Data Link Control and Protocols

Data Link Control

- Flow Control
- Error Control

Flow Control

- how much data sender can transmit before receiving the ack
- Why flow control?

  →

  Limitation with receiver

  1.Processing speed

  2. Limited memory to store incoming data

Flow Control

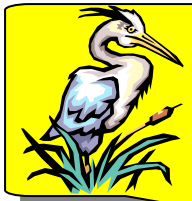
Flow control refers to a set of procedures used to restrict the amount of data that the sender can send before waiting for acknowledgment.

Error Control

- Error Detection + Error Correction
- Otherwise

  Error Detection + Retransmission
- ARQ

  - any time, an error is discovered in an exchange, specified frames are retransmitted



Error Control

Error control in the data link layer is based on automatic repeat request, which is the retransmission of data.

Flow and Error Control Mechanisms

- Stop and Wait ARQ
- Go-Back ARQ
- Selective Repeat ARQ

Stop-and-Wait Automatic Repeat reQuest

- simplest flow and error control mechanism
- the sending device keeps a copy of the last frame transmitted until it receives an acknowledgement
- Frames - alternately numbered as 0 and 1
- Ack for frame0 = ACK 1 and for frame1= ACK0
- Out of order frames and erroneous frames are discarded and no ack is sent
- Timers
- Control Variables

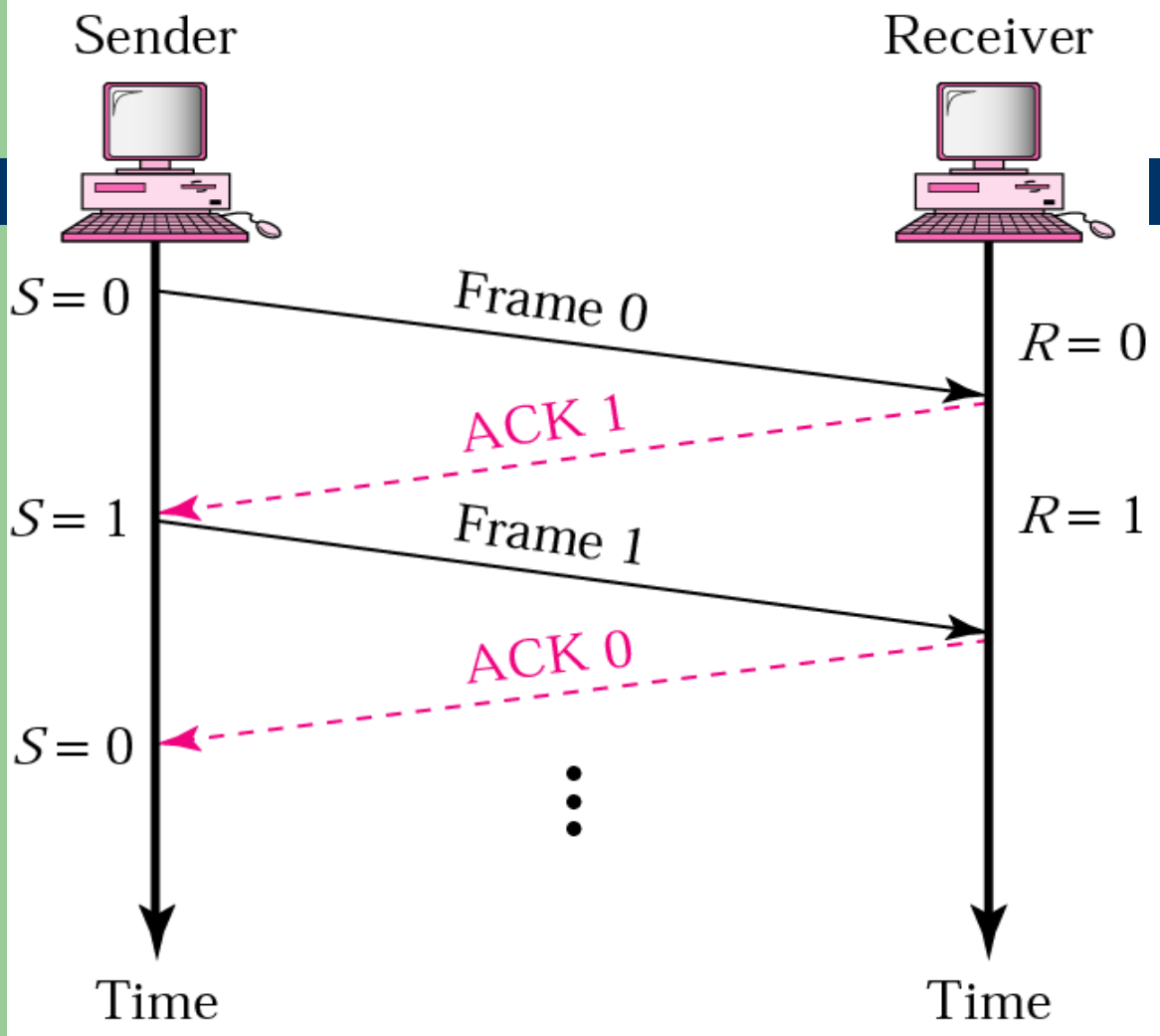
  sender – S (no of recently sent frame)

  receiver – R (no of next frame expected)

A simplex Stop and Wait ARQ

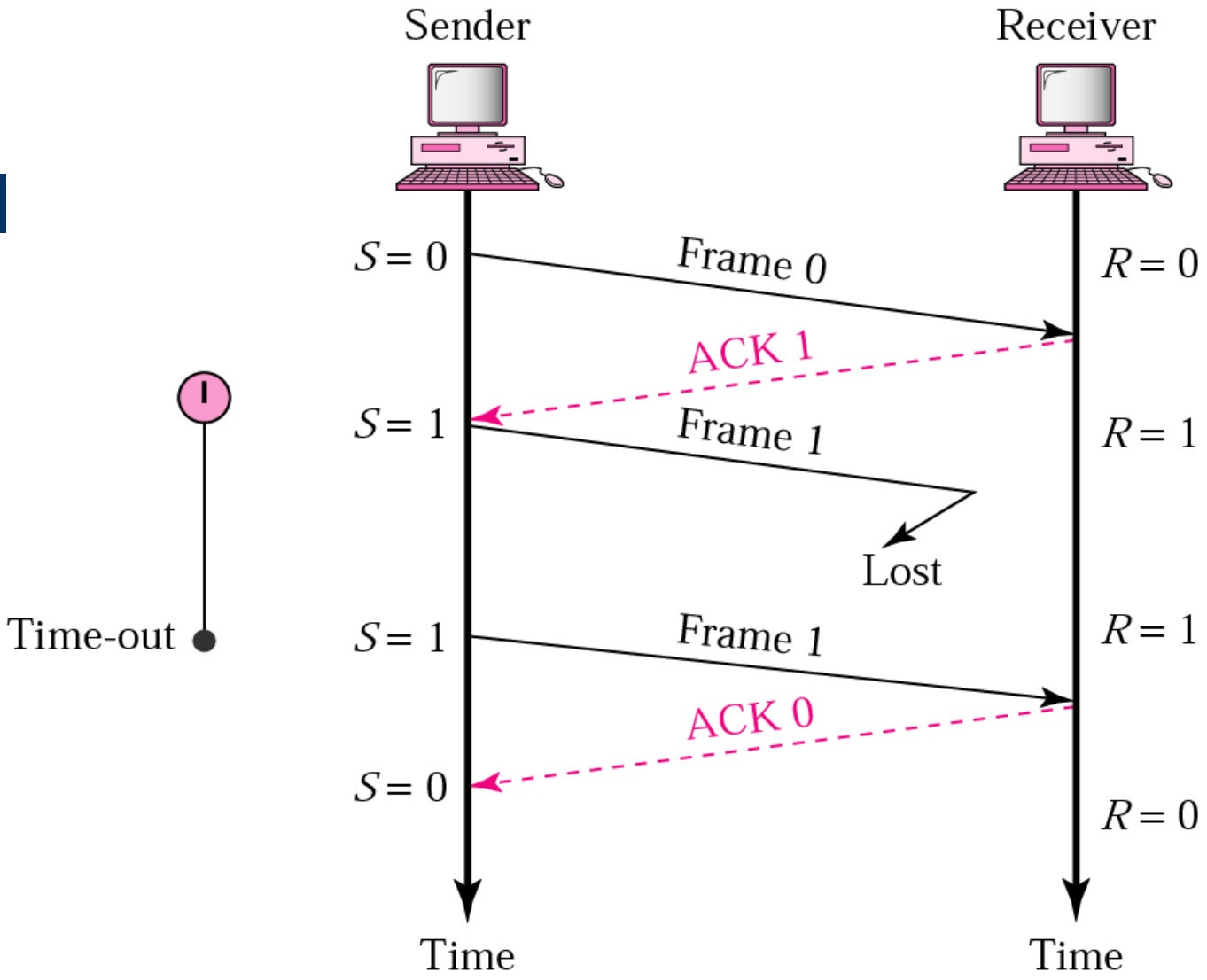
- Normal Operation
- Frame lost
- Acknowledgement lost
- Acknowledgement delayed

Normal operation

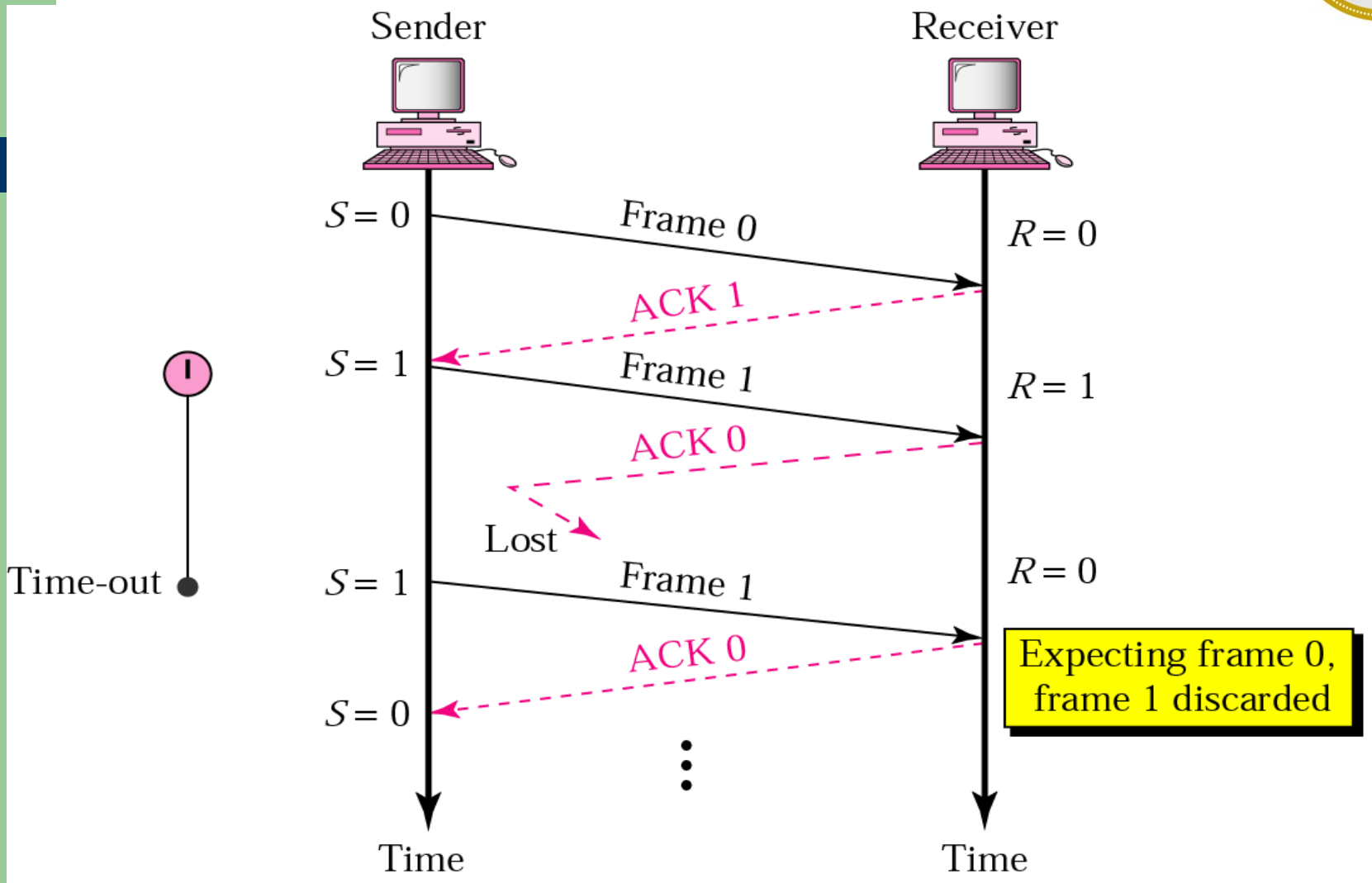


- The sender will not send the next piece of data until it is sure that the current one is correctly received

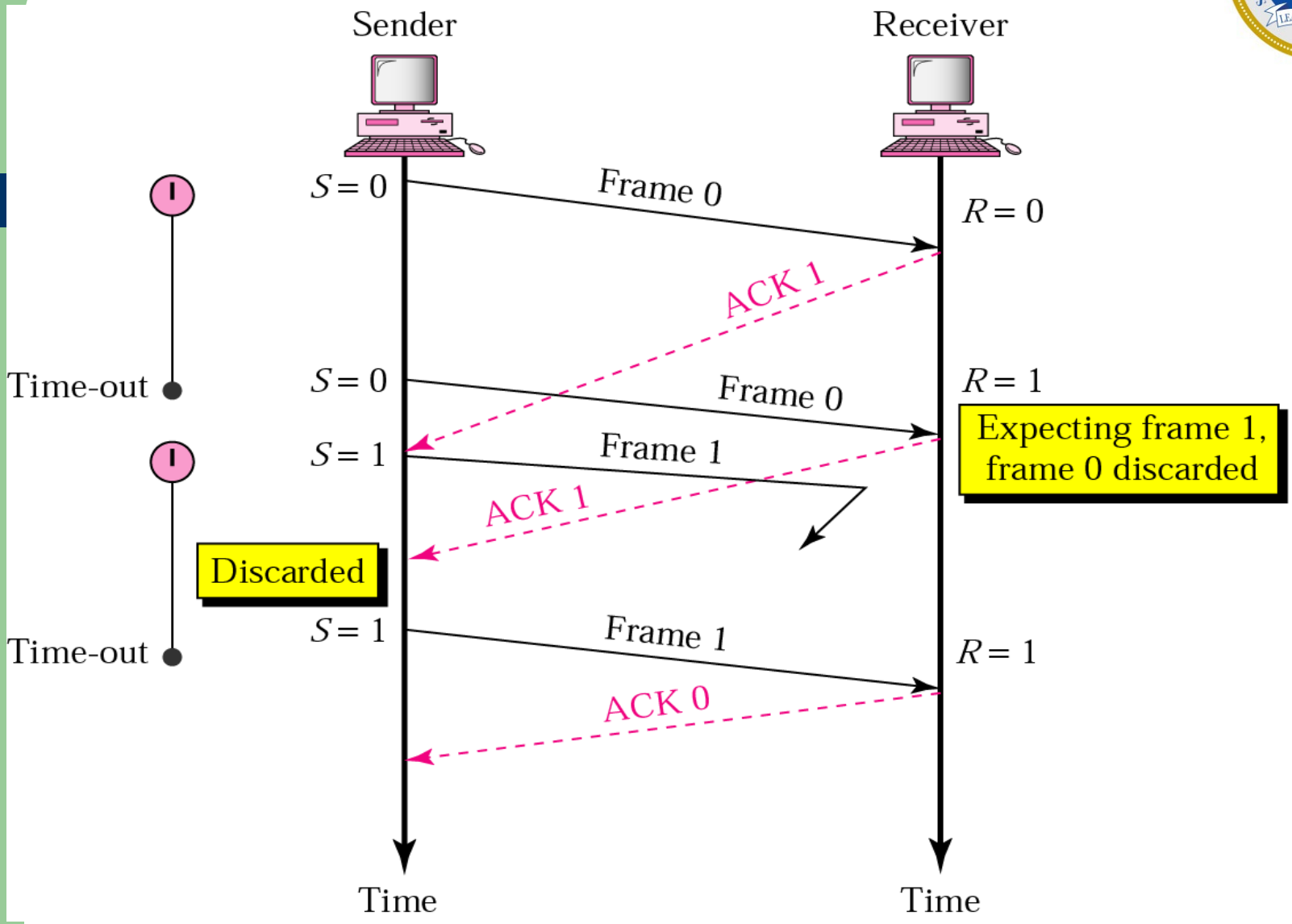
Stop-and-Wait ARQ, lost frame



Stop-and-Wait ARQ, lost ACK frame



Stop-and-Wait ARQ, delayed ACK

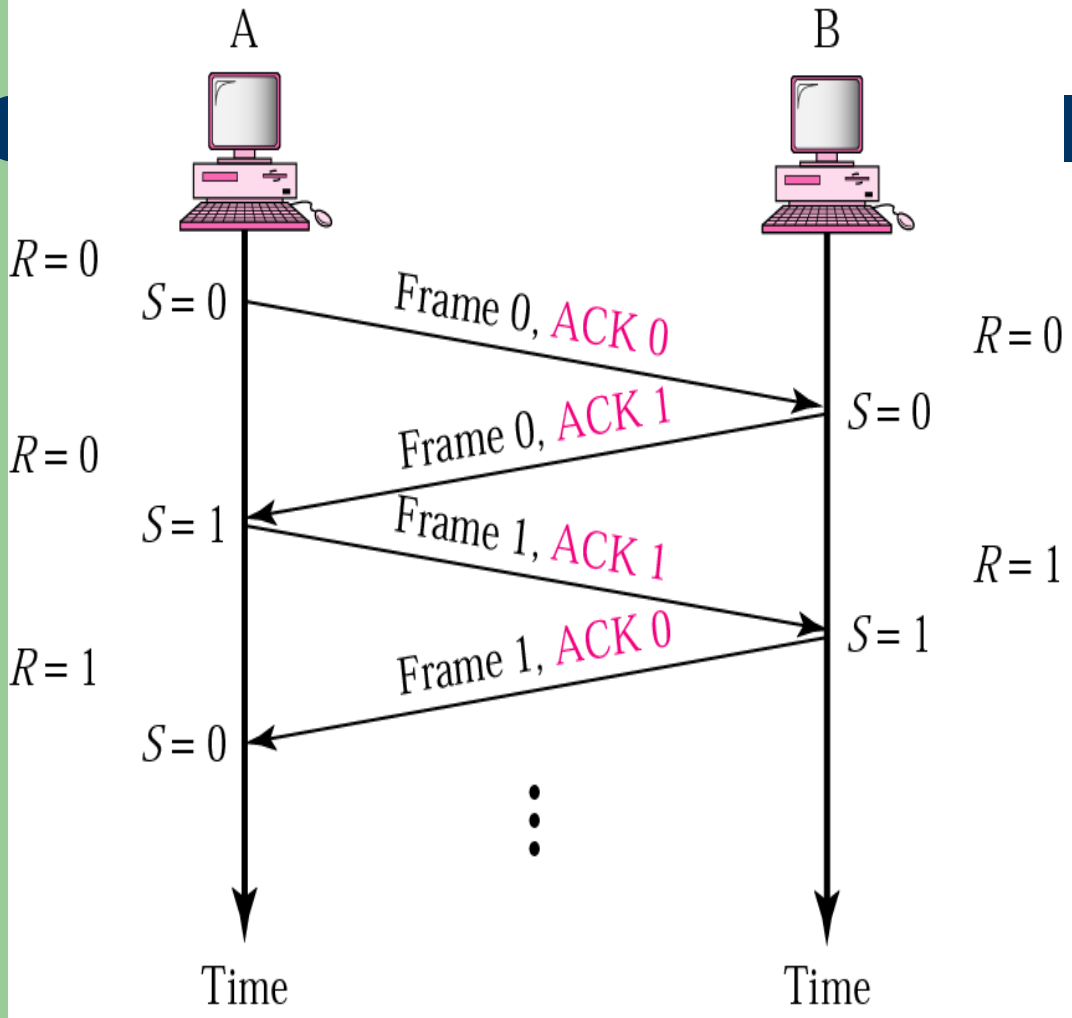


Why numbering frames?

In Stop-and-Wait ARQ,

1. ***numbering frames prevents the retaining of duplicate frames.***
2. ***Numbered acknowledgements are needed in case of delayed ack and next frame lost.***

Bidirectional Transmission



- Each party should maintain S and R to track frames sent and expected.
- Piggybacking (hooking ack with next outgoing data frame) can be used to save bandwidth.

Drawbacks of stop and wait

- Only one frame can be in transit at a time
- after each frame sent the host must wait for an ACK
  - inefficient use of bandwidth

- to improve efficiency, multiple frames can be sent before receiving acknowledgement

- **Alternatives: Sliding Window protocols**

  - one task begins before the other one ends (concept of pipelining)

  -increases efficiency in transmission

Sliding Window Protocols

- Sliding window
  - Holds the unacknowledged outstanding frames in sender
  - Holds the expected frames in receiver

- Sequence numbers
  - sent frames are numbered sequentially
  - if the number of bits in the header is *m* then sequence number goes from 0 to $2^m - 1$

- *Protocols*
  1. *Go back – N*
  2. *Selective Repeat*

Go Back - N

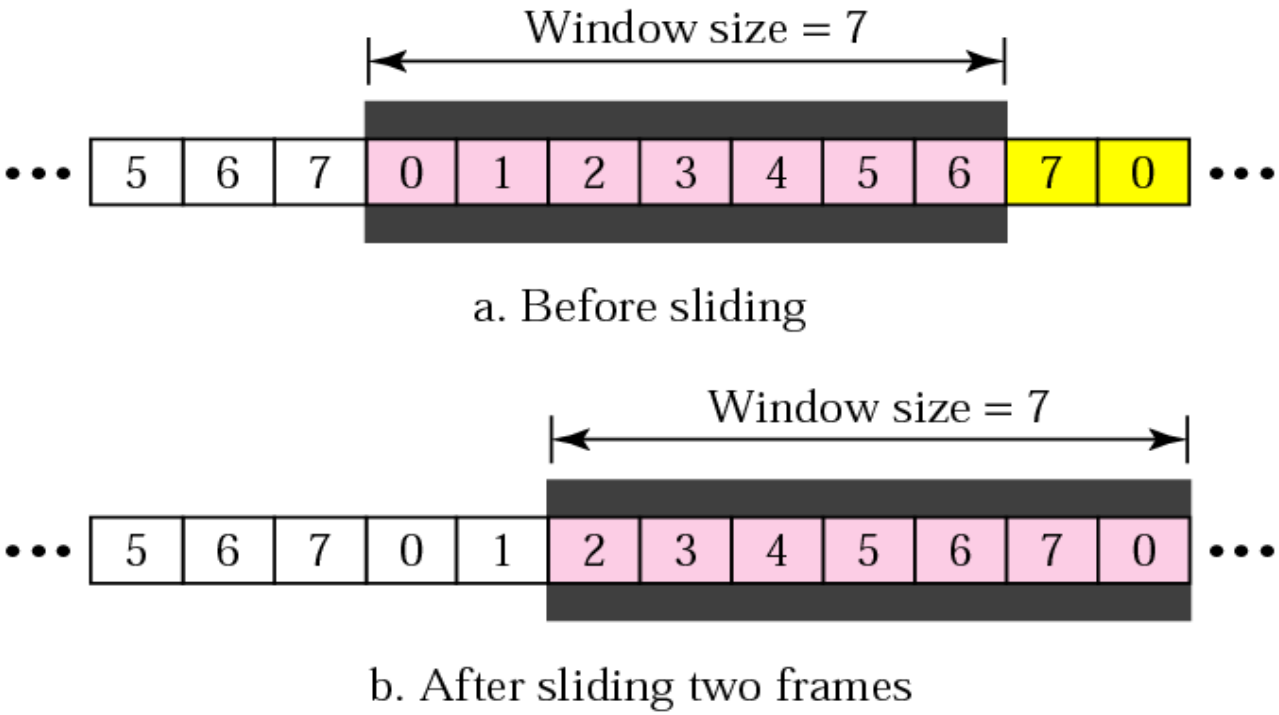
- Sender window size < 2^m
- Receiver window size = 1
- Why the name go back- N?
  - when the frame is damaged the sender **goes back** and sends a set of frames starting from the last one ACKn'd
  - the number of retransmitted frames is ***N***

Example:

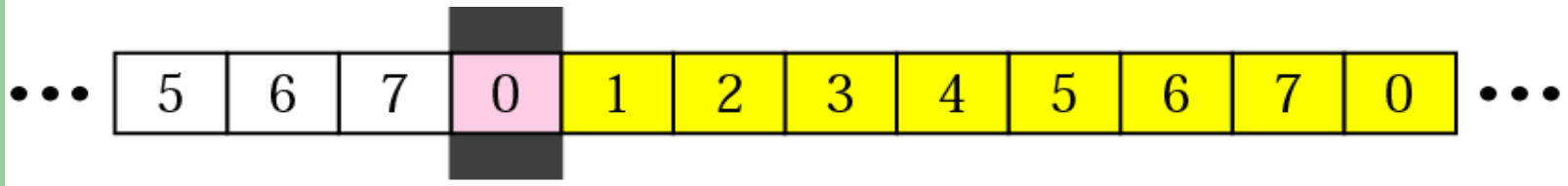
The window size is 4.

A sender has sent frame 6 and the timer expires for frame 3 (frame 3 not ACKn'd). The sender goes back and re-sends the frames 3, 4, 5 and 6.

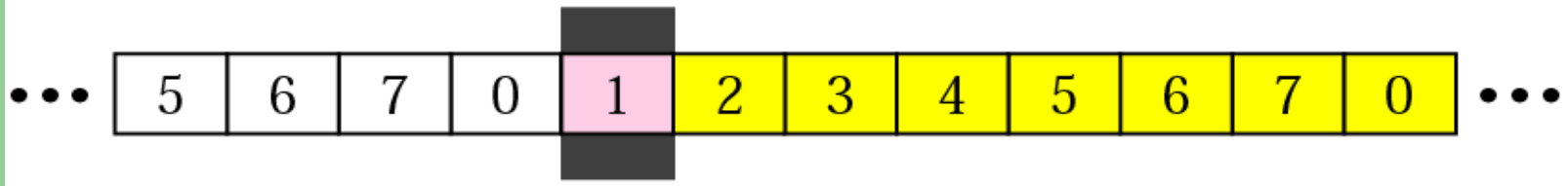
Sender sliding window



Receiver sliding window



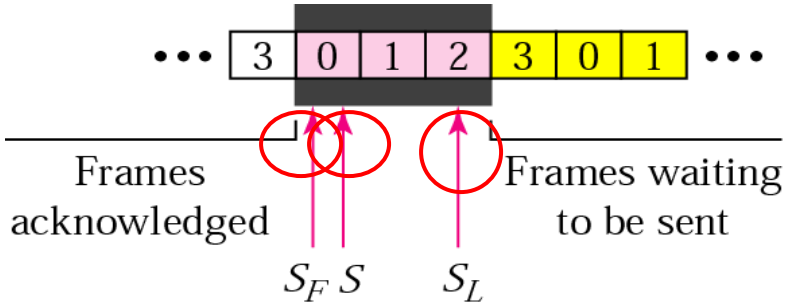
a. Before sliding



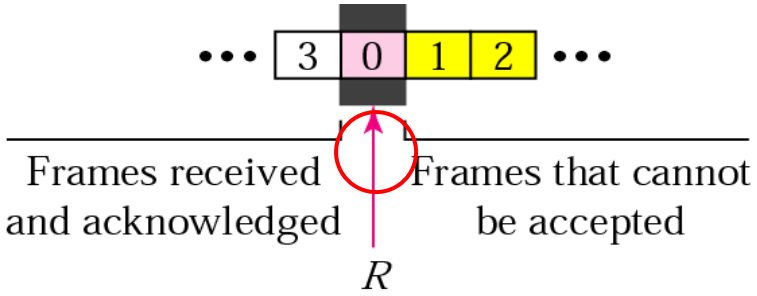
b. After sliding

Go-back-*N* : Control variables-

- *S*- holds the sequence number of the recently sent frame
- *SF* – holds sequence number of the first frame in the window
- *SL* – holds the sequence number of the last frame
- *R* – sequence number of the frame expected to be received

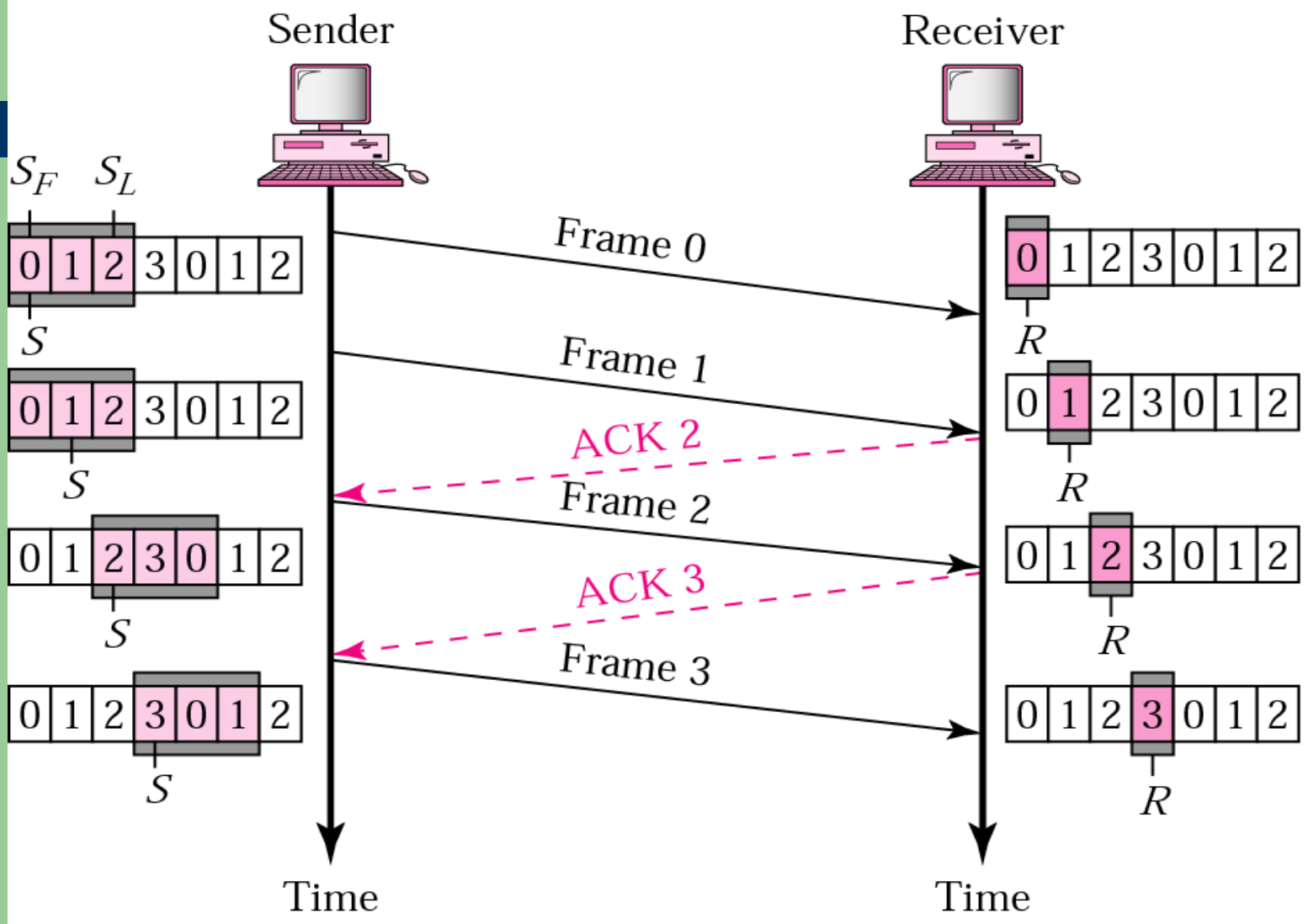


a. Sender window

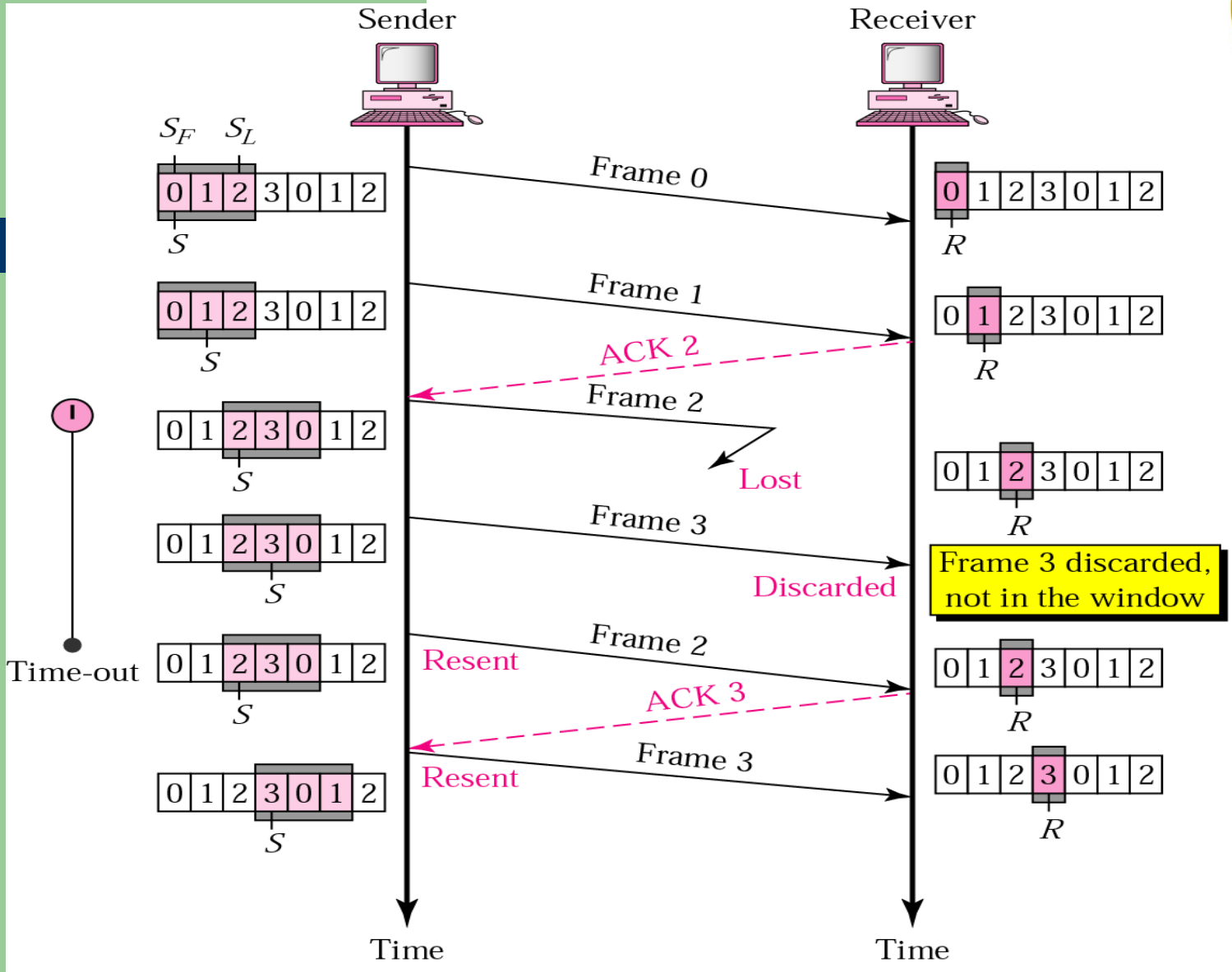


b. Receiver window

Go-Back-N ARQ, normal operation



Go-Back-N ARQ, lost frame



Try for (go back N)

- Damaged or lost ACK

  case 1 : next ack arrives before timer expires

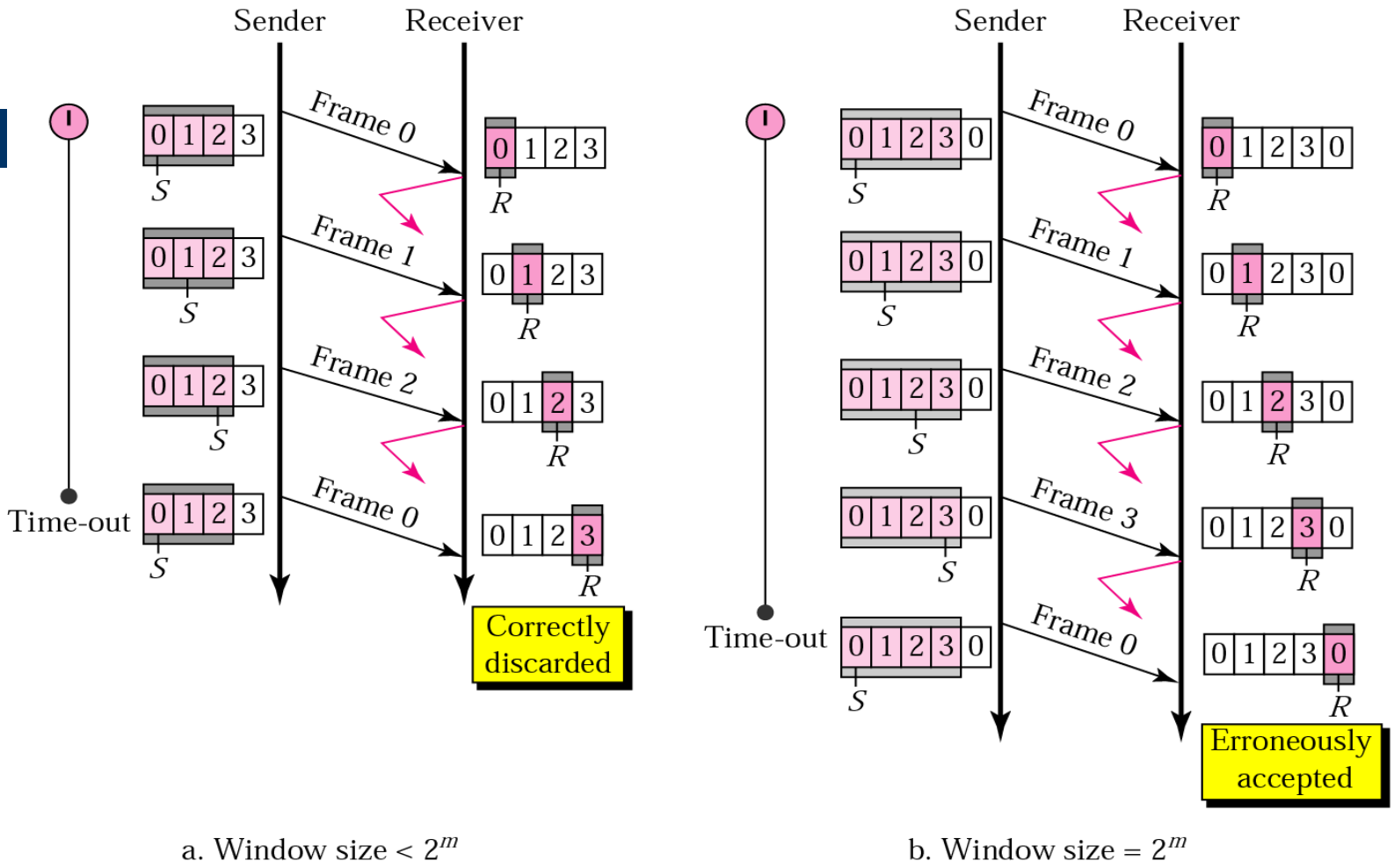
  case 2 : Next ack arrives after timer expires

- Delayed Ack

Note:

In Go-Back-N ARQ, the size of the sender window must be less than 2^m; the size of the receiver window is always 1.

Go-Back-N ARQ: sender window size



a. Window size < 2^m

b. Window size = 2^m

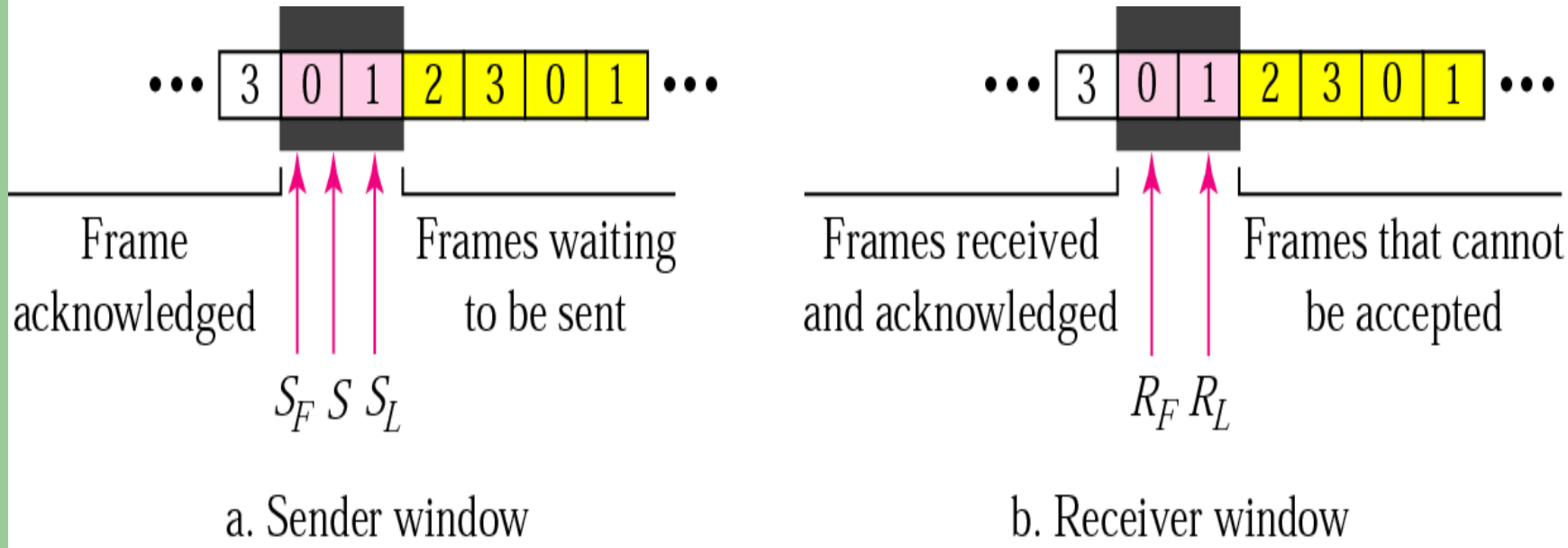
Drawbacks of Go-back-*N*

- Inefficient
  - all out of order received packets are discarded (recieiver side is simplified)
- This is a problem in a noisy link
  - many frames must be retransmitted -> bandwidth consuming
- Solution
  - re-send only the damaged frames
- Selective Repeat ARQ
  - avoid unnecessary retransmissions

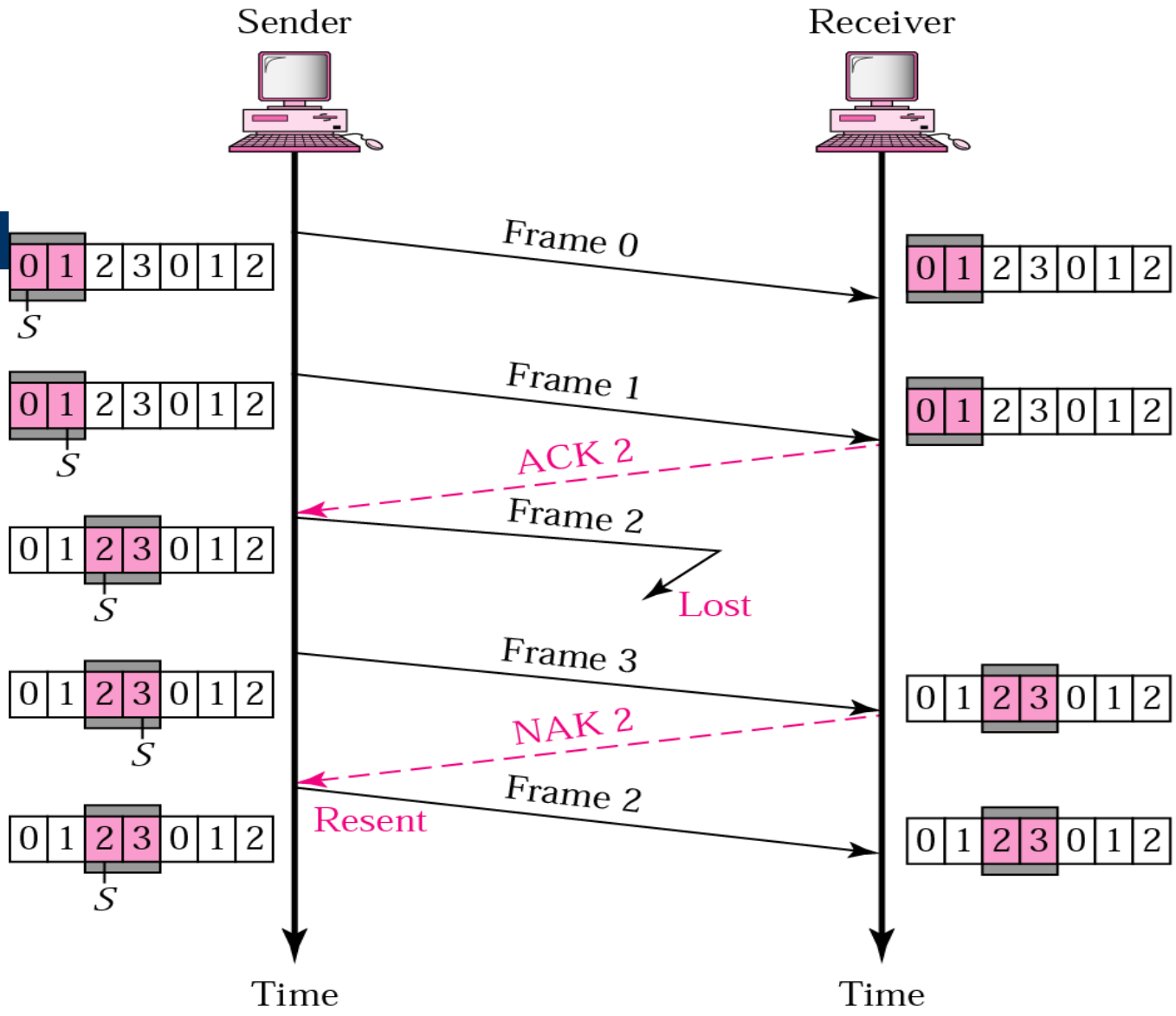
Selective Repeat ARQ

- Processing at the receiver more complex
- The window size is reduced to $2^{(m-1)}$ ($2^m/2$ at most)
- Both the transmitter and the receiver have the same window size
- Receiver expects frames within the range of the sequence numbers
- Negative acknowledgement

Selective Repeat ARQ - sender and receiver windows



Selective Repeat ARQ, lost frame



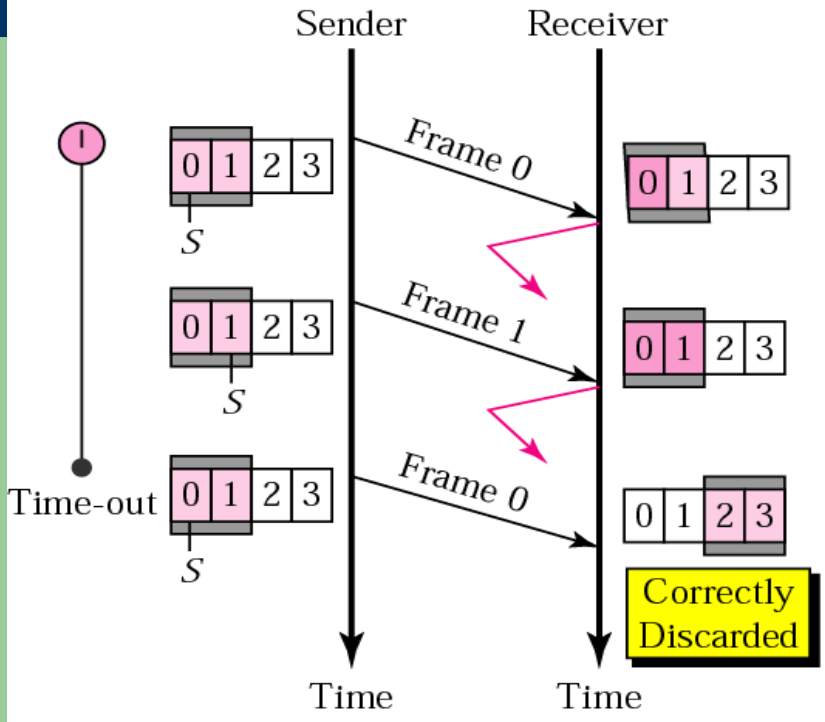
Try for – selective repeat

- Lost and delayed ACKs
- Bidirectional transmission (both side requires both sending and receiving windows)

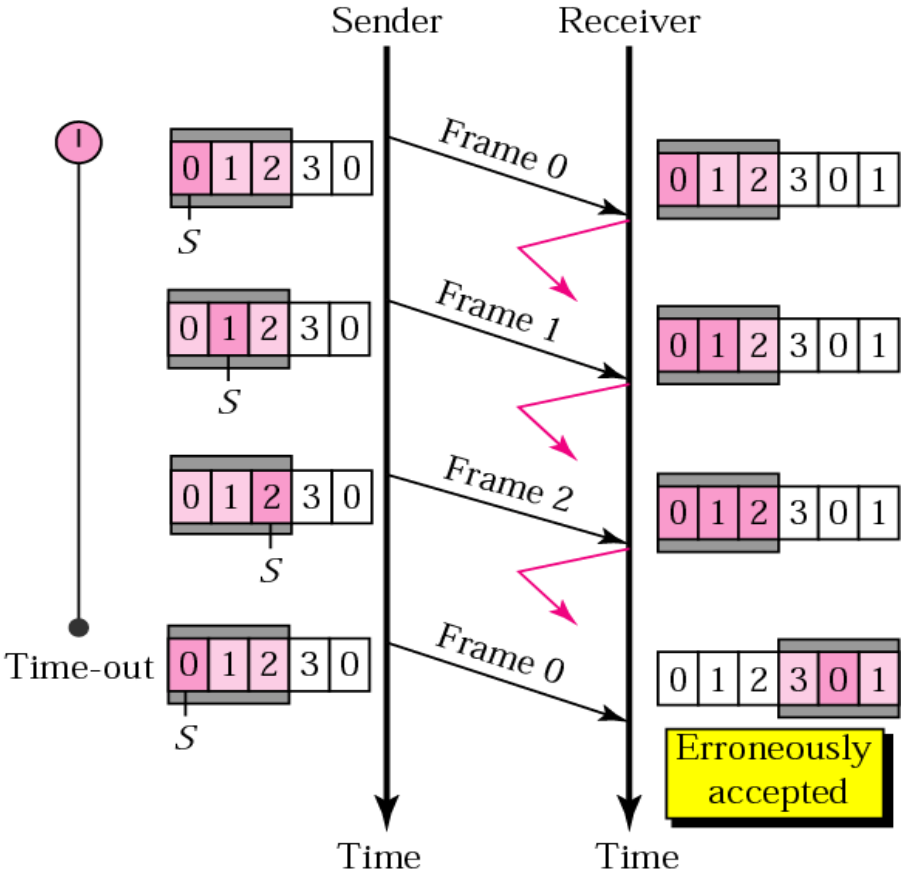
Note:

In Selective Repeat ARQ, the size of the sender and receiver window must be at most one-half of 2^m.

Selective Repeat ARQ, sender window size



a. Window size = 2^{m-1}



b. Window size > 2^{m-1}

Bandwidth – delay product

- A measure of efficiency of ARQ system

  = bandwidth(bits per second) * round-trip delay (in seconds)

- It is the measure of number of bits we can send out of our system while waiting for news from the receiver.

Example1:

- System : Stop and wait ARQ
- Bandwidth : 1Mbps
- Round trip for one bit : 20ms
- Frame length : 1000 bits
- **Utilization percentage of the link** = ?

Soln

Bandwidth – delay product = $1 * 10^6 * 20 * 10^{-3} = 20,000$ bits

So, the sender can send 20000 bits before it receives the ack.

But the system is stop and wait . So Only one frame (1000 bits) is sent at a time.

Hence, the link utilization = 20000 / 1000 = **5%**

Exercise

System : Go Back N with 15 frame sequence
Bandwidth : 1Mbps
Round trip for one bit : 20ms
Frame length : 1000 bits

Utilization percentage of the link = ?

Thank You
Queries?