Flow Control and Error Control

Error Control

- Network is responsible for transmission of data from one device to another device.
- Data can be corrupted during transmission.
- With the error control process, it is ensured that the transmitted and received data are identical.
- For reliable communication, error must be detected and corrected.

Flow Control

- Flow Control is one important design issue for the Data Link Layer that controls the flow of data between sender and receiver.
- Flow control refers to a set of procedures used to restrict the amount of data that the sender can send before waiting for an ack.
- Error control and flow control in data link layer is done by using Automatic Repeat Request (ARQ) mechanism.

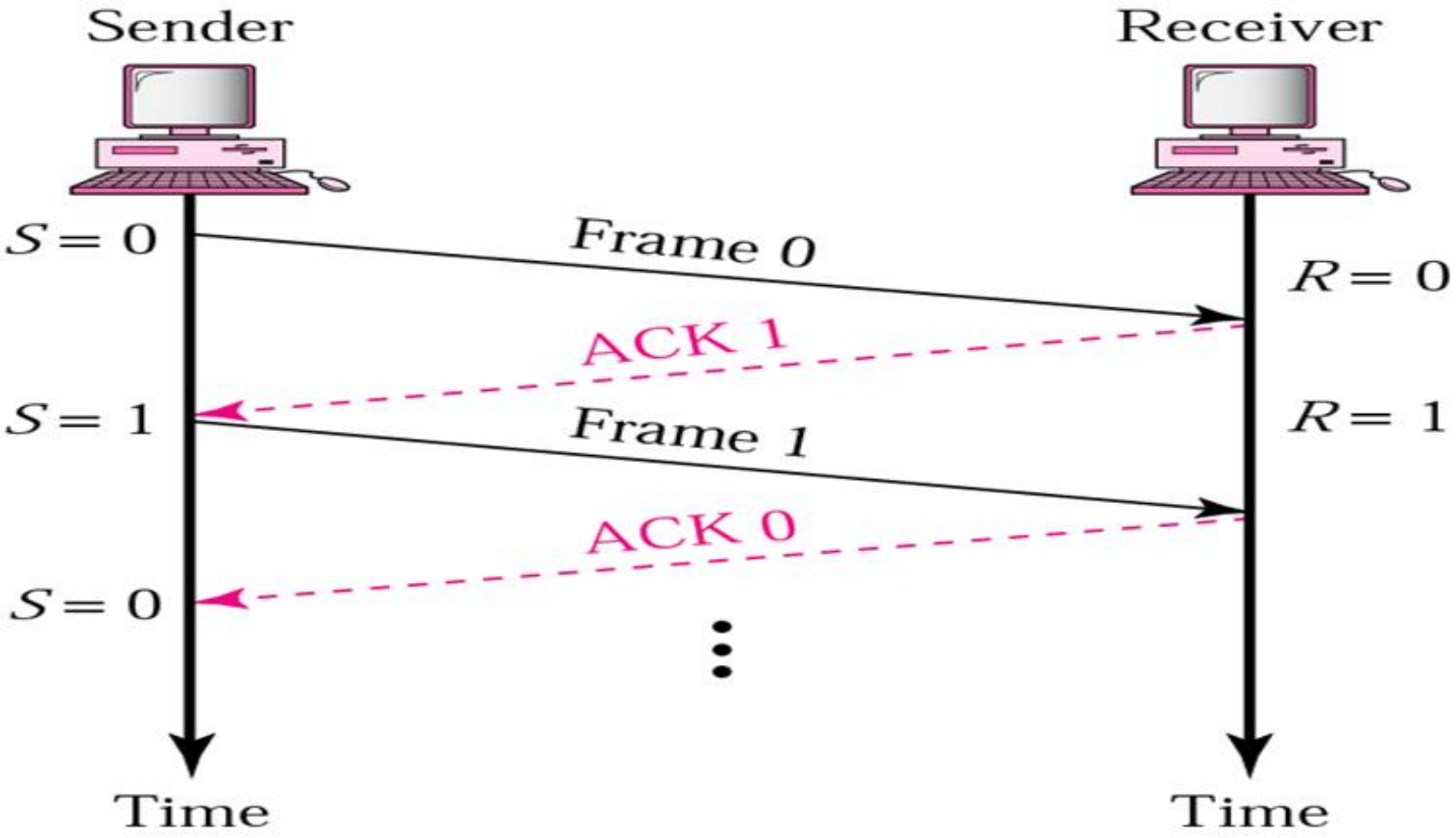
Automatic Repeat Request(ARQ)

- Once an error is detected in a transmission, the specified frames are retransmitted.
  - Errors → data lost, damaged frames, lost acknowledgment
- Types of ARQ
  - Stop-and-Wait Protocol
  - Sliding Window Protocol
    - Go Back N ARQ
    - Selective Repeat ARQ

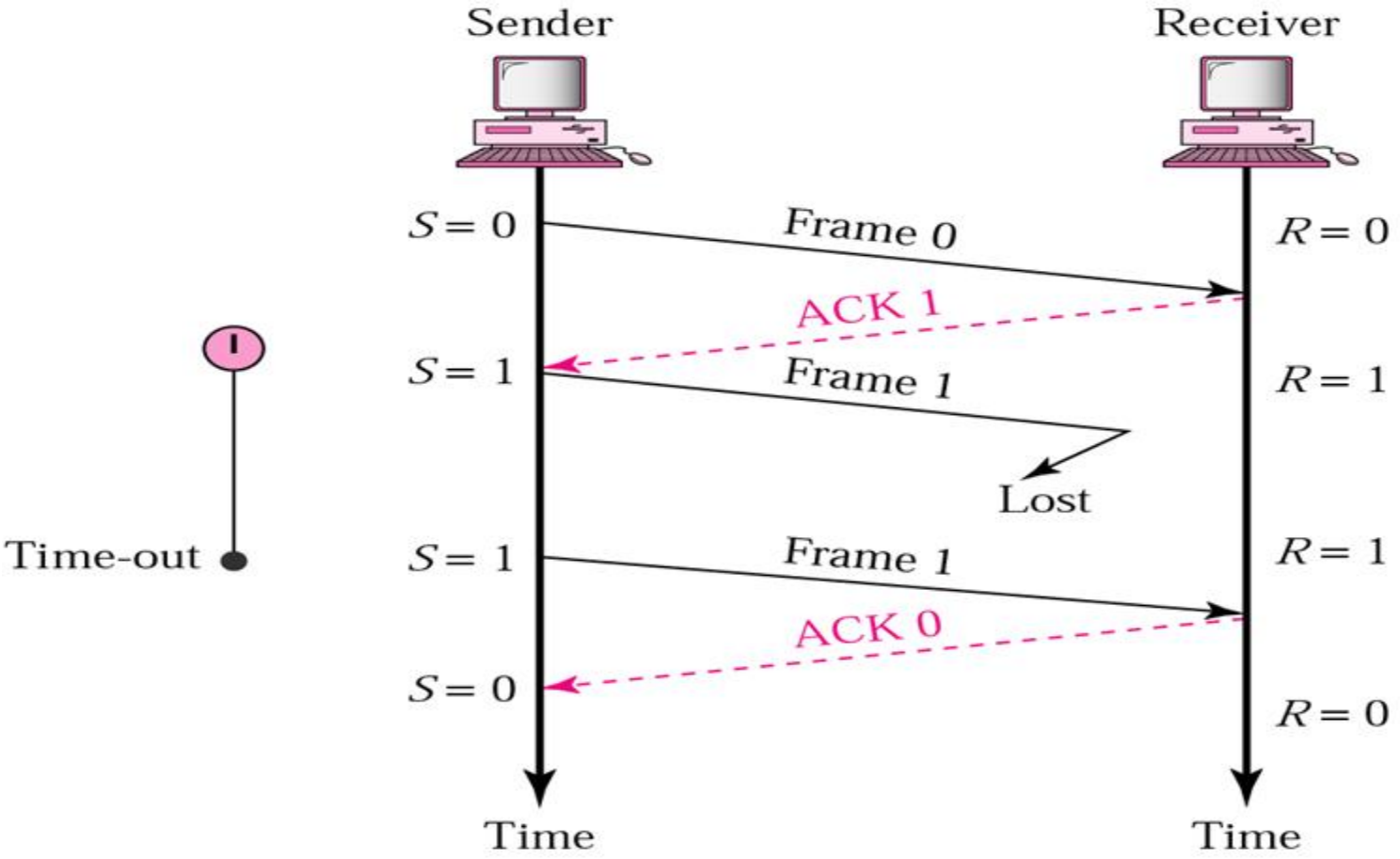
Stop and Wait ARQ

- Sender sends a frame, waits for an ack, after receiving the ack the sender sends the next frame, waits for an ack and so on…
- Frames are numbered as 0,1.
- Operations
  - Normal Operation
  - Lost or Damaged Frame
  - Lost Ack
  - Delayed Ack

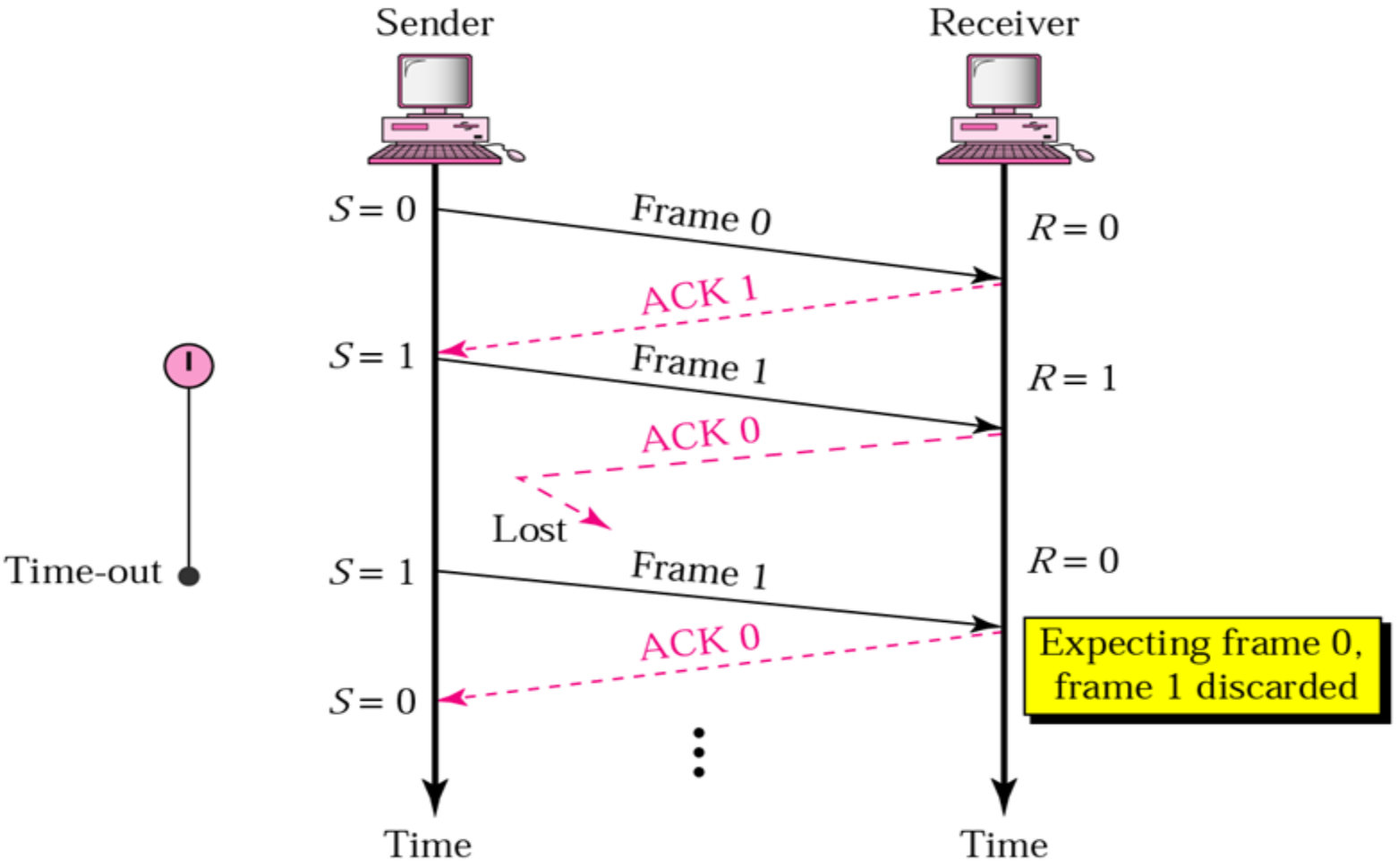
Normal Operation



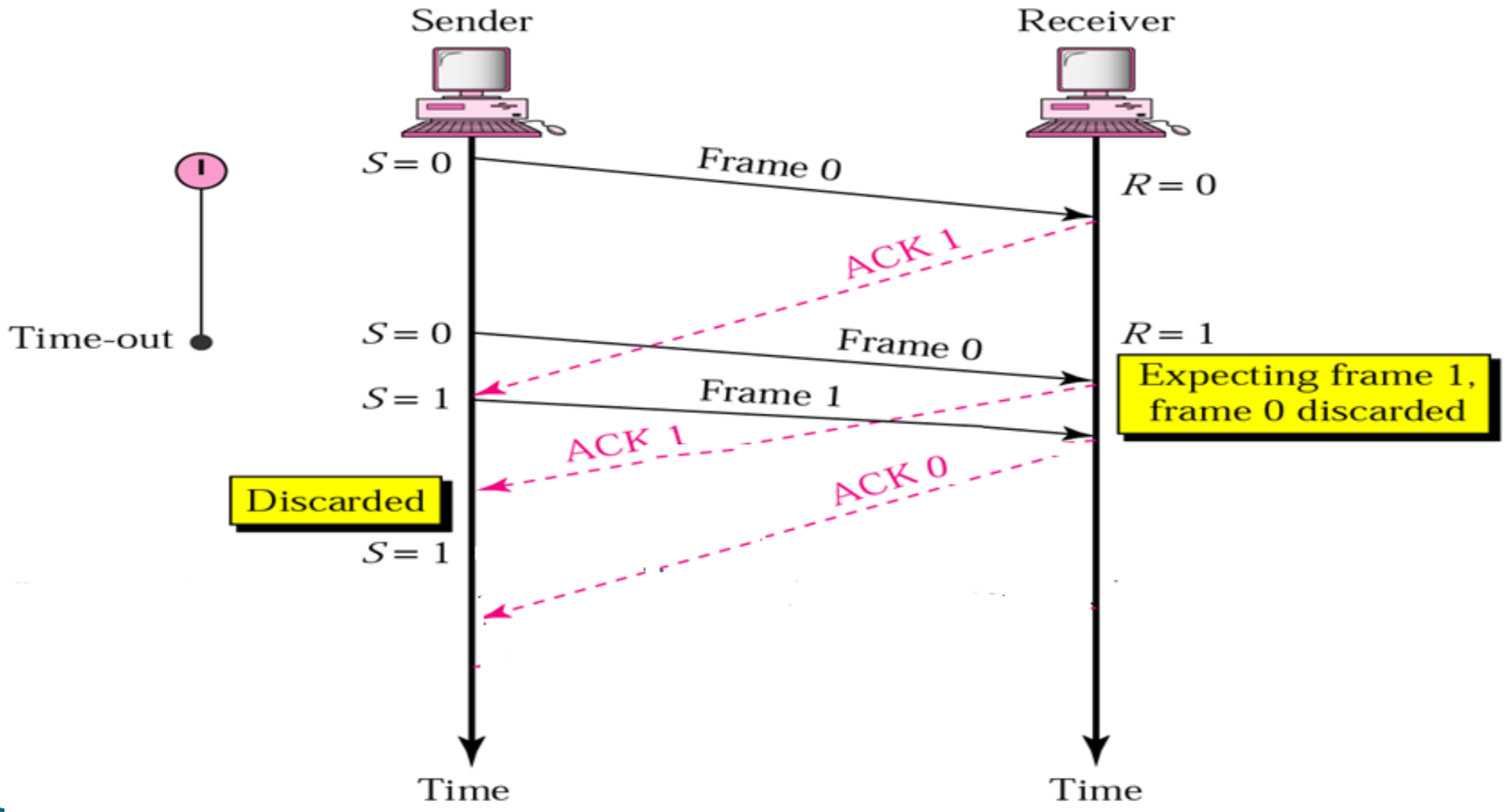
Lost or Damaged Frame



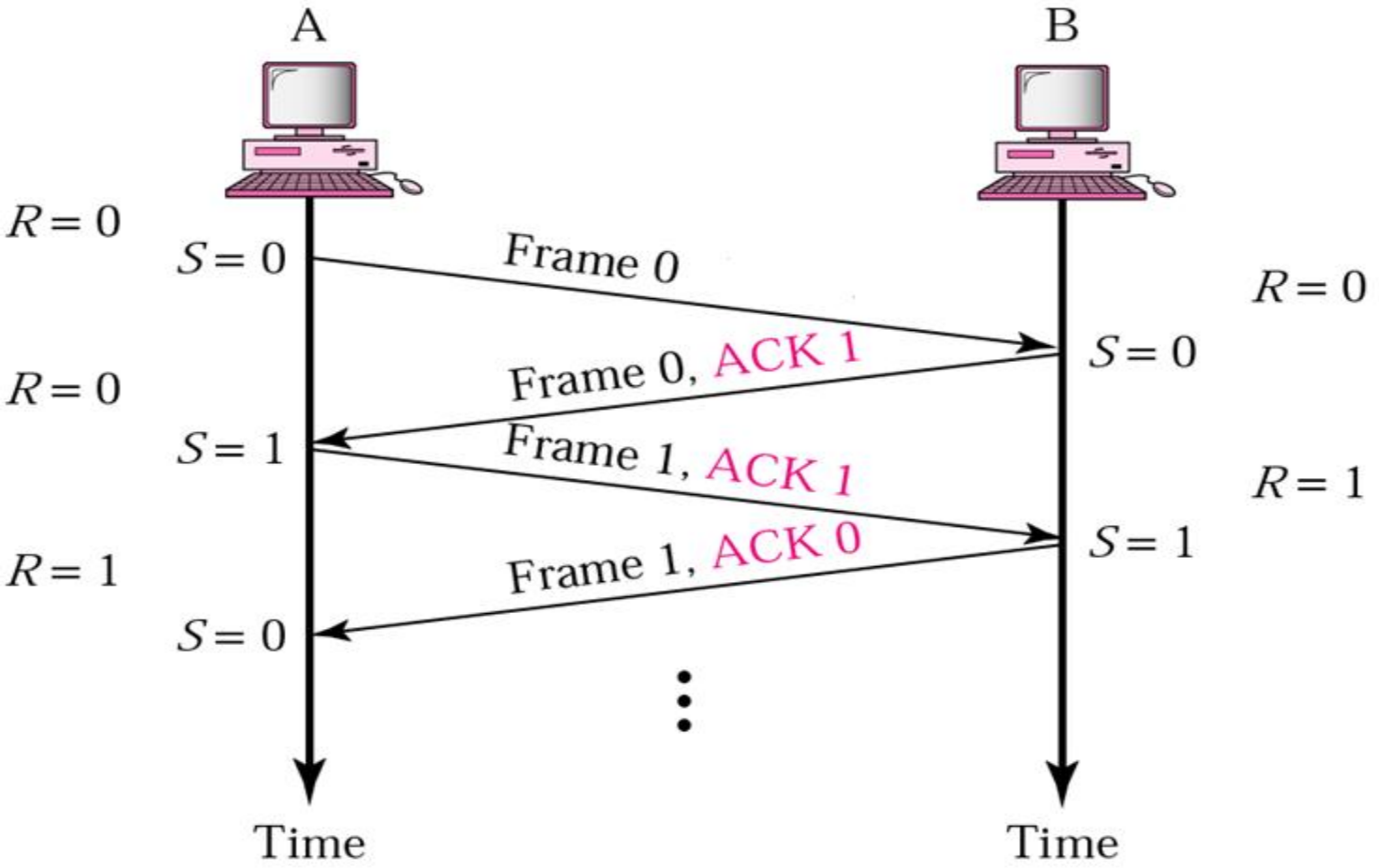
Lost Acknowledgment



Delayed Acknowledgement



Piggybacking



Stop and Wait ARQ

- The sender has only one outstanding frame on the link at a time
  - This may be far below the link's capacity
- Consider a 1.5 Mbps link with a 45 ms RTT
  - The link has a delay × bandwidth product of 67.5 Kb or approximately 8 KB
  - Since the sender can send only one frame per RTT and assuming a frame size of 1 KB
  - Maximum Sending rate
    - Bits per frame ÷ Time per frame = 1024 × 8 ÷ 0.045 = 182Kbps

      Or about one-eighth of the link's capacity
  - To use the link fully, then sender should transmit up to eight frames before having to wait for an acknowledgement

Sliding Window Protocol

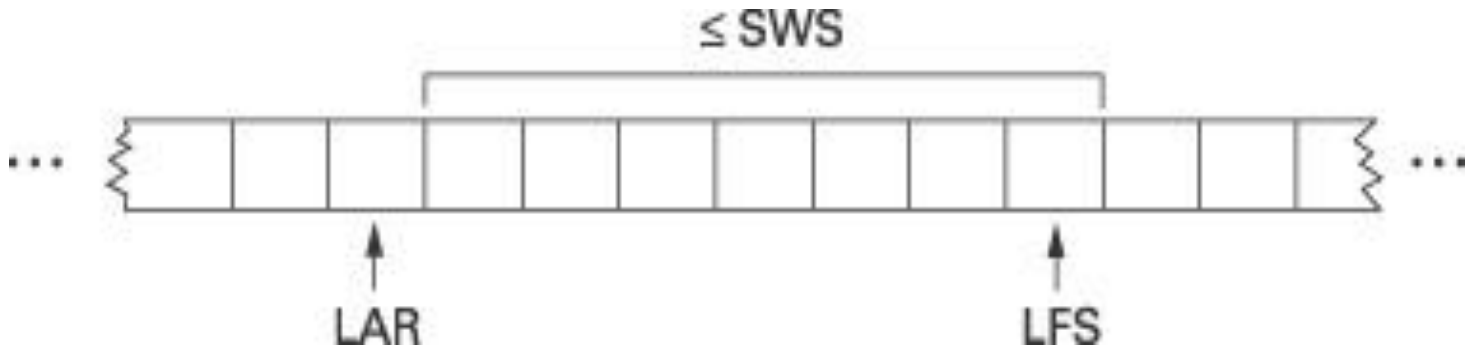
- Sender Window
- Receiver Window
- The sender and receiver slides its window

Sender Window

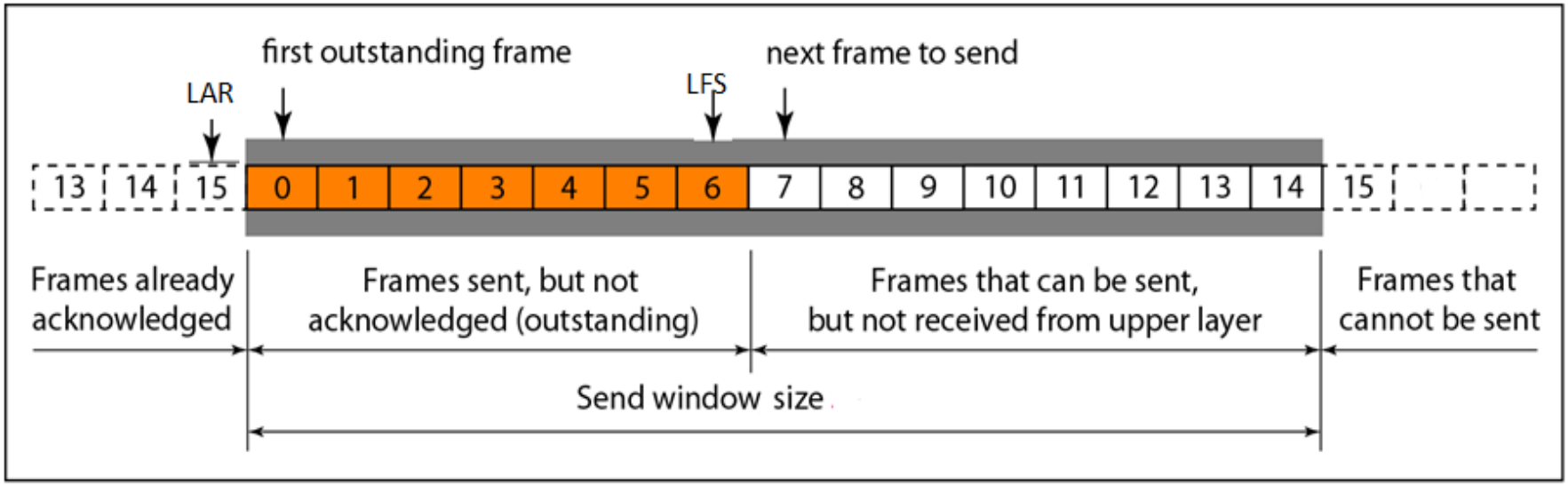
- **SeqNum** → Sender assigns sequence number to each frame which grows infinitely
- **SWS** →Sender Window Size, The size of sender window. ie The number of unacked frames the sender can transmit.(outstanding frames)
- **LAR** → Seqnum of the Last Ack Received
- **LFS** → Seqnum of the Last Frame Sent

Sender Window

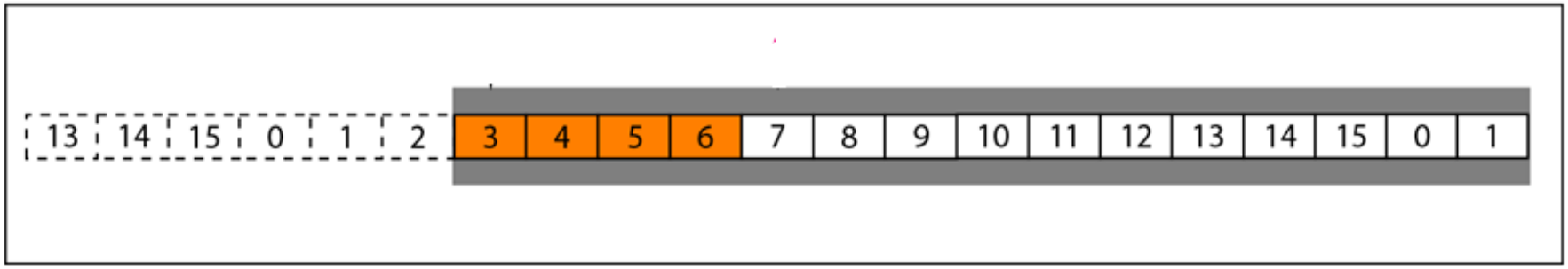
- Sender Maintains LFS – LAR ≤ SWS
- Ack → LAR to right (sliding)
- A timer is associated with each frame
- The sender buffers SWS no. of frames.



Sender Window



a. Send window before sliding



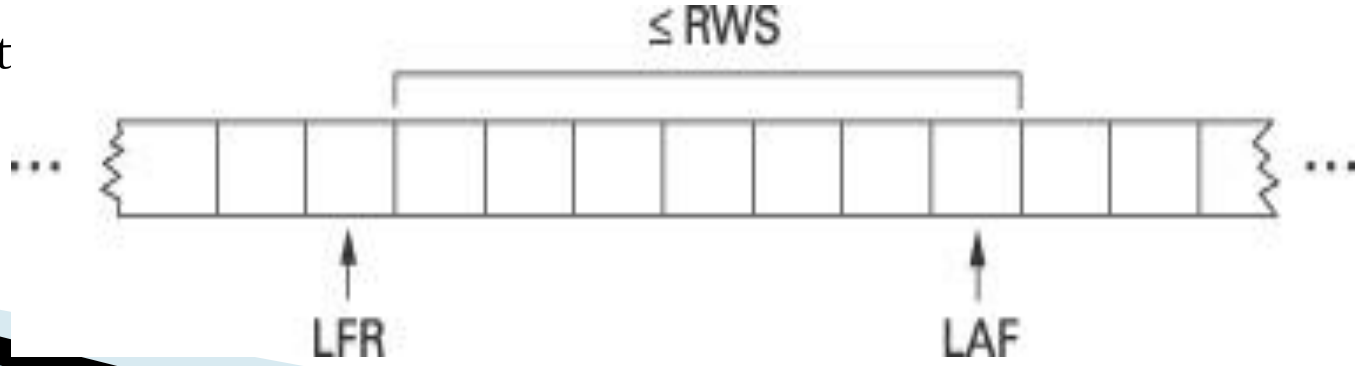
b. Send window after sliding

Receiver Window

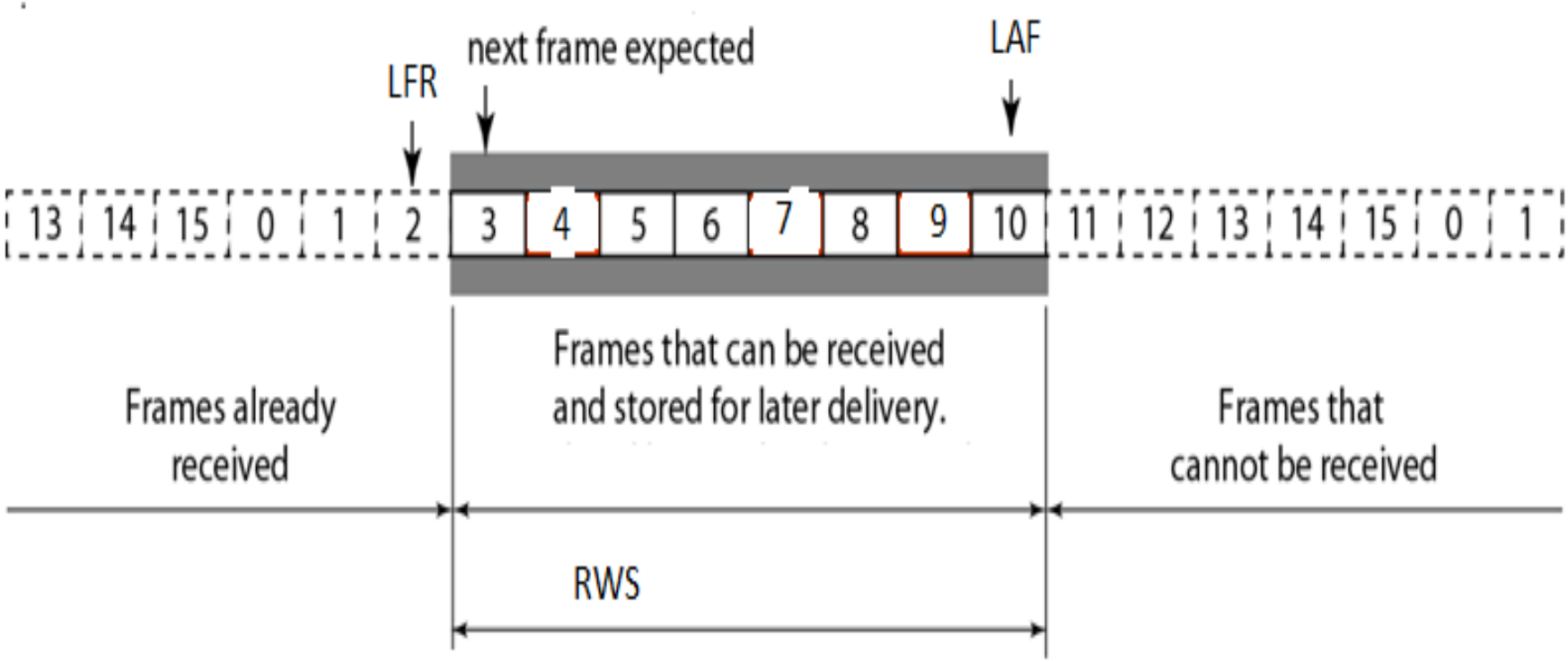
- **RWS** → Receiver Window Size, The size of receiver window. ie The number of frames the receiver willing to accept.(out-of-order frames)
- **LAF** →Seqnum of the Largest Acceptable Frame
- **LFR** →Seqnum of the Last Frame Received

Receiver Window

- Receiver Maintains **LAF − LFR ≤ RWS**
- **SeqNum ≤ LFR or SeqNum > LAF**
  - frame is outside the receiver's window, discard it
- **SeqNum within the window and arrived in out of order** → buffer the received frame and send a NAK to the sender for the expected frame.
- **SeqNum within the window and arrived in order** → slide the window
- **LFR < SeqNum ≤ LAF** →frame within the receiver's window and it is accept



Receiver Window



Finite Sequence Numbers

- Header allocates m bits for seqnum. (0 to 2^m-1)

  If $m = 2$

  $0 – (2^2 – 1)$

  0 – 3 is the range. ie., 0, 1, 2, 3

- **MaxSeqNum → 2^m**
- 9 frames to be sent, and 2 bits (m=2) are allotted in the header of frame for SeqNum,
  - Frames are numbered as : 0 1 2 3 0 1 2 3
- **SWS ≤ MaxSeqNum − 1**

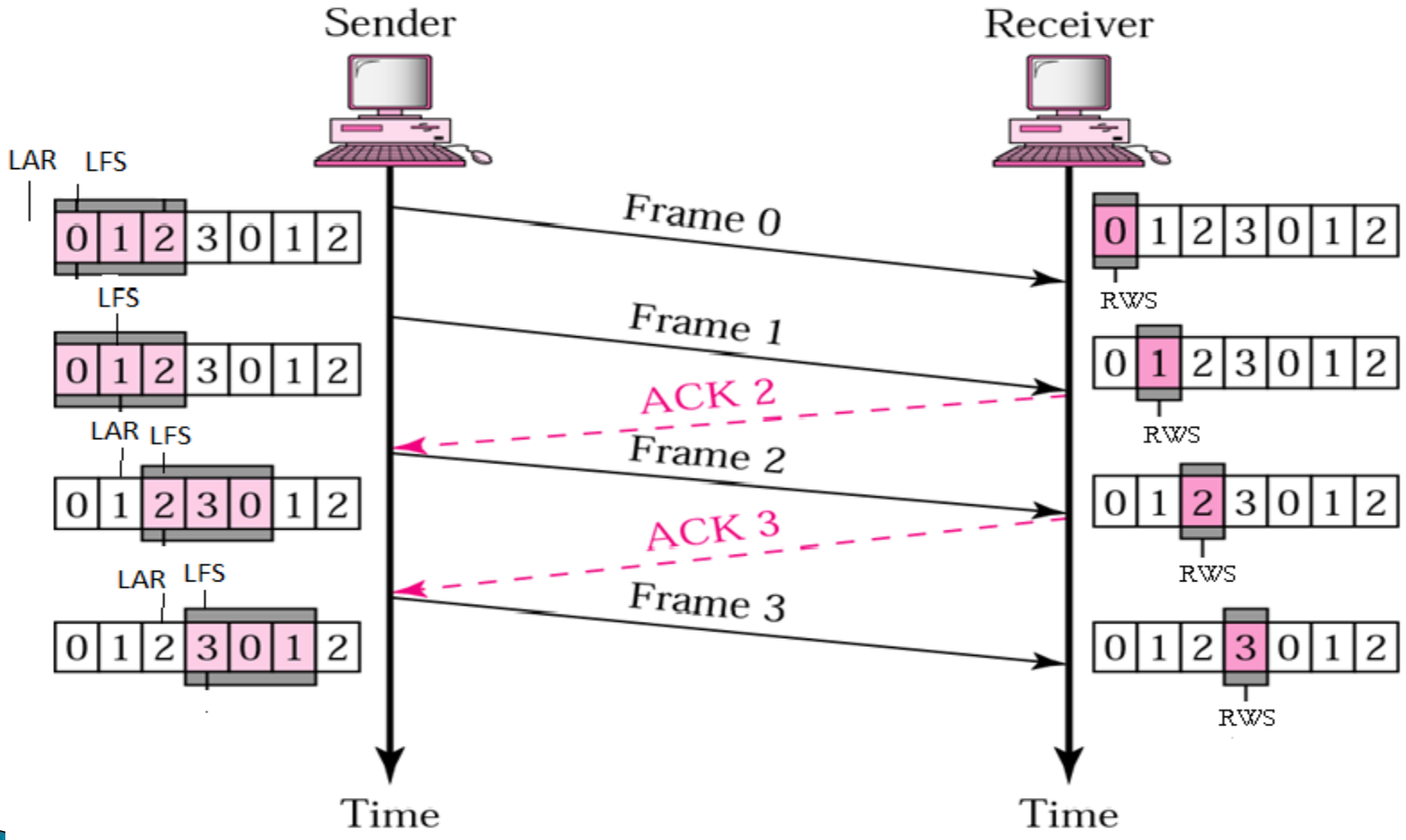
Go Back N ARQ

- Sender Wndow
  - The frames are stored in a buffer.
  - The frames that are sent but the ack is expected (outstanding frames) are enclosed in a window.
  - The size of the window is SWS ≤ MaxSeqNum − 1
  - On receiving an ack the window slides.
  - Each frame is associated with a timer
- Receiver Window
  - RWS= 1
  - The receiver is expecting a specific frame to arrive in specific order
  - No Timer

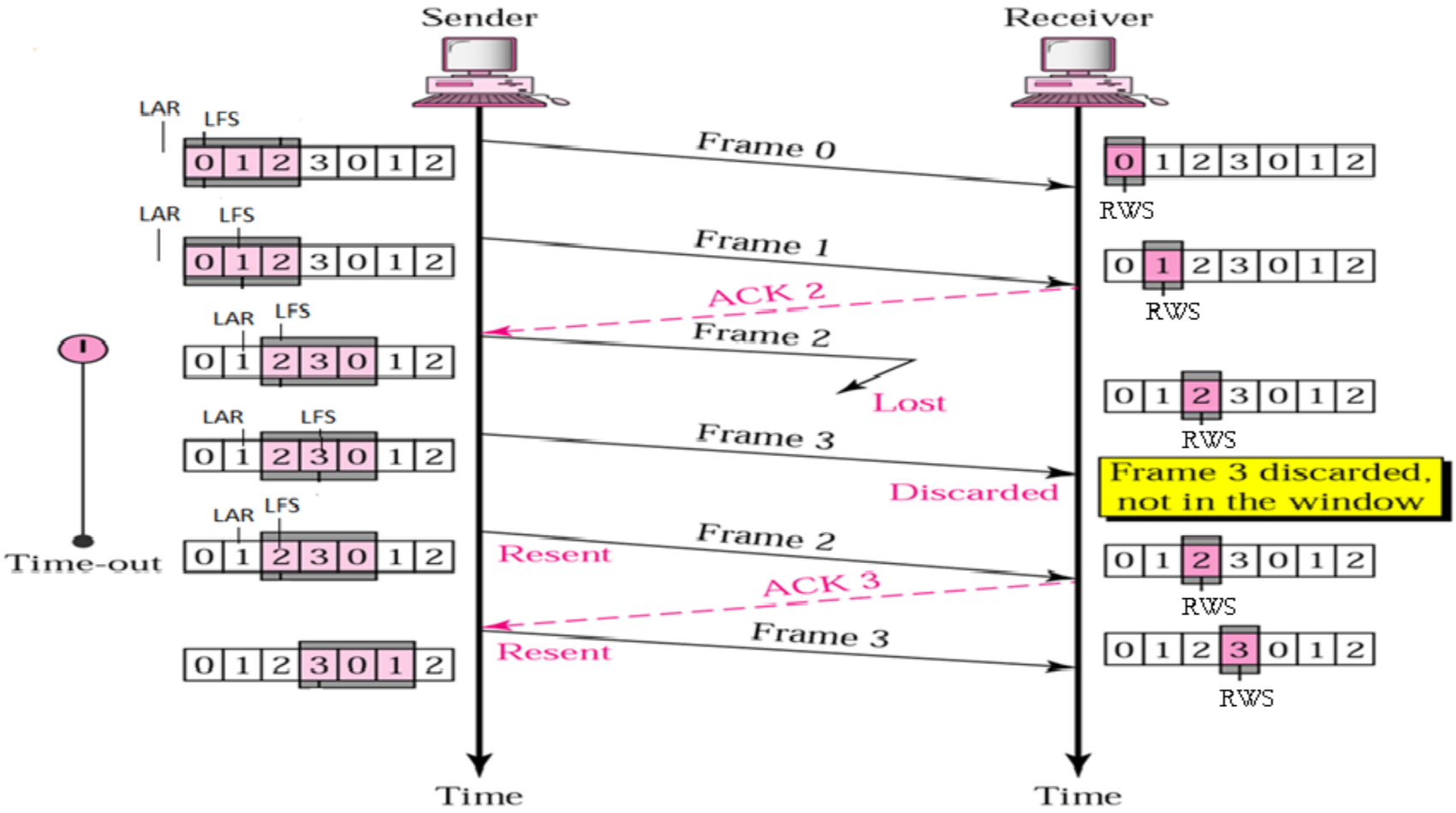
Go Back N ARQ

- Acknowledgment:
  - Frame received without damage → receiver sends an ack
  - Damaged Frame received or out of order → receiver keeps silent without sending acknowledgment. So the timer in sender side expires and the sender retransmits the frames starting from the one with expired timer.
  - Can send single or cumulative ack.
  - No concept of NAK

Go Back N ARQ (NO)



Go Back N ARQ (Frame Lost)



Go Back N ARQ

- Lost Ack
  - Resend Frame
  - When cumulative ack reaches the sender, then there is no need for retransmission.
- Delayed Ack
  - Resend Frame

Selective Repeat ARQ

- In Go back N if one frame is damaged or lost, N frames are resend.
- Selective repeat ARQ uses a mechanism to retransmit only the damaged or lost frame instead of N frames.

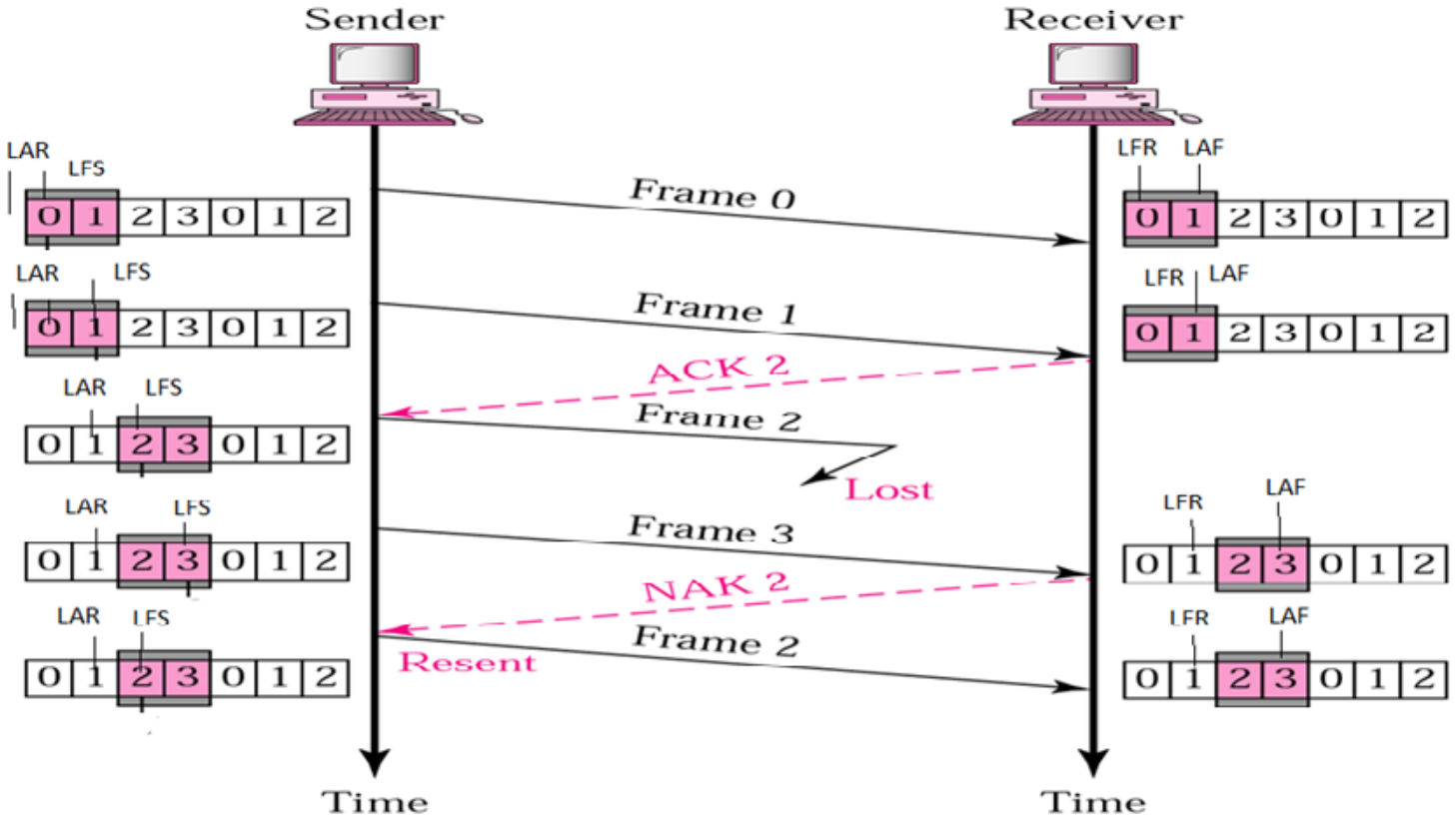
Selective Repeat ARQ

- Sender Window
  - SWS < (MaxSeqNum + 1)/2
  - On receiving an ack the window slides.
- Receiver Window
  - RWS = SWS
  - After receiving a frame, the receiver sends an ack to the sender.
  - The receiver window slides to its right side if the frame is received successfully.

Selective Repeat ARQ

- Normal Operation
  - Sender sends the frames in the window one by one
  - After the ack is received the sender window slides
- Frame lost or damage
  - If frame 2 is lost, frame 3 is sent by the sender sequentially.
  - The Receiver accepts frame 3 if frame 3 is in receiver window and sends and NAK 2 to sender requesting for the retransmission of frame 2.

Negative Ack (NAK)



Selective Repeat ARQ

- Lost and delayed Ack's
  - The sender sets a timer when a frame is sent.
  - If NAK or Ack is not received in the specified time the sender retransmits the frame.