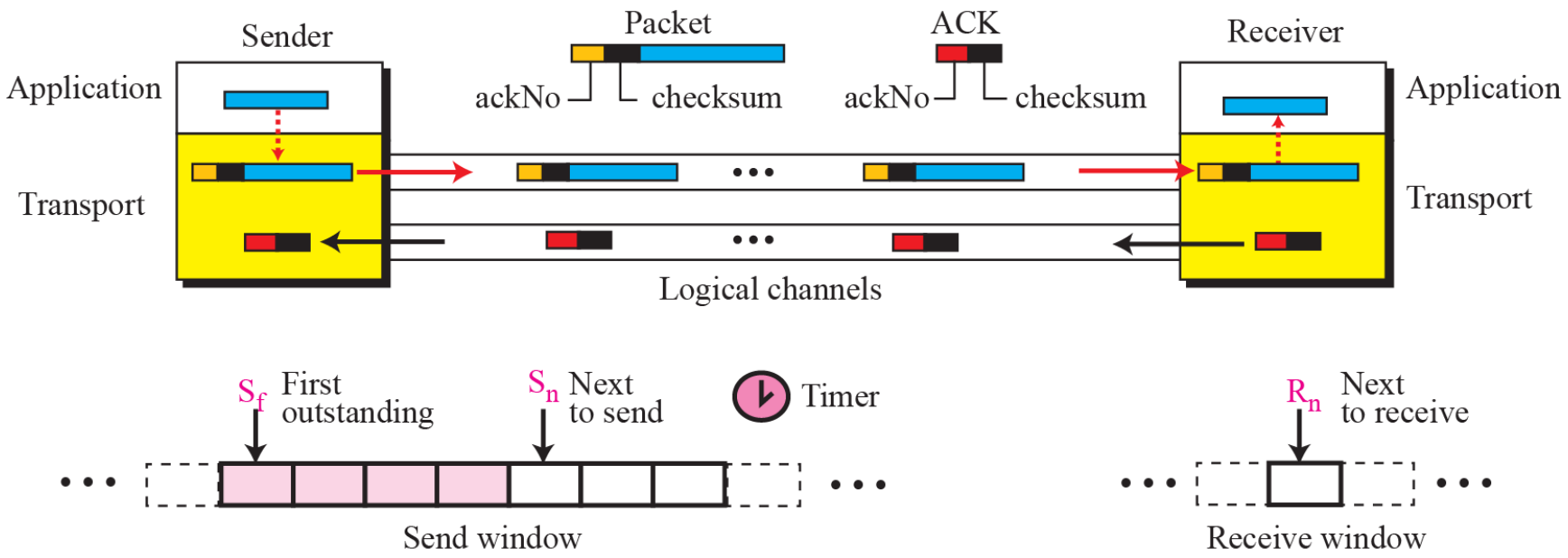
Transport Layer

Anand Baswade

anand@iitbhilai.ac.in

Go-Back-N protocol



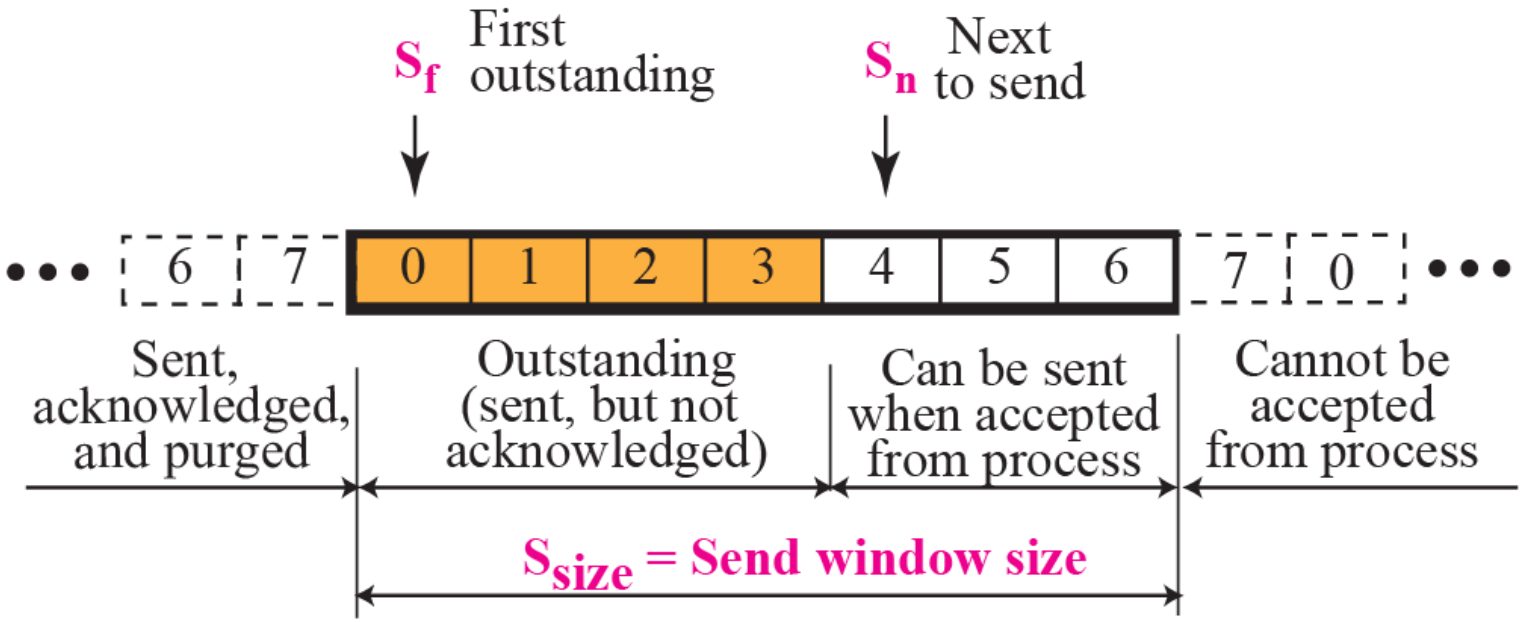
When the sender doesn't receive ACK, it retransmits the packet in error plus all the succeeding packets. Hence, the name of the protocol is go-back-N ARQ.

Go-Back-N protocol Cont..

- 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.

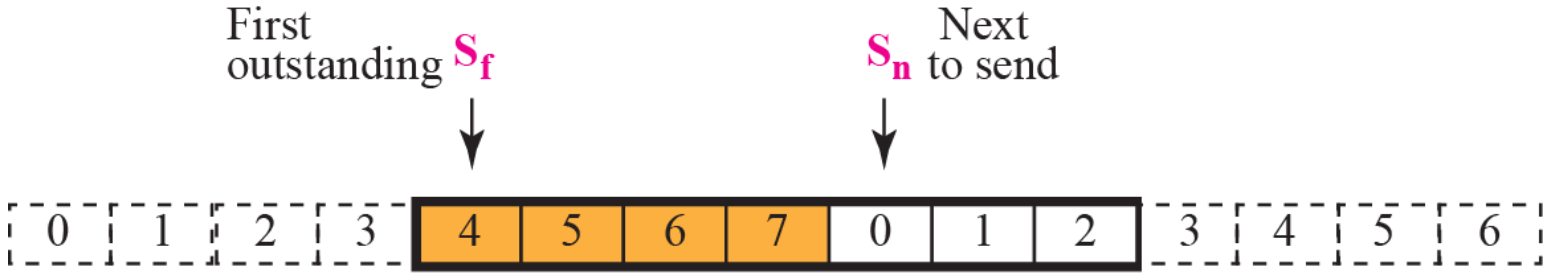
- In the Go-Back-N protocol, the acknowledgment number is cumulative and defines the sequence number of the next packet expected to arrive.

Send window for Go-Back-N

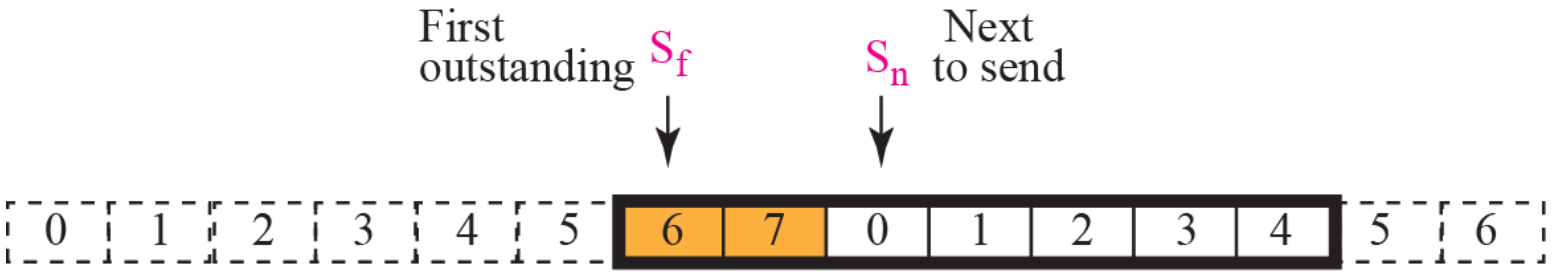


- *The send window is an abstract concept defining an imaginary box of maximum size = $2^m - 1$ with three variables: S_f, S_n, and S_{size}.*
- *The send window can slide one or more slots when an error-free ACK with ack No. between S_f and S_n (in modular arithmetic) arrives.*

Sliding the send window

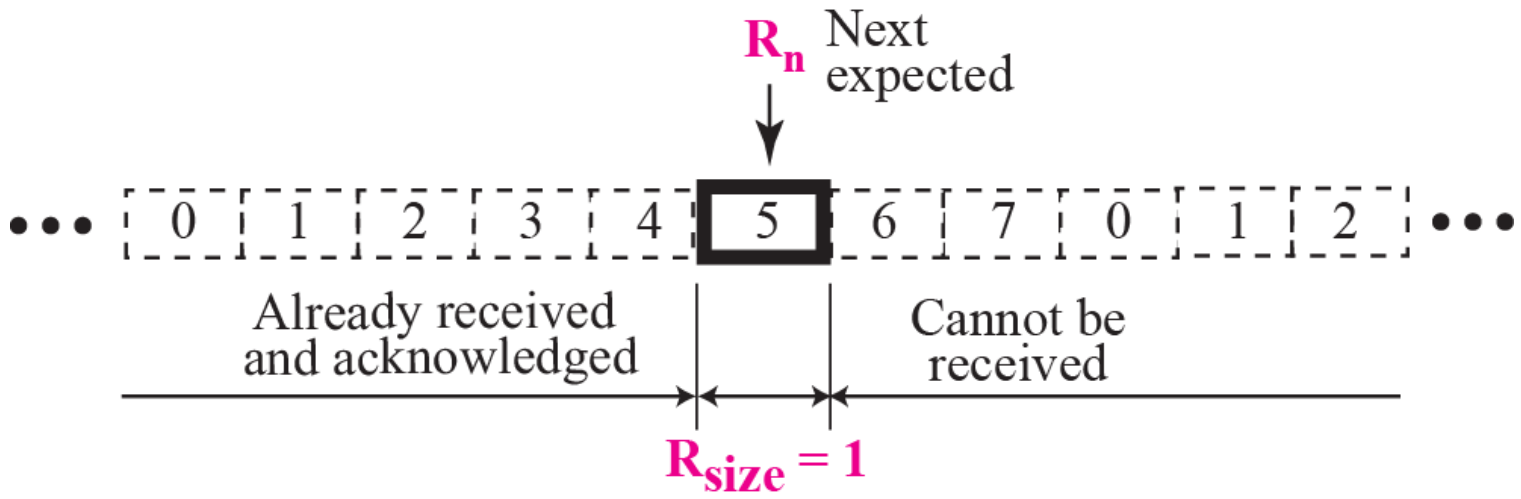


a. Window before sliding



b. Window after sliding (an ACK with ackNo = 6 has arrived)

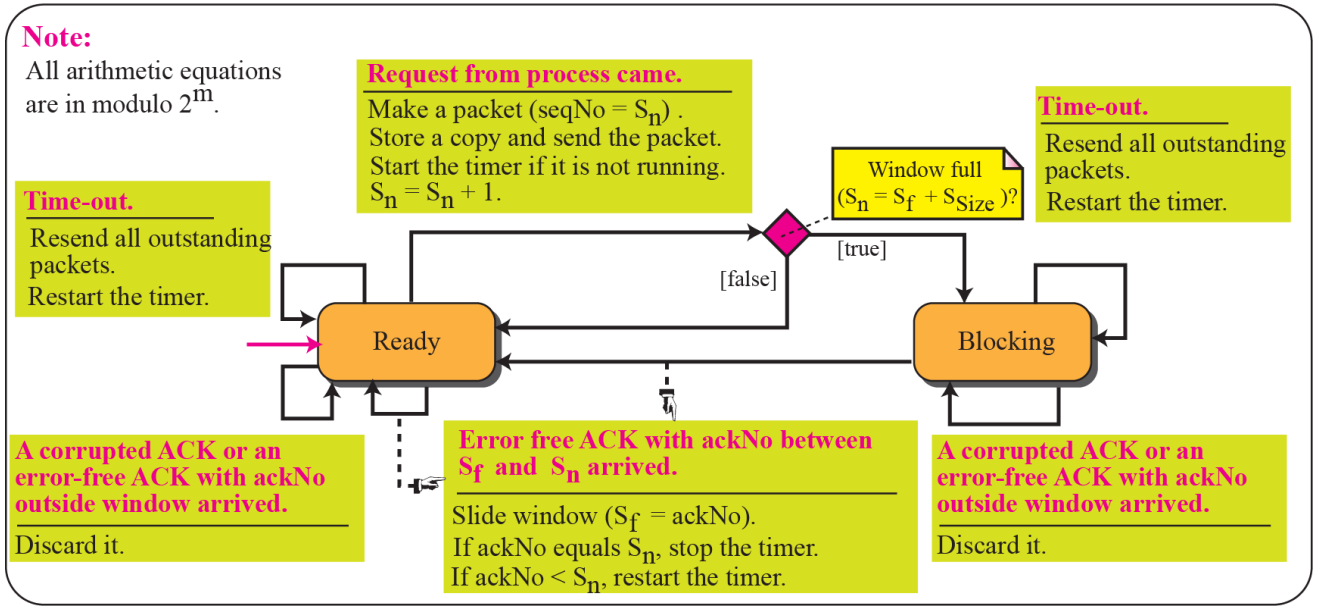
Receive window for Go-Back-N



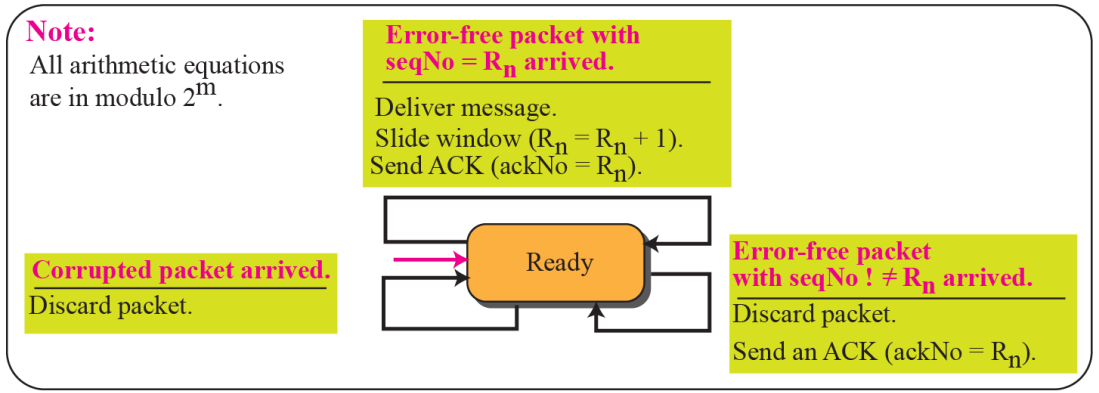
- The receive window is an abstract concept defining an imaginary box of size 1 with one single variable R_n.
- The window slides when a correct packet has arrived; sliding occurs one slot at a time.
- *In the Go-Back-N protocol, the size of the send window must be less than 2^m; the size of the receive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

FSMs for Go-Back-N

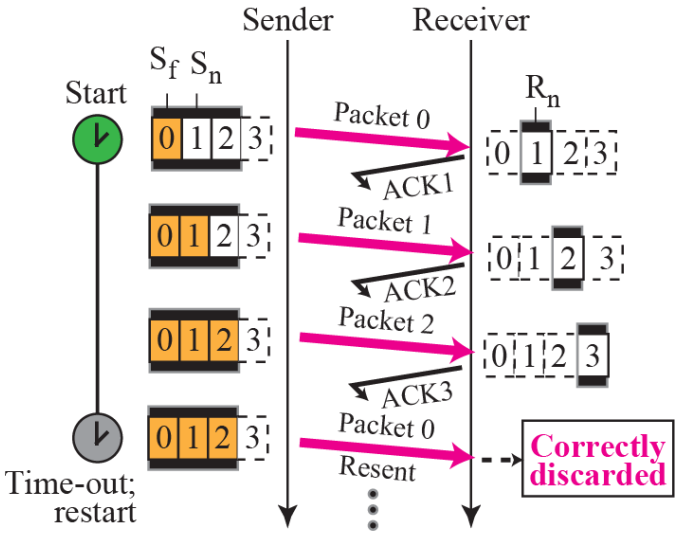
Sender



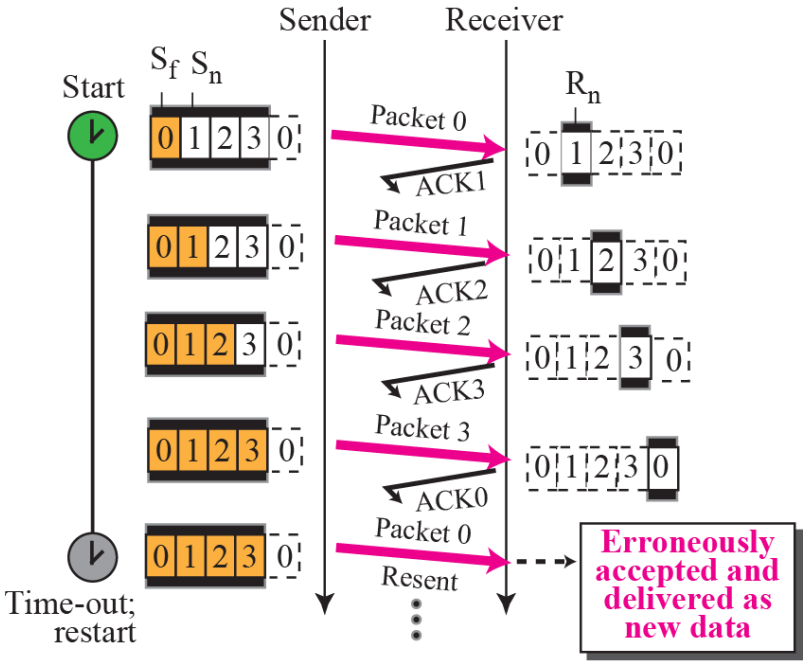
Receiver



Send window size for Go-Back-N

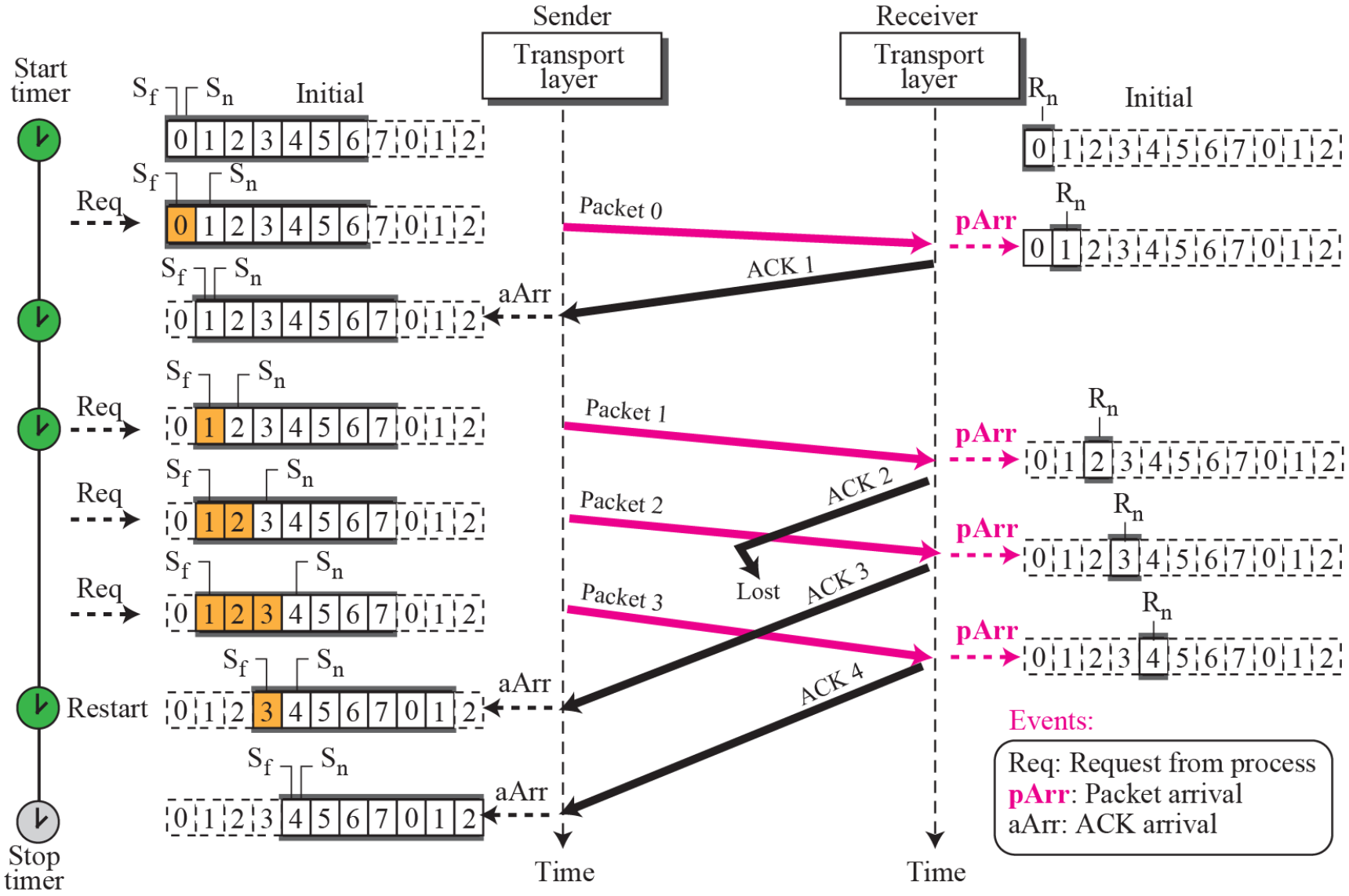


a. Send window of size < 2^m



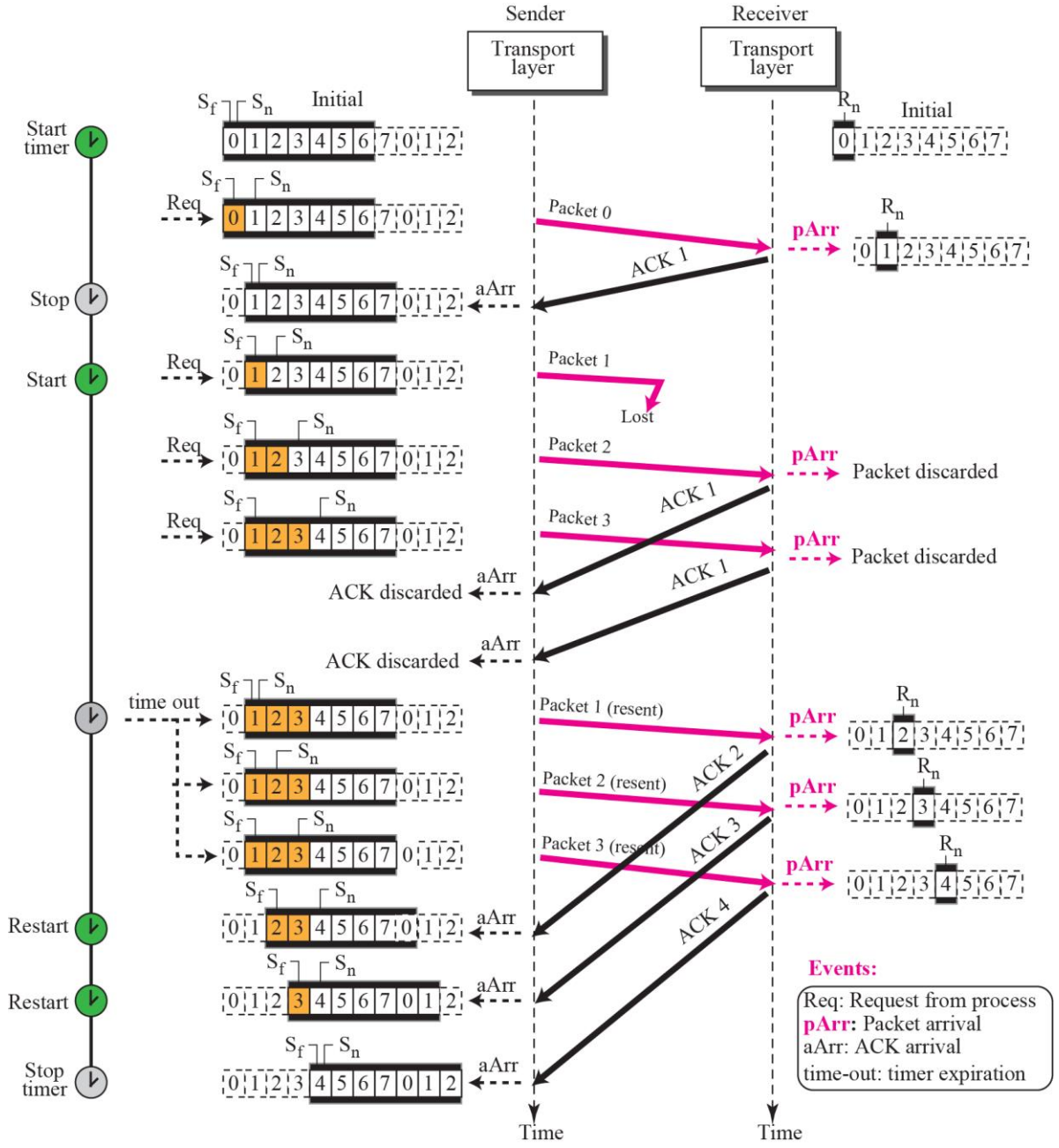
b. Send window of size = 2^m

Example

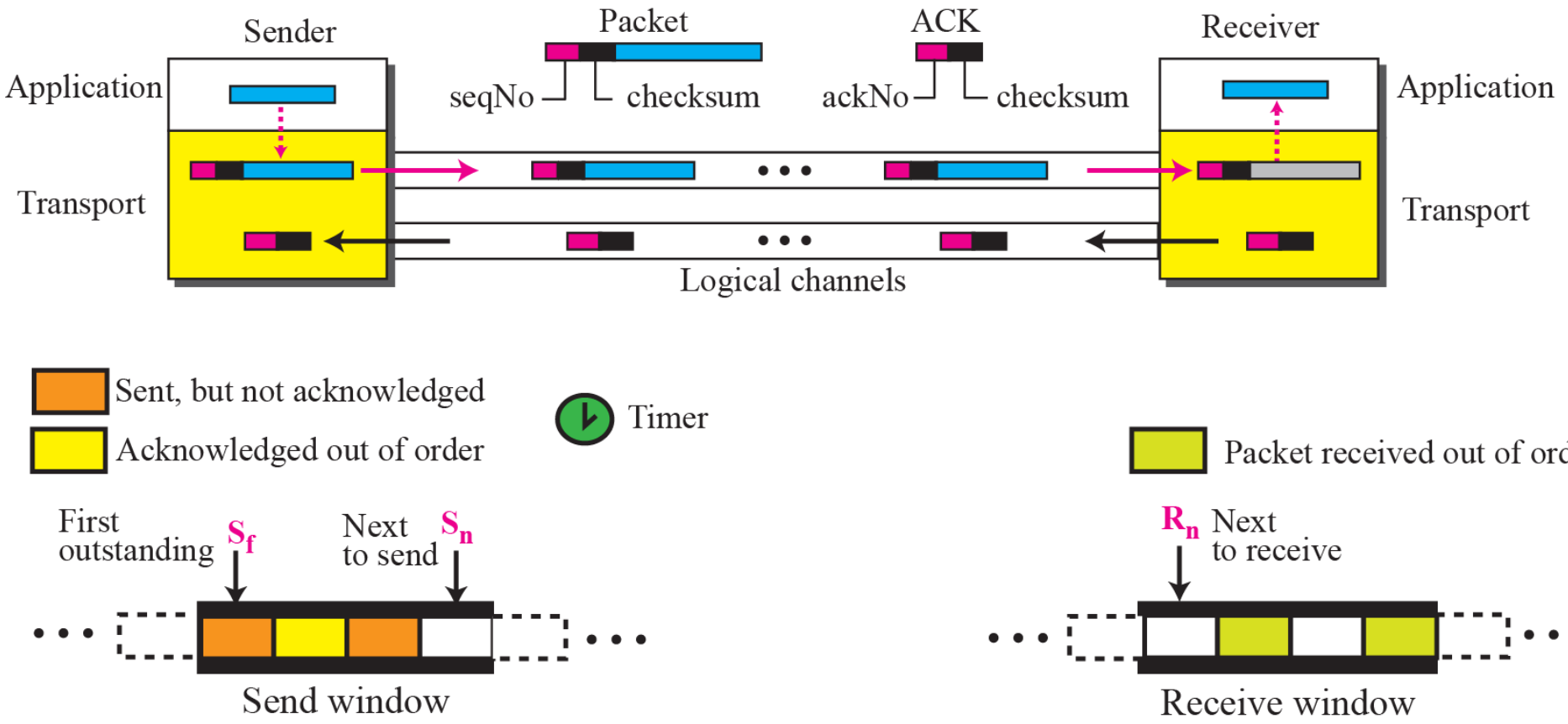


Cumulative acknowledgments can help

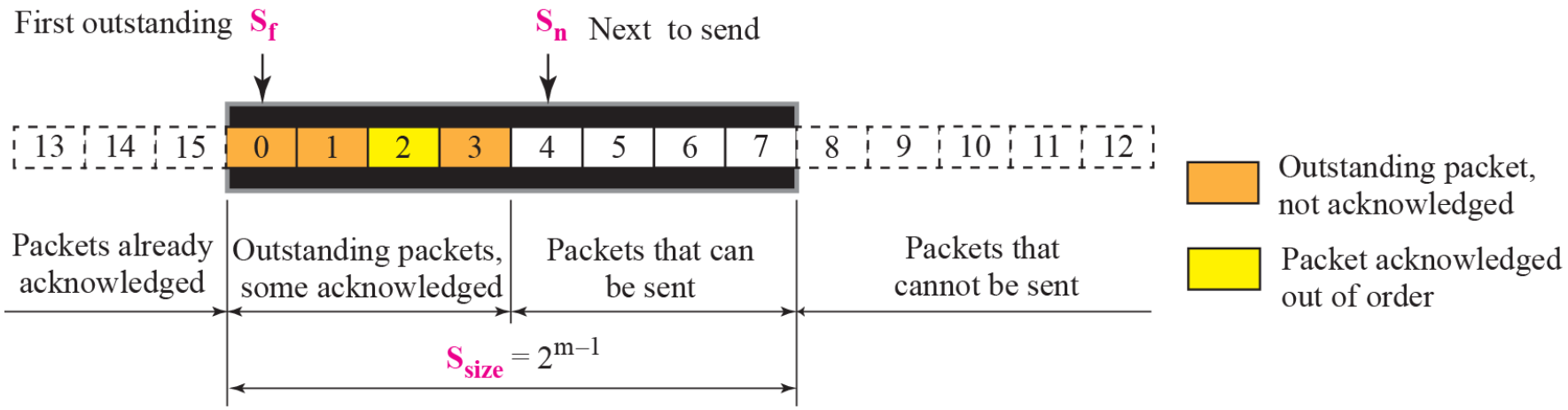
Example 2


Sender
Transport layer
Receiver
Transport layer
Initial
Start timer
Req
Packet 0
pArr
ACK 1
Stop
aArr
Start
Packet 1
Lost
Packet 2
Packet discarded
Packet 3
ACK discarded
time out
Packet 1 (resent)
Packet 2 (resent)
Packet 3 (resent)
ACK 2
ACK 3
ACK 4
Restart
Restart
Stop timer
Events:
Req: Request from process
pArr: Packet arrival
aArr: ACK arrival
time-out: timer expiration
Time
Time

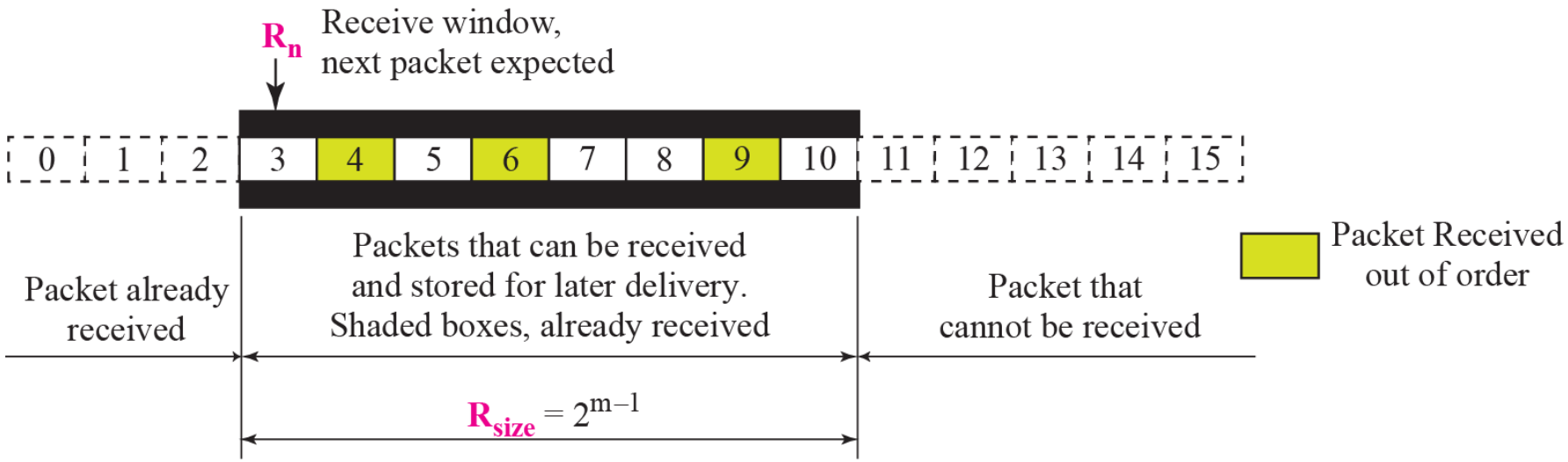
Outline of Selective-Repeat



Send window for Selective-Repeat protocol



Receive window for Selective-Repeat protocol



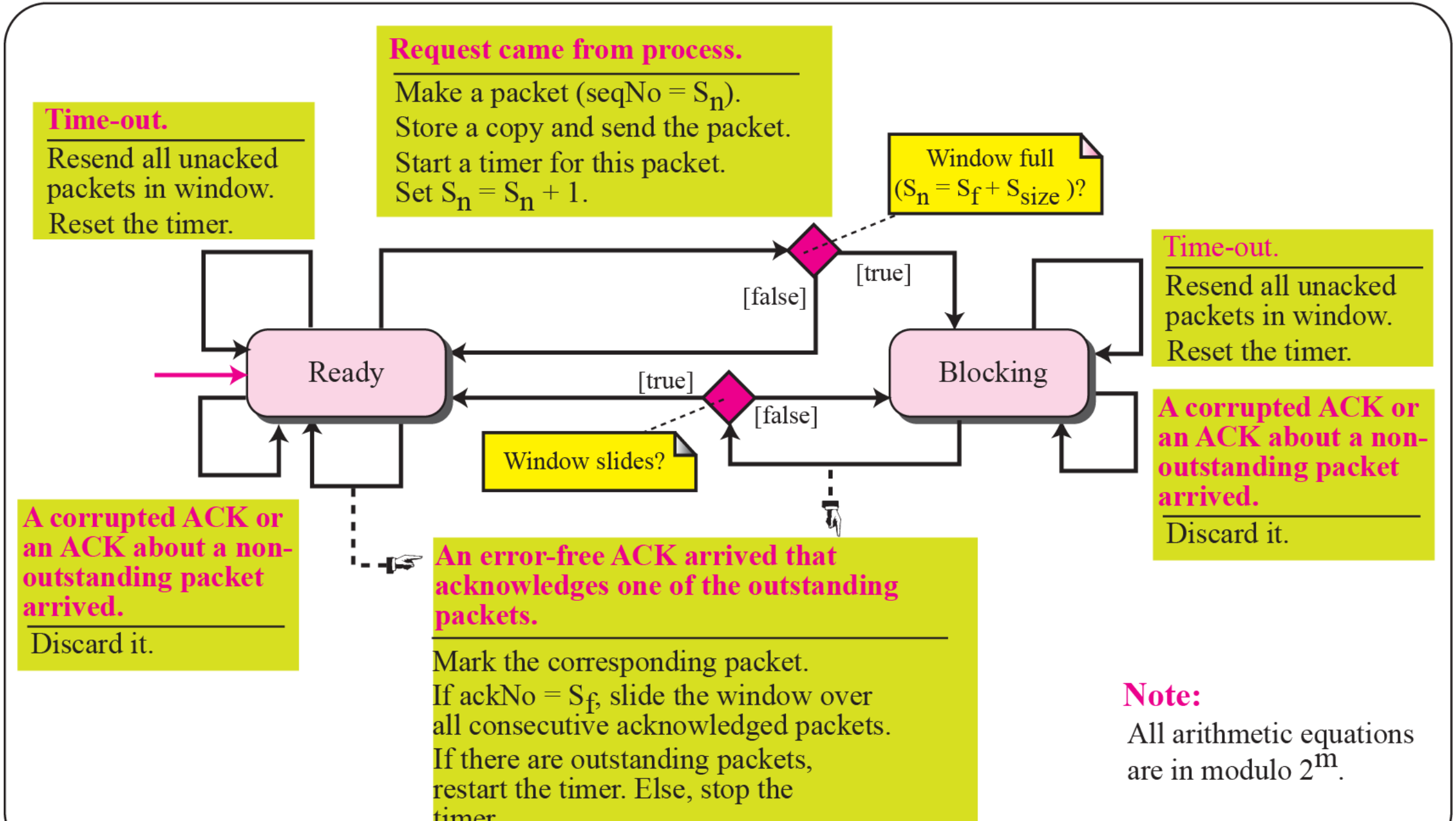
Example

Assume a sender sends 6 packets: packets 0, 1, 2, 3, 4, and 5. The sender receives an ACK with ackNo = 3. What is the interpretation if the system is using GBN or SR?

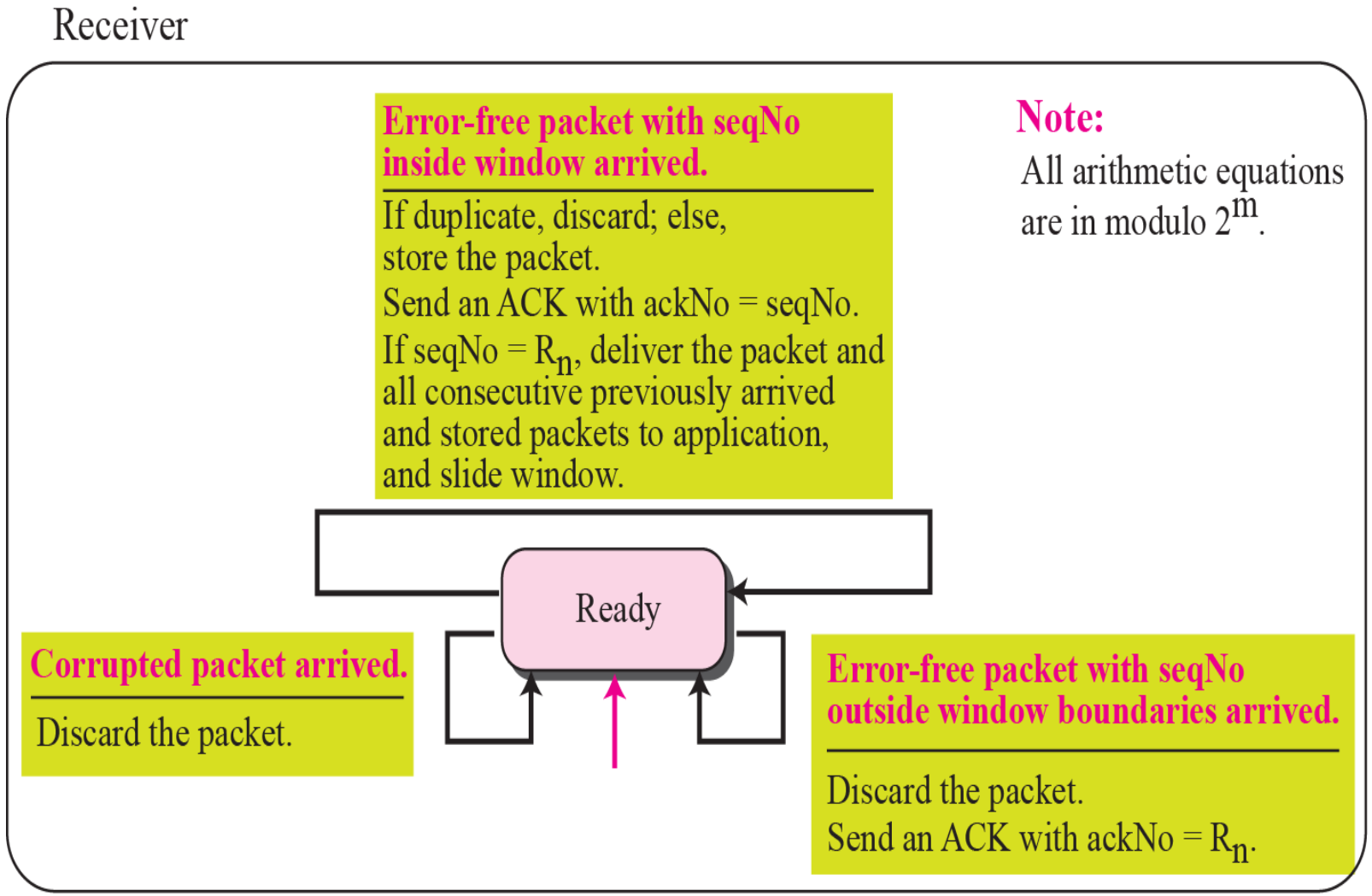
Solution

If the system is using GBN, it means that packets 0, 1, and 2 have been received uncorrupted and the receiver is expecting packet 3. If the system is using SR, it means that packet 3 has been received uncorrupted; the ACK does not say anything about other packets.

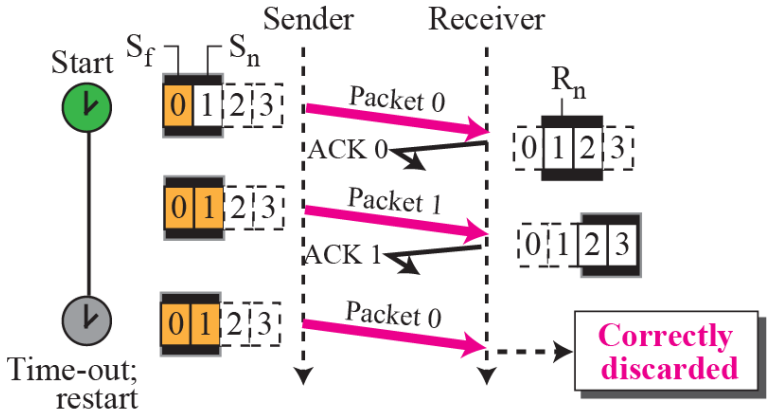
Sender


Request came from process.
Make a packet (seqNo = S_n).
Store a copy and send the packet.
Start a timer for this packet.
Set $S_n = S_n + 1$.
Window full
($S_n = S_f + S_{size}$)?
[true]
[false]
Time-out.
Resend all unacked packets in window.
Reset the timer.
Ready
Blocking
[true]
[false]
Window slides?
A corrupted ACK or an ACK about a non-outstanding packet arrived.
Discard it.
An error-free ACK arrived that acknowledges one of the outstanding packets.
Mark the corresponding packet.
If ackNo = S_f, slide the window over all consecutive acknowledged packets.
If there are outstanding packets, restart the timer. Else, stop the timer.
Time-out.
Resend all unacked packets in window.
Reset the timer.
A corrupted ACK or an ACK about a non-outstanding packet arrived.
Discard it.
Note:
All arithmetic equations are in modulo 2^m.

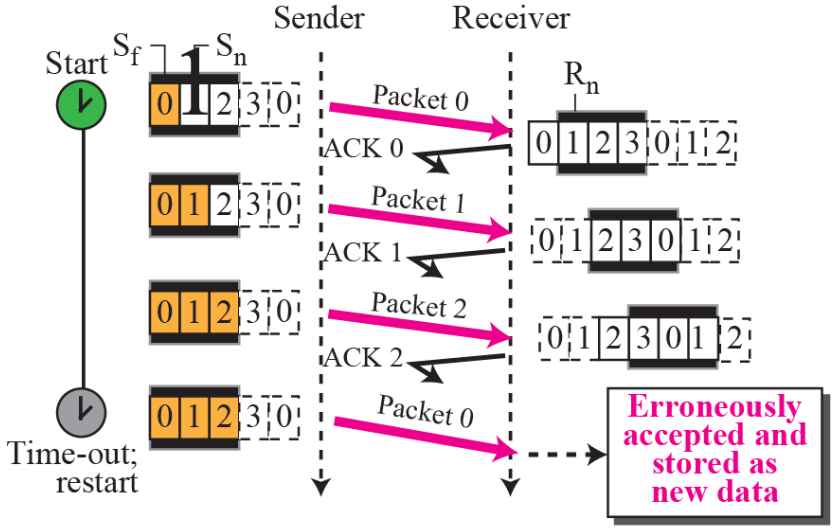
FSMs for SR protocol



Selective-Repeat window size

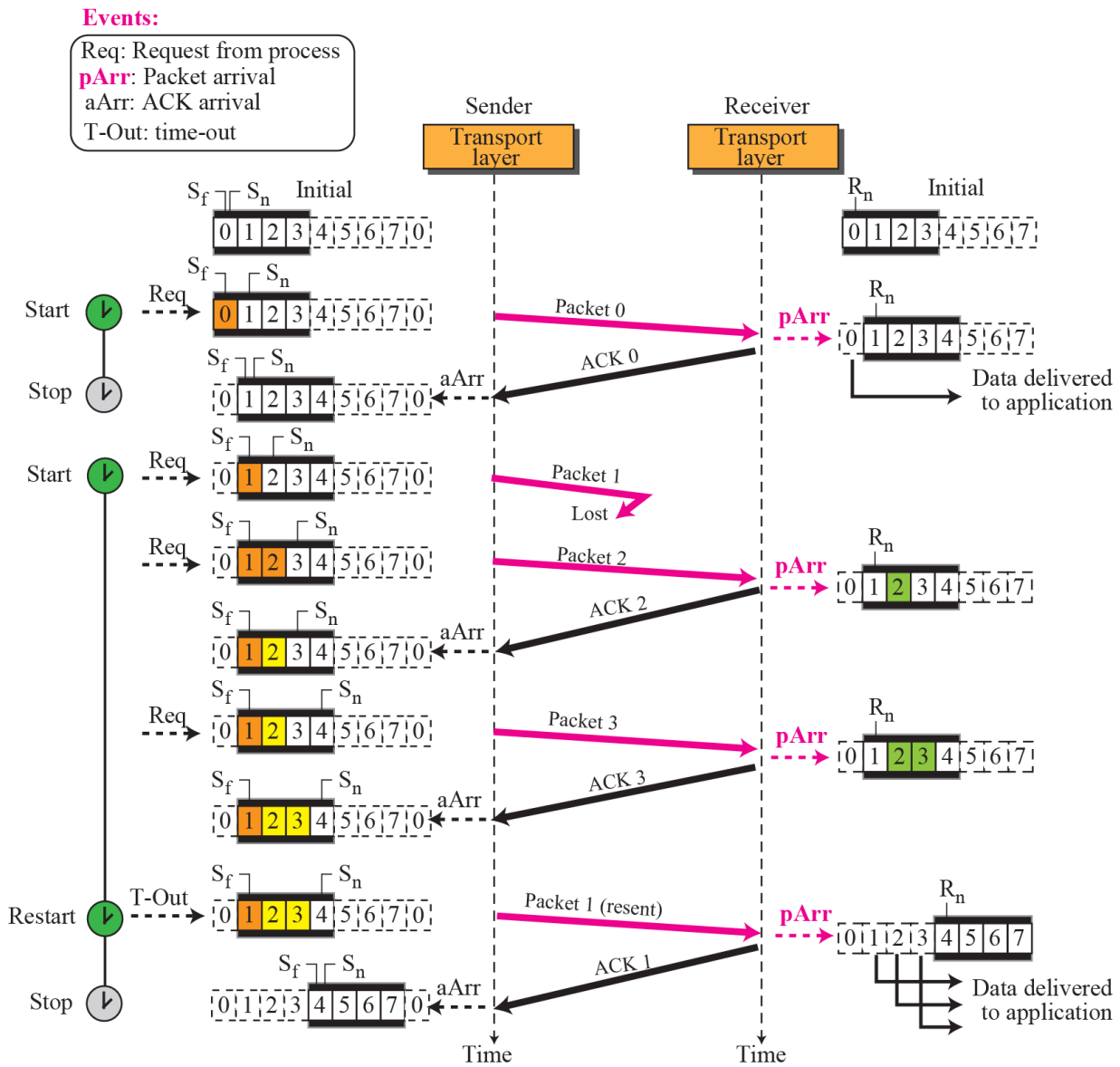


a. Send and receive windows of size = 2^{m-1}



b. Send and receive windows of size > 2^{m-1}

Example


Events:
Req: Request from process
pArr: Packet arrival
aArr: ACK arrival
T-Out: time-out
Sender
Transport layer
Receiver
Transport layer
Initial
Start
Stop
Req
Packet 0
ACK 0
pArr
Data delivered to application
Packet 1
Lost
Packet 2
ACK 2
Packet 3
ACK 3
T-Out
Restart
Packet 1 (resent)
ACK 1
Time
Time

Selective-Repeat

This is the most efficient among the ARQ schemes, but the sender must be more complex so that it can send out-of-order frames. The receiver also must have storage space to store the post-NAK frames and processing power to reinsert frames in proper sequence.