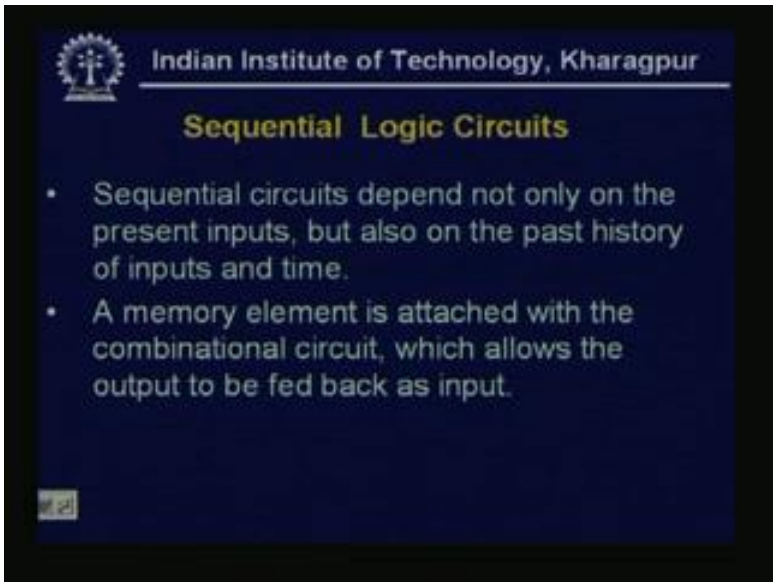
Digital Systems Design
Prof. D. Roychoudhury
Department of Computer Science and Engineering
Indian Institute of Technology, Kharagpur

Lecture - 21
Synchronous Sequential Circuit Design

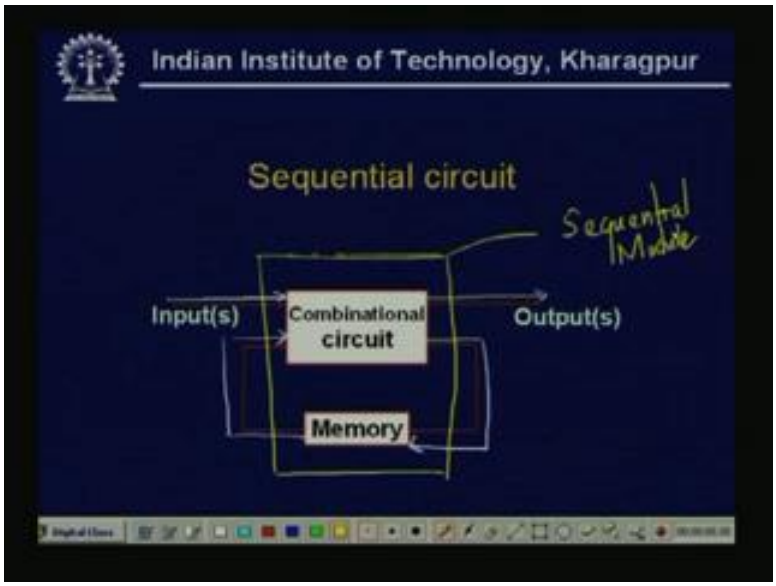
Already, we have learnt how to design the combinational circuit design or simple combinational circuits, complex combinational circuits. How we can synthesize the combinational circuits from, given Boolean equations or given a problem? How we can design a combinational circuit? Today, we will start discussion on sequential circuit design particularly the Synchronous Sequential Circuit Design. Now, first we consider the design of sequential modules, what do you mean by sequential logic circuits.

(Refer Slide Time: 01:29)



Now, in combinational circuits, we have seen that the combination output of the circuit depends only on the inputs of the particular circuit. Now, the sequential circuits depend not only on the present inputs. But also on the past history of inputs and time, means that the present output depends the present input. As well as, the previous state and that state is a time of or is a function of the inputs and time, so that is why it is written that past history of inputs and time. Now, a memory element is attached with the combinational circuit, which allows the output to be fed back as input, mainly that how the sequential blocks are developed.

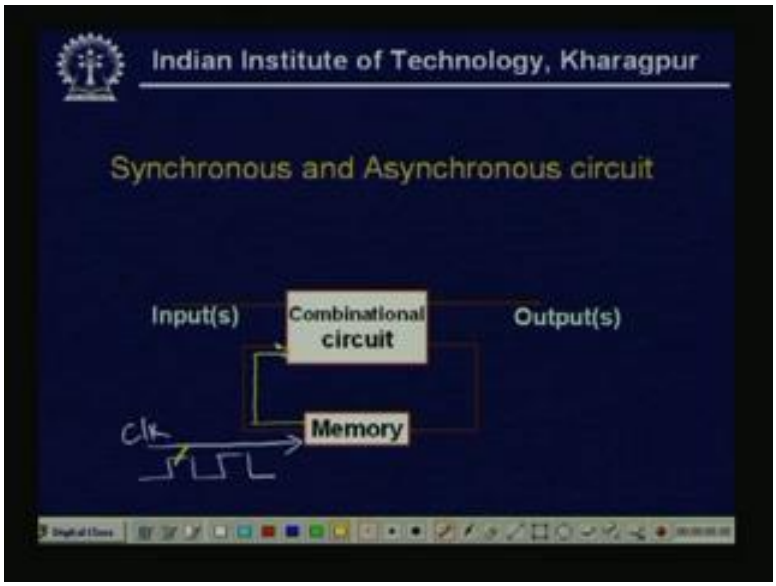
(Refer Slide Time: 02:33)



Now, see the normally this is the design of a simple sequential modules, one combinational circuit is there and inputs all ready we have seen that, these are the present inputs. Now, for combinational circuit these outputs are the only the function of present input, but here one memory element is attached. And that it is the sum of the outputs are attached with the memory and it is then fed back to the combinational circuit.

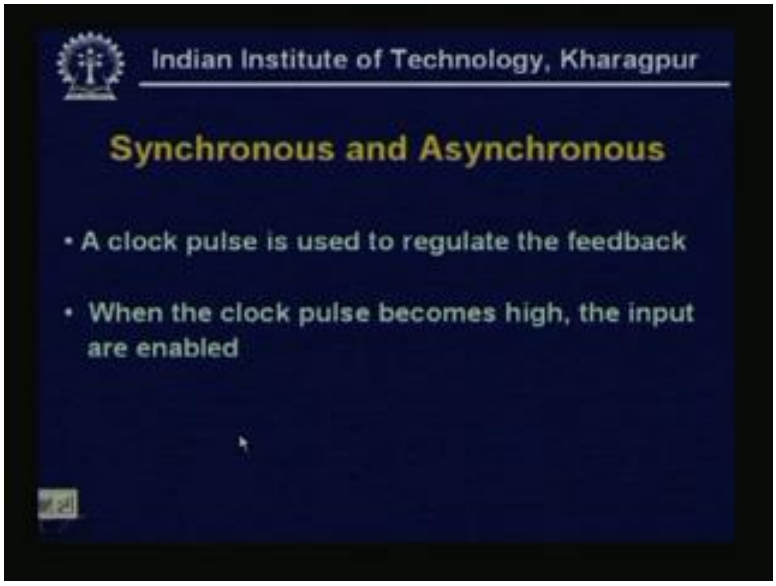
So, these thing, these combinational circuit attached with a memory elements which feedbacks some of the outputs is a sequential module. So, this is the basic module of a sequential circuit, now what do you mean by synchronous and asynchronous circuits.

(Refer Slide Time: 04:09)



Now, the with the memory elements we will attach a clock, so whatever was there in the sequential circuits the present input is there the outputs are there, some of the outputs attached with the memory element and they are fed back to the memory. Now, we are another input, this is a clock input that is given to the memory, so what it will be doing.

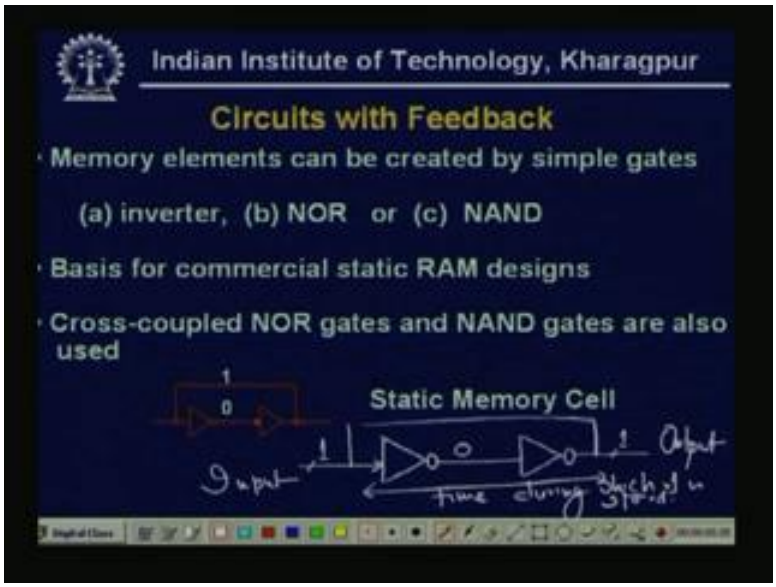
(Refer Slide Time: 05:04)



See the clock pulse is used to regulate the feedback, when the clock pulse becomes high the inputs are enabled. So, if we see the picture, so when this clock pulse say here we are giving say one clock pulse, so when this clock pulse is high means this is high, then only

this inputs are this memory inputs are enabled to the combinational circuit, so this is the circuitry description the sequential module description. So, mainly there are another control input and that is the clock pulse is attached with the memory element.

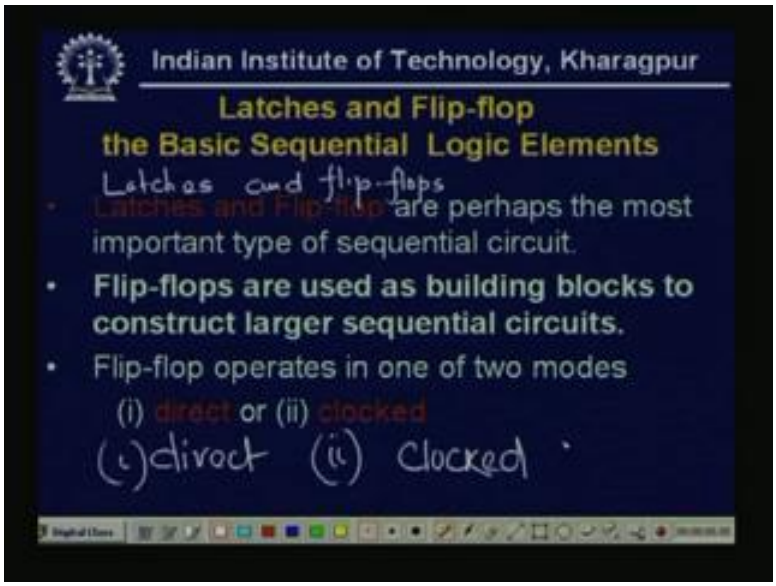
(Refer Slide Time: 06:11)



Now, this circuits with feedback, so memory elements can be created by simple gates this is the inverter NOR gate and NAND gate. And mainly this is the basis for commercial static RAM designs, means the one bit memory element. And cross coupled NOR gates NAND gates that are also used that means, inverter NOR gate NAND gate mainly, they are the they are used to design one bit memory.

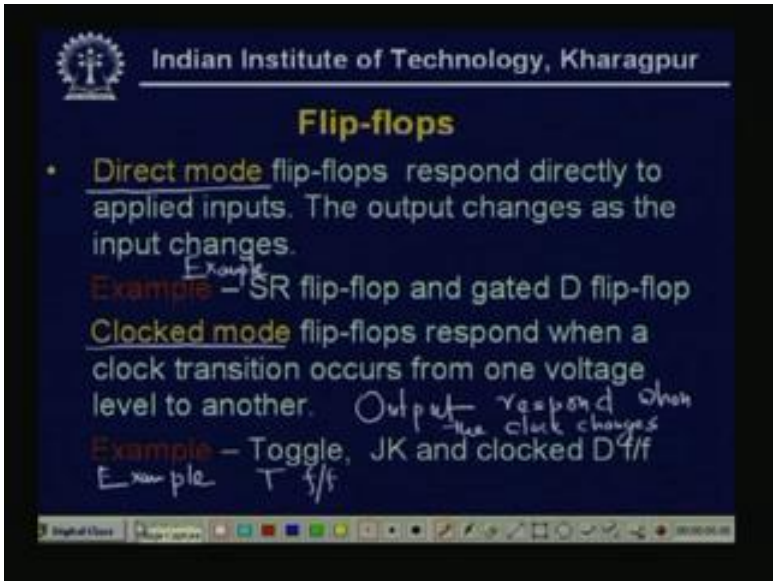
First we see the simplest design, see we consider here 1 inverter we if we draw say first 1 inverter is there we apply 1 1, so what will be the output, output will be 0. Now, another inverter is cascaded with this, so it will be the output will be 1, now if it is feedback. So, here I am giving 1 and that 1 and getting here and see here the concept is, as if the delay of two inverters up to this time this. The some of these delays up to this time this 1 is stored in this circuit, my input is 1 input is 1 my output is 1 and as if this is the time and stored this one the time during which the 1 is stored. So, this simple concept is being used. Now, before we read the in details the design of flip flops or this memory elements using NOR and NAND circuit first we define a flip flop and the latches.

(Refer Slide Time: 08:42)



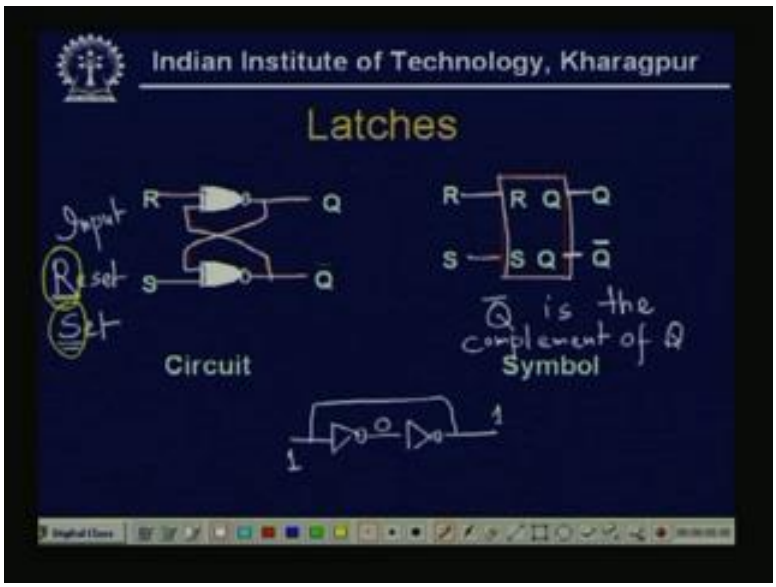
So, mainly the latches and the flip flop they are the basic sequential logic elements, now the latches and flip flops mainly, the latches and flip flops they are the most important type of sequential circuit. So, flip flops are used as building blocks to construct larger sequential circuits and flip flop operates in one of two modes, one is called the direct mode and another is the clocked mode. Now, first we see the definition of this direct mode flip flops and the clocked mode flip flops.

(Refer Slide Time: 09:58)



Now, direct mode flip flops that respond directly to the applied inputs means the output changes as the whenever, the input changes. And the example is the we will see later that SR flip flop examples are SR flip flop and gated the D flip flop, now the clocked mode flip flops respond when a clock transition occurs that means, there must be 1 clock input in the circuit. Already we have seen that in the sequential module, 1 clock input we can give to the memory elements. So, when a clock transition occurs from one voltage level to another that time only the output responses. So, the output respond when the clock changes, now the examples are the T flip flop or sometimes we called the toggle flip flop it is normally called T flip flop, when JK and clocked D flip flop. Now, one by one we will see the different type of flip flops.

(Refer Slide Time: 11:46)



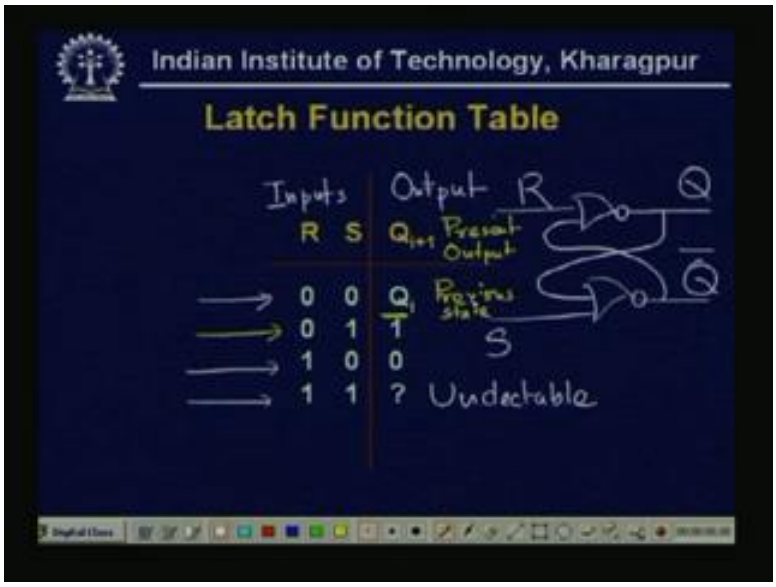
Now, first we see the latches because that is one basic elements, so we consider all ready I mentioned that the way I have one bit memory we have designed using inverter, say two cascaded inverter see here two cascaded inverter and the output is feedback. If I give 1 intermediate 0 then it will be a 1, now I am taking 2 NOR gates, so this is 1 NOR gate this is 1 NOR gate. There are 2 inputs one is called the R means the reset, another is set S. So, normally these are the 2 inputs R and S, the reset and set reset means R set is S, so this is the 2 inputs.

Now, the circuit made that output of the first NOR gate, that is being fed back as the input of the NOR gate where S is another input. Similarly, the output of the S that is

being fed back to the NOR gate where R is another input. So, thus this is the circuit that means, two NOR gates we have taken R is 1 input S is input of the two NOR gates and the other inputs are actually the output of S and R or output of the two NOR gates.

So, the symbol normally used is that one rectangular box the 2 inputs are R and S 2 outputs are Q and Q bar Q bar that means, Q bar is the complement of Q. So, the 2 outputs is are the complements and the these are the symbols.

(Refer Slide Time: 15:00)

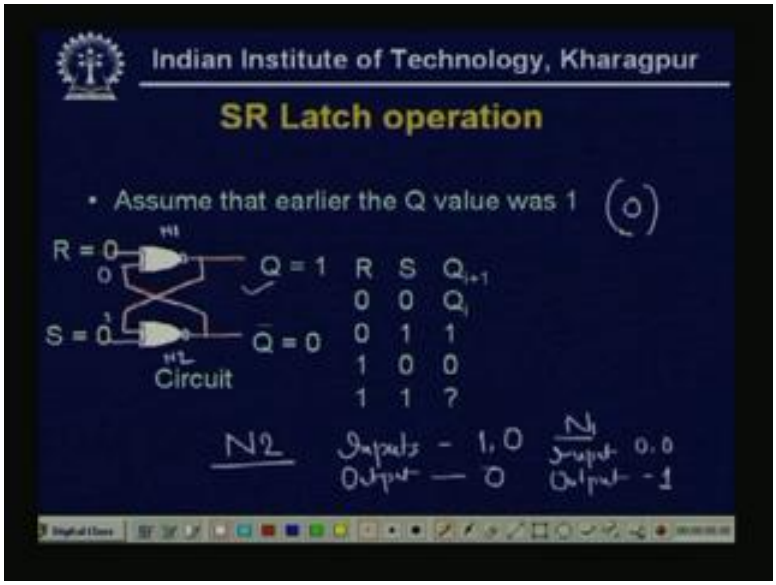


Now, we see the function table, see that these are my inputs of the circuits these are the inputs, one of the output Q this is my one of the output circuit I thing. We remember that two NOR gates are there one R inputs, one S input and the output of this is fed similarly, the output of this is fed this is Q this Q bar. Now, this is the function table that when the inputs are 0 0 then it is denoted as Q i output is Q i plus 1. Means we can write that this is the present state, say this is my as all ready we have mentioned that the present output. Or better I write that present output is a function of the present input and the, so this is my present output and the previous state.

So, this is my present output and this Q i we denote, that sum of the previous state that previous state. So, from where we have started after that if we give the R S inputs at 0 0 then the it will be the previous state. So, if it is 0 1 now, if I give 0 1 then output is 1 if it is 1 0 it is 0 0 and if it is 1 1, then question mark means this is undetectable this is

undetectable. So, now each of the input cases we see each of cases that how the function table forms by this circuit.

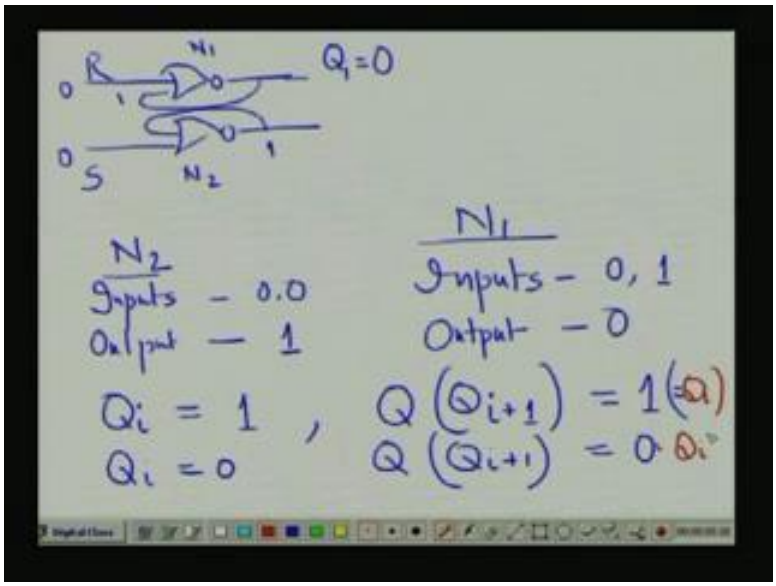
(Refer Slide Time: 17:55)



See first we assume that the first case that R equal to 0 and S equal to 0, so input is fed, so R is 0 and S is 0 this is my circuit. Now, if the earlier Q value was 1 and then that if R equal to 0 what will happen? See earlier Q value was 1. So, 1 must come here, but this is 0, so what will be the outputs say if we mark the first NOR gate as the N 1 or the top NOR gate is the N 1. But we will now get as the N 2 then what will be the function of the or what will be the input of N 2.

So, if we consider N 2 the inputs are inputs are 1 0 then the output will be, because it is a NOR, so output will be 0, so that means, here 1 0 will come. So, for N 1 for N 1 the inputs are 0 0 output is 1, so that is why it is written as 1. Now, what will happen it Q equal to 0 say that means, if we assume that Q value was a 0 if we assume that Q value was a 0.

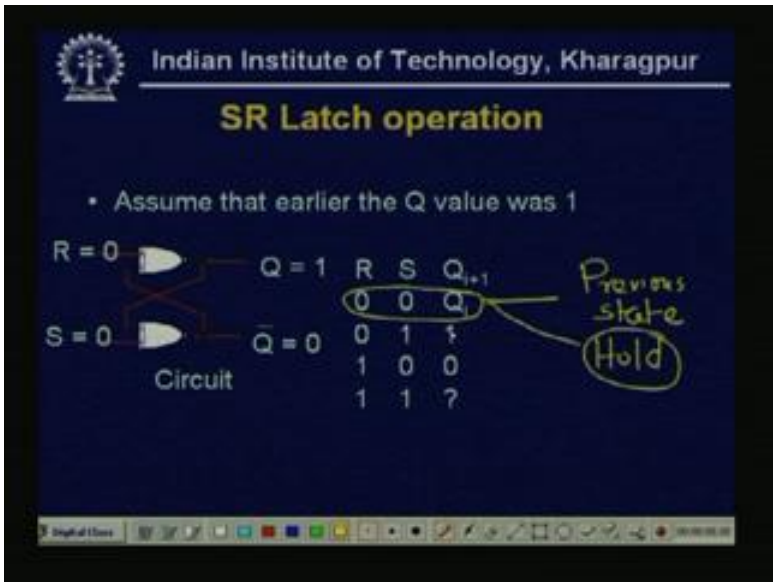
(Refer Slide Time: 20:07)



Now, if my Q value again if we draw the circuit say this is my NOR gate this is my R input, another S input it is fed back this is also feedback. Now, we have assumed that this Q equal to actually Q i means the previous state that equal 0. So, far, so far N 2 the inputs are 0 0 0 0 output is 1, so what will happen for N 1 this 1 will go here. So the inputs for 1 the inputs are 0 1, because the output of N 2 is 1 and that is being fed back to the 1 of the input of N 1 NOR gate and the output will be 0.

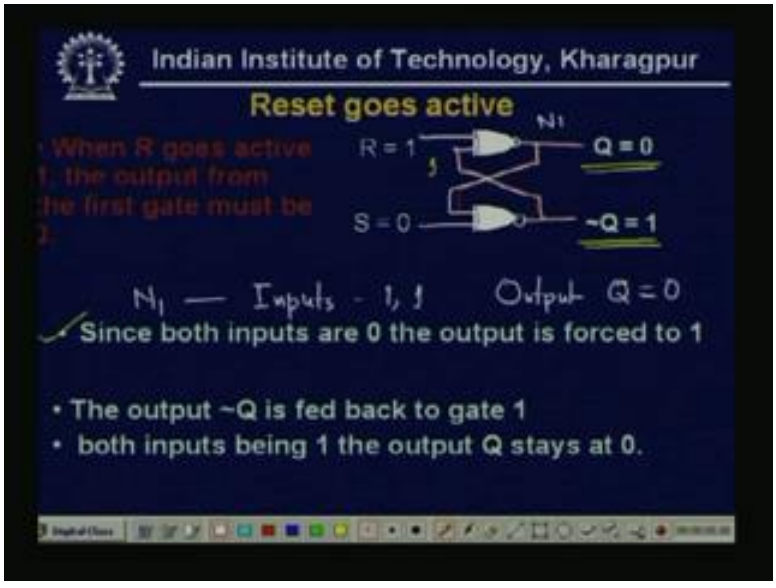
So, what we have seen when the previous output of the circuit, so previous output of the circuits say Q i that was 1 my the present output Q or we can write Q i plus 1. If previous we can denote Q i then the new current 1 is Q i plus 1 that will be 1. If it is Q i equal to 0 just now, what have seen then the present output you can denote Q i plus 1 and that is 0, so that is equal to equal to Q i and this is equal to Q i.

(Refer Slide Time: 22:37)



So, what we can tell that this is the we this case we are discussing that if R S inputs are 0 0, then this is the previous that we can tell the previous state or we can tell that this is hold. That means whatever signal level was there in the output it continues that mean it holds that value it holds the signal value, so R S 0 0 it is hold state.

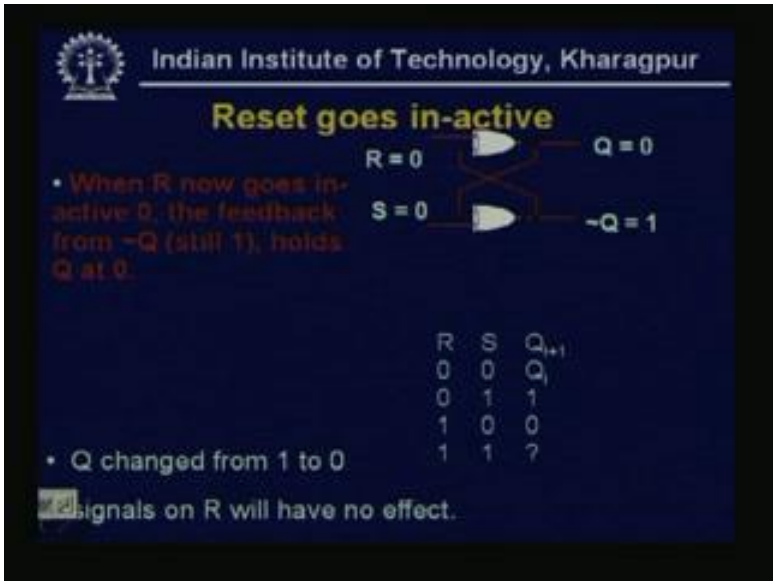
(Refer Slide Time: 23:35)



Now, if it is a 0 1 similarly, if we see that if it is a 0 1 or 0 0, if it is 1 0 means the reset goes active. So, what will happen if R equal to 1 and S equal to 0 then in the similar way, we are getting that Q equal to 0 and the complement of Q, Q 1 is 1. Since both inputs are

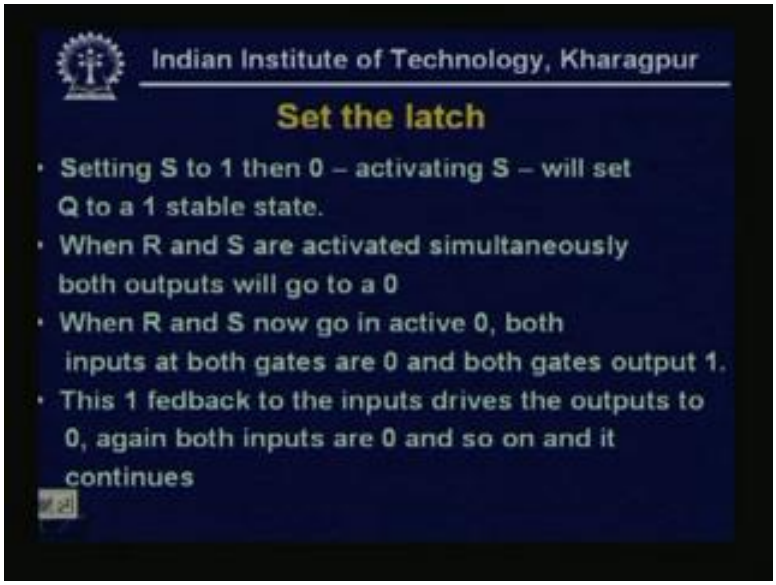
0 the output is forced to 1 the earlier situation all ready we have seen. Now, the output Q bar is fed back to the gate and both inputs being 1 to the output Q stays at 0. That means, if it is 1 then if it is 1 then R equal to 1 and the other input is also 1 that means, for N 1 what we can we tell N 1 the inputs are 1 1.So, the output Q is equal to 0, so that we are telling reset as the output is 0.

(Refer Slide Time: 25:21)



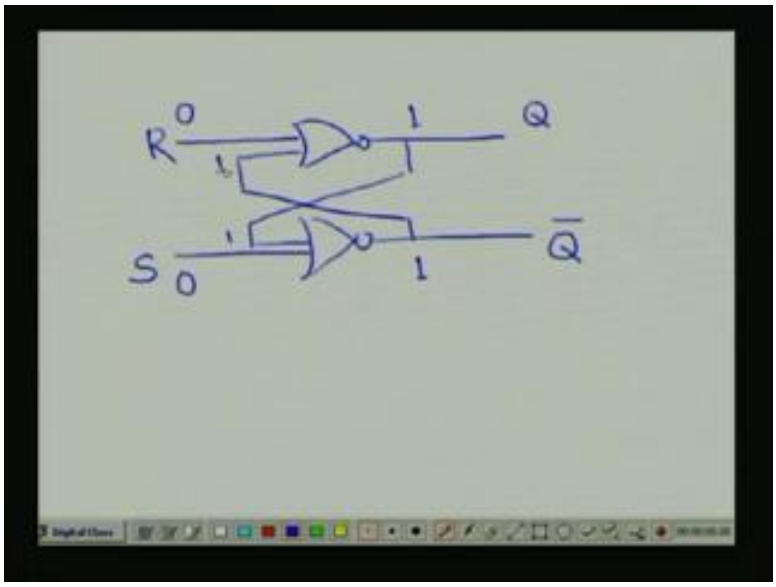
Now, again if we give R equal to 0 S equal to 0 then this reset goes inactive that means, Q changed from 1 to 0. And the signals on R will have no effect, now set the latch.

(Refer Slide Time: 25:49)



Now, setting S to 1 then 0 now, if we see that setting S to 1 that means, then 0 that means, 0 1 that activating S will set Q to a 1 stable state. When R and S are activated simultaneously both outputs will go to a 0, when R and S now, go is active 0 both inputs at both gates are 0 and both gate outputs are 1. Now, these I fed back to the input drives the outputs to 0 again, so both inputs are 0 and, so on and it will continue, so what will happen.

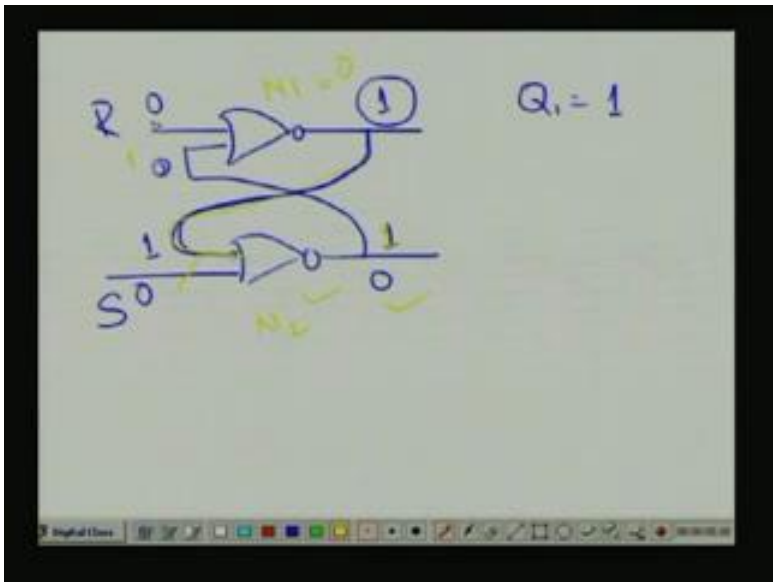
(Refer Slide Time: 27:00)



So, if we consider if we see that again if we draw the circuit say the NOR gate, now say both inputs are these are 1, now if R 0 and equal to S equal to now, then what will happen. That here also is it has 1 has come, so this will become 1 1, because NOR gate, so it will become 1 and this will become 0, so and it will continue that thing. So, when R and S now go inactive 0 both inputs, when R and S now go inactive 0 both inputs at both gates are 0 and both gates output are 1 just now we have seen.

Now, these one fed back to the inputs drives the outputs to 0 again, because once the output of the NOR gate one of the NOR becomes 1. Then it will force to the other NOR gate output be 0 provided the other input is 0 just now we have seen that thing. So, again both inputs are 0 and so on and it continues.

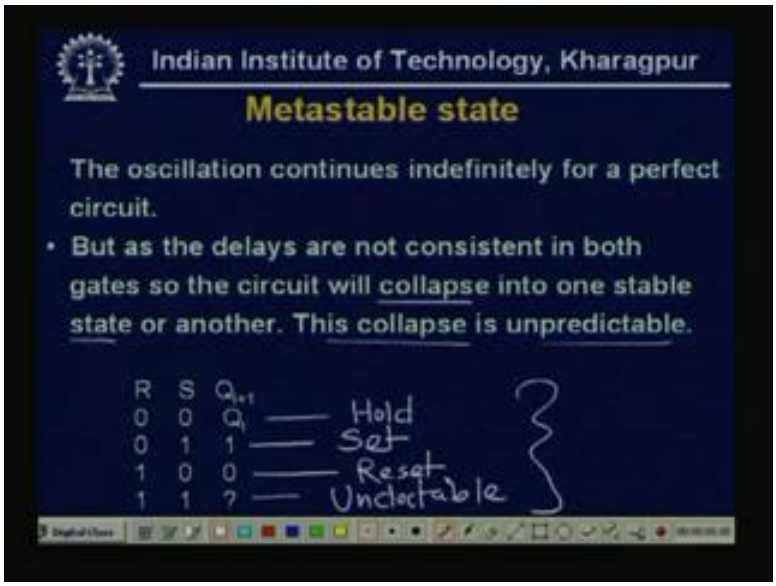
(Refer Slide Time: 29:21)



So, what we see that again if we consider that means, say again if we draw two NOR gate, see this is my R input and this is my S input now if it is 0. So, we are considering some previous state and previous state just now we have seen, if it is are R equal to 0 S equal to 0 then this becomes 1. So, previous state is 1 my Q i was 1. Now, when this becomes 1 this is 0 plus 1, this becomes 0 this is 1 and then this is again this will go 0, so this becomes 1.

Now, if the R input and if this becomes 1 then what will happen, see that these output forces actually, this is the reason because my the other input is 0. So, this is the reason that it forces the output of this N 2 NOR gate to be 0. Similarly, if this output is 1 then it will force the N 1 NOR gate output to be 0, if the R input is 0 and it will continue, ((Refer Time: 31:18)) so that is mentioned here. So, this is 1 fed back to the inputs drives the outputs to 0 again both inputs are 0 and so on and it continues.

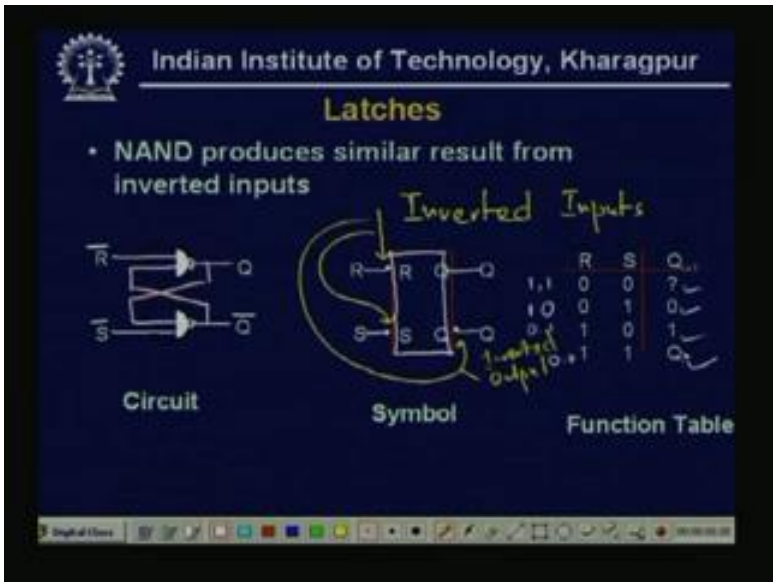
(Refer Slide Time: 31:35)



Now, these oscillations…So this will be a oscillation continues indefinitely for a perfect circuit. But as the delays are not consistent, because the even they are 2 NOR gates, but the delays are not sometimes there will be a change in the time some processing time of that gate that we have calling the delay. So, delays are not consistent in both the gates, so the circuit will collapse into one stable state or another and this collapse is unpredictable, because we do not know that which 1 will happen 0 or 1.

So, what we can summarize that this 0 0 this means this all ready we have seen this is a hold state. So, 0 1 this is 1 means this is my set and 1 0 all ready we have seen this is my reset and actually, 1 1 this is unpredictable or we call undetectable whatever, so undetectable, so mainly these are the four situations of the RS latch.

(Refer Slide Time: 33:02)

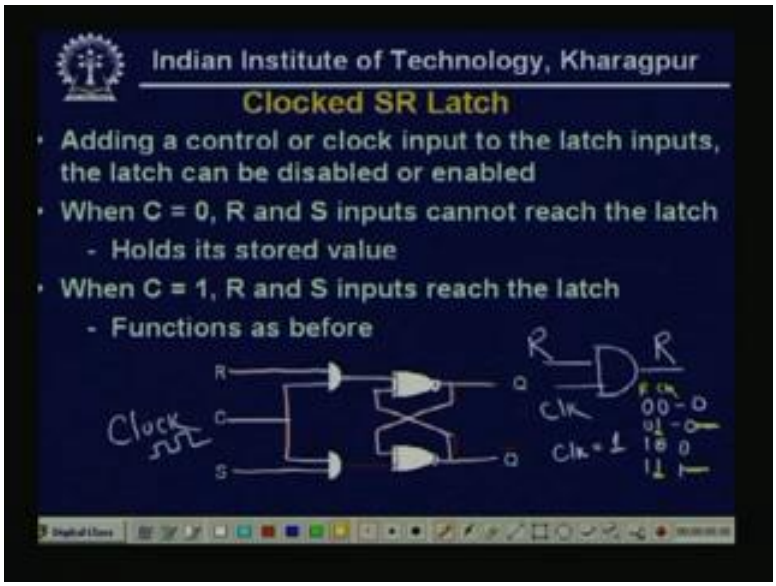


Now, the same operation we can get using NAND gate also, see now we are taking some the similar type of structure, the taking that again one NAND gate R input is there. Then this is with inverted inputs means, my R is actually R complement otherwise the same circuitry I am taking, this is NAND, S output again this is Q and Q bar. Always that inputs are inverted means R is R is complement S is S complement and that similar type of symbols. We can use only see here this bubble means here this bubble means it is inverted inputs.

Now, this symbol is same that R bar and S bar, again we are taking a rectangular shaped box and Q and Q bar are the again this bubble is the inverted means that this is also inverted, so this is actually inverted output. Now, the R S here the will be the same that this is my inputs, so it will be as it is inverted, so actually R bar S bar 0 0 R S 0 0 means this is actually, the situation of 1 1 of our NOR case and 1 1 means this is unpredictable.

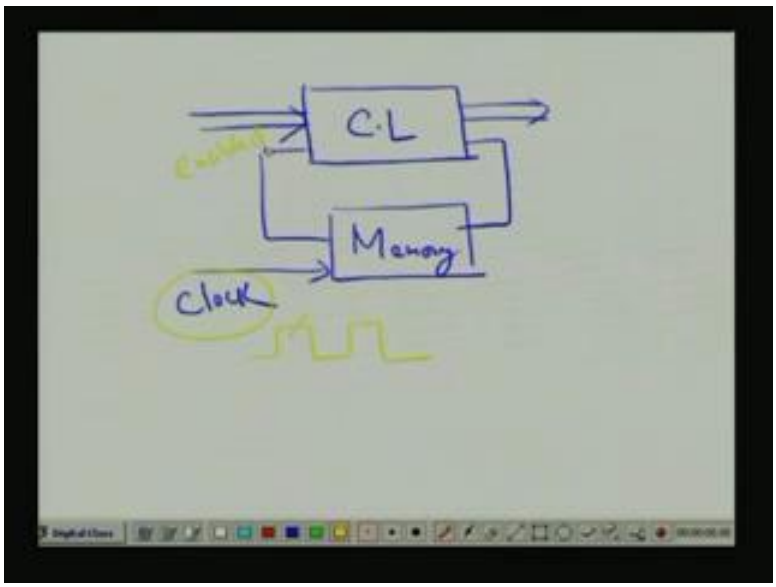
Similarly, it is 0 1 means actually 1 0 and that is my reset case it is 1 0 means 0 1, so actually this is my set case and 1 1 means this is my 0 0, so this is my the hold case. So, actually only we are getting that as if the inverted inputs, so the cross coupled NAND gate or the cross coupled NOR gate both forms the same type of latch circuitry.

(Refer Slide Time: 36:23)



Now, we take the clocked SR latch, because all ready we have seen the sequential module when the basic sequential module we have discussed.

(Refer Slide Time: 36:42)



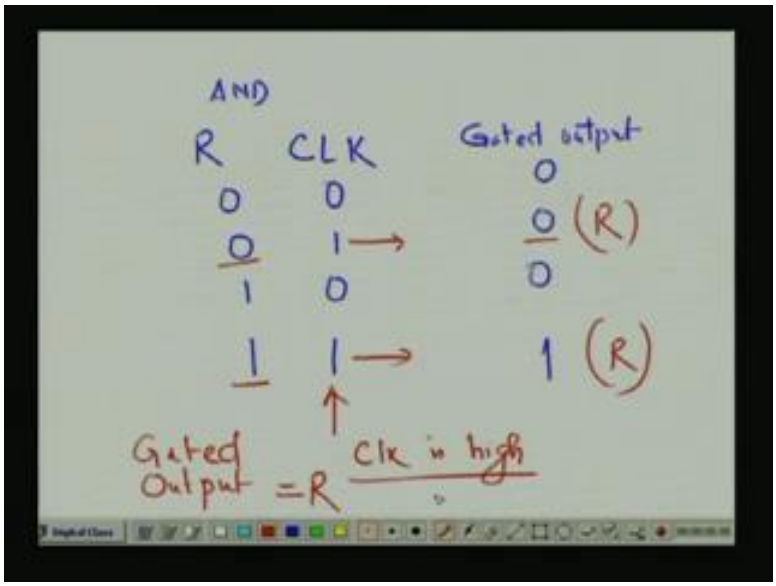
What we have seen, that actually this is a combinational circuits say combinational circuits C L, combination logic. Some inputs are there outputs are taken some memory elements, this is the memory elements whose design we are discussing know. And it is fed back, one clock is the another input of the memory elements. And when the clock

becomes, so this is one clock pulse when the clock becomes high then only the inputs are enabled, so these are these are enabled, so mainly we are discussing this one.

Now, adding a control or clock input to the latch inputs the latch can be disabled or enabled. So, now we are seeing the same NAND gate or say one AND gate as if this is a gated thing. First we see the clocked SR latch, so this is one R is fed to one AND gate and 1 clock is, so this is my clock input and that is the input of both the gates. The output of these gated input is fed to the NOR gate of the R S latch output is Q bar. And similarly, we have taken… See that means, here when the clock is on or the clock is high then only the R input is enabled or because as it is the AND gate. So, what we know that say it is R and this is my clock.

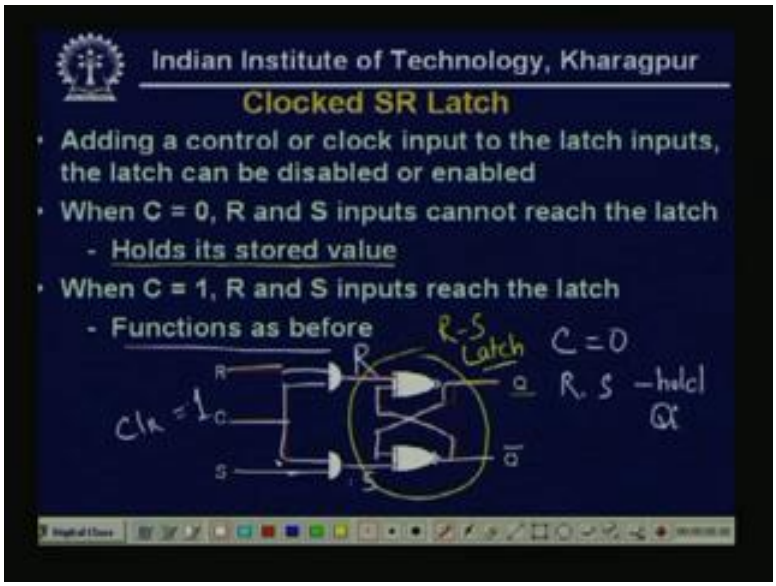
So, when clock is 1 then only I will get R here, because my AND truth table is 0 0 0 0 1 0 1 0 0 1 1 1. So, if we see these two situation as if this is the R input and this is my clock input. So, when the clock is 1 when these two clocks are 1 or this in this case that means, the if we if we draw.

(Refer Slide Time: 40:15)



Say as this is a nothing, but AND, so R and the clock these are the 2 inputs, so the gated output we can tell that is 0 0 0 only 1. Now, we consider only this situation and this situation when the clock is high, then this is 0 this is also 0 means R this is 1 this is 1 that is means R. So, actually when the clock is high then the clock is high, then output equal to R, output means my gated output, so that is why it is called a gated.

(Refer Slide Time: 41:28)



Now, similar thing will happen for the S inputs also that means, that here also that S will configure, so this will be the situation. Now, when C equal to 0, R and S inputs cannot reach the latch just now we have seen because it is a AND gate. So, if C equal to 0 then both inputs are these are 0, because AND gate if any 1 of the inputs is 0 the output will be 0, so if clock is my clock is 0 if to my clock is 0 then it will be 0.

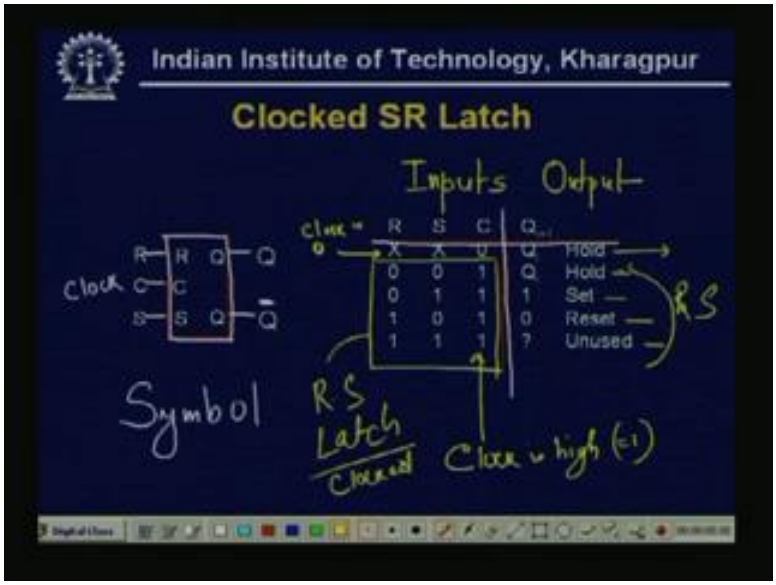
So, now I know if R equal to 0 S equal to 0 this is my hold case, means hold its previous value all ready we have seen. So, when it holds its stored value this is the hold its stored value so that means, when my clock is 0, now see we are trying to define the memory or function the function of the circuit as a memory with respect to the clock. That means, when my clock is 0 then whatever, value was there in the previous case that is being hold.

So, this is the clock equal to 0 means R input and S input both are 0 and all ready we have seen, for the RS latch that this is R equal to 0, S equal to 0 means the Q value is the actually the previous case the Q i we have denoted. So, this is my this is my Q i the previous case hold. Now, when C equal to 1 that means, our clock is now, my clock is high, so if we see my clock is 1.

So, what we have seen actually here not 0, R and S will come into come as input, because we all ready we have seen if clock is 1 then this is my whatever value we have given in R and S. Then if clock is 1 then whatever R value and S value means this

becomes actually, when clock is 1 this is behave as a whole circuit will behave as R S latch, so it is the functions as before.

(Refer Slide Time: 44:51)



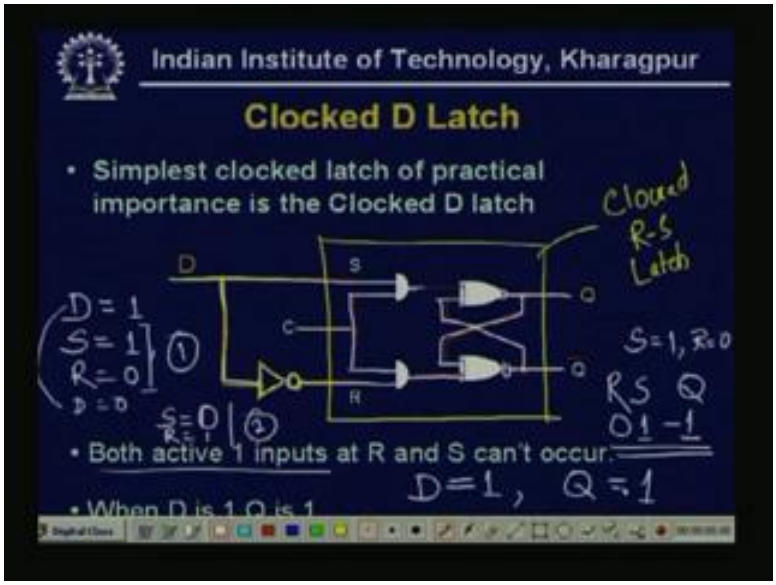
So, now if we represent the functional logic of the clocked SR latch, then it becomes first this is my symbol of the clocked. So, there are 3 inputs now, 1 is R input, 1 is S input and 1 is my clock C means the clock, then again that one rectangular box and outputs are as usual Q and Q bar. Now, if we draw the truth table then these are my inputs and 1 output we are showing other is the complement. So, if R is if clock is 0 see this is the situation when clock is 0.

So, whatever clock is 0, whatever value in the R S we give that will not reach in the R S output or will contain that R S latch input of the R S latch, so it will be it will be the hold case means the previous state. Now, if the clock is high, so for these four cases see the clock is clock is high. And if the clock is high or clock is 1 then whatever, R S value we give that will reach as the input to the R S latch. So, it will behave as the it will behave as a simple R S latch only clocked because only it will be enabled when clock is high.

So, it will be the same situation hold set reset and unused or undetectable or unpredictable or already we mentioned. So, this is my simple RS latch only this is a clocked RS latch better I write to that RS latch clocked. So, only we are getting one extra cases actually four extra cases, that when clock is 0 that whatever R S value we gives 0 0

0 1 1 0 1 1. Then it will be all the cases it will be the hold case means whatever, previous value was there that will be hold in the output.

(Refer Slide Time: 47:59)



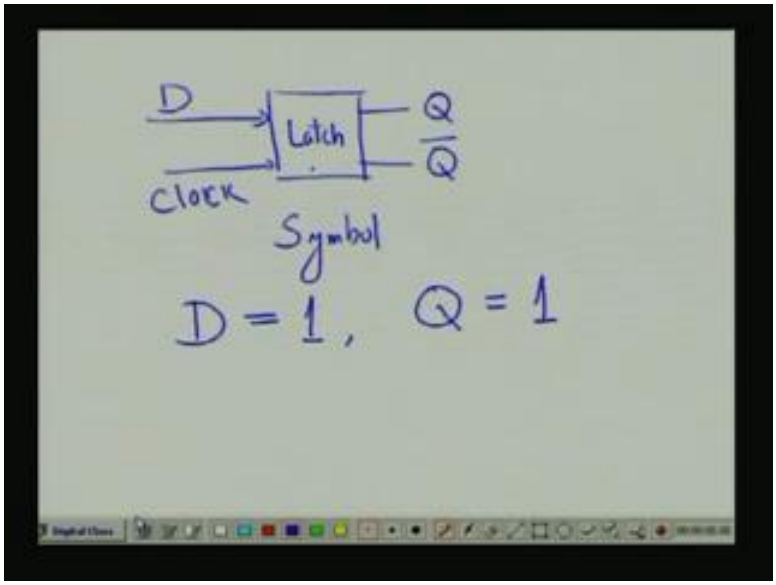
Now, clocked D latch see the simplest clocked latch of practical importance is the clocked D latch. So, it is again defined like first we see the circuit, so as if say this is my it is defined now, a D input it is 1 of the input of the AND gate say this is S NOR gate. Now, this D input is fed through a inverter, now what is the practical meaning of this thing see that as if this is my another input of the R S latch. So, both active 1 inputs and R and S cannot occur, that means see this is if we consider this circuit this is nothing, but the clocked R S latch, this is the clocked R S latch.

Now, the inputs are manipulated. Now, this S and R input these are symmetrical, so as if… 1 D input we are now telling this is a 1 input that is fed, and these D input being inverted is taken as the R. So, what will happen when D is 1 that means, my S is 1 and S is 1 as it is inverted. So, R is 0, so this is 1 case 1 case. When D is 0 when D is 0 then S is 1 and S is 0 and R is 1 this is my second case, see that means, never it will happen, because 1 input is inverted of the other input.

So, both will be active 1 that that type of situation will never happen, so this is R and S cannot occur. So, when D is 1 what will happen when D is 1 the situation is when D is 1. Actually, this is the S equal to 1 and R equal to 0 and we know that 0 1 that means, R S if we consider, RS latch clocked RS latch this is 0 1 means this is 1 that is my set case

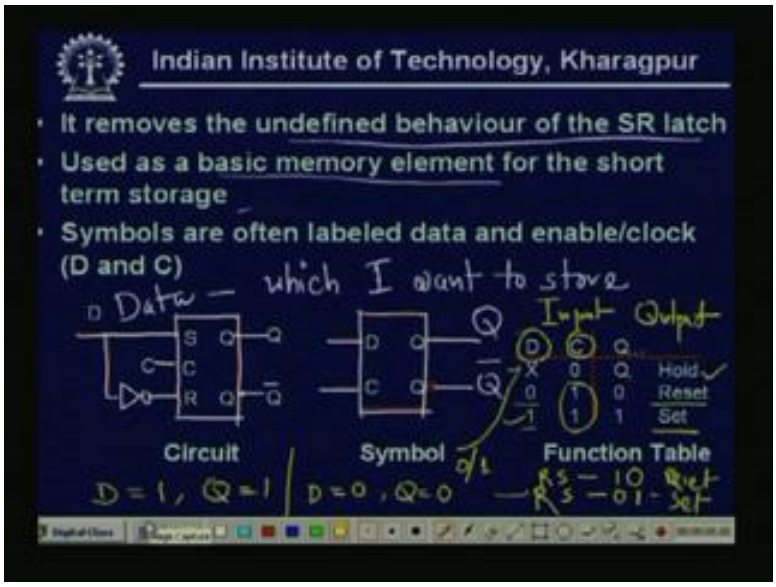
output R this is set case. So, D is 1 when D is 1 my output Q is 1, so now we will consider as if, so D latch as if I have only 1 input now if we draw that thing.

(Refer Slide Time: 52:07)



Say as if I have if we if this becomes the symbol as if I have the only D input clock is there obviously, because the it is clocked latch. And similarly Q and Q bar this is my flip flop D flip flop or better latch. Also we can tell, better we write I will write now, I write clocked latch, so thing is when D equal to 1 this is the Q equal to 1.

(Refer Slide Time: 53:09)



So, it removes a undefined the behavior of the S R latch, because S R latch when S equal to 1 R equal to 1 that was the only confusing situation that S equal to 1, R equal to 1 the output cannot be defined or we have mentioned that is unpredictable or undetectable. Now, we have removed that cases by inverting 1 input of the others that means if 1 input is 1 the other input never be 1 it becomes 0. So, it removes the undefined behavior and that is the beauty of the D latch, so this undefined behavior of the S R latch and this is used as a basic memory element for the short term storage.

So, symbols are often leveled data and unable clock D and C, so now as it is the function of only 1 input. So as if we are we are considering that my D means this is my D input, so this D means, my data input that data which I want to store in the memory. And this is being inverted this is my clock again that same symbol I can tell Q and Q bar and, so this is D and clock all ready I have shown this symbols and this is my Q Q bar.

So, the function table will be that if these are my inputs will be only one data input D and this is my clock and this is the output. So, if clock is 0 all ready we have seen if clock is 0 the input cannot be reached. So, whatever D value is there 0 or 1, whatever D value is there 0 or 1 that it should to be a hold case, because my R S inputs will be 0 0, because if clock is 0 all ready we have seen that it is 0 0, so it is a hold case.

Now, if clock is 1 then whatever D value is there that will be my input of the S input. So, D is 0 means S is 0, S is 0 means R is 1, so R S is 1 0 and that is my reset, so that is why when D equal to 0 this is a reset. If D equal to 1 D equal to 1 means my S equal to 1 and S equal 1 means my R equal to 0, so 0 1 means R S input 0 1 means this is a set case. So, this is set that means, when D equal to 1 this is a set.

So, what we can tell that D equal to 1 means Q equal to 1, D equal to 0, Q equal to 0. So, see the functions… also function becomes very simple that means, if D equal to 1 output Q equal to 1, if D equal to 0 Q equal to 0. So, this is treated as the basic memory element. And that is why this is the most important sequential element, that it taken as the basic building block for the higher memory design or that high dimension memory design.

So, next day we will be discussing the other flip flops and latches and the differences between the latch and flip flop and then how it is being used for the high dimension memories etcetera. So, mainly today's class we summarize that we have what to do we

mean by the latch or the flip flop and how it is being used to store one bit that we have told. So, will end the class here.

Thank you.

Digital Systems Design

Prof. D. Roychoudhury

Dept. Of Computer Science & Engineering

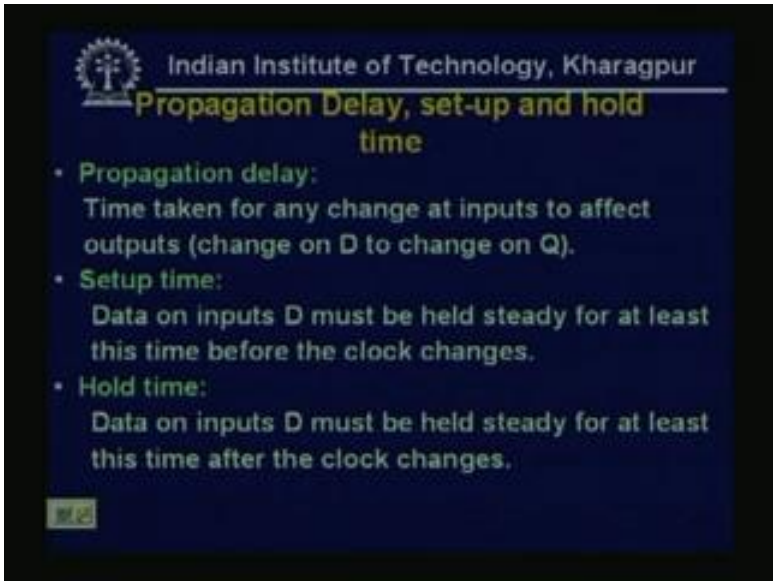
Indian Institute of Technology, Kharagpur

Lecture - 22

Design of Sequential Modules

We have learned how to Design a Sequential Modules or actually what do we mean by sequential circuits. And how we can store one bit memory or how the sequential modules are used as the memory, Then we have seen the construction or the design of the S R latch the clocked latch the D flip flop etcetera.

(Refer Slide Time: 58:38)



Now, today we will continue the discussion on the sequential modules, the other different type of sequential modules that are being used in real life circuits. But, before that we will see, that what do you mean by the propagation delay setup time hold time and actually how this times are effecting the actual value.