

# 行政院國家科學委員會專題研究計畫 期中進度報告

## 子計畫五：針對無線通訊上佈局驅動的資料路徑合成器之研究(1/3)

計畫類別：整合型計畫

計畫編號：NSC91-2218-E-009-013-

執行期間：91年08月01日至92年07月31日

執行單位：國立交通大學電子工程學系

計畫主持人：周景揚

報告類型：完整報告

報告附件：出席國際會議研究心得報告及發表論文

處理方式：本計畫可公開查詢

中華民國 92 年 5 月 23 日

針對無線通訊上佈局驅動的資料路徑合成器之研究

(1/3)

“The Study on Layout-Driven Datapath Synthesizer for Wireless Communication”

計畫編號：NSC 91-2218-E-009-013

執行期間：91 年 8 月 1 日 至 92 年 7 月 31 日

主持人：周景揚 交通大學電子工程系教授兼系主任

## 一、 中文摘要

由於半導體技術的突飛猛進，單晶片系統(SoC)已經變成了二十一世紀裡的主流科技，藉由單晶片系統的技術，可以加速促成三C(電腦、通訊、民生家電)之間的整合，這不管對企業或是科技來說，都是一股不可避免的潮流，對整個人類社會來說，三C的整合應用更是一個非常重要的進步。

以往人們的期望只是一個快速、可靠且容易使用的無線通訊系統，但現在人們的要求已不僅僅如此，他們期待能有更快速的方法來透過無線通訊存取資料，以便應付現在的各種應用，例如收發電子郵件、檔案傳輸協定、網路瀏覽器、甚至即時影像傳輸等等，對無線服務提供的廠商來說，它們若想繼續保有競爭力，就必須跟上這個時代潮流。在這個整合計畫中，我們主要就是針對 Beyond 3G(B3G)的無線通訊網路來做研究，特別是其中的單晶片系統技術。

在 B3G 的無線通訊系統中，如何設計一個速度又快、面積又小的乘法器是非常重要的，因為乘法器是系統中許多重要部分如中央處理器(CPU)、數位訊號處理器(DSP)、快速傅立葉轉換器(FFT)裡的關鍵元件。

在前一年的研究中，我們提出了一個以佈局導向為考量的自動乘法產生器。對於時序的計算，我們採用了以單位元件為基礎的延遲模型(cell-based delay model)，而非傳統電路設計中被廣為使用的互斥邏輯閘的延遲模型(XOR-based delay model)；同時，在電路合成的過程中，加入了繞線延遲(wire delay)的效應，使得時序的計算將能更貼近實際電路的情形。除此之外，在電路配置(placement)的過程當中，不僅對於時序做了最佳化，還同時考慮了電路的形狀，使其能趨近於方正。藉由將電路合成、配置以及再合成等步驟整合在乘法產生器的流程，我們將可以得到優於其它傳統作法所產生的乘法器。

在此次計畫中，我們提出了由誤差大小來決定硬體的自動乘法產生器。藉著容許些許的誤差，我們可以用更小的面積來實現更快速的乘法器。考慮到信號到達時間的曲線圖，我們也為以時間為考量的繞線和電路配置(placement)提出一些技巧，來做時序最佳化。在乘法產生器的流程中，藉由整合電路合成、配置以

及再合成的步驟，我們將可得到優於其他傳統作法產生的乘法器。

關鍵字

乘法器 量子化誤差 雙極吻合

## 二、 Abstract

With the advent of semiconductor technology, the System-on-a-chip (SoC) becomes a mainstream and focus of technologies on coming 21<sup>st</sup> century. The integration of Computing, Communication and Consumer Electronics will be speeded up through SoC, which causes the unavoidable trend of new era for technology and enterprises. Especially, the application of 3C integration is very important to the progress of human being society.

People all over the world have come to expect fast, reliable, and easily accessible wireless communications and now they are demanding faster ways to access data with applications as diverse as e-mail, file transfer protocol, Internet browsers, and even real-time video teleconferencing. Wireless service providers who want to remain competitive must keep pace with this demand. In this integrated project, we are targeting on the study of beyond 3G wireless networks. Particularly, we are focusing on **SoC technologies for OFDM-based SDR baseband processing**.

To design fast, area-efficient and low power multipliers is important because multiplication is a key operation in many processors such as **CPU, DSP and FFT/IFFT** processors for the **wireless communications**.

In previous research, we presented an automatic layout-driven multiplier generator. The cell-based delay model, rather than the XOR-based model, is used for timing estimation and the wire delay is also considered in the synthesis process. The timing optimization, by considering the shape of the circuit, is conducted in the placement process. Final adder is produced together with the column compression tree

for getting the simpler structure. By integrating synthesis, placement and resynthesis processes in the multiplier generation flow, the multipliers generated by our layout-driven multiplier generator outperform other previous works as shown in our experimental results.

In this project, we present an automatic error-controlled hardware-configurable multiplier generator. The determination of the hardware of a multiplier is based on the error constraint given by users. With allowing some rounding errors, a significant reduction in area and delay can be achieved. By considering signal arrival profile, we also proposed several techniques for timing driven routing and placement to optimize the timing. By integrating synthesis, placement and resynthesis processes in the multiplier generation flow, the multipliers generated by our multiplier generator outperform other schemes used for comparison as shown in our experimental results.

**關鍵詞:** multiplier, quantization error, bipartite matching

### 三、 計畫的緣由與目的

一般有兩種設計流程，一種是 cell-based 設計流程，設計者使用 Verilog/VHDL 描述抽象的硬體，經由自動的合成，放置以及繞線，這樣的設計流程，設計時間縮短了，但是設計出來的電路，在時序上以及面積都比較差，第二個流程是 full-custom 設計流程，設計出來的電路，可以達到相當高的品質，但是這種設計流程需要耗費相當多的人力以及時間。而且因為大量的人為操作，容易發生錯誤，因此需要耗費相當多的時間作驗證。

在深次微米時代，除了要考慮元件的延遲之外，還要考慮繞線的延遲，但是連線的延遲，在合成的時候，很難精準的估計時序，所以一般會把合成跟放置合在一起做，稱之為 physical synthesis，我們更加的延伸，把電路分成很多部分，每個部分都做合成與放置，這樣的方法可以達到更好的效果。

乘法器是一個很特殊的運算元，有很多的地方可以去調整它，但是需要一個系統化的方法，可以對 timing, area, power 做一些最佳化的設計，然而很多的問題是 trade-off 的，我們需要尋找一個好的平衡點，以達到我們的要求，因此在乘法運算上，我們會去分析 quantization noise, area, timing, power 等 trade-off 的影響，然後提出一些有效的演算法。

#### 四、 研究方法及成果

##### 1. Generation Flow

我們從 LSB 端的 VCS 做起，每一個 VCS 做 column compression tree generation, resynthesis, 做完這個 VCS, 再做下一個 VCS, 這樣的設計主要是因為 VCS 是一片一片的結構, 使用這樣的流程可以利用此特性, 這是一般電路所沒有的。在圖 1 可以看到整個 flow.

---

MULTIPLIER\_GENERATION()

#### **Begin**

```
1 FUNCTION_ANALYSIS_and_PARTIAL_PRODUCT_GENERATION()
2 BUFFER_INSERTION()
3 PLACEMENT()
4 For ( each VCS in increasing-weight order)
5   Do CCT_CONNECTIVITY_GENERATION()
6     REWIRING()
7     ROUTING_and_TIMING_ANALYSIS()
8 FINAL_ADDDER_ADJUSTMENT()
```

#### **End**

圖 1



## 2. Number System:

數字的表示，會影響到所需要的功率消耗，因此在電路設計時可能需要使用不同的表示法，因此也需要不同表示法的運算器，我們使用 2 補數來表示有號整數，我們的 partial product 產生器可以輕易的處理有號整數以及無號整數。

## 3. Truncated Multiplication:

在訊號處理中, fixed-point 的數字可以容忍一部分的誤差,例如 FFT 中,一定會有 quantization 的部分,而分析整個系統讓每個地方的誤差都在可容忍的範圍,需要做系統的模擬,在系統模擬中,可以分析出 quantization error V.S. cost 的 trade-off,這裡的 cost 可以是 area 可以是 power, 在系統模擬中,會去評估可容忍的 quantization 量,而使用 Truncated Multiplication 會是一個重要的技巧可以使系統 cost 大量的減少.

但是 quantization 量是跟 input 的分布有關係,例如 FFT 的分布並不是 uniform distribution,所以我們需要 input distribution 作為我們設計的依據,而 multiplication 中 partial product terms 的選擇是一個最佳化的問題, 我們使用 simulation-based 的演算法,來尋找對 output 最具影響力的 partial product terms.

在下圖中,我們會有 input distribution,我們會利用此分布找出每個 product term 的貢獻度,以決定哪些 product term 會保留下來,之後計算出他的誤差部分,因為 truncation 完之後 error 並不會是 random 的誤差,所以需要找出需要補償的部分,最後才會做 rounding 來達到規格的要求.

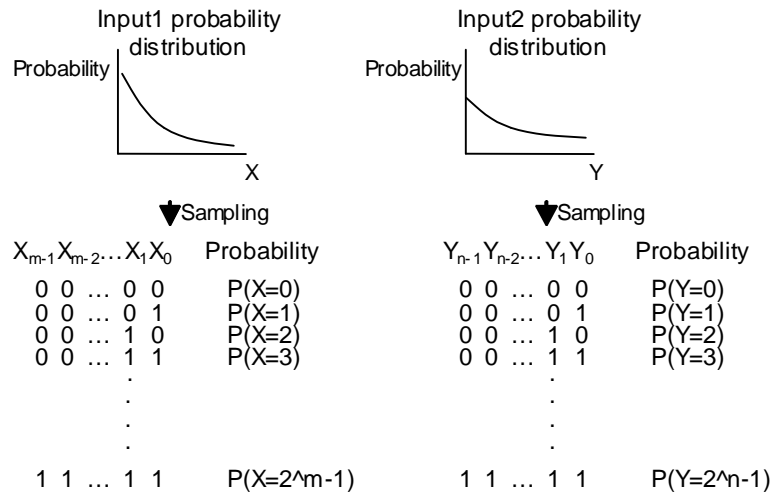


圖 2

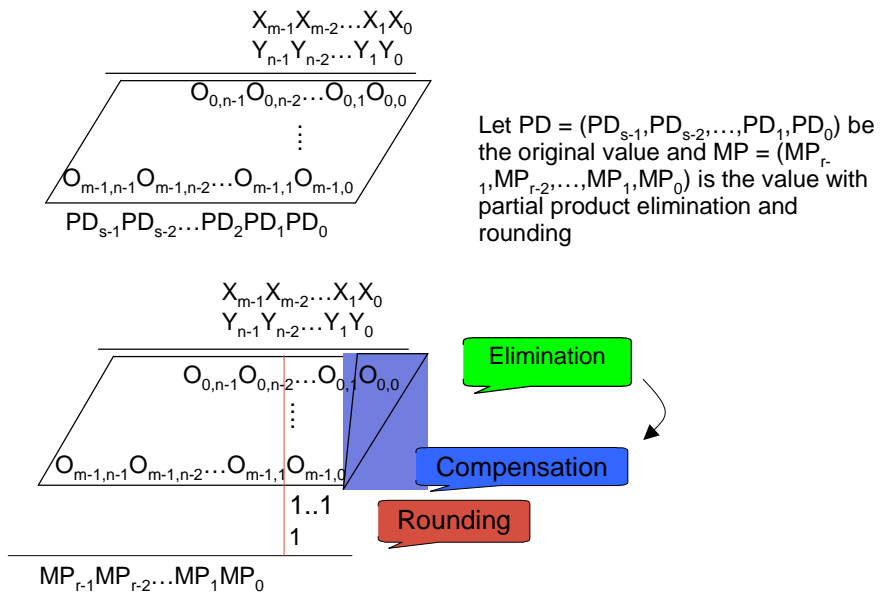


圖 3

下圖是一個設計流程圖，此圖描述了整個自動化的流程，可以利用下面的流程，根據 input distribution, input and output 的 word length, 找出最佳解。

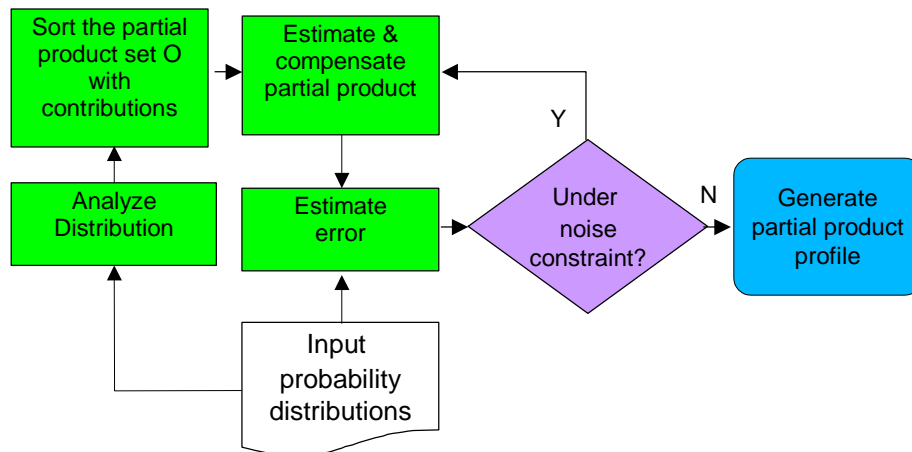


圖 4

#### 4. Timing Model:

在這個計畫中，我們使用 cell-based 的時序模型，這樣可以比使用 XOR-based 的模型更加的準確，而且此模型可適用於不同的 cell library。

在連線的延遲估計上，我們使用  $\pi$ -model 來模擬電路的結構，以及使用 elmore delay 的估算，來計算連線的延遲。

#### 5. Column Compression Tree Generation:

給定每個 compressed terms 的到達時序，以及 cell 的資訊，建立一個最佳的 tree，這個 tree 有最快到達的 sum 以及 carry。

在這個問題上我們提出的 top-down 的演算法，這方法可以對整個 tree 做最佳化，相較於 bottom-up 的方法，只能使用一些 greedy 的演算法，我們提出的演算法是比較具有整體性的演算法。

#### 6. Placement

乘法器具有 VCS 的結構，每個 VCS 跟前後 VCS 之間有連線關係，跟其他的 VCS 沒有連線，所以在放置的時候，針對這個特點，我們選定的放置位置也

是具有這樣的特色，也就是每個 VCS 跟之前與之後的 VCS 在位置上是相連的。

一般的 block 需要設計成方形，這樣在做 floorplan 的時候，處理上比較簡單。因為我們的乘法器產生器是可以輸出不同 bit-length 的乘法器，我們需要動態的決定每個 VCS 所使用的空間，因此找出一般性的通式是最重要，我們找出一些規則，可以把空間配置成方形，並且每個 VCS 還是保有原有的特性。

在考量到深次微米的繞線問題時，平行四邊形(如圖 5)的結構卻是比較適當的，因為整個 column compress tree 會在一直線上面，而且 carry out 會有很短的距離，然而平行四邊形的結構因為不是方形結構，在 placement 或者是 floorplan 上會比較困難，然而在整個系統中，乘法器只是一個小元件，所以可以把這部分跟其他的元件整合在一起，而乘法器會是 timing critical 的地方，所以他的結構比較緊密，而其他結構比較鬆散，會是一個比較好的設計，因此我們會把此乘法器內的各個相對位置固定，而讓其他電路散在各個地方。

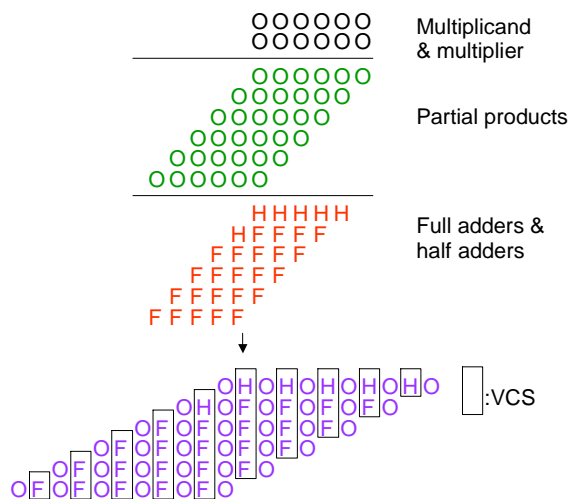


圖 5

## 7. Resynthesis:

因為在 synthesis 的時候，估計的線路延遲是不準確的，為了更進一步的改

善，我們交換相同 weight 的連線，來達到整體的最佳化，首先要找出可以交換的連線來，之後再決定如何的交換，我們提出系統化的方法，針對連線的到達時序，來決定可交換的連線，至於交換的配對，是把此問題變成一個 minimax 的數學問題。

如何的選出可以交換的連線是這個問題中最難的部分，並不是所有相同 weight 的連線都可以交換，因為不適當的交換會讓電路有不正確的 cycle 導致 function 錯誤，我們提出兩個方法，首先，會找出 critical path 來，之後會把這條 path 跟其他可能的部分作交換，如圖 6 所示，可以尋找出 arrive time 相近的 edge，來做交換，因為 arrive time 相近的最有機會可以互換，如果兩個 edge arrive time 差異很大，那換過去，另一邊會變成 critical path，這樣就沒有好處。

另一個方法是把 critical path 自己重組，因為第一個方法無法做大幅度的變動，所以需要最此 critical path 做重組的動作，如圖 7 所示，可以把他的結構變更。

利用這兩種方法，就可以把 critical path 的 timing 縮短。

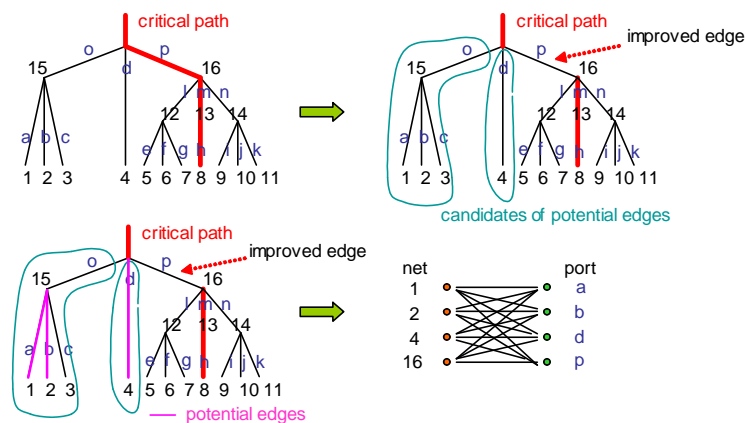


圖 6

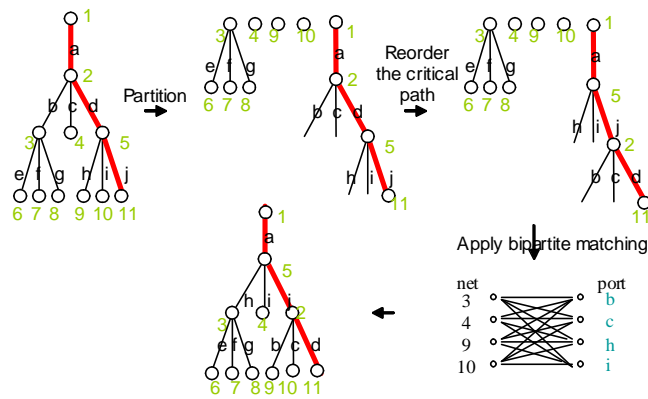


圖 7

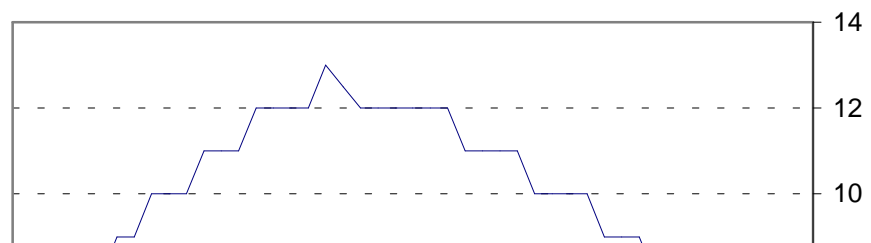
### 8. Speedup for the Final Adder:

乘法器在最後面需要加法器，但是由於此加法器的 input arrival time 是不一致的，因此需要特殊的加法器來處理。根據不同的 arrival time，我們會去分析 critical path，然後加入適當的 look-ahead 電路來加快速度。

### 9. Buffer Inserting

在乘法器中的 partial product 的產生，會有相當多的 gate 會需要同一條訊號來推動，這時候 buffer inserting 就會非常重要，但是 buffer 要放在什麼地方，要放多少卻是最佳化的問題，我們嘗試去分析，以 timing 為最重要的考量，也就是，不是 critical path 的部分就應該使用 buffer 切斷分開，使得 load 因此可以降低，然後直接使用大的 buffer 推動整個 critical 的部分，這是因為乘法器的特性；有相當多的 path 都是 critical path.

在下圖中，可以看到 critical delay 會發生在 column 29 的位置，所以 global line 在 column 29 以後就會使用 buffer 隔開，以致於可以減少 load.



## 五、 結論與討論

在本報告中，我們提出了一個完整的流程，可以自動產生出在誤差容忍度限制下之最佳固定寬度乘法器，初步的成果相當令人滿意。研究結果已陸續投稿至相關會議及期刊。而之前相關成果也陸續發表在會議中[1,3]以及投稿至國際性期刊[2]。

表一簡列近年本研究群的相關研究成果。89 年發表會議論文 5 篇，期刊論文 6 篇，並有 4 篇論文於 IEEE 期刊發表，90 年發表會議論文 10 篇，期刊論文 3 篇，並有 3 篇論文於 IEEE 期刊發表，91 年發表會議論文 8 篇，期刊論文 3 篇，並於 IEEE 期刊發表 2 篇論文。92 年及以後發表會議論文 6 篇，期刊論文 5 篇，並有 3 篇論文於 IEEE 期刊發表。

Year	Number of Papers				
	Domestic		International		SCI
	Conference	Journal	Conference	Journal	
2000	0	0	5	6 (IEEE:4)	6
2001	2	0	8	3 (IEEE: 3)	3
2002	3	0	2	3 (IEEE:2)	3
2003>			6	5 (IEEE: 3)	5

表一、本研究群近年相關研究成果

## 六、 參考文獻

- [1] Cheng-Yeh Wang, Ya-Chi Yang and Jing-Yang Jou, "An Automatic Layout-Driven Multiplier Generator," the 13th VLSI Design/CAD Symposium, Hsinchu, August 2002.
- [2] Cheng-yeh Wang, Ya-chi Yang and Jing-yang Jou, "Layout-driven Automatic Multiplier Generation," submitted to IEEE Trans. on VLSI.
- [3] Cheng-Yeh Wang, Ya-Chi Yang and Jing-Yang Jou, "An Effective Physical Synthesis Technique for Multiplier," International symposium on VLSI Technology, Systems, and Applications, April 2003.