Flow and Error Control

Computer Networks(CS31204)

Prof. Sudip Misra

Department of Computer Science and Engineering

Indian Institute of Technology Kharagpur

Email: smisra@sit.iitkgp.ernet.in

Website: http://cse.iitkgp.ac.in/~smisra/

Research Lab: cse.iitkgp.ac.in/~smisra/swan/

Data Link layer

Data link control functions include framing, flow and error control, and software implemented protocols that provide smooth and reliable transmission of frames between nodes.

Framing

- Framing in the data link layer separates a message from one source to a destination, or from other messages to other destinations, by adding a sender address and a destination address.
- The destination address defines where the packet is to go; the sender address helps the recipient acknowledge the receipt.
- Whole message could not be packed in one frame because:
  frame can be very large, making flow and error control very inefficient.

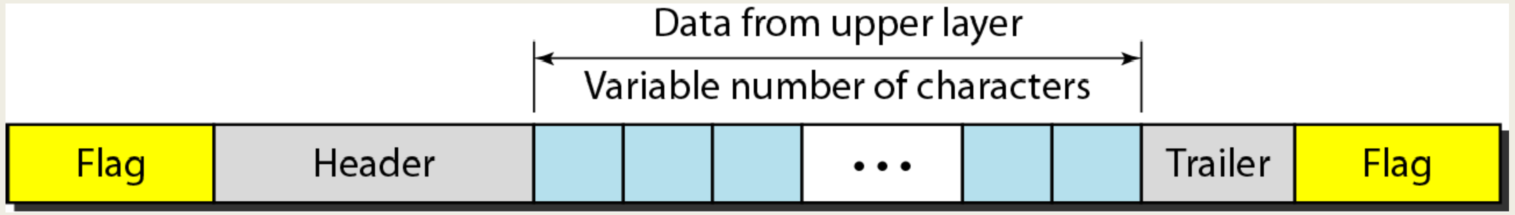


Fig: A frame in a character-oriented protocol

Source: B. A. Forouzan, " Data Communications and Networking ," *McGraw-Hill Forouzan Networking Series*,5E.

Bit Oriented Protocol

- The data section of a frame is a sequence of bits to be interpreted by the upper layer as text, graphic, audio, video, and so on.
- However, in addition to headers, we still need a delimiter to separate one frame from the other.
- Most protocols use a special 8-bit pattern flag 01111110 as the delimiter to define the beginning and the end of the frame.
- To differentiate the message from delimiter, byte stuffing is done.

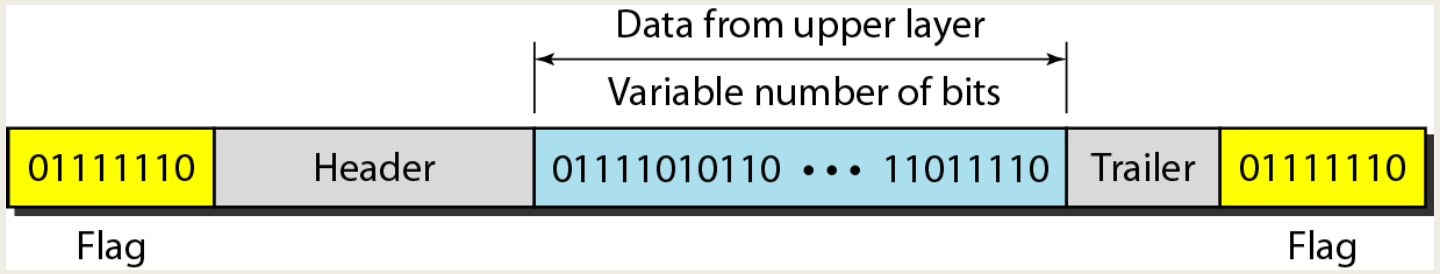


Fig: A frame in a bit-oriented protocol

Source: B. A. Forouzan, “ Data Communications and Networking ,” *McGraw-Hill Forouzan Networking Series*,5E.

Bit Stuffing

- Bit stuffing is the process of adding one extra 0 whenever five consecutive 1 follow a 0 in the data, so that the receiver does not mistake the pattern 0111110 for a flag.

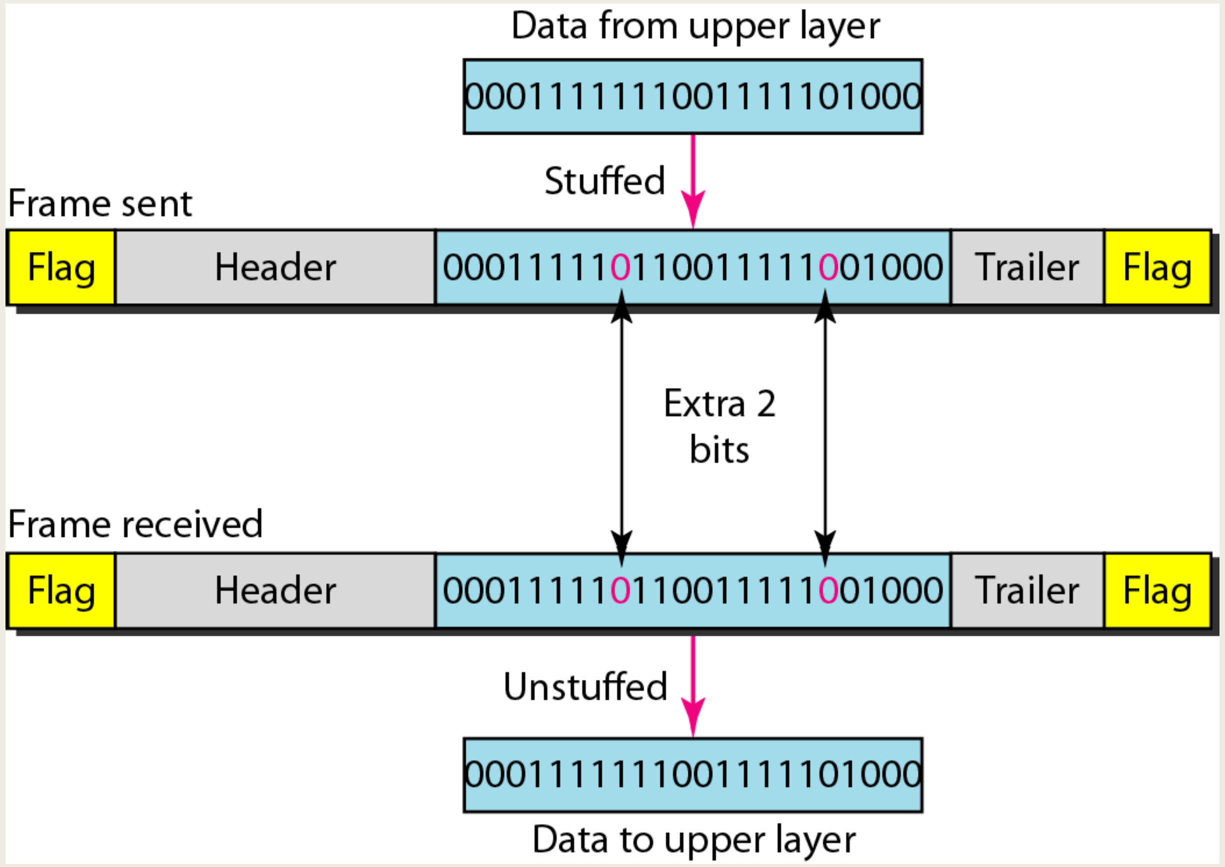


Fig: Byte Stuffing and Unstuffing

Source: B. A. Forouzan, " Data Communications and Networking ," *McGraw-Hill Forouzan Networking Series*,5E.

Flow and Error Control

- **Flow control** coordinates the amount of data that can be sent before receiving an acknowledgment
- One of the most important duties of the data link layer.
- In most protocols, **flow control** is a set of procedures that tells the sender how much data it can transmit before it must wait for an acknowledgment from the receiver.
- **Error control** in the data link layer is based on automatic repeat request, which is the retransmission of data.

Flow Control

- What if sender sends the bit pattern too fast? Receiver may not be able to process the data as fast as the sender is sending it
- Receiver can buffer, but buffer has finite size. After that is filled, data is lost
- Flow control – technique to control data flow between sender and receiver so that sender is blocked if receiver cannot accept any more data
- Sender must get an acknowledgement from the receiver before it can send more data

Stop and Wait Flow Control

- Sender sends a frame
- Receiver receives frame & acknowledges it
- Sender waits to receive “ack” before sending next frame (If receiver is not ready to receive another frame it holds back the ack)
- One frame at a time is sent over the transmission line
- Simple, easy to implement

Stop and Wait Flow Control

Problems:

- ❑ Works fine if there is only a few large frames to be sent
- ❑ Usually we want smaller frames
  - ➢ Receiver’s buffer size may be small
  - ➢ If a large frame is transmitted, the entire frame has to be retransmitted if there is an error
  - ➢ Longer the frame, larger the chance of an error
  - ➢ On a shared medium, do not want any one sender to occupy the medium for a long time
- ❑ With smaller frames, takes too long to send all frames

Link Efficiency of Stop and Wait Flow Control

- Bit Length of link in bits B = maximum possible number of bits that can be simultaneously present in the link
  - B = R × (D/V)
  - R = data rate of link in bps
  - D = length of link in meters
  - V = propagation speed in m/sec

- So for Stop & Wait, ideally we want frame length to be close to bit length of the link
  - The sender has enough bits to put out on the link while the earlier bits of the same frame reach the receiver
  - Links stays occupied for more time

- But we have also argued that we want frame sizes to be small

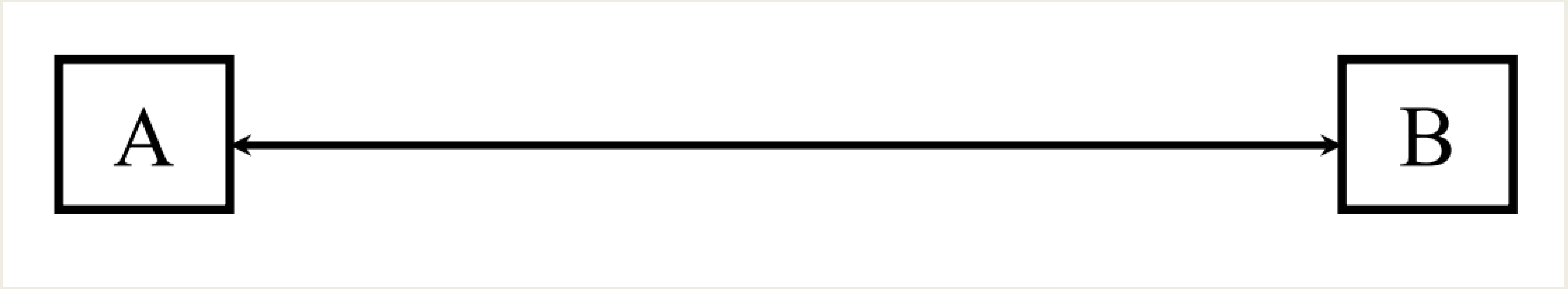
- Results in very poor line/link utilization for Stop & Wait, especially in fast links over longer distances
  - B becomes very large, so impractical to have such large frames

Link Efficiency of Stop and Wait Flow Control

- Consider a frame length of L bits
  - Transmission delay = Time to put the frame into the link = L/R
  - Time for last bit of the frame to reach the receiver = (L/R) + (D/V)
  - Assume that processing delay at receiver is negligible
  - Assume that transmission time of ACK is negligible (very small frame)
  - Time for ACK to come to sender = (D/V)
  - Total time to transmit the frame T = (L/R) + 2(D/V)
  - Total no. of bits that could have ben transmitted in this time = T × R
  - Actual no. of bits transmitted = L
  - Line utilization = L/(T × R) = L / (L + 2R(D/V))

Sliding Window Flow Control

- Reduces line inefficiency problem of stop-and-wait by transmitting multiple frames without waiting for acknowledgement
- Suppose two stations A and B are connected by a fullduplex link



Sliding Window Flow Control

- Station B allocates buffer space for W frames.
- Thus B can accept W frames, and A is allowed to send W frames without waiting for any acknowledgement
- Sender maintains a list of sequence numbers that it is allowed to send (sender window)
- Receiver also maintains a list of sequence numbers that it is prepared to receive (receiver window)
- Since the sequence number to be used occupies a field in the frame, it is clearly of bounded size
- For a k-bit field, the range of sequence numbers is 0 through $2^k – 1$, and frames are numbered modulo 2^k
- Maximum window size = $2^k – 1$ (why?)

Sliding Window Flow Control

- Each frame is labeled with a sequence number
- To keep track of the frames which have been acknowledged
- B acknowledges a frame by sending an ACK/RR that includes the sequence number of the next frame expected. This also explicitly announces that B is prepared to receive the next W frames, beginning with the number specified.
- This scheme can be used to acknowledge multiple frames
- B could receive frames 2,3,4 but withhold ACK until frame 4 has arrived. By returning an ACK with sequence number 5, B acknowledges frames 2,3,4 at one time

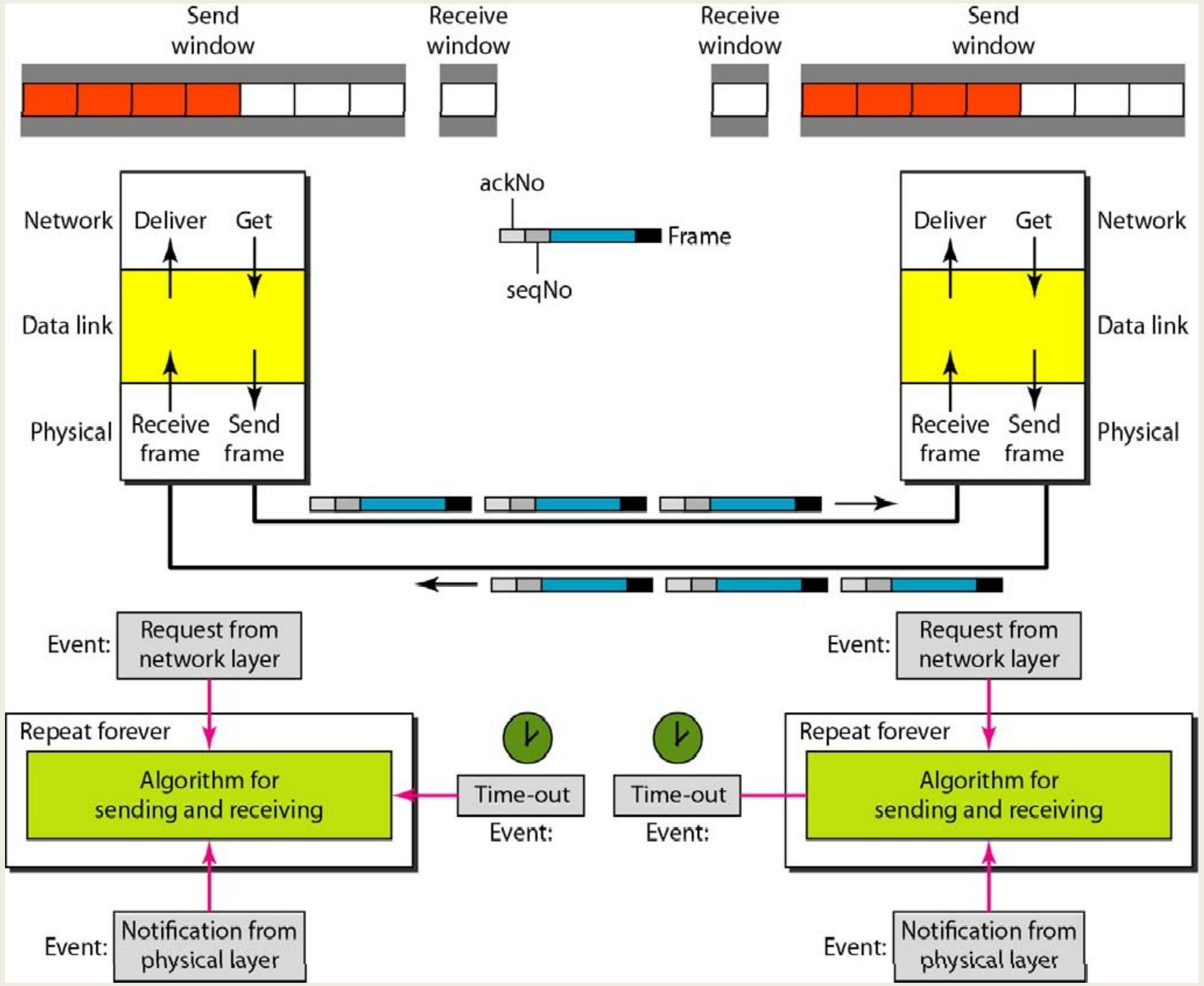
Sliding Window Flow Control

- The actual window size need not be the maximum possible size for a given sequence number length
- For a 3-bit sequence number, a window size of 5 can also be configured
- If two stations exchange data, each need to maintain two windows. To save communication capacity, a technique called piggybacking is used
- Each data frame includes a field that holds the sequence number of that frame plus a field that holds the sequence number used for ACK
- If a station has an ACK but no data to send, it sends a separate ACK frame

Piggybacking

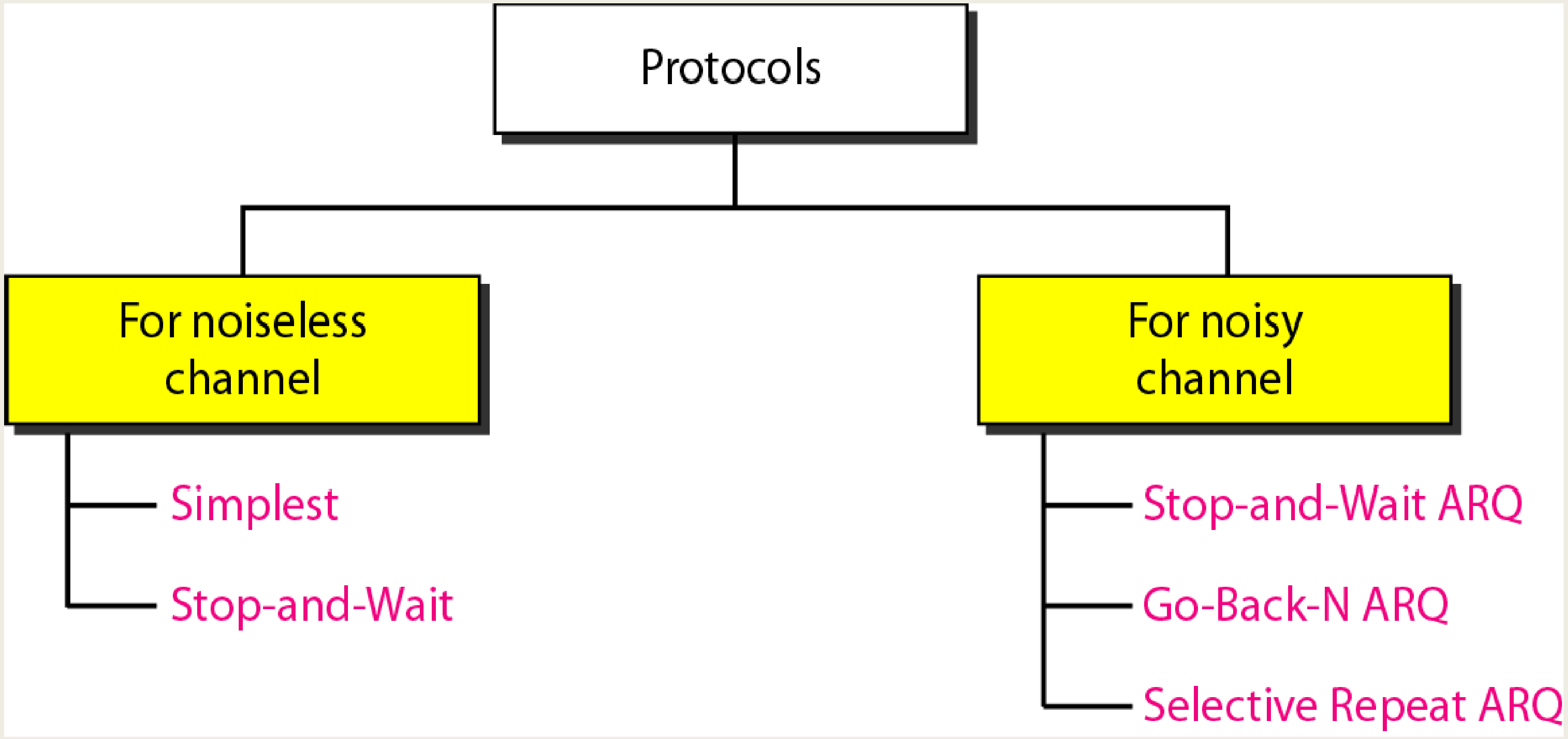
- Improve the efficiency of the bidirectional protocols.

- When a frame is carrying data from A to B, it can also carry control information about arrived (or lost) frames from B; when a frame is carrying data from B to A, it can also carry control information about the arrived (or lost) frames from A.



Source: B. A. Forouzan, " Data Communications and Networking ," *McGraw-Hill Forouzan Networking Series*,5E.

Error Control Protocols



Source: B. A. Forouzan, “ Data Communications and Networking ,” *McGraw-Hill Forouzan Networking Series*,5E.

Simplest Protocol

- It is a unidirectional protocol in which data frames are traveling in only one direction-from the sender to receiver.
- The receiver can immediately handle any frame it receives with a processing time that is small enough to be negligible.
- The data link layer of the receiver immediately removes the header from the frame and hands the data packet to its network layer, which can also accept the packet immediately.

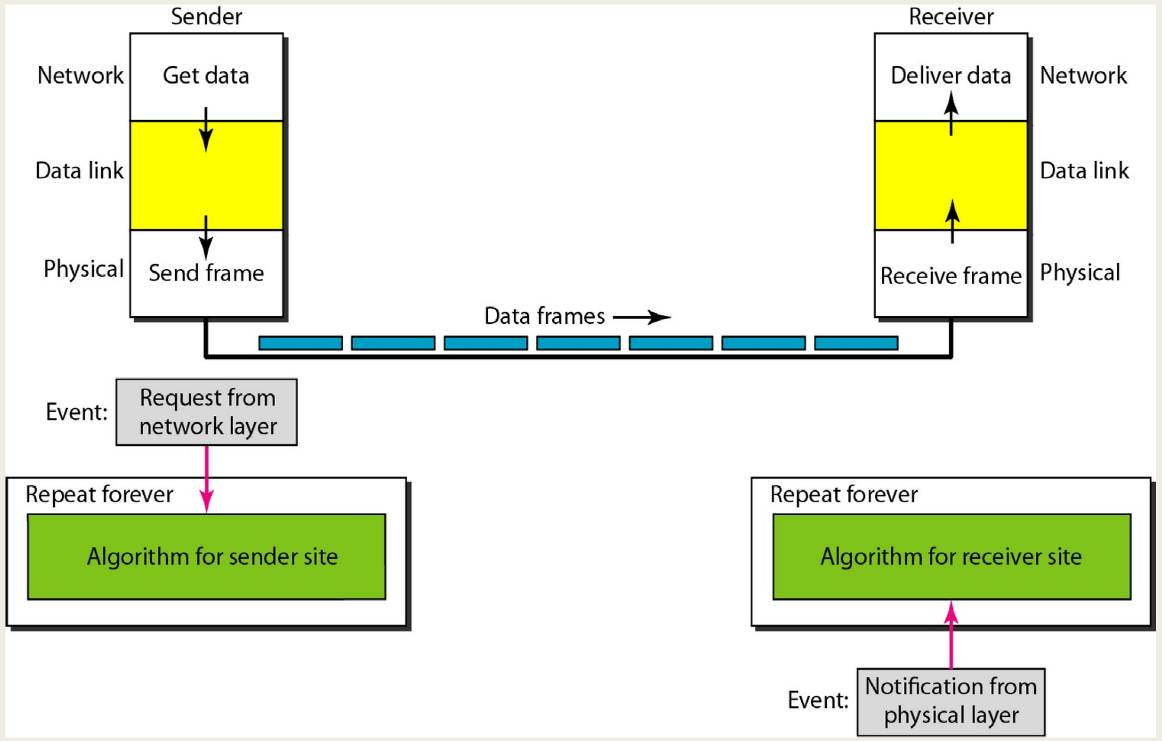


Fig.: The design of the simplest protocol with no flow or error control

Source: B. A. Forouzan, " Data Communications and Networking ," *McGraw-Hill Forouzan Networking Series*,5E.

Stop and Wait Protocol

- The sender sends one frame, stops until it receives confirmation from the receiver, and then sends the next frame.
- Unidirectional communication for data frames, but auxiliary ACK frames (simple tokens of acknowledgment) travel from the other direction.

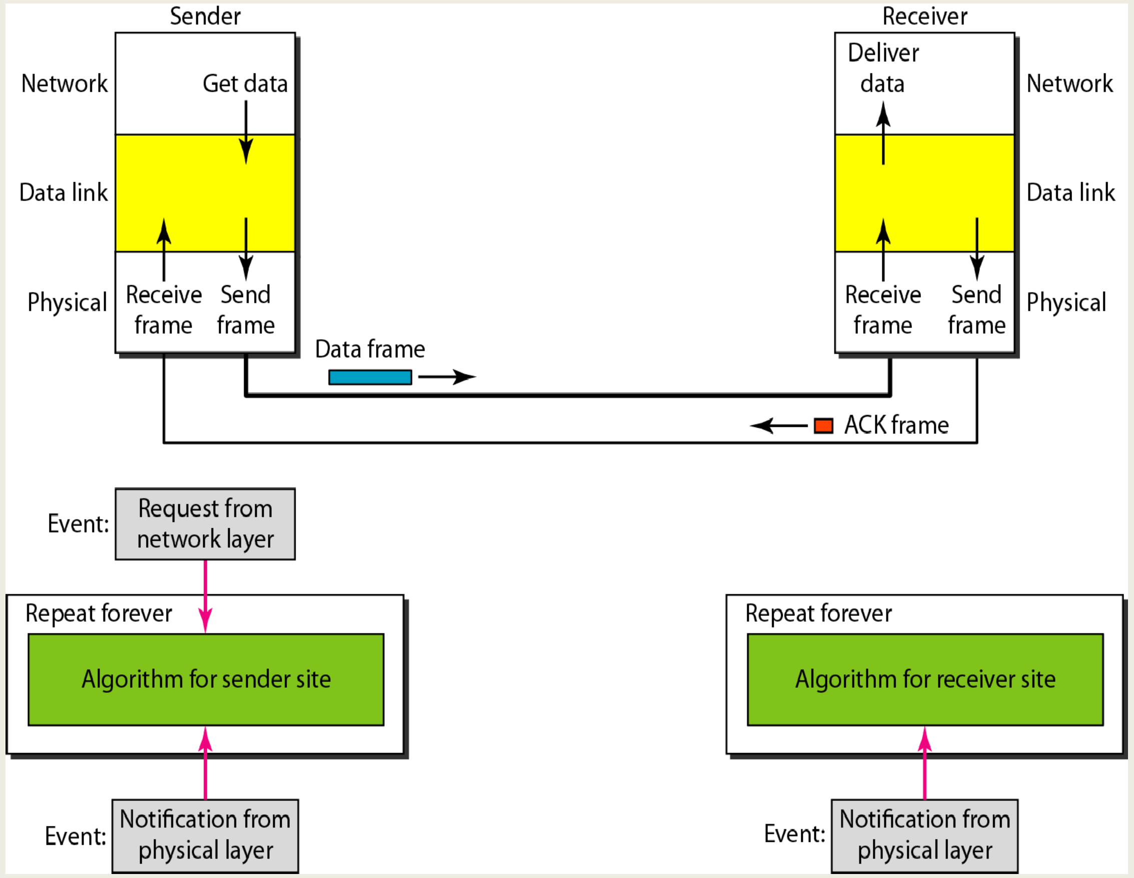
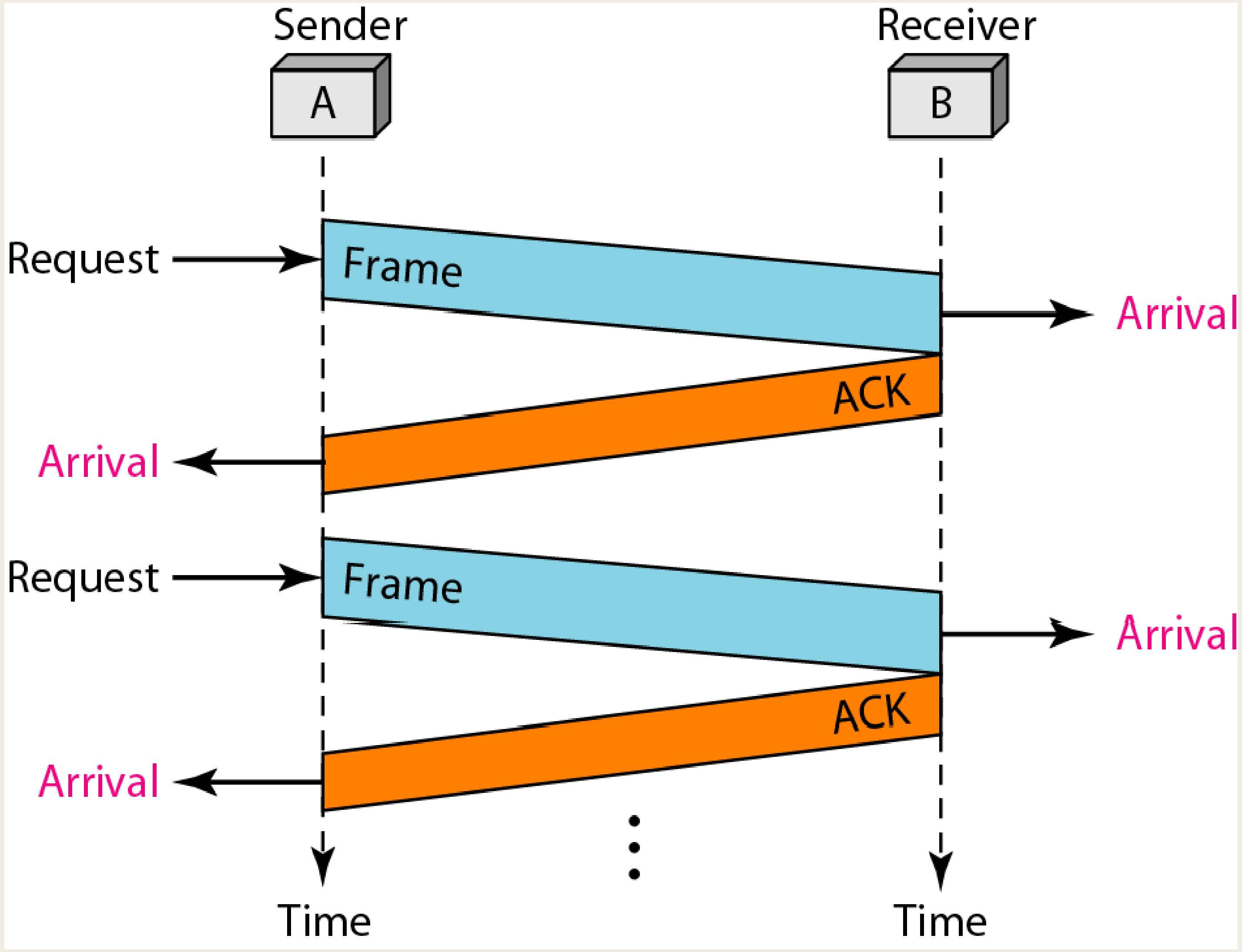


Fig.: The design of the Stop and wait protocol

Source: B. A. Forouzan, " Data Communications and Networking ," *McGraw-Hill Forouzan Networking Series*,5E.

Stop and Wait Protocol: Example



Source: B. A. Forouzan, " Data Communications and Networking ," *McGraw-Hill Forouzan Networking Series*,5E.

Stop and Wait Automatic Repeat Request

- The sender keeps a copy of the sent frame. At the same time, it starts a timer.
- If the timer expires and there is no ACK for the sent frame, the frame is resent, the copy is held, and the timer is restarted.
- The protocol uses the stop-and-wait mechanism, there is only one specific frame that needs an ACK even though several copies of the same frame can be in the network.

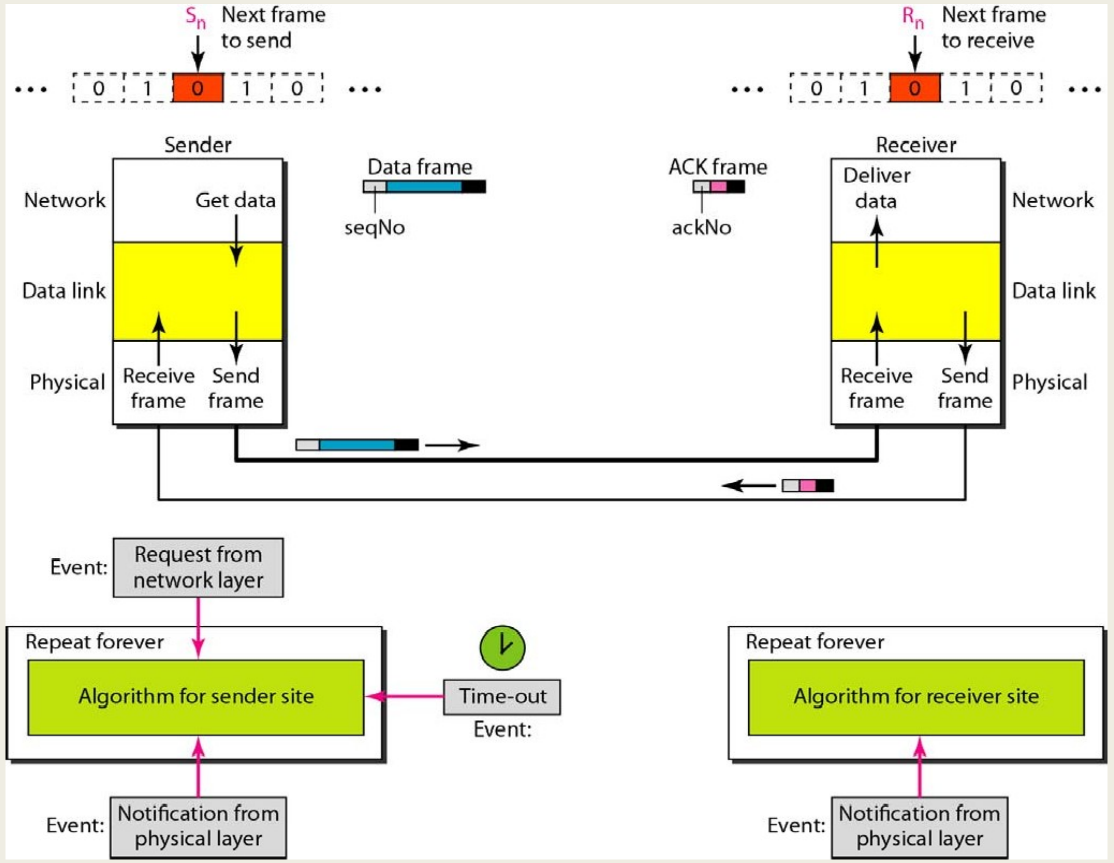
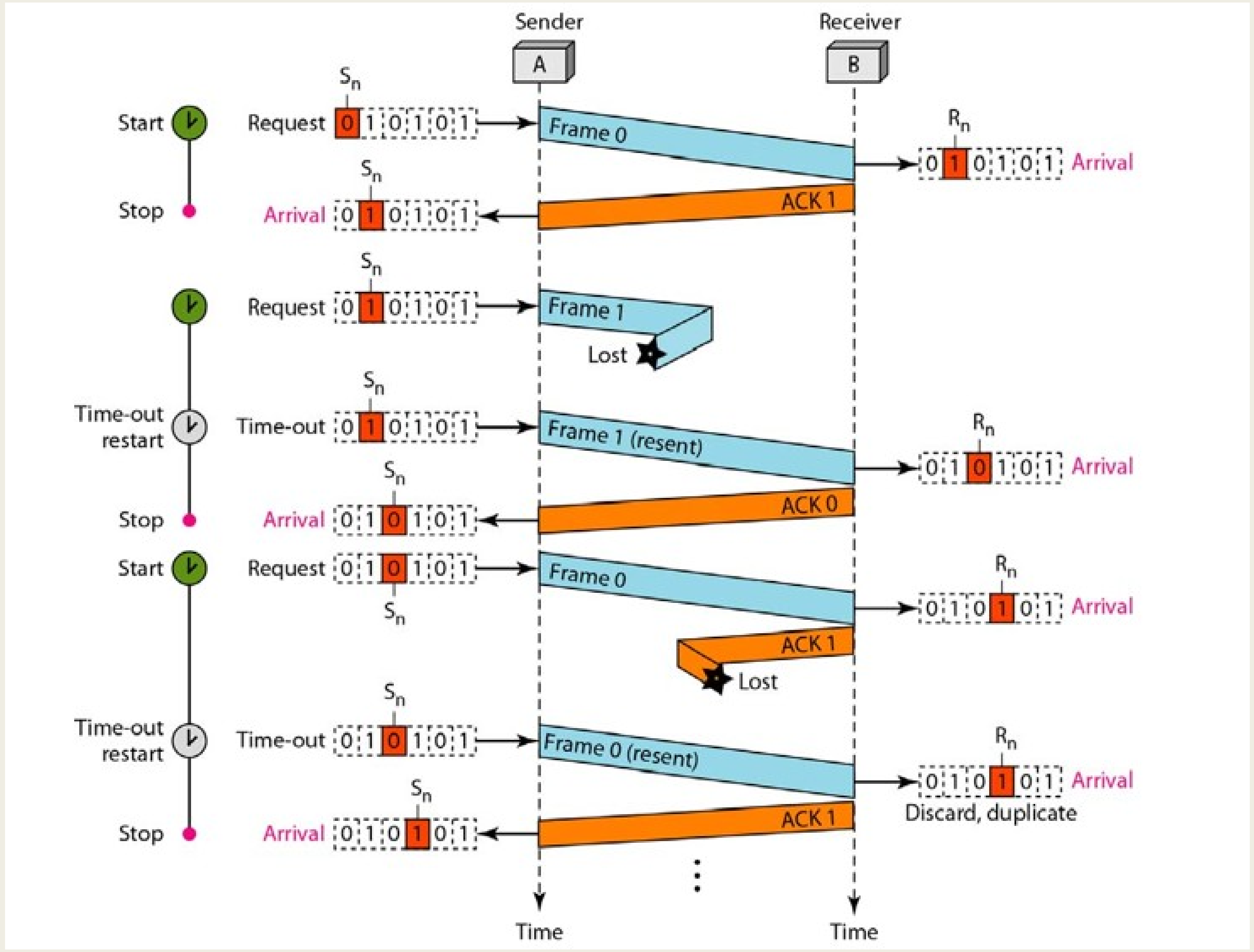


Fig.: The design of the Stop and Wait ARQ protocol

https://docs.google.com/viewer?a=v&pid=sites&srcid=ZGVmYXVsdGRvbWFpbnxlZWJhaHJpYTV8Z3g6MjE2MzY2MjcxMGU3MmQ4Nw

Example



Source: B. A. Forouzan, " Data Communications and Networking ," *McGraw-Hill Forouzan Networking Series*,5E.

Prof. Sudip Misra, IIT Kharagpur

Sequence Number

- A field is added to the data frame to hold the sequence number of that frame.
- The sequence numbers are wrap around. For example, if the field is *m* bits long, the sequence numbers start from 0, go to 2^m - 1, and then are repeated.
- The sequence numbers are based on modulo-2 arithmetic.

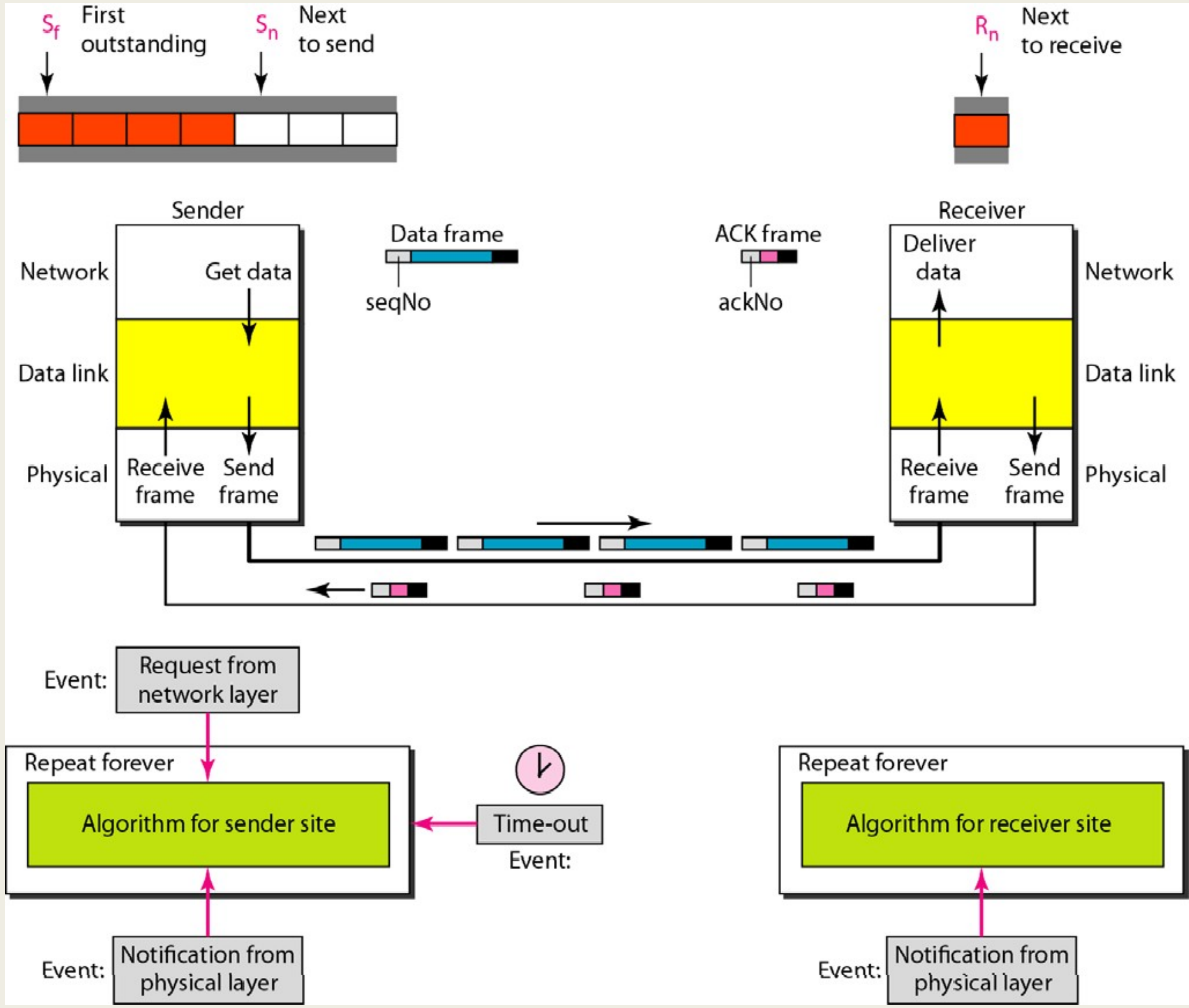
Acknowledgement Number

The acknowledgment numbers always announce the sequence number of the next frame expected by the receiver.

For example, if frame 0 has arrived safe and sound, the receiver sends an ACK frame with acknowledgment 1.
If frame 1 has arrived safe and sound, the receiver sends an ACK frame with acknowledgment 0 (meaning frame 0 is expected).

Go-Back-N Protocol

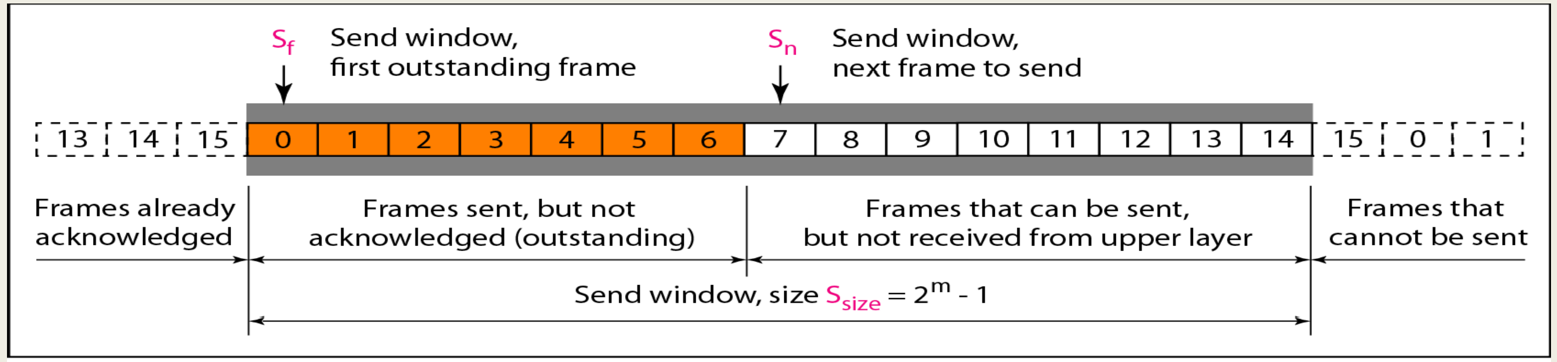
- In this protocol we can send several frames before receiving acknowledgments.
- Keep a copy of these frames until the acknowledgments arrive.
- Improve the efficiency of transmission
- In the Go-Back-N Protocol, the sequence numbers are modulo 2^m, where m is the size of the sequence number field in bits.



Source: B. A. Forouzan, " Data Communications and Networking ," *McGraw-Hill Forouzan Networking Series*,5E.

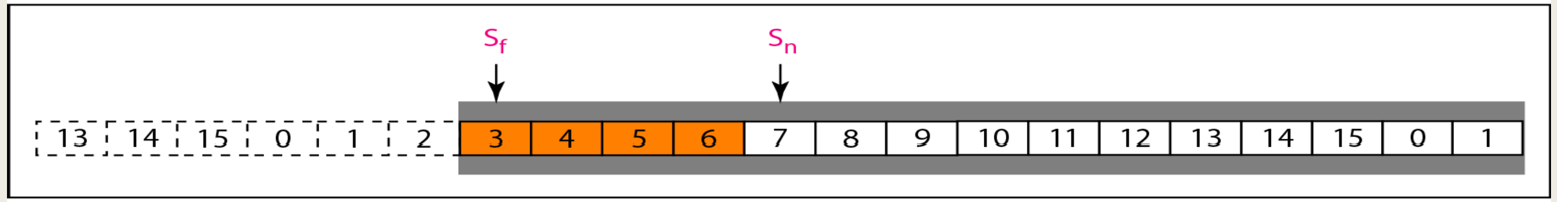
Sender Window

- The sender window is an imaginary box covering the sequence numbers of the data frames which can be in transit.
- In each window position, some of these sequence numbers define the frames that have been sent; others define those that can be sent.
- The maximum size of the window is $2^m - 1$.
- The sender window can slide one or more slots when a valid acknowledgment arrives.



S_f
Send window, first outstanding frame
S_n
Send window, next frame to send
13 14 15 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 0 1
Frames already acknowledged
Frames sent, but not acknowledged (outstanding)
Frames that can be sent, but not received from upper layer
Frames that cannot be sent
Send window, size $S_{size} = 2^m - 1$

a. Send window before sliding

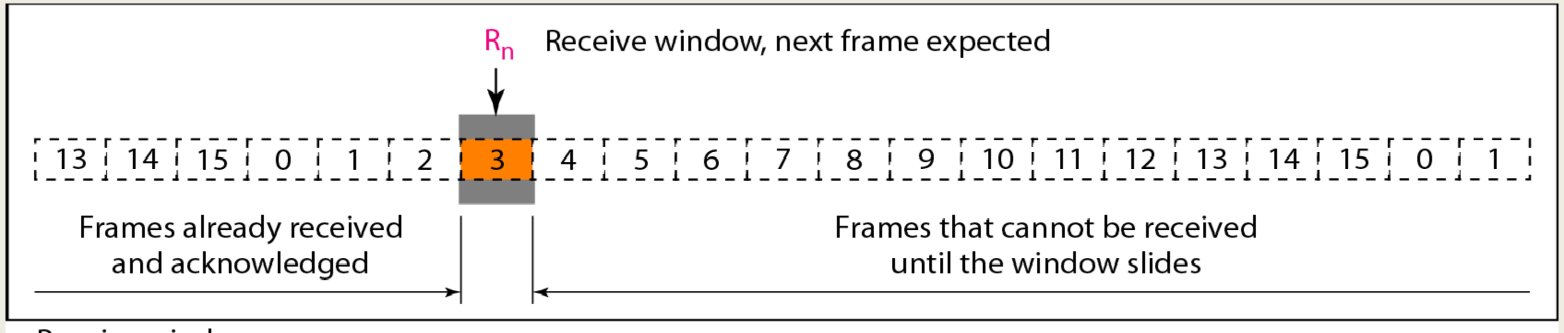


b. Send window after sliding

Source: B. A. Forouzan, " Data Communications and Networking ," *McGraw-Hill Forouzan Networking Series*,5E.

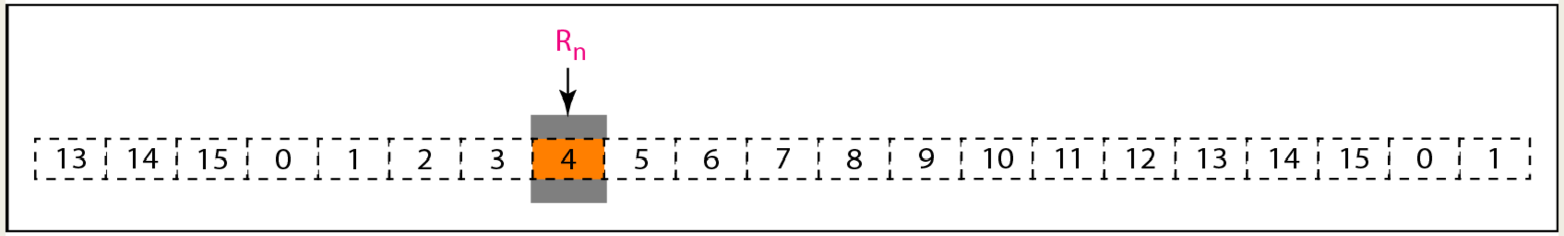
Receiver Window

- The receiver window is an abstract concept defining an imaginary box of size 1 with one single variable R_n.
- The window slides when a correct frame has arrived; sliding occurs one slot at a time.
- In Go-Back-N ARQ, the size of the send window must be less than 2^m; the size of the receiver window is always 1.



R_n
Receive window, next frame expected
13 14 15 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 0 1
Frames already received and acknowledged
Frames that cannot be received until the window slides

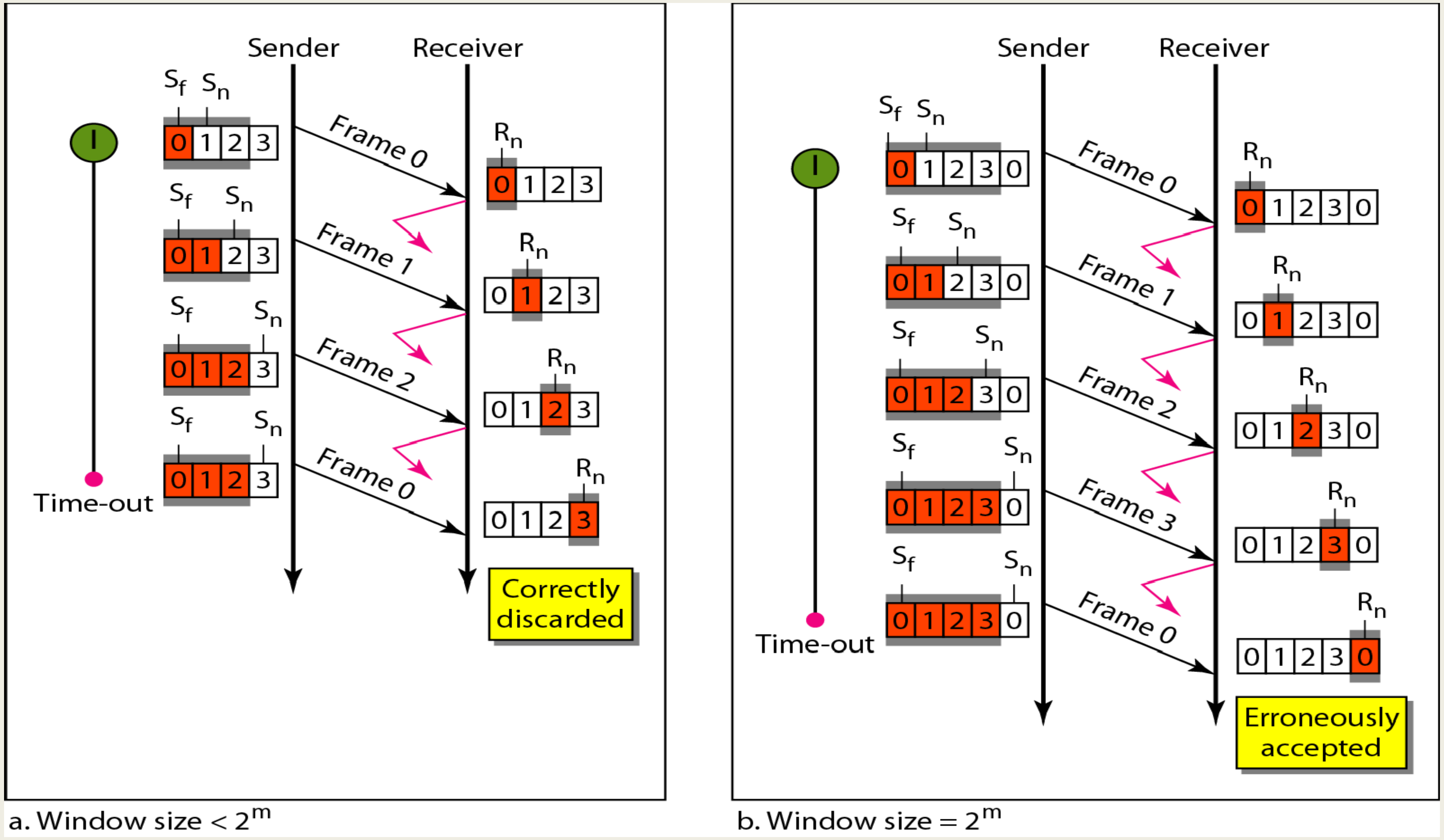
a. Receive window



b. Window after sliding

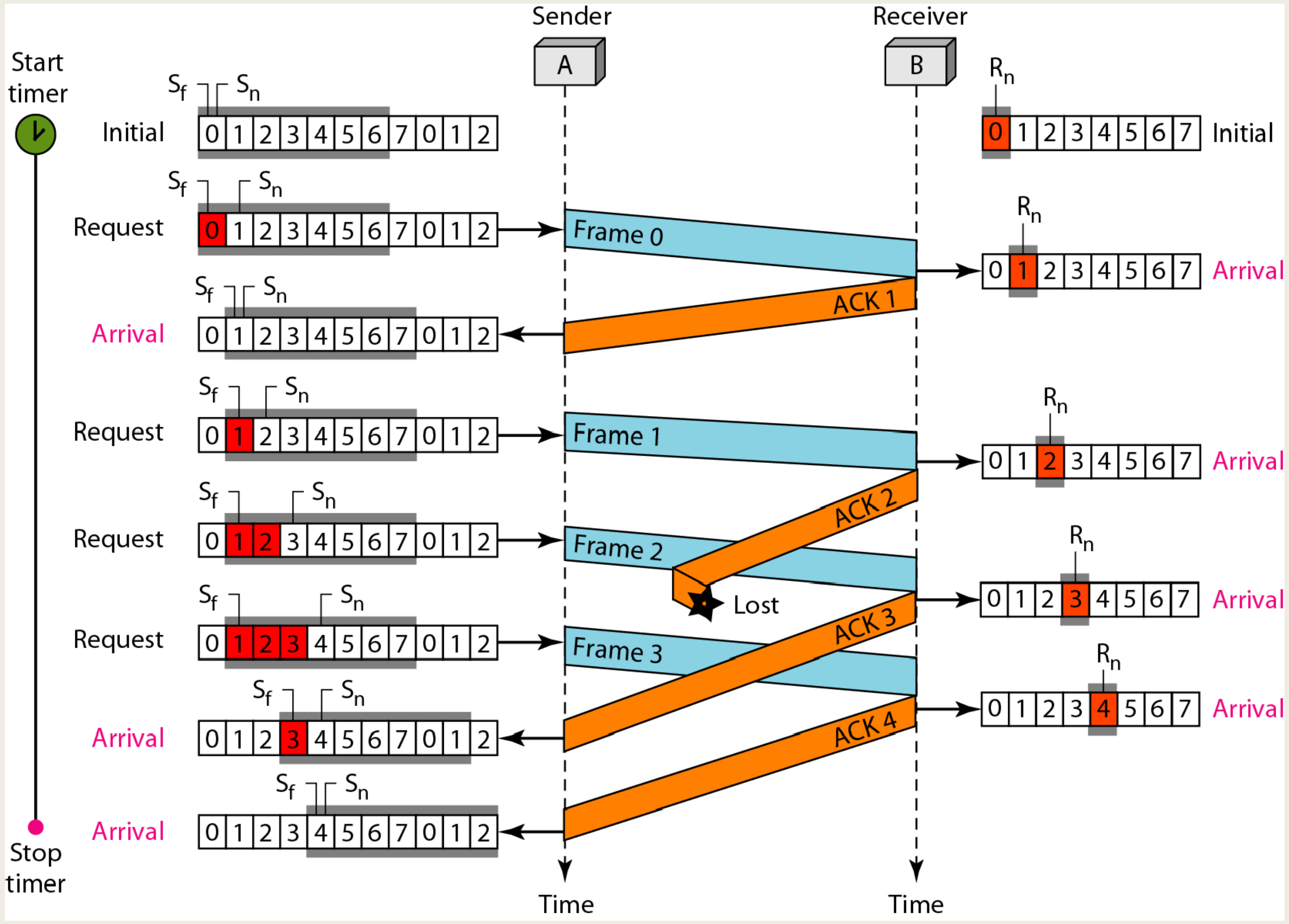
Source: B. A. Forouzan, " Data Communications and Networking ," *McGraw-Hill Forouzan Networking Series*,5E.

Example



Source: B. A. Forouzan, " Data Communications and Networking ," *McGraw-Hill Forouzan Networking Series*,5E.

Example



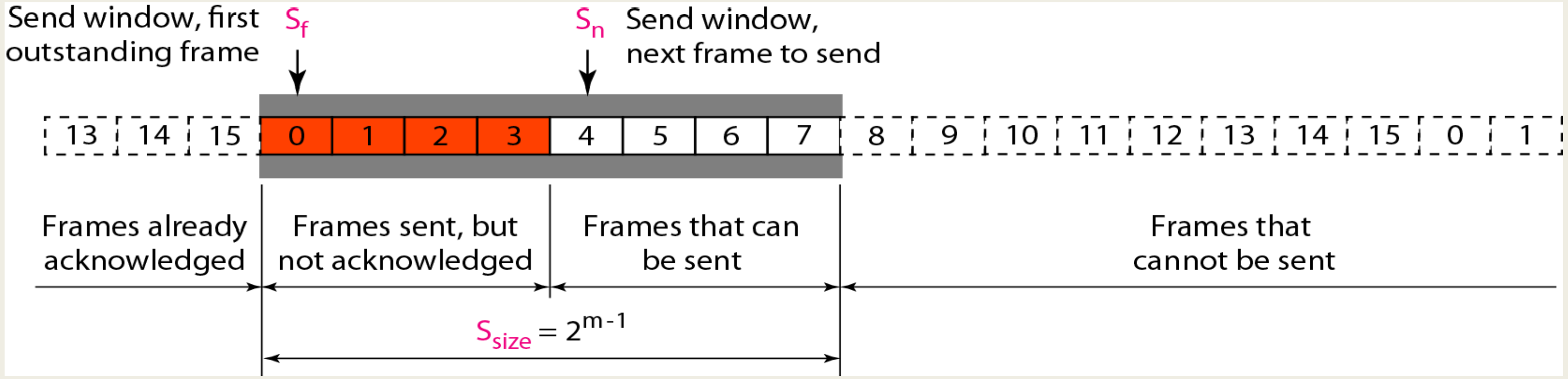
Source: B. A. Forouzan, " Data Communications and Networking ," *McGraw-Hill Forouzan Networking Series*,5E.

Selective Repeat ARQ

Does not resend *N* frames when just one frame is damaged; only the damaged frame is resent

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

More efficient for noisy links, but the processing at the receiver is more complex.



Send window, first outstanding frame
S_f
S_n
Send window, next frame to send
13 14 15 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 0 1
Frames already acknowledged
Frames sent, but not acknowledged
Frames that can be sent
Frames that cannot be sent
$S_{size} = 2^{m-1}$

Fig.: Send window for Selective Repeat ARQ

Source: B. A. Forouzan, " Data Communications and Networking ," *McGraw-Hill Forouzan Networking Series*,5E.

Selective Repeat ARQ: Receive Window

- The size of the receiver window is the same as the size of the sender window (2^{m-1}).
- The Selective Repeat Protocol allows as many frames as the size of the receive window to arrive out of order and be kept until there is a set of in-order frames to be delivered to the network layer.
- The sizes of the send window and receive window are the same, all the frames in the send frame can arrive out of order and be stored until they can be delivered.

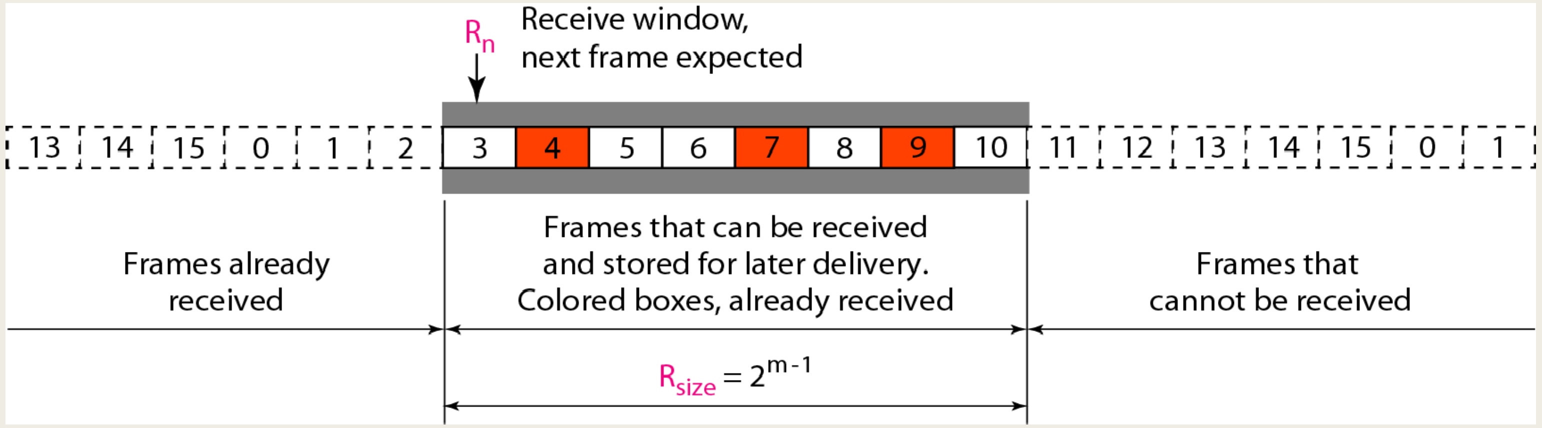
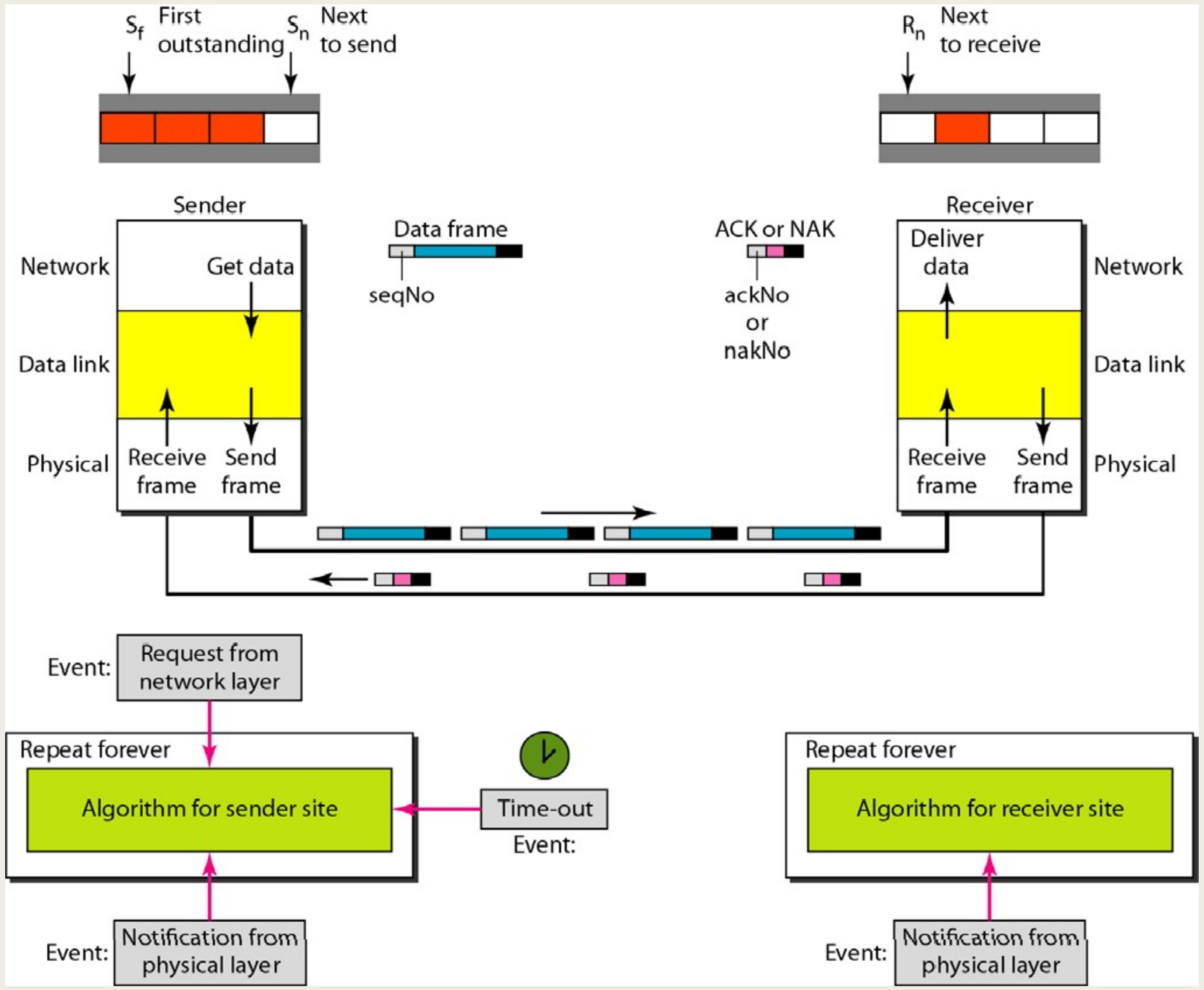

R_n
Receive window, next frame expected
13 14 15 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 0 1
Frames already received
Frames that can be received and stored for later delivery. Colored boxes, already received
Frames that cannot be received
$R_{size} = 2^{m-1}$

Fig.: Receive window for Selective Repeat ARQ

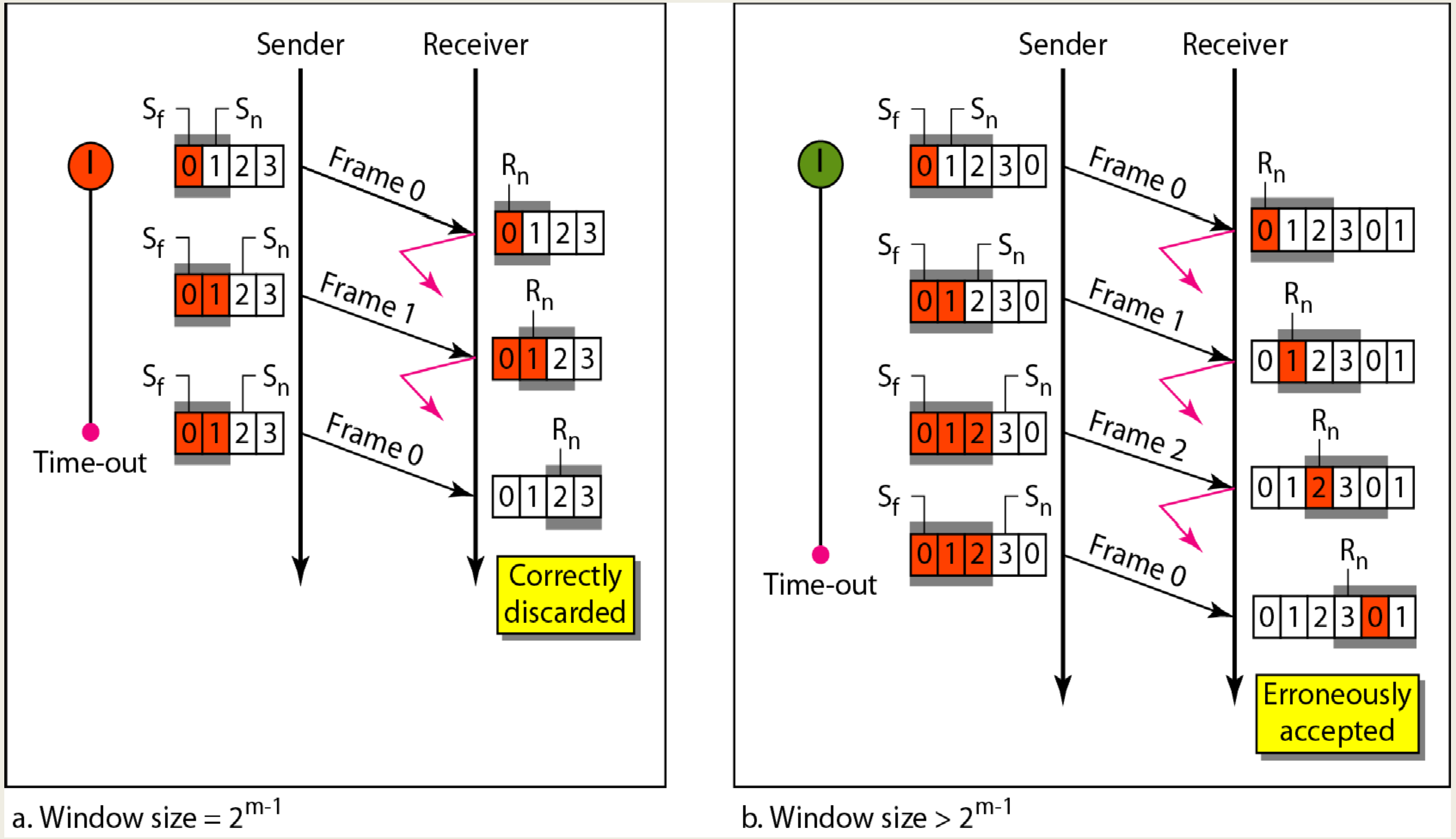
Source: B. A. Forouzan, " Data Communications and Networking ," *McGraw-Hill Forouzan Networking Series*,5E.

Selective Repeat ARQ: Design



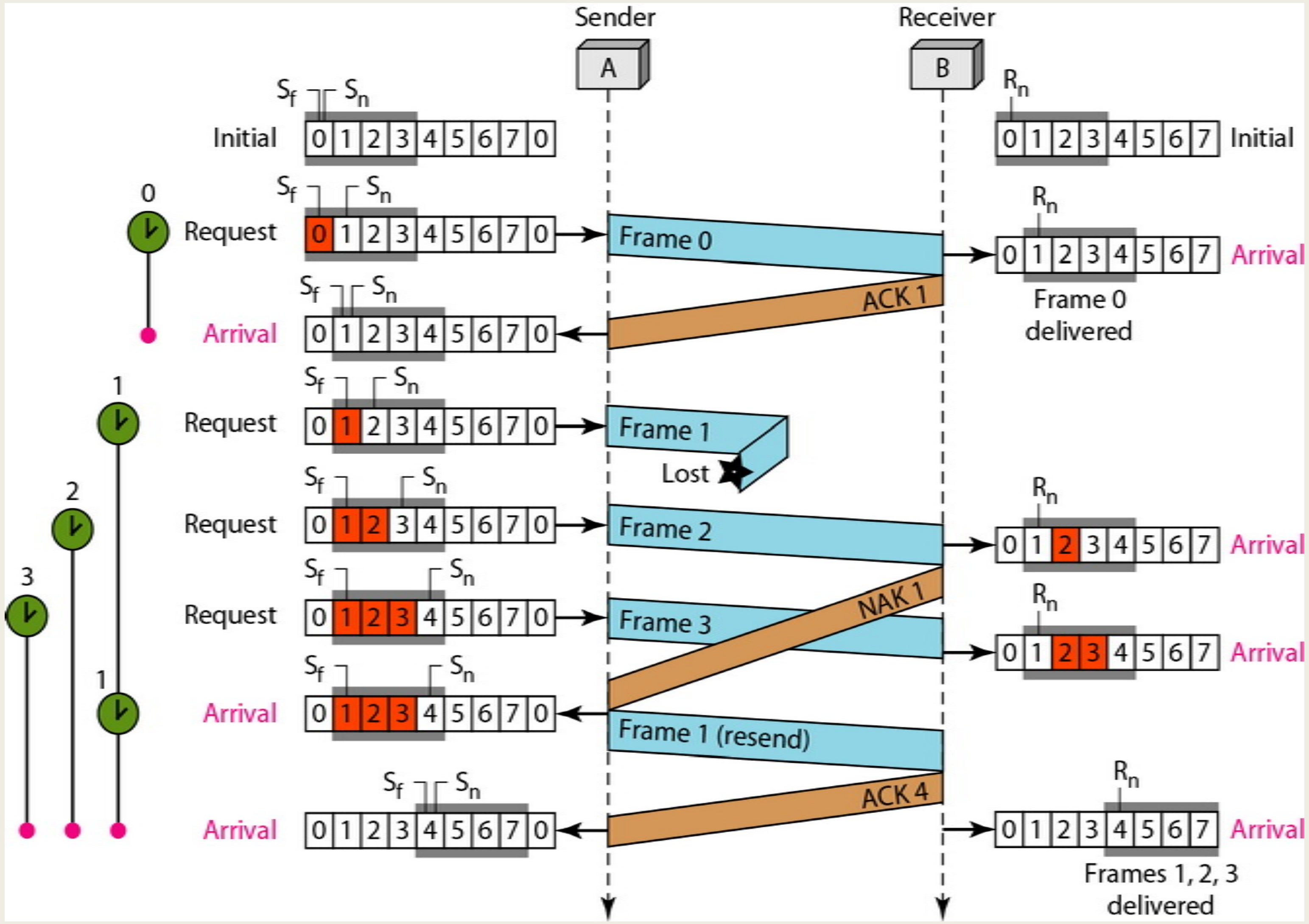
Source: B. A. Forouzan, " Data Communications and Networking ," *McGraw-Hill Forouzan Networking Series*,5E.

Selective Repeat ARQ: Window



https://docs.google.com/viewer?a=v&pid=sites&srcid=ZGVmYXVsdGRvbWFpbnxlZWJhaHJpYTV8Z3g6MjE2MzY2MjcxMGU3MmQ4Nw

Selective Repeat ARQ: Example



Source: B. A. Forouzan, " Data Communications and Networking ," *McGraw-Hill Forouzan Networking Series*,5E.

Thank You!!!