VLSI Physical Design with Timing Analysis

Dr. Bishnu Prasad Das

Department of Electronics and Communication Engineering

Indian Institute of Technology, Roorkee

Week - 02

Lecture 07

Overview of Timing Analysis

Welcome to the course on VLSI Physical Design with Timing Analysis. In this lecture, we will do an overview of timing analysis and its application in VLSI physical design. So, the content of this lecture includes the timing analysis, how it is, what is the meaning of timing analysis, what is the significance of static timing analysis, and we will see what is the use and where it is used inside a VLSI physical design flow. Then there are two types of timing analysis: one is called static timing analysis; one is called dynamic timing analysis. Then we have different types of path in a basically in our regular circuits. So, there are different types of path, for example, false path, critical path, and short path. We will discuss all of them with one examples. So, basically, what is timing analysis? So, timing is one of the most important parameter of any VLSI physical design or any VLSI design. In particular, VLSI physical design is intended for digital flow, but any VLSI design it may be on digital or analog, or mixed-signal, always time plays a very crucial role. We want to optimize the speed, or we want to optimize the delay of the circuits to get best performance out of it.

Let us say if we go to a market and we want to buy a smartphone. So, we want to look for which is running very fast, which has a very good processor inside it. Let us say we go to buy some laptop in the market, we will look for how many cores are there in the processor. What is the speed of the cores? So, the speed is one of the crucial factor which governs the customers. So, that the customers looks for speed. So, as a designer we also look for how to maximize the speed of our design. So, it is possible there is a method to analyze it. We need to have a method to analyze the speed of our circuits and that should be done at very fast manner. So, we have basically analysis the timing of the circuits and optimize the speed of the design. So, the timing parameters of the circuits includes setup and hold time, clock to queue delay and we have rise propagation delay, fall propagation delay, all these into take into account to find out and find out those paths and optimize those paths.

So, what is the significance of timing analysis? Whenever you are doing timing analysis or static timing analysis, what is the use of that? It basically determines the longest path of our design. So, whenever we talk about the speed that will be determined by the longest path in our design. The longest path in the design is called the critical path. So, critical path of our design. So, usually, people think that whenever we are looking for the speed, we only look for the longest path of the design, and it will reflect the speed of our design. But if we do only that thing, that is not enough for a chip design because sometimes what happens there are short paths in the design, which will basically very important in terms of design point of view. Let us say we only analyze the critical path or the longest path in the design and we found out the speed of the design. And at the same time we do not did any analysis for the short path. So, it is not good practice. So, we have to do analysis on the critical path or the longest path.

We have to do analysis for the short path or the minimum path. The longest path is done to basically to avoid setup violation and the shortest path is determined to avoid hold violation. So, ideally the clock reaches to each of the flip flops at time t equals to 0. But in general, whenever you route the clock, there is some kind of delay between them. So, that results in a clock skew. That results in a clock skew. Let us say I have a flip-flop here; I have another flip-flop here; both are connected by an edge. So, ideally, they reach at t equals to 0. But in reality, they do not reach at time t equals to 0 because we have some interconnect length and also we have introduced some buffers in between. We have introduced some buffers.

Why the buffers are needed? The buffers are needed to improve the transition time of the clock signal. So, the clock will rise sharply at the rise time. So, what is happening is that this buffer will add skew between the arrival time of the clock at node 1 and node 2. So, that results into clock skew. What is clock jitter? Because of some random phenomena inside the devices and inherent noises and substrate noise, supply noise, all these things, this clock arrival time may vary. So, that will also result in a clock jitter. These are non-ideal parameters which also impact your design, design timing. So, we also need to do the timing analysis considering the clock skew and clock jitter. So, there are another angle to the timing analysis that is called process variation. So, we have multiple corners there. We will discuss that in future slides, but those corners need to be analyzed properly in order to do the accurate timing analysis.

So, we need to consider the process variations and their corners into account while doing the timing analysis for an accurate prediction of the timing. So, this is the VLSI physical design flow, but in order to do the optimal timing closure at the end of the physical design, we need to check the timing of the design at each step of your physical design flow. So, after partitioning, after chip planning, every step, every step after partitioning again we have to check the run the STI tool whether we are able to meet the timing or not. After chip planning, again we need to run the STI tool to check that whether timing

is met or not. After the placement of the blocks, again we need to check whether our required specification given to us has met or not. And whenever we are going to the clock tree synthesis, because in the initial logic synthesis phase we assume that our clock tree is ideal, but whenever we are doing clock tree synthesis there will be a lot of buffer insertion and there will be lot of routing resources will be added in the clock tree, then the clock skew and clock jitter need to be considered. So, that will add to our timing. So, after the clock tree synthesis, we have to check the timing, and if it is violated again, you go back and fix the timing. And after the signal routing also we need to check the timing and check whether if it is meeting or not in case it is not meeting and again we go back and fix the timing in the we need to change the routing style. So, every phase of your physical design, we need to check the timing whether it is met or not; if it is not met, then we will go back and do the modification and try to fix the timing.

Then, only at the last step we will have to do less effort. Otherwise, what will happen is that it will not be possible to meet the timing closure at the end of the VLSI physical design cycle. The timing analysis is basically divided into two categories: one is called static timing analysis and the second one is called the dynamic timing analysis. So, the static timing analysis is basically is the timing analysis methodology at which we do the timing analysis of the circuit without applying any input vectors or monitoring the output of the without applying any input vectors and also we are not monitoring the output vectors. So, this was static, static means that no input is supplied to the timing analysis tool. So, what is the process of static timing analysis? We break the design into timing paths. Calculate the signal propagation delay along each path. So, we have multiple paths we break down into smaller paths and find a timing propagation of each path. Then we check that actual timing constraint is satisfied or not. If there is any kind of violation inside the design or input output interface if it is not met then again we need to modify the design and do the timing analysis again and again till we made the timing.

Let us say I design a processor, let us say the processor whatever I am designing the speed of the design is basically for example, is fixed at 1 gigahertz and the corresponding the frequency is 1 gigahertz if clock is 1 gigahertz. The corresponding time period t clock is 1 nanosecond. So, if my 1 nanosecond clock is not met that 1 nanosecond is supplied as a constraint in the timing analysis tool. This 1 nanosecond is supplied as a constraint in the timing analysis tool. Then the tool will evaluate the actual critical path in the design and check that how much slag is there at the end. If the slag is positive, then my timing is met; if the slag is negative, my timing is not met. Then again we need to modify the design to meet the timing. So, this is a methodology was followed in case of static timing analysis. What is dynamic timing analysis? It verifies the functionality of the design by supplying the input vectors and also it checks for the expected output at the output nodes. And it also checks for whether the design is logically correct or not, whether it is functionally working or not all these things is verified in case of dynamic timing analysis.

But it requires basically the simulation time required for dynamic timing analysis is more than static timing analysis as all the inputs need to be checked. So, basically it is suitable for designs wherever we have critical components like the clocks crossing the multiple domains. So, where we are critical points are there, there we need to follow the dynamic timing analysis. Otherwise static timing analysis, which is fast to determine the critical path that is basically employed in regular practice, but in critical points we go for dynamic timing analysis. So, let us look into this total design. You have inputs, we have clock, we have final output. In between, I have flip-flops. This is flip flop 1, this is flip flop 2. So, if you can see here, I have multiple paths in this design. What are the paths there? This is one path from input to the D pin of the flip flop 1. This is path 1. From input to the final output, so this is path 2. So from here till here this is the path 2 through the combinational logic. Then I have one more path between the flip flop 1 to the D pin of the flip flop 2. This is another path 3. Then you have flip flop 2 to the output, that is the path 4.

So, there are several paths inside an actual circuit. So, this is an example of different paths inside an actual circuit. So, there is also one path called clock path. So, this clock path is shown in a green line. So these are different paths inside the timing paths inside this particular example. So, if you can look into this one, we have a circuit given in the left-hand side and right-hand side is basically a graph. What type of graph? We have already discussed several graphs are there. This is basically called a directed acyclic graph or DAG. What is this DAG? Each node to the other node there is some direction is there and that is why it is called directed graph. From one each node to the other node there is a direction is there that is why it is called a directed graph. And there is no cycle there inside the graph that is why it is called acyclic graph. So this is called directed acyclic graph or DAG. So, if you can see here, we have a source node and each of the inputs primary inputs these are the primary inputs. I1 to I5 are the primary inputs. O1 and O2 are the primary output. These are the primary output. And this is the directed acyclic graph corresponding to these circuits. So these circuits if you can see all the primary inputs and primary outputs are denoted by the vertices or the nodes and the connection between them are represented are as edges.

So whenever we talk about the timing path it has multiple timing path inside this design. From, let us say, from the source node, I can go like this, like this, like this. This is my one path. So there are multiple paths are there I am giving you few examples of them. So I can go by this I2, then G3, G2, G3, G5, and this one. So there are multiple paths inside this design. So all the paths need to be analyzed by the STA tool in order to find the worst case critical path inside the design. So I have several paths are the, critical path and the short path.

Critical path is basically the path between the input and output with the maximum delay. The path between an input to the output with the minimum delay, the short path. But

whatever I told you, the critical path basically tells about the speed of a design, and the short path is also essential to be analyzed by the timing analysis tool to avoid hold violation. So, there is another path called false path. So, we discussed about the critical path, short path.

Now I have a false path. The false path is basically a path that exists in a design which is not functional at any point of time. But if you do not consider that path, it may be shows as a critical path and it will basically give you an idea that your critical path is this false path is treated as a critical path. However, that path will never be activated at any point of time. So, that path will never be activated at any point of the time. It is not required to meet the timing constraint in this path and because this design that path will not be activated at any point of the time. So, we need to find out those path in the design and tell the tools that this path should not be treated as critical path and tool will mark those path as the false path. So, let us take an example here. So, this circuit is given to you in the right hand side. We have bunch of gates here, NOR gate, inverter, and multiplexer. So, the delay of the NOR gate is 4 nanosecond, delay of the multiplexer is 3, NOR gate is 1 nanosecond.

So, in this design, we will find out the critical path, shortest path, and the false path in the given circuit. So, how do we do that? We will see what are the paths exist here. I have path 1, this is the path 1, this is the path 1 which is having a NOR gate and a MUX delay. So, this is P1. Then P2 is a NOR gate, multiplexer, again NOR gate and a multiplexer. So, this is P2. Then I have a path 3, NOR gate, multiplexer and multiplexer, this path. This is P3. Now I have P4 is a NOT gate, multiplexer and NOR gate like this. This is P4. Then P5, we have a NOT gate and one multiplexer and another multiplexer. So, this one is your P5. So, I have 5 different paths in this design. Which will be the critical, which will be the false, and which is the shortest path. So, now, what we did? We found out the delay of each of the path. The path 1 is 7 nanosecond, path 2 is 14 nanosecond, path 3 is 10 nanosecond, then path 4 is 11 nanosecond, path 5 is 7 nanosecond. Which is the largest among them? 14 is the largest. But so, these are the path and its corresponding delay, 14 is the largest. Now, we need to find out which is the critical path, which is the shortest path, and which is the false path in this design. So, if you can see here, if you look these circuits, which is the longest path here, it is the 14. But let us go into depth and analyze the circuit properly.

If let us say case 1, t equals to 0, and this will be 1. So, this is 0 means this path will be selected. And this is 1 means this is the path will be selected. So, if this is the case, then what is the delay of this one? Delay is NOR gate, NOR gate delay is 4, multiplexer is 3 and another multiplexer is 3. So, it is 10 nanosecond. So, this is 4 nanosecond, this is 3 nanosecond and this is 3 nanosecond. Total path delay is basically 10 nanosecond. Now in case of case 2, when t equals to 1, t equals to, I will write it in a different color. This is 1, this is 0, this is 1 means this path will be selected and this path will be selected like

this. This will come out like this. So, the delay of inverter is 1, this is 1, 1 nanosecond. Now the multiplexer is 3 nanosecond. Now the NOR gate here it is 4 nanosecond. Then you have this multiplexer is also 3 nanosecond. This is 3 nanosecond. So, 3,3,6 plus 4 plus 1 it is 11 nanosecond. So, out of these two, 11 is your critical path. But if I go back to, 11 is the actual critical path in the design. But in the previous case, whenever we discussed that all the paths, the 14 which is basically selecting both 0, basically this path I will do it in a different color. So, this path is a false path. Why this is a false path? Because these both cannot be 0 at the same time because of this inverter. And this path will never be excited or functional during the design of the circuits. So, during the design of the circuit, this path cannot be activated because whenever S0, is 0, this will be 1.

Whenever S0, is 1, this is 0 and both cannot be 0 at the same time. So, in that case, this path is called a false path. So, since it is a false path, we need to tell the tools that we should not consider this path as a critical path, so that it will not take into account while doing the critical path analysis. Then which is the shortest path here? We have this P1, if I go to the previous slide, this P1 and P5. So, the P1, this path is shortest path, and this path is the shortest path. So, both the paths are shortest in the design. So, both the things we need to be considered whenever you are doing the hold violation. Hold violation is not there because here we have not put any flip-flop here. But if we consider this circuit as a combinational circuit, let us say I have a flip-flop here, flip-flop 1, and I have this combinational design here, then I have a flip-flop 2, then this circuits as a whole is representing these combinational circuits, then the short path for this design is 7 nanoseconds. Short path for this design is 7 nanoseconds, which is used for checking the hold violation.

In this lecture, we discussed about the importance of timing in VLSI design, and we can analyze the circuits using static timing analysis. And the static timing analysis is essential at each step of your VLSI physical design cycle. And one more thing is that we need to check the timing at each step of our VLSI physical design in order to meet the timing at the end of the VLSI physical design. And finally, there are two types of timing analysis is there, one is called static timing analysis, one is called dynamic timing analysis. Static timing analysis is faster, dynamic timing analysis is slower, but whenever there is a critical nets like clock domain crossing, multiple clock domain crossing during that time dynamic timing analysis is useful. And we discussed about three timing paths, there are three important timing paths like critical path, short path, and false path with one example.

Thank you for your attention.