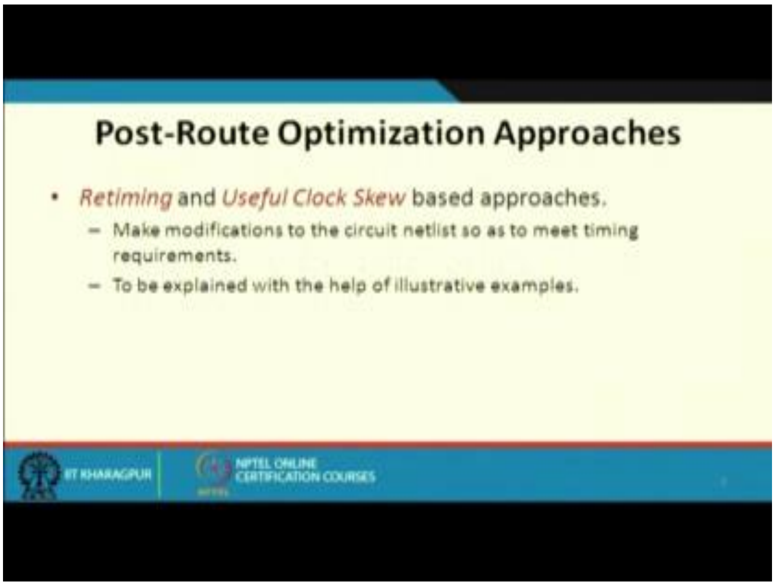
VLSI Physical Design
Prof. Indranil Sengupta
Department of Computer Science and Engineering
Indian Institute of Technology, Kharagpur

Lecture - 42
Miscellaneous Approaches to Timing Optimization

So, in this course we have seen. So, far some of the techniques where you can use the timing driven constraints in synthesis in the design flow for the physical design process. So, in this lecture we shall be looking at a few more techniques which also can be used which I have not discussed earlier. So, these we refer to as some of the miscellaneous approaches for timing optimization.

(Refer Slide Time: 00:56)



Post-Route Optimization Approaches

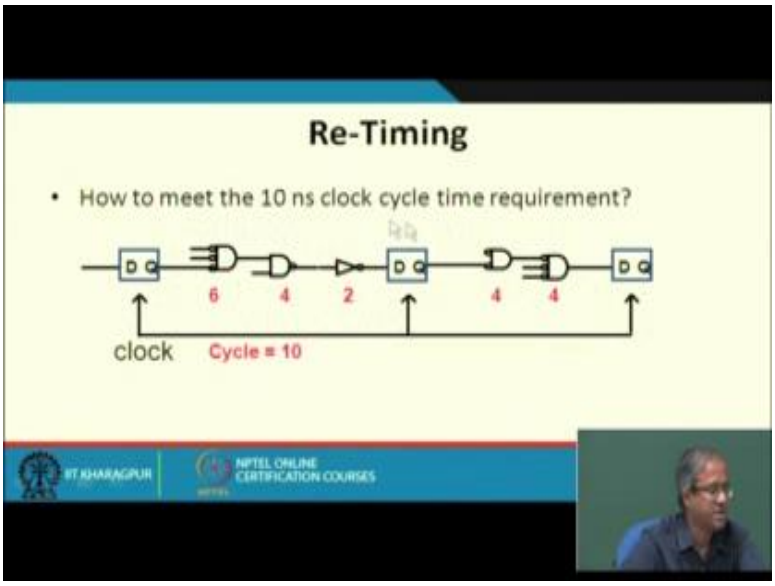
- *Retiming* and *Useful Clock Skew* based approaches.
  - Make modifications to the circuit netlist so as to meet timing requirements.
  - To be explained with the help of illustrative examples.

IIT KHARAGPUR
NPTEL ONLINE CERTIFICATION COURSES

Let us see some of the techniques, like a couple of very useful techniques which are practiced they refer to as a retiming and useful clock skew. Now retiming and useful clock skew what they refer to is that you do some kind of modifications in the circuit netlist. So, as to meet the timing requirement which is different from the ones we have already seen earlier. Like say we are given a circuit netlist and we have some timing parameters or aquastics to be met there are some violations. So, earlier what you have seen we can either increase the size of a gate, we can reduce the delay of a line possibly by inserting buffers, then we can also use a variety of techniques like restructuring the fanin and fanout paths of a circuit then, duplicating or cloning the gates using the rules of boolean algebra to make some simplification or modification to the gate level netlist and so on and so forth.

But there are some other techniques also which work at the structural level of the circuit and carry out some transmissions therein so that some of the delay related constraints which are not otherwise getting satisfied they can be satisfied.

(Refer Slide Time: 02:35)

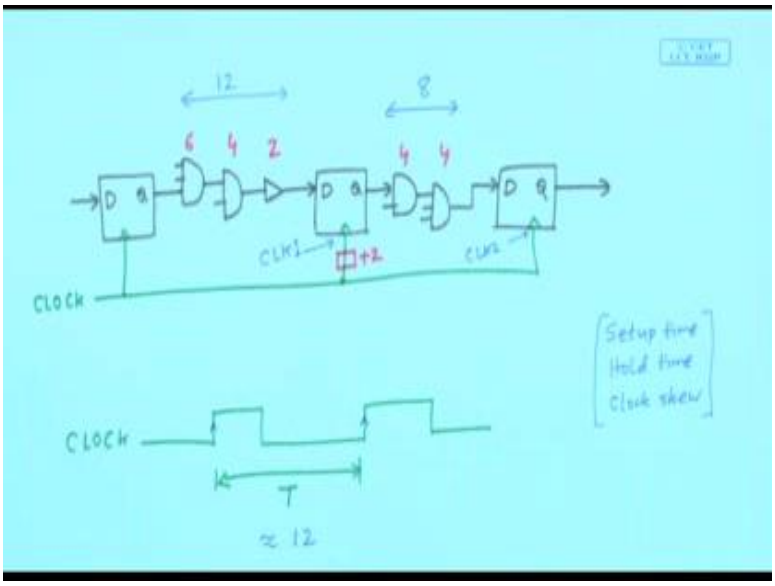


Re-Timing

- How to meet the 10 ns clock cycle time requirement?

So, let us see let us look at the concept of retiming. So, here we have a very simple example let us try to illustrate with this examples itself. You see there are 3 flip flops the clock signal is coming to all these 3 flip pops. And there are some gate level combination circuits in between.

(Refer Slide Time: 03:03)



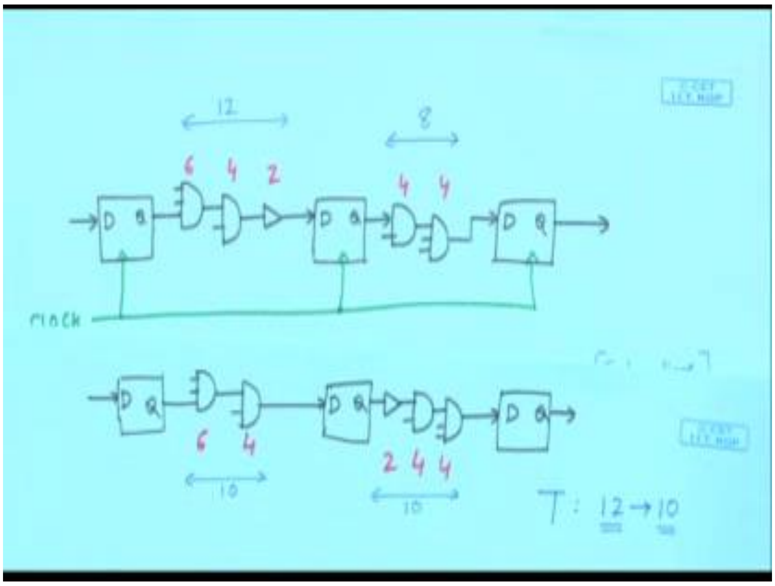
So, here let me just redraw this circuit again. So, here I shall be explaining these are the 3 flip flops. So, here what you are saying is that the input is coming here the output is coming to a 3 input gate, let us say like means I am using that same example. This is coming turn that input what type of gate is not important. So, I am not showing that this is coming to the input again here, let us say there is another gate which is coming to another gate, which is coming here. Let us say that the delay of these gates as was mentioned in the slides are let us say this is 6, this is 4, this is 2, this is 4, and this is 4. And the clock signal is being fed to all the 3 flip pops.

Now, the question is. So, when we talk about the clock signal. So, what should be the time period? When the here a few things we are ignoring here because our point of focus is a little different. We are ignoring setup time, hold time. We are also ignoring clock skew. Because earlier we have seen what these are and how we can handle this. What are the different constraints that arise out of these things? So, here we are assuming on I means an idealistic scenario that the clock edges are sharp everything takes place at the clock transition, let us say they will take place at the leading edge of the clock. So, we have seen that with respect to this register transfer level netlist with these discrete delays are shown what should be the time period of the clock.

Now one thing we can immediately see that in this circuit there are 2 combinational sections, the first path has a delay of 12 the second path has a delay of 8. So, in order for correct operation, so this time period should be at least equal to 12, again I am saying ignoring setup hold and clock skew. So, the actual time will be 12 plus all these things.

So, ignoring this the time period should be at least 12, because if it is less than 12 then this first set of combination circuit will not finish it is separation before the next stage comes alright. Now the point is that, when if you look structuring at this level what you can possibly do? Like you can make an observation that well I see that well the combinational logic is not uniformly distributed here. There is a delay of 12 here, there is delay of 8. So, can we move some of the gates across this flip pop to this side, without changing or modifying the functionality? So, this is a possibility for example, in this same example let us do this.

(Refer Slide Time: 07:02)

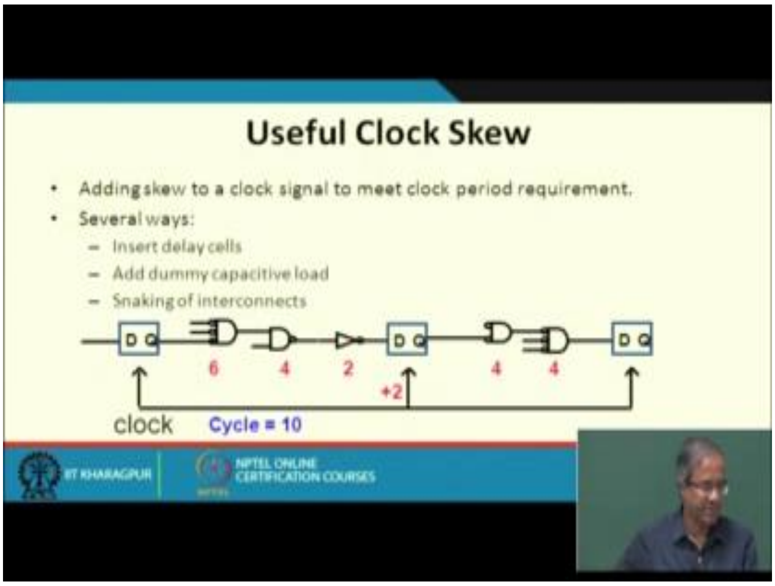


Let us try to make this change. Let say the same 3 flip flops are there, but the change that we are making is that here instead of these 3 gates we are keeping only 2 gates here. And this gate which was there we are shifting it out here, something like this. So, if you do this what advantage you are getting? As you can see for this example at least, the delay here is 6 plus 4, 12. Here it is 4 plus 4 8 and this gate delay was 2.

So, now the delay is getting uniformly balanced. Here also it is 10, here also it is 10. So, now, we have a we get a scenario that the clock period T which was earlier 12 because the maximum delay was 12, we have been able to bring it down to 10. Well not by doing I mean adding some hardware just one gate we shifted from this part of the circuit to this part of the circuit. Now in a typical data path when digital portion of a circuit we encounter such kinds of circuit where there are a number of flip pops or storage cells there are combination circuits in between. So, it is possible to move some gates or circuits around across flip-flops without modifying or changing the functionality of the circuit.

Well means of course, if the output of this flip pop is going somewhere else fanin out. So, those fanout connections you have to modify accordingly right. So, v timing is a simple technique as we have illustrated through this example where the clock cycle time which was earlier 12 we have been able to bring it down to 10 fine. So, this is exactly what is shown in this slide. So, what we have worked out. So, earlier it was 12 now by moving this gate out here, now the new clock circle time becomes 10.

(Refer Slide Time: 09:52)



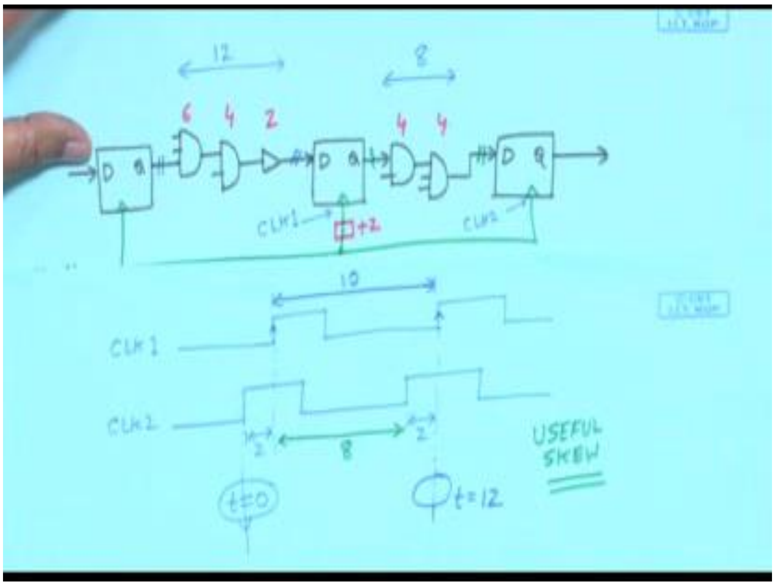
Useful Clock Skew

- Adding skew to a clock signal to meet clock period requirement.
- Several ways:
  - Insert delay cells
  - Add dummy capacitive load
  - Snaking of interconnects

Now, there is another important thing useful clock skew. So, let us look at this diagram again. So, let me explain this diagram I shall again go back to the slide, you see here this one is circuit where the combination circuit was not balanced. So, what you are saying, we are saying that well here you see that there is a combination part whose delay was 12 this was 8, and assuming that there is no clock skew the clock cycle time was estimated to be 12 the maximum piece.

Well now what you are saying that can we deliberately insert some delay in the circuit. How like say for example, we deliberately introduce a delay out here. So, see there are so many ways to introduce a delay. Like we can simply introduce a buffer, just we introduce a pair of inverters, we introduce a long zigzag wire that will be a interconnection delay there is. So, many wires so many ways to introduce a delay. So, what you are saying is that in one of the flip flop where the clock is coming, we are deliberately introducing a skew, introducing a delay. What I saying is that suppose we are deliberately introducing a delay out here, let us say a delay of plus 2. So, now, what will happen let us see? Let us keep some names because these 2 flip flops are in question. Let us say this clock is clock 1, let us call this as clock 2. So, now, let us see what happens. Let me just workout here again.

(Refer Slide Time: 12:05)



So, what will clock 1 say. Clock 1 was like this it was coming. Clock 2 what here clock 1 is a this is having a delay of 2. So, what will be a clock 2? Clock 2 will be having the edges coming through time units before this like this, this much is 2. So, this and this coming, so this much is 2; now because of this let us see what is the impact on our circuit performance and operation.

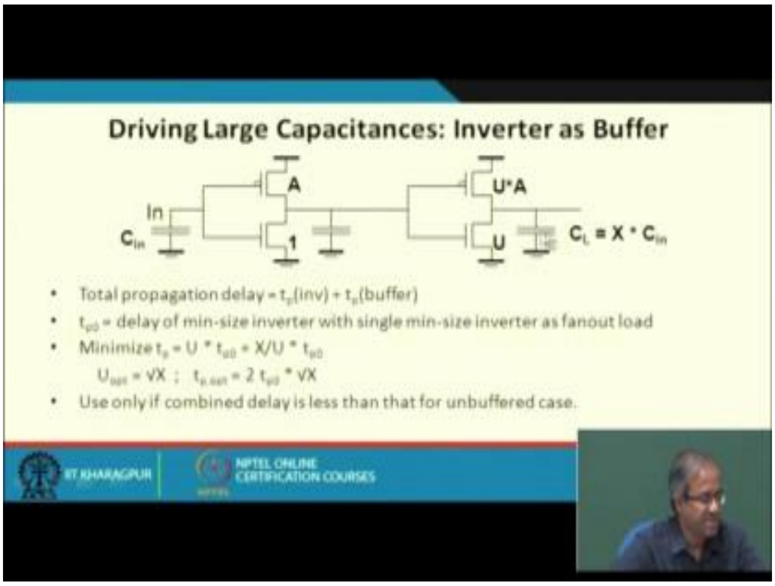
You see this was my circuit. The first let us say let us start from time t equal to 0 where we are assuming that this flip pop output has generated some data that has to be stored here. So, clock 1 what I am saying is that clock 1 we are generating a delay of deliberately. So, this is my let us say time t equal to 0. This is my time t equal to 0 where we have generated this data at Q. Now at this point what happens let us see. This data which is there this will be, this available at the input of the after time 12. Now after time 12 what will happen? And this let us say this let us assume our clock period is 10 now 12. So, our clock period this is 10, this is what we are wanting to achieve.

Earlier we did it by moving a gate around, but here let us see this 10. So, this is my time t equal to 0 where this Q is changing. So, after a time 12 this input of D should be ready. So, what is that 12 this is 2 this is the skew I have generated this clock 1 is coming after that plus 10. So, here this point refers to t equal to 12. So, at this t equal to 12, this data should be ready. So, when this clock h comes clock 1 this data is already ready. So, there is no problem fine.

Now, look at the second one. So, whenever this clock 1 h come, let us say the previous one whenever the clock 1 h comes, this this flip flop will be storing it is data. So, let us use a different color. So, the data will be stored here, let us say here. Now this circuit requires 8 units of time right to compute. So, after 8 units of from this point to this point it is 8. So, you see by the time the clock 2 h comes this time 8 is there. So, when clock 2 comes this data is already come here. So, you are correctly latching it here. So, you see this circuit works perfectly well. Just by adding a delay of 2 units in this clock 1 we have been able to balance it without moving these gates around right. This is something which is called useful skew. So, we are deliberately introducing skew in a circuit, such that the circuit can work faster, like in this example as we have illustrated. The circuit can work with a faster clock instead of 12 we had been able to work with a clock period of 10.

So, let us come back. So, clock skew just to summarize. Here you have saying that we are adding skew to a clock signal in this case the second one, the middle one. So, there are several ways as I had said you can introduce some buffers to delay the signal. You can introduce some dummy capacitive load instead of introduce anything just add some fanouts; that means, some loads here. That you slow down the signal or snaking make a zigzag connection here. So, that way interconnection delay becomes longer. So, by doing this as I have shown through that timing diagram, the cycle time becomes 10 here fine.

(Refer Slide Time: 17:34)



Driving Large Capacitances: Inverter as Buffer

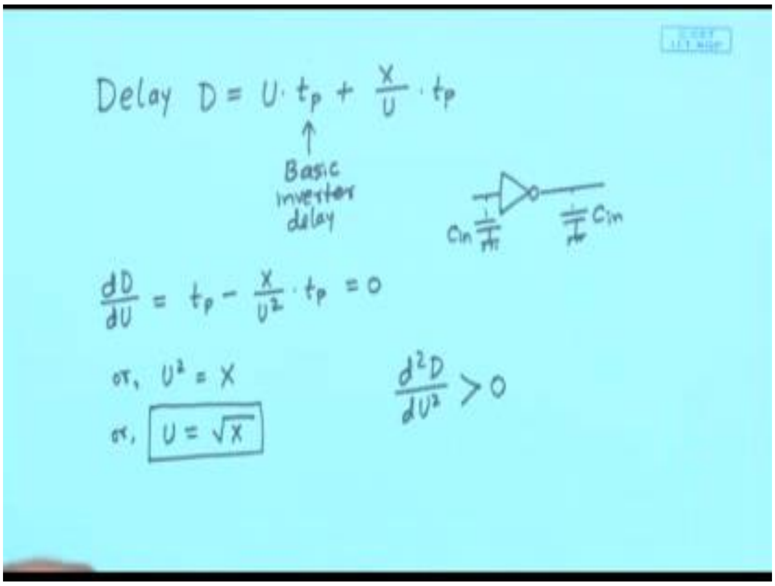
- Total propagation delay = t_p(inv) + t_p(buffer)
- t_{p0} = delay of min-size inverter with single min-size inverter as fanout load
- Minimize $t_p = U * t_{p0} + X/U * t_{p0}$
  $U_{opt} = \sqrt{X}$; $t_{p,opt} = 2\ t_{p0} * \sqrt{X}$
- Use only if combined delay is less than that for unbuffered case.

Now, let us look at another sub problem, driving large capacitances. So, here let us say we are using an inverter as a buffer. So, let us look at this cmos level diagram. Here we have a cmos inverter. Here we have another cmos inverter let us say this inverter is driving this inverter right. And this this means output load capacitance this say C L. And you recall earlier we said that this N type and t type transistors are approximately equal in size, but I also said because of difference in mobility’s of electrons and holes the t type transistor is slightly larger. So, let us say the sizes are 1 and capital A, where A is a little greater than one maybe 1.1 or 1.2 something like that. And this C in denote the base capacitance value of this smallest inverter, this smallest inverter can have the pull up and pull down sizes of 1 and A, for this smallest one the sum of the gate capacitances let us call it as C in.

So, this kind of a inverter is driving means another inverter which is larger, this gate is larger the sizes are like you see capital U and U times A. So, we have made both these inverters U times larger. So, that this inverter can drive current which is U times more because the resistances are U times less. They are wider; they are U times wider that is why their resistance value will be R divide by U. So, RC will be the total charging and discharging time if I can have made resistance as R by U charging discharging will become U time faster.

So, let us assume that the total output load capacitance is capital X times this minimum base capacitance C in. So, X is a factor. So, with respect to this let us try to estimate what will be the total propagation delay of this pair of inverters. So, I am just writing down the expression you just see whether it make sense.

(Refer Slide Time: 20:24)



Delay $D = U \cdot t_p + \frac{X}{U} \cdot t_p$

↑ Basic inverter delay

$$\frac{dD}{dU} = t_p - \frac{X}{U^2} \cdot t_p = 0$$

or, $U^2 = X$

or, $\boxed{U = \sqrt{X}}$

$$\frac{d^2D}{dU^2} > 0$$

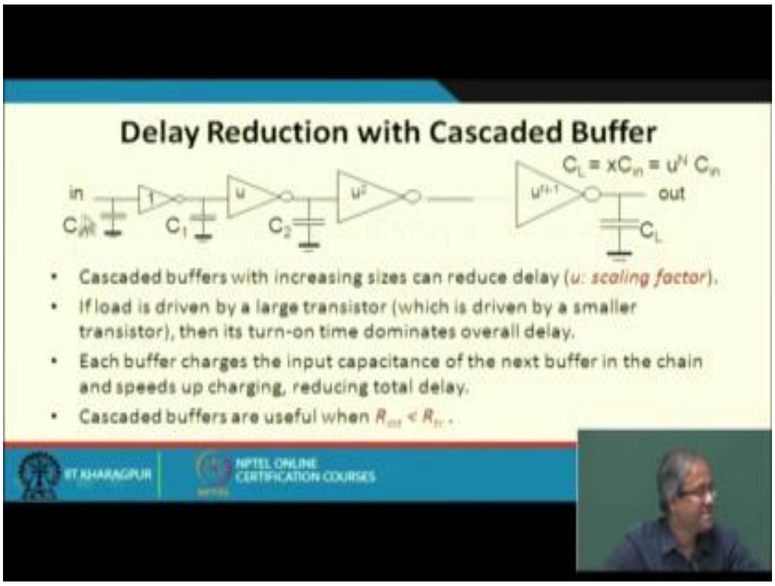
So, the total delay let us call it capital D. I am writing as capital U into t p plus X by capital U into t p, where t p is the delay is you can t p, is referred to as the basic inverter delay; that means, delay of a smallest inverter basic inverter delay when it is driving a similar load; that means, basic inverter delay means, an inverter whose input capacitance is as I had said we assume the smallest one to be C in. So, we are assuming that the output capacitance is also C in. So, it is driving another gate which is also of the same smallest size that you are calling as t p. Now let us try to understand how this expression is coming U multiplied by t p, you look at the slide once again.

You see the first inverter is driving a capacitance this capacitance this is U times larger. So, this delay will be not just t p, but U times t p because the charging and discharging will take U times more, because the capacitance value will be U into C in right. This is U into C in. That is why this would be U multiplied by basic inverter delay and what about the second one? Second one you see, second one is this transistor is U times larger, and it is driving a load of X time C in. So, how much will be the, I mean how many times of the basic inverter delay will it require? It will be X divided by U. Because larger the U less will be the time or higher the X more will be the time, that way you can estimate. So, for this inverter driving this C L the delay will be X by U multiplied by t p.

Now, we are trying to find out the minimum D, for what value of capital U D will be minimum. So, what do we do? We differentiate D with respect to U. So, how much will it be? T p minus X by U square into t p, to find the minimum value we equate this to 0. So, from this we just if you solve you get U square equal to X. Well or U equal to square root of X. This is the optimum value of X. Well if you do a double differentiation D 2 D U square with this value substituted you will find that will be D 2 D DU square will be positive meaning thereby that this is the minimum value not maximum. So, the value of U has to be chosen as square root of X, in order to minimize the delay. This is a very interesting equation. This says that if we have a scenario like this where you are driving a capacity load which is X times of C in, and you have one transistor a bit transistor like, what should be the value of U it should be square of root of X. Then the total delay will be minimized.

So, here I am saying that we are using means a pair of inverters to drive C L, like that you can imagine, one small inverter one large inverter. So, the large inverter size should be square root of X.

(Refer Slide Time: 24:48)



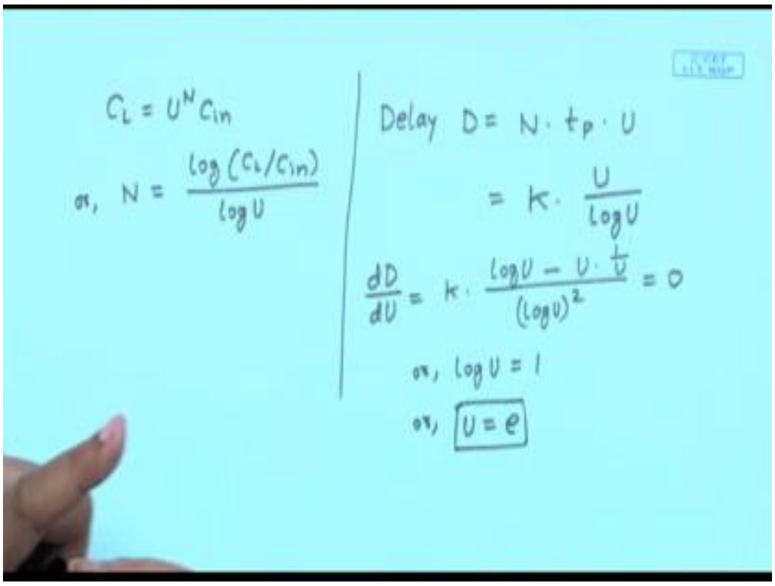
Delay Reduction with Cascaded Buffer

$C_L = xC_{in} = u^N C_{in}$
in
C_{in}
1
C_1
u
C_2
u^2
u^{N-1}
out
C_L
IIT KHARAGPUR
NPTEL ONLINE CERTIFICATION COURSES

- Cascaded buffers with increasing sizes can reduce delay (*u: scaling factor*).
- If load is driven by a large transistor (which is driven by a smaller transistor), then its turn-on time dominates overall delay.
- Each buffer charges the input capacitance of the next buffer in the chain and speeds up charging, reducing total delay.
- Cascaded buffers are useful when $R_{out} < R_{tr}$.

Now, let us look at a mode general case where you are not using 2 inverters, but a cascade of buffers, you see this is what is done in practice typically. To drive a large capacity load, we use a cascade of buffers. So, each of them will be U times larger, like if the first inverter is the smallest one with the size of 1 this would be size U, size U square, last one will be U to the power N minus 1. So, if C L is the total capacitance. So, and C in is the input capacitance. So, C L is determined as U to the power N C in. This is how you define U. So, for this case let us again similarly try to find out what should be the best value of U. So, that this this overall delay is minimized.

(Refer Slide Time: 26:01)



$C_L = U^N C_{in}$

or, $N = \frac{\log (C_L/C_{in})}{\log U}$

Delay $D = N \cdot t_p \cdot U$

$= K \cdot \frac{U}{\log U}$

$\frac{dD}{dU} = K \cdot \frac{\log U - U \cdot \frac{1}{U}}{(\log U)^2} = 0$

or, $\log U = 1$

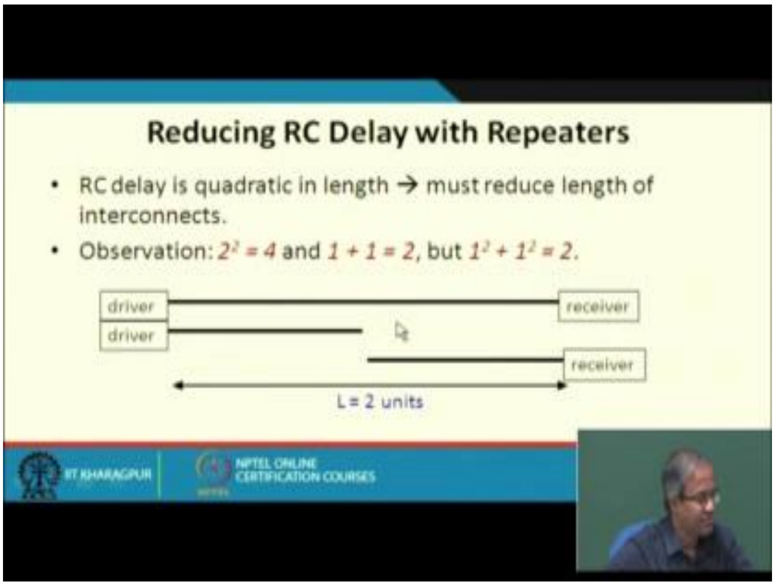
or, $\boxed{U = e}$

Now, what we have already seen is that from here, that that we have assumed that C L will be U to the power N C in. So, from there U can write what will be n? N will be log of C L divide by C in divide by log of U, this is N. And this is 1 and the delay total, what will be the total delay again in this case? You see here you look at this diagram again; here each inverter is driving another inverter which is U time larger - 1 into U into U square. So, so each inverter is driving another inverter which is U times larger, which means the affective delay will be t p 0, t p.

You can say t p into U. It will U, this is will be t p into U. This will be again t p into U. Because here resistance is U times less that the capacitance is U square times more. So, U multiplied by U square it will become one U will cancel out it will be U times. So, the total delay will be, N times t p U. So, if you substitute this value of N here this becomes you see this log C L by C in and t p these are constants. So, I am just writing them as a single K, it will become U by log U this kind of an equation. So, again you are trying to minimize D. You again differentiate D with respect to U. So, this K is a constant. So, this will be log U D U 1 minus U derivative of log U 1 by U divide by log U whole square. This will be 0. So, if you solve you will be getting this is 1 log U equal to 1.

Now, since you have taken derivative log U is 1 by u; that means, you have assumed the base to be e. So, you can say U equal to e the naperian base of 2 point 7 approximately. So, U the optimum value of U should be e. So, here you get this equation.

(Refer Slide Time: 28:50)

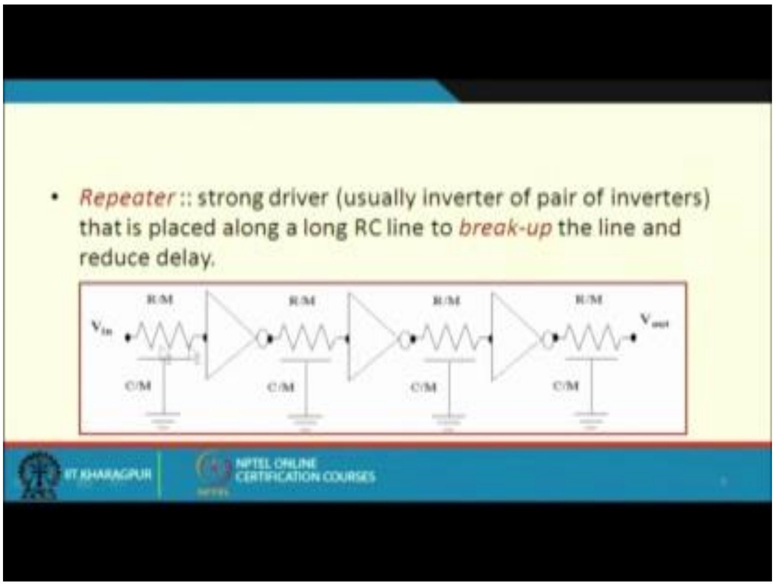


Reducing RC Delay with Repeaters

- RC delay is quadratic in length → must reduce length of interconnects.
- Observation: $2^2 = 4$ and $1 + 1 = 2$, but $1^2 + 1^2 = 2$.

Now, another interesting thing is that, this I have mentioned earlier that you can reduce RC delays with repeaters. Now I mentioned that the total delay along a line this quadratic in length which means, it is proportional to the square of the distance. So, if you a line of length 2, the delay will be 2 square 4, but if you use a repeater in between break the wire into 2 parts it will be 1 square plus 1 square. So, instead of 4 the delay now becomes 2. So, by introducing repeaters in between the total delay can be reduced.

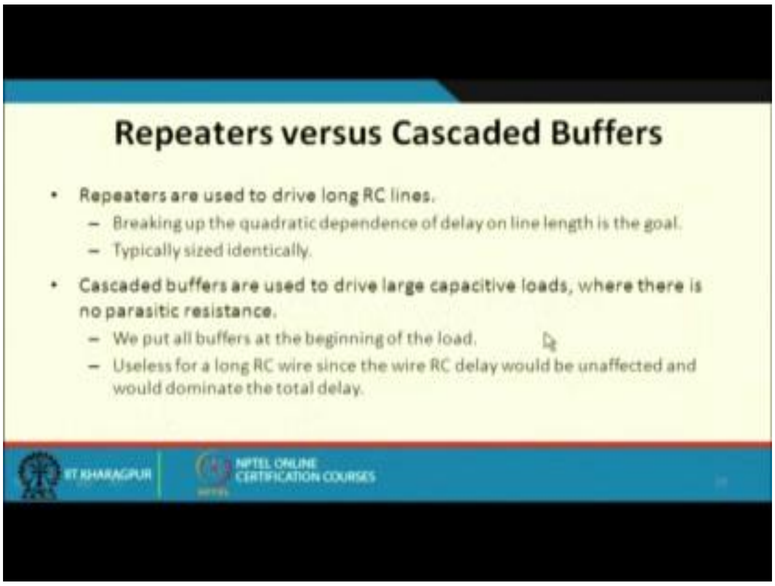
(Refer Slide Time: 29:38)



- *Repeater* :: strong driver (usually inverter of pair of inverters) that is placed along a long RC line to *break-up* the line and reduce delay.

So, repeaters are like this. Repeaters are nothing, but some strong drivers it can be inverters or a pair of inverters, which can be used to drive smaller segment of wires. Like here I have said a long wire you divide up. Similarly, here there are several repeaters I have inserted, so each of these wires. So, if the total resistance was capital R, I have divided them into M, M such segments R by M, R by M, R by M. Similarly, the capacitance is also getting reduced C by M C by M. So, the total delay will be less.

(Refer Slide Time: 30:19)

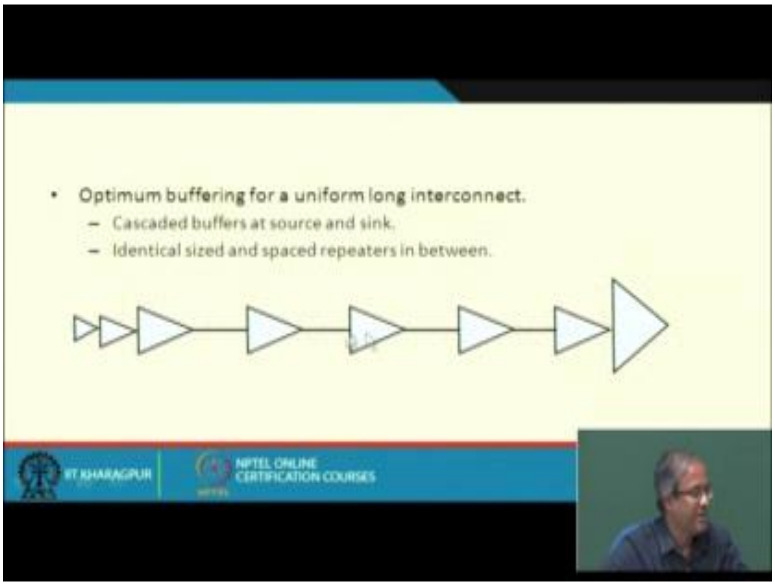


Repeaters versus Cascaded Buffers

- Repeaters are used to drive long RC lines.
  - Breaking up the quadratic dependence of delay on line length is the goal.
  - Typically sized identically.
- Cascaded buffers are used to drive large capacitive loads, where there is no parasitic resistance.
  - We put all buffers at the beginning of the load.
  - Useless for a long RC wire since the wire RC delay would be unaffected and would dominate the total delay.

So, now the question arises whether to use repeaters or cascaded buffers because both the options are there. Repeaters are normally used to drive long lines where the RC delay dominates. So, if you break up the line into smaller segments the delay reduces. And usually the repeaters are identical in size, but in contrast cascaded buffers are used whenever the load capacitance is very high, and the parasitic resistances are typically much less, but here for long RC lines the parasitic resistances are have got significance there you use repeaters, but for this kind of a cases you can put all the buffers at the beginning of the load and you can straightaway drive it.

(Refer Slide Time: 31:17)



- Optimum buffering for a uniform long interconnect.
  - Cascaded buffers at source and sink.
  - Identical sized and spaced repeaters in between.

But if you have a combination a long line, at the end there is a larges load capacitive load. Then you can have something like this. If we have a long line at the end there is a large capacitive load, they can have a cascaded buffer at the begging to drive this and then again a cascaded buffer at the end to drive this. So, these cascaded buffers will help you in tackling the large capacitive loads and this intermediate repeater help you in reducing the total RC delay of this long interconnection line. Because this can be a long interconnect.

So, actually this is the technique you can use any combination, of that you can have you have this 2 techniques one by using repeaters other by using cascaded buffers. Cascaded buffers are to be used for driving high capacitive load; repeaters are to be used to drive long lines. So, if you have a long line with high capacitance load at the end, you have to use a combination of the 2. So, with this we come to the end of this lecture.

Thank you.