

VLSI Physical Design
Prof. Indranil Sengupta
Department of Computer Science and Engineering
Indian Institute of Technology, Kharagpur

Lecture - 41
Performance-Driven Design Flow

So, to summarize we have earlier looked at the various physical design automations steps like partitioning, placement, floorplanning, routing and so on. Later on a looked at some of the performance driven issues, we looked at performance driven placement and routing and also some techniques for physical synthesis; how we can make some timing corrections. So, now, let us try to look at the overall picture now for this kind of high performance designs, where these performance driven issues are important, so what should be our so called performance driven design flow. So, we look at into the overall picture in this lecture.


(Refer Slide Time: 01:09)



Introduction

- We have discussed several techniques to improve timing of digital circuits.
- All these optimization techniques can be combined in a consistent *performance-driven physical design flow*.
- Due to the nature of performance optimizations, their ordering is important.
 - Evaluation steps (like STA) must be invoked several times.
 - Some optimizations like buffering must be redone multiple times to facilitate a more accurate evaluation.

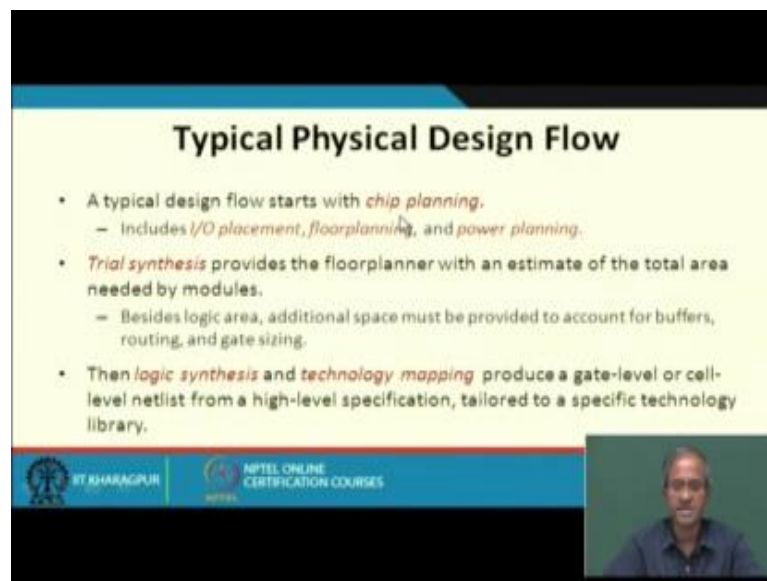
 IIT KHARAGPUR  NPTEL ONLINE CERTIFICATION COURSES



So, let us have a quick look at the so called performance driven physical design flow; which is more or less standard practice in modern day VLSI circuits. So, here what we are trying to tell you is that whatever techniques we have looked at, let us try to combine it in a consistent design flow. Now here we shall see that there are a few things like static timing analysis, which we do not consider as a step, we consider this as a tool and this tool can be used a number of times in different phases of the process. Similarly buffering

is a very common technique which may have to be done multiple number of times, there can be a initial level of buffering may be after 1 cycle we feel like still some timings are not been satisfied you may have to do some more buffering. So, there are some iterative steps which need to be carried out; there are a lot of iterative steps and even at the end you may find that still a few things are not been met, you may have to do a few more corrections there.

(Refer Slide Time: 02:33)



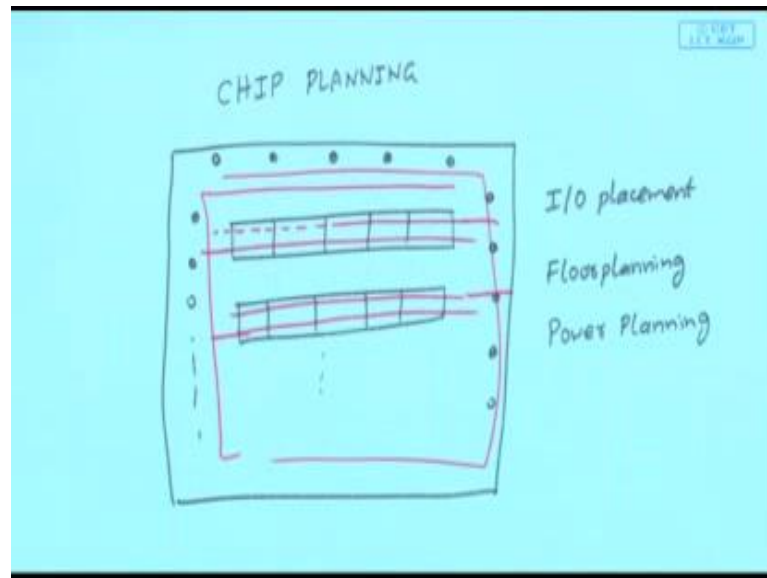
Typical Physical Design Flow

- A typical design flow starts with *chip planning*.
 - Includes *I/O placement*, *floorplanning*, and *power planning*.
- *Trial synthesis* provides the floorplanner with an estimate of the total area needed by modules.
 - Besides logic area, additional space must be provided to account for buffers, routing, and gate sizing.
- Then *logic synthesis* and *technology mapping* produce a gate-level or cell-level netlist from a high-level specification, tailored to a specific technology library.

IIT KHARAGPUR | NPTEL ONLINE CERTIFICATION COURSES

So, let us revisit the typical physical design flow once more. Now the typical physical design flow starts with something called chip planning, well we did not use this term earlier we called it floorplanning placement, but in chip planning we include 3 different things. Let me just try to just highlight.

(Refer Slide Time: 03:05)



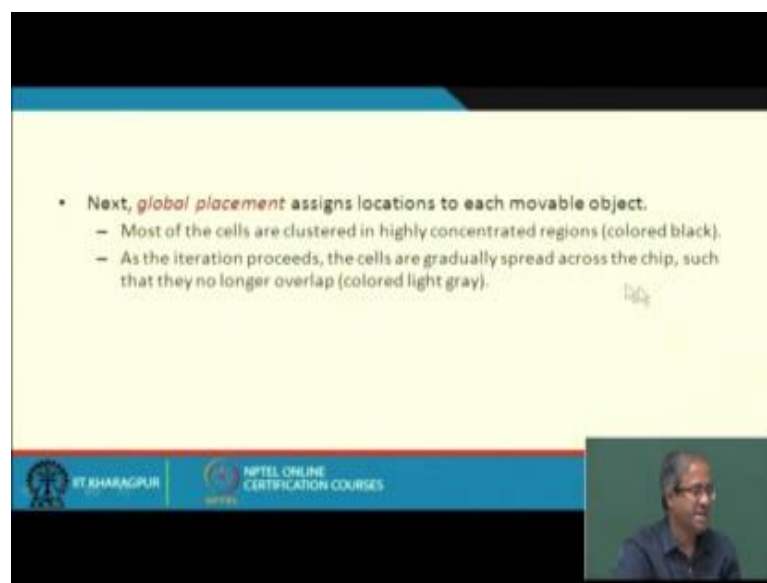
So, in chip planning we are talking about something called chip planning, which is also a very commonly used phrase first is a step called I O placement. So, I O placement means for this chip we have to place the I O pins or which pin should be carrying which signal. This is the I O placement like this, then we can have floor planning of course, I said that most of the circuits today are based on standard cells, floorplanning and placement arriving the identical in those cases. So, in floorplanning we are telling full standard cell has to place in which rows and there is another thing which is done together power planning.

Because now that we have the standard cells already laid out, we know that; where are the power connections that we need. V DD and ground here, V DD and ground here like that. So, you also plan overall whichever technical we use into supply the power supply and ground lines whatever way you do. So, this kind of a power planning network also you do. So, this entire process is sometimes referred to as chip planning; that overall the chip how does it look like, for or the pins what will be these pins carrying, these are the cells which cells would be put on which rows, and how will the power network look like that let powering that will provide you power to the different rows and the cells ok fine. So, here there is a something that trail synthesis, which is the initial thing which you have got, this provides the floorplanning tool so that estimate of the total area that is needed you can see that; and in this estimate you should provision for buffers so

additional place for routing, and gate sizing; some gates may have to be made larger. So, you have to keep some additional spaces for these things right.

So, this earlier we did not talk about all these things, but now we are saying that we need to keep provisions for all these things which need to be added in later stages of the design flow. Then logic synthesis and technology mapping of course, of this can be done earlier also, it produces a gate level cell level netlist; well when you are doing a floorplanning sometimes these are done together or this can also be done earlier.

(Refer Slide Time: 06:16)

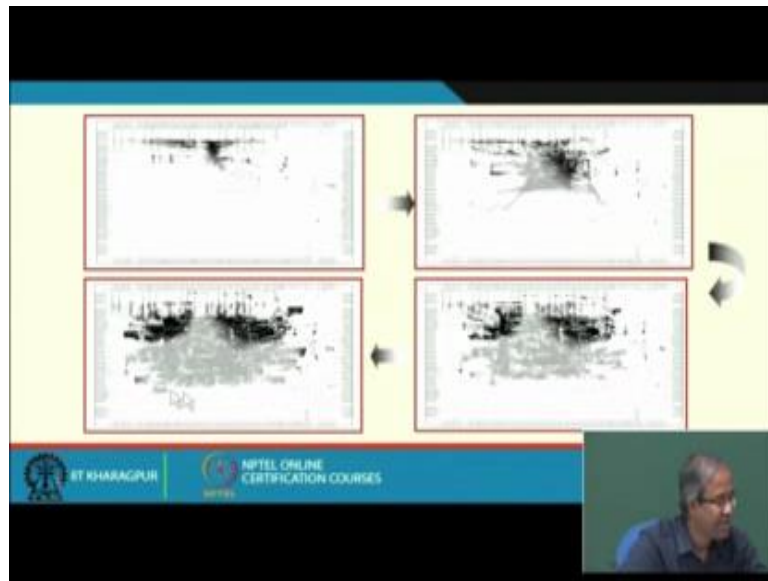


- Next, *global placement* assigns locations to each movable object.
 - Most of the cells are clustered in highly concentrated regions (colored black).
 - As the iteration proceeds, the cells are gradually spread across the chip, such that they no longer overlap (colored light gray).

So, after you have placed the cells you can carry out global placements. Now global placement whatever I have said that you place the blocks or the cells in rows of the standard cell. Now again I have said that in a typical VLSI design flow you cannot do the thing in 1 go it is a repetitive step. So, you make a placement, you may find that something is not working properly there is a lot of congestion somewhere, there is lot of delay in some places, you would may have to change the placement continuously.

So, here what I am saying is that this the process of global placement, it typically assigns locations to be objects where means I shall be showing a slide.

(Refer Slide Time: 07:25)



So, here we look at the clustering information, you try to spread the cells informally across the chips like what I am saying is that you may find that initially your chip plan looks like this, where this dark region means there is a lot of clustering and congestion here. Many of the cells the global placement would say that they will be mapping map here. So, you proceed this region slowly spreads, it spreads across the whole chip and many finally, you will be getting something like this. So, your whole placement must be spread across the entire area of the chip something like this. So, very very roughly I am saying.

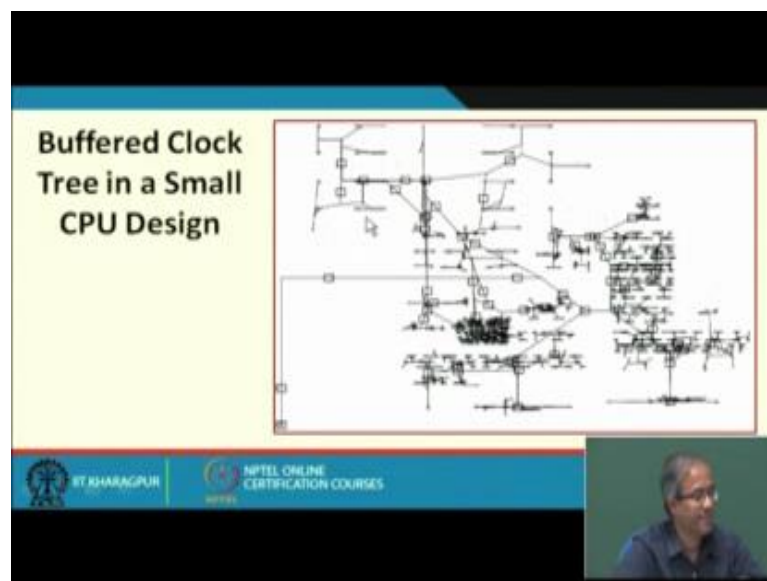
(Refer Slide Time: 07:58)



Then after the global placement is done, you see you cannot synthesis the clock network before you know where your flip flops are located. So, once you have done global placement, you know where your registers are where your flip flops are the sequential elements. So, with that information you can proceed to synthesis the clock network. So, this clock network as you know we have seen earlier, this can be a either a simple clock tree; it is like an H-tree or an MMM or something like that or you can have a more flexible network consisting of a tree at the higher level, and a very regular mesh at the lower level. Normally we have processors we have this kind of hybrid, because in a processor chip there are a lot of points where you need to carry the clock signals, there are large number of target or sink points for the clock signals.

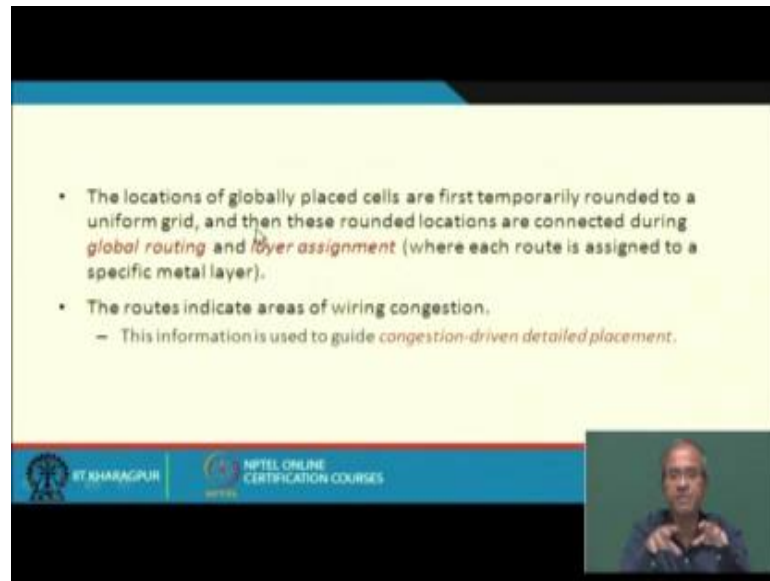
So, we normally have a mesh in the lower level and a regular tree at the upper level and then a H-tree usually fine.

(Refer Slide Time: 09:11)



So, this is just a sample slide showing a buffered clock tree in a small processor design, where this rectangles indicate the places where buffers have been inserted and the lines indicate the connections, and the x indicates the points where the clock has been taken fine.

(Refer Slide Time: 09:39)

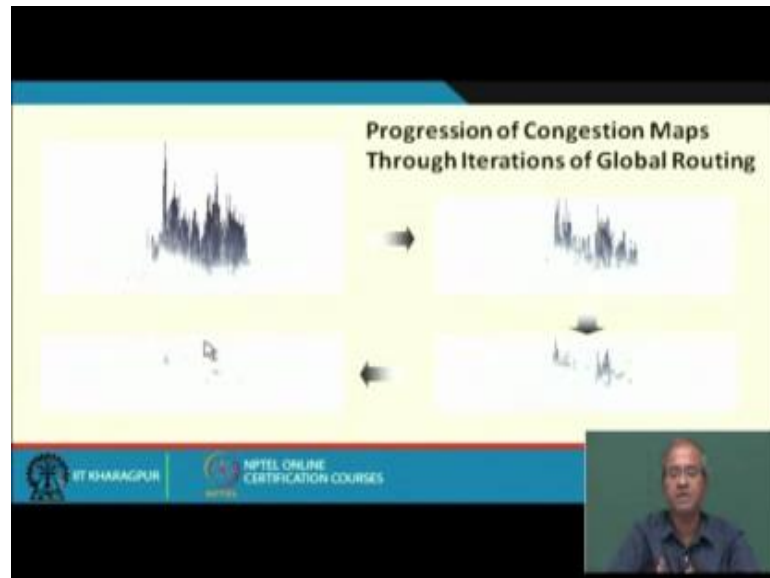


- The locations of globally placed cells are first temporarily rounded to a uniform grid, and then these rounded locations are connected during *global routing* and *layer assignment* (where each route is assigned to a specific metal layer).
- The routes indicate areas of wiring congestion.
 - This information is used to guide *congestion-driven detailed placement*.

So, after placement is done. So, the locations of these cells are well they aligned to a grid and after these align you see; during placement normally we do not have the grid concept. You can place a cell virtually anywhere, but when you do routing you often have some rows and column concept; tracks and some columns. So, there you have to align everything to a grid. So, after this initial level of placement is over, you have to do and of this alignment of this blocks or cells to grid locations to a uniform grid. And after you have done this, so you can proceed for global routing and layer assignment; where layer assignment says for each of the routes, which layer these layers are typically metal layer, which means which metal layer will be used to connect these nets.

Now, global router typically completes the routing on a single layer like you say the lease algorithm; they will the lease algorithm or the headlocks algorithm, they will try to find out a route on a single layer single metal layer. Now here when you do routing, you may also land up in something called wiring congestions, you may find there in the chip there many areas, where the wiring congestion is pretty bad. So, you may have to do iteration here again. So, you do something like congestion driven detailed placement, now you again modify your placement based on this information like I am just showing an example.

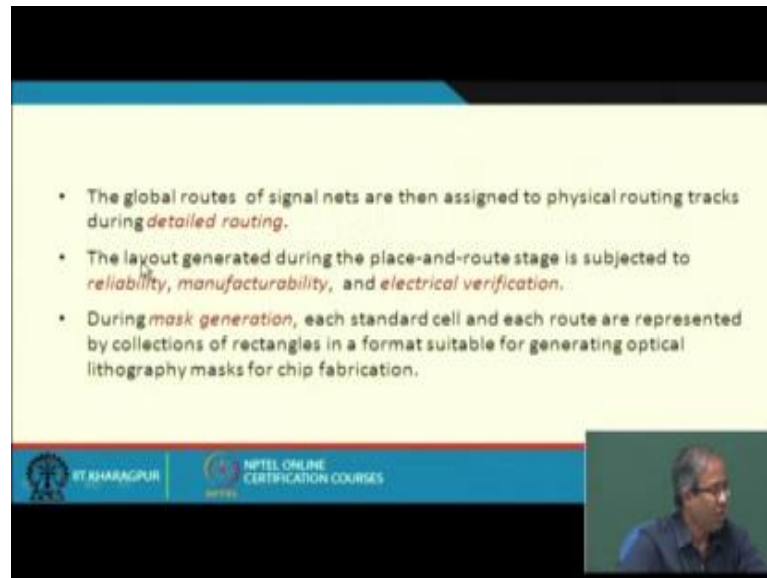
(Refer Slide Time: 11:39)



This shows you means over the area of the chip these peaks indicates the levels of congestion; there is an area where lot of congestion is there. Now with iteration we try to reduce congestions, reduce congestion and you will see finally, that there is almost no congestion they are uniformly placed.

Congestion means I refer to congestion with respect to routing. So, if there is congestion it is quite likely; that during the process of routing you will find that you are not able to complete the routing at all. So, such scenario should not occur. So, you should do something which I just mentioned called congestion are aware detailed placement. So, you modify the placement again such that the congestion value is get improved to a significant extent.

(Refer Slide Time: 12:41)



- The global routes of signal nets are then assigned to physical routing tracks during *detailed routing*.
- The layout generated during the place-and-route stage is subjected to *reliability, manufacturability, and electrical verification*.
- During *mask generation*, each standard cell and each route are represented by collections of rectangles in a format suitable for generating optical lithography masks for chip fabrication.

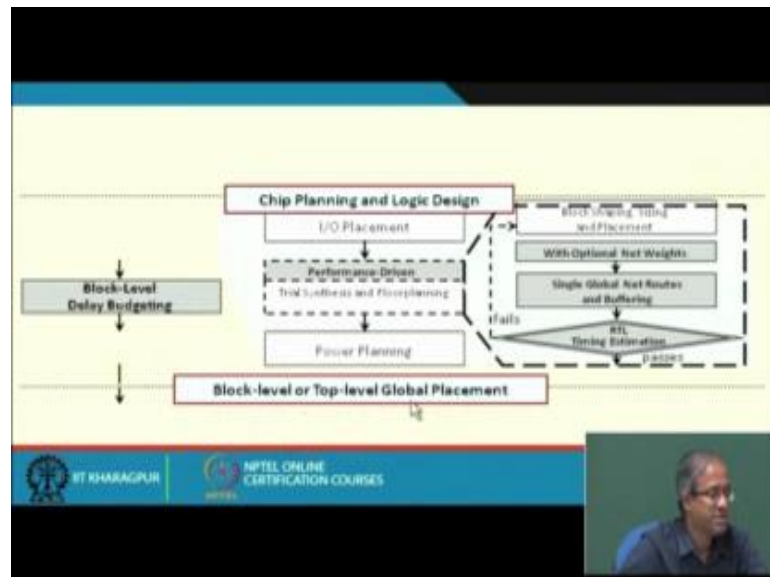
So, now after that you proceed to detailed routing; and the global routing gives you approximate routes, now detailed routing will just assign the exact horizontal and vertical metal layers for these routes. Now there is a additional optional step like these routes which are generated the wires. So, you may have to go through another iteration, which consist of reliability, manufacturability and electrical verification. Like let me tell you reliability refers to means you have seen the process of channel routing; there are horizontal tracks, vertical tracks, and there are some wire connections that connect the points across layers. So, you can have two layers or multi layer channel routing. Now the wire connection provides with you can say some source of unreliability you can say, because you are ultimately drilling a hole and making a connection across two layers. So, higher the number of wire connection, higher will be the possibility that there is in that the chip can fail because of some miss connection loose connection or something like that wrong connection.

So, one objective may be to reduce the number of bends in the connection, which may indicate number of wire connection. So, many a time means on the same layer when routing is carried out instead of using sharp bends, you use something like a 45 degree bend; because sharp bends are places where the metal is likely to break and form a disconnection right. These are something which are called reliability and manufacturability, these are something which have to be looked at it after everything is done. And electrical verification means well after you have laid out the wires, you see

the length of the wires whether the separation is sufficient, width of the wires are sufficient, there are some very simple set of design rules you can say. So, all those design rules are verified by a design rule checker, which can tell you that well. Here the places where I find some violations you please correct these things, in that case you may have to go back and make some corrections there some rules are getting violated ok.

So, after everything is done you proceed for mask generation where. So, every circuit element and interconnection are represented by rectangles. Rectangles around various layers polysilicon diffusion, metal, metal 1 metal 2 metal 3 and so on fine.

(Refer Slide Time: 15:32)

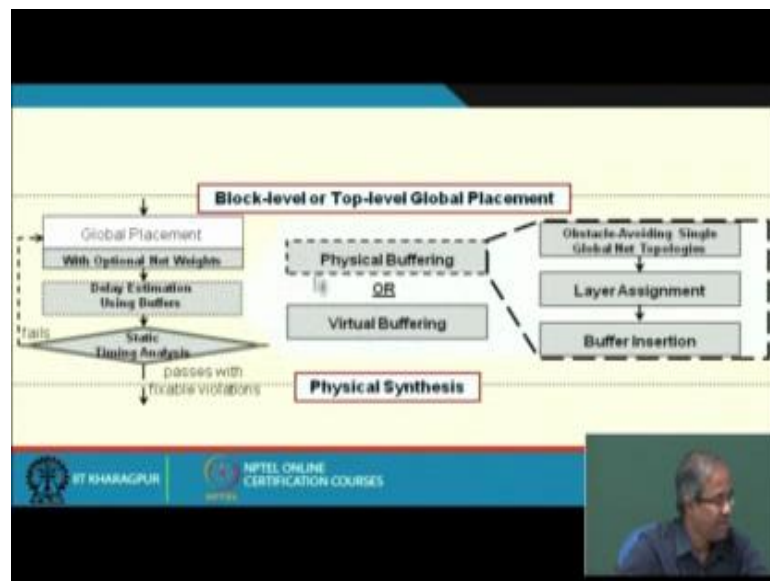


So, let us look at the overall flow in the form of flow chart. So, in the first step we assume that already chip planning is done. Chip planning is given as input to the physical design, logic design is also done. So, you do some kind of block level placement. So, you are doing some as output you are generating block level or higher level global placement. So, whatever things you are doing? Chip planning means you have already done I O placement, you have already done power planning. Now you are doing performance driven trial synthesis and floorplanning; this is a new step which we include in the performance driven design flow. So, what does this involve? This consist of block shaping, sizing and placement and as I had said earlier we can assign some weights to the nets net weights depending on the criticality, slag values, then you can just analyze the global net routes, and for the ones where the slag values are negative you use some

buffering and here at this level you can do some approximate timing estimation using some timing analysis.

So, if it passes move on to the next step, if it fails you again go back and again you modify your placement. So, I had said as I had said that this is not a single process, it is a continuous iterative process, several iterations are carried out this is the first step.

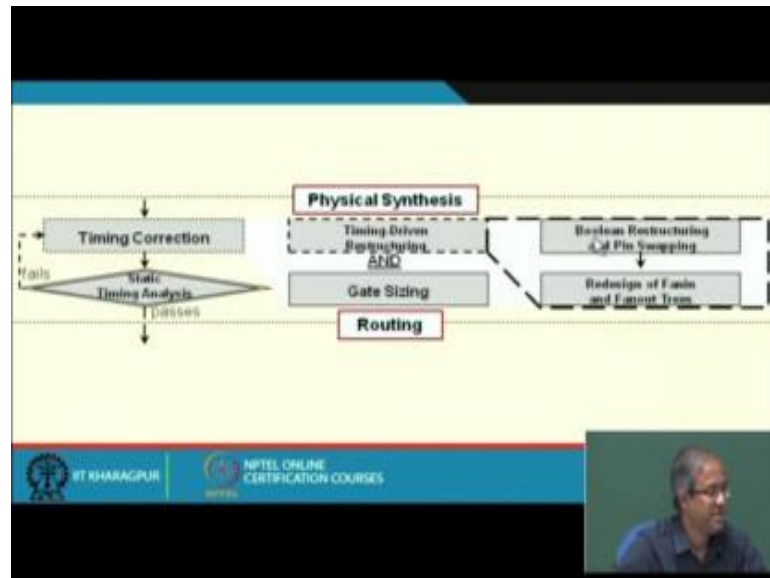
(Refer Slide Time: 17:16)



Then from block level global placement you move to physical synthesis. So, what you do? You start with a global placement with of course, net weight optional net weights, then you do some delay estimations, you can use buffers in this case then you carry out static timing analysis again. If it fails you again go back and change the placement. Now this delay estimation using buffers, so here the details are shown here, so what you do? You do some physical or virtual buffering.

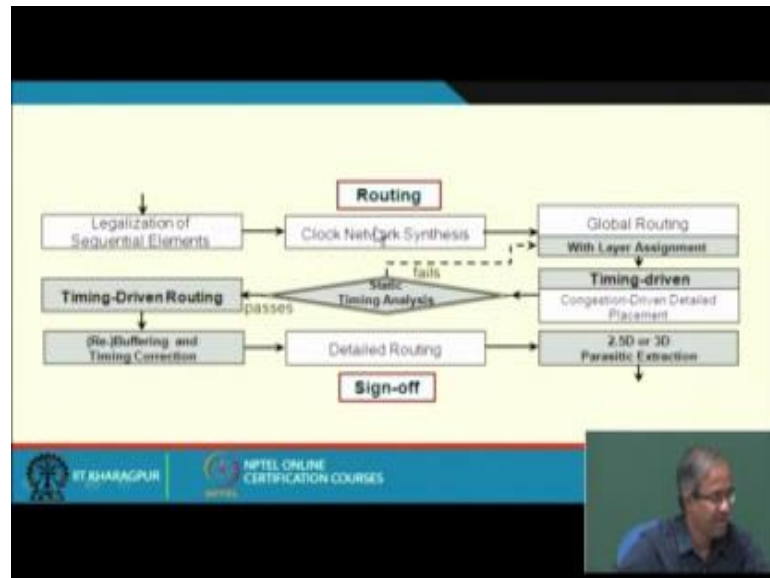
Physical buffering means you introduce some buffers, or virtual buffering means you indicate that here you may require a buffer, but you do not introduce right away you may need to introduce later. So, for physical buffering means you are actually introducing the circuits, the buffer the inverters. So, here you have to use an obstacle means obstacle avoiding global network topology, because you are inserting the buffers you have to connect them to the input and output lines, you have to see where the circuits and obstacles are already there, you have to lay the nets like that. So, layer assignment, buffer insertion these are the important steps here.

(Refer Slide Time: 18:46)



So, next comes after physical synthesis done to routing. So, here timing correction is one step, where wherever there is a negative slack you make some corrective actions and maybe using the methods I have already mentioned just sometime back. And again you use static timing analysis if there are some violations, you repeat this process. So, here timing correction can involve timing driven restructuring, gate sizing and restructuring can be used in Boolean restructuring, and pin swapping, redesign, fanin trees, fanout trees all the techniques that you have just now seen in the last lectures. So, you do all these things so that this timing constraint whatever was there was made, physical synthesis is done.

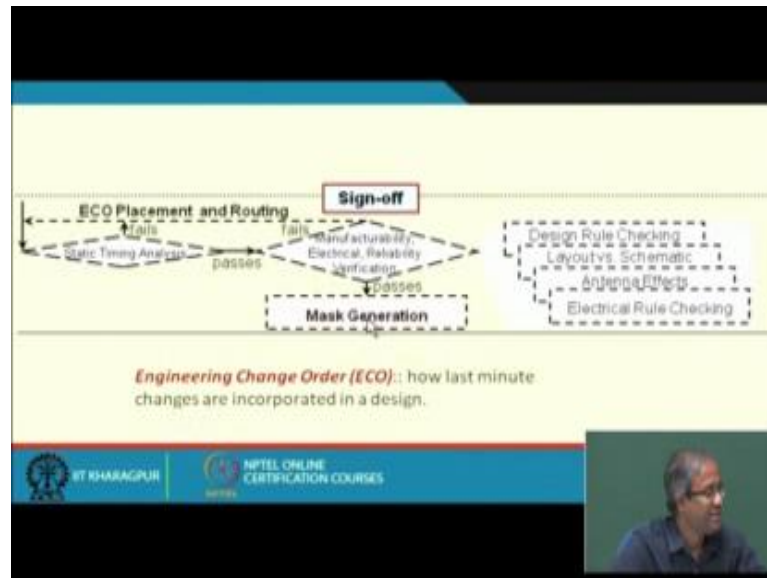
(Refer Slide Time: 19:42)



So, now this is the last step. So, you finalize locations of the sequential elements, now you synthesis the clock networks once they are finalized. So, after the clock network as you synthesis you do this once more global routing, layer assignment again you check the timing driven, congestion driven, detailed placement like you check for the timing here static timing analysis. If it fails again you go back again you modify it, but if the timing analysis is passed then you proceed to timing driven routing.

So, during this routing you may have to again insert buffers, you may have carryout some timing correction mechanisms here. So, once everything is done then you do detailed routing, then you can do some parasitic extraction, some additional steps of simulation and finally, you proceed to sign off.

(Refer Slide Time: 20:54)



Sign off what you do here, you check for as I said manufacturability, electrical properties reliability verification. So, everything passes then only you proceed to generate the mask for fabrication, but if it fails then you go for a step called equal placement and routing. Equal stands for engineering change order. So, what it refers is that, this refers to a last minute changes that mean you are in a step, which is just one step before the final fabrication. So, at this step you find that well with respect to manufacture reliability there are some problem let me make some small corrections. You go back make the small corrections, run timing analysis again if it passes; if it fails again make some small changes.

So, in this step what are the things that we have done? Design rule checking one thing I have just mentioned, layout verses schematic. So, you had a layout initially, you had this schematic diagram you can make a comparison that whether they are matching. There are some electrical effects like antenna effects, electrical rule checking, you may also have to do all these things at this step and there are some well defined templates using which these checks can be carried out. Because one thing you understand, ultimately your layout is a huge thing there are millions and millions of rectangles, you will have to do this check over this entire rectangle, unless it is a feasible and simple process you really cannot do it. So, this is you can.

So, I think with this we just come to the end of this lecture, in the next lecture we shall be seeing some additional timing correction methods, and we shall be having a relook at the insertion of buffers and drivers design of driver's buffers etcetera. So, that we can just use that information to just correlate with whatever we have said just now.

Thank you.