LESSON 35:
LINK-TO-LINK LAYERS

Objective

To provide a detailed understanding of the concepts of link to link layers

Introduction

Link-to-link Layers

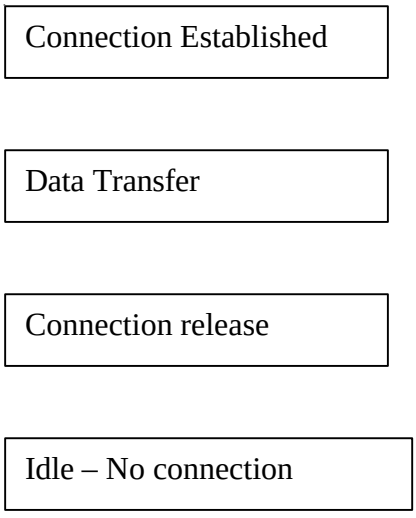
The first three layers, viz. physical layer, data link layer and network layer form the link-to-link layers of OSI reference model. Entities in an OSI layer perform certain functions to fulfill the stated purpose of the layer. They obtain services from the immediate lower layer and provide services to the immediate upper layer.

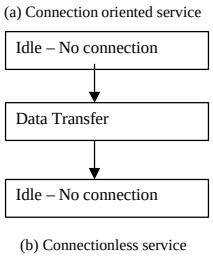
OSI services are of two types:

1. Connection Oriented Services
2. Connection less Services.

Connection Oriented Services

In connection oriented services, a connection is first established between the sender and the receiver before data transfer takes place. The connection may be virtual (logical) or physical. The type of connection depends upon the network capabilities and facilities. The essence of connection oriented service is that a connection acts like a tube or a pipe delivering the data to the receiver strictly in the same order in which the data was put into the connection by the render. These services are modelled after the telephone system. The example of connection oriented services is the virtual circuit service. The operation of connection oriented services is shown in Fig. 35.1 (a).





(a) Connection oriented service

(b) Connectionless service

Fig.35.1 Operation of connection-oriented and connectionless networks.

Connectionless Services

Connectionless service is modelled after the postal system. Each submission by the sender is treated independently of the others and is self-contained with the full address of the destination and the source indication which may be the full address too. In connectionless service, when two messages are sent to the destination one after another, it is possible that the first one is delayed and the second one arrives first.

Datagram service is the example of connectionless service. The operation of connectionless service is shown in Fig. 35.1 (b).

A connection-oriented service has provision for acknowledgements, flow control and error recovery. But a connectionless service does not generally have such provisions.

Peer entities of OSI layers communicate using peer protocols. Protocols are strict procedures and sequence of actions to be followed in order to achieve orderly exchange of information among peer entities. There are two sets of protocols corresponding to the two categories of services. One set for the connection oriented services and the other set for connectionless services. Layer protocols relate to the implementation of services of the layer and therefore are not visible to the users or other layers. This separation of services and the protocols provides complete freedom to change protocols at will without affecting the services.

Physical Layer

ISO-OSI architecture permits the usage of a realistic variety of physical, media and control procedures. These are enential for ISO-OSI model. Lowest layer of the OSI architecture has been identified as the physical layer. This layer performs functions associated with the activation and deactivation of physical connections. It deals with encoding/decoding of signals and the bit level transmission of electronic signals through the available transmission medium. The transmission may be synchronous or asychronous. Mode of data transmission may be simplex; half duplex or full duplex. The physical layer

provides mechanical, electrical, functional and procedural characteristics to activate maintain and deactivate physical connections for transmission of bits.

Data Link Layer

Some physical communication media like telephone lines have error rates that are hot acceptable for the great majority of data network applications. Therefore, we require special techniques to ensure error free transmission of data.

The data link layer deals with error detection and automatic recovery procedures required when a message is lost or corrupted. For this purpose, a user of this layer, i.e. the network layer is required to break up the data to be transmitted into frames which are then numbered and transmitted sequentially. The layer provides function and procedural means to establish, maintain and release data link connections for the entities in the network layer. A data link connection may be built upon one or several physical connections. Another important function performed by the data link layer is the link level flow control of frames.

Flow control is essentially a traffic regulation mechanism that will have to be enforced when the receiver is unable to accept frames as fast as the transmitter is able to send.

A data link may be of point- to-point type as in the case of terrestrial networks or broad cast type in the case of SBDNs, LANs or MANs.

The data link layer in the case of broadcast type channels will have to perform an additional function of acquiring or accessing the channel before data transmission can begin. Here we discuss only point-to-point links. We focus our discussion on efficient channel utilisation, error recovery and flow control mechanisms. These discussions are also applicable for data transmission in a broadcast type channel, once the channel is acquired.

The main source of error in a system

There are following main sources of error in a system:

1. **Thermal noise**: Which is internal to the system.
2. **Impulse or spike noise**: Which originates from man-made sources like automobiles or signalling in telephone systems and from natural sources like lightning.
3. **Cross talk**: Which occurs through electromagnetic radiation from parallel and adjacent wires which behave like an antenna.

Type of Errors

Errors are of two types:

1. Isolated Errors
2. Bursty Errors

Errors due to thermal noise are generally called isolated errors. Errors due to spike noise or cross talk are called bursty errors.

Error control mechanisms are chosen depending upon the type of-error that is predominant in a given system. There are three error controlling mechanisms that are commonly used:

i. Echo Checking

ii. Forward Error Correction (FEC)

iii. Automatic Repeat Request (ARQ)

Echo Checking:

It is normally used only when the source is intelligent. In echo checking, the data is echoed back to the sender and the data is retransmitted after detecting an error by sender. Since all the data is echoed back, the scheme requires double the transmission bandwidth required for one-way transmission. This scheme works well in a full duplex connection with a large channel capacity. Echo checking is typically used for terminal-to-computer communication. It is not suitable for large volume data transfer.

Forward Error Correction (FEC)

This mechanism is based on the hypothesis that it is possible to detect and correct errors at the receiving end by adding appropriate redundant information to the data at the sending end. This scheme is useful where isolated errors occur and the reliability of the channel is low. The most popular error correcting code is the Hamming code. In Hamming Code, a code set is formed by adding r number of redundant bits to the legal non-redundant information words. For example, an 8-bit character set has 256 legal combinations. By adding three bits to each of the characters, we generate a code set which has a total 2048 combinations of which only 256 care considered legal.

In a code set, the minimum number of bit changes required to produce one legal word from another is called Hamming distance between the two code words.

A code set in which at least one pair of legal code words has a distance d, but no pair has a distance less than d is called a distance- d code set.

Once a distance-d code set is formed we get two important results:

i. Bit changes up to d -1 bits would result in illegal code words. By recognising the illegal code words at the receiving end, we are in a position to detect upto d -1 bit errors.

ii. By applying simple nearest neighbour principle, an illegal code word can be corrected to the nearest legal code word, i.e. errors upto $(d - 1) / 2$ bit can be corrected.

Now we can note that a parity bit scheme is a distance-2 Hamming code and hence can detect single bit errors. The error, however, cannot be corrected as the resulting illegal code word is equidistant from the legal code words and hence the nearest neighbour principle fails.

Automatic Repeat Request (ARQ)

In this scheme, the information word is coded with adequate redundant bits so as to enable detection of errors at the receiving end. If an error is detected, the receiver asks the sender to retransmit the particular information word. ARQ scheme is useful where expected error is bursty in nature or error rate of the channel is low, i.e. the channel is fairly reliable. Mostly, the errors encountered in data communication systems are bursty in nature. Hence, ARQ schemes are used extensively in data networks.

Burst-error detection by one parity bit is not satisfactory. The error detection method which is used widely in ARQ protocols is called polynomial coding or cyclic redundancy coding (CRC)

Here, a check-sum of r bits is determined and appended to the message stream, such that the combined stream, when treated as polynomial, is exactly divisible by a predefined generator polynomial G(x) of degree r. An error is known to have occurred if G(x) does not divide exactly the received bit stream

There are three standard generator polynomials which are widely used in WAN applications.

1) CRC-12 = $x^{12} + x^{11} + x^3 + x^2 + x + 1$
2) CRC-16 = $x^{16} + x^{15} + x^2 + 1$
3) CRC-CCITT = $x^{16} + x^{12} + x^5 + 1$

CRC schemes are very effective in detecting errors.

A typical CRC scheme can detect the following errors:

1. All single and two bit errors
2. All odd bit errors
3. All burst errors of length less than r
4. Over 99.9% of all burst errors of length equal to or greater than r.

A ARQ protocol is characterized by four functional steps:

1) Transmission of frames
2) Error checking at receiver and
3) Acknowledgement
   (a) Negative ‡If error is detected (NAK)
   (b) Positive ‡ If no error is detected (PAK)
4) Retransmission if acknowledgement is negative (NAK) or if no acknowledgement is received within stipulated time.

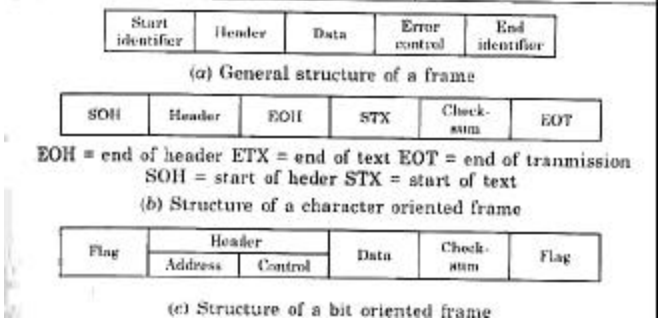
It can be noted here that ARQ protocols require two-way communication even if the information transfer is simplex (i.e. one way only). Information is exchanged in the form of frames. The beginning and the end of frames are identified by means of flags or special characters. The general structure of a frame is shown in Fig. 38.2 (a). The header information may contain destination address in the case of multidrop lines or command/response identifier in a point-to-point link. Some control information is also carried in the header. Data may be in the form of characters or continuous sequence of bits. According to type of data, we have two types of frames. These are:

1. Character or Byte Oriented frame
2. Bit Oriented frame.

Structures of Byte-Oriented and Bit-Oriented frames are shown in Fig. 38.2 (b) and Fig.38.2 (c), respectively. ARQ protocols that deals with two types of frames are known as Byte-oriented and Bit-Oriented protocols, respectively. The example of the Byte-Oriented protocol is the binary synchronous communication (BISYNC) protocol of IBM and example of the Bit Oriented protocol is the synchronous data link control (SDLC) protocol of IBM.

SDLC was modified by a number of standardizing agencies and adopted under different names: These are given as:

Modified Version of SDLC	*Name of Standardising Agency*
Advanced Data Communication Control Procedure (ADCCP)	ANSI
High Level Data Link Control (HDLC)	ISO
Link Access Procedure (LAP)'	CCITT
Link Access Procedure-Balanced (LAPB)	CCITT



(a) General structure of a frame

EOH = end of header ETX = end of text EOT = end of tranmission
SOH = start of heder STX = start of text

(b) Structure of a character oriented frame

(c) Structure of a bit oriented frame

Fig.35.2 Data Link Frame Structure.

It is possible that the control characters like STX, SOH or ETX which convey special meaning to the receiver. Control characters may appear as the part of the data in the character-oriented frames. Similarly, the flag pattern may appear in the binary stream of data in the bit oriented frames. A mechanism is required so that the receiver does not act on the special characters or flag patterns if they appear as part of the data. The mechanism used for this purpose are called character stuffing or byte stuffing and bit stuffing for the byte oriented and bit-oriented protocols respectively.

In character stiffing, the special character data link escape (DLE) is stuffed in front of control character when it appears as a part of data. When a DLE character is encountered by the receiver, it deletes the character from the main stream of data. The DLE character ignores the special significance, if any, of the character following DLE.

A data stream and the associated character stuffed .stream are shown in Fig. 35.3 (a)

Data stream	STX AB ... SOH ... DLE ... ETX ... STX ... EXT
Byte stuffed stream	STX AB ... DLE SOH ... DLE DLE ... DLE ETX ... DLE STX ... ETX

(a) Character stuffing

Data stream	0101 ... 0111111110 ... 001111100 ... 0101
Bit stuffed stream	0101 ... 01111111110 ... 001111110000 ... 0101

(b) Bit stuffing

Fig. 35.3 Various stuffing techniques for data frame.

The first STX and the last STX in Fig.35.3 (a) are the only characters recognised by the receiver as control characters. The significance of all other special characters SOH, ETX and STX are ignored.

Similarly, flag pattern, which is usually 0111110, is treated by stuffing a zero when five consecutive 1's are encountered in the data. This is shown in Fig. 35.3 (b).

A simple ARQ protocol is known as stop-and-wait protocol. It is also called idle RQ protocol. In this protocol the sender transmits a frame and waits for an acknowledgement. Fig.35.4 shows the operation of the protocol.

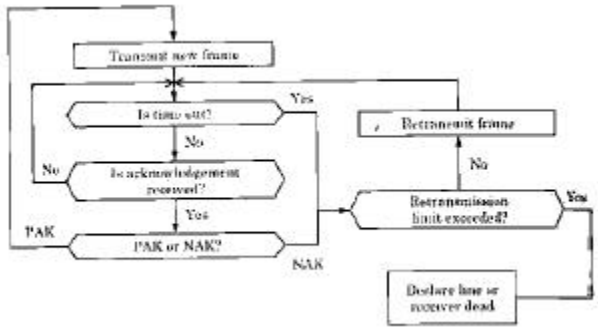


Fig. 38.4 Operation of the stop and wait protocol.

A retransmission count is used to avoid endless retransmissions. If a number of NAKs are received then one may conclude that the receiver is dead or the link is broken.

In idle RQ protocol, data transmission may be organised in a number of ways:

1. Full duplex (2 wire or 4 wire) or half duplex.
2. Separate or a common link for both data and acknowledgement packets.
3. Separate packet for acknowledgement, or acknowledgement as part of data flowing in the reverse direction, known as piggybacked acknowledgement.

To feel the performance of the protocol, let us consider the model shown in fig.35.5 (a). The data channel utilization details is shown in Fig.35.5 (b).

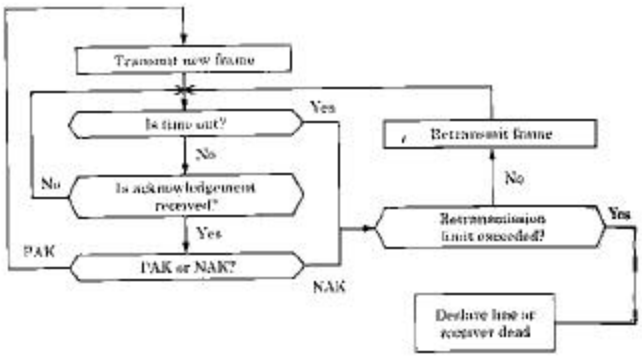


Fig. 38.5 An implementation model for the Stop-and-wait protocol.

The channel utilisation factor U may be expressed in terms of the times shown in Fig.35.5 (b) as

$$U = \frac{f}{f + 2p + k + a} * 100\% \quad \text{……..Eq.1}$$

for typical values, U may work out to be as low as 10%. The utilisation of channel can be improved by sharing the same channel for both data and acknowledgement or by piggybacking the acknowledgement with data. In the piggybacking scheme, it may be necessary to send a separate acknowledgement frame in case there is no reverse traffic.

Generally, all ARQ protocols require that a sequence number be used to identify a frame. Now we consider the sequence of steps in stop-and-wait ARQ protocol.

i. Error-free frame arrives at the receiver's end.

ii. Receiver transmits P AK frame

iii. The PAK frame gets lost and does not reach the sender

iv. The sender times out and retransmits the frame.

How would the receiver know that the frame that is received is a new one or a retransmitted one? To bring about this distinction, a 1-bit sequence number is used in stop-and-wait protocols. If the sequence bit is the same as the previous fame, the frame is a retransmitted one; otherwise it is a new one.

In piggybacked scheme, whenever a frame (new one or retransmitted) is sent, it must reflect the latest acknowledgement status. Otherwise, deadlocks may occur in the system.

Utilisation of the channels can be considerably improved by using what are known as continuous RQ or pipelined ARQ protocols.

In pipelined protocols, a sender may transmit frames continuously one after another without waiting for Acknowledgements. However, the number of frames that may be retransmitted in this manner is limited by the buffer capacity at the receiver. The acknowledgement traffic on the channel may be reduced by acknowledging the latest correctly received frame. When the receiver finds a frame in error, a retransmission is requested in either one of the following modes:

1. Selective repeat

2. Go back n

In selective repeat mode, the receiver requests the transmission of specific frame. But, in 'Go back n' mode, the retransmission of all the frames after the frame number n is requested.

The selective repeat case demands a large buffer capacity at the receiver than the case 'go back n'.

One the other hand, transmission overheads are higher in of case 'go back n'. There is an obvious need for numbering sequentially the frames in both cases. It is also compulsory that the frames must be delivered to the network layer in the same order in which they are received at the data link layer and the sequence numbering helps this process.

Now consider the limited buffer availability at the receiving ends. Therefore, the sequence numbering may be a modulo of some number m, usually, modulo-8 or modulo-128 is used.

The buffer can then be organised in a sliding window fashion such that frames may be received and buffered depending upon the window size. The maximum window size W is limited by the retransmission strategy used:

For selective repeat: W d•m / 2

For 'go back n' : W d•m -1

The restriction on the window size is necessary to know correctly the status of transmission. For example, if we use a window size of 6 in a modulo-8 scheme with selective repeat retransmission strategy, wrong interpretations can occur under the following conditions:

i. Frames 0-5 are send and successfully received.

ii. PAK-5 is sent and the window is set to receive frames 6, 7 or, 0-3.

iii. PAK-5 is lost, sender times out and retransmits from O.

In this case, the receiver would interpreter this frame 0 as a new one. If the window size were restricted to 4 or less, this anomaly would not occur. Similar situations can be visualised in the case of go back n strategy.

In ARQ protocols, although the window size is predetermined and agreed upon by both sender and receiver. It is possible that the receiver is unable to accept frames due to various other reasons such as lack of buffer space. In such cases, a flow control mechanism is invoked whereby the receiver requests the sender to 'Stop' until further advice to 'go' is given.

This flow control mechanism is called stop-and-go strategy. Sequence number, acknowledgement and flow control information are all part of the control field of a frame as shown in Fig. 35.6.

The field 'kind' is used to indicate the type of frame such as:

(1) Frame with data or

(2) Acknowledgement frame or

(3) Command frame etc.

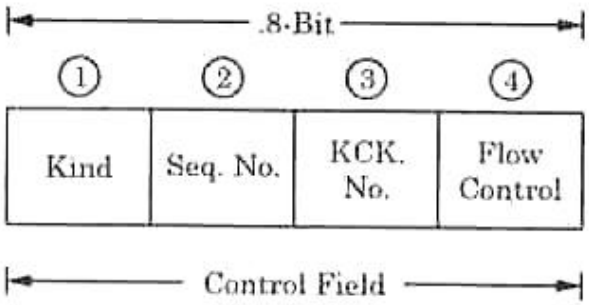


Fig. 35.6 Typical structure of the control field of a frame.

Network Layer

The highest link-to-link layer in the OSI model is the network layer. This layer functions on a link-to-link basis. But it concerns with, transmission of packets from the source node to the destination node. It deals with routing and switching considerations that are required in establishing a network connection. Such type of network connection may involve the use of several transmission resources in tandem including a number in intermediate switching nodes of different subnet works. The network layer makes invisible to the transport layer. This layer needs details of the underlying communication media and the different characteristics of the transmission and the network technologies.

It only assures a certain quality of service to the upper layers. Since an end-to-end connection may involve routing through a number of different networks. Internetworking is an important function of the different networks. Internetworking is an important function of the network layer. In the internetworking, we handle followings functions:

(1) Addressing schemes

(2) Network capabilities

(3) Protocol differences

(4) Accounting and billings.

Network congestion may occur due to too many messages on a particular route. Network congestion is also tackled by the network layer.

There is a wide variety of routing and congestion control algorithms. However, OSI network layer specifications do not recommend and discuss any routing or congestion control algorithms. OSI network layer specifications are left as implementation dependent features.

Routing strategies can be classified as shown in Fig. 35.7

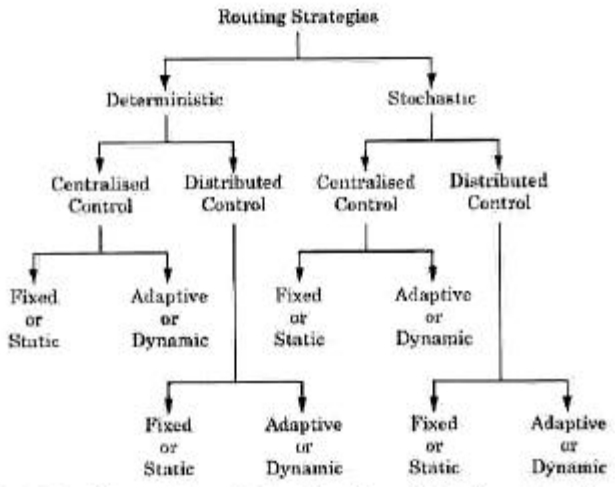


Fig. 35.7 Illustration of classification of Routing algorithms.

Routing algorithms may be deterministic or stochastic in nature. Here stochastic means random in nature.

Routing decisions may be taken centrally by collecting global information from the network or in a distributed fashion based on the local conditions. Finally, each strategy may use a fixed criterion or adapt itself to varying network traffic conditions.

A routing algorithm that uses a precomputed route from a given node to another is deterministic, distributed, static algorithm. If the route changes (say with time of the day) one may call the algorithm dynamic but still deterministic.

If the routing information is computed by a central node based on the knowledge of network topology etc. and distributed to the other nodes, then it is centrally controlled.

If the route is chosen based on some probability calculation (say a random number) the algorithm is stochastic in nature.

A number of measures maybe used in accessing the performance of a routing algorithm:

(1) Minimum Delay

(2) Minimum Number of Intermediate Nodes or Hops

(3) Processing complete

(4) Signalling capacity required on the network.

(5) The rate of adaptation in the case of adaptive algorithms.

(6) Fairness to all types of traffic

(7) A reasonable response time over a range of traffic intensities.

(8) Robustness**:** The ability to reach the destination even when parts of the network fail.

(9) Stability: The ability to reach the destination quickly without wandering. Following routing algorithms are widely used:

1. Shortest path algorithm
2. Flooding algorithm
3. Flow-based routing algorithm
4. Hierarchical routing algorithm
5. Broadcast routing algorithm.

Shortest Path Algorithm

A shortest path may be determined based on:

(1) Link length

(2) Minimum cost

(3) Minimum Delay or

(4) Minimum number of hops.

Recent versions of ARPANET routing algorithms use a shortest path algorithm using minimum delay as the metric.

Flooding Algorithm

In flooding, an incoming packet is sent out on all outgoing links except the one on which it arrived. Flooding generates a vast number of duplicate packets. Therefore, it can chock the network unless controlled. This algorithm is robust and simple.

Flow based Routing Algorithm

In some networks, where the mean data flow between each pair of nodes is relatively stable and predictable, it is possible to estimate the average flow on each line and then the mean packet delay for whole network. Flow based routing attempts to minimise the overall mean packet delay. When networks grow large then it is not possible for each node to keep track of the best route for every other node.

Hierarchical routing algorithm

This routing segment the network to different levels and a node is then only concerned about routing to the appropriate level to reach the destination.

Congestion occurs when there are too many packets in the network. There are two method of congestion control:

1. One method of congestion control is to preallocate buffers throughout the route before starting the packet transmission from a source node to the destination node.

   This method actually avoids congestion by taking a preventive measure. It is easily implemental in virtual circuit connections.

2. Second method is to limit the total number of packets than can exist on the network at any time. This technique is also called Isarithemic control.

Internetworking is not well defined in OSI model. Internet protocol (IP) of ARPANET is far more popularly used as it has been in use for a number of years and has stood the test of time. This inconjunction with the transmission control protocol (TCP), TCP is still being widely used.

Notes