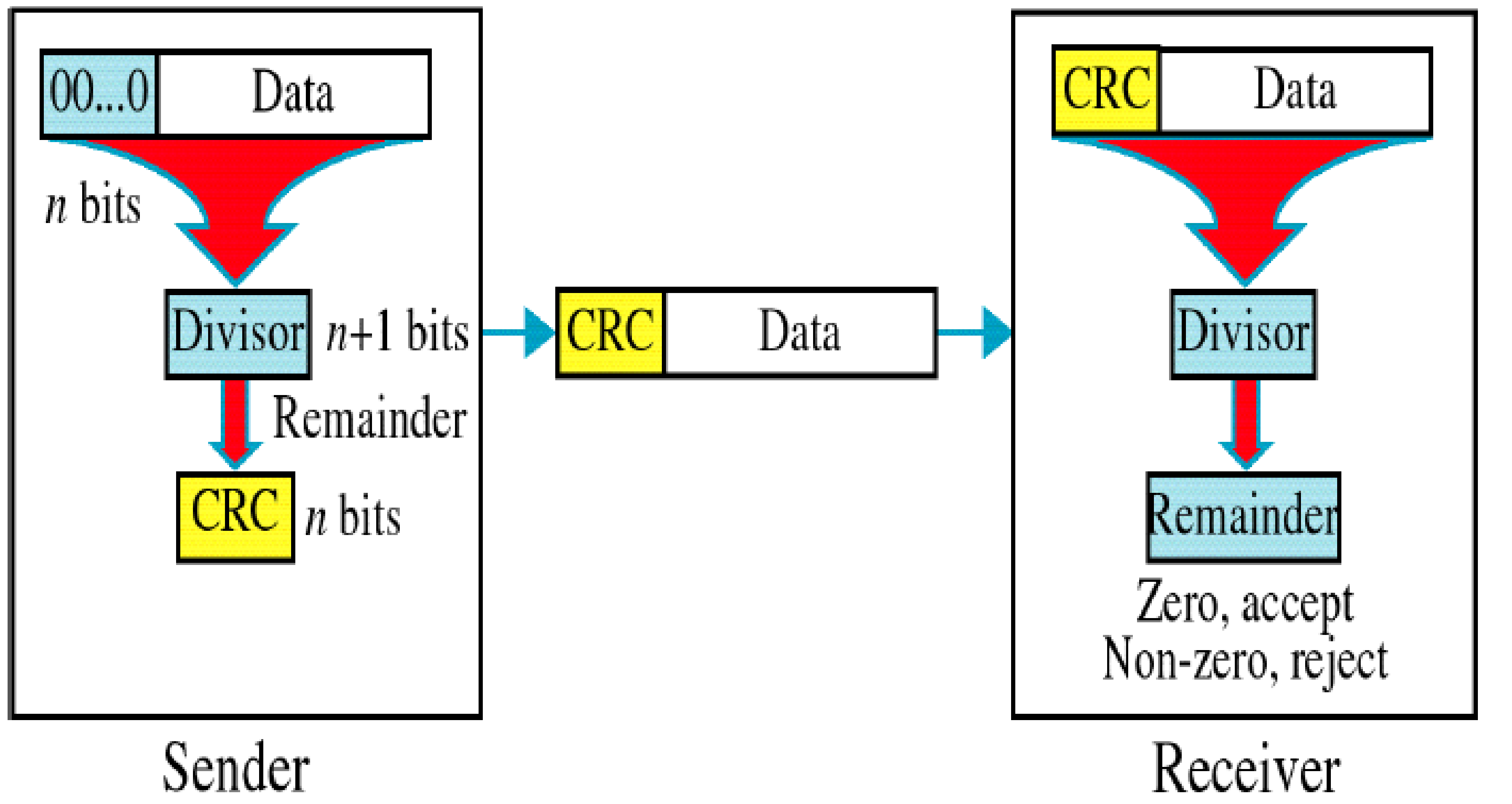
Single Parity Error Checking



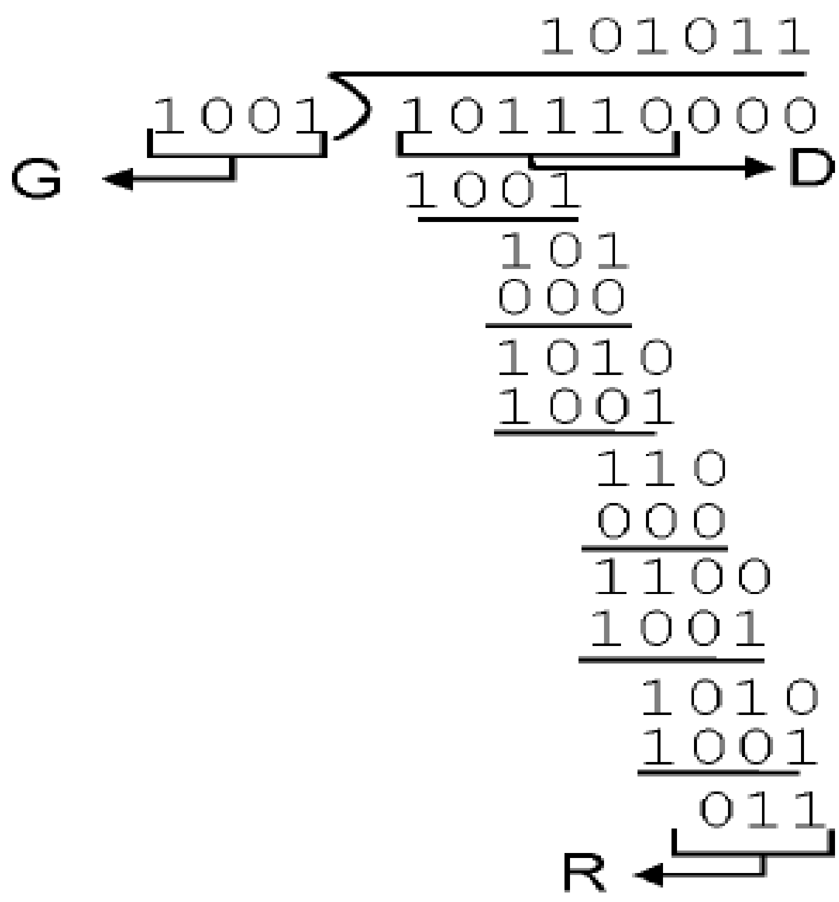
Cyclic Redundancy Check (CRC)

- For every block of k bits, transmitter generates n bit sequence
- Transmit $k+n$ bits which is exactly divisible by some number
- Receive divides frame by that number
  - If no remainder, assume no error
  - If reminder, an error is detected

CRC continued..



Example..

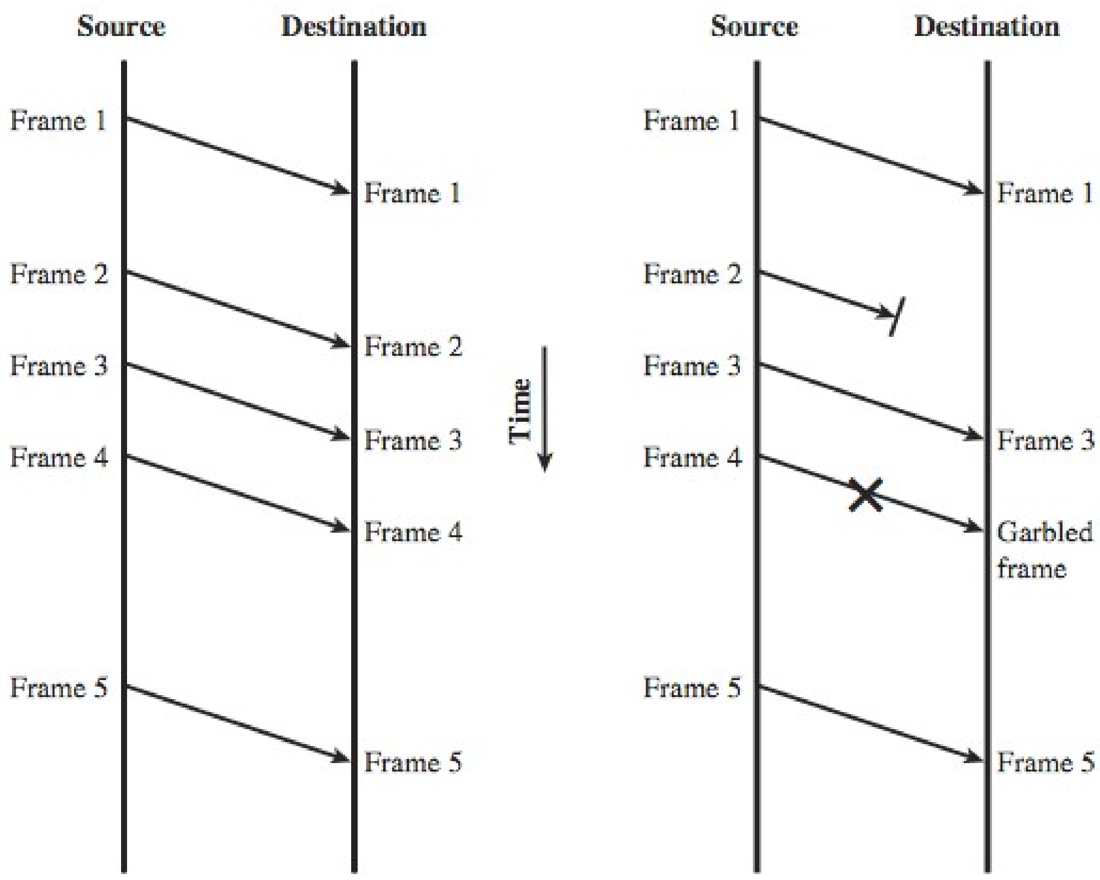


Binary Division

Flow Control

- ensure sending entity does not overwhelm receiving entity
  - by preventing buffer overflow
- influenced by:
  - transmission time
    - time taken to emit all bits into medium
  - propagation time
    - time for a bit to traverse the link
- assume here no errors but varying delays

Model of Frame Transmission



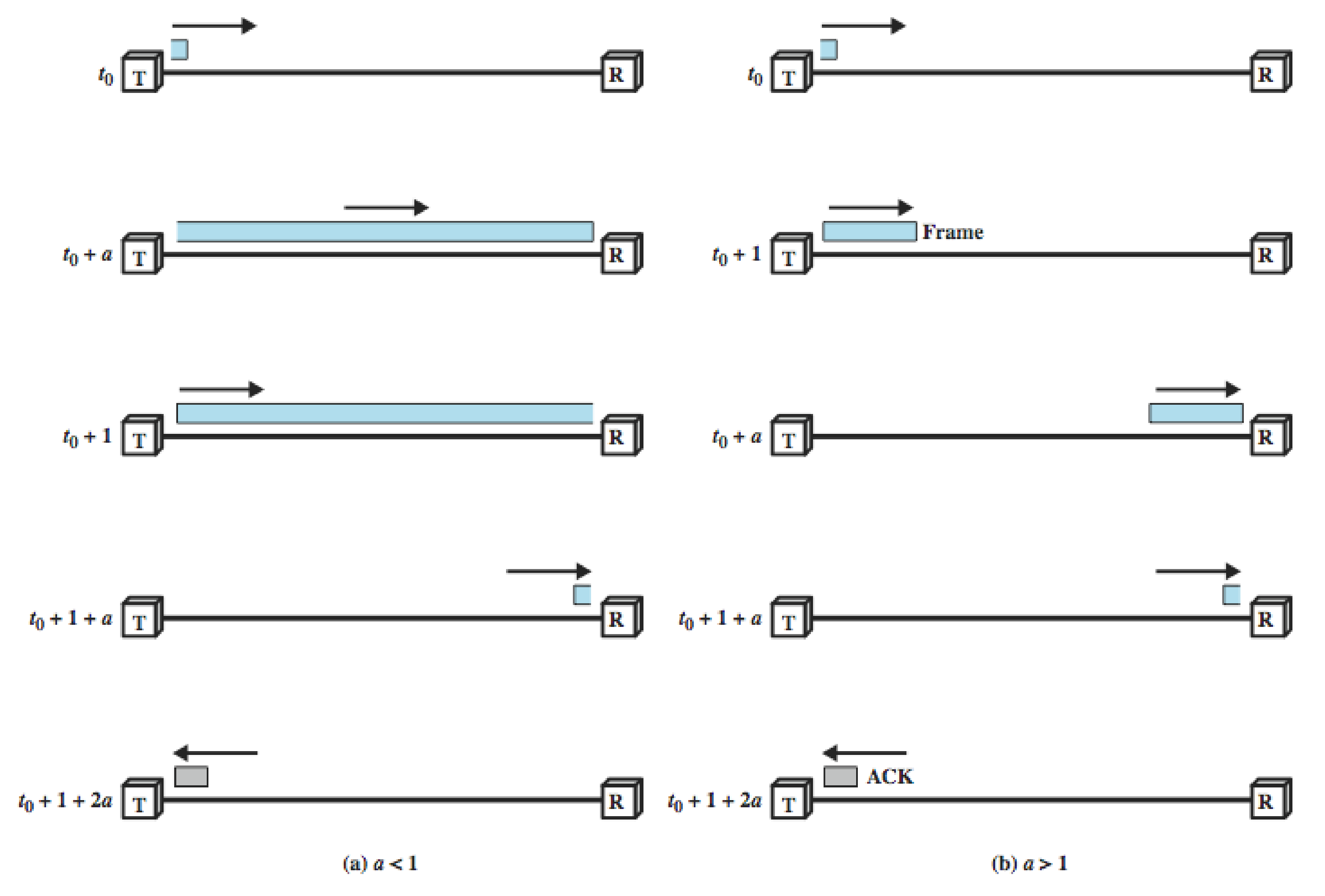
(a) Error-free transmission

(b) Transmission with losses and errors

Stop and Wait

- source transmits frame
- destination receives frame and replies with acknowledgement (ACK)
- source waits for ACK before sending next
- destination can stop flow by not send ACK
- works well for a few large frames
- Stop and wait becomes inadequate if large block of data is split into small frames

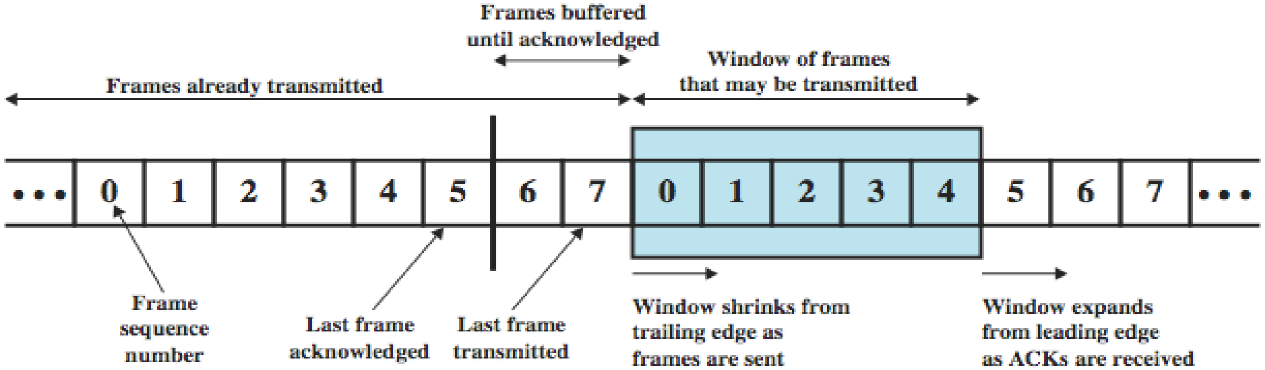
Stop and Wait Link Utilization



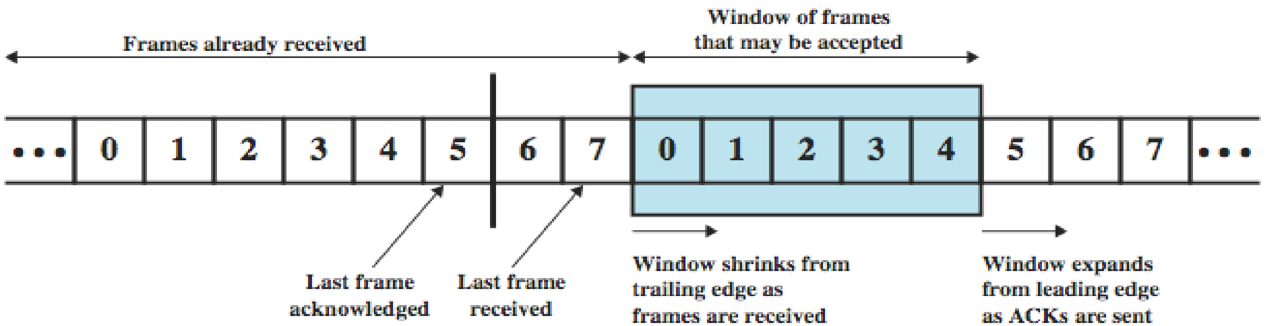
Sliding Windows Flow Control

- allows multiple numbered frames to be in transit
- receiver has buffer W long
- transmitter sends up to W frames without ACK
- ACK includes number of next frame expected
- sequence number is bounded by size of field (k)
  - frames are numbered modulo 2^k
  - giving max window size of up to $2^k - 1$
- receiver can ack frames without permitting further transmission (Receive Not Ready)
- must send a normal acknowledge to resume
- if have full-duplex link, can piggyback ACKs

Sliding Window Diagram

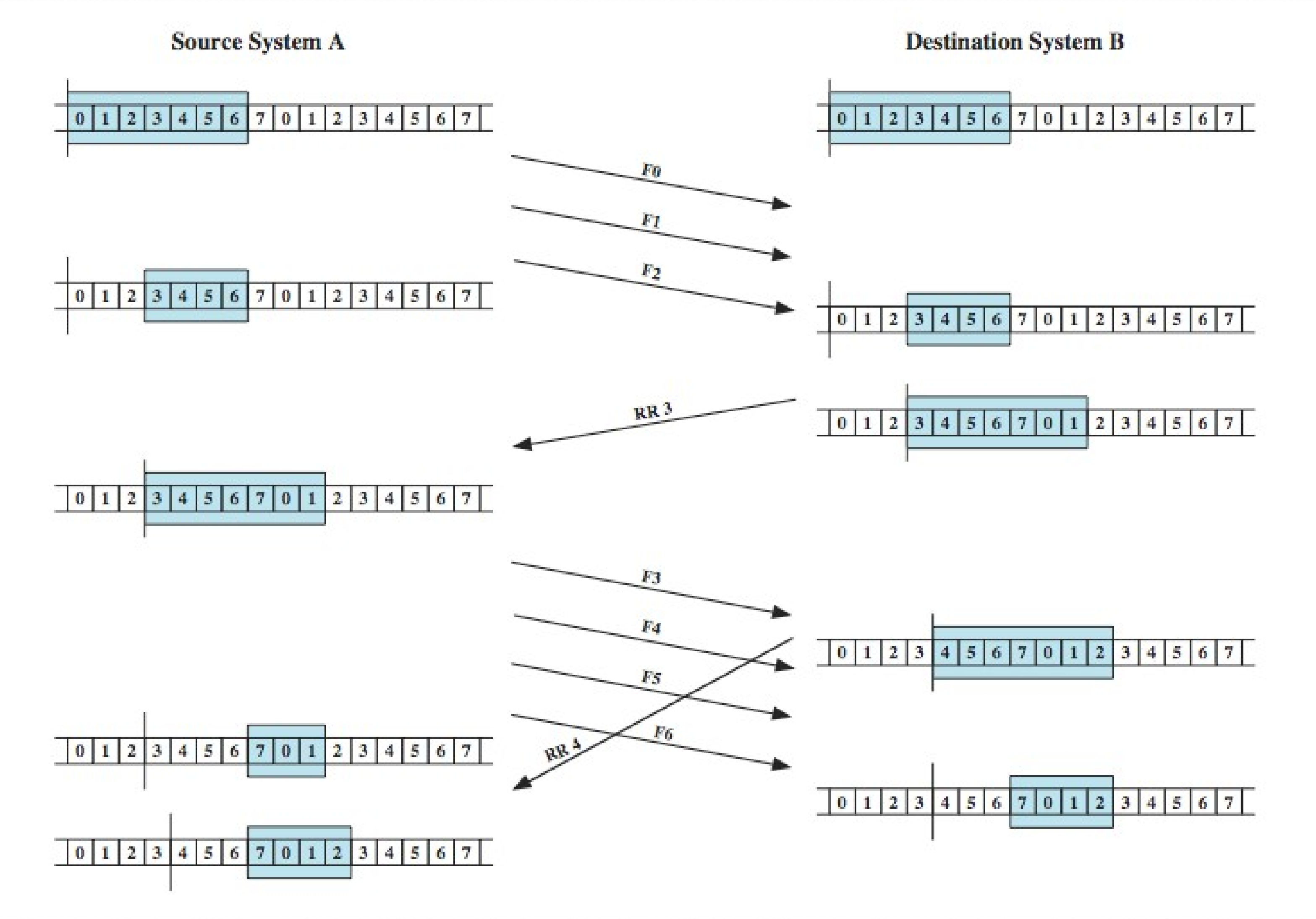


(a) Sender's perspective



(b) Receiver's perspective

Sliding Window Example



Error Control

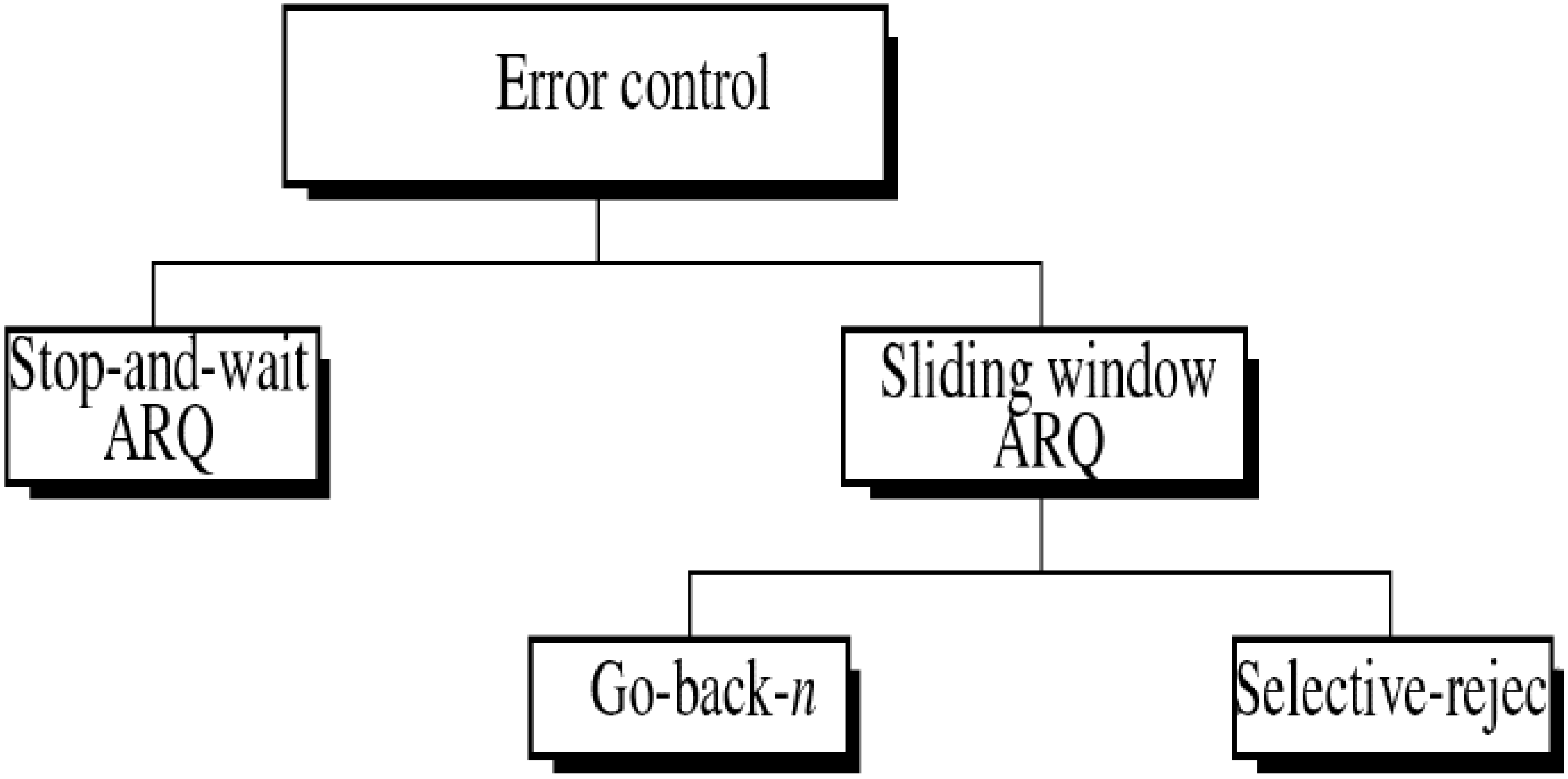
- detection and correction of errors such as:
  - lost frames
  - damaged frames
- common techniques used:
  - error detection
  - positive acknowledgment
  - retransmission after timeout
  - negative acknowledgement & retransmission

Collectively these techniques are referred to as **Automatic Repeat Request (ARQ)**

Automatic Repeat Request (ARQ)

- collective name for such error control mechanisms, including:
- stop and wait
- go back N
- selective reject (selective retransmission)

Automatic Repeat Request

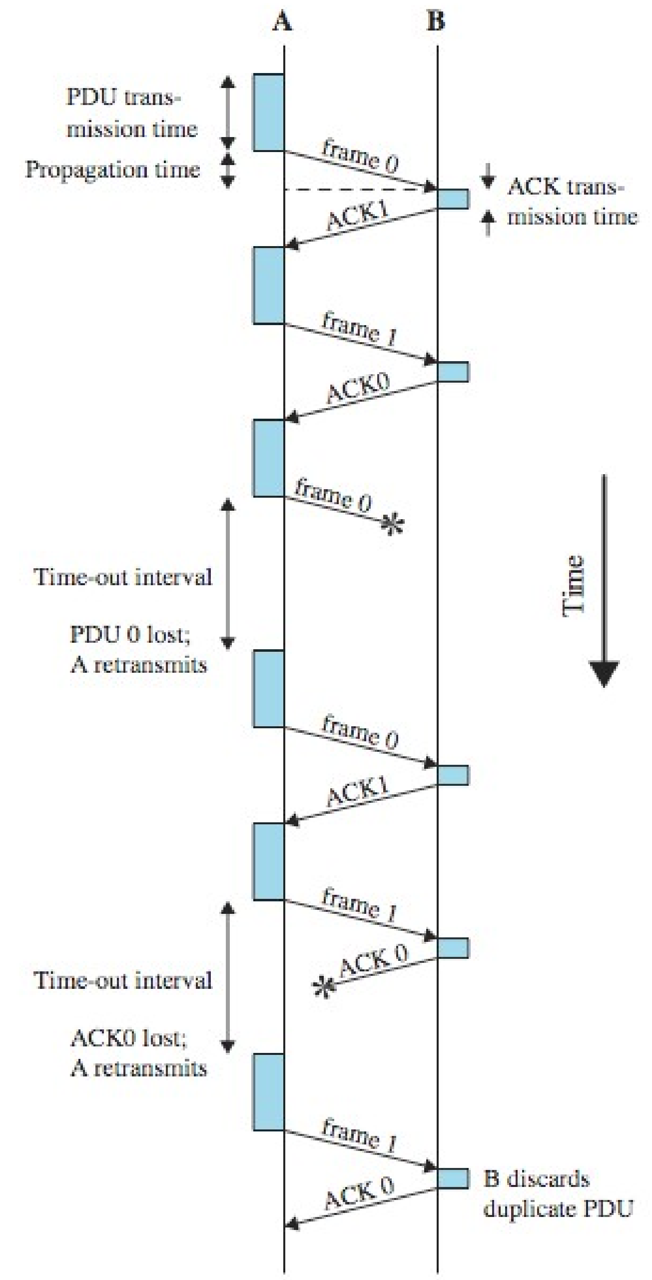


Stop and Wait

- source transmits single frame
- wait for ACK
- if received frame damaged, discard it
  - transmitter has timeout
  - if no ACK within timeout, retransmit
- if ACK damaged,transmitter will not recognize it
  - transmitter will retransmit
  - receive gets two copies of frame
  - use alternate numbering and ACK0 / ACK1

Stop and Wait

- see example with both types of errors
- pros and cons
  - simple
  - inefficient



Go Back N

- based on sliding window
- if no error, ACK as usual
- use window to control number of outstanding frames
- if error, reply with rejection
  - discard that frame and all future frames until error frame received correctly
  - transmitter must go back and retransmit that frame and all subsequent frames

Go Back N - Handling

- Damaged Frame
  - error in frame *i* so receiver rejects frame *i*
  - transmitter retransmits frames from *i*
- Lost Frame
  - frame *i* lost and either
    - transmitter sends *i+1* and receiver gets frame *i+1* out of seq and rejects frame *i*
    - or transmitter times out and send ACK with P bit set which receiver responds to with ACK *i*
  - transmitter then retransmits frames from *i*

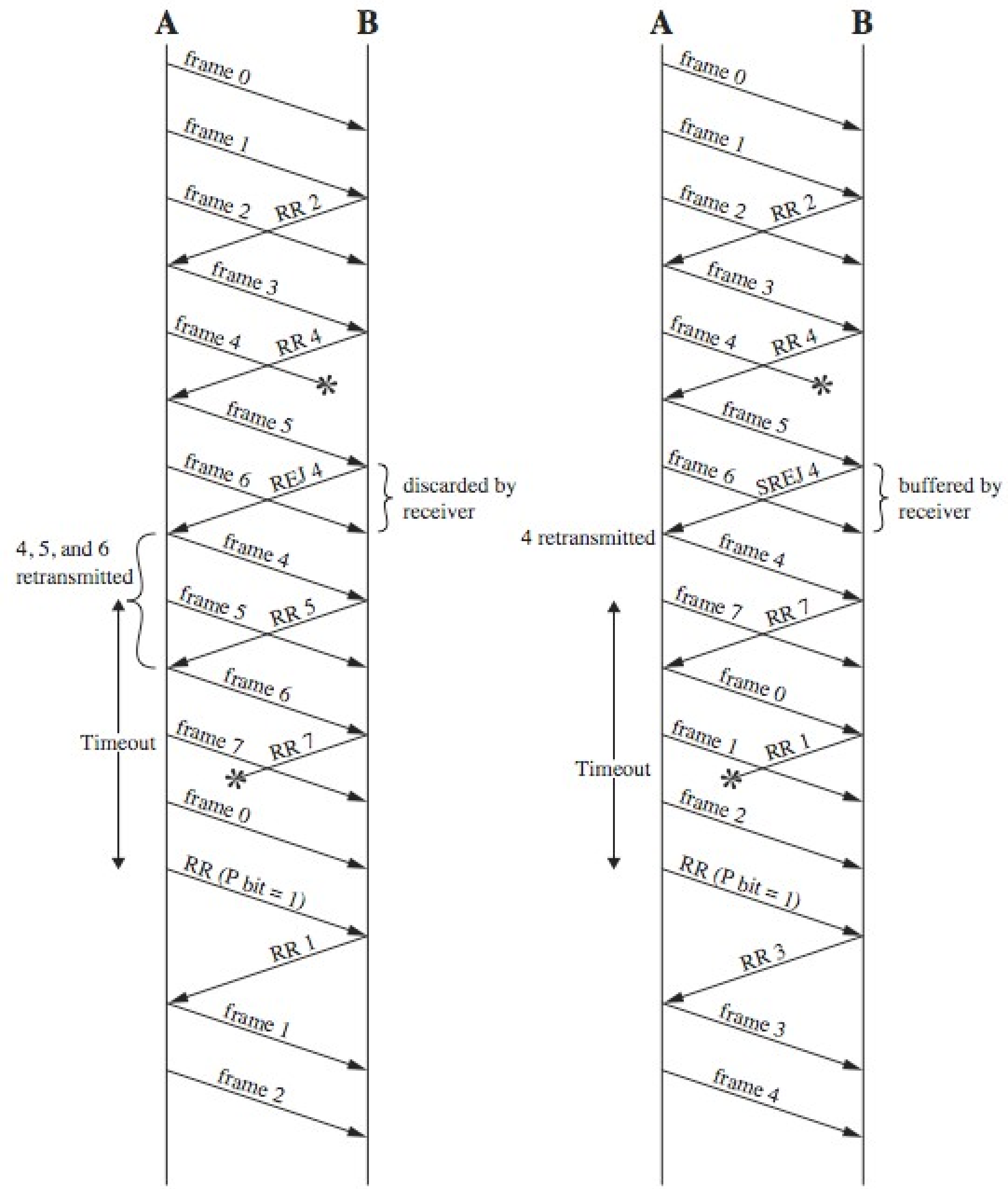
Go Back N - Handling

- Damaged Acknowledgement
  - receiver gets frame *i*, sends ack (*i+1*) which is lost
  - acks are cumulative, so next ack (*i+n*) may arrive before transmitter times out on frame *i*
  - if transmitter times out, it sends ack with P bit set
  - can be repeated a number of times before a reset procedure is initiated
- Damaged Rejection
  - reject for damaged frame is lost
  - handled as for lost frame when transmitter times out

Selective Reject

- also called selective retransmission
- only rejected frames are retransmitted
- subsequent frames are accepted by the receiver and buffered
- minimizes retransmission
- receiver must maintain large enough buffer
- more complex logic in transmitter
- hence less widely used
- useful for satellite links with long propagation delays

Go Back N vs Selective Reject



(a) Go-back-N ARQ

(b) Selective-reject ARQ

Summary

- introduced need for data link protocols
- flow control
- error control