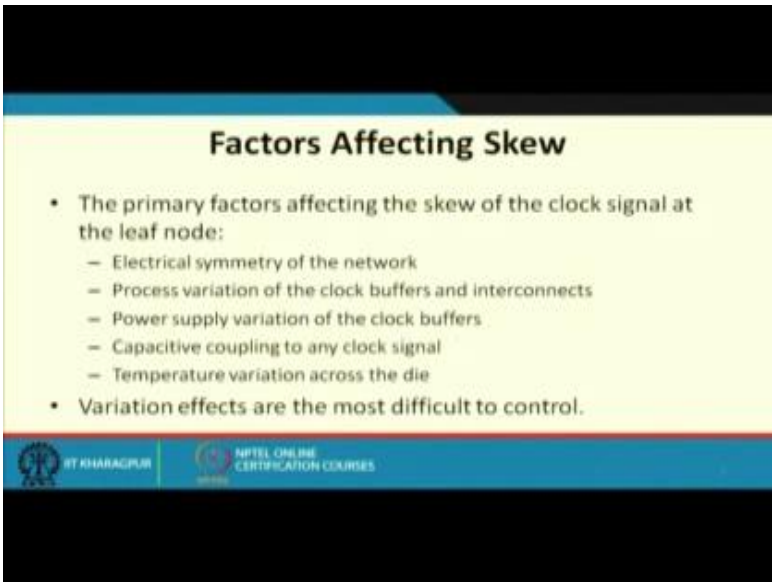
VLSI Physical Design
Prof. Indranil Sengupta
Department of Computer Science and Engineering
Indian Institute of Technology, Kharagpur

Lecture - 25
Clock Design (Part II)

So, let us continue a discussion on clock design so if we recall in our last lecture we talked about some of the parameters like skew jitter setup time and hold time and how they play a role in deciding the maximum clock cycle time during a so called pipe line kind of execution where there are storages stages with some logic circuitry in between.

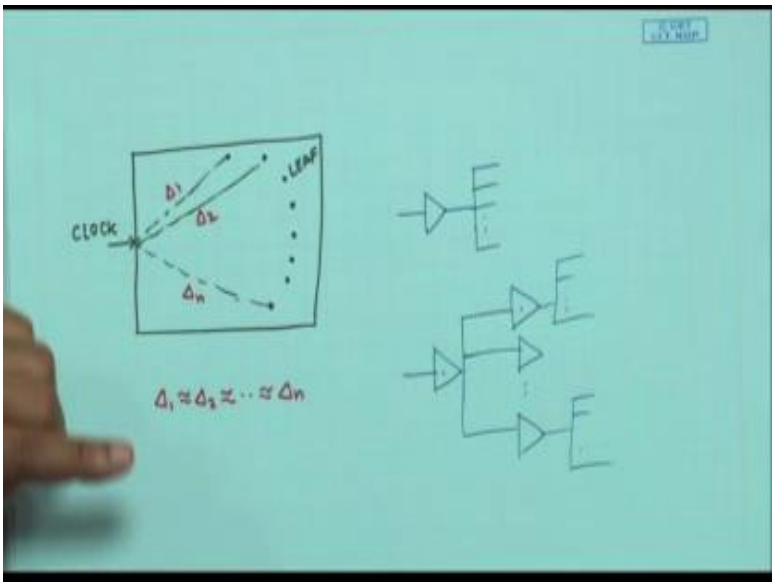
(Refer Slide Time: 00:58)



So, in this lecture we start today by trying to talk about the various factors that can affect the clock skew. So I shall be trying to explain this points one by one.

So, here as you can see I have use that term call leaf node. So let me just explain it once more.

(Refer Slide Time: 01:23)



So, I have a chip. I have an external clock signal that is coming from outside. And there are several terminal points where I have to feed my clock signal. Now these are called my leaf node or leaf points of my clock network. So in my clock network I have to connect this signal to each of these terminal points. I have to connect this to each of these points and these are called leaf nodes or leaf points. So the first factor that affects the skew is the electrical symmetry of the network. What does electrical symmetry mean? Electrical symmetry means that well I am generating the clock from somewhere I am sending it or distributing it to several leaf or terminal points. Now electrical symmetry means what would be the electrical delay of each of this paths. They will depend on number of factors because you know for any transmission line there will be resistive and the capacitive affect along the path and means also inductive effects.

So, if you lay out the wires in such a way that the delay on each of this wires will be approximately equal to the extent possible. Then you can achieve some kind of symmetry. This is what is meant by electrical symmetry of the network so the first point here electrical symmetry of network. So in this diagram whatever I have shown here if delta 1 is the delay of this path delta 2 is the delay of this path and delta in the delay of this path what I want is that these delays should all be approximately equal. This is a requirement that we have to satisfy and this delay is being in terms of electrical delay, where we are incorporating the resistive and capacitive affects.

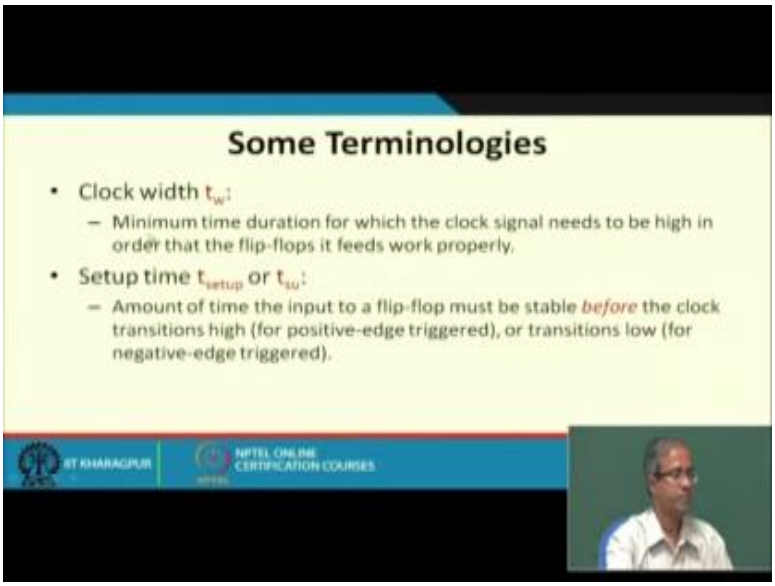
The next points are the process variation of the clock buffers and interconnects. Well, clock buffer is one thing that we shall again be coming back again and discuss in more detail, it means you see these clocks signals in one kind of a signal where there can be thousands of leaf nodes, there can be thousands leaf clocks, or even more. So I mean you cannot expect a single line to be connected directly to thousand other points. So you need some kind of a buffer either a current buffer which can be fit to a number of different clock points, or you can have some kind of a network of buffers like this so from here you can possibly feed. So there will be the interconnection lengths of course, the capacity rises the affects and also the delay of this buffers.

Both of this will be important here. So the process variation means when these are fabricated there will be some variations from one buffer to the other. So no 2 buffers can be exactly identically in terms of the delays. So because of that skews might be introduced. There is another property in VLSI circuits is that the delay of a circuit, delay of some kind of a sub circuits it depends on the power supply. So because of some again physical affect inside the chip parasite affect if there is a variation in the power supply, there will also be a plus minus variation in the delay of the resulting circuit that can again introduces skew.

So, the third point is the power supply variation of the clock buffered, then capacity of coupling for all the paths that are leading to the clock leaf nodes they may not be electrical identical. For some of them the capacity coupling affect with others lines we already running there can be more as compare to the other lines. So some of the lines might get delay it by greater amount then the other so this is one issues that needs to be addressed and of course, temperature variation across the die. Well if you do not design your chip in a proper way what might happen is that some part of your silicon might get heated more than the other part. Now you know that the physical parameters of this most devices they very quit significantly with changes in temperature.

So, if one part of chip is hotter than other part so may be the delay of one part will be different from the delay of the other part. So skews might get introduced because of this as well. So there are many such factors for which skews might be generated. And such variation affects are very difficult to control in some of them are almost impossible to control. This is I wanted to talk about.

(Refer Slide Time: 07:03)



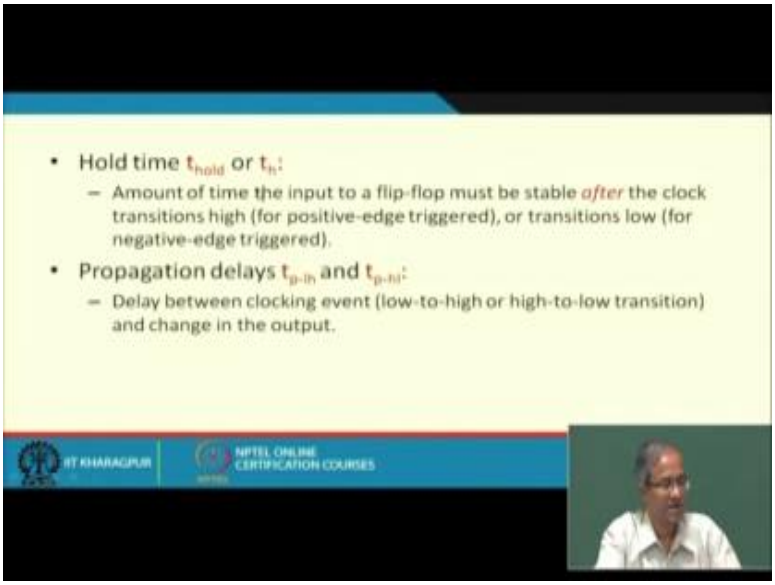
Well some of the terminologies is already we introduce this earlier in the last lecture, but let me reintroduce them again. So we sometimes talk about the clock width. Clock width is not equal to the clock period. Clock width means for how long time the clock remains high, because clock grows high again goes low, clock width is the time period for which the clock remains high, some of the flip flop may be having a minimum time requirement for this. Minimum time duration for which the clock signal needs to be high in order that the flip flop works properly, like you see you may be having a flip flop which is edge triggered, means that you expect whenever there is a 0 to 1 low to high I mean edge coming on the clock input it will have triggered and store the input value.

But what if the pulse that is apply to clock is so narrow that the flip flop is not able to register. For example, a one picosecond with pulse, that is usually not sufficient for the flip flop to triggered. That pulse must be of a minimum width only then the flip flop will be able to deduct the edge and work in a proper way. That is the clock width constraint. Then comes this setup time, this is typically denoted by either t setup or tsu both are same setup time.

I repeat. So, there is no harm in repeating this is the amount of time. The input to a flip flop must be stable before the clock edge appears. So for a positive edge triggered it is a high edge for a negative edge triggered it is a low edge. So before the edge comes what is the minimum amount of time I must hold my data to the input. That is the hold time

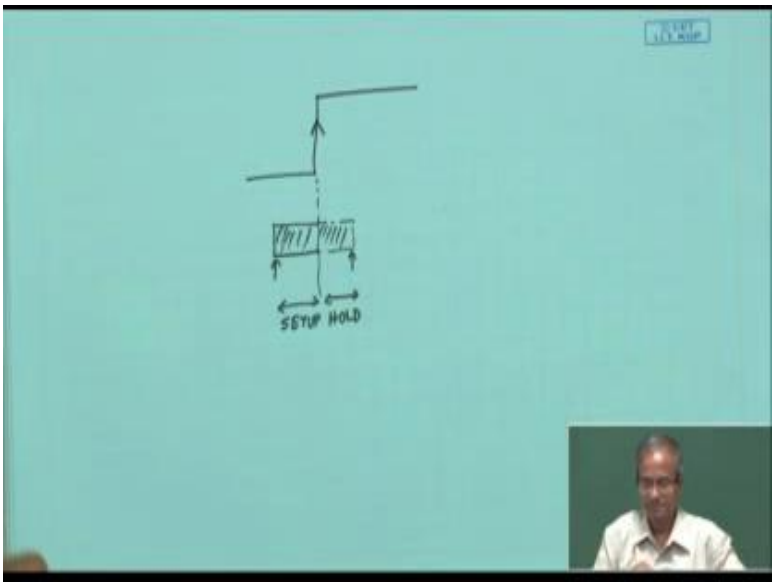
that is the so call setup time setup time is the time that must be given for me to apply the input data before the clock edge comes this is the setup time.

(Refer Slide Time: 09:26)



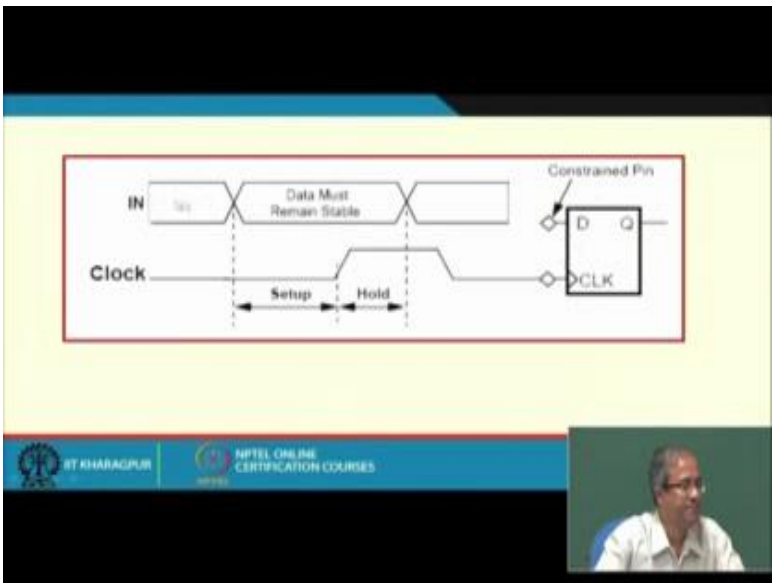
Similarly, comes the hold time which is denoted by t hold or th, t hold you recall this comes in to play after the clock edge comes. So this is the amount of time the input to the flip flop must be stable after the clock transitions high for low depending on the edge. So when there is a clock edge I must have a minimum time before the edge comes I must keep my data stable and also after the edge comes I must keep my data stable. It is not that whenever the edge comes I know that the think as happen and I am done. I remove my data immediately not that you can you can remove the data only after the hold time and before the clock edge comes you must apply the data a little before.

(Refer Slide Time: 10:20)



So, to summaries say for example, for leading edge flip flop my edge is coming here. This is my time so my input data must be ready starting from here and must be continue to be applied till here. This is my setup time. This is my hold time. This you should remember and the last thing are these, are the propagation delays. Propagation delays means staring from the clock edge, how much time is required for the output to change. So output means not the output of the flip flop, flip flop plus the logic the total delay propagation delay.

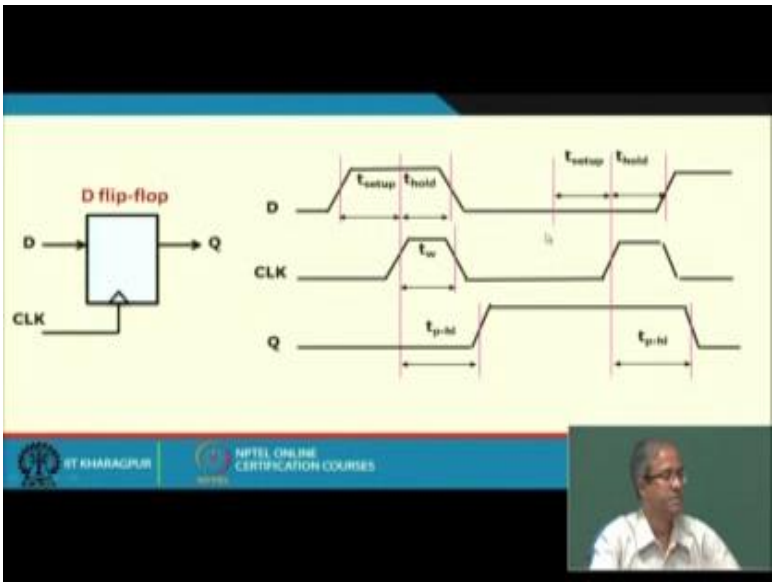
(Refer Slide Time: 11:38)



Starting from the clock edge what is the maximum propagation delays after which my output will become available. So it can be either low to high or high to low. So depending on the clock edge your working it can be tp low to high or tp high to low. There is the propagation delay, so setup hold as upset so here I am showing it diagrammatical you can see it very clearly.

So, so I have a flip flop. The input is coming this my input. Here I am showing one flip flop, but in a I means in a pipe line there will be a set of flip flop, so register that is why I am showing in the vector notation input data are coming this notation means the data is changing, so here my new data is coming. So after my new data has come to the input I must keep it stable for a minimum amount of time t set up, only after which my clock edge can come. And after my clock edge can come I must continue to keep my data stable for another time t hold. So my data must be kept stable a minimum time before the clock and also minimum time after the clock. This is the requirement of setup and hold time. So you can say that I must have to keep my data stable for a time which must be greater than t setup plus t hold with this time in constrained. Constrained pin means such timing constrained must have to be satisfied for the input data in right.

(Refer Slide Time: 13:07)

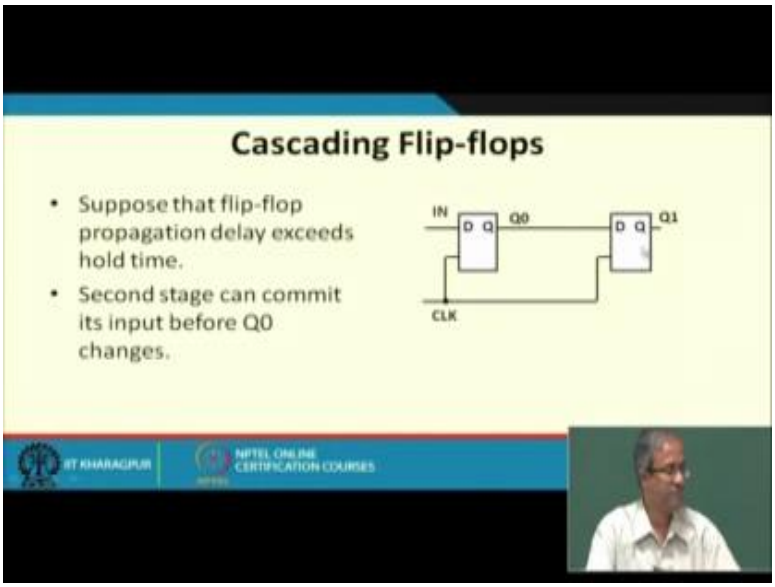


So, let us look at a scenario like this. This shows the different timing that I have just introduced, D flip flop. So the clock signal is coming I just said the width of the clock pulse that is your tw, and before the clock comes the D input must be applied minimum

of t setup time before and after the clock comes it must continue to be there minimum of t hold time, the sub setup and hold and this t propagation hl means after the clock edges comes after how much time this Q output is available. So after the D is applied the clock goes, there will be some t p hl may be here the key output is available the Q changes here.

So, this hold time is t p hl. So these are the times that we have discussed. Setup time hold time clock width and propagation delay high to low or low to high whatever depending on the polo relative of the clock.

(Refer Slide Time: 14:21)

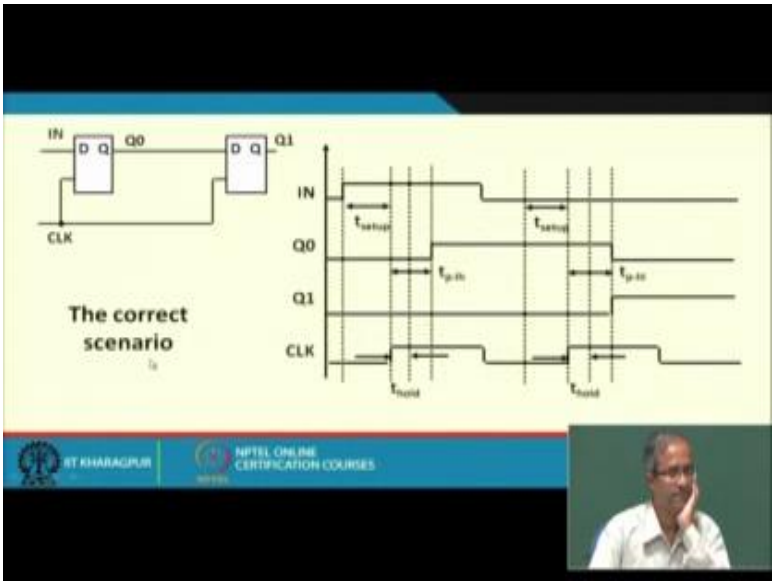


Now, if I knew cascading flip flops. There are lot of timing issues that we shall see now one by one. So if a flip flop propagation delay exceeds the hold time there can be a problem. Second stage can commit it is input before the output of the fist flip flop changes. See hold time is what you just recall once more hold time is the minimum time the data must be kept stable after the clock edge comes right. So the clock is being fed to both the flip flops.

So, after the clocks comes, so my input has to be kept stable for minimum that much time you think of the second one. So your propagation delays of the flip flop are something so after the clock comes after some delay, this skews role become stable and if that delay exceeds the hold time; that means, your second flip flop will be not be

getting that minimum time for which it can respond correctly. So there can be error in the timing.

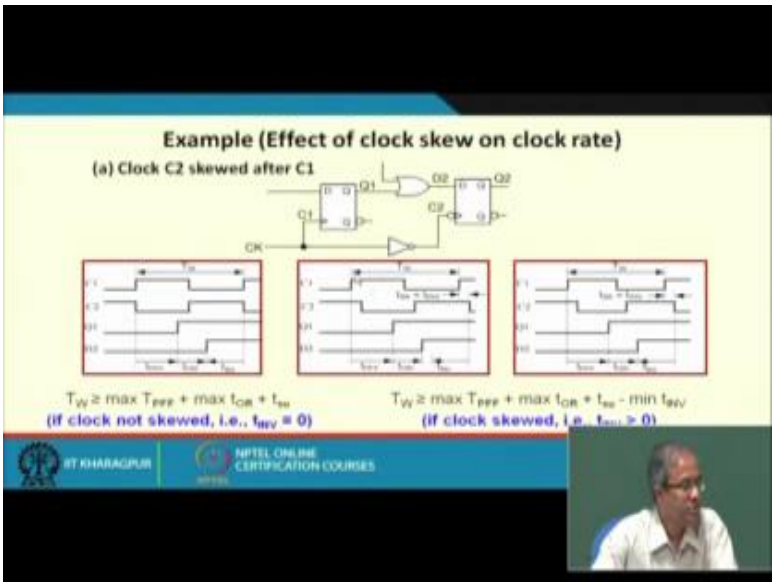
(Refer Slide Time: 15:49)



Now, this diagram shows the correct scenario. So what should be? You see that same 2 flip flop one feeding the other so the clock is shown in the last, this is the clocking. So here I am show this the input data, Input data is applied minimum of t setup time before the clock edge. And after the clock edge comes minimum of t hold time this input should be stable minimum of that much yes it is so it is. In fact, stable much more no problem there, but minimum it should be this much. And let say there is some circuitry in between or the propagation delay of this flip flop and this line this is t plh suppose this is tplh.

So, a input must be stable for this much time and only after this t plh Q0 will change state right. And again in the next clock when Q1 becomes Q0 become 1 in the next clock Q1 will become one again after the delay t phl t setup, and t phl. So in this way the flip flop will go on working right. This is the correct scenario of operation.

(Refer Slide Time: 17:16)



Now, let us see that when there is clock skew between 2 successive stages what are the means issues of problems that might take place let see it one by one some examples. Here I we have deliberately introduced a inverter here see there are 2 d flip flops the clock inputs are C1 C2. C1 is a positive edge triggered C2 is a negative edge triggered. So because of the inverter the same clock will be triggering both of them, but this inverter we introduce just to show that C2 will be delayed a little bit clock C2 is skewed after C1. And suppose here we have a commutation circuit simple or gates say. Let say what happens here. This is a scenario where here I am ignoring the delay of this inverter so I am assuming C1 and C2 are getting activated almost simultaneously 0 to 1 of C1 and 1 to 0 of C2 are happing together.

So, after that, there will be a delay of the flip flop, delay of this flip flop plus delay of the OR gate, delay of the OR gate plus setup time for the next flip flop. Before the next clock can come, so before the next clock can come D2 must be stable for this much time, say after this propagation delay and the delay of this or this output of D2 becomes 1 because q1 is as become 1 after the delay of or this output of war becomes 1 and this has to be kept stable for a minimum amount of t setup t has u only that it will work correctly.

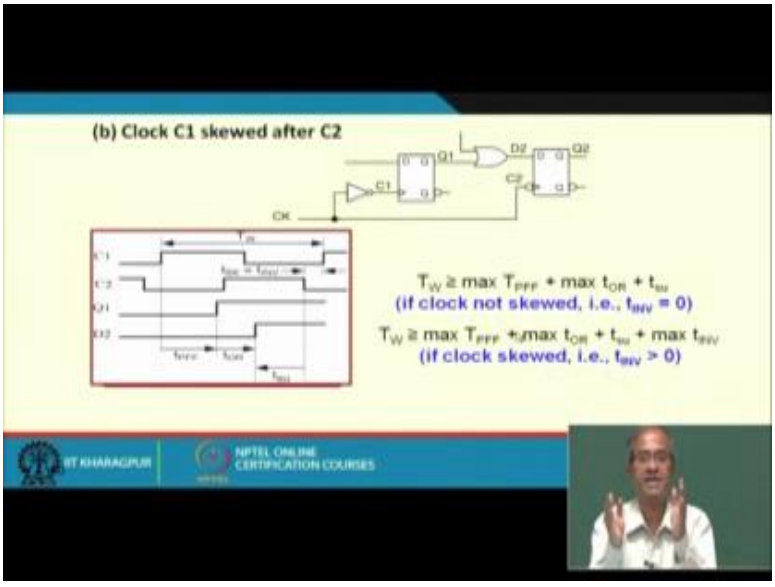
So the condition is this width of the pulse tw must be equal to maximum of flip flop propagation delay, maximum of this logic circuit delay this order I am saying plus setup of time, all these 3 things taken together. This is a scenario where I am ignoring the

inverter delay, if clock is not skewed, but here clock will be skewed. So what will happen is something like this. See because of the clock skew there will be delay here. C2 will get delayed. This will be your skew shown here. So clock going from 0 to 1 and clock 2 going from 1 to 0 there will be a delay, this is equal to t inverter, tiv and v so the total tw which was there, because of this this is actually getting subtracted.

Because any way you are needing to have some minimum delay, but already the clock is delayed. So after this delay you can see that this tw everything as getting delayed. So the first 1 whenever this clock comes these 3 delays are coming after that tff or tsu, but the second one start computing after this inverter delay because this clock edge comes after this t in delay.

So, actually for this second one the constrained will become max step plus minus minimum of t inverter. There are 2 scenario I am showing here. Here this is an OR gate this is setup of time where there are just equal this is the limiting case. So when clock is skewed you have to satisfy these conditions. This you have to know for this kind of circuit this will be the conditions.

(Refer Slide Time: 21:21)

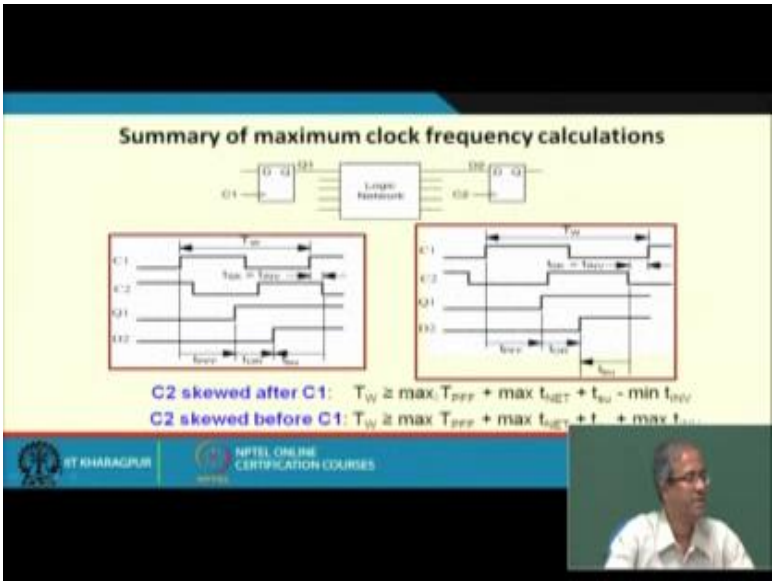


Next case reverse where C1 is delayed. So let us insert an inverter here and let us assume that whenever there is a positive edge coming on C1 and negative edge coming on C2 so both will have triggered together. Same way so this is the first scenario if the clock is not skewed tw will be greater than equal to some of these 3 the same way, but if the clock is

skewed like this was the previous thing clock is not skewed, but because of the clock skew you see C1 is getting delayed, C1 is getting delayed. So now, what will happen? This D2 so whatever is coming C2 edge that the whatever is coming here it is not getting total t setup time it is getting less D2 is getting stable here only. And it is getting stable for a time which is less than t setup. So there is an error situation here. So there are situations and the limiting case you can have this will be our inverter delay this much.

So, what is the condition we have to satisfy? This will get added now because this one is computing first and then you compute this, and because this is getting delayed everything is getting delayed so until this complete you cannot start this. So this inverter delay will get added here. So the idea is that if there a 2 flip flop one after the other if you delay the clock of the first one that is a bad situation, your total delay requirement increases, but if you delay the clock of the second one, that inverted delay gets minus negative subtracted. So you do not have to pay for an additional delay for the clock in that case. So these are some delay calculation you can use to actual estimate the clock cycle time and to improve or optimise one that.

(Refer Slide Time: 23:46)

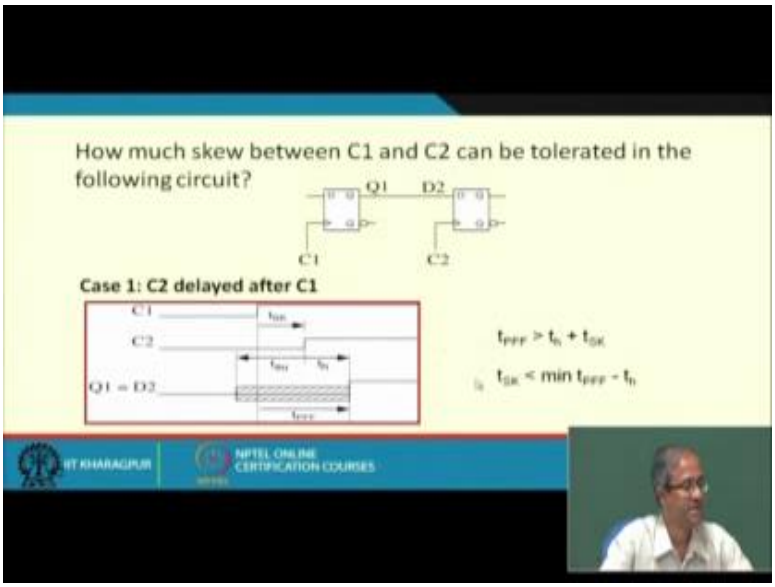


So, these are some simple calculations. Now to summaries maximum clock frequency calculation whatever you have seen, so in general will look at it like this. So I have a flip flop there is a logic network in between general there is another flip flop clock are C1 and C2. First scenario is C2 is skewed after c 1 that means, C2 getting delayed. So the

first case that you show so there your clock width time period must be greater than equal to the propagation delay of the flip flop, the maximum delay of the logic network setup time minus this delay. Suppose C2 is coming after inverter delay so I am calling tinv, but if c, but if it is the other round C1 is getting delayed then C2 is coming. So there is a inverter here in C1 then your T w is increasing which means your frequency would be decreasing.

So, your first case is better your frequency is improving, second case is worst your frequency is decreasing.

(Refer Slide Time: 25:14)

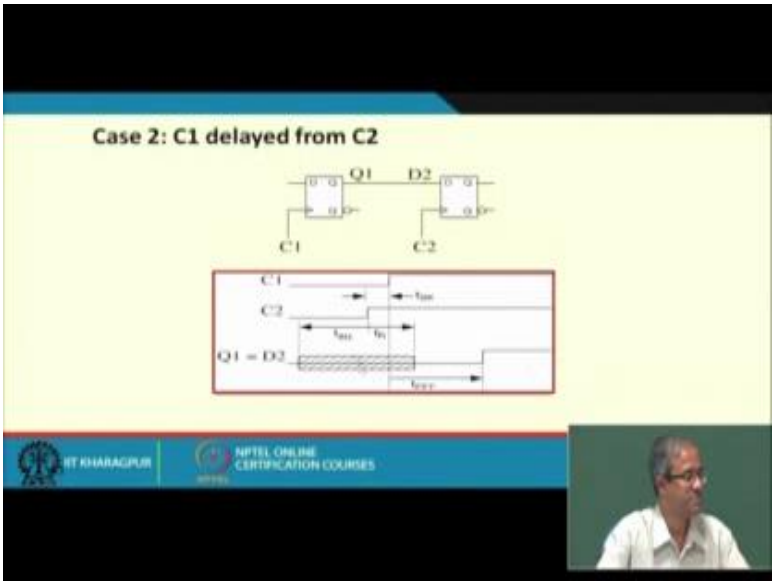


So, these are some criteria you can use you can follow to optimise on the clock frequency. Now there is another simple small example. Like you have a direct connection like this, from the output of one flip flop, to the other it is saying how much skew between C1 and C2 can be tolerated. Suppose C2 is delayed case 1 C2 comes after. So it is something like this C2 is coming after a delay of tsk right. So you have the setup time requirement, C1 edge minimum setup time and hold time after that this and suppose this is your propagation delay, before D2 reaches propagation delay of this line.

So, what happens is that that in the worst case after this delay what happens. This was without the delay, but if it getting delayed then everything is getting shifted to right, C2 is getting delayed further delay. This was little delayed the delay is increasing, so this edge will be touching this edge here. This is a limiting case, beyond this there will be an

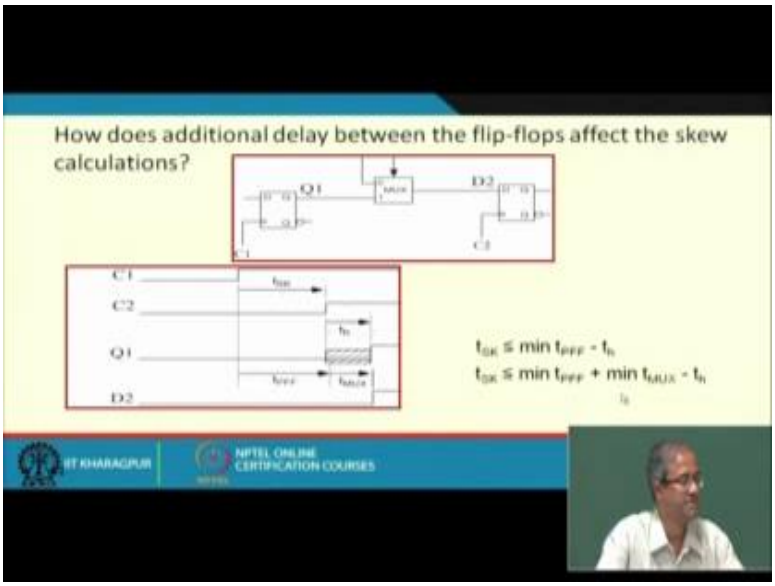
error if it is further reduced then the hold time requirement will be violated. So you get condition like this, tff should be greater than th plus tsk total. Your propagations delay should be should not be less than this among this 2, and of course, tsk should be propagation delay minus th should be less than that skew. You can check if this 2 are violated your correct operation will be disturbed.

(Refer Slide Time: 27:10)



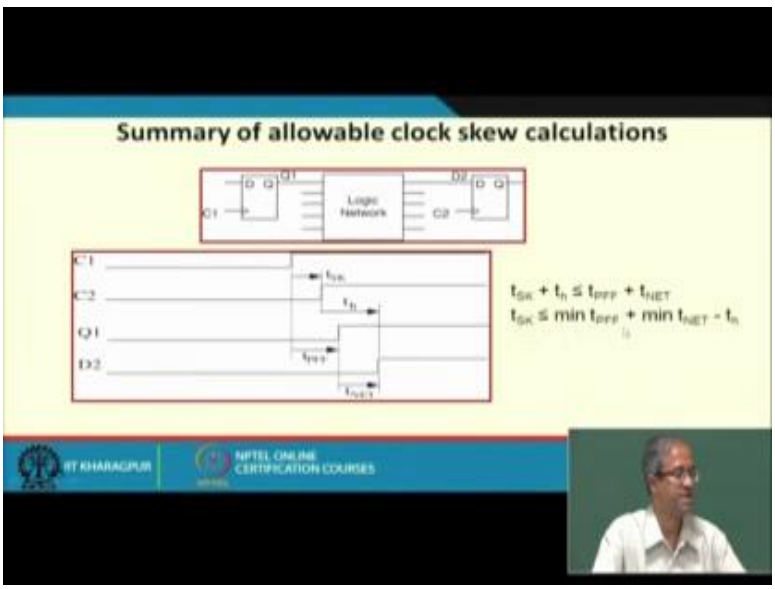
And if C1 is delayed in the other case, so you have a similar kind of situation you can similarly make a calculation. So I am just living it to you these are the scenarios C1 is getting delayed.

(Refer Slide Time: 27:26)



And additional delay between flip flop how it affects skew calculation one last example. Suppose there is a multiplexer in between. So let us consider a scenario like this. Skew C2 is delayed skew is minimum propagation delay time minus hold time total is tff minus t this is tsk and when you also include the delay of the multiplexer your D2 will change after that t ff plus t max D2 will change after that. And when it comes to the input of the second flip flop this will be the hold time for C2. After the c to h comes minimum this much time the data must be stable. So your second condition will be like this several time minimum t ff plus by multiplexer minus th, th will get subtracted.

(Refer Slide Time: 28:28)



So, to summaries for a circuit like this, skew these are the 2 question already we have talked about. So you can use these 2 equations and one you can check whether these 2 are getting satisfied if not; that means, there is some timing violation there.

So with this we come to the end of this lecture. So we shall be talking about some more issues about clocking in the next lecture as well.

Thank you.