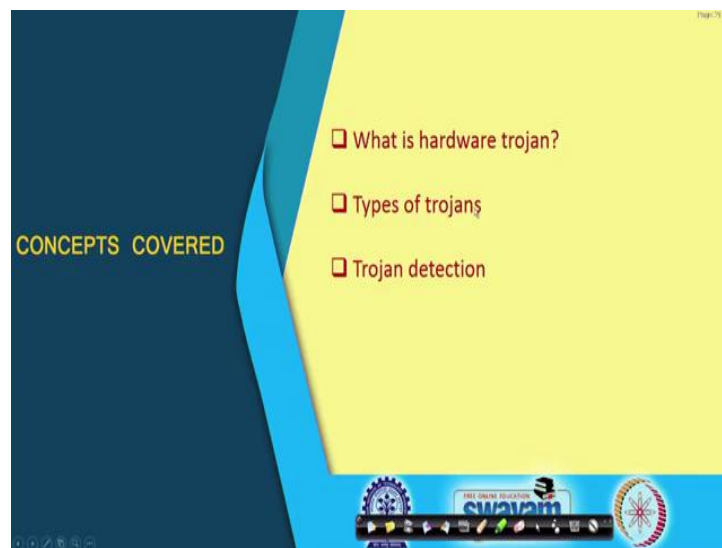


Ethical Hacking
Prof. Indranil Sengupta
Department of Computer Science and Engineering
Indian Institute of Technology, Kharagpur

Lecture – 50
Hardware Trojan

In this lecture, we shall be talking about Hardware Trojans. Now, we shall, we have mentioned in the previous lecture, that a hardware Trojan is something, a piece of hardware which is hiding inside another larger piece of hardware. It wakes up at some unpredictable times and does something which is again unpredictable with respect to the user of the system ok.

(Refer Slide Time: 00:47)





So, let us see, what this hardware Trojan is really like and why it acts like, how it does. So, in this lecture, we shall first be talking about hardware Trojan what it is; the different types of Trojans and how we can possibly detect the presence of Trojans?

(Refer Slide Time: 01:02)

What is Hardware Trojan (HT)?

- It is a malicious modification of the circuitry of an IC chip.
 - During design or fabrication
- A HT is completely characterized by its physical representation and its behavior.
- The payload of a HT is the entire activity that the trojan executes when triggered.



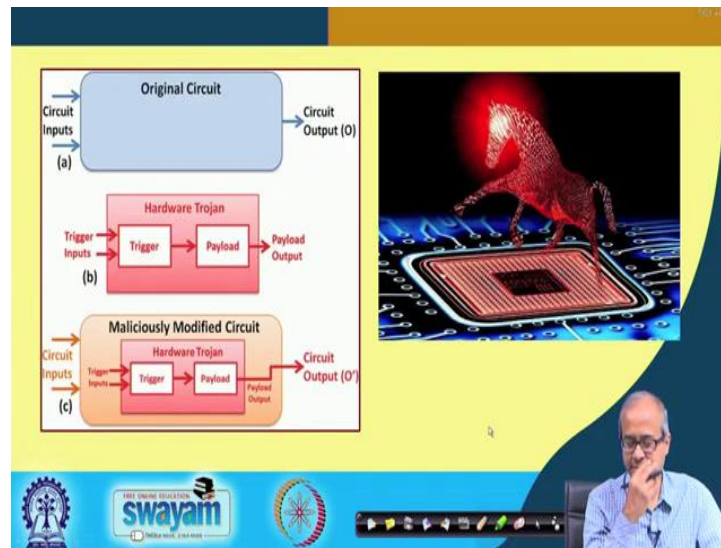
The slide features a yellow background with a blue header and footer. The title 'What is Hardware Trojan (HT)?' is in bold red text. The list of points is in black text with some words underlined. A small image of a wooden Trojan horse is on the right. A small inset image of a man speaking is in the bottom right corner. The footer contains logos for 'swayam' and other educational institutions.

So, hardware Trojans I mentioned before, that the name has come from the ancient story in Greece, where inside a horse a large wooden horse, some warriors were hiding; they entered a castle and they finally conquered the castle. So, this is the picture of that. So, in terms of hardware security, hardware Trojan is essentially some malicious modification of this circuit inside an IC chip. The circuit has been modified without the possible knowledge of the person who had designed the chip.

This is malicious and this modification can happen during design or it can happen during fabrication also. When you are designing the circuit, you may not be aware that someone has modified your design and when you are finally sending a designed for fabrication to a fab, there also when your chip is getting fabricated, you really do not know what is happening there; may be something extra is also getting fabricated inside that chip ok.

So, hardware Trojan, HT sometimes you call in short, is characterized by two things: physical representation, how it behaves, how it looks like and what is its action; how it actually shows up; what are the effects of that hardware Trojan? And there is also something which we called payload; whenever the hardware Trojan wakes up, the action that takes place is referred to as payload. So, payload is the entire activity that happens when the Trojan gets triggered, ok.

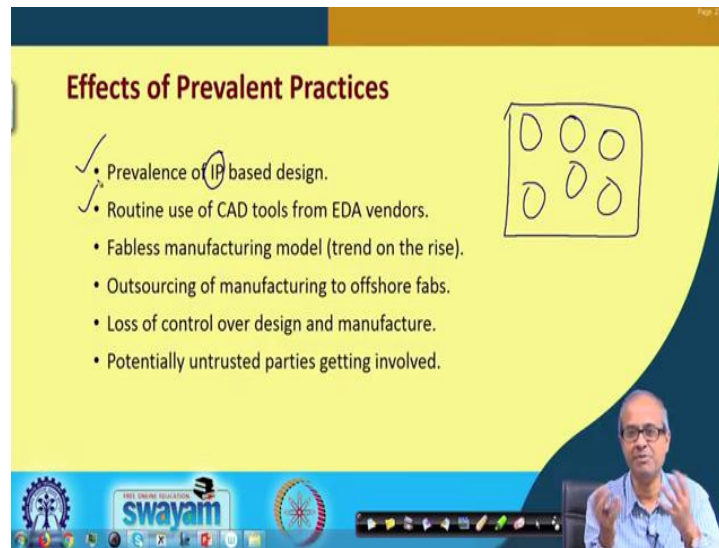
(Refer Slide Time: 02:58)



So, these are just some pictures. So, as I had said, the diagram on the left. Suppose this was my original circuit. So, some inputs are coming, some outputs are obtained, outputs are getting. Now, someone has designed a malicious hardware; this is my Trojan. Trojan consists of a trigger; it decides when the Trojan will wake up and the payload will decide what will happen when the Trojan wakes up. And this hardware Trojan gets inserted somewhere in the original circuit; that means, it goes inside the circuit like this.

To the user, user does not know that something like this has happened; still the inputs are applied; outputs are, outputs are obtained. Most of the time the circuit behaves normally; but sometimes the circuit may behave unpredictably, erratically, maliciously, whatever you say, whenever the Trojan gets triggered. This is the basic idea.

(Refer Slide Time: 04:8)



Effects of Prevalent Practices

- ✓ • Prevalence of IP based design.
- ✓ • Routine use of CAD tools from EDA vendors.
 - Fabless manufacturing model (trend on the rise).
 - Outsourcing of manufacturing to offshore fabs.
 - Loss of control over design and manufacture.
 - Potentially untrusted parties getting involved.

The slide features a hand-drawn diagram of a rectangle containing six circles, with three circles in the top row and three in the bottom row. The slide is part of a presentation, as evidenced by the 'swayam' logo and navigation icons at the bottom.

Now, there are several reasons why a Trojan might get inserted in a design, in a chip? Well here are few points which have been listed. So, maybe one or more than one of these points are responsible for that. First is prevalence of IP based design. See IP stands for intellectual property, not the IP protocol, the network protocol.

In this context for hardware, here IP stands for intellectual property core; when you design an IC chip, the chip can contain many building blocks; these are called IP cores. Now a days, the chips have become so complex that these IP cores we do not design all ourselves. Many of the cores we take from some other place; like a processor core you can take from somebody and mpeg decoder core I can take from someone else and so on.

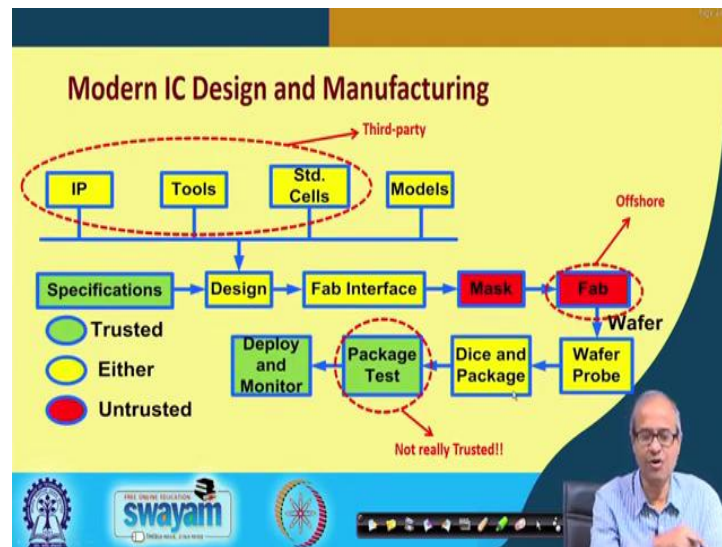
So, I trust that person from where I am getting it. I have not designed that. So, if that person is giving me a core with a Trojan inside, I have no control over it. I have trusted that person and inserted that core in my design, ok. And, secondly when you design some circuit, we use some CAD tools, some software. The software is also not designed by us.

We buy it from Cadence, Synopsis, Mentor Graphics, some company, large companies. So, we really do not know what that company, the person who wrote that tool, did to the software. So, in good faith we are using that software to design a circuit. But whatever circuit is getting designed, how do I have a guarantee that it is the same circuit I am wanting to design or something else has also gone in to it, some malicious piece of code.

Fabless manufacturing model, we do not manufacture ourselves; we give it to someone else to manufacture. So, there itself something wrong might happen. Like here, these are related, outsourcing of manufacturing to offshore fabs. So, we are slowly losing control over design and manufacture. We are not designing the whole thing. We are certainly not manufacturing, most of the time someone else is manufacturing. So, there lot of parties involved.

If one of the parties involved are malicious, then there is a possibility of a Trojan getting inserted in your design. So, there are potentially untrusted parties which are getting involved in the whole process.

(Refer Slide Time: 06:52)



So, let us have a overall picture here. This is a very simplified diagram that how typically we design an IC chip. As I had said that we have the intellectual property or IP cores, the designs we take from other two places. We have the computed design tools that you use. Some standard cells, standard cell library also we can take from some other places. And we can have some models based on which you are designing, some circuit models.

Based on this, we are creating our design. There is a process; we go through the steps of the design; we finally generate something called a gds to file, which is our fab interface; we can send it to the fab for fabrication. There the masks are manufactured and the finally, the chip is fabricated. Now you see, these yellow boxes are not totally trusted and

red boxes are definitely untrusted, because they are typically offshore and when these things are happening, we are not there at all.

We are very far away. So, when the things come back, when the chip comes back to us, when you are doing this package, testing, again we are using a tester from a third party; that is also not totally trustable. So, you see there are so many untrusted parties involved here. So, it is very surprising that, whatever design still works right, but well, we have to trust people and we have to work in a group in collaborative way, right. That is how things work. But, these are some of the reasons why we may have Trojans getting inserted in our design. That is what I wanted to say.

(Refer Slide Time: 08:51)

Hardware Trojans really are ...

- **Malicious modifications to design**
 - Can take place pre or post manufacturing.
 - Inserted by intelligent adversary.
 - Extremely small hardware overhead.
 - Stealthy => difficult to detect.
 - Causes IC to malfunction in-field.
- **Results:**
 - Potentially disastrous consequences.
 - Loss of human life and property.

The slide is part of a presentation, as indicated by the 'swayam' logo and navigation icons at the bottom. A presenter is visible in the bottom right corner of the slide frame.

So, essentially hardware Trojans are, as I had said, malicious modifications to the design, can take place prior to manufacturing, during design, pre or post manufacture; even after manufacturing when the chips are getting assembled, there also a small chip can be inserted in the wafer side by side that may be an extra thing getting inserted. So, you will not know; it can happen at many places.

These are inserted by some adversary who is certainly very intelligent; because no ABC can do this thing; only an expert can do this kind of malicious insertion, malicious modification. And as you can understand that the amount of hardware overhead that is required to insert a Trojan is not much, very small. So, maybe of your entire chip less

than 0.1 percent extra hardware is required to insert that Trojan. Maybe even less; the 0.1 is a very large number.

This is stealthy; stealthy means it is hiding; you cannot see it; very difficult to detect. The IC can manufacture, can malfunction in field sometime; you do not know when. The time is also unpredictable, ok. Result is that, result can be, this can be potentially disastrous; you can have some circuits which you have put on board as spacecraft. The spacecraft has gone to space and during its space maneuver, something wrong happens due to the Trojan. So, the effect can be catastrophic, right. There can be loss of a lot of property and even human life. This is the idea.

(Refer Slide Time: 10:49)



How Realistic are Hardware Trojans?

- Do hardware trojans really exist?
 - No concrete proof obtained as yet.
 - Tampering masks in fab is not easy (highly complex).
 - Reverse-engineering a single IC can take months.
- But there is strong evidence they do....
 - Numerous suspected military / commercial cases (as early as 1976!!).
 - Reverse-engineering of ICs is widely believed to be performed by reputed companies (IBM has patents).

The slide features a yellow background with a blue header and footer. The footer includes logos for 'swayam' and 'INDIA RISES WITH EDUCATION', along with a navigation bar. A presenter is visible in the bottom right corner of the slide frame.

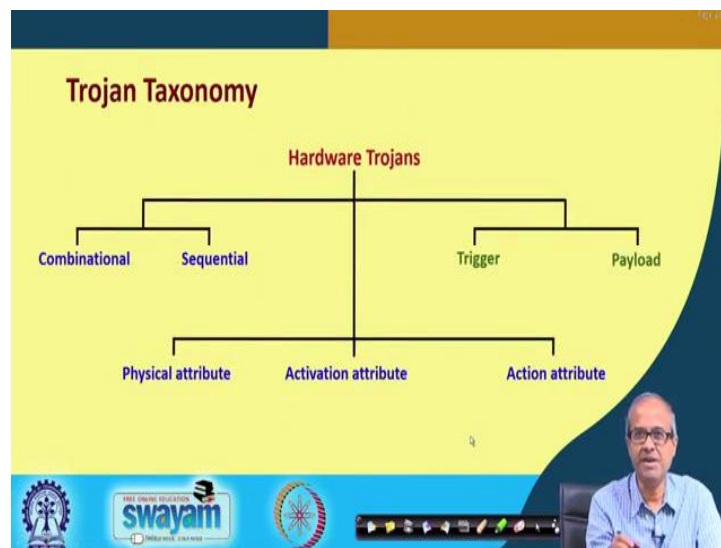
Now, the point is that, how realistic are hardware Trojans? All these, what have been talking about is theory and good stories; but do Trojans really exist? Well, there is no concrete proof that someone has actually inserted Trojan and has actually done it; but there are cases where, people suspect that something might have happened; but not 100 percent sure that it was due to a Trojan, ok.

You see tempering masks in the fab; something happening in the fab during fabrication is extremely unlikely; not impossible, because it is very very sophisticated and very expensive to do that, ok. Reverse engineering of an IC to understand what your original design is, then I insert a new circuit that can take a long time. So, that is also not so easy for someone to do reverse engineer and then insert a Trojan.

But there are number of suspected military and commercial cases where people suspect that something has happened. Like, I am giving one example; I am not taking any names, suppose you have purchased some equipment from some other place; let say guns; guns are very common things which people buy from other places, other countries.

When you buy, you test they were fine; but at time of war, you try to fire the guns and you see that, suddenly you see that the guns are getting locked; they are not firing. So, we suspect that there may be some Trojan inside; someone is possibly controlling your guns remotely from other places, maybe through satellite; something is getting controlled; you do not know, ok. So, reverse engineering of ICs is also quite common in many industries like IBM, many others also.

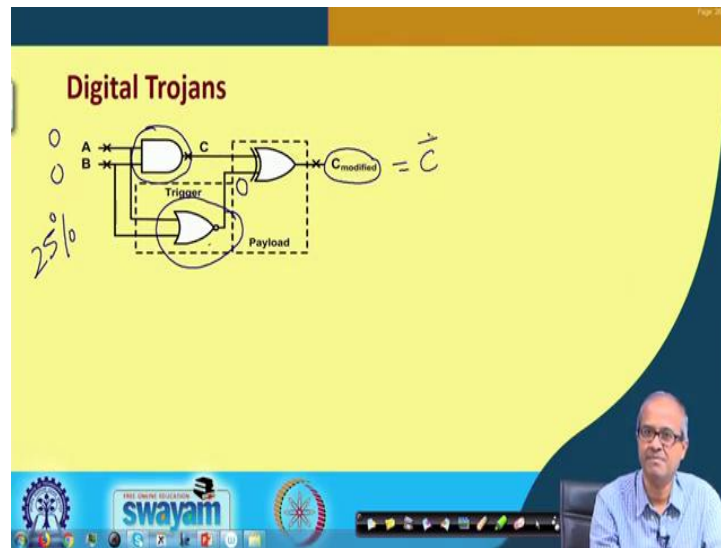
(Refer Slide Time: 12:56)



So, talking about Trojans, there are many different types of Trojans. Combinational, sequential; this is one category. There is another way of classifying Trojans; that what is the trigger condition; what is the payload; that is another way to look at Trojans. And another way to look at, what is the physical attribute? How it looks like? What is activation attribute? When does it get activated an action attribute?

What is the action? Well Trigger and payload are very similar to this activation and action. But exactly physically speaking how it actually looks like; when these things happen. So, there are many ways you can use; you can try and classify Trojan behavior, ok.

(Refer Slide Time: 13:45)

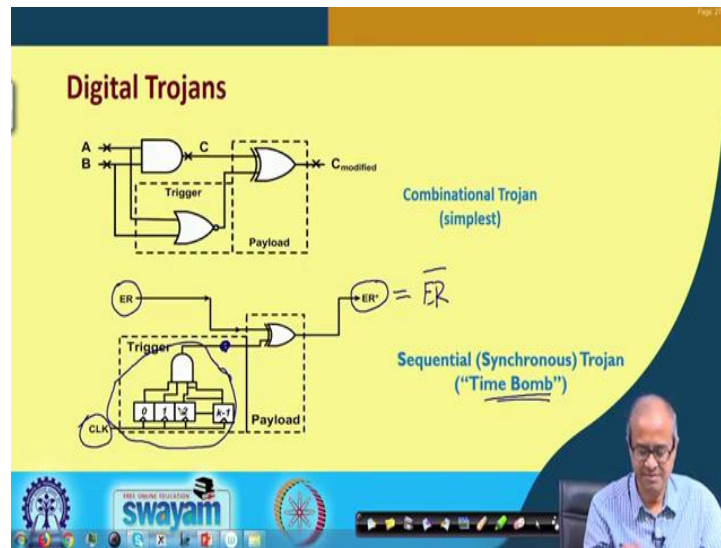


So, some of the Trojan types; let us look very briefly. Digital Trojans, well as the name implies, digital circuits are being targeted, where information are processed in zeros and ones digitally; well, here, let see; I am showing a very small circuit here; suppose our original function was this, a simple AND function. I am taking a very trivial example, a simple AND gate; A and B and this C. This was my original function. And this is a malicious circuit which got inserted.

As long as this output is 0. This C and $C_{modified}$ will be same, no change. So, we will not find any behavior; but as soon as this becomes 1, $C_{modified}$ and C becomes different and something might start happen. So, this trigger decides when this will become 1. You see this is A, in this example, this is NOR gate. This will become 1 when both the inputs A and B are 0 and 0, which has 25 percent probability in this case.

Because, in two inputs there can be 4 combination 0 0, 0 1, 1 0, 1 1. So, 25 percent of the case this Trojan will get triggered and if it gets triggered, this output of this NOR gate will become 1 and C, $C_{modified}$ will be will become \bar{C} , not of that, ok; it will change.

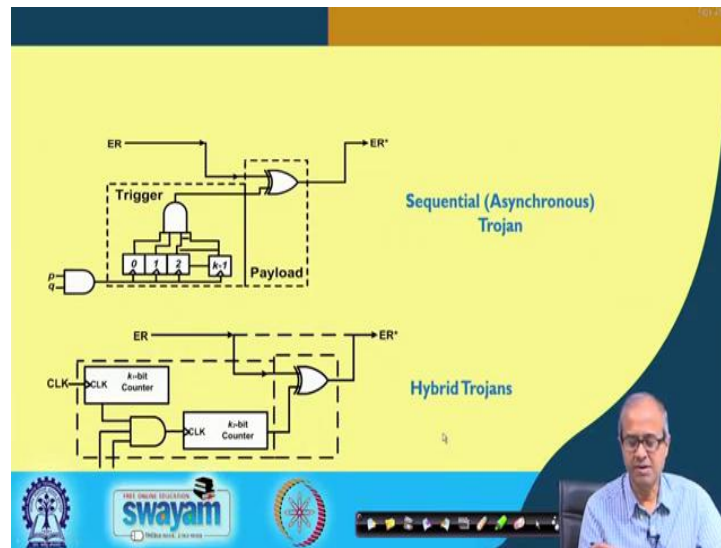
(Refer Slide Time: 15:27)



So, another type let see; this is called combinational Trojan; this is a combinational circuit. Here, it is a little more complex; this is a sequential Trojan. Sometimes, this also called a time bomb, because it depends on some time. The idea is that your trigger condition is something like this. Here I am not showing you this circuit; let us say ER is the value that is calculated; but the Trojan is modifying it to some value ER^* .

Now, what is this trigger? This trigger is some kind of a counter. A clock is coming, the counter is counting. Well this is a simplified diagram; there will be some gates in the counters also. This output value, this will become 1 when a particular count value is reached, let us say 1000; when 1000 clock comes, then only this value will become 1. So, after 1000 clocks, only then this ER will become ER^* , will become \overline{ER} . Before that, the Trojan will not be detected; the output will be correct. So, this is sometimes called a time bomb, because you are defining a time when this Trojan will get activated.

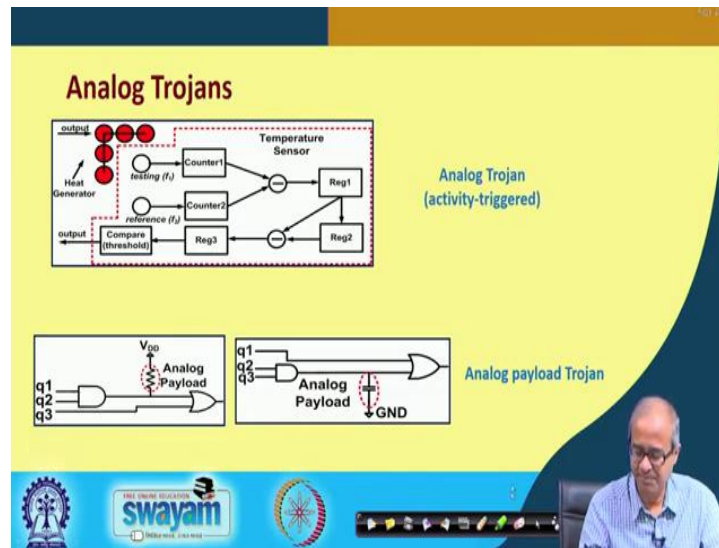
(Refer Slide Time: 16:55)



Now, there are other kinds also, sequential again, but asynchronous; that instead of a clock, you can have some kind of a sorry, you can have some kind of a gating mechanism; like two signals p and q , let say connected to an AND gate; that is connected to the clock of this counter. So, when the clock is coming that is also unpredictable. Whenever these, both signal p and q are 1 and 1, then only one clock will come.

So, you really do not know when the Trojan will fire. In the earlier case, after 1000 clocks, the Trojan was firing; that was more deterministic; but here you do not know. After 1000 such occurrences when p and q are both 1, then only the Trojan will fire. So, that is called asynchronous, because you cannot exactly correlate with clock. And hybrid mixture, there can be counter, there can be synchronous, there can be asynchronous; all sorts of combination can be there. So, I am not going into the detail; it can be as complex as you can think of.

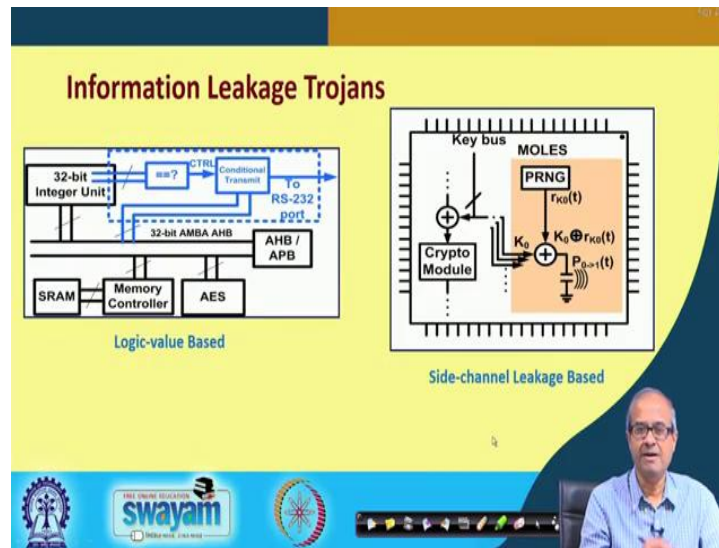
(Refer Slide Time: 18:07)



Well, not only digital, Trojans can be analog in nature also. Let say here, I have given a small schematic depiction; let us say some circuits are there and there is some heat generated; You are, something is happening and the chip is becoming hot; some heat is getting generated. And depending on that there is a temperature sensor. So, the Trojan, the trigger circuit has a temperature sensor inside. Whenever the temperature reaches some level, then only the Trojan will get triggered.

Well, I am not going to detail explanation of this; but just the idea is something like that. There can be some sophisticated analog circuitry which will be measuring the temperature of the chip. Whenever the temperature reaches let us say 60 degrees, then only some triggering will happen. And also some analog payload, you can have directly some kind load connected to the power supply or ground. They can forcibly pull a line to 1 or pull down to 0, whenever something happens, ok. I am not going to detail of these things, ok.

(Refer Slide Time: 19:23)



Now, more interestingly because we have already talked about these, there can be another kind of Trojan which is more dangerous. Like, we already talked about side channel attacks. If there is a side channel, something can leak out and we also have talked about that the designer can try to put in some countermeasures such that this kind of side channel leakages are minimized; we have a side channel resistant design.

But, suppose let us say well, mostly the target of this kind of things are cryptographic chips, where some security operations are going on. Let us look at the diagram on the right. Suppose I have a crypto module; something cryptographic operation is going on here. Somehow, when the chip is being manufactured, some persons, some doing the design, during manufacture some, during some phase has identified that some cryptographic operation is going on.

And some secret value is getting processed here, let say. They have identified some secret values processed here. So, what they do? They use some kind of a special circuit here, take this value and using some kind of a pseudo random number generator generate some random noise. It is not really random, because the person was inserted knows what kind of random patterns are generated here.

So, the pattern of the noise is known to that person and you are doing exclusive-OR with that. So, if the person does a side channel analysis, analyzes the waveform, he already knows what, random pattern was generated by this PRNG, which was the, this was the

payload of the Trojan. And whatever is the secret, that is bit by bit exclusive-ORed and that information is being captured.

So, you are exposing this circuit to side channel attack. The Trojan is exposing it through side channel attack. So, that side channel attack becomes possible. These are very interesting area and here the diagram on the left says, you may be having a processor kind of in format, processor, some processor, memory, bus. So, there can be Trojans sitting on the bus, whenever some activity is happening on the bus, the data on the bus is getting captured and something is happening, something modification is done. So, you can have so many different things, fine.

(Refer Slide Time: 22:09)



Multi-Level Attack

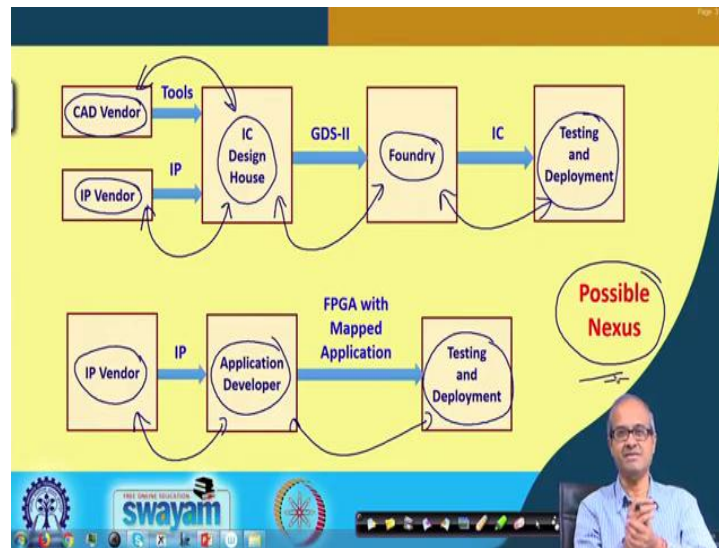
- Uses nexus between multiple parties.
- Only parties which are part of the nexus can benefit.
- The nexus eases the burden on individual parties.
- Additional challenges to detect.

The slide features a yellow background with a dark blue curved shape on the right. At the bottom, there is a blue banner with logos for 'swayam' and 'INDIA RISE, EDUCATION PROSPERS', along with a video inset of a man in a blue shirt speaking.

Well, more dangerous are multi-level attack. We have so far said that, somebody somewhere is malicious who is inserting something, this kind of Trojans inside the design. But if, what if there is a nexus between multiple parties, then your life is even more difficult. Well, the idea is that only the parties which are in nexus, they will benefit out of this thing; because the product that is developed should be some kind of a security product.

So, which if they can break, there will be some benefit, may be monetary benefit or otherwise; we do not know. Let us try to understand what you mean by this. So, I am showing a diagram.

(Refer Slide Time: 23:00)



This is again a situation of IC manufacturing, a very simplified diagram. Let us say, there are, so, this pink boxes are the different parties involved: CAD vendor; the persons who have written the software tools, computerized design tools. IP vendor: the persons from where you are taking the intellectual property cores, IP cores. IC design house: the place where you are designing; well, you can do it yourself or you might have given to a third party for designing.

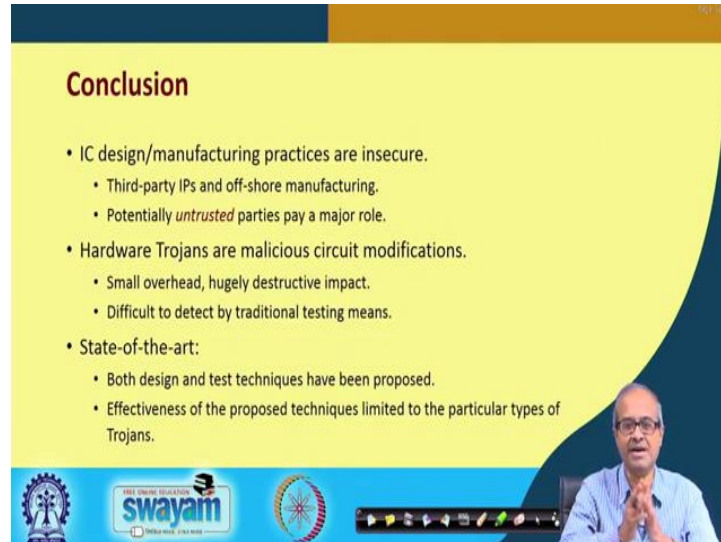
So, it can be a design house also and finally, the place where you are manufacturing the chip foundry and finally, the place where you are testing them. Now, you imagine there can be nexus between any set of people involved here. Like the CAD vendor and the IC design house may have a nexus. IP vendor, design house they may have a nexus. The design house and foundry may have a nexus, foundry and this testing site may have a nexus.

So, if they have a nexus then the effort of inserting the Trojan becomes that much simpler. Because two of the parties know about this thing and they can do it in two different places, two different ways to make it happen; it becomes much easier. Similarly, if you talk about FPGA kind of application, you can again have some IP vendors which are downloaded on FPGAs; you can have Verilog, VHDL codes.

You can have some application developers which may be yourself, maybe someone else and finally, someone will be testing them out. So, here again there can be nexus between

multiple parties. So, the idea is that if there is possible nexus, then the ease of inserting the Trojans become that much easier, ok. This is what I wanted to say.

(Refer Slide Time: 25:17)



Conclusion

- IC design/manufacturing practices are insecure.
 - Third-party IPs and off-shore manufacturing.
 - Potentially *untrusted* parties play a major role.
- Hardware Trojans are malicious circuit modifications.
 - Small overhead, hugely destructive impact.
 - Difficult to detect by traditional testing means.
- State-of-the-art:
 - Both design and test techniques have been proposed.
 - Effectiveness of the proposed techniques limited to the particular types of Trojans.

And just one thing let me say here; I did not actually mentioned about Trojan detection. Just only believe me that Trojan detection is very difficult. Because there are so many different kinds of Trojans that can theoretically exist; no one really knows whether Trojan actually exists in practice or not; but if someone wants, they can always inject Trojans in harder designs, ok.

So, to conclude there are a few points you need to remember and be aware of that the design manufacturing processes in ICs they are inherently insecure; because you are relying on many other part third parties, using their tools and their knowhow and using that you know you are designing your system, your chip. Third party IP is offshore manufacturing.

There are many untrusted parties, that are playing a, not paying, play a major role, there they are actually playing a major role in the IC design process. And this hardware Trojans, which are essentially small circuit modifications, they might easily be implemented if there is a nexus between these parties and these are not easy at all to detect by traditional testing. Because, we do not know when the Trojan will show up; normally the circuit is operating fine, maybe after one month the Trojan will get activated; you do not know ok.

So, state of the art, what is, see there are so many research work that have been carried out both design, how to design Trojans, how to test whether a Trojan is existing in design. They have been proposed, but all these works concentrate on particular types of Trojans; none of the methods are general; none of the testing techniques can detect all different types of Trojan; only specific types of Trojan can perhaps be detected through testing, ok.

So, with this we come to the end of this lecture. So, over the last few lectures we have tried to give you very brief idea, regarding various hardware security issues and techniques that are followed by the IC design community to try and secure circuits and devices. Now with these newer kind of attacks and newer kind of vulnerabilities in terms of hardware, you see the types of attacks we are talking about they become much more, you can say feasible and much more impactful.

So, when we are trying to secure a system, it is not only the software, but also the hardware devices on which the softwares are running, we need to look at the whole thing, the hardware/software ensemble and try to secure both of them. Just only securing software without looking at the hardware is not a good idea at all in the present day context.

Thank you.