Chapter Two

The Data Link Layer

The basic service provided by the physical layer is transportation of bits over the physical connection. This service is unreliable in the sense that disturbed line conditions of the transmission medium may introduce errors which are not taken care of by the physical layer. Error control and other associated functions are carried out by the second layer, the Data Link Layer, of the OSI reference model.

2.1 Need for Data Link Layer

Let us consider two digital devices A and B need to exchange information. These devices could be computers, network nodes, or other data terminal equipment.

To Exchange digital information between devices A and B we require an interconnecting transmission medium to carry the electrical signals, and the physical layer to convert bits into the electrical signals and vice versa.

Together with the transmission, physical layer of the devices provide the capability for transparent exchange of bits over the physical connection but this capability has certain limitations:

- If the electrical signal gets impaired due to the noise encountered during transmission or due to the medium characteristics, error may be introduced in the data bits. Therefore, there is need to establish mechanisms to control transmission error.
- Error can also be introduced if the receiving device is not ready for the incoming bits and some of bits are lost. Therefore, a data flow control mechanism also needs to be implemented.

The physical layer does not meet these requirements. Error and flow control functions are implemented in the data link layer which ensures error-free transfer of bits from one device to the other.

2.2 Data Link Layer

Data link layer together with the physical layer and the interconnecting medium provide a data link connection for reliable transfer of data bits over an imperfect physical connection, Figure 2.1.

Data link layer incorporates certain processes which carry out error control, flow control, and the associated link management functions.

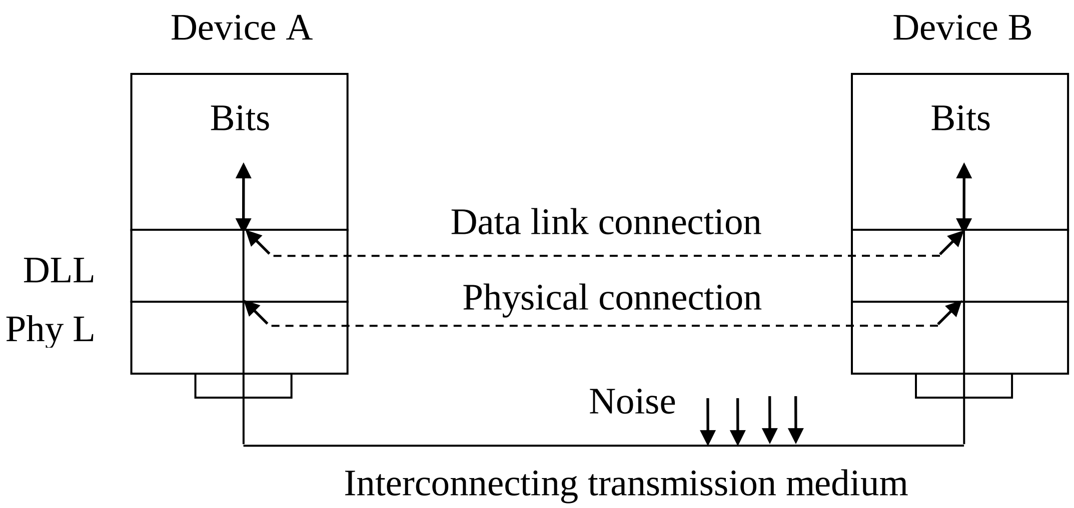


Figure 2.1 Reliable Transfer of Bits Over Data Link Connection

Data link layer receives the data to be sent to the other device from the next higher layer (network layer) and adds some control bits to a block of data bits. The data block along with the control bits is called frame. The frame is handed over to the physical layer. The physical layer converts bits of the frame into an electrical signal for transmission over interconnecting transmission medium, Figure 2.2.

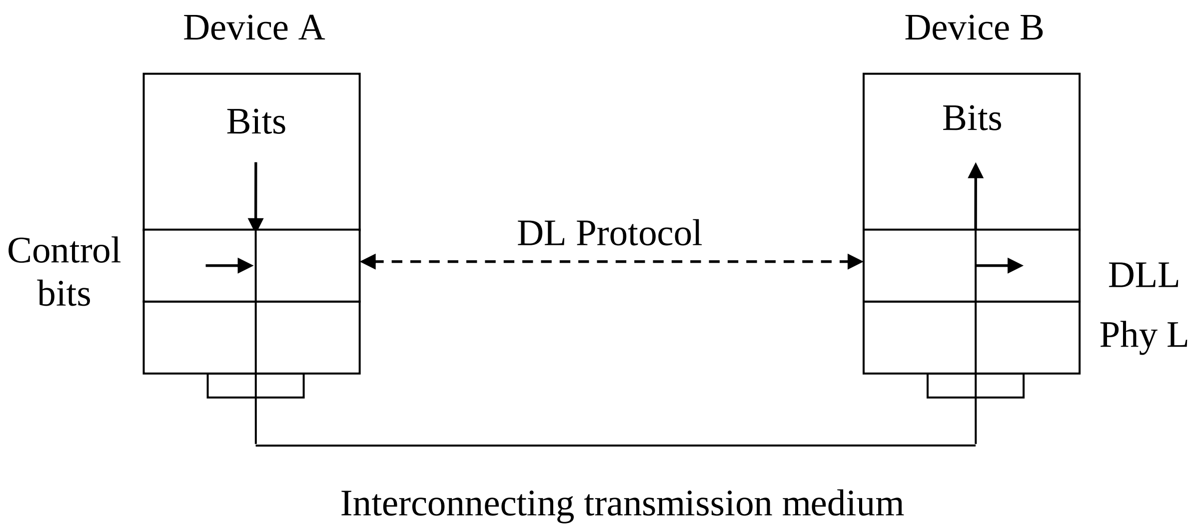


Figure 2.2 Formation of Frame in DLL

2.2.1 Service Provided by the Data Link Layer

Data link layer receives service from the physical layer and provides service to the network layer, which is the user of these services.

In the OSI terminology, the user data unit received from the network layer is called a Data Link Service Data Unit (DL-SDU) and the frame formed by adding control bits to the DL-SDU is called a Data Link Protocol Data Unit (DL-PDU) as showing in Figure 2.3. The control bits constitute a header and a trailer.

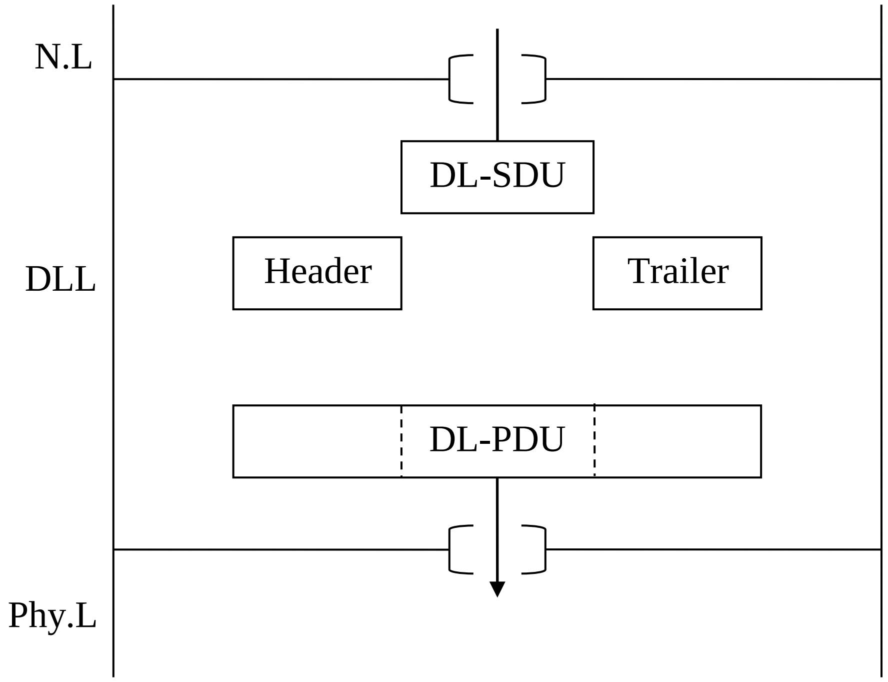


Figure 2.3 Data Link Service Data Units and Protocol Data Units

The basic service provided by the data link layer to the network layer is reliable transfer of DL-SDU over the data link connection which is established, maintained, and released by the data link layer on the request of the network layer. This basic service has the following associated features:

- **Sequencing**: The sequence integrity of the DL-SDUs is maintained.
- **Error notification**: If the data link layer detects an unrecoverable error, it notifies the network layer.
- **Flow control**: The network layer can control the rate at which it receives the DL-SDUs from the data link layer. This control may be reflected in the rate at which the data link layer will accept DL-SDUs at the other end.
- **Quality of service parameters**: The data link layer provides selectable quality of service parameters which include residual error rate, transit delay, throughput, etc. The selected quality of service is maintained during data link connection.

2.2.2 Data Link Protocol

It is essential that the structure of the frame is known to both the data link layers so that the control bits can be identified. The data link layers should also agree on the set of procedures to be adopted for exchange of control information. The specified set of rules and procedures for carrying out data link control function is called "data link protocol".

A data link protocol specifies the following:

- Frame format, i.e., locations and sizes of the various fields.
- Contents of various fields.
- Sequence of messages to be exchanged to carry out the error control, flow control, and data link management functions.

There are many data link protocols developed by various manufacturers and organizations. The frame formats and the contents of various fields are very specific to each protocol. Examples of data link protocols are:

- Binary Synchronous Data Link Control (BISYNC, BSC).
- Synchronous Data Link Control (SDLC)
- High- Level Data Link Control (HDLC)
- Advanced Data Communication Control Procedure (ADCCP)
- Point-to Point Protocol (PPP)

2.3 Frame Design Considerations

The first and foremost task of the data link layer is to format the user data as series of frames each having a predefined structure. A frame contains user data and control fields. Each frame is processed as one entity, for error control, i.e. if an error is detected the whole frame is retransmitted.

The format of a frame, in general, consists of three components, Header, Data, and Trailer, Figure 2.4.

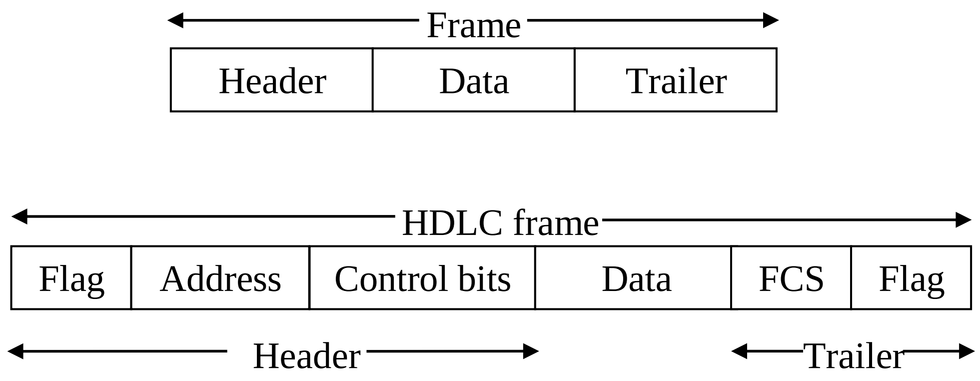


Figure 2.4 Data Link Layer Frame Format

The header and the trailer consists of one or more fields containing data link protocol control information. A typical example of the composition of a frame used for carrying user data in HDLC protocol is shown in Figure 2.4:

- Flag: Start of frame.
- Address: Source and destination addresses.

- Control: Sequence number, acknowledgement of the receipt frame, or other control information.
- Data: User data received from the network layer.
- FCS: Frame check sequence, contains CRC bits for error detection.
- Flag: End of frame.

There are two types of frame:

- Data frame: Which contains user data.
- Control frame: Which contains only the protocol control information, i.e. there no data user (g.e. acknowledgement).

2.3.1 Types of Frame Format

The frame format is so designed that the receiver is always able to locate the beginning of the frame and its various field, and is able to separate the data field. There are several types possibilities:

- Variable format-variable length,
- Fixed format-fixed length,
- Fixed format-variable length.

Variable format-variable length

All the field are optional and if a field is present, its size is variable. In all field, identifiers/delimiters are required, Figure 2.5a. X is the frame start identifier. The presence of each field is indicated by a field identifier which is also acts as delimiter for the previous field. As the size of frame is variable, an end delimiter Y is required to indicate the end of the frame.

Fixed format-fixed length

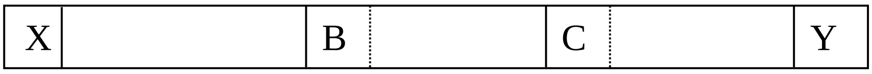
In this case the format of the frame is decided once for all and the frame sizes are also fixed in all the frames. The frame format is shown in Figure 2.5b, only one identifier is required at the beginning of the frame. On receipt of the identifier, the receiver is able to identify all the fields as the format of the frame and the sizes of the various fields are known to the receiver in advance.

Fixed format-variable length

In this case frame start and end identifier/delimiter are required, Figure 2.5c. The identifier of the first field is not required as the frame start identifier also identifies the field. As regards other fields, field delimiters are required to each field.

(a) Variable format-variable length

(b) Fixed format-fixed length



(c) Fixed format-variable length

Figure 2.5 Types of Frame Structures

2.3.2 Transparency

Transparency refers to providing a service to the users wherein no restriction is placed on the contents of the user data. Since any bit pattern can be sent by the user, problems may arise if the data field contains bit patterns similar to the field identifiers/delimiters. For example, if the data field of the HDLC frame shown in Figure 2.4 contains a bit pattern identical to the flag, the receiver may mistake it for the end flag of the frame. Therefore, the field identifiers and delimiters should not be present in any field apart from their predefined locations in the frame.

2.3.3 Bit-Oriented and Byte-Oriented Data Link Protocols

In a *bit-oriented data link protocol*, control information is coded at bit level and the length of the data field may not to be a multiple of bytes. Bit level implies that a control symbol need not to be full byte, (e.g. HDLC is bit-oriented protocol).

In a *byte-oriented data link protocol*, control information is coded at byte level, at least one byte long. The size of data field is also a multiple of bytes, (e.g. BISYNC is a byte-oriented protocol).

2.4 Error Control

Two types of errors can occur during transmission of frames from one device to the other:

- Content errors
- Flow integrity errors

Errors contained in the received frame are termed "content errors". "Flow integrity errors" refer to the lost or duplicated data frames and acknowledgments. Data link error control takes care of both the types of errors.

Content errors are detected using parity check or cyclic redundancy check (CRC) bits. The check bits are added as the trailer in the frame at the sending end. Their span of check usually over all the bits in the frame except the frame identifier, Figure 2.6. If the flag itself is corrupted the receiver will not recognize the arrival of the frame.

$\longleftarrow$ HDLC frame $\longrightarrow$

Flag	Address	Control bits	Data	FCS	Flag

$\longleftarrow$ Trailer $\longrightarrow$

$\longleftarrow$ Span of check $\longrightarrow$

Figure 2.6 Span of Error Check

The most common method of content error correction is retransmission of the frame. The receiver checks the received frame for possible transmission errors and then returns a short control message either to acknowledge its correct receipt or to request that another copy of the received frame is sent. This type of error control is known as "Automatic Repeat Request (ARQ)". It should be noted that, it is essential for the sending end to retains a copy of the transmitted frame until it is acknowledged by the receiver.

For flow integrity errors, the data link protocols specify the procedures to be adopted to detect and recover the missing /duplicate frames and acknowledgements. These procedures are built into the flow control mechanisms.

It must be remembered that no error method is 100% effective. There will always be some undetected content and flow integrity errors. Residual Error Rate (RER) refers to the error that still exist in the data stream after all the error control procedures have been completed.

There are two basic types of ARQ:

- Idle RQ (stop-and-wait), which is used with character oriented (or byte-oriented) data transmission schemes and,
- Continuous RQ, which employs either "a selective repeat" or "a go-back-N" retransmission strategy. Continuous RQ is used primarily with bit-oriented transmission schemes.

2.5 Error Control in Stop-and-Wait (Idle RQ) Mechanism

The stop-and-wait error control scheme is concerned with reliable transfer of data (Information) frames, (I- frame) between a sender (primary) and a receiver (secondary) across a serial data link.

The stop-and –wait scheme operates in a half-duplex mode. Since the primary after sending an I-frame, must wait until it receives and indication from the secondary as to whether the frame was correctly received or not. The primary then either sands the next frame, if the previous frame was correctly received, or retransmits a copy of the previous frame if it was not.

There two ways of implementing stop-and –wait mechanism:

- *Implicit retransmission*: In implicit retransmission, the receiver (secondary-S) acknowledges only correctly received frames and the sender (primary-P) interprets the absence of an acknowledgment as an indication that the previous frame was corrupted.
- *Explicit retransmission*: In explicit retransmission when the receiver (secondary-S) detects that a frame has been corrupted, it returns a negative acknowledgment to request that another copy of the frame is transmitted, explicit request.

2.5.1 Implicit Retransmission

Some example frame sequences with the implicit retransmission control scheme are shown in Figure 2.7. The following points should be noted when interpreting the frame sequences:

- P can have only one I-fame outstanding at a time (awaiting and acknowledgment frame or ACK-frame) at a time.
- On receipt of an error-free I-frame. S returns on ACK-frame to P.
- On receipt of an error-free ACK frame, P can transmit another I-frame – part (a).
- When P initiates the transmission of an I-frame, it starts a timer.
- If S receives an I-frame or P receives an ACK-frame containing transmission error, the frame is discarded.
- If P does not receive an ACK-frame within a predefined time interval (the timeout interval). Then P retransmits the waiting I-frame-part (b).
- If an ACK-frame is corrupted, then S receives anther copy of the frame (duplication case) and hence this is rejected by S-part (c).

As we can see in part (a) after initiating the transmission of a frame, P must wait a minimum time before transmitting the next frame. The wait time is equal to the time the I-frame, takes to be received and processed by S plus the time for the ACK-frame to be transmitted and processed. In the worst case, P must wait a time equal to the time out interval, which must exceed the minimum time by a suitable margin to avoid an ACK-frame being received after another copy of the previous frame has been retransmitted.

The relative magnitude of each component making up the minimum time varies from different types of data link. It is determined by such factors as the physical separation of the two communication systems (P and S) and the data transmission rate of the link. In general however, a significant improvement in the vitalization of the available link capacity can be obtained if S informs P immediately whenever it receives a corrupted I-frame, by returning a negative acknowledgment frame (NAK-frame).This is called Explicit retransmission”.

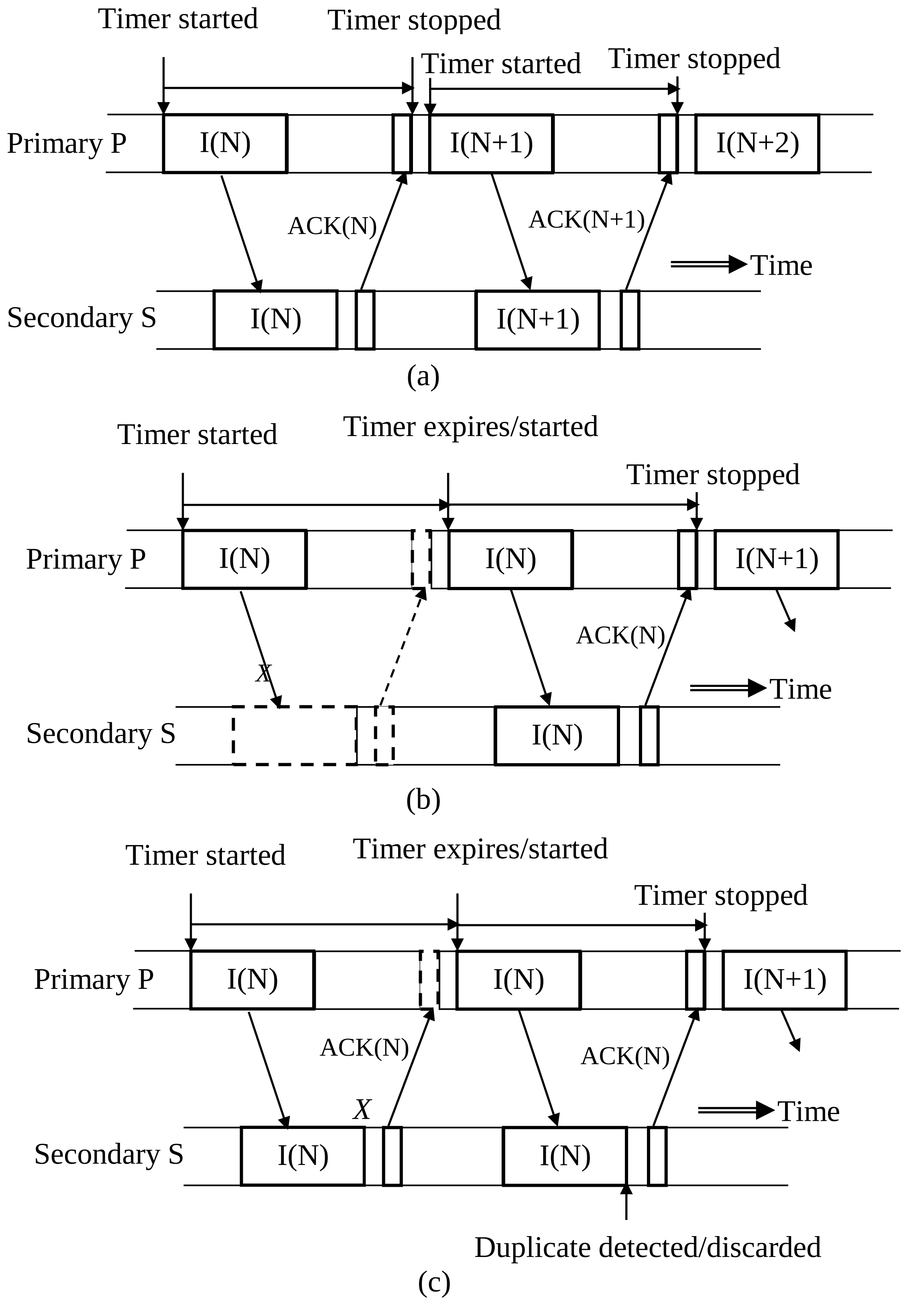
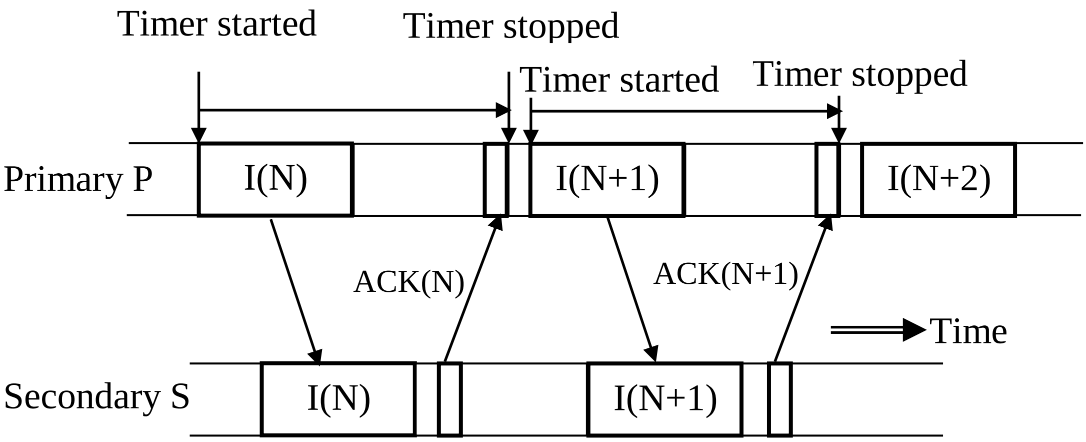


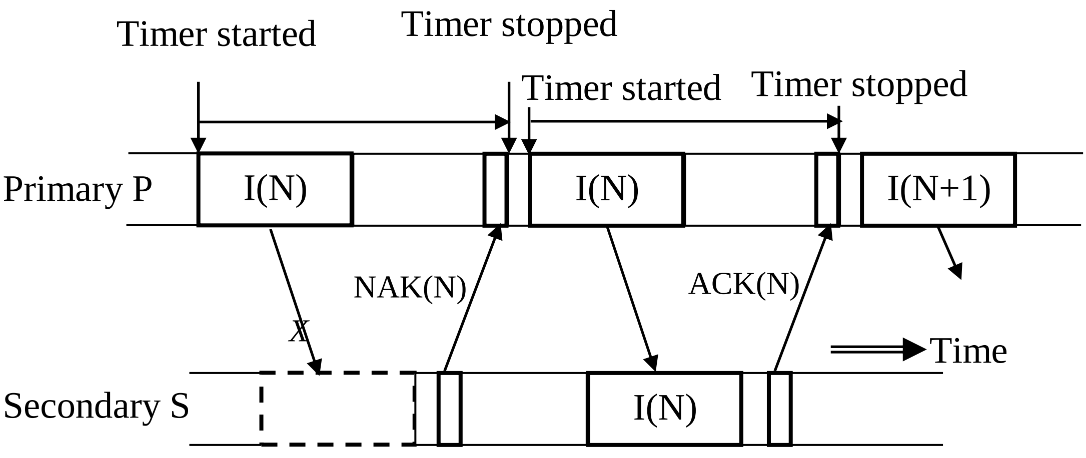
Figure 2.7 Implicit Retransmission

2.5.2 Explicit Retransmission

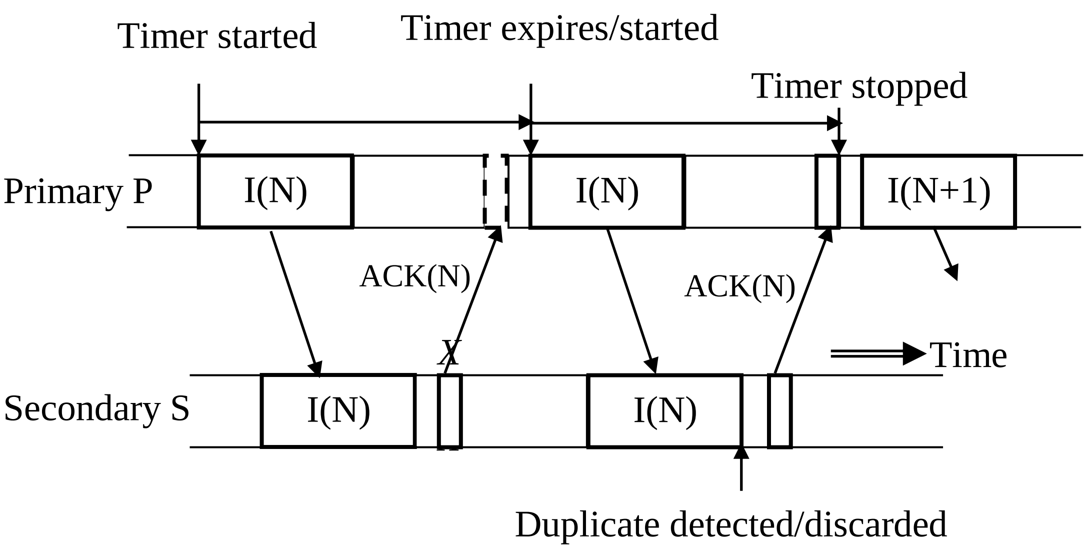
Some example frame sequence diagrams with this scheme are shown in Figure 2.8.



(a)



(b)



(c)

Figure 2.8 Explicit Retransmission

The following points should be noted when interpreting the frame sequences:

- As with the implicit acknowledgment scheme, on receipt of an error-free I-frame, S returns an ACK –frame to P.
- On receipt of an error-free ACK-frame, P stops the timer and can then initiate the transmission of another I-frame-part(a).
- If S receives an I-frame-containing transmission errors, the frame is discarded and it returns a NAK-frame-part (b).
- If P does not receive an ACK-frame or (NAK-frame) within the timeout interval, P retransmits the waiting I-frame- part (c).

Since with Idle RQ scheme the primary (P) must wait for an acknowledgment after sending a frame, the scheme is also known as "stop – and-wait" or "send – and-wait". As we can see from Figures 2.7 and 2.8 the scheme ensures that S receives at least one copy of each frame transmitted by P. With both schemes, however, it is possible for S to receive two (or more) copies of particular I-frame. These copies are known as "duplicates". In order for S to discriminate between the next valid I-frame (as it expects) and a duplicate, each frame transmitted contains a unique identifier known as the "sequence number" (N, N+1, etc). The receiver (S) must therefore retain a record of the sequence number contained within the last I-frame it correctly received. If (S) receives another copy of this frame then the copy is discarded. To enable (P) to resynchronize, (S) returns and ACK –frame for each correctly received frame with the related I-frame identifier within it.

We can observe the improvement in link utilization by using an "Explicit" request scheme by considering the frame sequences in part (b) of each scheme. With implicit retransmission, the time before the next I-frame can be transmitted is the timeout interval, whereas the time is much shorter with explicit retransmission, where a NAK-frame is used.

The relative improvement in link utilization is determined ultimately by bit error rate (BER) of the link and hence the number of frames that are corrupted and need to be retransmitted.

2.5.3 Link Utilization

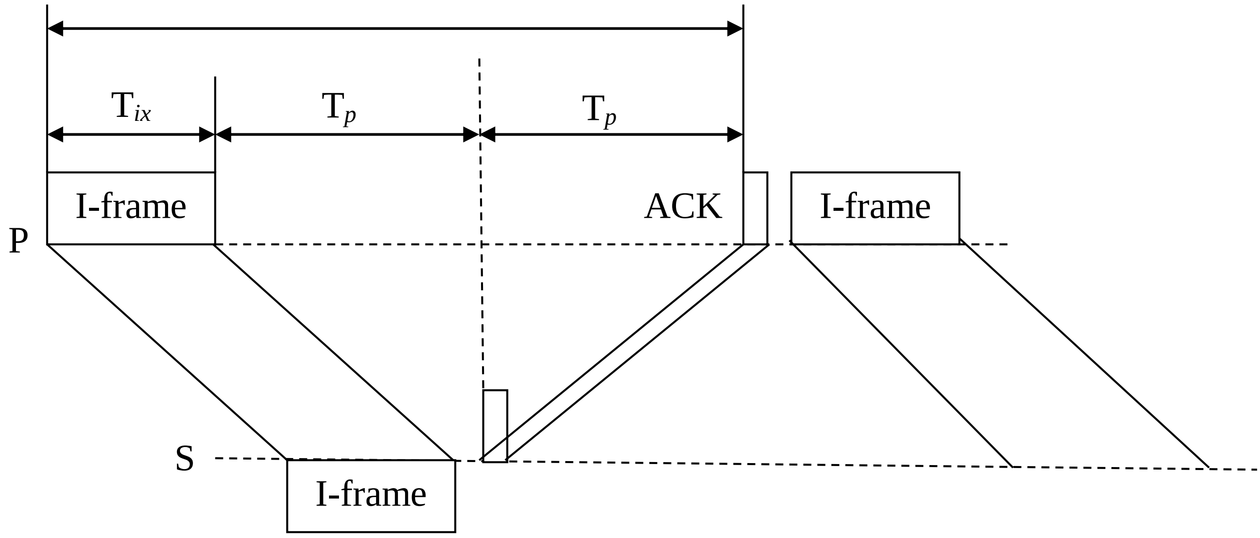
- *Link utilization without transmission errors*

In the stop- and –wait mechanism (Implicit/Explicit), the efficiency of utilization (U) in the case of error free transmission medium, is given by:

$$U = \frac{T_{ix}}{T_t}$$

Where T_{ix}: is the transmission time of I-frame, and

T_t: is T_{ix} plus any time the transmitter spends waiting for an acknowledgment. See the figure below.



$T_t = T_{ix} + 2T_p$

Where T_p is the propagation delay between P and S

$$U = \frac{T_{ix}}{T_{ix} + 2T_p}, or$$

$$U = \frac{1}{1 + 2T_p / T_{ix}}$$

$$U = \frac{1}{1 + 2a}$$

Where, $a = T_P/T_{ix}$

Note that "a" can range from a small fraction for low-bit transmission rate of modest length to a large value for long links and high bit transmission rate. For these two extremes, U varies between near unity (100%) and small fraction.

Example:

A series of 1000-bit frames is to be transmitted using an idle RQ protocol. Determine the link utilization for the following types of data link assuming a data transmission rate of: (i) 1 Kbps and (ii) 1 Mbps.The velocity of propagation of the link is 2 x 10^8 m/s and the bit error rate is negligible.

(a) A twisted pair cable 1 km in length

(b) A leased line 200 km in length

(c) A satellite link of 50000km

Solution:

The time taken to transmit a frame, T_{ix} , is given by:

$$T_{ix} = \frac{\text{Number of bit in frame, N}}{\text{Bit transmission rate in bps}}$$

At 1 kbps: $\quad Tix = \frac{1000}{10^3} = 1sec$

At 1 Mbs: $T_{ix} = \dfrac{1000}{10^6} = 0.001$ sec

Propagation delay (T_p) = $\dfrac{\text{distance}}{\text{Propagation speed}} = \dfrac{\text{distance}}{2\text{x}10^8}$ sec

Utilization:

$$U = \frac{1}{1+2T_p/T_{ix}} = \frac{1}{1+2a}, \qquad a = T_p/T_{ix}$$

(a) $T_p = \dfrac{10^3}{2\times10^8} = 5\times10^{-6}\,\text{sec}$

(i) $a = \dfrac{5\times10^{-6}}{1} = 5\times10^{-6}$ and hence (1+2a) ≈ 1 and U = 1 (100%)

(ii) $a = \dfrac{5\times10^{-6}}{10^{-3}} = 5\times10^{-3}$ and hence (1+2a) ≈ 1 and U = 1 (100%)

(b) $T_p = \dfrac{200\times10^3}{2\times10^8} = 1\times10^{-3}\,\text{sec}$

(i) $a = \dfrac{1\times10^{-3}}{1} = 1\times10^{-3}$ and hence (1+2a) ≈ 1 and U = 1 (100%)

(ii) $a = \dfrac{1\times10^{-3}}{10^{-3}} = 1$ and hence (1+2a) >1 and U = $\dfrac{1}{1+2} = 0.33$ (33%)

(c) $T_p = \dfrac{50\times10^6}{2\times10^8} = 0.25\,\text{sec}$

(i) $a = \dfrac{0.25}{1} = 0.25$ and hence (1+2a) >1 and U = $\dfrac{1}{1+0.5} = 0.67$ (67%)

(ii) $a = \dfrac{0.25}{10^{-3}} = 250$ and hence (1+2a) >1 and U = $\dfrac{1}{1+500} = 0.002$ (0.2%)

From the above example, we note the following:

- For relatively short links for which a<1, the link utilization is ≈ 100% and is Independent of the transmission rates of data. This means that an idle RQ protocol is perfectly adequate for short links and modest data rates. Examples are networks based modems and the analog PSTN.
- For longer terrestrial links, the link utilization is high for low data rates (and hence low values of "a"), but falls of significantly as the data rate (and hence "a") increases.
- The link utilization is poor for satellite links, even at low rate.

We can conclude that an idle RQ protocol is unsuitable for satellite applications and also for those that involve high bit rate terrestrial links, such as LANS, and most public carrier WANS.

- *Link utilization in presence of Errors*

In practice, the links have a nonzero bit error rate. Hence to transmit a frame successfully, an average N_r transmission attempts will be required.

- Average number of transmissions (N_r)

  If P is the probability that a bit is corrupted by a random error , and L is the number of bits in a frame. Hence, the probability that a frame is received without any error is $(1- P)^L$. Thus the probability P_f that a frame is received with one or more errors is given by:

  $$P_f = 1-(1-P)^L$$

  The probability of transmitting a frame successfully in the i^{th} attempt is

  $$P_f^{i-1}(1-P_f)$$

  We can calculate the average number of transmission (N_r) as:

  $$N_r = \sum_{i=0}^{\infty} i\, P_f^{i-1}(1-P_f)$$

  $$N_r = (1-P_f)\sum_{i=0}^{\infty} i\, P_f^{i-1}$$

  $$N_r = (1-P_f)\sum_{i=0}^{\infty} \frac{d}{dP_f} P_f^{i}$$

  $$N_r = (1-P_f)\frac{d}{dP_f}\sum_{i=0}^{\infty} P_f^{i}$$

  $$N_r = (1-P_f)\frac{d}{dP_f}\left(\frac{1}{1-P_f}\right) = (1-P_f)\frac{1}{(1-P_f)^2}$$

  $$N_r = \frac{1}{1-P_f} = \frac{1}{1-\left[1-(1-P)^L\right]} = \frac{1}{(1-P)^L}$$

  Where P_f the probability of receiving a frame in error, P is the probability of receiving a bit in error, and L is length of frame.

- *Link utilization in idle RQ(implicit retransmission)*

If the average number of transmission required for sending one data frame correctly is N_r, and time out interval is T_o. The total time (T_t) needed to receive successful frame at N_r th attempt is given by:

$$T_t = (N_r - 1)T_o + T_{ix} + 2T_p \qquad \text{(refer to fig. 2.7.b)}$$

$\therefore$

$$U = \frac{T_{ix}}{(N_r - 1)T_o + T_{ix} + 2T_p} = \frac{1 - P_f}{(1+2a)(1-P_f) + P_f\,(\frac{T_o}{T_{ix}})}$$

- *Link utilization in idle RQ (explicit retransmission)*

In this case the total needed time (T_t) to receive successful frame at the N_r^{th} attempt is given by:

$$T_t = N_r(T_{ix} + 2T_p)$$

Then

$$U = \frac{T_{ix}}{N_r(T_{ix} + 2T_p)} = \frac{1 - P_f}{1 + 2a}$$

Example: calculate the link utilization for stop-and –wait protocol (Explicit retransmission) If:

- Bit error rate(BER)= 10^{-4}
- Propagation delay (T_p) = 40 ms
- Transmission rate (R) = 1 mbps
- Frame length (L) =1000 bytes

$$T_{ix} = \frac{L}{R} = \frac{1000 \times 8}{10^6} = 8 \times 10^{-3} \text{ sec.}$$

$$a = \frac{T_p}{T_{ix}} = \frac{40 \times 10^{-3}}{8 \times 10^{-3}} = 5$$

$$P = 10^{-4}$$

$$P_f = 1 - (1-P)^L = 1 - (1 - 10^{-4})^{8000} = 0.550689$$

$$U = \frac{1 - P_f}{1 + 2a} = \frac{1 - 0.550689}{1 + (2 \times 5)} = 0.04 = 4\%$$

The major advantage of the idle RQ scheme is that it requires a minimum buffer storage for its implementation since both P and S need contain sufficient storage for only one frame. In addition, S must retain only a record of the identifier of the last correctly received frame to enable it to detects duplicates.

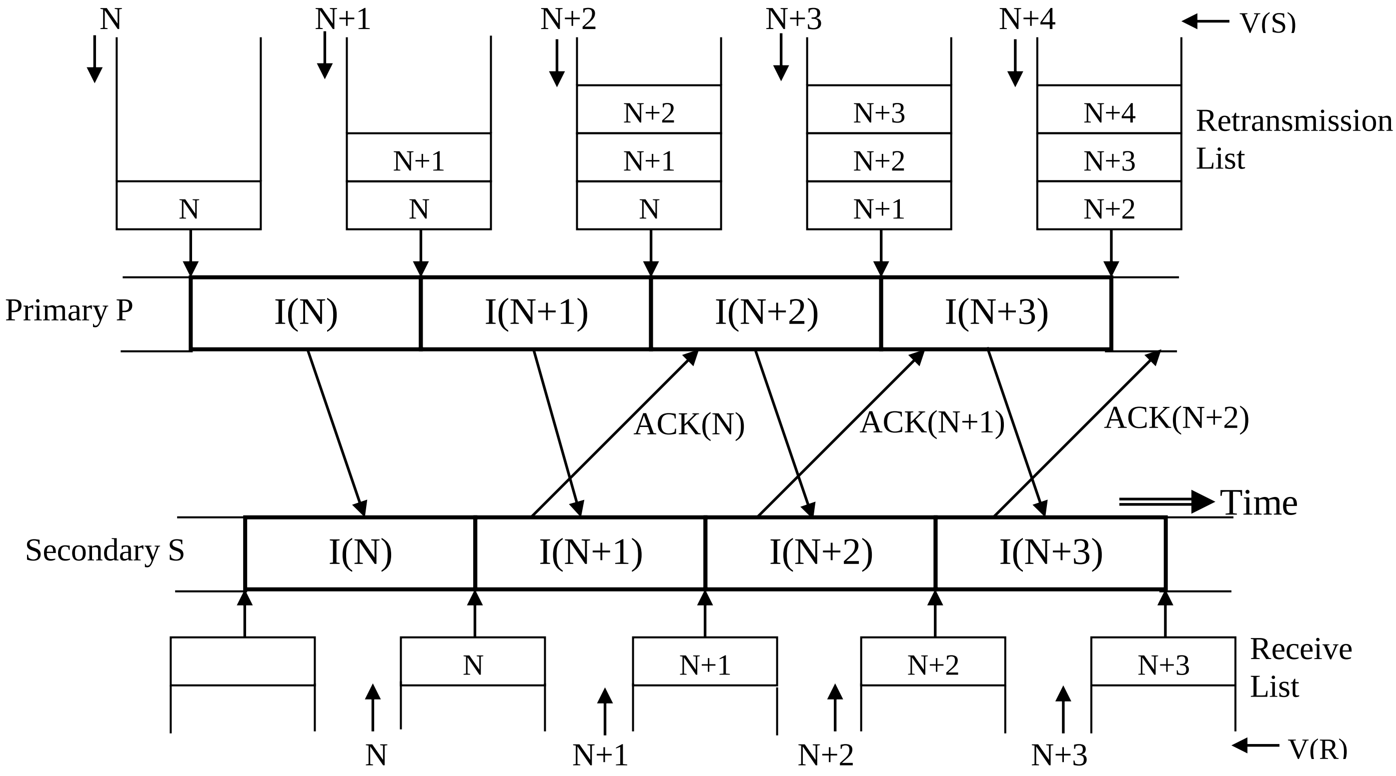
2.6 Error Control in Continuous RQ

With a continuous RQ error control scheme, link utilization is much improved at the expense of increased buffer storage requirements. A duplex link is required for continuous RQ error control scheme implementation.

An example illustrating the transmission of a sequence of I-frames and their returned ACK- frames (error free-case) is shown in Figure 2.9.

Note the following points when interpreting the operation of the scheme:

- P sends I-frames continuously without waiting for an ACK to be returned.
- Since more than one I-frame is waiting acknowledgement, P retains a copy of each I-frame transmitted in a retransmission list that operated on a FIFO Queue discipline.
- S returns an ACK –frame for each correctly received I-frame.
- Each I-frame contains a unique identifier which is returned in the corresponding ACK –frame.
- On receipt of an ACK -frame, the corresponding I-frame is removed from the retransmission list by P.
- Frames received free of errors are placed in the link receiver list to wait processing.



V(S) : Send sequence variable
V(R) : Receive sequence variable

Figure 2.9 Continuous RQ Frame Sequence

- On receipt of the next in-sequence I-frame expected. S delivers the information content within the frame to the upper (DL-SDU) layer immediately it has processed the frame.
- In the event of frames being received out of sequence. S retains these in the link receiver list until the next in sequence frame is received.

To implement the scheme. P must retain a send sequence variable V(S) which indicates the send sequence number N(S) to be allocated to the next I-frame to be transmitted. Also, S must retain a receive sequence a variable V(R), which indicates the next in-sequence I-frame it is a waiting.

Figure 2.9 assumed that no transmission errors occur. When an error does occur, one of two retransmission strategies may be followed:

- S detects and requests the retransmission of just those frames in the sequence that are corrupted. This scheme is called "selective repeat"
- S detects the receipt of an out-of-sequence I-frame and requests P to retransmit all out-standing unacknowledged I–frames from the last correctly received, and hence acknowledged. I–frame. This scheme is called "Go-Back-N".

2.6.1 Selective Repeat

Selective repeat can be implemented in one of the following two ways

- Implicit retransmission, and
- Explicit retransmission.

In both schemes, in the event of frames being received out of sequence, S retains these in the link receive list until the next in-sequence frame is received.

Selective repeat in implicit retransmission

Two frame sequence diagrams are shown in Figure 2.10. Part (a) assumes all ACK–frames are received correctly, While part (b) shows the effect of a corrupted ACK–frame.

To follow the sequence in Figure 2.10a, note the following:

- Assume I–frame N+1 is corrupted
- S returns an AC –frame for each correctly received I–frame
- S returns an ACK- frames for I–frames N, N+2, N+3, …
- On receipt of the ACK for I–frame N+2, P detects that N+1 has not been acknowledged
- To allow for the possibility of more than one I–frame being corrupted, on detecting an unacknowledged frame P enters the retransmission state.
- When in this state, the transmission of new frames is suspended until all unacknowledged frame have been retransmitted.

- P removes I–frame N+2 from the retransmission list and retransmits I – frame N+1 before transmitting frame N+5
- On receipt of I–frame N+1, the contents of the queued frames in the link receive list are delivered by S to the upper (DL–SDU) layer in the correct sequence.

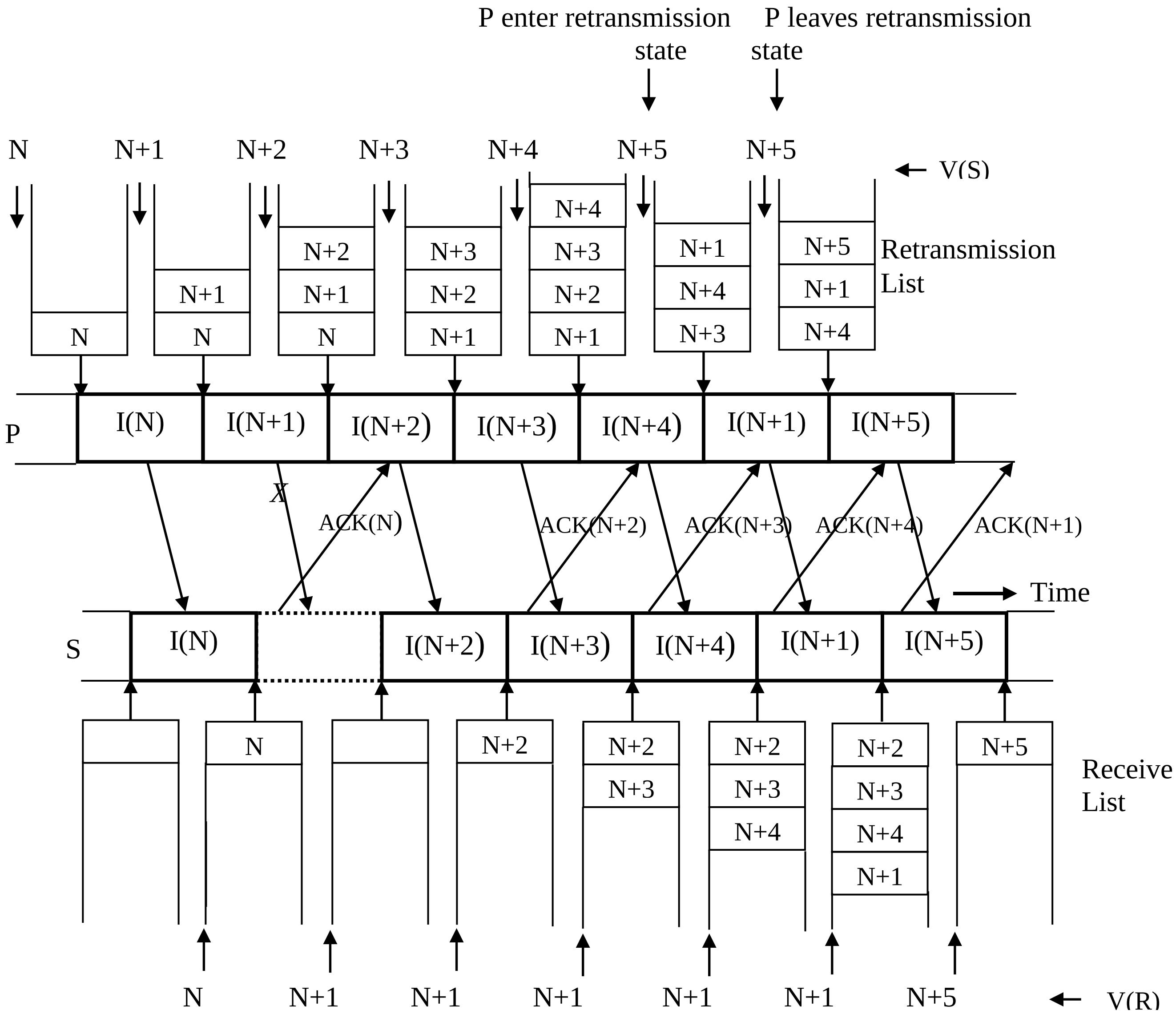


Figure 2.10a Selective Repeat-Implicit Retransmission
Corrupted I-frame

The frame sequence shown in Figure 2.10b relates to the situation in which all the I–frames are received correctly but ACK frame N is corrupted. Note that:

- On receipt of ACK–frame N+1, P detects that I–frame N is still a waiting acknowledgment and hence retransmits it
- On receipt of the retransmitted I–frame N, S determines from its receive sequence variable that this has already been received correctly and is therefore duplicate

- S discards the frame but returns an ACK–frame for it to ensure P removes the frame from the retransmission list.

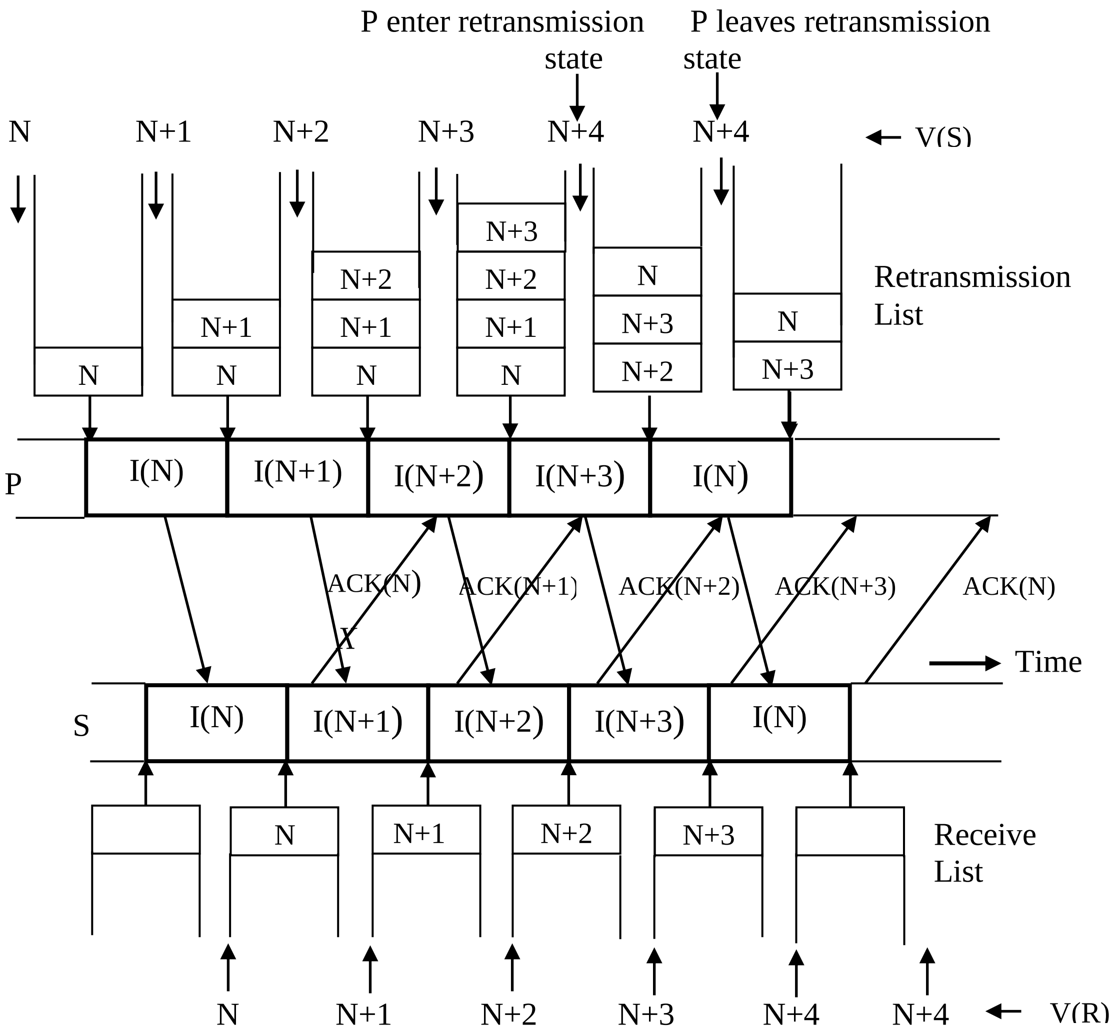


Figure 2.10b Selective Repeat-Implicit Retransmission ACK-Corrupted

Selective repeat in explicit retransmission

The alternative approach is to use an explicit negative acknowledgment frame to request for a specific frame to be retransmitted. The negative acknowledgment is known as selective reject. Again the sequence shown in Figure 2.11a assume no acknowledgments are corrupted. With the sequence in Figure 2.11b shows the effect of a lost acknowledgement. Note that:

- An ACK – frame acknowledges all frames in the retransmission list up to and including the I–frame with the sequence number the ACK contains.
- Assume I–frame N+1 is corrupted.
- S returns an ACK – frame for I-frame N.

- When S receives I–frame N+2, it detects I– frame N+1 is missing and hence returns a NAK –frame containing the identifier of the missing I –frame N+1
- On receipt of NAK N+1, P interprets this as S is still awaiting I–frame N+1 and hence retransmits it.
- When S returns a NAK –frame, it enters the retransmission state.
- When S in the retransmission state, the return of ACK –frames is suspended.
- On receipt of I–frame N+1, S leaves the retransmission state and resumes returning ACK frames
- ACK N+4 acknowledges all frames up to and including frame N+4,
- A timer is used with each NAK –frame to ensure that if it is corrupted (and hence frame N+1 is not received) it is retransmitted.

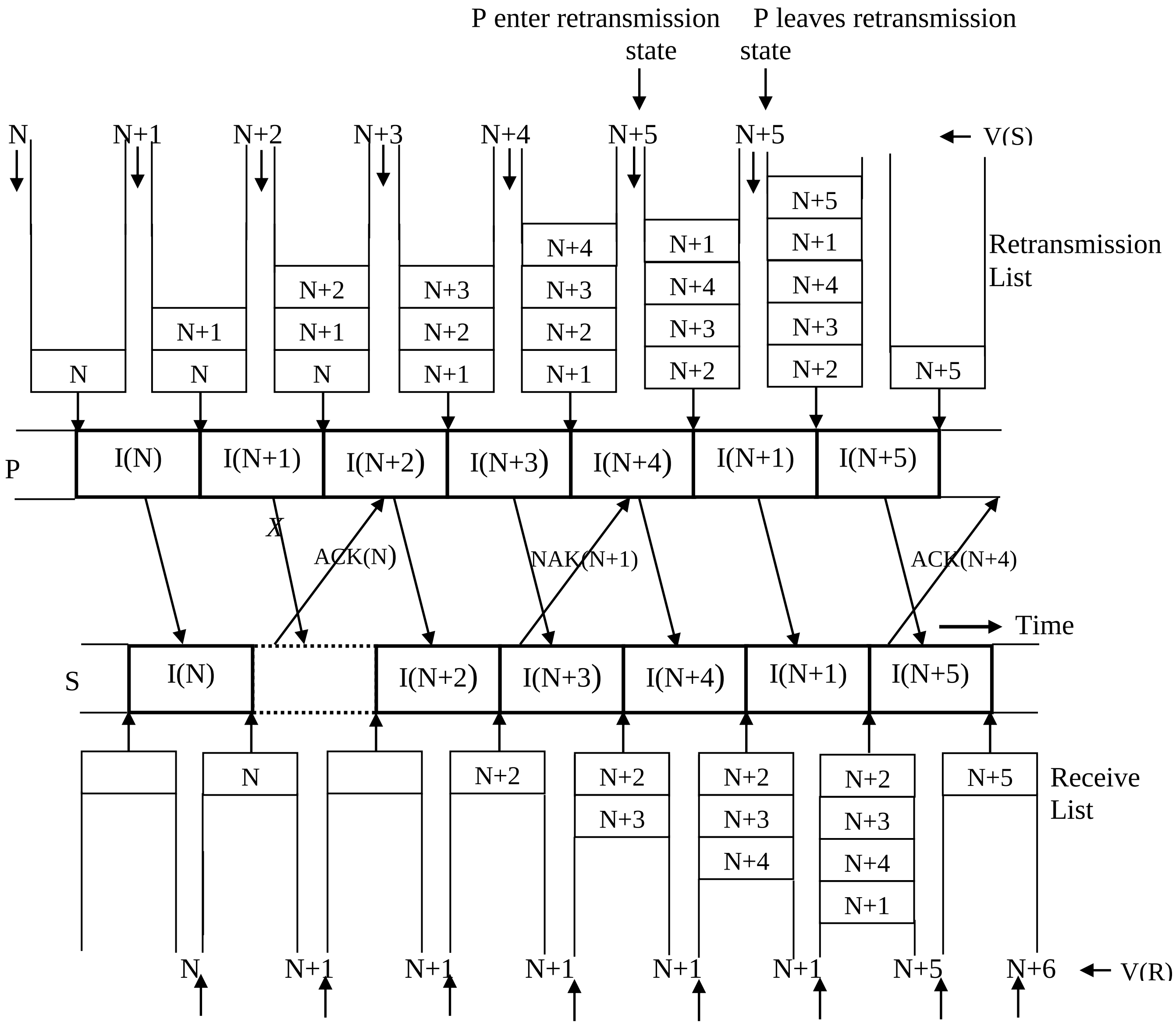


Figure 2.11a Selective Repeat-Explicit Retransmission Corrupted I-Frame

To understand why only one NAK –frame can be outstanding at a time. Consider the frame sequence diagram – shown in fig 2.11.b.

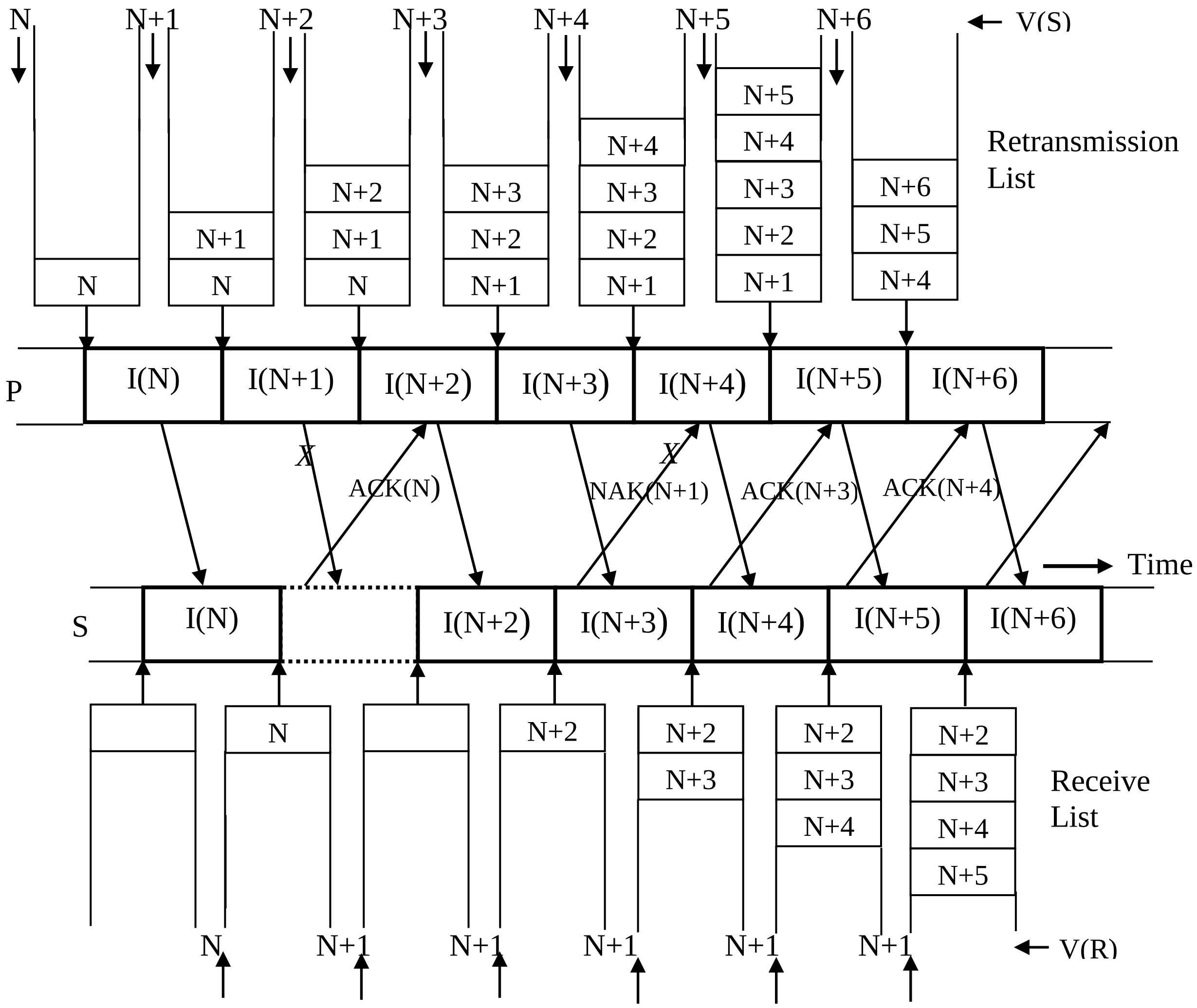


Figure 2.11b Selective Repeat-Explicit Retransmission
Corrupted I-Frame and NAK-Frame

Note that:

- Assume I –frame (N+1) is again corrupted.
- S returns NAK (N+1) as before but this time it is corrupted.
- On receipt of ACK–frame (N+3), this would acknowledge all frames including I–frame (N+1) and hence his frame will not be retransmitted.

We can deduce from the frame sequences shown that although (S) receives a correct copy of each frame sent by (P), the order of reception is not maintained.

In many applications, however, frames must be delivered on the same sequence as they were submitted. Hence frames received out of sequence must be buffered (Held) by (S) until the missing frame (s) is (are) received. Since the frames can often be large, and the number of frame buffers required can also become large, this makes the buffer storage capacity required in the communications subsystem unacceptably high. For this reason most applications of the type outlined, as well as most terrestrial networks, use the Go – back – N retransmission control scheme.

2.6.2 Go – Back – N

With go back N, as the name implies, when the secondary (receiver) detects an out – of – sequence frame, it informs the primary (transmitter) to start to retransmit frames from a specified frame number, Figure 2.12 shows the operation of Go – back –N scheme.

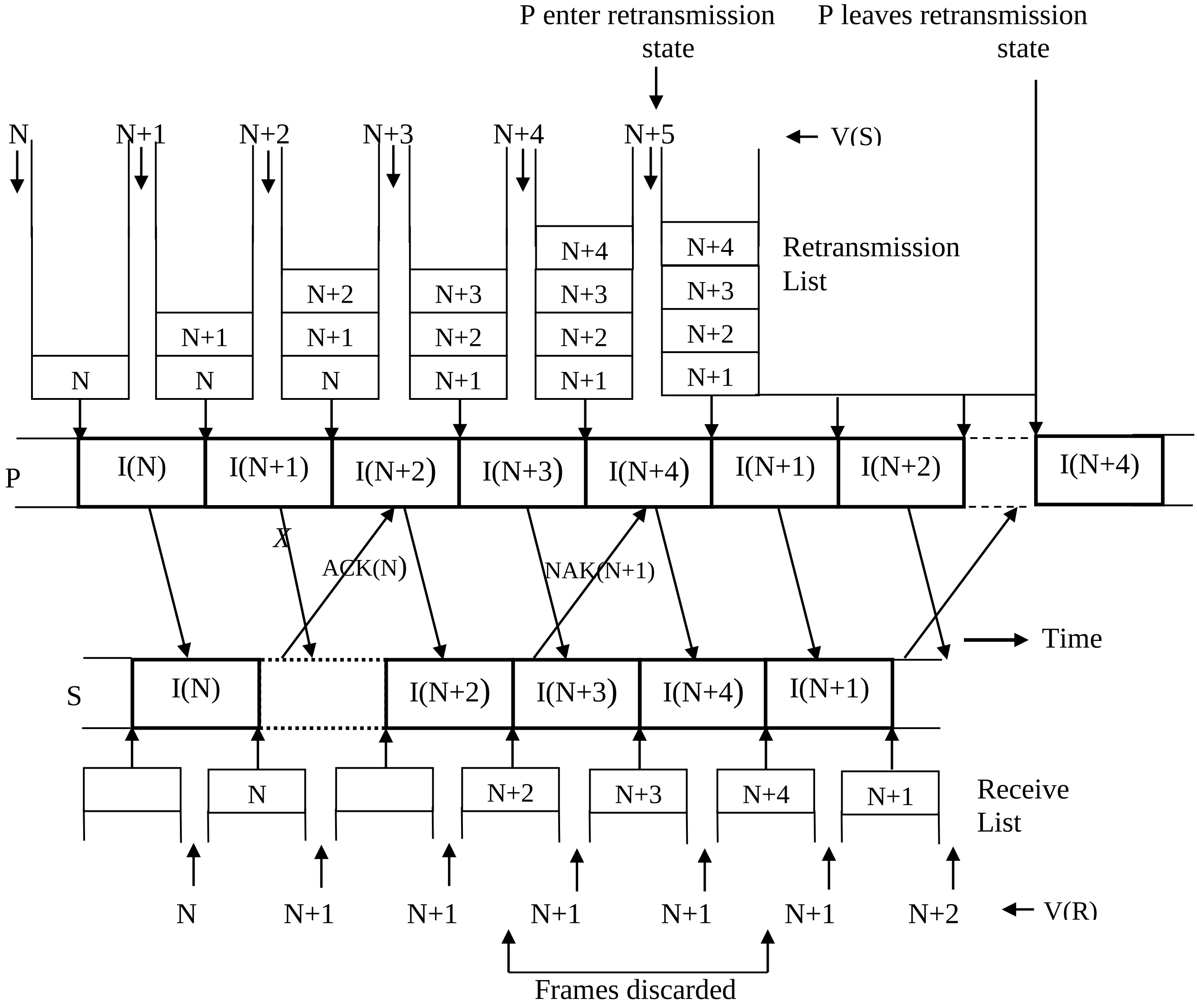


Figure 2.12a Go-back-N retransmission
Corrupted I-Frame

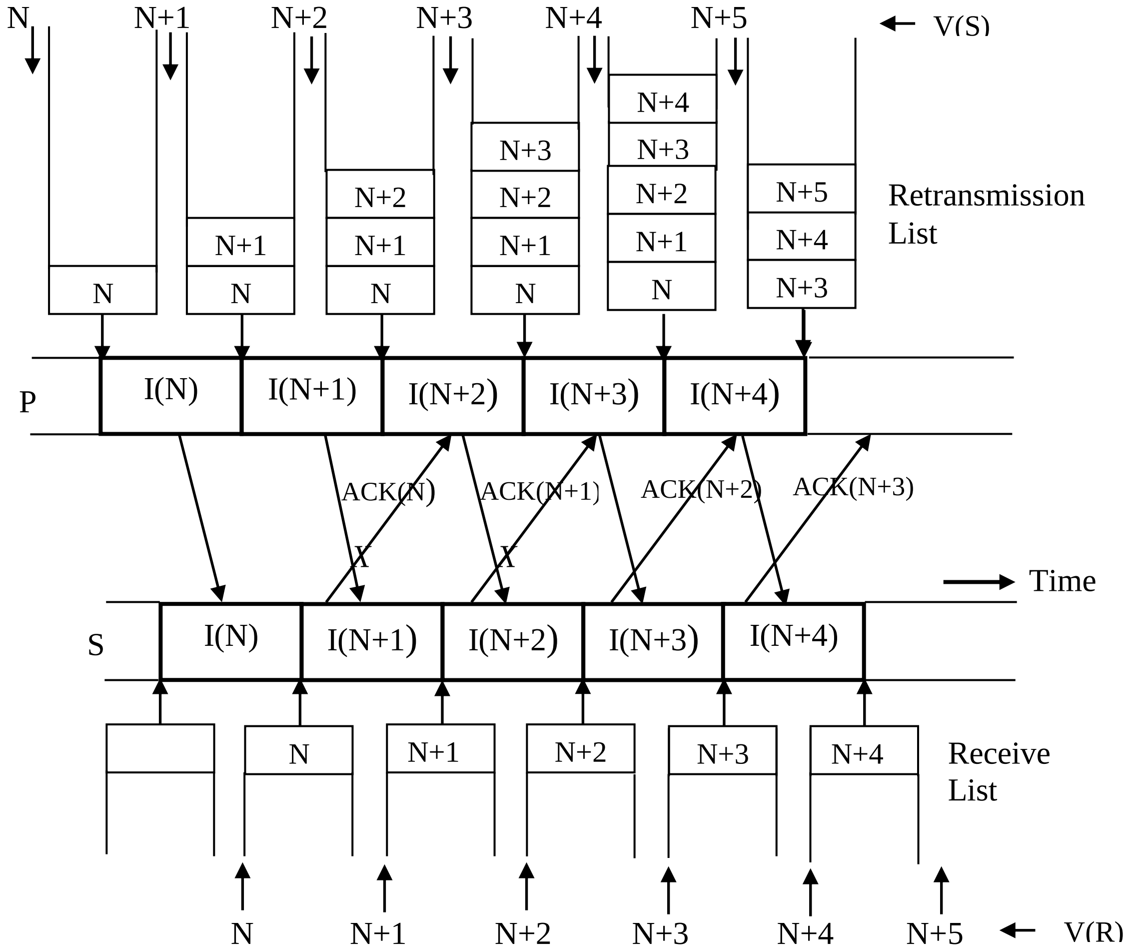


Figure 2.12b Go-back-N Retransmission
Corrupted ACK-Frame

- Assume I –frame N+1 is corrupted.
- S receives I–frame N+2 out of sequence.
- On receipt of I–frame N+2, S returns NAK N+1 informing P to go back and start to retransmit from I–frame N+1.
- On receipt of NAK N+1, P enters the retransmission state.
- When in this state, it suspends sending new frames and commences to retransmit the frames waiting acknowledgment in the retransmission list.
- S discards frames until it receives I–frame N+1.
- On receipt of I–frame N+1, S resumes accepting frames and returning acknowledgments.
- A time out is applied to NAK frames by S and a second NAK is returned if the correct in – sequence I–frame is not received in the time out interval.

Again, we have assumed in Figure 2.12a that an I–frame was corrupted and that the acknowledgment frames were received correctly. The effect of a corrupted acknowledgment on the frame transmission sequence is shown in Figure 2.12b. Note that:

- S receives each transmitted I –frame correctly,
- Assume ACK–frames N and N+1 are both corrupted,
- On receipt of ACK frame N+2, P detects that there are two outstanding I–frames in the retransmission list (N and N+1),
- Since it is an ACK–frame rather than a NAK–frame, P assumes that the two ACK–frames for I–frames N and N+1 have both been corrupted and hence accepts ACK-frame N+2 also an acknowledgment for the outstanding frames.

Note that, with a Go-back-N strategy, the correct frame sequence is maintained, thus minimizing the buffer storage required for its implementation. However, since some correctly received frames must be retransmitted, it is less efficient in its use of the available transmission capacity than a selective retransmission scheme.

2.7 Flow Control

Error control is only one component of a data link protocol. Another important and related component is "Flow control". As the name implies, it is concerned with controlling the rate of transmission of I –frames on the link so that the receiver always has sufficient buffer storage resources to accept them prior to processing. For example, with a frame – oriented selective retransmission scheme, the receiver may run out of buffer storage capacity if it is trying to buffer an indeterminate number of frames. In the following we shall consider the most common flow control, "sliding window".

2.7.1 Sliding Window Flow Control

To control the flow of frames across a data link a sliding window mechanism is used. Recall that an idle RQ error control scheme, although inefficient in its use of transmission bandwidth, requires a minimum of buffer storage capacity. Since, after a frame is transmitted by P, it must wait until an acknowledgment is returned by S before transmitting the next frame. The flow of I–frames across the link is therefore automatically tightly controlled.

With a continuous RQ error control scheme, however, P may send I–frames continuously before receiving any acknowledgments. With this type of scheme it is possible for the destination to run out of available buffer storage. If, for example, it is unable to pass on the frames at the rate they are received. It is usual to introduce an additional regulating action into such schemes. The approach is similar to the idle RQ control scheme in that it essentially sets a limit on the number of I–frames that P may send before receiving an acknowledgment. P monitors the number of outstanding (unacknowledgment) I –frames currently held in the retransmission list. If the destination side of the link is unable to pass on the frames sent to it, S stops returning acknowledgment frames, the retransmission list at P builds up and this in turn can be interpreted as a signal for P to stop transmitting further frames until acknowledgment stat to flow again.

To implement this scheme, a maximum limit is set on the number of I-frames that can be awaiting acknowledgment and hence outstanding in the retransmission list. This limit is the “send window, K” for the link. The limit (K-size) is normally selected so that, providing the destination is able to pass on or absorb all frames it receives, the send window does not impair the flow of I-frames across the link. Factors such as the maximum frame size, available buffer storage, link propagation delay, and transmission bit rate must all be considered when selecting the send window size. Operation of the scheme is shown in Figure 2.13.

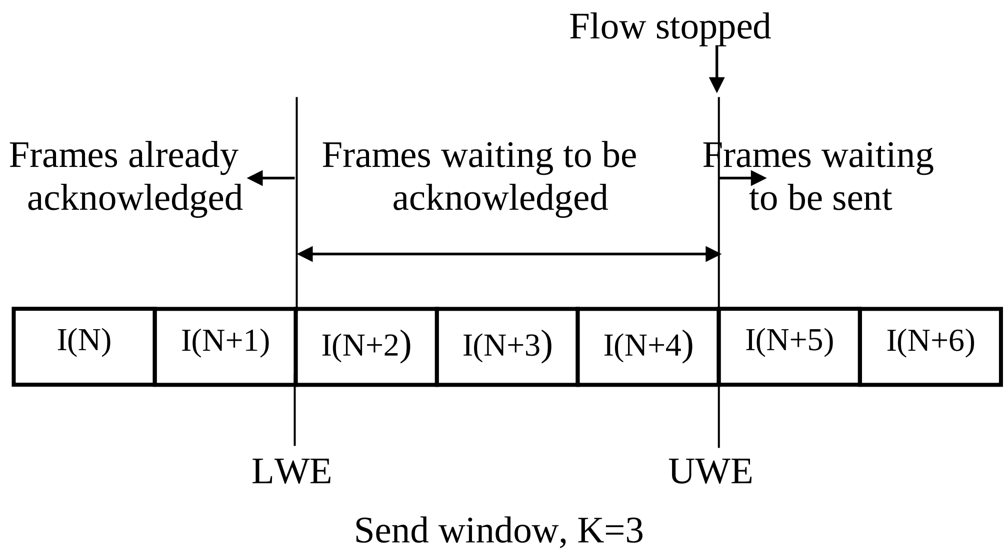


Figure 2.13 Sliding Window Example

As each I-frame is transmitted, the upper window edge (UWE) is incremented by unity. Similarly as each I-frame is acknowledged, the lower window edge (LWE) is incremented by unity. The acceptance of any new frames, and hence the flow of I-frames, is stopped if the difference between UWE and LWE becomes equal to the send window K. Assuming error – free transmission, K is a fixed window that moves (slides) over the complete set of frames being transmitted. The technique is thus known as “sliding window”.

The maximum number of frame buffers required at S is known as the receive window. We can deduce from the earlier frame sequence diagrams that in idle RQ and Go-back-N schemes only one buffer is required. With selective repeat, K frames are required. See Table 2.1.

Table 2.1 Send and Receive Window Limits

Protocol	Send window	Receive window
Idle RQ	1	1
Selective repeat	K	K
Go-back-N	K	1

2.7.2 Link Utilization (Error Free)

Unlike the stop and wait mechanism, in the sliding window flow control, each I-frame is not individually acknowledged, and therefore, the sending end (primary) can send a number of I-frames one after the other without waiting for acknowledgement, which results in better link utilization. To calculate link utilization, let us consider the following two possible situations

- The sender receives an acknowledgement before it exhausts the window.
- The sender exhausts the window before it receives an acknowledgement.

In the first situation, the sender can keep transmitting I-frames without interruption. The link utilization is unity (100%). If the window size is K, this situation will occur when the time required to transmit K I-frames is more than the earliest possible arrival of an acknowledgement, which is $T_{ix} + 2T_p$. It is assumed that the size of acknowledgement is very small and the receiver responds with acknowledgement immediately on receipt of an I-frame.

$$KT_{ix} \geq T_{ix} + 2T_p$$

or

$$K \geq 1 + 2a \quad , \text{ where } a = \frac{T_p}{T_{ix}}$$

In the second situation, the sender transmits K I- frames in time K Tix and then suspends further transmission of the I frames until an acknowledgement is received. If we assume that the receiver sends the acknowledgement at the earliest opportunity, i.e. immediately following the receipt of the first frame, the acknowledgement will be received after time $T_{ix} + 2T_p$. Therefore, the second situation will occur when

$$KT_{ix} < T_{ix} + 2T_p$$

or

$$K < 1 + 2a \quad , \text{ where } a = \frac{T_p}{T_{ix}}$$

In this case the link has been engaged for time $T_{ix} + 2\,T_p$ while ($a = T_p/T_{ix}$) has utilized it for time KT_{ix} only. Therefore, link utilization is given by:

$$U = \frac{KT_{ix}}{T_{ix} + 2a} = \frac{K}{1 + 2a}$$

Example: a series of 1000 –bit I- frames is to be transmitted using a continuous RQ protocol. Determine the link efficiency for the following types of data link if the velocity of propagation is $2x10^8$ m/s and the bit error rates of the links are all negligibly.

a. A 1 km link of 1 Mbps and a send window K = 2
b. A 10 km link of 200 Mbps and a send window K=7
c. A 50000 km satellite link of 2 Mbps and a send window K= 127

Solution

$$T_p = \frac{\text{distance}}{\text{propagation speed}} \quad , \quad T_{ix} = \frac{\text{I} - \text{frame length (in bits)}}{\text{transmission rate}} \quad , \quad a = \frac{T_p}{T_{ix}}$$

$$\text{(a)} \quad T_p = \frac{10^3}{2 \times 10^8} = 5 \times 10^{-6} \text{ sec.}$$

$$T_{ix} = \frac{1000}{1 \times 10^6} = 10^{-3} \text{ sec.}$$

$$\text{Hence} \qquad a = \frac{5 \times 10^{-6}}{10^{-3}} = 5 \times 10^{-3}$$

$$\therefore \quad (K = 2) > (1 + 2a), \quad \text{and hence} \quad U = 1 = 100\%$$

$$\text{(b)} \quad T_p = \frac{10 \times 10^3}{2 \times 10^8} = 5 \times 10^{-5} \text{ sec.}$$

$$T_{ix} = \frac{1000}{200 \times 10^6} = 5 \times 10^{-6} \text{ sec.}$$

$$\text{Hence} \qquad a = \frac{5 \times 10^{-5}}{5 \times 10^{-6}} = 10$$

$$\therefore \quad (K = 7) < (1 + 2a), \quad \text{and hence} \quad U = \frac{K}{1 + 2a} = \frac{7}{1 + 20} = 0.33$$

$$\text{(c)} \quad T_p = \frac{50 \times 10^6}{2 \times 10^8} = 0.25 \text{ sec.}$$

$$T_{ix} = \frac{1000}{2 \times 10^6} = 5 \times 10^{-4} \text{ sec.}$$

$$\text{Hence} \qquad a = \frac{0.25}{5 \times 10^{-4}} = 500$$

$$\therefore \quad (K = 127) < (1 + 2a), \quad \text{and hence} \quad U = \frac{K}{1 + 2a} = \frac{127}{1 + 1000} = 0.127$$

We can deduce from these results that the choice of K has a strong impact on the link utilization in certain cases.

2.7.3 Link utilization in Presence of Errors

As before we consider the following two possible situations for arriving at the expressions for link utilization in presence of errors:

- The sender receives an acknowledgement before it exhausts the window.
- The sender exhausts the window before it receives an acknowledgement.

When there are no errors, the link utilization efficiency U for these two situations will be

$$U = 1 \qquad \text{When } K \geq 1 + 2a$$

$$U = \frac{K}{1+2a} \qquad \text{When } K < 1 + 2a$$

- Link utilization in selective retransmission

- When $K \geq 1+2a$, the link is continuously busy for transmission of I-frames during the round trip time for acknowledgement (T_{ix} +$2T_p$). If P_f is the probability of receiving a frame with errors, we can calculate the link utilization in presence of errors as follows:
  - Number of frames transmitted during (T_{ix} + $2T_p$), = (T_{ix} +$2T_p$)/T_{ix}
  - Number of frames received with errors = P_f (T_{ix} +$2T_p$)/T_{ix}
  - Link time wasted by the frames in error = T_{ix} P_f (T_{ix} + $2T_p$)/T_{ix} = P_f (T_{ix} +$2T_p$)
  - Time for which the link is effectively utilized =

$$(T_{ix} + 2T_p) - P_f (T_{ix} + 2T_p) = (T_{ix} + 2T_p)(1 - P_f)$$

$$\text{Link utilization efficiency, } U = \frac{(T_{ix} + 2T_p)(1 - P_f)}{(T_{ix} + 2T_p)} = (1 - P_f)$$

- When K < (1+ 2a), the link is used for time K T_{ix}, out of (T_{ix} +$2T_p$) for transmission of I- frames. For the rest of the period, the link is idle. If P_f is the probability of receiving a frame with errors, we can calculate the link utilization in presence of errors as follow:
  - Number of frames transmitted during (T_{ix} +$2T_p$), = K
  - Number of frames received with errors = P_f K
  - Link time wasted by the frames in error = T_{ix} P_f K
  - Time for which the link is effectively utilized = $KT_{ix} - T_{ix} P_f K$

$$= K T_{ix} (1 - P_f)$$

- Link utilization efficiency, $U = \frac{KT_{ix}(1-P_f)}{T_{ix}+2T_p} = \frac{K(1-P_f)}{1+2a}$

Thus link utilization is reduced by a factor of $(1-P_f)$ in either case

- Link utilization in Go–back–N

We need to keep in mind that in Go–back–N all the succeeding I-frames are discarded by the receiver when a frame is received with errors. On each such occurrence, $2T_p+T_{ix}$ of the link time is lost, Figure 2.14. To simplify the mathematical model, we assume that NAK is sent immediately on receipt of an I-frame with errors instead of waiting for the next frame.

With this background, we can now derive an expression for link utilization. As before, we will consider two cases, $K \geq 1+2a$ and $K < 1+2a$.

- When $K \geq 2a+1$. On average an I-frame requires N_r transmissions to send it correctly once. Therefore, $(N_r-1)(T_{ix} + 2T_p)$ of link time is wasted. The sender utilizes the link effectively for time equal to T_{ix} for sending the last transmission. Therefore, link utilization efficiency is given by:

$$U = \frac{T_{ix}}{T_{ix}+(N_r-1)(T_{ix}+2T_p)} = \frac{1}{1+(N_r-1)(1+2a)}$$

$$\because N_r = \frac{1}{1-P_f}, \qquad \text{then}$$

$$U = \frac{(1-P_f)}{(1-P_f)+P_f(2a+1)} = \frac{1-P_f}{1+2aP_f}$$

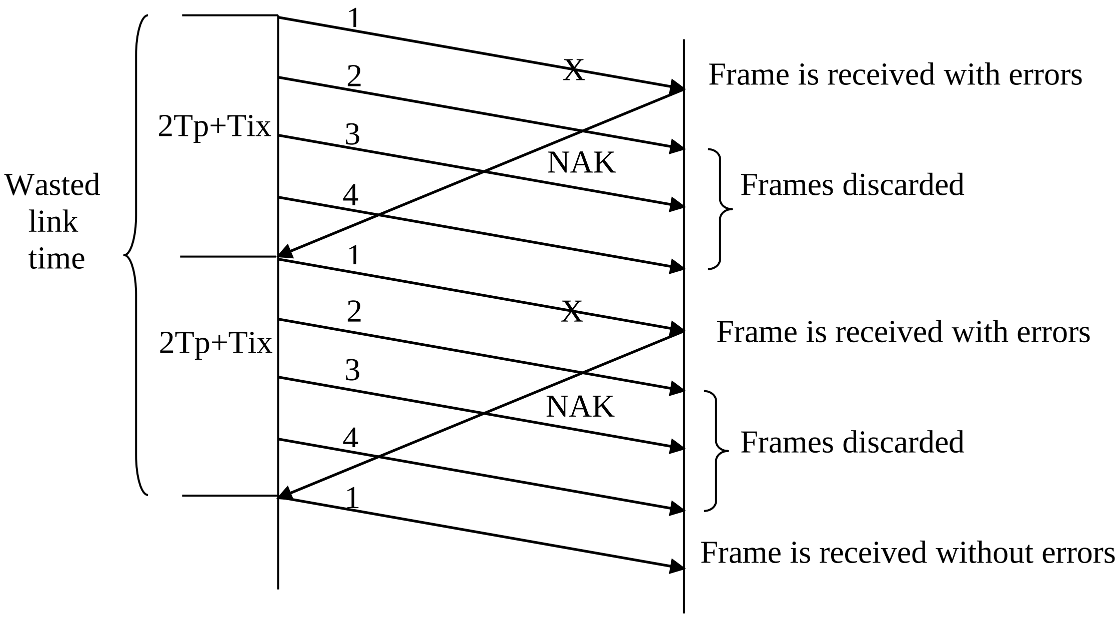


Figure 2.14 Link Utilization in Go-back-N When $K \geq 2a+1$

- When k < 1+2a. In the previous case, the sender was never idle because K $\geq$ 1+2a. But when K < 1+2a, the sender sends K I- frames and then it remains idle till ACK or NAK is received from the receiver. We need to take this factor into account. So long as a frame is received with error, the entire round trip time ($2T_p + T_{ix}$) is wasted. This includes the idle time of the sender. Therefore, $(N_r-1)(2T_P+T_{ix})$ of link time is wasted, Figure 2.15

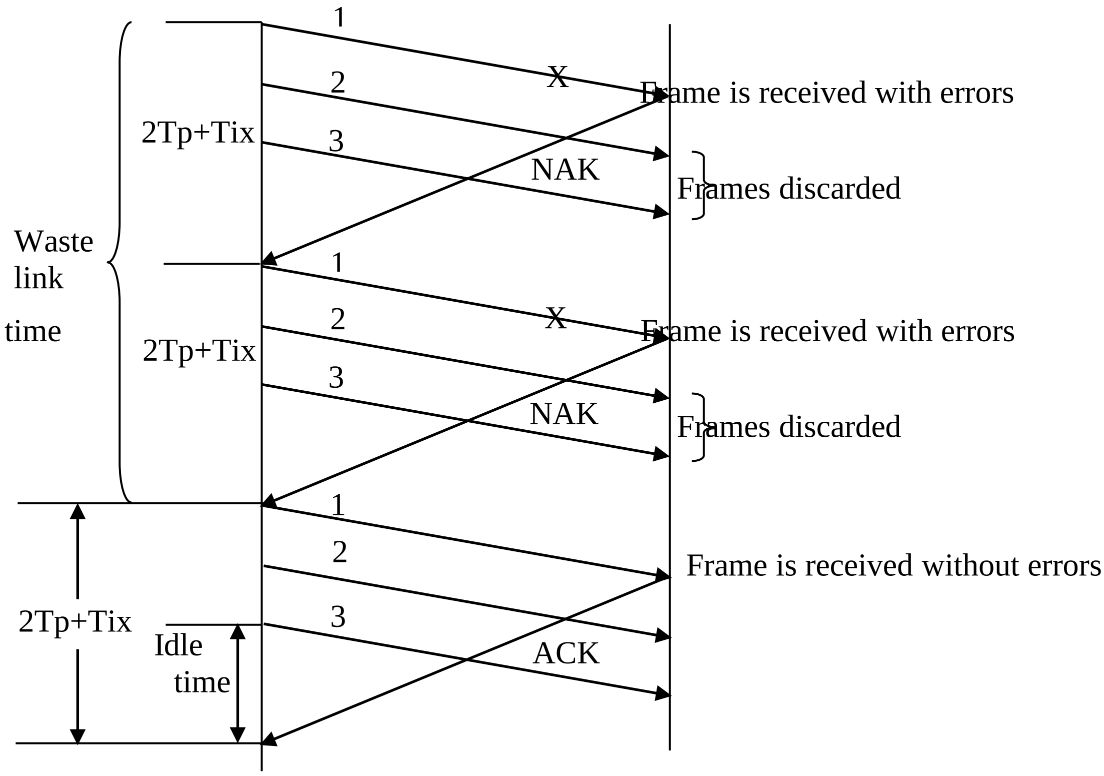


Figure 2.15 Link Utilization in Go–back–N When K < 2a + 1

When the sender sends the I-frame Nr^{th} time, the I-frame is received correctly. But this cycle of transmission of K I-frames includes idle time of the sender also. The link is engaged for time $(2T_p+T_{ix})$ for sending K I-frames. Therefore, on average an I-frame engages the link for time equal to $(2T_p + T_{ix})/K$. The total time for which the link is engaged for a transmission an I-frame correctly is:

$$(N_r - 1)(2T_p + T_{ix}) + (2T_p + T_{ix})/K$$

Therefor, link utilization (U) for Go-back-N mechanism, when K < 1+2a is given by:

$$U = \frac{T_{ix}}{(N_r - 1)(2T_p + T_{ix}) + (2T_p + 2T_{ix})/K} = \frac{K}{(2a+1)[1 + K(N_r - 1)]}$$

$$U = \frac{K(1 - P_f)}{(2a+1)(1 - P_f + KP_f)}$$

Example:

Calculate link utilization for the following link parameters: I-frame size is 1000-byte ,T_p = 40 ms, BER = $1x10^{-5}$, transmission rate = 2Mbps using the following link protocols:

i. Selective repeat, with K = 7 & K = 127
ii. Go-back-N, with K = 7 & K = 127

Solution

$$P = 0.00001, \qquad P_f = 1-(1-P)^L = 1-(1-0.00001)^{8000} = 0.076884$$

$$T_{ix} = \frac{8000}{2\times 10^6} = 4ms, \qquad T_p = 40ms$$

$$a = \frac{T_p}{T_{ix}} = \frac{40}{4} = 10, \qquad 1+2a = 21$$

(a) Selective repeat

i- K = 7, therefore K < 2a + 1

$$\therefore \quad U = \frac{K(1-P_f)}{2a+1} = \frac{7(1-0.076884)}{21} = 0.307$$

ii- K = 127, therefore K > 2a + 1

$$\therefore \quad U = (1-P_f) = 1-0.076884 = 0.923$$

(b) Go-back-N

i- K = 7, therefore K < 2a + 1

$$\therefore \quad U = \frac{K(1-P_f)}{(2a+1)(1-P_f+KP_f)} = \frac{7(1-0.076884)}{(2\times 10+1)(1-0.076884+7\times 0.076884)} = 0.21$$

ii- K = 127, therefore K > 2a + 1

$$\therefore \quad U = \frac{1-P_f}{1+2aP_f} = \frac{1-0.076884}{1+2\times 10\times 0.076884} = 0.36$$