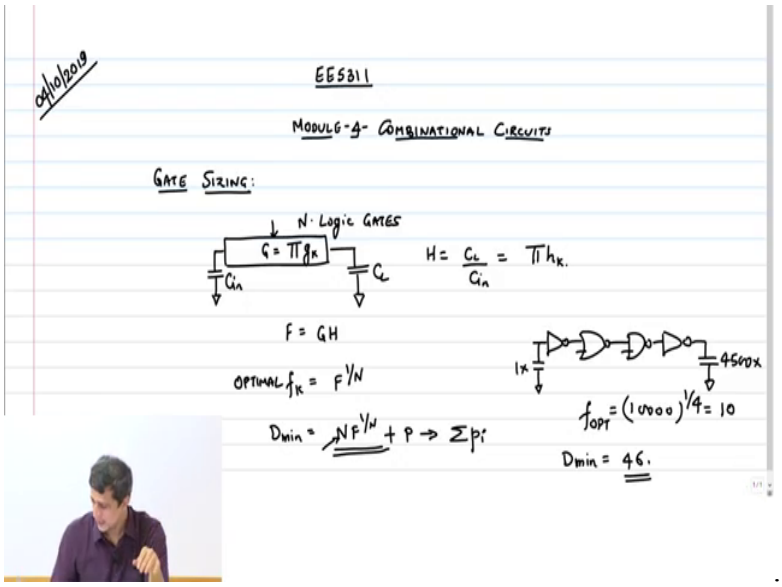
Digital IC Design
Prof. Janakiraman Viraraghavan
Department of Electrical Engineering
Indian Institute of Technology, Madras

Lecture – 43
Buffer Insertion

(Refer Slide Time: 00:20)



So, let us move on with our discussion on gate sizing right. So, what was the summary of gate sizing? If you have some arbitrary circuit with a path logical effort of G which is equal to product of the individual logical efforts right. And if the load capacitance is some value C L right and if this input gate presented a capacitance of C in then the elect path electrical effort was C L by C in and this again by the way is just the product of the stage electrical efforts.

So, the path effort F is just G into H right and we said that in order for the delay to be minimum each gate had to do equal effort right. Of course, the equal effort is not in terms of arithmetic mean. This is in terms of the geometric mean right F power 1 by N kind of thing.

So, the stage effort optimal. Stage effort optimal f K is F power 1 by N. So, that the job of charging the load capacitance is sort of evenly distributed amongst all the gates in that circuit ok. This circuit could contain any number of you know it contains N gates that is that is what this N is, so it has N logic gates ok. And therefore, the minimum delay was N into F power 1 by N plus P and what is P? This is nothing, but summation P parasitic. Unlike the pathological effort and path electrical effort, the path parasitic effort is just the sum of the delays. So, on the product of the delays ok. Just keep that in mind.

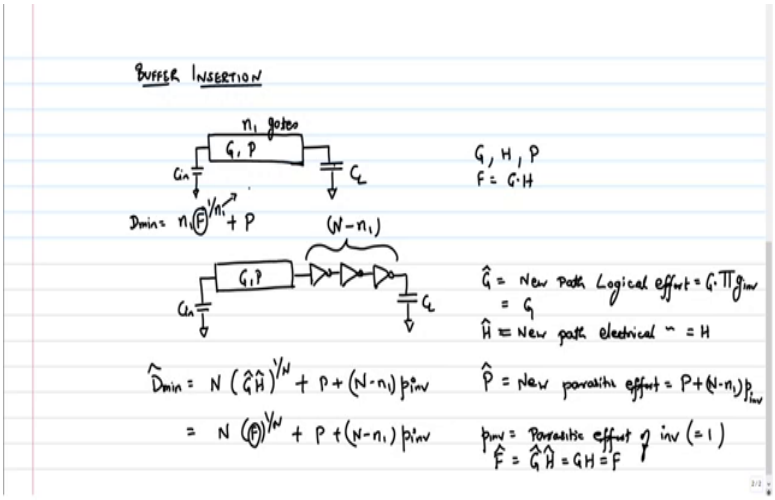
So, we did some examples and in one particular example we basically took an inverter NOR and then another NAND both 2 input final inverter and then we made it drive a very large load capacitance assuming that this is a one x capacitance. It turned out that for this particular circuit the optimal stage effort.

I will just say F OPT was 10,000 power 1 by 4 because there are 4 gates and therefore, this was 10 and the minimum delay was what? 46 normalized minimum delay was 46. The thing is that from this formula it is evident that if N happened to be a very small number then this F power 1 by N can come down only by that much because you have a only a square root or a cube root or a both power to you know 4 through to take.

And with that you cannot reduce this delay too much right. Remember that though delay is scaling with N; that means, if I increase the number of gates N is I mean the delay will go up because there is a multiplication by N. The power of F power 1 by N is much higher. It is bringing the delay down significantly right and therefore, we said that if you have a path with very few gates driving a very large load capacitance even this delay is technically not the most optimal.

It is optimal given that you have only those will many number of gates 10 gates. That is true, but if I am allowed to increase the number of gates then there is a chance that you can reduce the delay right. And we discussed yesterday that in order to add any gate on the path the best gate that you can add is the inverter because the logical effort is one ok.

(Refer Slide Time: 05:36)



So, we are going to now look at the problem of Buffer Insertion. So, I am going to take my old circuit which I had. I had a path logical effort of G path parasitic effort of P and this was my C in gap and I had a load gap here C L ok. So, the path logical effort is G and there is a H and there is a P. Now, I am going to alter the circuit as follows. Same C in because that does not change. Then I start inserting inverters in the path. Some number of inverters I am going to put ok. I am not going to worry about the logical inversion at this point C L.

So, this is G comma P that does not change. So, what is the new path logical effort? Is yeah? It is just G into the product of the logical effort of all my inverters correct. How many inverters am I adding? I am going to add let us assume that there are here there are n1 gates I am going to add N minus n1 inverters. So, that the total number of gates in my new path will be capital N ok. So, this is till G. Now, what is my new path electrical effort? Same thing, it is output load capacitance by input load capacitance that is not changing.

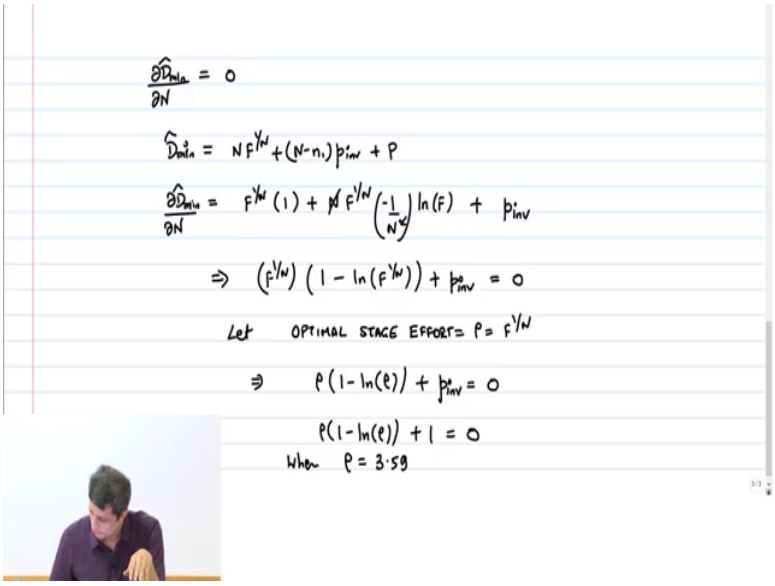
So, that is also equal to H correct. What is my new parasitic delay? Right. New parasitic effort is what? The old parasitic delay plus the delay of N minus n1 inverters. So, that is something that is quite obvious. If I add inverters I am adding a delay penalty of those inverters right. The only advantage is that delay goes linearly the other thing comes down exponential and therefore, I will still gave that is the intuition.

So, this is P plus N minus n1 into. I am going to call the parasitic effort of the inverter as P in ok. It is actually 1, but I am just giving it another variable because there are other inverters that I can construct which have parasitic effort less than 1 ok. So, we do that later. P in this parasitic effort of inverter which is equal to 1.

So, if I now have total number of N gates, what is the minimum delay that is achievable in this circuit? The D min right is simply going to be N into G hat into H hat power 1 by N right plus the parasitic effort of the entire circuit. What is that? That is P plus N minus n n1 into P. Of course, we know that this is nothing, but N into F power 1 by N ok. So, if F is equal to G into H for the old circuit then F hat is equal to G hat into H hat is also equal to G H is equal to F that does not change ok; N into F power 1 by N plus P plus N minus n1 into.

Now what was the delay of this earlier circuit here? Right. I will say D min hat because that is my new circuit. The D min here was n1 F power 1 by N 1 plus P. So, it is important to note that these 2 values are the same in both equations. I have only increased this n1 to capital N a larger number and of course, thereby I have incurred a linear increase in delay of the parasitic term correct.

(Refer Slide Time: 12:26)



So, now I want to find the optimal number of inverters that I can add in the path which will make my delay an absolute minimum right. So, what do I do? Dou D min hat by dou n equal to 0. Correct? So, can you please differentiate this and set it to 0 and tell me what you get. Do you get this? Can write this as F power 1 by N into 1 minus lon of F power 1 by N plus equal to 0. You all of you get this? Ok.

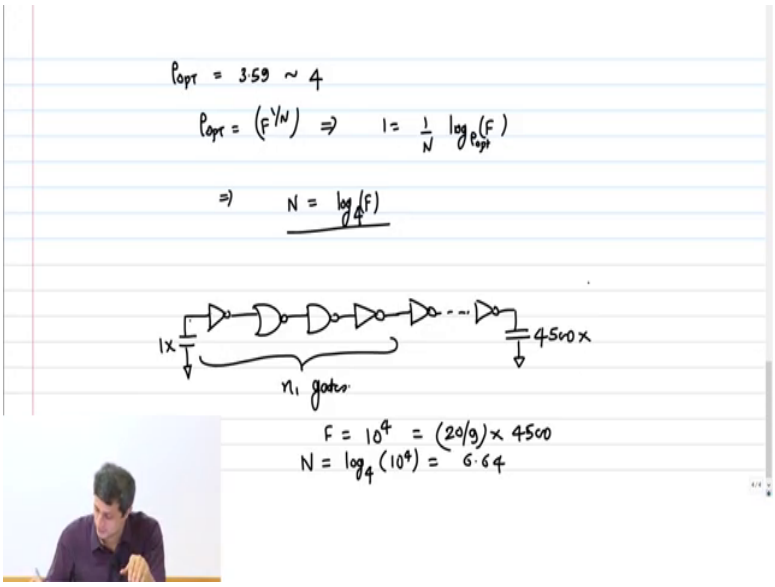
So, now let the optimal stage effort equal to. I am going to call it as rho as F power 1 by N implies I have the equation that I want to solve is rho into 1 minus lon of rho plus equal to 0 ok. Now, if P in is 1. We are talking about a standard reference static CMOS inverter right not reference static CMOS inverter then this will simply become rho into 1 minus lon of rho plus 1 equal to 0. Of course, the problem is you cannot solve this analytically. So, you have to

find the solution numerically and it turns out that when rho rho equal to about 3.59 you can evaluate this and check for yourself this quantity on the left hand side will be very close to 0.

So, therefore, it turns out that the optimal stage effort is about 3.59 ok. Now, 3.59 unfortunately is not very good when I want to do gate sizing because even yesterday I did not talk about it. We got some answers for the gate sizing, we got fractions. All these fractions are actually not available to you right because I have gates which are of fixed sizes 1X 2X 4X and you know sometimes if you want some very special gate you can make an arbitrary width as well.

But mostly from the standard cell library you will get only fixed gates. So, you have to after you get this continuous solution you have to discretize it ok. The it turns out that discretizing this optimal stage effort may to 4 is not a very very bad approximation. It turns out that the delay goes up only by about 2 percent ok.

(Refer Slide Time: 17:29)



So, my rho opt which was 3.59 can be approximately made 4. And the D min will go up very very very slightly ok. It is not too much of a problem there. So, therefore, it is an accepted approximation to make and that is why you will find this fan out of 4 to be such a magic number in digital design because that is the fan out or that is the optimal stage effort that is going to give you the minimum delay.

So, if I want to send one signal from here to there and I want to buffer it then what would I do. I would simply go in sizes of 4X, 1X, 4X, 16X and so on. You get a the reason is the theory that you do here. Inserting buffers and getting an optimal stage effort of approximately 4 is the best that you can get you cannot get better than that clear.

So, of course, now we need to translate this into how many buffers we wanted to add or how many inverters we wanted to add. So, of course, this rho opt we defined it as F power 1 by N

right. Therefore, if I just take a lon on both sides of with base of rho opt I would get 1 equal to 1 by N log to base rho opt of F implies that the optimal number of stages is log to base 4 of F. Yeah? Correct no no. So, that is what I am saying. The if the circuit changes then the number of buff inverters that I have to add will change. See this depends on F. It turns out that irrespect of the circuit the optimal stage effort is still 4.

It does not matter what the circuit is. That is a very good point. It is only the number of inverters that I add will change depending on the circuit that I am trying to optimize ok. So, why do not we go ahead and apply this to the previous problem that we solved yesterday. You know we did this problem right there we got this. Can you now tell me what how many buffers I need to add or how many inverters I need to add to the path right and what is the minimum delay.

Remember that the minimum delay that you get after inserting buffers should be less than this otherwise it is a useless exercise. Without buffering we got a delay of 46. Now, you tell me how many buffers you have to add and what the minimum delay is because of buffering that has to be less than 46 clear.

So, can you solve that? So, what did we have in that circuit? An inverter, a NOR, NAND another inverter right; this these are my n1 gates. I am now going to add some number of buffers and put this load capacitance here 4,500 X, this is 1 X. So, what is the optimal? So, what did we get as F? 10 power 4 and how did we get that? We got it as 20 by 9 into 4,500 which is basically 10 power 4 right. So, therefore, the optimal number of gates should be log to base 4 of 10 power 4. What is that number?

Student: (Refer time: 22:14).

About.

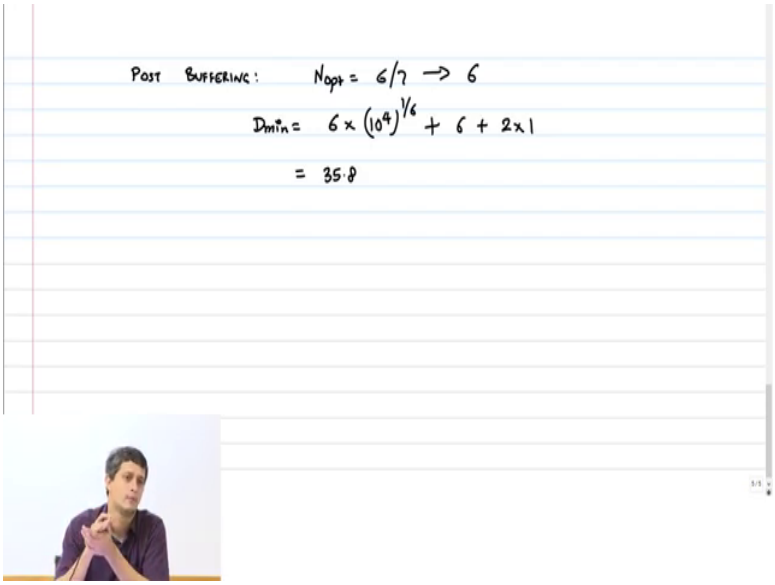
Student: (Refer time: 22:17).

About 6 point.

Student: (Refer time: 22:22).

Yeah you get 6.64 right. So, now, what do you do if you are with logical inversion you could even accept 7 because of course, there are how many 4 gates here. So, I have to add 3 more inverters right or if it is not then I can add only 6 of them. Either way you will get a delay which is lesser than that 46 that we got earlier.

(Refer Slide Time: 22:58)

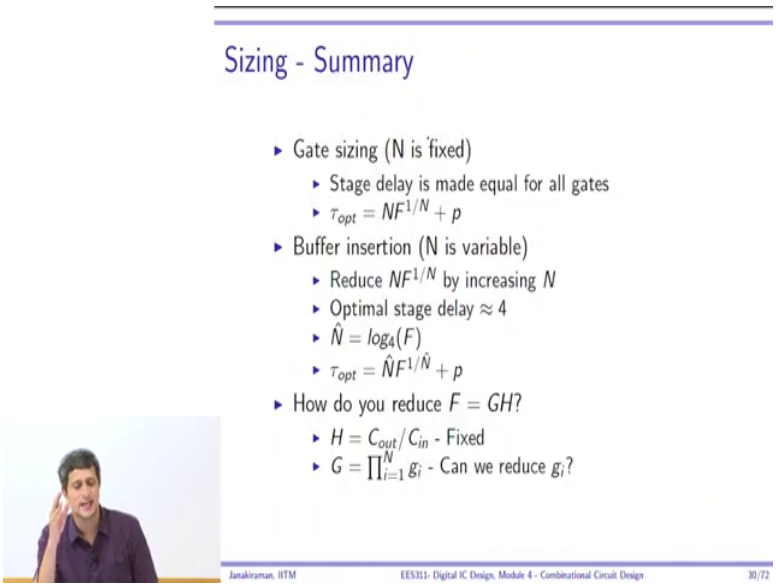


So, tell me what is the minimum delay we get because of this? Post buffering N opt equal to 6 or 7 ok. Let us take 6 so that we do not want any inversion we do not want logical inversion. So, let us take 6, I am going to make the 6. Therefore, D min is 6 into 10 power 4 power 1 by 6 plus the parasitic effort of the circuit original circuit. How much was that? 6 plus how many

no 2 because I added 6 gates totally right, so 2 into 1. What is the number you get? Yeah 35.8 as opposed to 46.

So, who was the; who was the guy who gave you maximum benefit? It is that F power 1 by N. I allowed that F power 1 by N to go to the sixth root and that kill the delay significantly. So, even though I am adding inverters and actually incurring a delay penalty I am still reducing the delay of the circuit eventually ok.

(Refer Slide Time: 24:49)



So, let me just quickly summarize gate sizing problem that we looked at. So, gate sizing if N is fixed that is the number of gates in the path are fixed then the stage delay should be made or the stage effort should be made equal for all the gates. That is the optimal delay is n F power 1 by N plus capital P not small p here.

If N is variable I can do something about N, then I reduce N into F power 1 by N by increasing N right, this is the counterintuitive part. Optimal stage delay or stage effort should be approximately 4 ok. And therefore, the optimal number of gates in the path should be log 2 base 4 of F right. Then the optimal effort again is this particular number. It is not just actually tau opt should be N hat F power 1 by N hat plus P plus the inverter delays right.

So, these are 2 methods in which we have reduced the delay of the circuit or optimize the delay of a path not a circuit right. I will tell you how to do it for a circuit later though it is not part of the syllabus. Question now is we have optimized N, we have done all that we could with N the only thing left for us to tweak is to tweak F if I want to reduce the delay of the path any further. Now, if I want to reduce the delay of F, F is G into H. What is H? H is output load capacity by input capacity that is fixed because I have to drive a fixed capacitance. It is coming it is specification coming from outside I cannot tweak that.

So, the only thing left for us to tweak is the path logical effort G which will simply translate to the gate logical effort G K. In order to reduce the delay of a path any further I need to reduce the logical effort of the gate. Remaining part of this module will be discussion of various styles of CMOS gates where I simply reduce the logical effort significantly step by step ok.