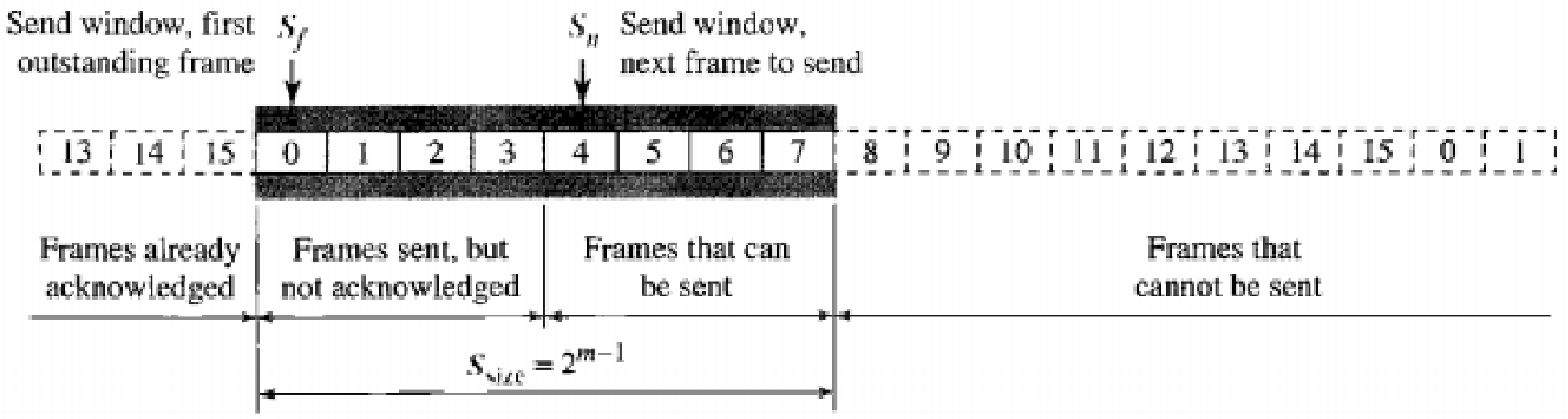
Figure 11.18 *Send window for Selective Repeat ARQ*



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
$S_{size} = 2^{m-1}$

Figure 11.19 *Receive window for Selective Repeat ARQ*

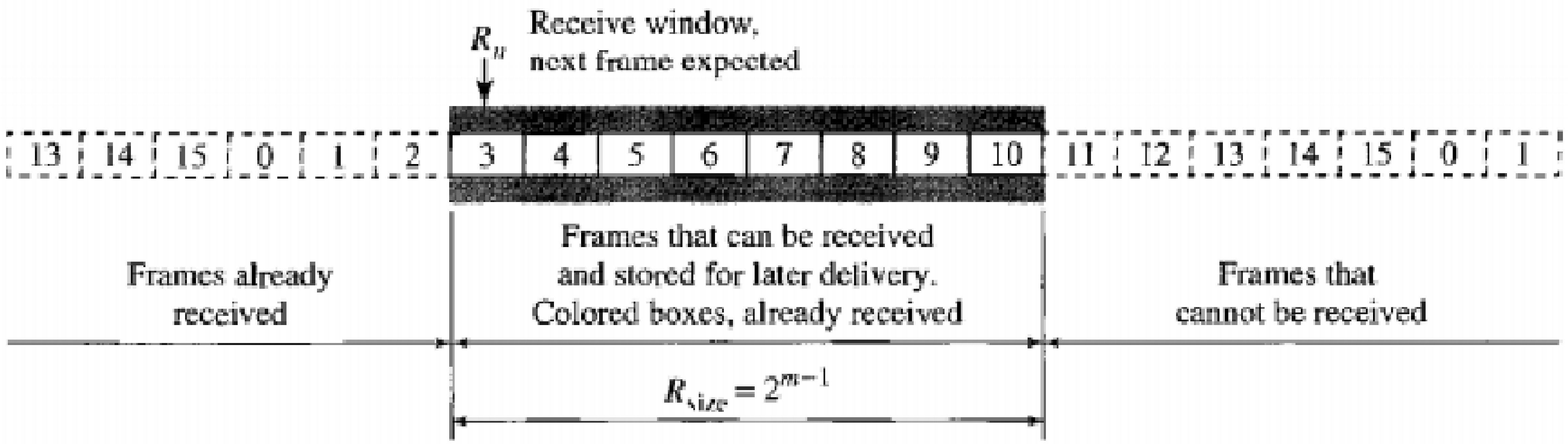


Figure 11.20 *Design of Selective Repeat ARQ*

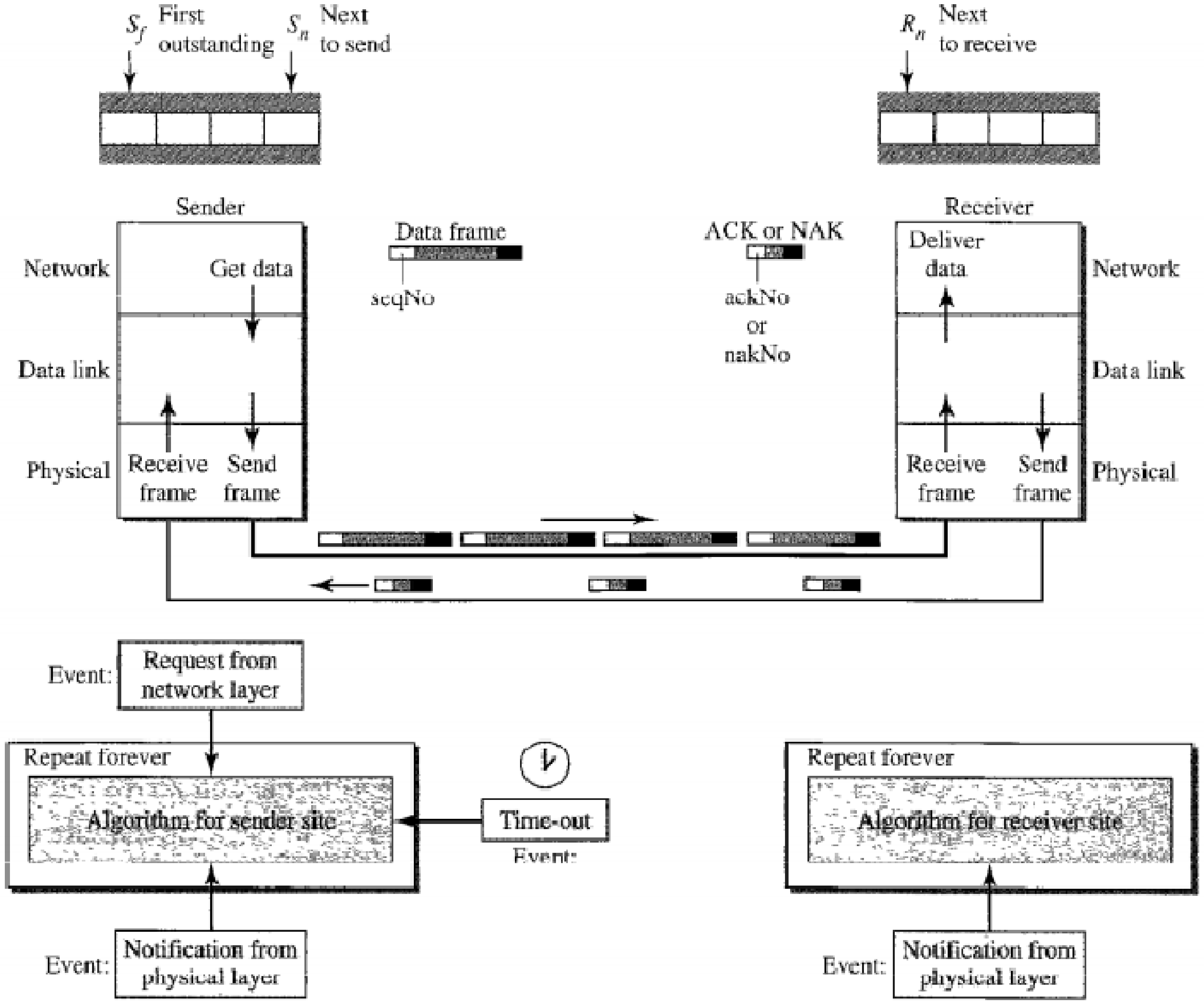
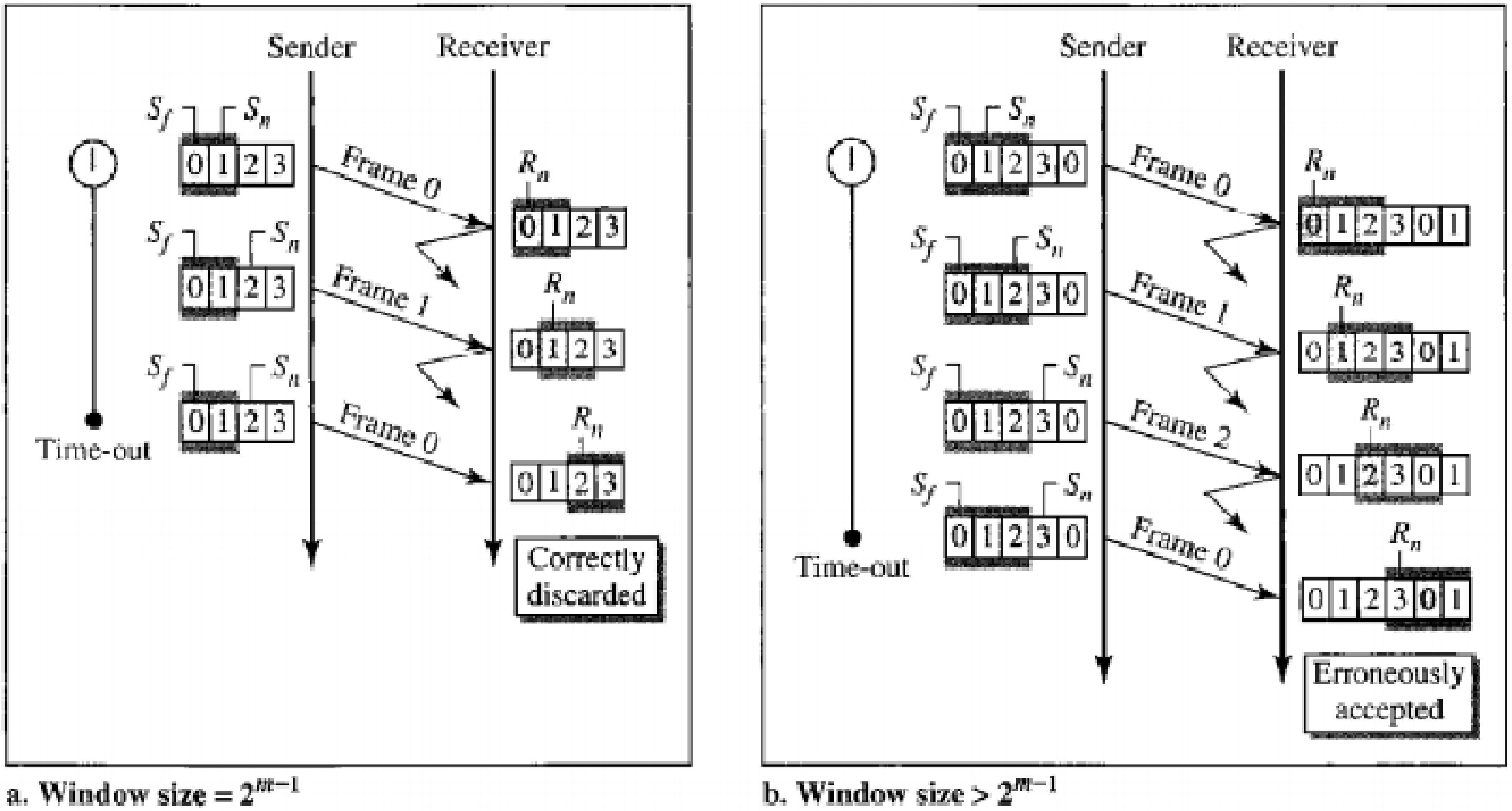
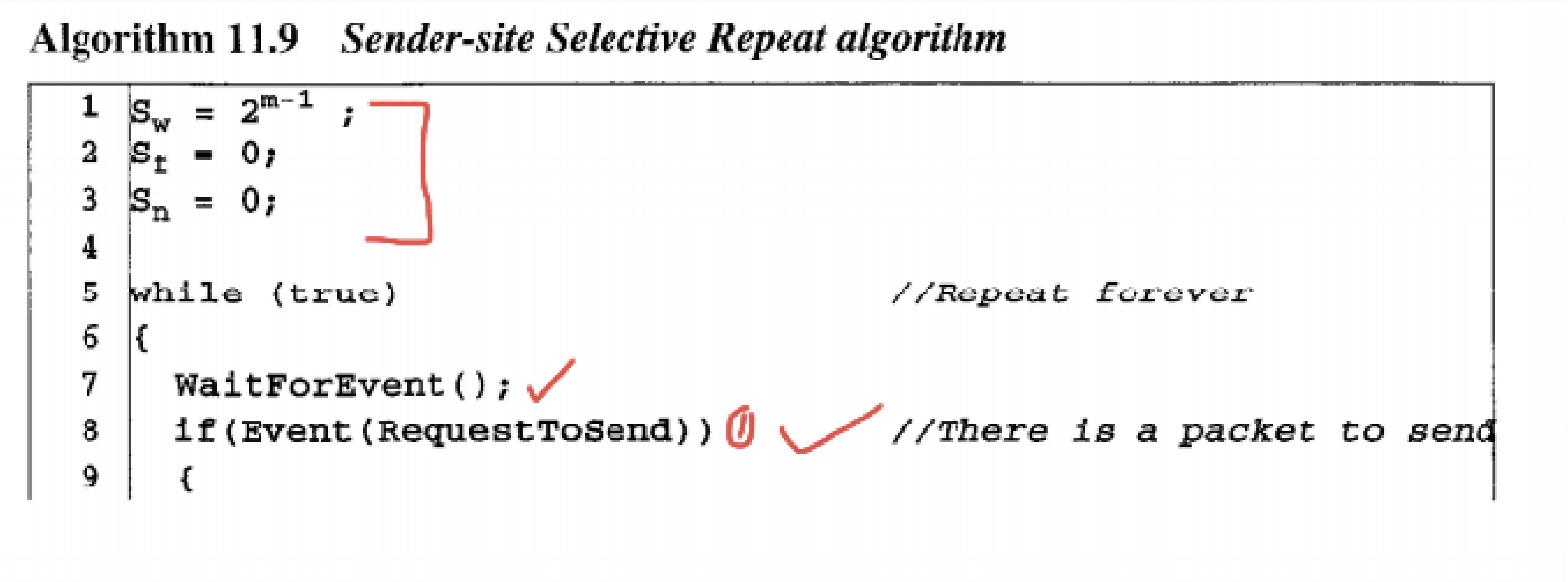


Figure 11.21 *Selective Repeat ARQ, window size*

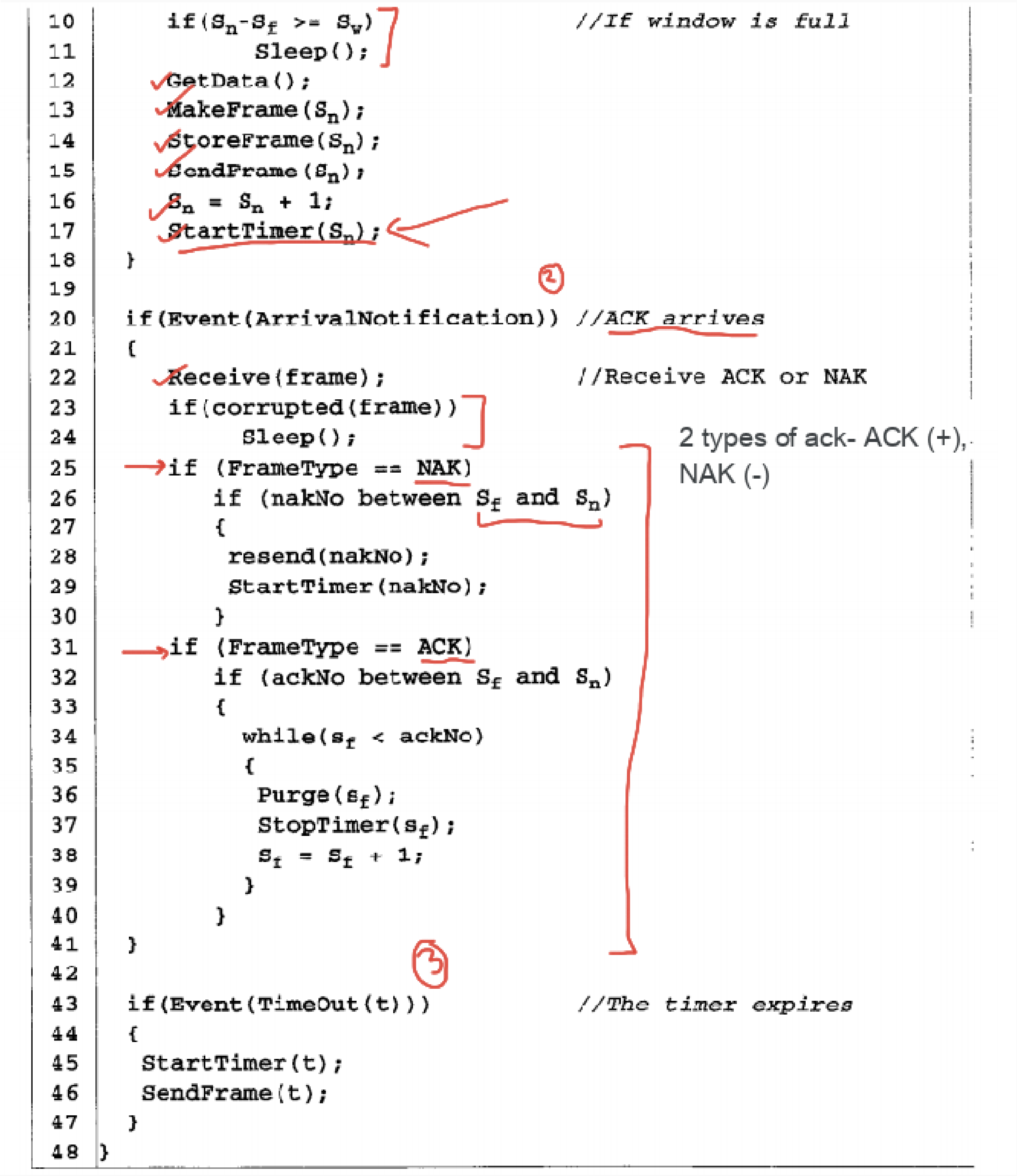




Algorithm 11.9 *Sender-site Selective Repeat algorithm*

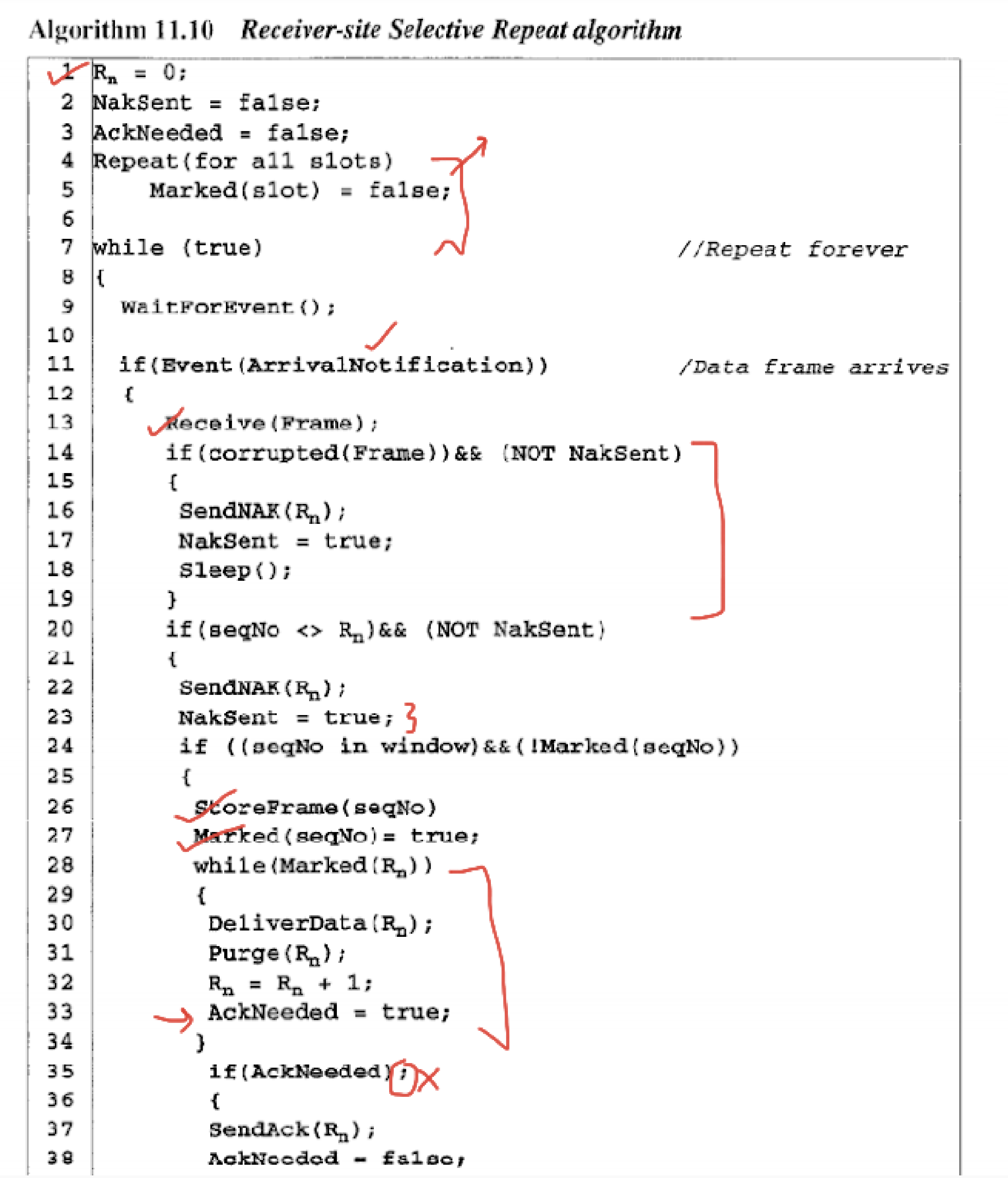
```
$S_w = 2^{m-1}$ ;
$S_f = 0$;
$S_n = 0$;

while (true)                              //Repeat forever
{
  WaitForEvent();
  if(Event(RequestToSend)) ①              //There is a packet to send
  {
```



```
     if($S_n$-$S_f$ >= $S_w$)                    //If window is full
            Sleep();
     GetData();
     MakeFrame($S_n$);
     StoreFrame($S_n$);
     SendFrame($S_n$);
     $S_n$ = $S_n$ + 1;
     StartTimer($S_n$);
  }
                                       ②
  if(Event(ArrivalNotification))  //ACK arrives
  {
     Receive(frame);                     //Receive ACK or NAK
     if(corrupted(frame))
           Sleep();
     if (FrameType == NAK)
         if (nakNo between $S_f$ and $S_n$)
         {
          resend(nakNo);
          StartTimer(nakNo);
         }
     if (FrameType == ACK)
         if (ackNo between $S_f$ and $S_n$)
         {
           while($s_f$ < ackNo)
           {
            Purge($s_f$);
            StopTimer($s_f$);
            $s_f$ = $s_f$ + 1;
           }
         }
  }
                                 ③
  if(Event(TimeOut(t)))                  //The timer expires
  {
   StartTimer(t);
   SendFrame(t);
  }
}
```

2 types of ack- ACK (+), NAK (-)

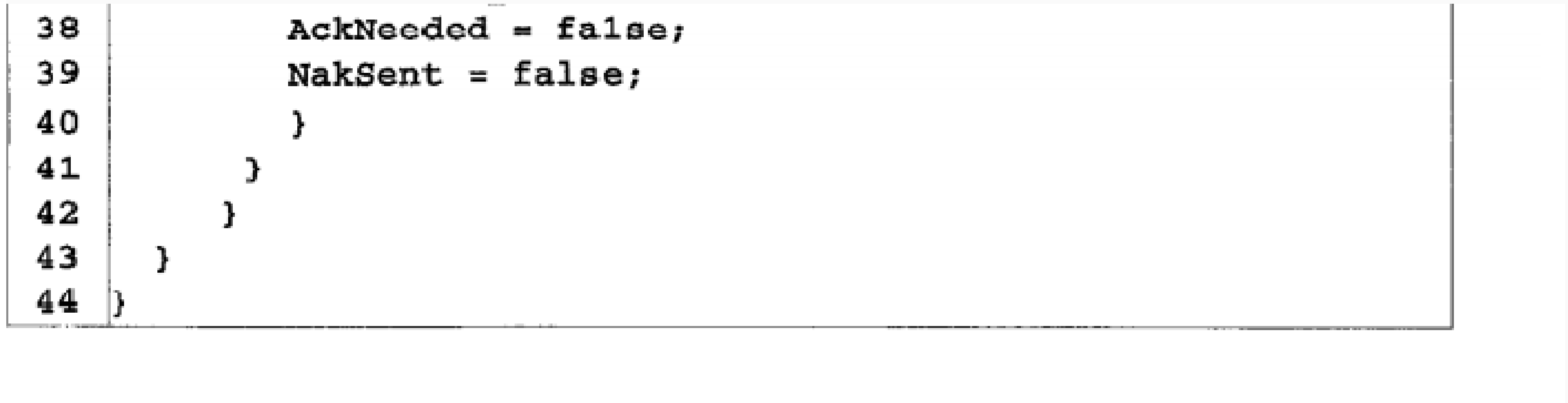


Algorithm 11.10 *Receiver-site Selective Repeat algorithm*

```
Rn = 0;
NakSent = false;
AckNeeded = false;
Repeat(for all slots)
    Marked(slot) = false;

while (true)                            //Repeat forever
{
  WaitForEvent();

  if(Event(ArrivalNotification))        /Data frame arrives
  {
     Receive(Frame);
     if(corrupted(Frame))&& (NOT NakSent)
     {
      SendNAK(Rn);
      NakSent = true;
      Sleep();
     }
     if(seqNo <> Rn)&& (NOT NakSent)
     {
      SendNAK(Rn);
      NakSent = true;
      if ((seqNo in window)&&(!Marked(seqNo))
      {
       StoreFrame(seqNo)
       Marked(seqNo)= true;
       while(Marked(Rn))
       {
        DeliverData(Rn);
        Purge(Rn);
        Rn = Rn + 1;
        AckNeeded = true;
       }
        if(AckNeeded);
        {
        SendAck(Rn);
        AckNeeded = false;
```



AckNeeded = false;

```
        NakSent = false;
        }
       }
      }
   }
}
```

$\text{Efficiency} = ws \times (L/L+BR)$

$ws = 2^{(m-1)}$

$\text{Throughput} = E \times B$

Question 1

Consider selective repeat protocol that uses a frame size of 1KB to send data on a 1.5Mbps link with one way latency of 50ms. What is the minimum number of bits required to represent sequence number field if link utilization is 60% ?

QUESTION 2

In SR protocol, suppose frames through 0 to 4 have been transmitted. Now imagine that frame 0 times out, 5 (a new frame) is transmitted. 1 times out, 2 times out and 6 (another new frame) is transmitted.

At this point, what will be the outstanding frames in sender's window ?