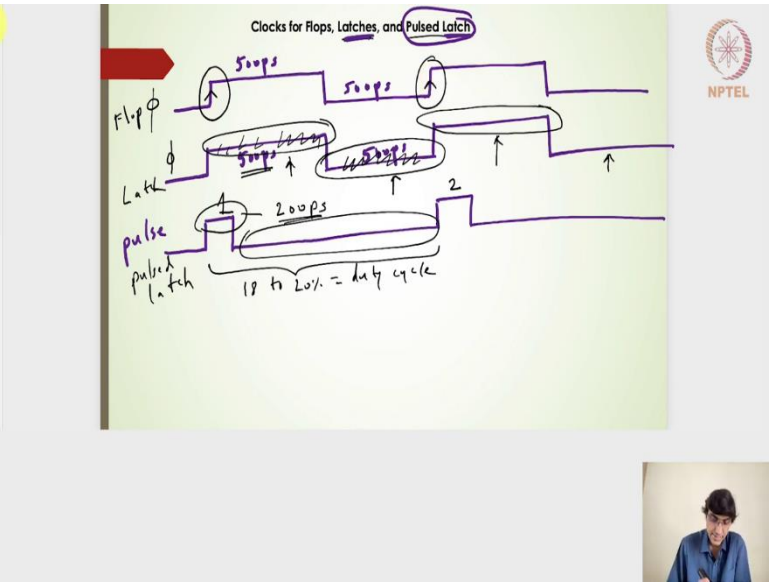
Design and Analysis of VLSI Subsystems
Dr. Madhav Rao
Department of Electronics and Communication Engineering
International Institute of Information Technology, Bangalore

Lecture - 79
Static Timing Analysis - Part 03

In this particular lecture most of the time I think we will be talking about the pulse latch. Pulse latch is another kind of a latch or another kind of a sequential elements, where the level of the pulse is not as high as that of the clock signals. In some of the digital circuit designing or the VLSI design we will require this pulse latch to be incorporated for getting the right output.

(Refer Slide Time: 00:47)



Moving ahead, what I will do is, in this particular slide I am just going to talk about the clocks that has been used for all the three sequential elements. Let us get started with the flops design, for the flop design we apply a clock here and then generally, the clock is of particular frequency.

For the 65nm technology we will generally use a clock of frequency 1Ghz. I am assuming, I will be getting around 500ps here and 500ps for the clock level low. The flip flop will be working only during this particular portion of the clock. Only when it sees for a positive edge triggered flip flop, only when the clock does the transition from 0 to 1 we will get the

output. Now, whereas for an if latch, the latch is always defined with respect to the level of the clock. In that sense the latch will be designed.

If I draw the another clock here of the similar frequency, this is my 500ps and the latch here is likely to work only in the positive level of the clock. The latch will throw the output only during the, whenever the clock is at a level high for a high level of the latch design.

Whereas for the flip flops, it is always with respect to the transition. It is of the clock whereas, in this particular case it is the high the positive edge triggered clock. In the positive level of transition, you will get the output. In the subsequent clock transitions, that is where we will get the output in the flops whereas, in the subsequent clock level high we will get the latch output.

Now, the third one is a pulse latch. Now, the pulse latch what it really where it will be really useful is, if you want a design or if you want the combinatorial circuit output to be captured really at the edge of the clock. But we may not have that kind of an allowance or rather we may not have that kind of a tight constraints that it will be timed properly at the edge of the clock.

But at the same time I do not want to use the level of the clock because it is high for almost majority portion of the clock signal. In the sense 500ps and 500ps for the clock level high. We do not have that kind of a luxury or the allowance also and in that sense what we do is design a clock or a pulse signal right for a shorter duration and then and then it will be level low.

What it really means is I do not have that kind of an allowance here as that of the clock that has been provided to the latch. I do not have that much of a room here the high level, but at the same time we do not have that much of a tighter constraint where the output should be captured at the level. There is some kind of a room here in the pulse signal. That is why you know this particular pulse signal will be acting as a clock for the pulse latch right. If I pass this particular pulse to any kind of a latch signal 25 kind of a latch signal, then it will be called as the pulse latch.

The pulse latch is, if I design the latch part of it is very much similar to the latch any regular latch. The only the clock signal that has been provided to the latch design if it is a pulse signal then it becomes a pulse latch. Just to complete this particular portion we will say

that this is also a clock, but it will be useful for the latch design, this is also clock which will be useful for the flip flop designs. This is a pulse which will be useful for the pulse latch operation. Hope this is clear.

Just to give a little bit additional information is the pulse level here, note that this is not of the same duration as that of the 500ps here. It turns out to be that the duty cycle will be nothing but the pulse level high divided by the pulse level high and then the low. It is generally around 18 to 20% the duty cycle, roughly around if it is 100 or rather 1000ps. This particular time duration will be a maximum of 200ps, it will be less than that. Hope you know this is clear.

This signal we want to apply it into the latch and then get the pulse latch output. The other additional information here is whereas, I mean when I compare the pulse latch as well as the regular latch, the operations the combinatorial circuits output will be captured in the positive level of the latch. Then another combinatorial circuits will be captured in the negative level of the clock and then so on, because we have so much of room here in the latch design, so much of room in the clock that has been provided to the latch design. We want to utilize the entire portion of this particular clock level high and do not wait for the clock level getting high in the next cycle. We also use the clock level low here.

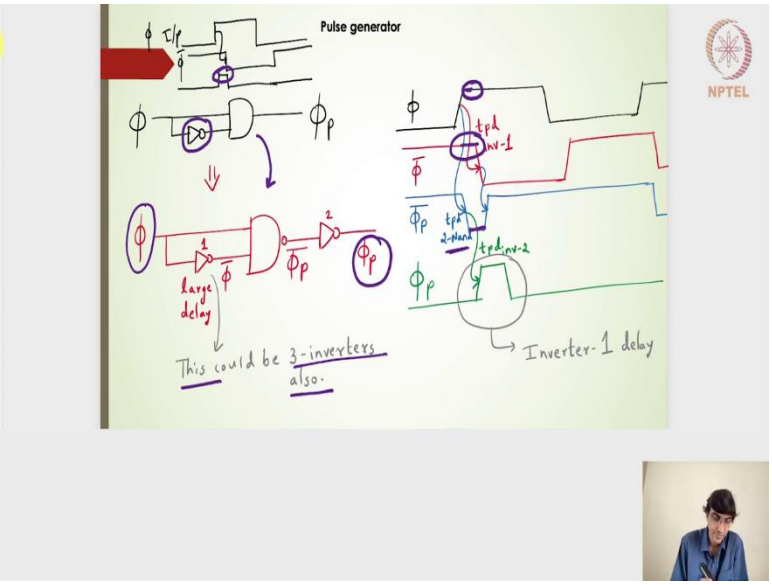
Most of the latches are actually used in pairs of and forms the phased transparent latch design. The combinatorial circuit number 1 will operate here and then the combinatorial circuit number 2 will operate here, combinatorial circuit number 3 will operate here, 4 will operate here and so on. That the combinatorial circuit output whatever this output will be getting captured in this low level here and that output will be provided to the combinatorial circuit number 2 and that is output will be captured in this particular level and then so on. Whereas, in the pulse latch this particular latch is, because it is it closely follows the flop designs also we do not have that kind of a tighter constraints.

But at the same time whatever is the combinatorial circuit data path, which we choose the similar thing will remain here for the pulse latch. If I am designing combinatorial circuit here it should be captured in this the output should be captured in this particular pulses in the pulse number 1 and then it will go to the subsequent combinatorial circuit and that should be captured in the second pulse. We do not utilize this low level of the pulse signal,

very very similar to that of not utilizing this entire portion of the clock that has been supplied to the flip flop designs.

Only in the latches because of this a majority of the room that has been available, we tend to use and capture the signal in the low level as well as apply the phase latch which will be able to capture this low level also. Here the latches they we always use it in pairs whereas, in the pulse latch it is always a single pulse latch and it will be operating or at every different pulses. Similar to that of the flip flop designs, where the flip flops is able to capture at every rising edge of the clock in this particular design. Hope this is clear to everyone.

(Refer Slide Time: 08:57)



Moving ahead, now how do we generate the pulse signal? One simple design is actually to use a clock and then feed it to an AND gate and the other input is nothing but the inverter. What really happens is, if I have an AND gate here and if this is the signal that has been provided to the AND gate and the inverted output of the input. This is the clock signal which is given to the input, the inverted of this clock signal will be nothing but high and then low and then high and then low, because of this delay in the inverter we are likely to get an output here, which will be nothing but a high for this particular duration and then low throughout. That is what we are trying to use this concept. Based on this inverters delay, we are likely to get this pulse duration or the pulse width.

Just in terms of the CMOS technology we cannot use the AND gate, we are using a NAND gate and then followed by the inverter here. We have a clock signal given and then the pulse signal is being generated from the clock signal. Remember that in a chip design you always have very few number of clocks source circuit called the clock generating circuit, giving a clock signal of ϕ.

From that we will be using the pulse signals or we will be generating the pulse signals. In this case the clock is provided to the NAND gate and then the inverted of the clock is given to another input to the NAND gate the output of that is go into the inverter. If I look into the timing diagram here very clearly it says the clock signal and then the $\overline{clock}$ signal which is provided as a second input to the NAND gate.

There will be a delay and that particular delay is due to this one particular inverter, inverter number 1. That is this inverter and then because of this being high and then this being high the output of the NAND gate will be low here, otherwise for all other cases the output of the NAND gate will be high.

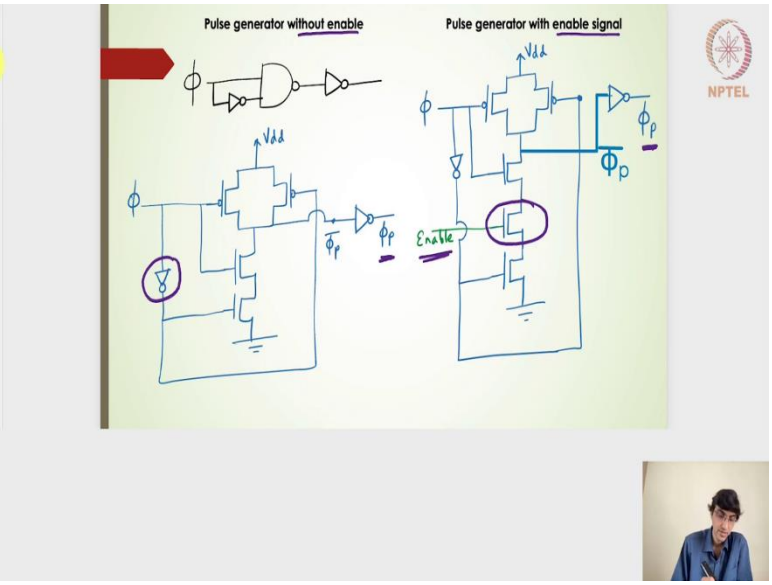
This will go to low with respect to the rising edge of this clock signal. The t_{pd} of the and then this will go high whenever the $\overline{\Phi}$ goes low. We will have this propagation delay with respect to the $\overline{\Phi}$ signal and this one is going low with respect to the propagation delay of the clock signal.

This particular duration is what is generating the clock signal. The output of the inverter here will be nothing but the inverted signal of this $\overline{\Phi}$ ϕp signal. We will get the pulse signal here and then the delay due to this, is due to the delay of the generating the pulse signal from the pulse bar signal, will be nothing but the delay of this particular inverter which is stated as t_{pd} inverter of 2.

This particular delay we call it as the output of the NAND gate, with respect to this clock signal we say that it is the t_{pd} of the 2-input NAND gate from the clock signal and then this one is the t_{pd} of the 2-input NAND gate coming from the $\overline{clock}$ signal. The pulse width here is actually determined by this particular overlapping once in ϕ and $\overline{\phi}$. That is arriving due to this particular the inverter which is generating this $\overline{\Phi}$ signal. The first inverter here, the pulse width is characterized by the inverters width of the inverter 1 dimensions and this could easily be also be the 3-inverters.

If I have a 3-inverters its likely to be an off the same dimensions which is there for the 1 inverter, then odd number of inverters is likely to give me in little bit more delay. Little bit more overlapping once here among the ϕ and $\overline{\phi}$ and thereby I will get a larger pulse width. For generating the pulse signal with a pulse width this simple circuit can be utilized comfortably.

(Refer Slide Time: 13:27)



Moving ahead, generally we will have a pulse generator signal with an enable signal. Without an enable signal is what we had seen now, the ϕ goes to the NAND gate the other input of the NAND gate is an inverted of the clock and then finally it goes to the inverter.

If I draw the schematic of this, it will be the NAND gate two of the PMOS transistors in parallel and two of the NMOS transistors in series and then the ϕ signal going to the PMOS and NMOS and $\overline{\phi}$ signal which is going to the PMOS and then NMOS. Then the output of this is going to give, it will be provided to an inverter to generate the pulse signal here.

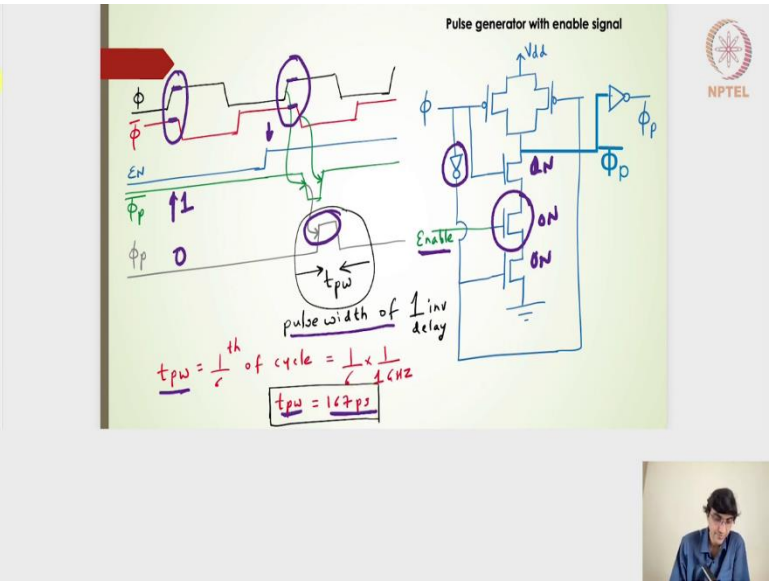
The pulse width is always determined or characterized by this inverters design. Whatever is the inverters delay is likely to give me the pulse width of value. Now, with an enable signal, if I want to add the enable signal ensuring that only when the enable is high that is when I will get the pulse signal here, pulse signal going high. Otherwise, the pulse signal will be low throughout.

For that particular case, an enable signal is provided to one additional transistor here in the pull down circuit of the 2-input NAND gate. This particular part of the circuit is very very similar to this particular part of the circuit. The only thing is the enable foot transistor or the enable additional transistor is added in the pull down circuit of the 2-input NAND gate.

In the pull down circuit here we have two transistors, here it is actually three transistors where one of the transistors is enable. If the enable is not 1, then the pull down circuit will not work, only when the enable is 1 we will have the pull down circuit. This ϕp the complement of pulse signal or the output will be able to discharge to the ground.

Unless that happens or unless enable is 1, if the enable is not 1 then the $\overline{\phi p}$ signal will always be considered to be in the previous state or to be high. Only when it is 1, this will go down assuming that ϕ and the ϕ and $\overline{\phi}$ are both 1, then it will go down and then the ϕp will be generated.

(Refer Slide Time: 15:48)



Hope that is clear to just to reiterate that in the timing analysis. We have the clock signal here and then the inverted clock which is we have the overlapping once here, because of the first inverter delay. Let us say the enable signal is made only high only in this particular instance, all other instant is low and then from this instance it is high.

Even if there is an overlap here we will not get the ϕ p going down, the ϕ p should go down right that does not happen because this particular transistor is off because enable is 0. Here

it will stay to 1, the ϕp signal and for $\overline{\phi p}$ signal and ϕ p the pulse signal will be 0 here, the moment the enable becomes one here this overlapping ones here we generated due to the inverter, inverter number 1 here. This particular overlapping ones will generate a low $\overline{\phi p}$ signal.

Now because the enable is high this particular transistor is on, this is also on this is also on. I am going to write it on and then this is also on. The $\overline{\phi p}$ signal will go down that is what is happening here and then once it goes back to low then it will go back to high. We will get this $\overline{\phi p}$ signal and then the pulse signal will be generated, with the pulse width here which is of course, determined by the first inverter.

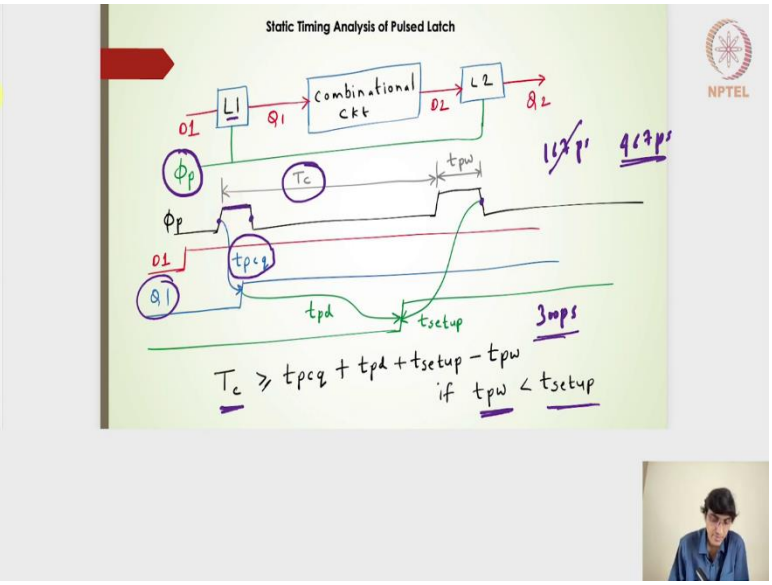
This is a very simple circuit providing the pulse signals only when the enable signal is high. An enable control signal is also been accommodated in this particular circuit. The pulse width of is nothing but of 1 inverter delay, in this particular case this is the inverter delay we are talking about.

Generally, it is one-sixth of the cycle, the pulse width is one-sixth of the cycle. If it is 1, the time period or the pulse width will be one-sixth multiplied by 1 giga 1 multiplied by 1 by 1Ghz.

$$t_{pw} = \frac{1}{6}\frac{1}{1\text{Ghz}} = 167\text{ps}$$

What I had said earlier was around a duty cycle of 18 to 25% so that is still valid. The maximum is 20% duty cycle, here the pulse width is 1/6 turns out to be 16% or something like that. Generally, we will have a pulse width of around 167ps for a clock frequency of 1Ghz. Especially in a 65nm technology node.

(Refer Slide Time: 18:24)



Moving ahead, now let us take a look into the static timing analysis of the pulse latch design. Very similar to how we have done for the flops design. We had a flop 1 here and flop 2 here in between the combinatorial circuits and we wanted the output to be captured in the subsequent clock cycles.

Here the same thing will be there, just that it is a pulse latch. A pulse signal is provided to the clock of this particular latch now, it is not a flop anymore, it is a latch design, but it is a pulse latch. A pulse signal is provided to the clock of this particular latch design.

We have a latch design L1 and L2 and the pulse signal is provided here. What we really want is the output to be time. The output should be captured in the second pulse signal and then third pulse signal and fourth pulse signal and then so on. Let us say that the D1 signal has arrived much much earlier than that of the pulse signal going high.

In fact, it is sufficiently arrived earlier than that of the t_{setup} time which starts from this going low remember that for the latch design, for a latch design the setup time is coming even for a latch design of a positive level of the latch, we will have the t_{setup} time actually coming from the falling edge, because it can be easily captured during the high level of the clock, but only when it does the transition it should be made available at least t_{setup} time before the clock level going low. Even for the latch design pulse latch design. The t_{setup} s starting from somewhere here, its defined from the negative edge of the pulse signals.

Anyways the D1 signal has arrived much earlier. In that sense the Q1 signal the output of the latch 1 will be generated the moment the pulse signal goes high. It is with respect to the propagation or delay of the clock to the Q for the pulse signal, for this particular latch design.

Once you have the Q1 output, then it goes to the combinatorial circuit the output of the combinatorial circuit will be available after the propagation delay of the combinatorial circuit. Let us say that it arrives much much earlier than the t setup time here of these pulse signal for the pulse latch.

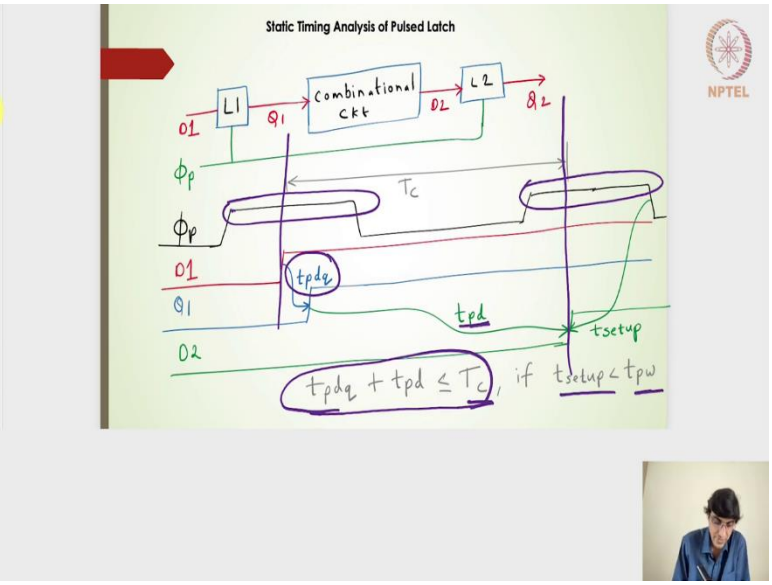
Remember that the setup time is always with respect to a negative edge of the pulse of the pulse signals, for a positive level of the pulse latch. If I have this particular timing diagram, I can easily find out what should be the clock time period. The clock time period will be nothing but this particular component plus this particular component, the clock time period should be at least greater than or equal to this particular component, this particular component, this particular component minus this particular component, which is the pulse width.

$$T_c \geq t_{pcq} + t_{pd} + t_{setup} - t_{pw}$$

Here the thing is we have used $t_{pw} < t_{setup}$. This t_{pw} the pulse width duration of whatever 167ps and t_{setup} is much much higher than that. In that case I will have this particular expression valid. Now what if I have let us say that t_{pw} = 167ps and the t_{setup} = 300ps then this particular expression is valid.

What if I have the t_{pw} instead of 167ps, I have 467ps, it is actually touching to the very very close to that of the clock level being high used for the latch design. The 467ps and I have the t setup time as 300ps then what happens?

(Refer Slide Time: 22:16)



In that sense we will have the same circuit diagram, the same block diagrams, where the latch combinatorial circuit is there and latch 1 and latch 2 are driven by these pulse signals. Instead of a small pulse here I will have a larger pulse width and in because I have a larger pulse width, now the D signal can actually arrive. In fact, even in the earlier case the D signal can actually arrive during that particular pulse width.

But let us say that because of the larger room here, the D1 signal arrives somewhere in between the clock level high of the pulse signal. The Q1 output which is nothing but the output of the latch 1, will be formed after the t propagation delay of D to Q. This is D to Q and in our earlier designs, let me just point it out it is a propagation clock to Q, because the D has arrived earlier, this clock will go high only when this or rather this output of the latch design will go high only when the clock goes high here, because D1 has arrived earlier, the Q1 output will depend on the whenever the clock goes high. The later signal whichever this is a later signal or whether D signal or a clock signal.

In this case the clock signal is the latter signal. We will have the Q1 going high the moment the clock goes high. Similarly in this particular case, between the clock and the D signals because the D is arriving late, I will get a Q1 signal with respect to the D signal. Now, I have the propagation D to Q here going high and that will be my output of the latch 1.

Then the output of the combinatorial circuit here which is the D2 signal and it will arrive after the propagation delay of the combinatorial circuit, and then it should be ensure that

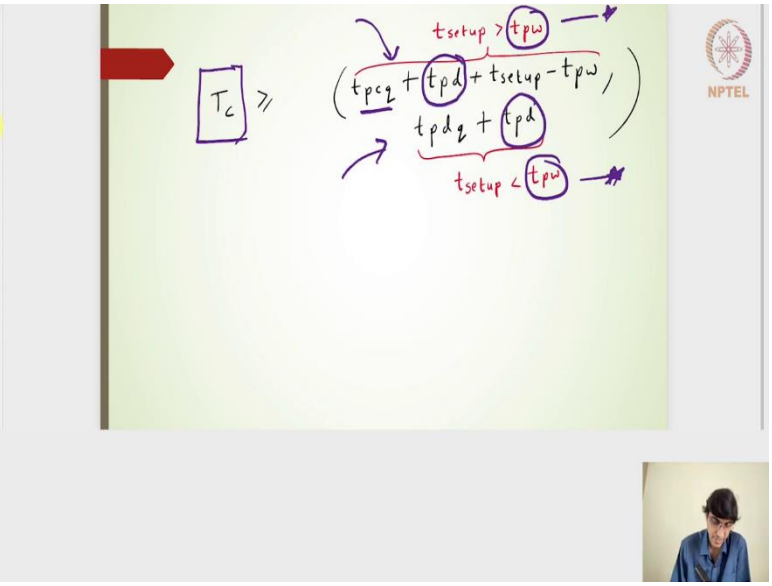
it is available just before the t setup time. Again there is a lot of room here to be captured here, it can arrive very very close to the t setup time. Then get captured into the latch 2 signal and then we will get a comfortable output of the output of the latch 2 signal. But if I look into this particular timing analysis it is very very clear that because the level here is or the pulse width is very very high here than that of the setup time.

I can have my clock design, I can have my clock design such that it is nothing but this component plus this component and I can design my clock in this two particular levels.

$$t_{pdq} + t_{pd} \leq T_c$$

If $t_{setup} < t_{pw}$

(Refer Slide Time: 25:28)



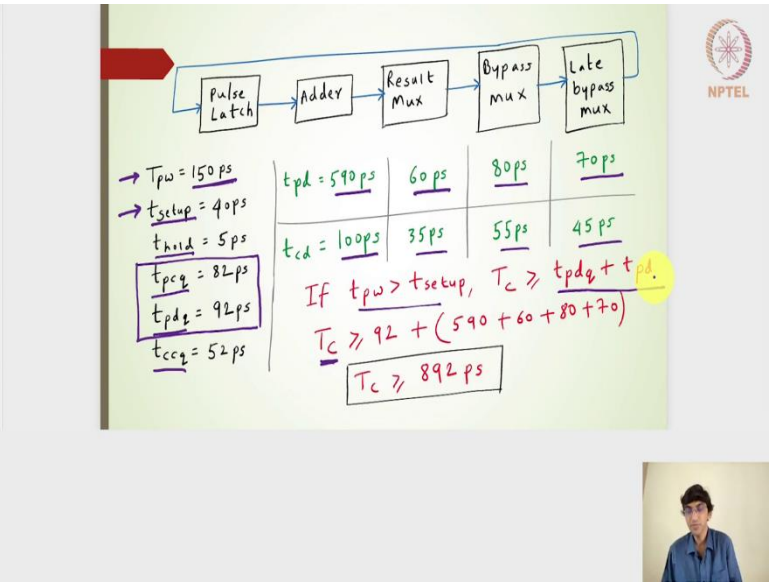
In a pulse latch we in fact have two such expressions, one is the expression for when the t_{setup} when the t_{pw} is actually low and another one is when the t_{pw} is high. When the t_{pw} is low with respect to the t_{setup}, it is nothing but the propagation clock to q, it works like on the flip flop design.

$$T_c \geq t_{pcq} + t_{pd} + t_{setup} - t_{pw},$$

$$t_{pdq} + t_{pd}$$

We have to look into this particular expressions and then whether conditions rather than the expression, if the t_{pw} is sufficiently high, then actually we can use this particular expression. If the t_{pw} is slow, then we will have to use this particular expression to design the clock signal 0r in this case the pulse signal or the pulse signal time period. The clock time period as well as the pulse signal time period are the same for the pulse latch.

(Refer Slide Time: 26:39)



Moving ahead. Let us take an example here, a very kind of a very chip level example. We have the pulse latch here, again given to the similar four combinatorial circuits of adder mux's, mux's, mux's and then fed back into the pulse latch here.

The pulse width is 150ps, that has been characterized that is the pulse width signal has been provided to the pulse latch and the pulse latch is characterized with this setup time, hold time, t_{pcq}, t_{pdq} and t_{ccq}. The t_{pcq} = 82ps and t_{pdq} = 92ps is given, we will have a deeper look into this later.

The t_{setup} = 40 and t_{hold} = 5ps. The propagation delay of this particular 4 combinatorial circuits are given as 590, 60, 80, 70ps and the contamination delay is given as 100, 35, 55 and 45ps and we have looked into the maximum delay constraint. We will try to design the apply the maximum delay constraint.

In last two slides what we had seen was is the maximum delay constraint. It gives us the expression for the t_{pd} in both the expressions in both the configurations, whether if the

t_{pw} is low or high. The t_{pd} will always configure or will help in determining the maximum delay constraints.

Moving ahead, for this particular expression, can we identify what is the clock time period or the pulse time period, the minimum clock time period or the minimum pulse time period so as to have no setup time failures in this case? for that particular purpose, we will have to identify what configuration or what expression we have to use because in the pulse latch we have two different expressions.

Based on the t_{pw} and t_{setup}, we can be able to choose one of the expressions. If the t_{pw} is actually very very higher than t_{setup}, then we will use we will go with that of the closer to the latch expression. If it is smaller then we will go with the closer to the flop expression, turns out that the $t_{pw} > t_{setup}$.

We will use the latch expression which is very very simple.

$$T_c \geq t_{pdq} + t_{pd}$$

$$T_c \geq 92 + (590 + 60 + 80 + 70) = 892\text{ps}$$

The minimum clock time period should be at least be 892ps without any setup time problems.