國立交通大學

電子工程學系 電子研究所

碩 士 論 文

三維積體電路的時序導向分割擺置演算法

Partition-Based Timing Driven Placement in Three-Dimensional Integrated Circuits

研 究 生：陳奕蓉

指導教授：陳宏明 教授

中 華 民 國 九 十 九 年 九 月

三維積體電路的時序導向分割擺置演算法

Partition-Based Timing Driven Placement in Three-Dimensional Integrated Circuits

研 究 生：陳奕蓉　　Student：Yi-Rong Chen

指導教授：陳宏明　　Advisor：Hung-Ming Chen

國 立 交 通 大 學

電子工程學系 電子研究所

碩 士 論 文

A Thesis

Submitted to Department of Electronics Engineering and
Institute of Electronics
College of Electrical and Computer Engineering
National Chiao Tung University
in partial Fulfillment of the Requirements
for the Degree of
Doctor of Philosophy
in
Electronics Engineering

September 2010
Hsinchu, Taiwan, Republic of China

中華民國九十九年九月

三維積體電路的時序導向分割擺置演算法

學生: 陳奕蓉　　　　　　指導教授: 陳宏明 教授

國立交通大學

電子工程學系 電子研究所 碩士班

摘要

在現今超大型積體電路設計中，由於製程技術進步和三維技術引進，藉由直通矽晶穿孔的堆疊結構發展，來達到三維空間的垂直整合。直通矽晶穿孔取代了二維空間中過長的繞線,如何有效適當的擺置區塊和直通矽晶穿孔來改善時序問題。在此篇論文，我們將電路分層，逐一對每層執行切割擺置演算法的標準元件擺置，同時考量到擺置直通矽晶穿孔的對準條件限制，接著使用模擬退火法來減少繞線長度和優化時序，最後處理擺置的重疊問題。實驗結果顯示，三維積體電路較二維積體電路提升與改善整體效能。

Partition-Based Timing Driven Placement in Three-Dimensional Integrated Circuits

Student: Yi-Rong Chen   Advisor: Prof. Hung-Ming Chen

Department of Electronics Engineering

Institute of Electronics

National Chiao Tung University

Abstract

The semiconductor technology has been advanced in modern VLSI design. Three-dimension (3D) concept imports an additional dimension for circuit design by using stack structures with through-silicon via (TSV). 3D ICs replace longer interconnect in 2D ICs with TSV cells. However, there are problems how to place cells and TSV cells to improve timing. In this thesis, we perform standard cell placement by min-cut partitioning for one layer after layer assignment and address alignment constraint at the same time. Then use simulated-annealing to optimize timing and reduces wirelength of interconnect. In the last, a legal placement by a greedy method removes operlap between cells and TSV cells. The experimental results show that 3D ICs improve wirelength and delay of critical path than 2D ICs.

Acknowledgements

謝謝指導教授陳宏明老師給予我研究上的協助與幫忙，在碩士生涯中提供許多自由發揮的空間，讓我能夠得到許多知識。

也謝謝實驗室的學長姐，佳蕙和學弟學妹們，在研究的生活中，因有大家的幫忙所以才得以順利，快樂的度過。

最後謝謝我的家人，讓我沒有後顧之憂的學習，取得碩士學位。

Contents

Abstract(Chinese) **i**

Abstract **i**

Acknowledgements **ii**

Table of Contents **v**

List of Figures **vi**

List of Tables **vii**

1 Introduction **1**

1.1 Review of Previous Work 1

1.2 Thesis Organization 3

2 Preliminaries **4**

2.1 Partitioning-Based Placement 4

2.2 Timing Model 5

2.3 Problem Formulation 8

3 Placement Methodology **10**

3.1 Flow of Our Methodology 10

3.2 Partition-Based 3D IC Placement 11

3.3 Alignment Constraint 15

3.4 Simulated-Annealing Timing Driven Placement 16

3.5 Legalization 18

4 Experimental Results **20**

5 Conclusion and Future Work **24**

Bibliography **30**

List of Figures

1.1 Interconnect in 2D and 3D ICs 2

2.1 The Elmore delay model. 6

2.2 TSV delay model 7

2.3 Bounding box of a net 9

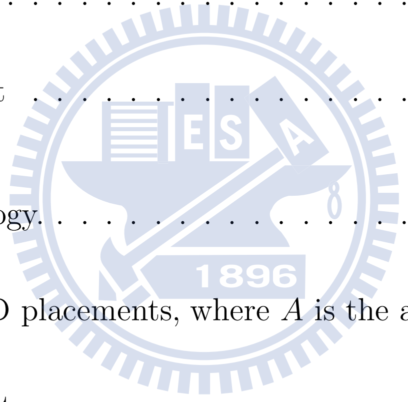
3.1 The flow of methodology. 10

3.2 Patterns of 2D and 3D placements, where A is the area. 11

3.3 TSV initial assignment. 13

3.4 TSV alignment constraint. 15

4.1 TSV-aware placement of 3-layer stacking. (a)Bottom layer, (b)Middle layer, (c)Top layer. 23



List of Tables

4.1 Characteristics of benchmark iscas85 after 3D IC partition. 21

4.2 2D placement after legalization. 21

4.3 3D (3 layer) placement after legalization and alignment constraint. 21

4.4 3D (4 layer) placement after legalization and alignment constraint. 22

4.5 3D v.s 2D placement improvement. 22

Chapter 1

Introduction

1.1 Review of Previous Work

The stacked multi-layer ICs of three-dimensional integrated circuits (3D ICs) become more and more popular recently, because 3D ICs have better performance than 2D ICs such as timing, wirelength and power. Thus many researchers focus on 3D integration today. First, 3D ICs overcome the problem of longer interconnect in 2D ICs [2]. It uses TSV cells to communicate signal among different layers. In that way, it reduces wirelength contribution significantly [3]. Second, 3D circuit placement of cells and TSV cells are important after partitioning. In the last years, the placement of 2D ICs have been studied. Many 2D ICs placement algorithms are mature today such as partitioned-based placement (Min-Cut placement), simulated annealing, and forced-directed placement.

In order to make the best advantages of 3D ICs to increase higher performance, the traditional design flow for 2D ICs needs some modifications. In other words, 2D ICs placement algorithms should consider the vertical issue then it would be appled to 3D ICs.

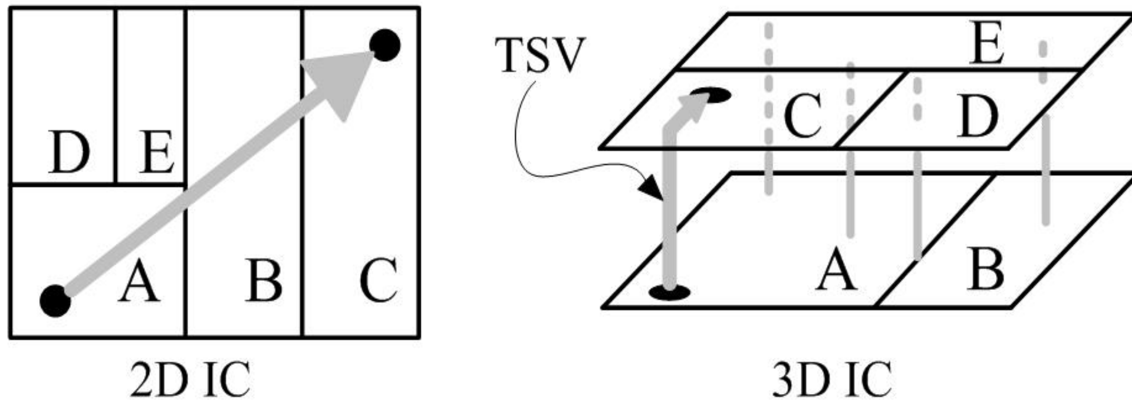


Figure 1.1: Interconnect in 2D and 3D ICs

Performance-driven placement [4]-[6] like timing driven placement [7]-[14] considers timing and wirelength at the same time. Its target is to minimize wirelength of timing critical path and to improve timing [15]-[19]. The existing timing-driven placement algorithms of 2D ICs are path-based and net-based. The path-based method minimizes the critical path delay directly. However, it needs computational cost due to a set of maximum paths. The net-based method is simple and less computational cost than path-based. The first one is net weight which transforms timing information into net weight. It assigns the critical path for higher weight than noncritical path in order to improve timing of critical path. The second one is net constraint which generates suitable constraints and does not limit solution space. They usually minimize the critical path delay or maximize the minimum slack to achieve timing constraints. The goals of these methods are that total wirelength minimization of the circuit achieves better timing performance.

The timing optimization is still important to 3D IC design. Many researches indicate that timing issue is significant in the design flow. It is essential to design a good circuit placement algorithm for 3D ICs that makes the best of its shorter interconnect and better timing performance than 2D ICs. In this regard, a design flow for 3D ICs timing-driven placement based on partition-based placement has been proposed in this thesis. [20], [21].

1.2 Thesis Organization

The remainder of this thesis is organized as follows: Chapter 2 depicts preliminaries about general partition-based placement problems, a design flow of 3D ICs timing-driven placement and a timing model for static timing analysis. Chapter 3 describes our 3D partition-based placement methodology. Chapter 4 presents the experiment results on 3D IC placement and compares it with 2D IC design based on the same algorithm. Finally, we summarize the conclusions and future work in Chapter 5.

Chapter 2

Preliminaries

2.1 Partitioning-Based Placement

The partitioning-based placement is also called min-cut placement. There are many advantages for this multilevel hypergraph partitioning placement algorithm like fast, suitable for large size design and total weirelength can also be reduced as well. Timing driven placement can be driven by wire-driven placement first in order to reduce wirelength cost efficiently.

Partitioning-based placement cut the placement region iteratively by cutline then the placement area is partitioned into two sub-circuits. Each sub-circuit is assigned to one part of the placement region. Repeat partitioning the circuit iteratively and swapping the cells minimize the cut cost. Stop partitioning until the size of sub-circuits are smaller enough to place cells. In the standard cell design, we stop cutting the placement region until the size of placement region equals to the height of row, then we assign the cells into rows.

In general, the partitioning-based placement is based on FM partition algorithm. The

moved mechanism is based on cells´ gain. First, it calculates the initial gain of cell and then move the cell which has the maximum gain. Update gain list after moving the maximum gain cell until all cells are moved. The process is finished after calculating the total maximum sums of gain. In the last, pick up the total maximum sums of gain and move the cells which contribute to the regions with minimum cut cost. The maximum gain means the smaller cuts between seperating two groups of cell. The iteratively moved process stops until moving cells which have negative gain.

The Iterative idea for partitioning-based placement attempts to minimize the total numbers of cut between the two sub-circuits and balance areas of two part at the same time. According to its moved based behavior, we try to group these closely connected cells together. The results of exchanging the cells in each iteration will improve cut size between two sub-circuits and total wirelength.

2.2 Timing Model

To guide timing driven placement, the accurate timing information such as arrival time, required time and slack are very important for timing analysis. Using precise timing model to analyze timing would improve timing effectively. Here we use Elmore delay model [25] for estimating delay of interconnect and TSV cells from source to target timing point. The Interconnect delay model is described as follow [26].

The interconnect delay T from driver to receiver can be modeled as (2.1)

$$T = R_d\left(C_w + C_\ell\right) + C_\ell R_w + \frac{1}{2}C_w R_w \tag{2.1}$$

where wire resistance R_w, capacitance C_w, output resistance of driver R_d, and load capacitance for its receiver C_ℓ. The formulation can also be replaced with another form.

$$T = \frac{1}{2}rcL^2 + \left(cR_d + rC_\ell\right)L + R_d C_\ell \tag{2.2}$$

where r is unit length wire resistance and c is unit length wire capacitance.

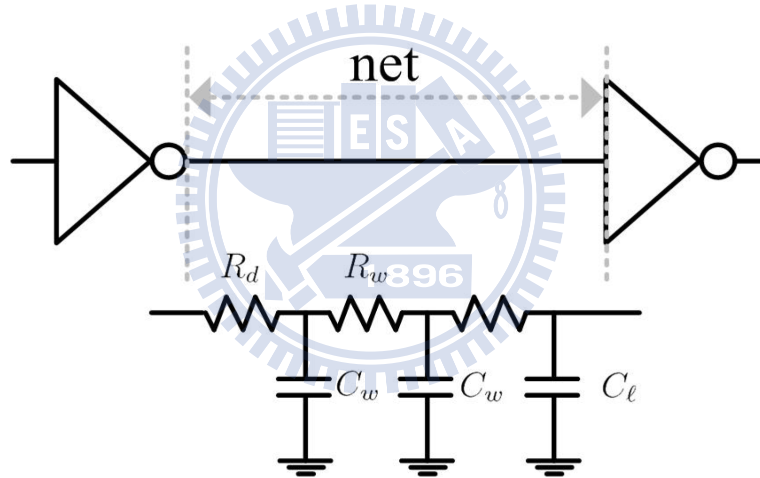


Figure 2.1: The Elmore delay model.

Another important issues for 3D ICs static timing analysis are TSV delay modelś construction and parasitic extraction. Due to the additional dimension of z, it would be a new delay model for z dimension timing calculation. Here we do not consider inductance effect, the z dimensional timing calculation also can be modeled as Elmore delay model[27]. The delay of TSV cell is one of interconnect delay. The parasitic of TSV cell has been described

in [28]. The resistance of a TSV cell, R_{TSV}, is a function of conductivity and cross-sectional area, the capacitance of an isolated TSV cell, C_{TSV}, and Elmore delay form for TSV cells are described as follow:

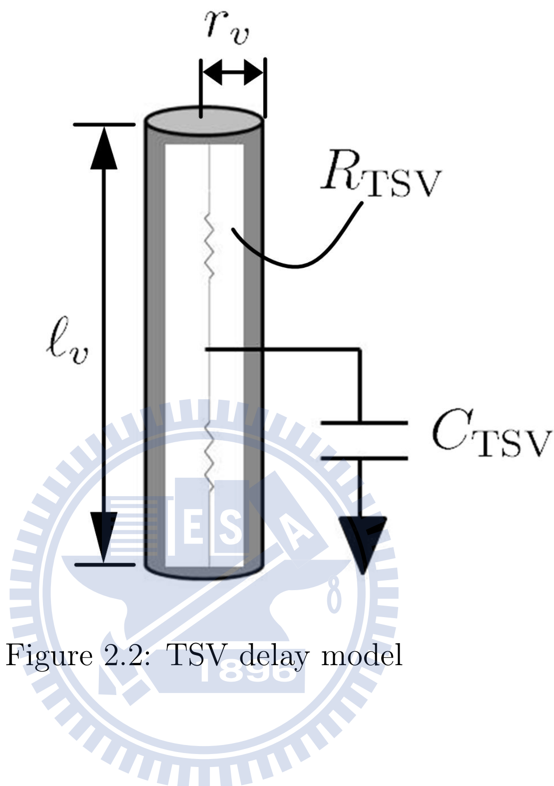


Figure 2.2: TSV delay model

1896

$$T_{\text{TSV}} = \frac{1}{2} rcL^2 + (cR_{\text{TSV}} + rC_{\text{TSV}})\,L + R_{\text{TSV}}C_{\text{TSV}} \tag{2.3}$$

$$R_{\text{TSV}} = \frac{\ell_v}{\sigma \pi r_v^2} \tag{2.4}$$

$$C_{\text{TSV}} = \frac{63.34\epsilon_0 \ell_v}{\ln\left(1 + 5.26\frac{\ell_v}{r_v}\right)} \tag{2.5}$$

where l_v is the length of TSV (um), r_v is the radius of TSV (um), σ is the substrate conductivity, and ϵ_0 is the permittivity.

2.3 Problem Formulation

The problems of 3D via-last ICs timing driven placement can be specified as follows. Given a directed hypergraph $G(C, E)$ which is equivalent to the def file, where C and E mean the cell and the edge, respectively. The numbers of movable cell is n. Each cell $c \in C = \{c_1, c_2, \cdots, c_n\}$, the coordinates is defined as (x_i, y_i, z_i). The cells are distributed on the layer after 3D layer assignment. A directed edge $e = (a, b) \in E = \{e_1, e_2, \cdots, e_n\}$ means signal direction is from cell $a \in C$ to cell $b \in C$. The primary input/output pads are assigned boundary of the bottom layer and the coordinates are fixed during placement. The number of 3D vias is m on different layers. Each 3D via $v \in TSV = \{v_1, v_2, \cdots, v_m\}$.

The timing of a circuit is represented by two metrics: worst negative slack WNS and total negative slack TNS. WNS is defined as the worst slack of all timing endpoints. Timing points is input/output pin of a cell or primary input/output pads of the circuit. TNS is defined as the sum of the timing endpoints which have negative slack. For each timing points i, its arrival time T_{arrival} is the time that signal arrives at the timing points, required time T_{required} is the time that signal requires at the timing points, and *slack* is the difference between required time and arrival time. The negative slack means that signal arrives later than the required time. All of them can be defined as (2.6)-(2.7), respectively.

$$T_{\text{arrival}}(i) = \max_{\forall j \in fanin(i)} \{T_{\text{arrival}}(j) + T_{\text{inst}}(i) + \text{delay}(j, i)\} \tag{2.6}$$

$$T_{\text{required}}(i) = \min_{\forall j \in fanout(i)} \{T_{\text{required}}(j) - T_{\text{inst}}(j) - \text{delay}(i, j)\} \tag{2.7}$$

The wirelength calculation model of a net is the bounding box enclosing all the cells connect to the net, estimating by half perimeter wirelength (HPWL)[23].

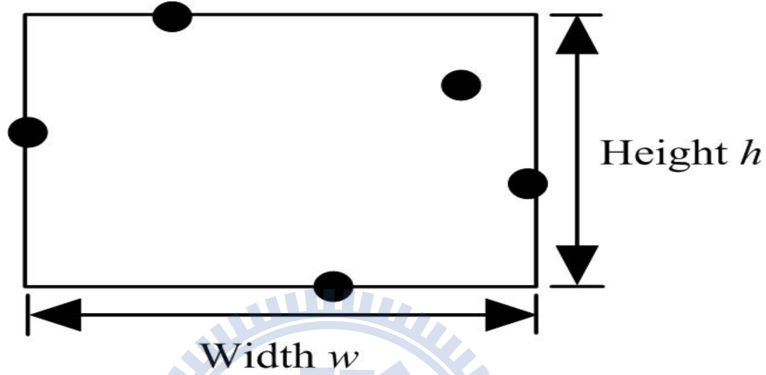


Figure 2.3: Bounding box of a net

1896

$$\text{HPWL}(e) = \left(\max_{c_i \in e} \{x_i\} - \min_{c_i \in e} \{x_i\}\right) + \left(\max_{c_i \in e} \{y_i\} - \min_{c_i \in e} \{y_i\}\right) \tag{2.8}$$

$$\text{TSV}(e) = \left(\max_{c_i \in e} \{z_i\} - \min_{c_i \in e} \{z_i\}\right) \tag{2.9}$$

where e is net, c_i is cell and its coordinates is (x_i, y_i, z_i)

The objective is to find a legal placement of cells and TSV cells with no overlap. At the same time, 3D IC designs can achieve better timing performance, shorter wirelength and minimize total numbers of TNS, WNS and critical path delay. Meanwhile, TSV cells always satisfy alignment constraint.

Chapter 3

Placement Methodology

3.1 Flow of Our Methodology

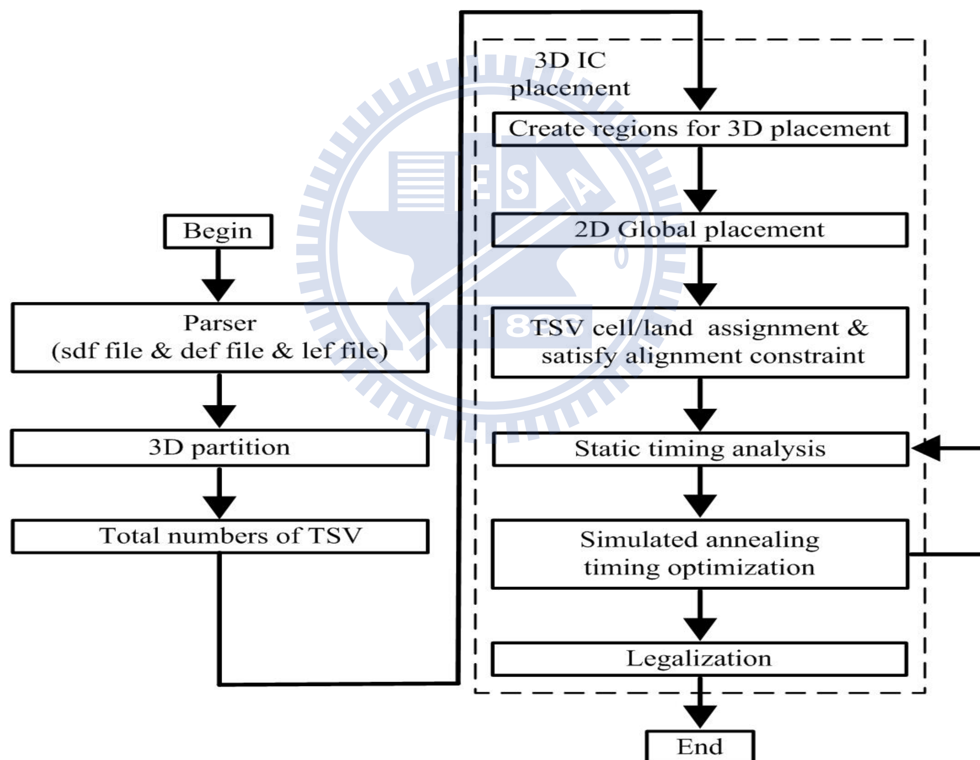


Figure 3.1: The flow of methodology.

In this chapter, a 3D partitioning-based timing driven placement algorithm that considers both timing and TSV assignment is proposed. Our methodology flow is illustrated

in Figure 3.1. The organization of this chapter is described as follows. Section 3.2 describes partitioning-based placement and how to assign coordinates of TSV cell. Section 3.3 describes how to use Elmore delay model to analyze delay of interconnect and TSV cells. Section 3.4 describes timing optimization by setting cost functions of simulated annealing timing driven placement. Section 3.5 describes legalization during detail placement.The via alignment constraint has been used to TSV-aware placement in section 3.6.

3.2 Partition-Based 3D IC Placement

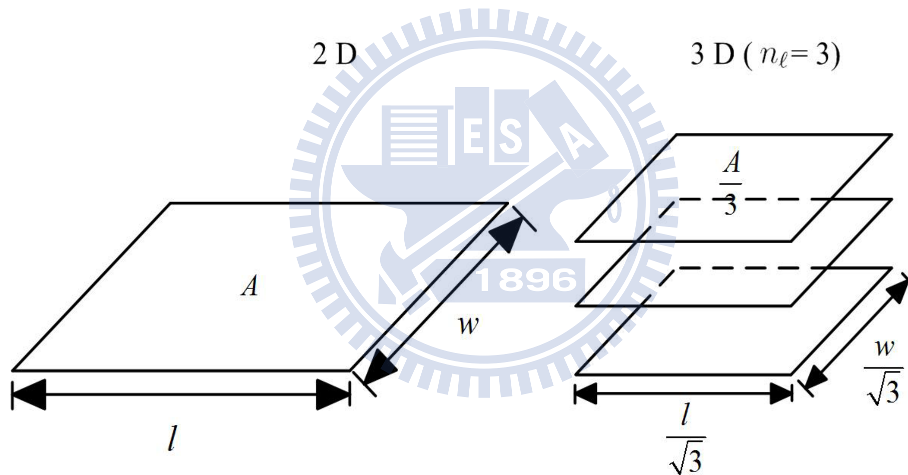


Figure 3.2: Patterns of 2D and 3D placements, where A is the area.

First, we begin our flow from partitioning a 2D circuit then transform it into a 3D design. The 3D placement region is calculated by dividing the 2D region to numbers of layer in Figure 3.2. The width of cell is reduced with the multiple of $\frac{1}{\sqrt{n_\ell}}$, where n_ℓ is the numbers of layer. The area of cell is reduced to $\frac{1}{n_\ell}$. In conclusion, the total placement region is still the same with 2D after scaling.

The communications of 3D ICs pass through by TSV cells, so a fine placement for cells and TSV cells would minimize total wirelength and have better timing performance. The numbers of TSV cell which are required for signal on each layer is determined after layer assignment Although TSV cells reduce longer interconnect delay than 2D ICs.However, the area of TSV cell occupy on a layer is the overhead for 3D IC after all. As a result, we do not want to increase the numbers of TSV cell. we focus on cells and TSV cells assignment. Although TSV cells reduce longer interconnect delay than 2D ICs.However, the area of TSV cell occupy a layer is the overhead area for 3D IC after all. As a result, we do not want to increase the numbers of TSV cell. we focus on cells and TSV cells assignment.

To avoid increasing overhead of TSV cells and to maintain the good wirelength on 2D plane, we simplify the problem of 3D ICs placement by iteratively 2D partition-based placement. Running Fiduccia/Mattheyses heuristic iteratively from top to bottom layer completes standard cells placement in row-based condition. In the condition, every plane repeats running bi-partition method to part the plane by vertical cuts. The cells on the same layer can be assigned gradually. After the recursive partitioning-based placement to minimize wirelength and balance area, we can finish the global placement.

The difference between 3D ICs and 2D ICs is the TSV cells. In other word, cells on different layers communicate the signals by using TSV cell. The locations of TSV cell may

affect timing of circuit delay. Different placement of TSV cells causes different delay. If TSV cells far away from cells, it may introduce longer interconnect delay. On the contrary, if TSV cells near cells that signal can pass faster with no additional interconnect delay. It is important for TSV cells placement during timing driven placement thereby.

The coordinates of cells are assigned after partitioning-based placement, then we can assign the coordinates of TSV cells. First, we distribute average numbers of TSV cells into rows. In that way, there are many benefits . One is that row has average TSV cells distribution that would not generate large TSV cells density, another is that TSV cells have more opportunities to exchange between different rows.

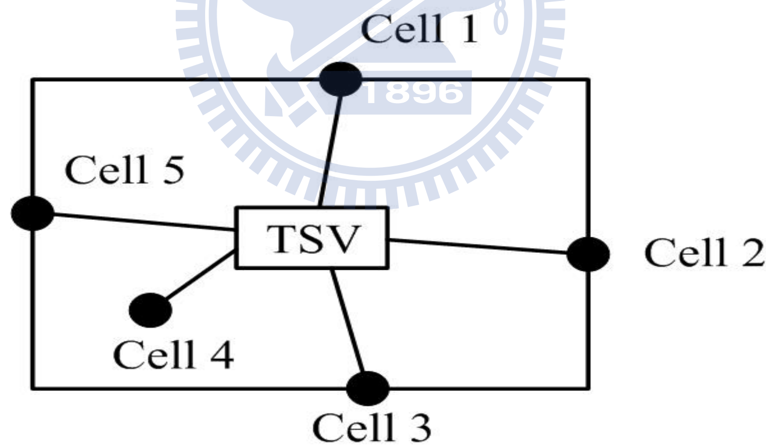


Figure 3.3: TSV initial assignment.

The coordinates of TSV cells is set by calculating the center of bounding box, so the coordinates of TSV cell are related to coordinates of cell. The reason is smaller wirelength and minimum cost. The center (x,y) of bounding box is decided by the smallest and largest x

coordinates of cells, and y coordinates calculation is the same. This step is initial coordinates set for TSV cells.

After initial setting, we want that TSV cells are individually set to one row. The setting method is according to the TSV-distribution row (a row which has TSV cells distribution). If the initial y coordinates of TSV cell do not map to any one TSV-distribution row. We use a greedy method to set the TSV cell to an appropriate and free coordinates. The greedy method is described as following. Once the coordinates of TSV cell is assigned, the signal communication across different layers can be correct. The delay time of circuit can be calculated by timing model when completing cells and TSV cells connection.

```
TSV cells placement
  for layer=top to bottom do
     for TSV-distribution row do
     1. get TSV distribution
     2. initial TSV cells assignment
        center of bounding box calculation
     3. find a best row
     4. set TSV cells to appropriate coordinates
     5. link signal
     end
  end
```

```
Set TSV cells to appropriate coordinates
  for TSV-distribution row r do
  if( r has free TSV cells )
     1. cost calculation for r cost_r
        cost_r = coordinates y of r - initial y of TSV cell
     2. choose the smallest cost cost_r ( the nearest row )
     3. set y coordinates of TSV cell
  end
```

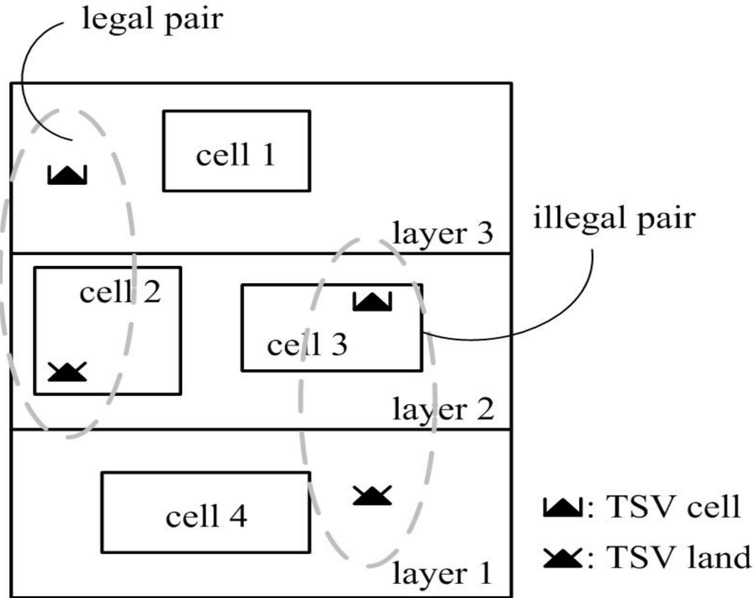


Figure 3.4: TSV alignment constraint.

3.3 Alignment Constraint

The signal across different layer is passed by using the TSV cell. Once we align TSV cell and TSV land, the re-distribution layer (RDL) cost can be reduced. In order to always satisfy alignment constraint the TSV cells on the layer i (the upper layer) should align with the landing cells on layer i-1 (the layer which is lower than TSV cells distribution). In this way, it would have correct connections between micro-bumps and landing pads.

The coordinates of TSV cells would be decided for delay calculation. In order to avoid the problem of cells overlap with TSV cells for TSV-aware placement. In [32] proposed a method to solve the problem. The method is to execute the placement flow from the top to the bottom layer that can always satisfy the alignment constraint. TSV cells on every layer

do not overlap with any cells on the same layer. On the contrary, if we assign TSV cells and landing cells from the bottom to the top layer. Because there are no TSV cells on the bottom layer, we assign only cells and landing cells. The TSV cells on upper layer in order to satisfy the alignment constraint, it causes the problem of the TSV cells which overlap cells on the same layer.

To sum up, the alignment constraint should be considered in 3D IC design. The existing idea is also used in our flow. We assign TSV cells from the top to the bottom layer and every layer can always satisfy alignment constraint. There are no overlap between TSV cells and cells in our experimental results.

3.4 Simulated-Annealing Timing Driven Placement

In our timing driven placement, we use simulated-annealing placement to improve and optimize timing . The simulated-annealing algorithm is an annealing process by setting the proper cost functions to achieve performance. The random behavior of exchanging cells can avoid local solution. We choose the exchanged cells one is connected with high critical edge and another one is connected with non critical edge. The exchange cells whether accept or not which is decided by cost difference and probability now. In other words, if difference of the cost is positive, it will be accepted for exchanging cells. On the other hand, if difference of the cost is negative, it will be accepted by probability. As a result, the setting of cost

function is a key point for timing driven placement.

In [33], assigns the cell that has smaller slack with higher criticality to handle critical gate. The criticality is defined as :

$$Criticality(i,j) = 1 - \frac{Slack(i,j)}{CriticalPathDelay} \tag{3.1}$$

where $Slack(i,j) = T_{\text{required}}(j) - T_{\text{arrival}}(i) - \text{delay}(i,j)$, $T_{\text{required}}(j)$ is required time of cell j , $T_{\text{arrival}}(i)$ is arrival time of cell i, $\text{delay}(i,j)$ is the interconnect delay of cell i and cell j.

Now we introduce how we set the cost function for timing driven placement. The cost function is composed of wirelength and timing. The performance of wirelength and timing optimization are our target. We try to get better timing performance without increasing the wirelength. In this condition, we set $Wire_Cost$ (3.2) and $Time_Cost$ (3.4) as our cost function.

$$Wire_Cost = \sum_{e=1}^{Nets} [HPWL(e) + TSV(e)] \tag{3.2}$$

$$Time_Cost(i,j) = \text{delay}(i,j) \cdot Criticality(i,j)^{Criticality_Exponent} \tag{3.3}$$

$$Time_Cost = \sum_{\forall i,j \in \text{cell}} Time_Cost(i,j) \tag{3.4}$$

$$\Delta C = \lambda \cdot \frac{\Delta Time_Cost}{PreviousTime_Cost} + (1-\lambda) \cdot \frac{\Delta Wire_Cost}{PreviousWire_Cost} \tag{3.5}$$

3.5 Legalization

Cells and TSV cells may overlap with each other after global placement. The detail placement handles this illegal placement problem [30]. In [29], they study the 3D-Via placement problem for 3D circuits and placing the 3D-Vias with no overlap with other 3D-Vias on the same layer.

Here we propose a greedy legalization approach based on the Tetris legalization algorithm [31] for 3D IC legalization. The flow is similar to Tetris, but it is different to the previous work, we solve this problem by legalization of TSV cells first and then cells.

TSV legalization

There are may be overlaps between TSV cells in one row. In the condition of average TSV cell distributions for one row, we only need to modify and consider its new x coordinates. The y coordinates of TSV cell was already assigned during the initial placement. In legalization, it only has to distribute TSV cells in one row. First, we get all the TSV cells in one row and calculate its new x coordinates X_c which is center of the bounding box. The reason is that the coordinates of cells may be changed after placement for better timing performance, so the bounding box of a net is not the same with initial assignment. Second, sort them according to their x coordinates. If the TSV cell still overlaps with another TSV cell after resetting its coordinates, we calculate its legal x coordinates again. The legal x coordinates

X_ℓ is decided after calculating its cost. The suitable coordinates is the coordinates which has minimum cost. The method is illustrated below.

```
TSV legalization
for each row r do
   1. r → getTSVCell
   2. calculate center X_c of bounding-box
   3. sort them according to X_c
   4. if( still overlap with another TSV cells)
         calculate legal coordinates X_ℓ
         calculate cost_diff = abs(X_ℓ, X_c)
   5. choose minimum cost_diff
   6. set legal coordinates X_ℓ for TSV cell
end
```

Cell legalization

After TSV cells legalization, we treat the TSV cells in a row as obstacles. We only need to handle the overlap between cell and cell. The legalization is similar to Tetris. We reassign coordinates of cell in one row. If the width of row is not enough, we find its the nearest row which has the same direction then set it.

```
Cell legalization
for each row r do
   1. r → getTSV
   2. calculate useful region for cells
   3. if(cell overlap with another cell in the useful region )
         replace cell on this row
         if(over row width)
            choose the nearest row which has the same direction with the cell
      end if
end for
```

Chapter 4

Experimental Results

This algorithm is implemented using C++. The platform is on the Intel Xeon Dual Core 3.0Ghz processor and 4096MB RAM in a Linux environment. In this thesis, our benchmarks are ISCAS85 circuits which were downloaded from [24].

The circuit is described in LEF/DEF format and intrinsic delay is from standard delay file (sdf). We modify original 2D ICs file into 3D ICs design. The area is reduced to original $(\frac{1}{n_\ell})$, where n_ℓ is numbers of layer. The locations of pad are also a little changed.

After partitioning the circuit, we generate a placement region which is the largest area among these three layers. The area of layer contains cells and TSV cells. All IOs are on bottom layer. The area of TSV IO is 85 μm^2 and TSV cell is about 255 μm^2. The aspect ratio of standard cell placement region is 1.13:1. Table 4.1 described the characteristic of benchmarks. Table 4.4 shows the three layers placement results.

Figure 4.1 shows the face to back via-last bounding technology legal placement results of

c1908, TSV cells and TSV lands distribution, the color of red line which means the placement region, the blue blocks mean TSV lands, the green blocks mean TSV cells, and pink blocks mean cells. At the same time, it satisfies alignment constraint. The location of TSV land on layer i-1 aligns to the TSV cell on layer i, and TSV lands can overlap with cells and TSV cells on the same layer.

Table 4.1: Characteristics of benchmark iscas85 after 3D IC partition.

Input file	#TSV cell	#TSV land	#cell
c1908	105	105	880
c2670	403	403	1269
c3540	139	139	1669
c5315	330	330	2307
c7552	531	531	3513

Table 4.2: 2D placement after legalization.

Circuit	HPWL	Critical path(ps)	TNS	WNS(ps)
c1908	73552.27	46970.87	28	-0.002
c2670	132360.72	38142.38	28	-0.002
c3540	197642.19	55242.35	28	-0.002
c5315	249188.64	57428.66	28	-0.004
c7552	480862.28	50797.82	28	-0.004

Table 4.3: 3D (3 layer) placement after legalization and alignment constraint.

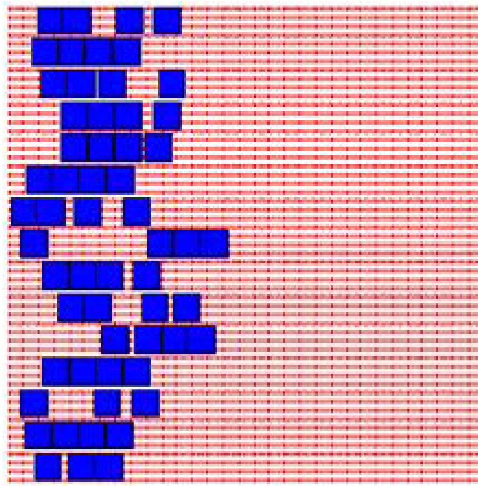
Circuit	HPWL	Critical path(ps)	TNS	WNS(ps)
c1908	39964.83	46923.97	28	-0.006
c2670	97814.48	38083.8	0	0
c3540	110990.16	54902.31	0	0
c5315	181797.53	57177.56	28	-0.004
c7552	286605.88	50379.83	28	-0.002

Table 4.4: 3D (4 layer) placement after legalization and alignment constraint.

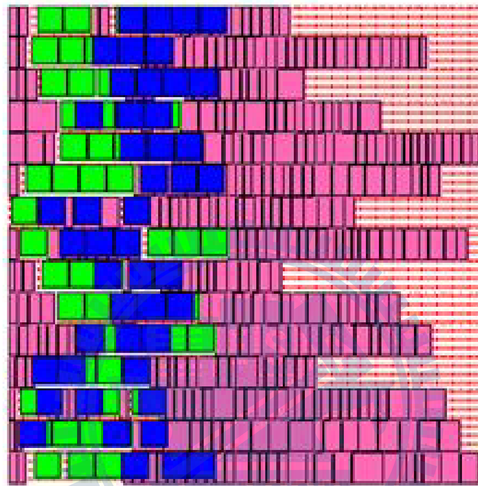
Circuit	HPWL	Critical path(ps)	TNS	WNS(ps)
c1908	32086.79	46891.95	28	-0.004
c2670	72576.26	37821.84	14	-0.001
c3540	78743.85	54861.85	28	0
c5315	161172.89	57208.76	7	-0.0007
c7552	205572.98	50315.58	28	-0.002

Table 4.5: 3D v.s 2D placement improvement.

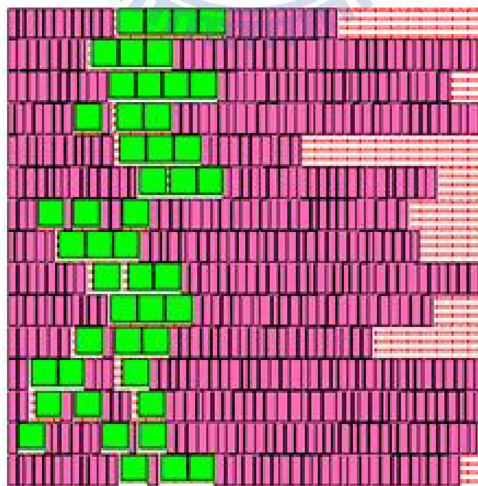
Circuit	#Layer	HPWL	Critical path(ps)
c1908	3	45.66 %	0.1 %
c1908	4	56.38 %	0.17 %
c2670	3	26.1 %	0.15 %
c2670	4	45.17 %	0.84 %
c3540	3	43.84 %	0.62 %
c3540	4	60.16 %	0.69 %
c5315	3	27.04 %	0.44 %
c5315	4	35.32 %	0.38 %
c7552	3	40.4 %	0.82 %
c7552	4	57.25 %	0.95 %



(a)



(b)



(c)

Figure 4.1: TSV-aware placement of 3-layer stacking. (a)Bottom layer, (b)Middle layer, (c)Top layer.

Chapter 5

Conclusion and Future Work

In this thesis, our contribution of this work is that we implement a 3D placement flow including timing analysis. We assign cells and TSV cells by partitioning-based placement and greedy algorithm in global placement. The TSV cells assignment always satisfy alignment constraint. Then simulated-annealing refines the timing performance. At last, we remove overlap during detail placement. The experimental results show that our methodology in 3D ICs has better wirelength performance than 2D ICs and less TNS in some benchmarks.

In the future, the most important thing is that models a more precise TSV delay model for 3D timing analysis. This is essential for delay calculation. The accurate TSV cell model is urgent in physical design of 3D ICs.

Bibliography

[1] H. Hua, C. Mineo, K. Schoenfliess, A. Sule, S. Melamed, R. Jenkal, and W. R. Davis, "Exploring compromises among timing, power and temperature in three-dimensional integrated circuits," *In Proceedings of ACM/IEEE Design Automation Conference*, pp. 997-1002, 2006.

[2] D. H. Kim, S. Mukhopadhyay, and S. K. Lim, "Through-Silicon-Via aware interconnect prediction and optimization for 3D Stacked ICs," *In Proceedings of International Workshop on System-Level Interconnect Prediction*, pp. 85-92, 2009.

[3] S. K. Lim, "TSV-Aware 3D physical design tool needs for faster mainstream acceptance of 3D ICs," *ACM DAC Knowledge Center (dac.com)*, 2010.

[4] G. Chen, and S. Sapatnekar, "Partition-Driven standard cell thermal placement," *In Proceedings of International Symposium on Physical Design*, pp. 75-80, 2003.

[5] W. Sui, S. Dong, J. Bian, and X. Hong, "Fast wirelength-driven partition-based placement for island style FPGAs," *In Proceedings of Joint Conference on Information Sci-*

ences , 2008.

[6] Y. C. Chou, and Y. L. Lin, "A performance-driven standard-cell placer based on a modified force-directed algorithm," *In Proceedings of International Symposium on Physical Design*, pp. 24-29, 2001.

[7] C. Hwang, and M. Pedram, "Timing-driven placement based on monotone cell ordering constraints," *In Proceedings of the Asia and South Pacific Design Automation Conference*, pp. 201-206, 2006.

[8] S. Raman, C. L. Liu, and L. G. Jones, "Timing-constrained FPGA placement: A force-directed formulation and its performance evaluation," *VLSI Design*, vol. 4, no.4 ,pp. 345-355, 1996.

[9] A. B. Kahng, and Q. Wang, "An analytic placer for mixed-size placement and timing-driven placement," *In Proceedings of IEEE/ACM International Conference on Computer Aided Design*, pp. 565-572, 2004.

[10] X. Yang, B. Choi, and M. Sarrafzadeh "Timing-driven placement using design hierarchy guided constraint generation," *In Proceedings of IEEE/ACM International Conference on Computer Aided Design*, pp. 177-180, 2002.

[11] C. Sechen, and W. Swartz, "Timing driven placement for large standard cell circuits," *In Proceedings of ACM/IEEE Design Automation Conference*, pp. 211-215, 1995.

[12] Z. Xiu, and R.A. Rutenbar, "Timing-driven placement by grid-warping," *In Proceedings ACM/IEEE Design Automation Conference*, pp. 585-591, 2005.

[13] P. Maidee, C. Ababei, and K. Bazargan, "Timing-driven partitioning-based placement for island style FPGAs," *IEEE transaction on Computer-Aided Design of Integrated Circuits and Systems*, vol. 24, no. 3, pp. 395-406, March 2005.

[14] S. L. Ou, and M. Pedram, "Timing-driven placement based on partitioning with dynamic cut-net control," *In Proceedings of ACM/IEEE Design Automation Conference*, pp. 472-476, 2000.

[15] D. Sinha, N. V. Shenoy, and H. Zhou, "Statistical Gate Sizing for Timing Yield Optimization," *In Proceedings of IEEE/ACM International Conference on Computer Aided Design* , pp. 1037-1041, 2005.

[16] W. P. Lee, H. Y. Liu, and Y. W. Chang "Voltage island aware floorplanning for power and timing optimization," *In Proceedings of IEEE/ACM International Conference on Computer-Aided Design*, pp. 389-394, 2006.

[17] H. Wu, M. D. R. Wona, and I. Liu, "Timing-constrained and voltage-island-aware voltage assignment," *In Proceedings of ACM/IEEE Design Automation Conference*, pp. 429-432, 2006.

[18] J. F. Lee, and D. T. Tang, "An algorithm for incremental timing analysis," *In Proceedings of ACM/IEEE Design Automation Conference*, pp. 696-701 ,1995.

[19] H. F. Jyu, S. Malik, S. Devadas, and K. W. Keutzer, "Statistical timing analysis of combinational logic circuits," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 1, no. 2, pp. 126-137 ,Jun 1993.

[20] M. C. Yildiz, and P. H. Madden, "Improved cut sequences for partitioning based placement," *In Proceedings of ACM/IEEE Design Automation Conference*, pp. 776-779, 2001.

[21] T. C. Chen, T. C. Hsu, Z. W. Jiang, and Y. W. Chang, "NTUplace: A ratio partitioning based placement algorithm for large-scale mixed-size designs," *In Proceedings of International Symposium on Physical Design*, pp. 236-238, 2005.

[22] l. Savidis, S. M. Alam, A. Jain, S. Pozder, R. E. Jones, and R. Chatterjee "Electrical modeling and characterization of through-silicon vias (TSVs) for 3-D integrated circuits," *Microelectronics Journal*, vol. 41, no. 1, pp. 9-16, 2010.

[23] J. Cong, and G. Luo, "Thermal-Aware 3D Placement," *Integrated Circuits and Systems*, pp. 103-144, 2010.

[24] http://dropzone.tamu.edu/x̃iang/iscas.html

[25] W. C. Elmore, "The transient response of damped linear networks with particular regard to wideband amplifiers," *Journal of Applied Physics*, vol. 19, no. 1, pp. 55-63, Jan 1948.

[26] H. Ren, D.Z. Pan, and D.S. Kung "Sensitivity guided net weighting for placement driven synthesis," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systemsn*, vol. 24, no. 5, pp. 711- 721, May 2005.

[27] A. Y. Weldezion, R. Weerasekera, D. Pamunuwa, and L. R. Zheng, and H. Tenhunen "Bandwidth optimization for through silicon via (TSV) bundles in 3D integrated circuits," *In Proceedings Design, Automation and Test in Europe Conference*, 2009.

[28] M. Grange, R. Weerasekera, D. Pamunuwa, H. Tenhunen, and L. R. Zheng, "Closed-form equations for through-silicon via(TSV) parasitics in 3-D integrated circuits," *In Proceedings Design, Automation and Test in Europe Conference*, 2009.

[29] R. Hentschke, and R. Reis, "A 3D-Via Legalization Algorithm for 3D VLSI Circuits and its Impact on Wire Length," *In Proceedings of IEEE International Symposium on Circuits and Systems*, pp. 2036-2039, 2007.

[30] P. Spindler, U. Schlichtmann, and F. M. Johannes "Abacus: Fast Legalization of Standard Cell Circuits with Minimal Movement," *In Proceedings of International Symposium on Physical Design*, pp. 47-53 , 2008.

[31] A. Khatkhate, C. Li, A. R. Agnihotri, M. C. Yildiz, S. Ono, C. Koh, and P. H. Madden "Recursive bisection based mixed block placemen," *In Proceedings of International Symposium on Physical Design*, pp. 84-89 , 2004.

[32] C. T. Lin, D. M. Kwai, Y. Fa. Chou, T. S. Chen, and W. C. Wu "CAD Reference Flow for 3D Via-Last Integrated Circuits," *In Proceedings of the Asia and South Pacific Design Automation Conference*, pp. 187-192 , 2010.

[33] A. Marquardt, V. Betz, and J. Rose "Timing-driven placement for FPGAs," *In Proceedings of ACM/SIGDA eighth International Symposium on Field Programmable Gate Arrays*, pp. 203-213 , 2000.