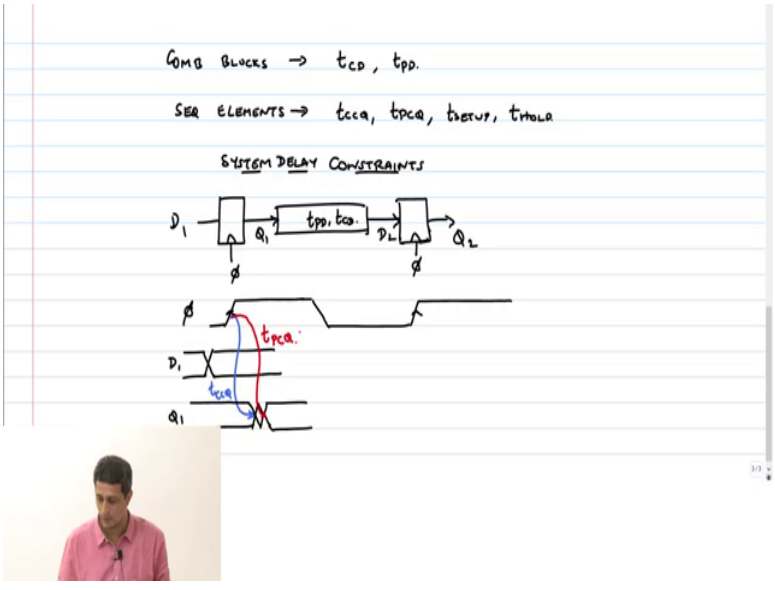
Digital IC Design
Prof. Janakiraman Viraraghavan
Department of Electrical Engineering
Indian Institute of Technology, Madras

Lecture - 74
Max and Min Delay of Flop Based Systems

So, now what we have done is we have built in module-4, we learned how to build any arbitrary combinational circuit right how to build any Boolean logic, design it with various styles of logic, so that we can meet certain specification, we can make it faster slower whatever it is right, lower the power. We learnt many many techniques to design combinational blocks. Now, in this module, we have designed all these sequential elements that we needed latches, flops, what is the set up time, hold time so on.

(Refer Slide Time: 00:53)



Right now what we have is we have combinational blocks, sorry blocks that have a delay t CD that is contamination delay right and a propagation delay right. We can characterize this and figure out what the two delays are right. Then I have sequential elements which have, so let us only deal with a flop for now ok.

So, we have t CCQ, t PCQ, right t set up and t hold. So, I can design these elements and hand them over to you, and tell you that you have to now put these things together in order to make the system function at a particular clock speed. Our aim now is to determine what the maximum clock speed is, what is the minimum clocks clock speed right, is there any minimum delay constraint, what is the maximum delay constraint, all of that we will study in over the next two days ok.

So, let us start with a basic clock based system, and I will call it system delay constraints right. So, I have a flop with a clock phi this is going to go through a combinational block ok. And it going to come and sit here with another flop ok. The left flop is called a launch flop; the right flop is called a capture flop ok. So, I have D Q 1, D 1, Q 1, D 2 and Q 2. So, right now I am assuming that the clock period is given to me I have a clock period and the time period is T c ok.

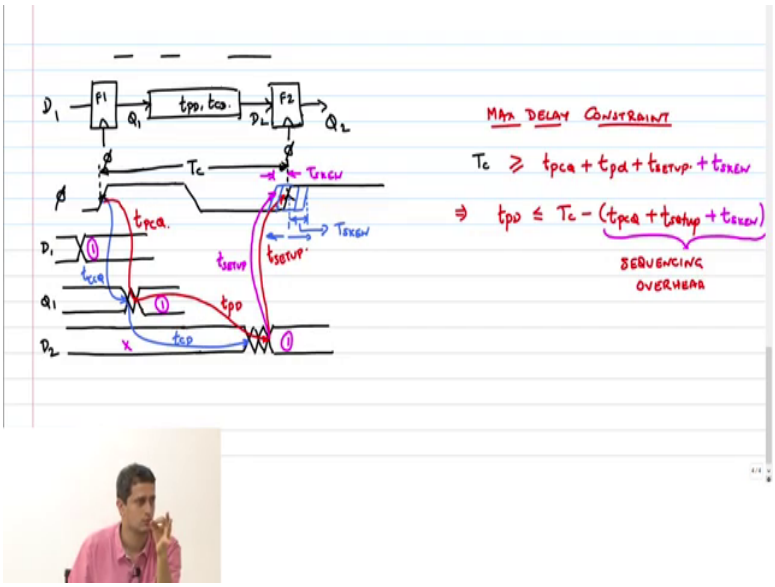
So, let us assume that the clock is like this ok. Now, I want to find out what is the this is t PD comma t CD. What is the maximum delay that I can insert between these two flops, so that this system will not fail? What is the combinational block and what is the maximum delay of the combinational block that I can insert between these two flops, so that the system does not fail ok.

So, what are these flops doing? These flops they are memory elements, but here they are not playing just the role of a memory element, their aim is not just to store something, their aim is to sequence events. You launch some data it has to calculate something that flop has to catch capture that particular event and hold it for the next guy to. So, the events get sequences 1, 2, 3 and so on, and no event should get missed that is the aim of putting it you know in a system with a flop launch, launch flop and a capture flop and so on ok.

So, let us see what happens I have a change in D in D 1 that is happening here right. So, after the clock Q 1 right, I mean after the clock rises right at phi, Q 1 will change after . So, now, I will do this ok. Q 1 from the clock edge, this is what t CCQ. What does this mean? It mean with the data had changed well before the clock edge, then this would have been the delay.

I do not know when the data is changing right, I have to account for worst case under all conditions that is what the analysis is going to is going to be all about. Now, this guy will change worst case. What is this delay t PCQ right ok.

(Refer Slide Time: 06:25)



So, now what do I do, I am going to look at D 2. D 2 is going through a D 2 is the output of the combinational block right. So, the moment Q 1 starts changing right they can be the earliest change that happens at the output of the combinational block which is the contamination delay of that block right which will be determined by the shortest path of

change right. So, D 2 will have another similar structure here right, where you have some changes like this.

From here you will have what is this delay t CD contamination delay of the combinational block, and this guy will be t PD. Now, in order right remember D 2 was earlier holding some state from the previous evaluation that state has changed, and that now needs to be captured by the flop F 2, F 1, F 2ok. Flop F 2 has to capture this new state appropriately right.

So, what I will do is I just put an event here, this event has reflected here and this given eventually reflects it. It is this event that has to be captured not this one ok. So, in order for this event 1 to be captured, what is the constraint now, it should be it should settle a setup time before the next clock edge. So, therefore, this has to be t setup correct. This is your clock period. So, what is the constraint now?

Student: (Refer Time: 09:08).

Right. So, let me write T c here, and I will write the sum of these delays which is basically t PCQ plus t what p d plus t SETUP. What is the inequality, greater than or equal. As long as the clock is faster than this I mean slower than this period right, all the delay clock period is greater than these three delays, your system will function correctly ok.

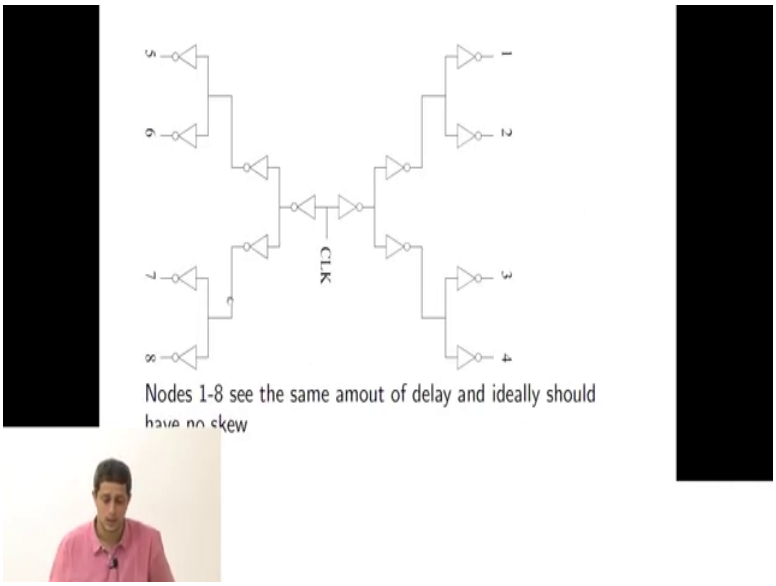
So, what does this mean, it implies t PD should be less than or equal to t c minus t PCQ plus t setup. Therefore, this constraint that we derived this now is called max delay constraint, it is the maximum delay that the combinational block can introduce without causing a failure in the system for a given clock period. So, this is basically called max delay constraint ok.

And this ideally if my flops were so perfect, I did not did not have a set of time and clock to delay and all that, then the entire clock period should have been available to me. I should have been able to use a fully clock period to do my combinational circuit evaluation, but that is not possible. So, this is called sequencing overhead ok. We will see next when we do pipelining.

When I remember that earlier also when I discuss the idea of pipelining initially; I told you that you cannot just keep adding sequencing elements and make it faster and faster. You cannot improve the throughput continuously right, because you have a concept of a sequencing overhead, and that is essentially the setup time plus the propagation clock to Q delay ok.

So, now what happens is maybe I should do this by the way, these two clocks right to F 1 and F 2 are coming from different points on the clock distribution network ok. So, how does crock distribution happen let me show you that very briefly here ok.

(Refer Slide Time: 12:05)



So, what you do is the clock generation happens at a central point in the chip, because the clock has to be insensitive to temperature, voltage changes and all that. So, you make a p l l base clock or something like that which is very stable right has no jitter right. What is jitter

basically, the clock edge just moves around the little bit randomly with time one edge can go this way or that way right. So, you want to design something with loads jitter and all that. And then you distribute this clock to the entire chip in a tree like fashion here.

So, what is shown here is the clock is generated at this point, and I just rout it two inverters in this fashion right. So, the nodes 1 to 8 are my leaf nodes, where I am going to connect flops in this example; of course, in reality it is like you know millions of nodes. So, distribution network is a very complicated thing, but the idea is to ensure that the delay on any path is basically the same ok.

You can optimize this delay you can make it lesser and all that. I do not want it is not necessary that the clock at node 1 and the clock at the origin should be in sync that is not necessary, of course, there will be a delay. The clocks at all these leaf nodes should be in sync, because that is what is going to control all my flops right.

So, we safely assumed here that this phi which is driving flop F 1 is also driving F 2 without any sync problems that is why we drew only one clock; otherwise I should have drawn phi 1 and phi 2, and you know delayed them or something right. So, ideally you want it to be like this, and that is what that is how you will design it with your clock distribution network.

So, if you take this network here all parts are the same if you ensure that the layout also is symmetric across all of them, the delay will be the same. In simulation when you try to simulate this, you will see that the delay at all leaf nodes 1 to 8 are exactly the same, there is no problem. However, when you manufacture this quite; obviously, you are going to have some process variations.

All these inverters will not be identical to each other right. And also spatially if you look at this distribution network over the entire chip, one end of the chip may have a slight bias with respect to process parameters with respect to the other way. So, all these inverters may be slightly different from all these inverters that is also possible. Apart from random changes even systematically you could have some variations ok.

Now, because of that, I have to deal with what is known as clock skew right. Clock skew is basically the clock edge can simply now start varying randomly about these points ok. So, what is clock skew? What is the exact you know delay here, it is from the reference clock edge where does this thing move right. So, for example, so I have shown you a band of curves here, right a family of curves. Clock skew is basically the worst case there. What is the worst case offset that I can have that is called clock skew, T skew ok.

Now, I want to alter my timing constraint, so that I am able to incorporate this as well. What do you think will happen? First of all before you start this analysis, you have to tell me which case should I consider, should I consider the case where the clock edge moves this way or does it move this way. It moves to the left, right. It has to move this way because it has to (Refer Time: 16:33). If it arrives later, then I have no problem, because my data is anyway being set up with respect to the original clock edge, and this thing has got delayed.

So, there is no issue. If it comes inside, then I have a problem. Then what is the thing that gets altered in this diagram, t setup becomes with respect to what, this skewed clock right. So, with clock skew this has to be t setup right. And of course, the delay from here to here is T skew right. So, how do I modify this expression now, t PCQ plus t PD plus t setup plus t skew right.

And therefore, here also it will simply get modified as plus t skew ok. And of course, this whole thing now is called, it is called sequencing for. Yes.

Student: (Refer Time: 18:17).

We D 1 it arrives.

Student: (Refer Time: 18:32).

So, like I told you the reference here to make with respect to one clock. So, I like I told you my reference clock is not this clock, it is not with respect to this clock that the leaf nodes start

deviating. This guy clock 1, the flop connected to the node 1 will become my reference clock with respect to that what happens to the next one is what we are saying. So, what you are saying is what if this guy at F 1, it arrives late.

Student: (Refer Time: 19:12).

So, if this arrives t skew later right,

Student: (Refer Time: 19:29).

So, question is should you count it twice or not?

Student: (Refer Time: 19:47).

Exactly. So, it is just a question of counting it as a which one you take as a reference that is all. And T skew I think it just boils down to what you define as t skew right that worst case number is with respect to one reference clock, where do these edges move right.

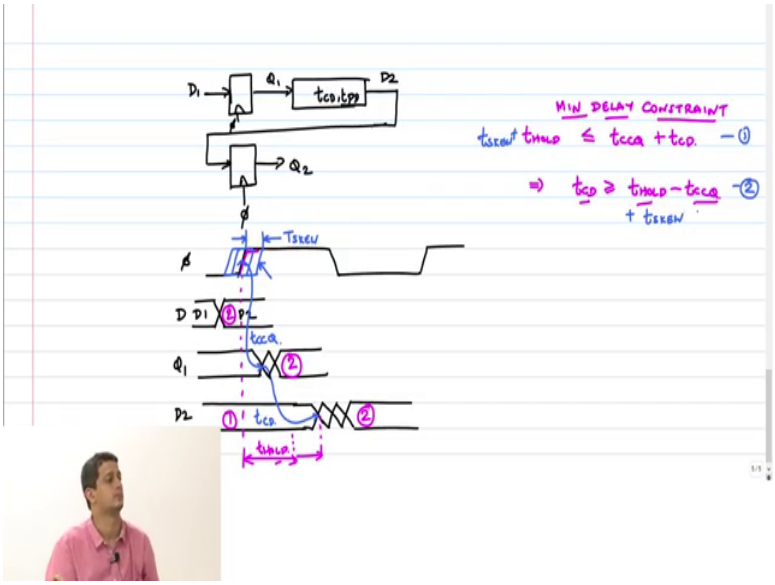
Student: (Refer Time: 20:21).

Correct.

Student: (Refer Time: 20:33).

Between any two clocks right. Therefore, then I if I consider t skew here once I am D 1, because I have considered between any two random clocks, this is the maximum number right. So, you could divide it as t skew by 2, here t skew by 2 that does not matter, but it is just one count, correct. So, I think that is right. So, what he says what he is saying is the t skew definition is the worst case difference between any two clocks on the leaf nodes that is what is t skew ok, so that we have taken care ok.

(Refer Slide Time: 21:13)



So, now what I want to do is, I am going to consider the following case now next D 1, Q 1, then I have the combinational block delay t CD, t PD ok. Now, I am going to look at the case where data can raise through right and affect D 2 right, this is D 2 in the same clock edge right. I am not waiting it for it to go and so. Like I told you here D 2 was maintaining some earlier state right, and that needs to be captured by the flop F 2 and it needs to be held for the whole period and all that right.

But what if data from D 1 races through and effects that earlier state that it is holding right that is also possible. So, the way we analyzed that is we bring this flop back here ok, just notionally. This is Q 2 ok. This is again phi, phi right. So, what I am doing here is my clock is going high, forget about the other edge ok. And data on D has changed ok. This is D 1; this is D 2 right. Q 1 changes starts changing at t CCQ right. Can you look at this?.

And D 2, where is D 2, it is going to start changing after t CD from t CCQ correct. So, the earliest change will happen here ok. So, t CCQ, then from this edge, you can have t CD ok. Now, earlier in the previous cycle right, D 2 was being, D 2, the at D 2 there was some earlier value that had to be sampled ok. So, if you look at the D 2 guy, this is some value let me.

So, let me call this event 2, event 2 has happened here, event 2 is happened here. Event 1 is what was sitting on D 2 ok, and that needs to be sampled for the flop F 1 to hold it correctly ok. Remember I am talking now I add this rising edge itself. So, the event 1 actually needs to be sampled and held there correctly. The event 2 should not raise through and overwrite the data.

Now in order for D 2 to sample event one that you see here, what is the constraint that I have to ensure. I have to do a whole time right if D 2 changes before the whole time then I have an issue right that is the basic point. So, what is the constraint that I can now impose, this guy here, this delay in the worst case should be t hold. So, therefore, I can now write t hold, and I have two delays there t CCQ and plus t CD. What is the inequality?

Student: (Refer Time: 26:24).

Yes, as long as the whole time as long as the data change happens after the whole time right. So, if this is my whole time right, so let us say that you know this is the worst case whole time, but let us say the whole time of the flop is somewhere here, let us say this is t hold. As long as this change happens later, there is no issue right and that is what we are saying. So, what does this mean, it implies that t CD should be greater than or equal to t hold minus Q minus t CCQ correct.

So, this is called a min delay constraint, min delay constraint. The combinational block that you put between two flops should have a minimum delay, so that you do not end up with a hold time violation right, correct. Now, let me ask you this question, suppose there is a set up time failure what will you do? Suppose, the set up time fails after you design the system

something happens, you are getting a set up time failure at the clock, increase the time period of the clock right.

When you have when you derive the max delay constraint, we said t PD was less than t c clock period minus some overhead, unfortunately in the min delay constraint, you do not have that luxury. You do not have that luxury of the clock period. Remember after the chip comes back, this number is fixed, this number is fixed, this number is fixed. It is determined inside the chip and you are done for.

So, you have to be very very careful about hold violations if at simulation or at design time itself you figure out there is a hold violation, what should you do, I had buffers and increase the dealing of the combinational block right that contamination delay, you should increase somehow. I add some buffers and you can solve that problem there, but after the chip is manufactured nothing ok.

On the other hand, if there is set up time by releasing you can reduce the clock frequency and make it at least work right. You may not be able to sell it appropriately with further enough profit, but at least you can get it to work. You want lose your job right, whole time violation you will surely lose your job. So, you are going to dump that whole chip and the entire lot, forget just that chip ok. So, you have to be very careful about this.

Now, as usual we have to now incorporate clock skew into this what happens if the clock edge moves around the little bit like this. First of all which edge should I consider, should worst case consideration is the edge moving to the left or the edge moving to the right? To the right, because I now need to ensure that the two is held for long enough after this edge right.

So, this if you look at this delay it is t skew, from there I need to ensure that there is enough whole time. So, how do I modify this equation, first equation here, 1 2? Where should I add t skew? To the left. So, what is the net thing now, I will add t skew ok.