

無進位式高速除法器之演算法則與硬體架構之研究與發展  
**The Design and Implementation of the Algorithm  
 and Architecture of the Carry-Free Divider**

計畫編號：NSC 88-2215-E032-001

執行期限：87年8月1日至88年7月31日

計畫主持人：江正雄 淡江大學電機系副教授 Email:chiang@ee.tku.edu.tw

### 一. 中文摘要

在傳統減法式的除法器 (Subtractive Divider) 中，往往需要一個減 (加) 法器來判斷部分餘數 (Partial Remainder) 的正負，或比較除數和部分餘數的大小，以決定部分商數的值。因為加法器中進位延遲常常成為此除法器效能的瓶頸。本研究計劃提出一個新式除法器演算法與架構，改善傳統除法器中疊代次數和選商的方法，以提高除法器的效率。

本研究計劃克服傳統除法器的缺點，將利用有號數 (Signed Digit) 及查表的方法來降低疊代的次數和進位的問題；在算術運算中，若除數大於  $1.1_2$  則被除數和除數同時乘上一個倍率 (Prescale) 0.75 以避免超出位元上限 (Overflow)，並利用重寫 (Rewrite) 來正確判斷下一次正確的運算方式，在算術運算中也利用平行運算處理來提高除法器的速度。

本研究計劃所提出的除法器演算法及架構，均使用 C 語言和 Verilog 來模擬驗證，由模擬驗證的結果顯示我們所使用的除法器演算法和架構效能良好，且提供了一種簡易且富彈性的除法器架構，此法可擴充到任何的位元數，使其易於在超大型積體電路 (VLSI) 中實現，改變傳統除法器的運算方式，提高除法器的速度，進而提昇算術處理器在作除法上的效能。

#### 關鍵詞：

有號數，位元上限，超大型積體電路，除法器，部分餘數，倍率，重寫，進位

#### Abstract:

In the carry-free design, we use tabular method to process the carry problem. By this approach, the division operation has the characteristic of parallelism and the speed can

be increased. The division is classified as non-restoring, therefore, signed bit is used in both the quotient and partial remainder.

The proposed algorithm and architecture are simulated by C language, and Verilog HDL respectively. The C simulation shows the result is correct. For the architectural implementation, we design a 32-b/32-b divider in Verilog HDL. From the Verilog HDL simulation, we find the speed and efficiency is very well. The Verilog HDL code is synthesized by the SYNOPSIS and the VLSI Layout (TSMC'S 0.6  $\mu$  m process) is also finished.

#### Key word:

rewrite, prescale, carry out, Verilog, VLSI, signed digit, divider, overflow

### 二. 計劃緣由與目的

在傳統的除法器中，需要一個加 (減) 法器來判斷部分餘數 (Partial Remainder) 的正負，以決定該部分商數的值，為達到高速運算，此加 (減) 法器須特別設計，而加法器中之進位延遲常常成為此除法器效能的瓶頸。我們提出一個新的演算法則與架構，使整個運算無進位延遲 (Carry Propagation Delay)，並以具有平行處理架構的方式、降低疊代運算次數、和簡化選商的方法來完成系統的算術運算，以提昇除法器的速度。

一般除法器，常有限制除數和被除數關係的範圍值，這是為了避免運算過程中發生溢位 (Overflow) 的現象。我們利用倍率 (Prescale) 的方法來避免溢位的問題，進而解決了除數和被除數之間關係限制的範圍值。對於進位延遲問題，利用兩次查表的方法來解決，也利用更新 (Refresh) 和重寫 (Rewrite)，再配合部分餘數之首位元 (Most Significant Bit, MSB) 來正確判斷下一次的運算方式，可以除去不必要的疊代運算，

同時也可根據部分餘數的 MSB 來選取正確部分商數；最後則利用 On The Fly Conversion [11] 的方法來得到正確的商數。

### 三. 研究方法及成果

本除法器在電路實現中，包括了：「倍率電路」(Prescale)、「加法電路」、「8-Bit 查表電路」、「進位選擇電路」、「4-Bit 查表電路」、「更新電路」(Refresh)、「重寫電路」(Rewrite)、「商的處理電路」、「控制電路」(Control)。

首先將被除數和除數利用 IEEE-754 作正規化處理，接著判斷除數的最高兩位元 (MSB) 是否同時為 1，若成立則表示運算過程中會發生溢位的現象，因此要經過倍率的處理。倍率的處理就是被除數和除數同時乘上 0.75，即利用減法來達到乘上 0.75 倍率的運算(乘上 0.75 相當乘上  $\frac{3}{4}$ ，而右移兩個 Bit 在算數上等於乘上  $\frac{1}{4}$ ，即利用  $1 - (\frac{1}{4}) = \frac{3}{4}$  的特性，用減法來達到作乘法的運算)，做法是利用原數減去原數右移兩 Bit 的值。

算術運算的最大瓶頸就是進位延遲的問題，本除法器架構中的加法器，利用兩個 Bit 來表示一個位元，使加法運算過程中完全無進位延遲的問題，再經用平行運算來提昇加法器的速度，避免因加法運算的進位而造成下一組加法運算的延遲，使每個模組的加法運算互相獨立，因此其速度可以變的非常快。

利用 8-Bit 查表來處理第一次進位的問題，並且作簡化位元的功用，接著再配合多工器和 4-Bit 查表把整體進位後的值查出。本除法器內用到的儲存資料，並不需要大量儲存記憶電路，以 PLA 作為本除法器查表法的硬體實現之電路架構。在考慮降低電路設計的複雜度、以及運算速度上的要求等各項因素之後，決定以靜態式的 PLA 電路做為查表電路的硬體實現方式[13, 14]。

8-Bit 的查表，只查出該級運算後的進位輸出 (Carry-Out) 和進位後的值，因此要求出整體的進位值，需再經過進位的電路推算出每一級進位的值，提供給 4-Bit 的查表並配合多工器 (Multiplexer) 的選擇，才能算出整體位元進位後的值。

4-Bit 的查表是根據進位選擇的值，來選出運算後整體進位後的真正值，以提供下一次運算正確的值。

本除法器電路中，作減法運算時，利用把除數加到負的部份餘數上來達到作減法的運算，因此最後還要利用更新電路，把下一次運算正的部份餘數和負的部份餘數的值正確的算出。當正的部份餘數和負的部份餘數相同位元上的值同時為 1 時，兩者就相消成 0。如此便可以降低除法器中，對加、減法器在硬體上使用複雜的設計。

重寫是把正的部份餘數和負的部份餘數最高的兩位元重寫。重寫的目的在除去不必要的疊代運算，正確的決定下一次的運算方式，和正確的選取下一次運算得到的商。

在處理商數時，用到兩個暫存器來儲存運算過程中得到的商數。其中一個暫存器功是真正儲存目前的商數，另一個暫存器則是用來儲存暫態的商數。儲存暫態商數的暫存器最大的功能就是當產生 (-1) 的部份商數時，可利用其儲存的值，左移一位再將 1 加在最後一個位元上，此結果就是原商數減去 1 的值。此法可降低電路設計的複雜度和能快速的得到我們想要的商數。

控制電路在除法器中，扮演偵測是否整除和是否達到需要的計算次數，當檢查到正的部份餘數和負的部份餘數均為 0 時，則表示被除數/除數是整除，就不要再做下一次的運算，否則利用計數器來控制達到所需的運算次數；當計數器的值未達到運算的次數和正的部份餘數、負的部份餘數不為零時，就繼續執行下一次的運算，反之就停止一切的運算。當不能整除時，計數器可用來控制除法運算的精確位數。

接著進行整體電路的邏輯功能模擬驗證，以確定各電路單元的功能無誤與控制信號是否能精確地驅動整體電路，達到正確的除法運算。如圖 3 所示。在整體電路的邏輯功能模擬驗證無誤後，即引入細胞元資料庫 (Cell Library) 進行電路合成 (Synthesis)，然後將合成的電路以 Verilog 格式儲存成 \*.vp 之檔案型式，再進行配合細胞元資料庫 (Cell Library) 的 Verilog 模擬，加上了實際細胞元電路的時間延遲考量，以進一步模擬接近實際電路的動作情形。如圖 4 所示。使用時脈信號的週期為 7ns，所以使用查表系統的除法運算速度為 143MHZ。

#### 四. 結論與討論

我們所使用的細胞元資料庫 (Cell Library) 是國科會晶片製造中心 (CIC) 所提供的, 此細胞元資料庫是使用 0.6 微米單層矽晶參層金屬半導體電路 ( $0.6\mu\text{m}$ 、 $1\text{p}3\text{m}$  CMOS)。整體電路的佈局, 首先將各模組電路的\*.vp 檔由 Cadence 的 Verilog In 來產生符號 (Symbol) 圖和電路 (Schematic) 圖, 然後利用各模組電路的符號圖和 PLA 電路圖的符號圖, 加上輸入、輸出的接腳電路圖等所構成的電路圖開始進行自動佈局與繞線 (Auto Place And Route)。部分電路佈局圖如圖 1、圖 2 所示。圖 5 為完整晶片的佈局圖, 其面積為  $3250 \times 3250\mu\text{m}^2$ , Gate 數目為 5783 個。

在這篇論文裡, 我們嘗試結合了 [6]、[15]、[16] 架構上的優點與特性而設計出新的架構來實現除法的運算, 在我們新的架構裡, 利用倍率 (Prescale) 來避免溢位 (Overflow) 的發生, 而倍率中用到的乘法器使用移位和減法器來完成, 有效的降低用乘法運算所需的時間; 利用兩次查表來避免進位延遲, 如此便可提昇加法器的運算速度; 利用更新 (Refresh) 處理來做疊代運算中的減法運算, 以提昇減法運算的速度; 利用重寫 (Rewrite) 處理來正確判斷下一次的運算方式和正確選出部份的商數, 以有效降低不必要的疊代運算和選商的時間; 整體的除法器都是使用基本的邏輯觀念來設計完成, 同時硬體上的架構極富彈性和模組化, 所以易於在超大型積體電路 (VLSI) 中實現。

#### 五. 參考文獻

- [1] M.D. Ercegovic and T. Lang, "Division and square root: digit-recurrence algorithms and implementations," *The Netherlands: Kluwer Academic Publishers*, 1994.
- [2] E. E. Swartzlander and Jr. "Computer arithmetic," vol. 1, Los Alamitos-California, *IEEE Computer Society Press*, 1990, pp. 156-157.
- [3] Kai Hwang, *Computer Arithmetic Principles, Architecture, and Design*, John Wiley and Sons, 1979.
- [4] A. E. Bashagha And M. K. Ibrahim, "A new digit-serial divider architecture," *Int. J. of Electronics*, vol.75,n1,pp. 133-140,Jul 1993.
- [5] Jordi Cortadella And Tomas Lang, "High-radix division and square-root with speculation," *IEEE Trans. On Computers*,vol.43 no.8,Aug.1994.
- [6] N. Burgess, "A fast division algorithm for VLSI," *IEEE Int. Conf. On Computer Design*, PP.560-563,1991.
- [7] M. D. Ercegovic and T. Lang, "Simple radix-4 division unit with operands scaling," *IEEE Trans. Computers*, vol. 49,pp.1204-1208,Sept. 1990.
- [8] P. Montuschj and L. Cimiera, "Over-redundant digit sets and the design of digit-by-digit division units," *IEEE Trans. on Comp.*, vol. 43, no. 3, pp. 269-3279, March. 1994.
- [9] T. E Williams and M. .A. Horovitz, "A 160ns 54-bit CMOS division implementation using self-timing and symmetrically overlapped SRT stages," in *Proc.10th IEEE Symp. Computer Arithmetic*, Grenoble, France, June 1991,pp.210-217.
- [10] J. B. Kuo H. P. Chen, H. J. Hunang "A BiCMOS dynamic divider circuit using a non-restoring iterative architecture with carry look-ahead for CPU VLSI," 1993 *IEEE Int. Symposium on circuit and systems*,pp.2027-2030.
- [11] M. D. Ercegovic, T. Lang. "ON-the-fly conversion of redundant into conversion representations," *IEEE Trans. on Computer*, vol. C-36, No. 7, 1987, pp.895-897.
- [12] C. Tung, "Signed-digit division using combination arithmetic nets," *IEEE Trans. Comp.*, vol. c-19, no. 8, pp. 746-748, Aug. 1970.
- [13] G. M, Blair, "PLA Design for Single-Clock CMOS," *IEEE JSSC*, vol.27, no.8, Aug.1992.
- [14] N. Weste, K. Eshraghian, *Principles of CMOS VLSI Design: a systems perspective, 2<sup>nd</sup>*, Addison-Wesley Publishing Company, 1993.
- [15] D. Ercegovic and Tomas Lang "Simple Raduix-4 With Scaling," *IEEE Trans. Comp.*, vol.39, no.9, pp. 1204-1208, Aug. 1990.
- [16].Hosahalli R. Srinivas and Jeshab K. Parshi "A Fast Radix-4 Division Algorithm and its Architecture," *IEEE Trans. Comp.*, vol. 44, no. 6, pp. 826-831, Aug. 1995

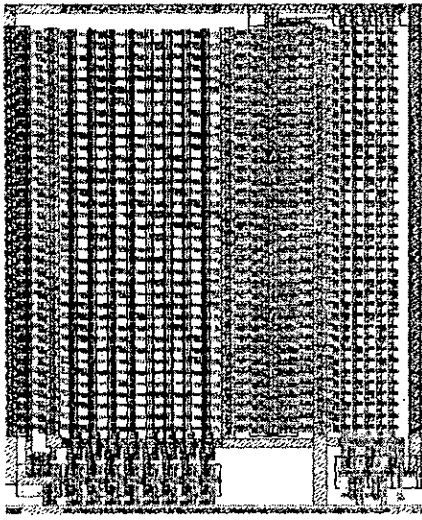


圖 1：8-Bit 查表的佈局圖

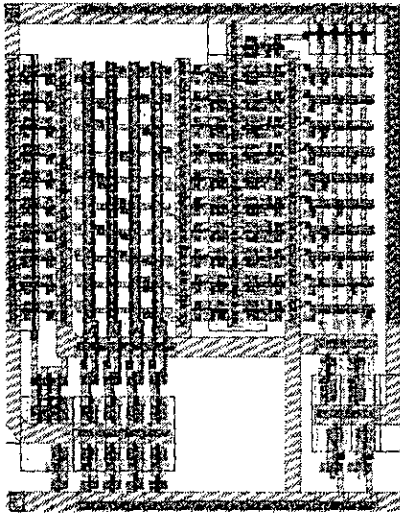


圖 2：4-Bit 查表的佈局圖

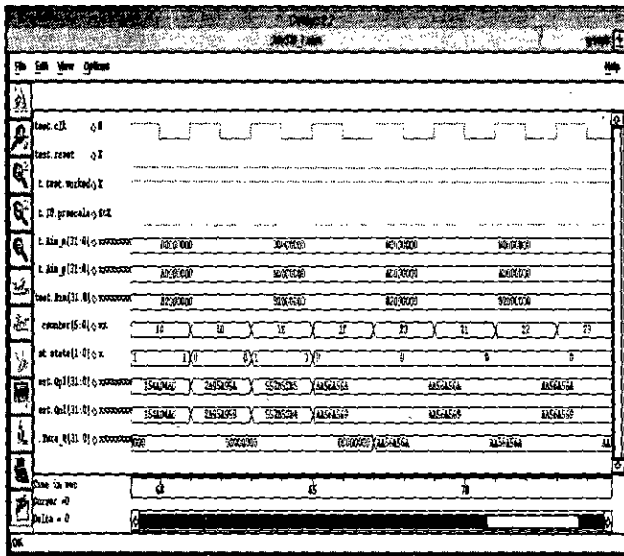


圖 3：除法器整體電路的 Verilog-XL 邏輯功能模擬波形圖

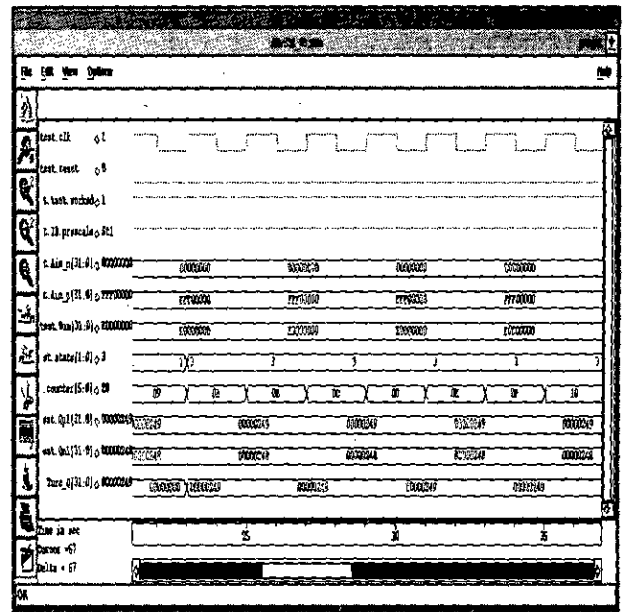


圖 4：除法器整體電路的 Gate-Level 模擬波形圖

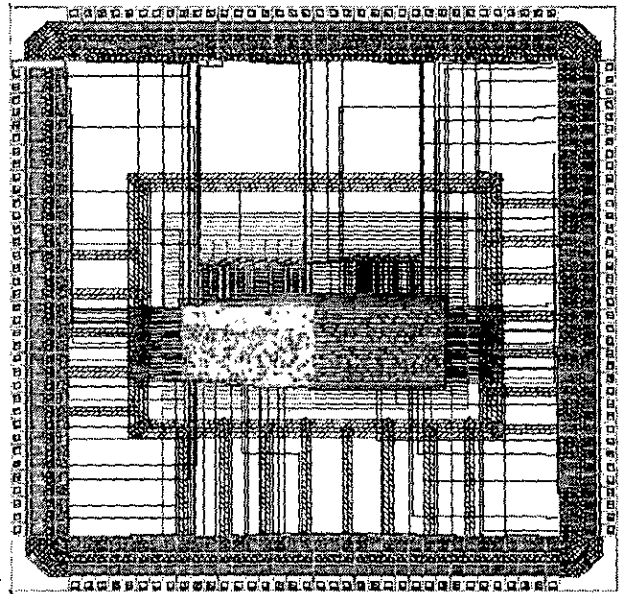


圖 5：完整晶片的佈局圖