**Assignment :- CSE306 (Computer Networks)**

**Set-1**

**M. Marks - 30**

**Name : Ashwani Kumar Section: K18PG Registration No: 11806645 Roll No: B-38**

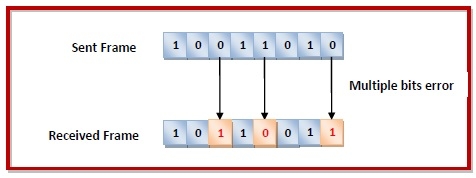
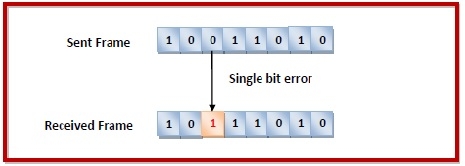
**PART-A**

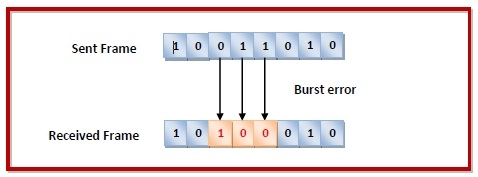
**Question 1.)** **Is the data link layer incapable of detecting an error in a received frame?** **(5 Marks)**

**Answer :- No,** Data Link Layer has an error control technique which can ensure that frames having control of bit streams of data which can be transfer from the source to the destination with a certain extent of utmost accuracy and reliable delivery of data or frame. So, The bits are transferred over the computer network, they tend to get corrupted because of interference and network issues. The corrupted or distorted bit leads to spurious or inconsistent data which is being received by the destination and are called errors.

## **Types of Errors :-**

* **Single bit error** :- In this Received Data have only one bit which is found to be corrupted, i.e., either 0 to 1 or from 1 to 0.



* **Multiple bits error** :- In this Received Data have More than one bits which are corrupted.
* **Burst error** :- In this Received Data have more than one simultaneous/consequent bits are corrupted.  
  

## **Error Control :-**

Error control can be done in two ways :-

* **Error detection** :- Error detection is to check whether any error has occurred or not.
* **Error correction** :- Error correction is to correct and specify the exact number of bits that has been corrupted or distorted and it’s location of the corrupted bits.

In both error detection and error correction, the sender needs to transfer some parity or redundant bit along with the data bits. The receiver has to checks based upon the additional parity bits. If it found that the data is error-free, then it has to remove the parity bits before delivering the message to the upper layers.

## **Error Detection Techniques :-** There are three main techniques for detecting errors in frames are as follows :-

1. Parity Check

## Checksum

## Cyclic Redundancy Check (CRC)

## **Error Correction Techniques :-**

The main error correction codes are :-

* Hamming Codes

Hence, The data link layer ensures error free link for data transmission. The issues it caters to with respect to error control for dealing with transmission errors, sending acknowledgement frames in reliable connections and retransmitting of the lost frames at receiver side.

**One Example of Hamming Code for detection and correction of received frame :-**

**In this, One formula is to find give the size of the frame :-**

* **2^p >= m+p+1**

**Where** p is for parity bit, and

m is for data bit

So when m = 4,

Then p = 3,

**When putting in formula,**

* 2^p >= m+p+1
* 2^3 >= 4+3+1
* 8 => 8 **(Condition is True)**

**Hence, The Size of the frame will be 7 bits.**

**L -> R (Left to Right) Even Parity Hamming Code Generation :-**

|  |  |  |  |
| --- | --- | --- | --- |
|  | **P4** | **P2** | **P1** |
| **0** | **0** | **0** | **0** |
| **1** | **0** | **0** | **1** |
| **2** | **0** | **1** | **0** |
| **3** | **0** | **1** | **1** |
| **4** | **1** | **0** | **0** |
| **5** | **1** | **0** | **1** |
| **6** | **1** | **1** | **0** |
| **7** | **1** | **1** | **1** |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **P1** | **P2** | **D3** | **P4** | **D5** | **D6** | **D7** |

**1 2 3 4 5 6 7**

P1 = D3, D5, D7

P2 = D3, D6, D7

P4 = D5, D6, D7

**Now At Sender Side (Tx), We need to set the value of Parity Bit (P1, P2 and P4) if the data bit is 1101 :-**

**MSB**

**LSB**

Since, it is for even parity, we get

P1 (D3, D5, D7) = 101 = 0; P2 (D3, D6, D7) = 111 = 1; P4 (D5, D6, D7) = 011 = 0

**Tx (Sender Side) =**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **0** | **1** | **1** | **o** | **0** | **1** | **1** |

**P1 P2 D3 P4 D5 D6 D7**

**Now we detect and correct the error bit :-**

**Now Suppose Data bit has changed to 1100 due to noise in the channel, then**

**Rx (Receiver Side) =**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **0** | **1** | **0** | **0** | **0** | **1** | **1** |

**P1 P2 D3 P4 D5 D6 D7**

Since it is by Even Parity,

**We solve to find the value of P1, P2 and P4 is correct or not and if it is not then we need to find the location of error bit and correct it :-**

P1 (P1, D3, D5, D7) = 0001 = 1 (Changed to 1 as The P1 value is not correct);

P2 (P2, D3, D6, D7) = 1011 = 1 ( Changed to 1 as The P2 value is not correct);

P4 (P4, D5, D6, D7) = 0011 = 0 (No change as The P4 value is correct);

**Hence, The location of error bit is 110 which 3 i.e., D3 has been changed due to noise.**

**So we correct it,**

**Rx (Receiver Side) =**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **0** | **1** | **1** | **0** | **0** | **1** | **1** |

**P1 P2 D3 P4 D5 D6 D7**

**Hence, This is how Data Link Layer detect and correct the errors in received frame.**

**Question 2.)** **Is flow control one of the services provided by the link layer?** **(5 Marks)**

**Answer :- Yes,** The data link layer can regulates flow control mechanism. So that A fast sender does not drown a slow receiver. When the sender transfer the frames at a very high speeds, a slow receiver may not be able to control it. It means that This is a potential problem, as a receiving host may receive frames at the rate faster than it can process the frames over given time interval. There will be loss of frame even if the transmitting data is error-free. Since, The nodes on each side of the link have a limited amount of payload buffering capacity. Without flow control, the receiver's buffer get overflow and frames will get lost.

**The two common approaches for flow control are :-**

**Feedback based flow control :-** In this, Receiver can send the information to the sender after receiving a frame to send the frame again or send the same frame again if it was lost or if the frame was received successfully.

**Types of Feedback based Flow Control :-**

1. **Stop and Wait ARQ,**
2. **Go-Back-N ARQ, and**
3. **Selective Repeat ARQ,**

**Note :- ARQ : Automatic Repeat Request**

**Rate based flow control :-** In this technique, we limit the rate at which sender may transmit data, without using feedback from receiver. All Congestion Techniques come under it.

**PART-B**

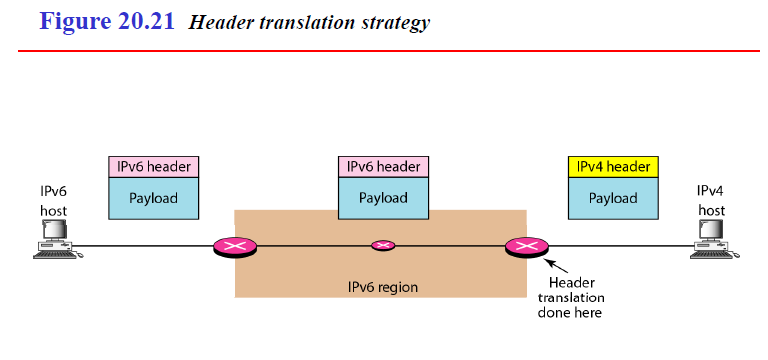
**Question 3.)** **If a source is using IPv4 address and Destination has IPv6? Were they able to transfer the data?** **(5 Marks)**

**Answer :- Yes,** The transition need not require any global coordination. Our sites and Internet service provider (ISP) can do transition at their own speed. But, Compatibility is also required for existing applications during the transition. Complete transition from IPv4 to IPv6 might not be possible because IPv6 is not backward compatible with IPv4, because these are two different protocols. Therefore, it is impossible to switch the entire internet over to IPv6 overnight. Hence, The pragmatic approach is to make these two version work side by side and this may last long even a decade and for that a transition mechanism is needed to bridge between IPv4 and IPv6. So, To overcome this short-coming, we have a few strategies that can be used to ensure slow and smooth transition from IPv4 to IPv6. So, If a source is using IPv4 and destination has IPv6 then we can transfer data by the following three transition strategies :-

**Dual Stack Strategy :-** In this, Header include both source and destination address in both IPv4 and IPv6 format In this technique, Node with IP stack will contain both IPv4 and IPv6. So, while communicating with IPv6 node, IPv6 part is used and same when communicating with IPv4 node. Source Host queries the DNS and if DNS server responds with a IPv4 address then source respond IPv4 packet and if DNS return IPv6 then source sends an IPv6 packet.

**Tunnel Strategy :-** In this, we have channel/tunnel having two router i.e., starting and ending router by which we can transmit data to two different host. This technique is implemented when different IP version exist on intermediate path. Suppose two IPv6 node want to communicate via IPv4 routers then IPv6 packets are placed inside Ipv4 packets as shown in the diagram and it holds true for IPv4 where the transit network is IPv6.

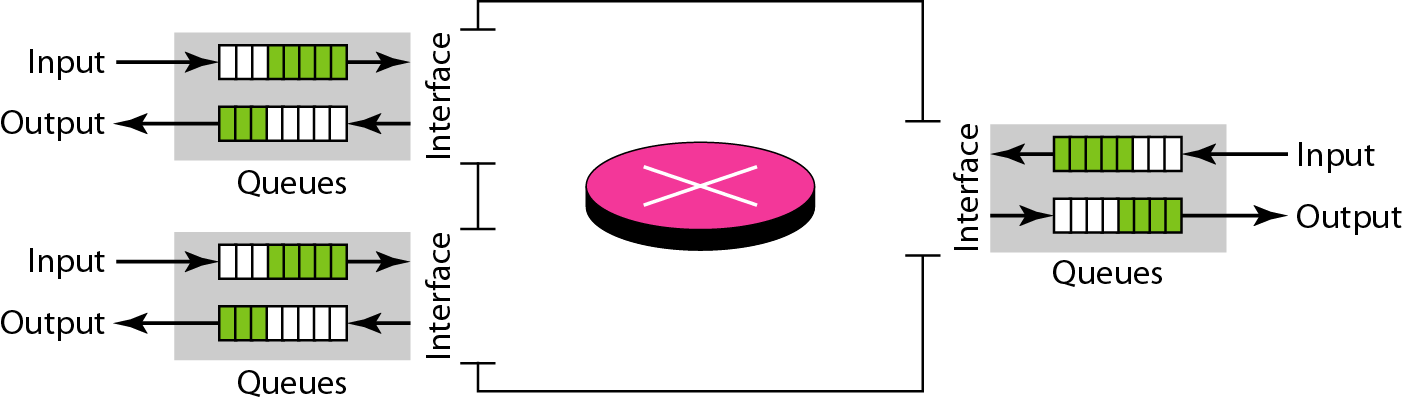
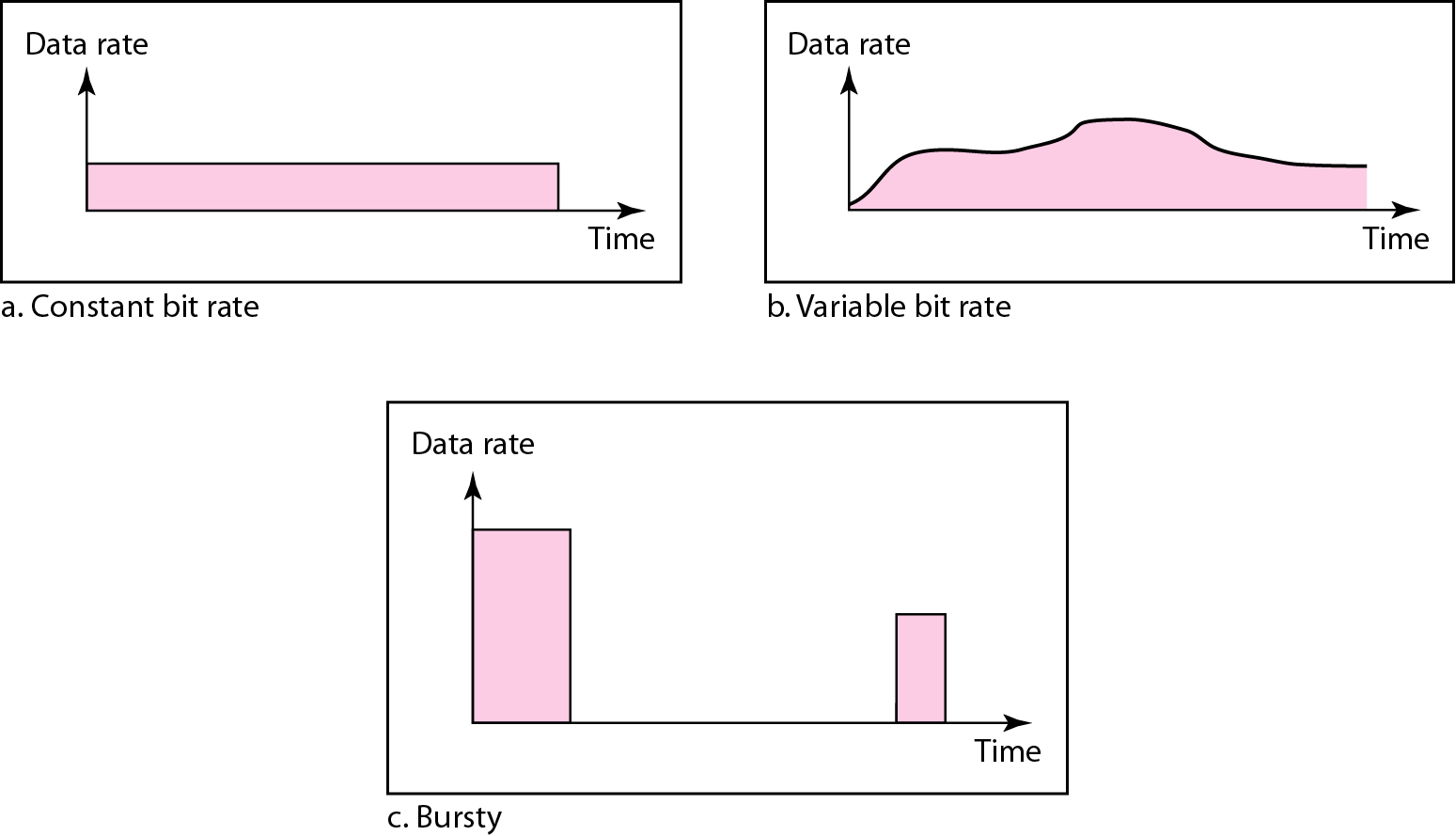
**Header Translation Strategy :-** In this, We have routers which act as a transmitter for us to transfer the data to different host. In this strategy, it is necessary when the majority of the internet has moved to IPv6 but some systems still uses IPv4. In this, sender uses IPv6 and receiver uses IPv4. So, Tunneling won’t help because receiver only understands IPv4. So, Header format is changed. It is converted from IPv6 to IPv4 through Header Translation by mapped address.

****

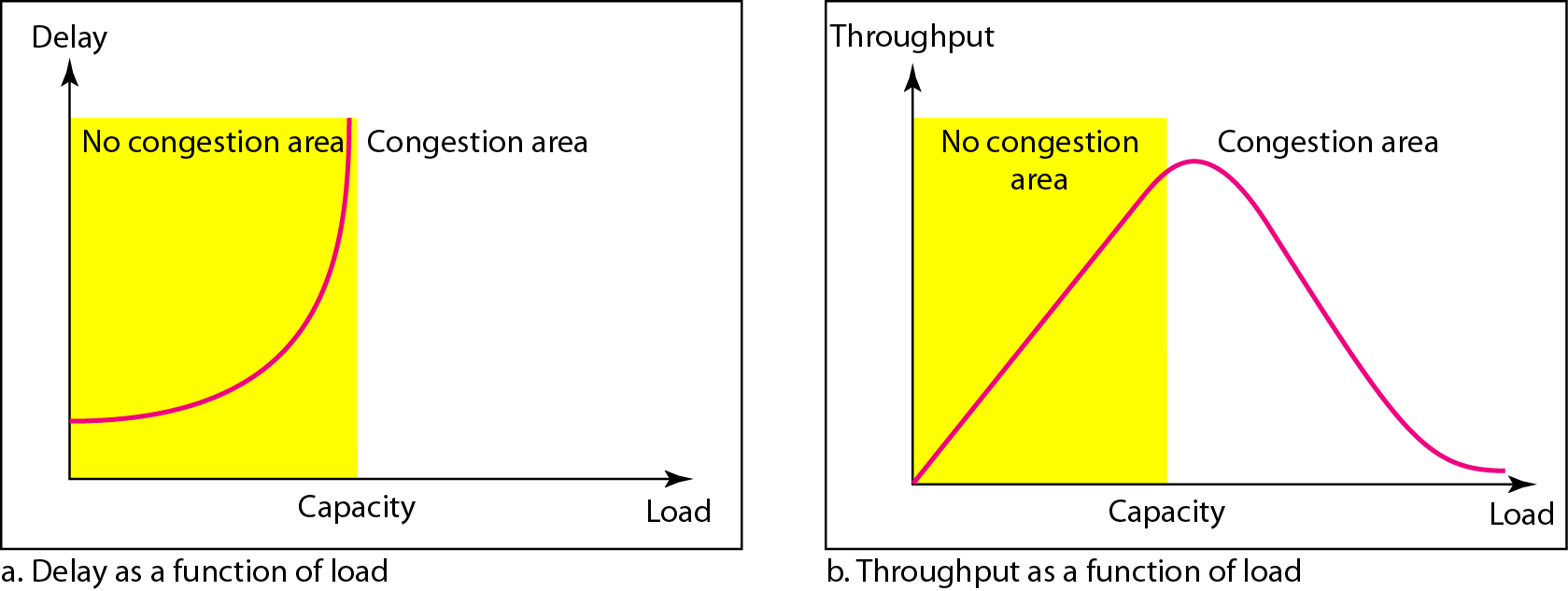
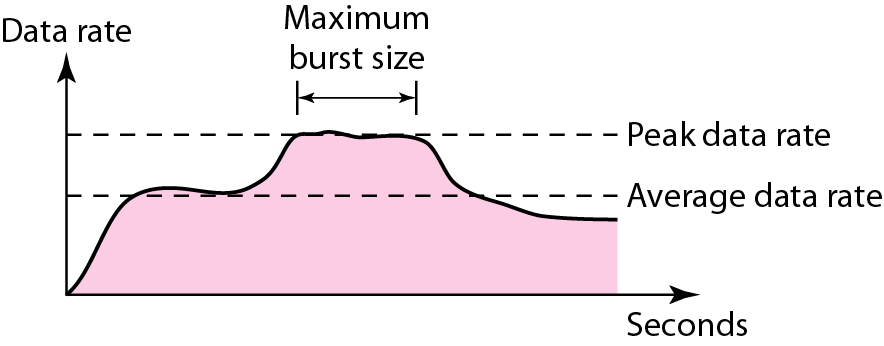
**In this way, we can transfer the data from IPv4 Source to IPv6 Destination.**

**Question 4.)** **Is congestion affect the overall performance of the network? Mention different congestion control techniques.** **(5 Marks)**

**Answer :-** **Yes,** Congestion affect the overall performance of the network because it leads huge data traffic on the network. It is due to the rate of input traffic which exceeds the limit of the output lines. So when suddenly a stream of data packet start arriving on three or four input lines and since they all need the same output line because of which a queue will be generated and if there is insufficient memory to hold all the data packets will lead to loss of the data packet. Even though, if we increase the memory to unlimited size will not solve the problem. It is because the time packets reached the front of the queue, they have been already timed out as they waiting in the queue. So, when timer goes off source sends duplicate packet that are also added to the queue. Hence, Same data packets are added again and again, which is increasing the load all the way on the transmitting line between the source and the destination.

******Queues in a Router Three Data Traffic Profiles**

In the given figure, As Congestion in a subnet can also occur, if the routers are Low. Furthermore, Low speed CPU at routers will perform the routine tasks such as queuing buffers and updating table very slowly. The router’s buffer is becomes incapable due it’s excess limiting capacity.

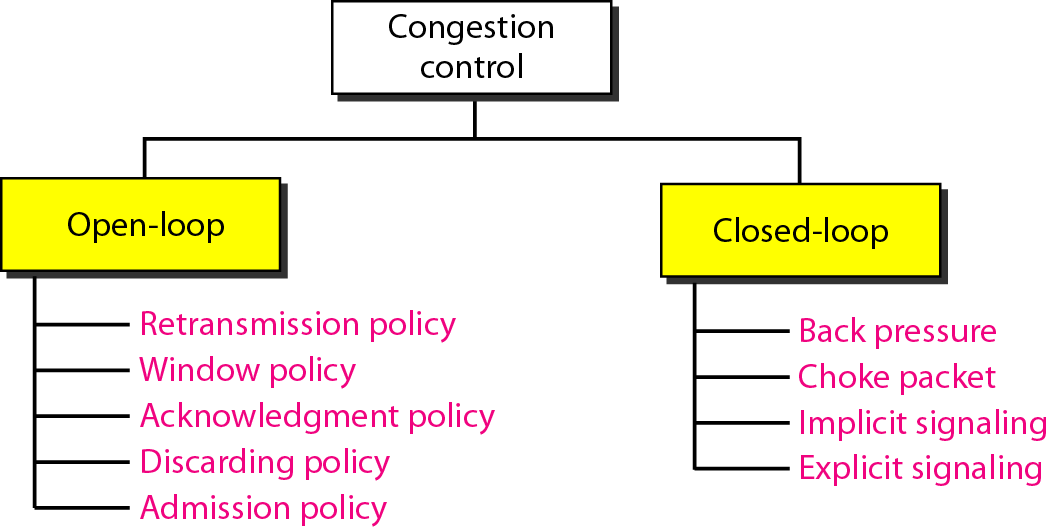


**Traffic Descriptor Network Performance: Delay and Throughput as Functions of Load**

Hence, Multiple transmissions of data packets will tend the congestion to take place at the sender side. Thus, Congestion in a network can occur if the loading on the network i.e., the number of data packets sent to the network i.e., is greater than the capacity of the network i.e., the number of data packets a network may hold.

So, To overcome congestion in data traffic,We need Congestion control techniques which refers to the mechanisms and techniques to control the congestion and keep the load below the capacity. It can either **prevent congestion**, before it can happen, or **remove congestion**, after it has happened. In general, we can divide congestion control techniques into two broad categories :-

1. **Open-loop congestion control (**Prevention**) and**
2. **Closed-loop congestion control (**Removal**)**



## **Open Loop Congestion Control :-**

• In this mechanism, policies has been used to prevent the congestion before it can happen.

• Congestion control has been handled either by the source or by the destination.

• The various methods are discussed below:-

### Retransmission Policy :-

• The sender retransmits a data packet, if he feels that the packet he has sent is lost or corrupted.

• However when retransmission can generally may increase the congestion in the network. But we need to implement well defined retransmission policy to prevent congestion.

•In retransmission policy, the retransmission timers has to be designed to optimize efficiency and prevent the congestion at the same time.

### Window Policy :-

• When we implement window policy, then selective reject window method has to be used for congestion control.

• Selective Reject method has been preferred over Go-back-N ARQ window since Go-back-N ARQ method, in which timer for a data packet is time out, several data packets has to be resent, although a few may have arrived safely at the receiver. Hence, This duplication or replication will make congestion worse.

• Selective reject method can sends only the specific lost or damaged or infected data packets.

### Acknowledgement Policy :-

• In Acknowledgement Policy can be imposed by the receiver which may also affect or control congestion or data traffic.

• When the receiver does not acknowledge about every data packet it has received it may slow down or inactivate or acknowledge the sender to stop or slow down from transmitting data packets and will help to prevent congestion.

• Acknowledgments is also add to the traffic load on to the network. Hence, By sending lesser acknowledgements, we will reduce load on the network.

• To implement it, several strategies can be followed :-

1. A receiver can send an acknowledgement only if he has a data packet to be sent.

2. A receiver can send an acknowledgement only when a timer expires.

3. A receiver could also decide to acknowledge only N number of data packets at a time.

### Discarding Policy :-

• A router can discard few sensitive data packets when congestion is likely to happen.

• So, When discarding policy will prevent congestion and it can’t harm the integrity of the transmission at the same time.

### Admission Policy :-

• In an Admission Policy where also found to have a quality of service(QoS) mechanism, which will also prevent congestion in the virtual circuit networks.

• In this, Switches in a flow will it first check the resource requirement of a flow before admits it to the network.

• In this, A router may deny establishing a virtual circuit connection if there will be congestion in the network or if there is a probability or chances of future congestion to occur.

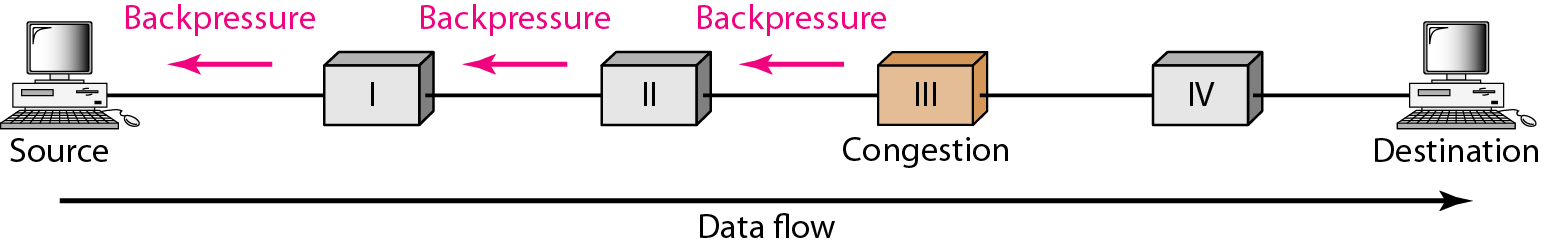
## **Closed Loop Congestion Control :-**

• Closed Loop Congestion Control mechanisms is implemented when we try to remove the congestion after it has happened.

• The various methods are discussed below :-

### Backpressure :-

• Backpressure is a node-to-node congestion control that begin with a node and propagates in the opposite direction of data flow.



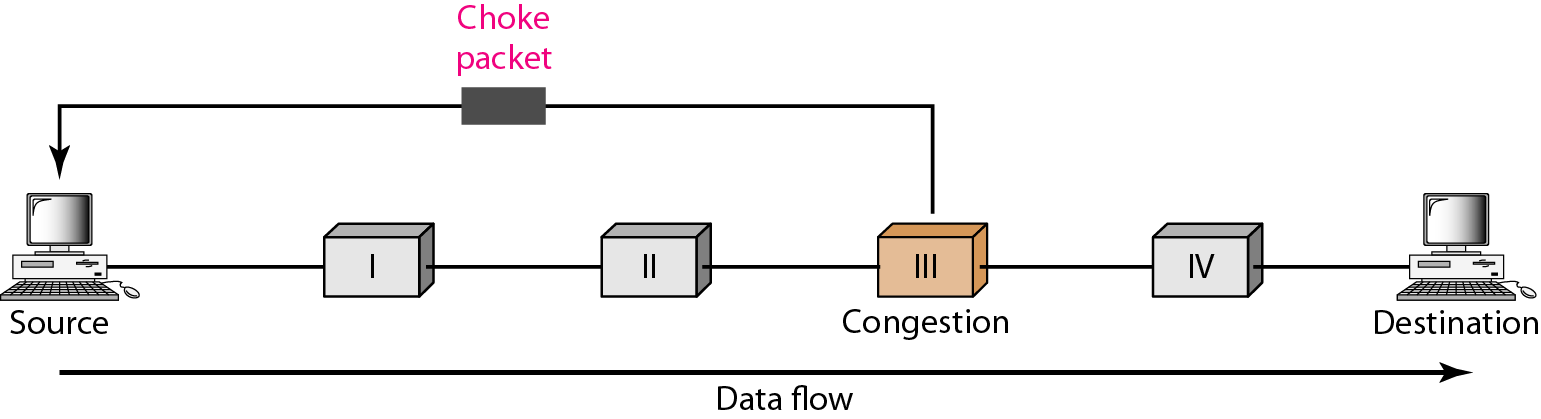
• In this method, the congested node will stop receiving data from the immediate upstream nodes.

• This will cause the upstream node on nodes to become congested and they in turn reject data packet from their upstream nodes.

• As shown in the above figure, Node III is congested and it stops receiving data packets and informs its upstream node II to slow down. Node II in turns will be congested and informs node I to slow down. Now node I will create congestion and informs the source node to slow down. In this way the congestion is alleviated or removed. Thus, The pressure on node III has been moved backward to the source to remove the congestion.

### Choke Packet :-

• In this method, Congested node will send a special type of packet called choke packet to the source and inform it about the congestion.



• Here, Congested node will not inform its upstream nodes about the congestion as in backpressure method.

• In this method, Congested node will send a warning directly to the source station that is the intermediate nodes through which the data packet has been transferred are not warned.

### Implicit Signaling :-

• In this method, There will be no communication between the congested nodes and the source.

• The source will guess that there is a congestion anywhere in the network when it does not receive any acknowledgment. Hence, The delay in receiving an acknowledgment has to be interpreted that is congestion in the network.

• On sensing that there is a congestion, the source will slow down.

• This type of congestion control policy is used by TCP.

### Explicit Signaling :-

• In this mechanism, the congested nodes can explicitly send a signal to the source or destination to inform about the congestion.

• Explicit signaling is different from the choke packet method. In choke packed method, a specific packet is used for this purpose whereas in the explicit signaling method, the signal has been included in the packets that can carry data .

• Explicit signaling can occur in two ways as follows :-

**• In backward signaling**, in which a bit will be set in a packet moving in the direction opposite to the congestion. This bit can also warn the source about the congestion and informs the source to slow down.

**• In forward signaling,** in which a bit will be set in a packet moving in the direction of congestion. This bit can also warn the destination about the congestion. So, The receiver in this case can use policies such as slowing down the acknowledgements as to remove the congestion.

**PART-C**

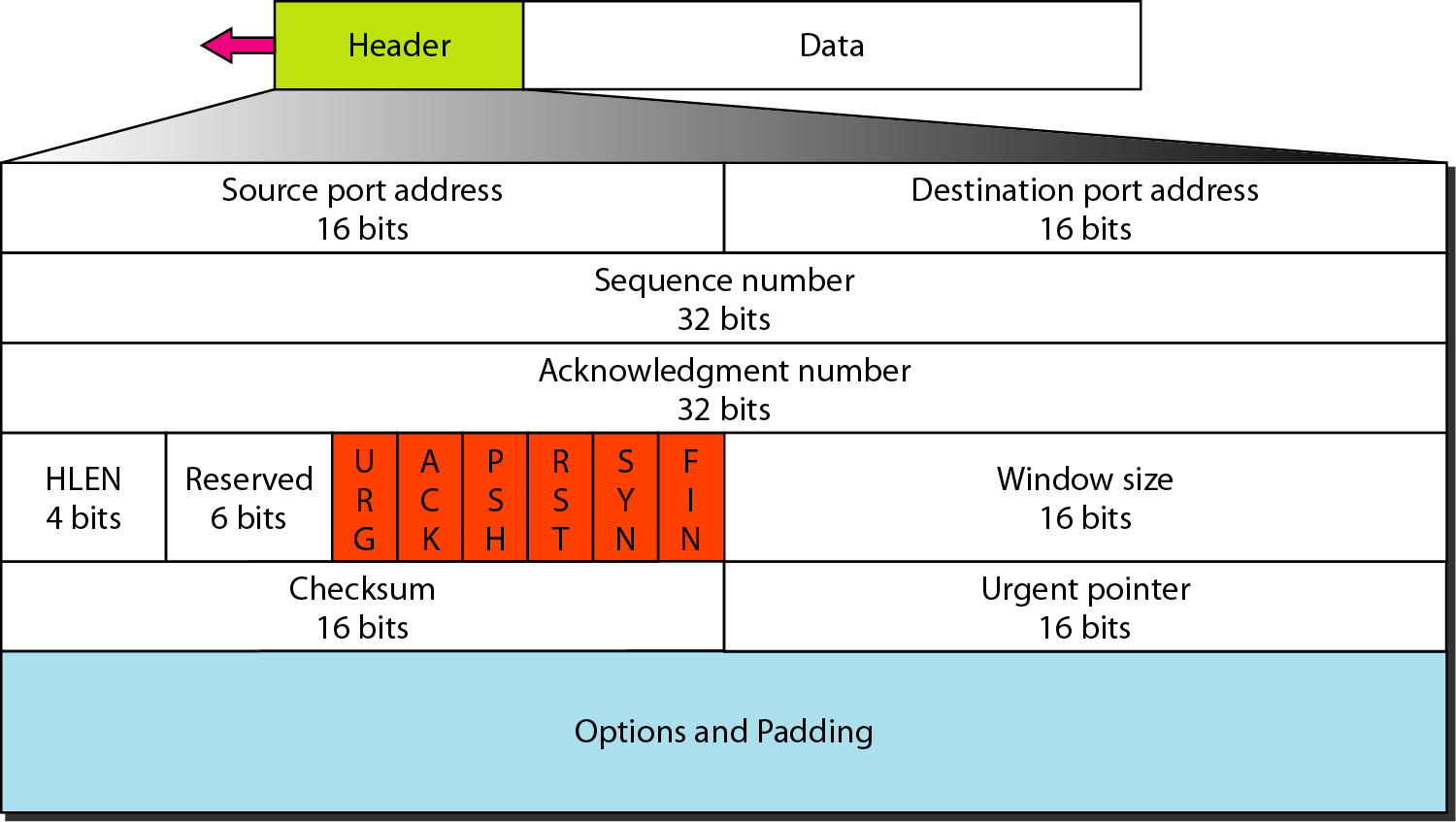
**Question 5.)** **What are the main differences between TCP and UDP? (5 Marks)**

**Answer :-**

|  |  |
| --- | --- |
| **UDP**  **(User Datagram Protocol)** | **TCP**  **(Transmission Control Protocol)** |
| 1. It is a Connectionless-Oriented Protocol. | 1. It is a Connection-Oriented Protocol. |
| 1. Less Reliability i.e., it doesn’t assure reliable delivery of data to the destination. | 1. More Reliability i.e., it assures reliable delivery of data to the destination. |
| 1. Error Control Mechanism is optional i.e., it doesn’t provides error checking mechanism as flow control and acknowledgement of data. So, It also doesn’t have any flag for acknowledgment. | 1. Error Control Mechanism is compulsory i.e., it provides extensive error checking mechanism such as flow control and acknowledgement of data. So, It has various flag for acknowledgement. |
| 1. Fast Transmission i.e., it makes fast service to transmit data. | 1. Slow Transmission i.e., it is comparatively slow because of these extensive error checking mechanism. |
| 1. Less Overhead. | 1. More Overhead. |
| 1. Delivery of data is not guaranteed if you are using UDP. | 1. Delivery of data is guaranteed if you are using TCP. |
| 1. No Congestion Control. | 1. Congestion Control is there. |
| 1. There is no retransmission of lost packets in UDP. | 1. Retransmission of lost packets is possible in TCP. |

**Question 6.)** **Explain the Header format of TCP with all fields. Also, mention the reason for using Flag bits in TCP.** **(5 Marks)**

**Answer :-** The header format of a TCP is given below :-



**The reason of having flag in TCP** is that it is connection-oriented protocol which need to have reliable delivery of data segment and for request and acknowledgment, it needs various flag bits as given below :-

1. **Source Port Address :-** It is of 16 bit which holds the port address of the application which is transferring the data segment.
2. **Destination Port Address :-** It is of 16 bit which holds the port address of the application in the receiving node which is receiving the data segment.
3. **Sequence Number :-** It is of 32 bit which holds the serial or sequence number that is the byte number of the starting byte which is sent in that particular segment. It is used to aggregate or reframe the message at the receiver side if the segments are received out in disorder.
4. **Acknowledgement Number :-**  It is of 32 bit which holds the acknowledgement number that is the byte number which the receiver predicts to receive next. It is also an acknowledgment for the earlier data has been received or not. Also, The acknowledgment number is generally cumulative.
5. **Header Length (HLEN) :-** This has a size of 4 bit which indicates the size of the TCP header by number of 4-byte words in the header.
6. **Control flags :-** These are of total 6 Bits. The function of the flag is given below :-
   1. URG: Urgent pointer is valid.
   2. ACK: Acknowledgement number is valid.
   3. PSH: Request for push.
   4. RST: Reset the connection.
   5. SYN: Synchronize sequence numbers.
   6. FIN: To Finish or Terminate the connection.

In this, Each bit is a control bit that control establishing the connection, terminating the connection, aborting the connection, flow control that means acknowledgement of received segment, and transmitting mode.

1. **Window Size :-** This field is to define or specify the window size of the sending TCP in bytes.
2. **Checksum :-** This field is to holds the checksum for error control. It is mandatory or compulsory in TCP not in UDP.
3. **Urgent Pointer :-** This field is valid only if the URG control flag is set and it is used to point to data which is urgently required and needs to reach the receiving process as soon as possible. The value of this field is added to the sequence number to get the byte number of the previous urgent byte.

**Plagiarism Report :-**

