INDIAN INSTITUTE OF TECHNOLOGY GUWAHATI

NPTEL
NPTEL INLINE CERTIFICATION COURSE
An Initiative of MHRD

VLSI Design, Verification &Test

Dr. Santosh Biswas
Department of CSE
IIT Guwahati

Module VIII: Fault Simulation and Testability Measures

Lecture IV: Testability Measures (SCOAP)

Design Verification and Test of Digital VLSI Circuits
NPTEL Video Course

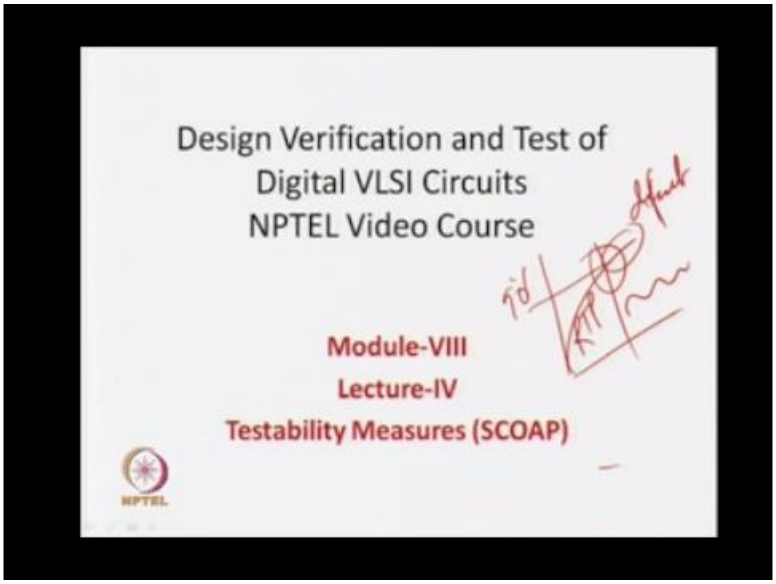
Module-VIII
Lecture-IV
Testability Measures (SCOAP)

So welcome to the lectures 4 of module 8, so in the last three lectures on fault simulation, so what we have basically seeing that, that when you do go for automatic test pattern generation that is for a given fault, you want to find out which pattern detected. So we have found that there are two schemes basically. So one is based on random test pattern generation and one is based on sensitize, propagate and justify approach.

So what we have done in random pattern generation based approach. So in random test pattern generation based approach what we have done. So we have taken basically a fault, basically a random pattern and then we have found out that which are the faults can be detected by the random pattern. So that we have seen four algorithms for that serial, parallel, detective and concurrent. And then the idea was that that we can easily go up to say around 90% of the faults which are easy to test.

So for the 90% of the faults you will find out that for a new random pattern the number of new faults that is detectable will be around say 90% of the faults can be a detected by the random pattern generation okay.

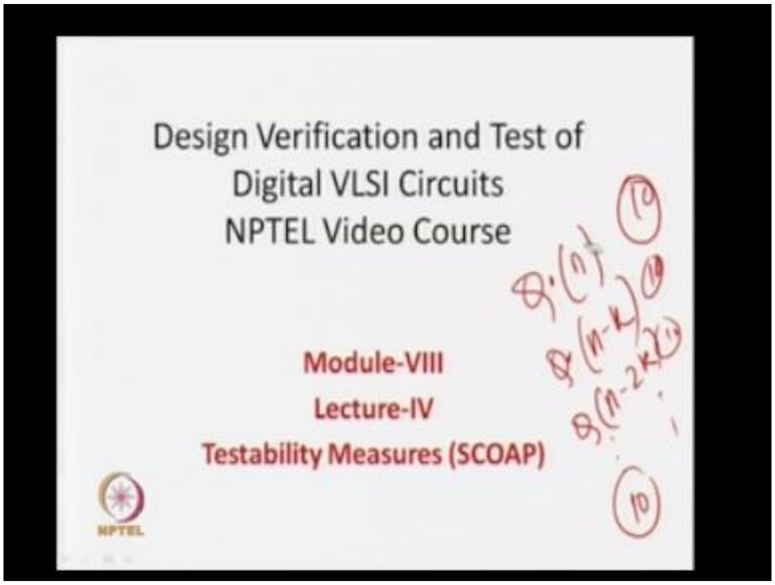
(Refer Slide Time: 01:27)



But after the 90% there are some faults which are actually called difficult to test faults so they are difficult faults. And for that you random patterns cannot detected. That is when if there a random pattern of 90% of your fault coverage when you find out that it does not detect a new pattern or it may detect a very less number of new patterns, and then when the next random pattern we applied may not detect any other fault and this can go on.

So at that time we stop the random test pattern generation and go for sensitize, propagate and justify approach okay it is this part of the curve. That is now we have seen that for that whatever the complexity of the random test pattern generation let us call it say complexity of secure okay.

(Refer Slide Time: 02:05)

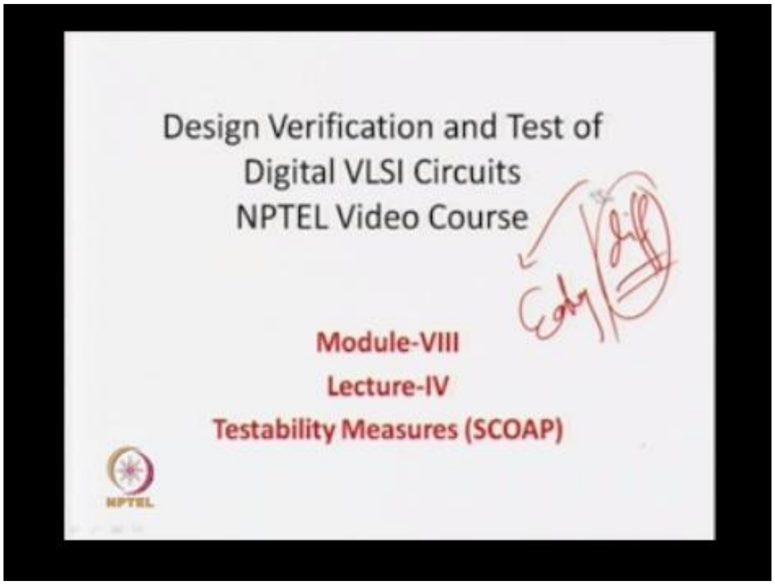


Then you have to multiply in the first iteration you have to consider all of these forms okay. So let us say that we have some N number of faults. Now again the complexity is Q and say it has detected K number faults at the first iteration, then you can say that next complexity is N-K. In the following iteration to the N-2K, so in next iteration also there 2K faults are there put … and say in the end 10 faults remain which are difficult to test and cannot be taken by the random parts.

Now you see this 10 faults we have considered here, these 10 faults we have considered here, this 10 faults we have considered.. So for all the iteration the difficult to test faults we have considered for random test pattern generation and the problem that happened was that could not be detected. So we first go and say around there were N patterns so N faults, so there were 100 faults N=100.

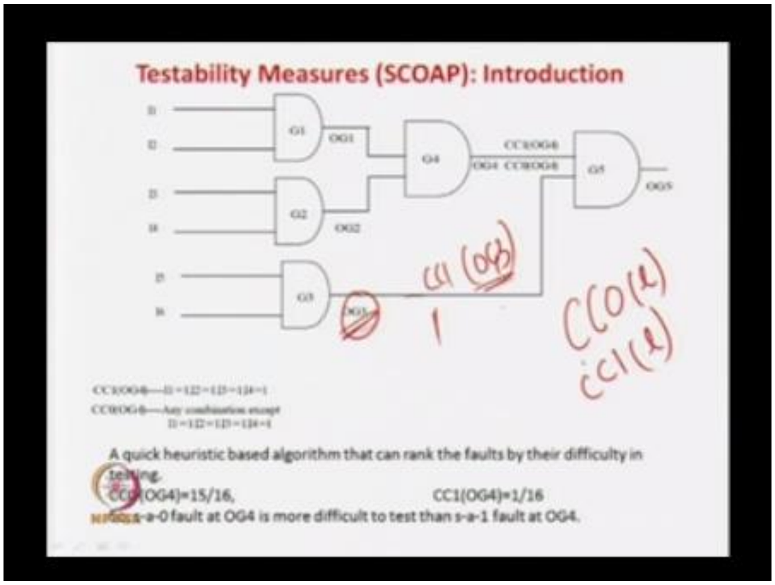
Then when the first pattern you apply and you verify for 100 faults we say 20 patterns, 20 faults get say detected, then next iteration you go for AT and but we measure that is after every iteration some of the faults are dropped, so they are easy to test faults and they are adding advantages to random test pattern generation with algorithm. But you see some of the faults remain the difficult to test faults remain. And these faults keep on linear in all the iterations of your random test pattern generation.

(Refer Slide Time: 03:22)



So we have thought that if you can find out some algorithm or some juristic which will say that these are easy to test fault and they are difficult to test fault. So what we will do that these difficult to test faults will not at all bring it into the paradigm for random test pattern generation. We will try with only that are easy to test faults and difficult to test faults we can keep it as E pride B and go directly for sensitive, propagate and justify approach.

(Refer Slide Time: 03:44)

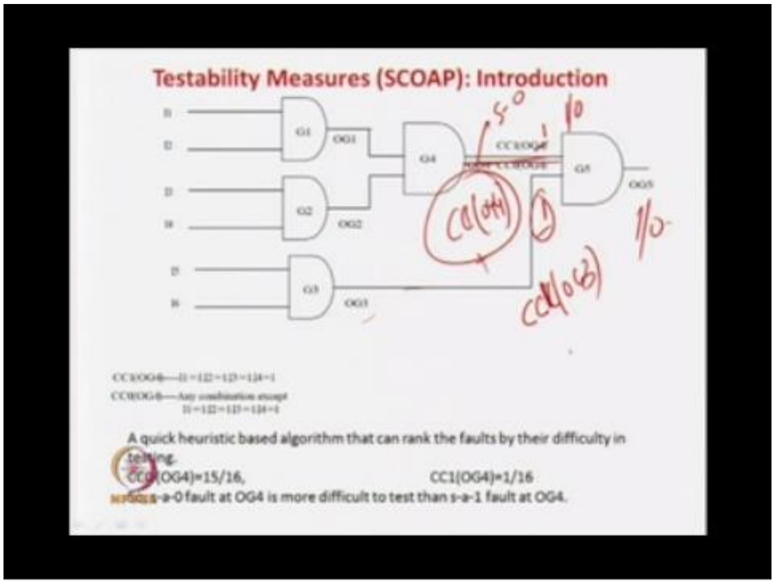


So what this will help, this will actually help in optimizing the random test pattern generation algorithm, because you will be not trying in every iteration the faults which linked are difficult to test fault. Now let us see how can we find out some kind of algorithm or a juristic that can tell you which is the easy fault and which is the difficult fault. So let us start with an example, so this is a simple circuit.

So let us consider that okay, let me just tell you some formal notations we will be using. So when we say that CC0 that is actually called combinational controllability of 0 okay. And if you say CC1 then we say there is combinational controllability of 1, and we also put a line L that is the net L. So if you say that CC0(OG3) so what does it mean, it will mean that how difficult is to get this line to get the value 0 combination controllability of 0 for this output the name if the this output is this one.

So if you say that CC1(OG3) that means how difficult is to make this line to 1 and this line. So there is two notations we are generally going to use and then we will add one more that we will tell you later.

(Refer Slide Time: 04:50)

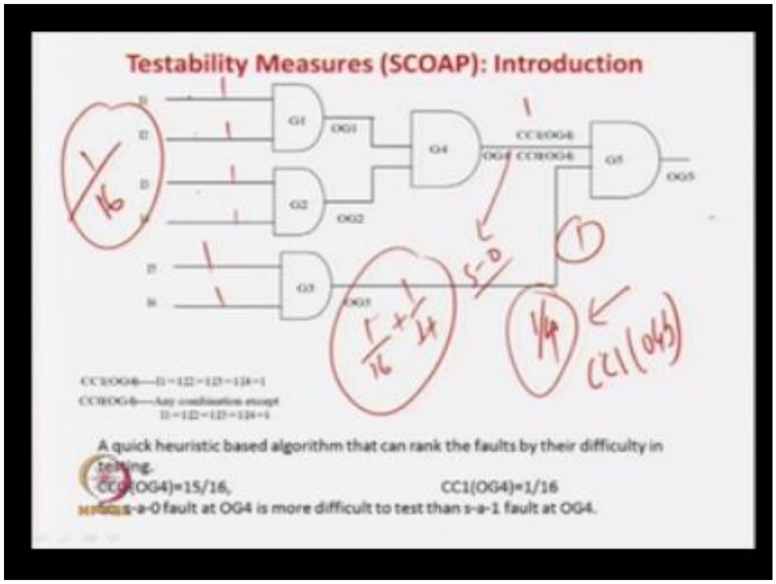


So let us consider this circuit now at this next we are considering a stack at 0 fault and a stack at 1 fault. Now we just try to find which is the easy fault to test and which is the difficult fault to test. So let us first consider that we are going for this stack at 0 fault. So if I allow your stack at 0 fault then you have a apply a 1 over here right. So that is what is there and then also we have to apply a 1 over here.

Then you can observe the value here correct, so this is your, so if this line is stack at 0, so you will get the answer 10, this is 1 and the answer is 10 and you can detect it. So instead we have to apply a 1 here and you have to apply a 1 here. Now so, you can say that difficulty of getting this stack at 1, stack at 0 fault here is combinational controllability of 1 you have to get at OG3 this you have to tell that you have to make this line 1 and this line also you have to make it to 1.

So it is combinational controllability of OG4 to this gate and you have to control it to 1. So if you add this to 1 that is you have to control this to 1 that is combinational controllability of OG3 plus combinational controllability of OG4 this one. If you add it then actually you can get the value of what is the difficulty of testing this one.

(Refer Slide Time: 06:06)

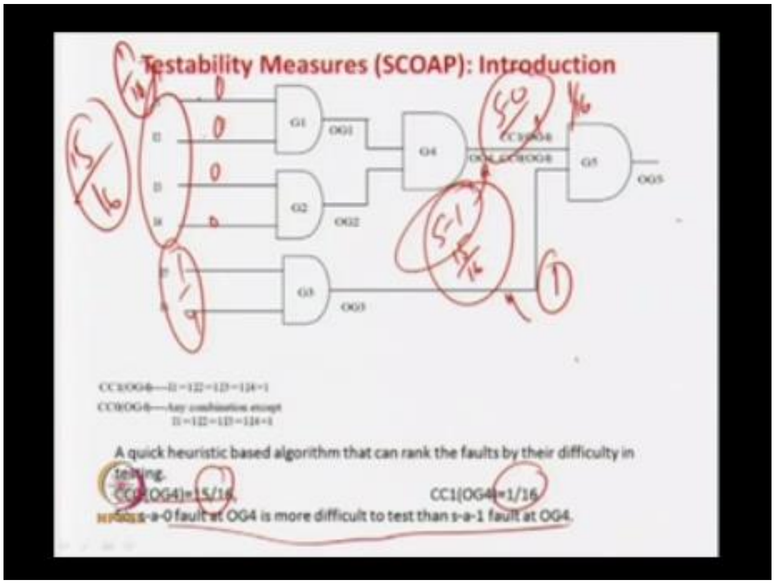


So let us see, I will just try to give some estimate okay, so in this case you have to get a 1, so what are the possible inputs here 00, 01, 10, 11 okay. So how difficult is to make it to 1 it is you can add a number that the difficulty in controlling this one is same ¼, because only one, because among these four patterns you can apply on the one of them will succeed in making this one as a 1.

So you can say that because 11 is required to test it make it a 1, so you say that 11 and so you can say this combinational controllability of 1 to OG3 is actually one fourth, because only one of the four combinations can do this one. Now in this case you have to apply a 1 over here, so you see 1234, 4 lines are there. So 16 possible combinations are there and when you can get a 1 you can get 1234.

So only 1/16 of a pattern can get you a 1 over here okay. So you can say that it is 1/16 plus the order of difficulty is ¼ something like this correct. So now you can say that for a stack at 0 fault here, so for a stack at 0 fault here remember that these are combinational complexity 1/16+1/4.

(Refer Slide Time: 07:26)



Now low let us try to do the same thing for the stack at 1 fault kind of a thing in the same net okay. So in the same net we are going to do that okay. So same net we are now going to take a stack at 1 fault, so stack at 1 fault means you have to apply a 0 over here obviously this is controllability so this is you have to apply a 1 at this net okay and this is the stack at 1 so if the answer is 1/0 and the 1/0.

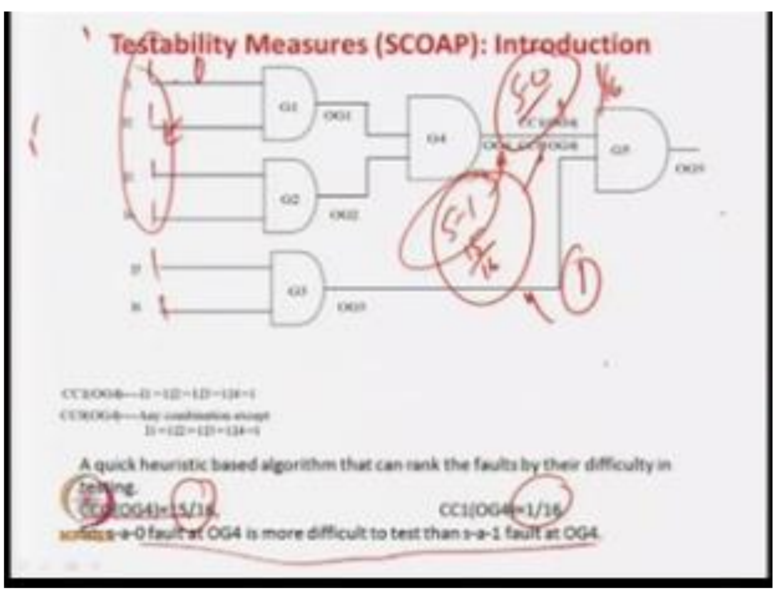
So now how do difficult is to state a stack at 1 fault here, it is ¼ because this line has to be made the 1 which is similar plus you have to make this net 0. So again this four inputs are there, so there is 16 combination and 15 out of them that is 0000 to 0111 actually accepting 111, 111 will give you the 1 people those are accepting this combination all other 15 combinations can give a 0 over here.

So you can say that the difficulty of controlling or how you can get the value is 15/16 and ¼. So this ¼ is common to both these stack at 0 fault error, stack at 1 fault. So let us illuminate this, so you can say that combinational controllability of OG4 to 0 is 15 by this one and combinational controllability basically here is 1/60. So now you can see what happens is that to test a stack at, because this ¼ this one get value 1 this net is same for all, so let us forget about it.

Now a stack at 0 fault N is the application of 1. So whose vulnerability is 1.16, now we apply a, there is 1/16 now to state the stack at 1 fault over here the probability is 15/16 that is 15 combinations out of 16 which is. So obviously testing a stack at 1 fault here is a easier than testing a stack at 0 fault here. So if I am going to apply a random test pattern generation then I could have said that you take this stack at 1 for random test pattern generation is easier because 15 out of 16 combinations can help you that is and one-fourth that is one fourth so this is common for all.

So out of 15, 16, 15 combinations can help you to detect this stack at 1 and only one out of 16 combination can help you detect the stack at 0 fault these are the difficult problem. So better for this stack at 0 fault you go for a, what do you call sensitize, propagate and justify approach and for the other you go for this stack at 0 fault you go for I mean sorry, for this stack at 1 fault you go for a random test pattern generation. So this stack at 0 fault at O4 is more difficult than a stack at 1 this one.

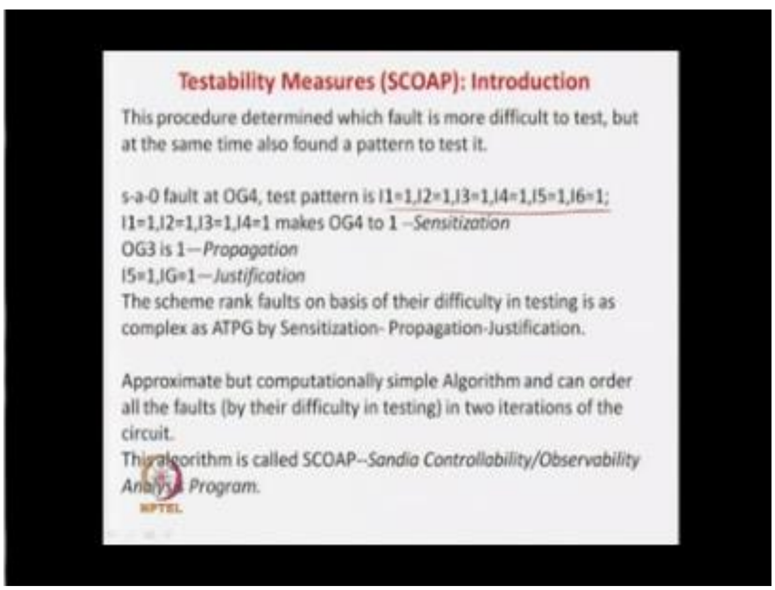
(Refer Slide Time: 09:54)



So by this algorithm kind of a thing you can easily find out that which is easier to test fault and which is difficult to test fault. But these are basic notion which is actually I will tell you what do you mean by SCOAP and all, what are the basic notion there, how can we determine that which is a, or you can say that how to determine which is the easy to test fault and which is a difficult to test fault so here we have found out a procedure but you all have to observe that the same time.

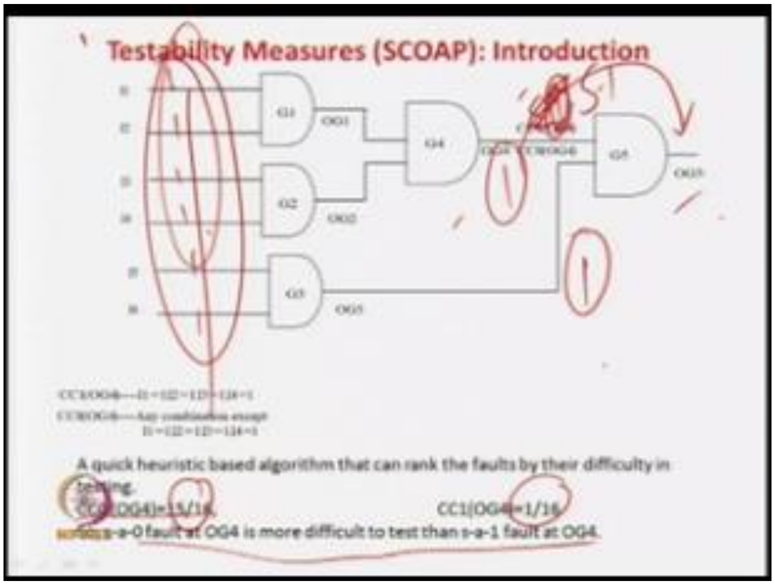
We have also generated the test pattern that is what we have done we have saved that you have this is you have to apply a one over here so one and one is a base pattern and here to test a stacked 0 fault you have to apply 1111 only one combination and here we have to any and this for this stacked one fault here any other combination other this 1 so indirectly what we have done.

(Refer Slide Time: 10:36)



Indirectly we have generated test patterns for finding out this which difficult to testy fault and which as the test fault so this is not a very good approach so what we have send that for this stacked 0 fault the test pattern is this say 1 part this is all once because we have said that.

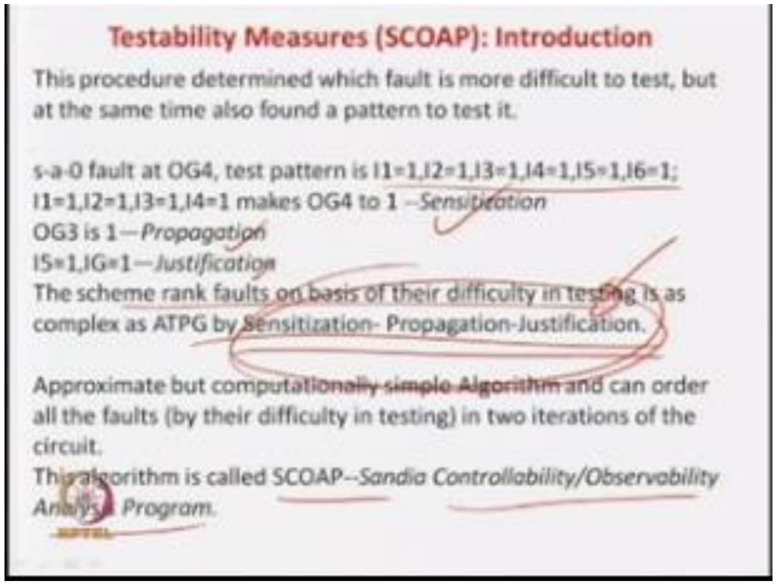
(Refer Slide Time: 10:50)



To test the stacked 1 fault let us take this case that why we are saying that to find out which is the easier or which are difficult to test for we are applying all the patterns say for the stacked one fault we have to apply a sorry for this stacked 0 fault we have to apply a 1 over here okay so I the answer is 1111 1 oaky and for this one also is true inter prorogated so every 11 so we have generated the test that when the sensory is portaged just if I approach indirectly so stacked 0 we apply a 1 so this sentience.

Now this analogy propagated to the output so you have to apply a 1 over and then we are saying that this is the patter that we apply and for the stacked 10 one forth so what we have done we have to say that any other pattern other than this thing with pull so basically what we have done.

(Refer Slide Time: 11:31)



So we have gone for sensate propagate and justify so we are ranking of their falls based on difficulty of testing but we are also generating the testing pattern so that is not a very good thing we are doing because of the complex so here we are actually say that we are trying to find out which is the easier fault and which is the difficult set of faults for that we are actually going for sensate prorogate and justify approaches is a very complex algorithm so to solve a simpler problem we are taking a bigger algorithm and trying to solve.

So that is actually wrung but we are put main emphasize that we want to find out a very simple algorithm may be this are lunatic may be it will give approximate results but our main goal was to find out that which are the easier set of faults and which are the difficult set of faults by doing this approach which we have disused in the last slide what we are doing we are doing a kind of a thing that we are using sensory propagate and justify approach and by that we are finding out which are easier fault and difficult to test.

So these have any meaning because we are generating the test pattern we have two algorithm what is random test pattern which is easier and which is no complexity and we have test sensor prorogate and justify approach which involves more complexity so what we are doing is we are using a more complex algorithm to find out which is a easier fault and which is a difficult to test that was many because now Idea was that we have to find out a very simple low complexity algorithm which can tell you which is the easy fault.

And which is the difficult fault so difficult faults will through to senor and propagate and justify approach and easier faults we will go by what you call random but now this algorithm is the last slide it actually finds out which is the easier to test difficult and difficult to test part but further that you are using an algorithm which is a very complex one that is sensory propagate and justify approach and then you are find out so here difficult.

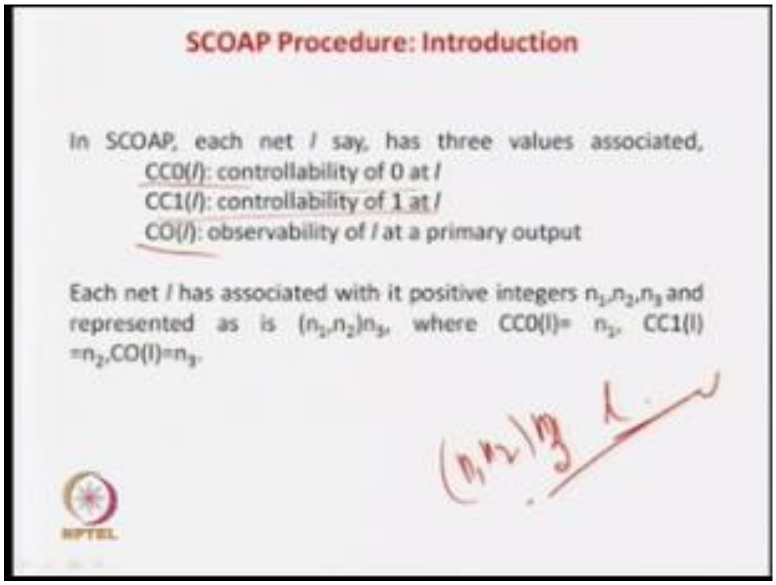
So I may say that why you want to classify is difficult and easy you apply sensory and propagate and justify approach and then have you think you get difficult faults as well as you are going to get the test pattern so that high complicity algorithm of sensory propagate and justify approach you where using the fine of which is the easier fault which is the difficult does not solve our parts our purpose is that we have to use sensory propagate and justify algorithm as minimum as possible and only for the difficult to respond.

So we cannot use our complexity algorithm to find out which is the easier fault and which I the difficult fault we have it find out such a easy to set faults and difficult to test faults using a very simple algorithm and may be but it may give some approximate result that is possible then for the earlier fault we are gouging to use a simple algorithm that is random test that I say and for the difficult to test fists we can use the complex algorithm for sensory propagate and justify so this is the aim.

So our so our idea of you have to find out a simpler algorithm to find out which a easier fault and difficult fault so we are going to study SCOAP which I the algorithm this can be control, ability observe analysis program so it is a very simple algorithm as will see it does not required to go and justify what all it will do it will sane the circuit and it will approximately take with a easier to test fault.

And which is the difficult to test part but it may be approximate and give you approximate result but it can give a clear idea that which are eraser and try to random and which are difficult then go for sensory propagate and justify so now we will study this algorithm in details.

(Refer Slide Time: 14:33)



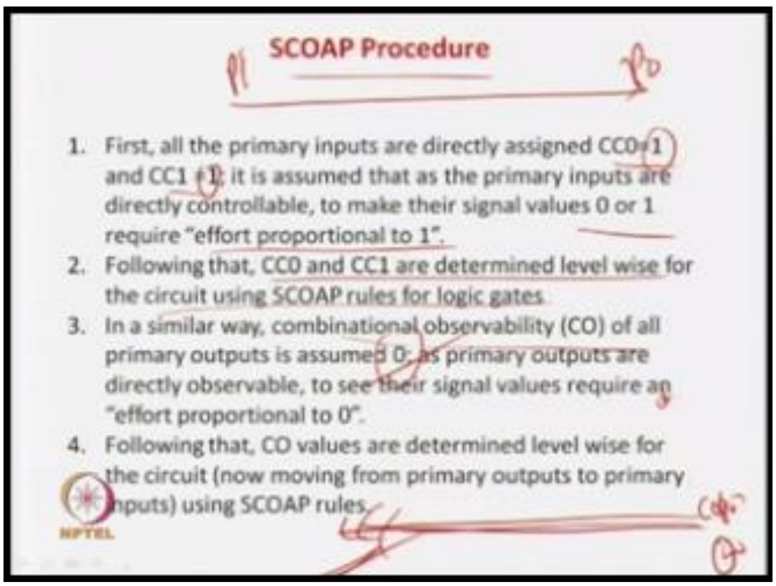
So before that we see some notations so CC0 as well already we told you it is control ability of net l how difficult this to control the net l to 0 CC1 how difficult this to control the net at one and this one more combination conserve ability of tell that is say for example this is a net l and this is lot of other circuits and this is the output primary output then CC of O that is CO of Ol that is the combination observability of l that is how difficulties to observe this the energy primary output.

So to test a circuits so if it is a stacked at o faults then you have to apply a 1 over here so we have to find that combinational controllability of 1 at l okay because if you apply 1 then stacked 0 fault can be test as well as this l value as to be observability the primary output that is CO of l so you have to add CC1 Ll + CO of l to get a stacked 0 for now if you add rest for stacked fault what you have to do so if you apply 1 to stacked 1 fault.

Then we have to apply a combinational of 0 at l use stacked 1 fault here you have to apply a 0 now this address as to be propagated to the output that you have to observe the evaluate the output so it is combinational observability of l at primary output you add them then you get the difficulty of how difficult you should take test a stacked one fault at the net l.

Okay so now sometimes you use a short notation like say n1 n2 n3 so n3 for net l so we say that nl stands for CC0 of l n2 stands for CC1 of l and CO stands for combinational observability and main this is some short notation with this sometimes use.

(Refer Slide Time: 16:06)



Now we discuss the SCOAP producer so what is the SCOAP producer this SCOAP producer is first all primary inputs are assigned 0 CC0 as CC1 so what is the idea so all the primary inputs we assume that it can be easily control level so the difficult of a control ability of CC0 that is making a primary input 0 primary input are 1 that is the assumption that is the assume that the primary signal is directly control level and then observability effect is proposition to 1 that is if you have an AND gate say these are all the primary inputs then CC0 is of this input is 1 and also for this one is 1 and control ability of 1 is also 1.

That is primary inputs can be directly control level and difficulty level is 1 then what we do CC1 and CC0 are determine level vise using some logic roots so in the last lecture we have see that for the deceive fault simulation there are some rules that is some level to another level in the circuit if you go you have some rule so now here will a study some of this what you call this combinational circuit rules will also study that is SCOAP rules.

That if you go for one level of this circuit to another level the circuit how these rule are Appling that is if you know that CC0 of this net and CC0 of this net and what is the value if CC0 if you now CC1 And CC1 of this and what is the value of CC1 of this thing what is that is that level propagation hoe difficult is to control and how difficult is to observe so all those things will study that is rotes so now primary inputs we apply CC of 0 and CC of 1 as 11 that is digital to control the primary inputs are 111 that is 5 then you go for the next level of the gates.

And then you apply some rules that we will study in the similar way so you have to go to the similar rule if you apply then we can go for the primary inputs and primary outputs have the net will have the value of CC0 and CC1 similarly we have to go for combinational observer combination control ability goes from primary inputs to primary outputs this is the level propagation but for the combination observability see we go for the reverse way is we assume that the primary output lines are very easy to observe directly if you prove and then you can observe so difficult level is 0 so primary input control ability difficult is 1 and primary output observability is 0.

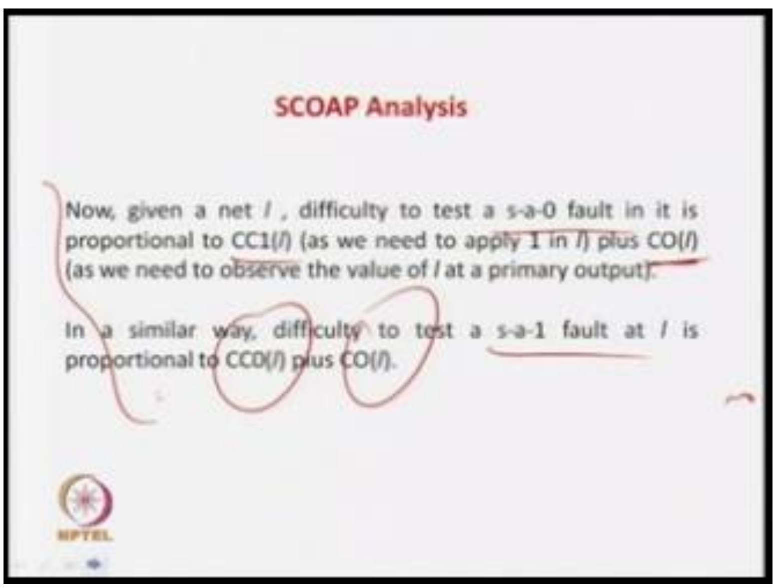
So you may ask the question that primary outputs are directly available as the output so why you make it as 1 and similar the control ability similarly the primary sorry what my question is that the primary input control ability is 11 that is the first statement but the observability of the primary output is 0 so you may ask the question that why it is like that that primary input is directly control levels you are giving a 1 and the primary output directly observe put you are not giving a value of 1 you're saying that S0 why is it possible.

So now here the assumption is the heretic is what you exact algorithm so the assumption it is there that to control alone you have to but some effect at least so they are saying that the primary input control l ability are 1 okay but the observation is the more simpler just you prohibit and you get the value so this is that observe difficulty is 1 less that is that is just the heuristic you can also try out with control ability level of 11 at the primary inputs and observability at the primary outputs is also 1 you can try out and find out that you may get another version of SCOAP but analysis is not different.

That is whatever SCOAP with primary input control level of 00 and output observability is familiar is 0 the value of I mean the cluster of easy to test for the digital for the gate will be very similar if we apply the logic that control ability of primary inputs is 1 and also observability primary output is 1 so this is not much difference in the results we use that and so where ever we go by the what does the propose by the literature that is difficult to control the inputs is 1 and on set is more easier so primary output observability is 0 following.

that we know that the primary output control ability value of all the primary outputs are 0 so now it is 0 basically now you have to come out from this level to this level and you have to apply some rules and then you can find out the controllability, observer ability all the dates but we have travel the circuit in the reverse direction.

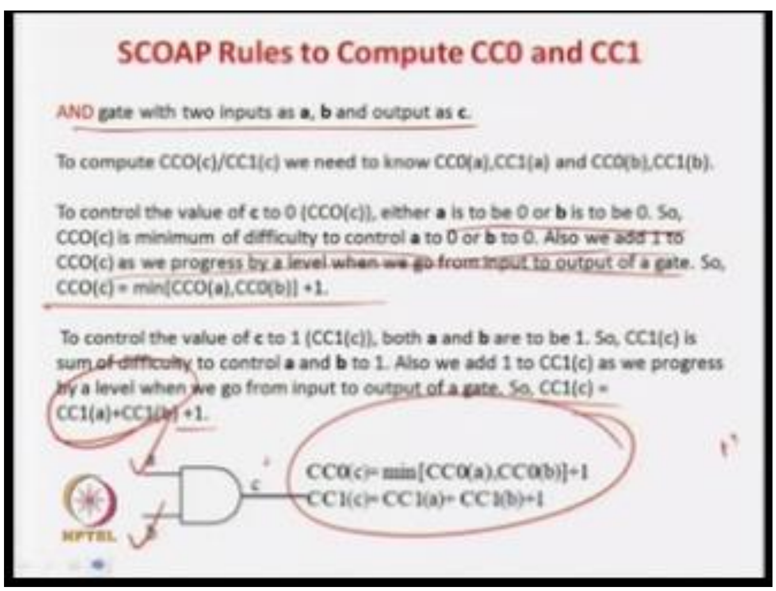
(Refer Slide Time: 20:01)



So once you have travel the circuit in two directions this direction which start with combinational control ability of 0 of primary net is equal to 1, and then we apply rules, rules, rules and at every net you get CC0 and CC1 so here also CC1 of the primary inputs are all 1s and finally for the primary output you get all the values of CC0 and CC1 you go by this way, and when you are going for the observably part of view the primary output observabilities are we say that primary output observer the combinational observability of the primary output equal to 0 and there are some rules you keep on applying then and then you get the there is a primary observability of all the nets in the circuit so primary output could be primary inputs.

So now as already discussed so even a net difficult to test a stack at 0 fault is you have to control the net to 1 and you have to observe the value of the outputs, so it is CC1(l) plus CO(l). Similarly, we stack at 1 fault means you have to apply a 0 so difficult to make that net to 0 and you have to observe the net at the output so this is how we can find out the difficulty to test the stack at 1 fault as stack at 0 fault.

(Refer Slide Time: 21:06)



Now we will study about what you call these rules because we said that we primary inputs control ability we know that they are equal to 1 observability primary output is 0 but now how to propagate so for that we require some rules, so let us see of an AND gate with two input output as this one and this one. So now say for example, these are rules we see rules later so now see how difficult is to control this C value to 0, okay so C value can be 0 when, when this is 0 or this is 0.

If either of then it is 0 so we know that this value is 0, so how difficult this to make this C to 0 so it is minimum of combinational control ability of 0(a) or combinational control ability of b to 0, plus you have to add a 1 that is because of the label shape from this label to this label so difficulty is infusing we assume and we add a 1, but to make a C0 how difficult it is, it is if it is to make C0 we have to apply either a0 or b0, so if a to 0 is easier then we say that we will apply a0 and we forget about b and the answer is 0.

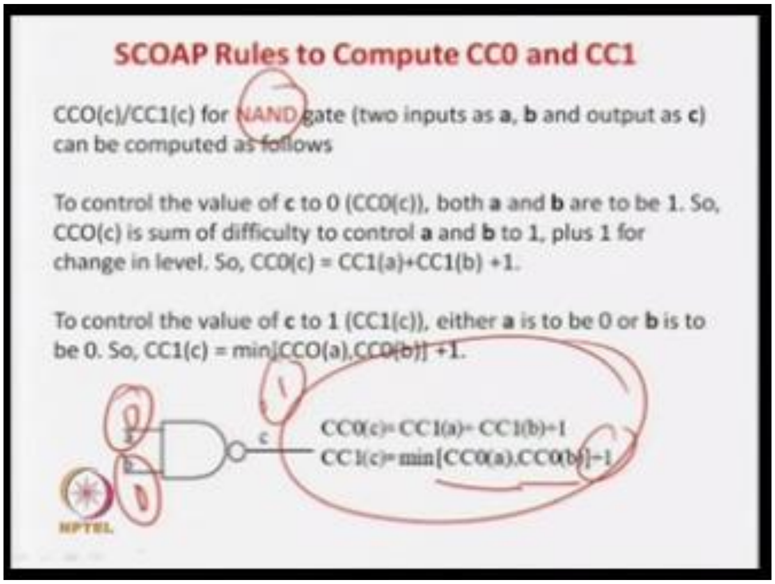
If b is easier to apply to 0 then we will apply b0 and forget about a and they automatically c will be 0, so minimum of the difficulty of making a0, b0 is the difficulty of controlling C to 0, that is which is east a is easy or b is east to make 0 that we will take and we will add 1 that is why we say that minimum of combinational control ability of a and combinational control ability of b that is easier to make a as 0 or b as 0 is the equivalent to combinational control ability of C to 0 and we have to add a 1 because this is a level shape.

Now let us see what is the difficulty of making C as 1, making C as 1 is a more difficult problem because a has to be 1 as well as b has to be 1, there is no choice or something so it is how difficult these two make a as 1 this value how difficult it is to make b as 1 those value you have to add because together we can make C=1 and then you have to add a 1 because there is a level shape, so this is the rule for CC(0) of AND gate and CC(1) of AND gate so I did thus we know the value of CC0(a), CC1(a), CC0(b), CC1(b) that is what is written.

You control the value of C to 0 either a to be 0 or b to be 0, so CC0 is the minimum difficulty to control a to 0 and b to 0 we add 1 as we propagate the level so CC0 is this one. Similarly for

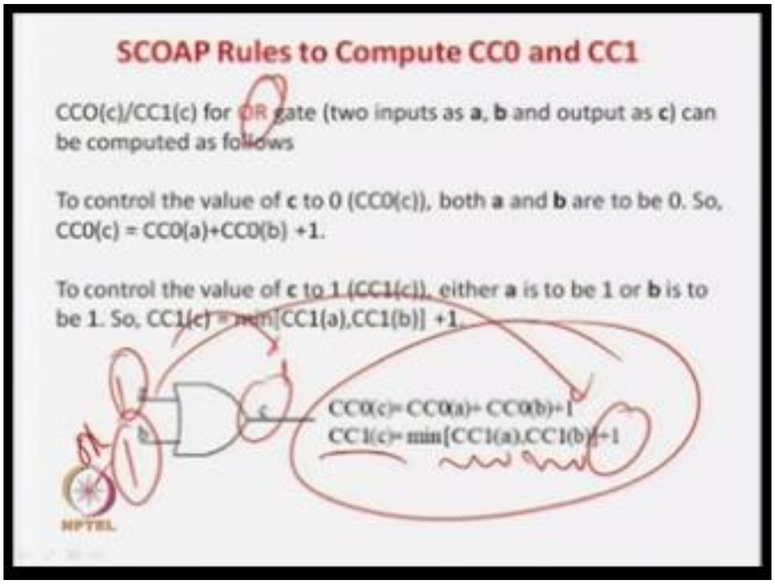
CC1 you have to add them because both of the a and b has to be 1 and these are 1 for label shape. So these are the two rules you have to remember for a AND gate.

(Refer Slide Time: 23:42)



Now we will see for a NAND gate, so how can we get a NAND gate to be 0 so NAND gate is 0 if you have 1 and 1 over here, so difficult to make C as 0 for the NAND gate we are looking at so it is CC1(a) they have to make this one to 1 as well as you have to also make this one to 1 so both the control ability difficulty you have to add plus 1 there is a label shape. Now so this is the value of CC0, now if you have to make C=1 then anything this is 0 or this is 0 any of them is 0 then it will makes this one as 1, so it is how minimum of this difficulty minimum of this difficulty plus you have to add a 1. So this makes you rules for the NAND gate, so you have to remember this rules.

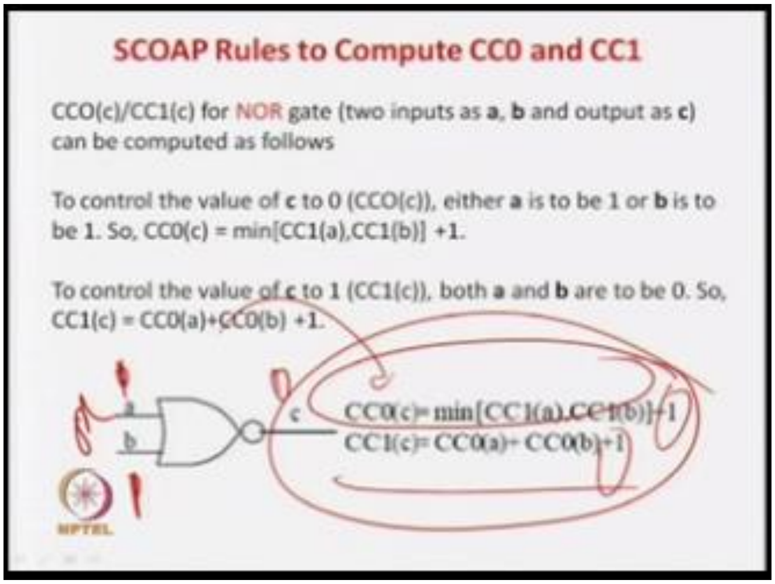
(Refer Slide Time: 24:25)



Now you go for OR gate, so OR gate is always the dual of a AND gate so let us see so if you want to get a 0 over here so you have to apply a 0 and 0 over here because any other combination make a 1. So the combinational difficulty of 0 at this plus combinational difficulty of 0 at b plus 1 is the label shape because both of them has to be 0 and 0. So this is the value for CC0 and this one.

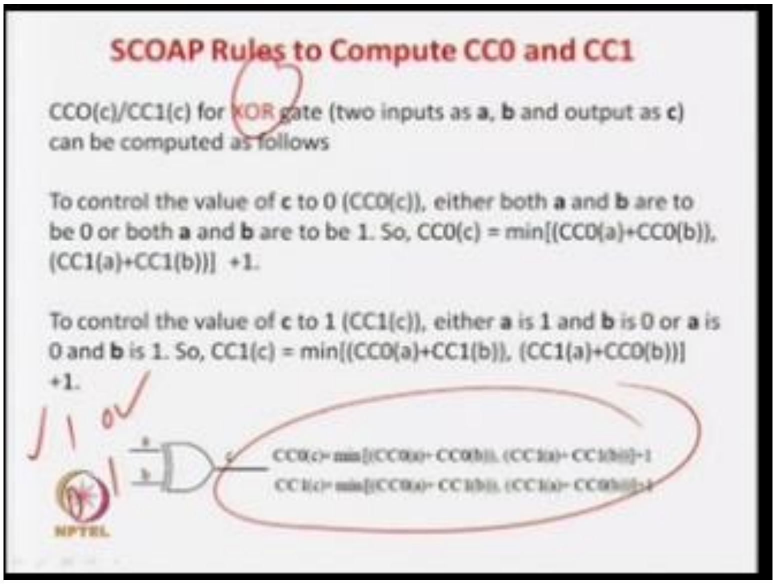
Now to make C=1 then these any either this one is 1 or this one is 1 any of them is 1 will get you this one so is minimum of the difficulty of this one to be 1 and or difficulty of this one into take the minimum of them and you have to add a 1 for label shapes, so CC1 of this one is minimum of difficulty to make a as 1, difficulty to make b as 1 so minimum of them you have to take and 1 you have to add for the label shape. So these are the rules for OR gate.

(Refer Slide Time: 25:13)



Similarly we can go for the NOR gate so NOR gate whenever you get 1 you get a 1 whenever you have a 0 and 0 so this is the value and whenever you get a 0 so it is either this one has a 1 or this one has a 1 so this is your rule, so this is a very simple rule now you can easily understand and 1 is for the label shape. So this is for the NOR gate you have to remember.

(Refer Slide Time: 25:37)



Now you see an interesting case is for a XOR gate that is why we require some fine times for explanation, so now what you was there so in case of the XOR gate what you see that sets 0 we want to get a 0 over here, so what are the possibilities so to get a 0 over here so we can apply a 00 over here or a 11 over here any of these two combinations we will apply a 0 over here, correct. So now what is the idea any of them will have a 0 so either if this is easier to apply then we will get this one if this is easier to apply then we will apply this one.

So either of this two which is easier to do we will apply and we will get a 0 over here, but now this exercise before ends between the OR gate in case of AND gate, AND gate and OR gate so what a idea so if you want to get a 0 over here so either you apply a 0 over here or you apply a 0 over here, either of them which is easier can be used to control this to 0. But there is either this or this any one of them, but in case of an XOR gate one pattern that is input pattern at one input is not going to survive you have to consider the input at both the patterns and then only you can say that which one is easier to test and which one is difficult to apply.

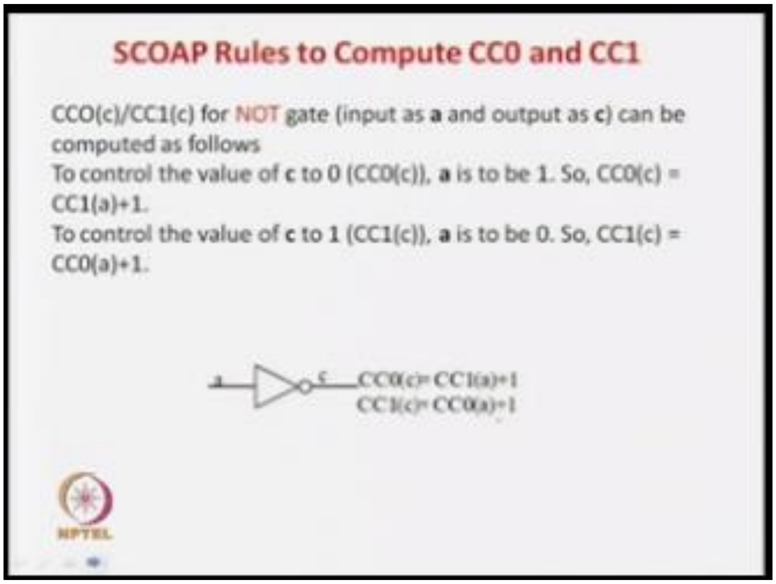
So that is in case of AND gate only one input this difficulty have this level and difficulty of this level if you or it or as you add it you are going to get the answer first combinational control is 0 and 1 at this output. But for the XOR gate it is somewhat different so one input you cannot use one input to get a 0 or a 1 directly for AND gate or a OR gate so one input if you control you can easily get a 0 and a 1 at the output by control use.

For 1 actually for AND gate you have to apply 11 but for a 0 you have to apply a 0 then any of the inputs if you apply a 0 you can get the as they outputs. But for the XOR gate we apply a 0 or a 1 so never you can say that 1 that is one input can control this value, so for XOR gate to get the value of 0 and 1 at the output both the input values we have to know this unlike AND gate will one get 0 implies that this is a 0 OR gate one is a 1 this if you do not know you know the answer is a 1.

So this does not hold for a OR gate that is why the control ability and observibility for the XOR gate is a big different. So to get a 0 over here either of this we apply, so we say that CC0(a), CC0(b) that is difficulty you applying a 00 CC1(a) and CC1(b) that is difficult to apply a 11 so minimum of these two so whichever pair is easier to apply that minimum you have to take that is actually you can give a 0 over here plus you add over so this is the control ability of XOR gate to 0.

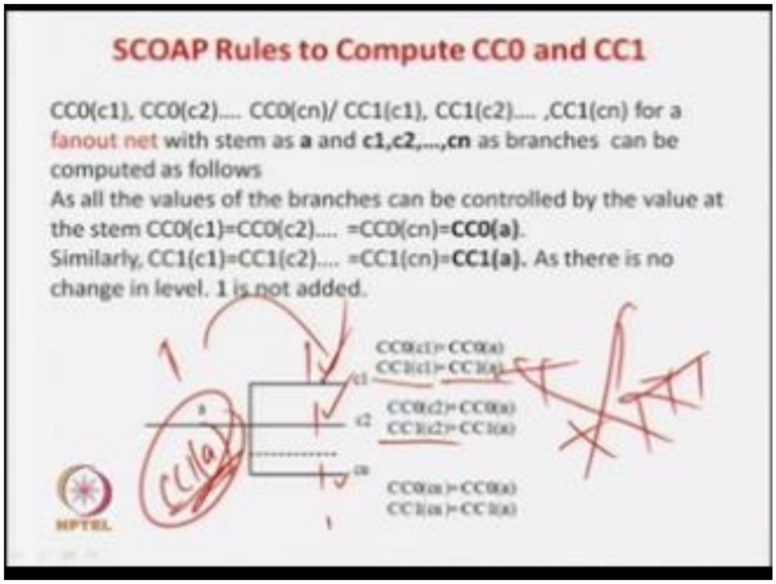
Now if you want to apply a 1 at the output how difficult is it so there are a and two patterns 10 and 01 then you get the answer as 1. So how difficult is it to apply to pattern CC0 sorry, how difficult is it to apply this pattern not this one, so this is CC0(a) and CC1(b) how difficult is to apply the pattern 10 this is CC1(a) and CC1(b). So easier of this two you will apply plus a 1 you have to go for a label shape so this from two formulas will give you what is the what you call it will give you what is the difficulty of CC0 and CC1 of XOR gate, so XOR gate is a big difficult to do this.

(Refer Slide Time: 28:46)



Now for the inverter it is very simple you get a 0 over here you get a 1 over here, so to get a 1 over here what you have to do this equal to apply a 0 over here plus label shape. Now if you want to get a 1 over here sorry, a 0 over here you have to apply a 1 over here so CC0 of this one is you have to apply a 1 this equal to apply so in a label shape. So inverter has a very simple want to add a 0 apply a 1 how difficult to apply a 1 that if I know add a 1 because of label shape you have done. Similarly for this, so inverter is a very simple one and XOR gate as you can think is the most complex one you think about this things.

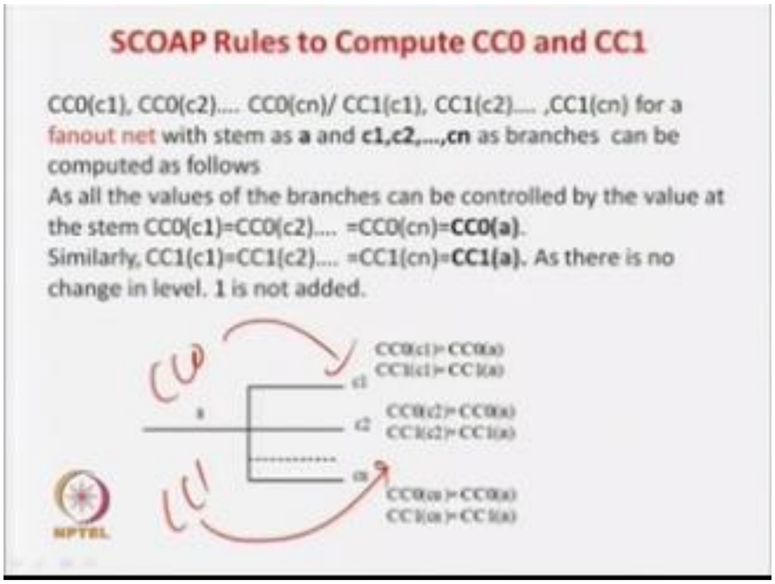
(Refer Slide Time: 29:22)



Now for a fan so say for example, you know that CC0(a) we know and CC1(a) also you know, then if you control this to 0 then automatically all these things will be 0. If you make this one to 1 then automatically everything will be 1, so if the difficulty of this net to be controlled to 0 is CC0(a) so obviously the difficulty in controlling this one to 0 sorry, we are starting with 0 so if you know that CC0(a) some value we know then the 09 is applied in all the nets will be 0 so if we are ply a 0 then automatically everything will be 0 so if you know the default of controlling cc of 0 to cc of a2 0 then automatically same value will apply for c1 of 0 cc0 of c2 cc0 everything will be same because if you just apply 0 automatically it is a carried over. And there is no level shift because there is no gate, so need not add a one.

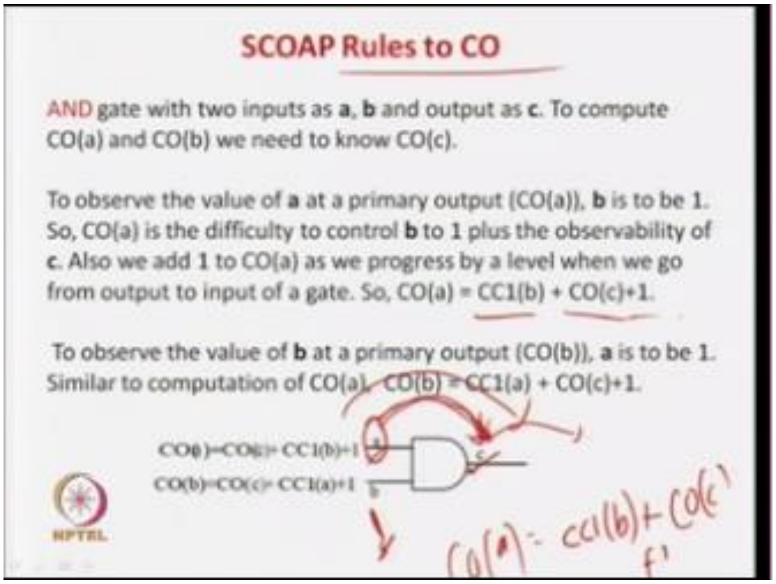
Similarly if you apply one over here then we say that cc of 1 a we know this difficult you know similarly apply over all 111 will go here so this one of c 1 is will be cc1 of a same value this is the how difficult is make a as one same value of same determine of difficulty will be use to apply one over her one over her because you are directly apply this one you are directly get this value, there is no gate enter between so you did not at one so one is not require to added this one is never require to be added.

(Refer Slide Time: 31:02)



At the next level so directly you can say that for the fan outs so whatever is the difficulty of cc sorry cc01 here same thing will be applied cc1 cc one value will be applied her because there is not level ship and nothing is here.

(Refer Slide Time: 31:15)



So still now we have seen the rules to control a line and also there is control line from 0 to 1 sorry so still now we have seen how difficult is true control a line to 0 and how difficult is to make control line to 1 that is the level that require giving a AND gate if you know the controllability value of the inputs how to calculate then output or get NAND gate OR gate all the values we have to find out that if you know this 0 and CC1 of the inputs you can we easily compute the value of cc0 and cc1 of the output.

Similarly we have seen for the Phelps now will see similar rule for CO that is you know that the operability the primary output how do you know that how to control how do you know the value of the primary where is reverse propagation, right so now let us see this case say for example we get CC0 the primary output may be whatever so we know that convenience competitive of C this value we know.

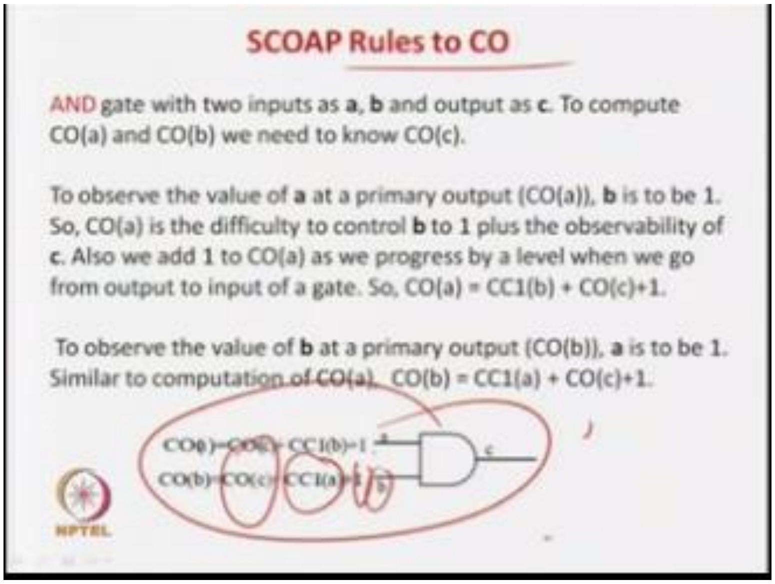
If is the primary output is 0 but say it only non primary input also for the time pay or just see that we know this values so now you know that difficult we are observing this lien this one okay, now we need to find out how difficult is to observe this thing of the primary so you see so doing over that travels the fault affect from this one to this one we need to apply a one at V correct so if you apply a one at V so we know that this value can be forward to the output correct.

So now what we say that so to observe this a so what we need to do first we need to make V1 so we note that combinational controllability of 1 to that has to be done then actually this has to be observe so if we are apply a one then a will be reflected at c correct so now but how difficult is to observe C because a is not reflected from her to here by applying a one over here, so we say that difficulty of combination observe ability of a = you have to make this b as one but if by making b as one a is now only propagated to c it is not propagated here to primary output.

So how difficult is to move it to a primary output that we know is that is equal to combinational observe ability of C, because combination ability have see will tell you how difficult is to observe c of the primary output, so we say that combination observantly of a = combinational controller b to one you make b as one value of a is reflected at c and we know that how difficult is to compute the value of c observe the value of see the primary output we know there is c of 0 this one plus one you have to add because there is a level ship so that is the value combination observantly of a is equal to combinational continue with b2 one conservative c plus one is a level ship.

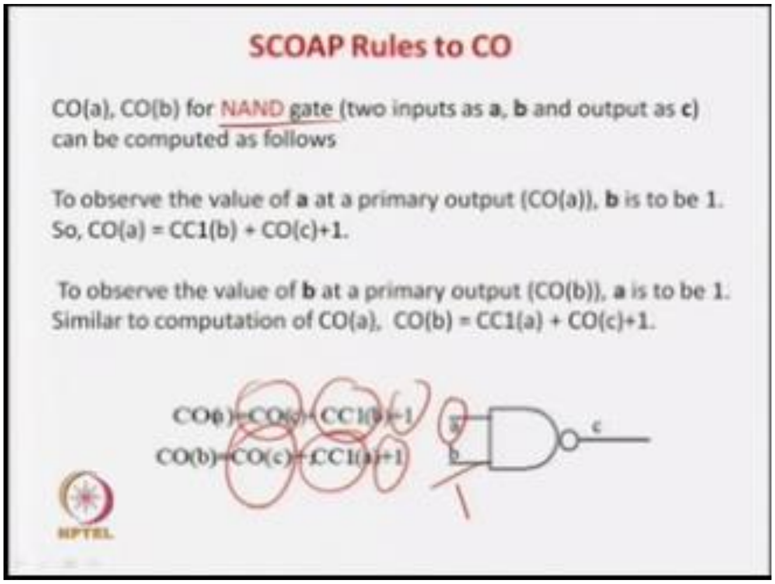
That is what is actually said in the line what I have explain then we say that the observe the value of a primary output b is to be one so co of a is the difficult to control b2 one plus observe of because you are observing a through c so we add one to co as your progressing by your level for output input and so we have to add this control of V observably of c because you are observing a through ca and one for a level sheet.

(Refer Slide Time: 34:20)



Are we get the formula similarly if you want to observe b so what you have to do you have to make a as one so combination control will be a as one plus you have to observe it because b will become observed as so this is the n1 for the level sheet so these are the rules for a observe ability of a AND gate.

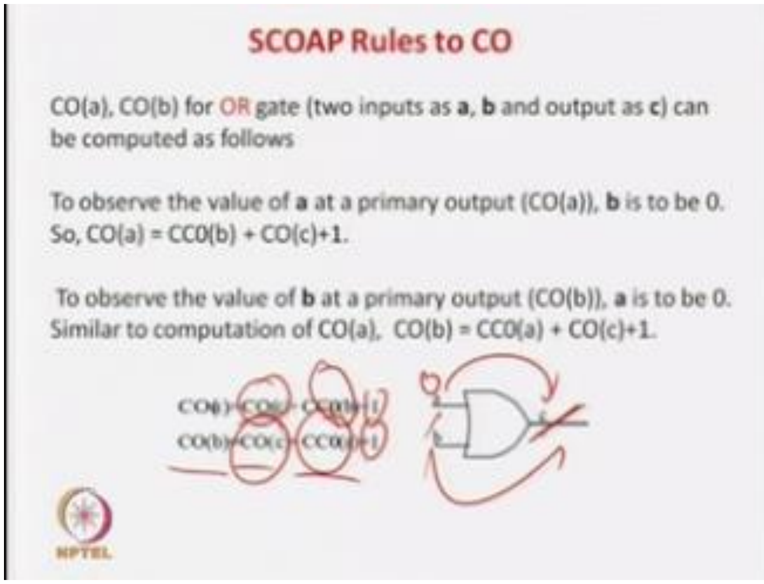
(Refer Slide Time: 34:38)



Now if you let us look at the observe ability conditions for a NAND gate okay so NAND say for example if you want to observe this a then what we have to do if you have to observe this as a we know that we have to apply a one because if you apply a 0 over here obviously it is b\not controllable and everything will be one and you whole purpose of test pastern is the last so you have to apply one so it is a combination control may be 21 A is observe by our c so this is the n1 for the level sheet.

Similarly for b it is a has to be made one so combination controllment of a21 plus you have to observe if b by our c so this is the case and one added for the level sheet.

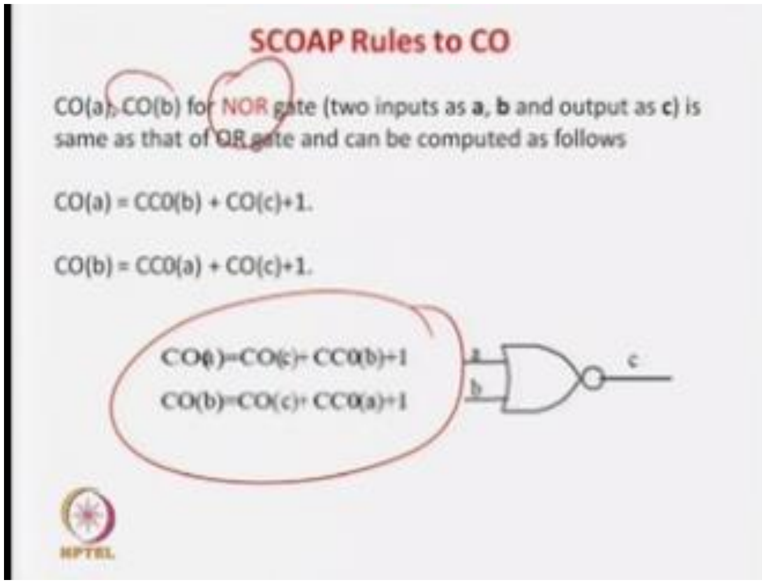
(Refer Slide Time: 35:16)



So these are your rules for observing and b for a AND gate. For the OR gate you always know that it is duel of AND gate so let us see that if you want to observe a then obviously b as to be made 0 if b is one then c will become one and the whole purpose of test patterns is loss so combination of control b21 0 you have to find out this is the case and then cco of c you have take because a is observed by out c but so that we have to add and plus one for level sheet.

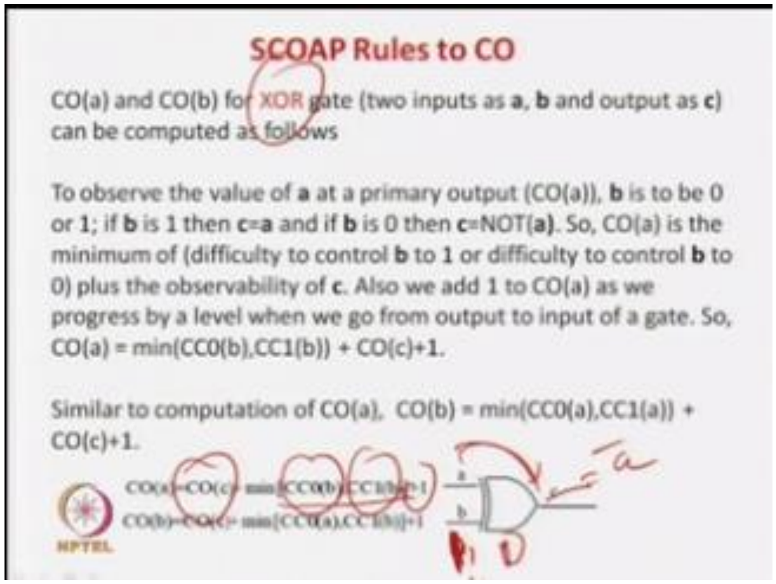
Similarly if you want to observe b then what we have to do the you want to observe b the output then a has to be made 0 one for the level sheet you have to do and you have to observe c we know that the value observe it a of c, that is cb is observe by our c so combinational control observe of b is difficult to observe c plus difficulty to make a as 0 plus one for a level sheet.

(Refer Slide Time: 36:11)



Similarly for a NOR gate also you can find out in the same way that these are the user not going to detail so these are the rules for the NOR gate.

(Refer Slide Time: 36:22)



Now we have to again observe carefully for the XOR gate because XOR gate is the most complex one so let us see how it can be done say for example you have to observe this at this output so now we all know that if b=1 then what is the value of c, c = a bar because we know that XOR gate is a controlled inverter if you apply one bit is a one then the value of you get is as a inversion.

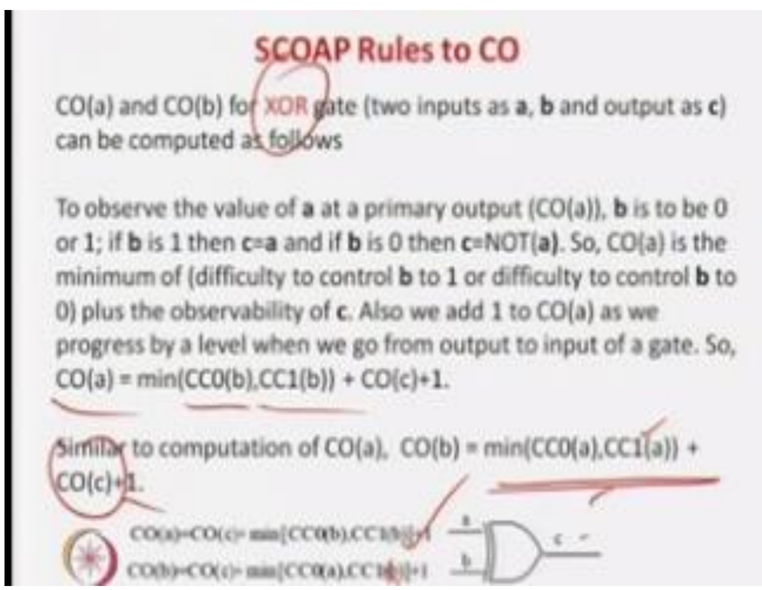
So if you are like say if you apply one over here and let ab 0 so 0 and a 1 we get the value of a one over here so this is x’ sorry the a’ so now if a is one then one and a one you get a 0 at the output c then again you get a a’, so if you apply one bit so that why we call what you called x circuit is the control inverter because if the one input is one may be other input actually get invert.

Similarly if you apply a 0 over here then c will be equal to a so that is the idea, so either you get a or you get a’ depending on whether you apply a 0 over here so if see now both of them observes are purposes because our goal is that we have to observe a you can propagate it so we can observe a also we can observe a’ also both of them solves the purpose like this may be say normal case 0 fault case one and this may lead to normal case one and fault case one.

And both of them will lead to our fault it is so propagation of a or a’ both of them is suffices our purpose so how difficult is to observe a so b is either 0 or one anything is possible so it is minimum of difficult to control minimum difficulty of controlling b20 or b21 so if it is easier to make b =0 you take b then you get c =1 if it is more easier to get b = 1 then you will get a’ five then you will apply one the choice is yours.

Whichever is easier you apply either you will get c = a or a’ so minimum of this one 0 or one whichever is easier to control is apply plus again a will observable by as see so that is control of observable of c is to be there and one for a level job, so these are some what we tricky for the XOR gate compare to the other gates observe ability and control both.

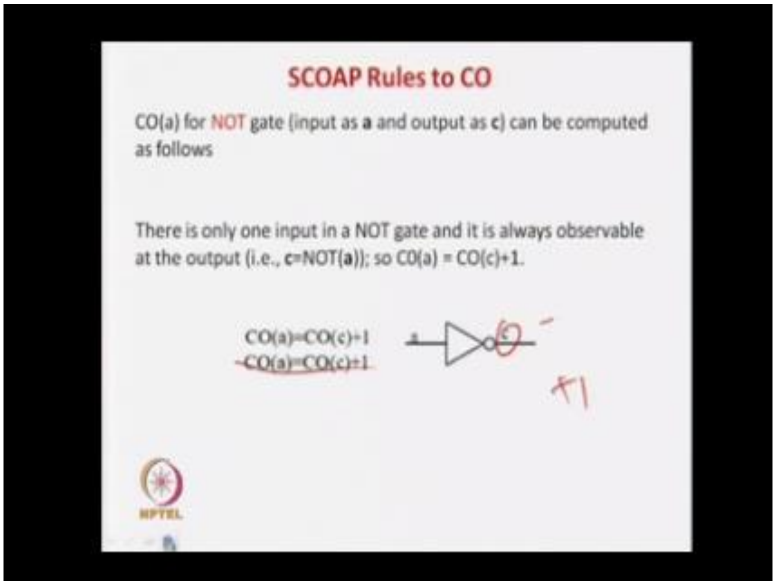
(Refer Slide Time: 38:39)



Now if you say one to observe b at the out so similarly if a is one then you get b’ if a = 0 then you get b okay so both of them is okay for us so is a is easier to make to apply that is a is easier to be get to 0 you do that so minimum of control ability of a to 0 and a’ okay so whatever may be the case you do that okay so there is a slide mistake in this figure I will just tell you so what it is written correctly over here so to control a to 0 it is minimum of cc0 of b and cc this as one so this is actually this is proper but in this case to control b this has to b and a this is correct.

so this is a small type over here just note it so this will be actually this one okay why because if you want to propagate b so if a is easier to control make a make a = 0 if one is easier to get a make cc1 of a so this is the case and one for the level job and it is observable by a c so make this one that is one, so there is a slight mistake so this one also be a correct so these are will go for the computation for XOR gate.

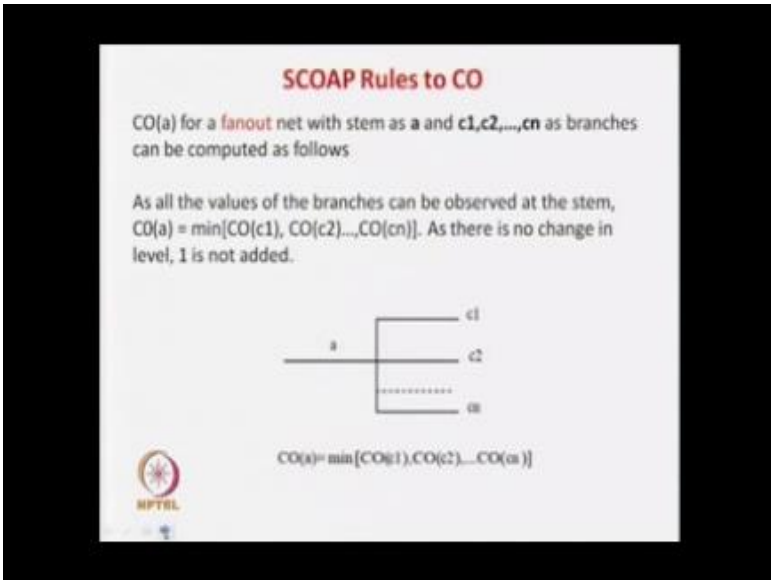
(Refer Slide Time: 40:00)



Now for a inverter, inverter is the most simplest one. So how difficult is to observe this. This one causes this one you have to observe form here so it is very easy to get equal to 0. So you will get A= 1 and it can be observed similarly if you apply 1, you will get 0 and you can observe. So only thing you have to remember is that this one is observed by a C. So difficulties of C only matting over here and this are a level. So if you want to observe A you have to observe C and last one for the level so this is the nothing how much out of A is CC of 0+1 and. This thing is not true.

So both for 0 and 1 this is the frequency of 0 and 1 both cases is validated. Both the cases means it is equal to 0 equal to 1 so both the cases the same phenomenal happens. We need to recognize whether it is not required so you have to observe C you have to observe a plus 1 for the level. So this is the simplest one.

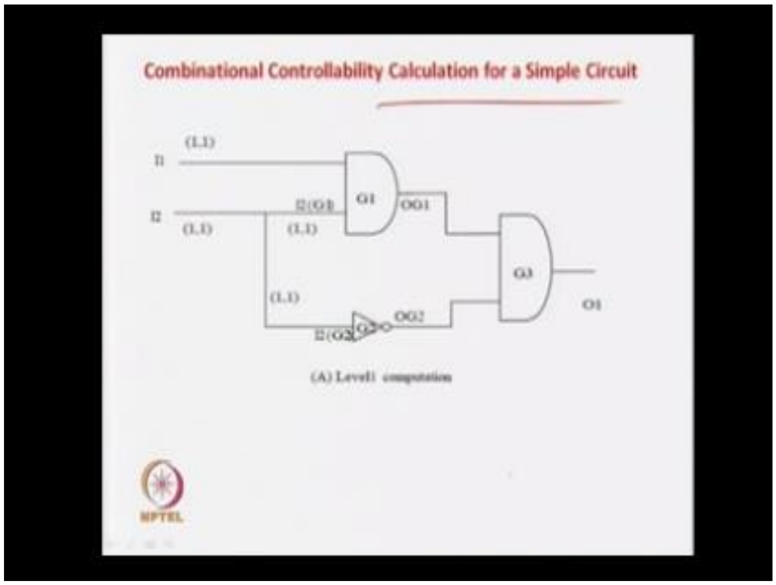
(Refer Slide Time: 40:55)



Now for the now we go for what we go for the fan out. So fan out is the most interesting or you can say fan out to be done in detail I mean it is shown in detail but there will be some thought. So you have to observe this one say why are this, why are this. Now you think that variable combination problem. The variable combination is out of here, but it is directly goes to primary. Then how difficult is to observe the line, how difficult is not observe this, because you want to conserve it so this one we are very complex crowd over here, we need to observe it is very difficult.

Here are also some difficult small line plots, but it is lightly primary over here. So where ever is the minimum difficulty we have to take that it is very simple. So complex observation for A is minimum of observable of this, whichever is minimum observe with difficulty or whichever is easiest so observe among this three that one is previous. So what do we write do you should see what we write we write that conversion observed of is minimum of this, minimum of this or minimum of this. Whichever is easy to observe we take the output of A- that. So this is your formula.

(Refer Slide Time: 42:03)



So now till now we have seen rules for what so we have seen the rules for combination control level of 01 for all the gates we also have seen the combination controllability sorry observably rule for all the gates. Now what we do we will see and we will ex[lore this for an algorithm so for the first circuit it is an example, so let us first see for combinational control we have 0, and combination control for1. So the notification is 00 means this one is formulation controllability of 0 and this one is contribution control of 1.

So this is how you write it, so let us see how we are going above for this. So in this case you see we write this one as 11 why because making this line 0 or 1, that is combinational controllability of 0 combinational controllability of primary input is always one okay. so we write as 11 so this is also 11. Now we have seen that for a fan out if you apply a 0 everything will get 0 arrays or 1 everything will be 11. So controllability of 10 is directly reflects over the output. So we just same so if the controllability of 1 and 0 and 1 and 1. So they directly let over 1, so whatever the value you give it comes.

So this is 1 And 1. Now so this is 11, I have written over here and it is 1. So now how difficult it is to for this level 1. Now you see this is 11 and this is 11. Now let us calculate for this and this level, how difficult is to get a 0 over here. This can be 0 or this can be 0. So how difficult is to make this line to 0 1. So how difficult it is to make 0 it is 1. So either of them if you make as this difficulty is combination controllability of 0 both of them are 1. Now either this is 0 or this is 0 we solve so minimum of 1, 1 is1 plus there is level short so we get the result. So that is 1 minimum of 1, 1 that is combinational controllability at this and this that is this one or this one.

Minimum you have to take last 1 for level. So minimum 11 is 1 +1 is there so it will be 2 over here. So combination controllability of 0 at this, now let us see the combinational construability of this to be 1 so if you want to get this as 1 so what you have to do you have to apply 1 over here and 1 over here. So combinational controllability of 1 at the net is =1. And combinational controllability level is 1. So what is the how can you get one over here this combination controllability of 1 at 1 net+ combinational controllability at V net is at A net okay +1.

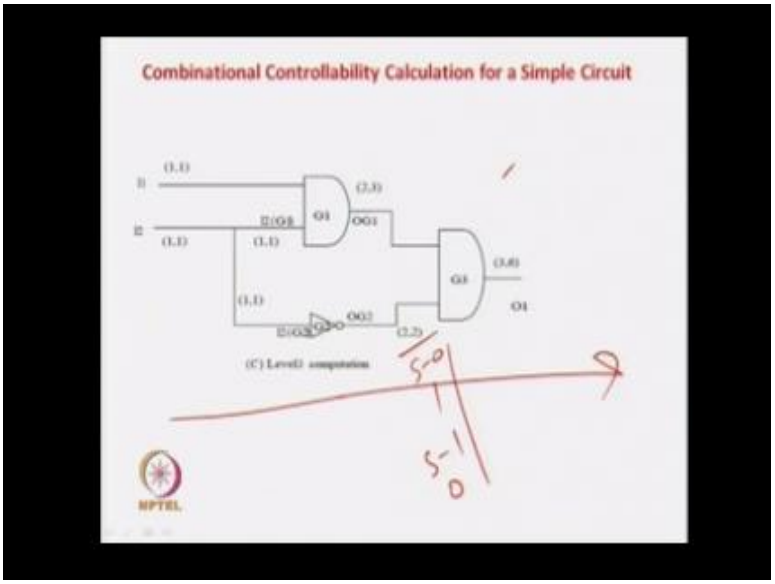
So 1+1 2+1 will be the levels so it is a tree. So we get what we get a simply we get a tree over here this is combinational controllability of 1 at this. That is 1+1 right, now for the OR gate so we want to it. So we know 11 is the combinational controllability of 0 and 1 over here so what is combinational controllability of this one simply this one plus you have to apply 11 for the level ship. So combinational controllability of 0 at 1 is combinational controllability of one here plus one so 1+1 is 2. So now we find out the combinational controllability of 1 here so it is combinational controllability of 0 here last one that is 1+1 is 2, 2 is the value.

So this solves for the level 2, now you have to compute for this one this is also very simple. So you know that we want to get a 0 over here, so to get a 0 over here you have to apply the 0 over here either a 0 over here. So what is the combinational controllability of this net so it is combinational controllability of this net is how much it is combinational controllability this net is how much it is 2 and combinational controllability is also 2. So minimum of 2 is 2+1 for a level ship. So you get a 3 over here okay. So these combinational controllability of 0 this net is 3. Now you want to get a 1 at this line, so to get a 1 at this you have to apply combinational controllability of 1 here and combinational controllability of 1 here.

So what is the combinational controllability of 1 at this is 3 combinational controllability of this net is 2. So it is 3+2+1 is 6. So combinational controllability of 1 at this net is 6, so these are how we can find out the combinational controllability of full circuit by traversing from this level to this level. Now you can easy analysis from some flash light say for example to test a circuit at here. To check socket over here we have to apply a 1 over here. So combinational controllability of applying a 1 here is 6 correct and observing this line because you are socket folder observation of this line does not matter, because it is same for both.

For socket 0 and socket 1 you can diagonal observe this is socket1 so to step on socket 1 you have to apply a 0 so in this case you have to apply a value the value is 3, because to get a socket 1 you have to apply a 0 for to stay at socket 1 for sorry to test the socket fault and you have to apply a 1 so difficult of applying 1 is 6. Socket 0 you have to apply a 0 here after applying 0 here it is 3. So stated 1 fault testing is easy is compared to socket 0 for the step okay. here for display it is the primary output so there observe it is we are not added.

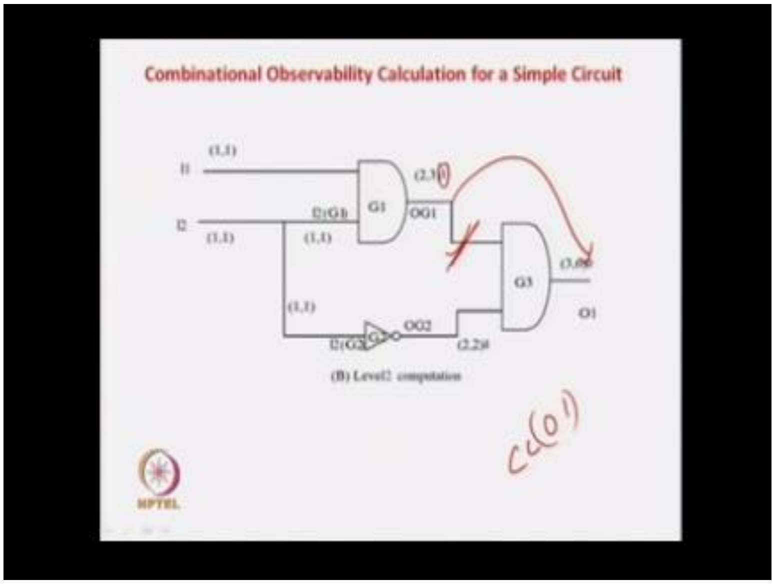
(Refer Slide Time: 47:55)



So this gives you some institution similarly if you consider this so stack at 0 you have to apply 1, socket 1 you have to apply a 0 difficulty at both are 2, 2. So they are equivalent to test because observe ability is same for socket 0 and socket 1. But say for example if I say that I want to compare what is the difficulty of testing at stocket0 fault here and testing stocket0 over here. Say that first I wanted to say socket 1 fault here and socket fault here. So to that you have to apply a 0 over her how difficult is to add a 0 over here. Now how difficult it is to apply a 0 at over this net it is again two, but now 2 can say that socket 1 fault and socket 1 fault is easier but that may not be the case. Why because you have to observe this line to determine the socket function over here you have to observe this line to determine this fault.

Now those lines and this line are not same at the lines are not same so you have to also find out also you have to add that how difficult is to observe this line this combination of this one and combination of this. So which ever will be higher that will be difficult to testing. But now foe example if you want to find out socket 0 is there to test a line or socket 1 is easier to test at this thing. So if you want to compare the value of the difficulty or easiness or socket 0 and socket 1.okay then conservative may not matter much because same line you are comparing the values so same line should be observed.

(Refer Slide Time: 49:30)



Combination observe ability of this net is same of socket 0 and socket1 or both but the difficulty is only coming in the matter but if you want to compare on be form at the is net and one for that net so both controllability and observably will matter because difficulty observing this net may not be same. We see for the same circuit how can we find out the combinational observe ability that us we have to do for this reveres property so now you sees observe ability of this one of 0 in the output. Now thus is level 1 we will go to next level.

So next level see you want to observe e this line this net you want to on so how difficult is to on this net so this net will be observed by this one okay so combinational observability of (01) that 0 as to be added right now if u want to observe this line, this line has to be one so how difficult is to make this line has one this 2 correct and then a making this line as 1 is 2 and then you have to add 1 for the level sheets so observing this line difficulties now say we want to observe this net.

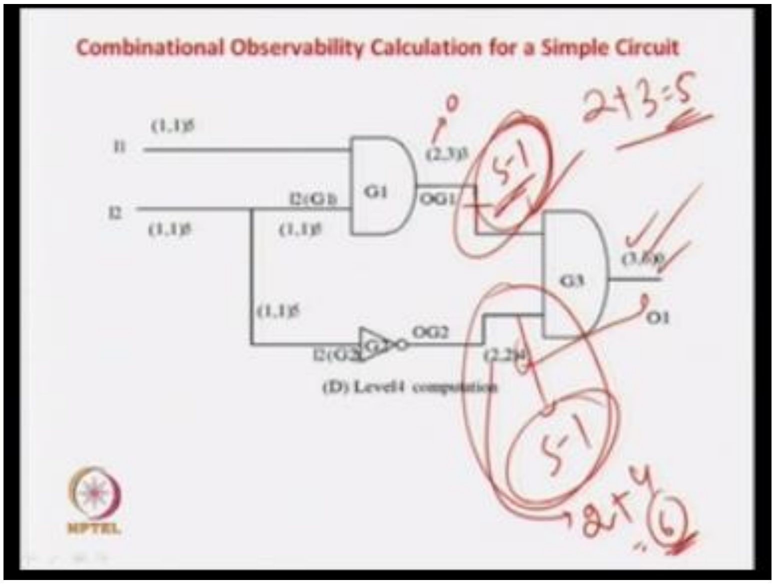
This net we want to observe so how difficult is to observe this net so difficult to observe this nets they have to make this net as 1 difficulties so 3+ so level shift is there so +1 but this net your observing through OR ,OR is the primary output so difficulty of observing OR is 0 so 3+1+0 is 4 so difficult to observe this net and the output is 4 now is about level 2 you have done .Now we will go to level 4 ,level 3 so say for example, you want to observe this net ,you want to observe this net.

Or else observe this net is actually how difficult is to observe this net class 1 so 4+1 is 5 so this is 5 now you say that you want to observe this net so if you want to observe this net you have to make this net to 1 so how difficult it is, it is 1 okay then you have to observe this net why are this nets so this has to be added did you difficult to observed this net is 3 C+1 again. So difficult is to control so to observe this net so you have to apply a, so observe this net.

We have to apply a one over here so how difficult is to apply one over here is 1 so you have added one how this has to observed by or this so this three is over here and one level jump so it is 3+1+1 is 2 ,5 so difficult to observe this net is 5 this how we can calculate similarly if you want to see observe this net ,this net is that we have observed to you have 2 make this net has 1 how difficult is to make this as 1is 1 then if you have to observed this inability wirer this net how difficult so now you have calculated for call accepting this net is have to calculate.

So we know that obviously this net 5 observe this net is 5 so this fan-out can be observed this and this so minimum of them difficulty of observing this or why this is be observability of there, but we are both them are equal so yet the value of 5 so add this been 4 for some reason then we instead say observing this line were this line is easier because it is 4 and this is 5 so could have enough 4 but now in this case what of tem about the stem with both the what you can call this fan-out and fan-out is 5 for this time we will also have a 5 so now this completes.

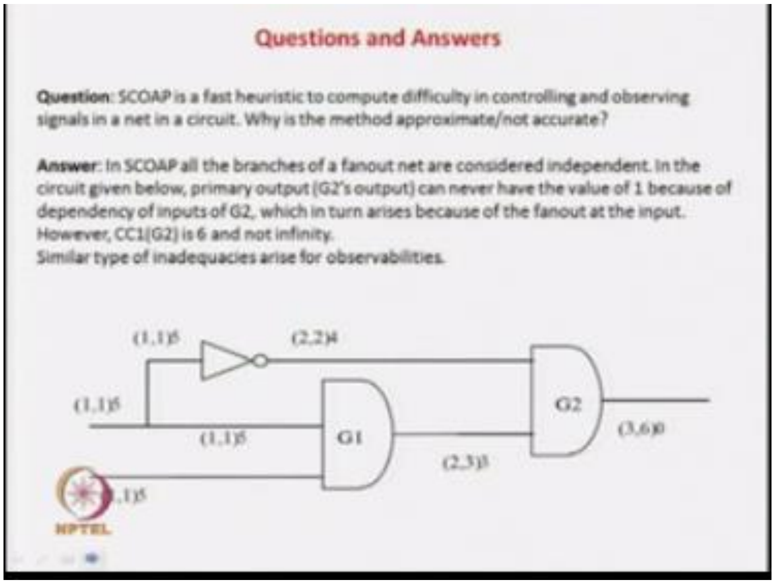
(Refer Slide Time: 52:49)



Our animating the whole circuit with controvelity of 0 controvelity 1 and observed. Now you can easily find out which is the easier fall 2 test which is the difficult fall test. So let us say that we test for a circuit one fault here so to apply circuit one fault here you have to apply a 0 over here so difficulty level is 2 and also to observe this at the output so how difficult is to observe.

This net at the output is the 2+3 =5 so difficulty for testing a stack fault here is a 5 now we say that we want to test a stack one fault this net together stack fault this net you have to apply a 0 difficulty level is 2 now observed this net at the output so difficult to the observe netted output is 4 so 2+4 =6 so you can say that stack one fault testing here is difficulty level is 5 and here difficulty level is 6 so this may be easier fault to test that stack one so this the stack one fault difficulty is 5 these net stack one fault difficulties 6so this may be easier fault than a 6 so similarly you can find out for in one goal.

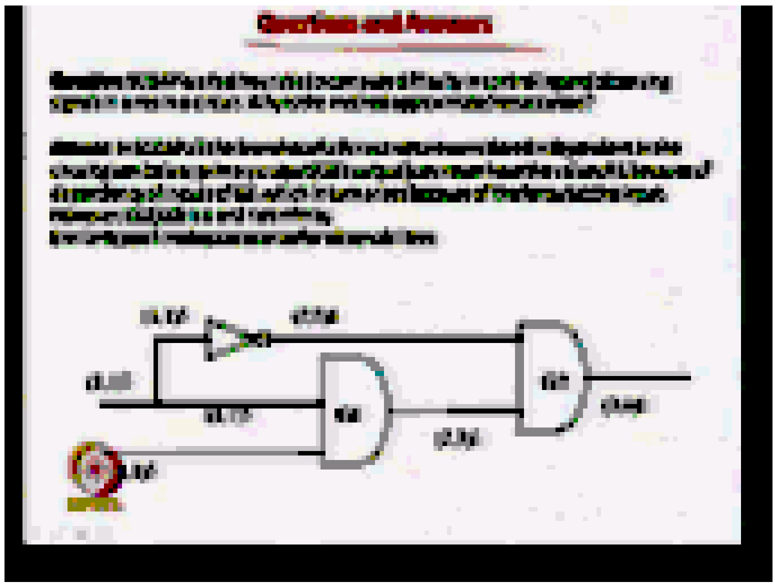
(Refer Slide Time: 53:54)



You can find out the difficulty level of testing the faults by sack faults you want stacked 0 faults if you want to test it will be combinational contribulitiy of 1+ observability and the similar way for if you want to go a stack one another fault stack 0 faults this is combination contribulitiy 1 thus observability this similarly all the faults you can find out the one scan of this circuit one scan of this circuit this will give you combination contribulitiy values the reverse can will give combination observability values the reverse can will give combination observability for all cases you find out what the level of difficulty of the faults now you can setup sexual that whatever is lower than that sexual will be easier to test fault and whatever are difficult to test faults would be higher than sexual now based on this demarcation.

You can easily find out which are the easy to test faults and difficult test faults easier one you will apply random pattern difficult one you can apply what you called the centered propagated and justify approach but this side you restrict so mean they can be lot of in ax curious and also the unitary approach properly if you take a very high trice hold then met me up and that you have very less number of faults and difficult routes series and then in random test patterns in the aimed you may land in problems but you have very low role then lot of false will go to sensational propagate and justify the approach then the beautiful layers and easy layers that is given to you by random will be loss so very careful you are Said the threshold. Now we go to the question answer section the question answer section we are saying that SCOAP is an heuristic which computes this controllability and observability is very fast way.

(Refer Slide Time: 55:15)



Now That mean there will be a lot of inaccuracy can be give some example why it is accurate in this answers yes SCOAP is a what you called an approximate algorithm the main reason of inaccuracy is are it does not in SCOAP all the branches of fan out are consider independent of each other.

That is you can have say this is a fan out so this is one branch and this is one branch okay there some difficulties and all but difficulty level and difficulty level are correlated because this two outputs are dependent on this equal because this are fan outing input and these are the fan out branches similarly you can say that in that same way you can say this guy is dependent on this one.

But because they all belong to a common transitive fan out point but to make the things easier SCOAP does not concentrate this. SCOAP considers that everything is independent this leads to inaccuracy so let us see why okay let us see that combinational controllability of CCG1 let us find out so CC one of G2 so that is CC1 of G2 that is how difficulty is to make this line to one so let us see inputs is one primary inputs that fan out will be 1 1 over here.

This is a level shift so this is 2 2 because inventory is add 1 so now in this case of AND gate how difficulty is to get 1 over here it is see the over here 1 1 minimum of the plus one minimum of 1+1 1,1 +1= it is difficult to this one and difficult to get 1 over here which is 1 1 is in this case so 1+1+3 is there and for the invertors this one is actually 1 1 so this is a level shift 2 2 so now 2 2 over here so how difficulty is to get 1 0 over here it is minimum of 2+1 that is 3 and again how difficulty is to get 1 over here so it is difficult to get 1 over at this one this is 2 how difficulty is to get 1 at this net is 3+1 that is 6 so we get the value of control this net to control.

This net to 1 we have the value 6 but now will see it is very interesting that you never get a 1 over here to get an 1 over here it is saying that combination of controllability of this is 1 is 6 that means shift apply a 1 and 1 to get a 1 correct now to get a 1 and 1 over here you have to get one over here 0 over here so to get 0 you have to get 0 has a here so this is 1 and this is a conflict so you can never get a 1 over here so difficult to get 1 over here is not 6 it is infinity okay.

But SCOAP will say that it is having an final level of difficulty at 6 that is not correct because you cant never get a 1 over here   so stack at 0 fall testing and this point is impossible but still SCOAP will say that you can test it the difficulty level is 6 observability is 0 so it is very good and the level of 6 you can test it but it should tell you in accurate whether difficulty of getting 1 over here is infinity and that is why stack at zero fall cannot be testing because of nontraditional but SCOAP will not able to tell you this because this two inputs.

This input and this input it comes as an independent but they are not independent because these two inputs are dependent on this common transitory fan out point this is dependent on these two okay so that it does not consider and there is lead to an inaccuracy. But it is an very fast algorithm because in one forward and one reverse can you can get out the values for everything which is very easy to do and it is a simple algorithm or sometimes still they are maybe inaccuracy and we have to leave the inaccuracy.

Because I have to get the answers fast so this comes to the end of this module so in the next class will go to another module where we will study in details about sensitize propagate and justify approach because in this lecture what we covered we have covered in details random fall simulation algorithm then we have seen how to find out the easier to test fall and difficult to test fall so this was all about random pattern generation now once I separate the easy fall and difficult fall so in next series of lecture so the next module will take of algorithm say called D algorithm which will solve in a formal way test pattern generation by sensitize propagate and justify approach so that will see in next class. Thank you.

Centre For Educational Technology

IIT Guwahati

Production

Head CET

Prof. Sunil Khijwania

CET Production Team

Bikash Jyoti Nath

CS Bhaskar Bora

Dibyajoti Lahkar

Kallal Barua

Kaushik Kr. Sarma

Queen Barman

Rekha Hazarika

CET Administrative Team

Susanta Sarma

Swapan Debnath