Indian Institute of Technology Kanpur

National Programme on Technology Enhanced Learning (NPTEL)

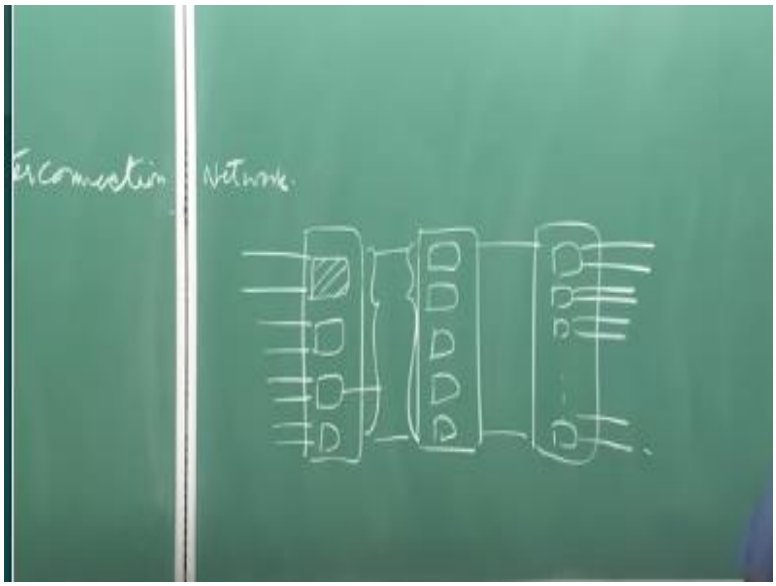
Course Title
Digital Switching

Lecture – 05

by
Prof. Y. N. Singh
Dept. of Electrical Engineering
IIT Kanpur

Okay, now let us move ahead we need to now ask a question.

(Refer Slide Time: 00:24)



I am going to build up a crossbar for n by n size n of course, whether I use complete all the cross points or I use the lower half triangle or upper half triangle does not matter, my complexity of the cross points is $0N^2$ okay. Now question is and of course, there is one fundamental property which I have told earlier that it is a strictly non blocking system. So what does that actually mean, if this is input and this is output of the switch if an input is free if an output is free

irrespective of whatever other connections are made without disturbing those connections I can always connect these free input and output okay.

That is what the strictly non blocking actually means. And that is a fundamental property of a crossbar okay, and the complexity is this. Now my problem is if I keep on increasing my switch size I go from 10 to 100, 100 to 1000, 1000 to 10,000 and so on my number of cross points which are required are growing at N^2 rate which is pretty large. So users have to pay a huge amount of cost per connection as which size actually grows.

So how to take care of this situation can I reduce my cross point complexity while maintaining the state you non blocking property that is the question. And of course that is one thing which I desire secondly can I increase this size to arbitrary very large values okay, people should ask why not yes probably we can do that. But you have to understand that all these cross points have to be controlled by a controller and mostly there will be one controller / crossbar.

So whenever a connection request will come it will process it will set up a connection it does take some computing load, so as my size actually grows my computing load also grows for which will be managing the cross points. So after certain size if you start increasing size to a very large value this controller which is going to control the cross point will not be able to take up the load.

So you cannot arbitrarily increase your size, size has to be limited okay. So because I cannot go arbitrarily very large I still require a strictly non blocking property of course later on I will tell you that actually it is theoretically fine but in real life situations I will probably will live with some small blocking probability, but I should know that what is that amount okay. So maybe how we can do this.

So idea is that let us use smaller crossbars and put lot of them together to build up a larger one not a larger crossbar, but a larger switch which will is to retain the same property of the strictly non blocking nature okay. So maybe how this can be done, so we can create something called

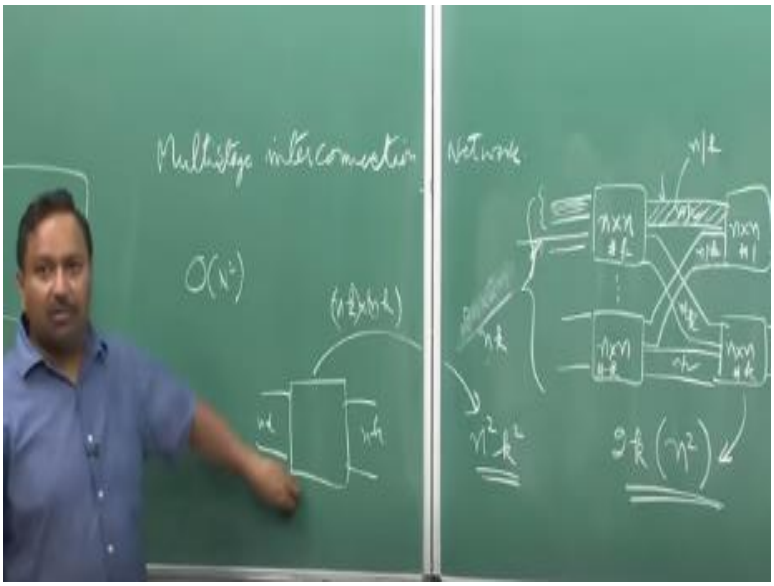
multistage interconnection network MIN. So that is what I am now introducing. So the way it will be done there will be not one they will be multiple stages.

So I will put my lot of cross bars in one stage I will create another stage I can actually when any number of stages I can put another one I can add any number of stages only important thing is that all my inputs are being coming here in the first stage and all my outputs will be emanating from the last stage. And of course, I had also need to ensure certain things for example I cannot leave a port free here all the ports here should be connected to all the incoming ports here.

So this is important how this will be done is a design problem how to do it, we will actually discuss these as we go along. And so means no input port no output port is free here no input port is free here all output ports here have to be connected to some input port on this side. And so on only these will become output ports T's will be the input port that is a multi-stage interconnection network.

Now idea is that obviously do not know how many stages will be required, I want a strip you non-blocking switch I only know these basic building block is a cross bar, so it is in itself is a shitty non-blocking. So I do not talk about this network which has to be strictly non blocking so can we do it with two stages let us ask this question. So let us try if we can do this, so I am going to build up a very simple switch and see if I can get a strictly non blocking switch or not.

(Refer Slide Time: 06:02)



So let me put a up a I am now putting a small n here okay and two of them so probably I want to build up at 2n / 2n is strictly non-blocking switch using n/ n small n by n cross bars and I am going to put them into stage lets me try it with two stages if this can be done so there n out outputs here there n outputs here now how to connect in between of course most common sense logic says because n input should be able to connect to any one of these outputs I should connect this switch to any one of these there are only n outputs coming here so I can take n/2 this side and n/ 2 on this side.

Same thing I can do here wow I got a 2n / 2n switch but is it restricted non-blocking so how to find it out see remember my condition was that an input is free an output is free I should not disturb any existing connection and should be able to make a connection between them so if I can create a situation where it is not possible where it is not possible to set up a connection when input and output both are free then I will say this is a blocking system this is not a strictly non blocking switch.

And henceforth this arrangement is not going to work so let us see how I can do this soul let us take n / 2 of these connections they are connected to n / 2 here in whatever pattern so I can all

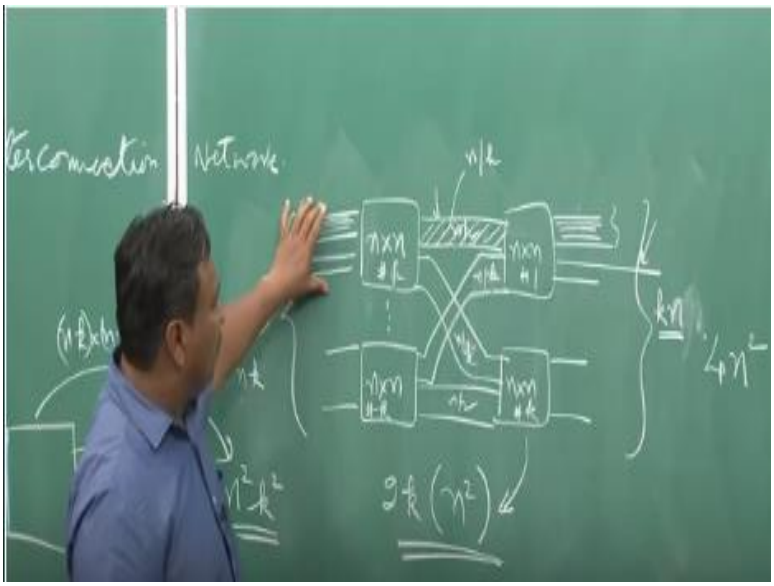
these lines are busy now there is one input which is free one output which is free I want to connect these two how to do this all the n / 2 are already occupied if I use bottom line I cannot come back here there is no way I can set up a connection so I have shown that by one example that this is not a strictly non-blocking switch is a blocking system.

So if a two-stage it is impossible to build up a strictly non blocking system okay and of course for this to n / 2 n my cross point complexity is actually 4 x n^2 for a 2n / 2n system their forces which is of course cross point complexity turn out to be 4 x n^2 here also so same cross point complexity but I ended up in a blocking system but I can generalize with 2n / 2n it turns out to the same but if I make it k such elements so 1 to k they many such things so I am actually dividing this will not be n / 2 this will be n / K.

So many n / k will be there as all of them will become n / K so total thing will be k into so nk that will be a total number of ports the ports on this side will be nk so when you build up nk /nk switch you require n^2 k^2 number of cross points when you have to stage kind of network I have 2k n / n switch so n^2 so these many cross points will be required and this will be a blocking system the both switch is are actually nk x nk both switches are nk x nk but if I build up a single crossbar with nk here and nk here I have these many cross points will be required if I build up a system like this I need these many cross points.

It is actually reduced by factor of k okay so you are divided you divide you actually multiply by 2, k /2 essential it is reduced by factor of K / 2 so if you divided by K / 2 you will end up in this you will require less cross for less cross points obviously but you are losing your restricted non-blocking property can I make it into a strictly non blocking system is still and using a smaller blocks remember even if I do that I require multiple CPUs so there will be separate controller for each one of these switch blocks.

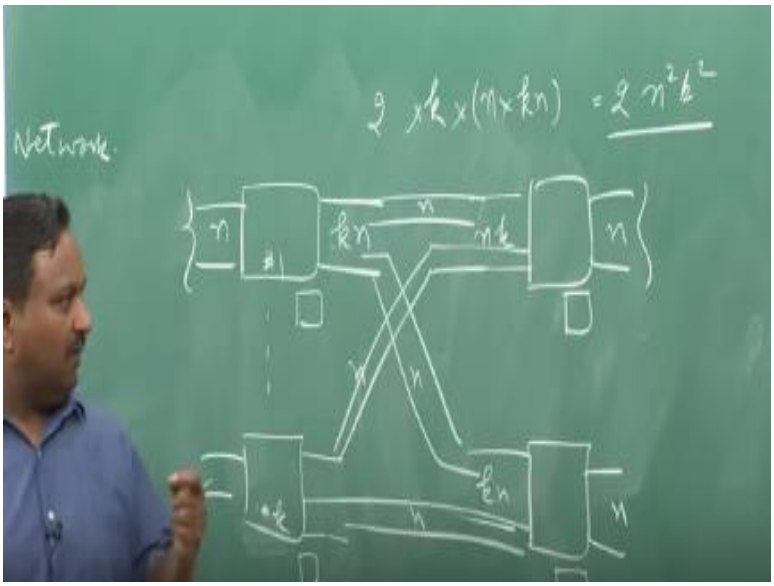
(Refer Slide Time: 11:19)



I am here only I will be requiring one control block one CPU which will be heavily loaded here CPU load will get distributed at least it become a distributed system so if you do that how this can be done so we have to actually come out of the notion that I can build up a crossbar of n / n only know in general I can build up a crossbar of n /k also so I want to build up a 4/8 I will have 4 input lines and I will have 8 output lines but remember if it is a one-to-one connection only 4 of these output lines will be occupied at any point of time.

But this is 4/8 crossbar so if I can do that in worst case all these small n number of lines has to connect to this the way it has to be done is if I want a strictly non blocking system.

(Refer Slide Time: 12:25)



With two stage interconnection I will take n as inputs I will take kn as the output so number of outputs I will increase because I can make 4/8 I can also make n /kn and I can do multiple of them okay such units and similarly I will have in the second stage also similar units so since their kn so I will be taking n here and to the last guy in same we will be doing in here n here n of course to all intermediate stuff so if all connections are required to be done here I can always do it is a strictly non blocking system but number of cross points which will be required is now n x kn into k such switches and 2 of the 2 stages will be there you will end up into n^2 k^2 twice the number what is required by a single crossbar.

You do not gain anything in terms of the cross point complexity you only gain in terms of using a smaller multiple controllers multiple separate CPUs which will be controlling so your computing load gets distributed which is setting up the connection which is dismantling the connection but we can actually further improve on this. What we will do is I will we need to essentially now discuss one more important point before we move ahead in real life situation as I mentioned I do not need is written on blocking property because I might have to put tremendous amount of hardware to create extra non-blocking property.

But do we want this kind of situation VIP treatment for everybody I might be happy if I want to make a call it does not go through once in and once in 10 years or once in here it is okay I will wait again make an attempt so I will I will most of the real-life situations will actually have a blocking switches because my cost goes down drastically okay my amount of hardware requirement goes down drastically and the performance maybe one in 100000 calls or 1 in 1,000 calls will get blocked and I have to attempt again after some time which is acceptable to as a user because it is going to reduce my cause if I want every call should go through immediately.

Then my cost is going to be huge in setting up of the system and even if you want a strictly known blocking switch there will be situations when you will know your call will not be able to go through that will happen when the destination is already busy is already occupied and you cannot connect to him because he is busy so there will be very small probability even for that.

So we accept this degradation in slight degradation in service which reduce determine which leads to reduction in lot of cost for us so now it is important that I have to now go to a situation so 2 stage will not work so I will try to go with three stages and then we will try to make an estimate that what is the blocking probability because I do not want a strictly non blocking switch I want a blocking system so we will look into how to estimate the blocking probabilities in the next week the lectures which will be given the next week.

Acknowledgement

Ministry of Human Resources & Development

Prof. Satyaki Roy

Co-ordinator, NPTEL, IIT Kanpur

NPTEL Team

Sanjay Pal

Ashish Singh

Badal Pradhan

Tapobrata Das

Ram Chandra

Dilip Tripathi

Manoj Shrivastava

Padam Shukla

Sanjay Mishra

Shubham Rawat

Shikha Gupta

K. K. Mishra

Aradhana Singh

Sweta

Ashutosh Gairola

Dilip Katiyar

Sharwan

Hari Ram

Bhadra Rao

Puneet Kumar Bajpai

Lalty Dutta

Ajay Kanaujia

Shivendra Kumar Tiwari

an IIT Kanpur Production