CMRINSTITUTE OF TECHNOLOGY

USN | | | | | | | | | | |

Internal Assessment Test II – April 2023

Sub:	Computer Networks							Sub Code:	22MCA14
Date:	25/04/2023	Duration:	90 min's	Max Marks:	50	Sem:	I	Branch:	MCA

Note : Answer FIVE FULL Questions, choosing ONE full question from each Module

	PART I	MARKS	OBE	
			CO	RBT
1	a) Explain frame format of HDLC protocol **OR**	[10]	CO4	L2
2	a) Explain Error Control and Flow control? b) Explain error detection and correction. c) Explain types of errors occur in data transmission	[3] [3] [4]	CO4	L2
3	**PART II** a) Demonstrate stop and wait ARQ protocol by considering acknowledgment, timer and sequence no with the help of flow diagram **OR**	[10]	CO4	L1
4	a) Explain the operation of Selective Repeat protocol	[10]	CO4	L2
5	**PART III** a) With a neat diagram, explain Go-Back-N ARQ protocol of noisy channel and explain flow control and error control is achieved. **OR**	[10]	CO4	L3
6	a) Explain PPP protocol frame format with a neat diagram, also mention the different transition phases of PPP.	[10]	CO4	L2
7	PART IV Using CRC, given the data word 101001111 and the divisor 10111 a) Show the generation of code word at sender side. b) Show the checking of code word at receiver side. **OR**	[5] [5]	CO4	L4
8	a) Define Hamming distance and minimum hamming distance. b) Calculate the pair wise hamming distance and minimum Hamming distance among the following code words. (10101,11110,01011) (00011110,10101001,10100110,00001110)	[4] [6]	CO4	L2
9	PART V a) What is the remainder obtained by dividing $x^7 + x^5 + 1$ by the generator polynomial $x^3 + 1$ b) Explain single bit parity check code encoder and decoder diagram with example. **OR**	[5] [5]	C04	L4
10	A bit stream 10011101 is transmitted using the standard CRC method. The generator polynomial is x3+1. a) What is the actual bit string transmitted? b) Suppose the third bit from the left is inverted during transmission. How will receiver detect this error?	[5] [5]	CO4	L2

HDLC

High-level Data Link Control (HDLC) is a bit-oriented protocol for communication over point-to-point and multipoint links.

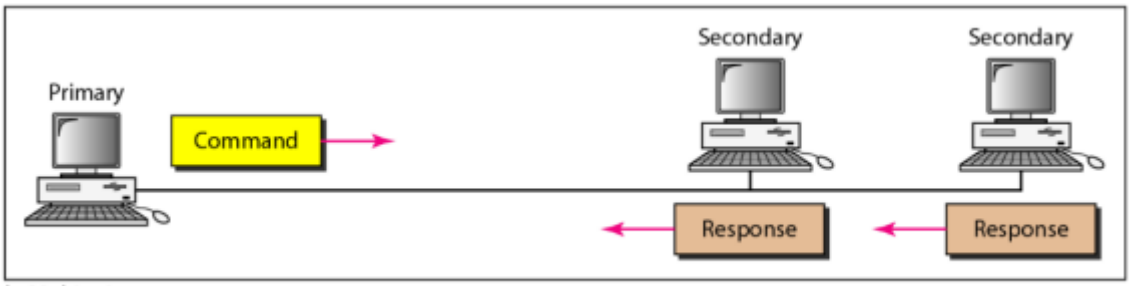
Configurations and Transfer Modes:

HDLC provides two common transfer modes :

1. Normal response mode (NRM)

- The station configuration is unbalanced.
- We have one primary station and multiple secondary stations.
- A primary station can send commands;a secondary station can only respond. The NRM is used for both point-to-point and multiple-point links.

a. Point-to-point



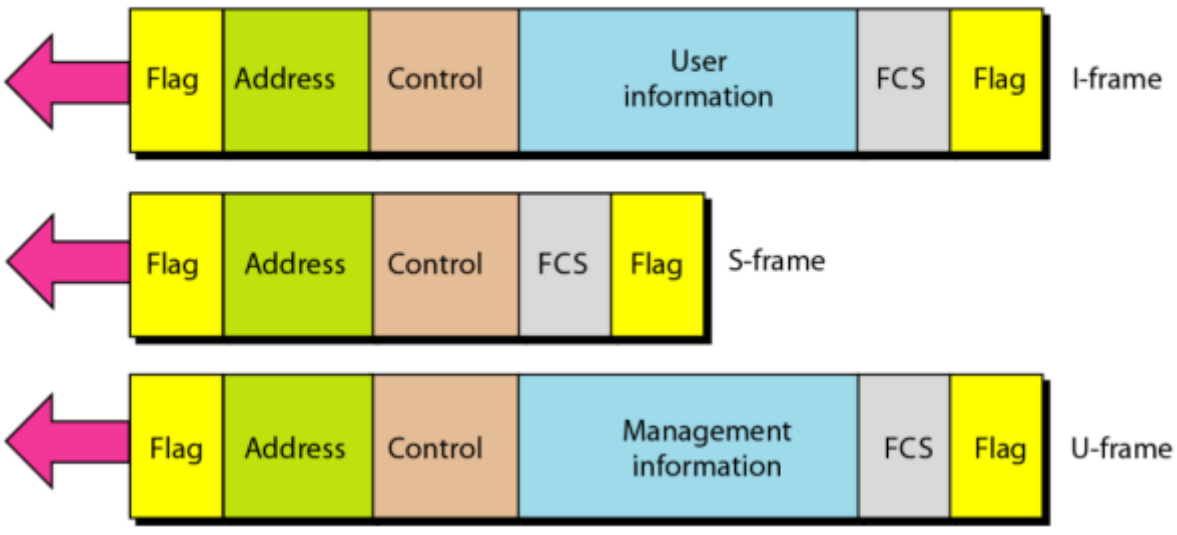
b. Multipoint

2. Asynchronous Balanced Mode(ABM)

- In asynchronous balanced mode (ABM), the configuration is balanced.
- The link is point-to-point, and each station can function as a primary and a secondary.

HDLC Frames:

Each frame in HDLC may contain up to six fields,a beginning flag field, an address field, a control field, an information field,a frame check sequence (FCS) field, and an ending flag field**.**



Fields :

1. Flag field: The flag field of an HDLC frame is an 8-bit sequence with the bit pattern 01111110 that identifies both the beginning and the end of a frame.

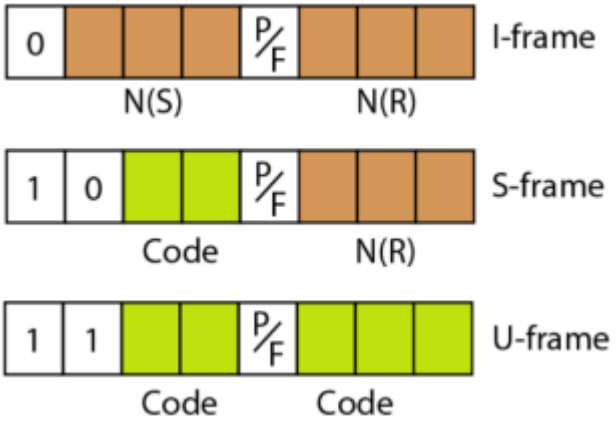
2. Address field: It contains the address of the receiver. If the frame is sent by the primary station, it contains the address(es) of the secondary station(s). If it is sent by the secondary station, it contains the address of the primary station. The address field may be from 1 byte to several bytes.

3. Control field : The control field is a 1- or 2-byte segment of the frame used for flow and error control.

4.Information field: The information field contains the user's data from the network layer or management information. Its length can vary from one network to another.

5. FCS field: The frame check sequence (FCS) is the HDLC error detection field. It can contain either a 2- or 4-byte.

6.Control Field : The control field determines the type of frame and defines its functionality**.**



Control Field for these Frames:

1. Control Field for I-Frames :

• I-frames are designed to carry user data from the network layer. In addition, they can include flow and error control information (piggybacking).

•The subfields in the control field are :

·The first bit defines the type. If the first bit of the control field is 0, this means the frame is an I-frame.

·The next 3 bits, called N(S), define the sequence number of the frame.

·The last 3 bits, called N(R), correspond to the acknowledgment number when piggybacking is used.

•The single bit between N(S) and N(R) is called the PIF(poll or final) bit.It means poll when the frame is sent by a primary station to a secondary (when the address field contains the address of the receiver). It means final when the frame is sent by a secondary to a primary (when the address field contains the address of the sender).

2. Control Field for S-Frames :

•Supervisory frames are used for flow and error control. •S-frames do not have information fields.

•If the first 2 bits of the control field is 10, this means the frame is an S-frame.

•The last 3 bits, called N(R), corresponds to the acknowledgment number (ACK) or negative acknowledgment number (NAK). •The 2 bits called code is used to define the type of S-frame itself. With 2bits, we can have four types of S-frames:

a. Receive Ready(RR):

▸▸If the value of the code subfield is 00, it is an RR S-frame. ▸▸This kind of frame acknowledges the receipt of a safe and sound frame or group of frames.

b. Receive Not Ready(RNR):

▸▸If the value of the code subfield is 10, it is a RNR S-frame. ▸▸This kind of frame is an RR frame with additional functions. ▸▸It acknowledges the receipt of a frame or group of frames, and it announces that the receiver is busy and cannot receive more frames.

c. Reject(REJ):

▸▸If the value of the code subfield is 01, it is a REJ S-frame. ▸▸It is a NAK that can be used in Go-Back-N ARQ to improve the efficiency of the process by informing the sender, before the sender time expires, that the last frame is lost or damaged.

d. Selective Reject(SREJ):

▸▸If the value of the code subfield is 11, it is an SREJ S-frame. ▸▸This is a NAK frame used in Selective Repeat ARQ.

3.Control Field for U-Frame:

▸▸Unnumbered frames are used to exchange session management and control information between connected devices. ▸▸U-frames contain an information field, but one used for system management information, not user data.

▸▸U-frame codes are divided into two sections: a 2-bit prefix before the P/F bit and a 3-bit suffix after P/F bit.

---

2)

a) Flow Control : Flow control refers to a set of procedures used to restrict the amount of data that the sender can send before waiting for acknowledgment.
b) Error Control : Error control in the data link layer is based on automatic repeat request, which is the retransmission of data.
c) Error Detection : Detection Versus Correction The correction of errors is more difficult than the detection. In error detection, we are looking only to see if any error has occurred. The answer is a simple yes or no. We are not even interested in the number of errors. A single-bit error is the same for us as a burst error. In error correction, we need to know the exact number of bits that are corrupted and more importantly, their location in the message. The number of the errors and the size of the message are important factors. If we need to correct one single error in an 8-bit data unit, we need to consider eight possible error locations; if we need to

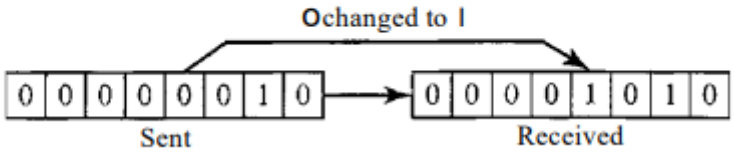
correct two errors in a data unit of the same size, we need to consider 28 possibilities. You can imagine the receiver's difficulty in finding 10 errors in a data unit of 1000 bits.

d) Forward Error Correction Versus Retransmission There are two main methods of error correction. Forward error correction is the process in which the receiver tries to guess the message by using redundant bits. This is possible, as we see later, if the number of errors is small. Correction by retransmission is a technique in which the receiver detects the occurrence of an error and asks the sender to resend the message. Resending is repeated until a message arrives that the receiver believes is error-free (usually, not all errors can be detected).

e) Types of Errors : Whenever bits flow from one point to another, they are subject to unpredictable changes because of interference. This interference can change the shape of the signal. In a single-bit error, a 0 is changed to a 1 or a 1 to a O. In a burst error, multiple bits are changed.

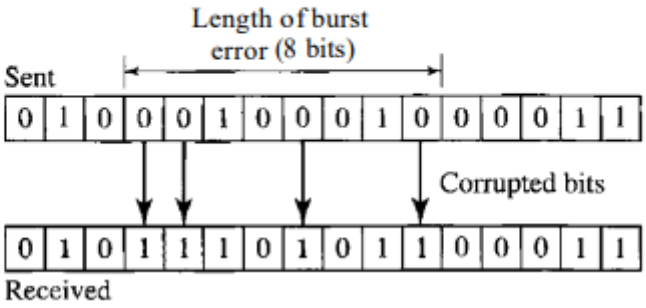
Single-Bit Error: In a single-bit error, only 1 bit in the data unit has changed.

Single-bit error



Burst Error:
A burst error means that 2 or more bits in the data unit have changed.

Burst error of length 8



3)

STOP-AND-WAIT ARQ PROTOCOL

● Stop-and-wait ARQ protocol comes under noisy channel. ● There is flow control as well as error control in this protocol. ● After transmitting one frame the sender waits for an acknowledgement before transmitting the next frame. If the acknowledgement doesn't arrive after a certain period of time then the sender again sends the same packet.

● Error Correction is done in this protocol by keeping a copy of the sent frames and retransmitting of the frame when the time expires.

Sequence Number -

There is also a sequence number in this protocol. Sequence number is used to number the frames, it is in the form of 0 and 1 only. The sequence number are based on modulo-2 arithmetic. A data frame uses a SeqNo(sequence number).

Acknowledgement Number

In stop-and-wait ARQ, the acknowledgement number always announces in modulo-2 arithmetic, the sequence number of the next frame expected. An ACK frame uses an ackNo(acknowledgement number).

Assume we have used x as a sequence number; 1. The frame arrives safe and sound at the receiver site; the receiver sends an acknowledgment. The acknowledgment arrives at the sender site, causing the sender to send the next frame numbered x + 1.

2. The frame arrives safe and sound at the receiver site; the receiver sends an acknowledgment, but the acknowledgment is corrupted or lost. The sender resends the frame (numbered x) after the time-out. Note that the frame here is a duplicate. The receiver can recognize this fact because it expects frame x + I but frame x was received.

3. The frame is corrupted or never arrives at the receiver site; the sender resends the frame (numbered x) after the time-out.

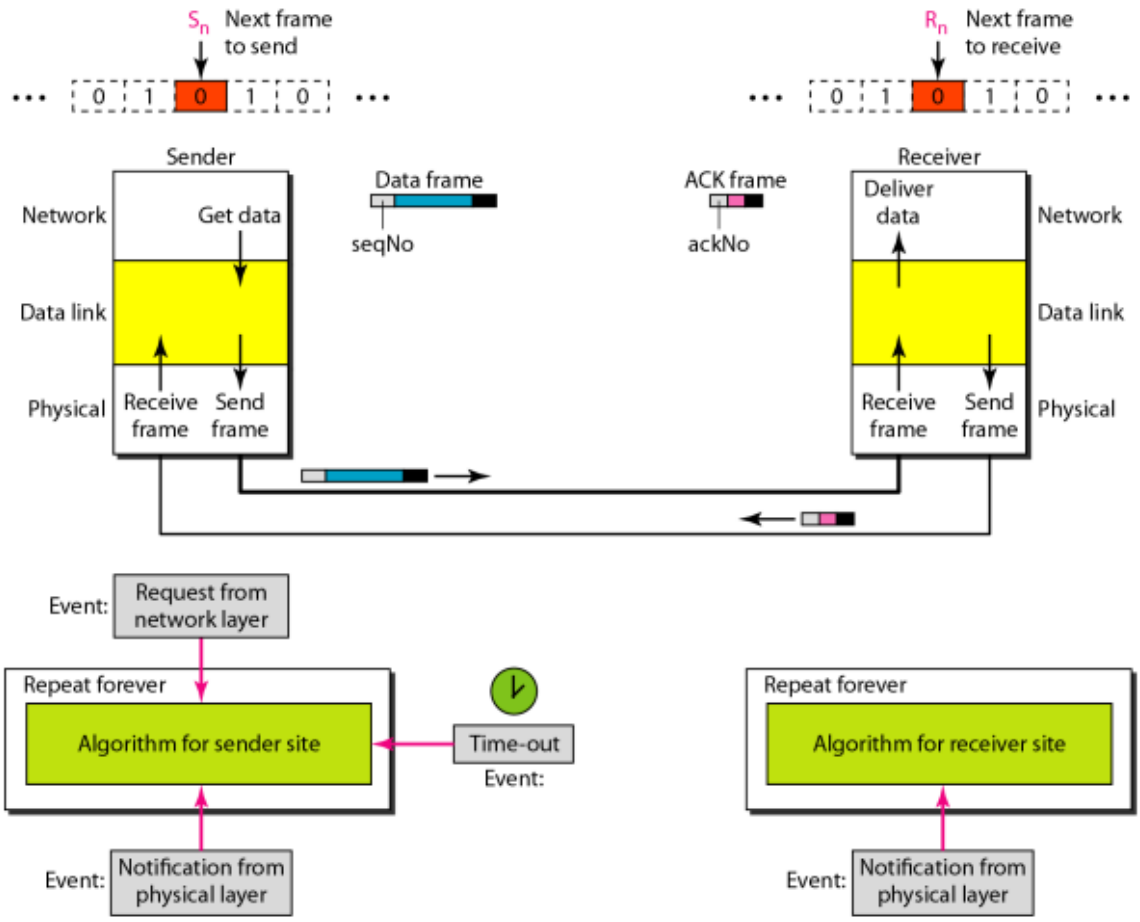
● In stop-and-wait ARQ, the sender has a control variable , which we call Sn(sender, next frame to send), that holds the sequence number for the next frame to be sent(0 or 1).

● The receiver has a control variable, which we call Rn(receiver, next frame expected), that holds the number of the next frame expected.

● When a frame is sent, the value of Sn is incremented(modulo-2) which means if it is 0, it becomes 1 and vice-versa.

● When a frame is received, the value of Rn is incremented (modulo-2), which means if it is 0, it becomes 1 and vice-versa.

Figure 11.10 *Design of the Stop-and-Wait ARQ Protocol*



The sending device keeps a copy of the last frame transmitted until it receives an acknowledgment for that frame. A data frames uses a seqNo (sequence number); an ACK frame uses an ackNo (acknowledgment number). The sender has a control variable, which we call Sn (sender, next frame to send), that holds the sequence number for the next frame to be sent (0 or 1). The receiver has a control variable, which we call Rn (receiver, next frame expected), that holds the number of the next frame expected. When a frame is sent, the value of Sn is incremented (modulo-2), which means if it is 0, it becomes 1 and vice versa. When a frame is received, the value of Rn is incremented (modulo-2), which means if it is 0, it becomes 1 and vice versa. Three events can happen at the sender site; one event can happen at the receiver site.
Flow Diagram :

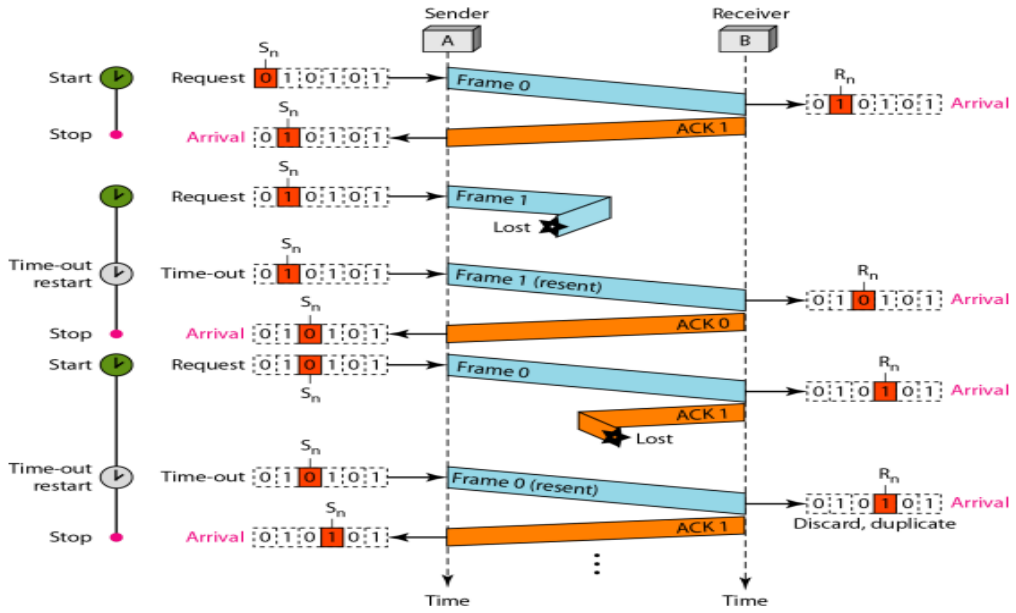


Figure 11.11 shows an example of Stop-and-Wait ARQ. Frame 0 is sent and acknowledged. Frame 1 is lost and resent after the time-out. The resent frame 1 is acknowledged and the timer stops. Frame 0 is sent and acknowledged, but the acknowledgment

is lost. The sender has no idea if the frame or the acknowledgment is lost, so after the time-out, it resends frame 0, which is acknowledged.

---

4)

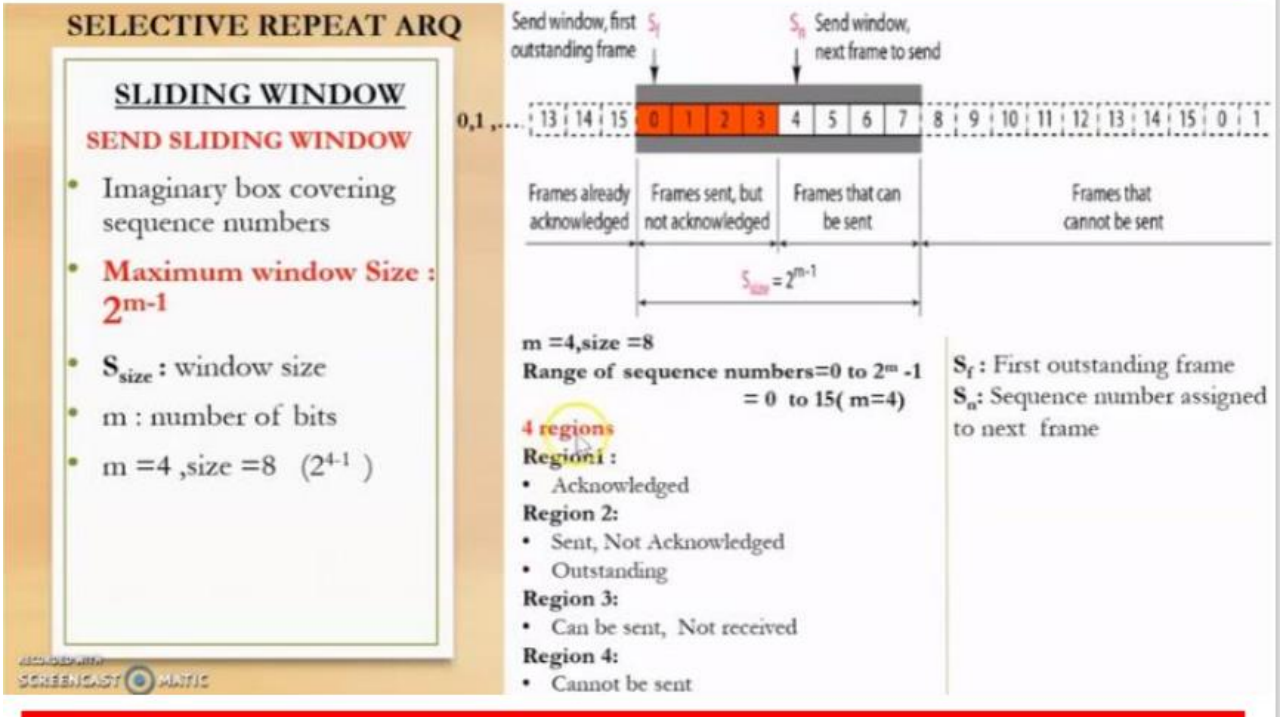
Selective Repeat Automatic Repeat Request: In stop and wait arq protocol, for a noisy link a frame has a higher probability of damage, which means the resending of multiple frames. This resending uses up the bandwidth and slows down the transmission. For noisy links, there is another mechanism that does not resend N frames when just one frame is damaged; only the damaged frame is resent. This mechanism is called Selective Repeat ARQ.

Windows : The Selective Repeat Protocol also uses two windows: a send window a receive window

The send window maximum size can be $2^m - 1$

For example, if m = 4, the sequence numbers go from 0 to 15, but the size of the window is just 8 The receive window in Selective Repeat is totally different from the one in GoBack-N. First, the size of the receive window is the same as the size of the send window $2^m - 1$

Figure 11.18 ***Send window for Selective Repeat ARQ***



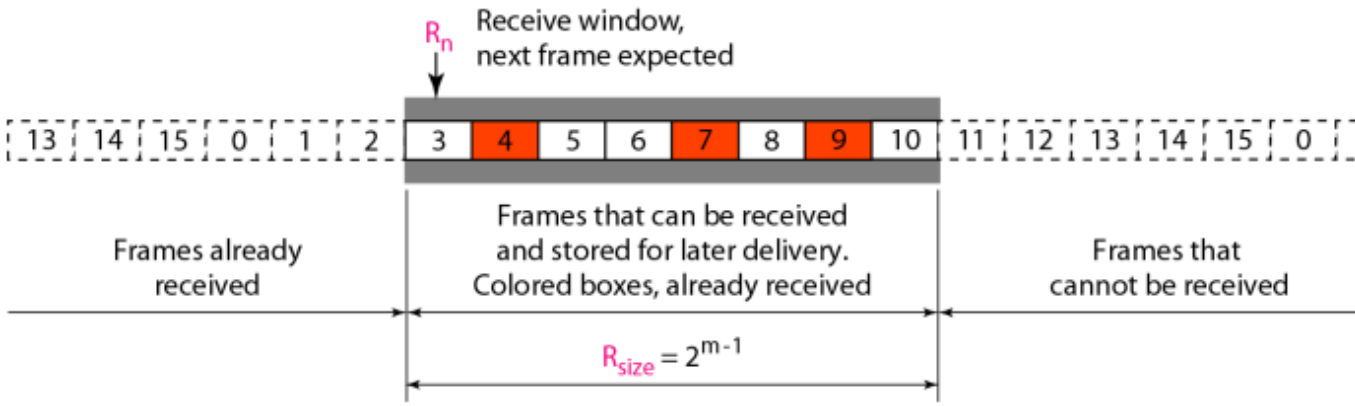


Figure 11.20 ***Design of Selective Repeat ARQ***

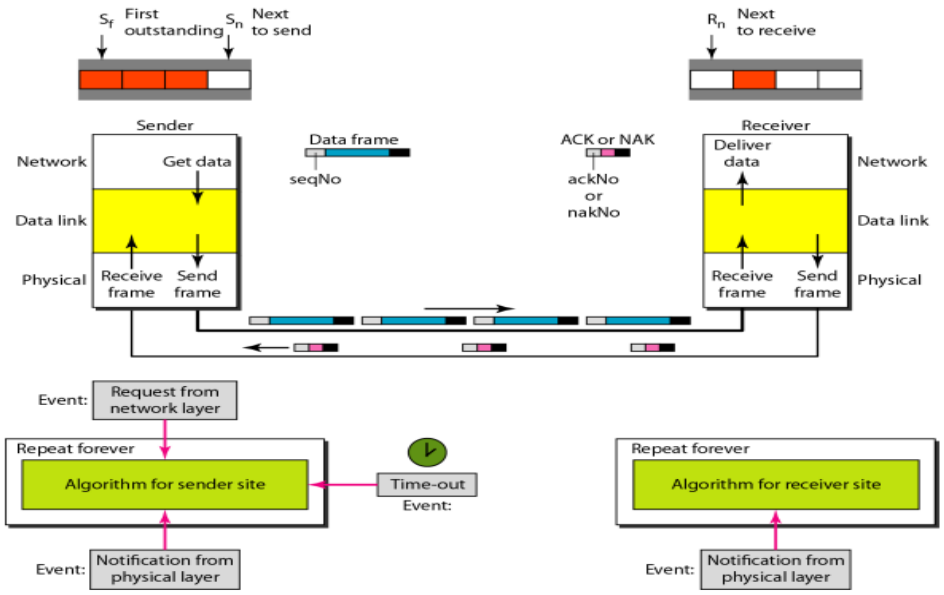
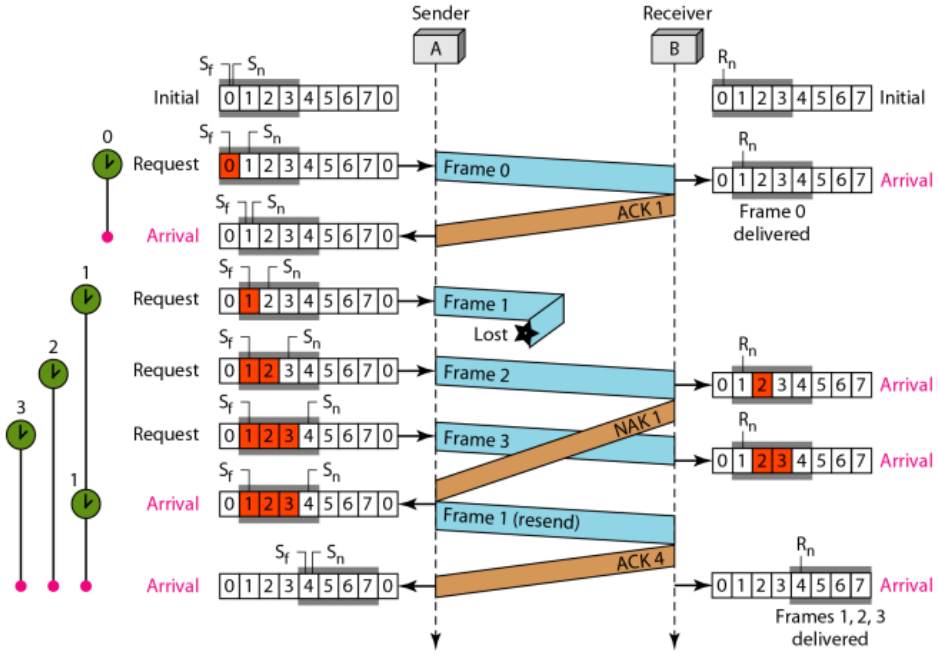


Figure 11.23 ***Flow diagram for Example 11.8***



5)

Go-Back-N ARQ PROTOCOL

● Go-Back-N ARQ protocol comes under noisy channel. ● Go-Back-N ARQ is mainly a specific instance of Automatic Repeat Request (ARQ) protocol where the sending process continues to send a number of frames as specified by the window size even without receiving an
acknowledgement(ACK) packet from the receiver. ● The sender keeps a copy of each frame until the arrival of acknowledgement.

Sequence Number-

● In the Go-Back-N ARQ protocol, the sequence numbers are module 2^m (2 to the power m) , where 'm' is the size of the sequence number fields in bits.

● Frames from a sending station are numbered sequentially. ● If the header of the frame allows 'm' bits for the sequence number, the sequence numbers range from 0 to 2^m - 1.

Window

1. Send Window:

● The send window is an imaginary box covering the sequence numbers of the data frames which can be in transmit.

● The window at any time divides the possible sequence numbers into four regions.

4 regions are :

● **Region 1:**

● Acknowledged

● **Region 2:**

● Sent , Not Acknowledged

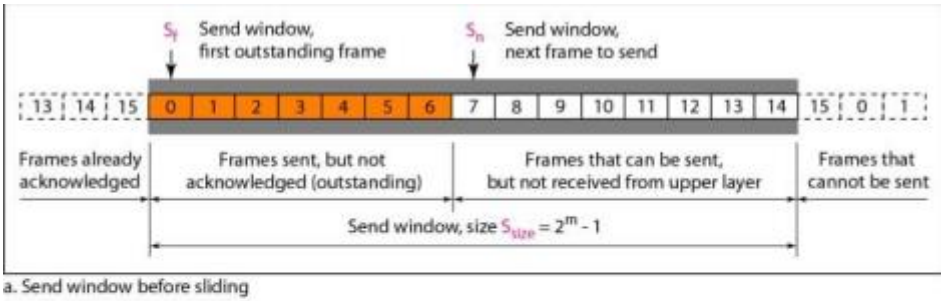
● **Region 3:**

● Can be sent

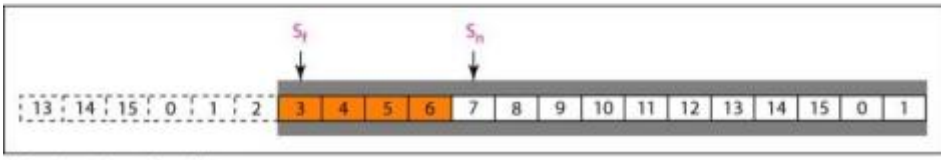
● Not received

● **Region 4:**

● Cannot be sent



a. Send window before sliding



b. Send window after sliding

E.g:

m=4, size=15

Range of sequence numbers = 0 to 2^m - 1

=0 to 15 (m=4)

· The send window is an abstract concept defining an imaginary box of size 2^m - 1 with three variables: Sf, Sn and Ssize.

• The send window can slide one or more slots when a valid acknowledgement arrives.

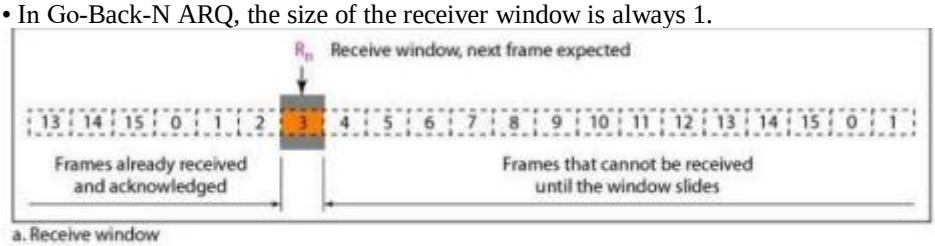
• In Go-Back-N ARQ, the size of the send window must be less than 2^m .

2. Receive Window:

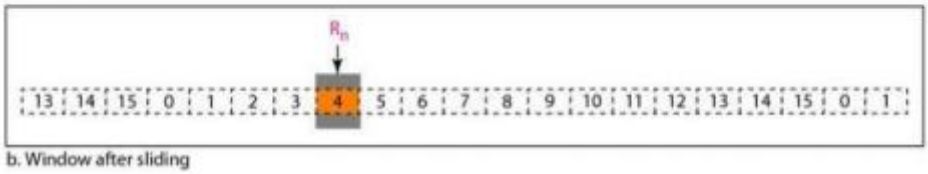
•The receive window is an abstract concept defining an imaginary box of size 1 with one single variable Rn. The window slides when a correct frame has arrived; sliding occurs one slot at a time.

•We need only one variable Rn (receive window, next frame expected) to define this abstraction. The sequence numbers to the left of the window belongs to the frames already received and acknowledged; the sequence numbers to the right of this window define the frames that cannot be received . Any received frame with a sequence number in these two regions is discarded. Only a frame with a sequence number matching the value of Rn is accepted and acknowledged.

• In Go-Back-N ARQ, the size of the receiver window is always 1.

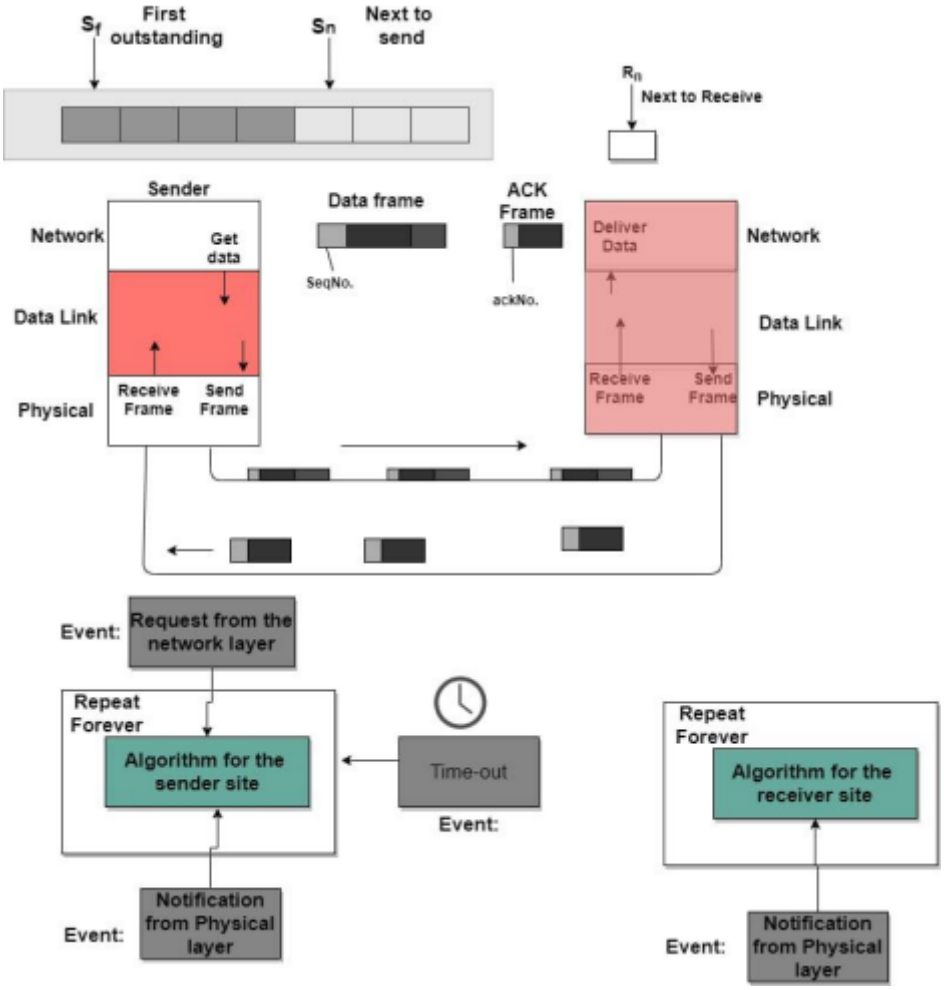


a. Receive window

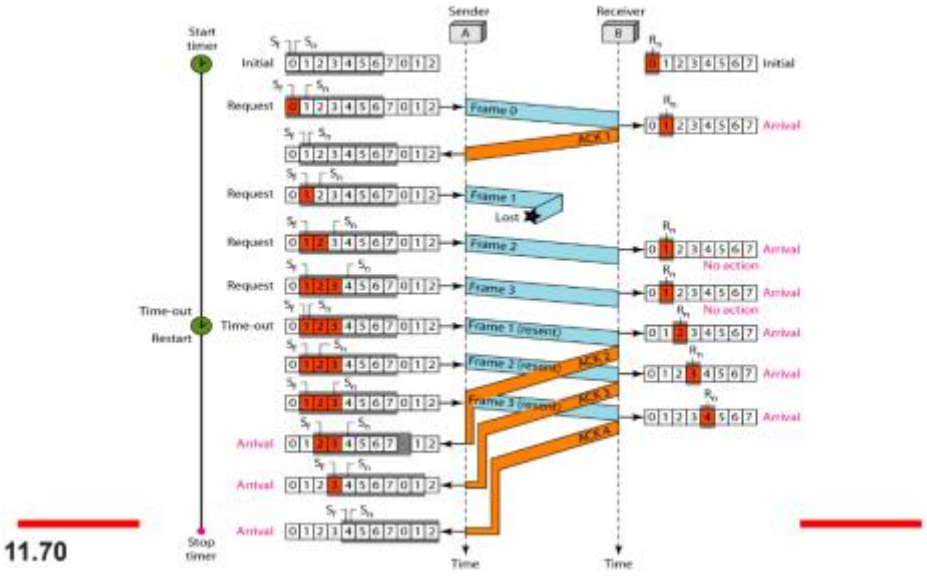


b. Window after sliding

Acknowledgement:
The receiver sends a positive acknowledgement if a frame has arrived safe and sound in order.
If a frame is damaged or is received out of order, the receiver is silent and will discard all subsequent frames until it receives the one it is expecting.
Resending a frame:
When the timer expires, the sender resends all outstanding frames.
For example: Suppose the sender has already sent frame 6, but the timer for frame 3 expires. This means that frame 3 has not been acknowledged; the sender goes back and sends frames 3,4,5 and 6 again. That is why the protocol is called Go-Back-N ARQ.
Design for Go-Back-N ARQ:



Flow Diagram for Go-Back-N ARQ:



11.70

Figure 11.17 shows what happens when a frame is lost. Frames 0, 1, 2, and 3 are sent. However, frame 1 is lost. The receiver receives frames 2 and 3, but they are discarded because they are received out of order. The
sender receives no acknowledgment about frames 1, 2, or 3. Its timer finally expires. The sender sends all outstanding frames (1, 2, and 3) because it does not know what is wrong. Note that the resending of frames 1, 2, and 3 is the response to one single event. When the sender is responding to this event, it cannot accept the triggering of

other events. This means that when ACK 2 arrives, the sender is still busy with sending frame 3.

---

6)

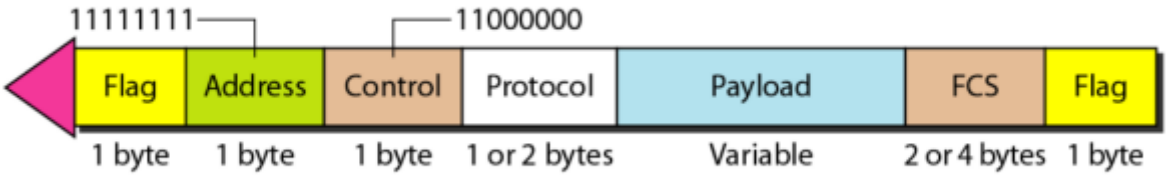
POINT-TO-POINT PROTOCOL

PPP is one of the most common protocols for point-to-point access. PPP is a byte-oriented protocol using byte stuffing with the escape byte 01111101.

PPP provides several services:

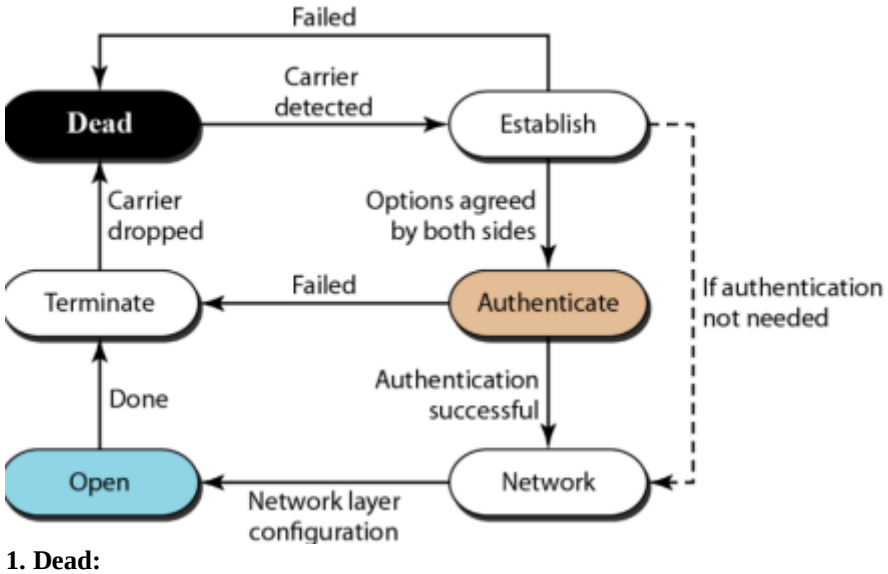
1. PPP defines the format of the frame to be exchanged between devices.
2. PPP defines how two devices can negotiate the establishment of the link and the exchange of data.
3. PPP defines how network layer data are encapsulated in the data link frame.
4. PPP defines how two devices can authenticate each other. **Services not provided by PPP**:
1. PPP does not provide flow control.
2. PPP has a very simple mechanism for error control. A CRC field is used to detect errors. If the frame is corrupted, it is silently discarded;

PPP Frame Format:



1. FLAG-
•A PPP frame starts and ends with a I-byte flag with the bit pattern 01111110.The flag is treated as a byte.
2. Address-
•The address field in this protocol is a constant value and set to 11111111 (broadcast address).During negotiation, the two parties may agree to omit this byte.
3. Control-
• The field is set to the constant value 11000000(imitating unnumbered frames in HDLC).
•PPP does not provide any flow control.
•Error control is also limited to error detection. This means that this field is not needed at all. The two parties can agree, during negotiation, to omit this byte.
4. Protocol-
•The protocol field defines what is being carried in the data field: either user data or other information.
•This field is by default 2 bytes long, but the two parties can agree to use only 1 byte.
5. Payload field-
•This field carries the user data or other information. •The data field is a sequence of bytes with the default of a maximum of 1500 bytes.
6. FCS-
•The frame check sequence (FCS) is simply a 2-byte or 4-byte standard CRC.

Transition Phase:



1. Dead:
In the dead phase the link is not being used.
2. Establish:
When one of the nodes starts the communication, the connection goes into this phase.
In this phase, options are negotiated between the two parties.

If the negotiation is successful, the system goes to the authentication phase (if authentication is required) or directly to the networking phase.

3. Authenticate:

The authentication phase is optional; the two nodes may decide, during the establishment phase, not to skip this phase. However, if they decide to proceed with authentication, they send several authentication packets.

If the result is successful, the connection goes to the networking phase; otherwise, it goes to the termination phase.

4. Network:

In the network phase, negotiation for the network layer protocols takes place.

PPP specifies that two nodes establish a network layer agreement before data at the network layer can be exchanged. The reason is that PPP supports multiple protocols at the network layer.

If a node is running multiple protocols simultaneously at the network layer, the receiving node needs to know which protocol will receive the data.

5. Open:

In the open phase, data transfer takes place.

The connection remains in this phase until one of the endpoints wants to terminate the connection.

6. Terminate:

In the termination phase the connection is terminated.

---

7)

data word = 101001111

divisor =10111 ...here divisor is 5 bits so pad 4 0's in data word

```
10111 ) 101001111 0000
          10111
        ----------------------
         000111111 0000
             10111
        ---------------------
           010001 0000
            10111
         ----------------------
            00110 0000
              101 11
         ----------------------
               011 1100
                 10 111
          ----------------------
                  01 0010
                   1 0111
                 --------------
                   0 0101
```

so code word will be 1010011110101.

```
_____________________
10111)1 0 1 0 0 1 1 1 1 0 1 0 1(
1 0 1 1 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 1
1 0 1 1 1

_______
1 0 0 0 1
1 0 1 1 1

_______
0 1 1 0 0
0 0 0 0 0

________
1 1 0 0 1
1 0 1 1 1

________
1 1 1 0 0
1 0 1 1 1

________
1 0 1 1 1
1 0 1 1 1

________
0 0 0 0 reminder of reciever side.
```

---

8)

Hamming Distance : The Hamming distance between two words is the number of differences between corresponding bits.

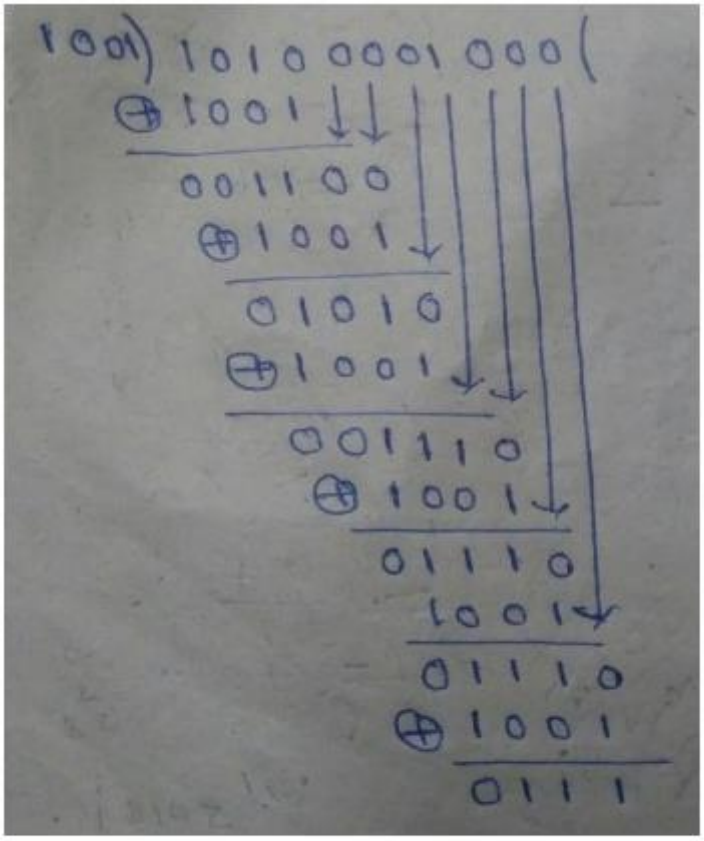
Minimum Hamming Distance :
The minimum Hamming distance is the smallest Hamming distance between all possible pairs in a set of words..

---

9)

a) What is the remainder obtained by dividing $x^7 + x^5 + 1$ by the generator polynomial $x^3 + 1$.

The remainder is x 2 + x + 1.



Remainder=$0 * x^3 + 1 * x^2 + 1 * x^1 + 1 * x^1$

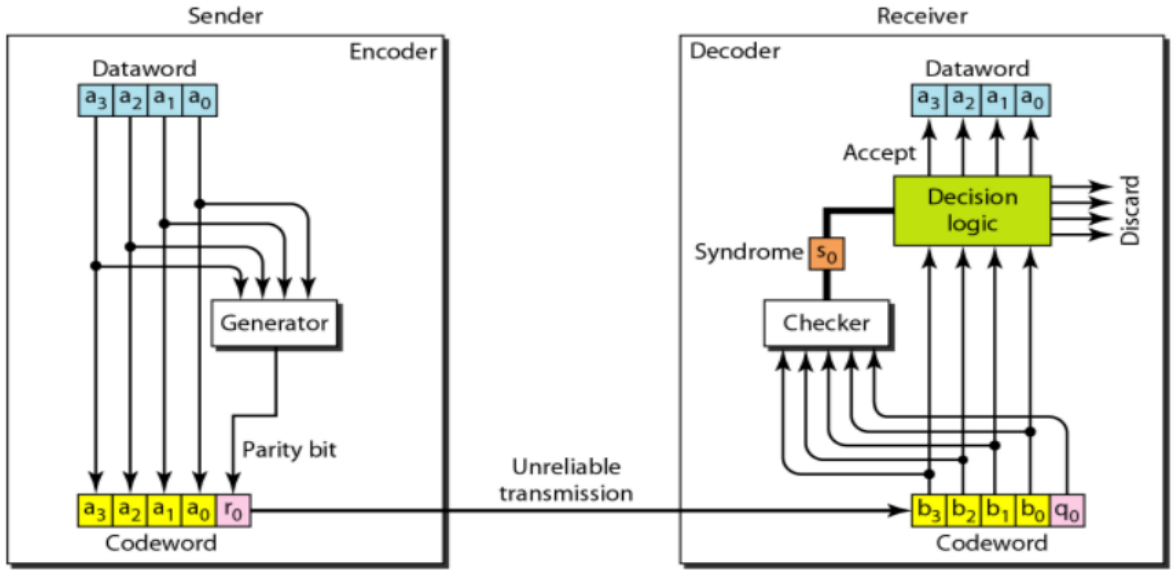
=$x^2 + x + 1$

9b)

Simple Parity-Check Code:

Perhaps the most familiar error-detecting code is the simple parity-check code. In this code, a k-bit dataword is changed to an n-bit codeword where n = k + 1. The extra bit, called the parity bit, is selected to make the total number of Is in the codeword even. Although some implementations specify an odd number of Is, we discuss the even case. The minimum Hamming distance for this category is dmin =2, which means that the code is a single-bit error-detecting code; it cannot correct any error. A simple parity-check code is a single-bit error-detecting code in which n =k + 1 with dmin =2.

Figure 10.10 ***Encoder and decoder for simple parity-check code***



Example 10.12 Let us look at some transmission scenarios. Assume the sender sends the dataword 1011. The codeword created from this dataword is 10111, which is sent to the receiver. We examine five cases: 1. No error occurs; the received codeword is 10111. The syndrome is O. The dataword 1011 is created.
2. One single-bit error changes aI' The received codeword is 10011. The syndrome is 1. No dataword is created.
3. One single-bit error changes roo The received codeword is 10110. The syndrome is 1. No dataword is created. Note that although none of the dataword bits are corrupted, no dataword is created because the code is not sophisticated enough to show the position of the corrupted bit.
4. An error changes ro and a second error changes a3' The received codeword is 00110. The syndrome is O. The dataword 0011 is created at the receiver. Note that here the dataword is wrongly created due to the syndrome value. The simple parity-check decoder cannot detect an even number of errors. The errors cancel each other out and give the syndrome a value of O.
5. Three bits-a3, az, and aI-are changed by errors. The received codeword is 01011. The syndrome is 1. The dataword is not created. This shows that the simple parity check, guaranteed to detect one single error, can also find any odd number of errors. A simple parity-check code can detect an odd number of errors.

---

10)

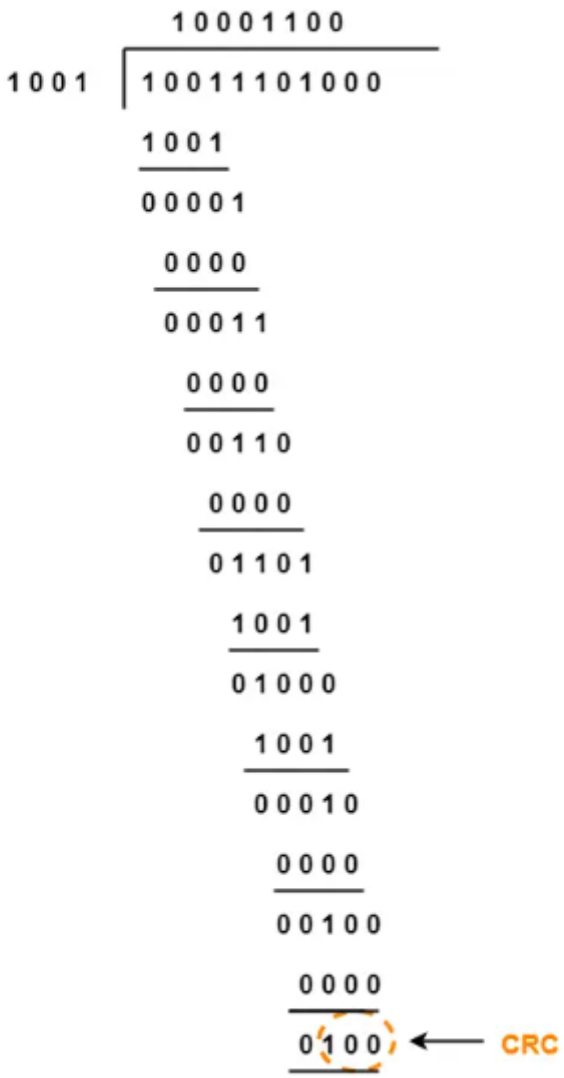
A bit stream 10011101 is transmitted using the standard CRC method. The generator polynomial is x^3+1.

1. What is the actual bit string transmitted?
2. Suppose the third bit from the left is inverted during transmission. How will receiver detect this error?

Solution-

Part-01:

- The generator polynomial $G(x) = x^3 + 1$ is encoded as 1001.
- Clearly, the generator polynomial consists of 4 bits.
- So, a string of 3 zeroes is appended to the bit stream to be transmitted.
- The resulting bit stream is 10011101**000**.



Now, the binary division is performed as-

From here, CRC = 100.

Now,

- The code word to be transmitted is obtained by replacing the last 3 zeroes of 10011101**000** with the CRC.
- Thus, the code word transmitted to the receiver = 10011101**100**.

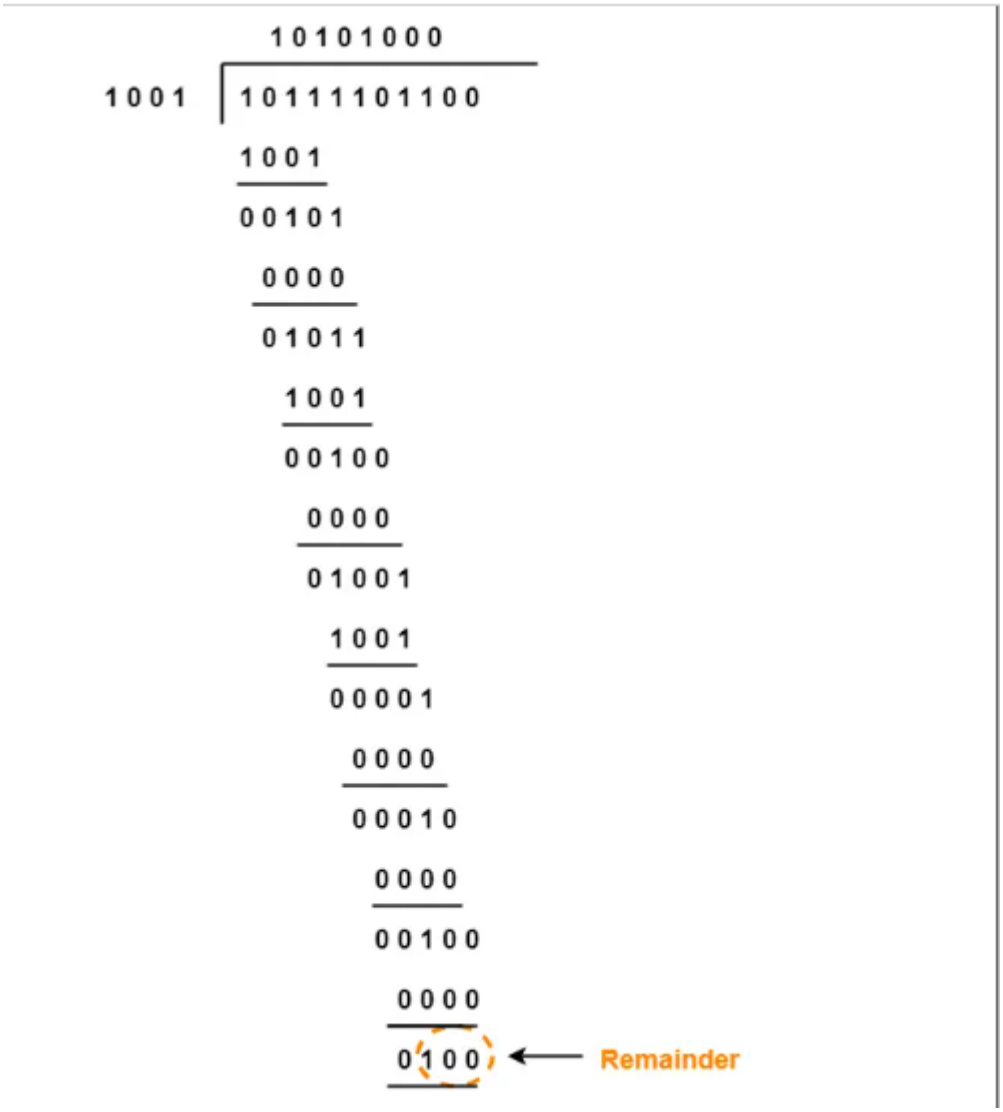
Part-02:

According to the question,

- Third bit from the left gets inverted during transmission.
- So, the bit stream received by the receiver = 10111101100.

Now,

- Receiver receives the bit stream = 10111101100.
- Receiver performs the binary division with the same generator polynomial as-



From here,

- The remainder obtained on division is a non-zero value.
- This indicates to the receiver that an error occurred in the data during the transmission.
- Therefore, receiver rejects the data and asks the sender for retransmission.