

行政院國家科學委員會專題研究計畫 期中進度報告

用於多媒體訊號處理之單晶片系統平台 ~ 組件, 界面與系統層次設計(2/3)

計畫類別：個別型計畫

計畫編號：NSC91-2215-E-009-033-

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研究所

計畫主持人：任建葳

計畫參與人員：李坤儉、張彥中、邱睦喻、林子傑、秦浩雲、許惠錚

報告類型：精簡報告

處理方式：本計畫可公開查詢

中華民國 92 年 5 月 26 日

用於多媒體訊號處理之單晶片系統平台～組件、介面與系統層次 設計(2/3)

A SoC Platform for Multimedia Signal Processing ~ Components, Interfaces and System-Level Design

計畫編號：NSC-912215-E-009-033

執行期限：91年8月1日至92年7月31日

主持人：任建葳 國立交通大學電子工程學系

一、中文摘要

本計畫的目標在於發展一適合多媒體訊號處理的低電源消耗單晶片系統平台—Star platform 與系統層級高階驗證環境。第二年完成的項目包括了建立系統平台所需的多層式記憶體控制器、高效率匯流排傳輸機制與具有較少記憶體需求的多媒體訊號處理的核心模組：包括 MPEG-4 的二元移動估計矽智產及 JPEG-2000 矽智產。

關鍵詞：記憶體控制器、匯流排傳輸機制、二元移動估計、JPEG 2000、矽智產

Abstract

This project is to develop a low-power SoC platform for multimedia signal processing and a high-level verification environment. In this year, we have developed a highly efficient memory controller with layered architecture for off-chip SDRAM memory, a high-efficient data transfer scheme, and low-memory requirement multimedia processing cores: MPEG-4 binary shape encoder (BME) and JPEG-2000.

Keywords: SDRAM, memory controller, data transfer scheme, MPEG-4, binary motion estimation, JPEG 2000, IP

二、計畫緣由與目的

在電腦、通訊及消費性電子等 3C 領域中，單晶片系統的設計是未來的趨勢。本計畫主要是探討發展一適合多媒體訊號處理的低電源消耗單晶片系統平台與系統層級高階驗證環境。

由於佔一個 SoC 最大面積與功率消耗的部份，往往就是記憶體的使用與資料的存取，基於這樣的觀察，系統模組除了本身內部在執行效率、速度、矽面積與低功耗的最佳化外，也考慮到整個系統層次的記憶體與系統匯流排的使用，減少記憶體的大小、記憶體的存取次數與系統匯流排使用的時間，進一步降低了整個系統所需的成本與功率消耗；也由於記憶體的存取次數與系統匯流排使用的時間變少了，系統中其它重要的功能模組在記憶體存取與系統匯流排的使用也就更為流暢，進而提升系統整體性能。

本篇報告將針對今年度計畫工作項目分項敘述與討論。包含用於多媒體系統晶片中的(1)多層式記憶體控制器，(2)高效率匯流排傳輸機制，(3)低記憶體需求

的形狀編碼器 IP，(4)低記憶體需求的 JPEG2000 IP。

三、研究方法及成果

(1) 多層式記憶體控制器

動態隨機存取記憶體 (DRAM) 在記憶體階層設計中，具有密度高、成本低特性，也因此被廣泛地應用在各種系統。但由於其複雜的介面協定以及繁瑣的時序限制，常造成一般的 DRAM 控制器無法有效的利用 DRAM 所能提供的峰值頻寬。另一方面，由於功能模組對記憶體使用的需求不同，如必須保證某些功能模組頻寬的取得或讓存取延遲越短越好等，一般通用型記憶體控制器無法有效滿足每個功能模組不同的需要。針對這兩個情形，我們設計一個分層式(layered)控制的 DRAM 控制器，系統設計者可依照其系統需求，選擇最適用的 DRAM 控制器的層級功能。圖 1 為分層式 DRAM 控制器的架構圖，主要為三層式的結構：

- Layer 0 Memory Interface Socket (MIS)：MIS 主要負責記憶體的基本控制功能，其會根據目前記憶體狀態，發出最適合的控制指令，以減少每次的存取延遲。MIS 硬體架構是針對 DRAM 的平均效能作最佳化。
- Layer 1 Quality-Aware Scheduler (QAS)：QAS 的目的在提供**最小存取延遲與保證頻寬**兩種服務，會根據目前記憶體的控制狀態、各功能模組存取記憶體的要求與存取位址三方面的資訊，調整各功能模組存取記憶體的順序。
- Layer 2 Data mover：內建地址產生器(Built-in Address Generator)的目的在提供 QAS 更多的存取位址資訊，進而可以有更好的排程；另一方面，也提供資料在記憶體擺放的方式，讓不熟悉 DRAM 系統的使用者，可以有個快速入門的參考。此外，內建記憶體產生器對某些介面協定，如 Advanced VCI，可以減少系統匯流排要傳送的資訊，由於不需再傳每筆資料的地址，可增加系統匯流排的效能。

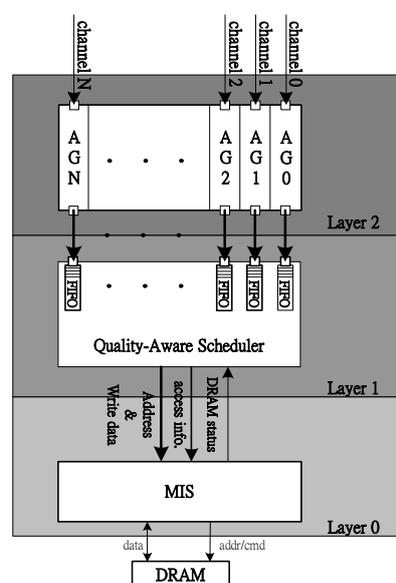


圖 1 分層式動態記憶體控制器

系統設計者可依照其系統需求，選擇最適用的 DRAM 控制器的層級功能，包括只使用 Layer 0、使用 Layer 0 + Layer 1、使用 Layer 0 + Layer 1 + Layer 2、或者使用 Layer 0 + Layer 2。

在效能評估方面，我們模擬數位電視盒(set-top box)單晶片的 DRAM 存取行為，在系統時脈為 100MHz 下時，多層式 DRAM 控制器，相較於傳統式多通道 DRAM 控制器，可增加頻寬利用率達 14%，同時對於中央處理器等需要低存取延遲的運算單元，平均存取延遲至多可降低達 50%。

(2) 高效率匯流排傳輸機制

我們觀察到晶內匯流排(on-chip bus)在深次微米時，不論是功率消耗或是時間延遲，都會佔系統蠻多的設計資源；而另一方面，影像處理的應用需要大量的資料處理，也需要大量的資料在模組間傳遞資料（特別是記憶體與運算單元間的傳遞），若能減少資料傳遞的頻寬，將可減少資料傳輸的代價，例如降低匯流排時脈或寬度、或用較簡單的匯流排協定等就可達到原來資料傳輸量的需求。有鑑於此，我們結合這定址模式、編碼技術與功能分割(functionality partition)三個觀念，提出一套晶內匯流排的傳輸機制，來減少資料傳輸時所需的頻寬，進而減少資料在晶內匯流排傳輸時所需要消耗的功率。我們的編碼方式考量到下面幾點：

1. 編碼方式必須是簡單的。如果編碼方式太過複雜，則花在編碼、解碼上的時間將有可能造成系統效能降低、硬體代價太高，而不符合原來的目標。
2. 所使用的資料傳輸方式必須不會用到特殊的晶內匯流排協定或額外的控制訊號。考量到 IP 可能會被整合到既有的晶內匯流排標準，例如 ARM AMBA 或 IBM CoreConnect，若用特定的協定或是控制訊號，將有可能造成將來 IP 整合時的一些問題。
3. 編碼後的資料傳遞時間一定不會比不編碼的資料時間長。
4. 編碼後的資訊，可以協助後級加快處理。

如前所述，由於晶內匯流排的架構可能造成的時間延遲，因此一般晶內匯流排的地址解碼器(address decoder)都只負責解碼地址匯流排(address bus)的一部份，剩下的再交由 IP 自行解碼。例如目前最熱門的晶內匯流排 ARM AMBA，其系統匯流排的地址匯流排共有 32 位元，就規範其地址解碼器最多只負責地址匯流排的最高 22 位元，也就是說每個 IP 內部至少都有 10 位元的定址空間。但事實上除了記憶體控制器外，很少有 IP 會需要用到這麼多的地址空間。有鑑於此，我們的編碼資料傳送方式會利用到這些地址匯流排的訊號線。

另一方面，由於影像處理的資料具有很多的重複性，因此我們選擇 fixed-length run-length 的編碼方式，來對資料作壓縮。以圖 2 為例，由於最上面兩個 row 的筆資料都是 0x3FFFF，因此我們可將這兩筆資料表示成(0x3FFFF, 0x02)，(m, n)裡的 m 代表資料本身的值，n 則代表資料出現的次數。我們將 m 透過資料匯流排 (data bus)，而 n 則透過地址匯流排來傳送，這樣原本需要傳送兩次的資料，現在只要一次就可傳完了。以形狀編碼是以 16×16 block 為單位，若 n 最多設為 16，則只會占到 10 位元的定址空間的 16 個位址，會用到 address bus 的 5 位元，而我們設計的形狀編碼 IP — Star SE — 其它內部控制與狀態暫存器 (Control and Status Register, CSR) 對記憶體的定址空間需求，只有 80 bytes。若這

16 個位址的編碼空間設為 0x01~0x10 (實體位置還要再加上 Star SE 本身的 offset address)，則匯流排的 run-length 解碼方式可輕易由計數器、address decoder、FIFO 解出；編碼方式也是類似的硬體需求。

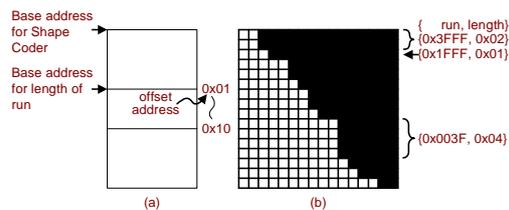


圖 2. 減少匯流排傳輸資料量的資料傳輸機制，(a)整個模組內的記憶體空間及不同 length 寫入的位址，(b)一個 boundary BAB 的 run-length coding 的例子。

根據這個資料傳輸機制，我們針對九組 MPEG-4 的測試影像序列作統計，我們所提出來的資料傳輸機制平均可以降低資料傳輸時間到原來的 9.39%。採用我們所提出的資料傳輸機制可以大量減少系統中匯流排傳輸資料所需要花的時間，減少匯流排的壅塞壓力，提升系統整體的效能。

這個編碼還有一個好處(也就是前述第四項的考量點)：當 length 為 0x10 而 run 為 0xFFFF 或 0x0000 時，則表示目前的 BAB 為 non-boundary BAB，可直接跳到 VLC 作最終編碼，加快整體運算。這種資料方式也可應用於其它多媒體系統如 JPEG2000。

(3) 低記憶體需求的二元移動估計 IP

移動估計模組向來是 MPEG 系列標準裡最佔運算量與記憶體存取的模組，而用在形狀編碼裡的二元移動估計(binary motion estimation)也不例外。我們針對形狀編碼的特性，對二元移動估計的運算排程與資料存取方式作最佳化，可有效降低所需記憶體存取次數與運算次數。

形狀編碼將物件平面切割成 16x16 的區塊來處理，每個區塊可分為三個種類：Opaque、Transparent 與 Boundary。因此我們將區塊的種類存在索引表(index table)裡，在讀取時先判斷該區塊是屬於哪一個種類：若是邊界區塊，則再去讀取這個 16x16 區塊的真實資料(存在 frame buffer)；反之，則每次使用此區塊內的資料，只需將對應的區塊種類的值(Opaque 或 Transparent)附加上去，而不用再去讀取 frame buffer。這樣的作法，可減少存取 frame buffer 次數達 46.7%。

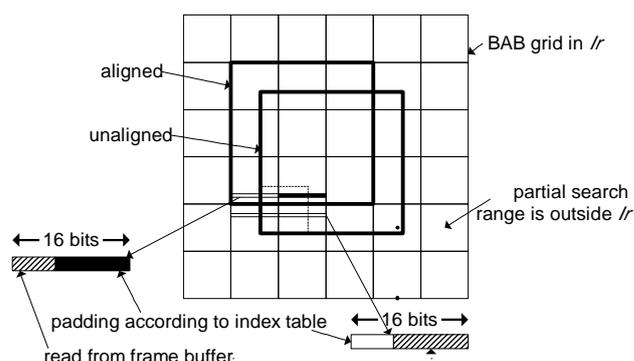


圖 3. 在不同搜尋位址時，可能會跨越不同數目的區塊。

在運算排程方面，基於形狀資訊總是群集性的出現，因此在找尋與目前要編

碼的資料最相符區塊所計算的絕對值的差總和 (sum of absolute differences) 時，是先計算候選區塊的最底部的一列資料。若此列資料與上一個候選區塊的後 15 列資料是一樣的，則表示此候選區塊內的資料一定是與前一個候選區塊是一樣的 (為 transparent 或 Opaque BAB)，因此剩下的 15 列資料就不必再讀取與計算。這樣的排程方式，用在單一運算元件 (processing element) 上，如一般微處理機，可節省 29.02% 的運算與記憶體存取；若用在 16 個運算元件架構，則可省 11.16%。

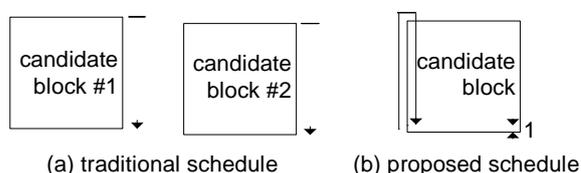


圖 4.傳統 top-down 與所提出的 wrapping 運算排程。

(4) 低記憶體需求的 JPEG2000 IP

在 JPEG 2000 的實現上，由於需要相當大量的緩衝資料，因此光是記憶體就佔據了晶片的絕大部分面積。為了要降低晶片的成本，最有效率的方法就是降低 JPEG 2000 緩衝記憶體的大小。

我們觀察離散小波轉換以及 EBCOT 熵編碼器的暫存資料，它們除了在自己本地端的運算部分需要使用緩衝區以外，在兩者資料流的中間也需要有個緩衝區，這是因為離散小波轉換的輸出資料型態是字元，然而 EBCOT 熵編碼器所需要的資料型態卻是編碼區塊的位元平面，因此離散小波轉換的輸出必須要先放在緩衝區，然後再轉換成 EBCOT 熵編碼器所需要的位元平面，如此才能正確地完成資料型態轉換，除此之外，我們觀察一般離散小波轉換的輸出順序，發現其輸出順序其實並不符合 EBCOT 熵編碼器所需的輸入順序，因此需要更大的緩衝區來儲存暫時用不到的離散小波轉換輸出資料，為了降低這塊緩衝區的大小，我們改變了離散小波轉換的輸入順序，從原本一行一行餵資料改成如圖 5 的順序，也就是在特定邊界就換到下一行餵資料的 Z 字型掃描順序，使其輸出順序較能符合 EBCOT 熵編碼器所需的編碼區塊輸入順序，經由這樣的改變，就可以有效地縮小兩者之間資料型態轉換所需的緩衝記憶體。

我們改變了離散小波轉換的輸入順序後，相對地原本內部運算所需的暫存資料也需要做更動，根據分析過後，我們發現改變輸入順序會導致離散小波轉換本地端所需的暫存資料量提高，然而離散小波轉換和 EBCOT 熵編碼器之間的資料型態轉換緩衝記憶體卻可以大幅的減少，以一張 256×256 的圖形，選用 32×32 大小的編碼區塊為例，總體 JPEG2000 大約可以省下 67% 的緩衝記憶體。

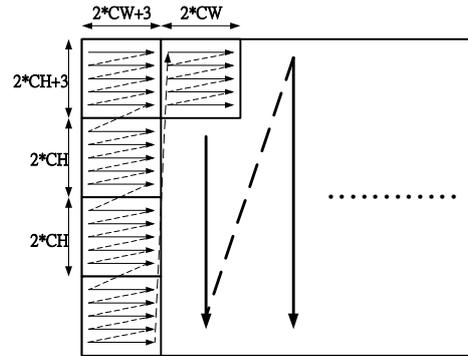


圖 5 最佳化 Z 字型掃描順序

四、結論與討論

本計畫已順利完成各項預期工作項目。其中部分研究成果已被國外期刊接受的有一篇、國外會議接受的論文有兩篇、已投稿國外會議的有 3 篇；其他部分仍陸續整理投稿於國際會議和期刊中。

五、參考文獻

- [1] K.B. Lee, J.Y. Lin and C.W. Jen, "A Multi-Symbol Context-based Arithmetic Coding Architecture for MPEG-4 Shape Coding," accepted by IEEE Trans. on CSVT.
- [2] K.B. Lee et al, "Optimal Frame Memory and Data Transfer Scheme for MPEG-4 Shape Coding," to be appeared in ECCTD'03.
- [3] K.B. Lee et al, "A Memory-Efficient Binary Motion Estimation Architecture for MPEG-4 Shape Coding," to be appeared in ICCE'03.