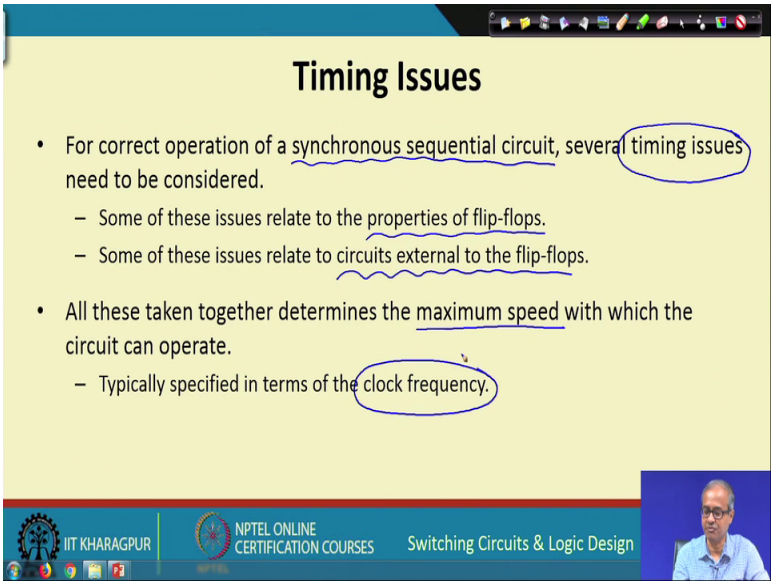
Switching Circuits and Logic Design
Prof. Indranil Sengupta
Department of Computer Science and Engineering
Indian Institute of Technology, Kharagpur

Lecture - 34
Clocking and Timing (Part I)

In the last few lectures if you recall, we had discussed the various types of latches and flip flops how they work. We talked about both edge triggered and the master slave variations of flip flops, which are very useful in designing synchronous sequential circuits. But there are some timing issues which you have not discussed so far; for instance, what should the clock frequency be, what is the maximum frequency with which a circuit can operate without any errors and so on and so forth. So, we shall try to address some of these issues in this lecture. So, the title of this lecture is Clocking and Timing.

(Refer Slide Time: 01:08)



Fine let us first try to understand why timing issues are important in a synchronous sequential circuit.

Now, let me repeat a sequential circuit is set to be synchronous if there is a clock frequency, and all state changes and outputs are generated in synchronism with the clock. So, synchronous sequential circuit means that; there is a clock which is fed to the flip

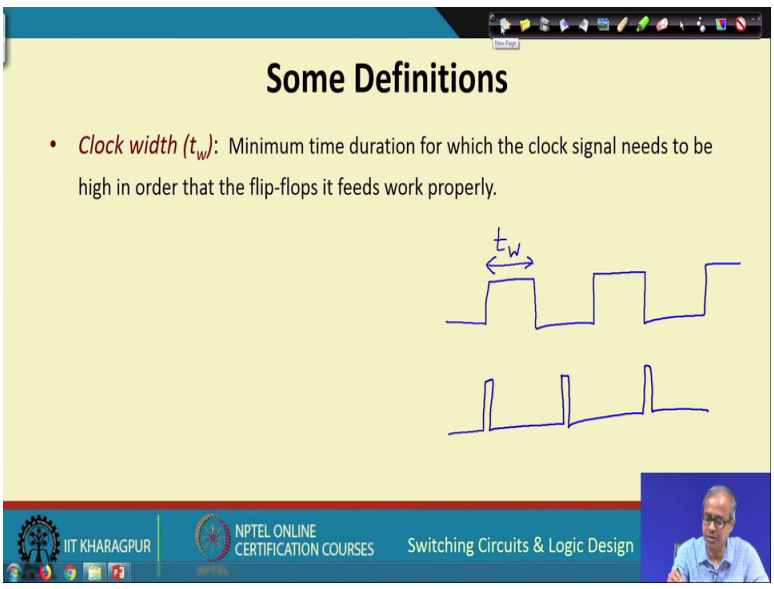
flops, and the flip flops would be working would be changing the states in synchronism with the clock edges.

Now as I said there are several timing issues that need to be considered that we shall be seen. Now these timing issues relate to some properties of the flip flops for example, given a flip flop when the clock edge arrives the question is; when should we apply the data inputs consider a D flip flop; some value should be applied to D and then the clock pulse should be applied the clock edge should come.

Now, minimum what is the time duration before which the data should arrive, and only then the clock edge come those timing constraints should be considered. And also after the clock edge come, how much more time I should hold my input data steady so that; the data can go inside the latch without any problem. These are some issues which relate to the properties of flip flops how fast they are and so on.

And of course, there are some issues which relate to the delays of the circuits which are outside the flip flops. So, we shall be considering all of these, all of these taken together will determine the maximum speed of operation of the circuit and we normally designate the maximum speed of operation in terms of the clock frequency right?

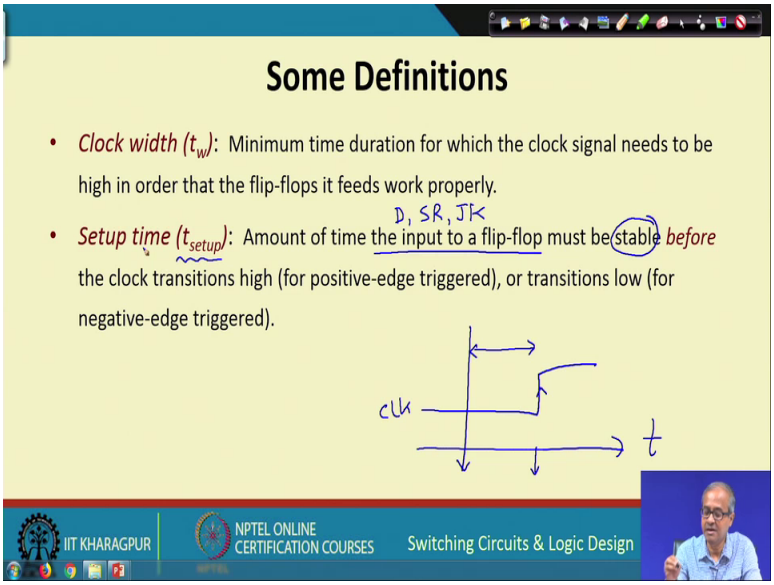
(Refer Slide Time: 03:34)



So, let us now look at some basic definitions with which we can explain these issues of timing in a much better way. The first definition we talk about is Clock width, Clock

width says suppose, this is my clock pulse. Now I can even give the clock pulse like this very narrow pulse. Now this clock with constraint says that what must be the minimum time duration for which the clock signal should be high, it is this duration this time is t w right? So, there is a minimum duration of the clock pulse width which should be there in order the flip flop works properly, this is the clock width constraint.

(Refer Slide Time: 04:40)

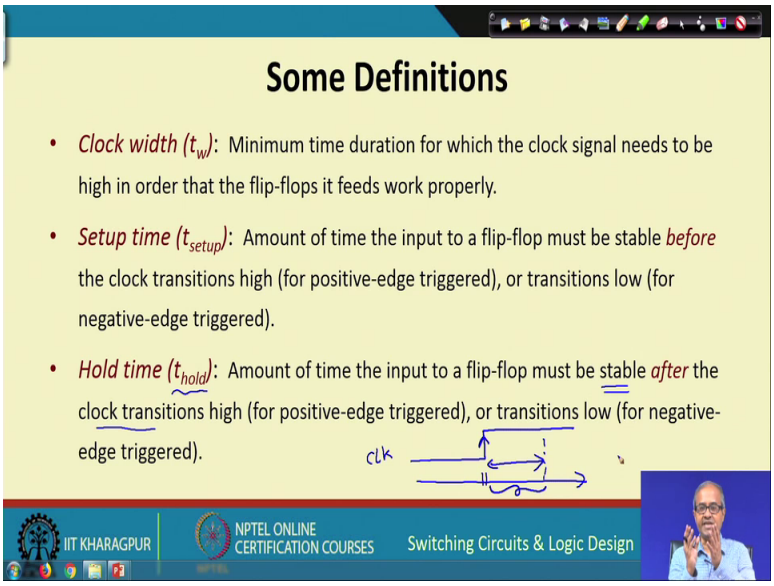


Then we have something very important this is called Setup Time. Setup Time I shall be giving some examples later.

But let us look at the definition we designate the setup time by a t setup this is defined as the amount of time the input to a flip flop, input to a flip flop means it can be a D input for a D flip flop, it can be S and R input for an SR flip flop, J and K for J K flip flop and so on. The time the inputs of a flip flop must be applied and must be held table before the clock transition comes like; if I look into the axis of time suppose this is the time where all the inputs are stable, and this is the time where my clock edge comes let us say for a positive edge triggered my clock edge comes here.

So, this is the duration for which my data remains stable before the clock edge that is defined as the setup time. And setup time is a characteristics of the flip flop that you are using depending on the flip flop that you use, it has a specified setup time and the input must be applied at least before that amount of time.

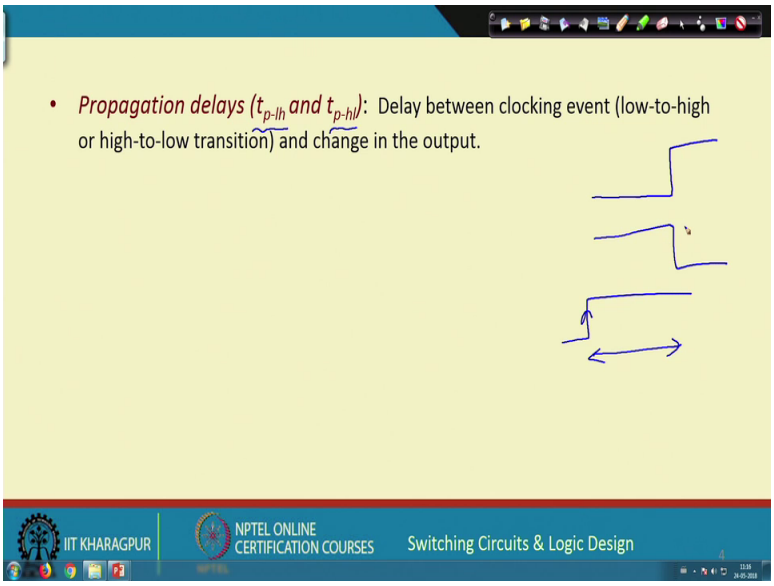
(Refer Slide Time: 06:07)



So in a similar way, you have another timing constraint called the Hold time denoted by t hold, this is on the other side of the clock edge this is the amount of time the input of the flip flop must be stable after the clock edge comes; that means, for a positive edge triggered the clock goes high or a negative edge triggered clock goes low

So, again in the axis of time if I say that my clock let us say a positive edge triggered, my clock comes here at this point in time. So, what is the minimum duration I must hold my input stable after this; let us say this much and this duration is my t hold my hold time. So, setup time specifies, with respect to clock edge how much before I must apply the inputs and hold time specifies, again with respect to the clock edge how much more I must maintain that input stable so that the flip flop works correctly. And of course, there is another delay to be considered it is the Propagation delay.

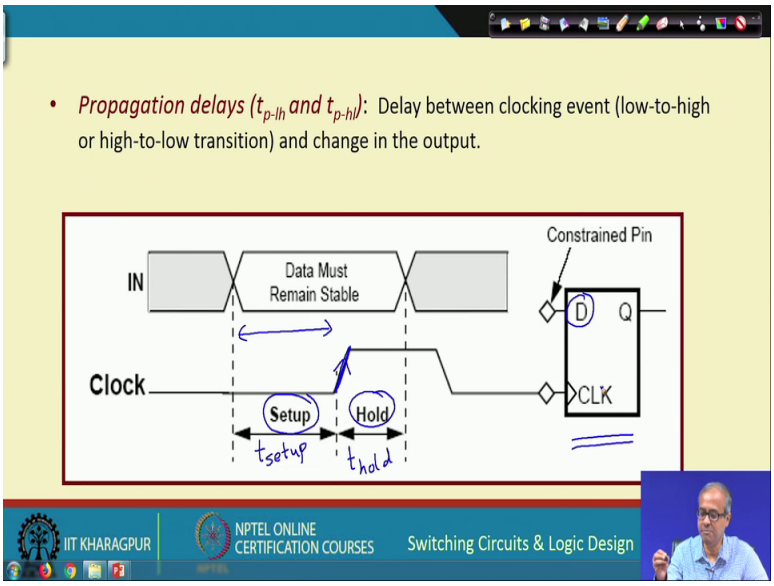
(Refer Slide Time: 07:27)



So, you see these are the constraints we talked about setup time, hold time, but the flip flop as a circuit it will be having it is own delay. There are so many gates all the gates will be having some delay. So, what is the minimum time after the clock edge has come that the output will start to change that is, the propagation delay of my hold circuit in this case the flip flop right?

So, the propagation delay we can define in 2 to w 2 ways propagation low to high transition and propagation high to low transition. So, after the clock has appeared this signal output signal, it can be it was low it was it becomes high or it was high it becomes low. So, what is the total delay? Suppose the clock came here this is the clock. So, with respect to the clock what is the delay, after which the output starts changing that is denoted as t propagation delay low to high or high to low as the case may be right? Let us take an example.

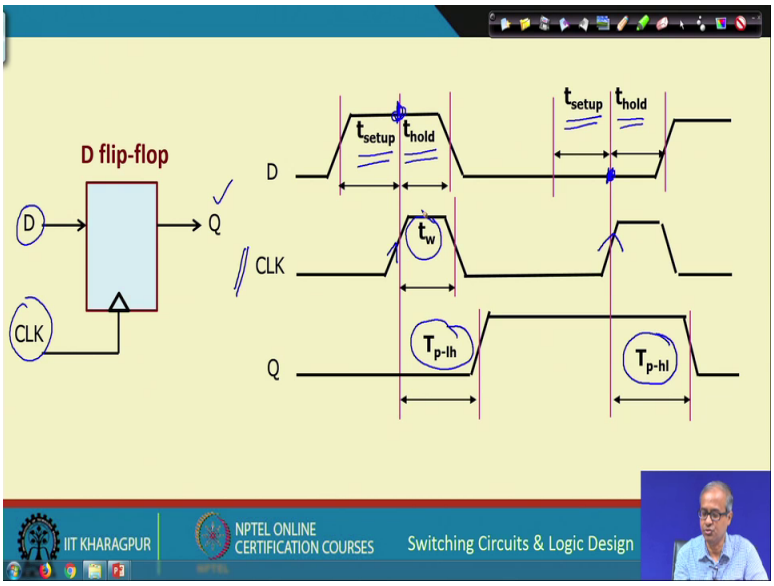
(Refer Slide Time: 08:42)



Consider a scenario like this; here we have a simple D flip flop with a D input and a clock. See the clock input I am applying a clock pulse like this. So, I am showing only one single clock pulse suppose this is positive edge triggered as the symbol shows. So, the circuit will be triggered by the positive edge of the clock; that means this edge.

Now, in this case is the input data which is only a single data D, the input data is coming. So, what I am saying is that before the clock edge comes my input data must remain stable for at least a time equal to t setup; this is my set up time. And after the clock edge has come I must maintain the data stable for another duration of time this is my hold time t hold. So, if you maintain this t step t setup and t hold constraints then your flip flop will be working correctly, otherwise if you apply the inputs too late or remove it too early the data that will finally, gets stored in the flip flop may be wrong. So, you need to satisfy the setup and hold constraints in a flip flop like this as I have showed you.

(Refer Slide Time: 10:25)



Let us take another diagram to show this part, there is other delays are also shown here. So, here again we have a D flip flop with a D input a clock input and output Q I am not showing Q bar I am only showing Q. So, let us assume that in the clock input I am applying a clock pulse and because this is positive edge triggered, it will be triggered in the positive edge of the pulse this is my triggering edge.

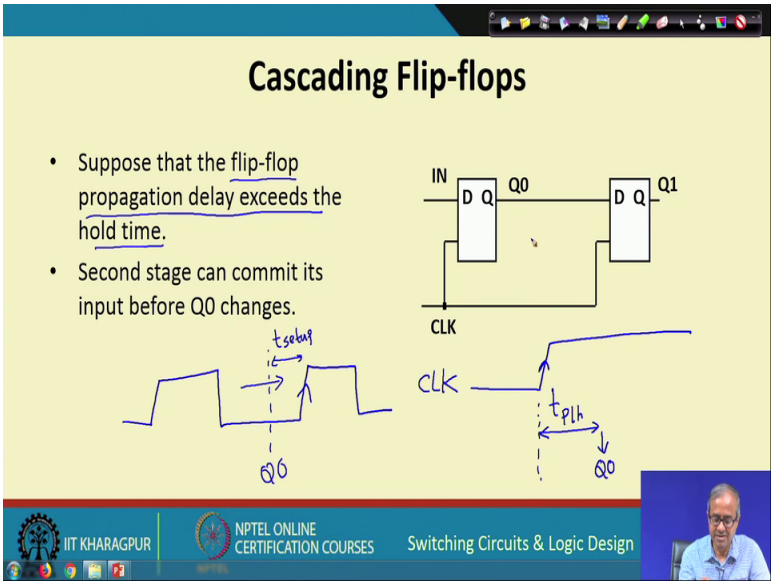
Now suppose in D input I am applying some signal like this. See you when this clock edge comes the value of D is 1 this point. So, this t setup is the minimum duration before this edge I must apply this D equal to 1 and leave it like that similarly, after the clock edge comes this hold time t hold I must maintain D for this much time again.

And after the clock edge come, let us say here this red vertical bar how much time it will take for the output to change? This is my propagation delay now since here D was 0 it is becoming 1 so Q will also become 1 so this is a low to high transition. So, this is denoted by propagation delay low to high. This is here and suppose later again when the next clock comes my D is 0, here the value of D is 0 here.

So, again I must apply this 0 t setup time before and maintain it for a duration of t hold after this. Now the output since it goes from 1 to 0 now the corresponding delay will be t p high to low. Now for a flip flop typically these 2 delays are equal, but in case they are unequal that is why you use 2 different symbols l h and h l.

So, in this diagram, we show all the relevant delays and of course, the width of the clock pulse that is t w, I mentioned that is also shown.

(Refer Slide Time: 13:01)



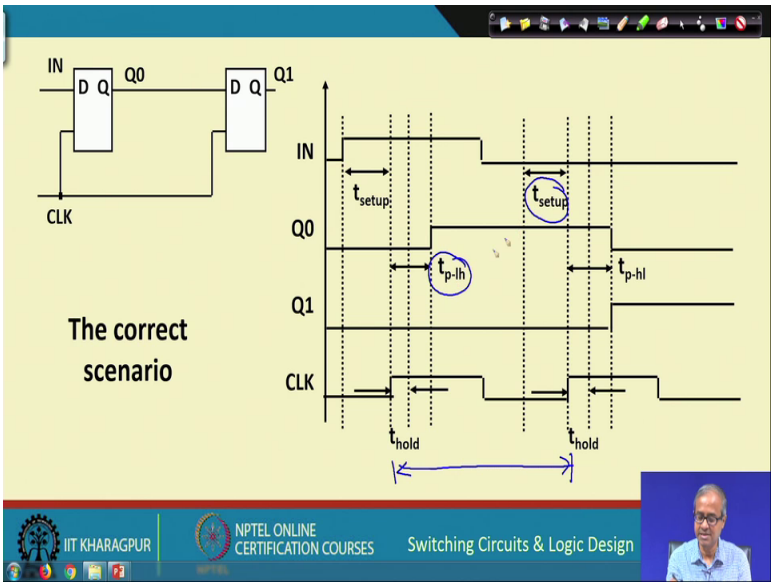
Now, the next question is; when we connect more than one flip flops, I mean one after the other we say that it is a cascade, cascade of flip flops what will happen? Let us look at a very simple scenario where 2 flip flops 2 D flip flops this simplest kind of flip flop are connected in cascade, such input signal in is applied here the first flip flop generates an output Q 0, which is fed as the input to the next flip flop and this Q 1 is the final output. And the clock signal is applied to both the clock let us assume these are both positive edge triggered.

Now, there are some considerations you need to look here. Like the first thing that we are saying is that what if the flip flop propagation delay exceeds the hold time, well what you mean by it, let us look at the clock signal. Suppose, the clock signal was coming like this was the active edge or the clock, just assume that hold time and set up time constraints are being satisfied. So, after a some propagation delay so I am showing it after the clock edge comes, the output Q 0 will be available this is equal to the propagation delay t p well either low to high or high to low. So, it is at this point that the output Q 0 is available after the delay of flip flop.

Now, what I am saying is that what if the flip flop propagation delay this t p l h exceeds the hold time because you see when the clock signal comes like this, this same clock

signal is applied to both the flip flops. Consider the second flip flop this second clock pulse has come now the value of Q 0; this Q 0 should be held stable at least t set up time before the clock edge comes. Now if this t p l h is too large. So, that it eats of into the t set up time then the set up time constraint will be violated, this can be a problem ok. So, you should be careful enough that the clock frequency should not be too fast that such a thing happens.

(Refer Slide Time: 15:51)



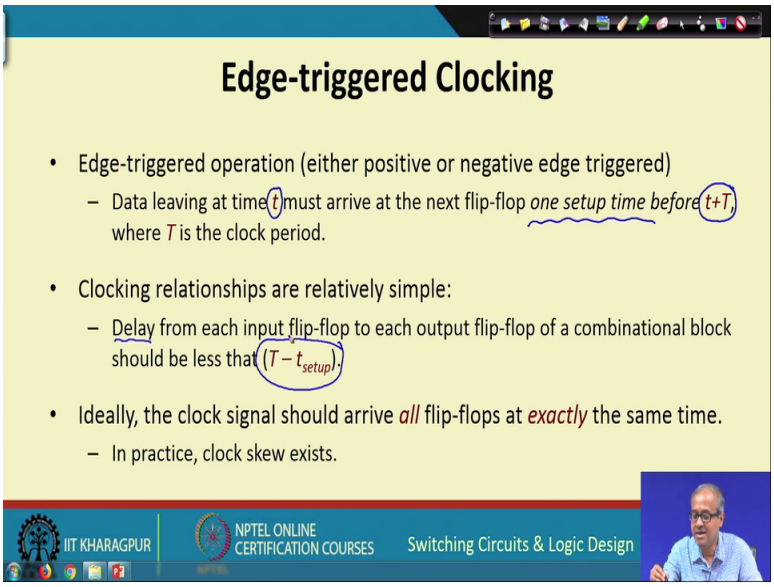
Now, let us show a scenario this is the correct scenario, the correct scenario where all the timing constraints are met. Let us see this is my clock signal, this is my first edge, this is my second edge. So, I am applying some value input here. So, when the clock edge comes it is here so I have applied the input at least t setup time before that and the input is stable till t hold this is fine. And output Q 0 will be available after the clock pulse after a delay of t propagation low to high, because Q 0 is changing from low to high.

Consider in the second flip flop second flip flop will be taking this Q 0 will be input and store it there. So, this input this new value of Q 0 ay the next clock, this next clock will be coming to the D input of the second flip flop. So, for the second flip flop again you see this setup time constraint is satisfied because this Q 0 is available here only. So, it is much buffer to sufficient buffer is here and off course hold is also there.

So, after that time Q 1 Q 1 will go from low to high. So, it is t p h l this after this delay this Q 0 1 value will get stored here right? So, this shows the basic requirement of the

timing constant which you see once more carefully. You see this is your total time period of the clock starting from here the first stage up to the next stage. Now within the time period what are the things that are included you must have one setup time, you must have this propagation delay and of course, hold time and propagation delay can overlap. So, setup time and this propagation delay at the most important delays that must be included here.

(Refer Slide Time: 18:11)

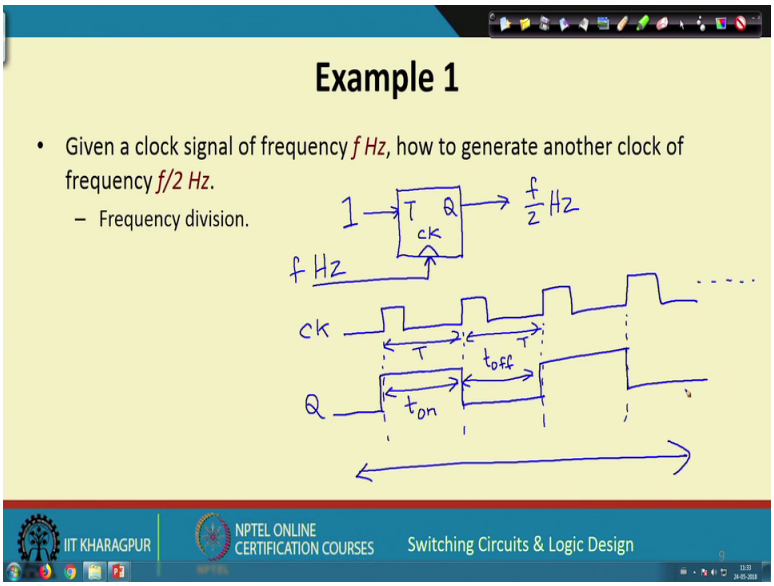


So, so when you talk about edge triggered clocking, well this edge triggered may be either positive edge triggered or negative edge triggered both ways. So, to summarize data that is generated from the output of the flip flop, let us say at time t must arrive at the next flip flop one setup time before t plus capital T, where T is the clock period. This is exactly what I mentioned in the previous diagram it must appear one setup time before this. And the second thing is that the delay from each input flip flop to each output flip flop of a combinational block should be less than T minus setup.

So, if the flip flop is too slow then it will be eating up into the setup time of the next flip flop and the timing violation will result. There is another issue which we shall be talking about very briefly later that the clock signal does not arrive exactly at the same time to all the flip flops, there will be some delay, because when the wires are laid out on a chip when a chip is fabricated there can be the flip flops, which is spread all round chip. The clock is generated at some point and it is connected to all the clock inputs of the flip flop

Now, the delays of tthese wires because the wires may be of unequal length that delays will also be un equal not the same. So, if you are not careful about this fact then the clock edge might reach the flip flops slightly varying in time either a little early or a little late. So, if you are not careful this may lead to incorrect behavior of the circuit ok, this is called clocks skew this is referred to as clocks skew, but here we are not discussed in detail regarding clocks skew.

(Refer Slide Time: 20:26)



Let us now look at a couple of examples. Let us say how this flip flops that we have designed we saw, how it can be designed how they work how they can be used to carry out certain very useful things. The first example we considered here is one of frequency division, what do mean by frequency division? We are saying that suppose we have I already have a clock signal of f hertz, what is hertz? F number of oscillations per second that is the definition of hertz.

So, we want to divide this frequency by 2 we want to generate another clock signal let us say whose frequency will be f by 2. So, how do we do it? Let us say I have a frequency of 1 kilohertz or let us say 2 kilo hertz and I required I mean another let us say clock of frequency 1 kilohertz, half of that. So, this is extremely easy to do using a special kind of a flip flop.

Let us take a T flip flop, T flip flop with the output and let us say this is my clock. So, what I do? Let us apply constant 1 on the T input and in the clock input I apply that f

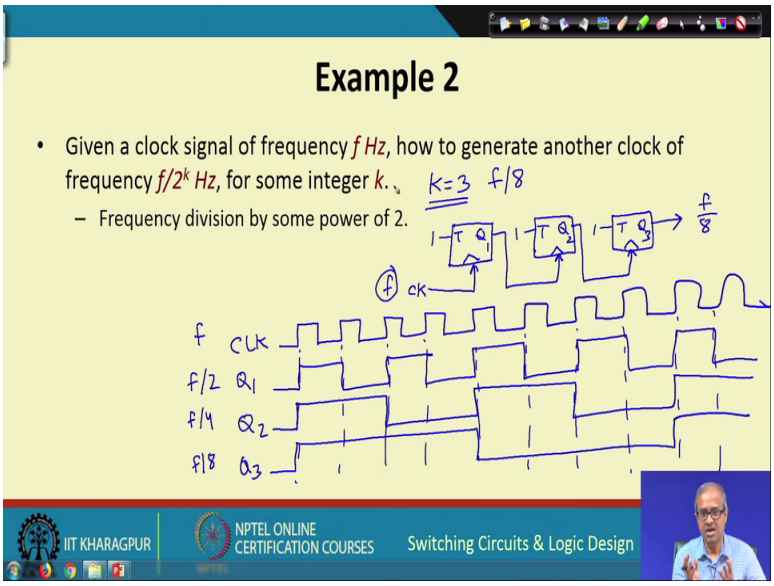
hertz clock frequency. So, what I claim is that at the output whatever will be generated that will be a f by 2 hertz clock signal, well how does it happen let us look into it with respect to a timing diagram.

So, I am showing a clock pulse I am showing 4 pulses. So, this is coming continuously now you think of a T flip flop this is positive edge triggered. So, this the active edge 0 to 0 to 1 going this is the active edge of the clock. Now what will be the value in Q suppose initially the value of Q was 0 let us say, because T is 1, what is the T flip flop? A T flip flop says that if the T input is 1 and a clock comes the output will be toggled, if it was 0 it will become 1, if it was 1 it will become 0. Here I am applying T continually as 1. So, every time a clock comes the output will toggle right because T is always 1.

So, whenever the first clock comes here let us say, this Q will toggle this will become 1, the next clock comes here it will again toggle it will become 0, next clock is here it will again become one next clock is here it will again become 0. So, you see over the original clock let us over this time duration which I have drawn, there were 4 pulses that were generated here. For every 2 pulses one pulse is generated only 2 which means I have reduced the frequency by half to half. So, a simple t flip flop like this can be used to divide the frequency by 2. Now means another advantage is there if you want this that in the output of Q.

The clock the on period of the clock and the of period of the clock if you call them t on and t of they will be exactly equal and they will be equal to the time period of the original clock signal in this was your time period both t on and t off will be equal to the time period. So, I will be getting an exactly square wave with equal on and of durations right. So, this is one example.

(Refer Slide Time: 25:24)



Let us extend this example further. So, here what I am saying is that; again we are given a clock of frequency f see again we are doing frequency division, but we are not dividing by 2 rather we are dividing it by some arbitrary power of 2, let us say 2 to the power k where k is some integer right let us take an example again here.

Let us assume k is 3, which means I am trying to divide a frequency by 8, 2 to the power 3 is 8. So, the required circuit here will be very simple well again we are using T flip flops, but not 1, but k number of T flip flops; that means, 3 3 T flip flops. And all the t inputs I am setting to 1, just like before and the clock these are the clock inputs. So, the clock input of frequency f, I am feeding here the output Q of the first T flip flop I am feeding in the clock of here and output of this I am feeding here and this is my final output. So, what I claim? That if my clock input frequency is f then here the frequency will be f by 8 ok.

Let us call this Q 1 Q 2 and Q 3, let us work out how it works suppose I have a clock pulse here 1 2 3 4 5 6 7 8 9 10. Let us say 10 pulses and this is the active edge right leading edge triggered, this is the active edge of the clock. So, if you plot Q 1, so Q 1 will toggle every time the active edge of f comes because T is 1 with respect to this C K. So, Q 1 suppose initially it was 0.

So, it will toggle one once here again here again here like this it will go on toggling right like this. So, the clock frequency was f for Q 1 it will be f by 2 just as we discussed

earlier in the previous example. Now think of Q 2 for Q 2 the clock input is Q 1, Q 1 is fed to the clock. So, same thing will happen, but with respect to the active edge of Q 1. So, if Q 2 was also 0. So, first toggling will be here next toggling will be here, next here, next here, next here and so on.

So, you see for every 2 pulses of Q 1, 1 pulse of Q 2 is generated. So, the frequency here will f by 4, and again Q 3 for Q 3 this Q 2 is fed as the clock of the last one. So, similarly if this is of 0 initially there will be one toggling here, one toggling here, one toggling here and so on. So, for 2 pulses of Q 2 one pulse of Q will be generated.

So, this will be f by 8. So, you see just by using a chain of t flip flops we can very easily divide an input clock frequency by any arbitrary power of 2. So, these are very interesting features of a flip flop of a T flip flop in particular that is, very useful whenever we want to generate some specific you can say timing signals with some frequency relations ships with respect to the original set.

So, with this we come to the end of this lecture, where we had discussed some of the timing and clocking issues that for 10 circuits when you use flip flops and we shall be continue our discussion in the next lecture as well.

Thank you.