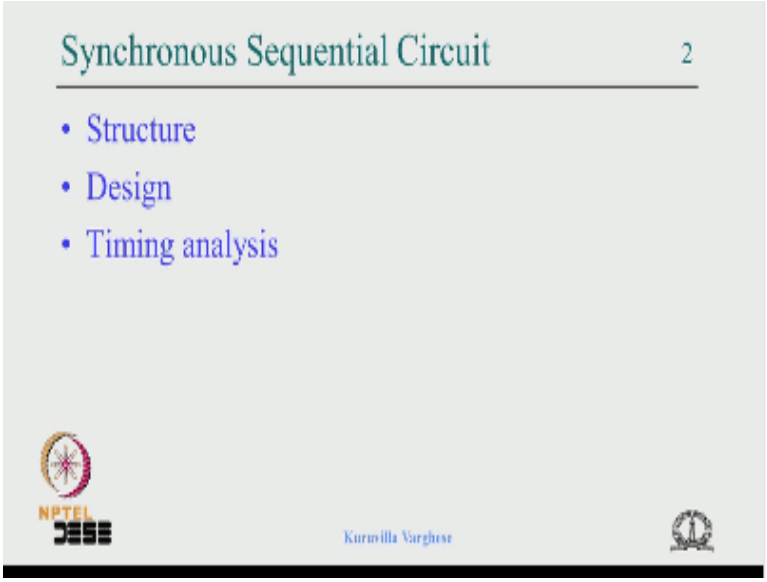
Digital Systems Design with PLDs and FPGAs
Kuruvilla Varghese
Department of Electronic Systems Engineering
Indian Institute of Science – Bangalore

Lecture-04
Analysis of Synchronous Sequential Circuits

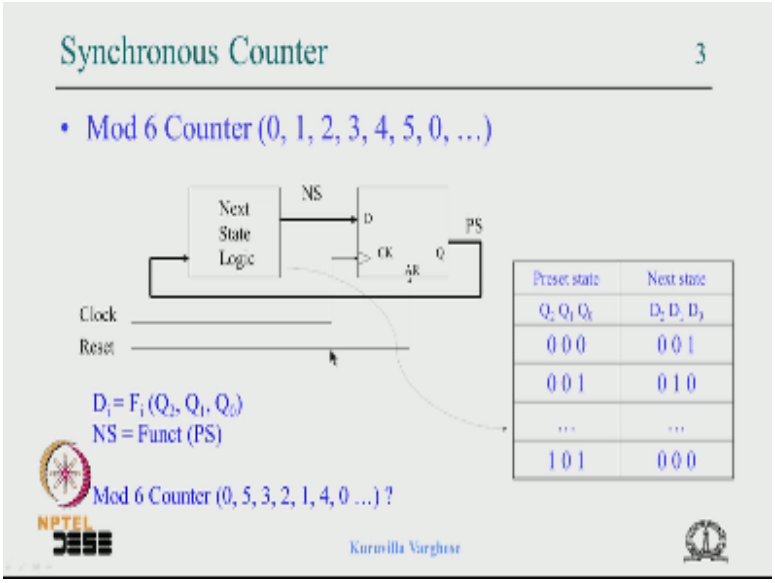
So welcome to the fourth lecture of digital system design with PLDS and FPGAs before continuing I will run through the previous lectures slides to be in sink.

(Refer Slide Time: 00:42)



So let us move on in the last lecture we have seen the synchronous sequential circuits, its structure the design and you will see the we will do the timing analysis today.

(Refer Slide Time: 00:58)



But quickly we said that the synchronous counter is the simplest synchronous sequential circuits we can learn, so we tried to build everything you know starting with the synchronous counter. So suppose we have taken we have taken an example of Mod 6 counter which counts sequentially from 0-5 and back to 0 and so on. So we said that you need for binary encoding 3 flip flop because logarithm of 6 to the base to use 2 point something and we take the integral number of flip flops 3.

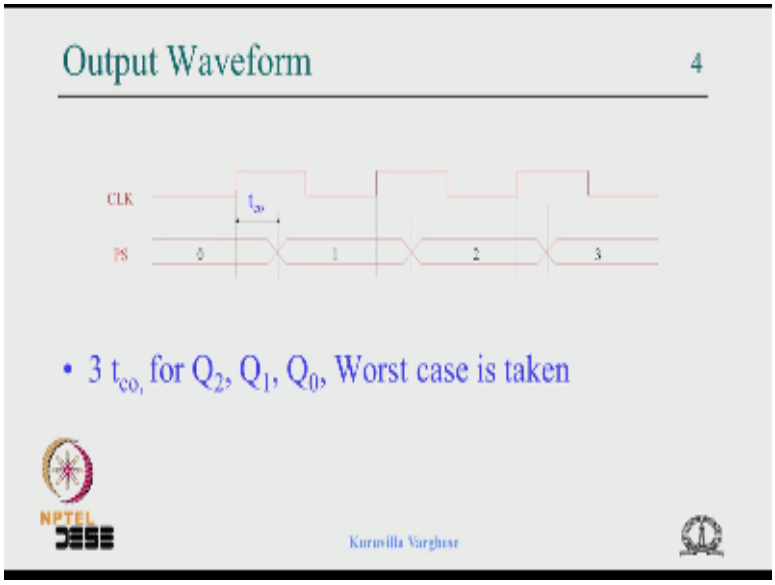
And I also said that we will not have show individual flip flops, we will show it together because we are trying to built complex systems, so all the clocks are tied together when the reset come this is 0. That is all we have, now in the next clock edge you have to have the 1 and next clock edge you have to the two. So the idea is to decode the next count from the present count, so we put a combination cute here before though the flip flops or registers and take the present count and try to decode the next count.

So that when the clock comes at the next count becomes a present count and we call it present state, state of the counter and this is the next stage which becomes a present state upon the clock and this combinational circuit has to be designed and that we know that we have designed the truth table, we have to write the truth table, so be able to design that and we write the truth table in terms of input ask Q2, Q1, Q0, D2, D1, Do.

Then from three equation for D2, D1 and D0. So you get Di as a 5 of Q2, Q1, Q0, I is 0, 1 and 2, so we say next state is a function of the present state and you should know that basically all the intelligence are the competition is in this combination circuit, the flip flop just wholesale store at work at the precise edge you know when the clock comes this is Latin ok that is the function of the registers it was a memory to hold and the precise timing is achieved by the clock.

So everything is in synchronous with the clock, the state of the count change in synchronous with the clock, that is why it is called synchronous counter and we said suppose you want to design a mod 6 counter which does not count in this sequence, this sequence maybe different it is not an order, but like the sequence could 0, 5, 3, 2, 1, 4, 0 and back and that case it is simple you do not do anything, you change the truth table to reflect this sequential.

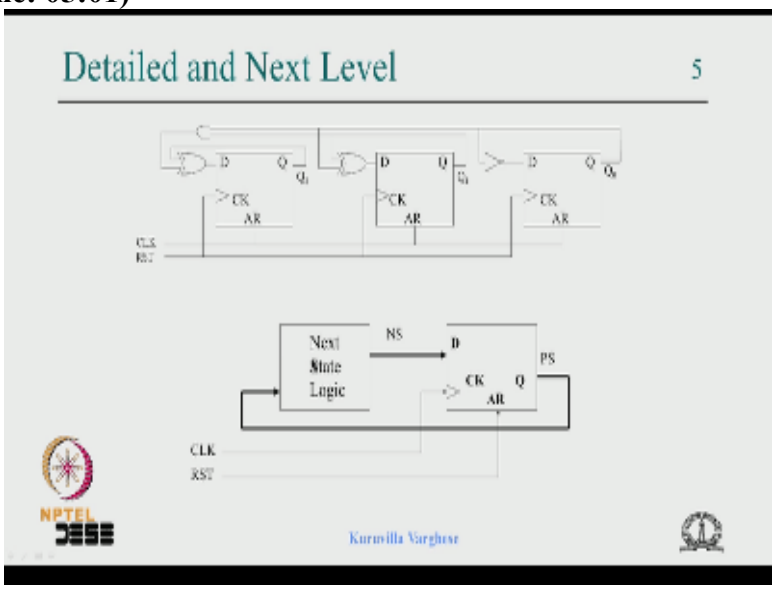
(Refer Slide Time: 04:01)



Then you get that counter, so that is very simple and we have looked at output waveform so the count changes up on the clock but not immediately there is a clock to output delay of the flip flop and we know that there are no this is this shows the multiple bit 3 bit and there are three delays Q4, Q2, Q1, Q0, so we represent the worst case delay and you should also know that many times this transition is not that.

Electrons move 3 from 1-2 because 1 is represented as 0,0,1 and 2 is represented as 0,1,0, in between there could be transit case state. In this is there could be a state tree is a brief duration here I am not shown that but normally when you simulate circuit you'll be able to see that for lease of drawing I have not shown and it is very difficult to say practically whether it will happen or not it might happen sometime.

(Refer Slide Time: 05:01)

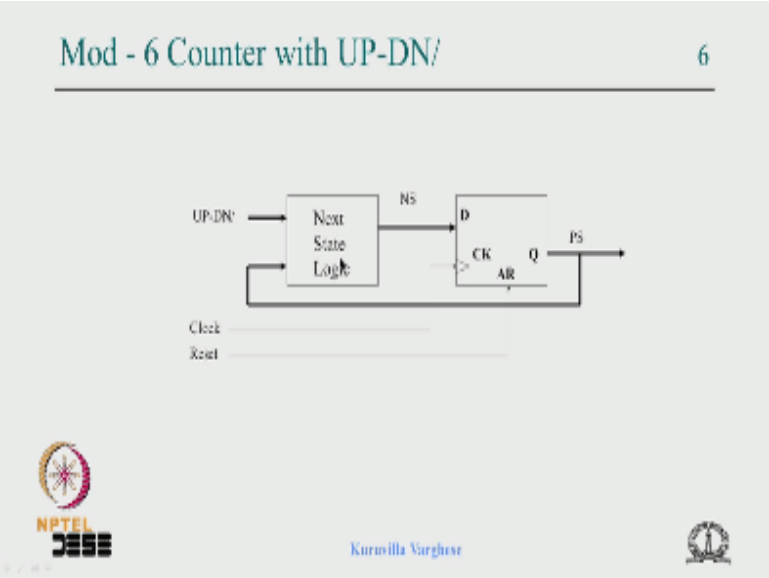


We will see where when it happens ok and I also reminded you last time saying that when we abstract to the next higher level do not lose sight of the detailed diagram. So we are showing the 3 flip flops together everything together but you should know that these are 3 flip flops separate. The D0 is a function of in this case only Q0, but in general case D0 can be a function of Q2, Q1, Q0 again same D2, and D1 could be a function of Q2, Q1, Q0.

So there are different circuit like you have a D0, there is one invertor, which is one level, but when it comes to D1 you have an XR gate, which is just one level but the gate is complex and when it comes D2 there is AND gate and an XR gate, so the delay from the present state to the next state is different for different the flip flops different parts. This should be kept in mind sometime when we see a picture like this that that is not a evident.

But that we should not lose sight of the detail, that inside you should be able to view. The D2,m D1, D0 a separate parts with different levels and different complexity and so on. So that is why I show this picture. So keep that in mind, so let us move on let us see the more complex counter.

(Refer Slide Time: 06:26)

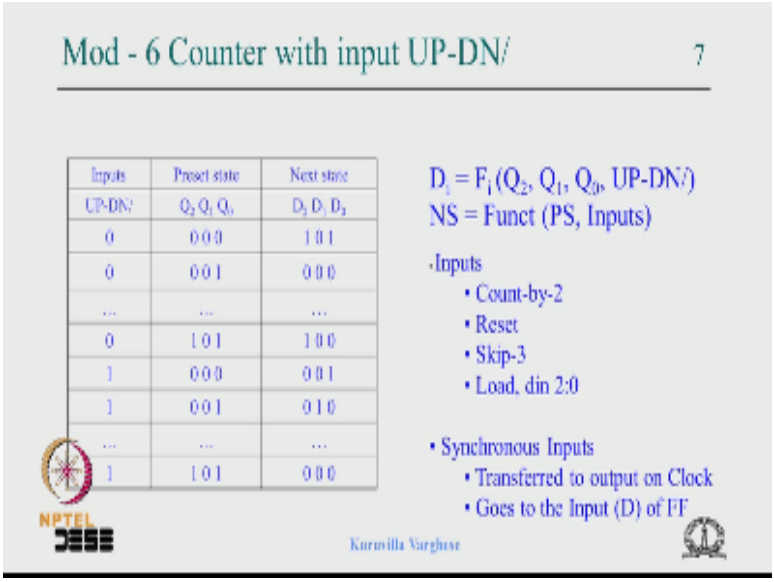


Suppose I want to design a mod 6 counter with up and down control that means there is a control call up, down, when it is 1 it will when it is 0 it will downtown, when it is up it is up count ok. That means we have to have an up down control when it is to say 0 it will be down counting, that means if it is we start with the reset 0 . When the clock comes if the up down is 0 then the next it will be 5 then 4, 3, 2, 1 with is 1 then from 0 it start 1, 2, 3, 4.

So word how do we are accommodate this, that is the question, but we know that in a synchronous counter the present status up on the clock the next state. So all this has to whatever happens has to happen in the input D here that means when the clock comes this is only moved here, so this up down should come in the next state logic ok. So essentially we give an input to the input it s a single bed its owners the bus.

But it is just single bed, so when it is 0 a downtown when it is up it is up count, so essentially we change the truth table of the next state logic then you get updown counter.

(Refer Slide Time: 07:58)



So let us look at the truth table now the truth table is becoming little more complex, you have the present state which is Q2, Q1, Q0, and the input up down. Now the next we decode as a function of the present state and input ok. So you have a column for D2, column for D1, typically in manual process will pick all the ones from the min term minimise it and all that. So but as I said we are going to use the higher level tools which will do the minimisation.

So we do not bother about the whole the minimization process we try to represent this table in some form of hardware description language, maybe we will enumerate every case in the most simplest case order some abstract form in high level form we will represent this truth table very concisely when we go to the VHDL language, but the thing is that we get free question.

Now is the function of not only the present state Q2, Q1, Q0, it is also a function of the input. So we say in this case next state is a function of the present state and the import. So you can

go back to the diagram. So the next state is a function of the present state, which is fed back from the flip flop and the inputs ok. This shows that you can give you know you can give a very complex counter.

Suppose 1 issue with this reset is that, this reset is asynchronous ok. It means that as soon as you make the reset active this goes to 0. But if you remove the reset and if the clock comes immediately then this will become 1. So otherwise all the other counts starting once you start the other accounts is of the duration of one clock period ok. So you can watch this the waveform, the waveform shows that this count remain there for 1 clock period.

But the 0 may not remain in the case of asynchronous reset, maybe we will reset the counter here immediately the cock comes then the 0 will be there only for a short duration. So there is an issue in this case with the starting state the 0 state, then once you started then everything is ok. But the next 0 which is kind of coming back will be ok, but the starting 0 is an issue.

So if you want that to be perfect of at least one clock duration then you have to have a synchronous reset. So it is very simple you give a reset here and change the truth table and say that when the reset is asserted this is 0. So clearly upon clock the 0 will come here and that will be remembered for the one cock period. So you could have any kind of suppose you want to load some value into this counter.

And start counting from there, say you want to load 3 very simple then you give a load control here and an input then when the load is high you say the next state is that input. Thereafter if the load is low it counts depending on up and down, then you get a presentable counter ok. So it is very simple you can write the truth table with the load at the reset up down everything.

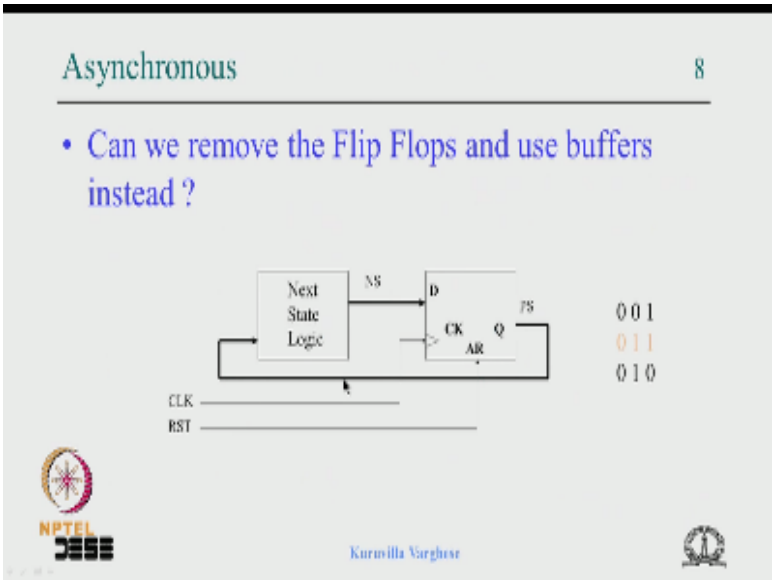
Any complex counter can be made and if you take if you think it's very easy even at in terms of the high-level function the research can be thought of as a marks here when the select line of the marks is the reset line and that is accepted we give a 0 and it goes there okay things like that, so I am here and output and we can view all these in terms of various multiplexers which we will see probably later.

For the time being let us assume that the combinational circuit functionality is changed by the truth table. So that is what I wanted to convey it, so you can have in principle various synchronous control like you can have control called count by 2, if it is active the counter will count by Q like instead of 0, 1, 2, 3 might go 0, 2, 4, 6, 0 like that. You can have a synchronous reset very properly to reset.

And this is very important because many counting application used for timing the precise timing and we cannot have any ambiguity that one state is only half. So in such cases it is better to have synchronous reset we can have a controlled call skip 3. Skip 3 is active then 3 will be skipped and as I said you can preset values by a certain load then there is an input 3 between which is this shows that into din 2, din1, din 0. So that is the meaning of this.

So when the load is high this value is loaded onto the counter. So all this can be incorporated that you should know, so you know all about how to built any counter which can count in any sequence any complex operation you need not worry like you need not learnt what is inside and worry about it and more over we have designing complex systems, so we can use hardware description language to represent all these behaviour in a you may obstruct we then enumerating like in a truth table.

(Refer Slide Time: 13:46)



We will see that when we learn the VHDL, so the next question I want to ask is that you we have taught about the synchronous you know synchronous sequential circuit synchronous counter can we have asynchronous counter, that means we do not have. In this case what

happens is that when the clock comes the next state come to the present state. That means if we start with 0 next clock comes the 1 comes here.

And that goes there and it decodes 2 and 2 is ready next clock comes 2 comes here, but all the intelligence is in the combinational circuit, but if you see the purpose of the flip flop is to kind of hold the value here so that this logical work properly and secondly it precise timing, when the clock comes the things change at the exact instant ok, of course with a delay.

If the delay is kind of fixed then at the same time at the same interval this change happens over and over, but the question is that if that is the purpose why not suppose the propagation delay of this flip flop is 1 nanoseconds and what we will do is that between D2 and Q2 between D1 and Q1, D0 and Q0 we insert buffers of 1 nanosecond delay and make at the beginning make this present state 0 by asserting a signal.

Will count 0, 1, 2, 3 like that, that is the question, we have not control over the timing, we will go very fast as depending on the delay, but will it work, so the answer is that it will work there is no issue, but should remember there is an issue here which I have mentioned earlier the decoding circuit of D2, D1, D0 might incur different delays with respect to this point ok.

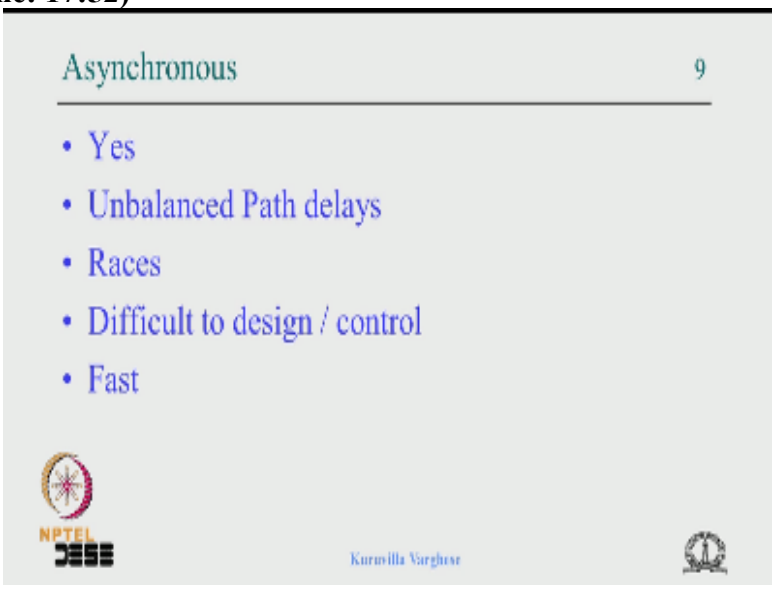
We might put 1 nanosecond buffers exactly here, but when from the present state the next state the 3 circuit 1 for D2, one for D1, one for D0, may be 1 is slower than the other, that can create problem, so take this case we have a count 1 here 0, 0, 1 which is going to count 2, which is 0, 1, 0. Now what happens is that there is a change in Q0, Q1 and Q2 does not change. Assume that with Q1 is faster than Q0.

So what happens is that before Q0 transit from 1-0, Q1 transit from 0-1, so in between going from 0 1-2 there is for a brief duration there is a state call 3 or the count call 3, now that remains there for this propagation delay, then the whole gain can change the the next state becomes 3 and we lose control, so the trouble with asynchronous circuit that is very fast when you want very high speed operation this is good.

But wherever there is feedback it creates problem because there is unbalance path delay in the multiple path and there could be raises and we call this raise 2 output can raise and there could be intermediate values things can go on. So that is why we use as far as synchronous

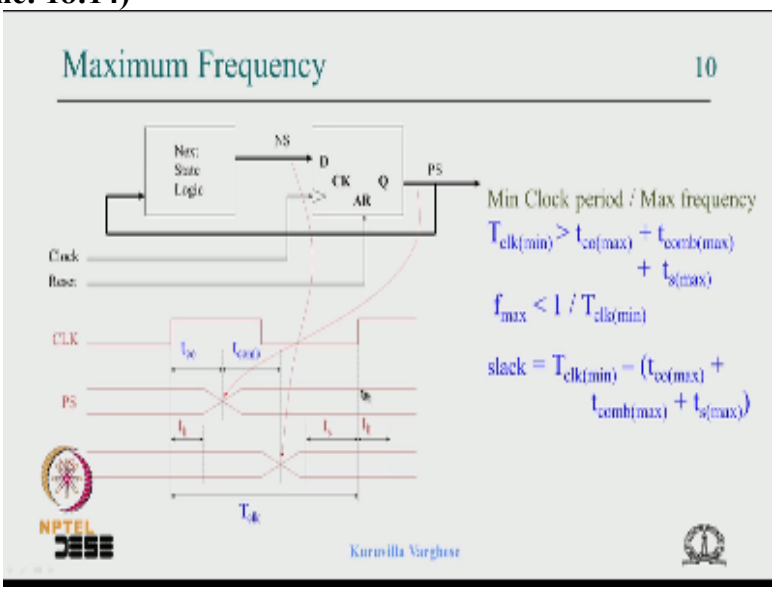
circuit particularly when there is a feedback it is easy to design synchronous circuit than a synchronous circuit , but whenever there is no kind of feedback we can one can go for asynchronous circuit, so that is summarise here.

(Refer Slide Time: 17:52)



Yes the circuit is possible but the trouble is unbalance path delay that could be races in output, it is difficult to design and control, it is very fast ok, so that is about the synchronous counter or asynchronous sequential circuits.

(Refer Slide Time: 18:14)



So let us move on to the timing part of the synchronous counter design. So the question I want to ask you is that what is the maximum frequency of this counter, that means we have a clock how much a frequency the maximum you can apply to this clock ok**,** that one that has to be determined because above which it may not work ok. So definitely that depends on the delays of the blocks involved.

The delay of the flip flop and delay of the next stage logic and you for a moment you think a clock edge active clock edge comes here then the present state will change after TCO delay, then that comes here it propagates through the next state logic and comes here ok. So there is a TCO delay and tcom delay and there are three parts you know D2, D1, D0, not only 3 path I mean 3 outputs.

But they could be 9 paths, we will see that 9 paths because for each D2 the input can come from Q2, Q1 or Q0, there are 9 path and we have to take the maximum delay and now you know that before the next clock edge comes data has to be steady here sometime before, call setup time. So we have seen that for the flip flop to work properly, the data should appear here sometime before the clock edge.

And it should remain sometime after the clock edge, so the cock period should accommodate all that delays and we will see it picked really, so let us take this as a cock and a clock ewdge comes here, so say it takes this much time the TCO min for the data to changes there, you see the present state changes there and then it propagates through this combinational logic and the data at this point data changes here after a combinational delays.

So that is what I have shown after the first clock edge the data changes at this point after TCO delay and at this point after the Tcom delay so the data raise here, the TCO+Tcom delay after clock edge, but we know that before the next clock edge comes the data has to be here sometime before calls setup time ok, so now that means there has to some gap at least this line and this line can touch.

But it cannot go before, ok, so that means the total clock period the T clock should accommodate TCO, Tcom and T setup ok, so that is what is shown here the minimum clock here, minimum you can go should be greater than TCO, maximum clock out time Tcom maximum like what there are many paths, we have taken maximum delay and at the maximum delay maximum setup time and that give the minimum you can go.

So the maximum of frequency will be less than 1/Tclock because this is becoming this is the maximum value so the frequency maximum we are taking it in the denominator, so F marks should be less than 1/Tclock, but it is not enough to like in this like it is not good to keep

equal then any temperature variation if the propagation delay become more than this initiate so we normally give a margin and that margin is cause lack which is the all these 3 quantities subtracted from Tclock will give you a slack ok.

But the surprising thing is that in this equation for the whole time does not picture at all okay does not look like, does not matter the whole time does not matter for the maximum frequency maybe at the beginning it is little bit of surprise, but then we have to we have to see whether the whole time is kind of max in this case and what is the condition that it can be violated.

So let us look at this picture, the whole time say when a clock edge comes okay here you say less let us take this clock edge at when the clock edge comes to say that whatever that I was there before the clock edge that should remain for some time afterwards ok, but we know that when a clock edge comes the data this point is going to change only after TCO+Tcom dealy okay.

So the whole time if TCO+Tcom you know that the next state will change only after this propagation here then it propagate here, then you changes there, so it is enough if the whole time is less than TCO+Tcom or other way TCO+Tcom should be greater than that whole time and mind you now we have to take the minimum PCO, and the maximum frequency is we picked the maximum delay of the TCO and Tcom to accommodate the Tcom.

But in this case the minimum is the constraint because there could be a case where one of the Q0 is very fast, TCQ0 is fast and from Q02 one of the input is very fast then there could be a problem, so we say TCO means Tcom in should be great than the T whole max ok, now that looks really looks kind of an impossible condition the violate and now if you pick a flip flop ok.

And just analyse the timing of the flip flop and if the flip flop itself TCO is greater than the T hold of the flip flop this cannot be violated at all. The the TCO min or at least TCO is always greater than the whole, there is no question of this violation ok. So looks like whole time can be violated but in whatever in this our analysis there is something hidden which we have not states.

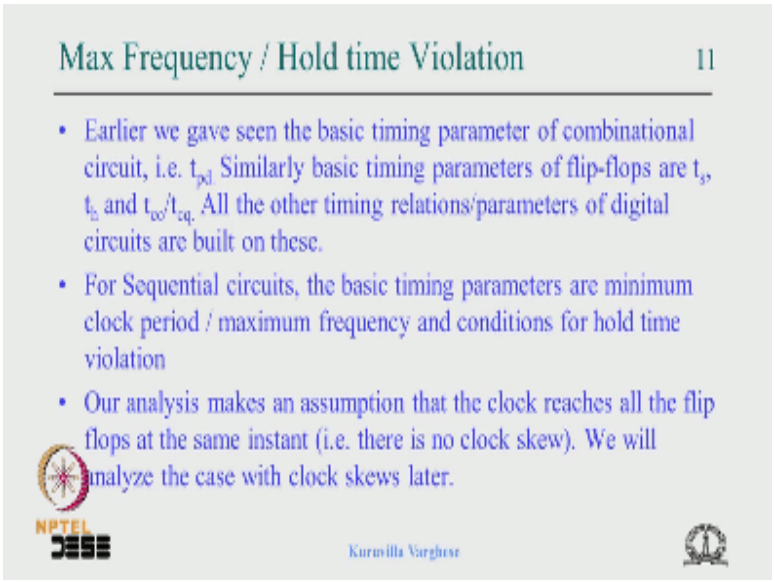
So that is very important, so here we have assume that the clocks for there are 3 cocks going to 3 flip flops, but we assume that the clocks are arriving at each flip flop at the same time. So but in real life that may not happen and that is skew, the clock maybe skewed with respect to each other, that means the clock 2 maybe coming earlier than the clock 1 and clock 0 maybe arriving later than the clock 1 and so on ok.

So this equation will change and there could be whole time violation and there is clock skew and the worst case can happen when there is no combinational delay like assumed that for some reason the combinational delays more or nil and you already know that we build a shift register the one flip flop output will go to the next flip flop input, so in such case there is no combination delay.

And if there is a clock skew there can be whole time violation ok. So this is important do remember and we will analyse the case for the clock skew, but this is the mean as far as a sequential circuits concern the next important thing is the maximum frequency of operation and the whole time violation and these 2 things are the important timing parameter of sequential circuits.

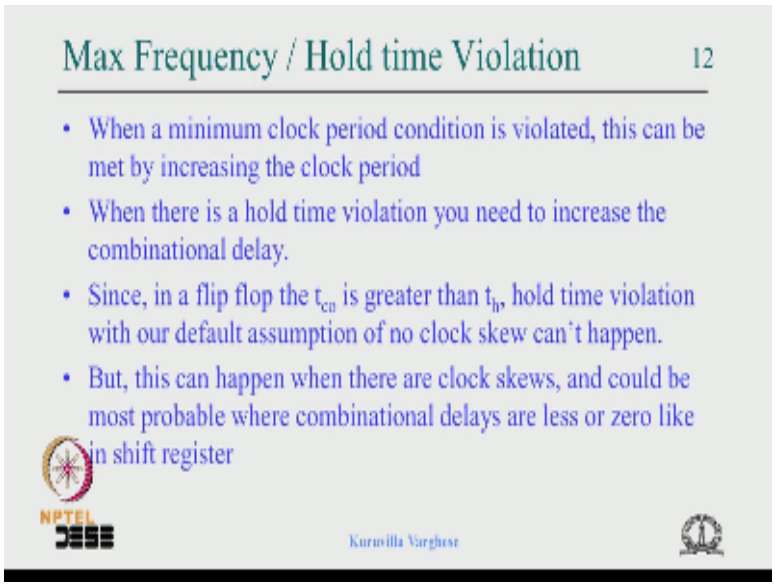
And sequential circuits and mind you that is built on the symbol combinational delay and flip flop timing parameters like TCO, T setup and T whole, so very simple things the propagation delay of the combination circuit and the propagation delay of the flip flop, the setup and hold time. The next level of the timing details are built on this the maximum frequency and the whole time violation.

(Refer Slide Time: 27:11)



So that is the next level that is the most important thing. So quickly I have stated whatever I have told these are the basic parameter and rest of all is built on that and we assume that the clock is not skewed in this case, when there is a clock skew we have to analyse it and you will analyse it later.

(Refer Slide Time: 27:32)

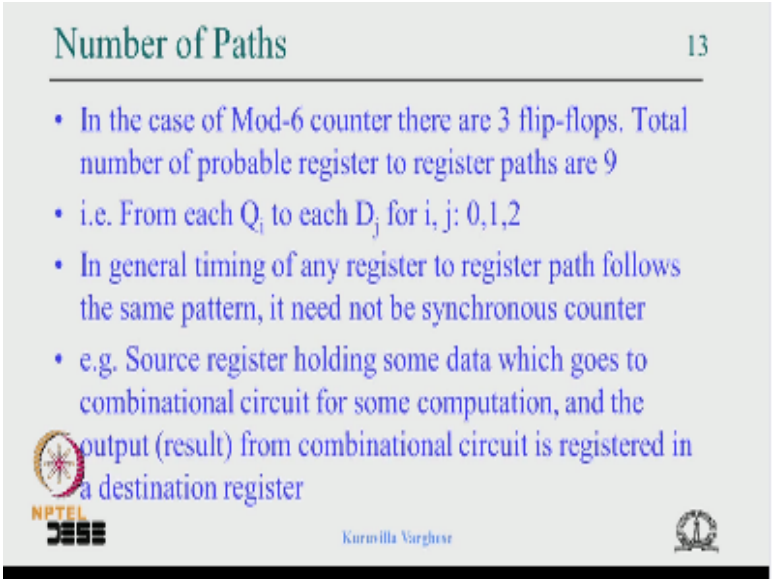


And the whole time violation okay one thing to remember suppose we build a secured like that and we give some cock we suppose we gave 500 mega hertz stock here which gives a 29 nanosecond of 1 mega, 2 nanosecond clock period and if this is violated then what to solve the problem of 2 kind of increase clock period. If this inequalities violated to solve it is to increase the clock period.

But assume that this is violated, TCO means class Tcom min is greater than the whole time, increase in the frequency would not help at all, because there is no T clock period in its inequality, so there is no point in. Suppose this is violated only ways to increase the combinational delay, so we may have to when there is a hold time violation we may have to introduce additional delay in the combinational part.

That is how to sort out the hold time violation, that is what I stated here when a minimum clock period conditional violated this can be made by increasing the cock period, but when there is a hold time violation you need to increase the combination daily. Again I stated in a flip flop TCO can be greater than the whole time and if the clock skew the whole time cannot be violated. But when there is a clock skew and when the combination delays minimum or 0 like in 1 shift register this whole time violation can happen.

(Refer Slide Time: 29:16)



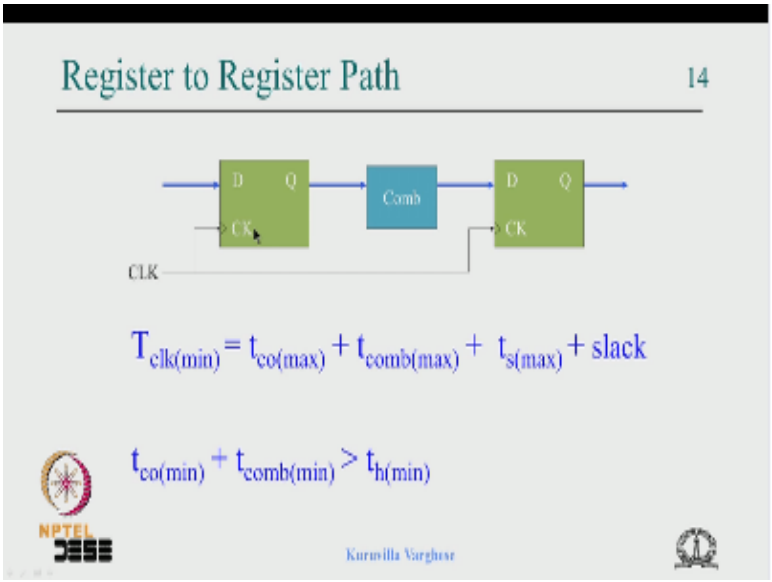
Then you have to introduce the additional long delay. So let us look at the number of paths probable paths in the previous circuit, so suppose in the case of a mod 6 counter there are 3 flip flops and total number of probable register to register paths are 9. Because for from each QI to each DJ there could be a path ok, it is a very simple sequence it may not be there, but in general there could be a path from like we know that the Q20 is a function of 3.

D0 is a function of Q2, Q1, Q0. So when built a circuit for that D0 there is a path from Q0 is a path from Q1,Q2, so 3 path. For each D0, D1 and D2 there are 9 paths. So essentially as far as the timing is concerned it is to analyse this registered to register path okay , it need not be

synchronous counter like in timing analysis be looked at from each register to one source register to do 1 destination register.

Path need to analyse for to find the maximum clock frequency and to find the whole time violation, so that is a basic gain. So in general when you look at the sequential circuit there is some sources say which is holding some value which is passing through a combinational circuits for some computation and it reaches the destination register and we have to make sure that they like inequalities are met.

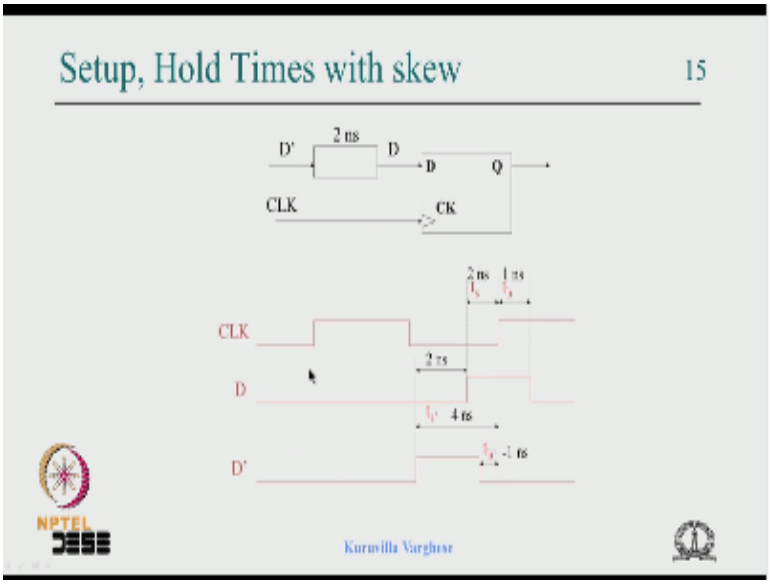
(Refer Slide Time: 30:53)



So here that there is a source register which supply the data to a destination register to a combination circuit, then the clock min should be greater than Tco+Tcomb+or =Tco+Tcomb+Tsetup with some margin and the whole time violation to avoid then you have the Tco min+Tcomb min should be greater than the hold minimum. So this is the most general case of the timing a registered to register path where and you can find the maximum clock frequency and the condition for hold time violation.

So we have abstracted it from the taking asynchronous an example and come to a very general sequential circuit in a synchronous counter anything can be analysed in this manner.

(Refer Slide Time: 31:50)



So let us move on so let us make things little more complicated of complex let us take a flip flop we know that the setup time and hold time is specified at the input that means the we know the timing parameters state that with respect to this clock there is set up time the data has to be set up sometime before, that has to be held sometime after. So we take an example of ceratin time being to nanosecond and hold time being 1 nanosecond.

Then the data bit with the minimum width of the data is like that, date is changing need from 0-1 just before 2 nanosecond, reminder for 2 nanosecond up to the clock and 1 more nanosecond and it changes ok. But many times like we may take this input on a chip like in on FPGA and we would like to know and that might like that a wire is going from the pin through some buffers and reached here.

So there could be additional delays in the data path as well as in the top path, but for analysis we will support this case, it is easy to analyse and grasp the concept and then we can you know you can put it together. So here this point in that deliver 2 nanosecond that means you give a date hear it appear here after 2 nanoseconds delay and when you supply clock assume that the clock has no delay.

We would like to know what is the setup time and hold time with respect to this point because we have control over this point. We are supplying the data from outside at this point and we are not worried about the setup and hold time here that does not help us because there is an additional 2 nanosecond delay. So whatever we put here appear here after 2 nanosecond., So you see this data here comes earlier by 2 nanosecond D.

So I move the data at this point by 2 nanosecond to the left. Now assume the clock is same, so what is the new setup time at this point d dash. Now the setup time you see with respect to this clock set up time is increased because a data this point has to be set up 4 nanosecond before this 2 nanoseconds plus this 2 nanosecond delay data D dash as we set up 4 nanosecond before the actual clock.

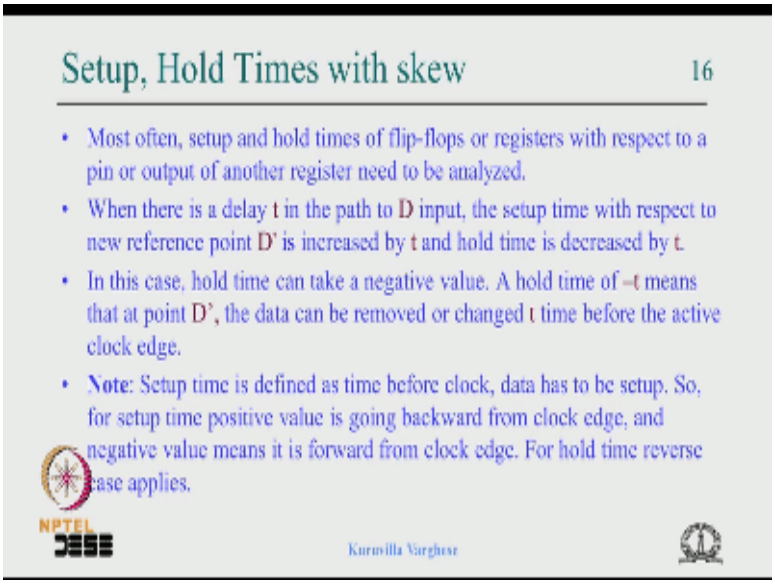
So this 2 nanosecond is added to the setup time, setup time is increase, now whole time you see whole time is 1 nanosecond towards left ok. Now one thing to remember is that channel setup time is defined as the time before the clock edge the date has to be set up. So and the whole time is defined as the data has to be held after the clock edge, but in this case you know that the data is removed before the clock edge.

So it becomes negative, so whatever was 2 nanosecond this delay is you know you have 1-2 you get -1 nanosecond and do not worry about the -1 nanosecond it essentially means that you can remove the data, normally you remove the data after the clock edge, when the whole become negative in the opposite direction it means that the you remove the data at this point even before the clock edge ok.

So if you do that at this point correctly is a data will match this window of the setup and hold time. So it essentially means whenever you have a delay in the data path or this path then the set up timing increase and the whole time decreases and the whole time can become negative, you need not worried means that the data can be removed even before the actual clock edge.

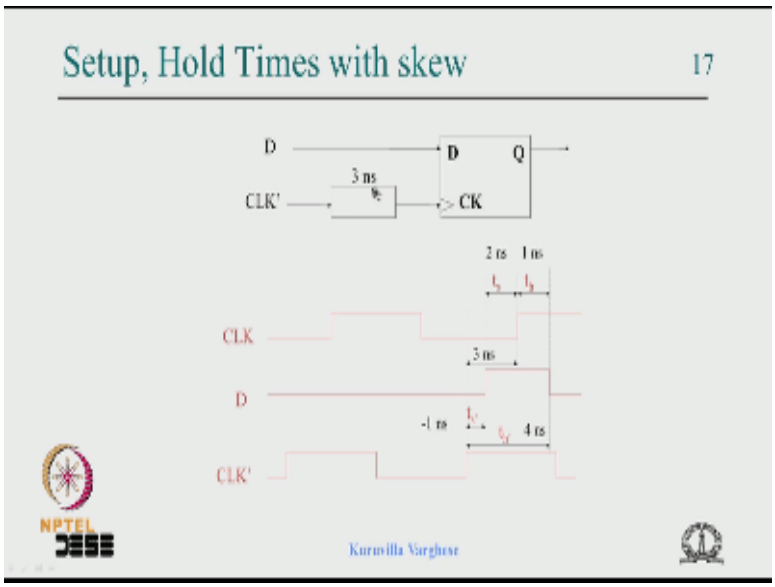
So that the input of the flip flop everything happens very correctly, so that the case with when there is a skew in the data path let us of analyse the other case that when there is a skew in the clock path what happens okay.

(Refer Slide Time: 36:21)



So this is summarised whatever I have told a delay in the data path will increase the setup time, decrease old time -T means data can be removed before the active clock edge set up time is defined before the set up before the clock edge and the whole time is defined after the clock edge, so whenever it goes in the opposite direction it is a negative value. So that is what it means so.

(Refer Slide Time: 36:48)



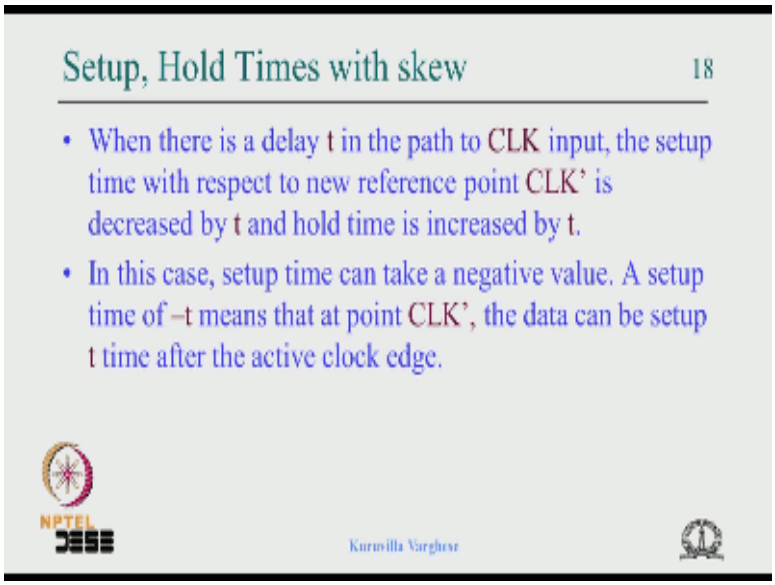
Let us take the next case where there is a skew in the clock path and that means we have a pin here we give the clock and that suppose the delay of 3 nanosecond before reaching the this particular flip flop and we would like to know what is the relation of the data setup time and hold time with respect to this clock, but that is where we have control. So the actual clock here is shown here.

This is the clock and it has a setup time with respect to this point 2 nanosecond and hold time 1 nanosecond, so in the normal case the data appear here with you know because there is a 2 nanosecond here and the clock you see here the clock dash will appear earlier to this clock by 3 nanosecond. So that it will appear correctly here. So now you can see that this edge of the clock is moved left by 3 nanosecond ok.

So now at this point our timing is with respect to the new clock or the clock dash. Now you see the data has to be set up at this point after the active clock edge, so the setup time was 2 nanosecond but now it has become 2-3 nanosecond which is -1 nanosecond that means data is set up at this point after the clock edge with respect to clock point. So that is a minus, but it has to be help for 1 + 3 nanosecond 4 nanosecond.

So the whole time is increased by 3 nanosecond by this Q, so it means that if there is a skew in the clock path the set up time is reduced by that much and the whole time is increase and in this case setup time can become negative it means at that point with respect to new clock the data is set up after the clock appears here, so that at this point everything you know works correctly.

(Refer Slide Time: 38:51)



So that is what is summarized here when there is a delay the setup time with the new reference point is decrease and hold time is increased and negative setup time would mean that it can be set up after the reference clock ok.

(Refer Slide Time: 39:08)

That is what about, so this is what I have you know covered in the last few slides basically we have looked at the structure of the synchronous sequential circuits we have taken an example of a synchronous counter, we have looked at the main components like the flop the, next year logic you know which decode the present state make the next state, so basically 2 components next state logic which decode the present state and the inputs.

When you give inputs it gives input to the next stage logic that it can control the next state in a decide way like synchronous reset load and up down and things like there any complex operation. Only thing is that we change the truth table accordingly then you get the functionality and we also said that you know that I am not lose sight of the detail view because the lot of parts within that simple diagram which is not be lost sight.

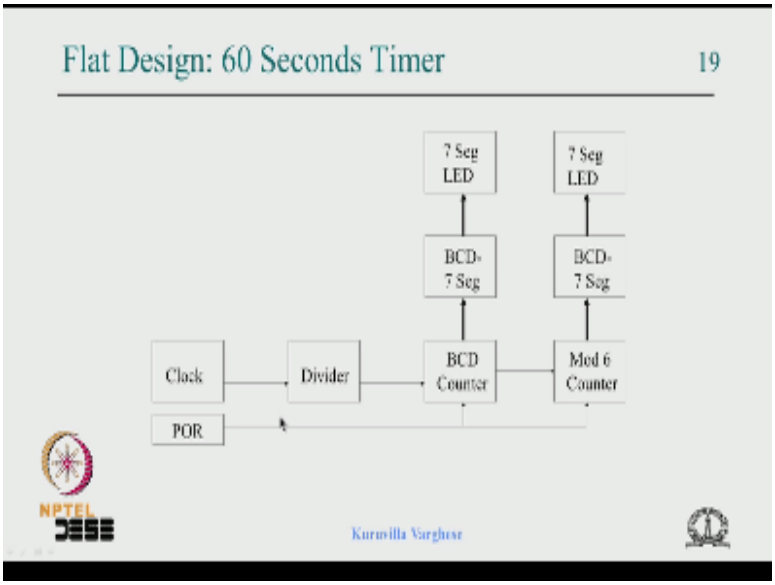
And we have looked at the asynchronous circuit and we have seen that could be races because there is unbalance path delay in the keys of feedback. So it is difficult to design and control service stick to the synchronous sequential circuits, synchronous counter in that case we have looked at the timing and we have looked at the condition for the maximum frequency operation.

We have looked at the condition for the hold time violation. These are the two main things which is built on the basic timing parameters, then we have looked at the case where there is what happens to the setup time and hold time when there is a skew in the setup path, when there is a skew in the clock path what happens with the set up and hold time with the new reference point.

That is very useful in analysis if you do not grass plant many times you can analyse things properly. So let us move forward, let us look at how to go about designing some system circuit.

So let us move on to say take an example suppose I want to design a 60 second simple timer ok, which counts likes a 0 to 60 that is all and display maybe for some practical purposes like we have counter which counts little accurately from 0 to 60 in sequence 0 to 59 mostly counter verifies like every one second you should think, so how to go about designing that it is a very simple thing.

(Refer Slide Time: 41:56)



Let us start with a simple design, so how do we design something, ok. So let us look at the to the slide, so the first thing you need is a 1 second clock, but you do not get a 1 second oscillator you to build, oscillator are high frequency oscillator you cannot have a crystal vibrating at 1 second period. So you have to have high frequency oscillator maybe 1 megahertz or 10 megahertz or final megahertz then you divide using counters to get a 1 seconds falls.

The moment you have one second clock you can give it to a counter and start counting. So let us put that in the picture, so we have a clock oscillator which is of high frequency maybe let us assume 1 megahertz clock and then we put a divider. So if you put a 1 megahertz clock you know that it has to be divided by some around you know nearing around twice 20.

Twice 20 is a binary number more than one min but you need at least 20 flip flops with some more to like Mod 1 may counter to divide this and here you get a 1 second falls, ok so we will make a very simple design, so for count from 0 to 59 we will put a counter 2 part 1 least significant digit will come from 0 to 9 I am going to reaches 9, it will increment the MOD 6 counter which is counting from 0-1 like that.

So we will put a BCD counter with counts the least significant digit when it reaches 9 it will increment a Mod 6 counter which goes from 0 then when it is 9 it goes to 1 and so on. Now you have to display it so we need a 7 segment from kind of display simple thing let us take a 7 segment kind of display a simple thing let us take a 7 segment LED. So we need a BCD to 7 segment decoder.

Here we can you still use a BCD 7 segment decoder or a mod 6 to 7 segment decoder. Then we can write some LED 2 light it up ok it is very simple when you have given some simple design like this you are able to design for and you know you start with the clock input the divide a put account put some decoder things like that. When you are through with visual design simple design like this can happen in a very very from online to other.

And you take a paper and do it and if you try to design this in using hardware description language it is very few statement, maybe you would not stay true statement for this divide a few statement for this and something for this and you are you connected the output to the LED is and it works perfectly fine you can add different control like reset and what not to so. All that we have learned you know you can incorporate many other controls in the counter if you need ok.

But I talked about at the beginning of the lecture there is function, there is timing, there is electrical pack. So let us in investigate weather that does not matter here, what is the function of a function we have designed ok. So is there any timing related issues here ok or even function like area such issues can be address in a simple case like this ok. So like if you look at this counter it count very low frequency.

Like it counts a kind of 1 seconds, so is not a very high frequency design but you take this divide you see the clock of the divider is 1 megahertz. So at least 1 megahertz, so this divider is working at a high frequency than this counter, a flip flop here are clock at 1 megahertz , the some flip flops are clock at one megahertz depending on the design, so this with lot of power here ok.

So that is an issue to keep in mind know there is a little highway frequency part compare to this, secondly I said this need to be accurate, how do we make sure that this one second counter is accurate and how to improve the accuracy so that you should know that suppose this clock source has some drift as I goes of an 1 megahertz you know clock**,** suppose it grips little bit.

You know some parts per million drift then you know that that drift is divided by this divider so since it is drifting in the higher frequency the drift in the low frequency is so much divider

for the actress is improve, so it shows that if you instead of 1 megahertz clock if you go to 10 megahertz clock with the same drift you will get a better accuracy in terms of the accuracy of 1 second.

So if it properly is worthwhile to go for a higher clock frequency then you should know that the divider size will increase like if you go to 10megahertz the number of flip flops in this and the frequency of operational in this and the participation will increase and suppose if it is a battery operated thing at the high accuracy might mean know the battery life and that you should know okay.

And similarly look at the area of the BCD counter you know that it count from 0 to 9 needs 4 flip flops and Mod 6 counter me 3 flip flops and that means it is 7 flip flop. So can we reduce number of flip flops ok. So things for a while this design take 7 flip flops, can we reduce for the number of flip flops for the same functionality. So the answer is yes because you have your splits the counter like a mod 9 counter or BCD counter and a mod 6 counter.

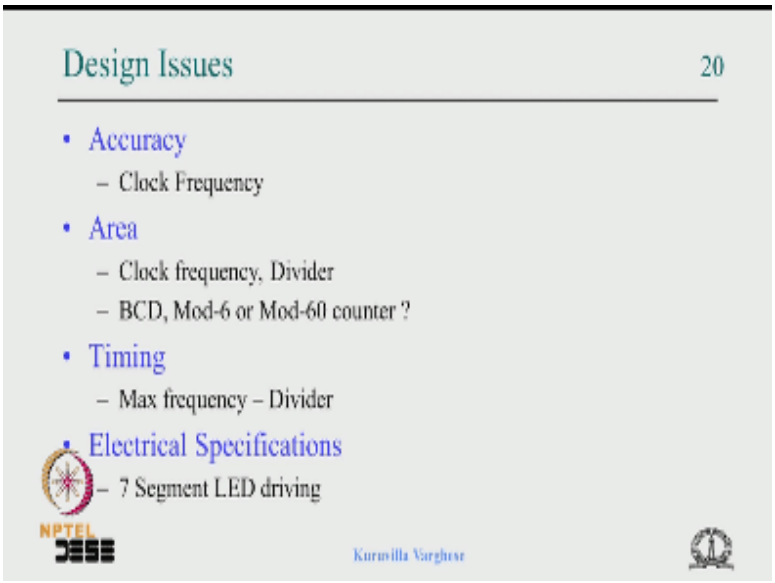
There are 4 bit out here and 3 but out here, so for a while let us assume what we need is a account of which come from 0 to 59, so assume that if we are not splitting it we are putting it together ok that it counts it we can put 6 flip flops which can count up to say 64 and but we will decide said that it count up to 59 and then we instead of having BCD and Mod 6 counter.

We design a counter which is MOD 60 counter that means we put 6 flip flops and redesign then we say one flip flop but the complexity is that now here we had a BCD to 7 segment Decoder decoding 4-7 like 4-7 Decoder here it is 3-7 Decoder. But in that case we need Mod 62, 2 7 segment LED Decoder okay that means we will have some 6 lines here which is using a combinational circuit we have decoders to 7 lines.

Maybe we will we do not know the complexity of that Decoder depends on the input output pattern. We cannot say anything about the redundancy. So if you go for a mod 3 counter we may save 1 flip flop here, but maybe the Decoder here can become more complex and their separate decoder I do not know unless you literally try and find out you should not assume that there will be simpler than this case may not be.

So one has to verify that whether that be a simpler thing and definitely you know that here it is driving the LED supposed LED to be very bright will take some current so maybe this BCD to 7 segment output pins has to give some 2200 milliampere for it to be visible at a distance so there is an electrical spark which needs higher than drive here so even in a symbol secured like this there are issues of the high frequency power dissipation.

(Refer Slide Time: 50:56)



The area conflicting requirements of area we do not know we cannot apply or you say that this is a better scheme counter or anything like that. So my advice is that saying something simple you should not assume that is very simple there are no issues that maybe issues like the accuracy as we see area, then the timing, then electrical specifications and things like. So this we should keep in mind whatever the design is small or big there could be these issues to be thought about.

And the text book may not teach you all this, it is for you to apply whatever you have learned the basic and applied to the particular case and that comes with the experience comes with many a times working on simple projects, implementing things then only you will be sensitize, then all will be quickly able to catch on this issue and solve the issue that comes with the experience, comes with thinking in learning it is not that you learn everything.

You understand everything, but thinking is an important thing and what you have learnt in the textbook have to apply to the real life then only you can find the issues in real life and that is that true engineer is a one who find the issues and quickly identify the problem areas and try

to solve it and there is no one can say the systematic approach will help but may time that alone will not help in a complex system.

Some issue come up many times have to quickly use a very oft-repeated free we have to zoom in to the to the issue and sort it out and that comes sometime very quickly without I like you cannot go from are remembering all the issues and one by one looking at it many times it clicks in the mind that comes with the experience that comes with your own deep understanding of the issues and the mind works very quickly seeing the behaviour you will be able to quickly debug.

I know that is one issue I want to emphasize with the real life circuit the debugging is very important that when you have a problem you should be able to quickly sorted out and debugging is not limited to the hardware it can be the software it can be mechanical system it can be true with the mathematics know you do not call and debugging a theorem or debugging a proof.

But essentially that is what is done you are like you are trying to suppose you are you have a hypothesis you are trying to prove that and you try work out something goes wrong that what things go wrong should. You out what is what is wrong with the hypothesis for what is wrong with your approach to proving it and think I have. So I would say that is debugging in that is debugging the maths, so you can debug a mechanical system something is going wrong something is not moving properly of to debug.

The concept ideas the technique the thinking is same respective of whether does abstract whether it is real life some time you will be surprised to see very lonely technician applying the mind you can learn a lot when you see the real technicians had some problem in the domain they can people go in the correct sequence of assembling something, dismantling something, trying to locate the issue with the with the problem and sort it out.

This is an eye opener maybe they are experienced that doing the things then they are out but the engineering domain need such approach sinking deep understanding experience and learning with the practical example always tried to implement and try to if something goes wrong or try to find out what is going wrong. So that you go wrong less time is many times people repeat that have to learn from mistake.

Yes have to learn from mistake but in your term career in your life you should make less nothing like you should learn as much as from the mistake that means wants you make something goes wrong then you should analyse it I literally so that that is not repeated. So the next slide I will give a just sensitize we are going to a bigger design ok. Really we will take a complex design and see how to know how in a complex case design can be done.

So I did not start the lecture just now because I have to terminate it half way, so the next step we have taken a very simple case and see how the digital design is done the next step is that have we will take a reasonably complex design and see how the design methodology can be applied in terms of the function, timing, the issues involved how to go about knowing with the minimum knowledge how you can design.

This is what I am going to power on the next lecture. So I have covered some minimal things but these are very important things have covered. So please go back review try to graph in your mind and think about it, work on a paper try to understand asked questions yourself like you should have the write it down, now this is a video lecture I have no way to interact with you.

But then you write down your question logically then try to answer it look for the answer properly can contact me if you are stuck very much you can contact me in email I will try to answer. So I wish you all the best and in the next lecture we will move with a complex case and thank you.