

# 以整數線性規劃法解決 3D 超大型積體電路增量式平面規劃

梁峰碩 鄭威佑 程仲勝

大葉大學資工系  
大葉大學資工所  
大葉大學資工系

[r9906003@mail.dyu.edu.tw](mailto:r9906003@mail.dyu.edu.tw)

[r9806011@mail.dyu.edu.tw](mailto:r9806011@mail.dyu.edu.tw)

[jscherng@mail.dyu.edu.tw](mailto:jscherng@mail.dyu.edu.tw)

## 摘要

3D IC平面規劃雖然可以改善電路連線延遲的問題，但因電路系統設計日趨複雜，難以設計出一個面積、時序、功率和線長都最佳的佈局結果，因此很可能需反覆執行電路平面規劃，以求得一個較佳的結果，但這會導致後端實體設計所需的時間增加。為了縮短重複進行平面規劃的時間，增量式平面規劃(incremental floorplanning)策略已被提出用以改進先前平面規劃之缺點。

在本文中，我們嘗試以數學規劃法(mathematical programming)發展增量式平面規劃。首先依據原始平面圖找出欲增量模組與其周圍模組間的閒置空間大小，及分析可增量之範圍，並在不改變模組間拓撲關係及不增加整體晶片面積之條件下，找出最佳模組增量後之佈局結果。以上作法以整數線性規劃(integer linear programming)加以實現，並以LINGO軟體求解。實驗顯示提出的增量式平面規劃系統可有效將模組增量，並改善繞線延遲問題。

**關鍵字：**3D IC，增量式平面規劃，整數線性規劃

## 1. 簡介

積體電路設計已進入高效能、低功率、高密度以及異質整合的時代，傳統2D IC設計漸漸地已無法滿足現今IC產品之需求，因此IC設計朝向三維方式發展已是目前克服設計困難度的最有效方式之一。

在製程的微縮下，晶片內部擁擠程度增加，導致繞線變得很困難，因此利用3D IC晶片層的堆疊將可減輕IC擁擠程度及提供更多的繞線資源[1-3]。如圖1所示，假設原本2D SoC中連接A與C的網列為晶片時序臨界路徑，若將2D SoC轉換成3D IC，在平面規劃時，將C模組堆疊至A模組上方，可使連接A與C網列連線縮短，改善整體晶片效能。

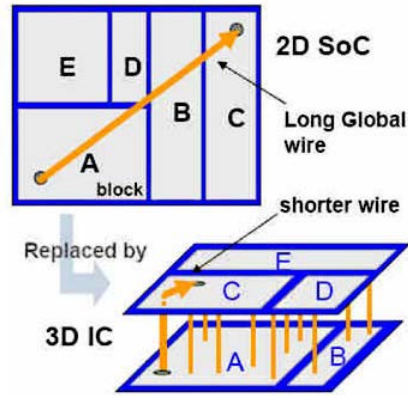


圖 1 2D SoC 架構與 3D IC 架構連線示意圖

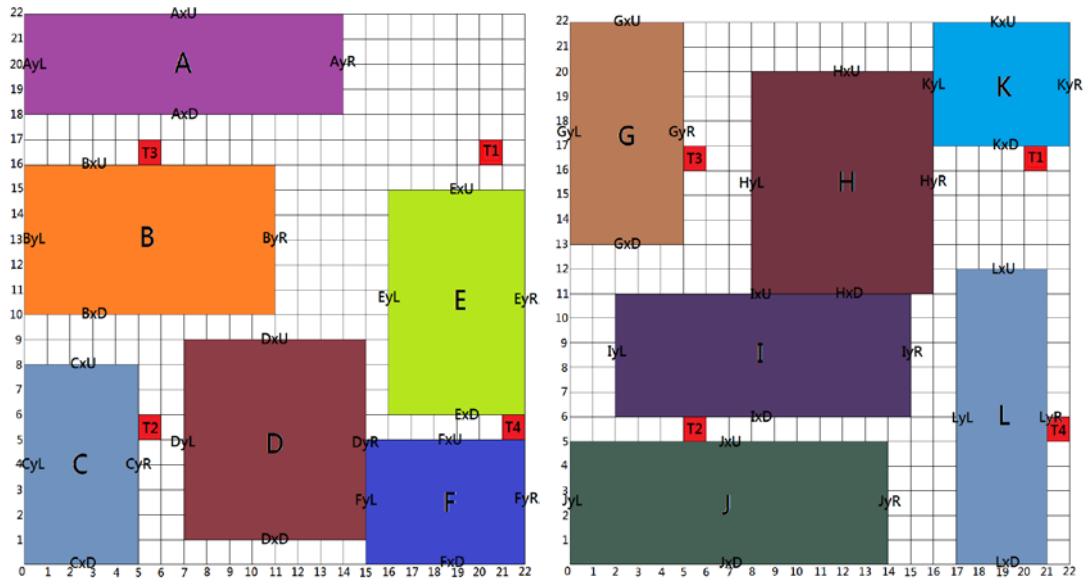
3D IC 平面規劃[4]雖然可以改善電路連線延遲的問題，但因電路系統設計日趨複雜，難以設計出一個面積、時序、功率和線長都最佳的佈局結果，因此很可能需反覆執行電路平面規劃，以求得一個較佳的結果，但這會導致後端實體設計所需的時間增加。為了縮短重複進行平面規劃的時間，增量式平面規劃(incremental floorplanning)策略已被提出用以改進先前所得之平面規劃中當某些模組位置、維度修改或少數模組間相對位置改變後，不論其修改範圍大或少，皆必須耗時地整個重新執行平面規劃之缺點[5-7]。綜上所述，所提論文的研究方向是設計一個適用於 3D IC 之增量式平面規劃系統，嘗試達到佈局最佳化以保障系統效能為目標。

本文中，使用整數線性規劃法來解決 3D 增量式平面規劃以及 TSV 擺放位置；首先會先得到一 3D 平面規劃初始圖以及初始圖相關資訊，以不改變晶片面積的條件為前提下，將模組增量面積、線長以及 TSV 擺放等轉換成線性問題，針對不同的需求給予不同的限制條件，使結果可達預期；實驗過程中，使用線性規劃將模組間的關連性、相對位置、線長對應以及 TSV 擺放記錄下來，以利於在增量過程中可針對特定需求做改變。

本文往後章節組織如下：第二節為問題描述，第三節將詳細說明如何使用線性規劃解決 3D IC 增量式平面規劃以及 TSV 擺放，第四節為實驗結果，第五節為結論與未來展望。

## 2.問題描述

先得到一個兩層之 3D IC 初始圖，並且得知每個模組寬、高以及模組之間的拓撲關係，如圖 2 所示，模組 A 寬為 14、高為 4、位於 TSV  $T_3$  上方。在不影響晶片面積的條件下，對電路模組做增量的動作。藉由整數線性規劃，得到最後 3D 增量式結果圖，使得模組增量面積，網列連線長度及相關 TSV 擺放位置最佳化。



(a)第一層平面規劃圖

(b)第二層平面規劃圖

圖 2 初始圖

圖 2 為 12 個模組(編號 A~L)所組成兩層初始圖，圖中白色部分為閒置空間(deadspace)，整個初始圖座落在 22×22 的格線座標上，其中假設模組 A、B、I 及 L 面積需增量 34、38、40 及 27 單位，網列資訊為網列 1(A、E、H、K)、網列 2(C、I、J)、網列 3(B、G、H)、網列 4(F、J、L)、網列 5(A、D、E)、網列 6(B、C、F)、網列 7(G、I、J)、網列 8(H、I、L)。

### 3. 整數線性規劃

在此節中將說明如何使用整數線性規劃(integer linear programming)來解決所提 3D 增量式平面規劃問題。模組在增量之前，必須先得知模組周圍是否有可擴增的閒置空間，在增量後模組間擺放仍不能互相重疊。基本上，3D 增量式平面規劃所處理的模組擴增問題為一個多目標最佳化問題，即需同時考慮模組面積擴展，網列連線長度以及 TSV 的擺放最佳化。

增量式平面規劃為一個二維最佳化問題，在此論文中，我們將此問題簡化為執行多個一維最佳化問題，亦即將解多個垂直方向及水平方向一維增量問題。以下將以圖 2 為例，說明如何將垂直方向一維擴增問題以整數線性規劃加以描述。另外，由於水平方向一維擴增問題之整數線性規劃描述方式與垂直方向類似，因此本文中不贅述。

目標函式配合多個限制式，可以將 3D 增量式平面規劃問題轉換為多目標整數線性規劃的問題，透過閒置空間的利用，可使模組增量，並局部重新擺放 TSV 使線長縮短，以下小節將會詳細說明目標函式及各種限制條件。

目標函式：

$$\max \{ \alpha \times \text{minslack} + \beta \times \text{totalslack} + \gamma \times \text{total\_Deltaarea} + \delta \times \Delta \text{wirelength} + \varepsilon \times \text{TSV\_Delta wirelength} + \omega \times \text{TSV\_Delta Location} \}$$

公式 1 中， $\alpha$ 、 $\beta$ 、 $\gamma$ 、 $\delta$ 、 $\varepsilon$ 、 $\omega$  為權重值，透過權重值的設定，可以在目標函式尋找最佳解時，決定是以線長、增量完成度、或是 TSV 擺放為優先。

目標函式架構上主要會分為三個部分：

- (1) 增量面積相關的變數： $minslack$ ：增量模組後，閒置空間最小的剩餘量。 $totalslack$ ：增量模組周遭剩餘閒置空間的總和。 $total\_Deltaarea$ ：為模組實際擴增面積。
- (2) 繞線長度相關變數： $\Delta wirelength$ ：增量模組未跨層繞線長度的變化量。 $TSV\_Delta wirelength$ ：增量模組跨層繞線長度變化量。
- (3) TSV 安插位置相關變數： $TSV\_Delta Location$ ：增量模組關連 TSV 與預估 TSV 最佳安插位置的位置變化量。

限制式：

#### (一) 位移限制

位移限制是避免模組在增量時超過所擁有閒置空間範圍。

$$lower_j \leq X_i \leq upper_j \quad (2)$$

公式 2 中  $X_i$  為增量模組  $j$  之上或下邊界， $lower_j$  與  $upper_j$  為固定常數，代表模組  $j$  在增量過程中可使用閒置空間範圍的邊界。

#### (二) 水平垂直限制

模組在增量過程中，會從初始圖中得到模組邊界拓撲關係，進而產生水平或垂直限制；水平垂直限制的目的是確保模組在增量過程中不會有相互重疊的情況發生。

$$X_i \leq X_j \quad (3)$$

公式 3 中， $X_i$ 、 $X_j$  為任意兩個相鄰模組的邊界，此限制不只能避免模組重疊，當增量模組周遭閒置空間不足時，也會推動臨近模組以增加閒置空間。

#### (三) 邊界擴展限制

當欲增量模組上下兩邊都有足夠的閒置空間讓模組增量時，此限制條件可讓模組平均的往上、下擴展，使閒置空間可有所保留，減少往後繞線困難度。

$$\begin{aligned} E_{ij} &\geq (upper_i - X_i) \\ E_{ij} &\geq (X_j - lower_j) \\ ME - E_{ij} &\geq M_{ij\_slack} \\ minslack &= \min (M_{ij\_slack}) \end{aligned}$$

$$totalslack = \text{sum}(M_{ij\_slack}) \quad (4)$$

公式 4 中  $E_{ij}$ 、 $M_{ij\_slack}$ 、 $minslack$  及  $totalslack$  皆為整數變數， $ME$  為常數，會依照增量模組周遭閒置空間大小而給定； $X_i$  及  $X_j$  分別代表同一模組未增量前的上、下邊界； $upper_i$  與  $lower_j$  分別代表同一模組增量後的上、下邊界；透過  $minslack$  和  $totalslack$  與目標函數配合可使增量模組往垂直方向做增量動作。

#### (四) 增量面積限制

為了避免模組在增量過程中，增量面積已達欲增量面積卻還繼續增量，因此給予限制條件，使其模組增量時，不會超過欲增量面積。

$$\begin{aligned} width_m \times [(upper_i - X_i) + (X_j - lower_j)] &\leq Expect\_AA_m \\ \Delta Area_m &= width_m \times [(upper_i - X_i) + (X_j - lower_j)] \\ total\_Deltaarea &= \text{sum}(\Delta Area_m) \end{aligned} \quad (5)$$

公式 5 中  $X_i$  及  $X_j$  分別代表同一模組  $m$  增量前之上及下邊界， $upper_i$  與  $lower_j$  分別代表同一模組  $m$  增量後之上、下邊界； $width_m$  代表增量模組  $m$  之寬度，而  $Expect\_AA_m$  則為期望模組所能增加的增量面積，透過  $total\_Deltaarea$  與目標函式配合，即可將增量面積控制在欲增量面積內。

#### (五) 寬高限制

為了使模組在增量過程中，避免模組因周遭閒置空間充裕而使模組單一朝垂直或水平方向擴展，導致拓撲關係有所改變，因此給予寬高限制。

$$\begin{aligned} Ratio_i &= width_{i\_after} / width_{i\_before} \\ ExpectMin\% &\leq Ratio_i \leq ExpectMax\% \end{aligned} \quad (6)$$

公式 6 中  $width_{i\_after}$  及  $width_{i\_before}$  分別代表模組  $m$  增量後與未增量的高(寬)值， $Expect\%$  為設定允許高(寬)值變化量，當設定值越低時，垂直與水平擴展越平均。以圖 2 為例， $Expect\%$  設定為 25%：

$$\begin{aligned} Ratio_a &= MaH/4 \\ Ratio_b &= MbH/6 \\ 0.75 &\leq Ratio_a \leq 1.25 \\ 0.75 &\leq Ratio_b \leq 1.25 \end{aligned}$$

#### (六) 繞線長度限制

在繞線長度的部分，會因為模組連線是否會有跨層的原因而分成平面繞線與 3D 繞線兩個部分，平面繞線是指網列中各模組位置都在同一層上，3D 繞線則是指網列中模組有透過 TSV 在不同層連線。繞線長度的估算是以 bounding box 的半周長去估算，bounding box 會以連線網列中所有模組的中心點都被包含在最小矩形範圍之中。

##### (1) 平面繞線

##### (2)

$$\begin{aligned} Line_x &= \max\{M_x, N_x\} - \min\{M_x, N_x\} \\ Line_y &= \max\{M_y, N_y\} - \min\{M_y, N_y\} \end{aligned}$$

$$Line_k = Line_x + Line_y$$

$$TotalLine = \sum_{k=1}^n Line_k$$

$$ML - TotalLine \geq \Delta wirelength \quad (7)$$

公式 7 中  $Line_k$  為單一網列 K 線長， $Line_x$ 、 $Line_y$  為 bounding box 的長與寬，線長估算方式為 bounding box 的半周長長度， $ML$  為原始所有網列線長， $TotalLine$  為所有網列估算的長度，透過  $\Delta wirelength$  與目標函式結合，可計算出同層中繞線長度並加以給予限制。以圖 2 中網列 5(A、D、E)、網列 6(B、C、F) 為例：

$$Line_x = \max\{A_x, D_x, E_x\} - \min\{A_x, D_x, E_x\}$$

$$Line_y = \max\{A_y, D_y, E_y\} - \min\{A_y, D_y, E_y\}$$

$$Line_5 = 10 + 15.5$$

$$Line_x = \max\{B_x, C_x, F_x\} - \min\{B_x, C_x, F_x\}$$

$$Line_y = \max\{B_y, C_y, F_y\} - \min\{B_y, C_y, F_y\}$$

$$Line_6 = 16 + 9.5$$

(3) 3D 繞線

$$LayerLine = TSV\_Line_x + TSV\_Line_y$$

$$TSV\_Line_k = Layer_1Line + Layer_2Line$$

$$TSV\_TotalLine = \sum_{k=1}^n TSV\_Line_k$$

$$TSV\_ML - TSV\_TotalLine \geq TSV\_ \Delta wirelength \quad (8)$$

公式 8 中  $TSV\_Line_k$  為單一網列 K 線長， $Layer_1Line$  為網列 K 在 Layer 1 半周長長度， $Layer_2Line$  為單一網列 K 在 Layer 2 半周長長度， $TSV\_ML$  為原始線長， $TotalLine$  為所有網列估計的長度。3D 繞線由於網列有透過 TSV 連接上下兩層模組，所以在估算半周長的部分，必須將兩層的 bounding box 相加。以圖 2 中網列 1(A、E、H、K) 為例，網列 1 將透過  $TSV_1$  連接上下兩層：

$$TSV\_Line_x = \max(A_x, E_x, TSV_{1x}) - \min(A_x, E_x, TSV_{1x})$$

$$TSV\_Line_y = \max(A_y, E_y, TSV_{1y}) - \min(A_y, E_y, TSV_{1y})$$

$$Layer_1Line = 10+9$$

$$TSV\_Line_x = \max(H_x, K_x, TSV_{1x}) - \min(H_x, K_x, TSV_{1x})$$

$$TSV\_Line_y = \max(H_y, K_y, TSV_{1y}) - \min(H_y, K_y, TSV_{1y})$$

$$Layer_2Line = 7+4$$

$$TSV\_Line_1 = 19+11$$

(七) TSV 位置限制

$$\begin{aligned}
 TSV_x &= \{\max\{M_x, N_x\} - \min\{M_x, N_x\}\}/2 + \min\{M_x, N_x\} \\
 TSV_y &= \{\max\{M_y, N_y\} - \min\{M_y, N_y\}\}/2 + \min\{M_y, N_y\} \\
 TSV\_length &\geq |TSV\_Best_x - TSV\_Actual_x| + |TSV\_Best_y - TSV\_Actual_y| \\
 TSV\_Range - TSV\_length &\geq TSV\_Location
 \end{aligned}
 \tag{9}$$

公式 9 中， $TSV\_Actual$  為 ILP 考慮兩層間置空間後，計算出 TSV 實際擺放位置， $TSV\_Best$  為預期 TSV 最佳擺放位置， $TSV\_length$  為 TSV 最佳擺放位置與實際擺放位置之間的長度， $TSV\_Range$  為固定值，為晶片的半周長，公式 9 透過  $TSV\_Location$  與目標函式配合，可以使與增量模組相關的 TSV 擺放位置達到預期。

將  $TSV\_bounding\ box$  定義為該 TSV 所屬網列中，晶片上下兩層所有模組的中心點，都被包含在最小矩形範圍之中，圖 3 為  $TSV\_bounding\ box$  示意圖，圖 3(a) 中，A、B、C、D、E 模組分別在兩個堆疊層中，有一條網列與 A、B、C、D、E 相連，圖 3(b) 在同一平面上建構出一個包含所有模組的最小矩形，而 TSV 安插最佳位置則為此矩形的中心點。

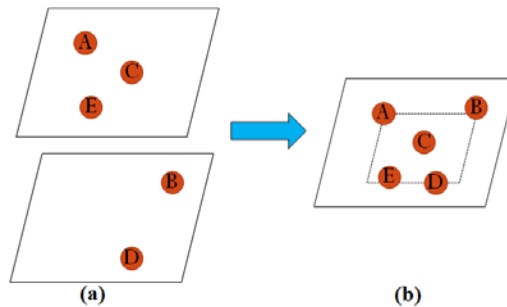


圖 3 策略  $TSV\_bounding\ box$  示意圖

以圖 2 中網列一為例，A、E、H、K 模組間，有一條網列連線。可得知最小矩形範圍為 A、K、E，則此矩形的中心點 ( $TSV_x, TSV_y$ ) 就為 TSV 擺放最佳位置。

#### 4. 實驗結果

本實驗目的為在不改變晶片面積限制下，將某些模組進行增量並且透過 TSV 的擺放有效的縮短線長。本文中，實驗所用的測試電路(benchmark)取自 MCNC 之 ami33 電路模組作為實驗，ami33 電路模組中，模組數為 33 個，繞線數為 123 條繞線；在實驗中，測試電路利用擺置器所規劃出來的 3D 初始圖(如圖 4 所示)進行增量式平面規劃，以不增加晶片面積的條件且不新增及刪除模組之情況下進行模組修改動作，模組修改時，將會以垂直方向與水平方向進行，若是水平方向修改完後，模組仍有未增量面積且模組周遭還存有足夠的閒置空間以供模組增量，將會再以垂直方向進行增量動作。本實驗以執行兩層增量式平面規劃為主，其中可對相關模組進行增量或減量動作。圖 5 及 6 分別顯示垂直及水平優先之 3D IC 增

量式平面規劃結果圖。而表 1 及 2 分別列出垂直與水平優先增量式平面規劃連線長結果變化，由兩表可看出兩種方式之連線長皆比初始連線長來的短。

## 5. 結論

在本論文中提出一個整數線性規劃法解決 3D 超大型積體電路增量式平面規劃，利用整數線性規劃的方式將模組變更，以縮短設計時間；透過模組與 TSV 相關資訊，局部變更 TSV 所安插位置，以縮短繞線長度，提升效能。

由於本論文在增量模組面積以及變更 TSV 安插位置上，是使用線性規劃方式，這使得應用在模組數量多的電路設計上，找出最佳解的時間會比其他演算法來的久，未來將可考慮透過如何將電路分割，利用階層式處理來縮短時間。

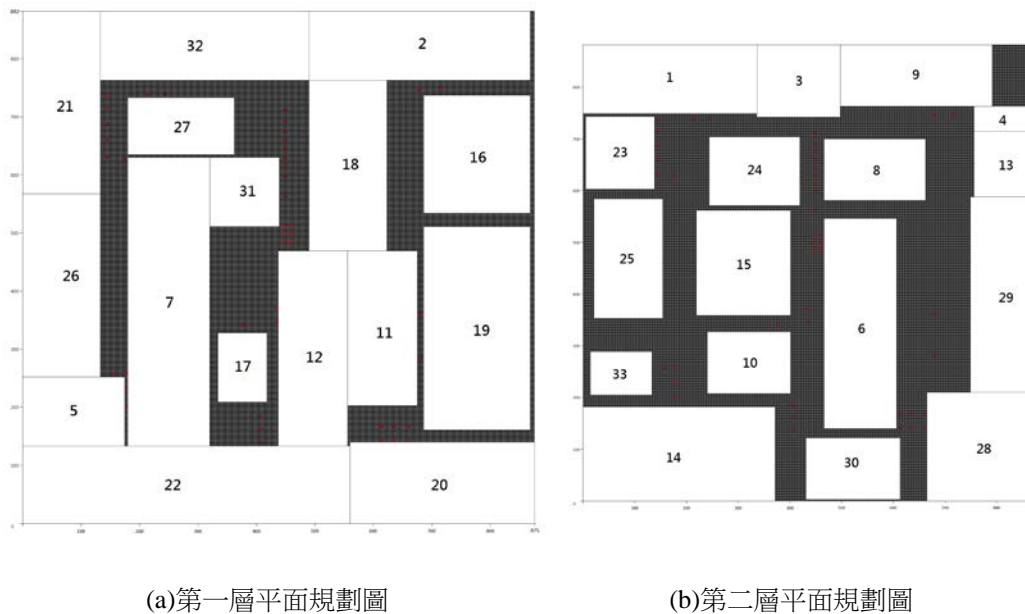
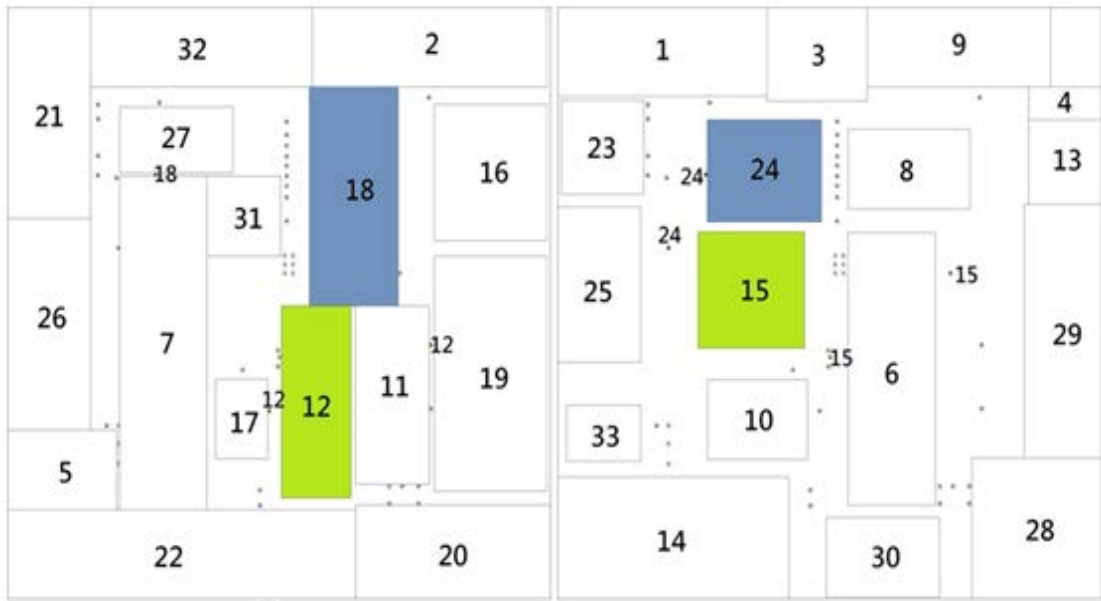


圖 4 AMI33 初始平面圖





(a)第一層平面規劃圖

(b)第二層平面規劃圖

圖 5 垂直優先增量式平面規劃結果圖



(a)第一層平面規劃圖

(b)第二層平面規劃圖

圖 6 水平優先增量式平面規劃結果圖

表 1 垂直優先增量式平面規劃線長結果變化

模組編號/ 擴增比例	原始連線長			增減量後連線長		
	同層	跨層	模組連線長	同層	跨層	模組連線長
12/-15%	154	1298	1452	132.5	1152	1284.5
15/-15%	0	2621.5	2621.5	0	2051.5	2051.5
18/0%	0	828.5	828.5	0	652	652
24/5%	1615.5	1647.5	3263	1613	1311.5	2924.5
線長加總	1769.5	6395.5	8165	1745.5	5167	6912.5

表 2 水平優先增量式平面規劃線長結果變化

模組編號/ 擴增比例	原始連線長			增減量後連線長		
	同層	跨層	模組連線長	同層	跨層	模組連線長
12/5%	154	1298	1452	128.5	1152	1280.5
15/-5%	0	2621.5	2621.5	0	2051.5	2051.5
18/20%	0	828.5	828.5	0	651.5	651.5
24/15%	1615.5	1647.5	3263	1613	1311	2924
線長加總	1769.5	6395.5	8165	1741.5	5166	6907.5

### 參考文獻

- [1] 3D IC stacking technology Mar. 2010. [http://www.2cm.com.tw/coverstory\\_content.asp?sn=1002260022](http://www.2cm.com.tw/coverstory_content.asp?sn=1002260022).
- [2] D. H. Kim, S. Mukhopadhyay, and S. K. Lim, "Through-Silicon-Via Aware Interconnect Prediction and Optimization for 3D Stacked ICs", SLIP, San Francisco, California, USA, July, 2009.
- [3] D. H. Kim, K. Athikulwongse, and S. K. Lim, "A Study of Through-Silicon-Via Impact on the 3D Stacked IC Layout", ICCAD'09, San Jose, California, USA, November 2-5 2009.
- [4] J. Cong and M. Sarrafzadeh "Incremental Physical Design" Proceedings of International Symposium on Physical Design. 2000.
- [5] L. Yang, Y. Ma, X. Hong, S. Dong and Q. Zhou "An Incremental Algorithm for Non-Slicing P. D. Franzon, W. R. Davis, M. B. Steer, S. Lipa, E. C. Oh, T. Thorolfsson, S. Melamed, S. Luniya, T. Doxsee, S. Berkeley, B. Shani and K. Obermiller, "Design and CAD for 3D Integrated Circuits", DAC, Anaheim, California, USA, June 2008.
- [6] Y. Liu, H. Yang and R. Luo "An Incremental Floorplanner Based on Genetic Algorithm" Proceedings of 5<sup>th</sup> International Conference on ASCI Vol.1, 2003, Pages 331-334.
- [7] Floorplan Based on Corner Block List Representation" Chinese Journal of Semiconductors Vol.26 No.12, 2005, Pages 2335-2343.