Nalanda Open University

Course Name: BCA Part II

Paper-X (Networking)

Topic- MULTIPLEXING AND SWITCHING

Coordinator: A. N. Pandey

E-mail ID : amarnathpandey@nou.ac.in

UNIT 4-MULTIPLEXING AND SWITCHING

4.1 INTRODUCTION

Nowadays it is possible to send pictures, images, and tweets through the internet without any superimposition of data from one source or the other. The transmission of data over a network, whether wireless or not, requires us to solve the problem of being able to send a large amount of data among different groups of recipients at the same time without the message of one being mixed up with those of the others. For example, think of the thousands of radio or television channels that are being broadcast throughout the world simultaneously, or of the millions of telephone conversations that are taking place every minute of the day. This feat is achieved by multiplexing the transmissions in some way. In this unit, we will look at the different methods of multiplexing that are commonly used, frequency division multiplexing and time division multiplexing. Time division multiplexing again can be synchronous or statistical.

Here we will see the techniques that make this sort of data transfer a possible reality. For example, if you send a message or email from your PC to your friend who resides in a different city, then your mail joins enroute with your neighborhood and other mails as well. After being clubbed together with other mails, your mail is fed into a larger transmission line, and then joined with other mails from your city. How do all these mails get joined together and transmitted without getting mixed up?

This process is done by a technique called Multiplexing. Multiplexing combines multiple analog or digital signals bound for transmission through a single communication line or computer channel. This technique has been introduced to increase channel utilization in multicomputer communication systems and time sharing systems and also to reduce the communication

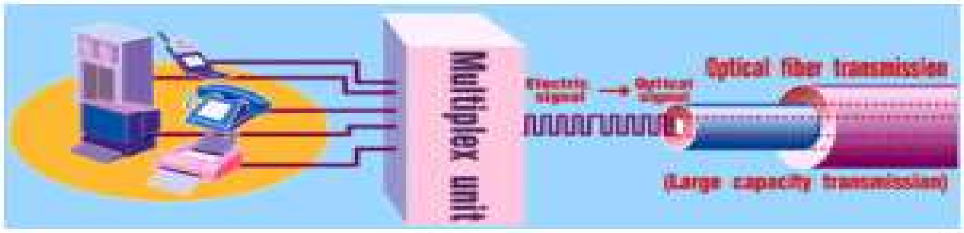

Figure 1: Multiplexing

There are some differences in the characteristics of voice and data communication. Although, finally everything is really data, these differences dictate the design of the telephone network as well as the techniques of switching. When one wants to transmit data over a telephone line, we have to make some changes to be able to get sufficiently high speeds. We will look at ADSL, a scheme for doing this at rates that can compete on cost and quality with those offered by cable television service providers. We will also look at the advantages and disadvantages of the two methods and of high-speed data access.

When constructing networks of any size, a fundamental requirement is that any two nodes of the network should be able to communicate with each other, for, if they cannot do so, they are not on the same network. This brings up the issue of how to switch the data stream, that is, to make sure it reaches its destination and not some other random location on the network. This unit will, describe the different switching techniques available to make this possible. We conclude the Unit with a brief summary followed by an exercise and some suggested readings for the students.

4.2 OBJECTIVES

After going through this unit, you should be able to:

- state what multiplexing and switching mean;
- state how a telephone line can be used for transmitting data using ADSL;
- describe the differences between ADSL and cable television for data access;
- describe the different kinds of multiplexing and switching;
- state the characteristics of Frequency Division Multiplexing;
- describe the features of Time Division Multiplexing, both synchronous and statistical, and
- differentiate between the different kinds of switching.

4.3 MULTIPLEXING

If, one wanted to send data between a single source and a single destination, things would be comparatively easy. All it would need is a single channel between the two nodes, of a capacity sufficient to handle the rate of transmission based on Nyqist's theorem and other practical considerations. Granted that there would be no other claimants for the resources, there would be no need for sharing and no contention.

The transmission channel would be available to the sole users in the world at all times, and any frequency which they chose could be theirs. In a broadcasting case, we could have the single source transmitting at any time of its choosing and at any agreed upon frequency.

However, things are not so straightforward in the real world. We have a large number of nodes, each of which may be transmitting or receiving, from or to possibly different nodes each time. These transmissions could be happening at the same moment. So, when we need to transmit data on a large scale, we run into the problem of how to do this simultaneously, because, there can be many different sources and the intended recipients for each source are often different. The transmission resource, that is the available frequencies, is scarce compared to the large number of users. So, there will have to share the resource and consequently, the issue of preventing interference between them will arise.

If, all possible pairs of nodes could be completely connected by channels, we would still not have a problem. But that is obviously out of the question, given the very large number of possible source and destination nodes. There would be over a billion telephones in the world today, for instance.

This problem is solved by performing what is called multiplexing. It can be done by sharing the available frequency band or by dividing up the time between the different transmissions. The former is called Frequency Division Multiplexing (FDM) while the other scheme is Time Division Multiplexing (TDM).

Another reason for multiplexing is that it is more economical to transmit data at a higher rate as the relative cost of the equipment is lower. But at the same time, most applications do not require the high data rates that are technologically possible. This is an added inducement to multiplex the data so that, the effective cost can be brought down further by sharing across many different, independent transmissions.

In this context, a multiplexer is a device that can accept n different inputs and send out 1 single output. This output can be transmitted over a link or medium to its destination, where to be useful, the original inputs have to be recovered. This is done by a demultiplexer. We need to realise that for this kind of scheme to work, the creation of the composite signal by the multiplexer needs to be such that, the original component signals can be separated at the receiving end – otherwise we would end up with just a lot of noise!

When we transmit data over a cable, we are really setting up a different medium. We can transmit data over different cables at the same frequency at the same time without interference because the media or links are different. Even in radio transmission, where the medium is space and hence is a single medium available to all transmissions, geographical distance can in many cases give rise to different links. A low power medium wave transmission happening in India can be done simultaneously with a similar transmission in Europe without interference or the need for any special precautions. However, throughout the rest of the unit, when we talk of multiplexing, we are referring to the simultaneous transmission of data over the same medium or link.

Frequency Division Multiplexing

In the 20th century, many telephone companies used frequency-division multiplexing for long distance connections to multiplex thousands of voice signals through a coaxial cable system. For shorter distances, cheaper balanced cables were used for various systems like bell systems K-and N-carrier, but they didn't allow large bandwidths. The FDM is an analog multiplexing that combines analog signals. Frequency division multiplexing is applied when the bandwidth of the link is greater than the combined bandwidth of the signals to be transmitted. FDM (Frequency Division Multiplexing) takes advantage of passband transmission to share a channel. It divides the spectrum into frequency bands, with each user having exclusive possession of some band in which to send their signal. AM radio broadcasting. The allocated spectrum is about 1 MHz, roughly 500 to 1500 kHz. Different frequencies are allocated to different logical channels (stations), each operating in a portion of the spectrum, with the interchannel separation great enough to prevent interference.

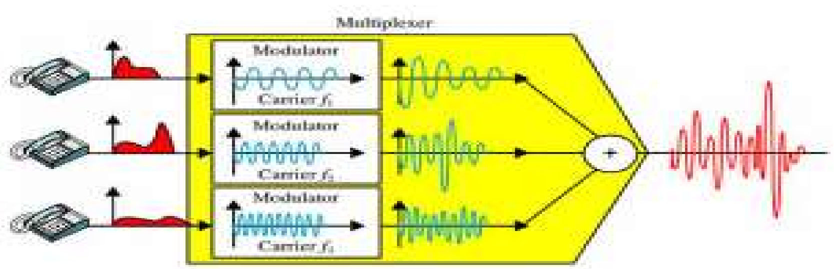

Figure 2: Frequency Division Multiplexing

In this type of multiplexing, signals are generated by sending different device-modulated carrier frequencies, and these modulated signals are then combined into a single signal that can be transported by the link. To accommodate the modulated signal, the carrier frequencies are separated with enough bandwidth, and these bandwidth ranges are the channels through which different signals travel. These channels can be separated by unused bandwidth. Some of the examples for the time division multiplexing include radio and television signal transmission.

The composite signal to be transmitted over the medium of our choice is obtained by summing up the different signals to be multiplexed. The transmission is received at the other end, the destination and there, it has to be separated into its original components, by de-multiplexing. In practice, a scheme like this could result in interference or cross talk between adjacent channels because the band-pass filters that are used to constrain the original data between the agreed upon frequencies (300 to 3400 kHz) are not sharp. To minimise this, there are guard bands, or unused portions of the spectrum between every two channels.

Another possible cause of interference could arise because of the fact that the equipment, such as amplifiers used to increase the strength of the signal, may not behave linearly over the entire set of frequencies that we seek to transmit. Then, the output can contain frequencies that are the sum or difference of the frequencies used by the input. This produces what is called intermodulation noise.

In this kind of division, it should be realised that the actual modulation technique used is not of consequence. So, one could use analog modulation (AM) or Frequency Modulation (FM). Also the composite signal that we have produced could again be modulated over a different frequency altogether. For example, the three voice channels, that have been modulated for commercial broadcast radio to produce a spectrum from 300 kHz to 312 kHz could be modulated onto a 2 GHz satellite channel for long distance transmission to the other side of the earth. This second modulation could use a technique different from the first one. The only thing that needs to be take care of is, that the recovery of the original signals has to be done in the reverse order, complementing the method used for producing the composite signal.

The International Telecommunication Union (ITU) has standardised a hierarchy of schemes that utilise FDM for the transmission of voice and video signals. To begin with, a cluster of 12 voice channels, each of 4 kHz is combined to produce a signal of bandwidth 48 kHz that is then, modulated and transmitted over the 60 to 108 kHz band. This is called a group.

The next level of the hierarchy is the super-group, where 5 such groups are combined to use 240 kHz of bandwidth, occupying from 312 to 552 kHz. The next level is the master-group that consists of 5 such super-groups. This uses 1232 kHz from 812 to 2044 kHz. Again, 3 such super-groups form a super-master group that contains 3.872 MHz from 8.516 to 12.388 MHz. The United States uses a similar hierarchy as decided by AT&T (the telecommunications company) that is not entirely identical to the ITU standard.

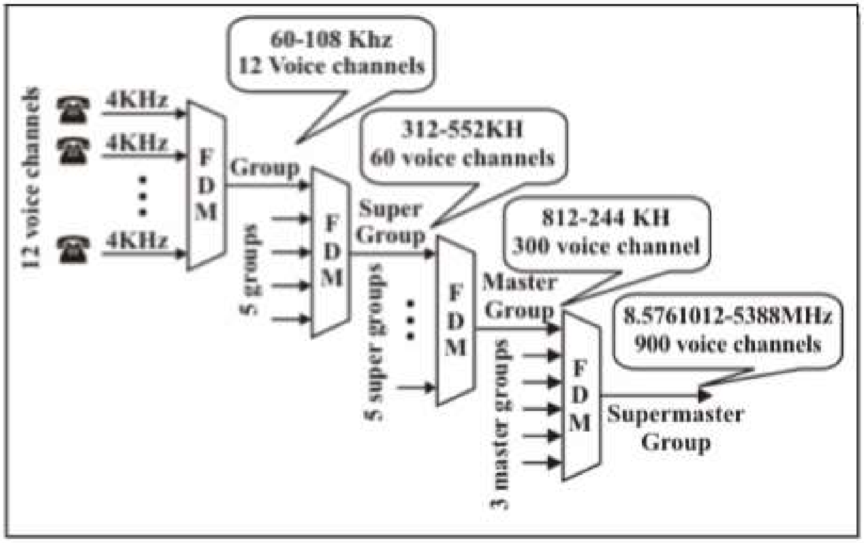

Figure 3: Analog hierarchy

We, thus, see that the original signal could find itself subjected to modulation several times, each of which may be of different kinds, before it is finally transmitted. This raises the possibility of interference and noise, but, with the very good equipment available today, this need not be a matter of concern.

FDM has the disadvantage of not being entirely efficient if the transmissions that are multiplexed together have periods of silence or no data. Since, a frequency band is dedicated to each data source, any such periods are simply not utilised.

EXAMPLE

Let us consider a simple example of transmitting three voice signals simultaneously over a medium. As was mentioned, the bandwidth of a voice signal is generally taken to be 4 kHz, with an effective spectrum of 300 to 3400 Hz (Figure 4a). If such a signal is used to amplitude-modulate a 64-kHz carrier, the spectrum of Figure 4b results. The modulated signal has a bandwidth of 8 kHz, extending from 60 to 68 kHz. To make efficient use of bandwidth, we elect to transmit only the lower sideband. If three voice signals are used to modulate carriers at 64,68, and 72 kHz, and only the lower sideband of each is taken, the spectrum of Figure 4c results.

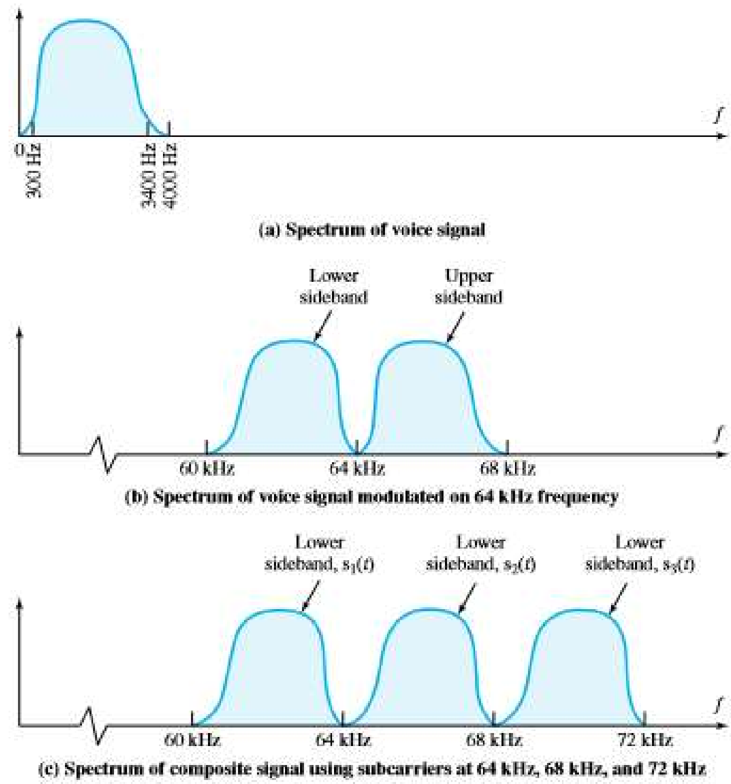

Figure 4 FDM of Three Voiceband Signals

Figure 4 points out two problems that an FDM system must cope with. The first is crosstalk, which may occur if the spectra of adjacent component signals overlap significantly. In the case of voice signals, with an effective bandwidth of only 3100 Hz (300 to 3400), a 4-kHz bandwidth is adequate. The spectra of signals produced by modems for voiceband transmission also fit well in this bandwidth. Another potential problem is intermodulation noise. On a long link, the nonlinear effects of amplifiers on a signal in one channel could produce frequency components in other channels.

Wavelength Division Multiplexing

Wavelength division multiplexing (WDM) is a technology in fiber optic communications; and, for the high capacity communication systems, wavelength division multiplexing is the most promising concept. This system uses multiplexer at transmitter to join signals and demultiplexer to split the signals apart, at the receiver end. The purpose of WDM is to combine multiple light sources into a single light source at the multiplexer; and, at the demultiplexer the single light is converted into multiple light sources.

WDM is designed to use the high data rate capability of the fiber optic cable. The data rate of this cable is higher than the metallic transmission cable's data rate. Conceptually, the wavelength division multiplexing is same as the frequency division multiplexing, except for the transmission through the fiber optic channels wherein the multiplexing and demultiplexing involves optical signals.

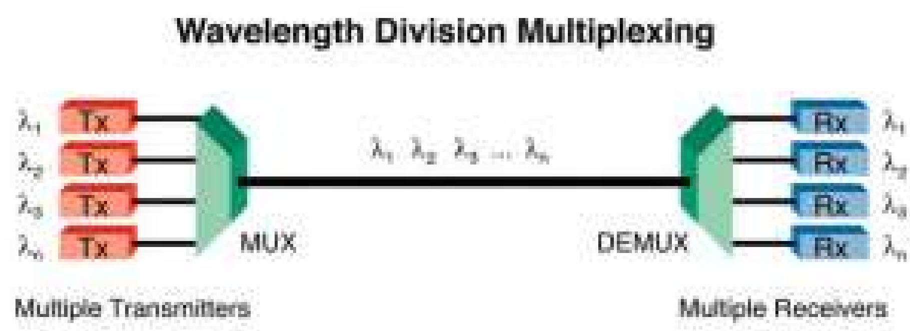

Figure 5: Wavelength Division Multiplexing

Time-Division Multiplexing

Time division multiplexing is a technique used to transmit a signal over a single communication channel by dividing the time frame into slots – one slot for each message signal. Time-division multiplexing is primarily applied to digital signals as well as analog signals, wherein several low speed channels are multiplexed into high-speed channels for transmission. Based on the time, each low-speed channel is allocated to a specific position, where it works in synchronized mode. At both the ends, i.e., the multiplexer and demultiplexer are timely synchronized and simultaneously switched to the next channel.

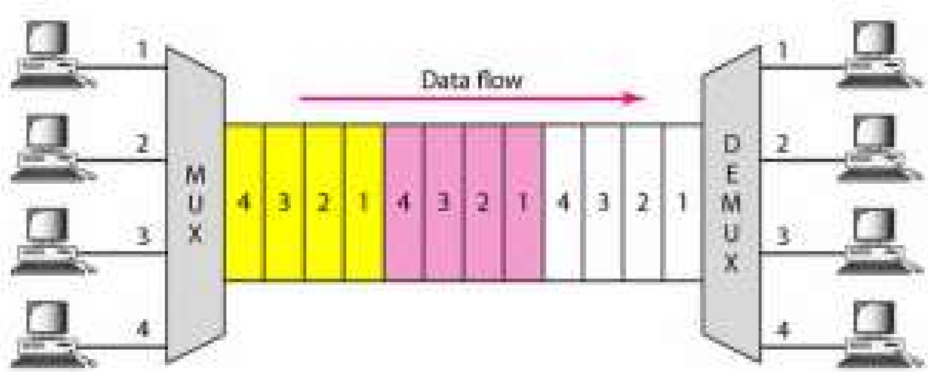

Figure 6: Time-Division Multiplexing

In this case, we have four different transmissions occurring with each being ¼ of the time slice. The slice should be small enough so that the slicing is not apparent to the application. So, for a voice transmission a cycle of 100ms could be sufficient as we would not be able to detect the fact that there are delays. In that case, each transmission could be allotted a slice of 25ms.

At the receiving end, the transmission has to be reconstructed by dividing up the cycle into the different slices, taking into account the transmission delays. This synchronisation is essential, for if the transmission delay is, say 40ms, then the first transmission would start reaching 40 ms later, and would extend to 65ms. If, we interpreted it to be from 0 to 25 ms, there would be complete loss of the original transmission. This interleaving of the signal could be at any level starting from that of a bit or a byte or bigger.

Synchronous Time Division Multiplexing

Synchronous time division multiplexing can be used for both analog and digital signals. In synchronous TDM, the connection of input is connected to a frame. If there are ‘n’ connections, then a frame is divided into ‘n’ time slots – and, for each unit, one slot is allocated – one for each input line. In this synchronous TDM sampling, the rate is same for all the signals, and this sampling requires a common clock signal at both the sender and receiver end. In synchronous TDM, the multiplexer allocates the same slot to each device at all times.

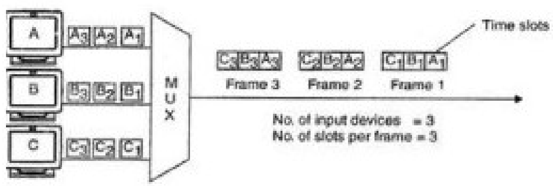

Figure 7: Synchronous Time Division Multiplexing

In this kind of Time Division Multiplexing (TDM), the simpler situation is where the time slots are reserved for each transmission, irrespective of whether it has any data to transmit or not. Therefore, this method can be inefficient because many time slots may have only silence. But it is a simpler method to implement because the act of multiplexing and demultiplexing is easier. This kind of TDM is known as Synchronous TDM. Here, we usually transmit digital signals, although the actual transmission may be digital or analog. In the latter case, the composite signal has to be converted into analog data by passing it through a modem. Here, the data rate that the link can support has to at least equal the sum of the data rates required for each transmission.

So, if one has to transmit four streams of data at 2400 bps each, we would need a link that can support at least 9600 bps. For each data stream, we would typically have a small, say 1 character buffer, that would take care of any data flow issues from the source.

Although, the transmission has to be synchronous, that is not the reason the scheme is called Synchronous TDM. It is because of the fact that each data source is given fixed slots or time slices. We can also support data sources with differing data rates. This could be done by assigning fewer slots to slower sources and more slots to the faster ones. Here, a complete cycle of time slots is called a frame. At the receiving end, the frame is decomposed into the constituent data streams by sending the data in each time slot to the appropriate buffer. The sequence of time slots allotted to a single data source makes up a transmission channel.

We have already seen that the transmission in TDM must be synchronous because otherwise, all the data would be garbled and lost. How is this achieved? Because this unit is concerned with the physical layer and not the data link layer, we will not concern ourselves with the problem of flow control or error correction or control. But even at the physical layer, we have to ensure that each frame is synchronised to maintain the integrity of the transmission.

One scheme to attain this is to add an extra bit of data to each frame. In succeeding frames, this extra bit forms a fixed pattern that is highly unlikely to occur naturally in the data. The receiver can then use this fact to synchronise the frames. Suppose each frame has n bits. Starting from anywhere, the receiver compares bit number 1, n+1, 2n+1 and so on. If the fixed data pattern is found, the frames are synchronised. Otherwise, the receiver commences to check bit number 2, n+2, 2n+2, … until it can determine whether the frames are in synchronization or otherwise. This continues till, it is able to synchronise the frames and can then start receiving the data from the various channels.

Even after this, it is important to continue to monitor the fixed pattern to make sure that the synchronisation remains intact. If it is lost, the routine of re-establishing it needs to be repeated as before.

The second problem in synchronising the transmission is the fact that there can be some variations between the different clock pulses from the different input streams. To take care of this, we can use the technique of pulse stuffing. Here, the multiplexed signal is of a rate that is a bit higher than that of the sum of the individual inputs. The input data is stuffed with extra pulses as appropriate by the multiplexer at fixed locations in the frame. This is needed so that the demultiplexer at the other, receiving, end can identify and remove these extra pulses. All the input data streams are thus synchronised with the single local, multiplexer clock.

Asynchronous Time-Division Multiplexing

In asynchronous time-division multiplexing, the sampling rate is different for different signals, and it doesn't require a common clock. If the devices have nothing to transmit, then their time slot is allocated to another device. Designing of a commutator or de-commutator is difficult and the bandwidth is less for time-division multiplexing. This type of time-division multiplexing is used in asynchronous transfer mode networks.

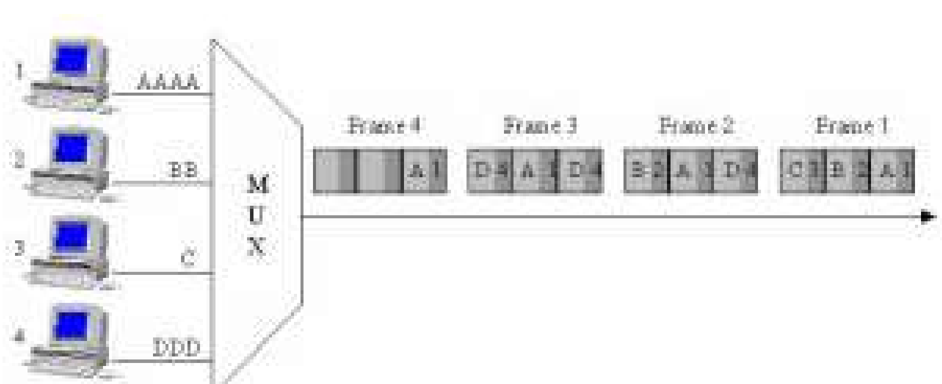

Figure 8: Asynchronous Time-Division Multiplexing

Interleaving

Time-division multiplexing can be visualized as two fast rotating switches on the multiplexing and demultiplexing side. At the same speed these switches rotate and synchronize, but in opposite directions. When the switch opens at the multiplexer side in front of a connection, it has the opportunity to send a unit into the path. In the same way, when the switch opens on the demultiplexer side in front of a connection that has the opportunity to receive a unit from the path. This process is called interleaving.

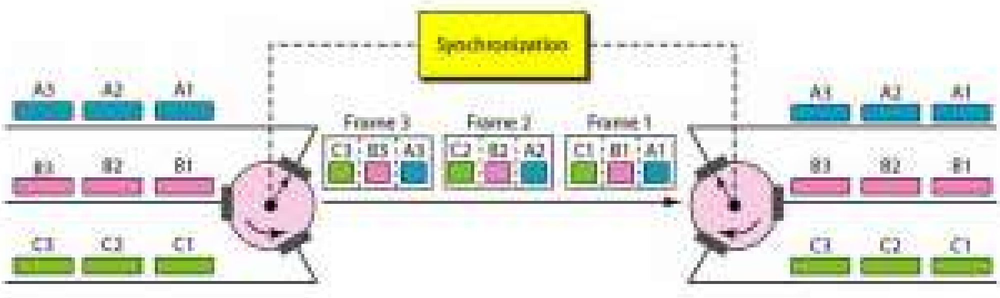

Figure 9: **Interleaving**

Statistical Time-Division Multiplexing

Statistical time-division multiplexing is used to transmit several types of data concurrently across a single transmission cable. This is often used for managing data being transmitted via LAN or WAN. The data is simultaneously transmitted from the input devices that are connected to the network including printers, fax machines, and computers. This type of multiplexing is also used in telephone switch board settings to manage the calls. Statistical TDM is similar to dynamic bandwidth allocation, an in this type of time-division multiplexing, a communication channel is divided into an arbitrary number of data streams.

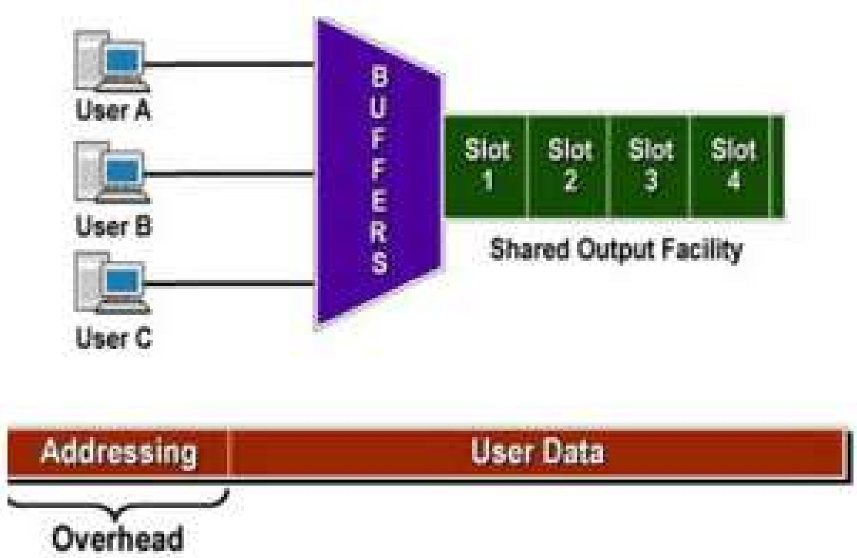

Figure 10: Statistical Time-Division Multiplexing

We have seen that synchronous TDM can be quite wasteful. For example, in a voice transmission, much of the bandwidth can be wasted because there are typically many periods of silence in a human conversation. Another example, could be, that of a time sharing system where many terminals are connected to a computer through the network. Here too, many of the time slots will not be used because there is no activity at the terminal. In spite of being comparatively simpler, synchronous TDM is therefore, often not an attractive option.

To take care of this problem, we can use statistical TDM. This is a method where there are more devices than the number of time slots available. Each input data stream has a buffer associated with it. The multiplexer looks at the buffer of each device that provides the input data and checks to see if it has enough to fill a frame. If, there are enough, the multiplexer sends the frame. Otherwise, it goes to the next device in line and checks its input buffer. This cycle is continued indefinitely. At the receiving end the demultiplexer decomposes the signal into the different data streams and sends them to the corresponding output line.

The multiplexer data rate is not higher than that of the sum of all the input devices connected to it. So statistical, TDM can make do with a lower transmission rate than needed by synchronous TDM, at the cost of more complexity. The other way in which this is of benefit is, by the ability to support higher throughput for the same available data rate of the multiplexer. This capability is easily realised during the normal, expected data transmission periods when the amount of data that the different input devices have available for transmission is, in fact, lower than the capacity of the multiplexing device. But the same attribute becomes a disadvantage during peak loads, where all or many of the devices may have data to transmit for a short time. In such a situation, the slack that was available to us for more efficient transmission is no longer present. We will see later how to handle peak load situations in statistical TDM.

In this kind of scheme it is not known to us which input device will have data to send at a given time. So we cannot use a round robin positional scheme that was possible in synchronous TDM. Each frame has to be accompanied by information that will tell the demultiplexer which input device the frame belongs to, so that it can be delivered to the appropriate destination. This is the overhead of statistical TDM, besides increased complexity of equipment.

If, we are sending input data from only one source at a time, the structure of a frame would need to have an address field for the source followed by the data for it. This is really the statistical TDM data frame. It would form the data part of a larger, enclosing HDLC frame if we are using HDLC as the transmission protocol. This frame would itself have various other fields that we will not discuss here.

Such a scheme is also not as efficient as we can make it. This is because the quantum of data available from the source, in that time slot, may not be enough to fill the TDM sub frame. So, while it may be an adequate method if the load is not heavy, we also need to think of a method that can utilise available resources better.

The way to do this would be, to have more than one input data source transmit in a single TDM frame. We would then need to have a more complex structure for the frame whereby, we would have to specify the different input devices in the frame followed by the length of the data field. More sophisticated approaches could be used, in order to optimise the number of bits, we need to encode all this addressing and data information.

For peak loads, there is need for some kind of buffering mechanism, so that, whenever there is excess input from the data sources that the multiplexer cannot immediately handle, it is stored until it can be sent. The size of the buffer needed will increase as the transmission capacity of the multiplexer decreases, as it will become more likely that an input data stream will not be transmitted immediately. It will also depend on the average data rate of the input devices taken together. No matter what the buffer size we choose, there is always a non-zero probability that the buffer itself will overflow, leading to loss of data. If, the average data rate of all devices is close to the peak rate, then we are approaching a situation of synchronous TDM where we are not able to take advantage of periods of silence that is the basis of statistical TDM.

4.4 DIGITAL SUBSCRIBER LINES

Stands for "Digital Subscriber Line." DSL is a communications medium used to transfer digital signals over standard telephone lines. Along with cable Internet, DSL is one of the most popular ways ISPs provide broadband Internet access.

When you make a telephone call using a landline, the voice signal is transmitted using low frequencies from 0 Hz to 4 kHz. This range, called the "voice band," only uses a small part of the frequency range supported by copper phone lines. Therefore, DSL makes use of the higher frequencies to transmit digital signals, in the range of 25 kHz to 1.5 MHz. While these frequencies are higher than the highest audible frequency (20 kHz), then can still cause interference during phone conversations. Therefore, DSL filters or splitters are used to make sure the high frequencies do not interfere with phone calls.

Symmetric DSL (SDSL) splits the upstream and downstream frequencies evenly, providing equal speeds for both sending and receiving data. However, since most users download more data than they upload, ISPs typically offer asymmetric DSL (ADSL) service. ADSL provides a wider frequency range for downstream transfers, which offers several times faster downstream speeds. For example, an SDSL connection may provide 2 Mbps upstream and downstream, while an ASDL connection may offer 20 Mbps downstream and 1.5 Mbps upstream.

Like in voice circuits, it would have been possible to use half the channels for communication in each direction. But, statistics show that most users download much more data than they upload. So usually 32 channels are dedicated to uploading, that is, transferring data from the users to the provider and the remaining 216 channels are used for downloading data to the users. This typically translates into 512 Kbps to 1 Mbps download and 64 Kbps to 256 Kbps upload data rates. This then, is the asymmetric aspect of the DSL line.

A problem with ADSL is that the physics of the local loop is such that, the speed at which it can be driven depends heavily on the distance between the subscriber's premises and the provider's nearest termination point. The speed falls sharply with distance and so, distance can become a limiting factor in being able to offer competitive speed compared to that of the cable television providers.

However, ADSL is comparatively simple for a service provider to offer, given an existing telephone network, and does not require much change to its already available equipment. It necessitates two modifications, one each, at the subscriber end and at the end office. On the user's premises, a Network Interface Device has to be installed that incorporates a filter. This is called a splitter and it sends the non-voice portion of the signal to an ADSL modem. The signal from the computer has to be sent to the ADSL modem at high speed, usually done these days by connecting them over a USB port.

At the provider's end office, the signal from the users is recovered and converted into packets that are then sent to the Internet Service Provider, which may be the telephone company itself.

In order to access the Internet using DSL, you must connect to a DSL Internet service provider (ISP). The ISP will provide you with a DSL modem, which you can connect to either a router or a computer. Some DSL modems now have built-in wireless routers, which allows you to connect to your DSL modem via Wi-Fi. A DSL kit may also include a splitter and filters that you can connect to landline phones.

NOTE: Since DSL signals have a limited range, you must live within a specific distance of an ISP in order to be eligible for DSL Internet service. While most urban locations now have access to DSL, it is not available in many rural areas.

4.5 ADSL Vs. CABLE

At first glance, a comparison between ADSL and cable may seem like a no contest. Cable television, sent over coaxial cables, has a bandwidth that is potentially hundreds of times that of the twisted pair Cat-3 cable used for telephone connections. But, as we go along, it turns out that there are considerations in favour of both sides.

First, there are specific assurances regarding bandwidth that we get from telephone companies who provide ADSL connectivity. An ADSL link is a dedicated connection that is always available to the user, unlike television cable that is shared by scores or even hundreds of subscribers in the immediate neighbourhood. So, the kind of speeds that we can get over cable can vary from one moment to the next, depending on the number of users that are working at the time.

As an ADSL system acquires more users, their increasing numbers have little effect on existing users, since each user has a dedicated connection. With cable, as more subscribers sign up for Internet service, performance for existing users will drop. The only cure is for the cable operator to split busy cables and connect each one to a fiber node directly. Doing so costs time and money, so there are business pressures to avoid it.

Availability is an issue on which ADSL and cable differ. Everyone has a telephone, but not all users are close enough to their end offices to get ADSL. On the other hand, not everyone has cable, but if you do have cable and the company provides Internet access, you can get it. Distance to the fiber node or headend is not an issue. It is also worth noting that since cable started out as a television distribution medium, few businesses have it.

Being a point-to-point medium, ADSL is inherently more secure than cable. Any cable user can easily read all the packets going down the cable. For this reason, any decent cable provider will encrypt all traffic in both directions. Nevertheless, having your neighbour get your encrypted messages is still less secure than having him not get anything at all.

There are security risks associated with the fact that cable is shared. Potentially other users can always tap in and read (even change) what you are sending or receiving. The problem does not exist on ADSL, because, each channel is separate and dedicated to the specific user. Though, cable traffic is usually encrypted by the provider, this situation is worse than ADSL where other users just do not get your traffic at all. Also because the channel is dedicated to you, the total number of users does not have any effect on your access speeds as there is no contention with other users.

The telephone system is generally more reliable than cable. For example, it has backup power and continues to work normally even during a power outage. With cable, if the power to any amplifier along the chain fails, all downstream users are cut off instantly.

Finally, most ADSL providers offer a choice of ISPs. Sometimes they are even required to do so by law. Such is not always the case with cable operators. The conclusion is that ADSL and cable are much more alike than they are different. They offer comparable service and, as competition between them heats up, probably comparable prices.

4.6 SWITCHING

Switching is process to forward packets coming in from one port to a port leading towards the destination. When data comes on a port it is called ingress, and when data leaves a port or goes out it is called egress. A communication system may include number of switches and nodes.

In a switched communication network, data entering the network from a station are routed to the destination by being switched from node to node. For example, in Figure 11, data

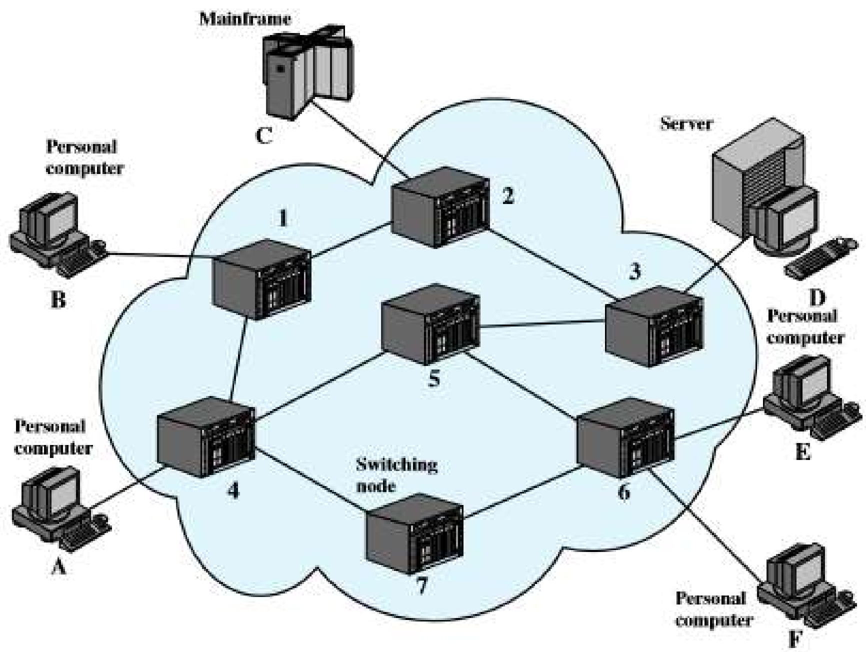

Figure 11 Simple Switching Network

from station A intended for station F are sent to node 4. They may then be routed via nodes 5 and 6 or nodes 7 and 6 to the destination. Several observations are in order:

1. Some nodes connect only to other nodes (e.g., 5 and 7). Their sole task is the internal (to the network) switching of data. Other nodes have one or more stations attached as well; in addition to their switching functions, such nodes accept data from and deliver data to the attached stations.

2. Node-station links are generally dedicated point-to-point links. Node-node links are usually multiplexed, using either frequency division multiplexing (FDM) or time division multiplexing (TDM).

3. Usually, the network is not fully connected; that is, there is not a direct link between every possible pair of nodes. However, it is always desirable to have more than one possible path through the network for each pair of stations. This enhances the reliability of the network.

Two different technologies are used in wide area switched networks: circuit switching and packet switching. These two technologies differ in the way the nodes switch information from one link to another on the way from source to destination. In circuit switching, we create a circuit or link between devices, for the duration of time for which they wish to communicate. This requires a mechanism for, the initiating device to choose the device that it wants to send data to. All devices must also have an identifying, unique address that can be used to set up the circuit. Then, the transmission occurs, and when it is over, the circuit is dismantled so that the same physical resources can be used for another transmission.

The packet switching approach can be used when we are using datagrams or packets to transmit data over the channel. Each datagram is a self-contained unit that includes within itself the needed addressing information. The datagrams can arrive at the destination by any route and may not come in sequence.

Circuit Switching

When two nodes communicate with each other over a dedicated communication path, it is called circuit switching. There 'is a need of pre-specified route from which data will travels and no other data is permitted. In circuit switching, to transfer the data, circuit must be established so that the data transfer can take place.

Circuits can be permanent or temporary. Applications which use circuit switching may have to go through three phases:

- Establish a circuit
- Transfer the data
- Disconnect the circuit

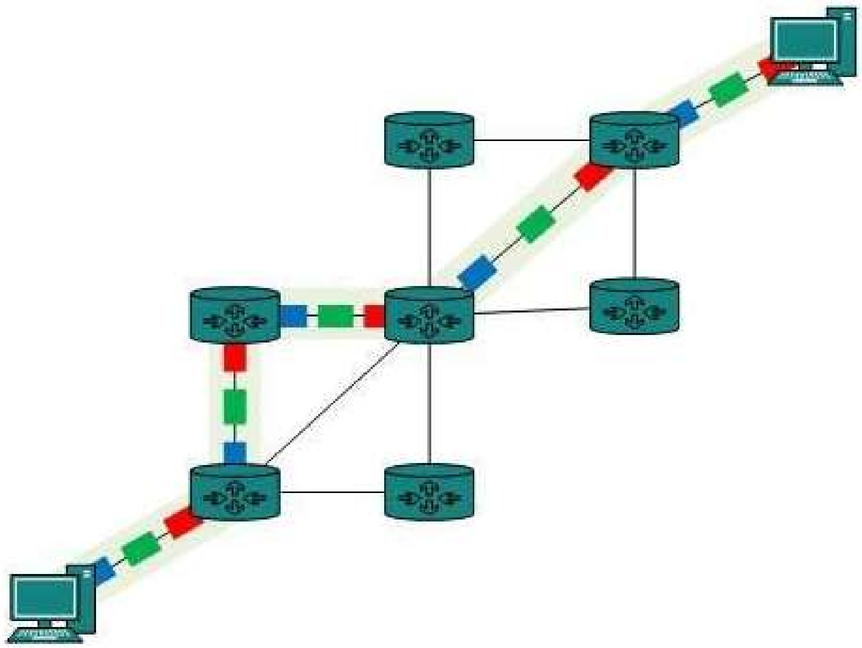

Figure 12: circuit switching

- **Establish a circuit:** Before any signals can be transmitted, an end-to-end (station-to-station) circuit must be established. For example, station A sends a request to node 4 requesting a connection to station E. Typically, the link from(figure 11) A to 4 is a dedicated line, so that part of the connection already exists. Node 4 must find the next leg in a route leading to E. Based on routing information and measures of availability and perhaps cost, node 4 selects the link to node 5, allocates a free channel (using FDM or TDM) on that link, and sends a message requesting connection to E. So far, a dedicated path has been established from A through 4 to 5. Because a number of stations may attach to 4, it must be able to establish internal paths from multiple stations to multiple nodes. How this is done is discussed later in this section. The remainder of the process proceeds similarly. Node 5 allocates a channel to node 6 and internally ties that channel to the channel from node 4. Node 6 completes the connection to E. In completing the connection, a test is made to determine if E is busy or is prepared to accept the connection.

- **Transfer the data:** Data can now be transmitted from A through the network to E. The transmission may be analog or digital, depending on the nature of the network. As the carriers evolve to fully integrated digital networks, the use of digital (binary) transmission for both voice and data is becoming the dominant method. The path is A-4 link, internal switching through 4, 4-5 channel, internal switching through 5, 5-6 channel, Internal switching through 6, 6-E link. Generally, the connection is full duplex.

- **Disconnect the circuit:** After some period of data transfer, the connection is terminated, usually by the action of one of the two stations. Signals must be propagated to nodes 4, 5, and 6 to deallocate the dedicated resources.

- Note that the connection path is established before data transmission begins. Thus, channel capacity must be reserved between each pair of nodes in the path, and each node must have available internal switching capacity to handle the requested connection. The switches must have the intelligence to make these allocations and to devise a route through the network.

  Circuit switching can be rather inefficient. Channel capacity is dedicated for the duration of a connection, even if no data are being transferred. For a voice connection, utilization may be rather high, but it still does not approach 100%. For a client/server or terminal-to-computer connection, the capacity may be idle during most of the time of the connection. In terms of performance, there is a delay prior to signal transfer for call establishment. However, once the circuit is established, the network is effectively transparent to the users. Information is transmitted at a fixed data rate with no delay other than the propagation delay through the transmission links. The delay at each node is negligible.

Let us understand an example of circuit switching by given this **figure 13** where there are 7 devices. These are divided into two groups of 3 on the left and 4 on the right. To ensure complete connectivity between them at all times, we would need 12 physical links. But, if connectivity is not required at all times, we can achieve connectivity between any of the devices by grouping them together and using switches to achieve temporary links.

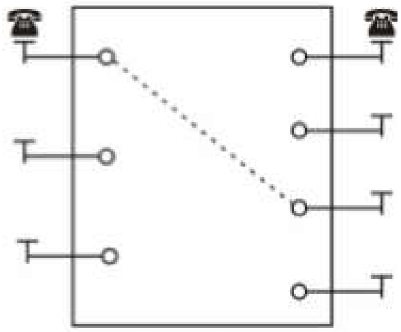

Figure 13: Circuit switching

For example, we can have 7 links that connect the devices A, B and C on the left to the switch. The other devices D, E, F and G on the right are also connected to the same switch. The switch can connect any two devices together using only these seven links as desired.

The capacity of the switch is determined by the number of circuits that it can support at any given time. In the above example, we have seen a switch with 1 input and 1 output. If devices C and E are communicating with each other, the others cannot communicate at the same time, although, there are available links from them to the switch. A circuit switch is really a device that has n inputs and m outputs (n need not be equal to m).

There are two main approaches to circuit switching, called space division or time division switches. Space division switches are so called because the possible circuit paths are separated from one another in space. The old, and now obsolete, crossbar telephone exchanges are an example of space division switching. The technique can be used for both digital or analog systems. There were other designs of such switching but the only one that went into large scale use was the crossbar. Because of the way it is constructed, such switching does not have any delays. Once the path is established, transmission occurs continuously at the rate that the channel can support.

In essence, a crossbar connects p inputs to q outputs. Each such connection is actually performed by connecting the input to the output using some switching technology such as, a transistor based electronic switch or an electromechanical relay. This requires a large number of possible connections, called cross points. For a 10,000 line telephone exchange, it would mean that, each of the 10,000 possible inputs be able to connect to any of the 10,000 possible outputs, requiring a total of 100,000,000 crosspoints. As this is clearly impractical, the pure crossbar design is not usable on a commercial scale. Moreover, there are inherent inefficiencies in this design as statistically, only a small fraction of these crosspoints are ever in use simultaneously.

The way to get around this limitation is, to split the switch into different stages. If we consider a 36 line exchange where each of 36 inputs needs to be connected to 36 outputs, we can do so in, say, 3 stages. The first stage could have 3 switches, each with 12 inputs and 2 outputs to the two second stage switches. These intermediate switches could each have 3 inputs from the 3 first stage switches and 3 outputs to the 3 third stage switches. The last stage of the switches would then have 2 inputs from the 2 second stage switches and 12 outputs, each to 12 of the 36 devices. It is thus, possible for each of the 36 inputs to connect to each of the 36 outputs as required, using only 72 + 18 + 72 = 162 crosspoints, instead of the 1296 crosspoints that would have been required without the multistage design. The following Figure 14 shows the multistage switch.

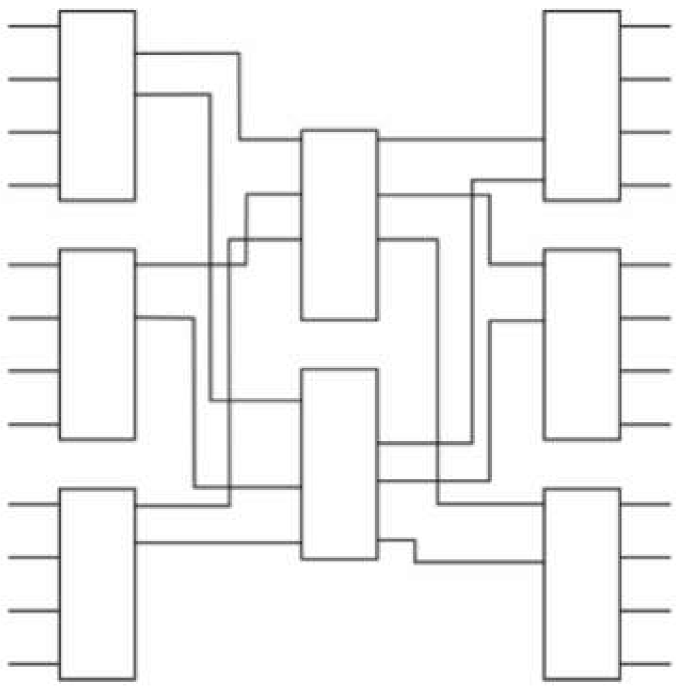

Figure 14: Multistage switch

If, we use a single stage switch, we will always be able to connect each device to any other device that is not already busy. This is because all the paths are independent and do not overlap. What this means is that, we will never be starved of circuits and will never suffer the problem of blocking. For multistage switching, the reduction in the number of cross points required comes at the cost of possible blocking. This happens when there is no available path from a device, to another free device, that we desire to connect to. In the Figure 14, if we have a switch in the first stage that is already serving 2 input devices, then there are no free outputs from that stage. We cannot therefore service more than 2 input devices connected to one switch at a time and the 3rd device would not be able to connect anywhere, getting a busy signal.

It is possible to minimise the possibility of blocking by using stochastic analysis to find the least number of independent paths that we should provide for. However, we will not go into such an analysis in this unit. But, we can see that the limiting factor in the Figure 14 is the middle stage where there are only 3 inputs and 2 outputs. This is the point at which congestion is most likely to occur. You too may have experienced blocking when trying to make a telephone call and found that you got an exchange busy tone, indicating that it was not the number you called but the exchange (switch) itself that was congested. Such a problem is, most likely to occur during periods of heavy activity such as at midnight on a New Year's Day, when many people might be trying to call one another to exchange greetings.

Another advantage of multistage switching is that, there are many paths that can be used to connect the input and output devices. So, in the above case, if there is a failure in one of the connections in the first stage switch, two input devices can still connect to it. In the case of single stage switching, that would have meant that we could not set up a circuit between those devices.

Let us, now look at another method of switching that uses time slots rather than spatial separation. You have already seen how synchronous TDM involves transmission between input and output devices using fixed time slots dedicated to each channel. But, that is not switching because the input-output device combinations are fixed. So, if we have three devices A, B and C transmitting and three devices D, E and F that are receiving the respective transmissions, there will be no way to change the circuit path so that A can transmit to E or F.

To achieve switching, we use a device called a Time Slot Interchange (TSI). The principle of such a device is simple. Based on the desired paths that we want to set up, the TSI changes the input ordering in the data streams. So, if the demultiplexer is sending the outputs to D, E and F in order, and if we want that A send data to E instead of F, and that B send data to F rather than E, then the TSI will change the input ordering of the time slots from A, B, C to C, A, B. The demultiplexer will not be aware of this and will continue to send the output the same way as before. The result will be, that our desired switching will be accomplished. However, unlike space division switching, our output will be subject to delays because we might have to wait for the input from the right device before it can be transmitted. This is unlike space division switching where the data can be sent in a steady stream.

How does the TSI work? It consists of a control unit that does the actual reordering of the input. For this, it has to first, buffer the input it gets, in the order it gets it. This would be stored in some kind of volatile memory. The control unit then sends out the data from the buffer in the order in which it is desired. Usually the size of each buffer would be that of the data that the input generates in one time slice.

We do not have to confine ourselves to a single type of switch. There can be switches that are based on a combination of both kinds of switching. For example, we could have a multistage switch where, some of the stages are space division switches whiles others are time division switches. With such an approach, we can try to optimise the design by reducing the need for crosspoints while keeping the delays in the whole system to a minimum. For example, we could have a TST three stage switch where the first and last stages use time division switching while the middle stage uses crosspoints.

Circuit switching is useful for voice based communication because of the characteristics of the data transfer. Although, a voice conversation tends to have periods of silence in between, those periods are usually brief. The rest of the time there is data available for transmission. Secondly, in such communication, we cannot tolerate delays of more than about 100 ms as that becomes perceptible to the human ear and is quite annoying to the speaker and listener.

Again, because it is human beings that are present at both ends, the rate at which data is generated at both ends is similar, even if one person talks a bit faster than the other! Also, the other human can usually understand what is being said even if he cannot at the same rate. And if really required, one can communicate to the other to speak slower or louder. But, when we are dealing with data generating devices, there can always be a mismatch between the rate at which one device generates data and at which the other device can assimilate it. Moreover, there can be long periods when there is no data generated at all for transmission. In such a situation, circuit switching will not be a suitable method and we have to look at something that takes care of the characteristics of data communication between devices.

Message Switching

This technique was somewhere in middle of circuit switching and packet switching. In message switching, the whole message is treated as a data unit and is switching / transferred in its entirety.

A switch working on message switching, first receives the whole message and buffers it until there are resources available to transfer it to the next hop. If the next hop is not having enough resource to accommodate large size message, the message is stored and switch waits.

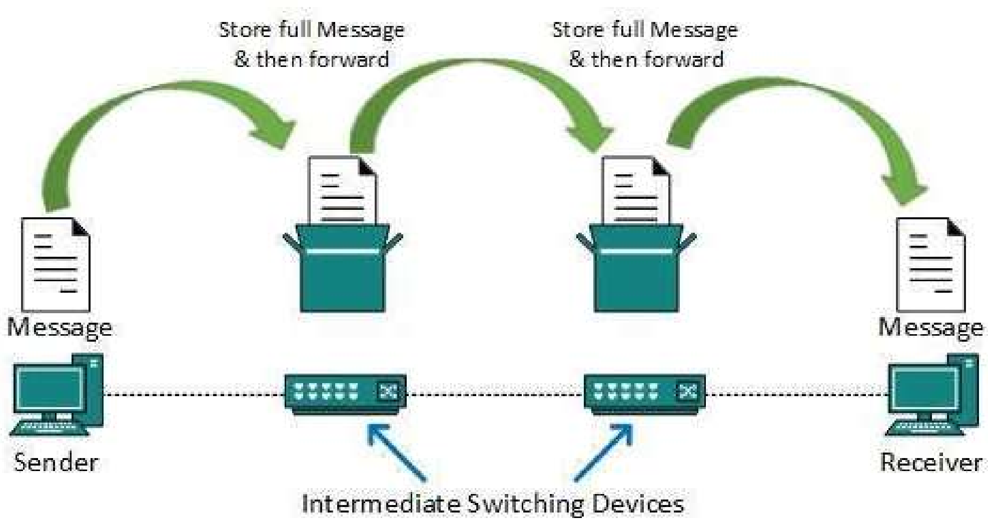

Figure 15: Message Switching

This technique was considered substitute to circuit switching. As in circuit switching the whole path is blocked for two entities only. Message switching is replaced by packet switching. Message switching has the following drawbacks:

- Every switch in transit path needs enough storage to accommodate entire message.
- Because of store-and-forward technique and waits included until resources are available, message switching is very slow.
- Message switching was not a solution for streaming media and real-time applications.

Packet Switching

Shortcomings of message switching gave birth to an idea of packet switching. The entire message is broken down into smaller chunks called packets. The switching information is added in the header of each packet and transmitted independently.

It is easier for intermediate networking devices to store small size packets and they do not take much resources either on carrier path or in the internal memory of switches.

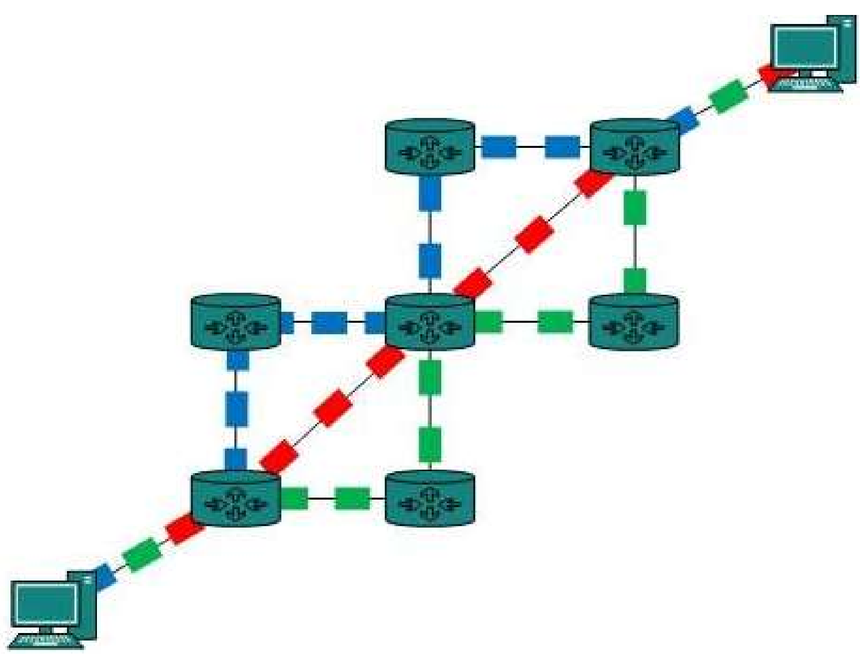

Figure 16: Packet switching

The long-haul circuit-switching telecommunications network was originally designed to handle voice traffic, and the majority of traffic on these networks continues to be voice. A key characteristic of circuit-switching networks is that resources within the network are dedicated to a particular call .For voice connections, the resulting circuit will enjoy a high percentage of utilization because, most of the time, one party or the other is talking. However, as the circuit-switching network began to be used increasingly for data connections, two shortcomings became apparent:

• In a typical user/host data connection (e.g., personal computer user logged on to a database server), much of the time the line is idle. Thus, with data connections, a circuit-switching approach is inefficient.

• In a circuit-switching network, the connection provides for transmission at a constant data rate. Thus, each of the two devices that are connected must transmit and receive at the same data rate as the other. This limits the utility of the network in interconnecting a variety of host computers and workstations.

To understand how packet switching addresses these problems, let us briefly summarize packet-switching operation. Data are transmitted in short packets. A typical upper bound on packet length is 1000 octets (bytes). If a source has a longer message to send, the message is broken up into a series of packets (Figure 16). Each packet contains a portion (or all for a short message) of the user's data plus some control information. The control information, at a minimum, includes the information that the network requires to be able to route the packet through the network and deliver it to the intended destination. At each node en-route, the packet is received, stored briefly, and passed on to the next node.

Now we can understand packet switching from figure 11. It depicts a simple packet switching network. Consider a packet to be sent from station A to station E. The packet includes control information that indicates that the intended destination is E. The packet is sent from A to node 4. Node 4 stores the packet, determines the next leg of the route (say 5),and queues the packet to go out on that link (the 4–5 link).When the link is available, the packet is transmitted to node 5,which forwards the packet to node 6,and finally to E. This approach has a number of advantages over circuit switching:

• Line efficiency is greater, because a single node-to-node link can be dynamically shared by many packets over time. The packets are queued up and transmitted as rapidly as possible over the link. By contrast, with circuit switching, time on a node-to-node link is pre-allocated using synchronous time division multiplexing. Much of the time, such a link may be idle because a portion of its time is dedicated to a connection that is idle.

• A packet-switching network can perform data-rate conversion. Two stations of different data rates can exchange packets because each connects to its node at its proper data rate.

• When traffic becomes heavy on a circuit-switching network, some calls are blocked; that is, the network refuses to accept additional connection requests until the load on the network decreases. On a packet-switching network, packets are still accepted, but delivery delay increases.

• Priorities can be used. If a node has a number of packets queued for transmission, it can transmit the higher-priority packets first. These packets will therefore experience less delay than lower-priority packets.

Switching Technique

If a station has a message to send through a packet-switching network that is of length greater than the maximum packet size, it breaks the message up into packets and sends these packets, one at a time, to the network. A question arises as to how the network will handle this stream of packets as it attempts to route them through the network and deliver them to the intended destination. Two approaches are used in contemporary networks: datagram and virtual circuit.

In the datagram approach, each packet is treated independently, with no reference to packets that have gone before. This approach is illustrated in Figure17, which shows a time sequence of snapshots of the progress of three packets through the network. Each node chooses the next node on a packet's path, taking into account information received from neighbouring nodes on traffic, line failures, and so on. So the packets, each with the same destination address, do not all follow the same route, and they may arrive out of sequence at the exit point. In this example, the exit node restores the packets to their original order before delivering them to the destination. In some datagram networks, it is up to the destination rather than the exit node to do the reordering. Also, it is possible for a packet to be destroyed in the network. For example, if a packet-switching node crashes momentarily, all of its queued packets may be lost. Again, it is up to either the exit node or the destination to detect the loss of a packet and decide how to recover it. In this technique, each packet, treated independently, is referred to as a datagram. In the virtual circuit approach, a pre-planned route is established before any packets are sent. Once the route is established, all the packets between a pair of communicating parties follow this same route through the network. This is illustrated in Figure17. Because the route is fixed for the duration of the logical connection, It is somewhat similar to a circuit in a circuit-switching network and is referred to as a datagram.

In the virtual circuit approach, a pre-planned route is established before any packets are sent. Once the route is established, all the packets between a pair of communicating parties follow this same route through the network. This is illustrated in Figure18.Because the route is fixed for the duration of the logical connection; it is somewhat similar to a circuit in a circuit-switching network and is referred to as a virtual circuit. Each packet contains a virtual circuit identifier as well as data. Each node on the pre-established route knows where to direct such packets; no routing decisions are required .At any time, each station can have more than one virtual circuit to any other station and can have virtual circuits to more than one station.

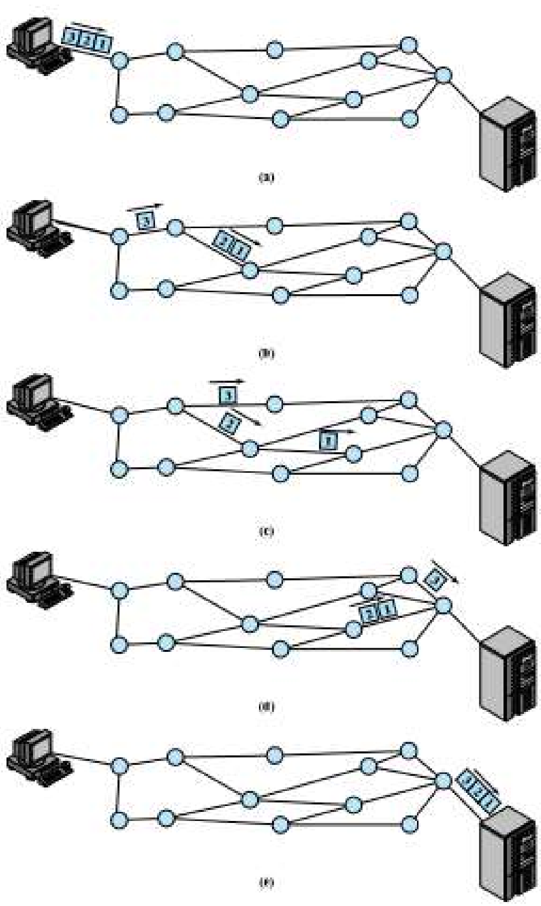

Figure 17 Packet Switching Datagram Approach

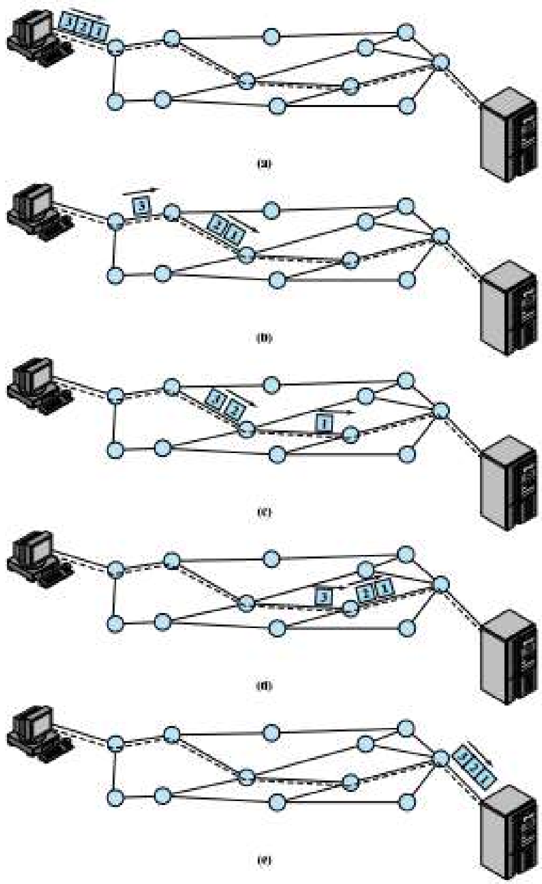

Figure 18: Packet Switching Virtual-Circuit Approach

So the main characteristic of the virtual circuit technique is that a route between stations is set up prior to data transfer. Note that this does not mean that this is a dedicated path, as in circuit switching. A transmitted packet is buffered at each node, and queued for output over a line, while other packets on other virtual circuits may share the use of the line. The difference from the datagram approach is that, with virtual circuits, the node need not make a routing decision for each packet. It is made only once for all packets using that virtual circuit.

If two stations wish to exchange data over an extended period of time, there are certain advantages to virtual circuits. First, the network may provide services related to the virtual circuit, including sequencing and error control. Sequencing refers to the fact that, because all packets follow the same route, they arrive in the original order. Error control is a service that assures not only that packets arrive in proper sequence, but also that all packets arrive correctly. For example, if a packet in a sequence from node 4 to node 6 fails to arrive at node 6, or arrives with an error, node 6 can request a retransmission of that packet from node 4. Another advantage is that packets should transit the network more rapidly with a virtual circuit; it is not necessary to make a routing decision for each packet at each node.

One advantage of the datagram approach is that the call setup phase is avoided. Thus, if a station wishes to send only one or a few packets, datagram delivery will be quicker. Another advantage of the datagram service is that, because it is more primitive, it is more flexible. For example, if congestion develops in one part of the network, incoming datagrams can be routed away from the congestion. With the use of virtual circuits, packets follow a predefined route, and thus it is more difficult for the network to adapt to congestion. A third advantage is that datagram delivery is inherently more reliable. With the use of virtual circuits, if a node fails, all virtual circuits that pass through that node are lost. With datagram delivery, if a node fails, subsequent packets may find an alternate route that bypasses that node. A datagram-style of operation is common in internetworks.

Packet Size

There is a significant relationship between packet size and transmission time, as shown in Figure19. In this example, it is assumed that there is a virtual circuit from station X through nodes A and B to station Y. The message to be sent comprises 40 octets, and each packet contains 3 octets of control information, which is placed at the beginning of each packet and is referred to as a header. If the entire message is sent as a single packet of 43 octets (3 octets of header plus 40 octets of data),then the packet is first transmitted from station X to node a (Figure 19a).When the entire packet is received, it can then be transmitted from a to b. When the entire packet is received at node b, it is then transferred to station Y. Ignoring switching time, total transmission time is 129 octet-times (transmissions).

Suppose now that we break the message up into two packets, each containing 20octets of the message and, of course,3 octets each of header, or control information.

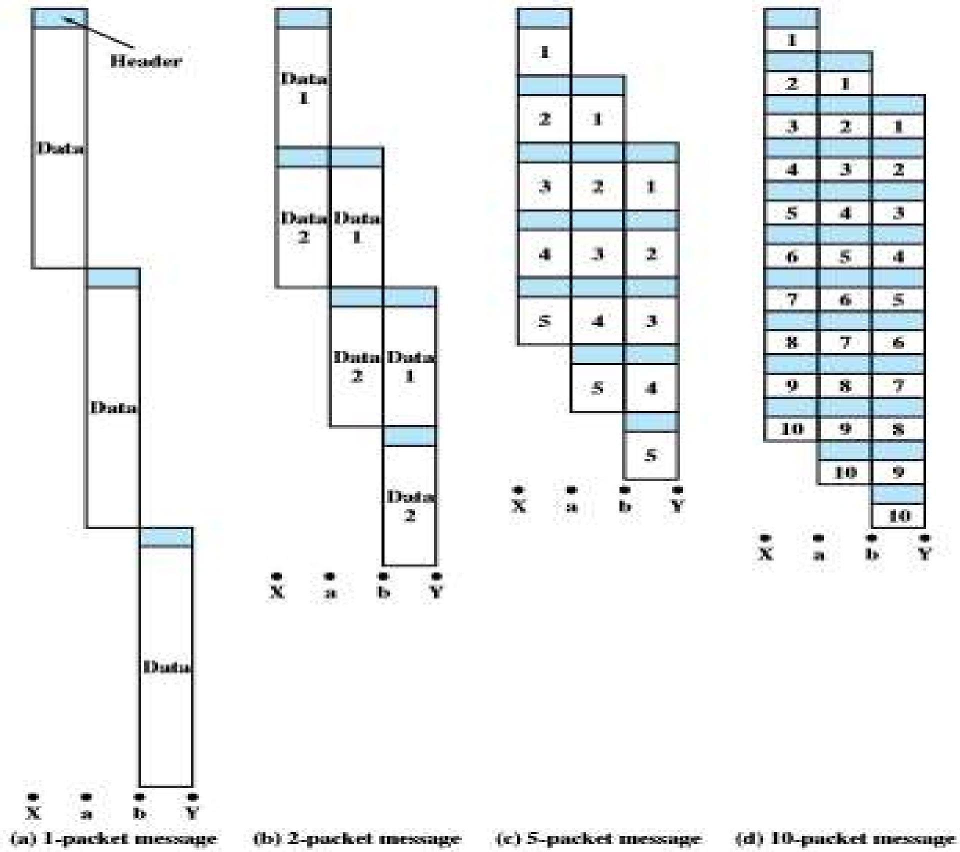

Figure 19: Effect of Packet Size on Transmission Time

In this case, node a can begin transmitting the first packet as soon as it has arrived from X, without waiting for the second packet. Because of this overlap in transmission, the total transmission time drops to 92 octet-times. By breaking the message up into five packets, each intermediate node can begin transmission even sooner and the savings in time is greater, with a total of 77 octet-times for transmission. However, this process of using more and smaller packets eventually results in increased, rather than reduced, delay as illustrated in Figure 19d. This is because each packet contains a fixed amount of header, and more packets mean more of these headers. Furthermore, the example does not show the processing and queuing delays at each node. These delays are also greater when more packets are handled for a single message. However, we shall see in the next chapter that an extremely small packet size (53 octets) can result in an efficient network design.

Comparison of Circuit Switching and Packet Switching

Having looked at the internal operation of packet switching, we can now return to a comparison of this technique with circuit switching. We first look at the important issue of performance and then examine other characteristics.

Performance

A simple comparison of circuit switching and the two forms of packet switching is provided in Figure20.The figure depicts the transmission of a message across four nodes, from a source station attached to node 1 to a destination station attached to node 4. In this figure, we are concerned with three types of delay:

• **Propagation delay:** The time it takes a signal to propagate from one node to the next. This time is generally negligible. The speed of electromagnetic signals through a wire medium, for example, is typically 2 X 10^8 m/s

• **Transmission time:** The time it takes for a transmitter to send out a block of data. For example, it takes 1 s to transmit a 10,000-bit block of data onto a 10-kbps line.

• **Node delay:** The time it takes for a node to perform the necessary processing as it switches data.

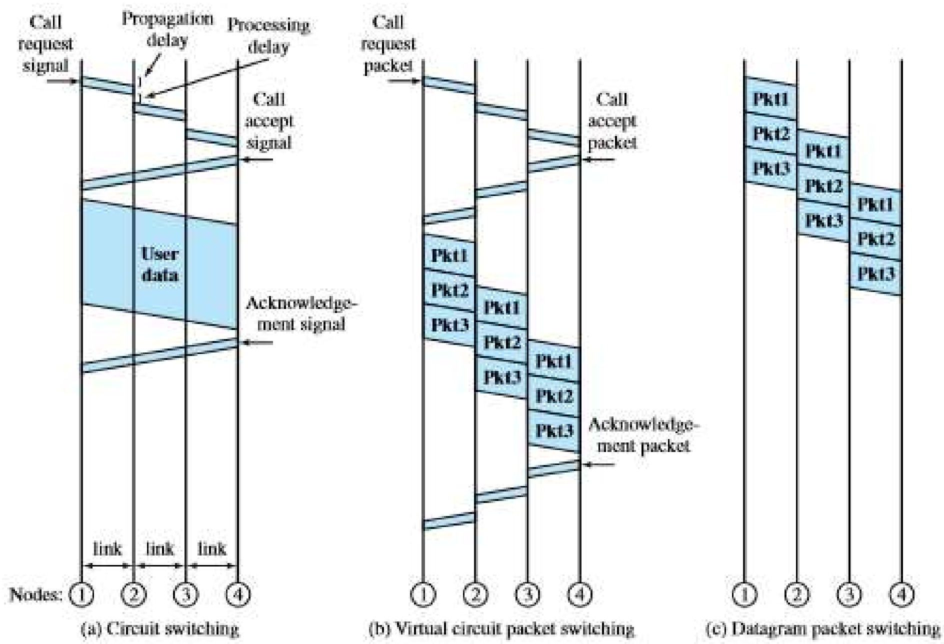

Figure 20: Event Timing for Circuit Switching and Packet Switching

For circuit switching, there is a certain amount of delay before the message can be sent. First, a Call Request signal is sent through the network, to set up a connection to the destination. If the destination station is not busy, a Call Accepted signal returns. Note that a processing delay is incurred at each node during the call request; this time is spent at each node setting up the route of the connection. On the return, this processing is not needed because the connection is already set up. After the connection is set up, the message is sent as a single block, with no noticeable delay at the switching nodes.

Virtual circuit packet switching appears quite similar to circuit switching. A virtual circuit is requested using a Call Request packet, which incurs a delay at each node. The virtual circuit is accepted with a Call Accept packet. In contrast to the circuit-switching case, the call acceptance also experiences node delays, even though the virtual circuit route is now established. The reason is that this packet is queued at each node and must wait its turn for transmission. Once the virtual circuit is established, the message is transmitted in packets. It should be clear that this phase of the operation can be no faster than circuit switching, for comparable networks. This is because circuit switching is an essentially transparent process, providing a constant data rate across the network. Packet switching involves some delay at each node in the path. Worse, this delay is variable and will increase with increased load.

Datagram packet switching does not require a call setup. Thus, for short messages, it will be faster than virtual circuit packet switching and perhaps circuit switching. However, because each individual datagram is routed independently, the processing for each datagram at each node may be longer than for virtual circuit packets. Thus, for long messages, the virtual circuit technique may be superior.

Figure 20 is intended only to suggest what the relative performance of the techniques might be; actual performance depends on a host of factors, including the size of the network, its topology, the pattern of load, and the characteristics of typical exchanges.

Besides performance, there are a number of other characteristics that may be considered in comparing the techniques we have been discussing. Table 1 summarizes the most important of these. Most of these characteristics have already been discussed. A few additional comments follow.

As was mentioned, circuit switching is essentially a transparent service. Once a connection is established, a constant data rate is provided to the connected stations. This is not the case with packet switching, which typically introduces variable delay, so that data arrive in a choppy manner. Indeed, with datagram packet switching, data may arrive in a different order than they were transmitted.

An additional consequence of transparency is that there is no overhead required to accommodate circuit switching. Once a connection is established, the analog or digital data are passed through, as is, from source to destination. For packet switching, analog data must be converted to digital before transmission; in addition, each packet includes overhead bits, such as the destination address.

Table 1: Comparison of Communication Switching Techniques

Circuit Switching	**Datagram Packet Switching**	**Virtual Circuit Packet Switching**
Dedicated transmission path	No dedicated path	No dedicated path
Continuous transmission of	Transmission of packets	Transmission of packets data
Fast enough for interactive	Fast enough for interactive	Fast enough for interactive

Messages are not stored	Packets may be stored until delivered	Packets stored until delivered
The path is established for entire conversation	Route established for each packet	Route established for entire conversation
Call setup delay; negligible transmission delay	Packet transmission delay	Call setup delay ;packet transmission delay
Busy signal if called party busy	Sender may be notified if packet not delivered	Sender may be notified of connection denial
Overload may block call setup; no delay for established calls	Overload increases packet delay	Overload may block call setup; increases packet delay
Overload may block call setup; no delay for established calls	Overload increases packet delay	Overload may block call setup; increases packet delay
Electromechanical or computerized switching nodes	Small switching nodes	Small switching nodes
User responsible for message loss protection	Network may be responsible for individual packets	Network may be responsible for packet sequences
Usually no speed or code conversation	Speed and code conversion	Speed and code conversion
Fixed bandwidth	Dynamic use of bandwidth	Dynamic use of bandwidth
No overhead bits after call setup	Overhead bits in each packet	Overhead bits in each packet

4.7 SUMMARY

Multiplexing is needed in communication networks because of the scarcity of bandwidth compared to the large number of users. Frequency Division and Time Division Multiplexing are the two ways of multiplexing, of which Time Division multiplexing can be synchronous or asynchronous. The asynchronous method is more complex but more efficient for data transmission. In frequency division multiplexing, signals are generated by sending different device-modulated carrier frequencies, and these modulated signals are then combined into a single signal that can be transported by the link. Wavelength division multiplexing (WDM) is a technology in fiber optic communications; and, for the high capacity communication systems, wavelength division multiplexing is the most promising concept. Time division multiplexing is a technique used to transmit a signal over a single communication channel by dividing the time frame into slots – one slot for each message signal.

ADSL is a means of utilising the existing capacity of the local loop in the telephone system for providing subscribers with high speed data access. Such access is also provided by companies over the cable television network. ADSL is more secure and predictable in terms of service quality, while cable does not have the limitations of distance from the end office that ADSL has.

Switching is necessary to connect two nodes or devices over the network that intend to communicate for a limited duration. Switching is process to forward packets coming in from one port to a port leading towards the destination. Circuit switching is more suitable for voice communication, and can be done using space division, time division or a combination of both kinds of switches. Multistage switching is needed to optimise the number of cross points needed. In message switching, the whole message is treated as a data unit and is switching / transferred in its entirety. Packet switching is used for data transmission and allows for prioritising of data packets, alternate routing as needed and is also more efficient for the bursty traffic pattern of data communication. Datagrams are self-contained packets of data that are routed by the intermediate nodes of the network. Switched or permanent virtual circuits can also be utilised, where the route is established at the beginning of the session but can be altered without disrupting the channel in case of failure of any part of the route.

4.8 Questions for exercise

1) What is the problem in offering high speed ADSL connection to all subscribers that have a telephone?

2) Which method is more efficient in terms of capacity utilisation?

3) Do you think voice transmission could be done using packet switching?

4) What sort of problems could arise in trying to completely connect all data transmission devices in the world?

5) What are some of the requirements for a multiplexed signal to be useful?

6) Is multiplexing needed for transmissions over different links?

7) What are the problems that can occur in Frequency Division Multiplexing?

8) What is the method that FDM uses to mix signals that can be recovered later?

9) List the considerations that make FDM useful and possible.

10) What are the considerations in choosing the length of the time slice for Time Division Multiplexing?

11) Why must the multiplexer and demultiplexer be synchronised? Why then is synchronous TDM so called?

12) What are the problems in synchronising the transmission and how can they be taken care of?

13) What are the inefficiencies inherent in synchronous Time Division Multiplexing and how does statistical TDM seek to reduce them?

14) What price do we have to pay for increased efficiency in statistical TDM?

15) What would happen if the load from the input devices exceeds the capacity of the multiplexer?

16) How does ADSL enable high speed data access although voice lines are so slow?

17) Why is ADSL called asymmetric and why is it not kept symmetric?

18) How can a cable provider improve quality of service after the number of users becomes larger?

19) Why is ADSL more secure than cable for data communication?

20) Why is switching necessary?

21) What is circuit switching?

22) What is packet switching? How does it differ from circuit switching?

23) What is meant by multistage switching? Is the capacity of such a switch limited?

24) List five important features of space division switching.

25) What are the characteristics of time division switching?

26) How can we combine space and time division switching? What are the advantages of such an approach?

27) Why is circuit switching suitable for voice transmission?

28) What are the features needed for a switching mechanism that transmits data?

29) What is a virtual circuit? How does it differ from circuit switching?

30) What are datagrams and how can they be used to transmit data?

4.19 Suggested Reading:

1) Computer Networks, 5^{th} Edition, A.S. Tanenbaum, Prentice Hall of India, New Delhi.

2) Communication Networks fundamental concepts and key architecture, Leon Garcia and Indra Widjaja, Tata McGraw Hill, New Delhi.

3) Data Communication and Networking, 4^{th} edition, Behrouz A. Forouzan, Tata McGraw Hill, New Delhi.

4) Data.And.Computer.Communications.8e.WilliamStallings 8th edition.

References

1) https://www.edgefx.in/what-are-the-different-types-of-multiplexing-in-communication-system/