International Journal of Engineering and Advanced Technology (IJEAT)
ISSN: 2249-8958 (Online), Volume-8 Issue-6, August 2019

Effective Usage of Chip Area by Optimizing the Dead Space

Ganugaphati Venkata Sai Mohan, Venkat Rao Ganjanaboyina

Abstract*: In the modern VLSI (Very Large Scale Integration) physical design, floor plan is the main step to optimize the circuit. The objective of floor plan is to optimize the interconnection between modules, area optimization and minimize the dead space. For very deep micron technologies, one of the major issue to design an chip is Dead space in physical design. In this paper, we introduced an algorithm for reducing dead space. This algorithm wrote in tickel programming language and implemented in Cadence Innovas Encounter Tool. By comparing to default algorithm floor plan, this algorithm reduces more dead space in the floor plan stage of the design.

Keywords* : *VLSI , Floor plan , Dead space, physical design.

I. INTRODUCTION

Physical design is the layout implementation of the design. After getting netlist, we perform various steps as floor plan, power plan, placement, clock Tree synthesis and routing and finally we get GDSII. Among all these steps, floor plan is the main step in physical design. The quality of our design depends on floor plan is good or bad. If floor plan is good, it is very easy to further steps as placing standard cells, building an clock tree and interconnections of macros and standard cells. If floor plan is bad, it leads to various issues like timing, congestion and routing problems etc. Generally, in floor plan we define the size of the chip, core boundary, die area, IO core spacing and aspect ratio. A top level design, the shape of chip is rectangular or square shape whereas in sub block design it will be rectangular or rectilinear shape. Floor plan will do both in top level and block level. In top level, all modules must be in rectangular shape. In design, it will meet conditions such as Each module lies parallel to the coordinate axes and No module overlaps each other. Core boundary is the area where we placing standard cells and macros. Floor planning means to place macros and interconnections between them. It is very necessary to optimize the size of the floor plan, reduce the effective area, reduce the dead area and reduce the interconnections between the modules. In this step of physical design, we estimate the locations of modules.

Revised Manuscript Received on August 30, 2019.

* Correspondence Author

Ganugaphati Venkata Sai Mohan*, Electronics and Communication Engineering, LakiReddy Bali Reddy College of Engineering, Mylavaram, INDIA. Email: ganugaphati.vsmohan@gmail.com

Venkat Rao Ganjanaboyina, Assistant Professor, Electronics and Communication Engineering , LakiReddy Bali Reddy College of Engineering, Mylavaram, INDIA. Email: gr1729@gmail.com

II. FLOOR PLAN PARAMETERS

Before starting the floor plan, we define parameters like height and width of core for calculating area of the core, aspect ratio and dead space.

Aspect ratio : The ratio of height to width of the module is known as aspect ratio.

Floor plan area : The product of width and height of core is an floor plan area.

Core utilization : The sum of total standard cell area and total macro area in total core area.

Dead space : It is an unused area which is not occupied any standard cell or macro.

$$\text{Dead space} = \frac{floorplanarea - sumofallmacrosarea}{floorplanarea}$$

Area of each module is the product of width and height of module.

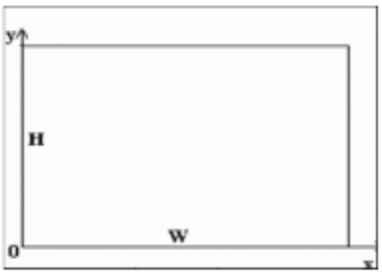

Fig.1. Representation of floor plan

III. EXISTED DESIGN

Now a days, we use different algorithms for reducing dead space in a chip level. In physical design, the first step is floor plan. Generally, tool follows default algorithm for floor plan. We have to plan for placing macros to avoiding issues like routing problems, dead area and meet timing of all standard cells. After fixing core area, die area and aspect ratio we have to run automatic macro placement. In that placement, it takes more iterations to avoid overlapping of macros. After placing automatically, we manually moved macros to avoid issues. The automatic floor plan by applying default algorithm is as shown in Fig.2. In this floor plan, we observed more dead space in the design. It cannot be modified that particular area and it will be wasted. It is the main drawback for floor plan.

If we placed macros by using default algorithm, it leads to more issues in further steps like place, clock tree synthesis and routing. Generally we fill the wasted area in design with filler cells after route for well continuity. If more unwanted area exists in design we want more number of filler cells. By using more fillers, the cost of the design will be more.

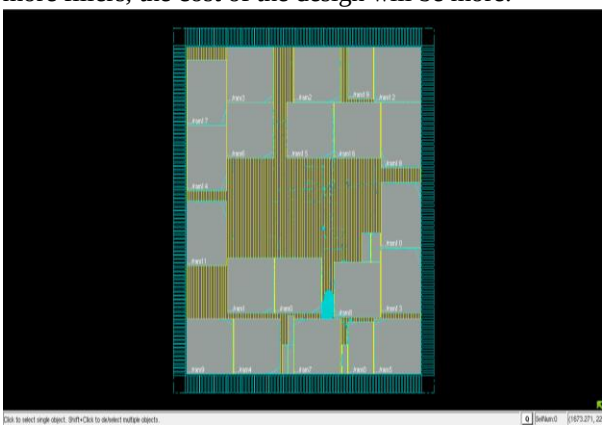

Fig.2.Existed Design

So the main drawback of existed design is it has more dead space and also design has high cost.

IV. PROPOSED PROCEDURE

In default algorithm, it causes more dead space in the design. To avoid this drawback we implemented one algorithm which was wrote in tickel programming language. After specifying core and die parameters, we place macros in the design. We created one algorithm for placing macros with efficient core utilization. First, we consider all macros complete name. We have to find each macro width and height. After getting these, we find area of each macro. All these parameters got by using different database commands. Consider initial coordinates of core boundary. Then we have to find which macro has smallest area and which has largest area. We start macro placement with smallest area one or largest area one. We started largest macro area and placing one by one with some distance from initial coordinates of core boundary. We have to give some distance for the purpose of interconnections from macro to macro or macro to standard cells. We consider width and height of entire core because whenever the boundary of core occurs while placing macros, the macro should be placed in the next alignment of core. Every macro should be checked the height of macro and spacing should be less than core boundary height it will place in that position. But if macro height should be higher than the core boundary height it will aligned in the next alignment of core boundary which is next of the first macro position. The macro should be placed after the width of initial macro and spacing. Likewise, we place all macros in our entire core area. If we place smaller area macros first, it chances an overlapping of macros and sometimes it goes to outside of the core boundary. It will be complex because further macros has larger areas. If we place larger areas macros first, it will gives an good macro placement. The algorithm is as shown in the Fig.3.

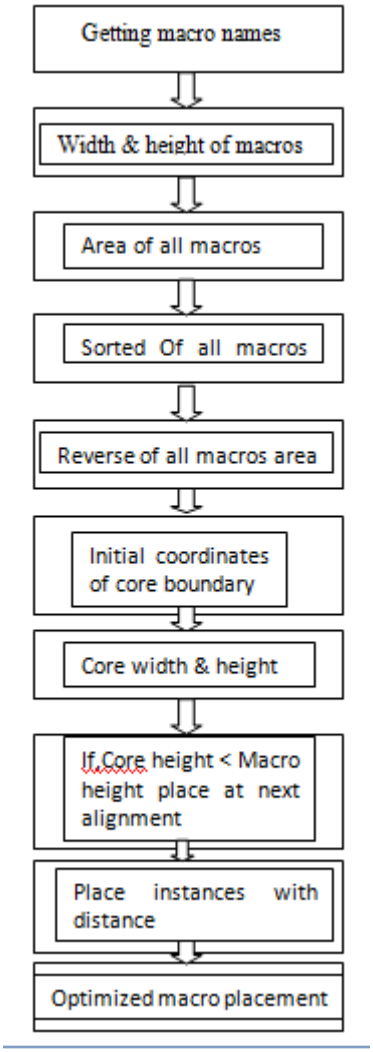

Fig.3.Proposed Algorithm

V. OUTPUT DESIGN OF PROPOSED

To design this work, tickel programming language and cadence encounter tools are used. After wrote the above algorithm in tickel programming language and sourcing in cadence encounter tool the macro placement is as shown in Fig.5. In this design, the area should be used efficiently. The area between the macros should be very less in the design. At the time of manufacturing the chip, we cut the right side of core boundary area. Because it is an unused area in the entire core of design. To reduce this area we can minimize the cost of the chip. The algorithm advantage is the unused area should be existed at end of the core boundary. By placing macros one by one efficiently the dead space occurs at end of core boundary. Whenever we use an efficient area in floor plan, then the design as good floor plan. If more unused area exists in the design it leads to issues in further steps. If we design with good floor plan it will be easy to do further steps and easy for manufacturing chip process. Otherwise it takes complexity to design an chip. For sourcing an algorithm for macro placement, it gives an effective usage of chip area.

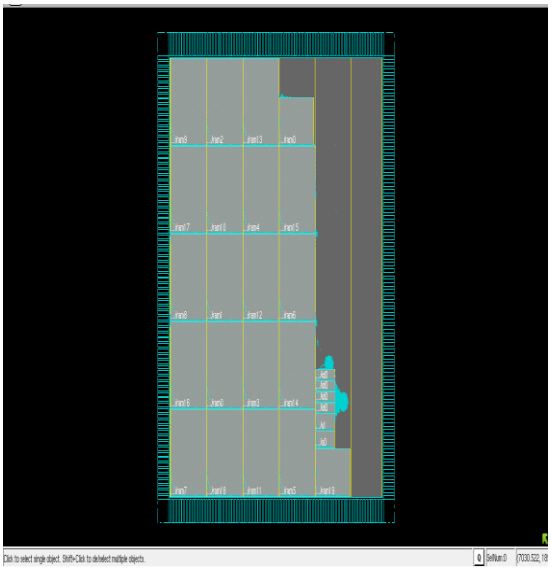

Fig.4. output design of proposed

VI. RESULTS

The design summary of proposed design is as shown in Fig.6. It has number of macros and standard cells and corresponding areas occupied in the entire core area.

Number of macros	27
Macros area	15657077.5Um^2
Number of standard cells	17886
Standard cells area	77261.84Um^2

Fig.5.Design summary of proposed

The comparison of dead space by sourcing default algorithm and proposed algorithm as shown in Fig.7. By comparing to default existed algorithm it reduces dead space in the design.

Parameter	Existed design	Proposed design
Dead space	0.56	0.30

Fig.6. Comparison of percentage of dead space

VII. CONCLUSION

The main aim of this project is to minimize the dead space in floor plan stage. The proposed algorithm wrote in ticel programming language and sourcing in cadence encounter tool. By comparing to default algorithms it reduces more dead space in the floor plan stage of the design. In existed algorithms it takes more number of iterations for placing macros in floor plan but we use this algorithm it less number of iterations for placing macros. The usage of filler cells for filling spaces for the purpose of well continuity is less compared to default algorithms. For that reason, we minimize the cost of the chip also. At the time of manufacturing the chip, we cut the unused area which is at the right side of the core in the chip. It is an advantage of this algorithm.

VIII. FUTURE SCOPE

In this paper, we created algorithm to minimize the dead space. As an extension, we place macros as per fly lines connectivity it will give a more perfect macro placement and also reduce dead space more.

REFERENCES

1. Rajendra Bahadur singh,Anurag singh Baghel, "Dead Space reduction of floorplan using simulated annealing algorithm", 2017 International Conference on energy, communication, data analytics and soft computing. **DOI:**10.1109/ICECDS.2017.8389822
2. Laskar, Naushad Manzoor, "A survey on VLSI Floorplanning: Its representation and modern approaches of optimization", Innovations in Information, embedded and communication systems, 2015 International Conference on IEEE.2015.
3. Singh, Rajendra Bahadur, anurag singh Baghel and ayush agarwal "A review on VLSI floorplanning optimization .using metaheuristic algorithms", Electrical, Elctronics and optimization techniques, International Conference on IEEE 2016.
4. *Phuong Hong Phan, Thanh Duc Tong, "Genetic progamming approach for SOC/IP floorplanning applications", The 2010 International Conference on Advanced Technologies for communications.* ***DOI:****10.1109/ATC.2010.5672733*
5. Lalin L.Laudis, S.Anand, Amit Kumar Sinha, "Modified SA algorithm for wirelength minimization in VLSI circuits", 2015 International Conference on circuits, power and computing Technologies, **DOI:**10.1109/ICCPCT.2015.7159500
6. J.Jenifer, S.Anand, Y. Levingstan, "Simulated annealing algorithm for modern VLSI floorplanning problem", ICTACT Journal of microelectronics, **DOI:**10.21917/ijme.2016.0030

AUTHORS PROFILE

Ganugaphati Venkata Sai Mohan, was born on 4th july 1995 in A.P, INDIA. He received his B.Tech in 2017 in ECE from NRI Institute of Technology , Agiripalli. Presently he is pursuing M.Tech Vlsi & Embedded Systems in Lakireddy Bali Reddy College of Engineering, Mylavaram, INDIA. His area of interest is Physical Design in VLSI.

Venkat Rao Ganjanaboyina, has received his M.Tech in Embedded Systems from JNTU Hyderabad in 2011 January. He is a member of IETE,ISTE. Presently, he is working as a assistant Professor at Lakireddy Bali reddy college of engineering, Mylavaram, Krishna (dist), A.P.India. His area of interest is VLSI and Embedded Systems .