

單晶片多處理機可程式實驗平台之設計與實現(III)

Design and Implementation of a Programmable Experimental Platform

for a Single Chip with Multiple CPUs (III)

計劃編號: NSC88-2213-E-009-039

執行日期: 87/8/1 - 88/7/30

主持人: 王國禎 國立交通大學資訊科學系副教授

一、中文摘要 (關鍵詞: 仲裁者、匯流排配置、內接網路、多處理機系統。)

在本計劃中,我們提出了一個建構在單晶片多處理機可程式實驗平台的疊流式單邊縱橫交換鍵。疊流式單邊縱橫交換鍵是由單邊縱橫交換鍵、仲裁器、處理機介面、以及記憶體介面四大部分所組成。處理機介面與記憶體介面控制了疊流式協定中仲裁、請求、窺探、回應和資料共五個階段的進行,它們是我們提出的架構中最主要的部分。仲裁器接收來自處理機之處理機介面的匯流排配置請求,以控制單邊縱橫交換鍵來達成處理機和記憶體模組之間的連接。我們使用 Verilog 硬體描述語言及 Xilinx 的 FPGA (現場可程式邏輯陣列) 分別描述及實現了一個 2x2 疊流式單邊縱橫交換鍵, Verilog 模擬的結果驗證了疊流式設計功能。我們並使用 Aptix MP3A FPCB (現場可程式化電路板) 與一些電子元件分別來模擬疊流式交換鍵以及處理機與記憶體模組的行為。此外,由於記憶體參考有兩種區域性: 時間區域性和空間區域性。如果處理機參考一個記憶體模組,它傾向於再次參考相同的記憶體模組。如果我們不馬上釋放連接處理機和記憶體模組的匯流排,當此處理機再次參考相同的記憶體模組時,我們可以直接使用這匯流排而不需重新設定。因此我們提出了一個新匯流排配置演算法。實驗數據顯示使用新匯流排配置演算法的效能比原先高了 1.5 到 3 倍。本研究的貢獻是設計出一個高產

量及具成本效益的疊流式單邊縱橫交換鍵以配合高效能的多處理機,從而消除效能瓶頸。

英文摘要 (Keywords: arbiter, bus allocation, interconnection network, multiprocessor system)

In this project, we propose a pipelined one-sided crossbar switch for a single-chip multiprocessor system. A processor interface for each processor and a memory interface are the main parts of our proposed structure, which control the five phases of the pipelined protocol: *arbitration*, *request*, *snoop*, *response*, and *data*. We have described and implemented a 2x2 pipelined one-sided crossbar switch using Verilog HDL and Xilinx FPGAs, respectively. Verilog simulation results have validated the functionality of the pipelined design. We also use an FPCB (Field Programmable Circuit Board) and some electronic components to emulate the pipelined switch and the behavior of the processors and the memory modules, respectively. In addition, memory references usually have two kinds of locality: *temporal locality* and *spatial locality*. If a processor references a memory module, it tends to reference the same memory module again. Thus, we propose a new bus allocation algorithm. Experimental results show that the performance (throughput) using the new bus allocation algorithm is 1.5 to 3 times higher than that using the original algorithm. The

contribution of this work is designing a high throughput and cost-effective pipelined one-sided crossbar switch to match high performance multiprocessors and to eliminate performance bottleneck.

二、計劃緣由與目的

內接網路是一個高效能多處理機系統中極重要的部分。單邊縱橫交換鍵改良了雙邊縱橫交換鍵之缺點，具有容錯及不阻塞的特性，極適合用來作多處理機之內接網路。在傳統式單邊縱橫交換鍵中，處理機和記憶體模組是放置在同一邊，而它們之間的資料和訊息傳遞則是透過交叉節點來連結。然而每次透過單邊縱橫交換鍵來傳遞資料時，均需要花費額外的時間來做重新設定。如果我們能減少重新設定的次數，那就能改善整個系統的效能。

匯流排競爭及記憶體衝突是影響內接網路表現之兩項主要因素。為了降低匯流排競爭可利用不阻塞之高頻寬內接網路。為了減少記憶體衝突則可增加記憶體模組個數或縮短記憶體存取時間。另外，疊流式控制也能降低上述因素所造成的效能影響。疊流式控制是當某筆交易等待資料被送回時，釋放匯流排的使用權供其它的交易使用。傳統的方式則需等待資料送回才釋放匯流排。利用疊流方式可使內接網路的產量提高。現今許多高效能的微處理機，像是Pentium Pro和Power PC，為了達成計算的高平行性，都提供了疊流式的匯流排協定來處理記憶體和輸出入裝置的存取。

在本篇報告中，我們致力於高效能疊流式內接網路之設計，以降低記憶體衝突所造成的影響。當某匯流排因故障而無法使用時，疊流式設計也能降低匯流排競爭的發生機會。我們以硬體描述語言來描述疊流式單邊縱橫交換鍵及以模擬來驗證內接網路的功能正確性。此外我們也使用了

FPGA與FPCB來模擬疊流式交換鍵及處理機與記憶體模組的行為，以實際驗證我們的設計。另外我們也根據了記憶體參考的區域性而提出了一個新匯流排配置演算法，以提升整個系統的效能。

三、研究方法及成果

圖(一)顯示一個多處理系統的架構，其中疊流式單邊縱橫交換鍵包含了處理機介面、記憶體介面、仲裁器，以及單邊縱橫交換鍵，茲分述如下：

- 單邊縱橫交換鍵：它是一個不阻塞的交換鍵，而且每條匯流排包括一個位址匯流排和一個資料匯流排。它們的寬度都是32位元。
- 單邊縱橫仲裁器：它決定哪一個請求將被配置匯流排，以建立連接通路。
- 處理機介面：它從處理機接收位址（讀取交易）或位址和資料（寫入交易）並送至相對應的記憶體介面，同時它也以疊流的方式來傳送位址與資料。
- 記憶體介面：它從處理機介面接收位址（讀取交易）或位址和資料（寫入交易）並送至相對應的記憶體模組。

我們提出的疊流式協定有下列五個階段：

- 仲裁：處理機以循環的方式將一個交易選出，然後相關位址（和資料）接著被送至疊流式介面。
- 請求：疊流式介面送出位址給單邊縱橫仲裁器來配置匯流排。如果沒有記憶體衝突的問題，則交易進入下一個階段；否則就須進行重送位址的動作。
- 窺探：在這個階段，交易已完成了請求，此時需要等待記憶體模組的就緒。
- 回應：目的裝置（這裏指記憶體模組）送出回應給提出請求的疊流式介面，這表示記憶體模組或處理機可以送出資料。
- 資料：如果沒有處理機的衝突，則資料可以傳送至匯流排上。

我們用連接表來記錄每一個處理機、每一個記憶體模組，以及每一條匯流排的狀態。對於每一個處理機或記憶體模組，其相對應的欄位如圖(二)所示，其中 C 表示該處理機或記憶體模組是否連接到匯流排， bus 則是表示連接到哪一條匯流排。對於每一條匯流排，其相對應的欄位如圖(三)所示，其中 $state$ 表示該匯流排的狀態，00 表示未連接；01 表示已指定給某個處理機和記憶體模組，但尚未連接；02 表示已連接上處理機和記憶體模組，而且資料正在匯流排上；03 表示已連接上處理機和記憶體模組，但是資料不在匯流排上。 $processor$ 表示連接到哪一個處理機，而 $memory\ module$ 表示連接到哪一個記憶體模組。我們的新匯流排配置演算法如下：首先選出一個處理機 p 成為匯流排的擁有者，而該處理機參考到記憶體模組 m ，有以下四種情形：

- p 連接到匯流排 i ， m 連接到匯流排 j 。
如果 i 等於 j ，我們就使用這匯流排。如果 i 不等於 j ，我們就使用匯流排 i ，並釋放匯流排 j 。
- p 連接到匯流排 i ，而 m 不連接到匯流排。我們使用匯流排 i 。
- p 不連接到匯流排 i ， m 連接到匯流排 j 。我們使用匯流排 j 。
- p 和 m 都不連接到匯流排。由於單邊縱橫交換鍵具有不阻塞的特性，我們可以找出一條目前未被使用的匯流排來用。

另外我們也使用可程式化電路板和 FPGA 建構出疊流式內接網路，並使用電子元件模擬處理機與記憶體之行為。圖(四)顯示如何將我們的設計對應到可程式化電路板上。我們必須準備兩個檔案：FPGA-grouped logic netlist 和 board level netlist，其中 FPGA-grouped logic netlist 描述了我們疊流式單邊縱橫交換鍵的設計，而 board level netlist 描述了 FPGA 和其他電子元件間的內部連接。將我們的設計對應到可程式化電路板上有六個步驟：

- 使用 Verilog 硬體描述語言來描述我們

的設計。

- 使用 Synopsys FPGA Compiler 來做合成。
- 使用 Xilinx XACT Step M1 軟體將設計實現出來，並產生 FPGA 設定資料以及腳位資訊。
- 使用 Explorer 軟體，根據 board level netlist 以及 FPGA 腳位資訊來產生可程式化電路板設定資料。
- 將可程式化電路板設定資料下載至 MP3A System Explorer。
- 將 FPGA 設定資料下載至 FPGA

為了實現疊流式單邊縱橫交換鍵，我們使用 Xilinx 4062XL FPGA 來模擬疊流式單邊縱橫交換鍵，並使用電子元件模擬處理機與記憶體之行為。圖(五)顯示輸入信號的模擬方式，而圖(六)則顯示輸出信號的模擬方式

四、討論與結論

我們比較了疊流式單邊縱橫交換鍵在採用新匯流排配置演算法之前和之後的產量，其中有五項參數： P 表示處理機的數目； M 表示記憶體模組的數目； B 表示匯流排的數目； P_r 表示在每一個 clock cycle 中，每一個處理機產生新交易(transaction)的機率； P_s 表示一個處理機產生新交易時，跟前一個交易參考到同一個記憶體模組的機率。圖(七)顯示產量與 P_s 的關係，其中 $M = B = P$ ， $P_r = 1.0$ 。實驗數據顯示在原先的設計中， P_s 越高則產量越低。原因是當兩個處理機同時參考到相同的記憶體模組時， P_s 越高則這兩個處理機再次參考到相同的記憶體模組的機率越大，而導致更多的記憶體衝突。而在新的設計中，雖然 P_s 越高造成更多的記憶體衝突，但是有更多的交易以較少的時間完成，因此產量上升。實驗數據顯示使用新的匯流排配置演算法可提升效能 1.5 到 3 倍。因此，我們的疊流式單邊縱橫交換鍵確實具有高效能的特

性，可以解決常發生在多處理機系統中內接網路效能瓶頸的問題。

五、參考文獻

[1] K. Hwang, "Advance Computer Architecture: Parallelism, Scalability, Programmability," *McGraw-Hill*, 1993.

[2] F. Pong, M. Browne, A. Nowatzky, and M. Dubois, "Design Verification of the S3.mp Cache-Coherent Shared-Memory System," *IEEE Trans. on Computers*, pp. 135-140, Jan. 1998.

[3] J. Ghosh, A. Varma, and Krishnamurthy, "Distributed Control Schemes for Fast Arbitration in Large Crossbar Networks," *IEEE Trans. on VLSI Systems*, pp. 54-67, Mar. 1994.

[4] N. Sarangdhar and G. Singh, "An Overview of the Pentium Pro Processor Bus," *Digest of Papers, COMPCON '96*, pp. 383-387, Feb. 1996.

[5] S. Brown and J. Rose, "FPGA and CPLD Architectures: a Tutorial," *IEEE Design and Test of Computers*, pp. 42-57, June 1996.

[6] T. H. Cormen, C. E. Leiserson, and R. L. Rivest, "Introduction to Algorithms," *McGraw-Hill*, 1990.

[7] K. Wang and A. Y. Liu, "HDL Design and FPGA Implementation of a Pipelined One-Sided Crossbar Switch for Multiprocessor Systems," in *Proceedings of the 9th VLSI/CAD Symposium*, pp. 419-422, 1998.

六、圖表

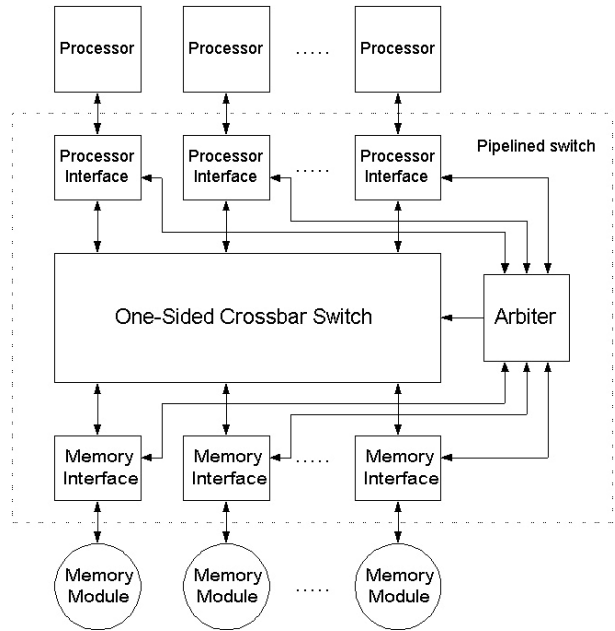


圖 (一) 多處理機系統架構圖。

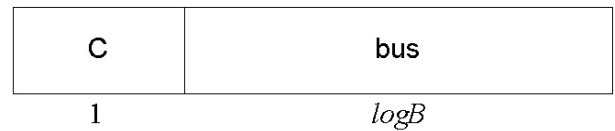


圖 (二) 連接表中對應於處理機或記憶體模組之欄位。

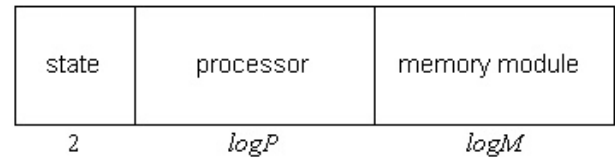


圖 (三) 連接表中對應於匯流排之欄位。

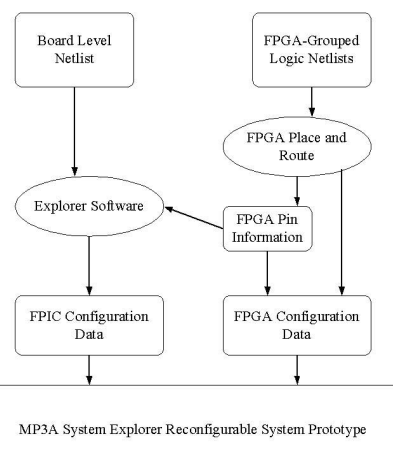
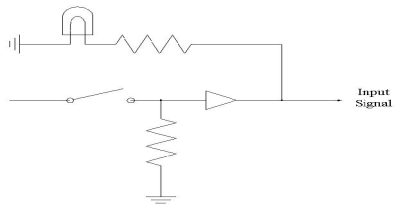
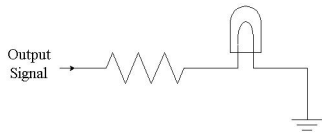


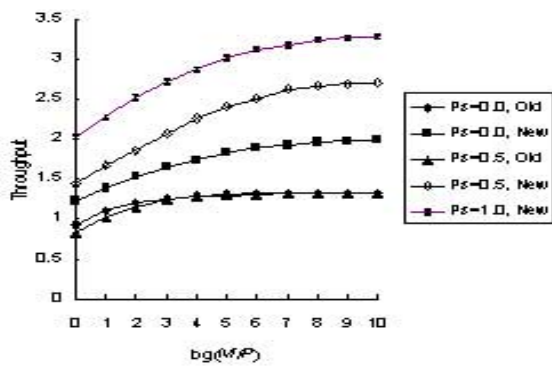
圖 (四) 將設計對應到可程式化電路板



圖（五）輸入信號的模擬方式



圖（六）輸出信號的模擬方式



圖（七）產量與 P_s 之關係。