VLSI Physical Design with Timing Analysis

Dr. Bishnu Prasad Das

Department of Electronics and Communication Engineering

Indian Institute of Technology, Roorkee

Week - 02

Lecture 08

Timing Arcs and Unateness

Welcome to the course on VLSI Physical Design with Timing Analysis. In this lecture, we will discuss about Timing Arcs and unateness. So, the content of this lecture includes definition of the timing arcs, then we will discuss about the what is source pin and a sink pin in a timing arc, then there are different types of unateness like positive unateness, negative unateness and non-unateness types of gates and what is the use of unateness in a any kind of digital circuits and how it is used for timing analysis. So, first of all we will discuss about the timing arc. So, what is this timing arc? So, it is basically defined as the relationship between two pins of a logic gate. It can be this logic gate can be a combinational or a sequential. So, this timing arc is also useful it is useful for it is useful for the STA tool to do timing analysis. And we have different types of timing arc. So, it is divided into two parts. So, it is basically a part of the timing path. It is divided into basically two categories, two types. One is called your cell arc and the second one is called the net arc. So, we will discuss what is a cell arc and a net arc with an example. So let us say I have a path; here, if you can see, let us say it is a NOR gate, then we have an AND gate and a buffer. So, it is a mixture of gates are there.

Then again you have a XNOR gate like this. So, which one is your cell arc, and which one is your net arc? So, these gates we can use different colour. This logic gates are basically your cell arc. So, because this cell or logic gates are part of the timing path and this is these are called the cell arcs and we have net, net means interconnect or a wire. So, these I can draw it in a different colour. So, these are my net arc. So, we need to consider both the cell arcs and the net arcs in the timing path. So, this is a timing path from input to the output we have a timing path. So, it consists of some cell arc and some net arc. So, this is the basic of your timing arc.

Now we will go into what is source pin and the sink pin. So, if you can see here one example I will show you then I will explain what is source pin and sink pin. Let us say I have a gate here. This is a NOR gate followed by a NAND gate or you take a AND gate. So let us say this is A pin, this is B pin, this is C pin.

This is let us say the gate 1, this is gate 2 and this is your output. There are two outputs let us say. This is O1, this is O2. So, here, if I can see you, basically your gates. So, what are the source pin? So, for gate G1 your B and C are the source pin and for and your O2 is the sink pin. So, source then the sink. Now, for gate 2, your basically A and O2 are the source pin, and O1 is the sink pin. So, this is the concept of source pin and sink pin. So, now we will discuss a few more examples. In case of a, I will generalize few things.

In case of shell arcs, your input pins are source pin. And output pins are sink pin. But in case of a net arc, the output pin of the, so input pins of the cell, input pins of cell, here input pins of cell, output pins of cell. So, here basically output pin of the cell are source pin and input pin of the cells are sink pin. For example, if you can see here this is a net arc. So the output pin is the, so this is a net arc, and the output pin is your source pin, and the input pin is a sink pin. This is the input, so this is a sink pin, and this is the source pin for the net arc. This is the source pin for net arc, this is the sink pin for net arc. Similarly, these are your source pin for the cell arc, these are the source pin of, these are the source pin of cell arc, and this is the sink pin of the cell arc. So, this is the concept behind the source pin and the sink pin.

Now we will go into the details about unateness. What is unateness? What is unateness? So this unateness is one of the important parameter for timing analysis and it is used by the STA tool and we will explain this with some analysis. So basically, each timing arc has a timing sense, actually. Each timing arc has a timing sense. So, each timing arc has a timing sense. What is the timing sense is that? It is denoted by unateness. So, what is this timing sense is that, what it says that change in the output transition with respect to input transition that is called timing sense. So, this is used in case of static timing analysis. So, if you know how your input and output transitions are related then the timing analysis tool will pick the right delay value from the dot leaves. We will discuss that in detail.

So now we have three types of unateness, one is called positive unate. What is this positive unate? So if your input signal is rising then your output signal is also rising is rising or no change. Or you can say the input signal is falling and your output signal is falling or no change. So, we will discuss this one with one example. Let us take a case of a buffer. So, we have a buffer. I have this, this is the buffer. So, how many timing arc is there? You have seen here if your input is rising here, output will also do a rise transition in case of a buffer. So, I have, this is one arc. So, this is first arc, this is the second arc. If your input is falling, then your output is also fall after a certain delay. So, that is why it is called positive unate. So, input is rising, output is rising. Input is falling, output is falling. So, let us take a case of a AND gate. Let us take a second example AND gate.

It is interesting, why? Because it has multiple timing arcs. For example, I have an AND gate. It has two pins, A and B, the two inputs AND gate, and your Y is your output. So, let us say your B is 1, B is fixed at 1. So, input is rising, then your output will also rise. So,

this is one arc. So, this is one arc. Then the second one is, let us say my input is falling, B equal to 1, my output will fall. So, this is the second arc. Then the third arc is basically, I have, let us say, A equal to 1, and B is rising, your output will also rise. Then the fourth arc is, A is 1, my input is falling, my output will also fall. This is the fourth arc. So all the cases, your input is rising, output is rising. Input is falling, output is falling. So then there is another, these four arcs is actually used, and there is another arc is also there, but it is not used. We will explain why it is not used. Let us say one of the inputs is 0; let us say B is 0. So if I do a rise transition here, your output will not change. So here there was a condition that either rising or no change, output is not changed. But it is not an arc. Why it is not an arc? Because if I put this gate with B equal to 0 in a path, then the signal will not propagate from input to the output in that path. So, it will stop there, it will stop there, the signal will stop there. That is why this is not an arc. So, your AND gate has four timing arc. Similarly, you can do it for A equal to 0 also. Those are not arc because your signal will not further proceed in the path. So, your AND gate has four timing arc. Now we will discuss about the positive unate. Now we will discuss about the negative unate. So what is negative unate? Is that here the opposite. So, what is the opposite is that, let us say your input signal is rising. So, your output signal is falling or no change. Your output signal is falling or no change.

Similarly, you can say input signal is output signal is falling means that output signal is rising or no change. So here, so let us say take an example of an inverter. One example I will take inverter which is having two arc. Let the first arc is, let us say I have a inverter here. Let us say I will do a rise transition in the input, output will fall. So this is a timing arc. Similarly, the inverter, it will fall then output will rise. So inverter is negative unate because input is rising, then the output is falling, and input is falling, then the output is rising, and it has two timing arcs. It has two timing arcs for a inverter. Now take an example of a NAND gate ok.

Let us do it in the next slide. So, two input NAND gate. So, what is happening? Let us say I have a NAND gate is there. If my B input is 1, let us say my other input is rising, your output y, this is A, output y will fall. Similarly, if I have another case B equal to 1 and my input A is falling, then your y will do a rise transition. So, this is the second arc. Similarly, let us say your A is 1 and B pin is doing a rise transition, your output will do a fall transition. Your A is 1 and B is doing a fall transition, then your output y will do a rise transition. So, these are the valid arcs. You have four arcs, but there is an arc is there which is not valid because let us say if I do a NAND gate, if I do one of the input, let us say your B is 0 and A is doing a rise transition or fall transition together let us say I am doing rise and fall transition.

Your A is doing a rise-and-fall transition. Your y is basically A B bar, so B is 0 means A into 0 bar, so 0 bar is 1. So, output will be held to VDD irrespective of change in the A input. So even if you do transition in the A input the signal will not pass to the next stage,

so it is not a valid arc, timing arc. So, your two input NAND gate has four timing arcs. So now we discussed about the positive unate and negative unate. Now we will discuss about the non-unate gate. So non-unate gate is such a gate, your output transition, output rise and fall transition will not basically directly reflected on the input rise and fall transition. So, it depends upon the what is the condition in the other pin. We will discuss this with one example. So let us say you have an XOR gate. So what is the table of XOR gate? So you have A, B and Y both the inputs are same, your output is 0 and 0, 1, 1, 1, 0, 1, 1, 1, 0.

So this is a XOR gate. So let us say you have an XOR gate, this is an XOR gate. So let us say my A is 0, then B is doing a transition from 0 to 1, B pin is doing a transition from 0 to 1. What will happen to my Y pin? So initially B is 0, A is 0, output is 0 and when B is changing from 0 to 1, so it will go to 1, you will get rise transition. So, your input and output is behaving like a positive unate, if my A equal to 0. Let us say I have another case, I have XOR gate, this is your Y, this is A equal to 1 and B is doing a rise transition.

So let us say your A is 1, A is 1 here and B is initially 0, so your output will be 1. So now B changing from 0 to 1, so now A is 1, B is 1, A is 1 and B is 1, so your output will change from 1 to 0, output do a change from 1 to 0. So here what is happening that it is a negative unate, if my A input is 1. So XOR gate is a non-unate type of gate, because your output transition will depend upon the state of the other input, whether A is 0 or 1. So, based on that you cannot tell that a positive unate or you cannot tell that a negative unate. So once you do this gate timing analysis in a path, we have to know what is the condition in the other pin, what is the condition of A pin or the B pin in case of transition happening in the A pin. So, this is an example of non-unate gate. So, this is a dot lib which is used for timing analysis. So if you can see here, this is timing sense whatever I was talking about how my output transition depends upon input transition. So is defined here the timing sense which is a negative unate.

Negative unate means that whenever you do a rise transition at the input, output will do a fall transition and vice versa. So, this is a gate inverter basically, this is inverter. So here, if you can see, it has mentioned that for inverter, the timing sense is a negative unate. So this is used in a timing analysis tool. The timing analysis tool will check that it is a negative unate and based on that it will do the timing analysis, based on that it will do the timing analysis. So this is one of the use of basically negative unate. So we discussed about the timing arc of combinational circuits. What about the sequential circuits? So the timing arc of sequential circuit. So let us take a sequential circuit.

This is a D flip-flop. So, you have a clock pin. So, this is a clock, this is a D pin, or a data pin. Then you have a Q, then you sum up the D flip-flop as Q bar and you have asynchronous reset. So, if you can see here, it has many timing arc. So for synchronous input, in case of synchronous input you have basically setup arc, setup check arc. So, it has two variety rising and falling. Now we have hold check arc, rising and falling. Then these

are for the synchronous input. So similarly you have asynchronous input, basically your clock for synchronous input, your clock to D, you have setup check and the hold check. So, these are the arcs for setup check arc from the clock pin to the D pin. Similarly for asynchronous input, what is the asynchronous input here? Your reset pin is asynchronous input. So here we have two types of arc is there, recovery check arc, recovery check arc, then second one is removal check arc. Now you have third arc is there which is called for the synchronous output. So, what are the synchronous output? Your Q and Q bar are the synchronous output. So for that one you have one arc is there called clock to Q, clock to output, propagation delay. So that is also have rising and falling. So these are the arcs basically involved in case of a sequential circuits. These are the arcs used in case of a sequential circuits. First of all we will discuss about what is setup arc and hold arc. So we have a flip-flop and we have basically we have a flip-flop, it has a clock signal and it has a D pin and a Q pin. So, now we can have basically first we will plot the clock signal, this is your clock. Then you have a data D which is coming before the rising age of the clock, which is coming before the rising age of the clock, let us say it is going like this. So what is the minimum time before the rising age of the clock, your data should be stable that will be sampled by the flop properly that is your setup time. So, we are checking before the rising age of the clock. So, this is the setup time. Similarly, after the rising age of the clock what is the minimum time I can keep the data stable such that it can be captured by the flop properly that is your hold time.

So, this one, this corresponds to setup arc and this corresponds to hold arc. So, these two is very essential for checking the timing. Now we have one more component which is also with respect to synchronous output which is basically clock to Q propagation delay. If your setup hold constraints are met properly, you have a Q signal, this Q, which is the output of the flip-flop. So, let us say this is your clock signal, your data may be anything before the rising age of the clock but since the data is satisfying your setup and hold requirement it will come after some time after the rising age of the clock.

So, then this will be available at the output of the flip-flop. So this is basically your, this distance is basically your clock to Q delay, T clock to Q delay and this is basically clock to Q delay. Now we have one more arc is there which is related to the asynchronous input reset. We will discuss that in detail. So, what is a reset? The reset is a signal if you apply to the flip-flop the output of the flip-flop will be 0. So, I have a flip-flop, this is clock, this is a D pin, this is Q and we have a another signal let us say here reset.

So in this case reset is active high, reset is active high means that when reset is 1 means that your Q becomes 0 irrespective of the whatever the data D applied to the flip-flop. So, what is the situation here? If I have any clock here let us say this is clock. Now I have a basically D input, D input is changing like this then I have a reset I can apply the reset anywhere let us say it is active high, this is active high here. So now let us say I will apply here again. So, what is the point here is that so here I we are assuming that our reset is

active high so here it is going from 0 to 1 so it is called assertion. This process is called assertion, and when it is going from 1 to 0 is called de-assertion. So here what is happening is that we had applied a reset and we have D input is there, we have a reset is given to us then which will be basically driving the Q output. So, here the reset is applied or asserted so your Q will be 0 after some time. So here the Q can be anything before that it can be anything before that but after the reset is appearing so, it will move from whatever the states to 0 state once the reset is applied.

Now again here it is again it will go to 0. So, this is the basically the meaning of assertion and de-assertion. Assassin means that we are changing the reset from 0 to 1, de-assertion means that we are changing the reset from 1 to 0 where we are assuming that our reset is active high. Now with this background we are going to discuss what is recovery check arc and removal check arc. Recovery check arc okay first we will discuss the recovery check arc. In case of recovery check arc we are basically de-asserting the reset signal and then we are giving control to the flip flop and flip flop should be driven by the D input instead of the reset input.

One more thing is that what is the minimum time okay what is the minimum time required with between the de-assertion of the reset signal and the arrival of the clock signal such that the new data can be captured by the flop that is your called your recovery time. Okay, so what is the situation here I have a clock signal I have a clock signal this is my clock and I have a reset signal so it is active high okay so it is active high so it is changing here. So, then we have a data input D which is changing here okay. Now we have a Q, which is the output of the flip flop, so what is this minimum time before the rising edge of the clock reset should be de-asserted? So what is the recovery check arc is that what is the minimum time this time before the rising edge of the clock your reset signal should be deactivated or de-asserted such that your new data D can be captured by the flip flop in this edge. So that is your recovery time so, in this rising edge your new data can be captured it will take some delay and it will appear at the output. So, this time is your recovery time this time is your recovery time okay. So, this is all about recovery check arc now we will discuss about the removal check arc. So, here we have a clock edge we are drawing the clock first this is clock. So now I am drawing the reset signal so the reset is de-asserted after the rising edge of the clock and whenever you are de-asserting the reset how much minimum time after the rising edge of the clock it should be de-asserted the output will be 0 output will be 0 what is the minimum time my reset should be de-asserted after the rising edge of the clock such that my Q should be 0.

So here, if I have a data if I have data like this, then my Q will be changing and capturing the data in the next clock edge in the next clock edge. So this time is called my removal time. So, in summary, this is the minimum time after the rising edge of the clock. The reset signal should be de-asserted such that my Q is 0, and it will act as a normal flip-flop in the next rising edge. So basically here why you are giving this minimum time here is that

basically your Q output is driven by two inputs: the reset input and the D-input. So there is a chance that both the things can create a metastability inside the flip-flop to avoid that we need some amount of minimum time after the rising edge of the clock your reset signal should be stable such that your output will be reset Q should be 0.

So, this is all about removal check arc. So, if you look into this dot lib or the timing library this is the setup rising the timing type is setup rising and this was the table for that one. So, this is giving an example of a setup rising arc in case of a sequential circuit. So how this tool requires this information to do the timing analysis in the complete path was stated here. So, the unateness and timing arcs are essential for the timing analysis of any digital circuits. So, in this lecture we discussed about unateness, source pin and sink pin and different types of unateness like positive unate, negative unate and non-unate types of gates and we also discussed about the timing arcs of the sequential circuits.

So, this is very useful for timing analysis of digital circuits. Thank you for your attention.