CMR
INSTITUTE OF
TECHNOLOGY

USN | | | | | | | | | |

Internal Assesment Test - II

Sub:	Computer Networks-1							Code:	10CS55
Date:	04 / 11 / 2016	Duration:	90 mins	Max Marks:	50	Sem:	V	Branch:	CSE

Answer Any FIVE FULL Questions

		Marks	OBE CO	OBE RBT
1(a)	What is Time Division Multiplexing? Explain how statistical TDM overcomes the disadvantages of synchronous TDM	[04]	CO1	L2,L3
(b)	If we have four sources, each creating 250 characters /sec. If the interleaved unit is one character and 1 synchronization bit is added to each frame. Find, i) The data rate of each source ii) The duration of each character in each source iii) The frame rate iv) The duration of each frame v) The number of bits in each frame vi) The data rate of the link	[06]	CO1	L4
2	Explain selective repeat ARQ. Justify how selective repeat ARQ outperforms Go-Back-N and Stop-and-Wait ARQ	[10]	CO2	L2,L3
3	Explain the working of HDLC protocol with relevant frame formats	[10]	CO2	L2
4	Explain the working of Bit and Byte oriented framing protocols	[10]	CO2	L2
5	Explain following random access protocols i). Slotted ALOHA ii) CSMA/CA	[10]	CO3	L2
6	What do you mean by channelization? Explain the working of CDMA protocol with an example	[10]	CO3	L1,l2
7	Write short note on, i). Fast Ethernet ii). Gigabit Ethernet	[10]	CO3	L1

Course Outcomes		PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1:	Understand two dominant network models Knows as OSI Reference model (a theoretical framework) and the Internet Model (The model used in today's Data Communications)	1											
CO2:	Define the types of data signals, their properties and the and the methods of signal transmission	2	2										
CO3:	Construct efficient codes for data on impaired communication channels.	1	2	3	1								
CO4:	Describe the various Link Control and Access control strategies for Noiseless and Noisy Channels.	3	2										
CO5:	Differentiate between the Standards and protocols for Wired and Wireless LANs.	1	2				2						
CO6:	Differentiate between the implementation of IPV4 and IPV6 Internet addressing schemes.	1					2						

Cognitive level	KEYWORDS
L1	List, define, tell, describe, identify, show, label, collect, examine, tabulate, quote, name, who, when, where, etc.
L2	summarize, describe, interpret, contrast, predict, associate, distinguish, estimate, differentiate, discuss, extend
L3	Apply, demonstrate, calculate, complete, illustrate, show, solve, examine, modify, relate, change, classify, experiment, discover.
L4	Analyze, separate, order, explain, connect, classify, arrange, divide, compare, select, explain, infer.
L5	Assess, decide, rank, grade, test, measure, recommend, convince, select, judge, explain, discriminate, support, conclude, compare, summarize.

PO1 - *Engineering knowledge*; PO2 - *Problem analysis*; PO3 - *Design/development of solutions*; PO4 - *Conduct investigations of complex problems*; PO5 - *Modern tool usage*; PO6 - *The Engineer and society*; PO7- *Environment and sustainability*; PO8 – *Ethics*; PO9 - *Individual and team work*; PO10 - *Communication*; PO11 - *Project management and finance*; PO12 - *Life-long learning*

Solutions

1a. Time-division multiplexing (TDM) is a digital process that allows several connections to share the high bandwidth of a link. Instead of sharing a portion of the bandwidth as in FDM, time is shared.

Statistical time division multiplexing.

- Asynchronous TDM is called so because is this type of multiplexing, time slots are not fixed *i.e.* the slots are flexible.
- Here, the total speed of input lines can be greater than the capacity of the path.
- In synchronous TDM, if we have *n* input lines then there are *n* slots in one frame. But in asynchronous it is not so.
- In asynchronous TDM, if we have *n* input lines then the frame contains not more than *m* slots, with *m* less than *n* $(m < n)$.
- In asynchronous TDM, the number of time slots in a frame is based on a statistical analysis of number of input lines.
- In this system slots are not predefined, the slots are allocated to any of the device that has data to send.
- The multiplexer scans the various input lines, accepts the data from the lines that have data to send, fills the frame and then sends the frame across the link.
- If there are not enough data to fill all the slots in a frame, then the frames are transmitted partially filled.
- Multiplexer scans the various lines in order and fills the frames and transmits them across the channel.

1 b. i). Data rate of each source=2000bps

ii). Duration of each character in source=4ms

iii). Frame rate=250 frames/second

iv). Duration of output frame=4ms

v). Frame size=33bits

vi). Data rate of the link=8250bps

2 a. Working of Selective repeat ARQ:

For noisy links, there is a 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. It is more efficient for noisy links, but the processing at the receiver is more complex.

Basis for Comparison	Go-Back-N	Selective Repeat	Stop and Wait
Basic	Retransmits all the frames that sent after the frame which suspects to be damaged or lost.	Retransmits only those frames that are suspected to lost or damaged.	Retransmits the frame once the timer Expires.
Bandwidth	If error rate is high, it wastes a lot	Comparatively less bandwidth	Wastes a lot of

High-level Data Link Control (HDLC) is a bit-oriented protocol for communication over point-to-point and multipoint links. It implements the ARQ mechanisms we discussed in this chapter. Configurations and Transfer Modes HDLC provides two common transfer modes that can be used in different configurations:

normal response mode (NRM) and asynchronous balanced mode (ABM).

Normal Response Mode

In 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.

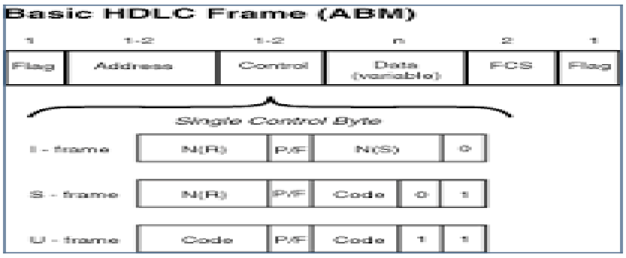
Basis for Comparison	**Go-Back-N**	**Selective Repeat**	**Stop and Wait**
Utilization	of bandwidth.	is wasted in retransmitting.	bandwidth
Complexity	Less complicated.	More complex as it require to apply extra logic and sorting and storage, at sender and receiver.	Simple to implement
Window size	N-1	<= (N+1)/2	Not using window concept
Sorting	Sorting is neither required at sender side nor at receiver side.	Receiver must be able to sort as it has to maintain the sequence of the frames.	Sending of frames in sequential order.
Storing	Receiver do not store the frames received after the damaged frame until the damaged frame is retransmitted.	Receiver stores the frames received after the damaged frame in the buffer until the damaged frame is replaced.	No sorting is needed
Searching	No searching of frame is required neither on sender side nor on receiver	The sender must be able to search and select only the requested frame.	-
ACK Numbers	NAK number refer to the next expected frame number.	NAK number refer to the frame lost.	NAK number refer to the next expected frame number.
Use	It more often used.	It is less in practice because of its complexity.	Not used

3. Working of HDLC

Asynchronous Balanced Mode

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 (acting as peers)

Frames Format:



To provide the flexibility necessary to support all the options possible in the modes and configurations just described, HDLC defines three types of frames: information frames (I-frames), supervisory frames (S-frames), and unnumbered frames (V-frames). Each type of frame serves as an envelope for the transmission of a different type of message.

I-frames are used to transport user data and control information relating to user data (piggybacking). S-frames are used only to transport control information. V-frames are reserved for system anagement. Information carried by V-frames is intended for managing the link itself.

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. In multiple-frame transmissions, the ending flag of one frame can serve as the beginning flag of the next frame.

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 used to define these functions. 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. Note that with 3 bits, we can define a sequence number between and 7; but in the extension format, in which the control field is 2 bytes, this field is larger. 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* bit. The *PIP* field is a single bit with a dual purpose. It has meaning only when it is set (bit = 1) and can mean poll or final. 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).

Control Field for S-Frames

Supervisory frames are used for flow and error control whenever piggybacking is either impossible or inappropriate (e.g., when the station either has no data of its own to send or needs to send a command or response other than an acknowledgment). 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) depending on the type of S-frame. The 2 bits called code is used to define the type of S-frame itself.

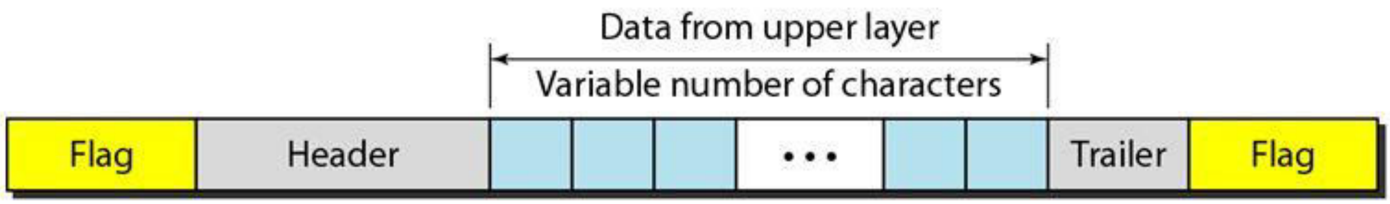
Control Fieldfor U-Frames

Unnumbered frames are used to exchange session management and control infonnation between connected devices. Unlike S-frames, U-frames contain an information field, but one used for system management information, not user data. As with S-frames, however, much of the infonnation carried by U-frames is contained in codes included in the control field. U-frame codes are divided into two sections: a 2-bit prefix before the PtF bit and a 3-bit suffix after the PtF bit. Together, these two segments (5 bits) canbe used to create up to 32 different types of U-frames.

4. Byte-oriented Framing protocol:

In a character-oriented protocol, data to be carried are 8-bit characters from a coding system such as ASCII (see Appendix A). The header, which normally carries the source and destination addresses and other control information, and the trailer, which carries error detection or error correction redundant bits, are also multiples of 8 bits.

To separate one frame from the next, an 8- bit (I-byte) flag is added at the beginning and the end of a frame. The flag, composed of protocol-dependent special characters, signals the start or end of a frame. The following figure shows the format of a frame in a character-oriented protocol.

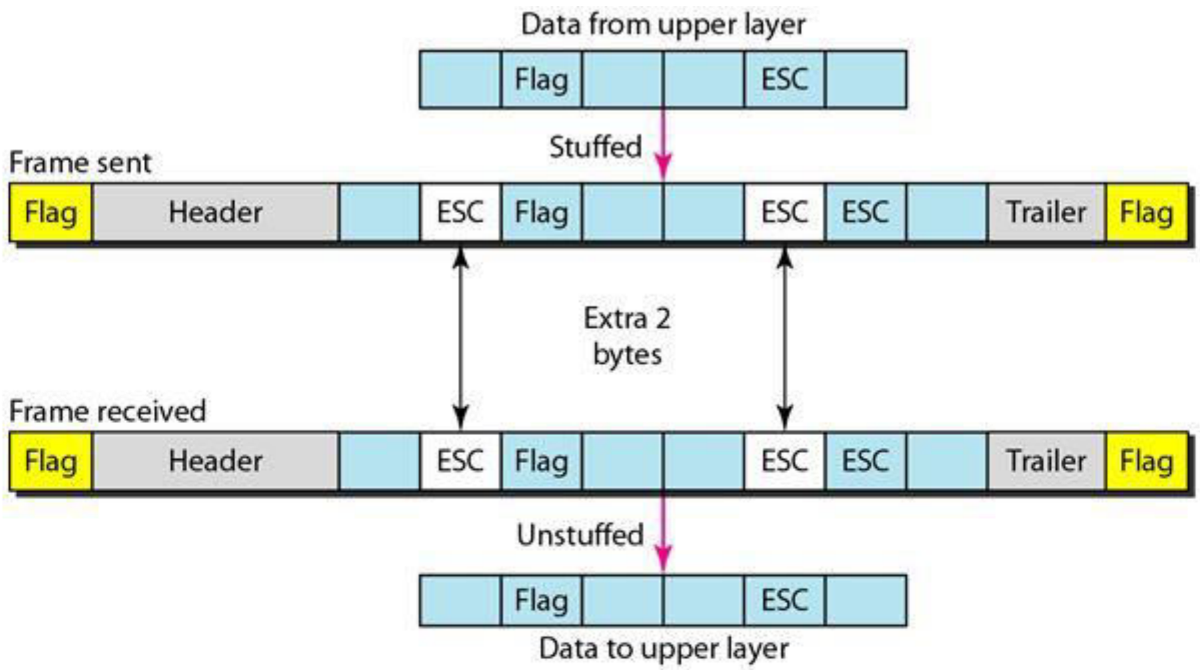


Character-oriented framing was popular when only text was exchanged by the data link layers. The flag could be selected to be any character not used for text communication.

However, if we send other types of information such as graphs, audio, and video. Any pattern used for the flag could also be part of the information. If this happens, the receiver, when it encounters this pattern in the middle of the data, thinks it has reached the end of the frame.

To fix this problem, a byte-stuffing strategy was added to character-oriented framing. In byte stuffing (or character stuffing), a special byte is added to the data section of the frame when there is a character with the same pattern as the flag. The data section is stuffed with an extra byte. This byte is usually called the escape character (ESC), which has a predefined bit pattern. Whenever the receiver encounters the ESC character, it removes it from the data section and treats the next character as data, not a delimiting flag.

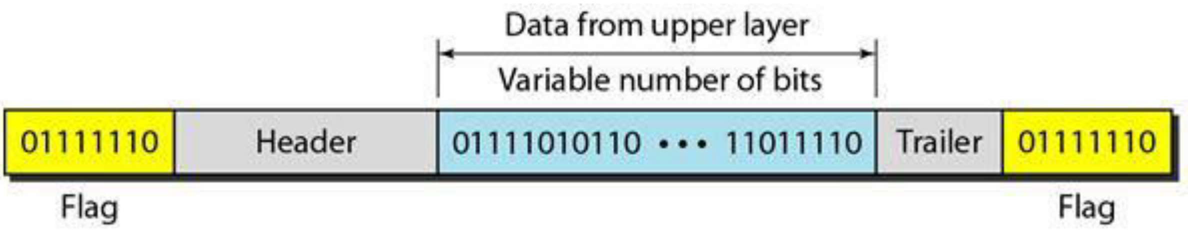
Byte stuffing by the escape character allows the presence of the flag in the data section of the frame, but it creates another problem. What happens if the text contains one or more escape characters followed by a flag? The receiver removes the escape character, but keeps the flag, which is incorrectly interpreted as the end of the frame. To solve this problem, the escape characters that are part of the text must also be marked by another escape character. In other words, if the escape character is part of the text, an extra one is added to show that the second one is part of the text. The following figure shows the situation.



Character-oriented protocols present another problem in data communications. The universal coding systems in use today, such as Unicode, have 16-bit and 32-bit characters that conflict with 8-bit characters. We can say that in general, the tendency is moving toward the bit-oriented protocols.

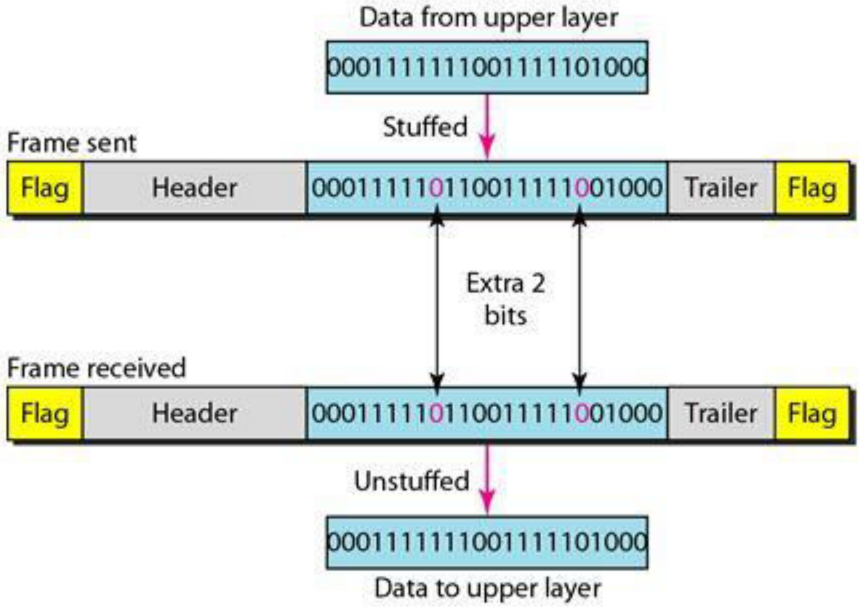
Bit-Oriented Protocols: In a bit-oriented protocol, the data section of a frame is a sequence of bits to be interpreted by the upper layer as text, graphic, audio, video, and so on. However, in addition to headers (and possible trailers), we need a delimiter to separate one frame from the other. Most protocols use a special 8-

bit pattern flag 01111110 as the delimiter to define the beginning and the end of the frame, as shown in the following figure.



This flag can create the same type of problem we saw in the byte-oriented protocols. That is, if the flag pattern appears in the data, we need to somehow inform the receiver that this is not the end of the frame. We do this by stuffing 1 single bit (instead of 1 byte) to prevent the pattern from looking like a flag. The strategy is called bit stuffing. In bit stuffing, if a 0 and five consecutive 1 bits are encountered, an extra 0 is added. This extra stuffed bit is eventually removed from the data by the receiver. Note that the extra bit is added after one 0 followed by five 1s regardless of the value of the next bit. This guarantees that the flag field sequence does not inadvertently appear in the frame.

The following figure shows bit stuffing at the sender and bit removal at the receiver. Note that even if we have a 0 after five 1s, we still stuff a 0. The 0 will be removed by the receiver.



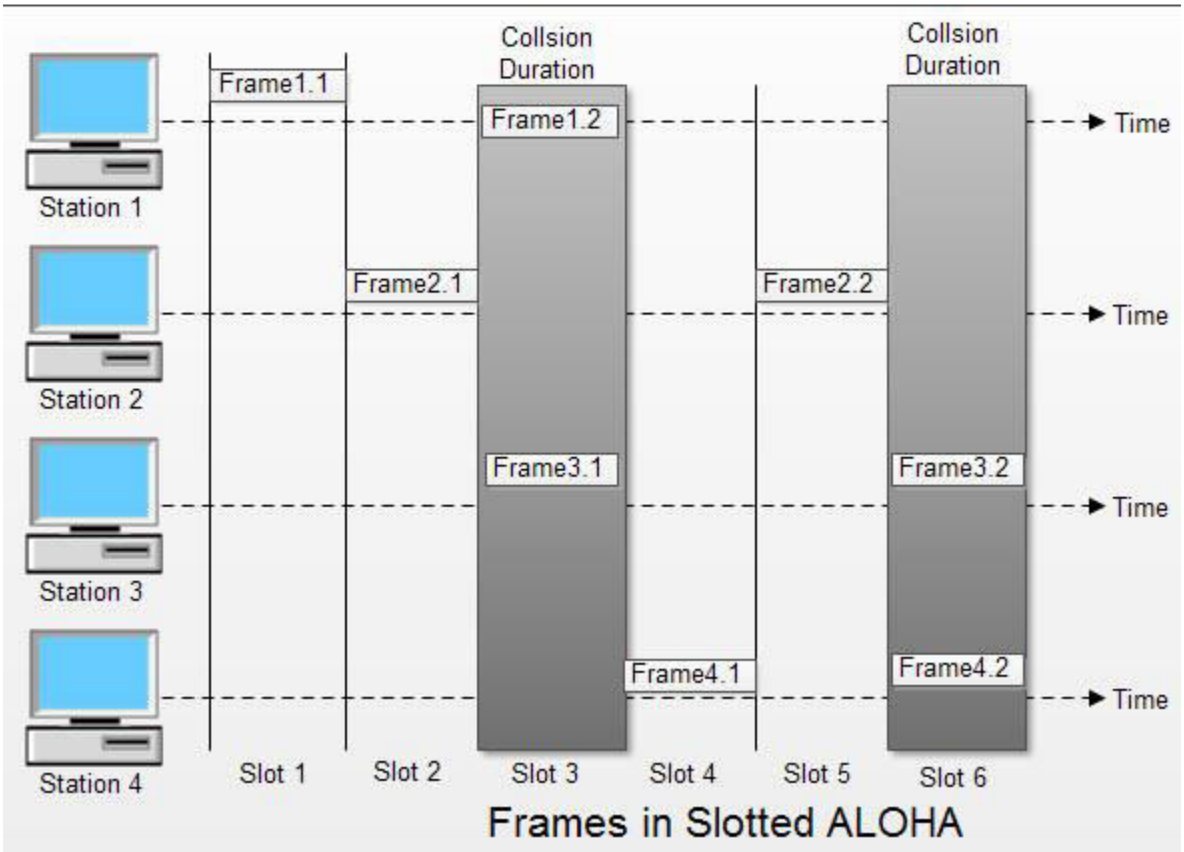
This means that if the flag like pattern 01111110 appears in the data, it will change to 011111010 (stuffed) and is not mistaken as a flag by the receiver. The real flag 01111110 is not stuffed by the sender and is recognized by the receiver.

5. a. Slotted ALOHA:

• Slotted ALOHA was invented to improve the efficiency of pure ALOHA as chances of collision in pure ALOHA are very high.

• In slotted ALOHA, the time of the shared channel is divided into discrete intervals called slots.

• The stations can send a frame only at the beginning of the slot and only one frame is sent in each slot.

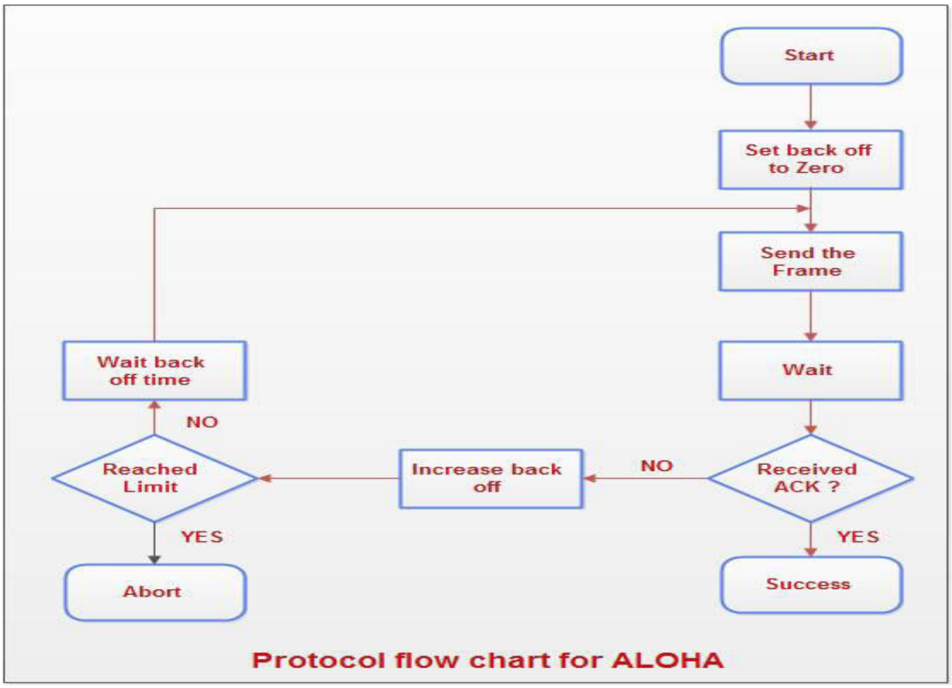


• In slotted ALOHA, if any station is not able to place the frame onto the channel at the beginning of the slot *i.e.* it misses the time slot then the station has to wait until the beginning of the next time slot.

• In slotted ALOHA, there is still a possibility of collision if two stations try to send at the beginning of the same time slot as shown in fig.

• Slotted ALOHA still has an edge over pure ALOHA as chances of collision are reduced to one-half.

Flow Chart for Slotted ALOHA:



A station which has a frame ready will send it.

• Then it waits for some time.

• If it receives the acknowledgement then the transmission is successful.

• Otherwise the station uses a backoff strategy, and sends the packet again.

• After many times if there is no acknowledgement then the station aborts the idea of transmission.

CSMA/CA :

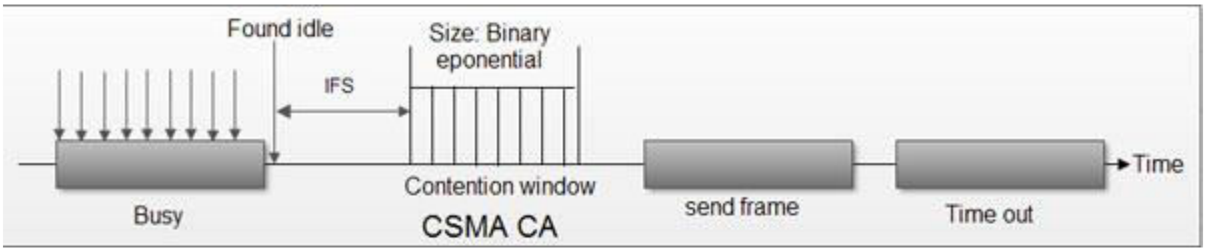
CSMA/CA protocol is used in wireless networks because they cannot detect the collision so the only solution is collision avoidance.

• CSMA/CA avoids the collisions using three basic techniques.

(i) Interframe space

(ii) Contention window

(iii) Acknowledgements



1. Interframe Space (IFS)

• Whenever the channel is found idle, the station does not transmit immediately. It waits for a period of time called interframe space (IFS).

• When channel is sensed to be idle, it may be possible that same distant station may have already started transmitting and the signal of that distant station has not yet reached other stations.

• Therefore the purpose of IFS time is to allow this transmitted signal to reach other stations.

• If after this IFS time, the channel is still idle, the station can send, but it still needs to wait a time equal to contention time.

• IFS variable can also be used to define the priority of a station or a frame.

2. Contention Window

• Contention window is an amount of time divided into slots.

• A station that is ready to send chooses a random number of slots as its wait time.

• The number of slots in the window changes according to the binary exponential back-off strategy. It means that it is set of one slot the first time and then doubles each time the station cannot detect an idle channel after the IFS time.

• This is very similar to the p-persistent method except that a random outcome defines the number of slots taken by the waiting station.

• In contention window the station needs to sense the channel after each time slot.

• If the station finds the channel busy, it does not restart the process. It just stops the timer & restarts it when the channel is sensed as idle.

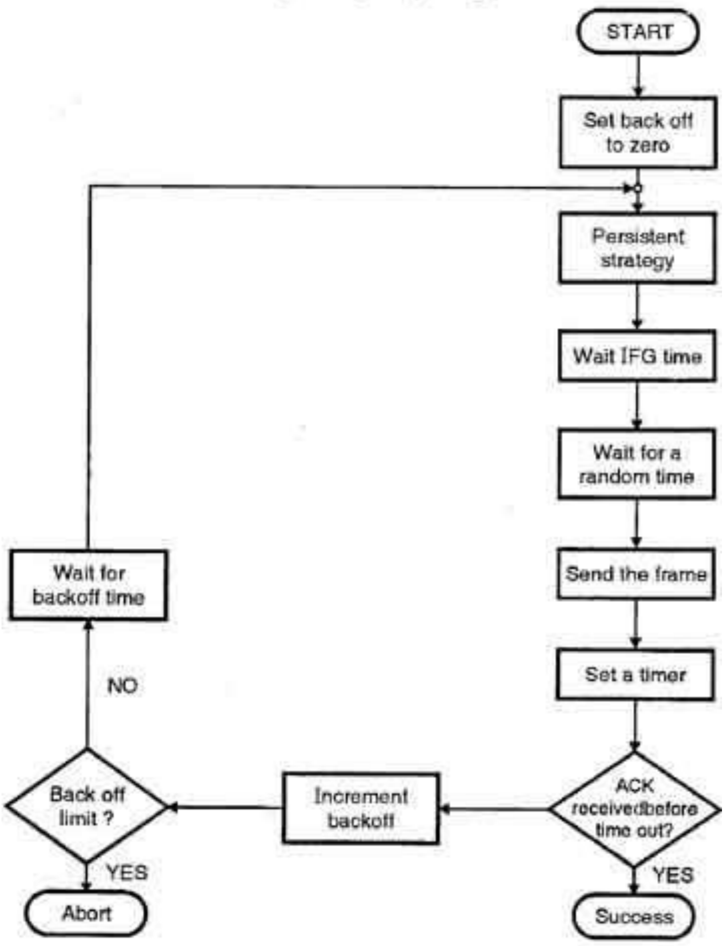
3. Acknowledgement

• Despite all the precautions, collisions may occur and destroy the data.

• The positive acknowledgment and the time-out timer can help guarantee the reception of the frame.

CSMA/CA Procedure:

Fig. Shows the flow chart explaining the principle of *CSMA/CA*.



CSMA/CA procedure

This is the CSMA protocol with collision avoidance.

• The station ready to transmit, senses the line by using one of the persistent strategies.

• As soon as it find the line to be idle, the station waits for an IFG (Interframe gap) amount of time.

• If then waits for some random time and sends the frame.

• After sending the frame, it sets a timer and waits for the acknowledgement from the receiver.

• If the acknowledgement is received before expiry of the timer, then the transmission is successful.

• But if the transmitting station does not receive the expected acknowledgement before the timer expiry then it increments the back off parameter, waits for the back off time and re-senses the line.

6. Channelization:

Channelization is a multiple-access method in which the available bandwidth of a link is shared in time, frequency, or through code, between different stations. Protocols in this category: FDMA, TDMA, and CDMA.

Working of CDMA: CDMA simply means communication with different codes. For example, in a large room with many people, two people can talk in English if nobody else understands English. Another two people can talk in Chinese if they are the only ones who understand Chinese, and so on. In other words, the common channel, the space of the room in this case, can easily allow communication between several couples, but in different languages (codes).

Let us assume we have four stations 1, 2, 3, and 4 connected to the same channel. The data from station 1 are d_1, from station 2 are d_2, and so on. The code assigned to the first station is c_1, to the second is c_2, and so on. We assume that the assigned codes have two properties. 1. If we multiply each code by another, we get O.
2. If we multiply each code by itself, we get 4 (the number of stations). With these two properties in mind, let us see how the above four stations can send data using the same common channel, Station 1 multiplies (a special kind of multiplication, as we will see) its data by its code to get $d_1 . c_1$ Station 2 multiplies its data by its code to get $d_2 . c_2$ And so on.The data that go on the channel are the sum of all these terms, as shown in the box. Any station that wants to receive data from one of the other three multiplies the data on the channel by the code of the sender. For example, suppose stations 1 and 2 are talking to each other. Station 2 wants to hear what station I is saying. It multiplies the data on the channel by c_1' the code of station 1.
Because $(c_1 . c_1)$ is 4, but $(c_2 . c_1)$, $(c_3 . c_1)$, and $(c_4 . c_1)$ are all Os, station 2 divides the result by 4 to get the data from station 1.

data $= (d_1 . c_1 + d_2 . c_2 + d_3 . c_3 + d_4 . c_4) . c_1$

The result is divided by number of stations which are sharing the channel.

7. a) Fast Ethernet was designed to compete with LAN protocols such as FDDI or Fiber Channel (or Fibre Channel, as it is sometimes spelled). IEEE created Fast Ethernet under the name 802.3u. Fast Ethernet is backward-compatible with Standard Ethernet, but it can transmit data 10 times faster at a rate of 100 Mbps. The goals of Fast Ethernet can be summarized as follows:

1. Upgrade the data rate to 100 Mbps.
2. Make it compatible with Standard Ethernet.
3. Keep the same 48-bit address.
4. Keep the same frame format.
5. Keep the same minimum and maximum frame lengths.

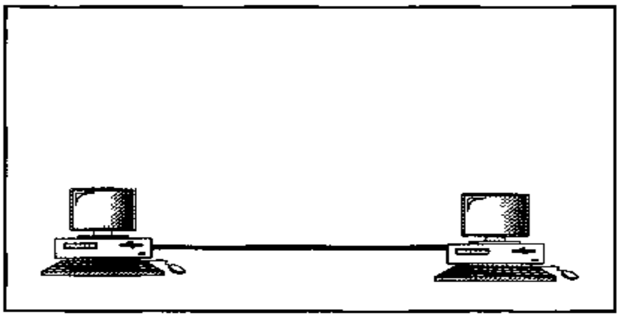
A new feature added to Fast Ethernet is called autonegotiation. It allows a station or a hub a range of capabilities. Autonegotiation allows two devices to negotiate the mode or data rate of operation. It was designed particularly for the following purposes:

 To allow incompatible devices to connect to one another. For example, a device with a maximum capacity of 10 Mbps can communicate with a device with a 100 Mbps capacity (but can work at a lower rate).

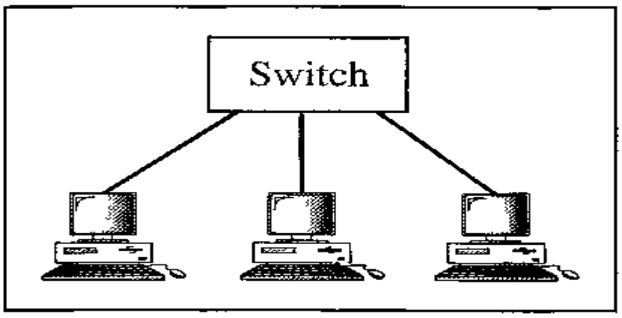
 To allow one device to have multiple capabilities.

 To allow a station to check a hub's capabilities.

Topology Fast Ethernet is designed to connect two or more stations together. If there are only two stations, they can be connected point-to-point. Three or more stations need to be connected in a star topology with a hub or a switch at the center.



a. Point-to-point

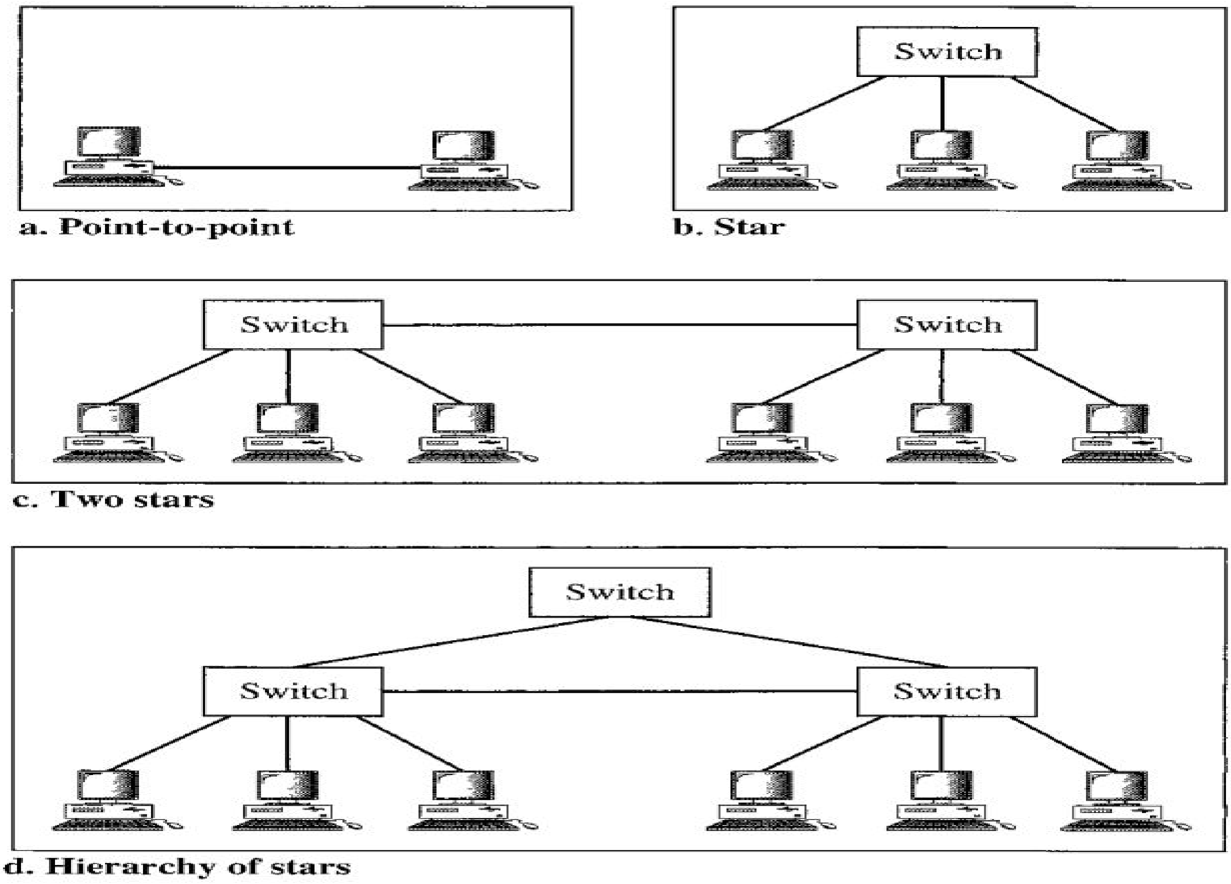


b. Star

The need for an even higher data rate resulted in the design of the Gigabit Ethernet protocol (1000 Mbps). The IEEE committee calls the Standard 802.3z. The goals of the Gigabit Ethernet design can be summarized as follows:

1. Upgrade the data rate to 1 Gbps.
2. Make it compatible with Standard or Fast Ethernet.
3. Use the same 48-bit address.
4. Use the same frame format.
5. Keep the same minimum and maximum frame lengths.
6. To support autonegotiation as defined in Fast Ethernet.

Gigabit Ethernet is designed to connect two or more stations. If there are only two stations, they can be connected point-to-point. Three or more stations need to be connected in a star topology with a hub or a switch at the center. Another possible configuration is to connect several star topologies or let a star topology be part of another as shown



Ten-Gigabit Ethernet The IEEE committee created Ten-Gigabit Ethernet and called it Standard 802.3ae. The goals of the Ten-Gigabit Ethernet design can be summarized as follows:

1. Upgrade the data rate to 10 Gbps.

2. Make it compatible with Standard, Fast, and Gigabit Ethernet.

3. Use the same 48-bit address.

4. Use the same frame format.

5. Keep the same minimum and maximum frame lengths.

6. Allow the interconnection of existing LANs into a metropolitan area network (MAN) or a wide area network (WAN).

7. Make Ethernet compatible with technologies such as Frame Relay and ATM.