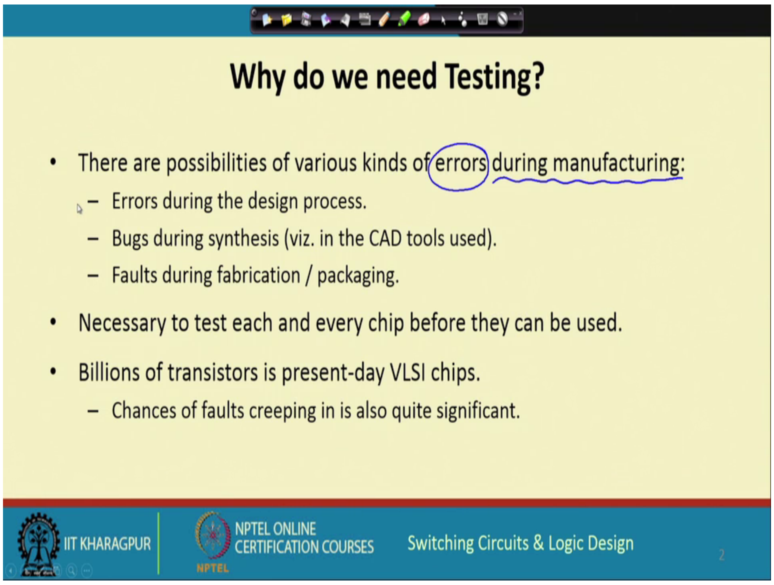
Switching Circuits and Logic Design
Prof Indranil Sengupta
Department of Computer Science and Engineering
Indian Institute of Technology, Kharagpur

Lecture - 55
Testing of Digital Circuits

So, far in the lectures that you have gone through, we have basically discussed different ways in which you can design a digital circuit. But today in this lecture will be discussing slightly different issue related to digital circuits; namely, once you have design the digital circuit, it can be a circuit designed using gates on a breadboard or it can also be a VLSI chip, how to test it? Because, during the process of manufacturing or fabrication or prototyping, there can be so many different kind of errors that can creep in. So, a very important step in the entire design cycle is how to test the manufactured hardware. So, the topic of today's lecture is Testing of Digital Circuits ok.

(Refer Slide Time: 01:16)



Why do we need Testing?

- There are possibilities of various kinds of errors during manufacturing:
  - Errors during the design process.
  - Bugs during synthesis (viz. in the CAD tools used).
  - Faults during fabrication / packaging.
- Necessary to test each and every chip before they can be used.
- Billions of transistors is present-day VLSI chips.
  - Chances of faults creeping in is also quite significant.

IIT KHARAGPUR
NPTEL ONLINE CERTIFICATION COURSES
Switching Circuits & Logic Design

So, let us first try to motivate our self and an understand why we need testing ok? So, I have just now said that during the process of manufacturing. So, when you say manufacturing it can be manufacturing of a chip, manufacturing of a circuit on a breadboard, using available chips you interconnect them using wires, it can be at different levels. It can be printed circuit boards, many chips which are solded together ok.

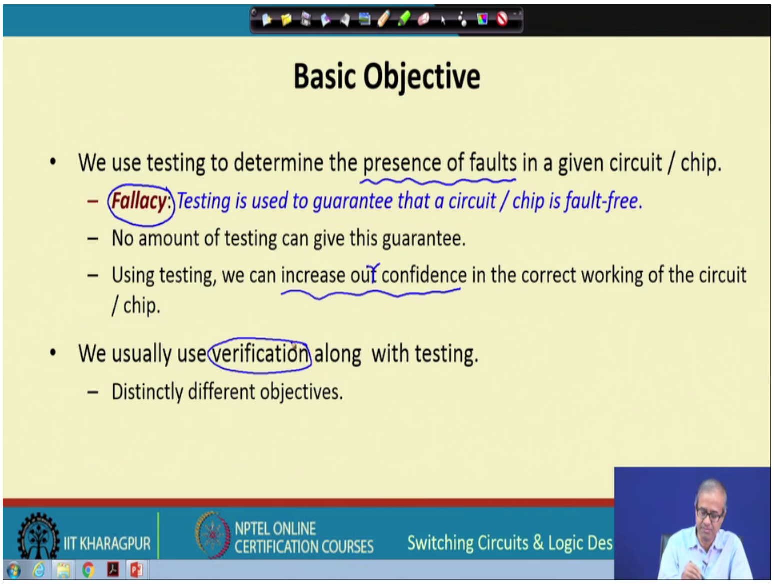
So, once you have such a manufactured hardware, there can be various kind of errors which can come into the picture. Now, here I have highlighted 3 different kind of errors, but there can be other kind of errors as well. First is that but in the design process itself, let us see there was some errors there are some errors during the design process.

So, when you are designing the circuit, because of a human error or maybe the specification was not correct, there has been some problem there or normally we use some software tools for design and synthesis. These are called Computer Aided Design or CAD tools.

There can be some errors in the tools itself, because the tools are huge software. And testing a huge software is an enormous task, there can always be some errors that are setting inside those tools themselves ok. So, there can be some errors or bugs in those tools and finally, during the fabrication or packaging there can be some faults. For example, when you are connecting two points using a wire there can be a loose connection for example, or there can be a short circuit, this kind of errors.

So, because of such error occurrences, it is necessary to test each and every chip or manufactured device before they can be used. If you talk about the modern day VLSI chips, there are of the order of billions of transistors inside them. So, you can understand how complex the circuits are and chances of faults creeping in such a very complex VLSI chip is also pretty high ok because of that the problem of testing is important. Now the basic objectives of testing, what is the basic objective that you trying to meet ok.

(Refer Slide Time: 04:08)



Basic Objective

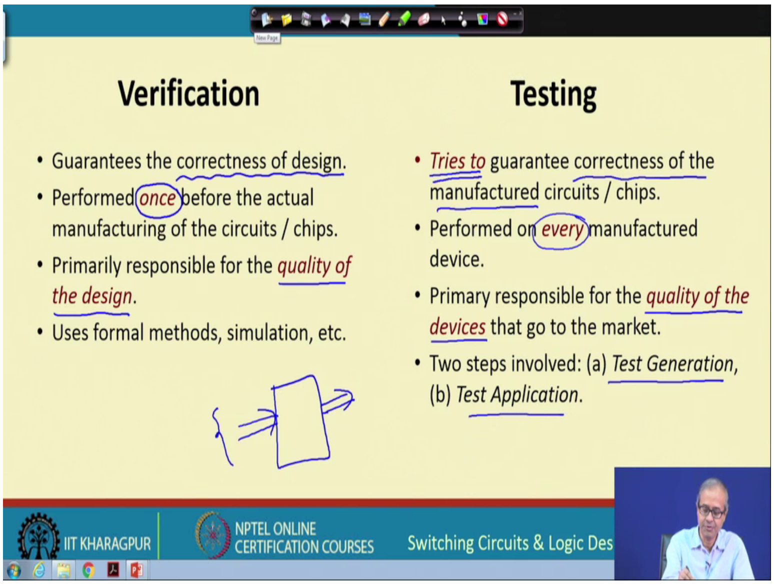
- We use testing to determine the presence of faults in a given circuit / chip.
  - Fallacy: *Testing is used to guarantee that a circuit / chip is fault-free.*
  - No amount of testing can give this guarantee.
  - Using testing, we can increase our confidence in the correct working of the circuit / chip.
- We usually use verification along with testing.
  - Distinctly different objectives.

IIT KHARAGPUR
NPTEL ONLINE CERTIFICATION COURSES
Switching Circuits & Logic Des

The point to notice that we use testing to determine the presence of faults if there are any but, there is a fallacy that someone thinks that well, we do testing to guarantee that a circuit is fault free. But the problem or issue is that no amount of testing can give you this guarantee. Well, you can for example, test and find out that your circuit is working perfectly at room temperature, but as soon as you increase the temperature by 5 degree Celsius your circuit fails.

So, it is not possible to test the circuit against all possible environmental circumstances. Temperature, humidity, there are so many parameters, right. Then the fault fluctuation in the supply voltage there are so many issues so, testing to guarantee that a circuit is fault free this is a false statement no amount of testing can give this guarantee. But, however, what we do using testing we can increase our confidence it should be r, we can increase our confidence in the correct working of the circuit. That is why we use testing and along with testing, we use another term called verification, it is briefly talk about what are the main differences between testing and verification.

(Refer Slide Time: 05:37)



Verification

- Guarantees the correctness of design.
- Performed once before the actual manufacturing of the circuits / chips.
- Primarily responsible for the *quality of the design*.
- Uses formal methods, simulation, etc.

Testing

- *Tries to* guarantee correctness of the manufactured circuits / chips.
- Performed on *every* manufactured device.
- Primary responsible for the *quality of the devices* that go to the market.
- Two steps involved: (a) *Test Generation*, (b) *Test Application*.

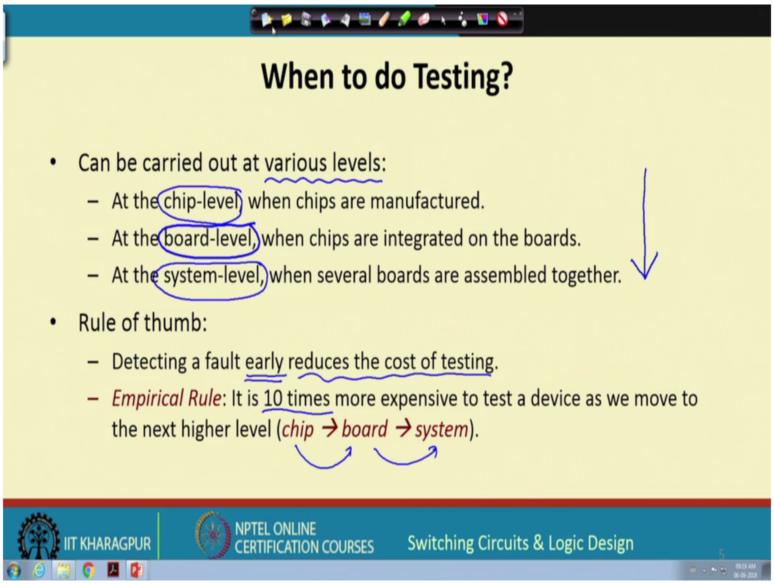
IIT KHARAGPUR
NPTEL ONLINE CERTIFICATION COURSES
Switching Circuits & Logic Des

Verification talks about the design, we want to ascertain whether the design is correct or not, not the actual hardware. Design we can do on a piece of paper, say a function minimize using Karnaugh map that is an example of a design ok. So, verification concentrates on the design and it can guarantee the correctness of a design ok. Now in contrast testing as I just now mentioned, it tries to it can never give a 100 percent guarantee.

It tries to guarantee correctness of the manufactured hardware, it can be circuits chips anything. For verification because we are talking of the design which possibly of done on a piece of paper, this is performed only once before you start the actual manufacturing right. Now in contrast, faults can occurred in each and every device therefore, testing must be performed on every manufactured device, this is one important difference. Verification assures the quality of the design that the design is free of any errors, testing ensures quality of the devices, that the devices are probably free of errors.

Well again I am repeating, you can never give a 100 percent guarantee ok and verification is typically carried out using mathematical techniques formal methods theorem proving simulation. But these are the things we shall not be discussing here, but for testing we normally use a two step process given a circuit with some inputs and outputs, first we find out that, what are the inputs we have to apply for testing? That is called test generation. Then for every circuit or chip we have to actually apply those inputs and verify whether the output is coming correctly that is called test application ok fine.

(Refer Slide Time: 08:05)



When to do Testing?

- Can be carried out at various levels:
  - At the chip-level when chips are manufactured.
  - At the board-level when chips are integrated on the boards.
  - At the system-level when several boards are assembled together.
- Rule of thumb:
  - Detecting a fault early reduces the cost of testing.
  - *Empirical Rule*: It is 10 times more expensive to test a device as we move to the next higher level (*chip* → *board* → *system*).

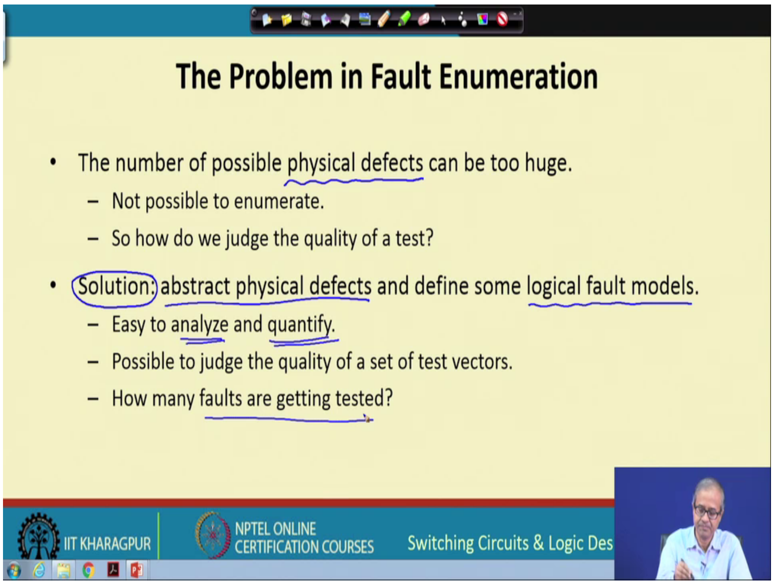
IIT KHARAGPUR
NPTEL ONLINE CERTIFICATION COURSES
Switching Circuits & Logic Design

Now, the question is when we can do testing? You see, when a chip is manufactured let us say, talk about a chip because chip is the most basic of the devices that we use today no design. So, when to do the testing? So, the point to notice that, testing can be carried out at various different levels. Well, when the chips are manufactured we can do testing at the chip level. When several such chips are integrated on a printed circuit board, we can carry out testing at the board level.

And when you build a more complex system, there can be several such printed circuit boards that make up a system. So, when several boards are assembled together, we can finally test at the system level. But the point to notice that as we move up the hierarchy so, as the system becomes more and more complex; the problem of testing becomes more difficult. Testing a single chip may be easier but, when I say I have a system which consists of 1000 chips and it is not working, testing that one thousand chip assembly is a real much more complex problem ok, this is something you have to understand.

So, rule of thumb says, that you need to detect the fault early if possible which reduces the cost of testing. There is an empirical rule well as you move from one level of hierarchy to another chip level board level system level it becomes 10 times more expensive to test the device; when you go from chip to board or board to system. So, it is always good to test a circuit at the level of the chip as soon as they are manufactured ok.

(Refer Slide Time: 10:11)



The Problem in Fault Enumeration

- The number of possible physical defects can be too huge.
  - Not possible to enumerate.
  - So how do we judge the quality of a test?
- Solution: abstract physical defects and define some logical fault models.
  - Easy to analyze and quantify.
  - Possible to judge the quality of a set of test vectors.
  - How many faults are getting tested?

IIT KHARAGPUR
NPTEL ONLINE CERTIFICATION COURSES
Switching Circuits & Logic Des

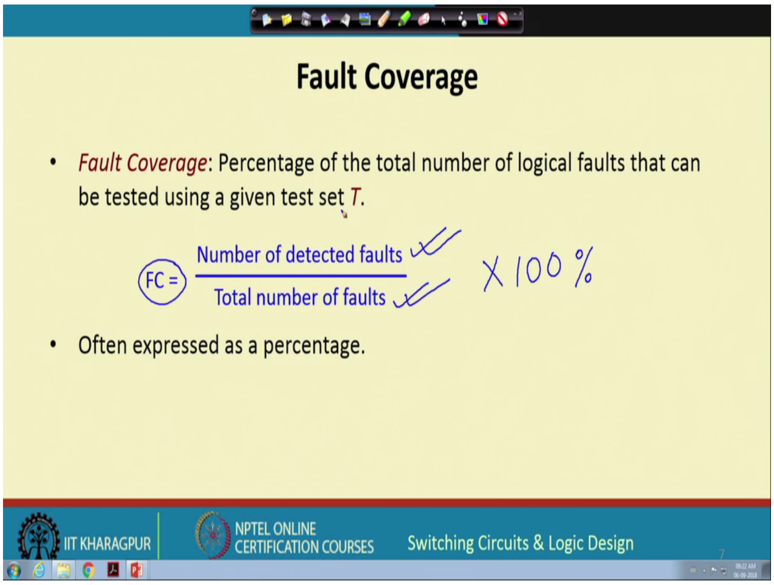
Now, talking about the faults; how many faults may be there when you are doing testing? There are some important points to note. You see, in a circuit, I mean the number of physical defects can be enormously large, well when you say physical defects there can be a short circuit, there can be an open circuit.

Say, a transistor may not be working properly; the gain of the transistor may vary. Now these are all you can say continuous parameters, the gain of a transistor can vary. So, how much varying by how much? You cannot quantify, it can be 1.1 times, 1.15 times, 1.155th times, there are infinitely many possibilities.

So, you really cannot count how many such physical failures are there, they can be quite enormous. So, how do you judge the quality of a test? Now the accepted solution in the industry is that, you use something called logical fault models, where you are abstracting physical defects. Like, you are not thinking about the actual physical defects, like there is a short circuit, open circuit; there is a change in gain of a transistor. Rather, at a much higher level your abstracting the failures and thinking about faults occurring at some other level. Like for example, a gate you assume that one of the inputs of the gate is permanently at 0, that is one kind of a fault model, you can think of ok, is just an example.

So, the advantage is that if you assume such a fault model, then it is easy to count the total number of faults and also you can analyze how good your testing process is; that means how many of those faults are getting detected. How many faults are getting tested ok, fine?

(Refer Slide Time: 12:23)



Fault Coverage

- *Fault Coverage*: Percentage of the total number of logical faults that can be tested using a given test set *T*.

$$FC = \frac{\text{Number of detected faults}}{\text{Total number of faults}} \times 100\,\%$$

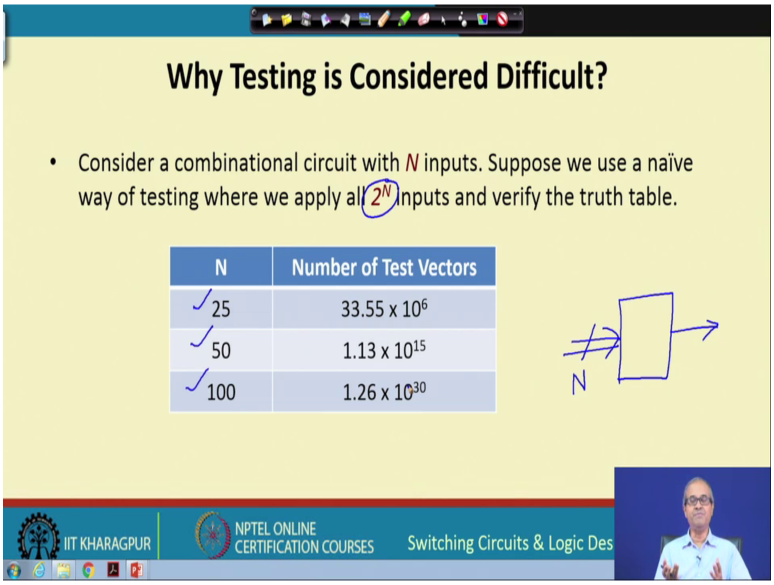
- Often expressed as a percentage.

IIT KHARAGPUR
NPTEL ONLINE CERTIFICATION COURSES
Switching Circuits & Logic Design

There is one parameter we use to assess the quality of testing, this is called fault coverage. Fault coverage is determined by 2 parameters; here the implicit assumption is that we already have a fault model. So, we already have a mechanism using which you can count how many faults are there in the circuit? This is our inherent assumption ok. So, the denominator here says total number of faults and numerator is well we are doing testing and how many of these faults were able to test.

The ratio of these two is defined as the fault coverage. Sometimes you multiply this factor by a 100 and express this fault coverage as a percentage. So, it is essentially percentage of the total number of logical faults that can be tested using a given set of test vectors T. Suppose for a circuit I say, that well these are the 10 test vectors I have to apply to test the circuit. So, these test these 10 test vectors, how many faults can be detected by them this is something which is determined by this parameter called fault coverage ok.

(Refer Slide Time: 14:00)



Why Testing is Considered Difficult?

- Consider a combinational circuit with *N* inputs. Suppose we use a naïve way of testing where we apply all 2^N inputs and verify the truth table.

N	Number of Test Vectors
✓ 25	33.55×10^6
✓ 50	1.13×10^{15}
✓ 100	1.26×10^{30}

IIT KHARAGPUR

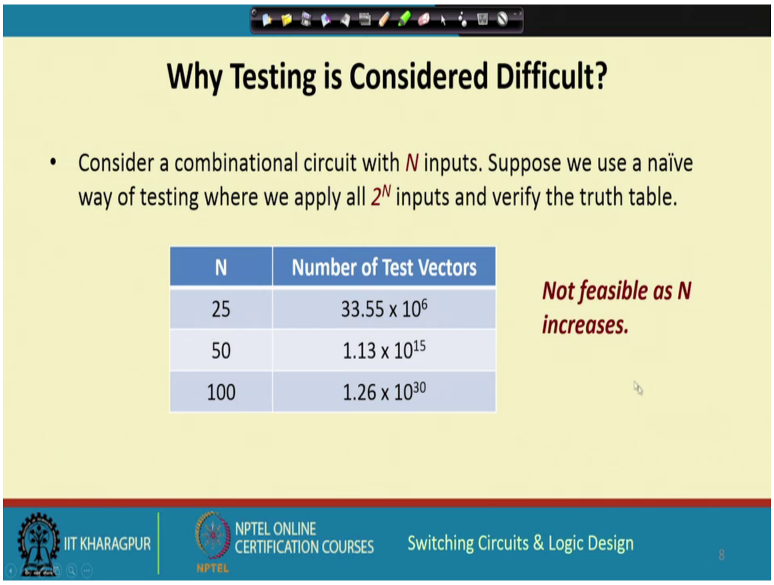
NPTEL ONLINE CERTIFICATION COURSES

Switching Circuits & Logic Des

Now, why testing is considered difficult? Well, you consider a simple combinational circuit. Let us say I have a combinational circuit with certain number of inputs and there can be one or more outputs. Let us say there are there N inputs, well, I can say that well let us follow a simple process, let us verify the truth table. That is the best possible testing we can do we apply all possible inputs and see whether the truth table outputs are matching. So, at the inputs we have to apply 2 to the power N combinations. Now let us think of practical circuits the value of N can be pretty large. So, here I am showing values of 25, 50, 100, but practical circuits can have of the order 1000 inputs as well.

But, if we just show here what is the value of 2 to the power N you see, 2 to the power 25 is about 33 into 10 to the power 6, 33 million, 2 to the power 50 is 1 into 10 to the power 15, 2 to the power 100 is 10 to the power 30. So, you see numbers are become becoming enormously large; you really cannot apply so many test vectors in practice. So, it is just out of the questions so, you would be requiring years and centuries to complete your test. So, this is not possible ok this is why we say testing is a difficult problem, not feasible as N increases.

(Refer Slide Time: 15:37)



Why Testing is Considered Difficult?

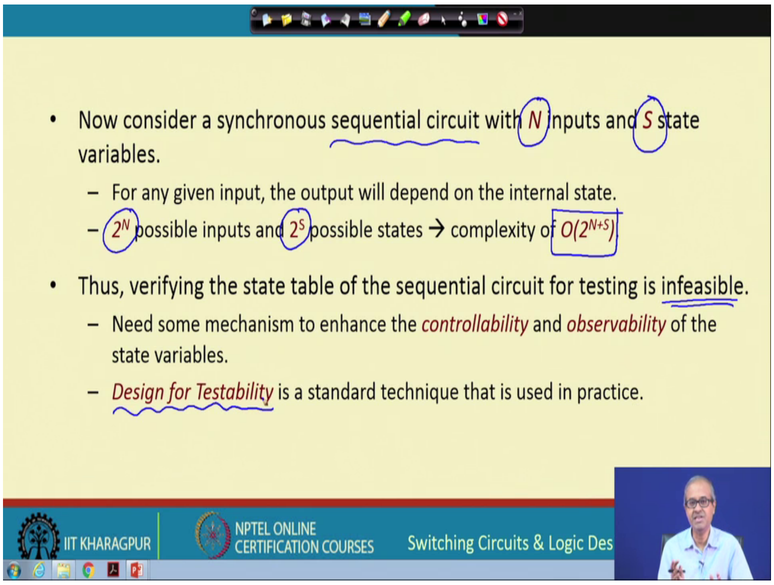
- Consider a combinational circuit with *N* inputs. Suppose we use a naïve way of testing where we apply all 2^N inputs and verify the truth table.

N	Number of Test Vectors
25	33.55×10^6
50	1.13×10^{15}
100	1.26×10^{30}

Not feasible as N increases.

IIT KHARAGPUR | NPTEL ONLINE CERTIFICATION COURSES | Switching Circuits & Logic Design

(Refer Slide Time: 15:40)



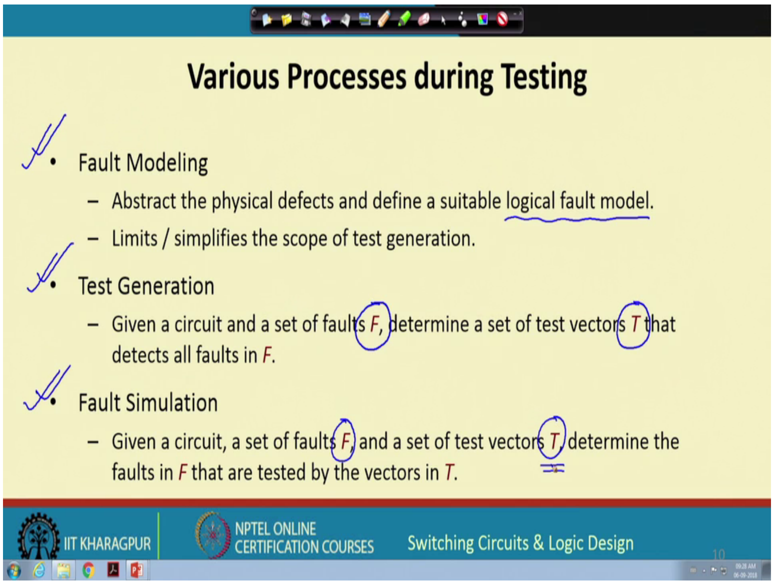
- Now consider a synchronous sequential circuit with *N* inputs and *S* state variables.
  - For any given input, the output will depend on the internal state.
  - 2^N possible inputs and 2^S possible states → complexity of $\boxed{O(2^{N+S})}$
- Thus, verifying the state table of the sequential circuit for testing is infeasible.
  - Need some mechanism to enhance the *controllability* and *observability* of the state variables.
  - *Design for Testability* is a standard technique that is used in practice.

IIT KHARAGPUR | NPTEL ONLINE CERTIFICATION COURSES | Switching Circuits & Logic Des

Well we have talked about combination circuits, but if you consider sequential circuits the problem becomes even more complex. Suppose there are N number of inputs and S number of flip flops or state variables. So, there can be 2 to the power N number of primary inputs and 2 to the power S number of states. So, the machine can be in 2 to the power N multiplied by 2 to the power S that many possible states.

The output will depend on so many combinations, so, the complexity becomes of the order of 2 to the power N plus S. So, the problem is even more complex as compared to a combinational circuit, verifying the state table of a sequential circuit is also in feasible. So, we need some mechanism for handling this complexity. We shall see later that we talk about some method called design for testability; which gives us a practical and feasible solution with which we can address this issue.

(Refer Slide Time: 16:53)



Various Processes during Testing

- Fault Modeling
  - Abstract the physical defects and define a suitable logical fault model.
  - Limits / simplifies the scope of test generation.
- Test Generation
  - Given a circuit and a set of faults *F*, determine a set of test vectors *T* that detects all faults in *F*.
- Fault Simulation
  - Given a circuit, a set of faults *F* and a set of test vectors *T*, determine the faults in *F* that are tested by the vectors in *T*.

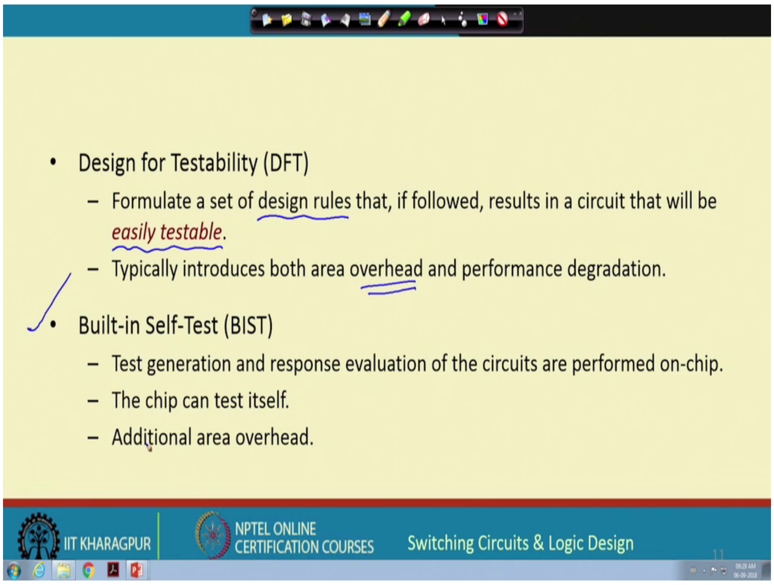
IIT KHARAGPUR NPTEL ONLINE CERTIFICATION COURSES Switching Circuits & Logic Design

Now, let us briefly talk about the various processes or steps that we carry out during the testing of a circuit. Some of this we shall be discussing well, 1 or 2 of this we shall not be discussing as part of this lecture series. The first point I have already mentioned, this is most important that modelling of the faults. Fault modelling here, we abstract the physical defects as we mentioned earlier and we define a suitable logical fault model. So, we shall see examples of this later, this simplifies the scope of test generation. Because, now you can count how many faults are there, let us say for a circuit you say that I have 20 faults.

So, let me try to generate a test set which can detect these 20 faults. But unless this number 20 was given to you, it would be very difficult for you to generate the test set. So, you are testing for what? That question would come right. So, after fault modelling next natural step is test generation. So, you are given a circuit and of course, after fault modelling you have a set of faults.

So, you have to generate a set of test vectors T that can detect this faults this is called test generation. So, here we shall also be very briefly talk about test generation, then comes fault simulation. Fault simulation says, well we have a circuit; we have a set of fault and also set of test vector. Fault simulation is a software that will analyze the circuit and tell you that how many of these faults are getting tested by this set of test vector. Fault simulation is a very useful tool, but however, in this lecture series we shall not be talking about fault simulation. But just remember fault simulation is also a useful step in testing.

(Refer Slide Time: 19:02)



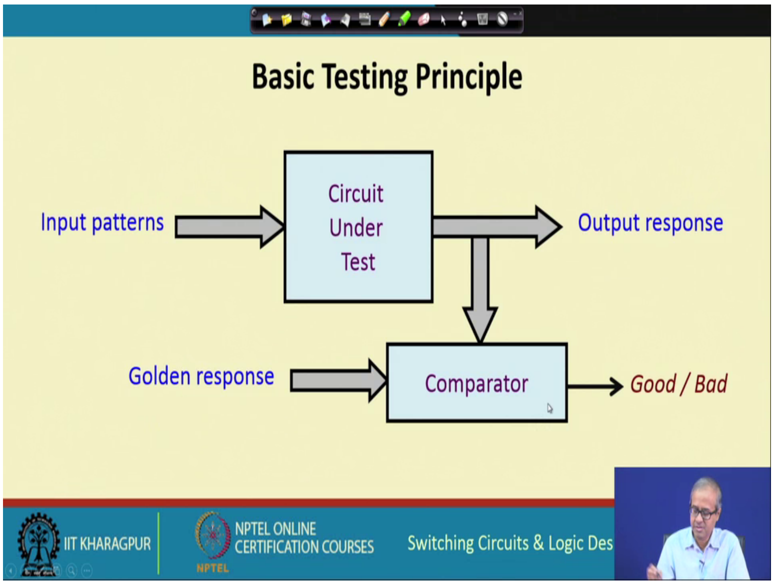
- Design for Testability (DFT)
  - Formulate a set of design rules that, if followed, results in a circuit that will be *easily testable*.
  - Typically introduces both area overhead and performance degradation.
- Built-in Self-Test (BIST)
  - Test generation and response evaluation of the circuits are performed on-chip.
  - The chip can test itself.
  - Additional area overhead.

IIT KHARAGPUR
NPTEL ONLINE CERTIFICATION COURSES
Switching Circuits & Logic Design

Then you have something called design for testability. Design for testability says well we talked about the thing, that the problem of testing is quite difficult. Combination circuit is difficult, sequential circuit is even is even more difficult. So, design for testability or DFT in short, it consists of a set of design rules which if you follow during design, it will make a circuit easily testable. This is the basic idea ok, well of course, if you do this there will be some additional overheads that will come in.

We shall look at one of the techniques later and finally, you have something called built in self-test; where when you fabricate a chip some testing hardware is also fabricated inside the chip itself. The chip can test itself from outside you need not have to do anything, the chip will tell you that I am good or I am bad ok. So, these are the different processes that you typically talk about during the process of testing.

(Refer Slide Time: 20:23)



Basic Testing Principle

So, this is the overall picture, here you have a circuit that you want to test. We apply some input patterns; you get some outputs you know what is your expected output, that is called the golden response. You compare the output response against the golden response and if they match, you said is ok, if they do not match, you can say that your circuit is bad it is not working ok.

This is a very overall simplified schematic diagram. So, with this we come to the end of this lecture; where we have introduced you to the basic problem of testing. Now, over the new over the you can say next few lectures we shall be talking about the different steps that you talked about the important steps, fault modelling to start with that are required in the process of testing.

Thank you.