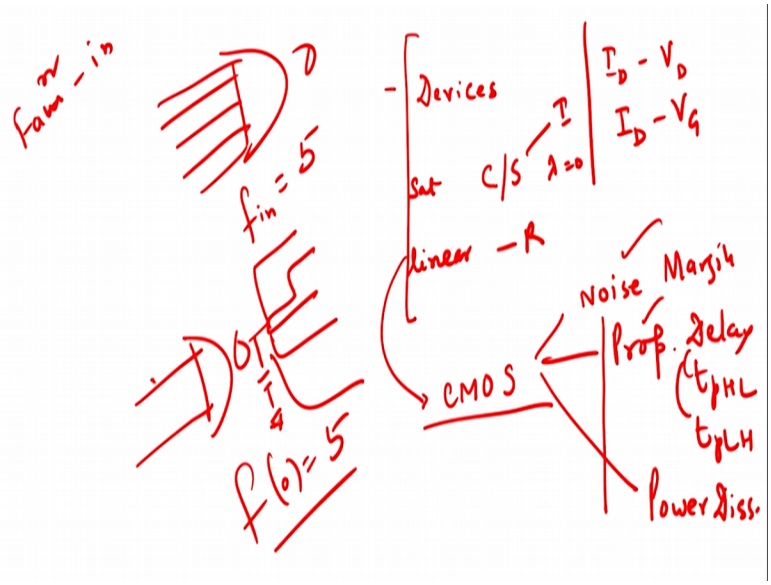
CMOS Digital VLSI Design
Prof.SudebDasgupta
Department of Electronics and Communication Engineering
Indian Institute of Technology – Roorkee

Module No # 03
Lecture No # 12
Combinational Logic Design –I

Hello and welcome to again to the NPTEL online certification course on CMOS digital VLSI design. We start this module by on combinational logic design so this module is primarily on understanding the combination logic design let me just before moving forward therefore let me give to an idea about what we have learnt till so that it is easier for you to gather information on what we expected to do in the next modules of combinational logic. What we have finished till now is basically two are three important things as far as this course is concerned.

(Refer Slide Time: 01:04)



First we have done devices and therefore we have understood what is basically a enhancement mode and depletion mode MOSFET? What is the meaning of the threshold voltage of the device? What is the meaning of NMOS and PMOS? And how can I change my threshold voltage my application of a sub-state bias we also understood what is the meaning of IDVD and IDVG and finally from the device point of view.

So we have understood IDVD we have understood IDVG right and from device point of view we have understood that if the device is saturated is kept in the saturated region it behaves as a

current source and saturated region right assuming that lambda = 0 which is no CLF and it behave like a resistor in the linear region right and it behaves as a resistor this behaves as a current source.

With this knowledge we then shifted our function to what is known as the complementary MOS right and we saw what is the meaning of complementary MOS what is the meaning of threshold voltage for NMOS and PMOS and we also saw that if threshold voltage are known to you can you predict how the output will vary if the input varies from 0 to 1 and 1 to 0 right and we were able to explain to you the various concepts of what is known as noise margin right.

We have also understood what is the meaning of propagation delay right propagation delay low to high which is tpHL high to low and tpLH low to high. So these two we have already understood what is the meaning of that on one factor these depends and third very important properties primarily the power dissipation. So we have devoted about 2 and half module for individual each one of them.

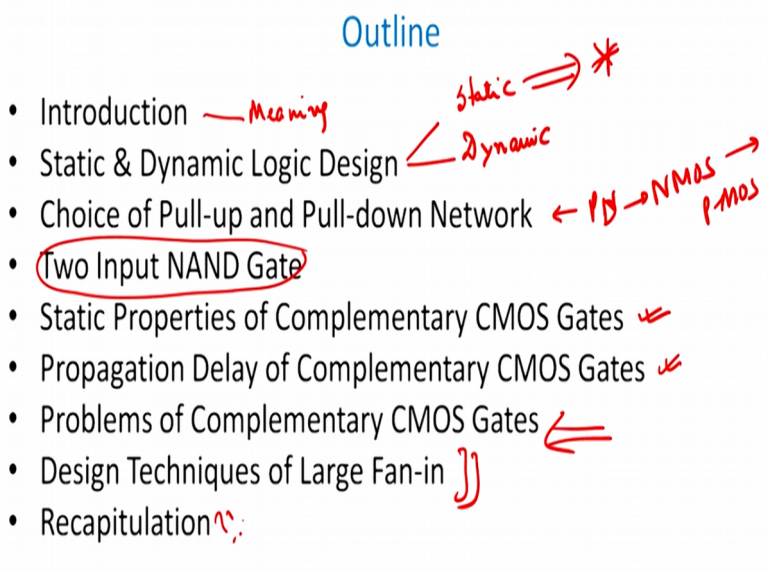
So we know how to therefore calculate noise margin on what factors they depend and how to optimize it we know how to calculate the tpHL and tpLH and therefore the intrinsic delay of the gate and also how to calculate it. We also know the meaning of power dissipation, what are the various factors which influence the power dissipation and therefore what is the methodology to optimize it.

We also learn very important issue that power dissipation and the propagation delay they do not go hand in hand. So if we want your power dissipations to be low you actually end up having larger delay. So this is the few important issues on the we finally in the previous module discussed about the about simulation tools which help to me why because for 1 or 2 active devices you can easily do it using pen and paper.

But when you have very large number of active devices even say 5 or 6 active devices with large number of resistance and capacitances solution, solving a multiple KVL and KCL is an another difficult task. Similarly solving more than even a second of differential equation is relatively difficult task by hand and therefore we took the help of spice simulations right so primarily we finish with spice also we will explain to you as move along for this course structure.

With this let me give you a brief outline or what we will be doing it in the next half an hour to 45 minutes of this module.

(Refer Slide Time: 04:23)



I will be giving an introduction what is the combination what is the meaning of combinational logic I will give you an idea about that so the meaning will be known to you here right the meaning of combinational logic we will come to then what is the known as the static and dynamic logic so we have two types of logics available to us static and dynamic.

We will be at this module only referring to or concentrating on static parsley so dynamic will be postponing it for subsequent modules but currently we will be looking at the static then we will also do an important job which is the third part is that if as I discussed with you all my pull down network VD in and pull down is basically of NMOS and all your pull up is basically your PMOS right.

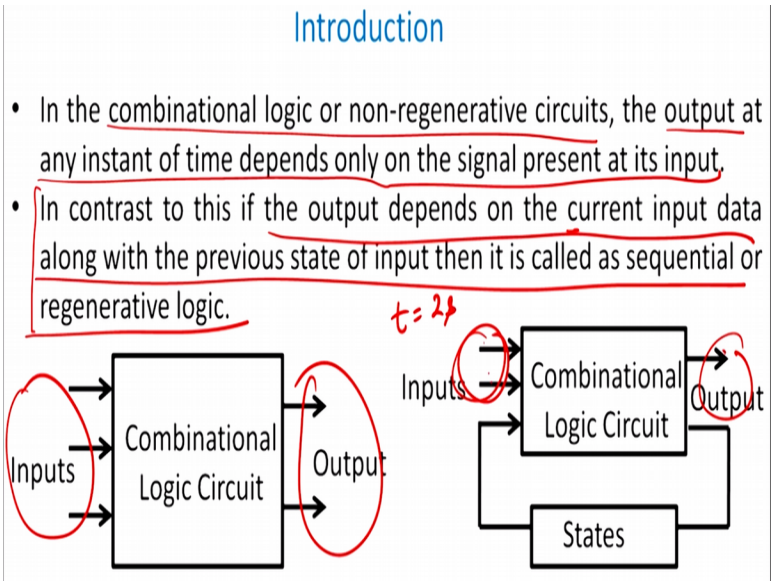
So why is that like that in most practical sense we have to look into it we will take an example of a two input NAND gate and we will see how it works out in terms of finding out the output characteristic and at what factor does the noise margin and these change as far as combinational logic is concerned. We will look at the properties of the CMOS gate and then look at the propagation delay we have already seen him we will just do it and we will show some problems in complementary gates and then we will look at the design of large fan in gates of design.

And as I discussed with you how to define fan in and fan out before we move forward we defined fan in as total number of input which is active on to particular devices let us suppose I have a NAND gate here and there are five input let us suppose available to you then we define fan in = 5 right. Similarly if I have aNAND gate at driven by two gates here but this itself is driving say five gates here then we define fan out to be approximate = 5 right.

We have already discussed this point that if you have large fan out you end up having very large capacitances here and the speed might decrease right. So we need to find out where is the means of having large fan in combination logical block and how to reduce it to low fan in combination logical blocks without effecting my overall power dissipation. So this will be having an look into the in subsequent sections and then we will finally recapitulate overall module for this stock right.

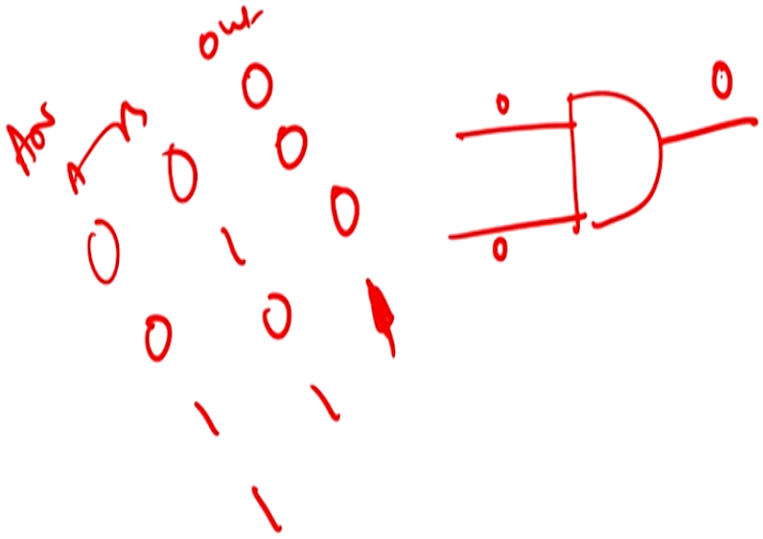
So this will be the general scheme of things which will be taking up in this case. Let me explain to you before we move to anything else what is the meaning of combinational logics what is the meaning of sequential logics there are two types of logics which is available as a designer in front of you. The first logic is defined as combination logic and if you can go through sentence which is written here.

(Refer Slide Time: 07:09)



That combinational logic or non-regenerative circuits the output at any instant of time depend only on the signal present at that instance in its input right brief example let me tell you.

(Refer Slide Time: 07:26)

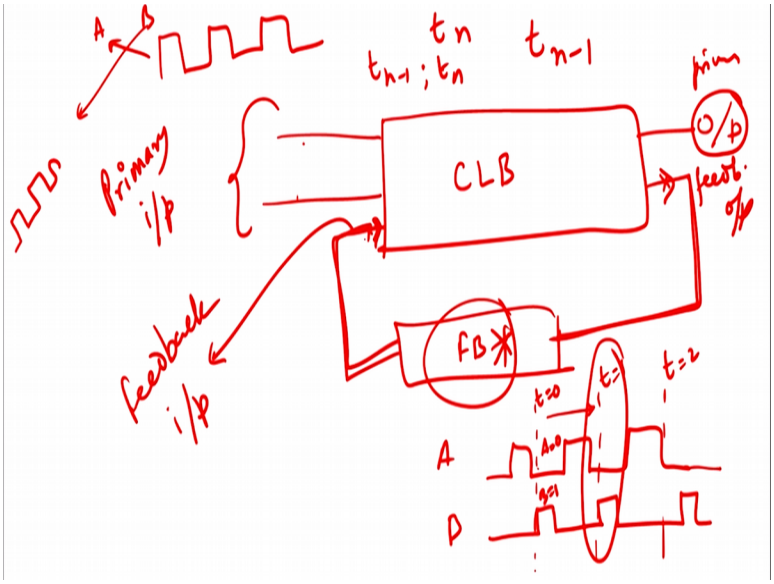


Say I have a simple AND gate right what is AND gate? AND gate obviously you are aware of the fact that it an AND gate A, B and out is there then I get something like this right the and I get 1 here right which means that if this is 0, 0 at any 1 particular point of time then the output here will always be equals to 0 independent of what is happening else anywhere else right.

So please be very careful what do mean by combinational logic? Combinational logic as the name suggest is non-regenerative which means it does not depend upon any value of input which was there earlier right what is the present state depending on that only you will have the output available to you. So this is what you get as your **as your** input here right in contrast when I say that this one this is sequential logic if the so if you look at the sequential logic if the output depends on the current input data.

Not only that but it also depends upon the previous state of the input then it is called as a sequential or a regenerative logic. So have two types of logic one as in one case it is combinational logic in which the output here only depends upon the input right where is it a sequential logic you have output not only depending upon the inputs but also of the state which was occurring to it at t – 1th second.So let us suppose at t = 2 second these are the inputs right I will explain to you in much more detail manner.

(Refer Slide Time: 09:10)



Let us suppose I have combinational logical block here this is combinational logical block CLB combinational logic block I have got two inputs here right and I have a output here and this output here and this is as I discussed with you is the feedback element here so this if you go back is the feedback element here right this is the feedback and then this goes here which means that this is defined as here primary input this is your feedback input this is your primary output which you see and this is your feedback output.

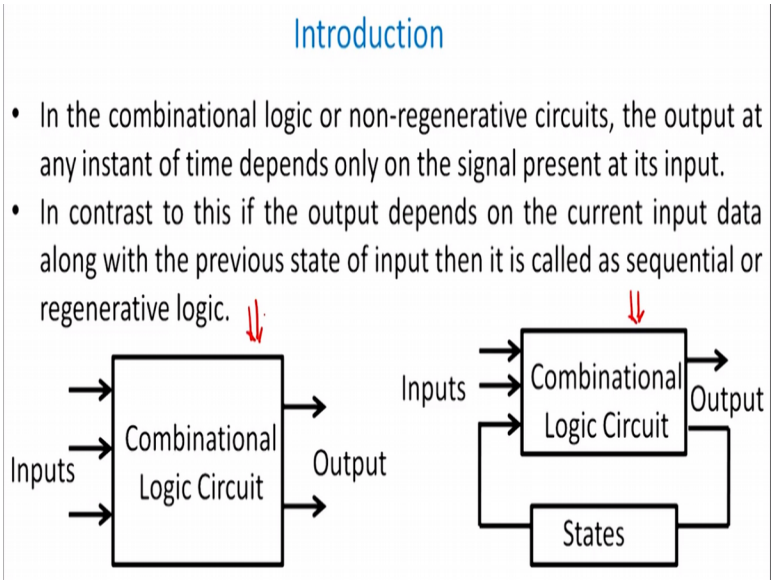
Now you see if you would have been a simple CLB or simple combinational block this whole data would not have been there this which means there this part and this part would not have been there depending on this two inputs I get an output. But it is an sequential logical block because please understand that data is coming in time domain. So I will have suppose A, B right so A is coming like this right this is A let us suppose and if I want to look at B let us say B is coming like this right

So let me give an idea so A is coming something like this is A let us suppose B is coming something like this right this is B right which means that say at equals to 0 this is T = 0 this is T = 1 let us suppose this is T = 2 right and this so on and hence so forth let us look at a combinational logical block and see what happens. Not at equals to 0 A was 0 B was equals to let us suppose 1 A was 0 and B was equals to 1.

If you could have been a PO combinational block I would have got an output depending on the type of combinational logical block which is available to me. Now what was happened is by the time and T = 1 is available to me I have output of this combinational logical block been again fed back here right by the time next train of pulse comes and I have another pulse which was actually generated from the T – 1th part so if this is happening at tn then the feedback loop was available at tn-1th time that will start influencing the output of the combinational logical block such a design is therefore known as sequential logical block.

And the best example therefore is that means you are able to store a data for at least 1 period of time in your feedback loop between tn- 1 and tn this feedback path is able to store the output data right. Therefore all your memory elements are primarily based on sequential logical right and all your ordinary combinational logical blocks for example NAND gate, NOR gate, ZOR gate and or inverter simple inverters using all these technologies you do that fine I think I made myself clear what is the difference between the combinational logic block and what is the difference between the combinational and sequential logical block.

(Refer Slide Time: 12:31)

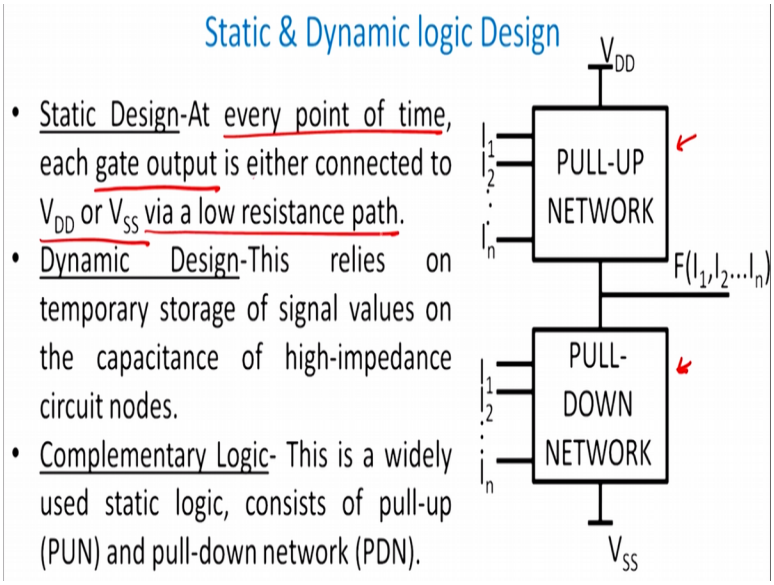


As you can see therefore you will always face very grave timing issues in case of sequential logical blocks but in case of combinational logical blocks you would not get a large amount of timing issues because it only depends so give me any time I can tell you what is the value depending upon the input but in the second case sequential give me anytime I can tell you the

value of course but then I should know what was the value of the output at the previous instance for which the data is available to you for the feedback path.

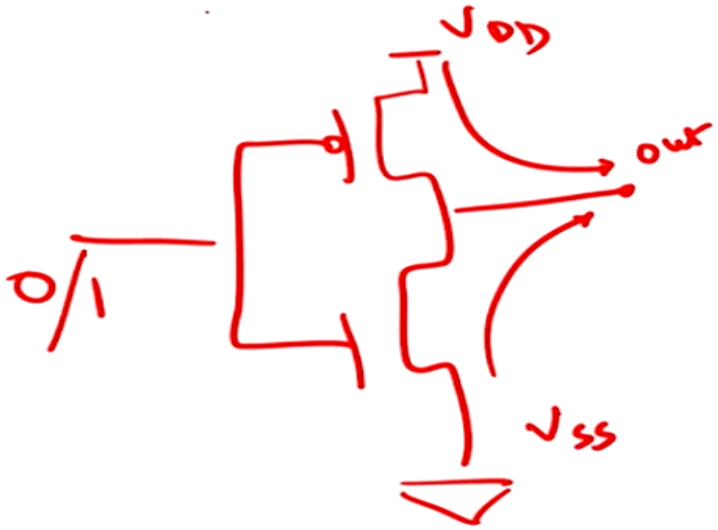
So this where is the difference between the combinational and sequential in this whole module next four modules possibly we will be only concentrating on combinational and then switch ourselves to sequential logical blocks as such let me therefore now discuss with you what is known as static design and we will also discuss about dynamic design complimentary design.

(Refer Slide Time: 13:20)



Static design as the name suggest that at any point of time right the output of the network right I have got a two networks here I have got a pull up network I have a pull down network right. Each output gate is either connected to VDD or VSS via load resistance path.

(Refer Slide Time: 13:43)



This is almost the same as we discussed in our earlier discussion that I have got a simple CMOS inverter right that you see when you apply a pulse here and you give say you apply 0 here PMOS is switched on and this VDD is connected to by out right so out and when you got 0 at this stage my ground is connected to out so this is VSS this also say this we say VDD.

So going by this discussion let us come back to the next stage that at every point of the time if the gate output, output which is available to you is either connected to VDD or VSS right via a low resistance path why no resistance because your PMOS and NMOS as been switched on and therefore it is defined as the low resistance path. If you would have been switched off obviously that would have been high resistance path or almost infinitely large resistance path but they are connected directly and that design is again known as the static design.

So for all practical purposes we will be using what is known as static design for this (()) (14:54) so we will be using two things one is combinational logical blocks and we will be using static design for our practical purposes. What is the dynamic design? Well dynamic design relies on temporary storage of signal values on the capacitance of high impedance nodes. What does it mean?

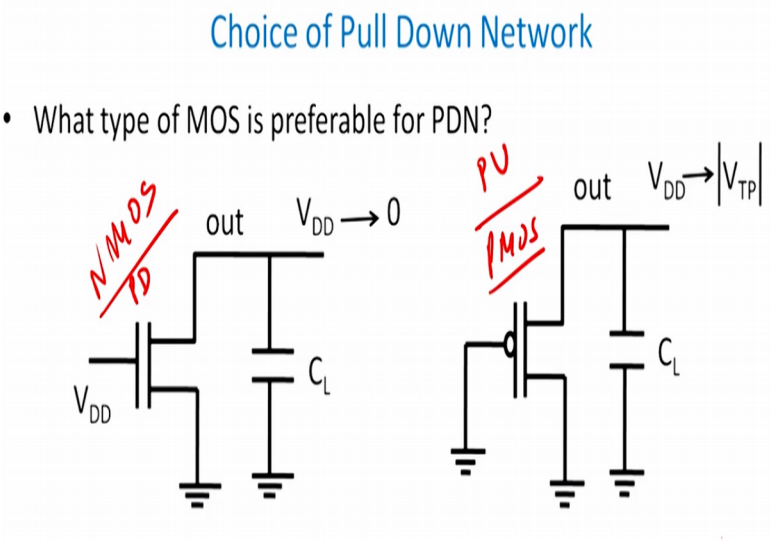
Depending upon the signal available to me if the output node of if the output node is connected to a capacitance of high impedance right for example that the storage value of the signal value on the capacitance of high impedance of circuit node which means that I have a high impedance

circuit node I have a capacitance there and it dependence upon how good or bad capacitor is able to store the value of charge at the particular point we define that the circuit to be a dynamic circuit right.

We come to what is know the complementary logic circuit? And this is also this is the most widely used static circuit. So complimentary logic circuit is part of static design only added is the mode widely used static so complementary logic circuit is part of the static design only added is the most widely used complementary why because this is of pull up network and this is pull down network.

So they are the complementary nature which means that if this is made up of PMOS this will be made of NMOS right and we will see later on that they are perfectly complementary in nature means any parallel network PMOS with a series network in NMOS and any series any parallel network in CMOS will be series network in PMOS. So there will be exactly complementary with respect to each other ensuring the fact very important ensuring the fact that there is always a low resistance path between the output and the ground or VDD right.

(Refer Slide Time: 16:39)



So please be please be cautious about it and understand this issue what idea is a what type of device we would like to use that is the pull down network or pull up network right. The most preferred device is basically your NMOS for pull down NOMS is used for pulling down you could also use PMOS to pull down it is generally used to pull up so I use NMOS for pull down I use PMOS for pull up right.

But the obvious question asked is can I not swap the places like can I make PMOS pull down and NMOS pull up right and this is where the whole issue comes into picture I will explain to you why we cannot do that right and we can do that but then we will comprise certain structure parameter or certain output characteristics of the PMOS inverter.

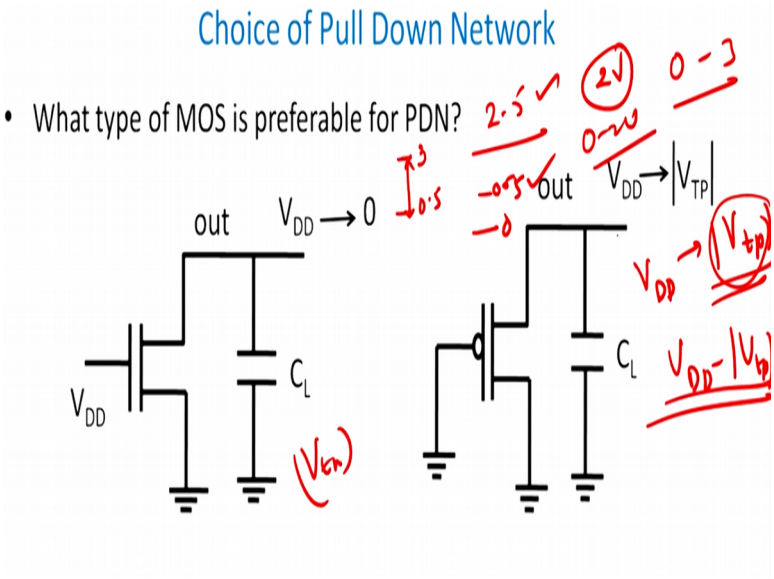
Please remember that at NMOS when you give gate voltage greater than the threshold voltage of the device then only it switches on enters into saturation beyond a particular gate voltage when VDSgreater than VGS – VTH and it starts to be a current source that is the time when it easily able to discharge all the charge stored in the capacitor to the ground. Before that please understand it was is resistive element and much before than as the deep try out almost in sub-threshold region on very small currents and therefore it takes definitely large amount of time.

So if you want that the pull down network should work very fast you should ensure that the device whether PMOS NMOS at this stage I am not telling you should be in either in saturated stage so that you are able to use as the current source and you can easily drop the current or charge the current from the CMOS from the capacitor on to the ground. So i will have resistive path to be more specific let me have a resistive path makes it easy to understand and therefore a charge can be discharge e to the power t by rc is the ah your delay time right.

But the problem in NMOS is that if you keep if you go on increasing your VGS right it becomes only on beyond VTH or VTN of the device right and you please understand that your output has to be lashed is only on beyond VTH or VTN of the device right and you please understand that your output has to be lashed to 0 almost close to 0 to get the whole swing from VDD to 0 right if let us suppose NMOS was pull up then it is drain is always connected to your output and source connected to VDD.

So your VDD S values are relatively small and therefore it will not allow the NMOS enter into resisting even resistive region or saturation region and therefore it is never be on properly and you would not be able charge the capacitor very easily right.

(Refer Slide Time: 19:33)



Second thing is in a PMOS case VDD get only be lashed to a value = mode of VTP so if it is a PMOS the maximum it can go up to is VTN the lowest value and here it can go up to mode of VTP. So since I can lower the value of VDD I get VDD – mode of VTP right and therefore I can easily reach to the value of VDD. So the swing available to me are very large if your NMOS is your pull down and PMOS is your pull up right and this helps as lot in terms of the choice which you have got pulled up right.

I will recommend that you have spice simulation to see whether what is happening so if you have a NMOS in the pull up we will see that your are never able to pull up beyond your particular point and it will be much lower than VDD similarly we have PMOS in the pull down network you would not be able to pull down to 0 there will be always a mode of VTP drop this mode of VTP might be actually higher than 0 so your swings are restricted.

So your restricted why? Because in any case VDD you drop down to VDD – VTN here and let us suppose this is 3 volt this is 0.5 so 2.5 you go in the pull up network similarly you are using PMOS and pull down and your VTP is 0.5 then rather than 0 you end up as 0.5 so what is the swing available from 2.5 to 0.5 from 2.5 to 0.5 this two volts you wanted an idea to be 0 to 3 volts you end up between 0 to 2 volts.

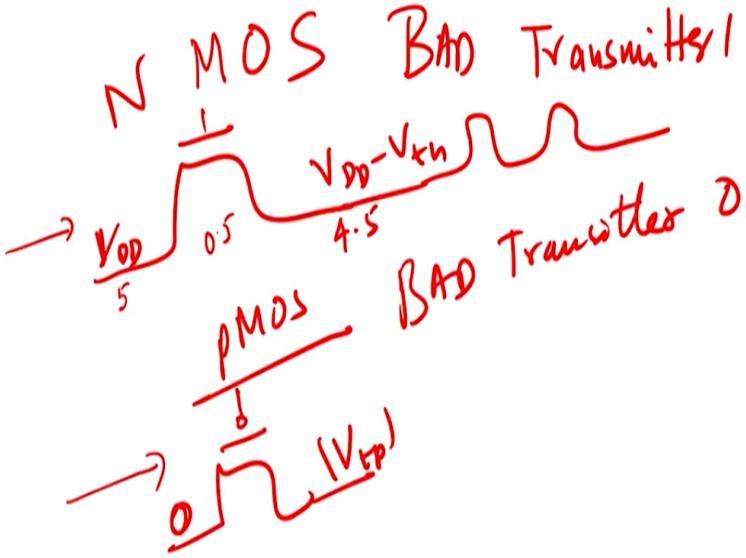
So your string as got restricted right and this might effect your noise margins at a later stage right so always advisable your would have NMOS in the pull down and PMOS in the pull up pull up

case right. As I discussed with you that why MOS is so what type of MOS is preferred for full for pull up network this my pull up network right this is the pull up network as I discussed with you that if this has been a pull up network it would have been connected to VDD here right and this rather than 0 it will actually look at VDD – VTN and that is the problem area in this case right.

And whereas this will allow you to go from 0 to VDD why because why I allows you to go to VDD – VTN because beyond this point as I discussed with you that therefore VDD – VDD – VTN is basically your VDS right. So your VDS will be how much is nothing but VTN right please remember for device to be saturated my idea was VDS should be greater than equal to VGS – VTN right.

Your VTS is always = VTN right so VGS – so which means that something like this VTN if you put VGS use to VTN let me suppose equals to less than equals to VGS – VTN right which means that I am restricting the swings available to me and I am only restricting it by a value = VDD – VTN. So please understand two important properties that NMOS therefore is a bad transmitter of 1 so I will just put very important two important statements.

(Refer Slide Time: 22:42)



NMOS right is a bad transmitter of 1 which means if you have NMOS here and you give 1 here you get 1 means VDD then you will get VDD – VTN at the output side right which means that if I use 5 volts here and if the threshold voltage is 0.5 I will get a 4.5 which means the signal as

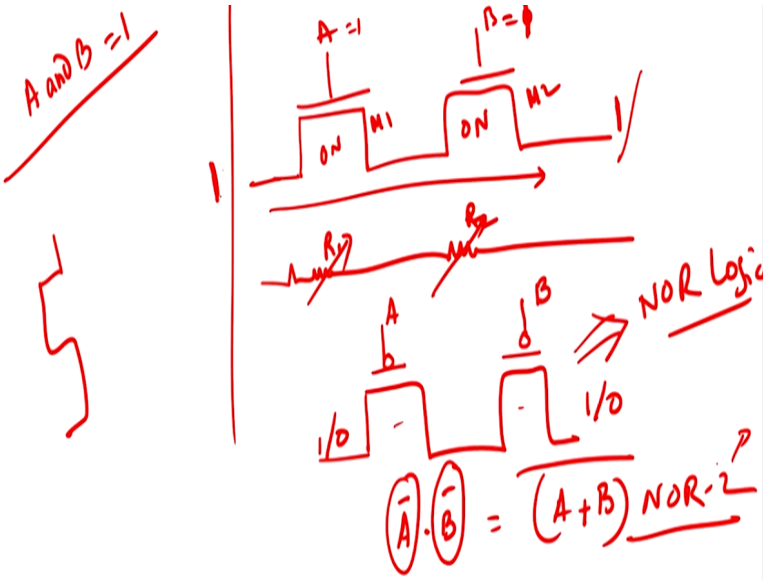
degraded right and therefore if there are multiple such type of transistors are available the signal will go on degrading as you move forward.

So NMOS is giving you a degrading effect in terms of potential right where as PMOS if you look it is a bad transmitter of 0. So if you want to transmit a PMOS right a PMOS and you want to have a 0 here then what you will get here is basically you will have a VTP mode available here they will always a mode of VTP available in the output side.

Which means that 0 will be not looked at 0 but mode of VTP so if you VTP is – 0.5 volts it will be looked as 0.5 volts available here. So NMOS is a bad transmitter of 1 and PMOS is a bad transmitter of 0 right. So this quiet interesting and will be very useful in logically understanding why NMOS should be pull down network and PMOS should be pull up network right therefore with this knowledge or with this idea of pull up and pull down network .

We come to the next preferred choice so we have therefore fixed up the it is always a good additive in the NMOS in the pull up in the pull down sorry pull down case and PMOS in the pull up because NMOS will let you go down upto 0 I PMOS will let you go up to VDD right so give the maximum available in this case let us therefore now we enter into the combinational logical understanding explanation and let us see how it works out in this case.

(Refer Slide Time: 24:44)



I will give you a brief example here let me say I am got two so what we are trying to do is this is the CMOS technology and basic NMOS technology can we realize the gates basic gates that is the only motivation of this module as far as understanding this module is concerned. So let me say I have got two gates here these are A and B right I give a 1 here right for this two assuming that there is no voltage drop which you have threshold voltage drop I can see this one available here right only when both A and B are both equals to one.

I hope you understand why because when both are equals to one both A = 1 and B = 1 = 1 this is on this is on and therefore this is a low resistance path and this 1 can be transmitted to this place very easily then what is the this thing and then this is this so I have is R1 this is R2 this is M1 this is M2. R1 is the on resistance of transistor M1 and M2 is the on resistance of on resistance of M2 and therefore I get easily see it.

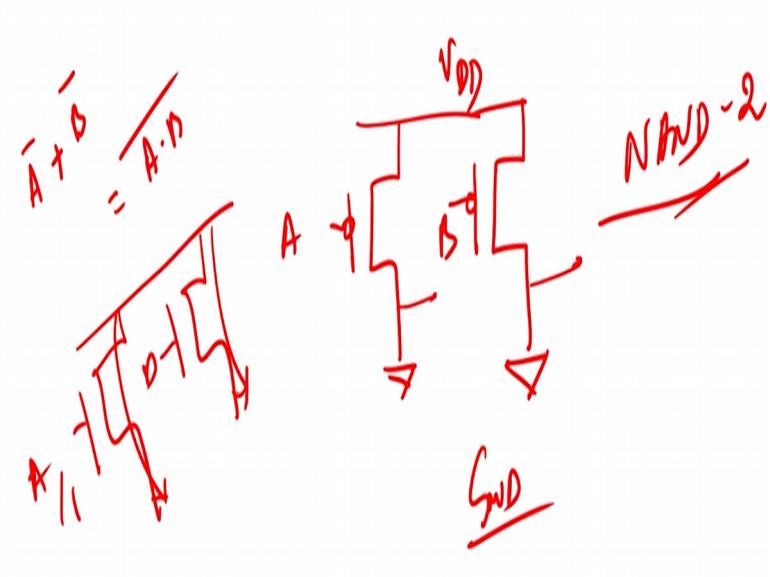
If either of it 0 A or B either of them is 0 then this is cut of or this is cut of and 1 will not be able to achieve here so either have the previous value or it will be a floating node right. So I can therefore achieve a simple AND gate by having two transistors in series or cascaded manner in this manner right two transistors in cascaded manner there is a very good way of looking at it. Now I will give you another example let me say I have this transistor and I have got two PMOS actually right and I wanted to see what is the output here and I give A here and I give B here right.

Now the device to be at ON state right I will like this to be a A bar dot B bar so A and B are both low then only I will have a 1 going from 1 here and 0 going to 0 here . But A or B can be written has A + B like this Dbogan's law. So therefore this starts to behave like a nor logic but please understand this is not a CMOS implementation this is simple implementation using static logic where you have NMOS and PMOS we will come to that after this slide.

Similarly you can have two PMOS's in parallel right I will explain to you so I hope I was able to explain it to you this one and I have got 2 PMOS's is series if you got A and B then I very know that for PMOS to switch on A should be low so I define the signal A to B as A bar which is low and B to B as B bar but both should be together low in order to switch it on I get A bar dot B bar once I do that De Morgan's law I get A or B whole bar.

So this is basically your NOR 2 basically means NOR input 2, 2 input nor gate fan in = 2 let me see therefore if you understood this basic concept of combinational logic.

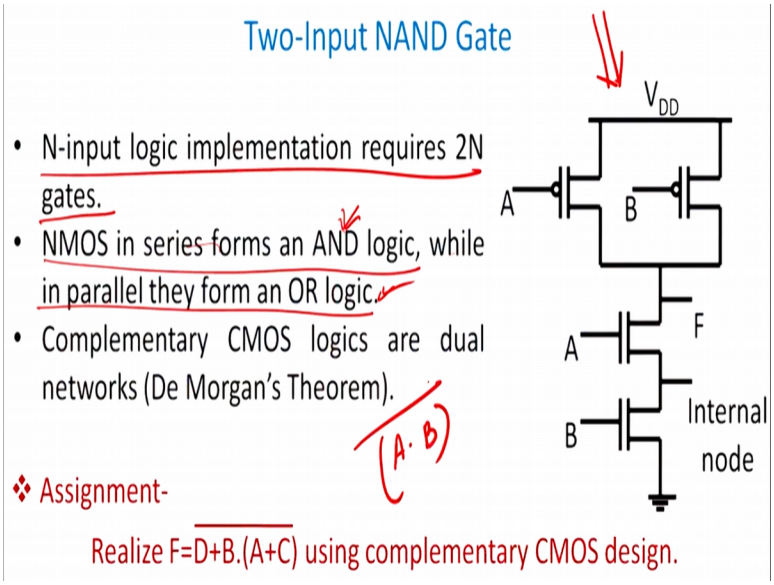
(Refer Slide Time: 28:01)



Then can we have something like this I will have two PMOS's right and this is sorry this is A and this is B and this is VDD and it goes to ground here then if this two PMOS's are in series for this two work as a gate and I want this to be appearing in output side on the output side I get very easily same I can easily understand that this will be A bar or B bar either of the two is on right if this is the output here available to me either the two is high I will get as a high and therefore this is nothing but A dot B bar by De Morgan's law and therefore this access as NAND gate.

So this is two input NAND gate right again let us CMOS implementation because there is no one NMOS appear available to you with this knowledge or with this idea let me give you a CMOS implementation of the switch.

(Refer Slide Time: 29:14)

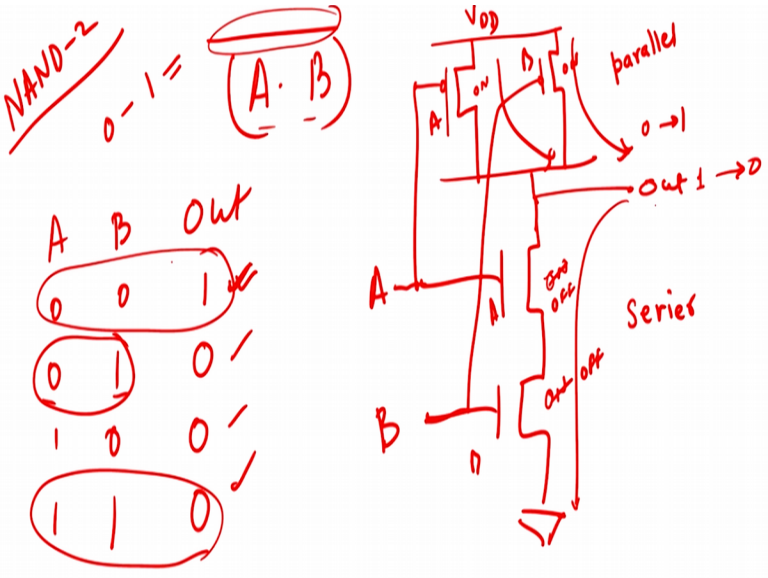


So what we have seen that NMOS therefore if it is a input logic implementation in complementary form it require to N gates to do it now NMOS in series is an AND logic while in parallel they form R logic I think again I made it clear to you why NMOS is in parallel I think it will be clear so if you have got NMOS in parallel right and if this is A and this is B right if either of them is A or B either of them is 1 either of them is 1.

This circuit is closed right and automatically it gets switched on element available to you 2 NMOS is in parallel therefore two NMOS is in parallel forms an OR logic and two NMOS is forms an AND logic right and complementary logics are dual of each other that I will explain what do I mean by that. This what you see in the right hand side this is basically a two implementation of two input NAND gate in CMOS logic what is the concept here?

Concept here is that NAN effectively means that I will have A dot B right and you have something like this and how do you generate it and how do you generate the right hand side diagram which you in front of you which is the diagram which you see in front of you how to generate it form the logic.

(Refer Slide Time: 30:33)



So what is the logic given to you is A dot B this which means that I have to make it two input NAND gate right let us suppose I have to make a AND gate finally that is not a big problem because then you have to put a static invertor of this to get it but let us suppose my job is to get a NAND to logic. So is using CMOS how will I make it right so I get A dot B bar so what I try to do is very simple is forget about the bar at this stage so just forget about bar and concentrate on A dot B I just now discussed with you that I can have a AND gate by simply having NMOS's in series agreed.

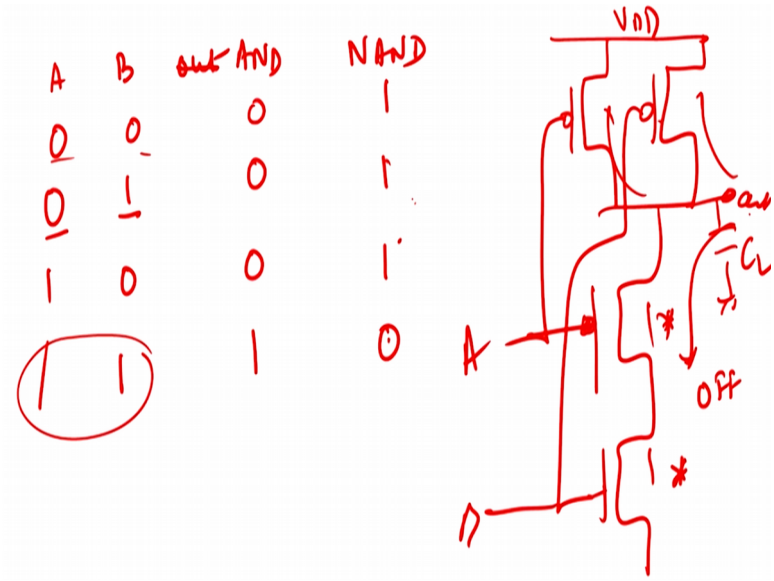
If this is so let me just NMOS's in series right this is A and this is B right and this is my output which is this is my output so I will just make it out I do not have to like this one. So I what do here is I try to make this one as the out capacitance out now the trick here is that you have to meet the PMOS exactly complementary of the NMOS so means that if there are two NMOS's in series here I have to make NMOS's in parallel here right.

So let me make that and let me see how it works out this is my VDD and so what I do I take one here I take another here and then I just take it like this right I connect to what I connect this to A or maybe I can connect this to A right and this to B is it okay. So I have got A here B here and I have got A here B here let us look at functionality now. So two table is what A, B and out I got 0, 0, 1, 1, 0, 1, 0, 1 and therefore when 0, 0 both are low and will be low and therefore out will be 0.

0, 1 output will be 0 bar will be this this this so I have 1 0, 0, 0 right let us look at it if you are both A and B are equals to 1 then this is on this is on and therefore output has got a load is resistive path to ground and therefore initially 1 was stored goes to 0 let us see I go to 0 here see this path is sufficient. Let us come to this path both are 0, 0 if both are 0, 0 then A at this will be on this will be on this will be off and this will be off and therefore this out gets charge from both the paths and therefore initially if it is 0 it is store 1 here.

And therefore our logic is fulfilled here again if you look at 0 and 1 here output will be again = 1 and so on and hence so forth you can actually have a look into it 0 and 1 will give you it is an AND gate basically so will give you output I am sorry there is a mistake here just make some small correction I am sorry I did a mistake here and I will just erase it.

(Refer Slide Time: 34:07)



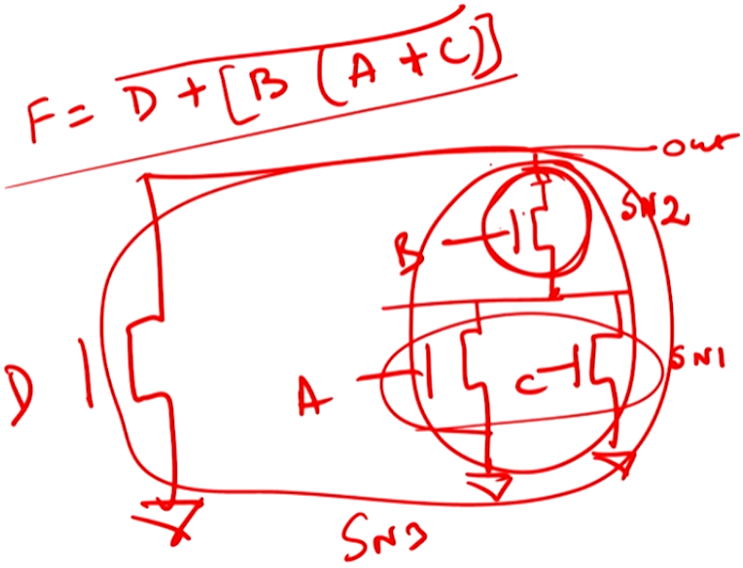
And I will just try to make a A, B out and I will make 0, 0, 1, 1, 0, 1, 0, 1 if this is a AND gate output right and this is AND gate output right I will just make it 0, 0, I can get 0 I get 0 I get 1 therefore 1, 1, 1, 0 will be there I am sorry I made a wrong mistake, I did a mistake in the previous slide it is my fault and I will explain to you from that angle again. That I have got this into consideration I have got sorry this is my sorry this is my NMOS and this is my PMOS so I have A here and I have got A here and I have got B here and this is my B here and this is my output.

So let us fulfill this one that for 1, 1 you will get 0 that is have already explain in 1, 1 output will always got to 0 let us look at 0, 0 if it is 0, 0 output will go to 1 because I have got two path true to be charged the CL will charge here right CL will charge here. If it is 1, 0 or 0, 1 then what I get I get 1 and this is true also if I get 0, 1 then you see either of these two NMOS's either this as NMOS or this NMOS will be in off state and therefore it will not allow you to have a low resistance path between output and the ground.

So what do I do and if it is 0, 1 or 1, 0 I will have at least 1 path available to me from VDD to out available and therefore that will give me 1. So I have made therefore clear that how do you actually how do you actually make it how do I actually make it. You try to first of all make pull down network depending upon the logic given to you and your pull out network will be perfectly complementary pull down network.

Well that is the major issue or the major output issue here so I will just show to you what I mean to say that. So as you can see that you are supposed to do this right let me realize the function which is given here in this example I will use and example here.

(Refer Slide Time: 36:17)



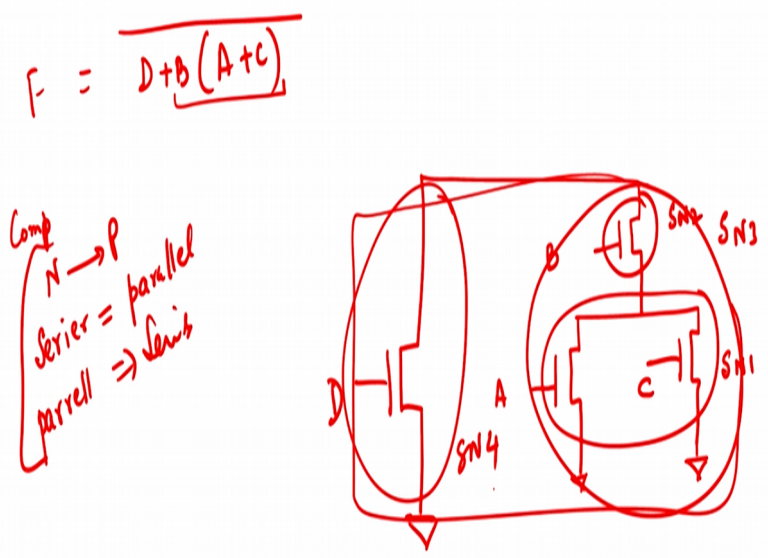
And it is F = right = your D + B right so I am sorry D + B into A right A + B is AC A +C let us suppose right let us suppose I have to realize this function the methodology people adopt is that first design the pull down network A + C primarily meaning that I have NMOS's and there should be in parallel that is very because there should be parallel to each other. So what I do

make two NMOS's right and make it parallel let us suppose then my D is in D is B is in series to it.

So what I do I make D in series to it so this is my D B this is A and this is C and D will be parallel to whom to whole of it the whole of it this hole so I make a D value here and this is D you got the point that this is D and this is so this is A + C A or C which is ending with B and therefore in series with B and which is in turn or in with D and therefore D is coming here therefore is the output here we defined these as sub next or we define these small once sub next.

So when you got two parallel and name it as sub net 1 right I define it sub net 1 then I define this to be as sub net 2 right and then I define this as whole block as sub net 3. So I have two elements sub net 1 in parallel I can use it as single device and then that is in series to sub net 2 right so this is let as sub net 2 make it more clear to you so let me draw for you the whole thing once again it will become clear to you how to make it therefore?

(Refer Slide Time: 38:23)

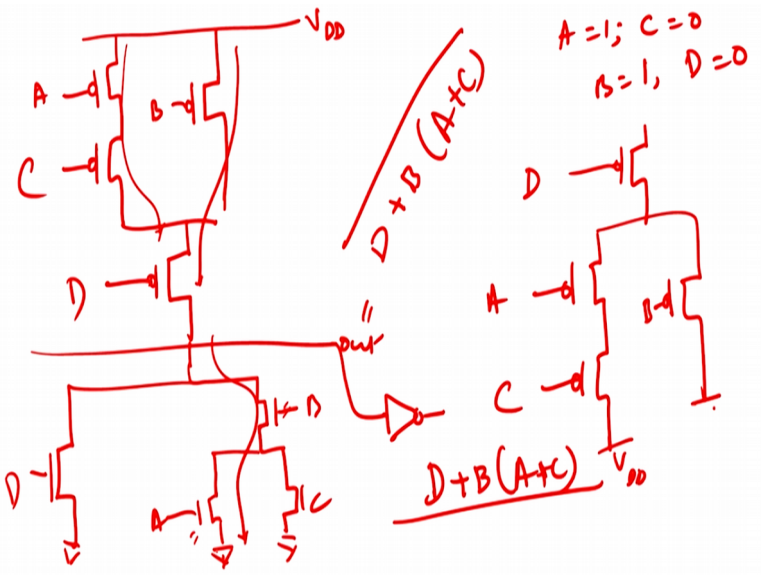


That I have got D + B into A + C NAND is my F right so what I do initially I have got A and C in parallel so A and I have got C in parallel to each other right connect this together I have B in series. So B is in series to each other so I have B here I have A here and I have C here right I also have a D which is actually Oring this whole quantity So what I do I take up D here and draw (()) Oring here I take D here.

Now you start defining the because now please understand I have to compliment this in the pull up state right I have to complement this in pull up state which basically means that I have to use all parallel to series and all series to parallel and all NMOS to CMOS. So when I say complementary I mean to say two things NMOS to PMOS all series combinations to parallel combinations parallel and all parallel combinations to series combinations. So these three activities you to together to do that we define individual networks we define this to be as sub net 1 we define this to be as sub net 2.

So when it goes to the pull up device they can convert into PMOS but in subnet 1 and subnet 2 will be in parallel to each other are you getting up my point? This whole this is subnet 3 and then this + this is basically your subnet 4 right so let me say this as subnet 4 explanation. So subnet 4 is parallel to subnet 3 but when you go to complementary it will be series to each other right. So let us see how to how do you how do you get a help and do it. So this is for NMOS right pull down and this basically a pull down network let me do the pull up network and do that same thing. So B was if you go back it was A + C.

(Refer Slide Time: 40:41)



So let me just start subnet 1 and A and C will be series to each other so A and series A and series C to each other this will be this will be what it was initially in series with subnet 2 which is nothing but B therefore it will be subnet 2 and therefore I will get B here this is for PMOS B here and then what was happening D was will be so if you look back here D was in parallel to sub net 3 and therefore D will be series to subnet 3 hence so forth I get D here signals here fine.

You just flip it then you automatically gets this which means that you connect this to VDD this you connect to VDD and you connect to I just make it for you the whole this thing I have got VDD here and I have got A here and then C here right and I have got this this then I get B here I get B here which is this this is B this is A this is C and then series to it you have D right this is D and then this is my output here out here and then is start downwards.

So downwards what was the clear condition I had B here right and then I have got if you go back to previous this thing A and C here so I have A here right I have C here right and both are grounded C here and there will be a D which is basically in parallel to all of them this is D. If you look now this out will be defined as whatever we are written in the previous state that it will be it will be D + B, A+ C NAND who is doing this NAND principle buy the very virtual to be complementary right.

So therefore if you want to come back to signal similarly simply put this connect to a single inverter and you automatically get D + B A + C. So what we have understood in this presentation is given any complicated Boolean expression you can realize that Boolean expression on to a platform based on pure CMOS complementary logic principles right. This will allow you to give a pull down and pull up path now you I leave us an exercise tomorrow we will see or that you will only for a set of inputs A, B, C, D you will get only one path between either ground and VDD or between ground and between VDD and output and output and ground.

They cannot be multiple paths of available right take any take any case you say A = 1 and C = 0 let us suppose and B = 1 you take an D = 0 you take. Let us suppose you take this then you take when D say when you take D= 0 and you take B = 1 this path is blocked because B = 0 will not allow you to do that A = 1 means this path is blocked so your pull up is blocked let us go to pull down with A = 1 B = 1 this part is open to you right.

So V out is connected to V as in easy fashion similarly you change any input and you will never get a direct input between VDD and ground but you will always get a low resistance path from where from output to either VDD or output to either ground fine. So this is the overall scheme of things which I want to do talk here. So today in this 50 minutes of lecture we have understood

the basic concept of complementary MOS logic and how to design the complementary loss logic from subnet principles.

So I leave as an excise to you to take up any Boolean expression logic and try to implement what we discussed today right. So when we start next time we will start from this basic concept and then we will see how to move about and generate much more complicated combination logical blocks with this I thank you for our patience really.