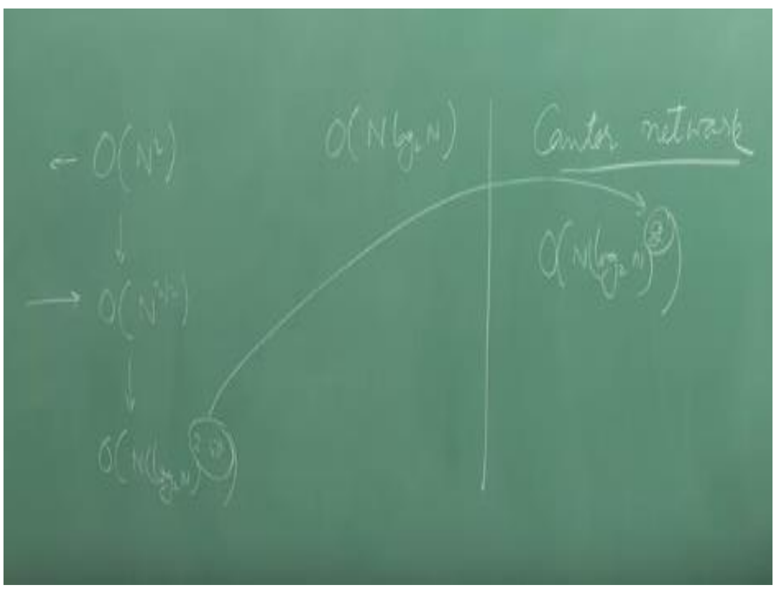
Indian Institute of Technology Kanpur
National Programme on Technology Enhanced Learning (NPTEL)
Course Title
Digital Switching

Lecture – 21

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

Okay in the previous video what we had discussed was recursively constructed reimaging non blocking switch as well as restrictive non blocking switches and as a consequence what will be the cross point complexity okay so and then of course we have estimated that we could improve my cross point complexity from.

(Refer Slide Time: 00:40)



When square which for a cross bar using a Clos network I was able to compute 3/2 by doing recursive construction I was able to come to log to n 2.58 okay while for rearrangibly non blocking switch we actually came up with a cross point complexity of this of course I had already mention that while coming from here to here this was actually strictly non blocking n/n cross bar for uni-casting as well as multicasting scenario any arbitrary multicasting scenario can actually be handled here but when we came to this Clos network this was not for arbitrary multicasting was not supported for this one I have not actually given you the cross point complexity when the multicasting.

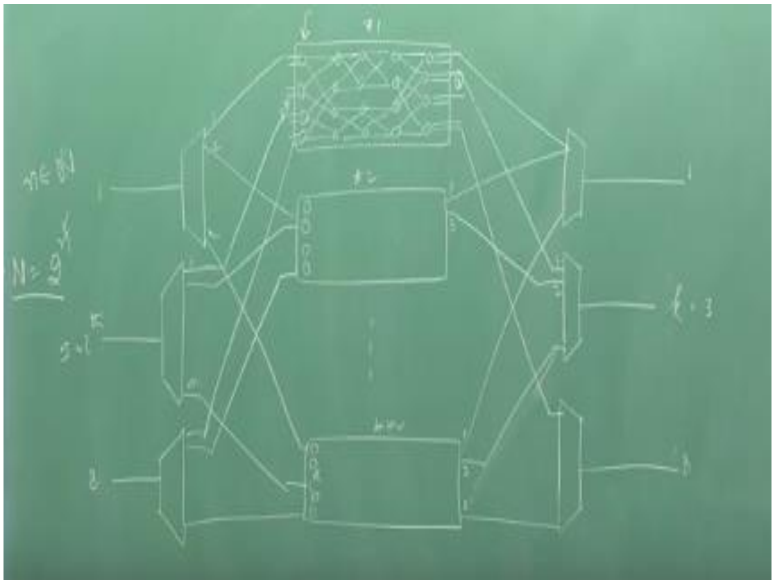
Will be multicasting will be taken care of so if f is become equal to r3 which is number of switches in third stage then it become arbitrary multicasting actually can be supported so we have not computed the cross point complexity of that but if use Clos network for uni-casting coming from here to I actually lose the strict non blocking each or for multicasting but return it for uni-casting in my cross point complex reduces if I do recursive construction I can actually further reduce it okay so and this was done for the rearrangeably non blocking switch, now we would like to figure out if we can further improve on this particular bound can I further go down, so this question was also is still actually prevalent we still do not know what is a bound, it is going to be worse than this.

But we do not know how much so it has to be somewhere between these two, so canter actually discussed this particular issue and he came up this something called Canter network and he computed that I can he can build up a, a strictly non blocking switch in this configuration and cross point complexity for this after you estimated everything was log 2n this power 2, so from 2.58 I was able to come to 2, okay.

From 2.58 I was able to come to 2, we still do not know whether we can further improve it where it is going to be O and log 2n 1.75 also we do not have the answer so we will leave with this particular bound and then we will move to our next video on white sense non blocking system, so how the canter network will look lie. So once I actually today what I am discussing is the canter network topology.

And we will be estimating what is going to be the cross point complexity for the scenario, so cross point network this canter network will look something like this.

(Refer Slide Time: 03:54)



I have already told you the structure of a Ben's network so for 8/8 for example I am drawing a 8/8 canter network I have to draw at a Ben's network here so bens network will look something like this, this should have 5 stages and this will the connections so I think all of you can recall this so this is 8/8 bens and we know as we have said earlier one of the videos that this is the rearrangable non blocking switch okay so blocking will happen in this case.

So we actually kantar came off with the very simple idea that let one input coming I will use 1/ m switch and I will use m such switching paths so this is the first one first bens network this is the second one and this is the n^{th} one so input one can be actually rooted through this particular bens network see this is the blocking system if it cannot the connection cannot be setup of this can be set you wire two if this also you found the things are blocked then it will go to m and so on.

So if I keep on increasing and this actually means this is how it is connected this is the n^{th} connection okay so this is 1 2 and there n^{th} connection here so if I keep on expanding this on the similar on same lines on the output side we will have something like this actually there is an output and m lines from m different Ben's network will be coming and will be put together. So this output will be connected to anyone of them.

So if I keep on increasing the same this switch should come sit in one blocking, so question which is now is that what value of m this will become a strict in one blocking. So we will investigate this question now. So in this case I have to draw other inputs because it is not one there actually n inputs so there is a n^{th} thing, I can have the i^{th} input also let me put something intermediate.

So that I can show the connectivity, so in this case it will be 8/8 so this will value should be maximum should be 8, in general it can be n/n then we will have n/n base network which will be there and of course we are assuming that $n = 2^n$ okay. So is equal to this 2^n is a small thing and n in end belongs to an integer, positive integer, okay. So I can call it unnatural number set so n belongs to that.

And this is the size of n, in this diagram because I have drawn 8/8 Ben's network I will be having 8 inputs here, so the first input in the i^{th} switch the first input will be always connected to the first switch and its i^{th} input actually, so this is i^{th} which is in this case is 5^{th} so I should make it equal to 5 actually. So this I will be connected to the 5^{th} one, okay. So i^{th} switch here let us connceted to the i^{th} input from this 5 input of this particular switch.

Because the number of this marks or demarks is 5, okay. So similarly the m^{th} here will be connected to the m^{th} in this case the i^{th} input here will be connected to the m^{th} input of this one, this logical will hold true for everything, okay. So this two will get connected here this two will get connected here and on similar kind of connections will be happening on the other side, on the output side.

So I will have one and this will be again I am taking some values A I can call it K, in this is the 8 which is equal to n in this case, so if K can be here 3 so the third number line will come up and connect to the first one first from first switch, so this will be 3 so remember this port number corresponds to this input number this port number corresponds to this witch number, on this same side also again.

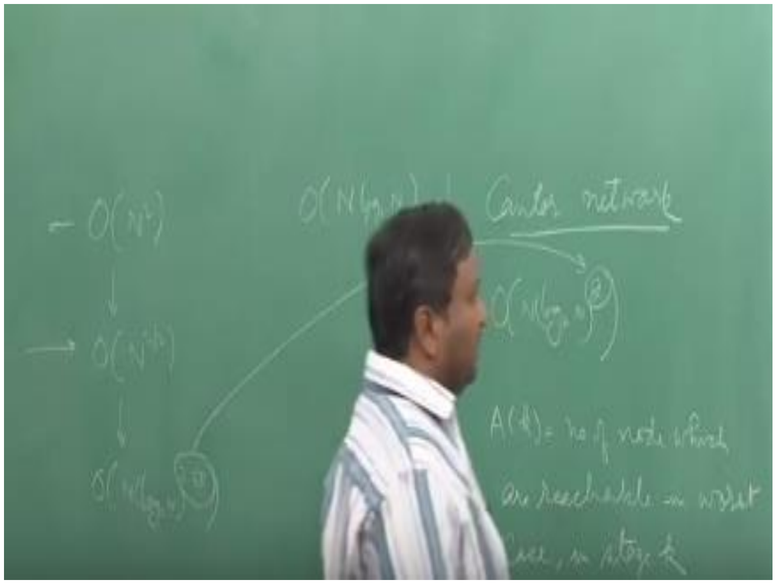
This port number corresponds to this switch number and this port number corresponds to this switch number, okay. So this correspondence is always done so that you can symmetrically construct this particular switch. So the last one will be coming here in this fashion this last one will be coming here to the third one, so this will be third one which will be coming and this is the 2, so that is what the kind network is.

Now only question which remains is what will be the value of n? So now let us compute so we will do a very different thing this time, what I will try is because let all the 7 other connections are already set up and I have one free input and I have one free output which need to be connected, okay. So if what I will do is, I will try to find out to how many nodes in the first stage the free input can go.

Assuming all of the connections have been made and I will count not only these four I will count even these first stage also, there first switches even setting in here so I will count all these actually. So in the first stage in this case it is 32, 8 x 4 number of switches in the first stage which are present, so how many of them you will be one will be able to reach, one can actually reach or whatever is a free port.

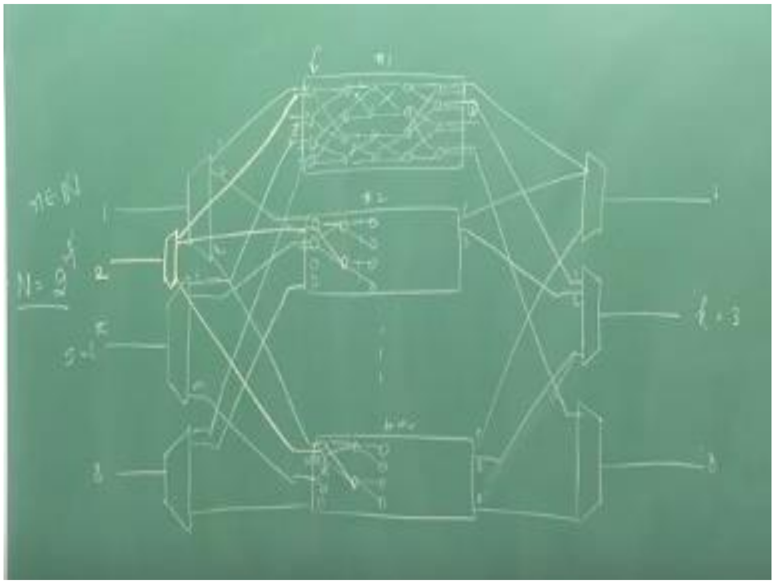
So let me assume it is 1, it is true for any one of the input port if it is true for one, it can be each 1 here, 1 here and 1 here so it can be in total m or is equal to 8, so I define this number.

(Refer Slide Time: 11:26)



I define AK as number and I call each one of them as a node, okay now so number of nodes which are reachable in worst case in stage K, okay. So A1 in the first stage what is the value of A1 will be m.

(Refer Slide Time: 12:08)



In this example it will be 8 so now once you can actually see this all these 8 switches this one this they now can actually be reached by 2 inputs one and if there I report have been another one so let me draw it with the different colour it is there I have not drawn it so 2 somewhere sitting in between so let me just draw it for clarity so this 2 will also be connected to the same port this 2 will also be connected to the same port so one and two the same switch.

So amazingly this one 1 into both are able to reach to the same set of notes in the first stage okay and since I am assuming that all other inputs are busy so when you will try to look in to the second stage how many notes are reachable so from this note it can reach to these two from this not again it can reach to this 2 from this note also again it should be reach to this 2 okay but since 2 is already connected it should be actually connected either with one of on two three four and five one of this six actually with the once.

So one is already occupied so number of notes which are reachable in this second stage will be so a2 the number of notes which are reachable they should be twice the number because each one of this switch is connected to 2 further 2 note is the next stage third stage it will be again two more twice of whatever is reachable in the second stage so a_2 should be equal to 2x a1 but as I said that input 2 is already busy.

So it is already connect so one of them as to be minuses s to be subtracted so the inter fearing one the two had to be there so one node has to be made less so whatever noted which are reachable by 1 a1 those can be further reaching to the third stage and a3 will be twice of a2 but – how much now where I look at these two notes they are reachable from corresponding to third and forth inputs also.

And I am assume there also occupied similarly these two can also be reached where these two ports which corresponsive of course same third and forth inputs is there already connected it means two of these links which are going out from here it is going to these so this and this one which I am now putting so these are actually 1, 2, 3, 4, and these of 4 x whatever the 8 actually whatever numbers which are there here m 8 x m so those many switches are reachable now but two of the occupied by these two third and forth input.

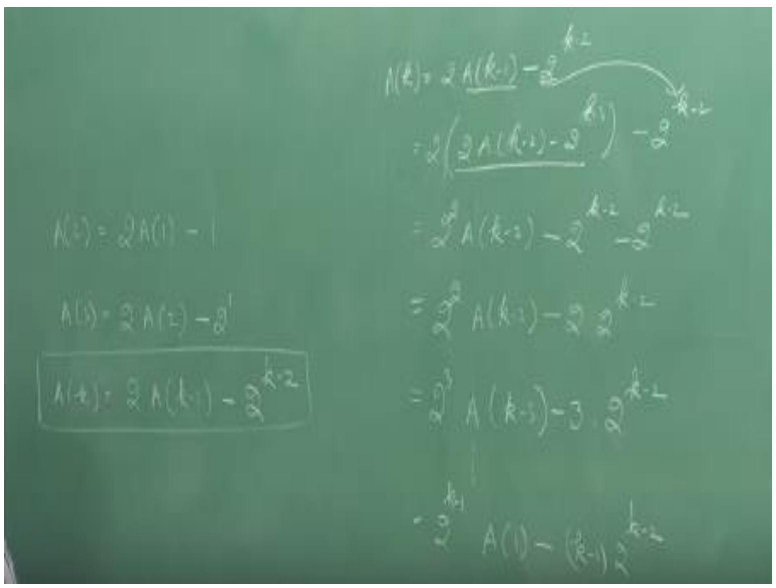
So now it has to be 2^1 in general if you take a larger switch you will find that a of k now remember I cannot go any further than the middle stage and up till middle stage I can do so here only I can have a_1 a_2 and a_3 I cannot compute for a4 here so I have to a 16/16 switch to compute for a4 and 32/32 for computing a5 and so on. So a k will be now given as 2 a^{k-1} -2 $^{k-2}$ so you can actually see the relation.

So once you have got this relation of reachable I am always interested and finding out in worst case scenario how may notes I can reach in the middle stage from the free input 1 okay so let us do that we have to do the recursive computation I can use this formula and then recursively compute because I can now put of expression for ak-1 is in the same re correction and compute what is ak in totality let us see.

So once I have ak which is being given here so what will be I can actually now put ak-1 expression so this will be using similar thing it will be 2 x ak- 2-2^{-k-3}- 2^{k-2} so this comes here is this, this I have expended in to this and if I solve it, it will be 2^2A(k-2) – 2^{k-2} – 2 ^{k2}. I can keep on doing it and if I keep on doing it, I might actually get 2^3A (k-3) -3 – 2^{k-2}. And so on once I reach to 2A(1), so remember this and this when you sum together, it will always come out to be case where is to be 1.

So it has to be k-1, when you sum these to it will be equal to k, this is nothing but number of terms, here it is 3, here it is 3, here it is k-1, so I should have k-1 here $2^{k-2.}$ Because this is a constant thing which will keep on coming, only this number will keep on increasing, So k-1 time we require and this will be the final expression for Ak. A1 we already estimated but equal to m, so I can replace this by m, the total number of base network which are there.

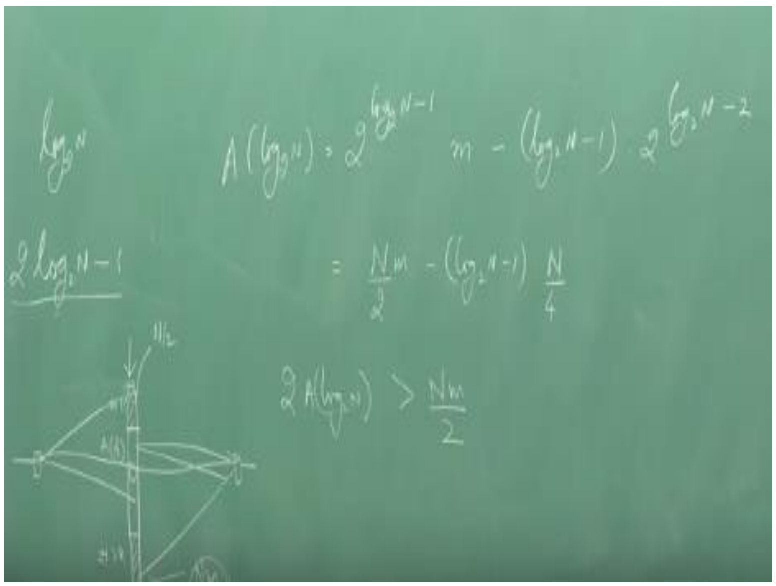
(Refer Slide Time: 20:07)



Now I'm interested, if it is, let's move to a general scenario, so general scenario is for n/n switch, so n/n base network is used, the middle stage will be $\log_2 n$, so total number of stages in the base network are $\log_2 n$, 2 $\log_2 n$-1, I'm looking for the middle one, this is always a odd number, remember. Ok $\log_2 n$ stages, how many nodes I can reach, I'm interested in that, so let me do that. A ($\log_2 n$), so I have to keep on putting the value of k and then keep on solving it.

So these are number of notes, which is free input 1 can actually reach, so solving it, it become $2\log_2 n-1$ will become 2, this will become, same we will do it will be n/4. Now the argument which we are going to have is, that there is one free input, in the $\log_2 n$ stage, there are total number of lot of number of switches. Total number of switches which will be there in middle stage will be nothing but, each base network will have n/2, and there are such base networks which are present are m.

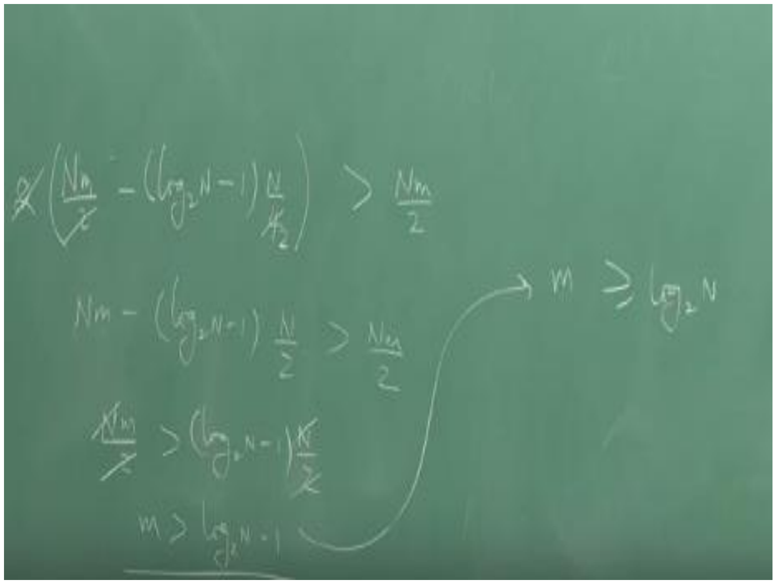
(Refer Slide Time: 21:38)



So total number of middle stage switches will be Nm/2. Now free nodes which I can reach from the free input will be some number, this number is given by A of k, here k is $\log_2 n$, same is two from the output side, so I can reach certain free number, now if I add this number Ak, and whatever I can reach this well, of course it will turn out be the same expression. These two numbers are added and whatever numbers of switches which are available, then there are bound to be some common free node, which is available.

There has to be some node which is common between in this set, and I can always use that to root the call without disturbing the existing calls. So in that scenario, I have to simply take 2A ($\log_2 n$) should be greater than Nm/2. This is the condition for restrictive non blocking operation of the kantar network. So once you take this, I can put A ($\log_2 n$) from here, the above expression, this will turn out to be, let me write it on the left side.

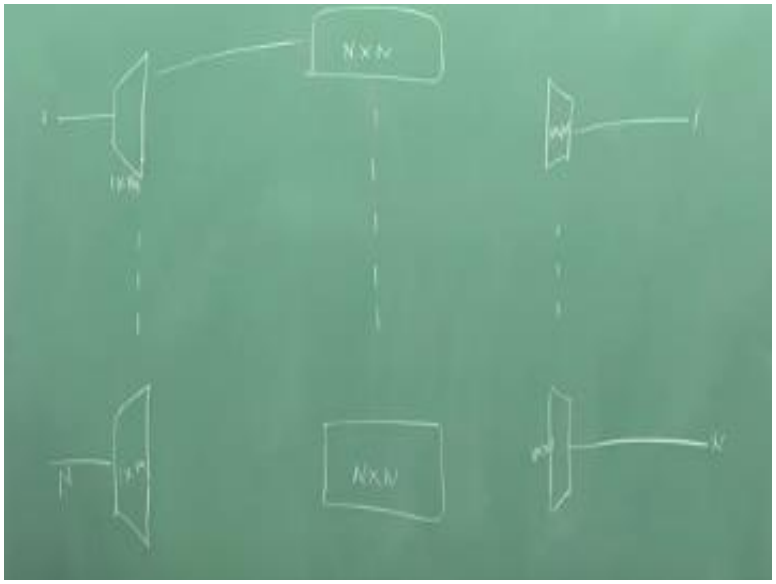
(Refer Slide Time: 23:51



So it will be two time A($\log_2 n$), I'm just putting Nm/2, two times of this minus $\log_2 n$-1 into N/4 twice of this has to be greater than nm/2 okay so once I do this ,this 2 will cancel with this and this will become 2,so actually I can write it has this minus this will be nm/2 this term I can take on that side it will become positive so 2 cancels with this n cancels with this $m>\log_2$ n-1 this actually implies the $m \geq \log_{2n}$ so if m is satisfying this condition the switch will be strictly non blocking .

Now lets compute what will be a number of cross points which will be required in total for this particular switch.

(Refer Slide Time: 25:18)



So if you look I actually used demarks this is the 1/N switch okay because 1/M actually where M is $\log_{2n}$ and there n such switches are used which are 1/m switches then i actually had used these network which is n/n and I had used M such beans network actually and then there was a m/1 switch, n such switches have been used.

Now let's compute the cross points total number of cross points which are required this one require m cross points their n such elements m*n will be there each beans. Network will require a n/2 switches are there two state switches in every stage and $2\log_{2}$ $_{n-1}$ stages are there so $n/2*2\log_2 n$-1and each 2/2 will require four cross points.

Okay and we have in total m such switches so i will put in m here plus on the other side I have m/1 so you require m cross points an n such switches so m*n so this a total number of cross points which will be required so i can now put it 2 mn this mn also becomes $\log_2 n$ so I can solve this one plus 2n so this will become $4n\log_{2n}$ -2n ,m becomes $\log 2_n$ here so this will be this cancels with y this your number of cross points which are required are 4n $\log_{2n}{}^2$ which implies my cross point complexity is $o(n(\log 2n^2)$ and which is a strictly non blocking switch .

Okay so thus we have proved we can improve the cross point bound which came from recursively built restricted non blocking switch and we can do better so that's what the entire network does .

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