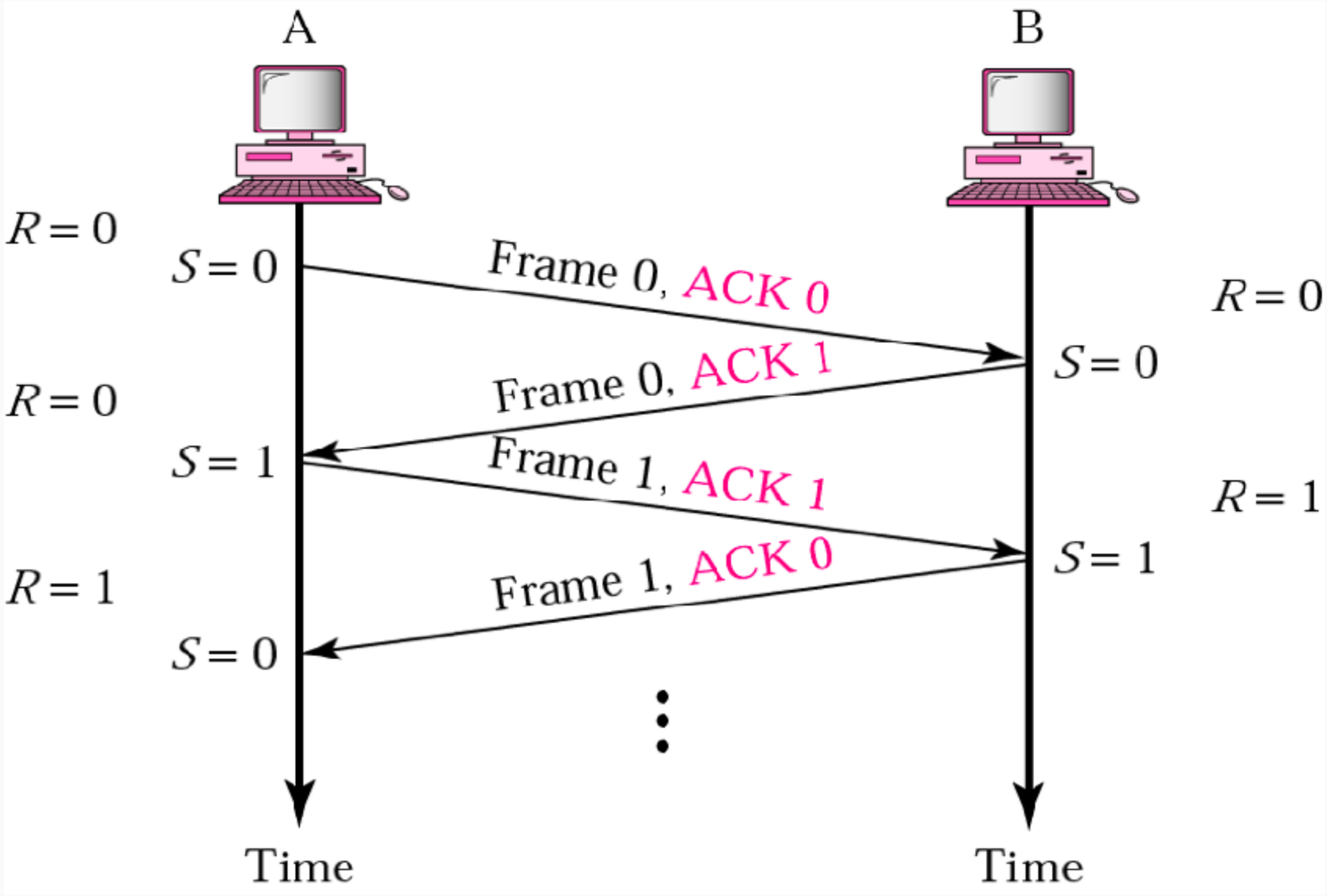
Piggybacking = defer sending your ack, and we can club data with ack



Bi-directional data transfer

Go Back-N ARQ

-Stop-and-wait => low link utilization

To inc utilization, solution - PIPELINING

-To improve the efficiency, multiple frames must be in transmission before waiting for an acknowledgement to come.
thus, we need multiple sequence numbers .

- If a frame is lost, the lost frame and all of the following frames must be retransmitted, that's why this protocol is known as Go-Back-N.

Sequence numbers

m bit field in frame header
=>2^m valid seq numbers

range of seq no. = 0 to 2^m-1

for eg, if m=3
then, range of seq no. = 0 to 7
i.e., 0 1 2 3 4 5 6 7 0 1 2 3 4 5 6 7 0 1 2 3 4 5 6 7 0 .... and so on

In Go-Back-N, the seq no.'s are modulo-2^m where, m is the size of the seq no. filed in the frame header.

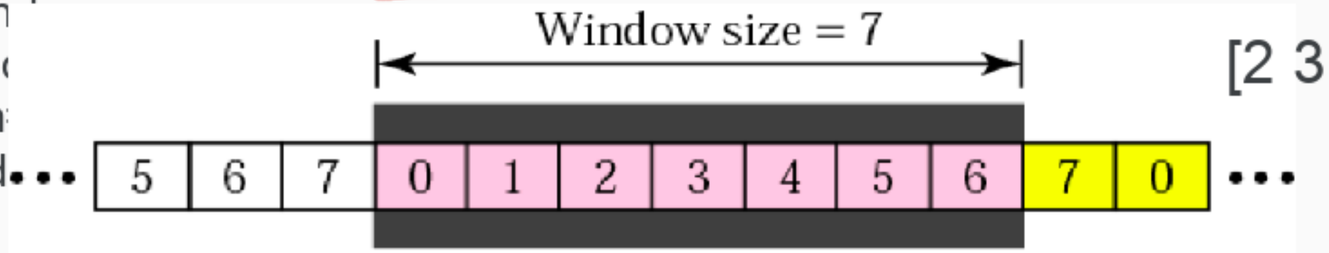
Sliding Window

-It is an abstract concept that defines the range of seq no. that is the concern of sender and reciever.

[0 1 2 3 - 6]

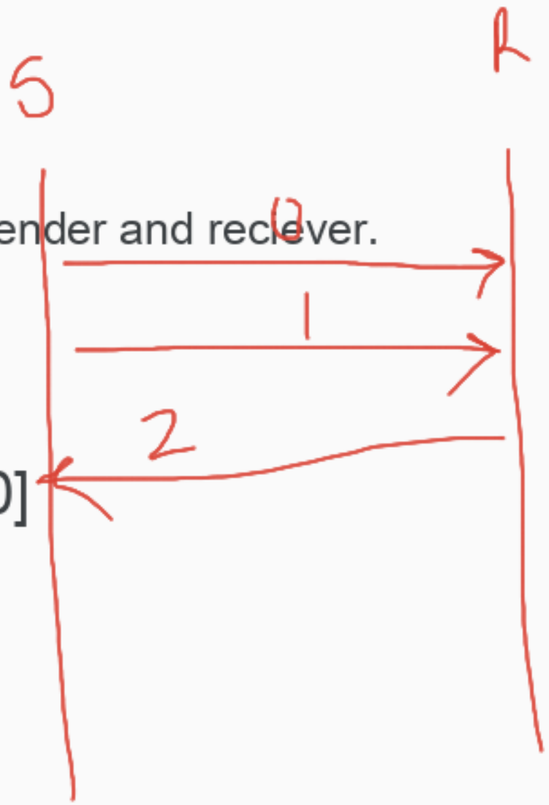
-2 sliding windows: 1 corres to sender and other corres to receiver

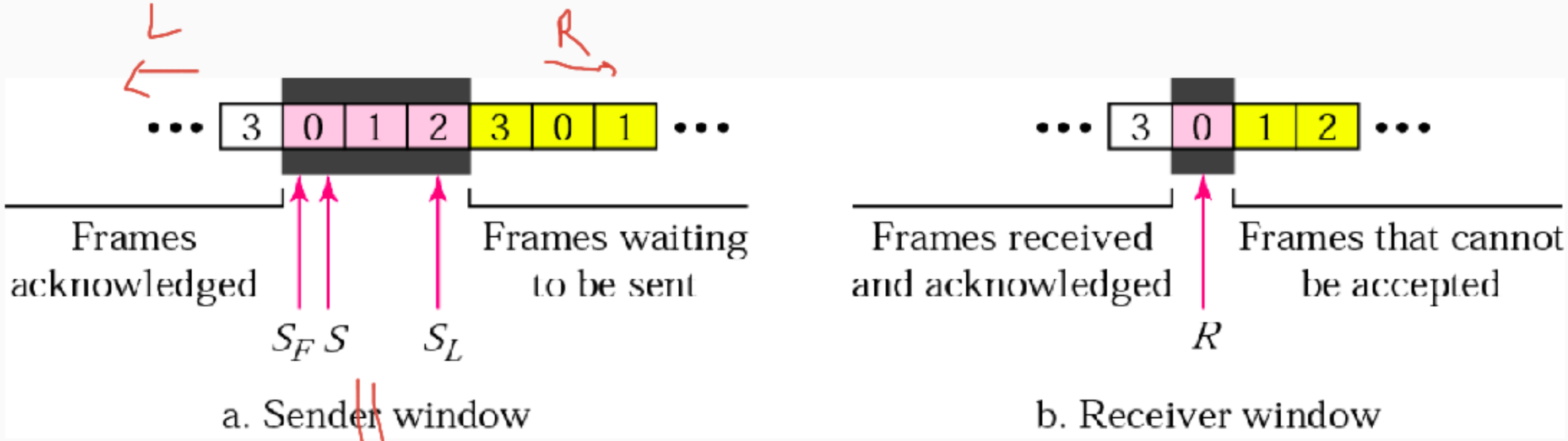
Sen
win
if m
and



[2 3 4 5 6 7 0]

Sender initial window [0 1 2 3 4 5 6] 7 0 1 2 3 4 5 6 7 0 1 2 3 4 5 6 7





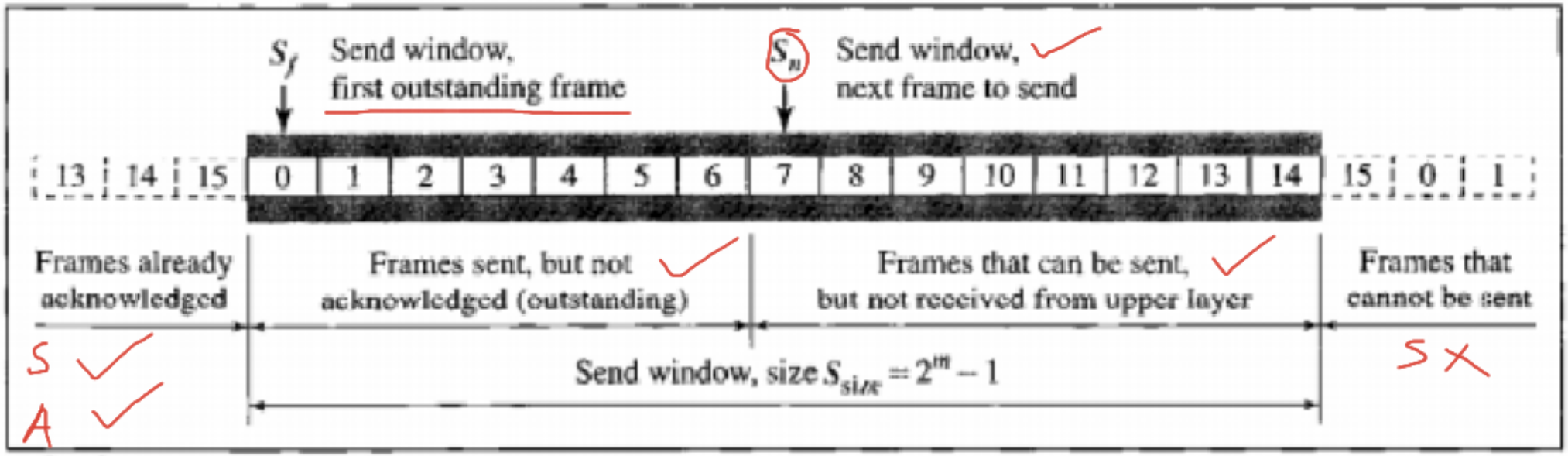
[0 1 2 3 4 5 6]

S_n

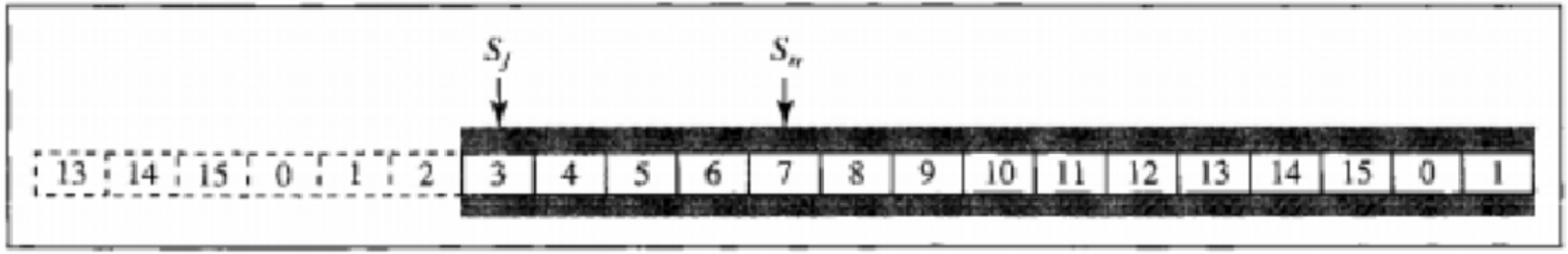
0

1

Figure 11.12 *Send window for Go-Back-N ARQ*

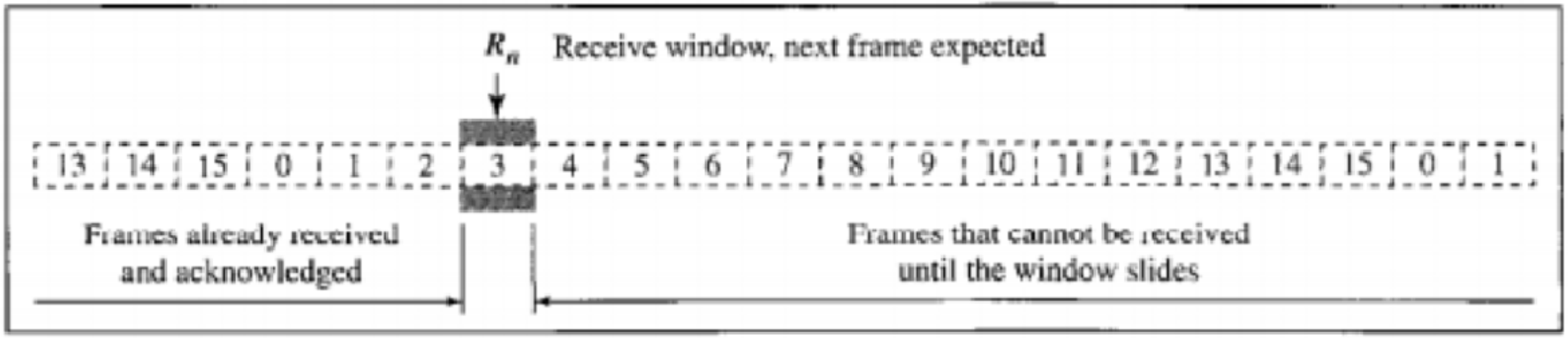


a. Send window before sliding

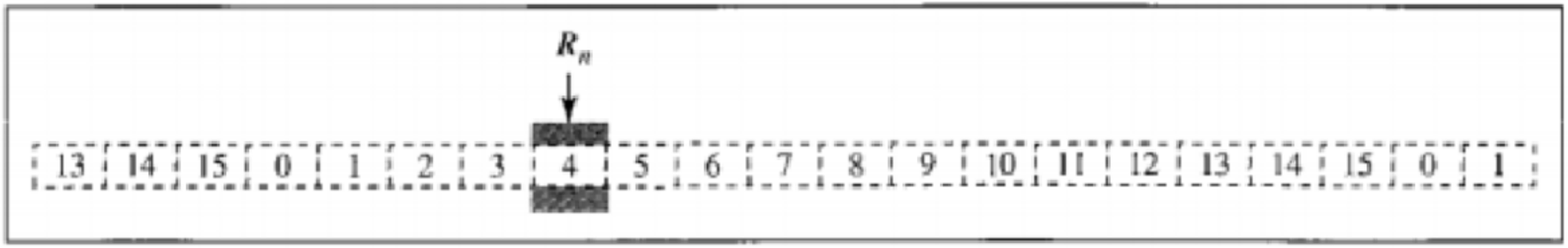


b. Send window after sliding

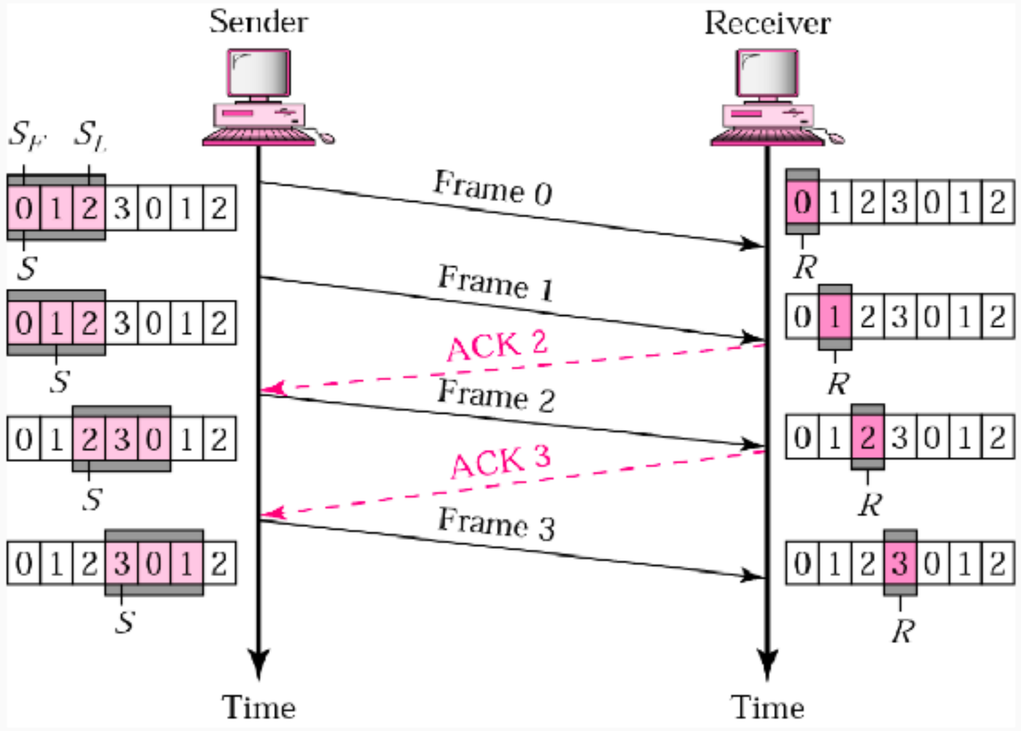
Figure 11.13 *Receive window for Go-Back-N ARQ*



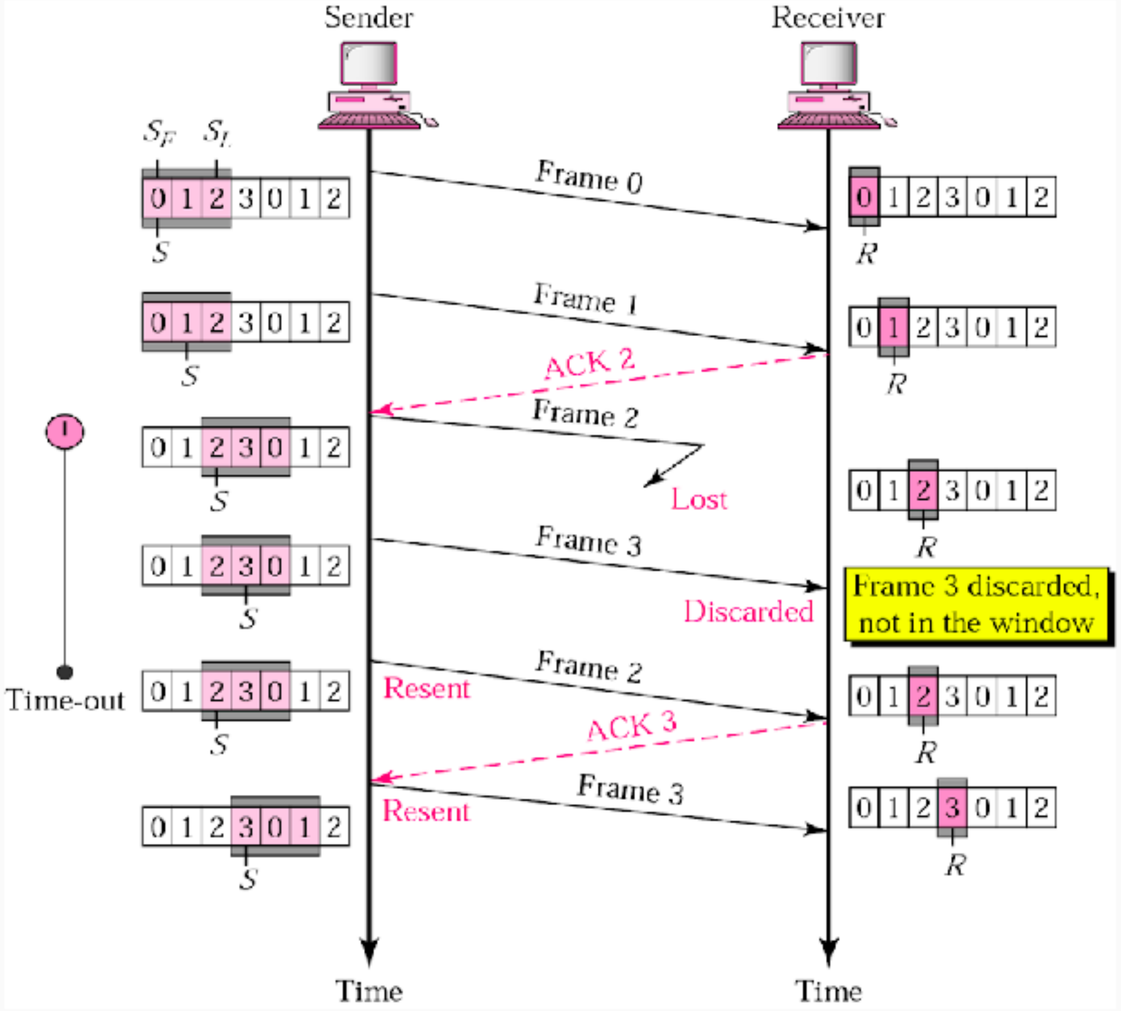
a. Receive window



b. Window after sliding



Normal Operation



Lost Frames

Figure 11.16 *Flow diagram for Example 11.6*

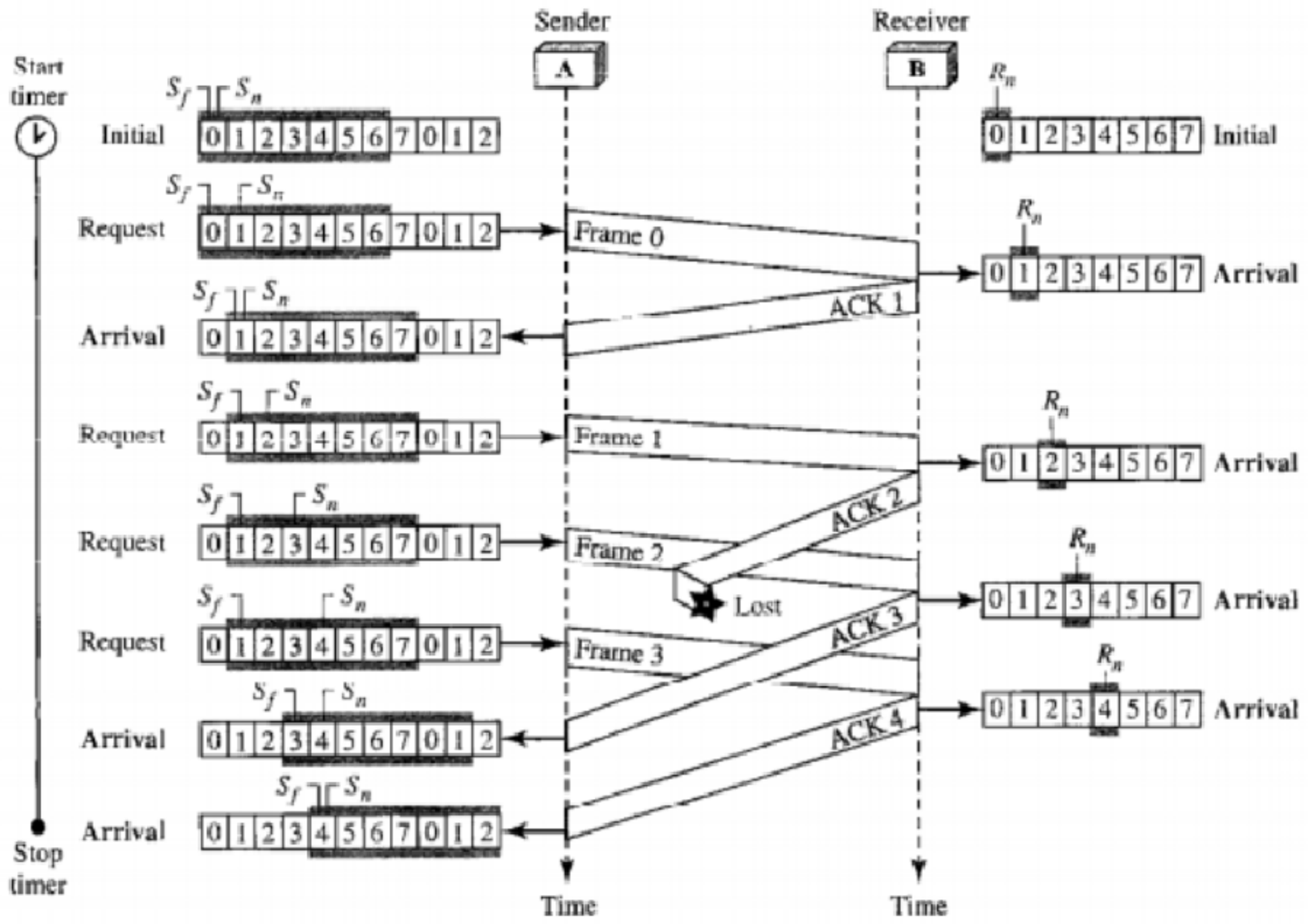
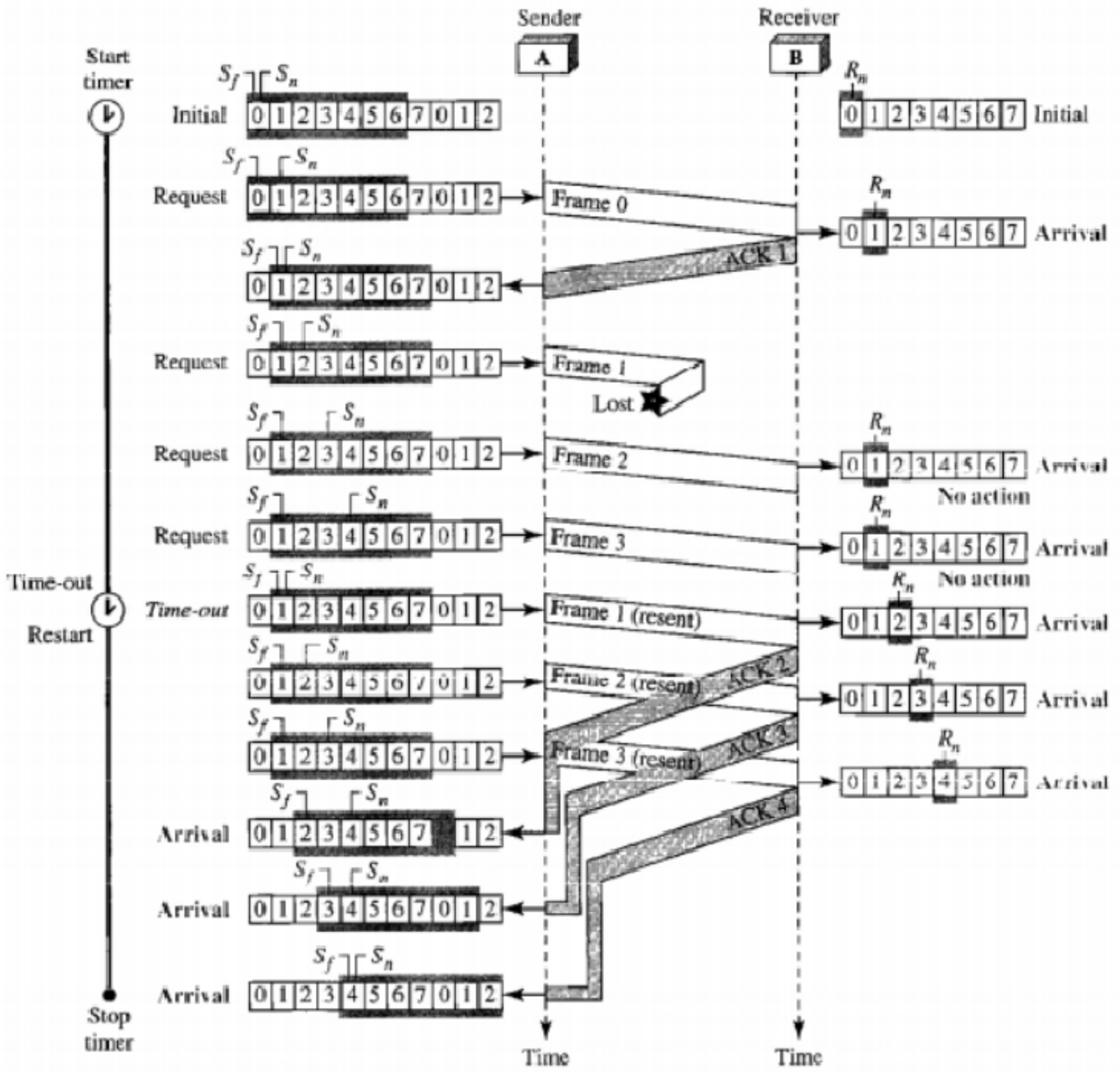


Figure 11.17 *Flow diagram for Example 11.7*

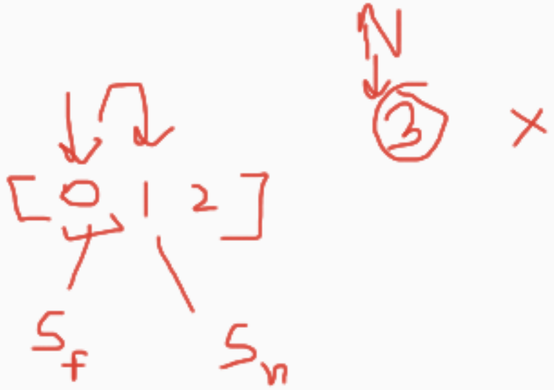


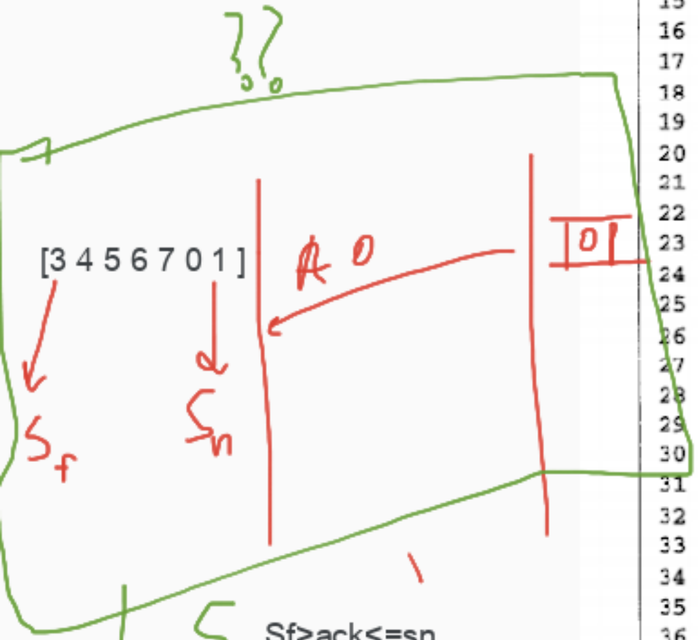
Algorithm 11.7 *Go-Back-N sender algorithm*

```
Sw = 2^m - 1;
Sf = 0;
Sn = 0;

while (true)                          //Repeat forever
{
 WaitForEvent();
  if(Event(RequestToSend))            //A packet to send
  {
     if(Sn-Sf >= Sw)                  //If window is full
          Sleep();
     GetData();
     MakeFrame(Sn);
     StoreFrame(Sn);
     SendFrame(Sn);
     Sn = Sn + 1;
     if(timer not running)
          StartTimer();
  }

  if(Event(ArrivalNotification))      //ACK arrives
  {
    Receive(ACK);
    if(corrupted(ACK))
         Sleep();
    if((ackNo>Sf)&&(ackNo<=Sn))       //If a valid ACK
    while(Sf <= ackNo)
     {
      PurgeFrame(Sf);
      Sf = Sf + 1;
     }
     StopTimer();
  }

  if(Event(TimeOut))                  //The timer expires
  {
   StartTimer();
   Temp = Sf;
   while(Temp < Sn);
    {
     SendFrame(Sf);
     Sf = Sf + 1;
    }
  }
}
```

1. data from nw layer
2. ack arrival from pl
3. timeout

m=2, w.size = 3, sn=0,sf=0
Initial w = [0 1 2] 3 0 1 2.....





Sf>ack<=sn
Sf<ack<=sn
Sf<ack>=sn

temp =temp +1

temp