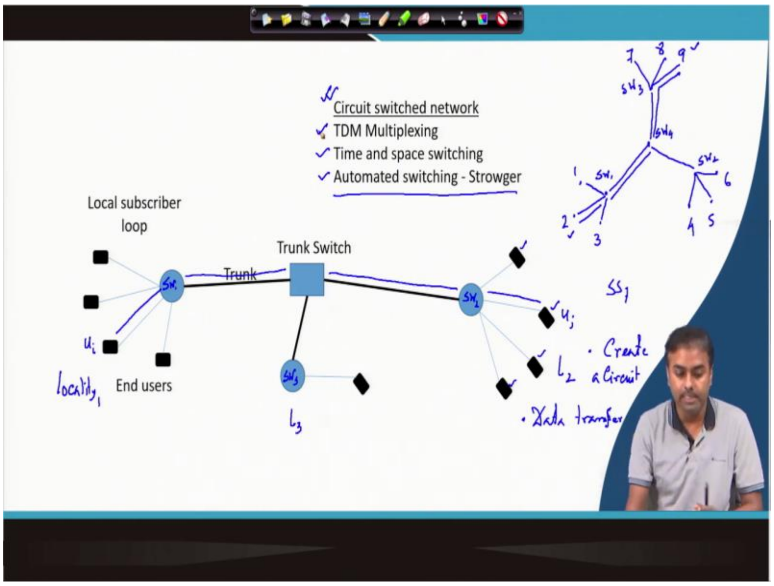
Communication Networks
Prof. Goutam Das
G. S. Sanyal School of Telecommunication
Indian Institute of Technology, Kharagpur

Lecture - 02
Introduction to Communication Networks cont'd

So, continuing our discussion of networking. So, what we have; a quick recapitulation.

(Refer to Slide Time: 00:36)



So, in the last class, we understood what communication is and what networking is. So, communication is a point-to-point thing, where you can successfully transfer information through a particular media; let us say wireless or wired media, so whatever it is, it might be fiber optic, it might be a coaxial cable.

So, we understood that part, and then we started discussing how we could do networking. So, multiple users simultaneously try to use that from this point-to-point link; how do we construct a network among those users. So, that is something we have started discussing.

And then, we have started discussing, in networking, what we can do to actually make it cost-effective and make the infrastructure more utilized so that we can scale the infrastructure as the number of users and the requests increases. So, this is something we have seen. So, from there, what we have started discussing is the first genesis of a network, probably that is called the means voice or telephony network.

So, today we will be concentrating on this particular telephony network, how people have started deploying them, and what is the salient feature of those kinds of networks. So, this is something we will be starting to discuss. So, if you now see, so, this is a typical telephony network.

So, in a telephony network, as we have discussed, networking has to be done like this. So, suppose I have a few nodes in a locality that wish to communicate. So, let us say this 1, 2, and 3 are the nodes that want to communicate with each other. So, basically, as we have discussed, so we’ll be putting a switch over here and will be giving connectivity from each node to the switch that is very good for doing local networking.

So, then if we have also talked about the Strowger automated switching so that switching can be made automated; so that if user-1 wishes to talk to 3, then the connectivity will be given accordingly through the switch; if user 2 wants to talk to 3, so accordingly the connectivity will be given.

So, that can be done locally. Now, what about suppose in another geographical location some other user, let us say users-4,5, and 6 are there. So, again how do we connect? We probably will not be connecting all these 4, 5, and 6; because they are distant users. So, if we again connect them to that same switch, that switch-1, let us say we have talked about. So, if we connect them to that switch 1, then a huge amount of infrastructure again has to be put in.

So, what will we be doing? We will be actually putting another switch close to in close proximity to these users, so and will connect them. So, the users 5, 4, 5, and 6 can talk to each other; but we have to also (because it is networking) give a facility so that 1, 2, 3, or any one of them can also talk to 4, 5, 6, ok.

So, there might be another locality also, where some more users are connected. So, let us say 7, 8, and 9, ok. So, this is actually switch-2. This is probably switch-3. Now, we have to give connectivity among these switches also, so we go hierarchically. So, basically, we will put another switch, which is more of a core switch probably, and then give connectivity to all these switches; this is something we also discussed last time.

So, once we do that, this is kind of, let us say, core switch. So, this is a bigger switch, actually. So, let us call that switch-4, ok. So, what will happen; if user-2 wishes to communicate with user-9? What will he do? He will first go to this switch and, from this switch, will be given connectivity to this one.

So, this and that will be routed through this, so this switch will make this connectivity. So, it will go to switch-3, and from switch-3, the connectivity will be given towards 9. So, this is how we'll be establishing the connection. So, that is what I have demonstrated over here. So, this is locality-1, this is probably locality-2, and this is locality-3. So, they have their switch. So, this is switch-1; let us say this is switch-3, and this is switch-2.

And then the users in each locality will be actually; this is actually called a local subscriber loop in telephony this one. So, all the end users will be connected to the local subscriber loop, so this is that switch.

And from there, the terminology was that from this switch to a bigger switch, they used to call that a trunk; that is why these calls used to be called trunk calls, whenever you are going means outside your locality, probably, that used to be called STD or trunk call, actually.

So, this trunk, so basically, you have a trunk link and then followed by a trunk switch, which connects all the other local subscriber loops, and then you used to get connectivity. So, whenever you dialed a particular number, so, there were unique numbers for everybody, each of this. So, whenever you dial a number, what used to happen? So, whatever number you dial, from there, they used to know where the user is.

Accordingly, the network used to construct a switching configuration through the entire network. So, it goes to that, then this switch from that number you have dialed, he will be knowing that I have to go to the trunk switch, it is not in this locality.

So, the trunk switch will know where that user is, so that might be in switch 2, so he used to give this connectivity. And then switch-2 will be knowing that the user the incoming call is intended to connect to is in my locality, so he used to make that switching.

So, this is how, whenever you dial a number, this switch configuration used to be made; so that means that is why it used to be called a circuit switch network. So, what does that mean? Actually, we construct a circuit even before giving connectivity.

So, even before you start talking, you first, with that number, construct connectivity through the entire network, or you reserve a circuit. So, a circuit for yourself you actually construct. So, suppose this user, let us call it user-i, wants to communicate with user-j. So, what do you do? You reserve a circuit; which way will be reserving circuit, I will talk about that later on.

So, you reserve this circuit end to end, and you actually mean you first construct that circuit through the switches. So, all switches are automated. So, they actually understand that numbering system. So, there is a protocol called SS 7 ok; through that, they will be understanding this particular thing, and they will be configuring themselves.

So, once it is configured, so this particular link will be completely dedicated to you, ok. So, or I should say user-i to user-j, this will be dedicated to that and whatever associated bandwidth they will have; that might be a 64 kbps bandwidth for a voice channel, so that will be completely dedicated to them, whether they are using it.

So, suppose once you create the circuit, then the data transmission will be going on. Now, in between, if you are not saying anything or your other counterpart is also not saying anything, that means no information is being carried, but the circuit is still for you as long as you are connected, ok.

So, that was the terminology they used to use for this. So, you first create and construct a circuit through the network, then you reserve that circuit; that circuit is for you only exclusively for the entire duration of your call, and then you start transmitting the data through that circuit, ok.

So, whenever you start transmitting the data, now the circuit is constructed, all the switch configurations are done; whatever data you will be throwing, it knows that it has to go through this path, go through this path, to the end user. So, it knows already. So, you, and that is why it used to be called a circuit switch network.

So, it is a constructed circuit over which you transfer your data. So, as you can see over here, two things are coming. So, first, you have to construct a circuit. So, basically, create a circuit, create a circuit, and then. So, this is the control part, basically with those dialing numbers you used to do that.

So, that is the control part; you create a circuit through the networking, and switches will help you to do that. So, once this is done, after that, you actually transfer your data. So, the next phase is original data transfer, original data transfer used to happen. So, basically, then you start transferring the data.

So, this is the technology they used to use. Now, let us try to go a little bit inside that. So, the switches we have already talked about, so they are automatic switching; of course, Strowger introduced it, and then after that, the switching has been much more sophisticated. We will talk about the switching when we will be discussing the circuit switch network in some future classes.

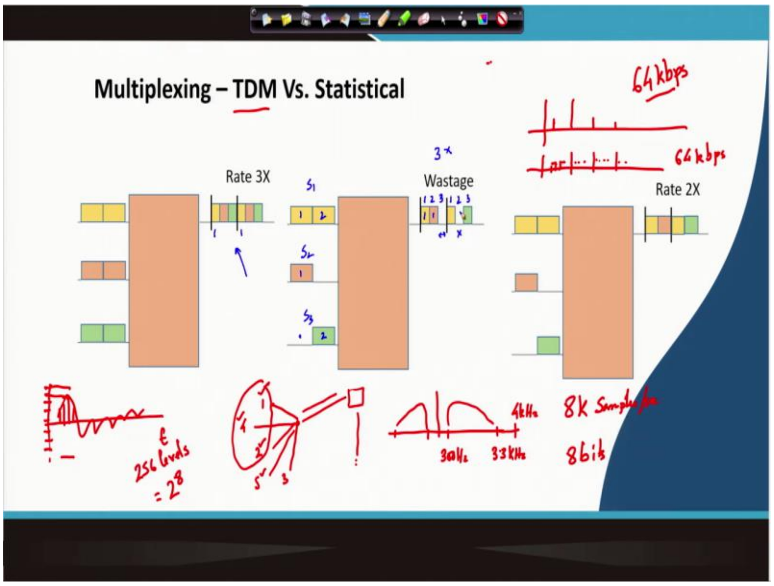
And what kind of switching do they do? There is a time switching and space switching. So, that is also something we will be discussing later on, that switching technology and all those things. And there is also other than switching; there is also some other thing which is called multiplexing, ok.

This is something I will give a brief idea about it; because if you wish to contrast it with the current networking, data networking, which is a packet switch network or IP network, you need to understand that multiplexing.

So, the multiplexing technique actually gets differentiated in that particular perspective. So, we need to understand this part; but this is the multiplexing they used to use, which is well known. You have already studied this part, probably; it is called time division multiplexing. So, that they used to do for the voice call.

So, there is a multiplexing; there is some associated switching, so that might be time switching or space switching. We still have not understood that part, but that will be clear in this course in due time. And the switchings are automated, so through this switch, you create a circuit; this is what is a circuit switch network. So, let us try to understand this multiplexing first, at least to get some idea, ok.

(Refer to Slide Time: 11:44)



So, let us try to see what this multiplexing is. So, over here, I will be concentrating on the first part ok, which is called TDM multiplexing. So, this is the part, ok. So, let us try to see what is TDM multiplexing. So, TDM multiplexing is something like this. So, multiplexing is somewhere if you have, let us say, two or three users connecting. So, let us say 1, 2, 3; you might have some more users let us say 4; let us say I also have 5, and it is coming to the switch.

Now, from this switch through the trunk, it has to go to some other, let us say, trunk switch to some other locality. Let us say user-1, 2 and 5 wish to go out, ok. So, basically, this user-1, 2, 5, and it might be even more than that. So, in a locality, there might be hundreds of users; out of them, ninety users probably at a particular time wish to communicate outside the locality. So, all those ninety users you have to take.

Do you want to take them individually over here, or do you actually; think the better idea will be for you to multiplex all of them together? So, suppose there are some users, let us say-1, 4 and 3. So, they want to talk to this locality. So, I can actually eventually multiplex them and create a single stream. So, that stream through the trunk will go, and this trunk switch will manage that single stream.

So, basically, if we multiplex the trunk switching and associated handling of those data will be a little bit easier. So, I can always do that; because three of them are going to the same locality, so why should I individually means, construct switching logic for everybody? Instead of doing that, I can actually multiplex them beforehand and then do all the things together in between so that in-between networking becomes simpler. So, that is one of the reasons why we wish to do multiplexing.

So, how do you do multiplexing? Let us try to see. So, let us say generally what happens; before going into multiplexing, let me talk a little bit about this voice traffic. So, voice we know that it; if you see the frequency domain, it generally from 300 hertz, it occupies up to 3.3 kiloHertz.

So, if I take the frequency spectrum, it is double-sided; some kind of pattern it will have, so 300 to 3.3 kiloHertz. So, if I just consider the voice bandwidth, I can go safely up to 4 kiloHertz, and we can say up to 4 kiloHertz, all the content of the voice signal will be already there.

So, my voice signal is of means bandwidth 4 kiloHertz, let us say in the baseband, ok. Now, what will I be doing? So, for voice, I have to convert it to, it is an analog signal, ok. So, I want to do a digital communication.

So, if that is the case, you all know about this PCM-encoded voice. So, what we do; we first do sampling. So, if it is 4 kiloHertz, according to the concept of the Nyquist sampling theorem, we have to do; means just to avoid aliasing and all those things; we have to do a double sampling rate. So, at least double the sampling rate.

So, if it is whole 4 kiloHertz, we have already given a guard band from 3.3 to 4. So, we can actually go and sample it at, which means double 4 kiloHertz. So, 8 kilo samples per second, ok. So, that should be my sampling. So, if I do 8 kilo samples, what will I be getting?

So, suppose I have a voice signal; this is in time, so I do this sampling. Each of those samples will have an amplitude value, but because the voice has a continuous kind of amplitude, that means it can take any value of amplitude, but that is not good for digital transmission.

So, I have first sampled it; after sampling, what I am trying to do is now we want to actually convert it into digital. So, for converting to digital, we have to give discrete levels. So, we start quantizing them. So, basically, whatever the maximum and minimum voice amplitude that can come, I actually subdivide this entire band into smaller bands; those are the quantization that I am trying to do.

So, let us say, generally in voice in PCM what they do; they actually construct such 256 levels, ok. Now, 256 levels in binary representation, how many bits are required? It is 2 to the power 8, so 8 bits are required; so 256 is nothing but 2 to the power 8. So, I know that 8 bits are required to represent each of these samples, ok. So, for each of these samples, I first see which level it goes to, the corresponding binary representation I give of 8-bit. So, basically, each of these samples will be 8-bit.

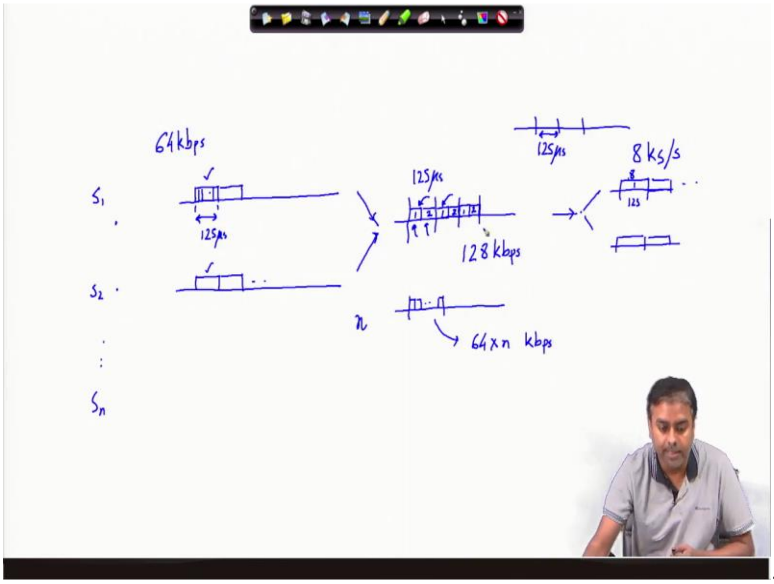
So, what is the overall data rate that is happening? So, there are 8k samples due to my sampling, and then each of the samples is of 8 bits. So, therefore, it is 64 kilobits per second. So, that is why any voice that we are actually means doing digitize it or PCM encoding it; so what is happening is we are getting overall 64 kilo means kilobits per second.

So, that is the magic number; 64 kbps is what we get for each of the voice signals. So, now, what is happening? So, if I now start seeing this particular thing, if I just enlarge it or enhance its time domain, so what happens? There are samples. So, each of the samples will now be represented by. So, let us say this is one sample, so this is another sample, this is another sample something like this. So, each of these samples will now be represented in a binary bit pattern.

So, there will be 8 bits that I can put over here; so 1 0 0 0 something like that 8 bit, again this will be represented by 8 bit, this will be represented by 8 bit and so on, ok. And the corresponding data rate should be 64 kbps, right? This is what will be happening. So, that is what a voice looks like after PCM encoding or pulse code modulation encoding. So, we are taking sampling, then each pulse we are actually representing it into a digital stream of 8 bits, and those 8 bits you can put one after another.

Now, what is multiplexing? Let us try to understand that part. So, what I can do is; we have got a 64 kbps sample, and now I want to multiplex them. So, basically, two voice signals are coming; let us say let me erase this at least. So let us try to see how we multiplex these two things.

(Refer to Slide Time: 19:33)



So, basically, what has happened; I have a voice signal. Let us say that 64 kbps is already PCM encoded. So, there is a voice source. So, let us say source; I have PCM encoded it, so how will that look in the time domain? So, it will look like every sample has 8 bits. So, this is the 8-bit representation. So, next, let us try to also understand, between samples, what is the time gap. So, because you have done 8 kilo samples per second, right?

So, basically, it should be 1 divided by 8 k, so that is 125 microseconds. So, this time slot will be 120. So, within these 125 microseconds, you are putting this 8-bit, ok. So, similarly next 125 microseconds next 8 bits, the next sample will be coming. So, this is what the first source will have. If I have another source s-2, this will also have similar things every 125 microseconds.

Now, what I can do; if I wish to multiplex, so basically, these two data I have to aggregate these somehow, ok. All I have to do is remember these 125 microseconds within that both these data have to be; because that is the one sample period. So, one sample can accommodate up to 125 microseconds because next to 125 microseconds, another sample will come. So, within this, somehow, I have to transfer both data because both have their 8-bit data.

So, what I can do, eventually, I can put them side by side, ok. So, this will be representing 8 bits; this will be representing 8 bits. So, now, within 125 microseconds, I have to transfer instead of 8-bit, 16-bit, ok. So, that is the crux of this multiplexing. So, what will be happening; because I am transferring more bits within 125 microseconds, so my data rate is getting increased. So, this will be instead of 64 kbps; this will now be 128 kbps, ok. But what is happening?

If I see in time, this is, let us say, source-1’s data; this is source-2’s data. So, again I will be putting interleaving them source 1s 8-bit data and source-2 8-bit data in the next 125, and I keep on doing that ok 1, 2.

So, on the other end, when I intend to demultiplex, what will I do? I construct from this 128 kbps, I will be actually selecting one data, and I put it in this 125 microseconds. So, time dilate it. If I do that, so in these 125 microseconds, now this 8-bit data will be there again next 125 microseconds.

So, just one’s data I will pick. So, I pick this one and put it over here, next to this one. So, I skip this one, next pick next 1s data, and I put it over here so on. And for 2, I will be doing the same thing; I post put 2s data over here, next 2s data over here, and I keep doing that. So, basically, I will get 1 and 2s data separately.

So, this is the crux of multiplexing and demultiplexing. So, if you wish to multiplex more than two things, you can always do that; only the data rate will increase. So, basically, what will be happening? Suppose n multiplexing I want to do. So, basically, there are n sources, and all of them I want to multiplex; what will be happening? Within this 125 microsecond, so there will be n such 8-bit symbols, ok.

So, basically, what will be the data rate corresponding data rate of the multiplex symbol? So, that will be 64 into n kbps. So, basically, the data rate gets increased as you multiplexed higher and higher numbers of streams. So, this is what will be happening, ok. So, this is how we wish to do multiplexes. So, as you can see, I have already demonstrated over here.

So, what we are trying to do is, we are trying to multiplex three streams, these are this width is 125 microseconds, so three of them. So, basically, they will keep on getting data. So, for source-1, this is the data one probably, then data two, and so on, it will keep on going.

For source-2, it is data one, data two; source-3, data one, data two. I have color coded them source one, source two, and source three. So, because three of them I am trying to multiplex, so rate should be 3 in 3 times that 64k. So, what I am trying to do, is within the

125 microseconds, we are actually taking data one. I am putting it over here; it will be squeezed in time because the data rate has been increased. Source-2 first data I am putting over here, and source-3 first data I am putting.

And their sequences are properly done; one then followed by two, followed by three, and that sequence you maintain, ok. So that is called TDM multiplexing. So, by this multiplexing, what is the facility you get? Basically, you can agglomerate all the incoming sources into a single stream, ok. So, that is one of the facilities that you get, that is one thing.

The second thing is, as you can very clearly see, within the time domain, as long as your times are synchronized, these whomever you multiplex, they have a particular address, ok. So, the first one of these slots is identified to be the data of the first stream. So, basically, as your this streams are synchronized or you can your frames are synchronized, you can actually pick the data from which source it is coming.

So, by seeing the location, position, and timing position of the data, you can already tell where from it is coming from the multiplex source, right? So, this is for time-division multiplexing; this is what is happening. So, this is the advantage. So, basically, you do not need to actually specify; from here, if you try to see, from the output, already from the location of the bits, you know which from which particular place this data is coming, so or which particular source the data is coming.

So, basically, the timing is already giving you the address from where it is coming; this is what is happening over here, and you are precisely multiplexing them one after another, ok. But that has a disadvantage; let us try to see that; probably this is the most important part where from the circuit switching, actually packet switching or I P switching actually came into the picture.

So, let us try to see if suppose these voices are all of 54 kbps. Now, suppose this 64 kbps is not always mean they are not filling the data all the time, ok. So, what might happen? So, within this 64 kbps, it might happen that sometimes there is nothing. So, basically, the voice signals, if you see, I continuously sample them, and I mean, do the PCM encoding, but sometimes there might be a null period, so nobody is talking.

So, at that point, I am actually not transferring any information; I am mostly transferring noise ok or transferring nothing as such; no information is being transferred. So, if that is the case, I am actually at that point. There might not be any data to transfer. So, let us take a scenario arbitrary scenario. So, I am trying to transfer multiplex these three. So, this is stream 1, this is stream-2 and this is stream-3.

Stream 1 has data in the first slot also, and the second slot also has data. But what is happening in the stream-2 has data in the first slot, but the second slot does not have any data; suppose this is the scenario. And stream 3, in the first slot, there is no data, but in the second slot, there is data. Now, I want to multiplex them. But because it is a TDM multiplexing, then what will happen?

The timing location is the address; so, therefore, I cannot actually play with this address. So, everybody should be; if I am multiplexing 3, everybody should be given its distinct location. So, this is location 1, location 2, and location 3. So, I have to give locations 1, 2, and 3 in the first slot, and then I have to also give locations 1, 2, and 3 in the second slot. Now, in the first slot, stream-1 has data, so I put that; stream 2 has data, so I put that ok.

So, stream 1 data 1, stream 2 data 1 I put; stream-3 does not have data over here. So, this goes blank over here. Similarly, in the second slot, as you can see, stream 1 has data, but stream-2 does not have data, so this goes blank again over here. So, what is happening? You make a three times rate, and then some of the locations are getting wasted over here. So, TDM multiplexing will always give you waste, ok. So, what we will try to look at in the next class is how we really reduce this wastage, ok.

So, that should be the major focus of these things; which means the next multiplexing that will be starting to discuss should be our major focus, that we are ok due to this kind of multiplexing, there is wastage. This was probably the building block of the circuit switch network, but we have to now appreciate that there is a wastage in this TDM multiplexing or circuit switch network.

So, can we really take out that wastage, do something else to mean again enhance the usability of the resources we put in or cost-effectively use it or reduce the cost of infrastructure? How can you do that? We will see that in the next class, ok.

Thank you.