International Journal of Hybrid Information Technology
Vol.9, No.1 (2016), pp. 279-290
http://dx.doi.org/10.14257/ijhit.2016.9.1.24

A Study of Floorplanning Challenges and Analysis of macro placement approaches in Physical Aware Synthesis

Shivani Garg[1] and Neeraj Kr. Shukla[2]

[1, 2] VLSI Design Group, Department of EECE, ITM University, Gurgaon (Haryana) India

Abstract

The macro placement is a governing factor in design flow in terms of timing criticality and congestion metrics. Also tape-out schedules are affected because of quality of macro placement or floorplanning. As hierarchical approach is followed for design closure today, there are multiple macro iterations back and forth between synthesis and implementation tool until the design meet all timing and design constraints. Traditional flow takes much time to figure out the best floorplan in terms of timing, Quality of results (QoR) and congestion. Now, physical aware synthesis gives a user an opportunity to cut the implementation time at later stages. The approach used here is RTL level floorplanning to enhance the quality of floorplan and also save multiple iterations. In one shot we get automatic floorplan initially generated by tool and then used by Place and Route (P&R) team for further processing. Also a rough estimate of wire delays and routes to calculate parasitic value, hence delays is obtained. There are multiple ways of placing macros in a floorplan which varies with shape of die and core utilization of design. So when multi macro placement approaches are applied simultaneously, then one can choose the best floorplan in terms of QoR and design metrics and then take it to implementation tool. After studying floorplan and macro placement challenges, there is a need to reduce cycle time between synthesis and implementation tool. In this work, different macro placement approaches are applied on design for different shapes of die (rectangular and rectilinear) using various parameters and then analyzed the timing, design metrics and congestion of various approaches. The results obtained show that every macro placement approach has different effect for different types of die on timing, congestion and power. Thus we can easily automate floorplanning for different shapes of die and reduce cycle time from months to few weeks. As checked on a design with 500k instances and 18 macros, with utilization of 59% if rectangular die is chosen the approach 3rd gives best result in congestion of 0.1% and the cycle time reduced from weeks to days.

Keywords: *congestion, die, floorplanning, macro placement, QoR*

1. Introduction

As the interconnect delay dominates chip performance due to scaling of VLSI Technology. It is not necessary that if a design is meeting timing after logic synthesis then it will meet same after place and route because interconnect delays come into picture then. Physical synthesis is an important feature of modern VLSI methods. Physical synthesis begins with a mapped netlist generated by logic synthesis. The netlist generated after logical synthesis describes the logical connections of the components such as macros, input output blocks and pins. Physical synthesis generates a netlist which is optimized and a physical layout which provides a basic estimation of placement and routing in implementation tool.

ISSN: 1738-9968 IJHIT

Physical aware synthesis is generation of a floorplan by P&R (Place and Route) tool and feeding back of floorplan DEF to synthesis tool and hence physical layout estimation is used instead of wireload modelling estimation. The time taken in number of iterations from synthesis to P&R and vice-versa reduces and timing improves resulting in a better QoR. The Floorplanning step mainly includes macro placement, partitioning of design, input-output ports placement and power planning. The design parameters such as power, area, timing and performance need to be considered during floorplanning. These estimations are calculated at every step based on the feedback from the implementation team, IP owners, and RTL designers. In case of hierarchical designs, the number of iteration increase as all the blocks are joined at top level at a later stage. A floorplan is a proper arrangement of macros, blocks, power grid (rings or stripes) and input-output ports. Figure 1.1 shows an example of floorplan.

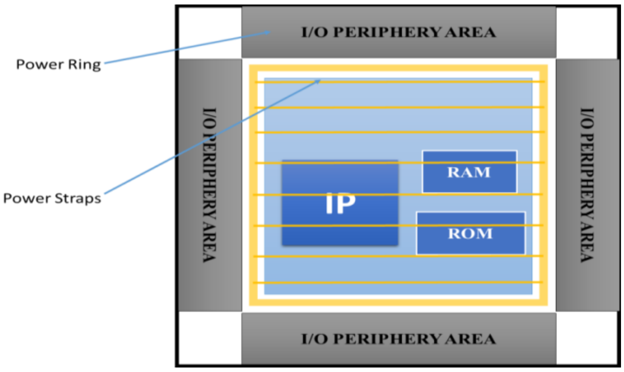


Figure 1.1. A Floorplan View

The paper is organized in six sections. The section II provides the literature review of the related work. In section III & IV floorplanning and macro placement challenges are discussed respectively. The analysis and results are shown in section V. The section VI provides conclusion and future scope.

2. Literature Review

An estimation model and a congestion aware floorplan have been discussed in [3], for 3D ICs. This model is based on analysis after considering Through Silicon Vias (TSV) location and the congestion aware floorplan uses multiple approaches to evaluate results. The approach discussed in [4] helps designers identify congestion related issues. This approach allows macro placement, to analyze and compare congestion. The paper [5] discussed challenges like: Automatic generation of data-path layout, More integrated timing and power optimizations, Layout-friendly high-level synthesis Lithography-aware physical synthesis and Quantifying the impact. A technique which improves the placement solution for routing has been discussed here. This technique achieves the placement and routing co-optimization to handle variation, contamination, and defect [1].

Placement algorithms optimize signal-net switching power. They ignore clock-network switching, responsible for more than 30% of total power. New techniques and a methodology to optimize total dynamic power during placement have been discussed for large IC designs [16]. During floorplanning, two approaches have been discussed to allocate level shifter regions which reduce time complexity [14]. The work discusses a unified method to handle alignment and cluster constraints on sequence pair representation which significantly reduces the solution space and speeds up the algorithm [15].

A new floorplan representation [17], the circular-packing trees which can resolve the problem of macro placement has been discussed here. This floorplan representation can pack movable macros toward corners or preplaced macros along chip boundaries circularly. This in turn optimizes macro orientations for better wirelength and routing congestion. This work used design space exploration of low-power adders for comparative analysis of Physical layout Aware Synthesis and Place and Route estimation flow [18]. In [19], to handle large scale mixed size placement an effective algorithm is discussed. Four steps for the flow are discussed which includes clustering of objects into blocks , floor planning on blocks , optimizing wire length to shift the blocks within the chip and placing the remaining objects incrementally keeping the big macro location as fixed.

Floorplanning [21] techniques handle macros effectively but sometimes do not scale to hundreds of thousands of placeable objects. In this floorplanning techniques are combined with placement techniques in a design flow that solves the placement problem. The techniques can also be used to guide circuit designers who prefer manual macro placement. The discussed flow used an arbitrary black-box standard cell placer to get an initial placement and then removes overlaps using a floorplanner at initial stage. The wirelength improvements of 10%-50% have been observed. In [23], in order to reduce the dynamic power a new design flow was presented for gated clock tree synthesis. An approach is presented for the optimum power saving during shut off mode and full operation mode on clock tree.

In [24], two well-known forces directed algorithm namely KK and FR is developed to present a module placement tool. Feasibility of combining these two algorithms into one placement tool is explored. Compared to that of current academic placement tool remarkable results were observed [24]. In order to perform the multiple optimizations on large design partitions an integrated transformation system is developed. A combination of cloning and register placement and physically-aware register retiming is used [25]. For analytical mixed-size placement in order to handle macro orientation a new rotation force is presented. During placement a cross potential model is also discussed to increase the rotation freedom [26].

In [27], to shorten the wire delay for dynamically reconfigurable processor, two iterative synthesis techniques are presented in between the place and route tool and a high level synthesizer (HLS). In the conclusion the wire delay has shortened the increased synthesis time with only a slight increase of delay [27]. In [28] techniques to solve the challenges from large-scale mixed sized designs of the circuits with wire length optimization are used. For the modern circuit designs various challenges and opportunities on routability and macro placement, timing, power and thermal- driven optimization of placement are induced for future research. In this paper automated floor planning methods are addressed which is essential for the efficient design space exploration. A few techniques that can improve the existing tools for early floor planning are also stated [29]. An approach for the floorplanning of rectangular blocks with various constraints on their connection and dimension is used such that the total wire length and area of the resulting floorplan are minimized [30].

A flexible and robust mixed size legalization scheme has been introduced to remove the overlap between standard cells which also preserves the legalization of macros. To reduce wirelength, a technique called as Sliding-window based cell swapping is applied in the end [34]. An efficient algorithm which works on a novel augmented constraint graph to remove overlap in the presence of fixed location, spacing and boundary constraints that are predefined or imposed on macro cells earlier has been presented in [35]. One of the operand isolation schemes is adopted to reduce switching in datapath which was causing overhead in terms of delay, power, and area. Isolation techniques which are based on supply gating reducing the overheads associated with isolating circuitry are presented [37]. In [38] Rent's rule has been used for estimation of interconnect power consumption. Accordingly, the interconnect power reduction is 72.9% and overall power reduction is 56.0% with 44.4% area overhead as compared to area optimized circuits.

A macro-cell placement algorithm uses a model for accurate delay analysis by handling path-delay constraints. The algorithm is flexible in physical design flow since it is iterative and incremental and gives better results in terms of path delay [39]. In [40] a complete chip design method is presented which incorporates a soft-macro placement and resynthesis method for area and timing improvements [40]. Three technologies which governs logic synthesis and physical layout optimization including early floorplanning, layout-driven logic synthesis, and post- layout resynthesis are discussed in [41].

3. Floorplanning Challenges

Many industries are moving towards lower technology nodes due to increase in demand for SOC speed and performance. The floorplanning affects the design cycle time in entire design flow from design planning to implementation. [6]. Some challenges such as large design sizes, increasing macro count, timing/power estimations, region shaping and pin assignment, predefined placement locations, macro orientations and pin positions , simultaneous standard cell and macro placement, congestion and timing-driven placement, reducing cycle time, high performance and low power targets are increasing for a floorplan designer. Sometimes, the macros need to be legalized to the periphery of the design in order to provide maximum area for the standard cell placement. In some designs the requirement is such that the macros need to be arrayed so that the address and data lines have optimum distance to the logic which is connected to it. This is a manual and iterative process between synthesis and implementation which has a large impact on cycle time in designs [7].

As the design approach is now shifting towards hierarchical closure which depends on the architectural requirements, tool limitations due to higher gate counts, late IP deliverables, and different power modes in an SOC. The hierarchically partitioned blocks are first implemented in terms of placement, routing, timing, and noise closure .Then they are merged together into top level at later stage which causes back and forth iterations between synthesis and implementation to optimize floorplan. Sometimes there are cases of many top-level nets detouring across the boundary of these partitions which on a later stage results in timing violations, wasted routing resources due to large wirelength of nets and buffer insertion to avoid design-rule violations on these nets. Ultimately, results in increased power consumption and increased placement density [6].Some of the common issues [10, 11] which arise for physical design teams are:

3.1. IO Pad Placement

Sometimes we don’t have proper predefined input output constraints from the packaging teams which can result in suboptimal order and location of input output pads. This causes a problem for chip level timing closure. When detailed placement and routing is complete, then designers have to iterate back and forth to fix IO pad locations and again perform placement and routing. The unpredictable design schedules causes risk of project delivery in time.

3.2. Macro Placement

Achieving optimal macro placement is a challenge for SoCs with large number of cores and memories. Earlier macro placement was more of manual because the number of macros was very less. But now there is a need of automated floorplan by a tool before manual refinement by designer. Tools use to cluster macros based on wire length but there are other metrics to be considered including critical path timing, multi voltage power domains and reducing congestion. The challenge lies in the fact that designers need better seed placement and a powerful macro placement editing that can assist in spacing and alignment of macros.

3.3. Region Shaping and Pin Assignment

Once the regions are determined, the regions or partition pins are assigned. The challenge is to create pin placements that suits all design criteria but out of all, the timing requirements between blocks for all analysis modes and corners are critical. Pin assignments are based on wire length mainly as determined by Steiner-based estimations. Poor pin assignments can be an issue for inter-partition timing paths which may lead to costly sign-offs between the block level and chip level at later stage in the design cycle.

3.4. Timing and Power Estimations

One of the main objectives of floorplanning stage is to have accurate estimations of timing, power, and area as quickly as possible. The timing analysis should include as many corner cases as is feasible that does not mean it should be sign-off accurate. The floorplanning timing estimations are needed for all the timing scenarios which may be later needed at timing sign-off stage. To avoid timing criticality, sometimes memories needs to be placed near the core area. This in turn places the related logic near it reducing the wire buffering and slew and slack. It also impacts congestion. Power estimations at floorplanning stage should have a good correlation with more detailed analysis later in the flow. If there are inaccurate power estimates, there can be errors which can cause difficulty in closure. Multiple power meshes are there in multi-Vdd designs, so at floorplanning stage the power domains must be created and connected properly.

3.5. Creation of stacks

When the design contains multiple memories, stacks need to be created. Memories are of different aspect ratios and different pin densities which results in complexity of stack creation. One of the challenges in stacking is that smaller memories have pin counts similar to bigger ones. For better routing accessibility and reducing congestion, smaller memories are kept closer to the core area.

4. Macro Placement Challenges

As we know, macro placement can be manual or automatic. Manual macro placement is more efficient when there are few macros to be placed and logical connectivity of all is known. Automatic macro placement is more needed if there is not enough information for initial macro placement and/or the number of macros is large. During the macro placement step, the slicing tree is used by the floorplan algorithm for slicing floorplan and to define routing channels between the blocks. This should ensure minimum area between blocks and the channel definition should be properly done [12].

Macro placement depends on factors including fly lines**,** ports communications, macro's are placed at boundaries, macro grouping (logical hierarchy), spacing between macro's, macro alignment, notches avoiding, macro’s orientation, placement and routing blockages [20]. As the number of logic gates and associated hard macro instances increases manual macro placement becomes nearly impossible. Therefore user needs design automation solutions to speed-up hard macro placement. Some of the macro placement challenges are as follows:

4.1. Increasing Macro Count

With shifting of technology towards lower nodes, there is an increase in number of macros of varying aspect ratios to be packed into a smaller which makes it difficult to pack them tightly without wasting silicon area. There is a big challenge in identifying the optimal solution through whole place and route flow. Also, the increase in design size, macro counts, and macro shapes leads to longer runtimes and more design cycle time which needs to be controlled [7].

Some algorithms used for automatic macro placement considers that macros are connected to each other by nets. In hierarchical flows, macros are grouped on the basis of connectivity, clock generation logic or power domains. Also, floorplans are partitioned into functional groups which put physical constraints on placement of standard cells and macros. Poorly shaped regions and broken logical hierarchy may contribute to delayed schedules and lack of optimization in terms of timing, power and area.

4.2. Predefined Placement Locations

Sometimes there are predefined locations of input output ports provided by packaging team like some hard macros must be placed next to specific IO cells. Then the pins of these hard macros must align and be close to specific I/O cells to allow wide-wire connections. Generally, macros are pre-placed in specific locations within the core area, and then their placements are fixed so they cannot be moved by automated placement algorithms [10]. However, this approach of pre-placing and fixing hard macro locations produces placement blockages for the rest of the design logic. Thus a rectilinear area is available now for placing standard cells and placement algorithms need to be capable of handling rectilinear placement areas.

4.3. Macro orientations and pin positions

In many placement algorithms, standard cells usually are assumed to be all the pins at the center because macros are very large in size as compared to standard cells. The length of a net between cells is simply the Manhattan distance between their centroid. So now optimization of total wire length becomes a problem in respect to distance calculation. If the pins are at different positions from center, then macro orientations play a significant role. Depending on the orientation of the macro, the lengths of the nets connecting to the macros can be different which further complicates the optimization formulation.

4.4. Standard Cell and Macro Placement Simultaneously

Earlier, macro placement was done in two stages: manual placement of macros pushing to edge of block/chip and fixing of the macro locations, then followed by placement of the standard cells which assume the fixed macros to be placement and routing blockages for standard cells. But during wire length optimization, both macros and standard cells are not considered simultaneously. The pushing of macros to edges may result in long routes. Therefore the timing results are sub-optimal.

4.5. Timing and congestion issues due to macros

The congestion due to hard macros is because it acts as placement and routing blockage since it blocks first three to four layers of routing. Also the hard macro pins needs a specific number of tracks causing congestion. Automatic estimation of required tracks is needed to improve congestion. Reduction in total wire length and the length of critical nets is needed to improve timing. Timing estimation should consider detouring of routes which takes place because of macros.

4.6. Wide variety of sizes and shapes

Placing macros of varying sizes and aspect ratios without good optimization engines can result in division of placement and routing space causing design to be unroutable. When there are narrow columns between macros or closed space then it may cause difficulty in routing.

4.7. Reduce Cycle Time

Key to meeting design schedule requirements is the ability to produce results quickly. The large numbers of hard macros are not optimized consistently in every iteration manually which results in long cycle time to obtain optimal configuration or floorplan.

5. Analysis and Results

After studying floorplan and macro placement challenges, there is a need to reduce cycle time between synthesis and implementation tool. There are multiple kind of approaches for macro placement considering factors such as macro legalization, macro grouping, avoiding notch, macro alignment, refining partitions, distance between two macros, halos around macros, orientation of macros and placement/routing blockages.

The objectives and sub objectives of work are:

1. To apply different macro placement approaches on design for different shapes of die (rectangular and rectilinear) using various parameters.
2. Analyze the timing, design metrics and congestion of various approaches.

A data flow driven floorplan directly from RTL that concurrently optimizes for timing, power, area and physical constraints is feed forward as initial guidance to place & route teams. The Figures 2 & 3 shows the general synthesis and floorplan flow and Implemented flow respectively.

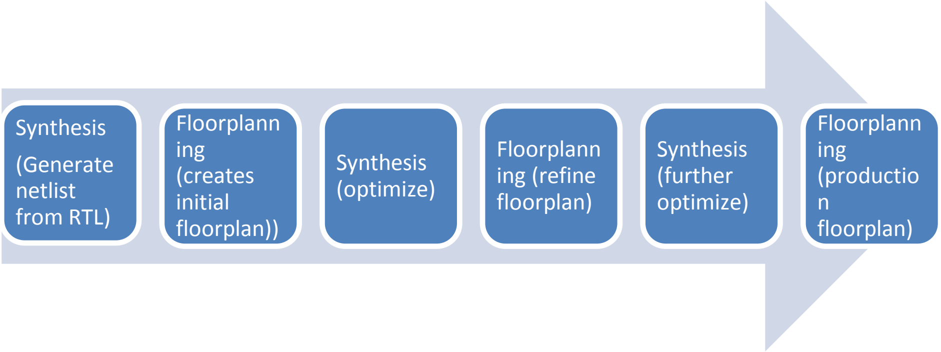

Synthesis (Generate netlist from RTL)
Floorplanning (creates initial floorplan))
Synthesis (optimize)
Floorplanning (refine floorplan)
Synthesis (further optimize)
Floorplanning (production floorplan)

Figure 3. General Synthesis and Floorplan Flow

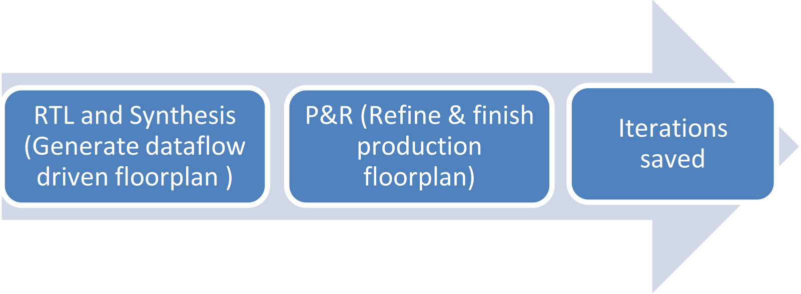

RTL and Synthesis (Generate dataflow driven floorplan)
P&R (Refine & finish production floorplan)
Iterations saved

Figure 4. Implemented Flow

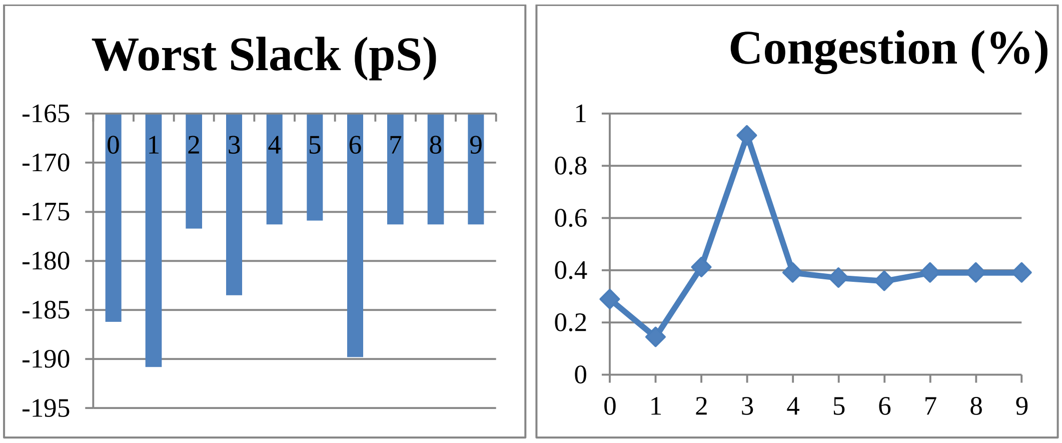
A sample design is chosen to perform analysis of different macro placement approaches. Four types of die shapes are used to observe the behavior of multiple approaches. There are ten kinds of macro placement approaches based on factors including fly lines, ports communications, macro's are placed at boundaries, macro grouping (logical hierarchy), spacing between macro's, macro alignment, notches avoiding, macro’s orientation, placement and routing blockages. Below is the analysis of QoR and design metrics for all approaches. The designer can pick whatever feels best floorplan and take it to implementation tool. This will reduce number of iterartions back and forth between synthesis and implementation tool.

5.1. Rectangular die

A rectangular die is used in this case with utilization 59% approximately The Table 1 shows the comparison of congestion, timing QoR and power varying with each kind of macro placement. The graphs (Fig 5) shows congestion, WNS, TNS and Total Power consumption variations w.r.t various macro placement approaches.

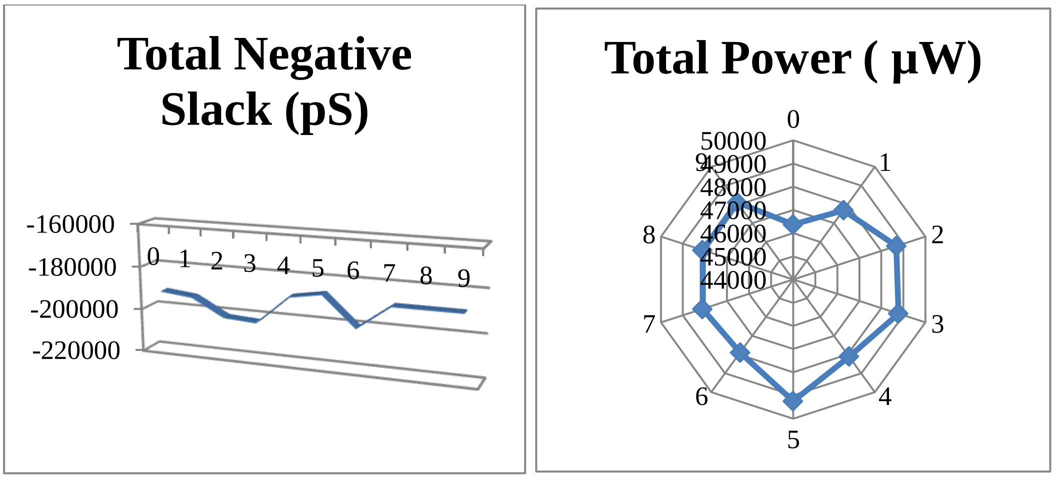
Table 1. Comparison of QoR for Multi-Approach Macro Placement for Rectangular Die

Macro Placement Approach	Congestion (% overflow)	Worst Negative Slack (ps)	Total Negative Slack (ps)	Total Power (uW)
0	0.289403	-186.2	-191848	46373.97
1	0.144116	-190.8	-193402	47689.4
2	0.411825	-176.7	-201532	48675.66
3	0.916702	-183.5	-202467	48775.18
4	0.390684	-176.3	-189146	48107.69
5	0.370876	-175.9	-186557	49249.63
6	0.35868	-189.8	-200442	47895.5
7	0.390684	-176.3	-189146	48107.69
8	0.390684	-176.3	-189146	48107.68
9	0.390684	-176.3	-189146	48107.69



(a)

(b)



(c) (d)

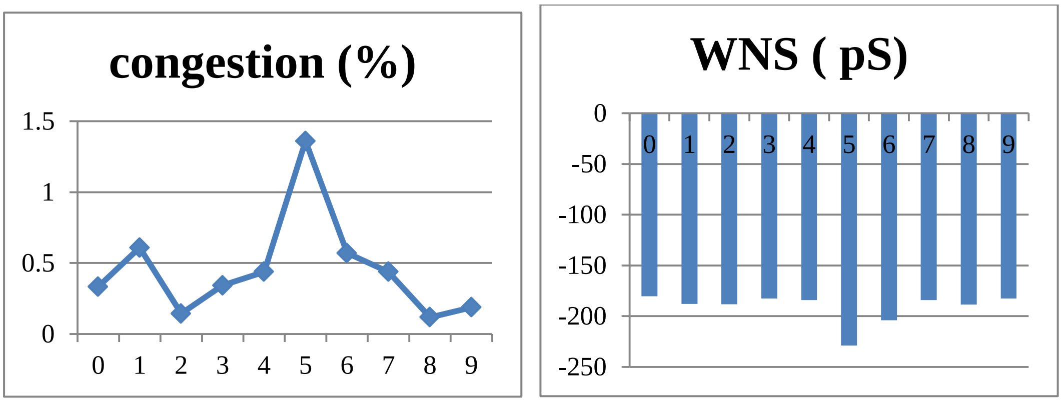
Figure 5. Variations with Respect to Various Macro Placement Approaches (a) Congestion (b) WNS (c) TNS (d) Total Power Consumption

5.2. Rectilinear Die (L-Shape)

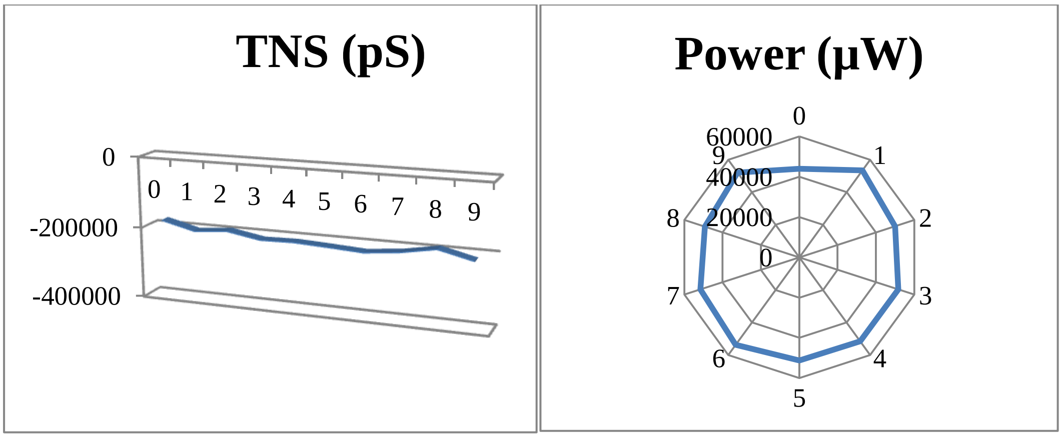
The Table 2 shows the comparison of congestion, timing QoR and power varying with each kind of macro placement. The graphs (Fig 6) shows congestion, WNS, TNS and Total Power consumption variations w.r.t various macro placement approaches.

Table 2. Comparison of QoR for Multi-Approach Macro Placement for Rectilinear die-L

Macro Placement Approach	Congestion (% overflow)	Worst Negative Slack (ps)	Total Negative Slack (ps)	Total Power (uW)
0	0.333775	-180.4	-181053	44003.23
1	0.609029	-188	-204321	53339.56
2	0.143939	-188.2	-195935	50009.36
3	0.341958	-182.5	-212123	51692.96
4	0.437779	-184.2	-209928	51572.3
5	1.35947	-229.1	-214538	51233.77
6	0.570334	-204.1	-219916	53753.56
7	0.437779	-184.2	-209928	51572.3
8	0.117789	-188.7	-192464	49233.25
9	0.188339	-182.5	-215152	52139.55



(a) (b)



(c) (d)

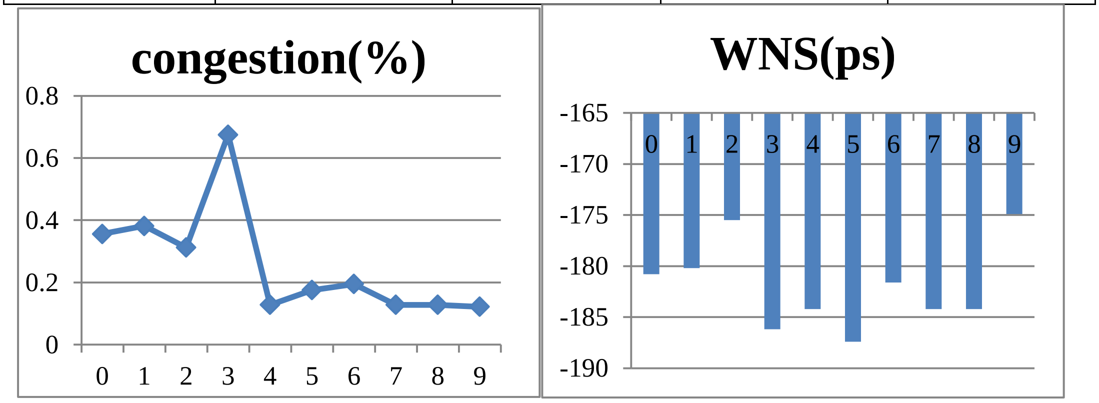
Figure 6. Variations with Respect to Various Macro Placement Approaches (a) Congestion (b) WNS (c) TNS (d) Total Power Consumption

5.3. Rectilinear Die (U-Shape)

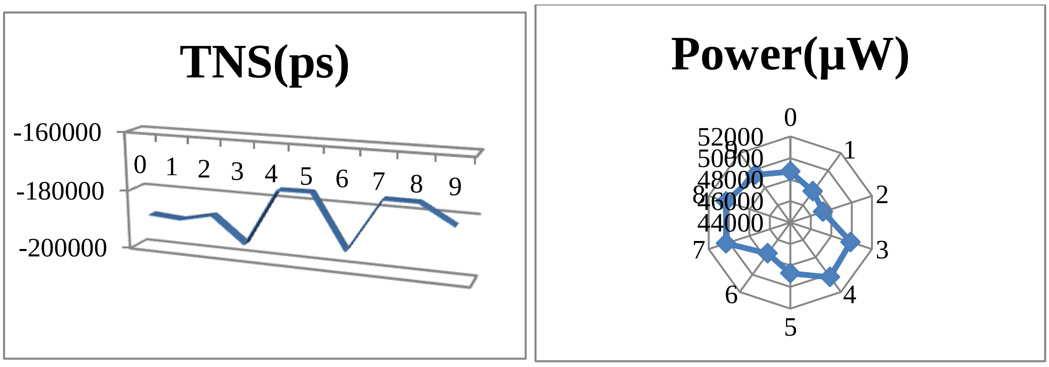
In this case rectilinear die of U-shape is used with utilization of approximately 49%. The Table 3 shows the comparison of congestion, timing QoR and power varying with each kind of macro placement. The graphs (Figure 7) show congestion, WNS, TNS and Total Power consumption variations w.r.t various macro placement approaches.

Table 4.3. Comparison of QoR for Multi-Approach Macro Placement for Rectilinear Die-U

Macro Placement Approach	Congestion (% overflow)	Worst Negative Slack (ps)	Total Negative Slack (ps)	Total Power (uW)
0	0.355747	-180.8	-188551	48744.23
1	0.381626	-180.2	-189317	47574.58
2	0.312003	-175.5	-186816	47217.29
3	0.675045	-186.2	-195554	49908.05
4	0.128027	-184.2	-176272	50292.61
5	0.175622	-187.4	-176064	48732.78
6	0.19481	-181.6	-194216	47567.34
7	0.128027	-184.2	-176272	50292.61
8	0.128027	-184.2	-176272	50292.61
9	0.122105	-174.9	-182709	49493.41



(a) (b)



(c) (d)

Figure 7. Variations with Respect to various macro placement approaches (a) Congestion (b) WNS (c) TNS (d) Total Power consumption

6. Conclusion

From previous discussions, results and analysis, it can be concluded that the macro placement has a huge impact on timing and congestion in design flow. Also the floorplanning is a very critical step in complete design implementation flow. A good or bad floorplan may be a determining factor in tape-out schedules. When hierarchical approach is followed for design closure, there are multiple macro iterations back and forth between synthesis and implementation tool until the design meet all timing and design constraints. RTL level floorplanning enhances the quality of floorplan and also saves multiple iterations. In one shot we get automatic floorplan initially generated by tool and then used by P&R team for further processing. So when multi macro placement approaches are applied simultaneously, then one can choose the best floorplan in terms of QoR and design metrics and then take it to implementation tool. The results show that every macro placement approach has different effect for different types of die on timing, congestion and power. Thus we can easily automate floorplanning for different shapes of die and observe results. As checked on a design with 500k instances and 18 macros, with utilization of 59% if rectangular die is chosen the approach 3rd gives best result in congestion of 0.1% and the cycle time reduced from weeks to days.

References

[1] Chen Liao and Shiyan Hu, “Physical-Level Synthesis for Digital Lab-On-a-Chip Considering Variation, Contamination and Defect,” IEEE Transactions on Nanobioscience, vol. 13, no. 1, pp. 3-11, March 2014

[2] Chia-Chun Lin, Susmita Sur-Kolay and Niraj K. Jha, “PAQCS: Physical Design-Aware Fault-Tolerant Quantum Circuit Synthesis,” IEEE Transactions on Very Large scale Integration (VLSI) Systems, 2014.

[3] Wenrui Li , Jaehwan Kim and Jong-Wha Chong, “A Novel Congestion Estimation Model and Congestion Aware Floorplan for 3D ICs”, in proc. of International Conference on Innovation, Management and Technology Research (ICIMTR),IEEE ,pp. 199-204, 21-22 May, 2012

[4] Ankit Goyal, “Evaluating Macro Placement in an SoC Block Based on a Congestion Estimate,” in proc. of IEEE Canadian Conference on Electrical and Computer Engineering (CCECE), 2012

[5] Igor L. Markov, Jin Hu and Myung-Chul Kim, “Progress and Challenges in VLSI Placement Research,” ACM, 2012.

[6] “Floorplanning: concept, challenges, and closure,” URL: http://www.edn.com/design/integrated-circuit-design/4396580/Floorplan

[7] Arvind Narayanan and James Forsyth, “Automatic Macro Placement for Advanced Nodes,” Mentor Graphics.

[8] Golshan K., “Floorplanning,” Physical Design Essentials : An ASIC Design Implementation Perspective, pp 37-69, 2007.

[9] Mike Clarke, Diego Hammerschlag, Matt Rardon and Ankush Sood, “Eliminating Routing Congestion Issues with Logic Synthesis,” Cadence Design Systems

[10] Ratan Devpura, Niraj Jani, Jignesh Panwala and Parth Lakhiya, “Physical Design Implementation Challenges in Highly Memory-Intensive Design in 40nm,” Electronics for You, pp. 36-38, July 2013.

[11] “Advanced Floorplanning with Olympus-SOC for Fast and Reliable Design Closure,” www.mentor.com

[12] “Hard Macro Placement in Complex SoC Design,” URL: http://www.soccentral.com/

[13] "Automatic Macro Placement for Olympus-SoC," User's Manual, Software Version 2013.1, Mentor Graphics, July 2013

[14] Jai-Ming Lin, Wei-Yi Cheng, Chung-Lin Lee, and Richard C.J. Hsu, "Voltage Island-Driven Floorplanning Considering Level shifter Placement",IEEE, pp.443-448, 2012

[15] Shantesh Pinge, Rajeev K Nain and Malgorzata Chrzanowska-Jeske, "Fast Floorplanning With Placement Constraints", IEEE, 2013

[16] Dong-Jin Lee and Igor L. Markov, "Obstacle-Aware Clock-Tree Shaping During Placement", IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, vol. 31, no. 2, pp. 205-216, February 2012

[17] Yi-Fang Chen, Chau-Chin Huang, Chien-Hsiung Chiou, Yao-Wen Chang, and Chang-Jen Wang, "Routability-Driven Blockage-Aware Macro Placement"

[18] Ivan Ratkovi, Oscar Palomar, Milan Stani, Osman Unsal, Adrian Cristal, Mateo Valero, "Physical vs. Physically-Aware Estimation Flow: Case Study of Design Space Exploration of Adders", in proc. of IEEE Computer Society

[19] Jackey Z. Yan, Natarajan Viswanathan, and Chris Chu, "Handling Complexities in Modern Large-Scale Mixed-Size Placement", pp.436-441

[20] "Macro Placement (Guide Lines)," http://vlsipd.blogspot.in/2013/06/macro-placement-guide-lines.html

[21] Saurabh N. Adya and Igor L. Markov, "Consistent Placement of MacroBlock Using Floorplanning and StandardCell Placement", ISPD'02, April 7-10,2002

[22] Sanjiv Taneja, "Physical-Aware, High-Capacity RTL Synthesis for Advanced Nanometer Designs," Cadence Design Systems, March 4-6, 2103.

[23] Teng Siong Kiong and Dr. Norhayati Soin, "Physical Aware Low Power Clock Gates Synthesis Algorithm for High Speed VLSI Design"

[24] Meththa Samaranayake, Helen Ji and John Ainscough, "Module Placement Based on Hierarchical Force Directed Approach", in proc. of International Conference on Signals, Circuits and Systems, IEEE, pp. 1-6, 2009

[25] David A. Papa, Smita Krishnaswamy, and Igor L. Markov, "SPIRE: A Retiming-Based Physical-Synthesis Transformation System",IEEE, pp. 373-380,2010

[26] Meng-Kai Hsu and Yao-Wen Chang, "Unified Analytical Global Placement for Large-Scale Mixed-Size Circuit Designs",IEEE, pp. 657-662,2010

[27] Takao Toi , Takumi Okamoto, Toru Awashima, Kazutoshi Wakabayashi,and Hideharu Amano, "Wire Congestion Aware Synthesis for a Dynamically Reconfigurable Processor", IEEE, pp. 300-303, 2010

[28] B. Sekhara Babu, Rajine Swetha R, Dr. Sumithra Devi K. A., "Comparison of Hierarchial Mixed-Size Placement Algorithms for VLSI Physical Synthesis", in proc. of International Conference on Communication Systems and Network Technologies,IEEE, pp. 430-435,2011

[29] Jeonghee Shin, John A. Darringer, Guojie Luo, Merav Aharoni, Alexey Y. Lvov, Gi-Joon Nam and Michael B. Healy, "Floorplanning Challenges in Early Chip Planning", IEEE, pp. 388-393, 2011

[30] Chang-Sheng Ying and Joshua Sook-Leung Wong, "An Analytical Approach to Floorplanning for Hierarchical Building Blocks Layout" , IEEE Transactions on Computer-Aided Design, vol. 8, no. 4, April 1989

[31] RealTime User Guide (14.2), Mentor Garphics.

[32] URL: http://www.oasys-ds.com/products/realtime-designer

[33] URL:https://www.semiwiki.com/forum/content/2388-oasys-announces-floorplan compiler.html

[34] Jason Cong, and Min Xie, "A Robust Mixed-Size Legalization and Detailed Placement Algorithm," IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, vol. 27, no. 8,pp. 1349-1362, August 2008

[35] Thenappan M, Senthil Arasu T, Sreekanth KM, Ramesh S Guzar, "An Overlap Removal Algorithm for Macrocell Placement in VLSI Layouts," in proc. of International Conference on Computing: Theory and Applications (ICCTA'07), IEEE, 2007

[36] Sudeep Pasricha, Nikil D. Dutt, Elaheh Bozorgzadeh, and Mohamed Ben-Romdhane, "FABSYN: Floorplan-Aware Bus Architecture Synthesis," IEEE Transactions on Very Large Scale Integration (VLSI) Systems, vol. 14, no. 3,pp. 241-253, March 2006

[37] N. Banerjee, A. Raychowdhury, S. Bhunia, H. Mahmoodi, and K. Roy, "Novel Low-Overhead Operand Isolation Techniques for Low-Power Datapath Synthesis," in proc. of International Conference on Computer Design (ICCD'05), IEEE, 2005

[38] Lin Zhon and Niraj K. Jha, "Interconnect-aware High-level Synthesis for Low Power," IEEE, pp.110-117, 2002

[39] Fan MO, Abdallah Tabbara and Robert K. Brayton, "A Timing-driven Macro-cell Placement Algorithm", IEEE, pp.322-327, 2001

[40] Hsiao-Pin Su, Allen C.-H. Wu, and Youn-Long Lin, "A Timing-Driven Soft-Macro Placement and Resynthesis Method in Interaction with Chip Floorplanning", IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, vol. 18, no. 4, pp.475-483, April 1999

[41] Massoud Pedram, "Logical-Physical Co-design for Deep Submicron Circuits: Challenges and Solutions", IEEE, pp.137-142, 1998

[42] S.V. Venkatesh, "Hierarchical Timing driven Floorplanning and Place and Route Using a Timing Budgeter", in proc. of IEEE 1995 Custom Integrated Circuits Conference, pp.469-472, 1995

[43] URL: http://www.synopsys.com/Company/Publications/SynopsysInsight/Pages/Art2-synthesis-technology-IssQ4-12.aspx

[44] URL: http://www.edn.com/design/integrated-circuit-design/4313674/Establishing-timing-correlation-between-tools

[45] Saxena P., "An Introduction To Routing Congestion," Routing Congestion in VLSI Circuits Estimation and Optimization, pp 3-29,

[46] M.J.S Smith, "Floorplanning and Placement," Application Specific Integrated Circuits, 1997