Unit II

2 Error Detection, Correction and Data Link Control

2.1 : Data Link Layer

Q.1 Explain services provided to network layer.

Ans. : • The primary responsibility of data link layer is to provide services to the network layer. The principle service is transferring data from the network layer on the source machine to the network layer on the destination machine.

- The two data link layer communicates with each other by data link control protocol.
- Following are the important services provided by data link layer to the network layer.

1) Unacknowledged connectionless service.
2) Acknowledged connectionless service.
3) Acknowledged connection-oriented service.

1) Unacknowledged connectionless service : As the name suggests, it is unacknowledged form of transmission. Here the source machine sends the data to the destination machine without any acknowledgement. For this, no connection is either established or released. If the data is lost due to noise or interference, the lost data is not even recovered by the layer.

2) Acknowledged connectionless service : In acknowledged connectionless service each data frame is acknowledged by the destination machine. If any data frame is lost or not arrived in time the same can be transmitted again. In this service no connection are used.

3) Acknowledged connection service : Acknowledged connection service establishes a connection prior to data transmission. Each frame is numbered before transmission and corresponding acknowledgement is also received. The transmission is carried out in distinct phases.

2.2 : Error Detection and Correction

Q.2 Explain types of errors ?

Ans. : Two general types of errors can occur

1. Single bit error 2. Burst error

1. Single bit error

- It means that only 1 bit of a given data unit is changed from 1 to 0 or from 0 to 1. A single bit error is an isolated error condition that alters one bit but does not affect nearby bits.
- A single bit error can occur in the presence of white noise, when a slight random deterioration of the signal to noise ratio is sufficient to confuse the receiver's decision of a single bit. Single bit errors are the least likely type of error in serial data transmission.

2. Burst error

- The term burst error means that 2 or more bits in the data unit have changed from 1 to 0 or from 0 to 1.
- Burst errors are more common and more difficult to deal with errors. Burst errors can be caused by impulse noise. Note that the effects of burst errors are greater at higher data rates.

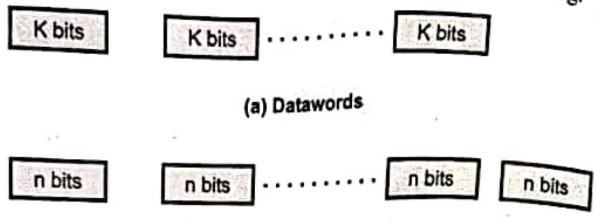
2.3 : Linear Block Codes

Q.3 Explain linear block coding, error detection and error correction.

Ans. : • In block coding, message is divided into blocks. Each block size is K bits and called as **datawords**. Redundant bits (r) is add to each block to make the length n = K + r. The resulting n-bit blocks are called **codewords**.

- With K bits, combination of 2^K datawords are possibe and with n bits, 2^n codewords combination are possible. The block coding process is one-to-one; the same dataword is always encoded as the same codeword.

- Fig. Q.3.1 shows the datawords and codewords in block coding.



(a) Datawords

(b) Codewords

Fig. Q.3.1 Datawords and codewords

Error Detection

- Following steps are used for detecting errors in the block coding.
  1. The receiver has a list of valid codewords.
  2. The original codeword has changed to an invalid one.
- Fig. Q.3.2 shows the role of block coding in error detection.

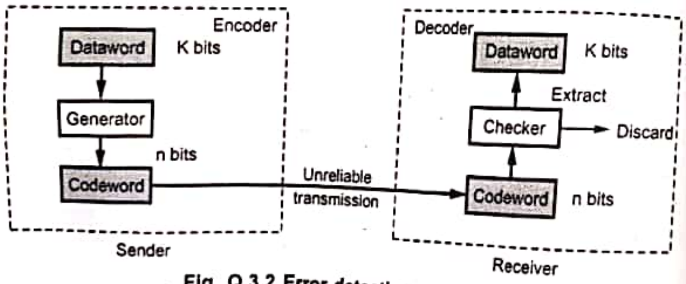


Fig. Q.3.2 Error detection process

- The sender creates codewords out of datawords by using a generator that applies the rules and procedures of encoding. Each codeword send to the receiver may change during transmission.
- If the received codeword is the same as one of valid codewords, the word is accepted; the corresponding dataword is extracted for use. If the received codeword is not valid, it is discarded.

- If the codeword is corrupted during transmission but the received word still matches a valid codeword, the error remains undetected.
- Block coding can detect only single errors. Two or more errors may remain undetected.

Error Correction

- Fig. Q.3.3 shows the error correction process. Error correction is much more difficult than error detection.

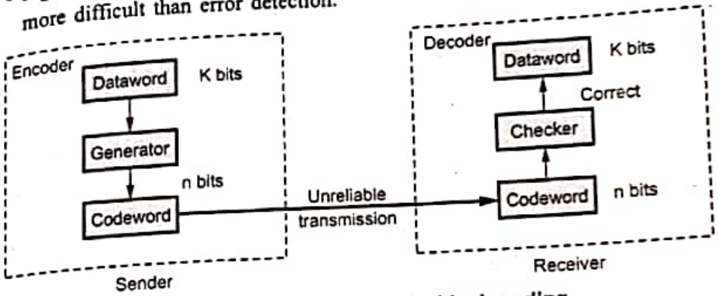


Fig. Q.3.3 Error correction in block coding

- In error correction, the receiver needs to find the original codeword sent. More number of redundant bits are required for error correction than for error detection.

Q.4 Explain hamming distance.

Ans. : - Hamming bits are inserted into the message at the random locations. Hamming code is a single error correcting code. It is most complex from the stand point of creating and interpreting the error bits. Let us consider a frame which consists of m data bits and r check bits. The total length of message is then n = m + r. An n-bit unit containing data and checkbits is often referred to as an **n-bit codeword.**

- If 10001001 and 10110001 are two codewords, then the corresponding bits differ in these two codewords is 3 bits. The number of bit positions in which two codewords differ is called the **Hamming distance.**
- If two codewords are a hamming distance d apart, it will require d single bit errors to convert one into the other. Determining the

placement and binary value of the hamming bits can be implemented using hardware, but it is often more practical to implement them using software.

- The number of bits in the message are counted and used to determine the number of hamming bits to be used. The equation is used to count the number of hamming bits.

$$2^H \geq M + H + 1 \qquad \text{... (Q.4.1)}$$

where M = Number of bits in a message

H = Hamming bits

- After calculating the number of hamming bits, the actual placement of the bits into the message is performed.
- Hamming code works as follows : Supose that frame consists of eight bits say $m_1\ m_2\ m_3\ m_4\ m_5\ m_6\ m_7\ m_8$. If n parity checks are used, there are 2^n possible combinations of failures and successes.
- If we use 4-bit parity checks, then there are 16 possible combinations of parity successes and failures. Total 12 bits are sent which contain 8-bit original message and 4-bit parity bits. The four parity is inserted into the frame.
- Four parity bits are $P_1\ P_2\ P_3$ and P_4. Let us consider following.

Data bit $m_1\ m_2\ m_3\ m_4\ m_5\ m_6\ m_7\ m_8$

Hamming code	P_1	P_2	m_1	P_3	m_2	m_3	m_4	P_4	m_5	m_6	m_7	m_8
	1	2	3	4	5	6	7	8	9	10	11	12

- The parity bits are inserted into the message. Position of the parity bit is calculated as follows. Create a 4 bit binary number $b_4\ b_3\ b_2$ and b_1 where

$b_i = 0$ if the parity check for P_i succeeds

$b_i = 1$ otherwise

for i = 1, 2, 3 or 4.

1) The parity bit P_1 is inserted at bit position 1 for even parity for bit positions 1, 3, 5, 7, 9, 10. In these bit positions it contains even number of 0s or 1s.
2) The parity bit P_2 is inserted at bit position 2, for even parity for bit positions 2, 3, 6, 7, 10, 11.
3) The parity bit P_3 is inserted at bit position 4, for even parity of the bit positions 4, 5, 6, 7, 12.
4) The parity bit P_4 is inserted at bit position 8 for even parity of the bit positions 8, 9, 10, 11, 12.

- For inserting the parity bit even or odd parity can be used. Each parity bit is determined by the data bits it checks. When a receiver gets a transmitted frame, it performs each of the parity checks.
- The combination of failures and successes then determines whether there was no error or in which position an error occurred. Once the receiver knows where the error occurred, it changes the bit value in that position and the error is corrected.

Minimum hamming distance (d_{min})

- The minimum hamming distance is the smallest hamming distance between all possible pairs in a set of words.
- To find the value of d_{min}, we find the hamming distances between all words and select the smallest one.

2.4 : Cyclic Codes

Q.5 What is meant by parity check ? Explain two-dimensional parity check method in detail. ☞ **[SPPU : Dec.-18, Marks 6]**

Ans. : • The simplest error detection scheme is to append a parity bit to the end of a block of data. Parity checking will detect any **single bit error**.

- The parity bit is transmitted with the data bits and the receiver checks the parity. If the receiver finds an odd number of 1 bits, an error has occurred.
- Suppose two bits change during transmission. If they were both 0, they change to 1. Two extra 1s still make the total number of 1 bits even. Similarly, if they were both 1 they both change to 0 and there are two

fewer 1 bits, but still an even number. If they were opposite values and both change, they are still opposite. This time the number of 1 bits remains the same. This means that the parity checks do not detect **double bit errors.**

- In general, if an odd number of bits change, parity checking will detect the error. If an even number of bits change, parity checking will not detect the error.
- Fig. Q.5.1 shows encoder and decoder for parity check.

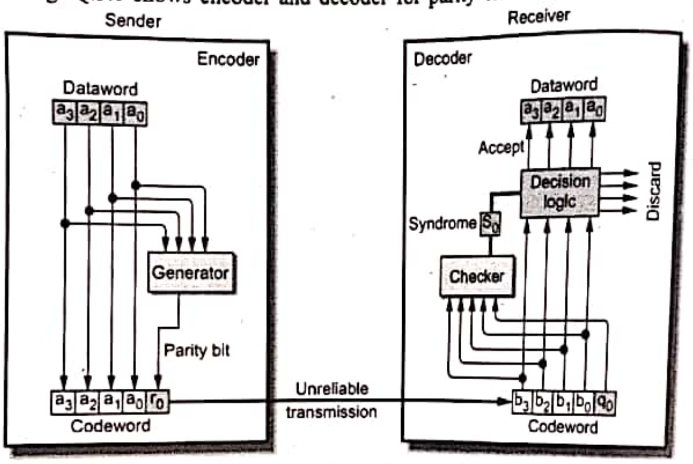


Fig. Q.5.1 Encoder and decoder for parity check

Q.6 Generate the CRC code for message 1101010101. Given generator polynomial

$g(x) = x^4 + x^2 + 1$ ☞ **[SPPU : May-17, Dec.-17, 18, 19, Marks 6]**

Ans. : For polynomial division *T(X)/G (X)*

where $T(X) = 1\ 1\ 0\ 1\ 0\ 1\ 0\ 1\ 0\ 1$

$$\left(x^9 + x^8 + x^6 + x^4 + x^2 + 1\right)$$

$$G(X) = x^4 + x^2 + 1 = 1\ 0\ 1\ 0\ 1$$

Polynomial division is done from an algebra.

Rules for addition and subtraction.

1.	Addition	2.	Subtraction
	0 + 0 = 0		0 − 0 = 0
	1 + 0 = 1		1 − 0 = 1
	0 + 1 = 1		0 − 1 = 1
	1 + 1 = 0		1 − 1 = 0

The steps are as follows :

Step 1 : Append 0 to the end of the string T(X).

The degree of polynomial $G(X) = x^4 + x^2 + 1 = 4$.

So we append 4 zeros to string $T(X)$.

The string becomes

1 1 0 1 0 1 0 1 0 1 0 0 0 0

Step 2 : Divide $B(X)$ by $G(X)$. After appending 0s to $T(X)$ it becomes $B(X)$. (Actually it is new $T(X)$ divided by $G(X)$).

```
              1110001110
10101) 11010101010000
       11010
       10101
        011111
          10101
          010100
            10101
            000011010
                 10101
                  011110
                   10101
                   010110
                     10101
                     000110      ← Remainder
```

```
  11010101010000
+           0110
  11010101010110   ⇐ Codeword
```

Q.7 Explain CRC generator and CRC checker with suitable example. ☞ [SPPU : May-19, Marks 6]

Ans. : • Parity method detects only odd numbers of errors. To overcome this weakness polynomial codes error detection method is used. Polynomial codes involve generating check bits in the form of a cyclic redundancy code (CRC). Therefore **polynomial also called cyclic redundancy codes (CRCs).**

- The theory of polynomial code is derived from a branch of mathematics called algebra theory. The theory of CRC checksums is developed by using algebra and polynomials. These polynomials are equations which have the form of powers of X :

$$X^{N} + X^{N-1} + ... + X^{2} + X^{1} + X^{0}$$

- Polynomial codes are used with frame transmission schemes. A single set of check digits is generated for each frame transmitted, based on the contents of the frame and is appended by the transmitter to the tail of the frame. The receiver then performs a similar computation on a complete frame and check digits. If no errors have been induced, a known result should always be obtained, if a different answer is found, this indicates an error. Consider an example for binary, the polynomial for binary 10011001 is

$$X^{7} + X^{4} + X^{3} + X^{0} \left(X^{0} = 1\right)$$

- The polynomial which represents the data bits is called the message polynomial, usually shown as G(X). There is a second polynomial, called the generator polynomial P(X). G(X) and P(X) both having same format. Combine two polynomials P(X) and G(X) to produce the CRC checksum polynomial F(X) calculating CRC error as follows :

a) Multiply the G(X) by X^{n-k}, where $n-k$ is the number of bits in the CRC checksum.

b) Divide the resulting product $X^{n-k}[G(X)]$ by the generator polynomial P(X).

c) Add the remainder C(X) to the product to give the F(X), which is representated as $X^{n-k}[G(X)]+C(X)$.

d) The division is performed in binary without carrying or borrowing. In this case, the remainder is always 1 bit less than the divisor. The remainder is the CRC and the divisor is the generator polynomial.

Working of CRC

- Let's now describe how CRC works. Suppose we want to send the bit string **1101011** and the generator polynominal is **G(x) = $x^4 + x^3 + 1$**

Step 1 : Append 0s to the end of the string. The number of 0s is the same the degree of the generator polynominal G(x) (in this case, 4). Thus the string becomes 11010110000.

Step 2 : Divide B(x) by G(x). We can write this algebraically as,

$$\frac{B(x)}{G(x)} = Q(x) + \frac{R(x)}{G(x)}$$

where Q(x) represent the quotient.

$$G(x) = x^4 + x^3 + 1 = 11001$$

String = 1101011 = After appending 11010110000

```
                 1 0 0 1 0 1 0   ←  Quotient
        11001) 1 1 0 1 0 1 1 0 0 0 0   ←  Dividend
Divisor ↗      1 1 0 0 1
               0 0 1 1 1
                 0 0 0 0 0
                 0 1 1 1 1
                   0 0 0 0 0
                   1 1 1 1 0
                     1 1 0 0 1
                     0 1 1 1 0
                       0 0 0 0 0
                       1 1 1 0 0
                         1 1 0 0 1
                         0 1 0 1 0
                         0 0 0 0 0
                           1 0 1 0   ←  Remainder
```

B(x) = 1101011000 bit string B

−

R(x) = 1010 bit string R

T(x) = 11010111010 bit string T

Step 3 : Define T(x) = B(x) − R(x). In this case,

Note that the string T is actually the same as string B with the appended 0s replaced by R. The sender transmit the string T.

Q.8 Information to be transmitted is 110011 and the generator polynomial is represented as $g(x)$ = 11001. Do a CRC check.

Ans. : Append by 4 bit 0 because coefficient of g(x) is 4.

The binary equivalent of d(x) = 1 1 0 0 1 1 0 0 0 0

```
              100001
       11001)1100110000
             11001
             11001
             0000010000
                  11001
                  01001        ←Remainder
```

Remainder is added to d(x) to give f(x) i.e.

1 1 0 0 1 1 0 0 0 0 + 0 1 0 0 1 = 1 1 0 0 1 1 1 0 0 1 ← f(x)

f(x) is transmitted.

2.5 : Framing

Q.9 What is mean by framing ? Explain character oriented protocol.

Ans. : • Framing in the data link layer separates a message from one source to a destination or from other messages to other destinations by adding a sender address and a destination address.

- To service the network layer, data link layer uses the service provided to it by the physical layer.
- Physical layer accepts the raw bit stream and delivers it to the destination. This bit stream may contain error i.e. number of bits received may not be equal to number of bits transmitted.
- The data link layer breaks the stream into discrete frames and computes the checksum for each frame.
- At the destination the checksum is recomputed.
- The breaking of bit stream by inserting spaces or time gaps is called framing. Since it is difficult and risky to count on timing and mark the start and end of each frame.

Fixed-size framing

- Frames can be of fixed or variable size. In fixed size framing, there is no need for defining the boundaries of the frames; the size itself can be used as a delimiter.
- ATM is the example of fixed size framing.

Variable Size Framing

- In variable size framing, end of the frame and the beginning of the next frame is defined.
- Two methods are used for this purposes.

  1. Character oriented 2. Bit oriented

2.6 : Flow Control

Q.10 Explain flow control.

Ans. : • When the sender is running on fast machine or lightly loaded machine and receiver is on slow or heavily loaded machine. Then the transmitter will transmit frames faster than the receiver can accept them.

- Even if the transmission is error free at a certain point the receiver will simply not be able to handle the frames as they arrive and will start to lose some.
- To prevent this, flow control mechanism is incorporated which includes a feedback mechanism requesting transmitter a retransmission of incorrect message block.
- The most common retransmission technique is known as Automatic-Repeat -Request.
- Error control in Data Link Layer (DLL) is based on Automatic Repeat Request (ARQ) i.e. retransmission of data in three cases.

  1. Damaged frames 2. Lost frames 3. Lost acknowledgements.

2.7 : Noiseless Channels

Q.11 Explain simplex stop and wait protocol.

Ans. : • Protocols in which the sender sends one frame and then waits for an acknowledgement before proceeding are called **stop-and-wait.**

- The communication channel is still assumed to be error free however and the data traffic is still complex.
- **Main problem :** How to prevent the sender from flooding the receiver with the data faster than the latter is able to process it.
- It is also assumed that there is no automatic buffering and queueing done within the receiver's hardware. The sender never transmit new frame until old one has been fetched by *from_physical_layer*.
- In some situations, delay is inserted by sender in the above protocol to slow it down sufficiently to keep from swamping the receiver.
- A more general solution to this dilemma is to have the receiver provide feedback (ACK) to the sender. After having passed a packet to its network layer, the receiver sends a little dummy frame back to the sender which, in effect, gives the sender permission to transmit the next frame.
- After having sent a frame, the sender is required by the protocol to bide its time until the little dummy (i.e., acknowledgement) frame arrives.
- Using feedback from the receiver to let the sender know when it may send more data is an example of the flow control.
- The simplest retransmission protocol is stop-and-wait. Transmitter (station A) sends a frame over the communication line and then waits for a positive or negative acknowledgement from the receiver (station B).
- If no errors occurred in the transmission, station B sends a positive acknowledgement (ACK) to station A.
- The transmitter can now start to send the next frame. If frame is received at station B with errors, then a negative acknowledgement

(NAK) is sent to station A. In this case station A must retransmit the old packet in a new frame.

- There is also the possibility that information frames and/or ACKs can be lost. To account for this, the sender is equipped with a timer. If no recognizable acknowledgement is received when the timer expires at the end of time out interval t_{out}, then the same frame is sent again.
- Fig. Q.11.1 shows the design of stop and wait protocol.

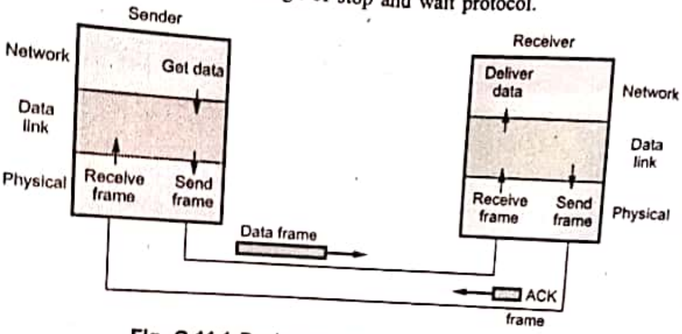


Fig. Q.11.1 Design of stop and wait protocol

- Protocols in which the sender sends one frame and then waits for an acknowledgement before process are called **stop and wait.**
- **Algorithm for sender**

```
void sender (void)
{
    frame s;
    packet buffer;
    event_type event;
while(true){
    from_network_layer(&buffer);
    s.info=buffer;
    to_physical_layer(&s);
    wait_for_event(&event);
```

```
    }
    }
```

- Fig. Q.11.2 shows the flow diagram.

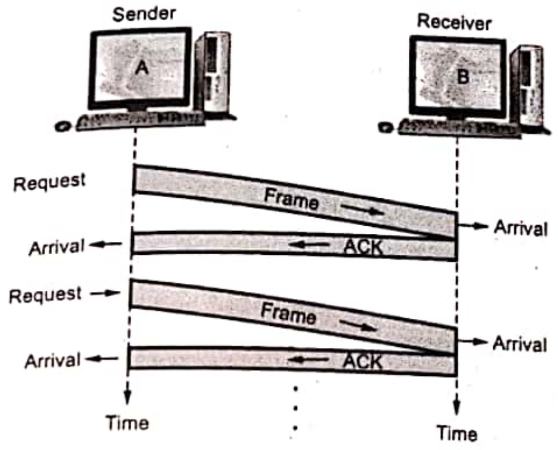


Fig. Q.11.2 Flow diagram for stop and wait

- **Algorithm for receiver side**

```
void receiver(void)
{
    frame r,s;
    event_type event;
while(true){

wait_for_event (&event);
from_physical_layer(&r);
to_network_layer(&r.info);
to_physical_layer(&s);
}
}
```

- **Major drawback of stop-and-wait flow control :**

1) Only one frame can be in transmission at a time.

2) This leads to inefficiency if propagation delay is much longer than the transmission delay.

2.8 : Noisy Channels

Q.12 Explain in detail go-back-N and selective repeat ARQ system.

☞ [SPPU : May-17,18, Dec.-17, Marks 6]

OR Explain in working mechanism of go-back-N and selective repeat ARQ system.

☞ [SPPU : Dec.-19, Marks 7]

Ans. : Go-Back-N ARQ Protocol

- Go-Back-N uses the sliding window flow control protocol. If no errors occur the operations are identical to sliding window.
- A station may send multiple frames as allowed by the window size.
- Receiver sends a NAK i if frame i is in error. After that, the receiver discards all incoming frames until the frame in error was correctly retransmitted.
- If sender receives a NAK i it will retransmit frame I and all packets i+1, i+2,... which have been sent, but not been acknowledged.
- The need for a large window on the sending side occurs whenever the product of bandwidth x round-trip-delay is large. If the bandwidth is high, even for a moderate delay, the sender will exhaust its window quickly unless it has a large window.
- If the delay is high, the sender will exhaust its window even for a moderate bandwidth. The product of these two factors basically tells what the capacity of the pipe is and the sender needs the ability to fill it without stopping in order to operate at peak efficiency. This technique is known as **pipelining**.
- As in Stop-and-Wait protocol senders has to wait for every ACK then next frame is transmitted. But in Go-Back-N ARQ W frames can be transmitted without waiting for ACK. A copy of each transmitted frame is maintained until the respective ACK is received.

Additional features of Go-Back-N ARQ

1) Sequence numbers : Sequence numbers of transmitted frame are maintained in the header of each frame. If k is the number of bits for sequence number, then the numbering can range from 0 to $2^k - 1$ e.g. for k = 3. Sequence numbers are 0 to 7 $\left(2^3 - 1\right)$.

2) Sender sliding window : Window is a set of frames in buffer waiting for acknowledgment. This window keeps on sliding in forward direction. The window size is fixed. As the ACK is received, the respective frame goes out of window and new frame to sent come into window. Fig. Q.12.1 illustrates sliding of window for window size = 7. (See Fig. Q.12.1 on next page)

3) Receiver sliding window : In the receiver side the size of the window is always one. The receiver is expecting to arrive frames in specific sequence. Any other frame received which is out of order is discarded. The receiver slides over after receiving the expected frame. Fig. Q.12.2 shows receiver sliding window.

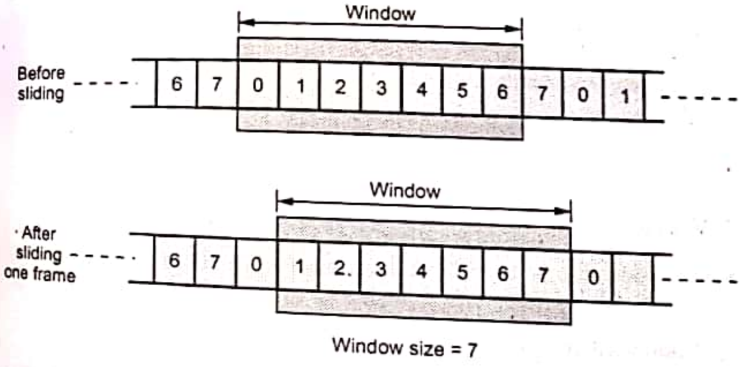


Fig. Q.12.1 Sender sliding window

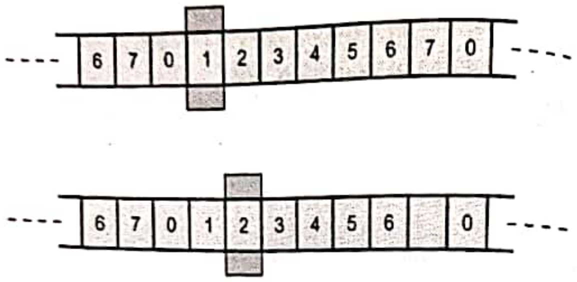


Fig. Q.12.2 Receiver sliding window

4) Control variables :

a) Sender variables

- The sender deals with three different variables.

  S → Sequence number of recently sent frame.

  S_F → Sequence number of first frame in window.

  S_L → Sequence number of last frame in window.

  ∴ Window size $W = S_L - S_F + 1$

  e.g. in previous feature, $W = 7-0+1 = 8$

b) Receiver variable

- The receiver deals with one variable only.

  R → Sequence number of frame expected

  If the number matches, then the frame is accepted otherwise not.

5) Timers

- The sender has a timer for each transmitted frame. The receiver don't have any timer.

6) Acknowledgment

- The receiver responds for frames arriving safely by positive acknowledgments. For damaged or lost frames receiver does not reply, the sender has to retransmit it when timer of that frame elapsed.
- The receiver may acknowledge once for several frames.

7) Resending of frames

- If the timer for any frame expires, the sender has to resend that frame and the subsequent frames also, hence the protocol is called Go-Back-N ARQ.

Operation

a) Normal operation

- The sender sends frames and update the control variables i.e. S_F, S, S_L and receiver updates variable R. Fig. Q.12.3 shows normal operation.

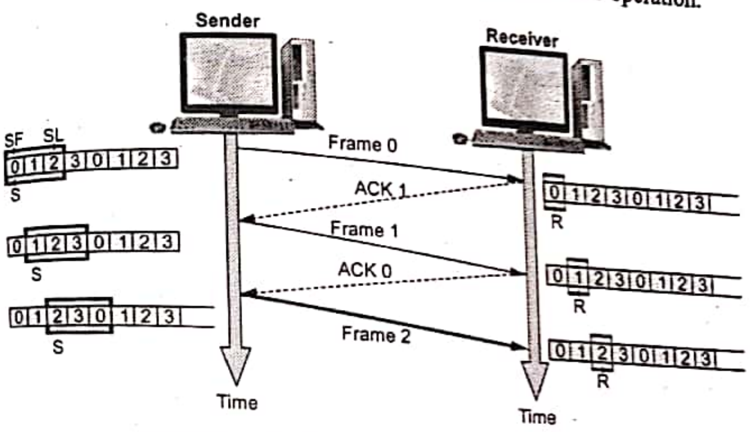


Fig. Q.12.3 Go-Back-N ARQ, normal operation

b) Damaged or lost frame

- Suppose frame 2 is damaged or lost and if receiver receives frame 3, it will be discarded since it is expecting frame 2. Sender retransmits frame 2 and frame 3. Fig. Q.12.4 shows this process.

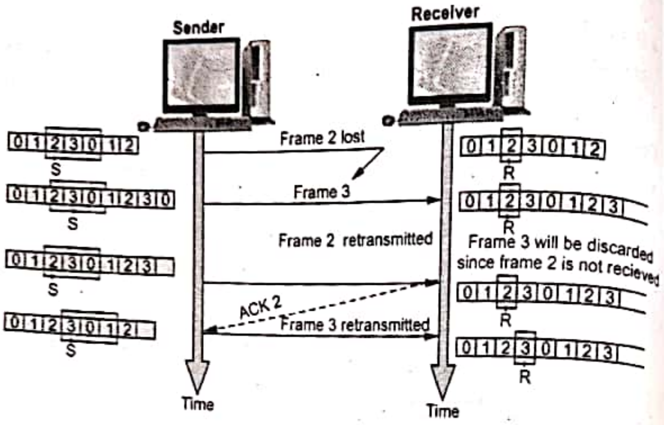


Fig. Q.12.4

Selective Repeat ARQ Protocol

- Selective repeat ARQ retransmitts only the damaged or lost frames instead of sending multiple frames. The selective retransmission increases the efficiency of transmission and is more suitable for noisy channel. The circuit complexities at the receiver side increases.
- The size of sender window is one half of 2^k. The receiver window size is of same length as that of sender. The receiver window includes the set of expected frames. The boundaries of receiver windows are defined by R_F and R_L. Fig. Q.12.5 shows the sender and receiver windows.

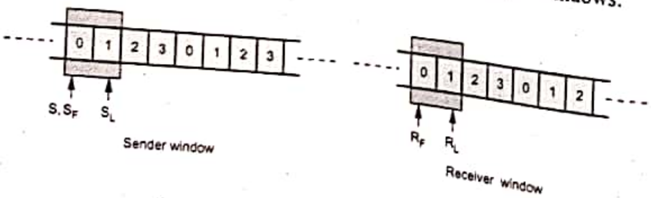


Fig. Q.12.5 Selective repeat windows

- Negative acknowledgement (NAK) is used for lost or damaged frames.

Operation

- In sequential transmission of frame 0, 1, 2, 3, suppose frame 2 is lost and the next frame 3 is already received then receiver sends NAK 2 frame to sender. Then sender retransmits frame 2 only. Fig. Q.12.6 shows operation of selective repeat ARQ.

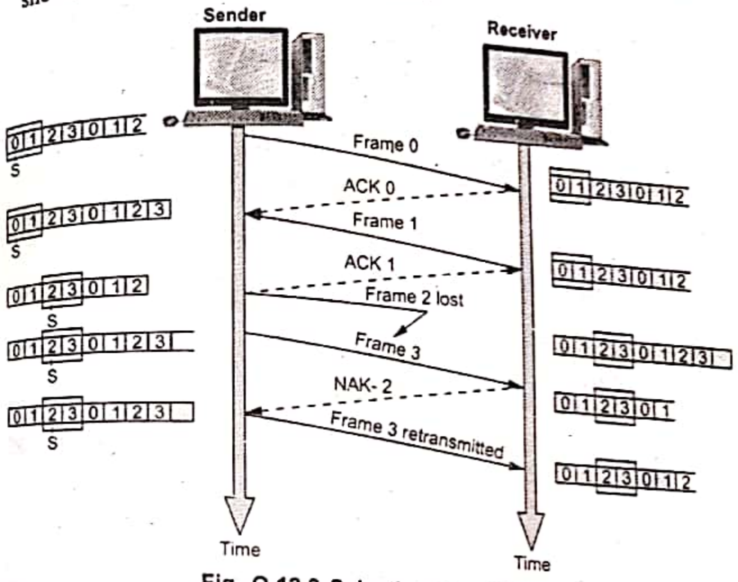


Fig. Q.12.6 Selective repeat ARQ

Advantage :

1) Fewer retransmissions.

Disadvantages :

1) More complexity at sender and receiver.
2) Each frame must be acknowledged individually (no cumulative acknowledgements).
3) Receiver may receive frames out of sequence.

Q.13 Give difference between go-back-n and selective repeat.

Ans. :

Sr. No.	Go-Back-N	Selective repeat
1.	Go-back-N requires all retransmission of the succeeding frame along with the lost or damaged frame.	In selective repeat, only the specific damaged or lost frame is retransmitted.
2.	Sender does not require any logic to select the specific frame for retransmission.	Extra logic is required for searching and retransmission of specific frame.
3.	Receiver do not required any sort of storage and sorting mechanism.	The complexity of sorting and storage mechanism is required by the receiver.
4.	It is not expensive.	It is expensive.

END...