2.3 Data Link Control and Protocols

HDLC,
Stop and wait ,
Sliding Window ARQ techniques,

Data Link Control

Framing

Flow Control

Error Control

Protocols(software)

Framing

- Data bits packed into frames
- separates a message from other by adding sender and destination address.
- Can be fixed size (e.g. ATM) or
- variable size framing (e.g. LAN)
- Start and end of frame need to define
- Character oriented and bit oriented protocol

HDLC

Configurations and Transfer Modes

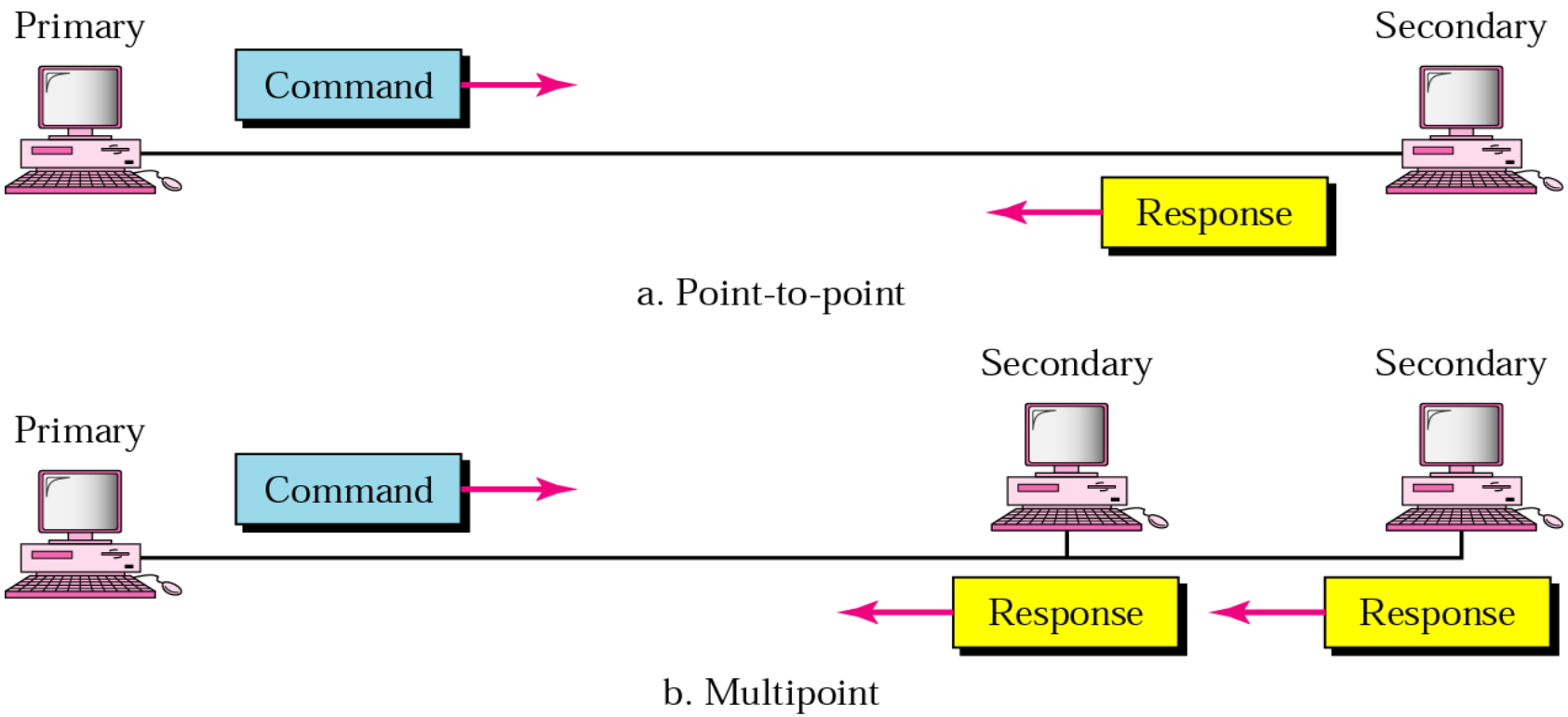
Frames

Frame Format

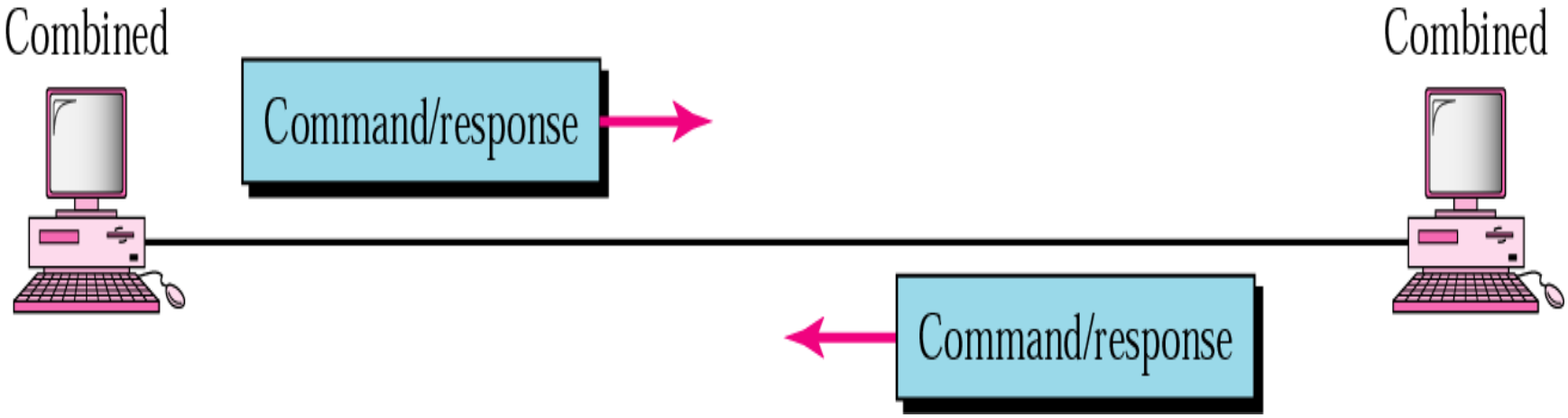
Examples

Data Transparency

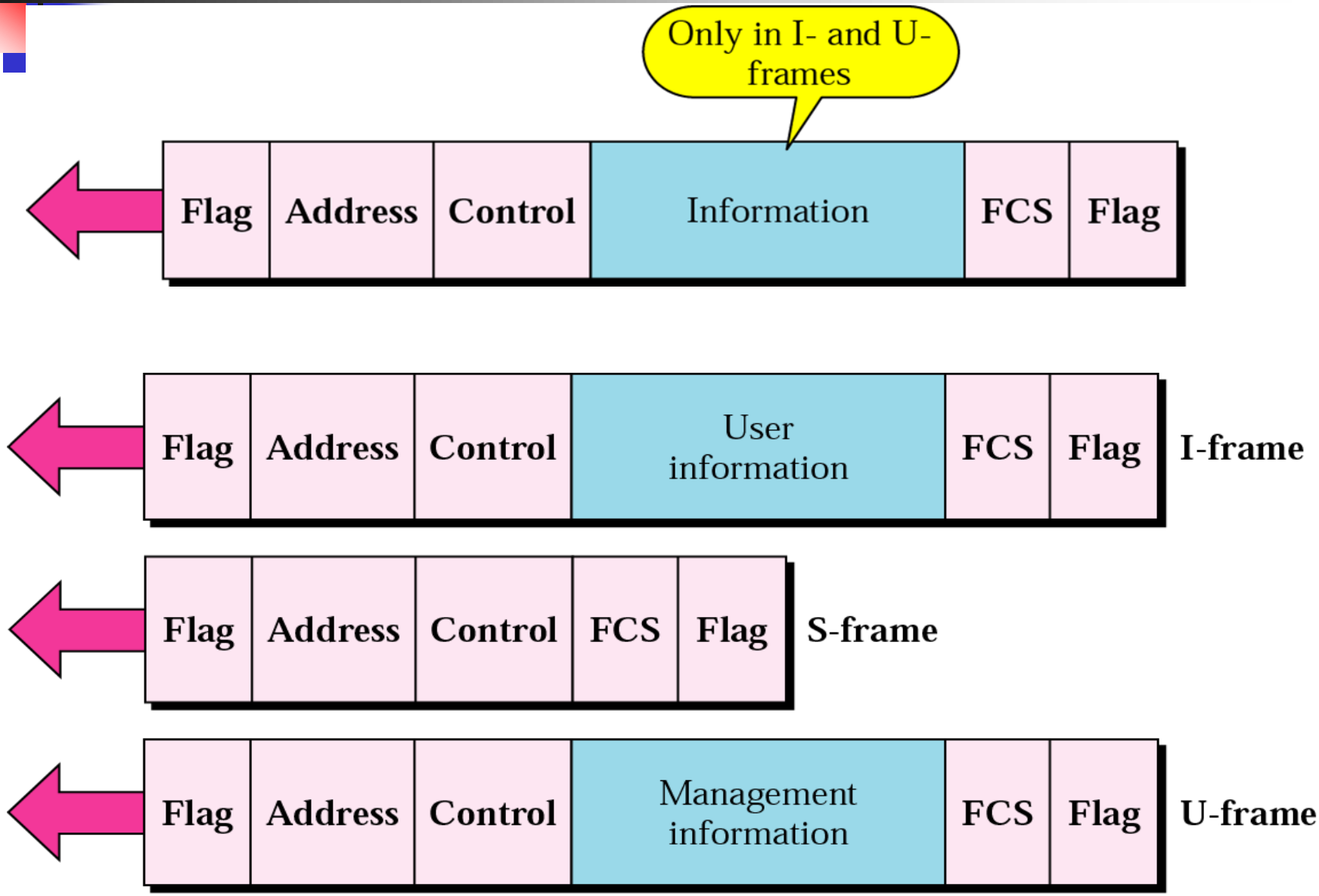
NRM



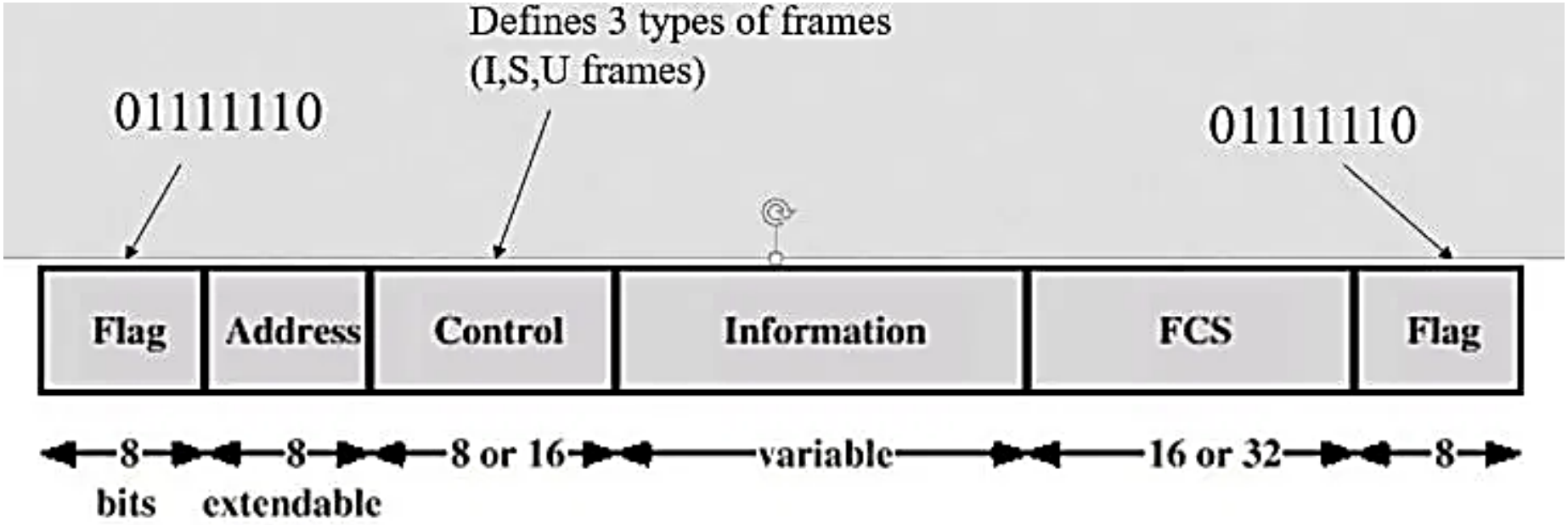
ABM



HDLC frame types



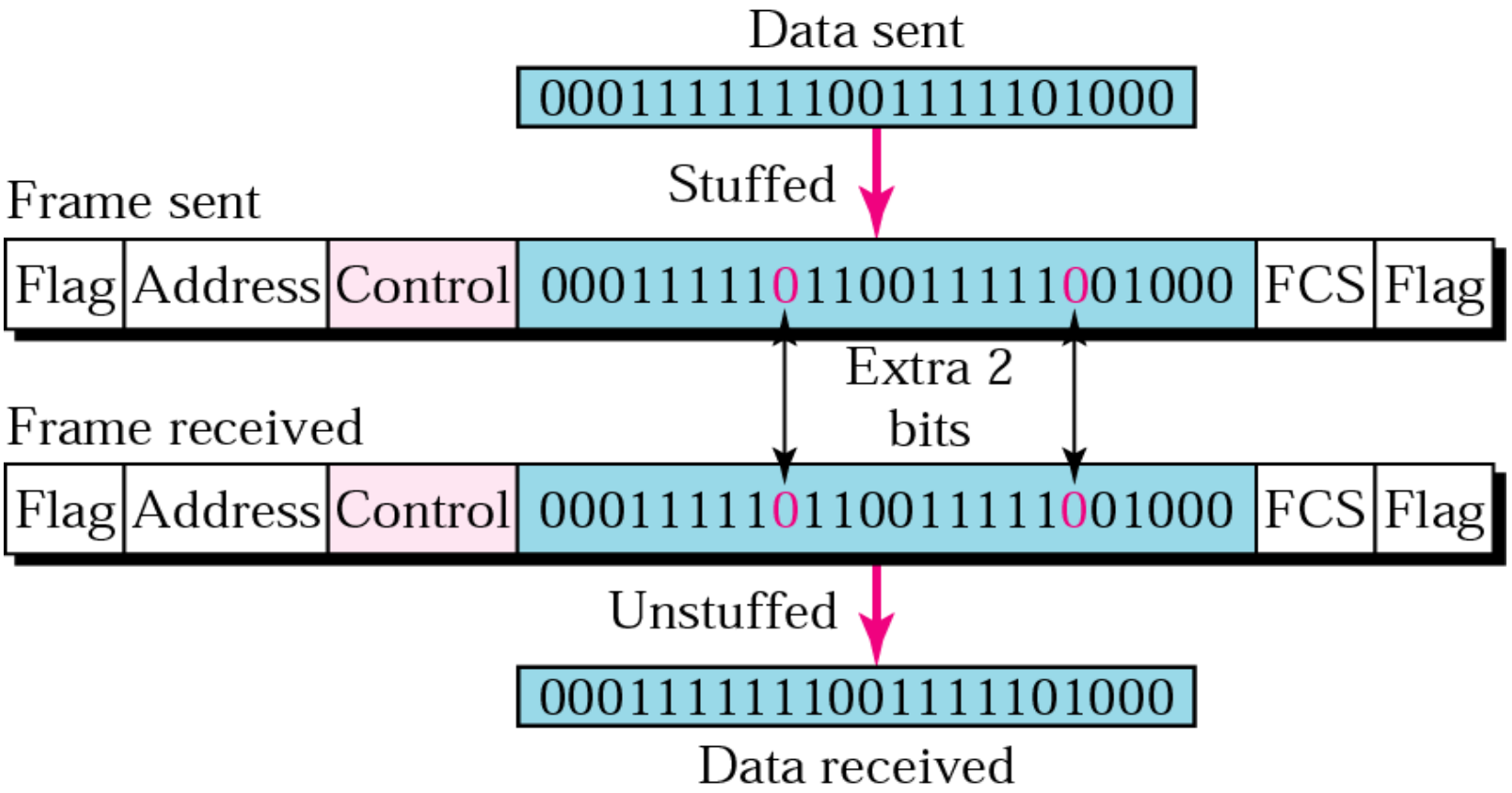
HDLC frame Format



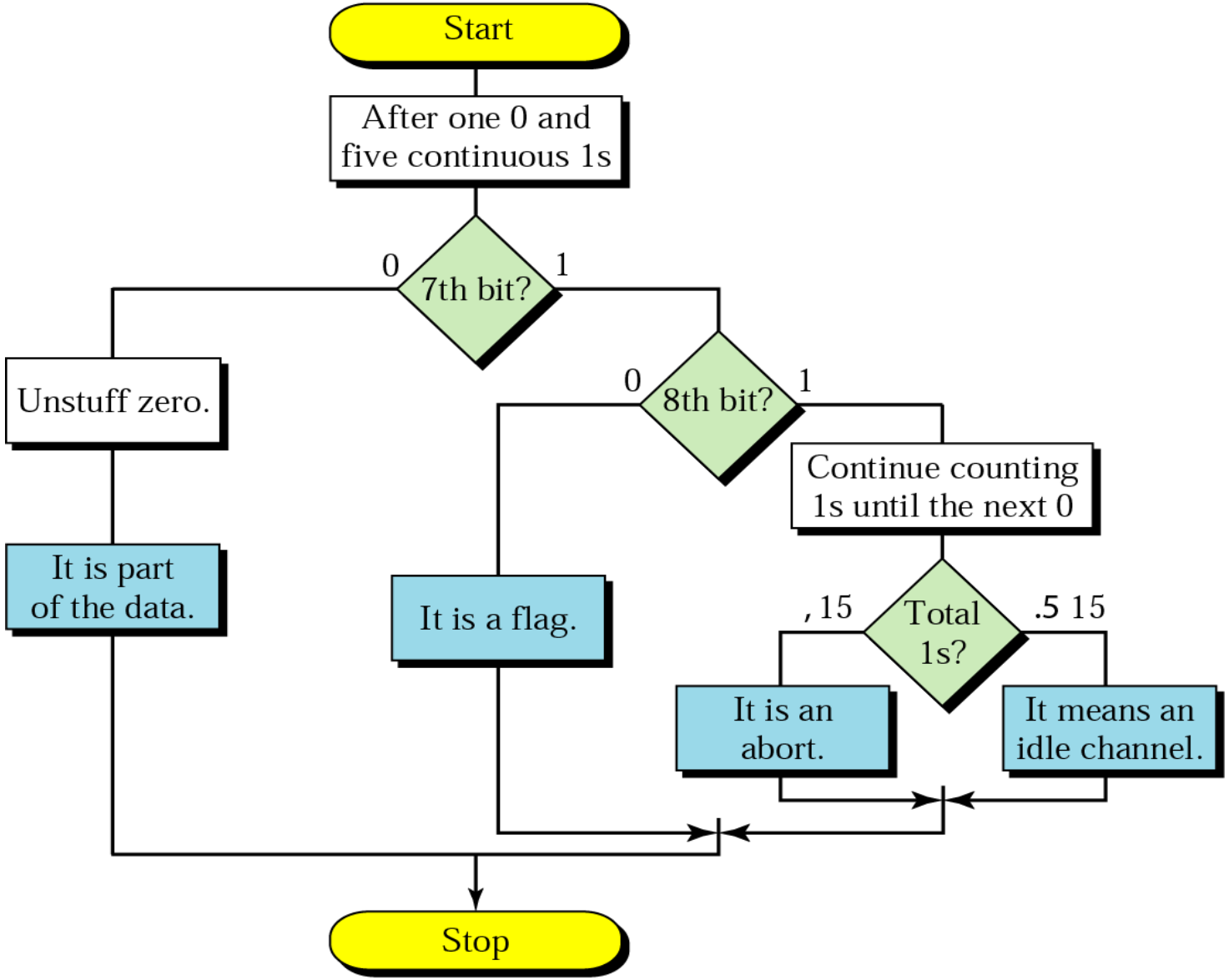
Note:

Bit stuffing is the process of adding one extra 0 whenever there are five consecutive 1s in the data so that the receiver does not mistake the data for a flag.

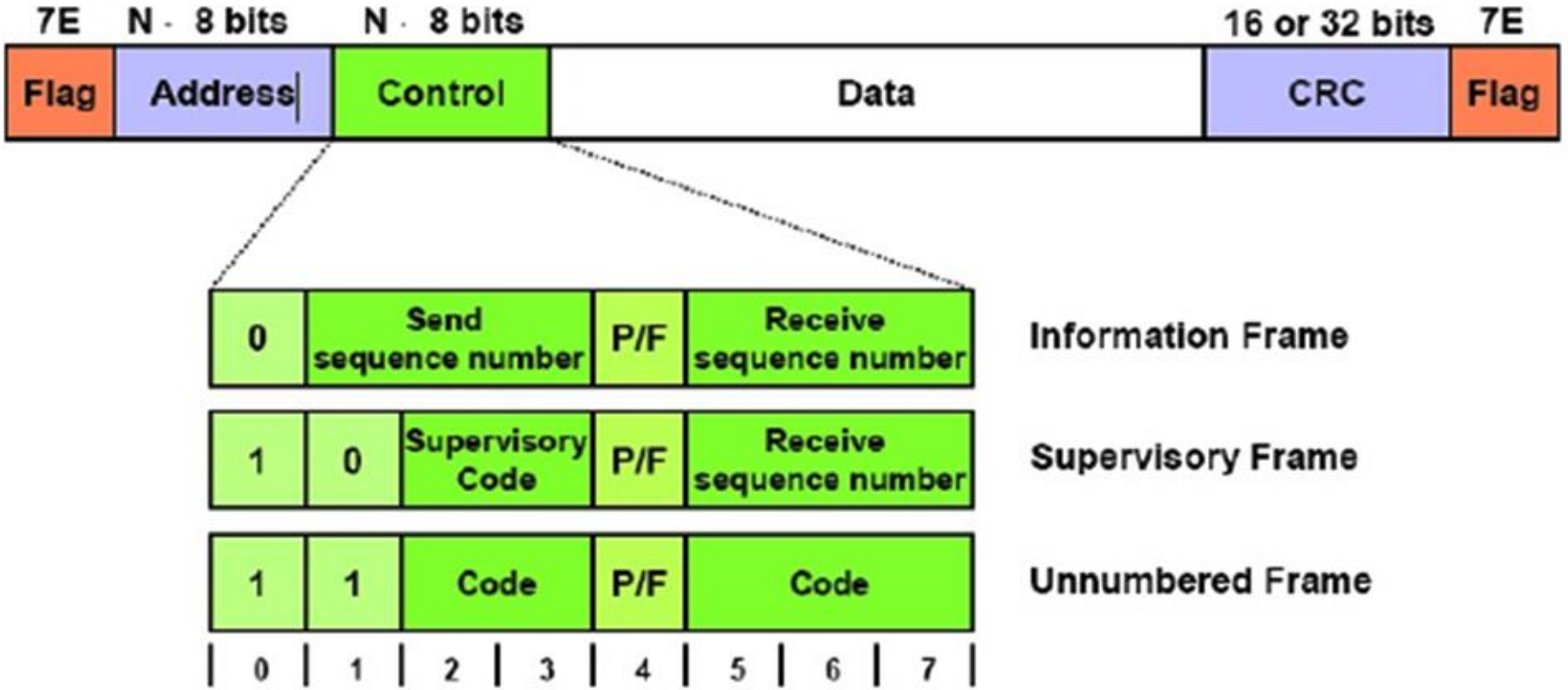
Bit stuffing and removal



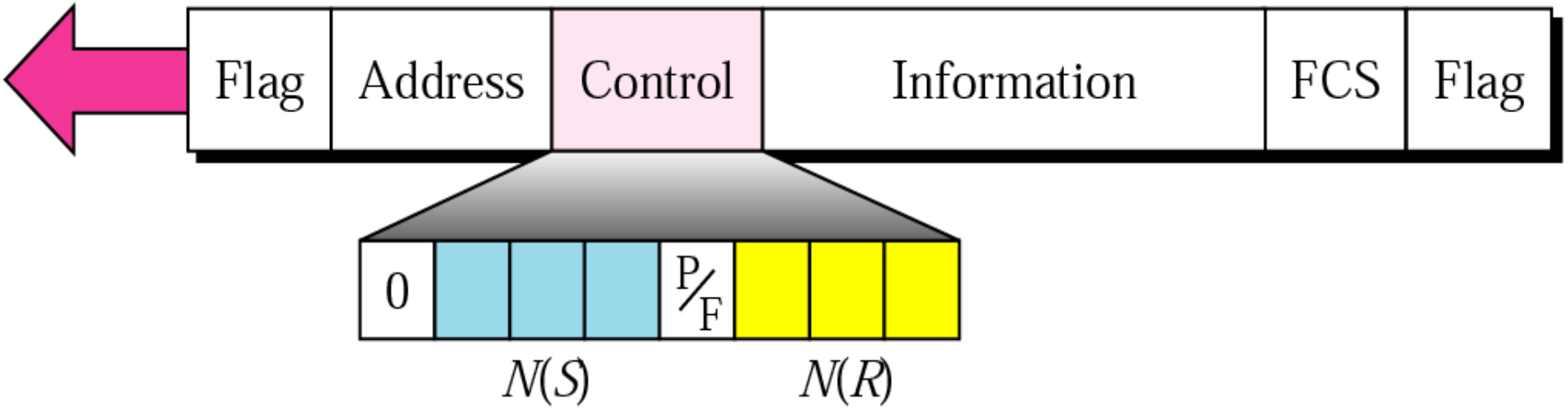
Bit stuffing and removal in HDLC



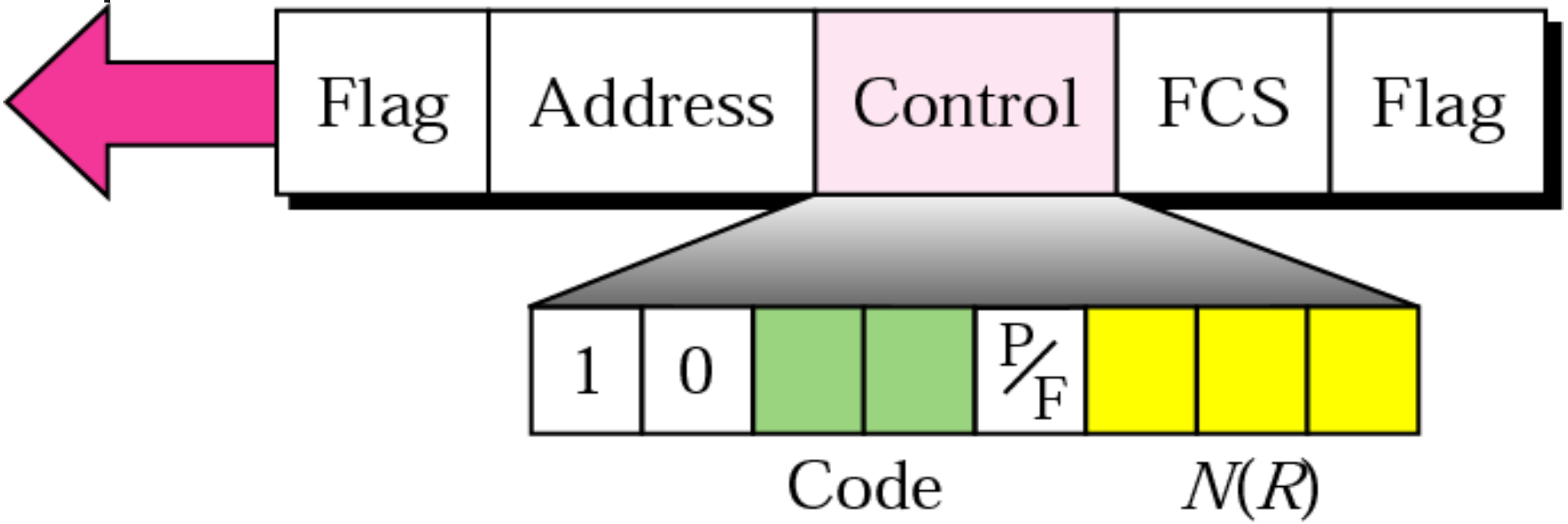
HDLC frame Format



I-frame

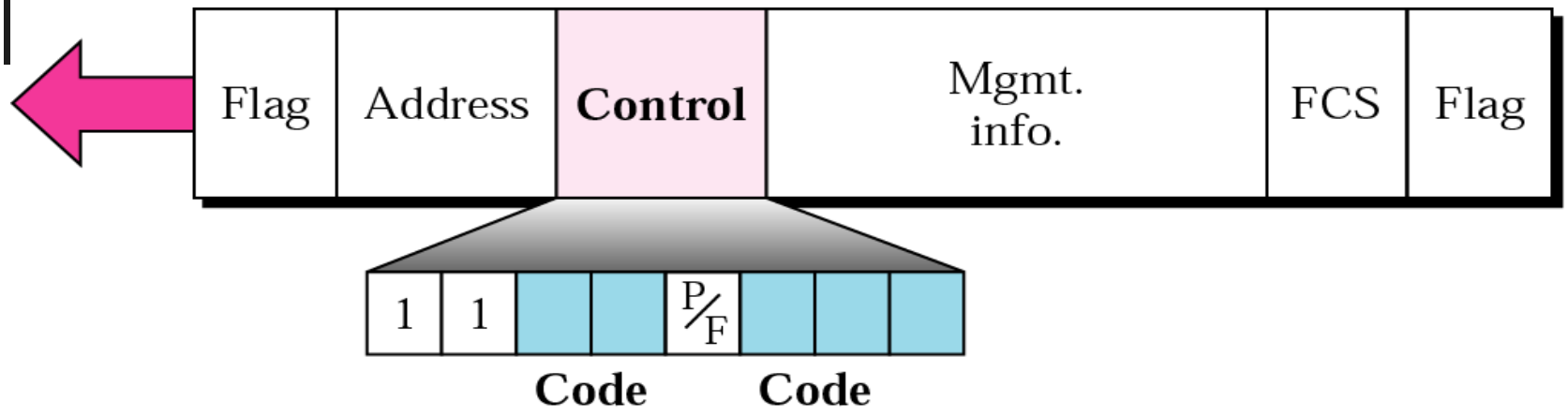


S-frame control field in HDLC



RR Receiver ready
RNR Receiver not ready
REJ Reject
SREJ Selective reject

U-frame control field in HDLC

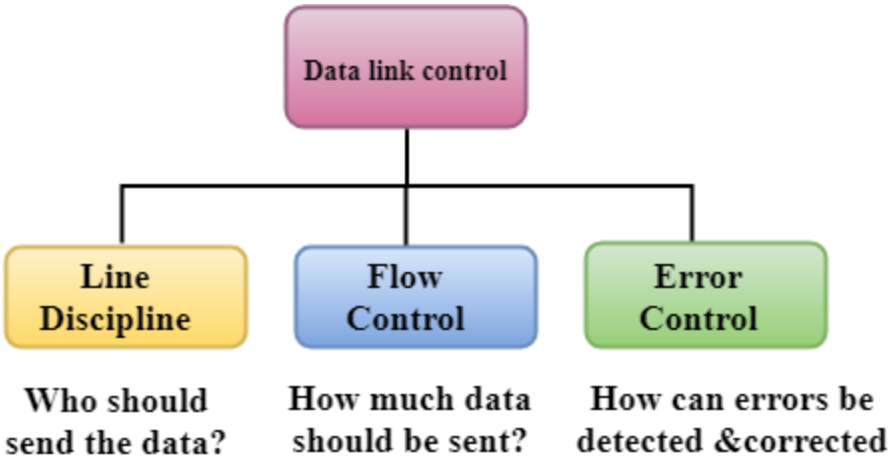


Code		Command	Response
00	001	SNRM	
11	011	SNRME	
11	100	SABM	DM
11	110	SABME	
00	000	UI	UI
00	110		UA
00	010	DISC	RD
10	000	SIM	RIM
00	100	UP	
11	001	RSET	
11	101	XID	XID
10	001		FRMR

U-frame control command and response

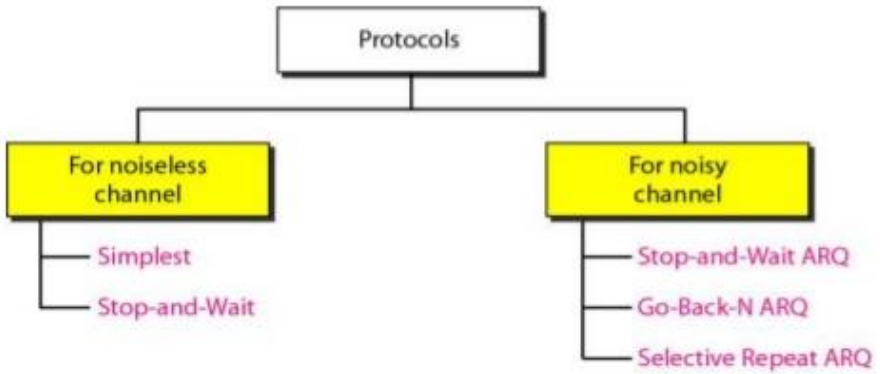
Command/response	Meaning
SNRM	Set normal response mode
SNRME	Set normal response mode (extended)
SABM	Set asynchronous balanced mode
SABME	Set asynchronous balanced mode (extended)
UP	Unnumbered poll
UI	Unnumbered information
UA	Unnumbered acknowledgment
RD	Request disconnect
DISC	Disconnect
DM	Disconnect mode
RIM	Request information mode
SIM	Set initialization mode
RSET	Reset
XID	Exchange ID
FRMR	Frame reject

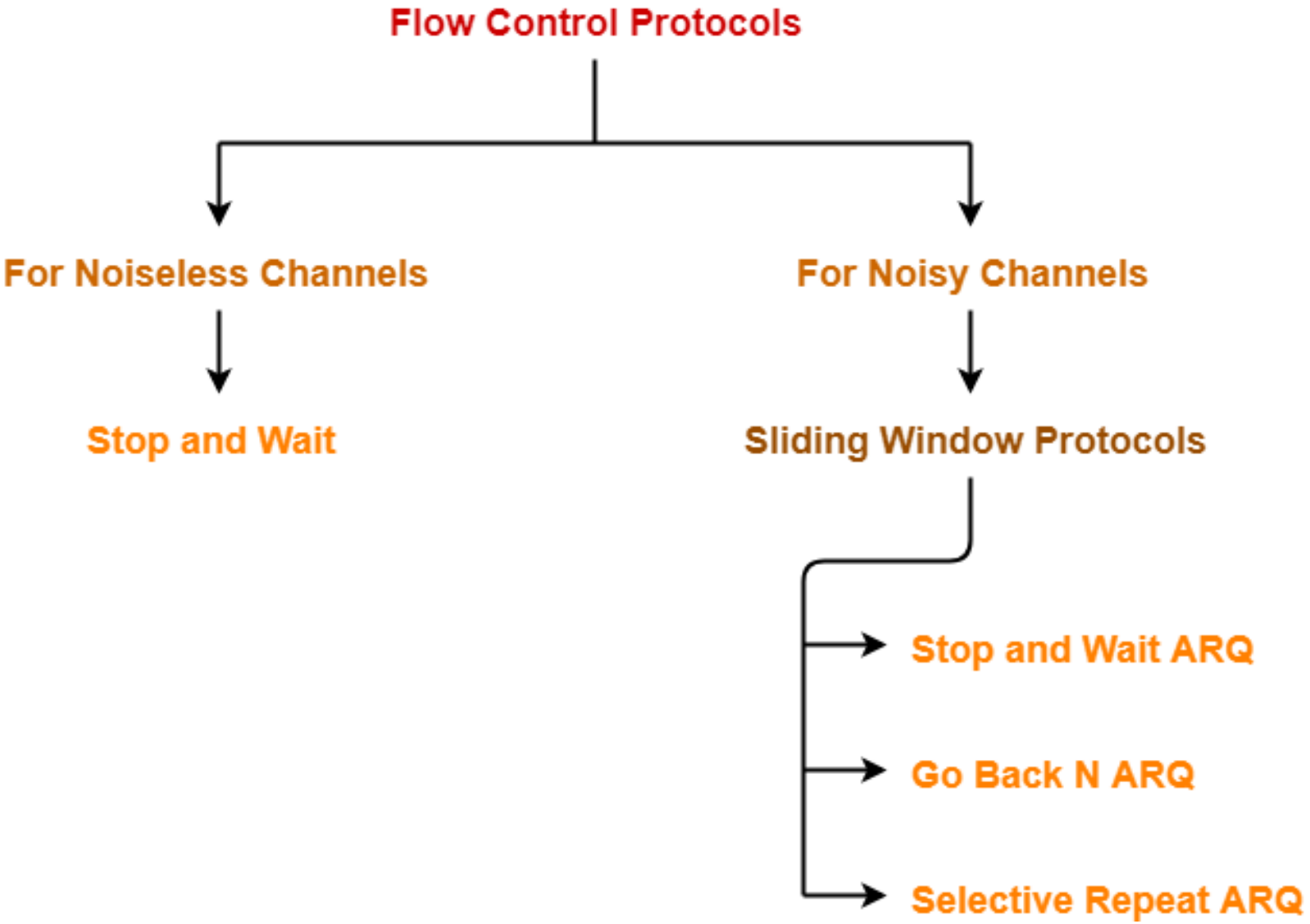
Data Link Control



Flow and Error Control

- *Data link control = flow control + error control*
- *Flow control* refers to a set of procedures used to restrict the amount of data that the sender can send before waiting for acknowledgement
- *Error control* in the data link layer is based on *automatic repeat request* (ARQ), which is the retransmission of data
- All the protocols are unidirectional i.e. the data frames travel from source to destination node. Only ACK, NAK(Negative ACK) frames can flow in opposite direction.





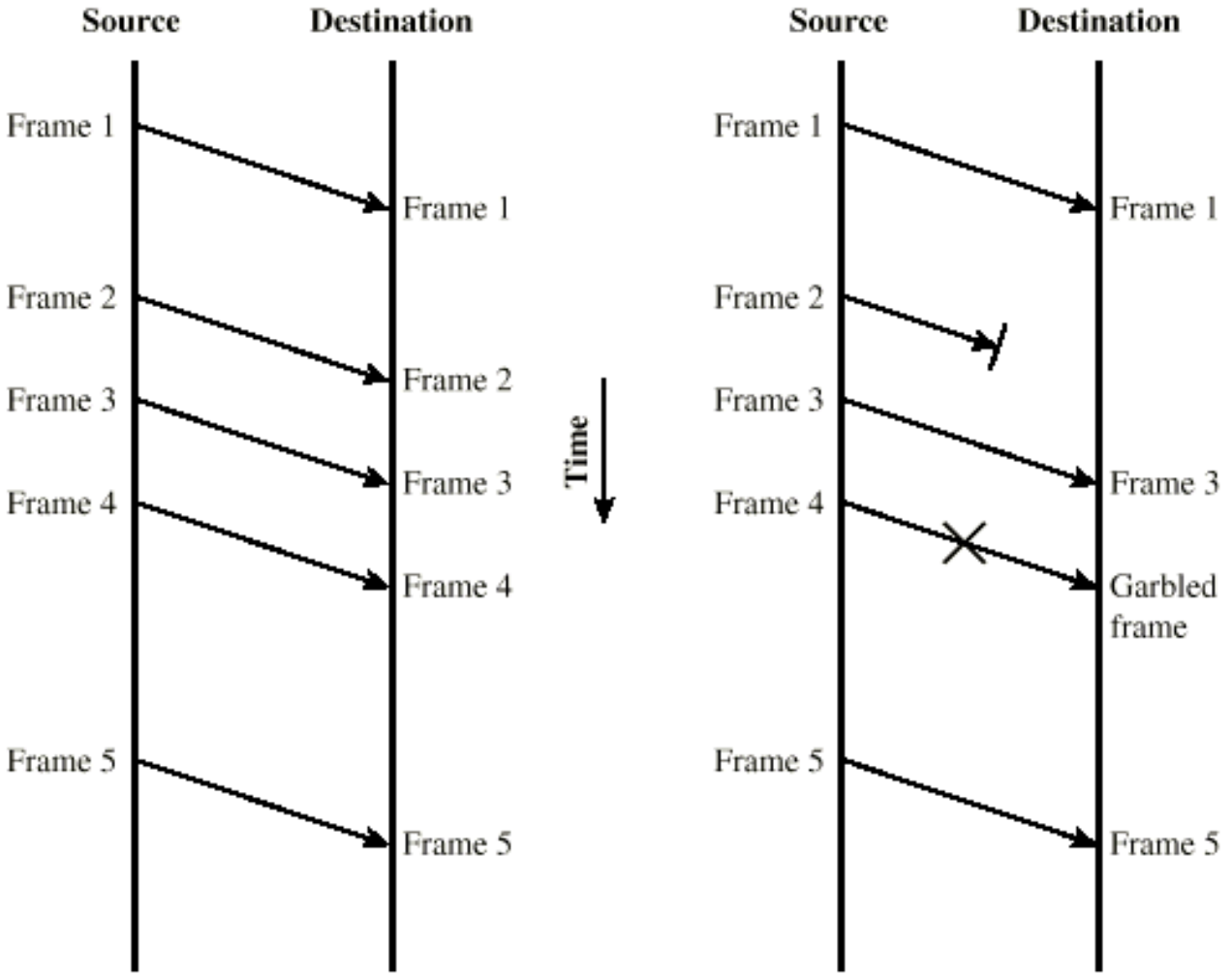
Note:

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

Flow Control

- Ensuring the sending entity does not overwhelm the receiving entity
  - Preventing buffer overflow
- Transmission time
  - Time taken to emit all bits into medium
- Propagation time
  - Time for a bit to traverse the link

Model of Frame Transmission



(a) Error-free transmission

(b) Transmission with losses and errors

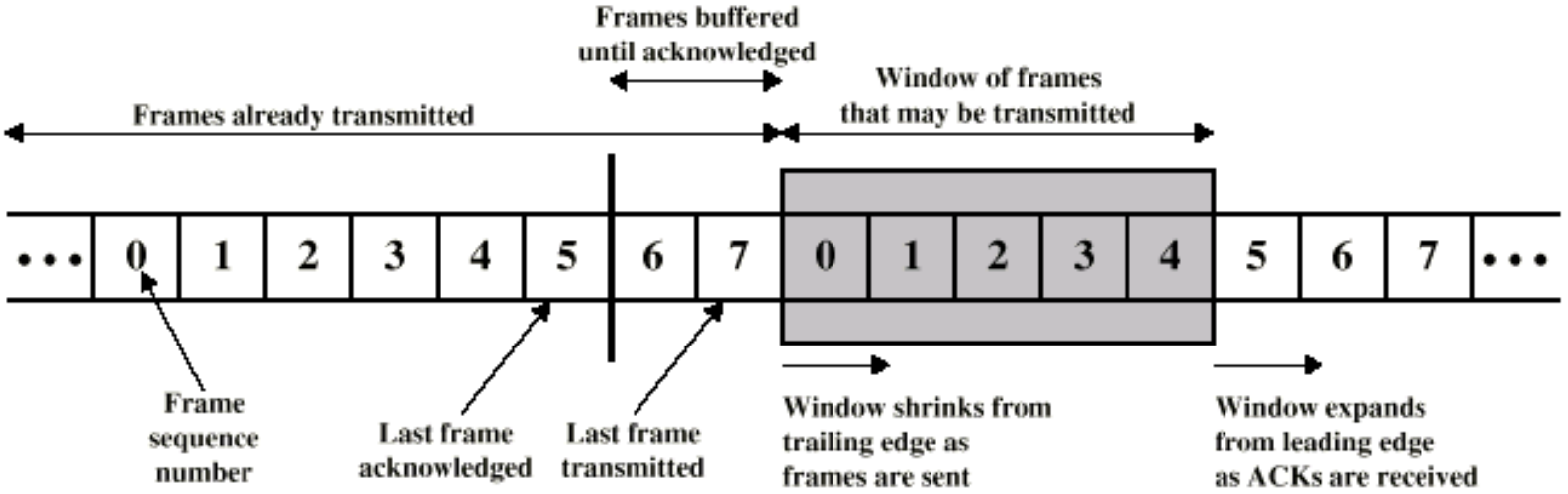
Stop and Wait

- Source transmits frame
- Destination receives frame and replies with acknowledgement
- Source waits for ACK before sending next frame
- Destination can stop flow by not send ACK
- Works well for a few large frames

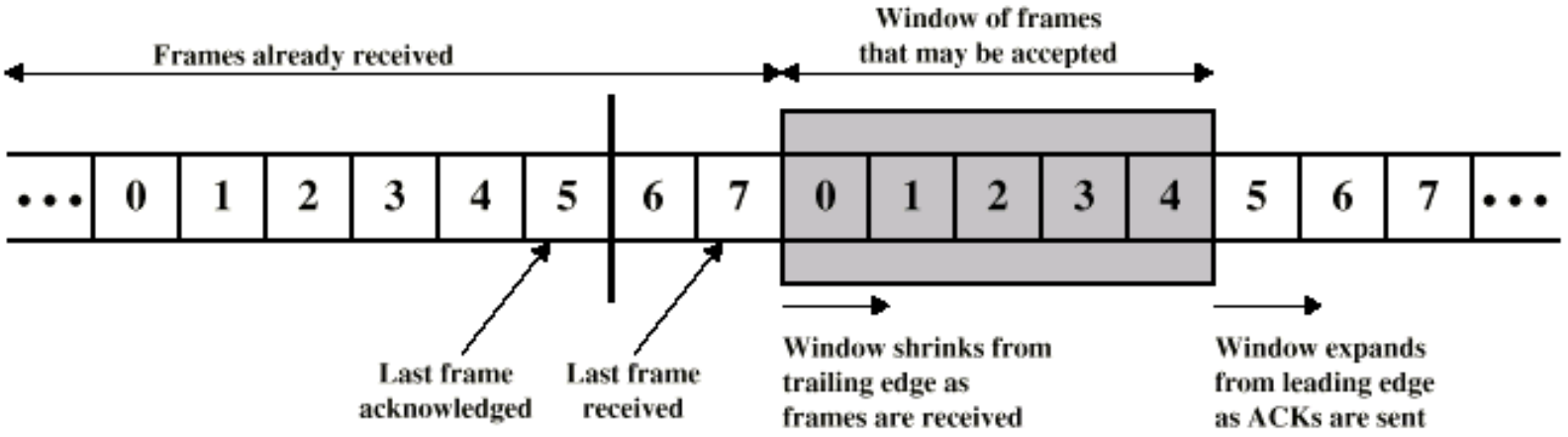
Sliding Windows Flow Control

- Allow multiple frames to be in transit
- Receiver has buffer W long
- Transmitter can send up to W frames without ACK
- Each frame is numbered
- ACK includes number of next frame expected
- Sequence number bounded by size of field (k)
  - Frames are numbered modulo 2^k

Sliding Window Diagram



(a) Sender's perspective



(b) Receiver's perspective

Example 1

In a Stop-and-Wait ARQ system, the bandwidth of the line is 1 Mbps, and 1 bit takes 20 ms to make a round trip. What is the bandwidth-delay product? If the system data frames are 1000 bits in length, what is the utilization percentage of the link?

Solution

The bandwidth-delay product is

$$1 \times 10^6 \times 20 \times 10^{-3} = 20{,}000 \text{ bits}$$

The system can send 20,000 bits during the time it takes for the data to go from the sender to the receiver and then back again. However, the system sends only 1000 bits. We can say that the link utilization is only 1000/20,000, or 5%. For this reason, for a link with high bandwidth or long delay, use of Stop-and-Wait ARQ wastes the capacity of the link.

Example 2

What is the utilization percentage of the link in Example 1 if the link uses Go-Back-N ARQ with a 15-frame sequence?

Solution

The bandwidth-delay product is still 20,000. The system can send up to 15 frames or 15,000 bits during a round trip. This means the utilization is 15,000/20,000, or 75 percent. Of course, if there are damaged frames, the utilization percentage is much less because frames have to be resent.

Error Detection Techniques

- Additional bits added by transmitter for error detection code
- CRC
- Internet checksum
- 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

Error Control

- Error Detection Lost frames

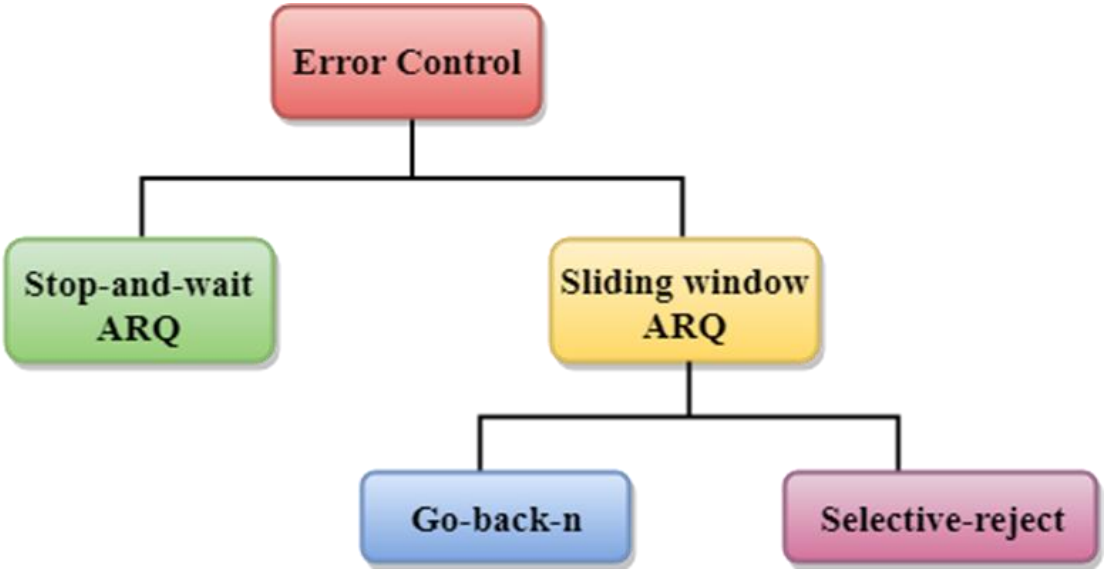
Damaged frames

Lost acknowledgment

- Error Control (Automatic repeat request)
  - Error detection
  - Positive acknowledgment
  - Retransmission after timeout
  - Negative acknowledgement and retransmission

Automatic Repeat Request (ARQ)

- Stop and wait
- Go back N
- Selective reject (selective retransmission)



Note:

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

Note:

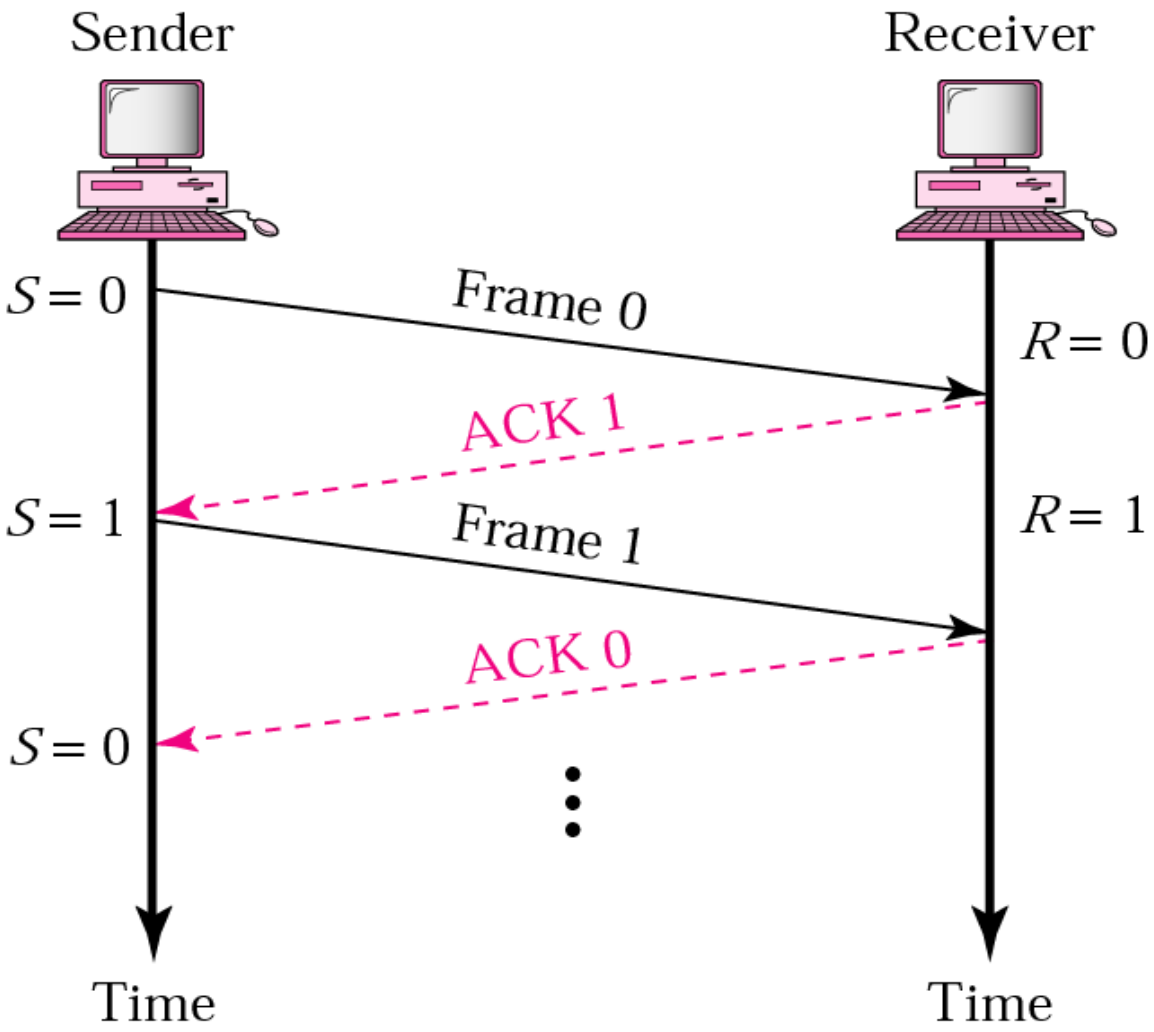
In ARQ, numbering frames prevents the retaining of duplicate frames.

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 ACK0 and ACK1

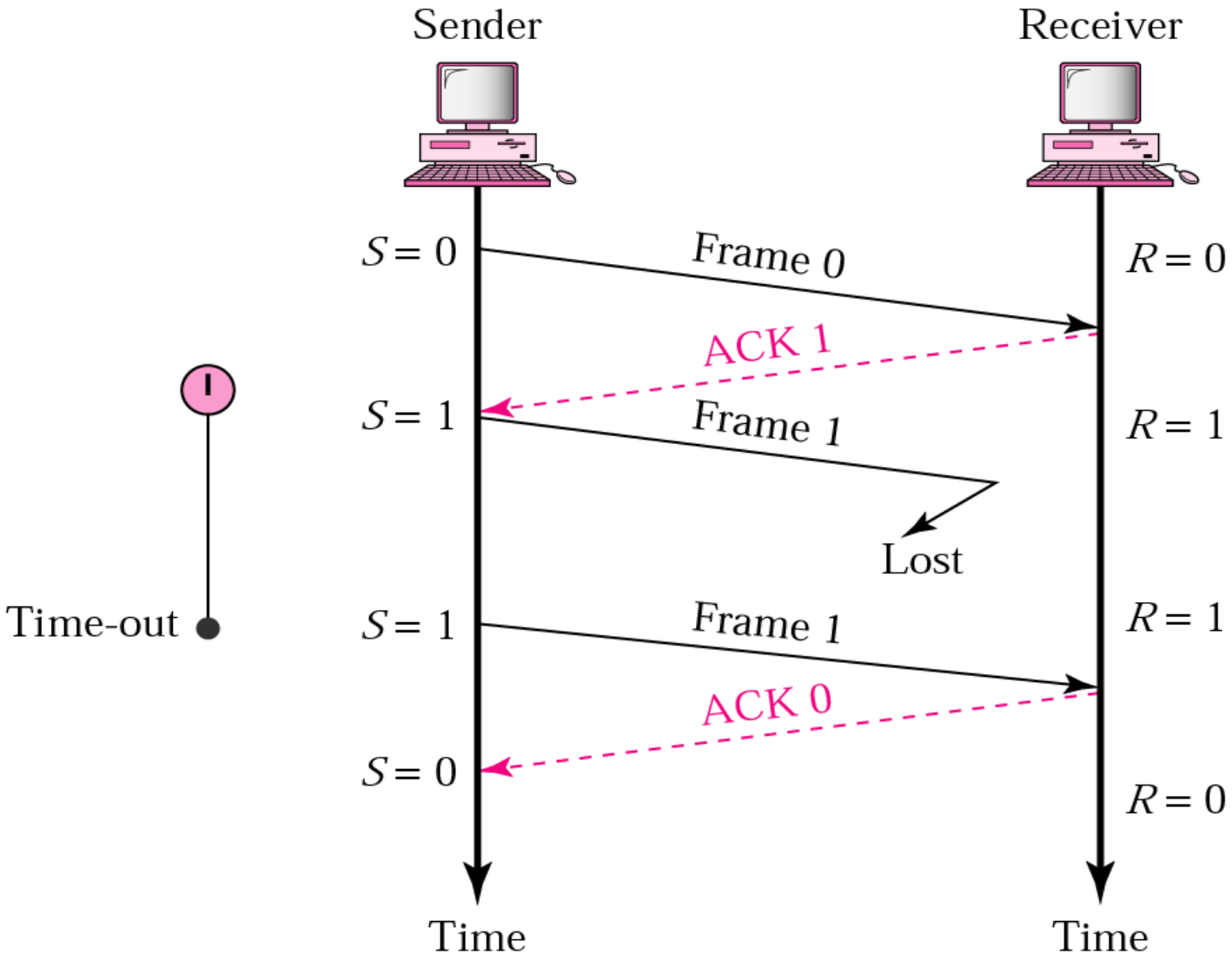
Stop and Wait

Normal operation



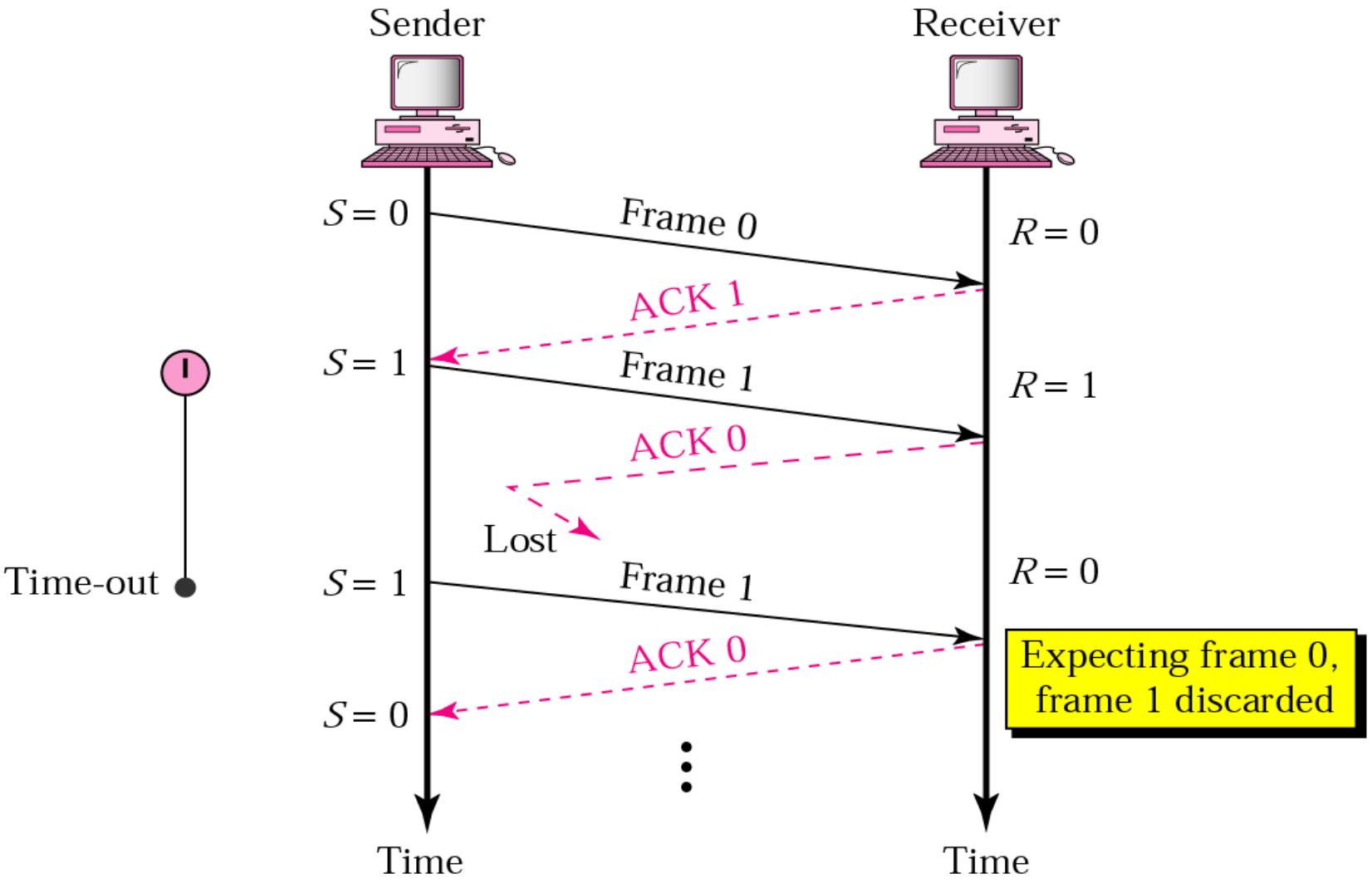
Stop and Wait

a. Stop-and-Wait ARQ, lost frame



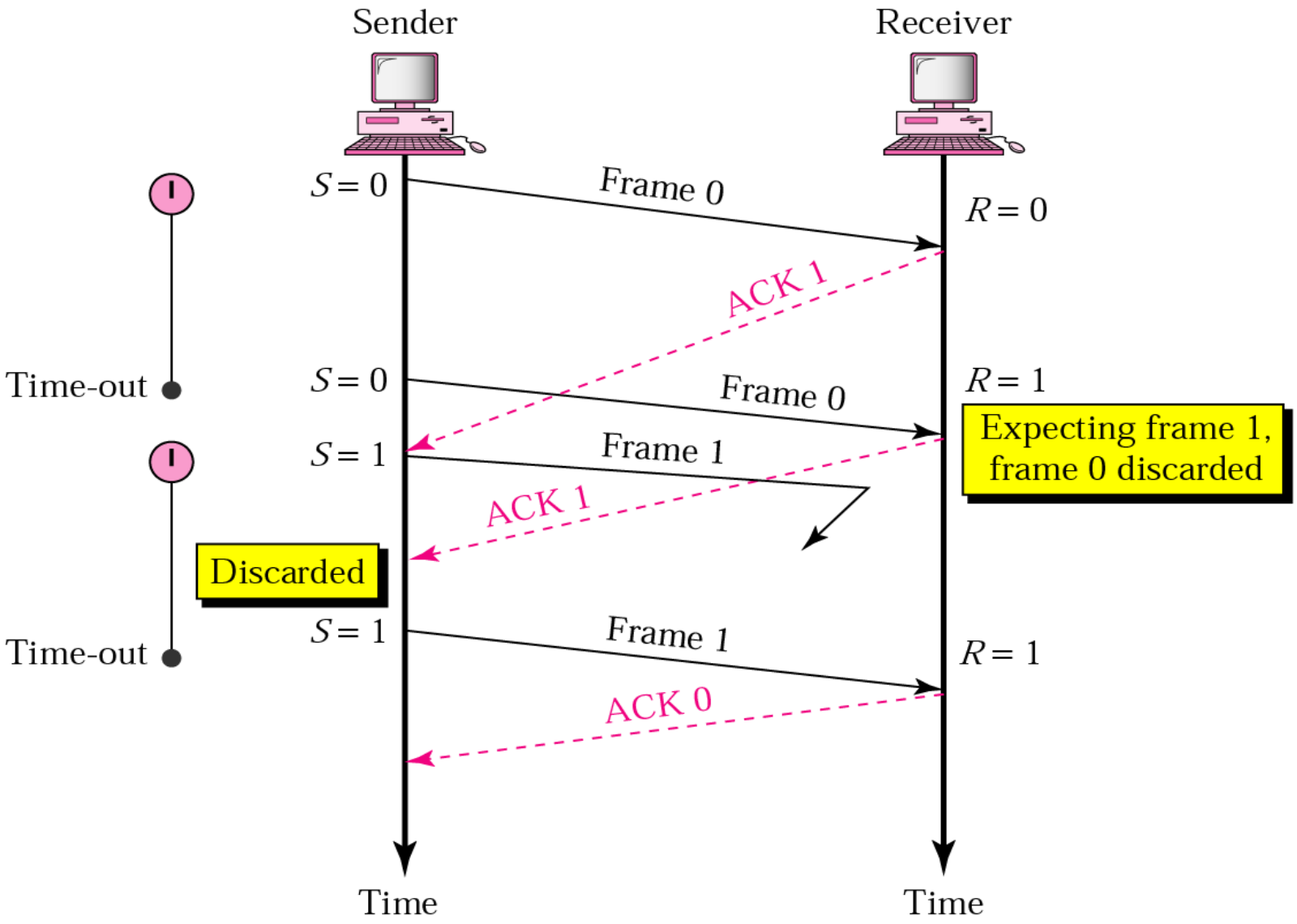
Stop and Wait

b. Stop-and-Wait ARQ, lost ACK frame

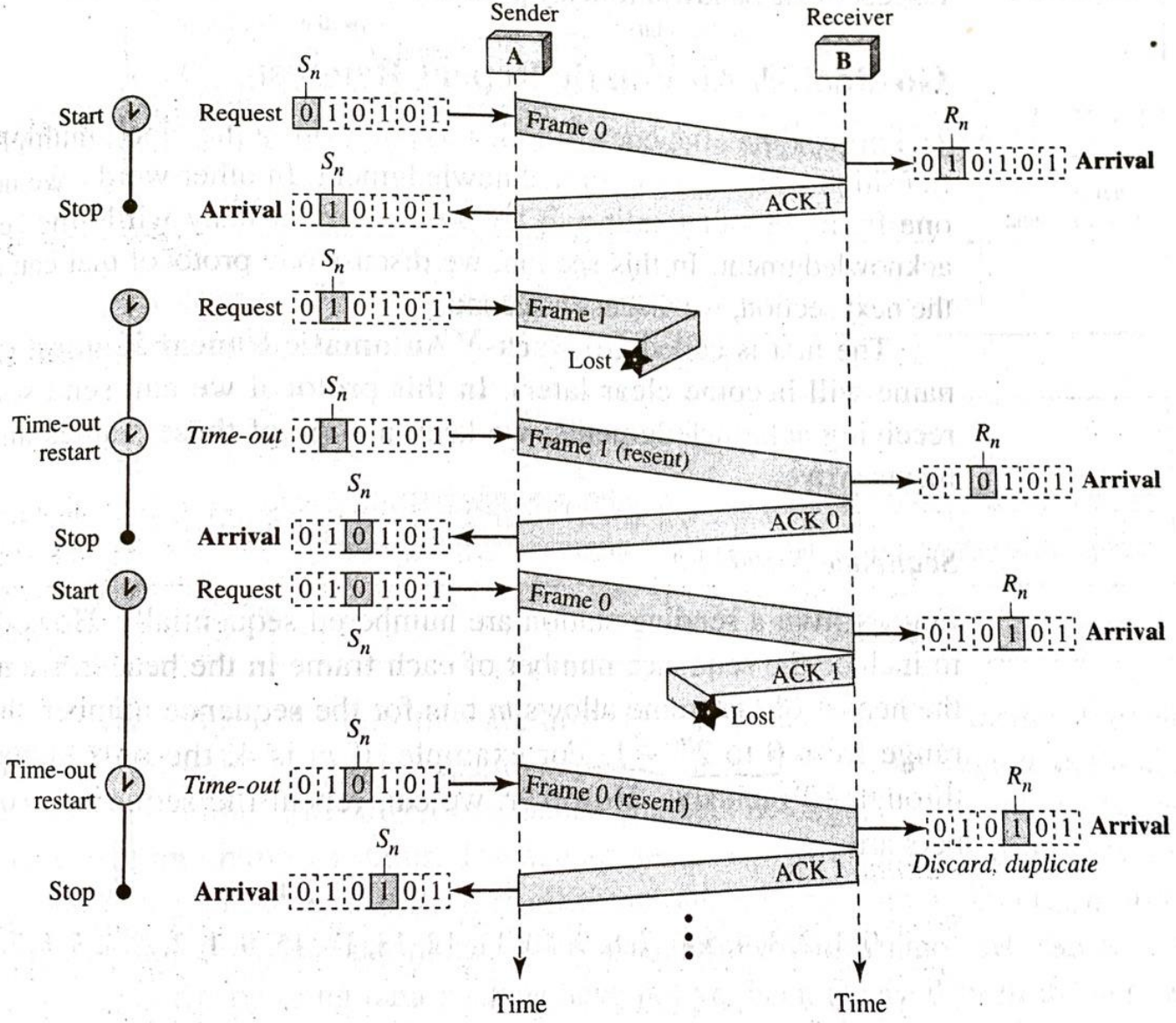


Stop and Wait

c. Stop-and-Wait ARQ, delayed ACK



Example



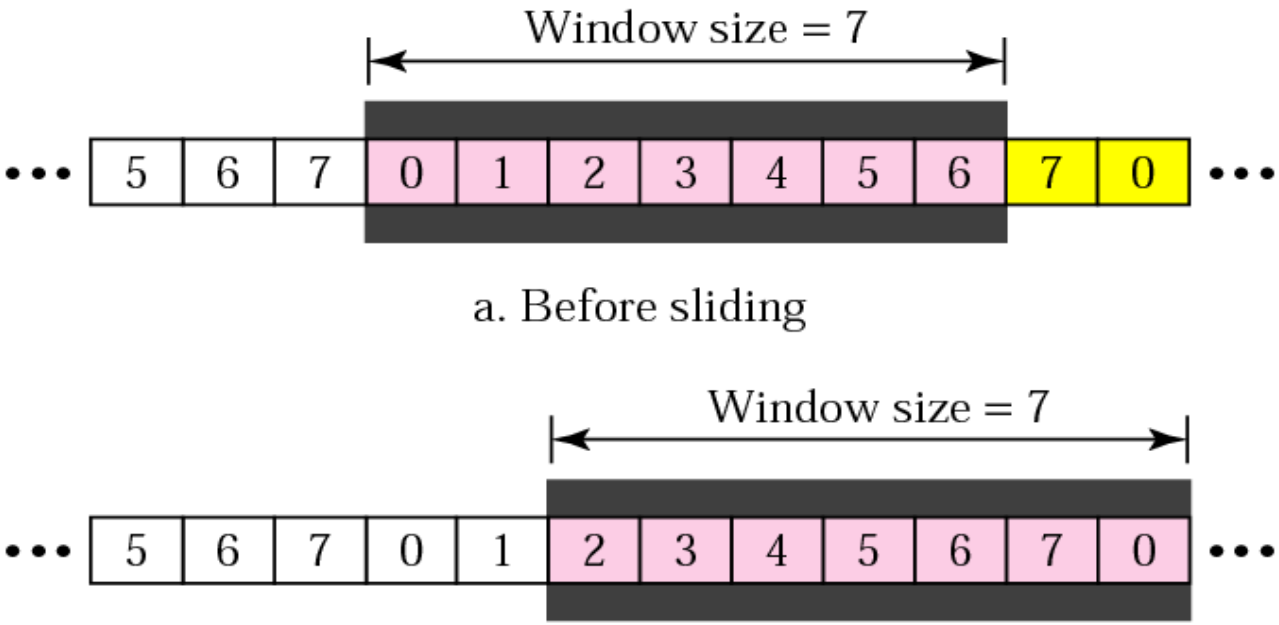
Go-Back-N ARQ

Note:

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

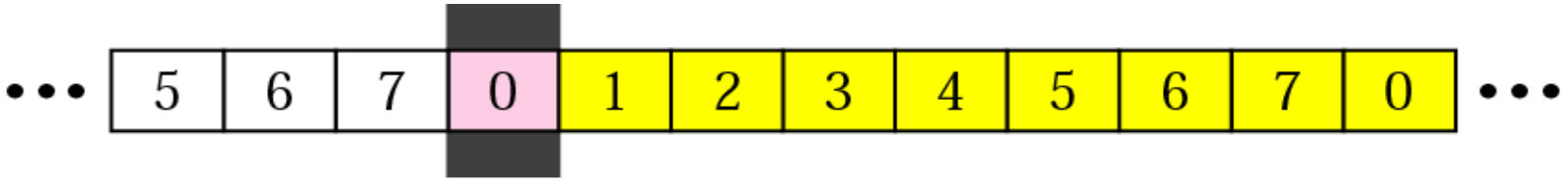
Go-Back-N ARQ

Sender sliding window

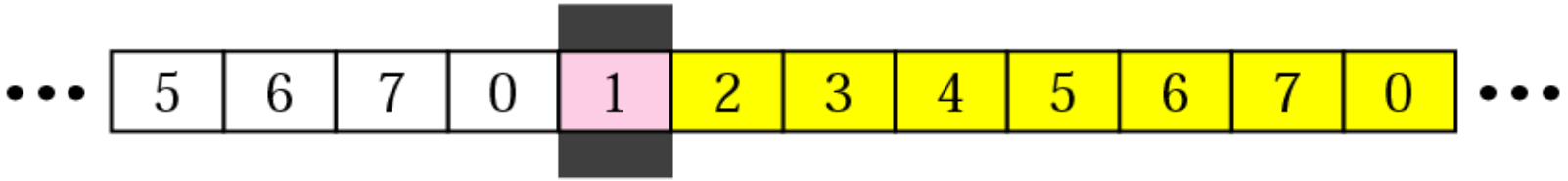


Go-Back-N ARQ

Receiver sliding window



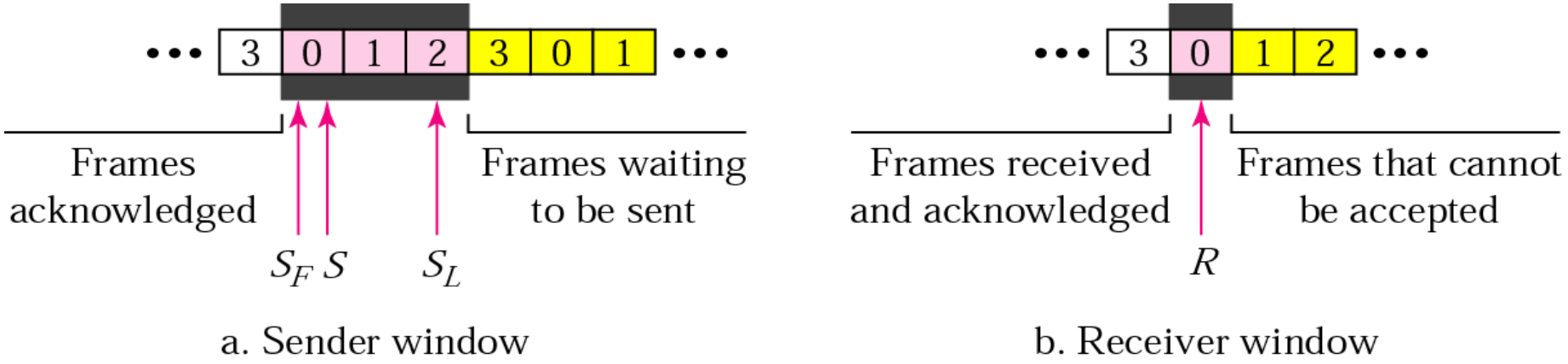
a. Before sliding



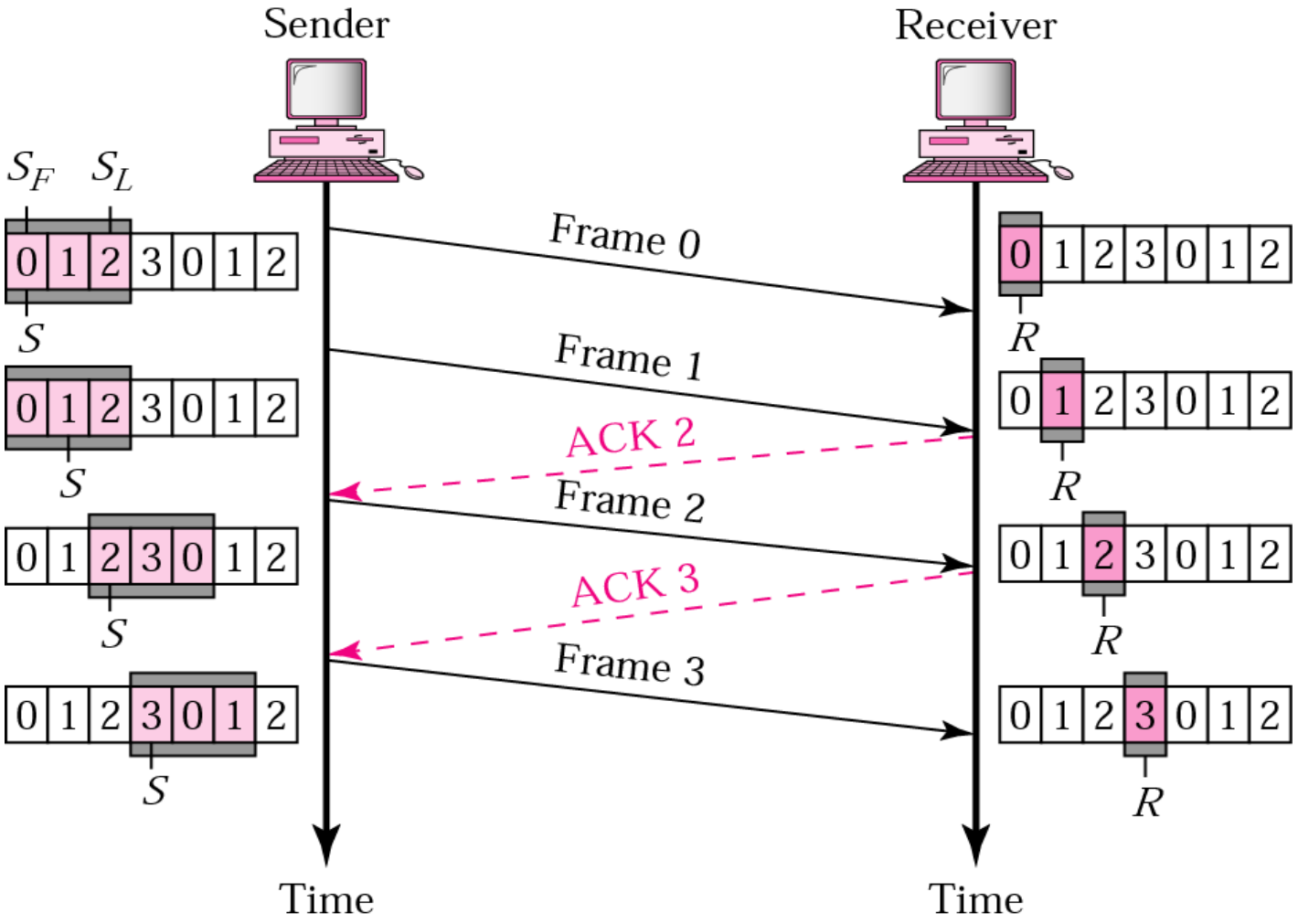
b. After sliding

Go-Back-N ARQ

Control variables

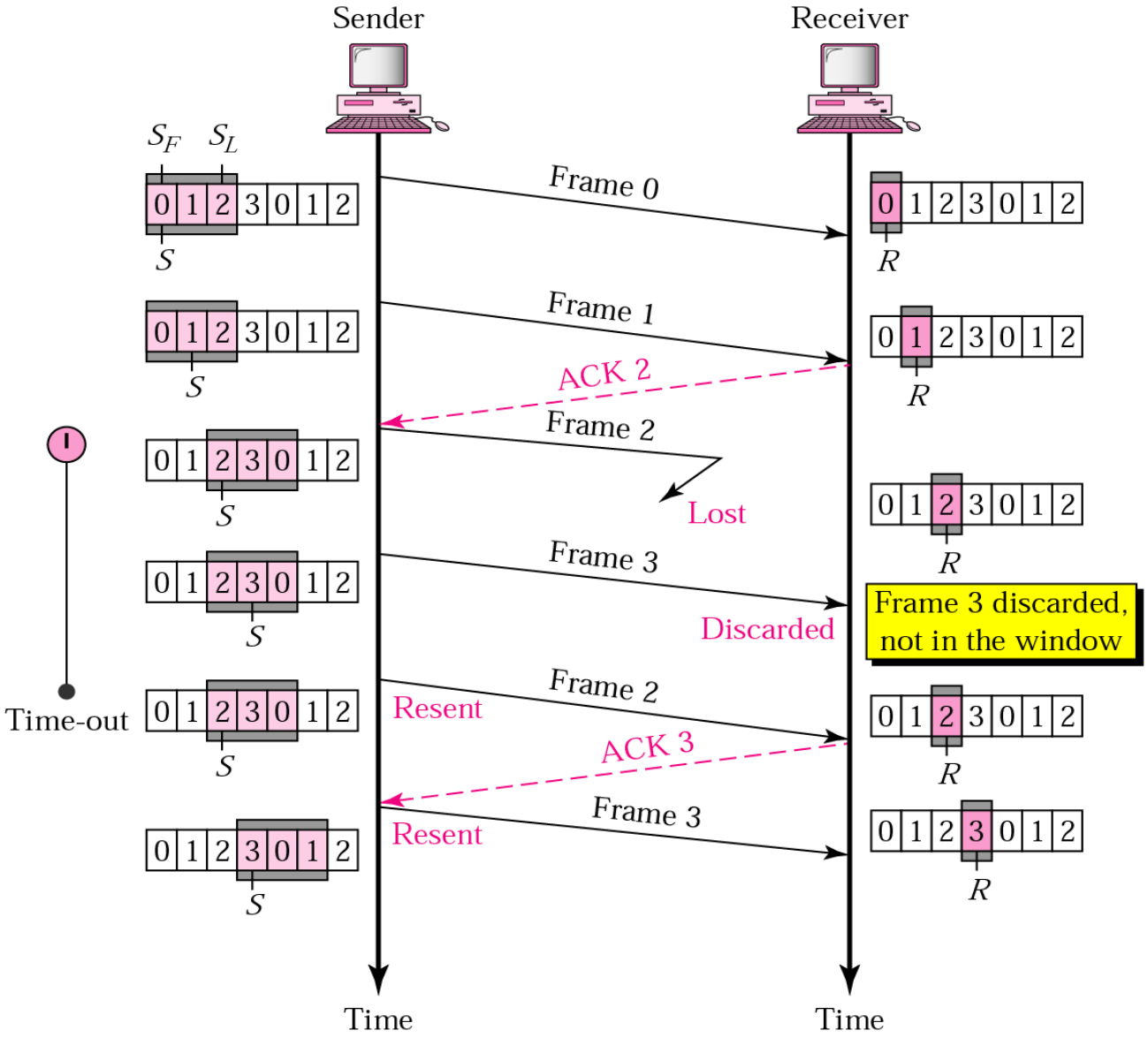


Go-Back-N ARQ



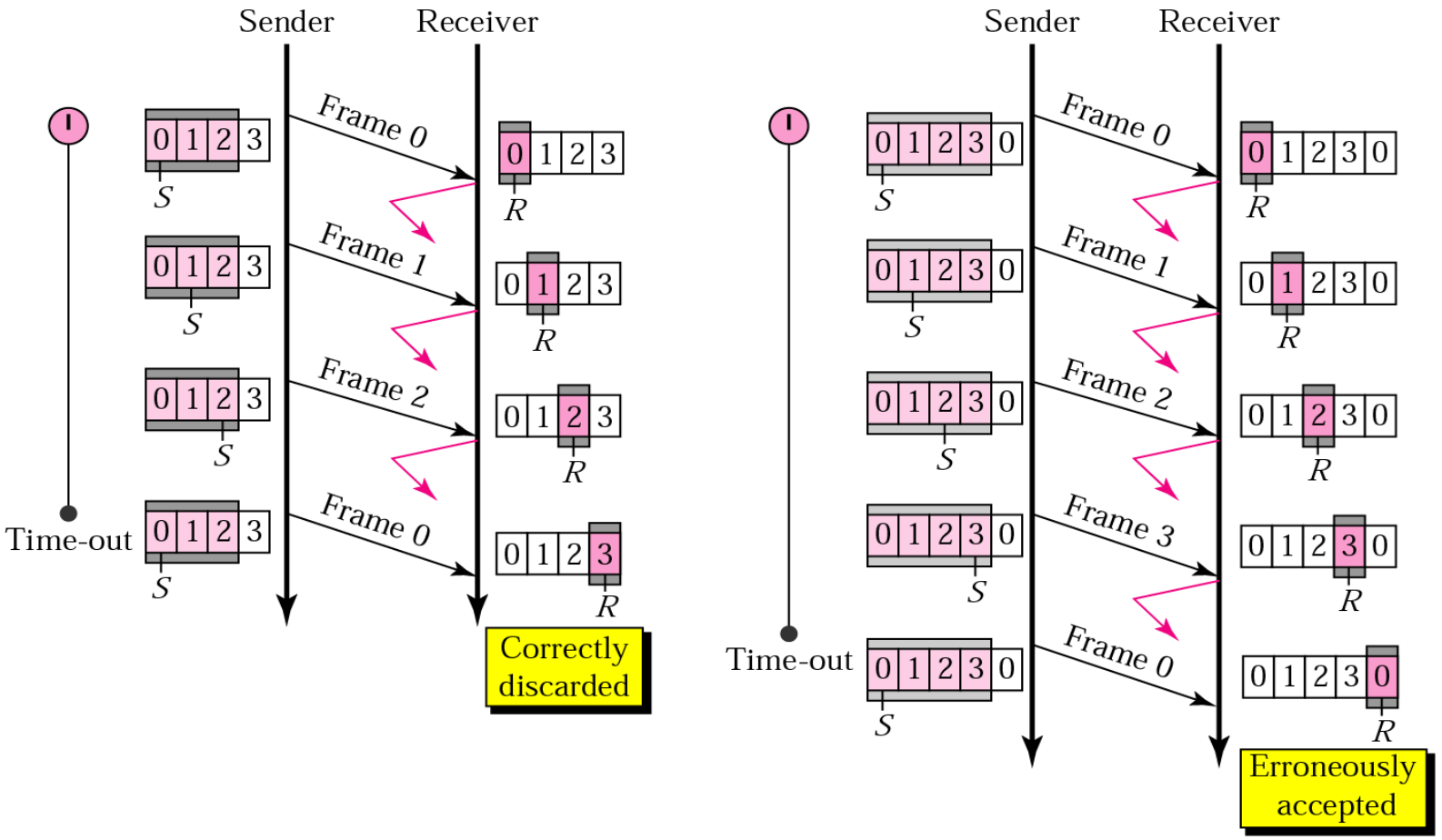
Go-Back-N ARQ

a. Go-Back-N ARQ, lost frame



Go-Back-N ARQ

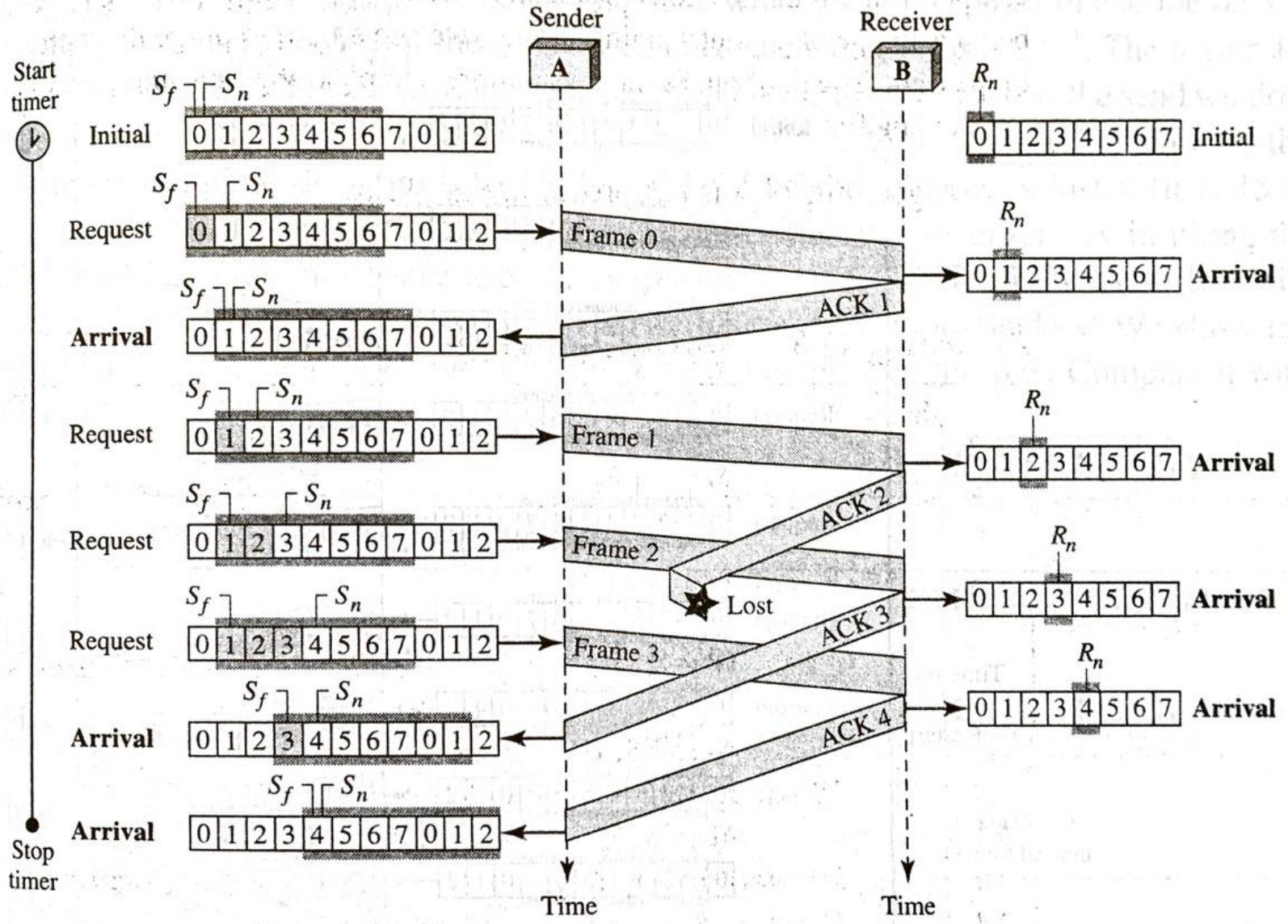
Go-Back-N ARQ: sender window size



a. Window size < 2^m

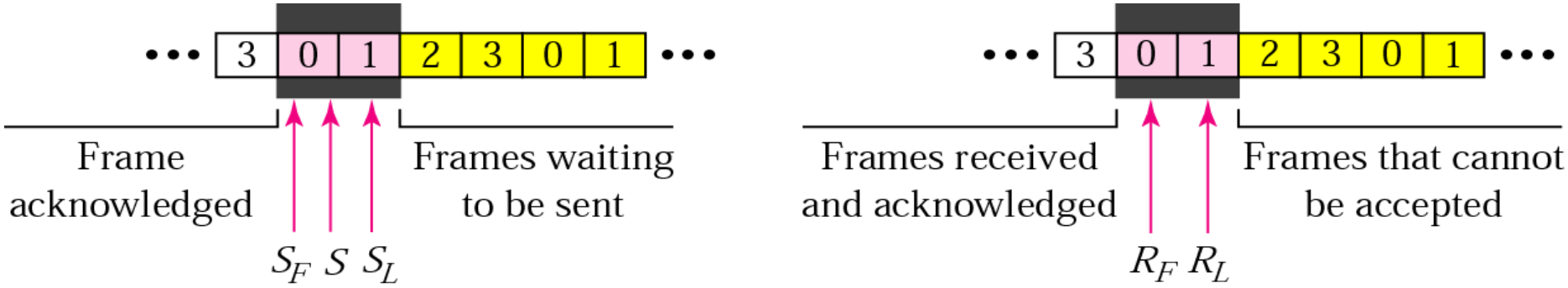
b. Window size = 2^m

Example



Selective Repeat ARQ

Selective Repeat ARQ, sender and receiver windows

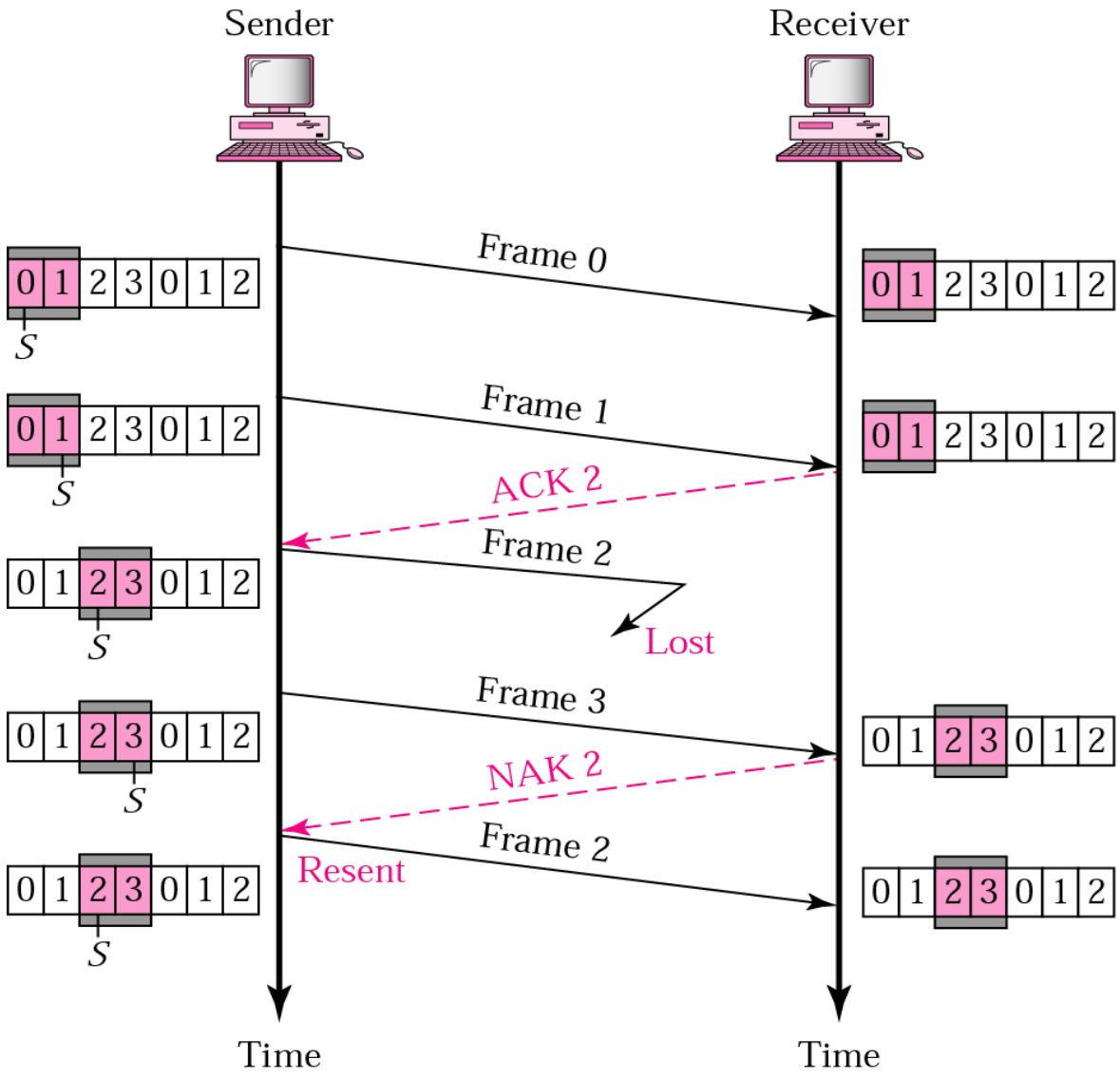


a. Sender window

b. Receiver window

Selective Repeat ARQ

Selective Repeat ARQ, lost frame

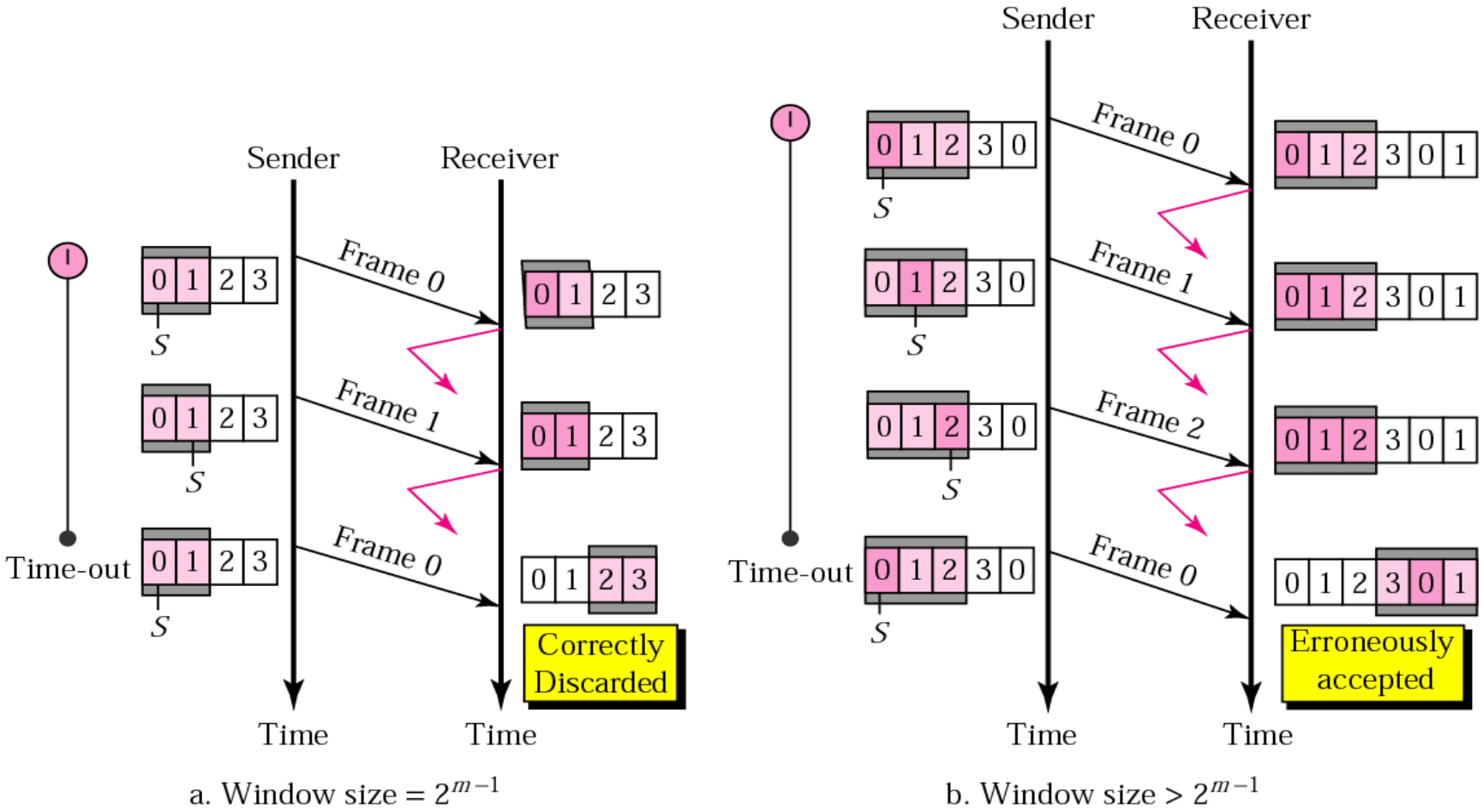


Note:

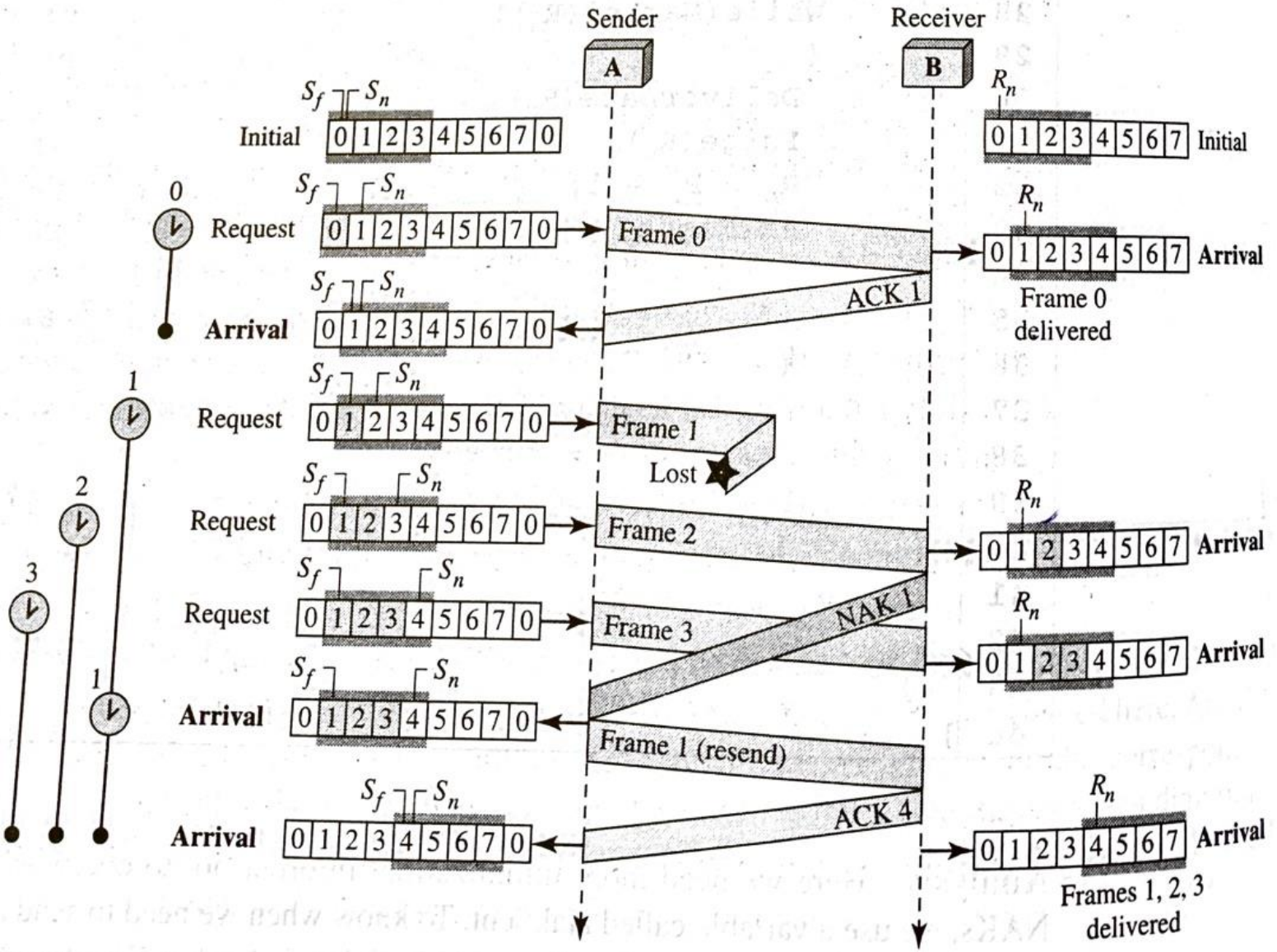
In Selective Repeat ARQ, the size of the sender and receiver window must be at most one-half of 2^m.

Selective Repeat ARQ

Selective Repeat ARQ, sender window size



Example



Comparison

PROTOCOL:-	GO-BACK-N	STOP AND WAIT	SELECTIVE REPEAT
Bandwidth utilization	Medium	Low	High
Maximum sender Size Window	2^m-1	N.A	2^(m-1)
Maximum receiver Size Window	1	N.A	2^(m-1)
Pipelining	Implemented	Not Implemented	Implemented
Out of order Frames	Discarded	Discarded	Accepted
Cumulative ACK	Applicable	N.A	Applicable
NAK	N.A	N.A	Applicable

Comparison…Pipelined protocol

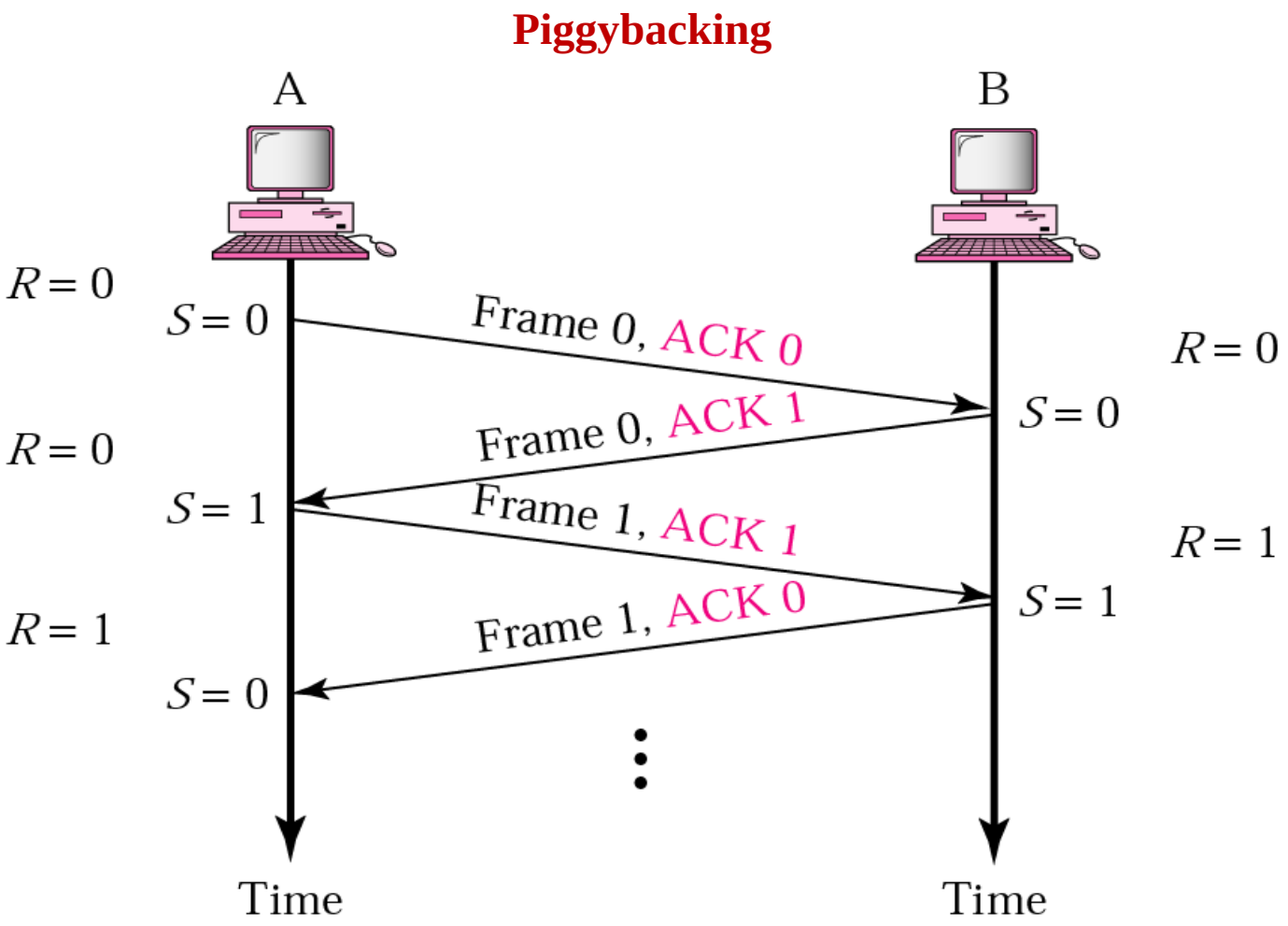
Go-back-N:

- sender can have up to N unacked packets in pipeline
- receiver only sends *cumulative ack*
  - doesn't ack packet if there's a gap
- sender has timer for oldest unacked packet
  - when timer expires, retransmit *all* unacked packets

Selective Repeat:

- sender can have up to N unack'ed packets in pipeline
- rcvr sends *individual ack* for each packet
- sender maintains timer for each unacked packet
  - when timer expires, retransmit only that unacked packet

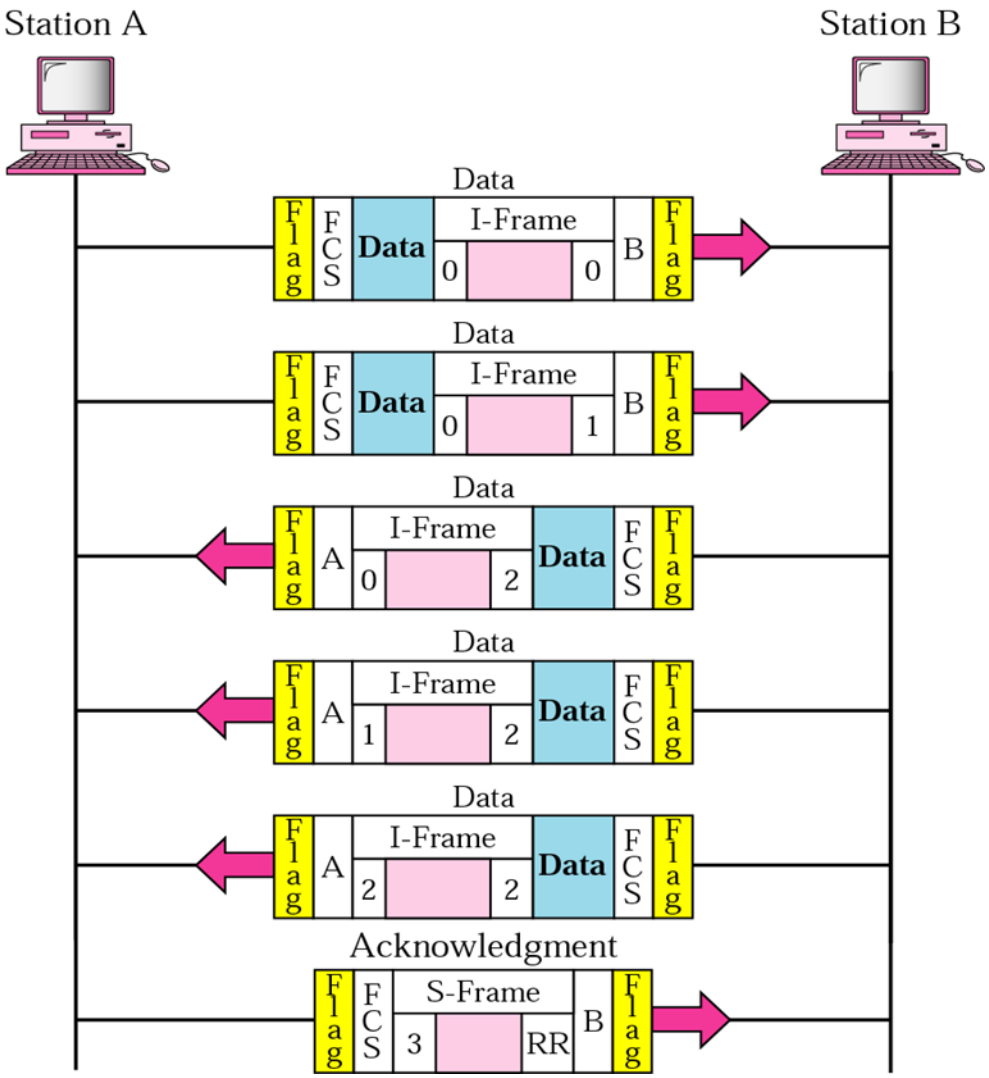
Piggybacking (Bidirectional flow)



Example 1

Figure shows an exchange using piggybacking where is no error.

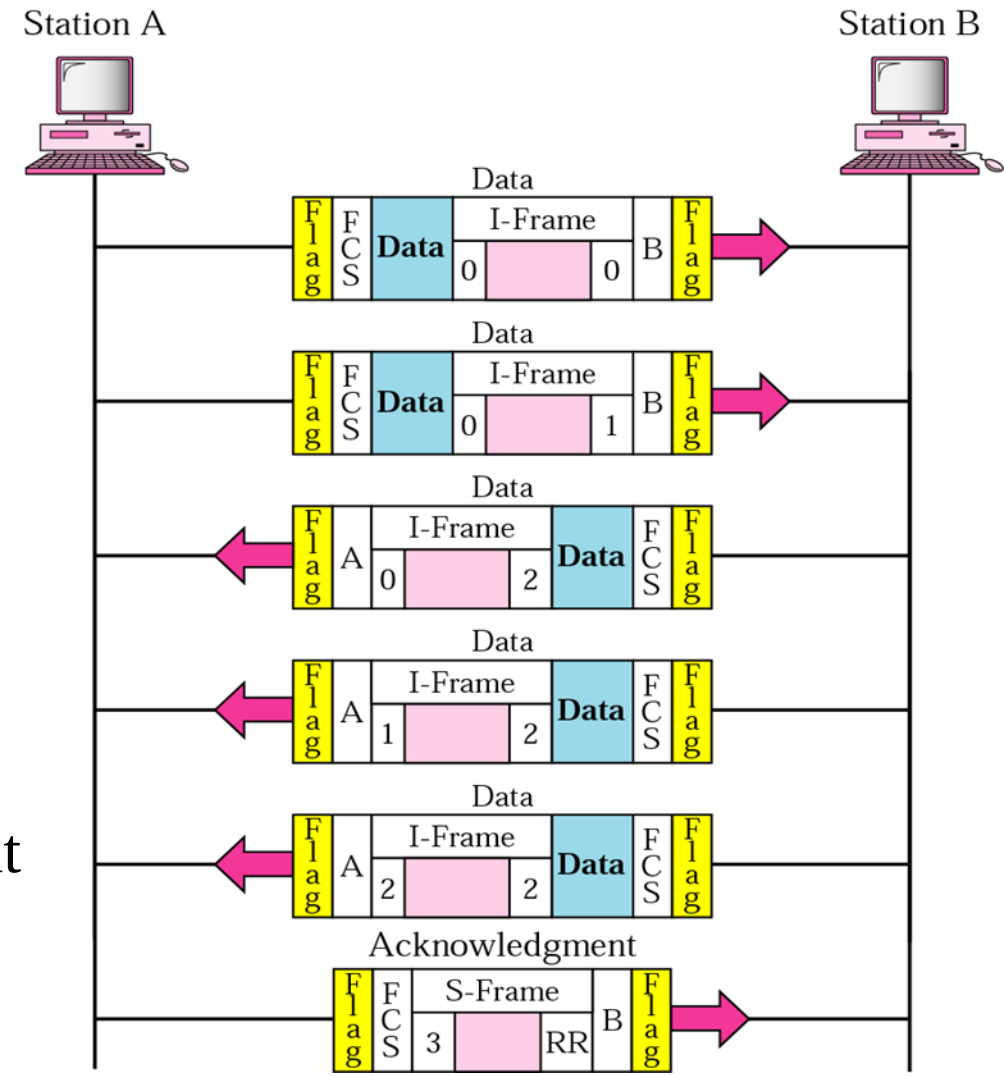
1. Station A begins the exchange of information with an I-frame numbered 0 followed by another I-frame numbered 1.



Example 1

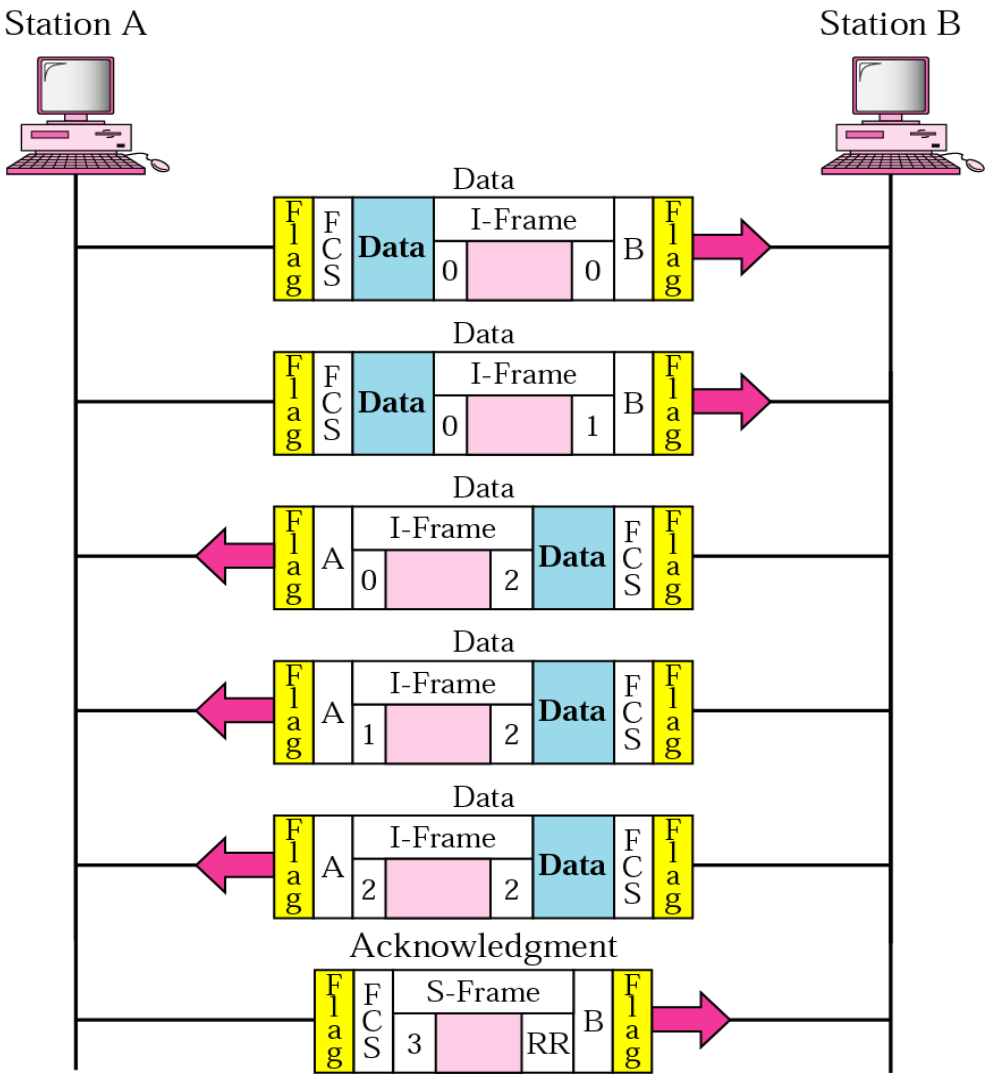
Figure shows an exchange using piggybacking where is no error.

2. Station B piggybacks its acknowledgment of both frames onto an I-frame of its own. Station B's first I-frame is also numbered 0 [N(S) field] and contains a 2 in its N(R) field, acknowledging the receipt of A's frames 1 and 0 and indicating that it expects frame 2 to arrive next.



Example 1

3. Station B transmits its second and third I-frames (numbered 1 and 2) before accepting further frames from station A. Its N(R) information, therefore, has not changed: B frames 1 and 2 indicate that station B is still expecting A frame 2 to arrive next.



Example 2

In Example 1, suppose frame 1 sent from station B to station A has an error. Station A informs station B to resend frames 1 and 2 (the system is using the Go-Back-N mechanism). Station A sends a reject supervisory frame to announce the error in frame 1. Figure shows the exchange.