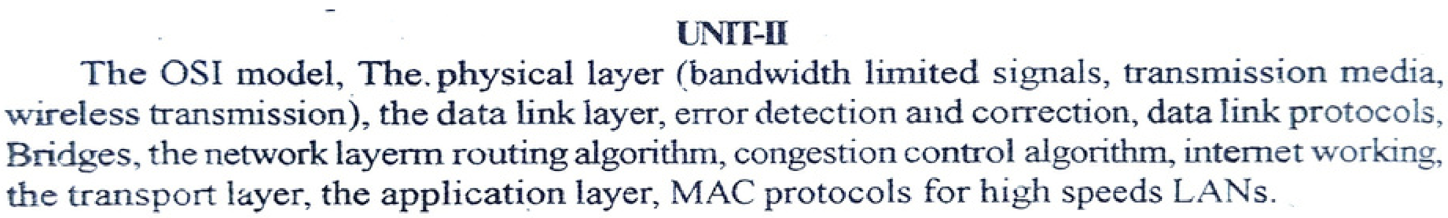

## UNIT-II

The OSI model, The. physical layer (bandwidth limited signals, transmission media, wireless transmission), the data link layer, error detection and correction, data link protocols, Bridges, the network layerm routing algorithm, congestion control algorithm, internet working, the transport layer, the application layer, MAC protocols for high speeds LANs.

## 2. The OSI Model 30-55

Chapter

# 2 The OSI Model

Established in 1947, the International Standards Organization (ISO)

A Multinational body dedicated to worldwide agreement on international standards, An ISO standard that covers all aspects of network communication is the open systems interconnection model. It was first introduced in the late 1970. An open system is a set of protocols that allow any two different systems to communicate regardless of their underlying architecture. The purpose of the OSI model is to show how to facilitate communication between different systems without requiring changes to the logic of the underlying hardware and software, the OSI model is not a protocol, it is a model for understanding and designing a network architecture that is flexible, robust and interportable.

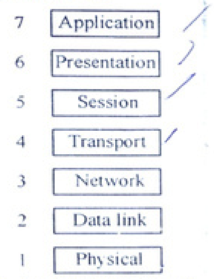

**Fig. 1 : Seven layers of the OSI Model**

The OSI model is a layered framework for the design of network system that allow communication between all types of computer system. It consists of seven of separate but related layeres, each of which defines a part of the process of moving information across a network.

**Layered Architecture :** The OSI model is composed of seven ordered layers : Physical (Layer 1), data link (layer 2) network (layer 3), transport (layer 4), session (layer 5) presentation (layer 6) and application (layer 7)

Fig. 2. Show the layers involved when a message is sent from device A to device B. As the message travels from A to B, it may pass through many intermediate nodes. These intermediate nodes usually involve only the first three layers of the OSI model.

In developing the model, the designers digitilized the process of transmitting data to its most fundamental elements. They identified which networking functions had related uses and collected these functions into discrete groups that become the layers. Each layer defines a family of functions distinct from these of the other layers. By defining and localizing functionality in this fashion, the designers created an architecture that is both comprehensive and flexible the OSI model allows complete interportability between otherwise incompatible system.

Within a single machine, each layer calls upon the services of the layer just below it layer 3 for example, uses the service provided by layer 2 and provides

services for layer 4. Between machines layer x on one machine communicate with layer x on another machine. This communication is governed by an agreed upon series of rules and convention called protocols. The process on each machine that communicate at a given layer are called peer to peer process. Communication between machines is therefore a peer to peer process using the protocols appropriate to a given layer.

## 2.1 Peer to peer process :

At the physical layer, communication is direct. In Fig 2 device A sends a stream of bits to device B (through intermediate nodes) At the higher layers however communication must move down through the layers on device A, over to device B and then back up through the layers. Each layer in the sending device adds its own information to the message it receive from the layer just above it and passes the whore package to the layer just below it. At layer 1 the extire package is converted to a form that can be transtritted to the receiveing device. At th receiving machine, the message is unwrapped layer by layers with each process receiving and removing the data mean for it for example layer 2 removes the data meant for it, then passes the rest to layer 3. Layer 3 then removes the data meant for it and passes the rest to layer Li and so on.

The interaction between layers in the OSI model

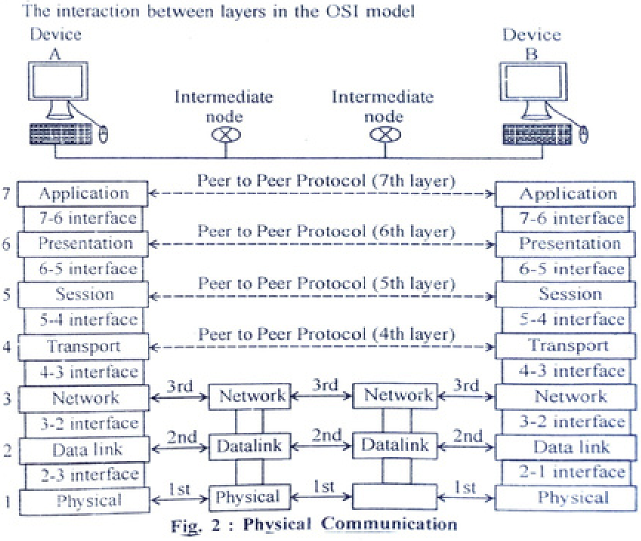

**Fig. 2 : Physical Communication**

**Interfaces Between Layer :** The passing of the data and network information down through the layers of the sending device and back up through the layers of the recieving device is mode passible by an interface between each pair of adjacent layers. Each interface define the information and services a layer must provide for the layer above it. Well defined interface and layer functions provide modularity to a network. As long as a layer provides the expected services to the layer, above it, the specific implamentation of ith functions can be modified or replaced without requring changes to the surrounding layers.

## 2.2 Organization of the layer :

The seven layers can be though of as belonging to the subgraphs. Layer 1, 2 and 3 physical, data link and network are the network repport layers; they deal with the physical aspects of moving data from one device to another (such as electrical specifications, physical connection, physical addressing and transport timing and reliability. Layer 5, 6 and 7 session, presentation and application can be through of as the user repport layer; they allow interperability among unrelated softwere systems. Layer 4 the transport layer, links the two subgroups and ensure that what the lower layers have transmitted is in a ferm that the upper layers can use. The upper OSI layers are almost always implemented in software lower layers are a combination of hardware and software, ercept for the physical layer, which is mostly hardware.

In fig. 3 which given an small view of the OSI layers, D7 Means the data unit at layer 7 D6 means the data unit at layer 6 and so on. The process ssarts at layer 7 (application layer) then moves from layer to layer in descending, requential order. At each layer, a header or possible a trailer, can be added to the data unit. Commonly, the trailer is added only at layer 2. When the formatted data unit passes through the physical layer (layer 1), it is changed into an electromagnetic signal and transported along a physical link.

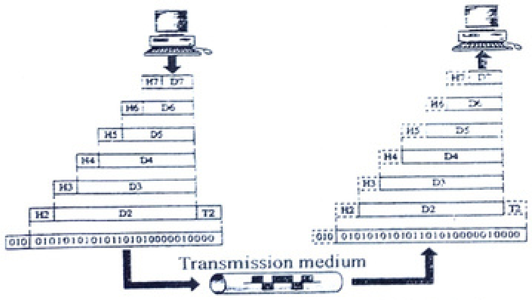

Fig. 3 : An Exchange using the OSI Model

## 1. Physical Layer

The physical layer coordinates the functions required to carry a bit stream over a physical medium. It deals with the mechanical and electrical specifications of the interface and transmission medium. It also defines the procedures and functions that physical devices and interfaces have to perform for transmission to occur. Fig 4 shows the position of the physical layer with respected to the transmission medium and the data link layer.

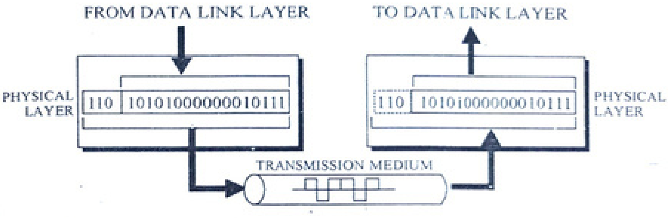

Fig. 4 : The physical layer

The physical layer is also concerned with the following :

- **Physical characteristics of interfaces and medium :** The physical layer defines the characteristics of the interface between the devices and the transmission medium. It also defines the type of transmission medium.
- **Representation of bits :** The physical layer data consists of a stream of bits (sequence of 0s or 1s) with no interpretation. To be transmitted, bits must be encoded into signals-electrical or optical. The physical layer defines the type of **encoding** (how 0s and 1s are changed to signals).
- **Data rate : The transmission rate-** the number of bits sent each second-is also defined by the physical layer. In other words, the physical layer defines the duration of a bit, which is how long it lasts.
- **Synchronization of bits :** The sender and receiver not only must use the same bit rate but also be synchronized at the bit level. In other words, the sender and the receiver clocks must be synchronized.
- **Line configuration :** The physical layer is concerned with the connection of devices the media. In a point-to-point configuration, a link is shared among several devices.
- **Physical topology :** The physical topology defines how devices are connected make a network. Devices can be connected by using a mesh topology (every device is connected to every other device), a star topology (devices are connected through a central device), a ring topology (each device is connected

to the next. forming a ring), a bus topology (every device is on a common link), or a hybrid topology (this is a combination of two or more topologies).

- **Transmission mode :** The physical layer also defines the direction of transmission between two devices : simplex, half-duplex, or full-duplex. In simplex mode, only one device can sent; the other can only receive. The simplex mode is a one-way communication. In the half-duplex (or simply duplex) mode, two devices can send and receive at the same time.

## 2. Data Link Layer

The data link layer tranforms the physical layer, a raw transmission facility, to a reliable link. It makes the physical layer appear error free to the upper layer (network layer). Figure 5 Shows the relationship of the data link layer to the network and physical layers.

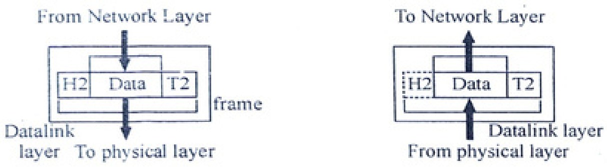

**Fig. 5. Data link layer**

**The data link layer is responsible for moving frames from one hap (node) to the next**

## 2.3 The data Link Layer Design issue

The main task of the data link layer is to take a raw transmission facility and transform it into a line that appears free of transmission errors in the network layer. It accomplishes this task by having the sender break the input data up into data frames (typically a few hundred bytes), transmit the frames sequentially, and process the acknowledgment frames sent back by the receiver. Since the physical layer merely accepts and transmits a stream of bits without any regard to meaning of structure, it is up to the data link layer to create and recognize frame boundaries. This can be accomplished by attaching special bit patterns to the beginning and end of the frame. If there is a chance that these bit patterns might occur in the data, special care must be taken to avoid confusion.

The data link layer should provide error control between adjacent nodes. Another issue that arises in the data link layer (and most of the higher layers as well) is how to keep a fast transmitter from drowning a slow receiver in data. Some traffic regulation mechanism must be employed in order to let the can transmitter know

how much buffer space the receiver has at the moment. Frequently, *flow regulation* and error handling are integrated, for convenience.

If the line can be used to transmit data in both directions, this introduces a new complication that the data link layer software must deal with. The problem is that the acknowledgment frames for A to B traffic compete for the use of the line with data frames for the B to A traffic. A clever solution (piggy bàcking) has been devised.

**Example :** HDLC

The Datalink layer is the second layer of the OSI model. The datalink layer performs various function depending upon the hardwere protocol used, but has four primary functions :

COMMUNICATION with the network layer above.

SEGMENTATION of upper layer datagrams (also called packets) into frames in sizes that can be handled by the communication hardware.

BIT ordering. .Organizing the pattern of data bits before transmission (packet formatting)

COMMUNICATION with the physical layer below.

This layer provides reliable transit of data across a physical link management, error notification, ordered delivery of frames, and flow control.

## 2.4 Data link layer issues

We have seen that the physical layer is responsible for actually transmitting bits over a transmission medium and in general a bit stream given from the higher layer is coded and modulated to do this.

Although the physical layer tries to maximize the transmission efficiency, i.e. make use of the entire capacity of the transmission medium, it is possible that bits will be transmitted in error.

Furthermore, it is possible that the receiver's higher layer are not operating fast enough (e.g. buffering and copying the received data to a disk file) to accommodate the rate at which the sender is sending the data.

An important function of the data-link layer is the establishment of reliable and efficient communication with another data-link layer peer.

This is achieved by having appropriate error control and flow control algorithms. The data -link layer in one machine / node collects data bits from its network layer (next higher layer), frames them and requests its physical layer to transmit to the other machine/node collects data bits from its network layer (next higher layer), frames them and requests its physical layer to transmit to the other machine / node.

At the receiving peer, communicated bits are collected by the physical layer and passed onto its data-link layer. The data-link layer in the receiver extracts the data and passes the data to its network layer.

## Framing

DU's to be sent from one data-link layer to another are framed into a sequence of bits before being given to the physical layer.

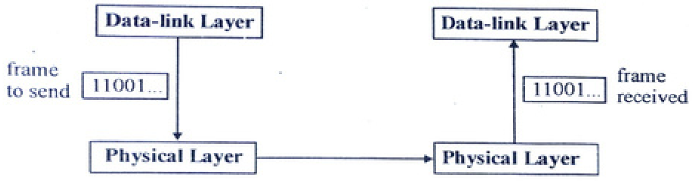

**Fig. 6 The use of a frame**

Some of the bits in the sequence say how many bits in total the sequence contains, i.s. the frame length. Most computer network physical layers don't provide connection oriented services. Framed data is inherently used for connection-less services. The entire frame is either sent for transmission or it is not.

Framed data allows breaks " in between the transmission of the data, so that the receiver can check for errors so far in the communication and can indicate if it is ready to receive more data.

### Error control

This is typically achieved by adding redundant bits in each frame during its transmission. Based on all the bits in the frame (data + error check bits), the receiver may be able to detect errors in the frame.

| Data | Check bits |
|---|---|

**Fig. 7**

A frame contains data and check bits.

We will discuss the other data in the frame later. For now let us understand how the check bits can be computed.

Flow control will be discussed when we discuss the protocols used by the data-link layer.

## 2.5 Flow control

Another important design issue that occurs in the data link layer (and higher layers as well as) is what to do with a sender that systematically wants to transmit frames faster than the receiver can accept them. This situation can easily occur when the sender is running on a fast (or lightly loaded) computer and the receiver is running on a slow (or heavily loaded) machine. The sender keeps pimping the frames out at a high rate until the receiver is completely swamped. Even if the transmission

is error free, at a certain point the receiver will simply not to be able to handle the frames as they arrive and will start to lose some. Clearly, something has to be done to prevent this situation.

The usual situation is to introduce flow control to throttle the sender into sending no faster than the receiver can handle the traffic. This throttling generally requires some kind of a feedback mechanism, so the sender can be made aware of whether or not the receiver is able to keep up. Various flow control schemes are known, but most of them use the same basic principle. The protocol contains well-defined rules about when the sender may transmit the next frame. These rules often prohibit frames from being sent until the receiver has granted permission, either implicitly or explicitly.

## 2.6 Error Correcting and Detecting Codes

Data is sent from the sender with redundancies so that errors in the received frame can be either detected (and the frame rejected) or detected and corrected.

A total of n - k check bits are added to a block of k data bits to generate a n-bit data block for transmission from the sender to the receiver. This is referred to as a (n, k) code with rate k/n. The (n - k) check bits are generated in a systematic fashion following some appropriate algorithm. The receiver essentially uses a similar algorithm to check for errors in the received block of n-bits

For error correction, a syndrome is generated at the receiver specifying the bits which need to be corrected. For error detection, the syndrome generated declares that there is an error somewhere in the received block of n-bits but does not specify the actual location of the errors.

Typically, the overhead associated with doing error correction is high so that such codes are usually of low rate and are not very efficient.

Error correction is typically only used in situations where the intrinsic error rate of the physical channel is poor and needs to be improved with error correction techniques.

In the following section we try to learn some basic concepts in coding theory from first principles. Our goal is not to make use of advanced mathematical results. From coding theory perspective, the entire physical layer can be construed as a channel enabling the two peers at the link layer to communicate with each other. Let us look at the model.

A communication Model

Message Source : Source for information in bits (k-bits).

Encoding Process : process of converting a block of k-bit information to n-bit codeword,

Channel : Medium in which n-bit codewords pass through. Decoding process : Process of converting n-bit received block to a k-bit message.

Message sink : Destination for k-bit information in bits.

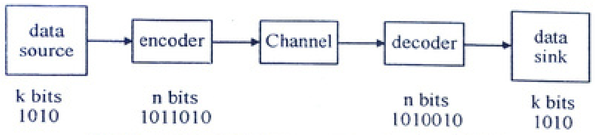

Fig. 8 A data link layer communication Model

### Error Model of Channel

There are various error models. The binary symmetric model is one such model. This is an appropriate model in situations where the errors are truly random. Many satellite channels follows this model. In practice the noise appears more bursty and you need more complicated models to simulate them.

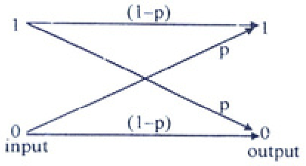

Fig. 9 : Binary Symmetric Channel

## Error Correcting codes

Definition 8 the error correction capability of a code is the maximum number of errors that can occur in transmitted codewords while always remaining possible for the receiver to recover the error or the original message.

Let $C_1$ = 11001100 and $C_2$ = 11111111 be two codewords.

If the received word is T = 11111100, and given the fact that there were two errors, you cannot determine whether the transmitted code word is c1 or c2.

This only means that error correction capability is LESS THAN 2.

We could not decide the original codeword as the received word is exactly in between c1 and $c_2$. How do we correct errors ? What is the limit on number of errors that can be corrected ?

Let be the transmitted codeword and be the received word (after affected by the channel). Hence

$$r = c \quad e,$$

where e is the error word. The effect of the error is that it increases the Hamming distance between the codeword and the received word. In error correction, we choose a most likely codeword nearest to the received word. The nearness is defined by the Hamming distance we defined earlier.

Let us consider the code in Example 2 where the block length 5 and the number of codewords is 4.

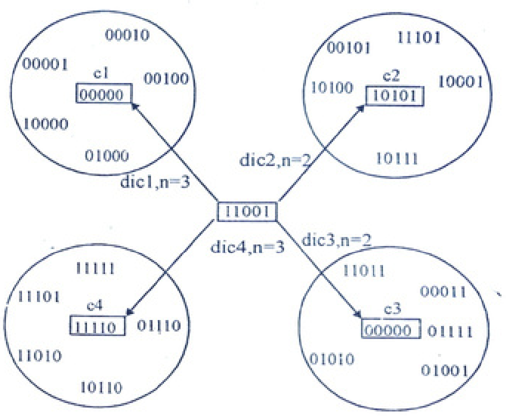

Fig. 10 : A decoding Example

Error Correction is possible only if the number of errors is less than half the minimum Hamming distance of the code because this will ensure that an erroneous received message will be closer the transmitted codeword than to any other codeword in the code.

Result : The error correction capability of a code is where [x] denotes the integer part of x, for example [1, 3] = 1 and [1.7] = 1

## Error Detecting codes

Assume the receiver has no prior knowledge of the block transmitted to it. If the transmitter sends only k-bits, the receiver does not know whether any errors were introduced.

Generally, the number of transmitted bits is n > k. The act of deciding whether a received block contains errors is called error detection.

Note that there is no way of always being able to detect errors in a received block.

Error Detection simply means that the receiver detects the fact that some errors exist in a received block.

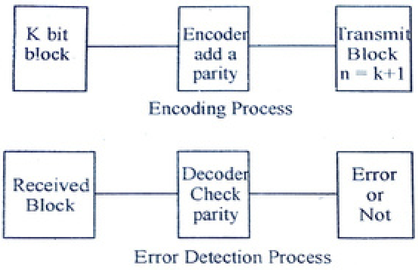

**Fig. 11 : Simple Error Detection Scheme**

## Data link protocol

A set of rules relating to data communications over a data link. Note : Link protocols define data link parameters, such as transmission code, transmission mode, control procedures, and recovery procedures.

## 2.7 HDLC

The HDLC protocol is a general purpose protocol which operates at the data link layer of the OSI reference model. The protocol uses the services of a physical layer, and provides either a best effort or reliable communications path between the transmitter and receiver (i.e. with acknowledged data transfer). The type of service provided depends upon the HDLC mode which is used.

Each piece of data is encapsulated in an HDLC frame by adding a trailer and a header. The header contains an HDLC address and an HDLC control field. The trailer is found at the end of the frame, and contains a Cyclic Redundanc Check (CRC) which detects any errors which may occur during transmission. The frames are separated by HDLC flag sequences which are transmitted between each frame and whenever there is no data to be transmitted.

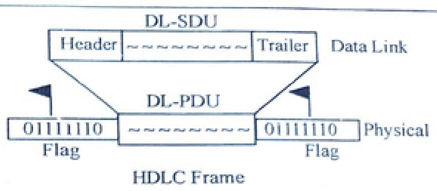
DL-SDU
Header
~~~~~~~~~
Trailer
Data Link
DL-PDU
01111110
~~~~~~~~~
01111110
Physical
Flag
Flag
HDLC Frame

**Fig.12**

HDLC frame structure showing flags, header (address and control), data and trailer (CRC-16).

### 3. Network layer :

The network layer is responsible for the source to destination delivery of a packet, possibly across multiple network (links). Whereas the data link layer oucrese as the delievery of packet between two systems on the some network (link) with connecting devices between the networks (link), there is often a need for the network layer to accomplish source to destination delivery. Fig. 7 shows the relationship of the network layer to the data link and transport layers.

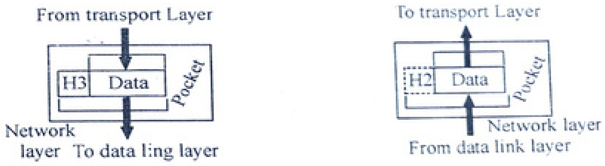

**Fig. 13 : Network layer**

The network layer is responsible for the delivery for the delivery of individual packets from the source host to the destination host

Other responsibilities of the network layer include the following :

**Logical addresing :** The physical addressing implemented by the data link layer handels the addressing problem locally. If a packet passes the newtork boundry we need another addressing system to help distinguish the source and destination systems. The netwwork layer adds a header to the packet coming from the uper layer that, among other thing, includes the logical addresses of the sender and reciever. We discuss logical addresses later in this chapter.

**Routing :** When independent network or links are connected to create internetworks (network of networks) or a large networks, the connecting devices (called routers or switches) route or switch the packets to their final destination. One of the functions of the network layer is to provide this mechanism.

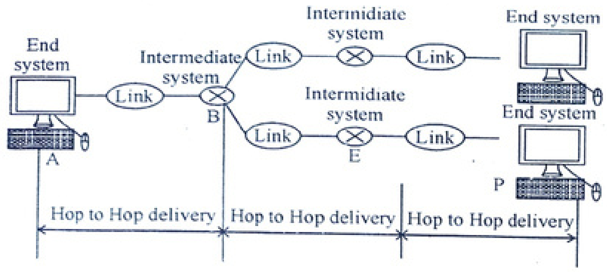

Fig. 14 : Source to destination delievery

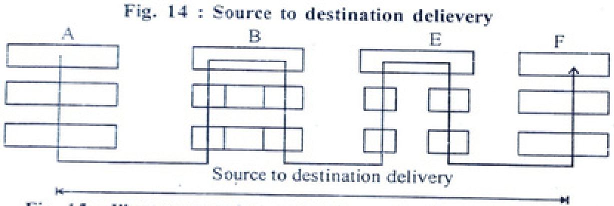

Fig. 15 : Illustrates end to end delivery by the network layer.

As figure shows, now we need a source-to-destination delivery. The network layer at A sends the packet to the network !ayer at B. When the packet arrives at router B, the router makes a decision based on the final destination (F) of the packet. As we will see in later chapters, router B uses its routing table to find that the next hop is router E. the network layer at B, therefore, sends the packet to the network layer at E. The network layer at E, in turn sends the packet to the network layer at F.

## 4. Transport layer :

The transport layer is responsible for process to process delivery of the entire message. A process is an application program running on a host. Whereas the network layer ouerseas source-to-destination delivery of indivisual packets, if does not recognise any relationship between those packets. If treats each one independently, as though each piece belonged to a separate message, whether or

not it does. The transport layer, on the other hand, ensures the whole message arrives inact and in order, overseling both error control at the source to destination level. Fig. 16. shows the relationship of the transport layer to the network and session layers.

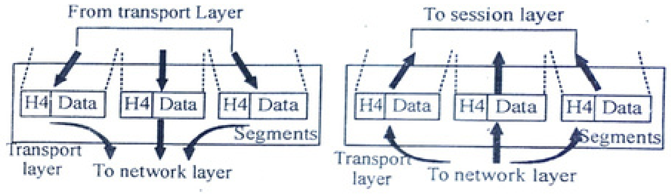

Fig. 16 : Transport layer

The transport layer is responsible for the delivery of a message from one process to another.

Other responsibilities of the transport layer include the following :

**Service point addressing :** Computers often run several programs at the same time. For this reason, source-to-destination delivery means delivery not only from one computer to a specific process (running program) on the other. The transport layer header must therefore include a type of address called a service point address (or port address). The network layer gets each packet to the correct computer, transport layer gets the entire message to the correct process on that computer.

**Segmentation and reassembly :** A message is divided into transmitteble segments, with each segment containing a sequence number. These numbers enable the transport layer to resemable the message correctly upon arriving at the destintion and to identify and replace packets that were lost intransmission.

. **Connection control :** The transport layer can be either connection less or connection oriented. A connectionless transport layer treats each segments as an independent packet and delivers it to the transport layer at the destination machine. A connection oriented transport layer makes a connection with the transport layer at the destiration machine first before delieving the packets. After all the data are transfered, the connection is terminated.

**Flow control :** Link the data link layer, the transport layer is responsible for flow control. However flow control at this layer is performed end to end rather than across a single link.

- At the physical level X.21 physical interface is being used which is defined for circuit switched data network. At the data link level, X.25 specifies the link access procedure-B (LAP-B) protocol which is a subset of HDLC protocol.
- At the network level (3rd level), X.25 defines a protocol for an access to packet data subnetwork.
- This protocol defines the format, content and procedures for exchange of control and data transfer packets. The packet layer provides an external virtual circuit service.

**Characteristics of X.25**

In addition to the characteristics of the packet switched network, X.25 has the following characteristics:

1. Multiple logical channels can be set on a single physical line
2. Terminals of different communication speeds can communicate
3. The procedure for transmission controls can be changed.

**Error control :** Like a data link layer, the transport layer is responsible for process rather than across a single link. he sending transport layer makes sure that the entire message arrives at the recieving transport layer without error (damage, loss or duplication). Error correction achieved through retransmission.

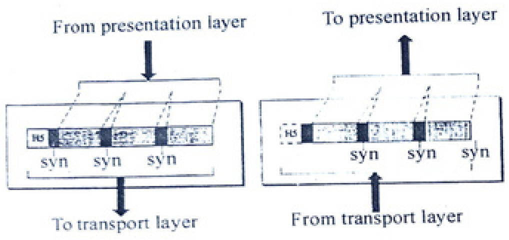

**Fig. 17 : Illustrate process to process delievery by transport layer.**

## 5. Session layer :

The services provided by the first three layers (physical, data link, and network) are not sufficient for some processes. The session layer is the network dialog controller. It establishes, maintians and synchronizes the interaction among communicating systems.

The session layer is responsible for dialog control and syncronization.

Specific responsibilities of the session layer include the following :

**Dialog Control :** The session layer allows a two systems to enter into a dialog. It allows the communication between two processes to take place in either half duplese (one way at a time) or full duplex (two ways at a time) mode.

**Synchronization :** The session layer allows a process to add checkpoints, or symchronization points, to a stream of data. For example, if a system is sending a file of 2000 pages, it is adjustable to insert checkpoints after every 100 pages to ensure that each 100 page unit is received and acknowledged independintly. In this case, if a crash happens during the transmission of page 523, the only pages that need to be resent after system recovery are pages 501 to 523 pages previous to 501 need to be resent.

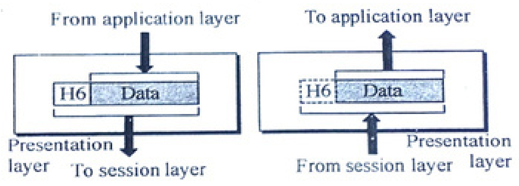

**Fig. 18 : Illustrates the relationship of the session layer to the transport and presentation layer.**

## 6. Presentation layer :

The presentation layer is concerned with the syntax and semantics of the information exchanged between two systems. Fig. 18 shows the relationship between the presentation layer and the application layer and session layers.

Specific responsibilities of the presentatin layer include the following :

**Translation :** The processes (running programs) in two systems are usually exchanging information in the form of character strings, numbers, and so on. The information must be changed to bit streams before being transmitted. Because different computers use different encoding systems, the presentation layer is responsible for interportability between these different encoding methods. The presentation layer at the sender changes the information from its sender-dependent format into a common format. The presentation layer at the receiving machine changes the common format into its receiver-dependent format.

**Encryption :** To carry sensitive information, a system must be able to ensure privacy. Encryption means that the sender transforms the original information to another form and sends the resulting message out over the network. Decryption reverses the original process to transform the message back to its original form.

**Compression :** Data compression reduces the number of bits contained in the information. Data compression becomes particularly important in the transmission of multimedia such as text, audio, and video.

## 7. Application Layer

The application layer enables the user, whether human or software, to access the network. It provides user interfaces and support for services such as electronic mail remote file access and transfer, shared database management, and other types of distributed information services.

Figure 19 shows the relationship of the application layer to the use and the presentation layer. Of the many application services available, the figure shows only three X.400 (message-handling services), X.500 (directory services), and file transfer, access, and management (FTAM). The user in this example employs X.400 to send an e-mail message.

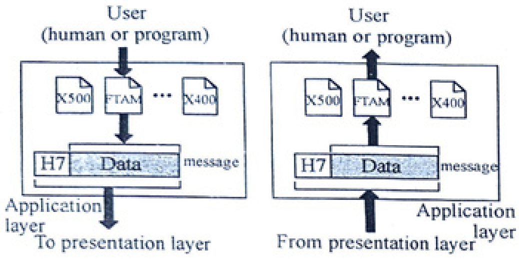

Fig. 19 : Application layer

**The application layer is responsible for providing services to the user.** Specific services provided by the application layer include the following :

**Network virtual terminal :** A network virtual terminal is a software version of a physical terminal, and it allows a user to log on to a remote host. To do so, the application creates a software emulation of a terminal at the remote host. The user's computer talks to the software terminal which, in turn, talks to the host, and vice versa. The remote host believes it is communicating with one of its own terminals and allows the user to log on.

## 2.8 TCP/IP Protocol Suite

The TCP/IP protocol suit was developed prior to the OSI model therefore, the layers in the TCP/IP protocol suit do not exactly mateh those in the OSI model. The original TCP/IP protocol suite was defined as having four layers : host to network, internet, transport, and application. However when TCP/IP is compared to OSI, we can say that the host to network layer is equivalent to the combination of the physical and data link layers. The internet layer is equivalent to the network layer and the application layer is roughly doing the job of the session, presentation, and application layers with the transport layer in TCP/IP protocol taking care of paut of the duties of the session layer. So we assume that the TCP/IP protocol suit is made of two layers procide physical datalink, network, transport and application. The first four layer provid. Physical standard, network interface, internet working, and transport functions that correspond to the first four layers of OSI model. the three topniost layers in the OSI model, however, are represented in TCP/IP by a single layer called the application layer. TCP/IP is a hierarehical

protocol made up of interacture modules, each of which provides a specific functionality; however, the modules are not necessarily inter dependent.

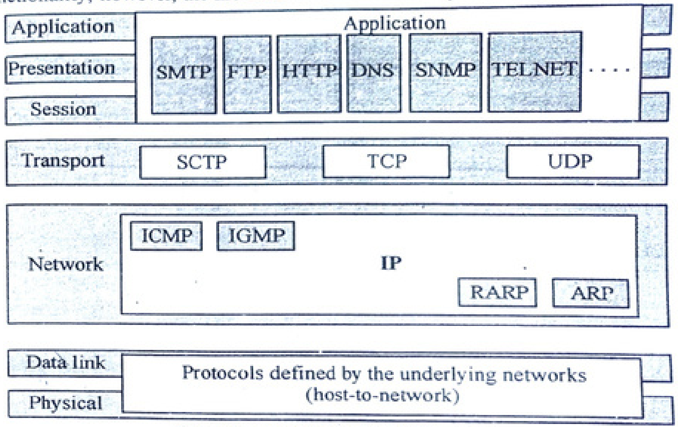

Fig. 20 : TCP/IP and OSI model

Where as the OSI model specifies which functions belong to each of its layers, the layers of the TCP/IP protocol suite contain relatively independent protocols that can be mixed and matched depending on the needs of the system. The term hierarchical means that each upper level protocol is supported by one or more lower levels protocols.

At the transport layer, TCP/IP defines three protocols; Transmission control protocol (TCP), User Datagram Protocol (UDP), and Stream Control Transmission Protocol (SCTP). It the network layer, the main protocol defines by TCP/IP is Internetworking Protocol (IP); there are also some other protocols that support data movement in this layer.

**File transfer, access, and management :** This application allows a user to access files in a remote host (to make changes or read data), to retrieve files from a remote computer for use in the local computer, and to manage or control fiels in a remote computer locally.

**Mail services :** This application provides the basis for e-mail forwarding and storage.

**Directory services :** This application provides distributed database sources and access for global information about various objects and services.

**Summary of Layers**

Figure 21 shows a summary of duties for each layer.

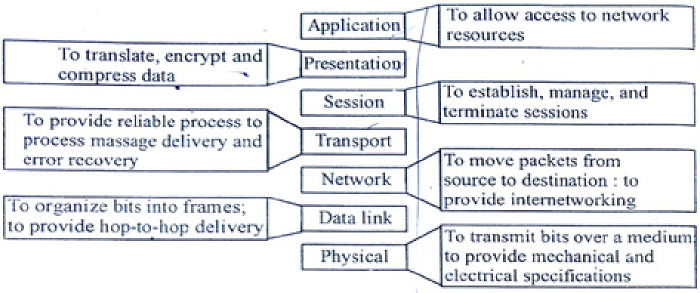
Application
To allow access to network resources
To translate, encrypt and compress data
Presentation
Session
To establish, manage, and terminate sessions
To provide reliable process to process massage delivery and error recovery
Transport
Network
To move packets from source to destination : to provide internetworking
To organize bits into frames; to provide hop-to-hop delivery
Data link
Physical
To transmit bits over a medium; to provide mechanical and electrical specifications

Fig. 21 : Summary of Layers

## Physical, Data link layers

At the physical and data link layer, TCP/IP does not define any specific protocol. It support all the specific standard and proprietery protocols. A network in a TCP/IP internetwork can be a local-area network or a wide area network.

## Network layer

At the network layer (or, more accurately, the internetworking protocol. IP, in turn, uses four suppouting protocols protocols : ARP, RARP, ICMP and IGMP. Each of these protocols is descuibed in greater details in later chapters.

### Internetworking Protocol (IP)

The Internetworking Protocol (IP) is the transmission mechanism used by the TCP/IP protocols. It is an unreliable and connectionless protocol-a best-effort delivery service. The term best effort means that IP provides no error checking or tracking. IP assumes the unreliability of the underlying layers and does its best to get a transmission through to its destination, but with no guarantees.

IP transports data in packets called datagrams, each of which is transported separately. Datagrams can travel along different routes and can arrive out of sequence or be duplicated. IP does not keep track ofthe routes and has no facility for reordering datagrams once they arrive at their destination.

The limited functionality of IP should not be considered a weakness, however. IP provides bare-bones transmission functions that free the user to add only those facilities necessary for a given application and thereby allows for maximum efficiency.

### Address Resolution protocol

The **Address Resolution Protocol (ARP)** is used to associate a logical address with a physical address. On a typical physical network, such as a LAN, each device on a link is identified by a physical or station address, usually imprinted on the network interface card (NIC). ARP is used to find the physical address of the node when its Internet address is known.

### Reverse Address Resolution Protocol

The **Reverse Address Resolution Protocol (RARP)** allows a host to discover its Internet address when it knows only its physical address. It is used when a computer is conected to a network for the first time or when a diskless computer is booted.

### Internet Control Message Protocol

The **Internet Control Message Protocol (ICMP)** is a mechanism used by hosts and gateways to send notification of datagram problems back to the sender. ICMP sends query and error reporting messages.

### Internet Group Message Protocol

The **Internet Group Message Protocol (IGMP)** is used to facilitate the simultaneous transmission of a message to a group of recipients.

### Transport Layer

Traditionally the transport layer was represented in TCP/IP by two protocols; TCP and UDP. IP is a **host-to-host protocol,** meaning that it can deliver a packet from one physical device to another. UDP and TCP are **transport level protocols** responsible for delivery of a message from a process (running program) to another process. A new transport layer protocol, SCTP, has been devised to meet the needs of some newer applications.

### User Datagram Protocol

The **User Datagram Protocol (UDP)** is the simpler of the two standard TCP/IP transport protocols. It is a process-to-process protocol that adds only port addresses, checksum error control, and length information to the data from the upper layer.

### Transmission Control Protocol

The **Transmission Control Protocol (TCP)** provides full transport-layer services to applications. TCP in a reliable stream transport protocol. The term stream, in this context, means connection-oriented; A connection must be established between both ends of a transmission before either can transmit data.

At the sending end of each transmission, TCP divides a stream of data into smaller units called segments. Each segment includes a sequence number for reordering after receipt, together with an acknowledgement number ofr the segments received. Segments are carried across the internet inside of IP datagrams.

At the receiving end, TCP colects each datagram as it comes in and recorders the transmission based on sequence numbers.

### Stream Control Transmission Protocol

The **Stream Control Transmission Protocol (SCTP)** provides support for newer application such as voice over the Internet. It is a tranport layer protocol that combines the best features of UDP and TCP.

### Application Layer

The application layer in TCP/IP is equivalent to the combined session, presentation, and application layers in the OSI model. Many protocols are defined at this layer. We cover many of the standard protocols in later chapters.

### Addressing

Four levels of addresses are used in an internet employing the TCP/IP protocols : physical (link) addresses, Logical (IP) addresses, port addresses, and specific addresses (see Figure 22).

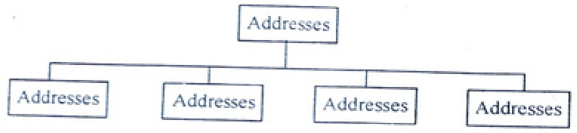

**Fig. 22 : Addresses in TCP/IP**

Each address is related to a specific layer in the TCP/IP architecture, as show in Figure 23.

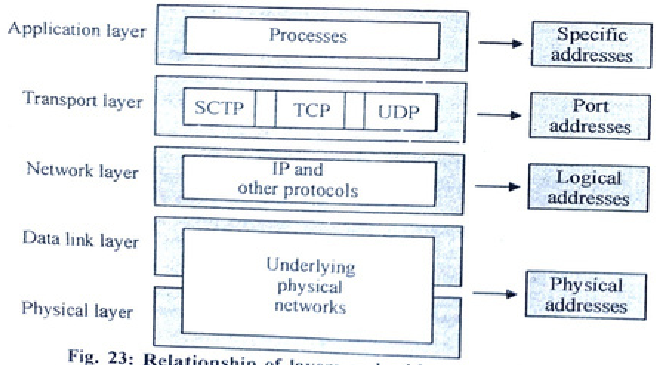

**Fig. 23: Relationship of layers and addresses in TCP/IP**

### Physical Addresses

The physical address, also known as the link address, is the address of a node as defined by its LAN or WAN. It is included in the frame used by the data link layer. It is the lowest-level address.

The physical addresses have authority over the network (LAN or WAN). The size and format of these addresses vary depending on the network. For example, Ethernet uses a 6-byte (48-bit) physical address that is imprinted on the network interface card (NIC). Local Talk (Apple), however, has a 1-byte dynamic address that changes each time the station comes up.

**Example 2.1 :** In Figure 24 a node with physical address 10 sends a frame to a node with physical address 87. The two nodes are connected by a link (bus topology LAN). At the data link layer, this frame conains physical (link) addresses in the header. These are the only addresses needed. The rest of the header contains other information needed at this level. The trailer usually contains extra bits needed for error detection. As the figure shows, the computer with physical address 10 is the sender, and the computer with physical adress 87 is the receiver. he data link layer at the sender receives data from an upper layer. It encapsulates the data in a frame, adding a header and a trailer. The header, among other pieces of information, carries the receiver and the sender physical (link) addresses. Note that in most data link protocols, the destination address, 87 in this case, comes before the source address (10 in this case).

We have shown a bus topology for an isolated LAN. In a bus topology, the frame is propagated in both directions (left and right). The frame propagated to the left dies when it reaches the end of the cable if the cable end is terminated appropriately. The frame propagated to the right is sent to every station on the network. Each station with a physical address other than 87 drops the frame because the destination address in the frame does not match its own physical address. The intended destination computer, however, finds a match between the destintion address in the frame and its own physical address. The frame is checked, the header and trailer are dropped, and the data part is decapsulated and delivered to the upper layer.

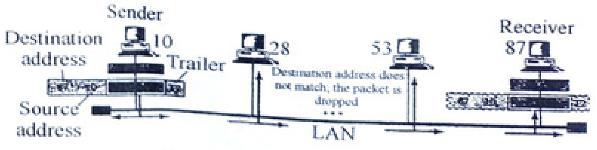

**Fig. 24 : Physical addresses**

**Example 2.2 :** Local-area networks use a 48-bit (6-byte) physical address written as 12 headecimal digits; every byte (2 hexadecimal digits) is separated by a colon, as shown below :

### Logical Addresses

Logical addresses are necessary for universal communications that are independent of underling physical network. Physical addresses are not adequate in an internetwork environment where different network can have different address formates. A universal addressing system is needed in which each host can be identified uniquely, regardless of the underlying physical network.

The logical address are designed for this purpose. A logical address in the Internet is currently a 32-bit address that can uniquely define a host connected to the Internet. No two publicly addressed and visible hosts on the Internet can have the same IP address.

**Example 2.3 :** Figure 25 shows a part of an internet with two routes connecting three LANs. Each device (computer or router) has a pair of addresses (logical and physical) for each connection. In this case, each computer is connected to only one link and therefore has only one pair of addresses. Each router, however, is connected to three networks (only two are shown in the figure). So each router has three pairs of addresses, one for each connection. Although it may obvious that each router must have a separate physical address for each connection, it may not be obvious why it needs a logical address for each connection.

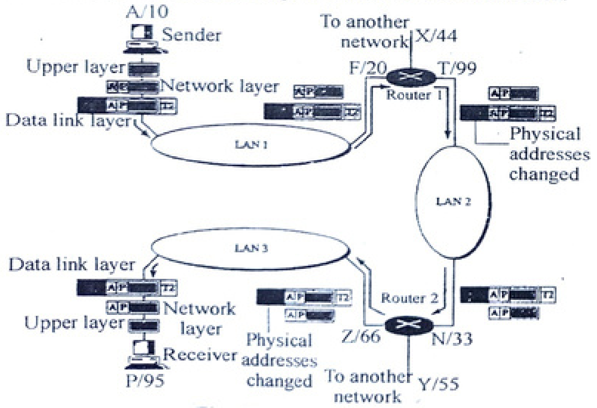

Fig. 25 : IP Addresses

The computer with logical address A and physical address 10 needs to send a packet to the computer with logical address P and physical address 95. We used letters to show the logical addresses and numbers for physical addresses, but not that both are actually numbers, as we will see later in the chapter.

The sender encapsulates its data in a packet at the network layer and adds two logical addresses (A and P). Note that in most protocols, the logical source address comes before the logical destination address (contrary to the order of physical addresses). The network layer, however, needs to find the physical address of the next hop before the packet can be delivered. The network layer consults its routing table (see Chapter 22) and find the logical address of the next hop (router 1) to be F. The ARP discussed previously finds the physical address of router 1 that corresponds to the logical address of 20. Now the network layer passes this address to the data link layer which in turn, encapsulates the packet with physical destination address 20 and physical source address 10.

The frame is received by every device on LAN 1, but is discarded by all except router 1, which finds that the destination physical address in the frame matches with its own physical address. The router decapsulates the packet from the frame to read the logical destination address P. Since the logical destination address does not match the router's logical address, the router knows that the packet needs to be forwarded. The router consults its routing table and ARP to find the physical destination address of the next hop (router 2), creates a new frame, encapsulates the packet, and sends it to router 2.

Note the physical addresses in the frame. The source physical address changes from 10 to 99. The destination physical address changes from 20 (router 1 physical address) to 33 (router 2 physical address) The logical source and destination addresses must remain the same; otherwise the packet will be lost.

At router 2 we have a similar scenario. The physical addresses are changed, and a new frame is sent to the destination computer. When the frame reaches the destination, the packet is decapsulated. The destination logical address P matches the logical address of the computer. The data are decapsulated from the packet and delivered to the upper layer. Note that although **physical address will change from hop to hop, logical addresses remain the same** from the source to destination. There are some exceptions to this rule that we discover later in the book.

**The physical addresses will change from hop to hop, but the logical addresses usually remain the same.**

### Port Addresses

The IP address and the physical address are necessary for a quantity of data to travel from a surce to the destination host. However, arrival at the destination host is not the final objectie of data communications on the Internet. A system that sends nothing but data from one computer to another is not complete. Today, computers are devices that can run multiple processes at the same time. The end objective of Internet communication is a process communicating with another process. For example, computer A can communicate with computer C by using TELNET. At the same time, computer A communicates with computer B by using the File Transfer Protocol (FTP). For these processes to receive data simultaneously, e need a method to label the different processes. In other words, they need addresses. In the TCP/IP architecture, the label assigned to a process is called a port address. A port addresss in TCP/IP is 16 bits in length.

**Example 2.4 :** Figure 26 shows two computers communicating via the Internet. The sending computer is running three processes at this time with port addresses a, b and c. The receiving computer is running two processes at this time with port addresses j and k. Process a in the sending computer needs to communicate with process j in the receiving computer. Note that although both computers are using the same application, FTP, for example, the port addresses are different because one is a client program and the other is a server program. To show that data from process a need to be dlivered to process j, and not k, the transport layer encapsulates data from the application layer in a packet and adds two port addresses (a and j), source and destination. The packet from the transport layer is then encapsulated in another packet at the network layer with logical source and destination addresses (A and P). Finally, this packet is encapsulated in a frame with the physical source and destination addresses of the next hop. We hae not shown the Internet. Note that although physical addresses change from hop to hop logical and port addresses remain the same from the source to destination. There are some exceptions to this rule that we discuss later in the book.

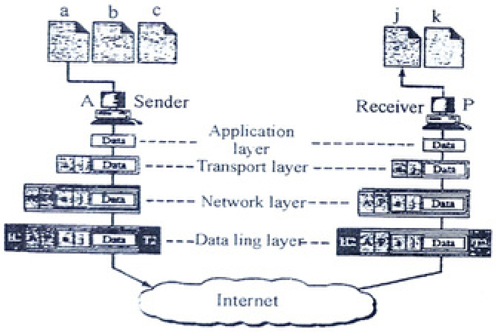

**Figure 26 : Port addresses**

**The physical addresses change from hop to hop, but the logical and port address usually remain the same.**

**Example 2.5 :** a port address is a 16-bit address represented by one decimal

number as shown.

753

A 16-bt port address represented as one single number

**Specific Addresses**

Some applications have user-friendly addresses that are designed for that specific address. Examples include the e-mail address (for example, forouzan @ fhda. edu) and the Universal Resource Locator (URL) (for example, www.mhhe.com). The first defines the recipient of an e-mail; the second is used to find a document on the World Wide Web. These addresses, however, get changed to the corresponding port and logical addresses by the sending computer.

## Exercises

**Very Short Questions** **[2 marks each]**

1. Which of the following layer is not a property of a packet filtering firewall?
2. Which layer filters the proxy firewall ?
3. What is port address ?
4. What is data link layer ?
5. What is network layer?
6. Explain Bridges.
7. What is error detection.
8. What is interface.
9. What is OSI model ? Give diagram.
10. Explain protocol.

**Short Questions** **[4 marks each]**

1. Explain OSI model with diagram.
2. How many types of layers we use. Explain
3. What is transmission media ?
4. What is transmission control protocol
5. Explain peer-to-peer process.

**Long Questions** **[12 marks each]**

1. What is OSI model ? Explain seven layers with diagrams.
2. What is error detection and corrections? Explain.
3. What is bridge and explain network routing algorithm.
4. What is transmission median and wireless transmission ?
5. What is bandwidth ? Explain with Network layer ?