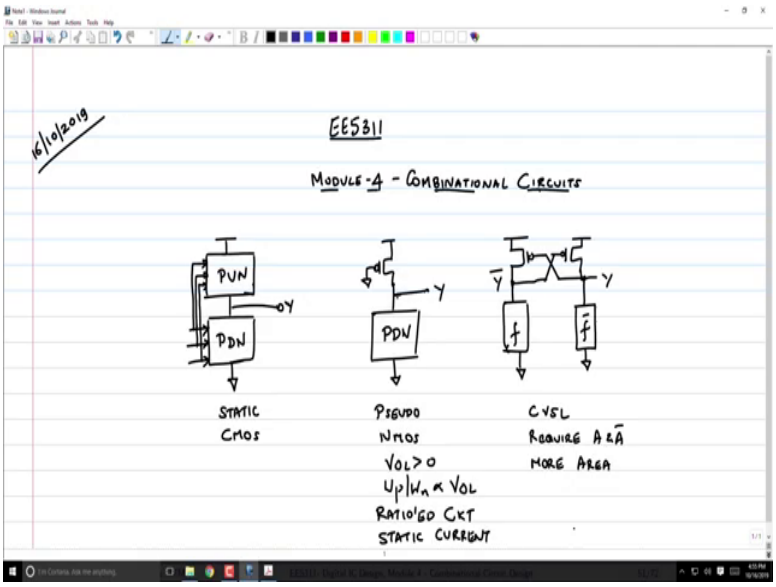
Digital IC Design
Prof. Janakiraman Viraraghavan
Department of Electrical Engineering
Indian Institute of Technology, Madras

Lecture – 50
Dynamic Circuits – Input Monotonicity

(Refer Slide Time: 00:14)



So, let us continue our discussion on reducing the logical efforts right, that is where we were; that is where we were last class and last two classes in fact. So, basically to summarize what we have discussed till now is a static CMOS logic is like this. There is a PMOS pull up network, there is an NMOS pull down network and right and the output is in between the two.

Then in order to reduce the logical effect of course, the same inputs go to both, right; this has very high logical effort because of the PMOS block. Therefore, we said let us take the PMOS

block off completely, keep a PMOS transistor which is on always; however, this has to be a weaker transistor than the NMOS right and you have only a pull down network, right this is my static CMOS.

In the today's class I will tell you why this is called static CMOS, right? Because we have something called dynamic logic later. So, this is a pull down network and beside that this is called Pseudo NMOS, because it is mainly NMOS logic; we have just one PMOS transistor.

PMOS is really not used to implement any logic function right; therefore, it is called Pseudo NMOS logic or Ratioed circuit. Ratioed circuit, because unless the ratio of W p by W n is correct this circuit cannot even function right. So, problem is VOL is greater than 0, right; W p by W n is or as proportional to VOL. So, this ratio is critical and therefore, it is called a Ratioed circuit, right.

And of course, there is also an other problem there is static current, when the pull down network is active; because the PMOS is always turned on, right. So, to in order to overcome this problem, we came up with the idea of implementing a cascode voltage switch logic.

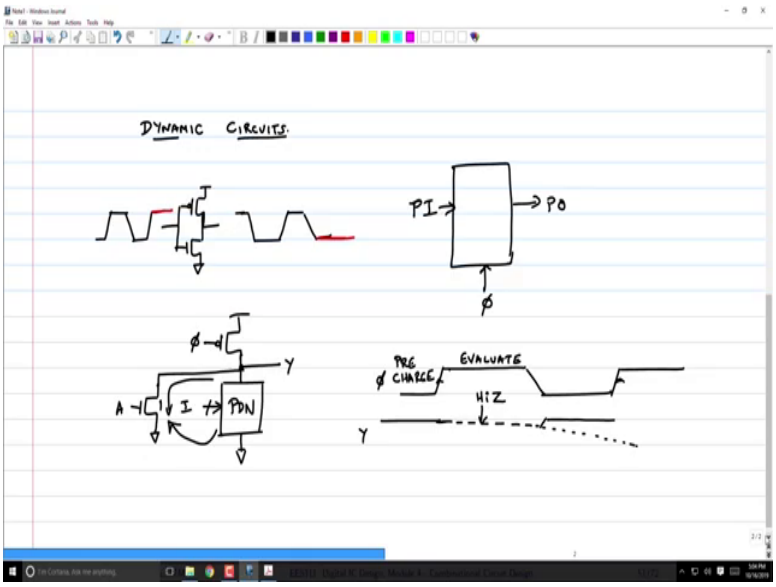
We just said that all you have to do is have two pull down networks. Of course, you have more area now; on one side you implement the function f, on the other side you implement the function f bar. Then the opposite sides of where you know the PMOS and NMOS are fighting will ensure that output is logic low and therefore, you just turn off the PMOS transistor and prevent any static current right, this is called CVSL logic, right. Problem is of course, you require.

Student: A and A bar.

Require A and A bar to be available and no doubt you have more area in general, ok. So, there are certain advantages in this and it is useful in certain places; but you are doing this at the penalty of throwing in more transistors and hence more area and also therefore, more power ok. So, you just have to keep this disadvantages in mind, you get y no y bar and y, ok.

So, what I really want to do now is, I want the benefits of you know this CVSL logic; but I still do not want put in so many transistors, ok.

(Refer Slide Time: 04:40)



It turns out that; that is possible by doing what is known as dynamic circuits. So, let us just quickly look at you know what is the philosophy behind the combinational circuit. Suppose I have an input making a transition here and then here and then here; the output will follow with some delay, it will follow exactly this, right. The point being that I can make a transition at any point in time and my static CMOS circuit will respond with some delay, right.

But most of the time the what we have is you have a system which is going to be, you have inputs; primary inputs, primary outputs and you are also going to drive it with the clock phi,

right. Then basically going to you going to have flip plops switch, you are going to sample the inputs, hold the input constant during the cloc.

Then you can evaluate and you have some implicit constraints are there on when the input can make transition so that the outputs will appear without any error. Implicitly these constraints are there in the system; so why not we exploit those constraints ok, that is the basic philosophy here.

The second thing is, if this input made a transition and this steady state value just remained forever ok; then this steady state value would also remain forever. As long as my power supply is connected, both of them will remain in that state forever, right.

These two conditions we are going to exploit and we are going to compromise, so that we get better logical effort; but do away with all those problems what we had in the Pseudo NMOS circuit or the CVSL logic, ok. So, what we do here is, instead of my Pseudo NMOS logic which was simply connected to ground right; the gate was connected to ground always, pull down network.

I say do not connect it to ground, but connect it to the clock phi, ok. I have a clock in the system anyway; I am going to connect it to that clock, right. So, what happens here, my clock is going to look like this ok; a realistic clock will have some finite rise time. What happens when my clock is 0? PMOS is turned on and therefore, the output is y will be logic high correct.

So, my y output is going to remain high here. Now let us assume that my pull down network never turns on, ok. The input combinations are such that they never turn on, ok. There is no transition happening by primary input, my inputs are here there here right, some different number of inputs and it turns out that the pull down network is never activated; the input never makes a transition. So, what will happen the moment my clock turns off or goes high, what will happen to the PMOS transistor?

Student: Off.

It is going to be off. So, what if the pull down network is never going to be turned on in that state, based on that assumption and the PMOS also now turns off, what state is my output in; high impedance or it is going to float. So, I will represent high impedance by a dotted line.

Now, question is will this be at V DD always? Clearly the answer is no, because now there is a leakage path through the pull down network right; let me replace this pull down network with something that we know, I am going to connect it to a single NMOS transistor A, right. I want to make an inverter, so this I will connect here. Let us say this is the pull down network. Dynamic inverter is what I am constructing, right. Clearly let me and what are the sizes now? NMOS will be, NMOS should be how much?

Student: Half.

Half why?

Student: (Refer Slide Time: 10:05).

No. So, let us assume that the minimum size is 1. So, it is 1. Similarly this PMOS size will fix later, ok. So, now, there is a leakage path; when I say output y is floating, there is a leakage path through this NMOS transistor at through the sub threshold leakage. And therefore, strictly this V DD will actually drop slowly with time; until the next clock comes, I mean the next falling gates comes and takes it back to V DD. So, here this will again go back to V DD right, this is the high impedance.

So, why is this circuit called a dynamic circuit; because when my clock goes high, that is when I am going to allow this circuit to evaluate the particular logic, ok. These two clock phases I want to call as precharge and evaluate. In the pre charge phase all the that the clock is doing is taking the output to one; in the evaluate phase the clock is turning off and it will

allow the pull down network to pull the logic low if necessary, depending on the input combination there, right.

So, the circuit instead of being a regular combinational circuit where it can evaluate in any phase of the clock; we are now putting the restriction that this circuit can work only in one phase of the clock, that is when the clock is high, ok. So, why is this is called dynamic now? Let us assume that I slow this clock down sufficiently, I make the clock period 1 Hertz right, one second I am going to you know my time period is 1 second.

Clearly in one second this output y, even though my input is 0 right my output is suppose to be one forever correct; but because the time period is so large, it can leak out and the output eventually can go below. If you just allow this to go on and on, it will eventually go below the threshold of the you know of a gate V t n and then it will effectively become logic 0.

So, it is called the dynamic circuit; because the output is dynamically floating on a capacitor and you can evaluate it only in a particular period of time. Static CMOS logic will hold its value as long as the power supply is on, even if your clock is down to a frequency of 1 Hertz. So, that is why that is called static logic and this is dynamic logic.

Student: Sir the face value of both NMOS and PMOS have leakage path, so instead of giving y as one family make it as slow throughout and it can actually come out as solo.

No that is why that is the whole point of doing this exercise was to get rid of my PMOS logic, because I want to reduce logical effort. If I predischarge to ground and pull up through the PMOS logic; I am just losing the entire advantage. NMOS inherently is smaller for the same drive straight; therefore, I have to do only precharge and then pull down logic.

Student: By evaluating both the PMOS and NMOS are off.

In this example both are off.

Student: So, like there will be leakage from V DD to output, right.

Correct.

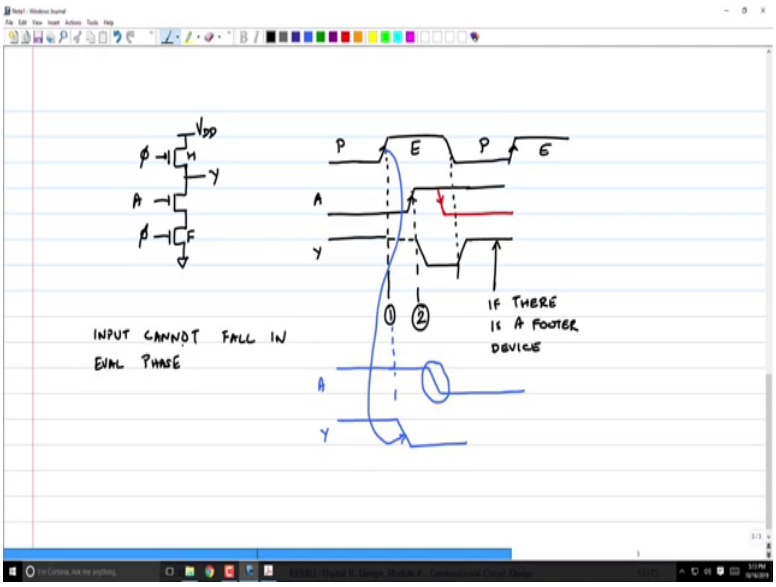
Student: (Refer Slide Time: 14:21).

Absolutely. So, you must be very careful, because again you have to size this correctly. Now the logic is why should I size my PMOS transistor if that is not going to take part in the logic evaluation; I will make that also one. So, you are right there will be some sort of a fight of leakage between these two and since the NMOS is now going to be stronger right, it will be closer to zero that state, you are right ok.

Student: So, if we make the PMOS stronger than even evaluate face comes likes (Refer Slide Time: 14:58).

Correct that is true, ok.

(Refer Slide Time: 15:10)



So, now let us consider you know this dynamic inverter again; but we will now allow the input to switch a little bit. So, this is my clock phi. So, let us say, my output, my input, right; so this is again precharge evaluate; precharge evaluate. Let us say my input was 0 and in the evaluate phase it went high ok, this is my input transition. So, how will the output y now look like? In the precharge phase A is turned off there is no problem, right. So, therefore, my output is going to be logic high. Now at this point my clock is turning off. So, between point 1 and 2, what will be the state of the output?

Student: High impedance.

High impedance. So, I am going to represent that by a floating line; of course, assuming that this is you know reasonably close you know fast time, it will not fall too much and assuming it is going to be very close to V DD, right

Now the input is going high. So, what will happen to the output?

Student: It can.

It can.

Student: Fall.

Fall, right; it will fall like this and then remain until the next pre charge phase comes, correct.

Student:

This. So now, what is the problem that you see here first of all? Both are.

Student: On.

On. So, in the precharge phase I have a constraint that the input cannot.

Student: Be high.

Be high, the pull down network should not be active, right. So, for that what I do is, I just add another transistor here called a footer transistor. Connecting the same clock. So, when the clock is 0, it is in precharge phase; PMOS will be turned on and the NMOS will be turned off, ok. This is my footer device and this is my let us call it a header device; the NMOS transistor, footer NMOS is off during the precharge phase right, it is a PMOS that is turned on.

Therefore, now input a can be any value, because there is no path to the to ground; because I have cut off that series path right and therefore, here there will be no problem. If there is footer device, clear. Now I will give you another interesting scenario. Let us say that this input went high and then it fell here; what will happen to the output here?.

Student: There will be no high.

So, the output is . right; when input A went high, output went to 0. Now my input fell, there is no path to pull that y up back to V DD and therefore, this will just remain at 0, right. So, there is a very serious constraint here that this kind of a circuit can tolerate only a rising input right or a falling output. You have to design your circuit, so that the inputs during the evaluate phase do not switch in the other direction; it can go from low to high no problem, but the input cannot go from high to low, right.

So, what I am saying is, suppose my input A was like this right with the footer device; during precharge my input A can be high no problem, right. And then my clock sort of turns off the header device and it goes to evaluate phase; what will happen, immediately my output y which was 1 will fall, right.

So, if this is high like this and let us say it falls somewhere in between. As soon as this clock goes high, this will result in the output falling. Now my input after some delay falls, but nothing can pull my output back high; therefore, this transition is not allowed.

Input cannot fall in eval phase ok. So, therefore, of course, it is very clear that it can be used only in certain conditions, certain constraints. If you have no control over the inputs and you have to guarantee that the circuit works for any transition, well you cannot use this style of logic. Use the static CMOS logic; you have lower logical effort and you have those problems. But at least logically it will work.

So, under the constraints where you are able to impose these conditions, like in that D rams circuit; the place where these circuits is used quite often is a dynamic random access memory

circuit. I want to have time to like show you exactly what happens; but its effectively this dynamic NOR gate is what gets used in a D RAM memory. In those places I can impose these constraints and therefore, use the logic and therefore, it is very fast, ok