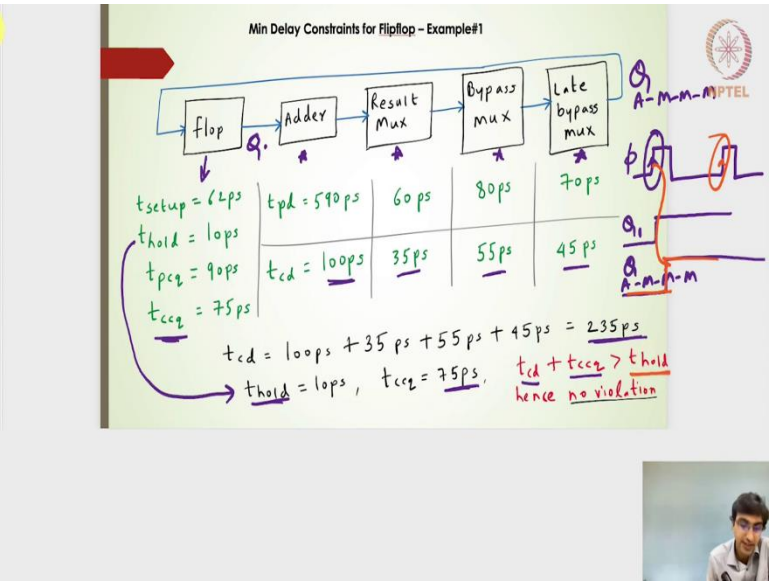
Design and Analysis of VLSI Subsystems
Dr. Madhav Rao
Department of Electronics and Communication Engineering
International Institute of Information Technology, Bangalore

Lecture - 77
Static Timing Analysis - Part 2

Hello students, welcome to this lecture on the Static Timing Analysis and there this Part 2. In this particular lecture, we will look into the flip-flop designs and then how do we not violate the hold time failures, rather how do we avoid the hold time failures. Also, we will start into the latch designs and then consider the setup time violation and the hold time failures as well for the latch designs. Let us begin let us move forward, meanwhile I will also get my pointer.

(Refer Slide Time: 00:53)



In the last part of the last lecture we had seen this particular design of the flops which is kind of parameterized or characterized with this particular parameters, like the t setup time, the t_{hold} time, the t_{pcq} and t_{ccq}. This is something the sequential circuit or the element characterized elements design is given to us, with all these set of parameters.

We have an adder, result mux, bypass mux, late bypass mux which is again going back to the flop design through the flip-flop design. We have this set of 4 combinatorial circuit

blocks, whose characterization in the form of propagation delay and the contamination delays are also given in the form of this particular values.

The one of the questions we had earlier seen was, whether this will have any hold time failures. If I actually calculate the t_{cd} of this particular 4 blocks of combinatorial circuits, which have annotated in the form of a stars, it is nothing but 100ps, 35ps, 55ps and 45ps.

$$t_{cd} = 100\text{ps} + 35\text{ps} + 55\text{ps} + 45\text{ps} = 235\text{ps}$$

That will be the contamination delay of all this, the 4 blocks which are in cascaded stages and then the output of that is going to the flop 2. What we are expecting is in the 1st clock output will give me this particular input to the adder circuit block. Then the output of this late bypass mux should be captured into the flop design into the flip-flop design only in the 2nd clock or in the 3rd clock and it should not get captured within the 1st clock itself.

$$t_{hold} = 10\text{ps}, t_{ccq} = 75\text{ps}$$

To avoid, the hold time failures we know that $t_{ccq} + t_{cd} > t_{hold}$ of 10ps then, what we can confirm is the output of the late bypass mux, will not be captured in the 1st clock edge of this flip-flop design.

$$t_{ccd} + t_{ccq} = 235 + 75 = 310\text{ps} > 10\text{ps}$$

Hence, we say that there is no hold time violation or there would not be any hold time failures, hope this is clear. What we really mean is, if I design the clock, somewhere here I will start designing the clock here. What we do not want is the output of this which is high level of edge of the clock, this is my clock design.

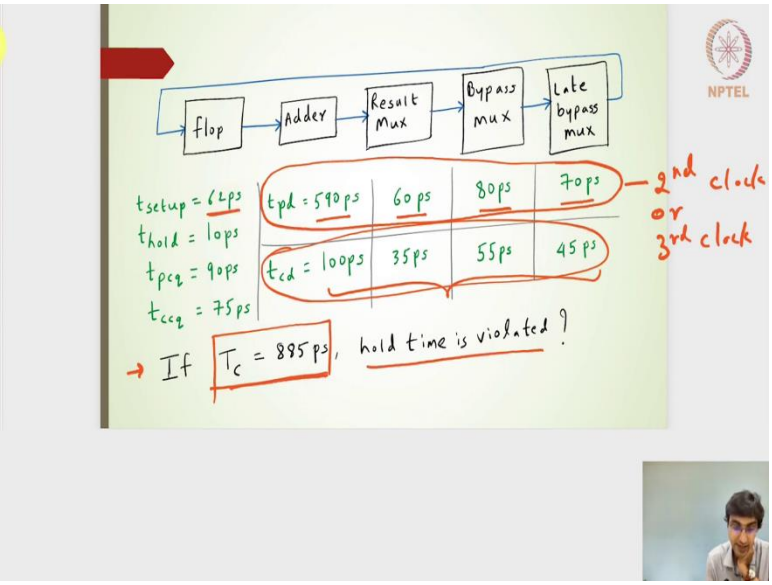
We do not want everything to be captured here, this I will write it as the Q signal, this one will be the Q1. So, what it means is for the 1st clock edge, I will get that the Q output which is nothing but, Q1 which will be after the t_{ccq} contaminated clock to Q. Then the adder output result, mux output, bypass mux output late bypass mux output, I will write it as Q of the adder, the result mux, the mux bypass mux, the late bypass mux, 3 muxes are there.

What we really do not want is with the contamination delay of all these 4 combinatorial box blocks, if the output reaches really close to the 1st clock edge. The output of this, will

be my combinatorial circuit which is nothing but, adder, muxes and muxes. The output of this should not be captured in the 1st clock edge itself.

Whatever is the t_{hold} time, this should make sure that they should be ensure that it is away from the t_{hold} time. Then only we can clearly say that the contaminated output does not get captured in the 1st clock. It should then easily captured in the 2nd clock edge. That is what we want and that is why we say that the $t_{cd} + t_{ccq} > t_{hold}$. Hope this is clear.

(Refer Slide Time: 05:34)



Moving on. Let us say if the Tc = 885ps and this is something which we had seen in the last lecture, where we calculated the propagation delays from the propagation delays of all this 4 combinatorial blocks we were able to evaluate, what should be the minimum clock time period and it was around some somewhere around 952ps.

But here, I am taking an example where the clock is compromised a bit, Tc = 885ps. Now the question is whether there will be any hold time violation. If I look into the Tc of 885ps, what we really want to know is if I take the propagation delay that means, the output of the adder, output of the result mux, output of the bypass mux, output of the late bypass mux and these outputs are actually the steady state outputs.

When we consider the propagation delay it is nothing but, the steady state outputs. It is not the glitch which is going up and down and then reaching to the steady state value. This steady state values, we need to know whether it is captured in the 1st clock or the 2nd

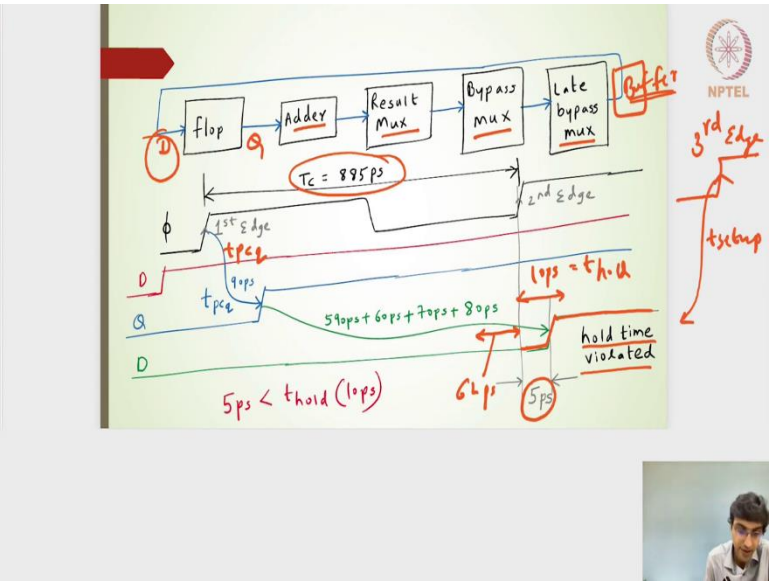
clock or the 3rd clock and so on. Then, I am going to write it as 2nd clock or 3rd clock and 3rd clock in the sense 2nd clock edge or the 3rd clock edge and then try to see whether the same applies for the contaminated output also.

If it is not, let us say that if the propagated propagation delay of the of all these 4 blocks, allows the output to be captured only in the 3rd clock edge. But the contaminated output due to this particular value is allowing the flop to capture in the 2nd clock edge then I think whether its a violation.

What we are supposed to do is both design the circuits in such a way that it does not only violate the hold time as well as the setup time failures but also make sure that, it is reaching the proper clock edge for both the propagated delay output and any contaminated delay output. In that sense we have to see whether it is kind of the hold time is violated or the setup time is violated and then proceed further.

If we have a Tc of 885ps, what we are likely to get is this clock, having at 885ps. Over the distance or the duration between the 1st clock edge and then the 2nd clock edge, we have the 885ps.

(Refer Slide Time: 08:05)



Let us say that this is the D input. The D input is coming and it has arrived earlier than the setup time or the 1st clock edge, it is readily available. The moment it sees the 1st clock edge, we will get the output of the flop. This is the output of the flop and this particular

delay is t_{pcq} propagation clock to Q. In this particular diagram, I am considering only about the propagation delays.

I am not yet come to the contamination delays, but looking at the setup time failures. The propagation clock to Q delay and then after that, there will be an adder output, result mux, bypass mux, late bypass mux. If I consider the propagation delay of adder, propagation delay of the result mux, propagation delay of the bypass muxes and a late bypass mux, which will be nothing but, 590 + 60 + 70 + 80.

I will get around 590 + 60 = 650 + 150 = 800ps, from here to here, it will be 800 + 90 = 890ps. It comes here at this particular stage, and the clock time period is 885ps. Here it comes the output here which goes to the input to the flop design. That is why I have written D again here, what it means is, the D provided here, after the combinatorial circuit output it comes back to the flop design with 890ps.

With respect to the 885ps of the clock it is 5 seconds, 5ps late. With respect to the 2nd clock edge, it is 5ps late. The question is if it is 5ps late, whether it is having any setup time failures? The answer is, No. The setup time failure will not be there, because we were supposed to arrive somewhere before the 2nd clock edge.

If it is not done that, it is also not done it is also not arriving in between in the setup time, setup time in between the setup time means the 2nd clock edge to whatever is the setup time, let me go back and then say the setup time is 62ps. It has also not arrived in between the 2nd clocked edge and then the 62ps.

It has not arrived in between the 2nd clock edge and then the 62ps. In that sense, if I say that this is my 62ps, it has not arrived in between that. In fact, it has arrived after the 2nd clock edge, there would not be any setup time failure.

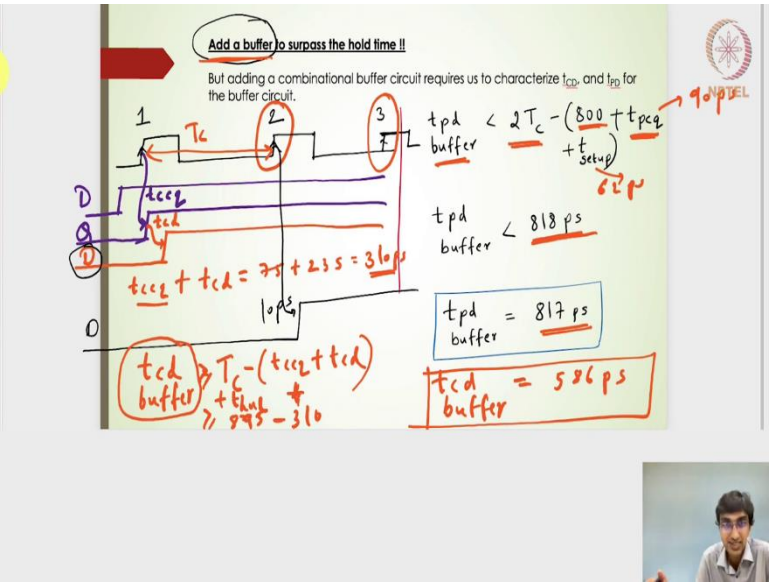
The only condition is, the output generated by the late bypass mux or the addition of this adder, muxes it is going to be collected by it is going to be captured by the flop at the output side, not in the 2nd clock edge now, it will be captured only in the 3rd clock edge.

Somewhere here it is going to get captured, this will be my 3rd edge. For the 3rd edge, if I consider the 62ps of the setup time it has arrived much much earlier, than the 3rd clock

edge and then the 62ps of that. There would not be any setup time failures. Now, the question is, whether there will be any hold time failures.

I have written here that the hold time is violated, but I have taken the propagation delays and then saying that, it is coming in between the 5ps of the hold time, the hold time is only 10ps, it is with respect to the 2nd clock edge. In that sense, it is coming in between, the hold time of 10ps here. I will just write it here, it is the 10ps which is the hold time. It is arriving in between that which is not correct, there will be an hold time violation.

(Refer Slide Time: 12:27)



There is another condition here, if I consider the only the contamination delay. If I consider the only the contamination delay. Let me draw the clock here first. I am drawing a very smaller clock this time. This is the 1st edge this is the 2nd edge. I am going to write it as 1 and then this is the 2nd and I want t_{ccq}.

Let us say that the D signal has arrived much earlier, this is the first input to the flop number 1 of the flop input and then the output is getting captured. Let us say that output is getting captured here just soon after the 1st clock edge. This will be my contaminated ccq contaminated clock to Q and this will be the output of the flop. After that, there will be an added block, there will be 3 maxes block.

I am going to say that all of them is a contaminated output. I am going to consider this as the contamination t_{cd} which is nothing but the contaminated delay of all the 4

combinatorial block blocks. If I consider this t_{ccq} + t_{cd}. Let me try to find out that particular value, which we have anyways calculated in the earlier slides.

Let me go back to the earlier slide number 2 and then say that this value was 75ps and 235ps. I am coming back to here. It is nothing but, 75ps + 235 = 310ps, with respect to the 1st clock edge, the output of the last mux or rather the output of all the 5 combinatorial blocks, is coming after 310ps.

With respect to the 1st clock edge, the t hold is only 10ps, its sufficiently away from the 10ps. The 310ps is sufficiently away from the 10ps. But the problem here is, it is now easily available. The output here, which goes to the input to the flop, it is now easily available and can get captured in the 2nd clock edge.

Whereas, the propagated the steady state signal, will not be captured here it is likely to captured after the 2nd clock edge, that means that, it will capture in the 3rd clock edge, which is incorrect. What we have is a propagation delay output to be captured in the 3rd clock edge. But the contaminated output is getting captured in the 2nd clock edge, which is inconsistent.

What we need to do is add a buffer. we need to add a buffer and that is what we are saying here we need to add a buffer after or in between in the 4 combinatorial blocks. So, that it surpasses the hold time with respect to the 2nd clock edge, with respect to the 2nd clock edge means the 10ps after the 2nd clock edge.

If we get, this D signal delayed, till the 10ps of the 2nd clock edge, if you can somehow delay that by adding a buffer and then choosing the contamination delay of that particular buffer. That the output is 10ps after the 2nd clock edge, we can be rest assured that it will not be captured the contaminated output will not be captured in the 2nd clock edge whereas, it will go and it will get captured in the 3rd clock edge. In the 3rd clock edge anyways the propagated signal, the steady state output will get captured.

The contaminated output will in fact not be, creating any problems because in the 3rd clock edge we know that, even if it is a contaminated output which is not captured in the 1st, which is not captured in the 2nd and whatever is the glitches, it will not be captured in the 1st and 2nd rather the steady state values will be captured in the 3rd clock edge.

In that sense, what should be contamination delay of the buffer and it is very simple to calculate because we know that t_{ccq} and t_{cd} will be there, for the contaminated output. It will be nothing but, the clock time period here, Tc which is nothing but, 885ps.

$$t_{\substack{cd \\ buffer}} \geq T_c - (t_{ccq} + t_{cd} + t_{hold})$$

This ensure that if I add a block after or in between the 4 combinatorial blocks it should ensure that the output of that particular output which goes as an input to the flop design should arrive only after the t hold of the 2nd clock edge. That it will not be captured in the 2nd clock edge, it will be captured only in the 3rd clock edge.

In that sense it will be nothing but 885 + t_{hold} time which is nothing but, 10ps. The 895 - 310ps, it turns out to be. In fact, it should be greater than or equal to Tc which is 885 + 10ps. The 895ps - 310ps.

In that sense it will be, 85 here and 585ps, you can choose 590ps as or we can design the buffer such that, the t_{cd} the contamination delay of the buffer is 590ps. So, that it can easily avoid, the 2nd clock edge and it can get captured only in the 3rd clock edge, where the steady state the value will be captured. Now, the question is what should be the propagation delay of that particular buffer which we have used.

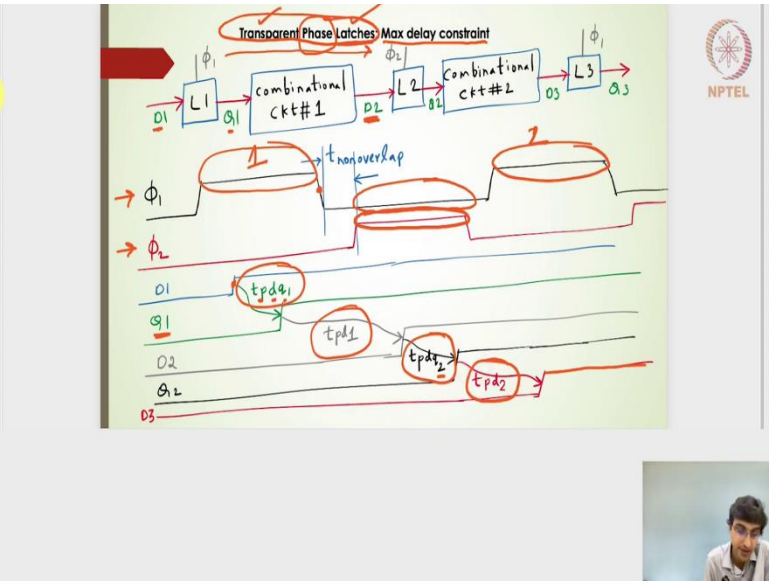
We the hold time failures is avoided not only in the 1st clock edge, but also in the 2nd clock edge. The t_{pd} of the buffer, if I look into the previous slide, the t_{pd} of the buffer says that, the t_{pd} = 590 + 60. The 590 + 60 = 650 + 150 = 800ps + 90 = 890ps and if I add a buffer at somewhere here. This buffer the maximum delay, this buffer can go is make sure that the 3rd clock edge and the output of this buffer, should be arriving at least the t_{detip} time for the 3rd clock edge. Whenever the 3rd clock edge comes, then the t setup time with respect to the t setup time of the flop design it should be available and that it will be captured in the 3rd clock edge.

Going back, this propagation delay of the buffer should be nothing but, the 1st clock edge to the 3rd clock edge. So, that we will have twice the time period of the clock, minus of the 800 which is nothing but the propagation delay of all the combinatorial blocks, t_{pcq} that is coming from the propagation of 1st clock to the Q output of the flop and the t setup for the 3rd clock duration.

It should be smaller than 818ps t_{setup} is nothing but, 62ps. This is 62ps and t_{pcq} was nothing but 90ps. If I do the calculation, I will get 818ps, that should be the maximum one. The t_{pd} buffer we can have it as 817ps and t_{cd} could be.

I am going to write here as t_{cd} of buffer, which we are going to add after the 4th combinatorial circuit, could be nothing but 5 and then 8 and 5. I can choose 586ps. We need to design a buffer with its propagation delay as 817ps and its contamination delay around 586ps. So, that the contaminated output as is avoided in the 2nd capturing in the 2nd clock edge, as well as the propagation delay which is the steady state output will be captured only in the 3rd clock edge. Hope this is clear to everyone. This is what we are going to follow in our designs to the sequential elements. whether it is a flip-flop designs along with the combinatorial circuit designs or the latch designs or the phase latch designs.

(Refer Slide Time: 22:05)



Let us proceed further to the next kind of the sequential elements, which is called as the latches. Generally, latches are not utilized in terms of how we have utilized the flops, the flops are generally utilized for the clock edges with respect to the clock edges. We have the combinatorial circuit design element, within the 2 subsequent flop elements.

Such that the output of the combinatorial circuit 1 will be captured in the 2nd clock edge and then and so on, right. But in the terms of the latches the latches are actually designed in terms of the levels it is working or operating in terms of the levels. If I design a clock here the latch 1 here is connected to the positive level of the clock, that means, that it will

work in the positive level of the clock and the L3 latches are working again in the positive level of the clock.

All the odd number of latches L1, L3, L5 those work in the positive level of the clock. Whereas, the L2, L4 and L6 are all connected to the clock number 2 which is nothing but kind of derived from the clock 1, but it is kind of phase shifted. The clock 1 and clock 2 are kind of non overlapping in this particular design or in this particular representation.

The clock 2 is connected such that, when the clock 1 is high, the clock 2 is low and when during some portion of the clock 1, during low level of the clock 1 the clock 2 will be high. The latch 2, latch 4, latch 6 and all the even number of latches, operates in the high level of the ϕ 2 clock. What it really means is it actually operates in the sum portion majority portion of the low level of the clock 1.

What we normally use the latches is the 1 combinatorial circuit will operate in the high level of the clock 1. The 2nd combinatorial circuit operates in the low level some portion of the low level of the clock 1 and then the 3rd latch will or the 3rd combinatorial circuit works in the again the high level of the clock 1 and then the 4th combinatorial circuit will work only in the some portion of the low level of the clock 1.

In terms of the latches and then with respect to the flip-flop, the differences because the flip-flops operate only in the clock edges, the combinatorial circuits output we make sure that it is captured in the clock edges, in the subsequent clock edges. Whereas, here because of the latch design, we have an ability to have the combinatorial circuit, maybe a faster combinatorial circuit to be captured in the levels positive levels as well as the negative levels of the clocks.

To do that, what we do is we have the clock 2 representation which is nothing but, coming from the clock 1 and with some time non-overlap. We will see what is the significance of this time known non-overlap in the later slide. What it really means is the ϕ 2 is derived from ϕ 1. The clock 2 is derived from clock 1, but clock 2 will operate or will be high only during the majority portion of the clock 1 being low.

This is about the discussion about the clocks, for this combinatorial circuit designs. Now, let us say that this is also called as the transparent phase latches. The reason is it is transparent it is CMOS latches works in the transparent mode as well as in the opaque

mode, but we are interested in only capturing the output only in the transparent mode, that is why it is called as the transparent and then it is a latch design and it is called as a phase latches.

The reason is nothing but, if you look into clock 1 and clock 2, clock 2 is its operating only when the clock 1 is low. It is kind of 180 out of phase. In that sense it is called as the phase latches. Now, let us say that the D1 signal is coming here, to the latch 1 and remember that the latch 1 will operate when the clock 1 is high.

Even if the D1 signal comes before the clock or after the clock level, after this particular rising edge does, not really matter, because even if it comes after the rising edge, during this entire portion of the clock level being high, it is likely to get captured. The farthest, the D1 signal can come with respect to the clock level high will be nothing but, the t_{setup} time before the clock goes to low.

The t setup time before the falling edge of this clock. Until then we have the high level of the clock. It should be able to capture it easily. If the D1 signal is getting high in between this particular clock level high the Q1 which is nothing but output of the latch 1 is going to get captured here with the propagation D to Q.

That is why I have written D to Q1, 1 represents the latch 1. Now, remember that here it is the propagation delay of the D to Q and not the propagation delay of clock to Q. Remember in the flip-flop designs the D used to come before it used to be ready before the clock edge going high and then the output was actually determined whenever the clock does the rising edge of the transition and that is why the output is with respect to the clock going high to the output.

Here, it is actually because the clock is already available, whenever the D comes the output Q1 will be high in this particular case and that is why we say that it is with respect to the D signal and not with respect to the clock signal, that is the output of the Q1. Then it goes to the combinatorial circuit which will have some propagation delay and then it will give the D2 signal.

I am considering the propagation delay of the combinatorial circuit 1 as tpd1 notice that this particular title is the maximum delay constraints. We are considering the maximum delay we can have it for the combinatorial circuit 1 and 2.

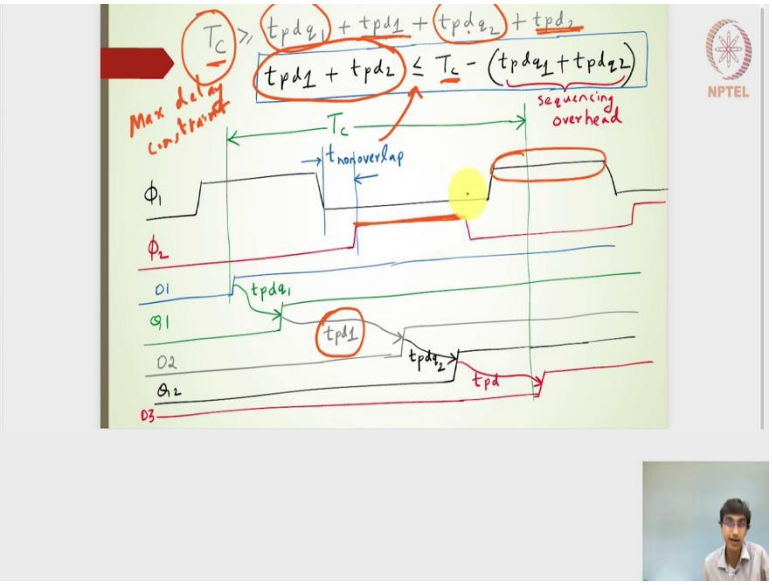
We can design the combinatorial circuit 1 and 2 such that it will have the maximum delay, that means that, we will have to consider the propagation delays of the combinatorial circuit, we will have to consider the propagation delays of the latch designs L1 and L2 and L3. In this case, the D2 signal comes after the propagation a delay of the 1st combinatorial circuit.

That is the D2 signal and after that, the latch 2 is going to get captured and remember that the output or rather the D2 signal is arriving during this ϕ 2 level being high. The ϕ 2 level being high that is when it arrives and then we will get the Q2 signal, which is now with respect to the propagation delay of the D, data D for the latch 2 and then the output of Q and then the 2 subscript is there, that it represents the latch 2 output.

This is the propagation delay of the D input to the Q output for the 2nd latch and then it goes to the combinatorial circuit number 2 goes to the next L3 latch which is kind of driven by the clock 1. I will have the D3 output which is now coming into the high level of the clock which is now sufficiently there, during the high level of the clock 1. Of course, in the 2nd clock level, this is the 1st clock level, this is the 2nd clock of the ϕ 1.

If I consider this particular propagated delay, now because of the sufficiently high levels of the clock, I can also you know if at all I have to design the clock signal here. Let us say that the ϕ 1 clock time period is not given to us, the ϕ 2 clock time period is not given to us and if we were to design the clock time period, then what we are likely to do is, we will calculate this t_{pdQ1}, we will calculate the t propagation delay 1 of the combinatorial circuit 1 t_{pdq} of the latch 2 designs. The t_{pd} which is nothing but, the propagation delay of the combinatorial circuit number 2. Add them up and say that will be anything those things will be my clock time period.

(Refer Slide Time: 31:16)



$$T_c \geq t_{pdq1} + t_{pd1} + t_{pdq2} + t_{pd2}$$

If at all I were to design the clock time period. What I will choose is, all this 4 components of the propagation delay and then anything greater than that will be my time period.

Now, if we were to let say that if we have the clock time period and if we were to design the combinatorial circuit blocks, then what should be the propagation delay of those 2 combinatorial blocks.

$$t_{pd1} + t_{pd2} \leq T_c - (t_{pdq1} + t_{pdq2})$$

That is why it is called as the sequencing overhead. If we were to use the sequential elements to time my outputs of the combinatorial circuit, then there will be an overhead and in this particular latch design the overhead comes in the form of t_{pdq1} and t_{pdq2}, that is why it is called as a Sequencing Overhead. Hope this is clear to everyone. This will be my maximum delay constraints.

The maximum delay constraints for the phase latches, where we want. Let me complete this writing, maximum delay constraint, this is especially for the phase latches where we want the 1 part of the combinatorial circuit to work in the high level of the clock and then the 2nd part of the combinatorial circuits to work in the low level of the clock and that is

why we have used 2 phase clock signals and then 2 phase latches. The 1 clock given to the latch number 1, latch number 3, latch number 5 and all the odd latches.

Then the 180 degree phase shifted clock is provided to the latch number 2, latch number 4, latch number 6 and so on, all the even number of latches. If suppose that we have to design the combinatorial circuits, then the maximum delay, if at all the time period is given then it should be like this, $T_c - (t_{pdq1} + t_{pdq2})$. The tnon-overlap time is does not come into the picture of this particular expression at all.

The tnon-overlap although it is there, it does not come because we have the t_{pd1} which is designed in such a way that, it comes during the clock level being high. The t_{pd} are designed to in a way that the output of the output of the combinatorial circuit's are coming during the clock levels being high. In this case the clock level being high and in this case the clock level being high.