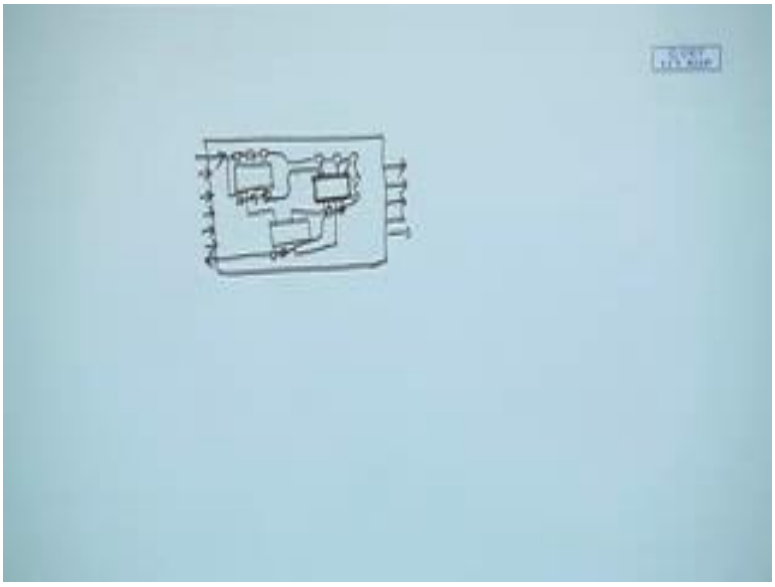
Digital VLSI Testing
Prof. Santanu Chattopadhyay
Department of Electronics and EC Engineering
Indian Institute of Technology, Kharagpur

Lecture – 40
Boundary Scan

(Refer Slide Time: 00:42)



So, next we will start with a new topic, which is known as boundary scan, and it is also used in core-based testing. So, boundary scan means as the name suggest, through the boundary, we will have some sort of scan lines. So, this was very much popular when we have this PCB based designs, where this individual chips like in the PCB if this is the PCB, we have got this chips were there. So, this is one chip, so this is another chip, this is another chip. So, what happens is that while, so there are connections between the chips. So, there may be signal lines running like this, there may be signal lines running like this. Now, this board; it has got some inputs. So, these are the board level inputs and these are the board level outputs. Now, you see that to test a particular chip, so if I want to test this particular chip, whether it is working or not, so at the time of manufacturing, so this chips were tested, so that is fine.

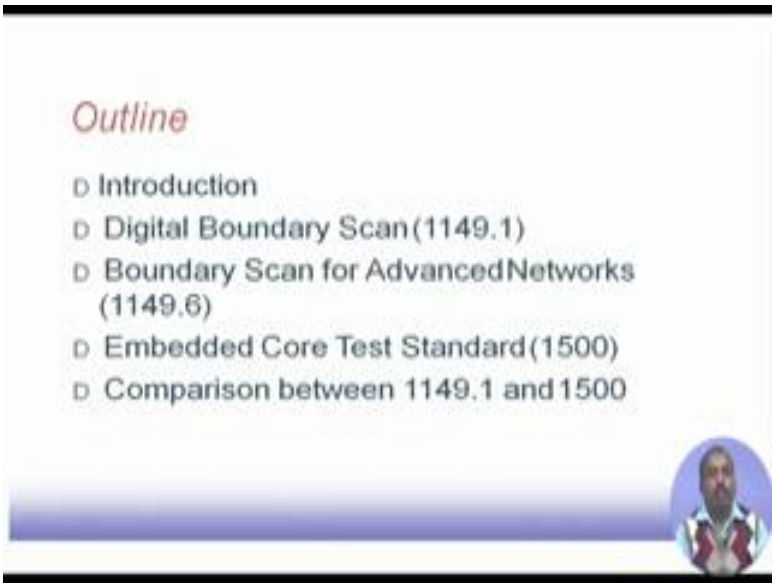
And at the time of manufacturing the board, so what we could do, we could we assume

that this chips are all right, but this interconnections we needed to test, so that was done. But later on when the board is operating, so if there is some problem that has grown. So, we need to check whether these chips are ok or not. So, for that pattern, for that purpose, so without picking the chip out of the slot, so if you want to test it then we have to apply some test pattern to this chip for example. Now, since this signal line is coming from here, so I need to make this chip operate in such a fashion, so that the desired output comes at this point and that is basically a very cumbersome process, because chips are not that simple as logic gates individual logic gates. So, at the circuit level, we could try to do a justification and try to get that using some ATPG algorithm. So, we could try to justify how this line can be made 0 or 1, but in at a board level that is not possible.

So, what is done is that every, every chip that we have. So, around that we have got a chain of flip-flops. So, we have got some sort of chain of flip-flops and they are connected in a chain like this. Similarly, this chip is also having some chain of flip-flops surrounding and this chain is connected to this. And ultimately, it is connected from a system input, and here it maybe this is the last few flip-flops and they are connected to the output. So, what is happening is that we can pass any pattern into this chain, and we can say after how many shifts the pattern will reach the particular chip input and that way these inputs are can be fed to this chip. So, maybe I have got connections like this, and these inputs can be fed to the chip and similarly the responses from the chip can also be collected, so maybe this side we collect the responses and they are ultimately connected to this chain. So, this way we can have a very elegant way of testing the chips without taking them out of the board.

So, if it is an automated process, that will send the patterns to the individual chips without taking them out of the board, so that way the testing becomes easy. So, this is the concept of boundary scan. And with the invention of this core-based design, so this testing process has become complex; and again the ideas of boundary scan has been incorporated into these core-based testing mechanisms. So, these are the 2 things that we are going to see.

(Refer Slide Time: 04:02)



So, apart from introductions, so we will see the standard 1149.1. So, this is an IEEE standard which is for digital boundary scan where the chips that we have they are all digital chips. Now, there is some advanced protocol 1149.6 which works for these analog chips also. So, if you have mix signals chips then this will work for that. So, we will not go into the details of 1149.6 knowing that it is otherwise similar to 49.1, but it can also handle the analogue chips in it way. Then we look into this embedded core test standard, which is 1500. So, this is an updated version of 1149, so that can that is at a chip level. So, this 1149 is at board level; and 1500 is at chip level. We will have a comparison between 1149 and 1500 also.

(Refer Slide Time: 05:02)

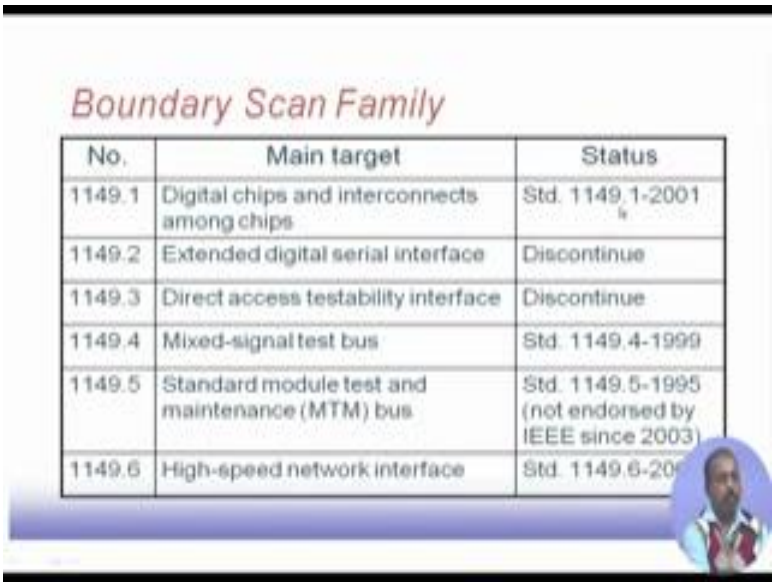
So, the basic objective of boundary scan was to develop board level digital testing. So, a board level, so we do testing, but this is for digitalize original objective is for digitalized ICs. Now, it is also applied to MCM, MCM stands for multi-chip module, so you have got a single chip, number of chips found for that of mounted on the same board, so that a multi chip module. And we have got filled programmable gate arrays. So, FPGA also the design is very much modular because we have got the similar set of configurable logic blocks distributed throughout the chip. So, if you want to test each of them then we have to again send some test pattern to each of these CLBS and that can be done using some chain architecture, so that the test patterns can be fed to this individual inputs of this CLBS. So, that way we have got this FPGA also, they have got this boundary scan thing.

Analog circuit and high-speed network, so there as I have already said the 1149.6, so that is actually targeted to that. So, it can use this analog circuits and high-speed networks. And we can also go for verification and debugging clock control power management chip reconfiguration, so for many other purposes we can use this boundary scan. Basically it becomes a tool to command to various chips, like if a chip has got say different power modes and if you want to change the power mode of a chip then you can send the command via this boundary scan. And then there may be an interpreter or analyzer available at the chip input which will be looking into this command coming from the

boundary scan, and accordingly it may reconfigure the chip to go into a low power mode, so that way we can have this thing.

Similarly, for verification and debugging, this is useful because if you want to verify a design, we need to check what sort of values we are getting at different pins inputs and outputs, and that can be obtained only by this. And debugging also like something is going out, so is not working properly, so you can do a debugging there. So, in mid 1980s, there was a there was an interface JETAG and then in 1988, so that get modified to JTAG. So, so then in 1990, it came up with the first boundary scan standard 1149.1.

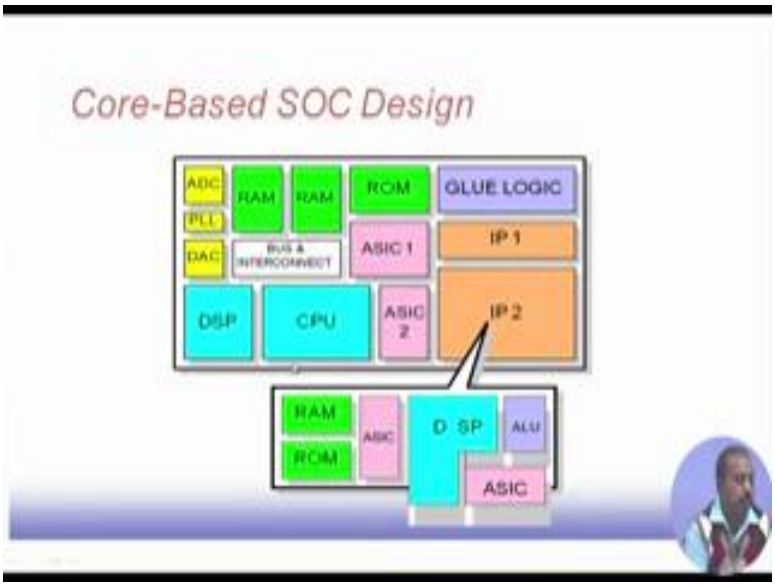
(Refer Slide Time: 07:42)



So, there is a family. So, this is the thing that I was talking about 1149.1. So, this is for digital chips and interconnects among chips. So, standard is 1149.1 in 2001. So, 49.2, 49.3, they got discontinued because they are for digital extended digital serial interface and direct access testability interfaces, but they were discontinued. Then this mixed signal ICs they came in a big way as a result this 1149.4, so this came for this mixed signal test buses. Then 1149.5, so this is for module test and maintenance standard module test and maintenance. So, this has came up in 1995, but from 2003, so this is no more valid. And ultimately, this one has come up 1149.6 which is now an accepted one which actually combines many of these. So, this high-speed network interface, and it can

also apart from high-speed, so it also takes care of this analog part. So, the previously whatever we have it has been discontinued, so that can be taken care of by this one 1149.6.

(Refer Slide Time: 08:53)



What is a core-based design? So, suppose it is like this. Suppose, I want to design a system, so where we have got all these components a CPU, DSP, some ADC, PLL, DAC, RAM, ROM some more intellectual property designs IP 1 and IP 2. Now, if we want to design such a system, so forget about IP 1, IP 2. Suppose, we take a very simple case DSP CPU and this memory chips, so they are there. Now, if you want to design such a system, so what we have to do is that we have to take different components from different vendors and mount them on a PCB and do connection between them.

So, the disadvantage is that this CPU to memory communication, so that is going to be an off chip communication. So, and we know that off chip communications are much, much slower compare to on chip communication. So, this off chip communication being slow one possible solution for this is to have this CPU and memory on the same chip, but that way so one integrator one system designer's requirement is different from another designer, so as a result there cannot be any unified system that way. So, what is required is that as a designer, I should be able to take different components from different vendors

and integrate them into my chip. Now, when we do this thing, so I have to go to say for example, Intel to get the Pentium design, I have to go to say TI for getting the DSP design, so like that, but none of them can give me the design because that is what they are doing. So, that is they cannot give it like that.

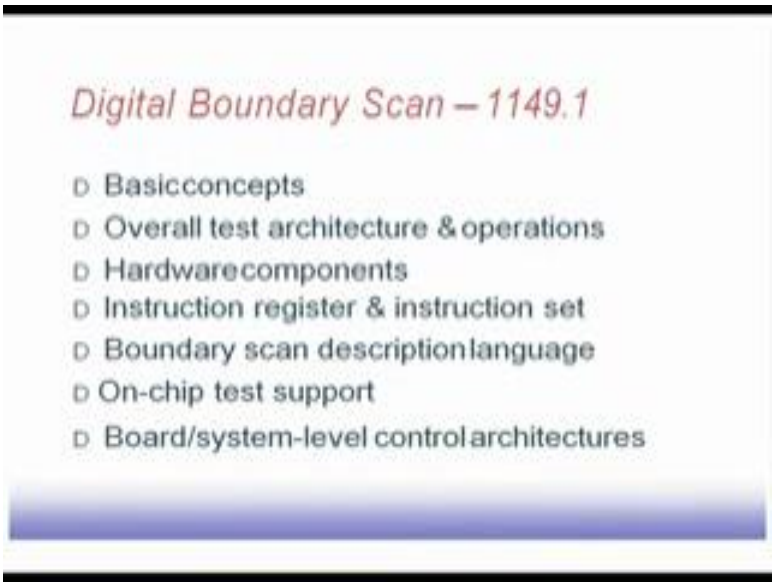
So, what they give is nothing but a layout level description of the whole structure, so and that layout level description if manufacturer properly fabricated properly that will give me the system. So, if I have that layout level description, so of this say CPU, DSP, RAM etcetera and synthesize this whole thing together then all the off chip connections between say CPU and RAM, they become on chip connection, as a result the speed of the system will be better the performance - the system performance will be better. So, this is actually called the core-based design. So, this layout level description that I was talking about, so they are called cores. So, these cores are taken from the vendors and this they are actually integrated by the system integrator.

So, apart from this say CPU, DSP, RAM, ROM, so you may need some extra logic which is not covered by any of them, so some extra glue logic. So, this is normal commonly known as glue logic, so that is just as if to connect all these components this extra logic is necessary. There will be some interconnection between these components, so that maybe via bus or that may be via some other interconnect mechanism. So, whatever it is some bus and interconnect mechanism should be there; some if it is a system talking to the analog environment then there must be some ADC and DAC components in it, so that is also there. Then we have some special ASIC may be necessary that may be known to the designer only. So, the ASIC are their descriptions are taken. And it may so happen that this IP which stands for intellectual property, so intellectual property core one intellectual property core 2 they themselves maybe another SOC, so they themselves may have more number of chips in them more number of layouts in them which will make that.

So, as a designer I put all these things into my manufacturing process, and get this whole thing done on a single chip. So, this is called a system on chip based design. Now, when you go for this system on chip based design, so they are all associated test challenges, so

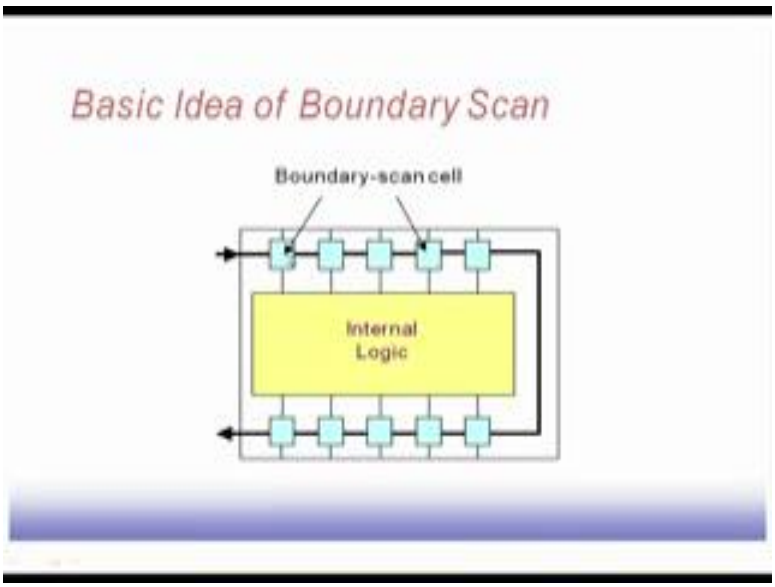
that will see as you proceed through the course.

(Refer Slide Time: 12:55)



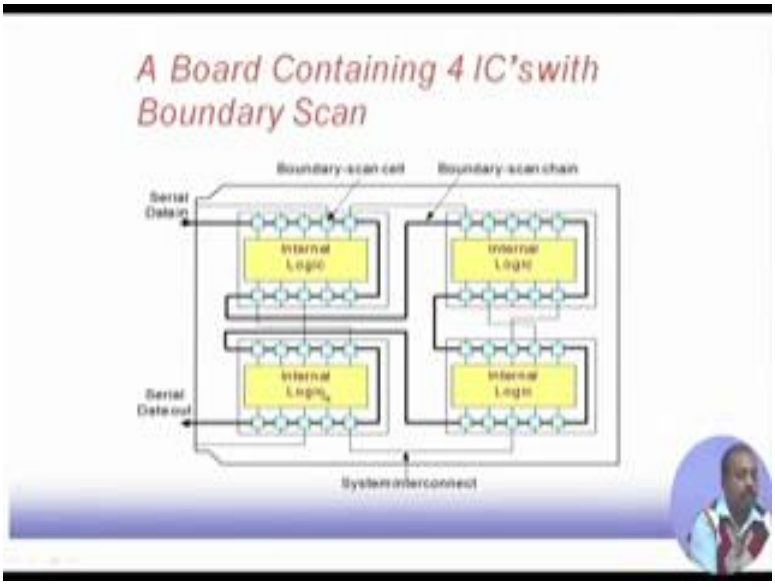
To start with, we look into the basic digital boundary scan policy 1149.1, so we will look into the overall test structure hardware components, instruction register instruction set description, scan description language then the test support and the control architectures.

(Refer Slide Time: 13:15)



So, this is basically the boundary cell boundary scan structure. So, this is my internal logic. So, one chip may have this as the internal logic. So, if it supports boundary scan then all its inputs, so they will be available on this scan cells and all its outputs will also be available on scan cells. So, there will be 2 separate inputs apart from these functional inputs for this circuit, which is called this scan input and this scan output. So, all the internal lines, all the internal lines in the sense that all the primary inputs and the scan inputs maybe inside there is a scan chain here, so there is a scan chain here, so that scan chain can also be accessible via this scan cells. So, that way this boundary scan chain will be made by having these boundaries scan cells connected over a chain.

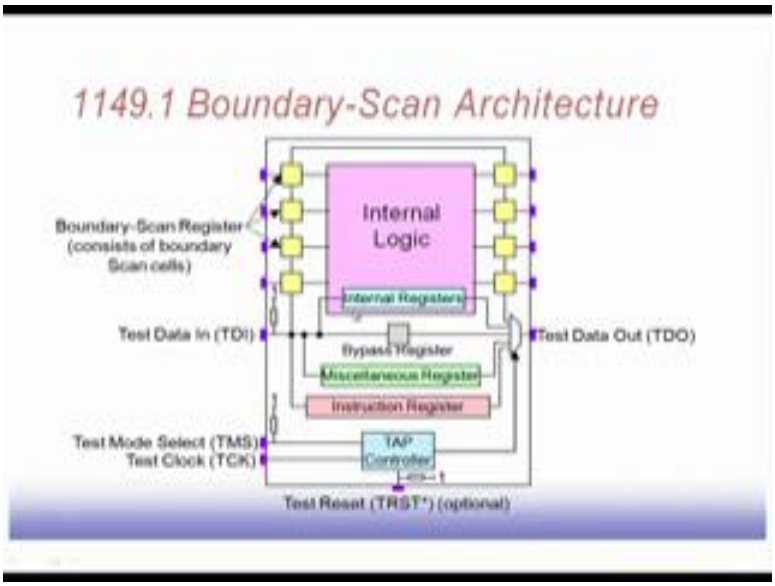
(Refer Slide Time: 14:10)



So, this source I board structure that has got 4 ICs. So, these are the 4 ICs that we have. And each IC is equipped with this boundary scan feature. So, all these inputs are connected over a scan chain like this. So, scan chain of this boundary scan cells now it is connected like this. Now, output from this is again going to the next chips the boundary scan input again that way it goes. So, it traverses, so this particular chain it traverses the entire board, so that way apparently it appears that it is very costly to reach individual chips, but still it is much better than going by the logic of the circuit. And most of the cases, the logic of the circuit is not known, because we are buying the chips from the vendors. So, we do not know the internal logic.

So, what should be the value given at a particular point at a chip level, so that I can get a signal propagated through the chip, so it is very difficult to tell. In logic level, this was possible because the circuit description was known to us the gate level description was known to us. So, it could use some sort of justification routine, but here I do not know the internal details of this individual chips, so I cannot to do any sort of logic justification there. So, the only way out is to have some additional chain made like this, so that all these inputs can be accessed.

(Refer Slide Time: 15:43)



So, how does it look like, so say this is the internal logic that we have? Now, there may be scanned scan chains inside. So, these are the internal registers, so all of them are accessible from this thing. Now, so these are the boundary scan cells and this whole thing constitutes the boundary scan registers. So, all this boundary scan cells put into a chain, so that will constitute a boundary scan register. So, there is a serial test data input pin and there is a serial test output pin. So, if this test data input pin is kept floating then it is tied high; similarly there are some other points where it is tied high.

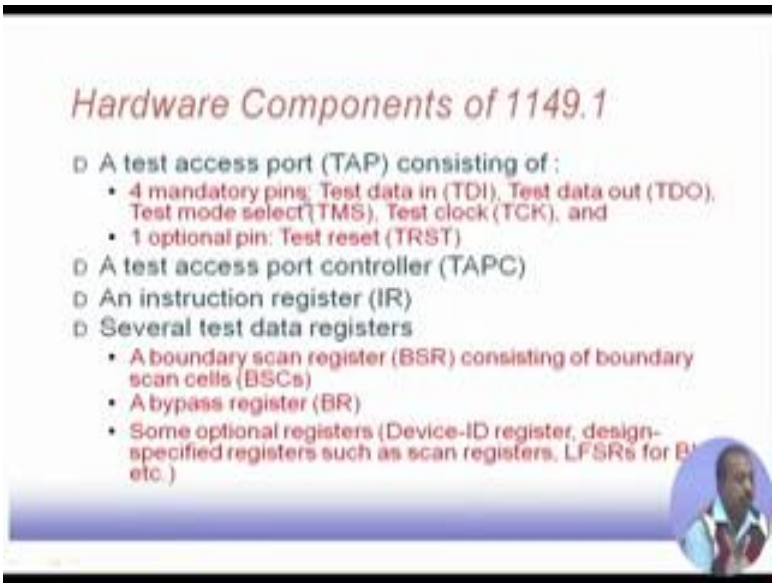
Now, this test data inputs, so it can go to various places like it can go to this chain of this scan cells it can go to this instruction register or there are some other registers miscellaneous registers that will see, and also there is a bypass registers. Similarly, this

test data output can get input from the internal registers, it can get input from this boundary scan chain, from this bypass register, from this miscellaneous registers or from this instruction registers. And there is a test access port control, so controller, so which is the TAP controller - test access port controller, so which will be giving commands to this test data output to select one of these inputs.

Now there is a test mode select test mode select signal, so that will us that will tell the TAP controller like what to do with the next input that is coming. So, whether it should go to the instruction register or some other miscellaneous register or how this test data out will be controlled so all of them, so they are obtained from this test mode select line this that will tell the TAP controller what to do. And then there is a test clock, so that is the clock at which this whole testing process will operate and there is an option on the reset signal. So, this test reset signal that is optional. So, if you make this high then the entire operation will be reset.

So, we will see how what is the meaning of this reset. So, essentially what we are having, so the apart from this internal logic that the chip will realize, so it needs to realize all these extra components to make the chip, this boundary scan compatible. So, if a chip follows this particular standard 1149.1 that means, it has got all these components in it; and also its step controller works in a particular way that we will see, so based on the command, so it will work in a particular way.

(Refer Slide Time: 18:40)



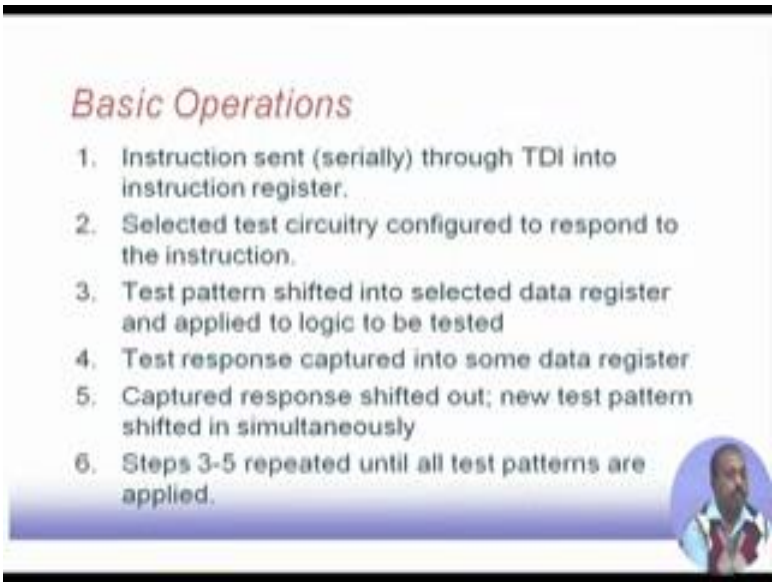
So, the hardware components of 1149.1, it consist of one test access port. So, this is the test access port that we have. So, test access port has got a number of signals. This 4 pins are mandatory test data input, test data output, test mode signal and test clock. So, these are the test data input, test data output, mode select and respect clock, so these are the 4 pins that must be there in any 1149.1 boundary scan. Then there is an optional pin, which is test reset that is optional. Now, one test access port controller TAPC like in this case we have got this is the TPAC the test port where test access port controller and this is the reset signal.

Then there is an instruction register. So, there are certain types of instructions that you can give the; you can tell the test access port controller to do. So, this is actually that instruction register, it can hold different types of instructions. And we have got several data registers like this boundary scan register, so this boundary scan register, so these boundary scan cells they will put, they will go in a chain and that forms the boundary scan register. Then we have got a bypass register; so this one is the bypass register, which can be used to send the test data input directly to test data output without affecting the chain that we have here.

So, if certain information is not meant for this particular chip it is for the next one or next

to next one, so all those intermediary chips, so TAP controller can be told that ok, it should configure the system in a bypass mode. So, then this TAP controller will select this bypass register and whatever is coming as in the TDI line will be same to the TDO line. So, that way it can make the operation faster, the transfer of data through the chain faster. Then this there are some other registers like device-ID register, design-specified registers as scanned registers, LFSR for BIST, etcetera. So, basically the other miscellaneous register that we are talking about, so these actually include all these things. The instruction register is there then the device-id register, so then if we are trying to support BIST, so there may be some BIST register, so all these are part of those special registers.

(Refer Slide Time: 21:25)

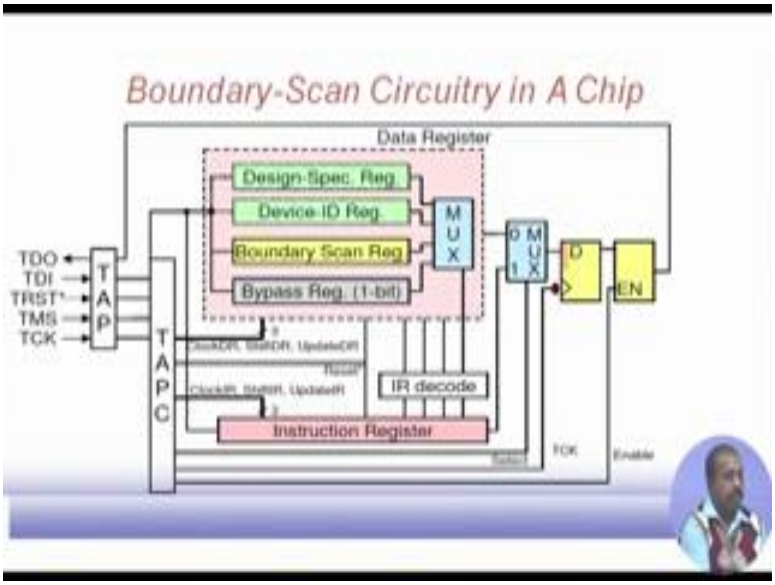


Now, how this whole thing operates; first the instruction is sent serially through the TDI to the instruction register. So, the test data input line it goes to the instruction register. And selected test circuitry configures to respond to the instructions. So, first this instruction is coming from this TDI, and it is reaching this instruction register. So, once it is there in a instruction register, so it will configure rest of the components to respond to the instruction, so this is what is done the selected test circuitry is configured to respond to the instruction.

Test pattern shifted into the selected data register and applied to logic to be tested. So, if we see that the command is to shift in some test pattern through the boundary scan cells into the boundary scan register, then it can be doing some shift operation over a number of cycles. So, if the instruction is to do a start a BIST operation, then it will start a BIST operation, it may be to capture some response, so that way it will be capturing the response. So, there are various instructions. So, this instruction register content is decoded and based on that the other controls are set, so that these operations are done.

And test response will be captured into some data register. And capture response shifted out new test pattern shifted in simultaneously. So, this is true for any scan chain configuration that the previous response and the current pattern, so they are shifted parallel. Now, this is steps number 3 and 5 are repeated until all test patterns have been applied. So, this is the way this whole thing works. So, what we need to do is somehow set this test data the instruction register with the proper values, so that it can be made to it can it can command the other components to work accordingly.

(Refer Slide Time: 23:18)

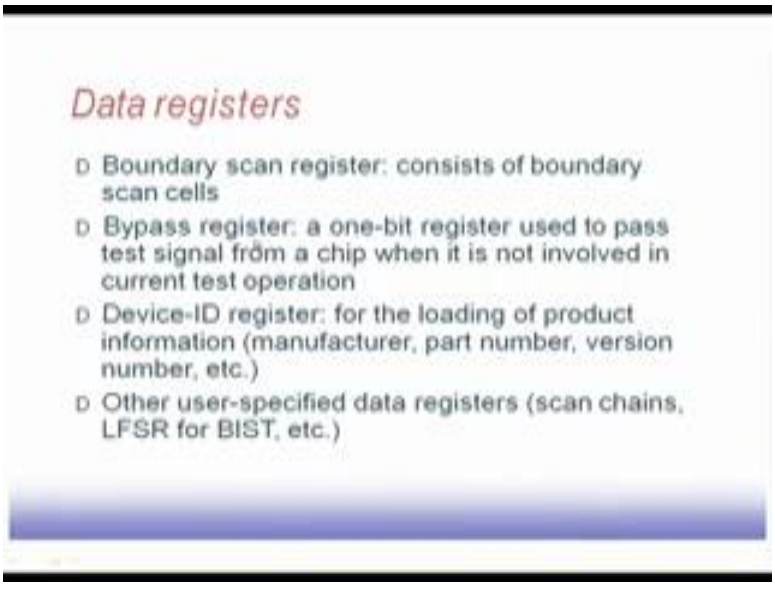


So, this is the boundary scan circuitry of a chip. So, the how also it is a more detailed structure. So, you see what has happened is from this in the test access port, we have got this line - this TDI, TDI test data input, TDO test data output, then this TMS test mode

select and the test clock, and this is an optional reset signal TRST. Then that goes to the this tap controller and also from the tap controller, so it gives the clock. So, we have got this registers like this boundary scan register, device-ID register, and design-specific register. So, bypass register.

So, they are there now there are some signals like clock to the data register - D register shift to the D register, update data register, so these are some signals. So, we will see this as we proceed. So, there are some cells some registers in that will see. And this instruction that is coming, so that there is an instruction register decoder, so which will be decoded accordingly it will control these all these data register and this multiplexers. So, that the value will be shifted out to the next stage by means of either this from this multiplexer coming from this multiplexer or it may be from the instruction register. So, this way this value is passed. And then this is another register where this value may be directly coming from this TAPC or it may be coming from this d flip-flop.

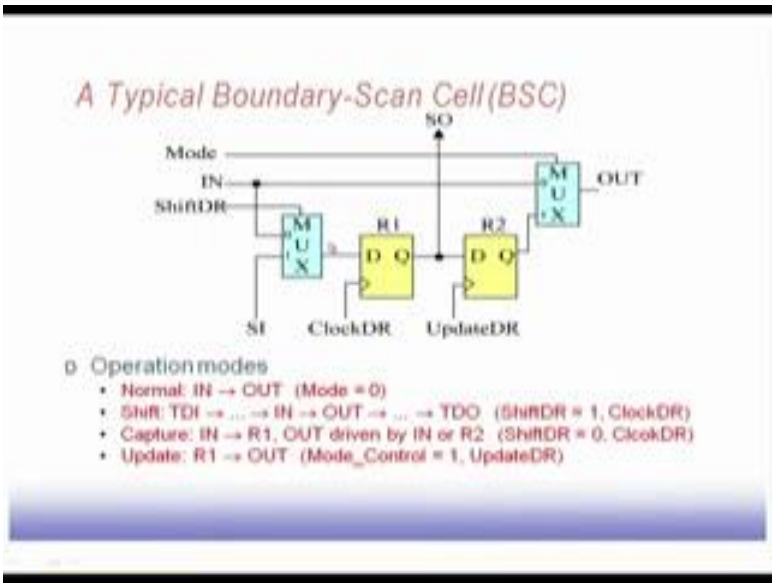
(Refer Slide Time: 24:56)



So, we will start with the boundary scan register it consists of the boundary scans cells then the bypass register, so this is you know one bit register used to pass test signal from a chip when it is not involved in the current test operation. So, this is basically bypassing the entire chain for a particular chip. Then there is a device-id register. So, for the loading

of product information like manufacturer, part number, version number etcetera, so there is a device id register. So, later on, we can send some command to check whether this device id matches with the device id for which we are sending the pattern and accordingly several such test patterns can be applied conditionally. And other the user specified data register is scan chains, LFSR, BIST etcetera, so they are also forming some data registers.

(Refer Slide Time: 25:44)



So, this is the structure of a boundary scan cell. So, a single cell of the boundary scans register. So, you see that this has got 2 registers in it R 1 and R 2. So, this R 1 register this is basically for storing the input pattern. So, whatever is so if you are trying to capture the response then what will have to do is that from let us start with the normal mode of operation, so normal mode of operation means from the input, the result should be out available at output. So, it should not go into this boundary scan cells. So, in that case it is mode is equal to 0. So, this multiplexer will be selected. So, mode equal to 0, so this is a selected, so this input goes to the output.

Then the shift operation that is whatever is coming on the test data input line, which is reaching here by this serial input line, so that will be going to the serial input line, so that will be going to this register R 1. So, TDI, so that will go to into output TDO, so that is

test data. So, basically it is coming from this shift data register. So, it will be coming from if this bit is say 1, then the bit will be coming from this shift input; if the bit input is 0 then the input will be coming from this on the circuit functional input in. So, whatever is happening, so this value is getting transferred on to this one say clock DR. So, shift DR equal to 1 and clock DR equal to 1, so this will put this SI the serial input into this register R 1.

Then in capture mode shift DR is equal to 0, and clock DR equal to 1. So, shift DR equal to 0 means this input line whatever is coming, so that will be put into this multiplexer that will be coming to this register R 1. So, essentially in shift mode, the test pattern is being shifted via this line; and in the capture mode whatever be the circuit outputs that is being captured via this input line. So, from in it is coming to this multiplexer to this R 1, so that is the shift and capture mode. So, one is mode 0, then mode one is shift mode, mode 2 is the capture mode. Then in the update mode, this content of this R 1 will be available at the output.

So, how this thing happens, so this mode control is equal to 1, and this update DR is given. So, previously there was so update will be given after capture only. So, after capture this R 1 is containing the value, so that will be when this update DR is given this, this value will be will come to this R 2 as well, so this value will be stored in R 2 and then this mode is selecting one. So, this is available at the output. So, this that way this, this R 2 will be updated with the value of R 1 and this will be output at the this thing also.

So, you see that in shift mode, so we are shifting in that test pattern and simultaneously the content of this D flip-flop will be available at the scan out. The next input comes here get last here previous input goes to the scan out through the scan out to the next cell in the sequence. In the capture mode, whatever is coming on the inputs, so it is getting captured on to this flip-flop R 1. Then after that if there is an update operation, then this update operation will put it the value into R 2 and that will be available on the outline. So, this is actually the way this boundary scan cells operate.