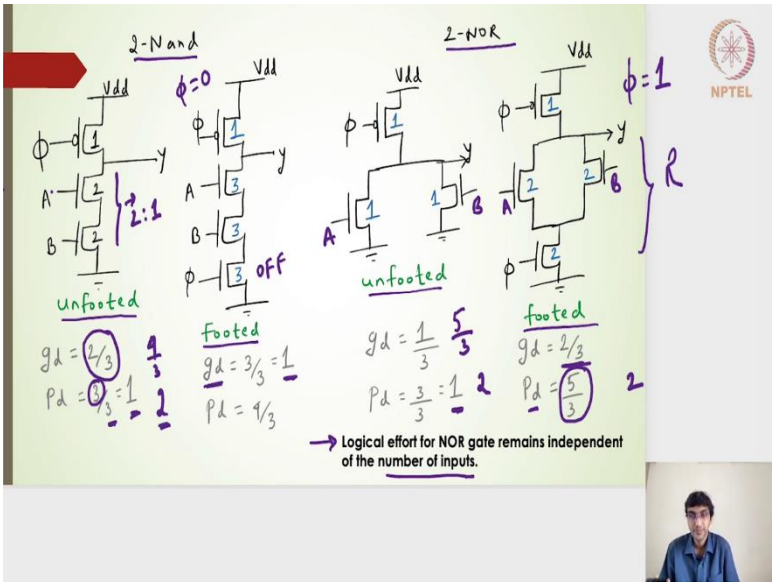
Design and Analysis of VLSI Subsystems
Dr. Madhav Rao
Department of Electronics and Communication Engineering
Indian Institute of Information Technology, Bangalore

Lecture - 47
Dynamic Logic and Domino logic

(Refer Slide Time: 00:16)



Hello students. Welcome back to this lecture on the dynamic circuit families. Here I have drawn different types of combinational gates, one has the 2-input NAND gate and another one is the 2-input NOR gate, on the other side I have drawn the 2 input NOR gates. This is a NAND and this is a NOR and here in the individual parts or the individual components I have got the un footed ones and then the footed ones.

Again, similarly the unfooted, footed NOR gates have been shown the schematic of that is shown. Let us go all one by one 2-input NAND gates for an unfooted dynamic circuit will be shown here with a clock PMOS transistor and the size of 1 here and then the A and B inputs are connected to the NMOS transistors here with the size of 2 there is no foot gate or there is no foot transistor.

My logical effort for going down will be nothing but 2 and this one will see a benchmark inverter of 2:1. My logical effort will be,

$$g_d = 2/3$$

Parasitic will be nothing but,

$$P_d = \frac{3}{3} = 1$$

If I look into the logical effort for the going down and compare with that of the regular CMOS family or the regular CMOS structure to input NAND gate, it turns out to have the logical effort is nothing but $\frac{4}{3} > \frac{2}{3}$.

This $\frac{2}{3}$ turns out to be better 1 and then the parasitic for the 2-input NAND gate was actually 2 in a regular CMOS structure, 2 input NAND gates here it is actually 1. Again a very very beneficial and much more faster circuits, but if there is a contention here the unfooted one always y has a probability of getting a contention between the pull-up and the pull-down circuit.

For example, if $\phi = 0$ during the precharge mode this PMOS will be on and for some reason A and B will be 1. Then my NMOS will also be on. I will have a current on the pull down side I will have a current on the pull up side thereby that contention will be there and that is the reason why we have a foot transistor.

Now the foot transistor is there which is the transistor closer to the ground rate and whenever the $\phi = 0$ this will be off. That contention or the path to the ground by the pull down circuit will not be complete, because even when $\phi = 0$, the pre charge mode this transistors will be off and that is the reason why the discharge path is cut off.

The output can only charge to V_{dd}, it eliminates the contention problem, but because it acts the transistor my width has to be different now. It has to be a size of 3 and 3 and 3 and thereby my logical effort and then the parasitic will go up. If I consider the input A here the logical effort will be the capacitance,

$$g_d = \frac{3}{3} = 1$$

Its still better than the $\frac{4}{3}$ regular CMOS structure 2-input NAND gates. The parasitic now will be nothing but,

$$p_d = \frac{4}{3}$$

Again its much much less than that of the value of 2, what we had seen for a regular CMOS structure 2-input NAND gate. 2-input NOR gate again on an unfooted side I will have 1 PMOS transistor which is nothing but connected to the clock signal and then the A and B inputs connected to the individual branches NMOS transistors.

For an unfooted one, the logical effort will be nothing but,

$$g_d = \frac{1}{3}$$

Its very very less than that of the $\frac{5}{3}$ what we had seen. For a regular CMOS structured 2-input NOR gate and the parasitic will be nothing but,

$$P_d = \frac{3}{3} = 1$$

Again, the CMOS structured 2-input NOR gate is nothing but gives a parasitic of 2 whereas, here the dynamic logic families gives a parasitic of 1. On the footed side just to avoid the contention we have one more transistor here the foot transistor with the size of 2 and A and B are transistors on the pull down side has a size of 2 and 2.

Overall, the switching resistance will be nothing but R because if either one of them is 1 the output will be discharge 2 ground in the evaluation phase. In the evaluation phase means when the $\phi = 1$ for the footed 2-input NOR gate the logical effort will be different than that of the unfooted one because there is an addition of one more foot transistor.

The logical effort will be nothing but the input capacitance of either of these inputs will be nothing but,

$$g_d = \frac{2}{3}$$

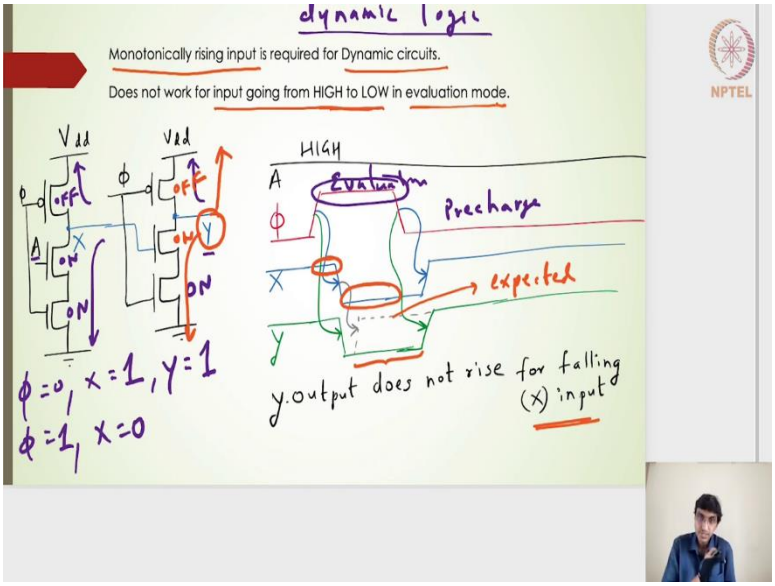
Again, it is much much better than that of the $\frac{5}{3}$ which we had seen earlier and the parasitic will be nothing but,

$$P_d = \frac{5}{3}$$

The parasitic seen by the regular CMOS structure 2-input NOR gate is 2, this $\frac{5}{3}$ is also better than that of the parasitic of the regular 2-input NOR gate. Note that the logical effort for the 2-input NOR gate or an n input NOR gate remains independent of the number of inputs. The logical effort is actually dependent on the, you know we take we consider one of the inputs. So, it really does not depend on how many number of the inputs are there. The logical effort for the NOR gate in that sense it does not really depend on the number of the inputs.

Whereas for the NAND gate if I keep on adding more transistors because to accommodate more inputs, the sizes of the transistor sizes will change and thereby the input capacitance scene will be changed and thereby the logical effort will change. Whereas in a 2 input NOR gate that is not the case the logical effort remains independent. Hope this is clear moving ahead.

(Refer Slide Time: 06:58)



There is one big disadvantage with the dynamic logic families and that is what is stated here. Let us take an example of dynamic inverter followed by another dynamic inverter. What we really want is an input A whatever is the input A either 0 or 1. The X will be the complement of that and Y will again be the complement of X and thereby I should see an output of the Y to be similar to the input of A at least during the evaluation phase.

Because in the evaluation phase that is where the computation happens that is when the output is been evaluated. I should get in the evaluation fed in the evaluation phase I should expect X to be a complement of Y and then Y should be a complement of X. Let us take a very simple case A being high, what we expect is X will go down to 0 and then Y will rise up from 0 to 1 and that is why I should expect Y to be very very similar to that of the A during the evaluation phase.

Let us say that the $\phi = 0$ and then it goes to the evaluation phase. $\phi = 0$ and then it goes to 1. when $\phi = 0$, X is anyways charged to V_{dd}. Y will also be charged to V_{dd}. When $\phi = 0$, I will say that X is has reached 1, y has reached 1. What it means is X is precharged to V_{dd} and Y is precharged to V_{dd} or logic level 1.

When $\phi = 0$, X is high and Y = high. Now, let us say that when ϕ goes to 1, that is, when this particular graph when $\phi = 1$ alright this is where the evaluation happens and this particular phase is called the evaluation phase because A is high always X will go down A is high and ϕ = high.

This will be on and then this will be on and X will now and then of course, this will be. What happens is in when $\phi = 1$ this will be off and then this will be on and then this is $\phi = 1$. This particular foot transistor will also be on and then the X = 0, that is what I have drawn here the X = 0.

But there will be some kind of a delay for X to go to 0 and that is what I have drawn in this particular transient graph. I have an X value which will go to 0, it stays high and then it goes to 0 and that is what I have drawn in the blue line here indicating that the moment ϕ goes to high, the X will take some time some nonzero time and then it will go to 0 and then it continues to stay at 0 till the evaluation phase is there because that is when A is continuously high during the evaluation phase and that is why the X is at 0 and then during the pre-charged mode this particular thing we will call it as a pre-charged and when $\phi = 1$, this one we can call it as the evaluation phase.

In the pre charged mode the X will come back to 1 again with some kind of a delay. The delay representation in this particular timing graph I have represented in the form of an arrows. The blue arrows for the X output, now X goes to 0 what really happens to Y. When $\phi = 0$ the Y has been pre-charged to 1, this particular portion.

But, at the moment $\phi = 1$, that means $\phi = 1$ represents this is on and X is still high in this particular case. If I consider this particular portion of X it is still high. X stays high for some time and then it goes to 0, because of the delay for the X to discharge in this particular case, X stays high for some time and then it goes to 0.

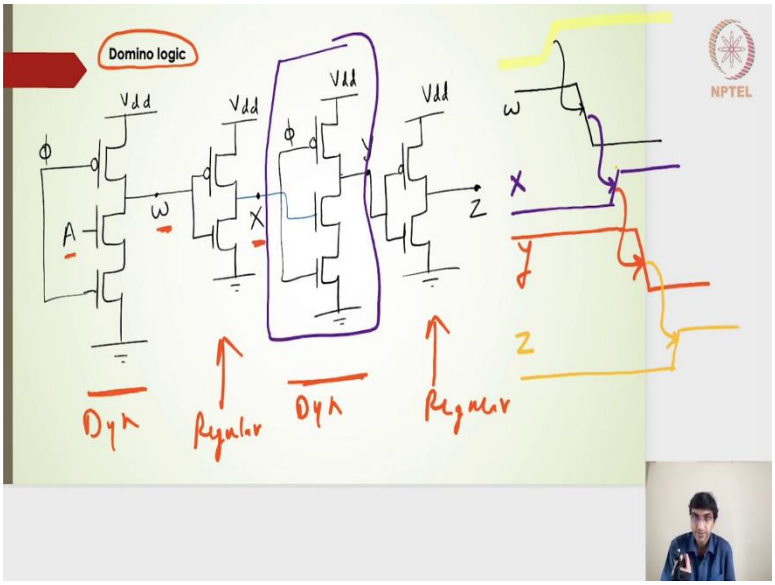
During that particular portion when X = high this transistor will be on and Y is going to discharge now as well, Y actually discharges to 0. The moment X hits 0, Y was supposed to go high because that is what we expect, but once the Y is completely discharged and then this particular transistor is off during the evaluation phase, there is no path for Y to go to V_{dd} or go to the logic level high. Because this transistor is off during the $\phi = 1$ during the evaluation phase. Y does not have any path for charging and it has completely discharged because it sees X for a small duration the X is high. I have drawn this dotted line representing that the dotted line is our expected profile, but what really happens is once the Y comes down there is no charging part.

Y stays at 0, the Y output does not rise for the falling X input, whenever the X falls. Falling during this particular X the Y does not rise, this is the real problem for the dynamic logic or the dynamic circuits. What I have said here is for a monotonically rising input it is required. The dynamic circuit's works only for the monotonically rising input.

In this case for X if it had been the opposite case the X was low and then it would have gone high then the Y would have worked very well. Then the Y will stay at the logic level or it would have come down to 0, because coming down from 0 it just depends on this particular X input and ϕ is anyways 1, it can easily come down.

It works for the monotonically rising input. It does not work for the input going from high to low. If the input is high for some duration and then it goes slow, during this particular duration when it is high almost all the dynamic logic circuit families it will go to 0 and once it hits 0 it does not know how to come back to 1. For a transition from high to low in the evaluation mode it does not work. It always needs a monotonic output monotonic input and that should be a rising input, low in the evaluation phase and then in the evaluation phase it should go high. It does not work the opposite way, very big disadvantage for this particular circuit.

(Refer Slide Time: 14:29)



That is why another for family of the circuit is been has been deduced that is called as a dynamic Domino logic. A Domino logic it is basically nothing but one can see that as a representation of both a kind of a mixed back. The dynamic logic family and in between there is the regular CMOS family structure. It is basically a mix of both and that is how the domino logic is formed. In this particular case you have seen that these are the 2 dynamic logic and in fact 2 dynamic inverter circuits and in between the regular inverter is been inserted. What this particular regular inverter does is whenever the input goes from high to low or low to high, we will get this output is fed to the regular inverter structure.

The regular inverter structure does not really depend on the evaluation phase or the precharged phase. It just does it gives me the output, which is a compliment of that of the w input right. In our previous case when X was actually going down in the evaluation phase Y was not able to come up, that is the case we are talking about.

In this particular case when w goes low in the evaluation phase. In the evaluation phase, hat I am going to do is I am going to draw the evaluation phase. This is my evaluation phase and then the w goes low it will take some time to go low and during this particular phase X output will take its own time of course and then it will be an inverted output, I will have a rising output.

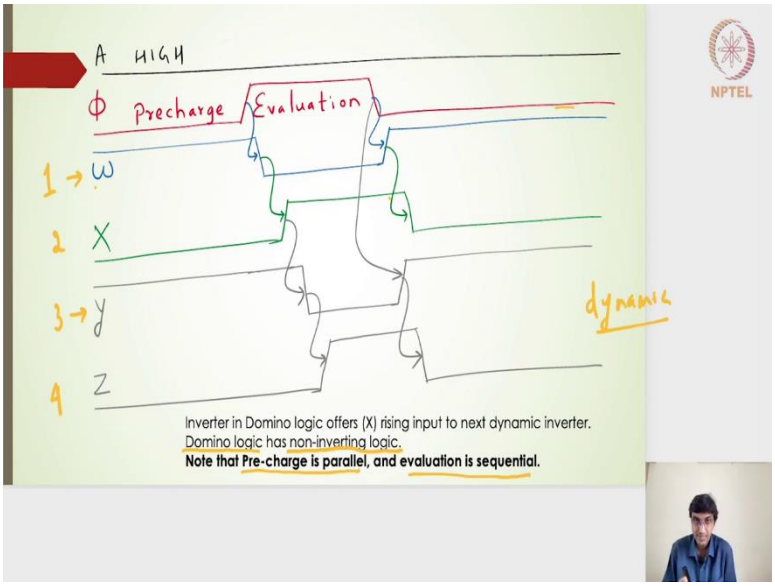
Now, this rising output is given to the next stage of the dynamic inverter and rising stage input is not at all a problem for the dynamic inverter only the falling output is a problem

the rising output is not at all a problem. In this particular case Y signal it is pre charged and then whenever the X stays at low it is Y will be connected to the V_{dd} level and then it is going to go low.

This will be my output of the Y and then finally Z will be nothing but it is a regular inverter again I will get low and then it is going high. What we are doing here is adding an inverter to ensure that, we always give a rising input to the dynamic inverter in this case or for any other dynamic logic circuits.

The inverter is added to feed always the rising input. The falling input is completely changed to a rising input by adding this particular inverter and that is why it is called as the domino logic because it contains both the dynamic logic as well as the regular inverter alright. Hope this is clear.

(Refer Slide Time: 18:13)



That is what I have drawn here the timing graph where A input is high it is the similar one, very very similar to that of what we had seen for the dynamic logic. Only thing is now I will have 4 additional signals w x y and z which is nothing but the output coming from the 1st dynamic inverter and then the 2nd inverter and then the 3rd stage dynamic inverter and then the 4th stage inverter.

I will have a pre-charge, precharge will ensure that w now pre-charge to V_{dd} and y is pre-charge to V_{dd} and x and the z are nothing but the inverted outputs of w and y respectively.

In the evaluation phase w goes down because input A is high. It goes down with some delay x will ensure that it goes high whenever the w goes low, because x is a rising input y will stay to V_{dd} and whenever it goes high it will drop to 0. Again, in the evaluation mode and z will be nothing but the complement of the y input and note that here once the evaluation phase is complete w comes back to 1 at the same time as that of the y comes back to 1, because it is a. The evaluation is fed parallelly rather sequentially the evaluation happens sequentially.

But the pre-charge is done at the same time. The precharge for all the dynamic circuits will be done at the same time because the clock is the same. I have written here the pre-charge is actually parallel and then the evaluation is kind of sequential. The evaluation sequential means based on A, w will go down and x will go high after some delay of w.

You know after some delay and then y will go down after some delay the moment the x goes high after some delay y will go down and z will go down after some delay from the y going down in that sense it is kind of very staggered or sequential in the computation. But the pre-charge, that means that the w and then the y output goes high simultaneously or in pattern.

The one another aspect about the dynamic inverter it is called as the dynamic inverter it is not called as a static inverter the reason is, it is dynamic in the sense that the clock is added is given to the PMOS transistor and the evaluation of the computation happens onto the NMOS side. That in that sense it is called as a dynamic inverter and not a regular inverter or a not a regular CMOS structured family of logic.

Hope you have understood this particular concept of starting with ganged CMOS family, where we saw 2 input NOR gates and then followed by the dynamic logic families and then finally we have arrived at the domino logic which prevents that problem of the dynamic inverters where the falling output is a problem for the dynamic logic families and thereby the falling output is converted into a rising output.

Note here that the domino logic is also applicable only for a non-inverting logic it does not work for the inverting logic. We always need even number of stages, the inverter is added in between. We always have the even number of stages and in this particular example we have seen for the non-inverting logic. It does not work for the inverting logic, that is

another disadvantage, but it avoids or eliminates that problem of the monotonically and the falling input. It converts that falling input into a rising input.