

# Physical Design Automation of Complex ASICs

Chentouf Mohamed<sup>1,2</sup> and Alaoui Ismaili Zine El Abidine<sup>2</sup>

<sup>1</sup> Mentor Graphics Corporation/ICDS Division  
Rabat, 10010, Morocco

<sup>2</sup> Information, Communication and Embedded Systems (ICES) Team, University Mohammed V  
Rabat, 10010, Morocco

## Abstract

Physical Designs of Application Specific Integrated Circuits (ASICs) is a challenging and paradigmatic process as the resources are limited or difficult to find. This paper shows the fundamental steps of Physical Design with the use of Electronic Design Automation (EDA) tools like Nitro-SoC and Olympus-SoC of Mentor Graphics. In order to illustrate the Place and Route flow, experiments were carried out using a test circuit and the results of each flow step performed were displayed with snapshots. Although the intent of this paper is to put forward a technical approach for mastering ASICs' back-end design in more details and engage engineers in practices, designers in the progression of the Physical Design of a complex chip may also get benefits from it, since it answers the central questions faced at the advanced nodes SoCs. This hands-on flow steps will be an added advantage, besides the traditional Place and Route (PnR) flow. It will also help professional scientist and engineers to enhance their skills and expertise in the back-end design, and to analyze and fix ASIC circuits.

**Keywords:** *Physical Design, Place and Route, Netlist, Specs, Floorplanning, Placement, CTS, Route, Skew, Latency, ASIC, Hold, Setup, Power Consumption.*

## 1. Introduction

Many books and articles were published lately to help specialists and professionals enter the physical design world, but given the multidisciplinary of the problem, each researcher focused on a specific aspect and approached the field from a different angle. Some focused mainly on the algorithmic aspects and intended to show specialists and users of VLSI CAD tools what is going on inside these tools. To gain knowledge about what can or cannot typically be performed by such tools [1]. A new approach was adopted by [2], it forces the specialist to truly understand the basic physical design algorithms and apply them to small but insightful problem instances. The author in [3] covers the broad spectrum of computer-aided design and optimization techniques, which include circuit performance modeling, design by optimization, statistical design optimization, physical design automation, computer-aided analysis and circuit simulations; and design automation, which includes system-level design,

performance modeling and analysis, and hardware-software co-design.

Other researchers treated the electrical aspects, like in [4] where authors treated the RC extraction of an integrated circuit and presented the key concepts of the extraction and various field-solver techniques, it shaded the light on the computational complexity, physical theory, numerical stability and robustness of the algorithm. [5] Leverages theory and techniques from fields such as applied physics, communications, and microwave engineering and apply them to the field of high-speed digital design, creating an optimal combination between theory and practical applications. [6] Dig deeper into the physical phenomenon raised from the deep sub-micron technologies and presents a bottom-up approach to VLSI conception from transistor to system level. [7] Gives a big attention to the problem of writing and understanding timing constraints in integrated circuit design and provide a guide on how to learn to write the constraints effectively and correctly, in order to achieve the desired performance of their IC or FPGA designs, including considerations around reuse of the constraints. [8] tries to resolve the paradigmatic question to achieve High Performance, Low Power, and Reliable 3D Integrated Circuits, it presents the results of the work done on the entire spectrum of design and testing for a test chip: architecture, layouts, CAD tools, package, board, and testing infrastructure. [9] Focuses on the power aspect, and presents a combination of different techniques, including architectural design choices, logic, and physical design, choice of circuit families and implementation technology to achieve Low-power consumption in integrated circuits and systems.

Available tools and methodologies represent separate axes of research, [10] Incorporates many aspects such as application-specific tools and methods, performance evaluation methods; power estimation methods, and design planning, both for digital and analog and mixed-signal designs. [11] Presented a traditional physical design flow using Cadence (Encounter) place and route and Synopsys (PrimeTime) for static timing analysis. [12] Presents the tools and techniques available for High-Performance ASIC Design and tries to show the opportunities existing in custom design methodology and apply them to close the

gap between the computational efficiency of ASICs and custom silicon.

Another interesting approach focused on the fabrication processes and all the issues raised from it, [13] presented the process of layout generation, but focused mainly on the issues related to the manufacturability aspects. A more advanced approach was covered by [14] about the three-dimensional (3D) integration, which provides a mechanism for space transformation of the traditional planar implementation of integrated circuits into three-dimensional space. Also [15] treated the challenges raised by 3D integration technology and proposed a 2.5 D as an in-between solution to overcome the accumulative yield loss problem hindering other 3-D integration schemes. [16] and [17] present the impact of Electro-migration at the circuit layout level and treat the electromagnetic compatibility of integrated circuits, they provide specific guidelines for achieving low emission and susceptibility derived from the experience of EMC experts. [18] Focuses more on the manufacturability aspect to prevent catastrophic failures due to the difficulty in applying proper optical proximity correction (OPC) or failure of the OPC/RET algorithm induced by layout designed with no knowledge of its impact on patterning.

While some of the available works treat the subject in its globality, others tried to focus only on one aspect and analyze it in depth, [19] Identifies new objectives, constraints and concerns in the clock-network synthesis for systems-on-chips and microprocessors and proposes new techniques and a methodology to reduce dynamic power consumption for large IC designs with macroblocks by integrating clock network synthesis within the global placement. [20] Gives a particular attention to the IC Interconnect, which has become a dominant factor for integrated circuit performance, it provides a comprehensive coverage of modeling and simulation of RC and RLC interconnect, including the interactions with gates. [21] Provides an understanding of various leakage power sources in nanometer scale MOS transistors and at the full chip level. And emphasize the most known leakage power reduction techniques, such as power gating, dynamic voltage scaling, body-biasing, and use of multiple performance transistors. [22] Enlarges the power's scope and covers all the low-level aspects of the design of low-power integrated circuits (ICs) in deep submicron technologies, [23] went a step further and proposed an approach to turn the leakage problem into an opportunity by exploiting leakage currents to perform the computation. It uses body biasing adaptively to compensate for PVT variations. [24] Presents a host of challenging global combinatorial optimization problems faced due to the enormous size and complexity of current and future integrated circuits (IC's). [25] Presents the Multi-voltage CMOS Circuit Design as a very effective technique to

reduce power consumption without degrading the speed by selectively lowering the supply voltage along non-critical delay paths. [26] Focuses mainly on the routing algorithms and presents some solutions to important aspects and issues related to routing algorithms, while [27] explores routing congestion management and progresses with a comprehensive discussion of the techniques available for estimating and optimizing congestion at various stages in the design flow. [28] Covers most of the VLSI timing related issues such delay calculation, timing constraints, interconnect parasitic and coupling, composite current source (CCS) timing and noise models, while [29] presents the optimization techniques used to fix the violations detected after the STA is performed, these techniques include Gate Sizing, Buffer Insertion, and Threshold Voltage Assignment. [30] Idea is to capture a technology snapshot of dominant placement algorithms from flat placement techniques to multilevel placement methods.

Due to this wild diversity of materials, many specialists get confused as for where to start their learning path and how to proceed to learn physical design, as a result, they get familiarized with one aspect and lack knowledge of other aspects. This is again due to the lack of comprehensive textbooks/papers which cover practical approaches in all aspects of VLSI physical design. The goal of this paper is to provide the essential steps required to start mastering the physical design of ASIC. It is our intention that the paper present enough detail and self-contained material to give the reader a basic idea of ASIC place and route process.

The remainder of this work is organized as follows. Section II provides a detailed description of the main place and route steps using Nitro-SoC of Mentor Graphics. Section III presents the Chip Finishing and Design for Manufacture techniques used in backend flow before starting the physical verification cycle. Finally, Section IV draws the conclusion.

## 2. Place and Route flow

Physical design is the process of placement and routing (P&R) of ICs (cell-based ASICs, custom ASICs, FPGA...). We will focus on cell-based ASICs P&R flow where each cell is assigned a geometric location and connected to other cells by way of metal lines. This is done automatically by EDA tools such as Nitro-SoC (Mentor Graphics), the resulting layouts are almost always correct by construction and design productivity is much better than for manual layout. This chapter explains the main steps of Nitro-SoC P&R flow, which begins with floorplanning and placement, and then handles physical synthesis, CTS, and routing. Signal integrity (SI) analysis and multi-corner/multi-mode (MCM) analysis can be performed at any stage during the design flow.

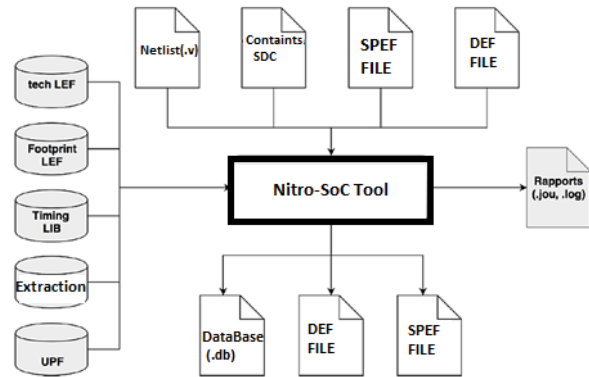


Fig. 1 Nitro-SoC Inputs/Outputs

### 2.1 Inputs and outputs of Nitro-SoC

Nitro-SoC P&R tool takes input from a number of different sources and generates output in various formats (Figure 1.). It is important to make sure that the environment setup is correct and the right data is loaded since wrong data can lead to many surprises. For example, if wrong constraints are loaded and latency are not budgeted properly, it is common to end-up re-implementing the design from scratch with updated latency/clock definitions.

As for outputs, Nitro-SoC P&R tool can export physical design information such as DEF for a particular partition

which contains complete physical information about cell placement and detail routing. Also, it can write out a GDSII file which is a geometric representation of the polygons that describe the actual layout of the design with all its connectivity. (Table 1 and Table2.)

After data preparation, the Back-End flow could be started. It consists of converting an RTL circuit description to a Physical Design, which can be divided into four major parts, floor-planning, placement, clock tree synthesis, and routing.

Table 1 : Nitro Input data

Nitro-SoC Input data		
Input data	File content	Extension
Design Synthesized Netlist	A gate-level description (physical entity) generated by a synthesis tool from RTL description when an ASIC library is selected.	.v
Physical Libraries (Lef of GDS file for all design elements like macro, std Cell, IO pads etc.)	Contains complete layout information and Abstract model for placement and routing like pin accessibility, blockages etc..	.lef
Timing, Logical and Power Libraries	Contains Timing and Power info	.lib
Constraints	Contain all design related constraints like Area, power, timing	.sdc/.tcl
Floorplanning	Contain floorplanning information if this step is done with a third party tool and needs to be imported	.def/.pdef
Design DB	A Nitro database format used to store all the designs info in a specific stage	.db
UPF (Unified Power Format)	Description of the power strategy of the design when some blocks being supplied with a different voltage than the top level	.upf
SPEF (Standard Parasitic Exchange Format)	Resistance and Capacitance info of cells and nets used to back annotate the design if a third party tool is used for extraction.	.spef

Table 1 : Nitro Output data

Nitro-SoC Output data		
Output data	File content	Extension
Logical description (Netlist)	A gate-level description generated by the P&R tool. It contains all the logical transformations added to fix the timing, propagate the clock, reduce area or power consumption.	.v
Physical description	Contains complete layout information of placement and routing, this format is used when multiple tools are used to carry out the P&R flow.	.def
QoR report	Transcript a report containing the QoR metrics values: timing, congestion, wire- lengths, area, utilization, power ...	.log
Run transcript	Report of the details of different operation run by the tool.	.log
Design DB	A Nitro database format used to store all the designs info in a specific stage	.db

## 2.2 Placement

Placement is a key step in the physical design flow. There are two types of placement: global placement and detailed placement. The objective of the global placement could be driven to meeting timing specification, congestion or wire length of interconnections; however, the objective of detailed placement is to ensure the legality of design and to respect the constraints such as timing and congestion [38].

The objective of the placement is to place the standard cells in order to:

The objective of the placement is to place the standard cells in order to:

- ↳ Keep the minimum distance between the cells to avoid congestion of the circuit.
- ↳ Respect the polarity of the cell supply pins.
- ↳ Maximize design routability.

Respect the timing constraints, and optimize the delays on the interconnections. Since the delay is directly correlated with the length of the wires, the placers often minimize the total wire length. It calculates the amount of wires to use to connect all the logic gates of a given Netlist .Then, it tries to find the locations of the standard the net. A super-bounding box (sbox) of a pair of nets is the union of the bounding boxes of the two nets, and an intersection box (ibox) of a pair of nets is the intersection of the two bounding boxes. (figure 2) [37]

One of the best-known methods for estimating the design wire length is Half-Perimeter Wire Length (HPWL =  $\Delta X + \Delta Y$ , as shown in figure 3) [36] or Bounding Box method (BBOX). A bounding box (bbox) of a net is defined to be the smallest rectangle which contains all the terminals of



Fig. 2 Illustrates the overall placement step

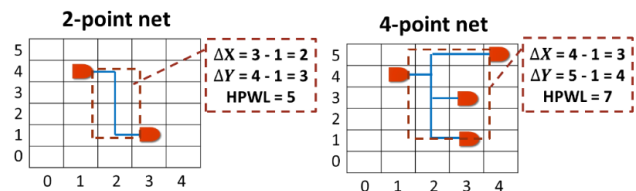


Fig. 3 Wire-length Estimation Example

### 2.2.1 Global Placement

Global placement computes the best position for each cells group (module) to achieve well-spread placements, while concurrently optimizing for wire length, spread, and timing. The goal is to find the best trade-off for all of the objectives. At the beginning of the global placement phase, all cells belong to one group that spans the entire chips core area.

A typical top-down hierarchical placement approach can be generalized as follows: at a given hierarchical level, the layout area is partitioned into several global bins. All cells of the circuit will be distributed into these global bins to minimize a certain placement objective. This cell



distribution problem is called a hierarchical placement problem. If a cell is distributed into a particular global bin, it will be placed within the area of this bin in the final layout. As we proceed to finer levels, the number of global bins increases and the physical size of global bins decreases. Thus we can get more and more detailed information about physical locations of cells as we proceed. The top-down hierarchical approach will terminate when there are only a few cells in each global bin. [39]

The Global Placement uses many algorithms [41] depending on the case to be dealt with:

- Optimize wire length.
- Avoid congestion in the circuit.
- Respect the timing of the circuit.

### 2.2.2 Detailed Placement

The detailed placement is used for creating a legalized placement after each step in the flow. Its objective is to increase cell-spreading and legalization with minimum cell displacement (figure 4). It uses cell bloats to reserve free space around each cell in order to improve the routability of the design. Detail placement algorithm can prioritize cells according to setup-slack, where cells that are timing-critical will suffer lesser displacement relative to other cells in the vicinity. A cell's timing criticality is determined by the timing criticality of nets that it connects to. A net is more critical for timing if it has its slack closer to the worst negative slack (WNS) or it is driven by a weak driver, and hence sensitive to wire-cap changes.

The legalization phase performs a DRC run on cell ports and blockages with respect to pre-routes in the design. It also ensures that a cell's port can be reached by dropping a via (pin-access check). This check ensures that if a via is placed to connect to the cell's pin, it will not have DRC violations with pre-routes.

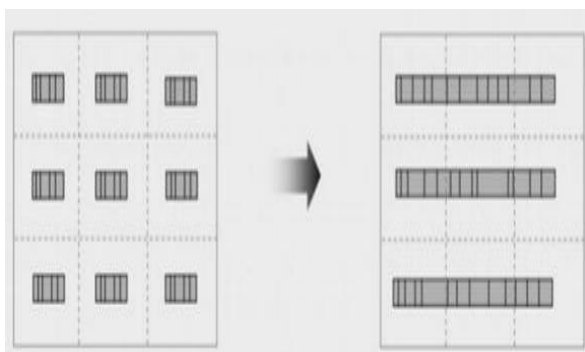


Fig. 4 Global placement vs detailed placement

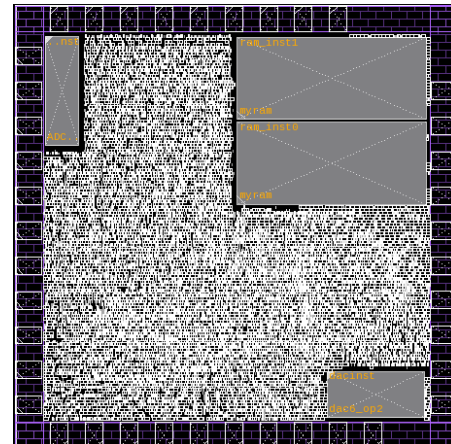


Fig. 5 Integrated circuit after placement step

### 2.3 Clock tree Synthesis

The synthesis of the clock tree consists of creating an electrical network to distribute the clock signal to all synchronous cells of an integrated circuit based on a previously prepared specification. A CTS specification defines a clock network for the CTS module. The clock network for the timer is specified using the SDC command "create\_clock" or "create\_generated\_clock". In typical usage, the clock network defined by a CTS specification will be contained within the clock network of the timer. Any pin that is on a path from the root to a valid leaf for the timer will be part of the clock network described by the specification. In cases where the root pin is not on the clock network for the timer, any pin that is not a buffer or inverter pin is considered a valid leaf pin of the network.

The most economical way of distributing a clock signal is that of a tree topology. An ideal clock tree is the H-tree topology where the basic building block at each level of the distribution network is a regular H-structure. Another scheme that yields equal-length interconnections is the X-tree where the basic building block is an X-structure. Such regular topologies also facilitate the addition of clock buffers in a symmetrical fashion [40].

H-trees (or X-trees), although effective in equalizing path lengths from a driver to a set of sinks, have serious limitations. These trees are best suited for regular layouts where the clock load is uniformly distributed over the entire chip. It is not particularly suitable for irregular placements with varying sink capacitances, which are common for cell-based designs. Moreover, a tree topology is more susceptible to the effects of variations in process parameters and operating condition because of its lack of redundancy; there exists only one unique path from the

clock source to a flip-flop. A very effective way of introducing redundant clock paths to a balanced clock tree is to add a trunk to connect all the leaf nodes of the clock tree, and branch off from the trunk to drive clock pins [40].

For MCM (multi-corner/multi-mode) designs, you can specify a set of constraints for the preferred corner with the -corner argument. CTS will scale the constraints and apply them to other corners automatically. If no preferred corner is specified, CTS will apply user constraints to the "slow" corner if it exists, or to the "default" corner otherwise.

The CTS engine buffers a clock network described by the CTS specifications. The primary objective is to build a buffered clock network that satisfies the design rule constraints and has acceptable skew at the root pins defined in the CTS specifications. The secondary objective is to optimize latency, area, and level skew.

- Level skew at a root pin is the difference between the maximum number of cells and the minimum number of cells along any path from the root to a leaf pin.
- Skew at a root pin is the difference between the maximum path delay and the minimum path delay from the root pin to a leaf pin.
- Latency at a root pin is the maximum path delay from the root pin to any leaf pin.

By default, the CTS engine forms clusters of leaves based on their down delay latency (latency of the clock network in the

fanout of the current leaf level). This method is used to minimize the overall CTS latency. After all, the nets in the clock tree are buffered, the cells on the clock tree, including the leaf cells, are legalized.

After CTS balancing, a pass of CTS refinement is used to optimize the clock network described by the specifications. The optimization criterion can be DRC, skew, area or timing. Figure 6 shows a simple example of the clock tree before and after the synthesis pass.

Regardless of which configuration is used to perform the clock tree synthesis, the objective that determines the clock tree quality is achieving a minimum skew and latency. For an ASIC design to perform properly, the skew must be less than the clock period. The clock skew can be expressed as:

$$\delta = t_2 - t_1 \quad (1)$$

Where  $\delta$  is the clock skew between two leaf registers,  $t_2$  and  $t_1$  are the latencies, as shown in Figure 6.

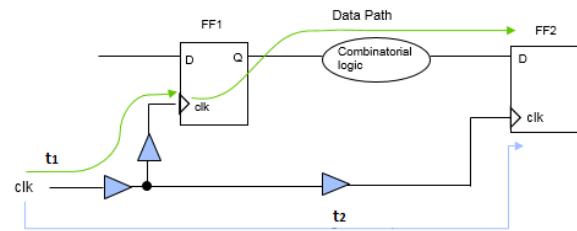


Fig. 6 Skew Between Two Registers Clock Ports

## 2.4 Routing

After the floorplanning, the placement and the CTS are complete, comes the routing to establish the connections between pins on the circuit as specified in the Netlist, while obeying design rule checking (DRC) and design for manufacturing (DFM) requirements. Its objective function is to minimize the wire-lengths and to avoid congestion and timing violations.

To start the routing stage, some requirements need to be satisfied in initial data:

- ✎ Design Need to be properly floor planned and power planned.
- ✎ The design should be Placed and fully legalized.
- ✎ Proper timing constraints must be set.

The Nitro-SoC routing flow consists of four major steps:

### 2.4.1 Clock routing

Clock wires represent the most critical foundation of the circuit functionality. Therefore, they should not be disturbed by any switching signal that can introduce noise on the clock signal and thus change the clock network registers value. When dealing with the clock routing we deal with the problem of assigning appropriate widths to wires in a clock tree to minimize the clock skew, the clock delay, and the sensitivity of the clock Clock wires represent the most critical foundation of the circuit functionality. Therefore, they should not be disturbed by any switching signal that can introduce noise on the clock signal and thus change the clock network registers value. When dealing with the clock routing we deal with the problem of assigning appropriate widths to wires in a clock tree to minimize the clock skew, the clock delay, and the sensitivity of the clock tree to process variations. The constraint on the maximum width of a wire is typically imposed by the available routing resources, whereas the constraint on the minimum wire width depends on the fabrication technology. Moreover, the maximum allowable current density through a wire also provides a lower bound

for the wire width, so that the wire can withstand the wear-out phenomenon caused by electromigration. Note that a long wire may be divided into several segments, and each segment may have different upper and lower bounds [40]. Figure 7 illustrates the circuit after the clock routing step.

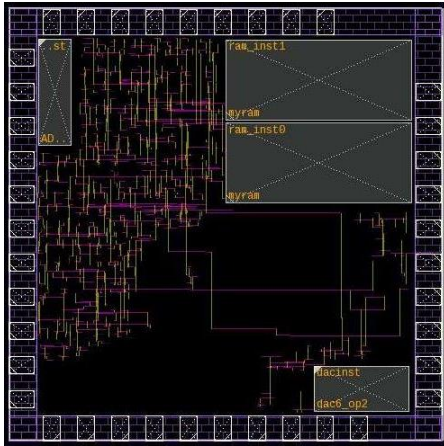


Fig. 7 Integrated circuit after clock routing

## 2.4.2 Global routing

Is invoked first to estimate congestion which allows knowing whether the design is likely to be routable. For accurate results, this estimation takes into account complex resource requirements such as the effect of vias or stacked via patterns, blockages, and staggered macros. Also consider design rule compliance and SI requirements like wire spreading, wire widening, and shielding. Once the GR results are acceptable, we can typically perform several other tasks with the design, such as optimization and clock tree synthesis, before continuing the routing process, this may cause some change in design topology, therefore GR options are available by the tool to repair the nets touched and also to optimize the global routing for better timing/congestion results.

The global routing creates a grid of equally sized square global cells (gcell) and calculates how many available tracks cross every gcell edge. Then, it routes all routable nets using global routes. These routes do not have a real width and do not connect the cell pins; instead, they end at the center of the gcell where the pins must be connected are located. The figure below describes the chip after the global routing. In this figure (Figure 8), each two-circuit cells having the same color are connected in the Netlist.

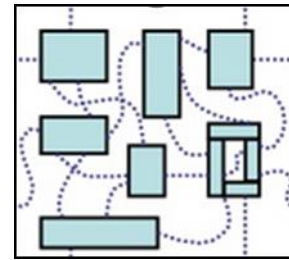


Fig. 8 Global Routing

## 2.4.3 Track routing

Track routing creates an initial detailed routing based on the global route topology. The main goal of the track router is to minimize conflicts between segments in global channels. The track routing engine attempts to honor major DRC rules (spacing and shorts) and non-default rules (NDRs). Any DRC errors remaining after track routing is completed can be cleaned during final routing. The track router generates detailed routing connected to all pins. The interconnections made by this router in figure 9 may use different levels of metals to allow crossings without adverse consequences for the circuit such as shorts.

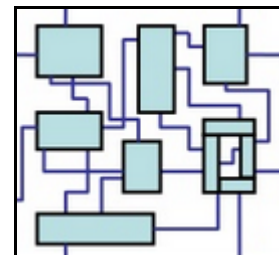


Fig. 9 Track routing

## 2.4.4 Final routing

The final routing engine is a multithreaded engine designed primarily for "search and repair" in existing incomplete or not entirely DRC-clean routing. It mostly follows TR and designed to ensure that the design is DRC/LVS clean. At this point, a variety of manufacturing-related operations can be applied, such as metal fill, antenna fix, and via reduction. Also, another typical application of final routing is ECO routing, which allows to repair the modified nets and even fully routing of new ones. This one is called mainly during post-route Optimization when cells are moved and new buffers inserted. Figure 10 illustrates a circuit.



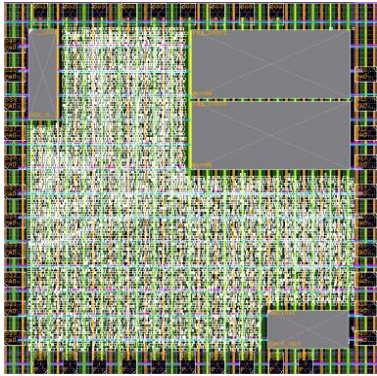


Fig. 10 Integrated circuit after final routing

Final routing corrects routing violations using two types of checks: DRC and LVS.

▪ **Layout vs Schematic (LVS)**

LVS check finds layout violations related to net connectivity. For each specified signal, net verification fetches all pins and routing objects. Then it builds a connectivity graph and checks it. If the net has open pins or more than one routing chain, then open violations are reported.

Global routing checking verifies pin connections only. It does not check for opens in global routing itself but merely checks if it reaches all the pins. A pin is considered unconnected if the closest global routing segment of its net is more than 2 gcells away. A net with no global routing is also reported as an open unless all the pins are confined in the same 2x2 gcell box.

It is a good practice to run `check_lvs` on global routing before proceeding to track routing on a design. If the global router has left unconnected pins behind then the track router is likely to fail.

▪ **Design Rule Check (DRC)**

DRC check finds layout design rule violations based on technology rules, it queries all layout objects, builds polygon shapes, and checks technology rules for them. It determines if the layout satisfies technology rules, if it doesn't, the nets with DRC violations will be rerouted/repared. The most common rules are minimum metal width (figure 11) and spacing between metals (figure 12).

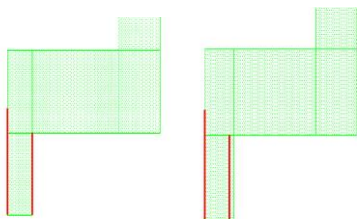


Fig. 11 Example of minimum metal width

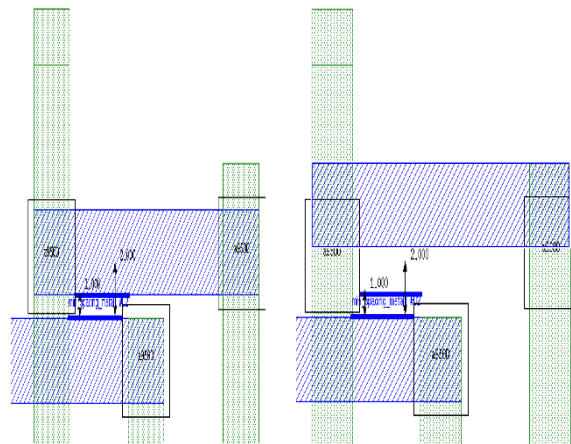


Fig. 12 Example of spacing between metals

### 3. Chip Finishing and Design for Manufacture

Chip finishing prepares the design for tapeout. The chip finishing steps depend on the design requirements, which can include (but are not limited to) meeting manufacturability requirements, improving yield, building in design robustness, and generating data for mask preparation. This step assumes that your design is fully routed and clean.

The following actions could be performed at this stage as needed:

- **Wire editing:** A process of manually changing the existing wire geometries, creating new wires, or both. This is typically used to fix the remaining DRC errors, refine existing routes, or perform spare cell consumption.
- **Filler cells insertion:** You need to fill the gaps in the cell row with filler cells. These are cells in your library that have no active circuits in them, just power and ground wires and NWEELL layers. The purpose of filler cells is to maintain continuity in the rows by adding Vdd and gnd lines and an n-well. The filler cells also contain substrate connections to improve substrate biasing. Typically, filler cells are used to fill any spaces between regular library cells to avoid planarity problems and provide the power and ground electrical continuity. For a design with multi-voltage threshold cells, the filler cell insertion process must take into account the voltage threshold type of adjacent cells, so the tool can select the appropriate filler cell to insert in the gap between two cells.



- **Inserting Decoupling Capacitor Cells:** Decoupling capacitor (dcap) cell insertion smooths the output of power supplies. These cells prevent over-voltage failures by absorbing the extra voltage and under-voltage failures by providing temporary power.
- **Fixing Antennas:** Antenna fixing is performed on long wires to reduce charge accumulation during manufacturing.

The following repair techniques are available to fix antenna violations:

- Jumper insertion (The term “jumper” indicates the point at which the routing wire changes to higher layers.)
  - Diode insertion
  - Spare diode consumption
- **Wire Spreading:** Wire spreading helps to improve timing by reducing coupling capacity and coupling noise. It also improves yield by reducing the probability of critical area defects, such as electrical shorts caused by random dust particles.
- **Reducing the Via Count:** Vias have a susceptibility to failures such as cut misalignment, electromigration, and thermal stress. By reducing the number of vias used in your design, you can enhance both yield improvement and reliability.
- **Replacing Single-Cut Vias:** You can increase reliability and reduce yield loss due to via failures by replacing single-cut vias with types that are more robust. More robust via types include:
  - Redundant (double-cut) vias — Two vias for one connection point
  - Extended vias (line-end extension) — Vias with a larger enclosure on all sides
  - Barvias — Vias that have a non-square cut
- **Inserting Metal Fill:** Metal fill insertion improves surface planarity for each metal layer by inserting metal fill segments to meet specified density requirements. Improved surface planarity helps decrease manufacturing variations that contribute to timing variability. This, in turn, helps increase yield numbers.

### 3.1 Output File Generation

After completing the place and route flow and fixing all timing, DRC, LVS, ERC, and DFM issues, the clean layout, usually represented in the GDSII Stream format, is generated for final physical verification with sign-off tools before it is sent for manufacturing at a dedicated silicon factory. The handoff of the design to the manufacturing

process is called tapeout, even though data transmission from the design team to the silicon fab no longer relies on magnetic tape. Generation of the data for manufacturing is sometimes referred to as streaming out, reflecting the use of GDSII Stream. At this stage, the stage the journey of physical design ends and another journey of chip fabrication start which is out of the scope of this article.

## 4. Conclusion

In this paper, a practical approach that helps and facilitates specialists understanding the physical design flow of cell-based ASICs. To develop the flow, a top-down approach was followed using Nitro-SoC of Mentor Graphics. The main steps of the flow were discussed and illustrated with examples to help the beginners enter the place and route world and to see the global picture before getting specialized in one area, like floorplanning, timing analysis, placement, routing or optimization. Of course, the P&R flows described here only scratches the surface of what the tools can do, its aim is to give confidence to every specialist before embarking on a complex SoC project by demonstrating the key concepts involved in the VLSI chip development process. It is my hope to help readers build a solid foundation for further advancement in this field.

## 5. Perspectives

This paper gave a general description of the multidisciplinary Physical Design Process and shed light on the major stages in a chip PnR flow. More work can be done to describe the metrics to monitor during the process progression, how to measure and compare the Quality of Results (QoR), and also when to decide to go to from a step to the next step in the flow.

## Acknowledgments

This paper was supported by Mentor Graphics Corporation. We thank our colleagues from ICDS division who provided insight and expertise that greatly assisted the research. We thank Dr. Hazem El Tahawy (Mentor Graphics, Managing Director MENA Region) for initiating and supporting this work, Chinnery David (Architect, ICDS P&R Solutions Optimization), Bhardwaj Sarvesh (Group Architect, ICDS P&R Solutions Optimization), for assistance, help and guidelines through the research, and Chafik Ouidiane (Product Validation, ICDS P&R Solutions) for the opportunity to work on such advanced topic.

## References

- [1] Naveed A. Sherwani, "Algorithms for VLSI Physical Design Automation", Springer Science & Business Media, Dec 6, 2012 - Technology & Engineering
- [2] Andrew B. Kahng, Jens Lienig, Igor L. Markov, Jin Hu, "VLSI Physical Design: From Graph Partitioning to Timing Closure", Springer Science & Business Media, Jan 27, 2011 - Technology & Engineering
- [3] Wai-Kai Chen, "Computer Aided Design and Design Automation", CRC Press, Jun 23, 2009 - Technology & Engineering
- [4] Wenjian Yu, Xiren Wang, "Advanced Field-Solver Techniques for RC Extraction of Integrated Circuits", Springer Science & Business, Apr 21, 2014 - Technology & Engineering
- [5] Stephen H. Hall, Howard L. Heck, "Advanced Signal Integrity for High-Speed Digital Designs", John Wiley & Sons, Sep 20, 2011 - Science
- [6] Neil H. E. Weste, David Money Harris, "CMOS VLSI Design: A Circuits and Systems Perspective", Pearson Education India
- [7] Sridhar Gangadharan, Sanjay Churiwala, "Constraining Designs for Synthesis and Timing Analysis: A Practical Guide to Synopsys Design Constraints (SDC)", Springer Science & Business Media, Jul 8, 2014 - Technology & Engineering
- [8] Sung Kyu Lim, "Design for High Performance, Low Power, and Reliable 3D Integrated Circuits" Springer Science & Business Media, Nov 27, 2012 - Technology & Engineering
- [9] Luca Benini, Giovanni DeMicheli, "Dynamic Power Management: Design Techniques and CAD Tools", Springer Science & Business Media, Nov 30, 1997 - Technology & Engineering
- [10] Luciano Lavagno, Grant Martin, Louis Scheffer, "Electronic Design Automation for Integrated Circuits Handbook - 2 Volume Set", Taylor & Francis, Apr 13, 2006 - Technology & Engineering
- [11] Erik Brunvand, "Digital VLSI Chip Design with Cadence and Synopsys CAD Tools", Addison-Wesley, 2010 - Computers
- [12] David Chinnery, Kurt Keutzer, "Closing the Gap Between ASIC & Custom: Tools and Techniques for High-Performance ASIC Design", Springer Science & Business Media, Jun 30, 2002 - Computers
- [13] R. Jacob Baker, "CMOS: Circuit Design, Layout, and Simulation", John Wiley & Sons, Jan 11, 2011 - Technology & Engineering
- [14] Chuan Seng Tan, Kuan-Neng Chen, Steven J. Koester, "3D Integration for VLSI Systems", CRC Press, Apr 19, 2016 - Science
- [15] Yangdong Deng, Wojciech P. Maly, "3-Dimensional VLSI: A 2.5-Dimensional Integration Scheme", Springer Science & Business Media, Sep 8, 2010 - Technology & Engineering
- [16] Cher Ming Tan, Feifei He, "Electromigration Modeling at Circuit Layout Level", Springer Science & Business Media, Mar 16, 2013 - Technology & Engineering
- [17] Sonia Ben Dhia, Mohamed Ramdani, Etienne Sicard, "Electromagnetic Compatibility of Integrated Circuits: Techniques for low emission and susceptibility", Springer Science & Business Media, Jun 4, 2006 - Technology & Engineering
- [18] Harry J.M. Veendrick, "Nanometer CMOS ICs: From Basics to ASICs", Springer, Apr 28, 2017 - Technology & Engineering
- [19] Anantha Chandrakasan, Frank Fox, William J. Bowhill, "Design of High-Performance Microprocessor Circuits", Wiley, 2001 - Technology & Engineering
- [20] Mustafa Celik, Larry Pileggi, Altan Odabasioglu, "IC Interconnect Analysis", Springer Science & Business Media, May 8, 2007 - Technology & Engineering
- [21] Siva G. Narendra, Anantha P. Chandrakasan, "Leakage in Nanometer CMOS Technologies", Springer Science & Business Media, Mar 10, 2006 - Technology & Engineering
- [22] Christian Piguuet, "Low-Power CMOS Circuits: Technology, Logic Design and CAD Tools", CRC Press, Nov 1, 2005 - Technology & Engineering
- [23] Nikhil Jayakumar, Suganth Paul, Rajesh Garg, "Minimizing and Exploiting Leakage in VLSI Design", Springer Science & Business Media, Dec 2, 2009 - Technology & Engineering
- [24] William W. Hager, Shu-Jen Huang, Panos M. Pardalos, Oleg A. Prokopyev, "Multiscale Optimization Methods and Applications", Springer Science & Business Media, Jun 18, 2006 - Mathematics
- [25] Volkan Kursun, Eby G. Friedman, "Multi-voltage CMOS Circuit Design", John Wiley & Sons, Aug 30, 2006 - Technology & Engineering
- [26] Maurizio Palesi, Masoud Daneshtalab, "Routing Algorithms in Networks-on-Chip", Springer Science & Business Media, Oct 22, 2013 - Technology & Engineering
- [27] Prashant Saxena, Rupesh S. Shelar, Sachin Sapatnekar, "Routing Congestion in VLSI Circuits: Estimation and Optimization", Springer Science & Business Media, Apr 27, 2007 - Technology & Engineering
- [28] J. Bhasker, Rakesh Chadha, "Static Timing Analysis for Nanometer Designs: A Practical Approach", Springer Science & Business Media, Apr 3, 2009 - Technology & Engineering
- [29] Ashish Srivastava, Dennis Sylvester, David Blaauw, "Statistical Analysis and Optimization for VLSI: Timing and Power", Springer Science & Business Media, Jun 21, 2005 - Technology & Engineering
- [30] Gi-Joon Nam, Jingsheng Jason Cong, "Modern Circuit Placement: Best Practices and Results", Springer Science & Business Media, Aug 26, 2007 - Technology & Engineering
- [31] J. Bhasker, Rakesh Chadha, "Static Timing Analysis for Nanometer Designs: A Practical Approach", Springer Science & Business Media, Apr 3, 2009 - Technology & Engineering
- [32] K. J. Nowka et al., "A 32-bit PowerPC system-on-a-chip with support for dynamic voltage scaling and dynamic frequency scaling," in IEEE Journal of Solid-State Circuits, vol. 37, no. 11, pp. 1441-1447, Nov 2002.
- [33] K. Roy, S. Mukhopadhyay and H. Mahmoodi-Meimand, "Leakage current mechanisms and leakage reduction techniques in deep-submicrometer CMOS circuits," in Proceedings of the IEEE, vol. 91, no. 2, pp. 305-327, Feb 2003.
- [34] Ajit Pal, "Low-Power VLSI Circuits and Systems", Springer, Nov 17, 2014 - Technology & Engineering

- [35] Gary K. Yeap, "Practical Low Power Digital VLSI Design", Springer Science & Business Media, Dec 6, 2012 - Technology & Engineering, Pages 175-178.
- [36] B. N. B. Ray, S. Das, K. Hazra, N. Patra and S. K. Mohanty, "An Optimized HPWL Model for VLSI Analytical Placement," 2015 International Conference on Information Technology (ICIT), Bhubaneswar, 2015, pp. 7-12.
- [37] Mysore Sriram, Sung-Mo (Steve) Kang, "Physical Design for Multichip Modules", Springer Science & Business Media, Dec 6, 2012 - Technology & Engineering
- [38] Sung Kyu Lim, "Practical Problems in VLSI Physical Design Automation", Springer Science & Business Media, Jul 31, 2008 - Technology & Engineering, Pages 102-103.
- [39] Bing Lu, Ding-Zhu Du, S. Sapatnekar, "Layout Optimization in VLSI Design", Springer Science & Business Media, Jun 29, 2013 - Computers
- [40] Laung-Terng Wang, Yao-Wen Chang, Kwang-Ting (Tim) Cheng, "Electronic Design Automation: Synthesis, Verification, and Test", Morgan Kaufmann, Mar 11, 2009 - Technology & Engineering
- [41] Chang, C.-C., Cong, J., Pan, Z., and Yuan, X. (2003). Multilevel global placement with congestion control. IEEE Trans. on Computer-Aided Design of Integrated Circuits and Systems, 22(4):395-409.

**Chentouf Mohamed**, was an FPGA R&D engineer with Zodiac Aerospace, he is now a product validation engineer with Mentor Graphics Corporation/ICDS Division- Rabat, Morocco.

**Alaoui Ismaili Zine El Abidine**, is an assistant professor and researcher in ENSIAS/Telecommunications and Embedded Systems Team, University Mohammed V- Rabat, Morocco.