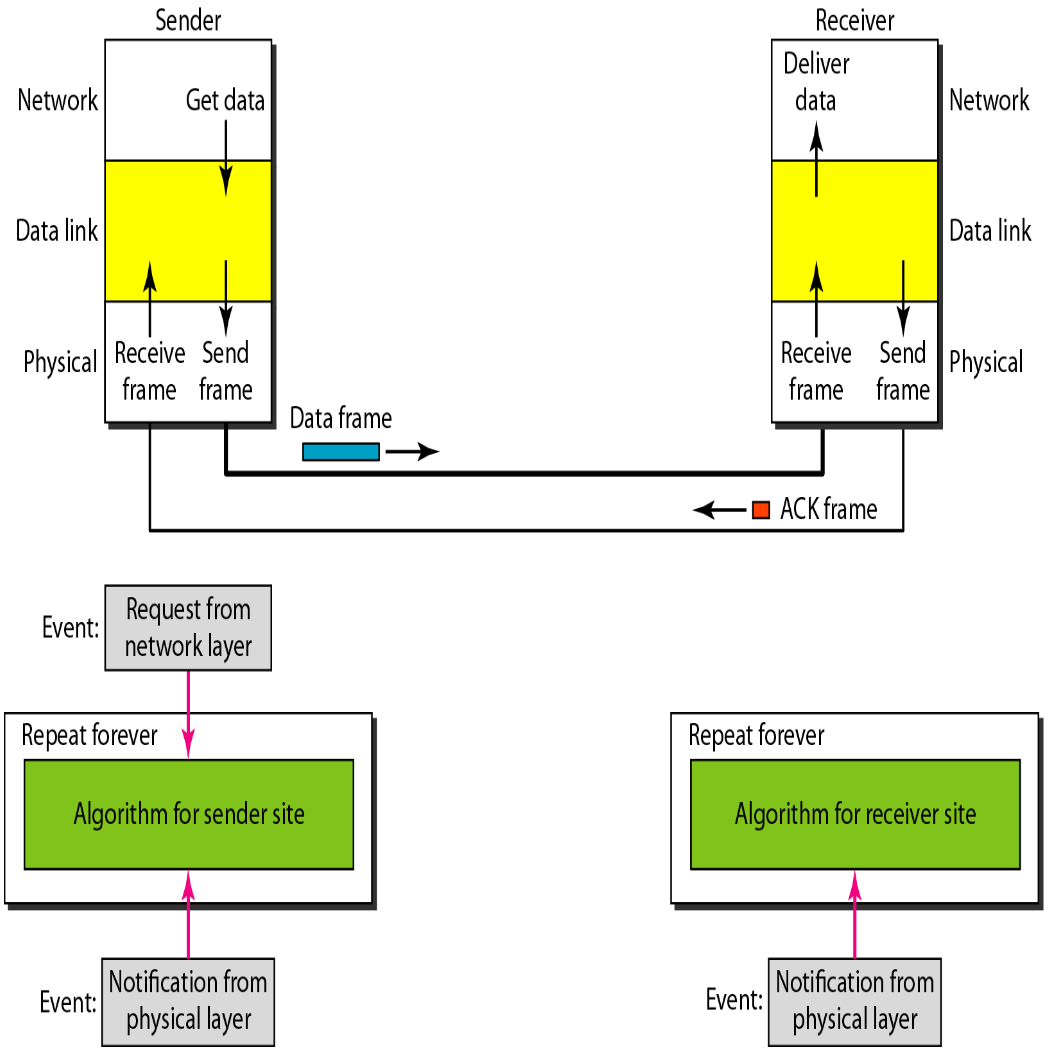
Computer Networks

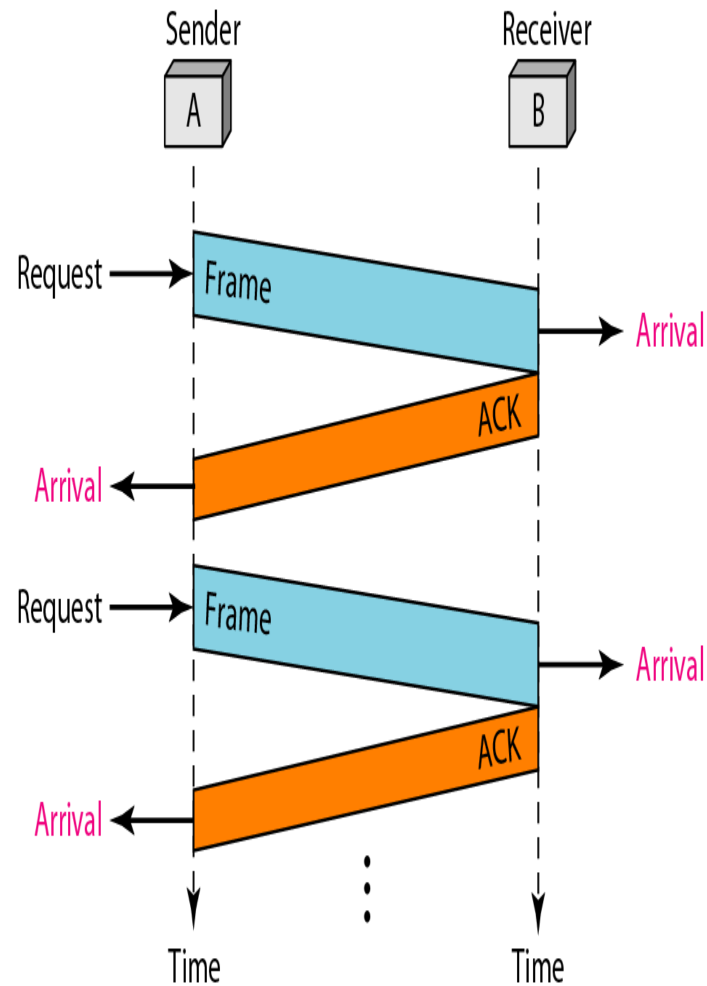
Day- 5

Recap

- Day-1 : Concept of Layering
- Day-2 : LAN Technologies (Ethernet)
- Day-3: Basic Wifi
- Day-4: Error Detection & correction, framing, DL protocols over noiseless channel

Design of Stop-and-Wait Protocol





Sender-site algorithm for Stop-and-Wait Protocol

```
while(true)                                //Repeat forever
canSend = true                             //Allow the first frame to go
{
  WaitForEvent();                          // Sleep until an event occurs
  if(Event(RequestToSend) AND canSend)
  {
      GetData();
      MakeFrame();
      SendFrame();                         //Send the data frame
      canSend = false;                     //Cannot send until ACK arrives
  }
  WaitForEvent();                          // Sleep until an event occurs
  if(Event(ArrivalNotification)            // An ACK has arrived
   {
      ReceiveFrame();                      //Receive the ACK frame
      canSend = true;
   }
}
```

Receiver-site algorithm for Stop-and-Wait Protocol

```
while(true)                                //Repeat forever
{
  WaitForEvent();                          // Sleep until an event occurs
  if(Event(ArrivalNotification))           //Data frame arrives
  {
      ReceiveFrame();
      ExtractData();
      Deliver(data);                       //Deliver data to network layer
      SendFrame();                         //Send an ACK frame
  }
}
```

NOISY CHANNELS

Although the Stop-and-Wait Protocol gives us an idea of how to add flow control to its predecessor, noiseless channels are nonexistent.

- Stop-and-Wait Automatic Repeat Request
- Go-Back-N Automatic Repeat Request
- Selective Repeat Automatic Repeat Request

Stop-and-Wait Automatic Repeat Request (One bit sliding window protocol)

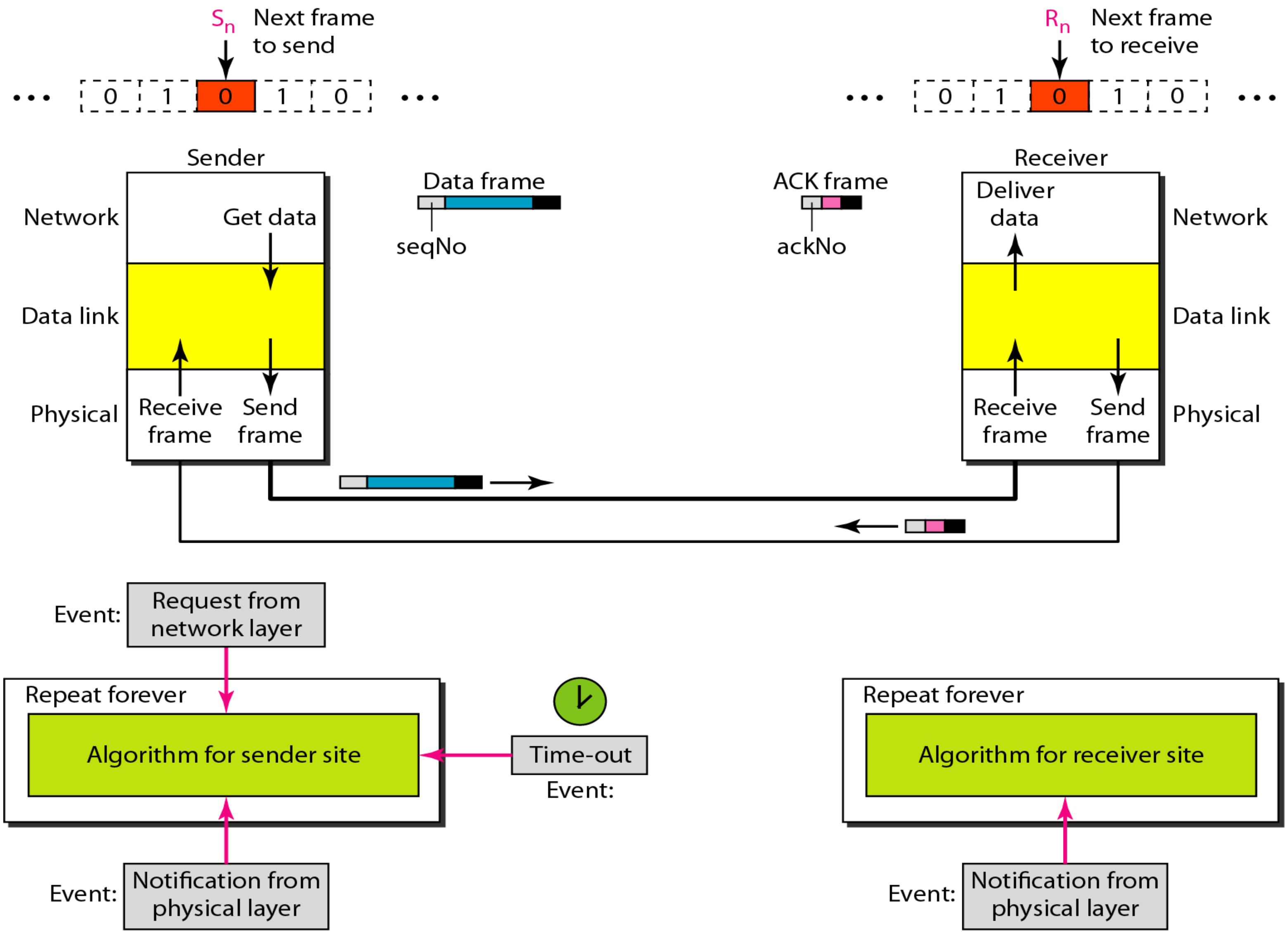
- Error correction in Stop-and-Wait ARQ is done by keeping a copy of the sent frame and retransmitting of the frame when the timer expires.

- In Stop-and-Wait ARQ, we use sequence numbers to number the frames.

- The sequence numbers are based on modulo-2 arithmetic.

- In Stop-and-Wait ARQ, the acknowledgment number always announces in modulo-2 arithmetic the sequence number of the next frame expected.

Design of the Stop-and-Wait ARQ Protocol



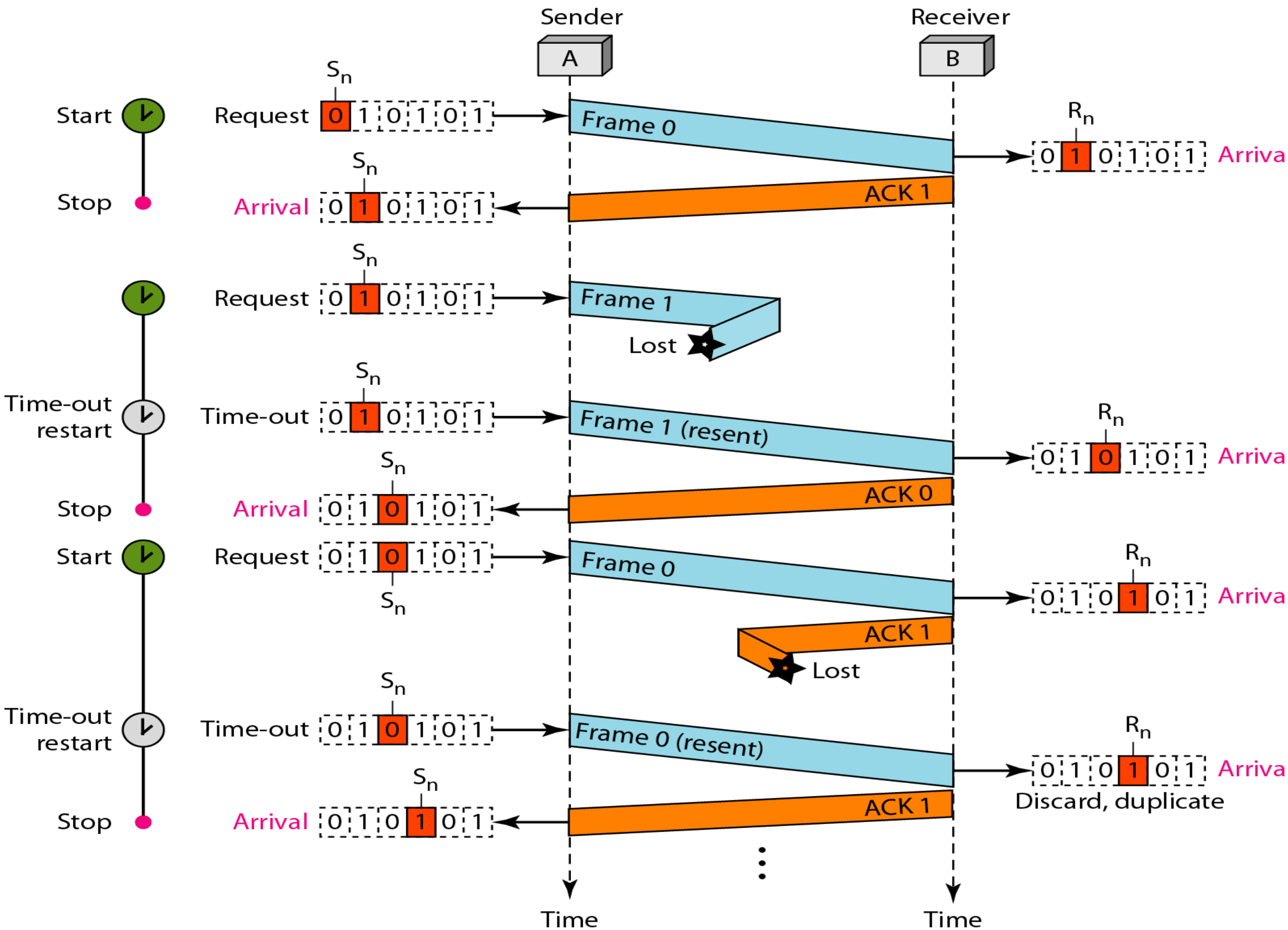
Sender-site algorithm for Stop-and-Wait ARQ

```
S_n = 0;                          // Frame 0 should be sent fi
canSend = true;                   // Allow the first request to
while(true)                       // Repeat forever
{
  WaitForEvent();                 // Sleep until an event occu
  if(Event(RequestToSend) AND canSend)
  {
     GetData();
     MakeFrame(S_n);                        //The seqNo is S_n
     StoreFrame(S_n);                       //Keep copy
     SendFrame(S_n);
     StartTimer();
     S_n = S_n + 1;
     canSend = false;
  }
  WaitForEvent();                           // Sleep
    if(Event(ArrivalNotification)     // An ACK has arrived
    {
      ReceiveFrame(ackNo);            //Receive the ACK frame
      if(not corrupted AND ackNo == S_n) //Valid ACK
        {
          Stoptimer();
          PurgeFrame(S_n-1);          //Copy is not needed
          canSend = true;
        }
     }

    if(Event(TimeOut)                 // The timer expired
    {
     StartTimer();
     ResendFrame(S_n-1);              //Resend a copy check
    }
}
```

Receiver-site algorithm for Stop-and-Wait ARQ Protocol

```
Rn = 0;                          // Frame 0 expected to arrive first
while(true)
{
  WaitForEvent();                // Sleep until an event occurs
  if(Event(ArrivalNotification))  //Data frame arrives
  {
     ReceiveFrame();
     if(corrupted(frame));
        sleep();
     if(seqNo == Rn)               //Valid data frame
     {
      ExtractData();
       DeliverData();              //Deliver data
       Rn = Rn + 1;
     }
      SendFrame(Rn);               //Send an ACK
  }
}
```

Flow diagram



Exercise

Assume that, 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?

The bandwidth-delay product : $(1 \times 10^6) \times (20 \times 10^{-3}) = 20{,}000$ 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.

link utilization = 1000/20,000, or 5 percent.

For this reason, for a link with a high bandwidth or long delay, the use of Stop-and-Wait ARQ wastes the capacity of the link.

Exercise

What is the utilization percentage of the link in above example if we have a protocol that can send up to 15 frames before stopping and worrying about the acknowledgments?

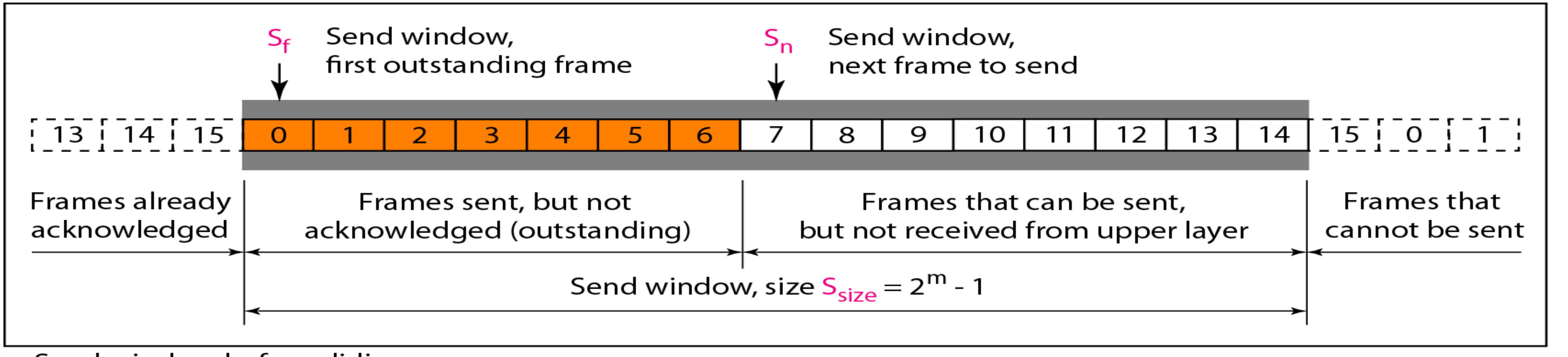
Solution

The bandwidth-delay product is still 20,000 bits. 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.

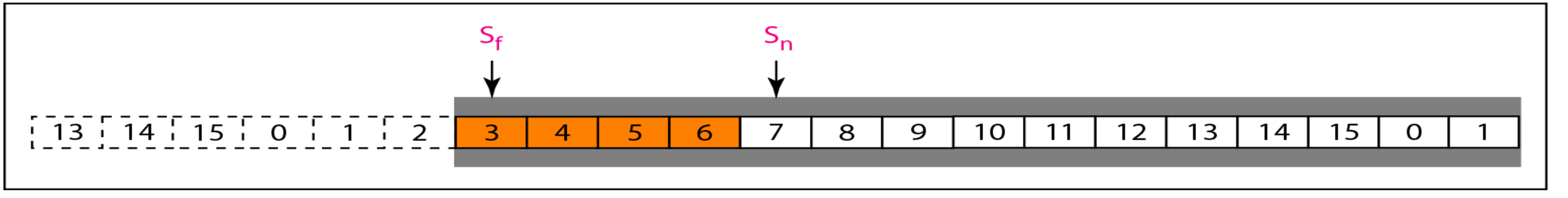
Go-Back-N Protocol

- 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

Send window for Go-Back-N ARQ



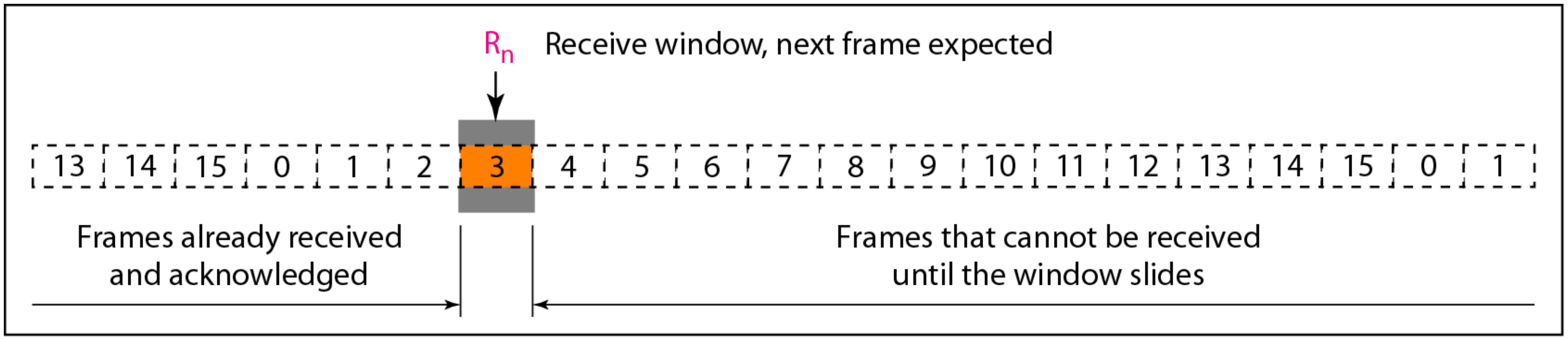
a. Send window before sliding



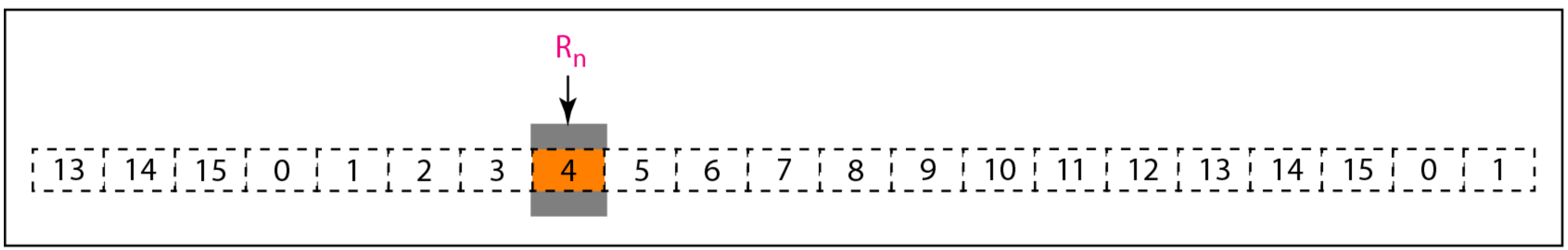
b. Send window after sliding

The send window is an abstract concept defining an imaginary box of size $2^m - 1$ with three variables: Sf, Sn, and S_{size}.
The send window can slide one or more slots when a valid acknowledgment arrives.

Receive window for Go-Back-N ARQ



a. Receive window

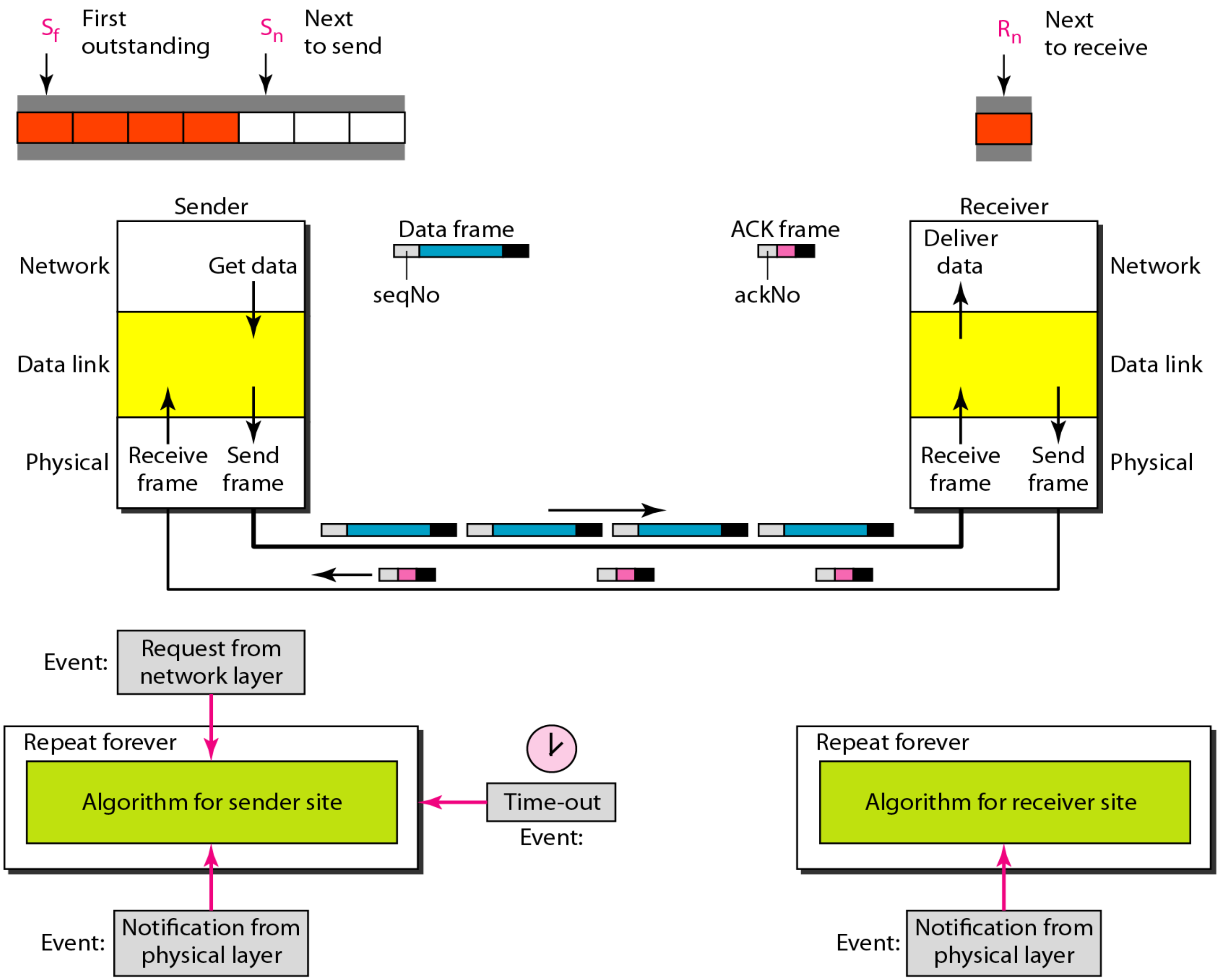


b. Window after sliding

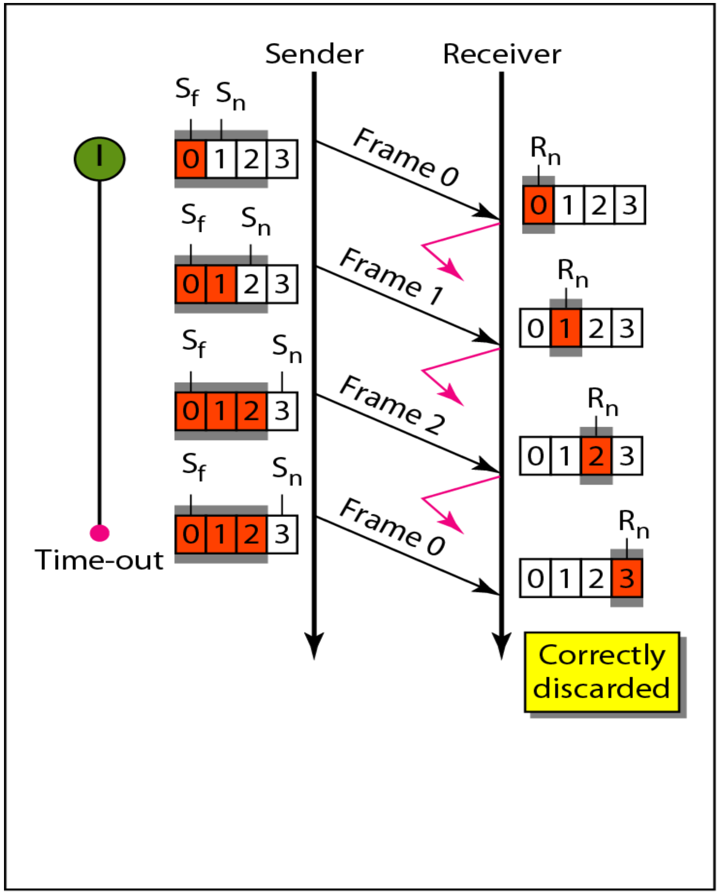
The receive window is an abstract concept defining an imaginary box of size 1 with one single variable R_n.

The window slides when a correct frame has arrived; sliding occurs one slot at a time.

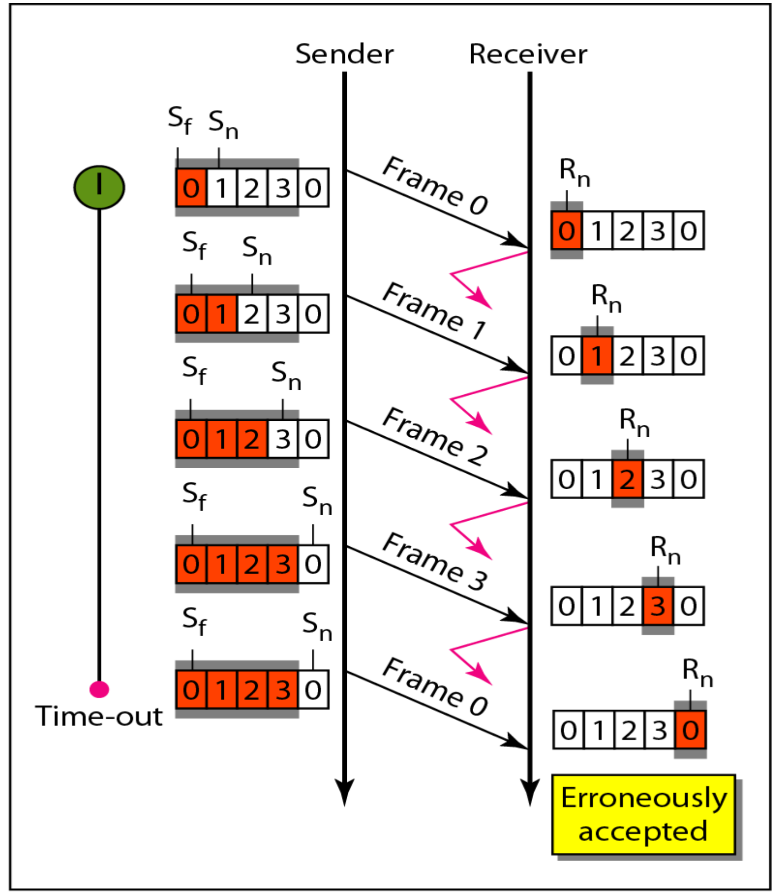
Design of Go-Back-N ARQ



Window size for Go-Back-N ARQ



a. Window size $< 2^m$



b. Window size $= 2^m$

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.

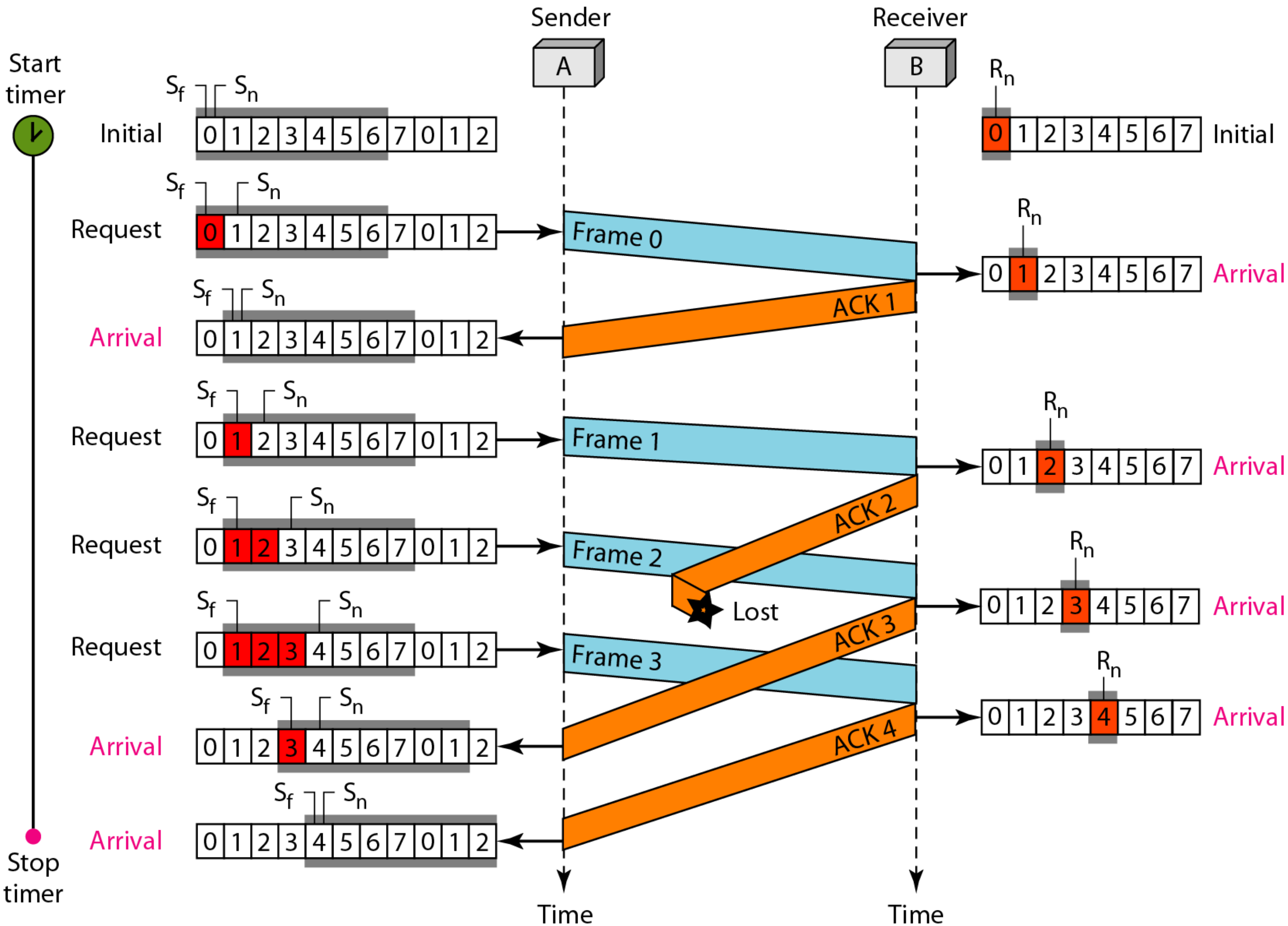
Go-Back-N sender algorithm

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

while (true)                          //Repeat forever
{
 WaitForEvent();
  if(Event(RequestToSend))            //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;
     if(timer not running)
          StartTimer();
  }

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

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

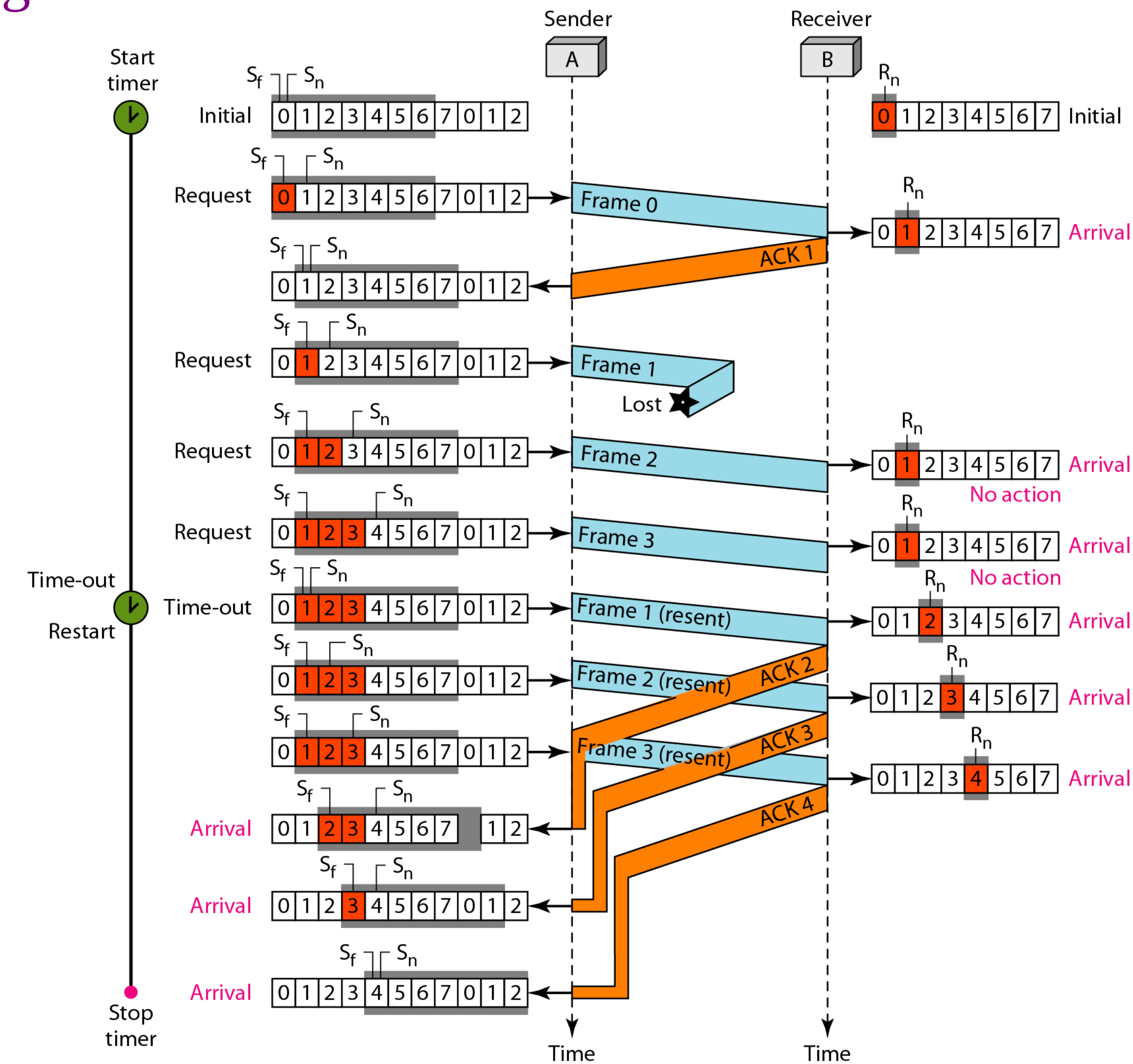
Go-Back-N receiver algorithm

```
Rn = 0;

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

  if(Event(ArrivalNotification)) /Data frame arrives
  {
     Receive(Frame);
     if(corrupted(Frame))
          Sleep();
     if(seqNo == Rn)                      //If expected frame
     {
       DeliverData();                     //Deliver data
       Rn = Rn + 1;                       //Slide window
       SendACK(Rn);
     }
  }
}
```

Flow diagram



Flow Diagram

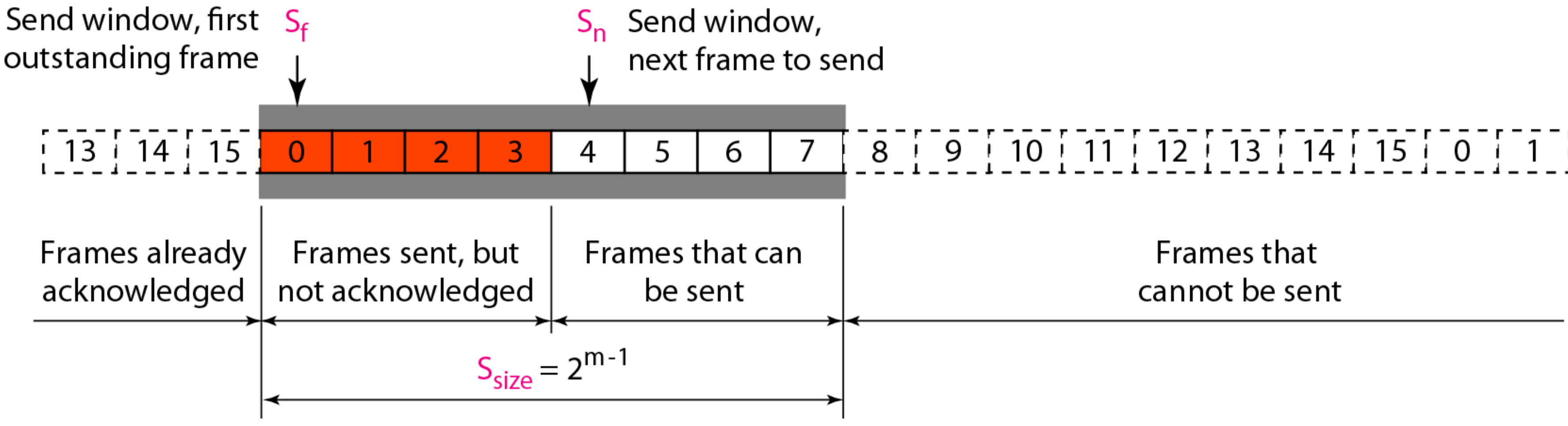


Stop-and-Wait ARQ is a special case of Go-Back-N ARQ in which the size of the send window is 1.

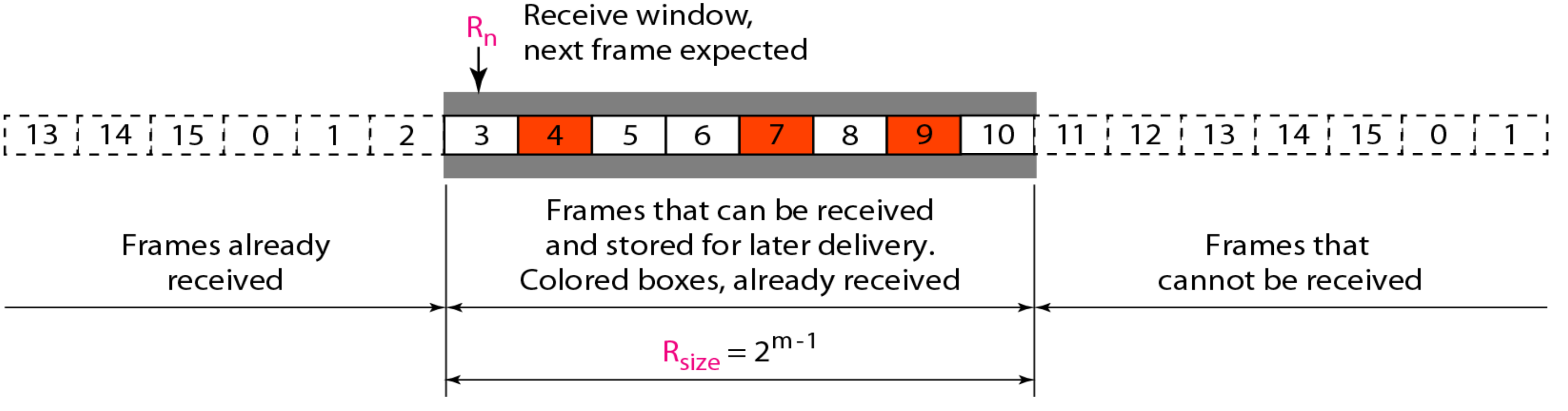
Selective Repeat Protocol

- In the selective repeat Protocol, the sequence numbers are modulo 2^m, where m is the size of the sequence number field in bits

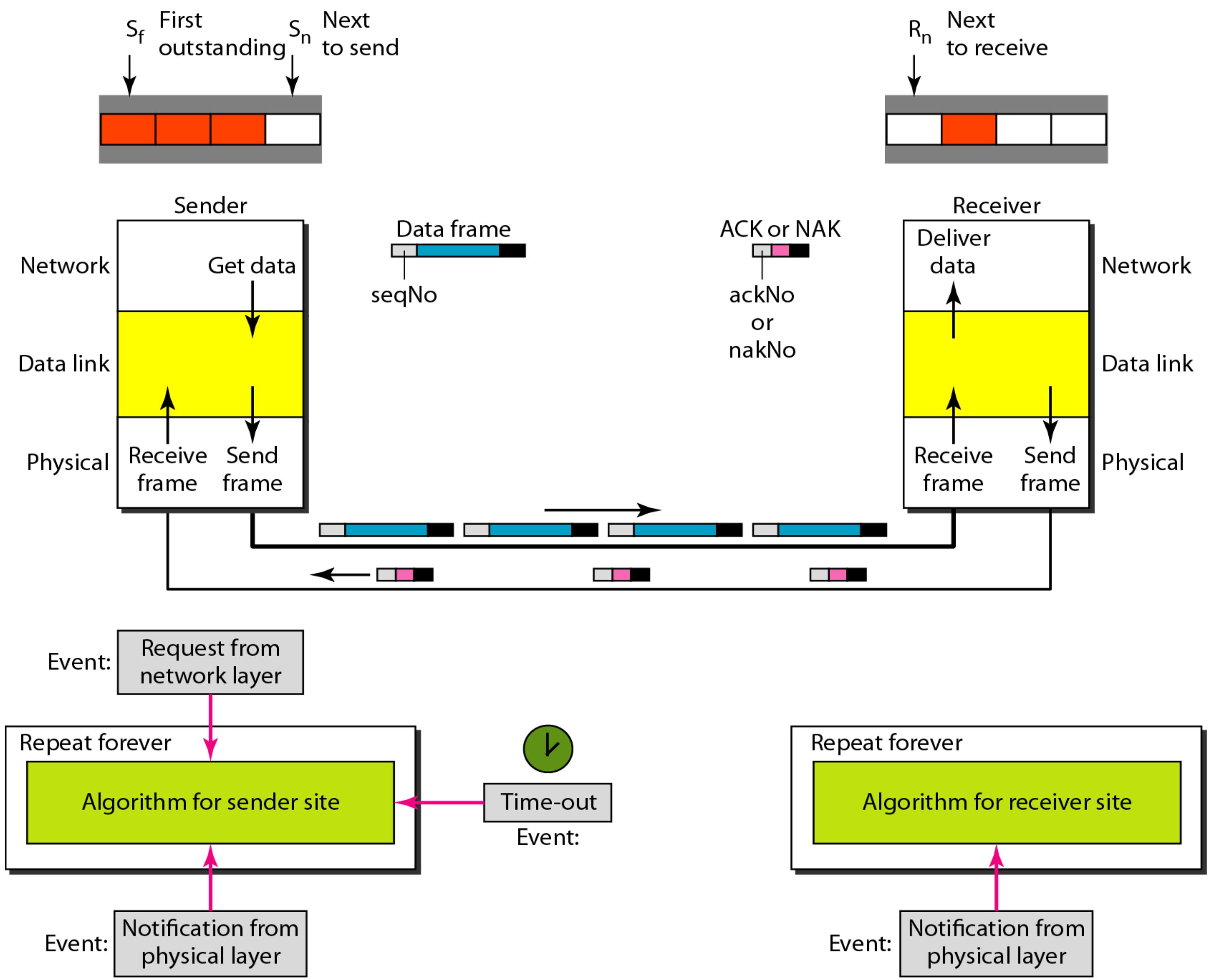
Send window for Selective Repeat ARQ



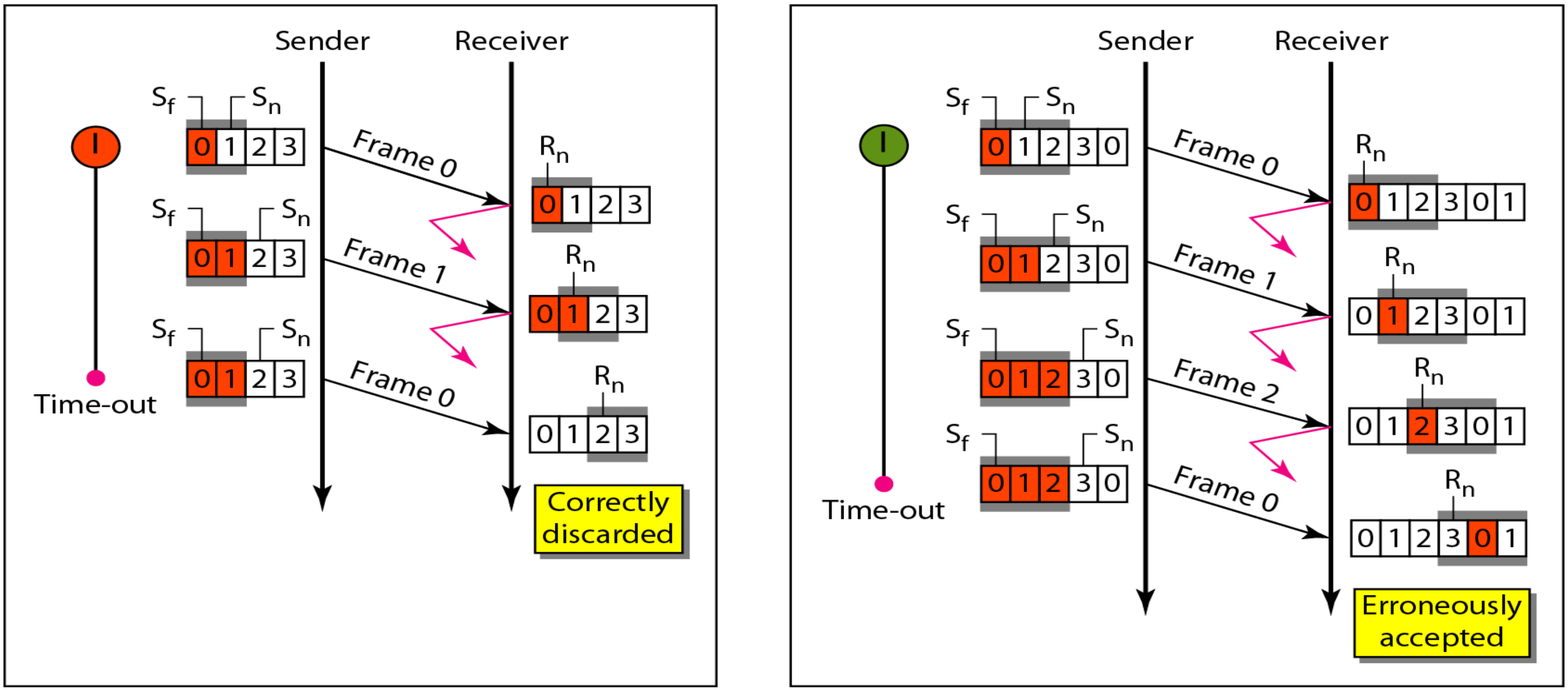
Receive window for Selective Repeat ARQ



Design of Selective Repeat ARQ



Selective Repeat ARQ, window size



a. Window size = 2^{m-1}

b. Window size > 2^{m-1}

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

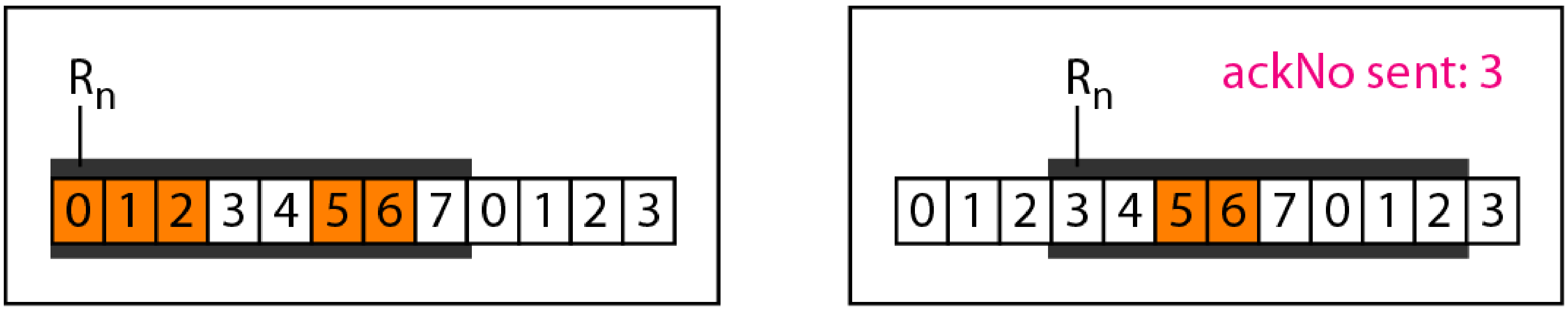
Sender-site Selective Repeat algorithm

```
S_w = 2^(m-1) ;
S_f = 0;
S_n = 0;

while (true)                              //Repeat forever
{
  WaitForEvent();
  if(Event(RequestToSend))                //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);
  }

  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;
          }
        }
  }

  if(Event(TimeOut(t)))           //The timer expires
  {
   StartTimer(t);
   SendFrame(t);
  }
}
```

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

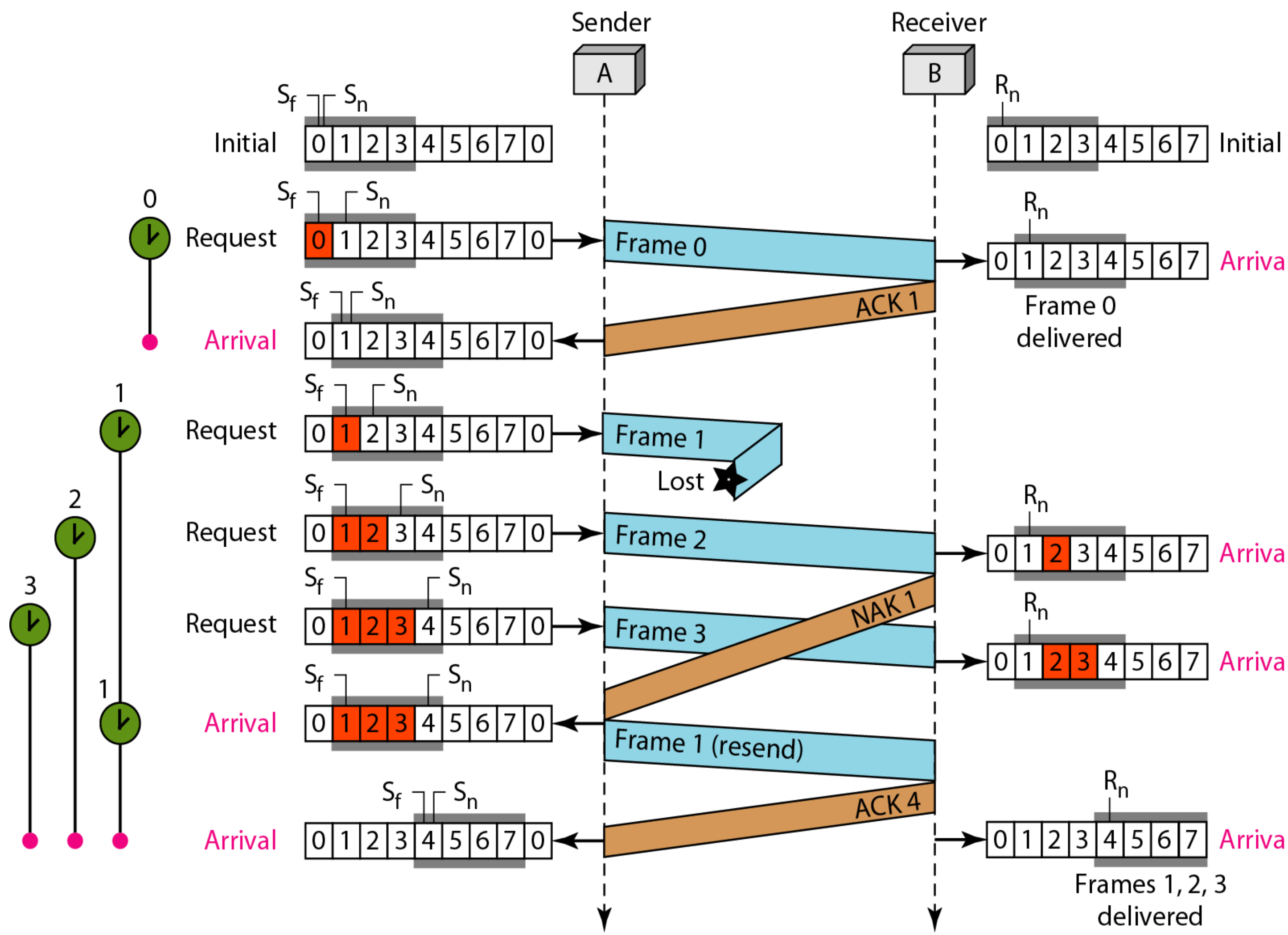
Delivery of data in Selective Repeat ARQ



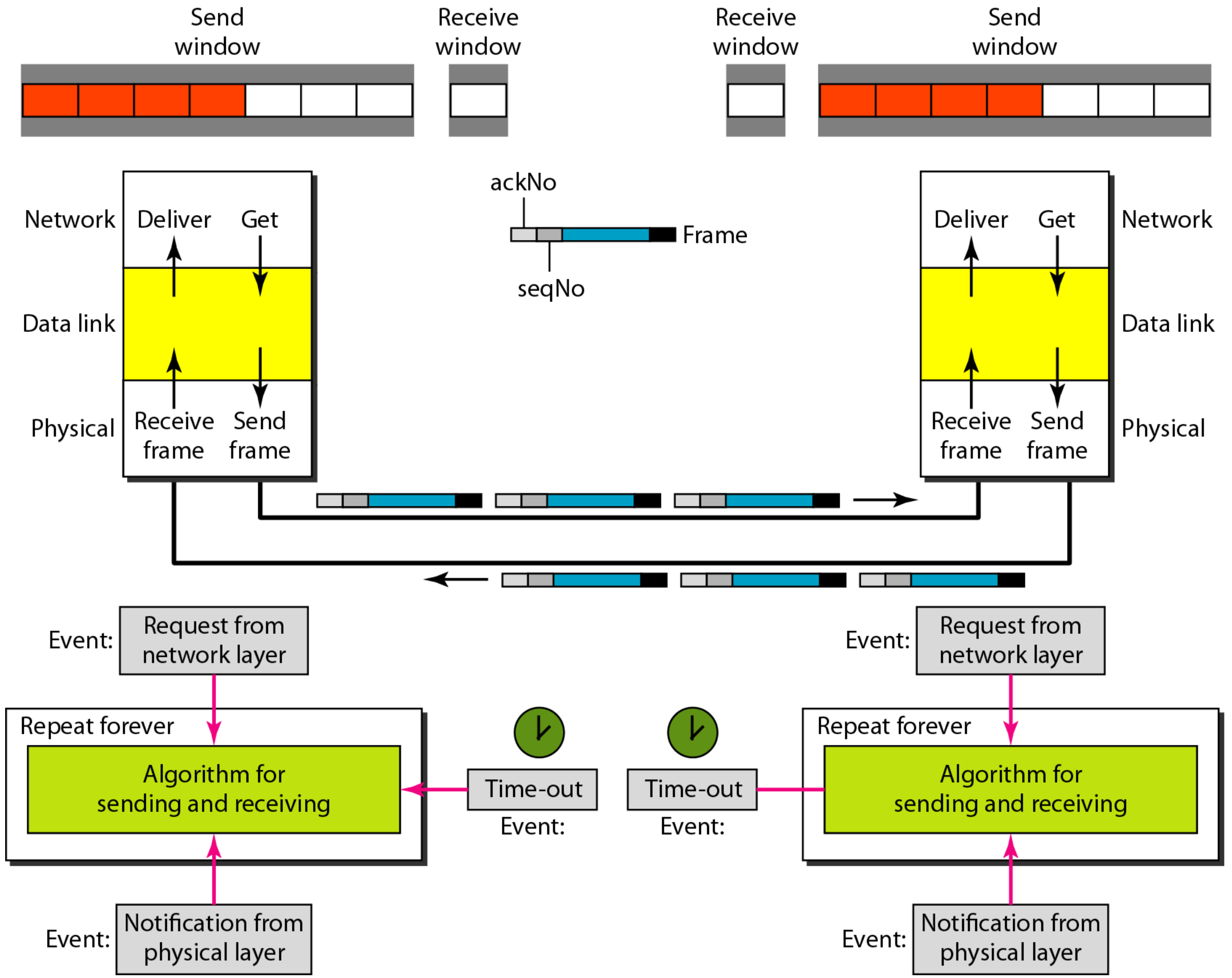
a. Before delivery

b. After delivery

Flow diagram



Design of piggybacking in Go-Back-N ARQ



Exercise

1. **If the bandwidth of the line is 1.5 Mbps, RTT is 45 msec and packet size is 1 KB, then find the link utilization in stop and wait.**

Sol: Bandwidth = 1.5 Mbps

RTT = 45 msec

Packet size = 1 KB

Transmission delay (T_t) = Packet size / Bandwidth = 1 KB / 1.5 Mbps

= (2^{10} x 8 bits) / (1.5 x 10^6 bits per sec) = 5.461 msec

Propagation delay (T_p) = Round Trip Time / 2 = = 45 msec / 2 = = 22.5 msec

Link utilization = Tt / (Tt+2Tp) = 5.461 / (5.461 +45) = 0.108 = 10.8%

= 1/(1+2a) a = Tp/Tt

Exercise

2. Consider a selective repeat sliding window protocol that uses a frame size of 1 KB to send data on a 1.5 Mbps link with a one-way latency of 50 msec. To achieve a link utilization of 60%, the minimum number of bits required to represent the sequence number field is ________.

(A) 3

(B) 4

(C) 5

(D) 6

Exercise

2. Consider a selective repeat sliding window protocol that uses a frame size of 1 KB to send data on a 1.5 Mbps link with a one-way latency of 50 msec. To achieve a link utilization of 60%, the minimum number of bits required to represent the sequence number field is ________.

(A) 3

(B) 4

(C) 5

(D) 6

Sol:

Transmission delay = Frame Size/bandwidth = (1024*8)/(1.5 * 10^6)=5.46ms

Propagation delay = 50ms

Efficiency = Window Size/(1+2a) = .6

a = Propagation delay/Transmission delay = 50/5.56 = 9.157

window size = 0.6(1+18.314) = 11.588(approx)

$w= 2^{m-1}$

$m = \log_2 (w*2) = \log_2 (11.588*2) = 4.534 = 5$

Exercise

3.The distance between two stations M and N is L kilometers. All frames are K bits long. The propagation delay per kilometer is t seconds. Let R bits/second be the channel capacity. Assuming that processing delay is negligible, the minimum number of bits for the sequence number field in a frame for maximum utilization, when the sliding window protocol is used, is:

(A) $\left\lceil \log_2 \frac{2LtR + 2K}{K} \right\rceil$

(B) $\left\lceil \log_2 \frac{2LtR}{K} \right\rceil$

(C) $\left\lceil \log_2 \frac{2LtR + K}{K} \right\rceil$

(D) $\left\lceil \log_2 \frac{2LtR + K}{2K} \right\rceil$

Exercise

3.The distance between two stations M and N is L kilometers. All frames are K bits long. The propagation delay per kilometer is t seconds. Let R bits/second be the channel capacity. Assuming that processing delay is negligible, the minimum number of bits for the sequence number field in a frame for maximum utilization, when the sliding window protocol is used, is:

(A) $\left\lceil \log_2 \frac{2LtR + 2K}{K} \right\rceil$

(B) $\left\lceil \log_2 \frac{2LtR}{K} \right\rceil$

(C) $\left\lceil \log_2 \frac{2LtR + K}{K} \right\rceil$

(D) $\left\lceil \log_2 \frac{2LtR + K}{2K} \right\rceil$

Sol:

Tp = Lt

Tt = K/R

a=Tp/Tt = LtR/K

Eff = w/(1+2a) => w = 1+2a => w = (K+2LtR)/K

=> m = $\log_2$ (2LtR+K)/K

Exercise

4. Pick the incorrect statement in error retransmission used in contiguous ARQ method:

(a) Go-back-N method requires more storage at the receiving site

(b) Selective repeat has better line utilization

(c) Selective repeat is complex than Go-back-N

(d) Go-back-N has better line utilization

Exercise

4. Pick the incorrect statement in error retransmission used in contiguous ARQ method:

(a) Go-back-N method requires more storage at the receiving site

(b) Selective repeat has better line utilization

(c) Selective repeat is complex than Go-back-N

(d) Go-back-N has better line utilization

Exercise

5. Using stop and wait protocol, sender wants to transmit 10 data packets to the receiver. Out of these 10 data packets, every 4th data packet is lost. How many packets sender will have to send in total?

Sol: 13 packets

1234 456778 9 10 10

Exercise

6. The maximum window size for data transmission using the selective repeat protocol with n bit frame sequence numbers is-

(a) 2^n

(b) 2^{n-1}

(c) 2^n-1

(d) 2^{n-2}

Exercise

6. The maximum window size for data transmission using the selective repeat protocol with n bit frame sequence numbers is-

(a) 2^n

(b) 2^{n-1}

(c) 2^n-1

(d) 2^{n-2}

Exercise

7. **Station A uses 32 byte packets to transmit messages to Station B using a sliding window protocol. The round trip delay between A and B is 80 milliseconds and the bottleneck bandwidth on the path between A and B is 128 kbps. What is the optimal window size that A should use?**

   (A) 20

   (B) 40

   (C) 160

   (D) 320

Exercise

7. **Station A uses 32 byte packets to transmit messages to Station B using a sliding window protocol. The round trip delay between A and B is 80 milliseconds and the bottleneck bandwidth on the path between A and B is 128 kbps. What is the optimal window size that A should use?**

(A) 20

(B) 40

(C) 160

(D) 320

```
Round Trip propagation delay = 80ms
Frame size = 32*8 bits; Bandwidth = 128kbps
Transmission Time = 32*8/(128) ms = 2 ms
Let n be the window size.
UtiliZation = n/(1+2a) = n/(1+80/2)
where a = Propagation time / transmission time =
For maximum utilization: n = 41 which is close to option
(B)
```

Exercise

8. A bit-stuffing based framing protocol uses an 8-bit delimiter pattern of 01111110. If the output bit-string after stuffing is 01111100101, then the input bit-string is

(a) 0111110100

(b) 0111110101

(c) 0111111101

(d) 0111111111

Exercise

8. A bit-stuffing based framing protocol uses an 8-bit delimiter pattern of 01111110. If the output bit-string after stuffing is 01111100101, then the input bit-string is

(a) 0111110100

(b) 0111110101

(c) 0111111101

(d) 0111111111

Exercise

9. What is the hamming distance between 001111 and 010011 is

(a) 1

(b) 2

(c) 3

(d) 4

Exercise

9. What is the hamming distance between 001111 and 010011 is

(a) 1

(b) 2

(c) 3

(d) 4

Exercise

10. The technique of temporarily delaying outgoing acknowledgements so that they can be hooked onto the next outgoing data frame is called _______________

a) piggybacking

b) cyclic redundancy check

c) fletcher's checksum

d) parity check

Exercise

10. The technique of temporarily delaying outgoing acknowledgements so that they can be hooked onto the next outgoing data frame is called ________________

a) piggybacking

b) cyclic redundancy check

c) fletcher's checksum

d) parity check

Exercise

11. A link has a transmission speed of 10^6 bits/sec. It uses data packets of size 1000 bytes each. Assume that the acknowledgment has negligible transmission delay, and that its propagation delay is the same as the data propagation delay. Also assume that the processing delays at nodes are negligible. The efficiency of the stop-and-wait protocol in this setup is exactly 25%. The value of the one-way propagation delay (in milliseconds) is ___________.

(A) 4

(B) 8

(C) 12

(D) 16

Exercise

11. A link has a transmission speed of 10^6 bits/sec. It uses data packets of size 1000 bytes each. Assume that the acknowledgment has negligible transmission delay, and that its propagation delay is the same as the data propagation delay. Also assume that the processing delays at nodes are negligible. The efficiency of the stop-and-wait protocol in this setup is exactly 25%. The value of the one-way propagation delay (in milliseconds) is ____________.

(A) 4

(B) 8

(C) 12

(D) 16

$Tt = 8*1000/ 10^6 = 8ms$
Efficiency = $Tt/(Tt+2Tp)$
$0.25 = 8/(8+2Tp) \Rightarrow Tp = 24/2 = 12ms$