

# 行政院國家科學委員會專題研究計畫 期中進度報告

可重組式計算於複雜算術系統的應用--以圖形顯像為例

(2/3)

計畫類別：個別型計畫

計畫編號：NSC92-2213-E-009-025-

執行期間：92年08月01日至93年07月31日

執行單位：國立交通大學資訊工程學系

計畫主持人：鍾崇斌

共同主持人：單智君

計畫參與人員：黃士嘉、吳智斌，陳俊一，呂理聖，翁綜禧，劉彥志，賴敬中

報告類型：精簡報告

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 93 年 5 月 31 日

行政院國家科學委員會補助專題研究計畫  成果報告  
 期中進度報告

可重組式計算於複雜算術系統的應用 -- 以圖形顯像為例(2/3)  
**Reconfigurable Computing for Complex Arithmetic Systems**  
**-- Using Rendering as an Example (2/3)**

計畫類別： 個別型計畫  整合型計畫

計畫編號：NSC 91-2213-E-009-062

執行期間：92年 08月 01日至 93年 07月 31日

計畫主持人：鍾崇斌 國立交通大學資訊工程系

共同主持人：單智君 國立交通大學資訊工程系

計畫參與人員：黃士嘉、吳智斌、陳俊一

呂理聖、翁綜禧、劉彥志、賴敬中

成果報告類型(依經費核定清單規定繳交)： 精簡報告  完整報告

本成果報告包括以下應繳交之附件：

赴國外出差或研習心得報告一份

赴大陸地區出差或研習心得報告一份

出席國際學術會議心得報告及發表之論文各一份

國際合作研究計畫國外研究報告書一份

處理方式：除產學合作研究計畫、提升產業技術及人才培育研究計畫、  
列管計畫及下列情形者外，得立即公開查詢

涉及專利或其他智慧財產權， 一年 二年後可公開查詢

執行單位：國立交通大學資訊工程研究所

中 華 民 國 93 年 05 月 31 日

# 行政院國家科學委員會專題研究計畫成果報告

## 可重組式計算於複雜算術系統的應用--以圖形顯像為例(2/3) Reconfigurable Computing for Complex Arithmetic Systems -- Using Rendering as an Example (2/3)

計畫編號：NSC 91-2213-E-009-062

執行期限：92 年 08 月 01 日至 93 年 07 月 31 日

計畫主持人：鍾崇斌 國立交通大學資訊工程系

共同主持人：單智君 國立交通大學資訊工程系

計畫參與人員：黃士嘉、吳智斌、陳俊一

呂理聖、翁綜禧、劉彥志、賴敬中

### 一、中文摘要

由於半導體技術的急速演進，現今的晶片已經可以達到數百萬甚至數億個電晶體的容量。這樣的技術將晶片設計導入新的系統整合的設計規劃，單一晶片即可整合系統的所有功能。同時，消費性產品擁有產品週期較短，量小且高毛利率的特性，所以快速的產品設計時程、產品個別化及易於修復錯誤已經成為目前系統單晶片的重要課題。本計畫針對 3D 繪圖顯像之應用，提出可重組式計算進行相關研究。在此架構之下，產品工程師可以快速的研發下一代的產品且當有錯誤發生在現行產品亦可立即更正以降低整體成本。

根據上述的想法，我們選擇 3D 圖形顯像上的應用，因其具有較長的運算時間、大量且平行性資料及管線式運算結構的性質，特別適合於專屬硬體實作以發揮高平行性計算的優點，同時需要設計相關的編譯技術做緊密的結合。在三年期計畫中，工作項目區分如下：第一年，標竿程式分析與編譯技術之設計；第二年，算術單元與可重組式管線之設計；第三年，算數架構與可重組式管線之實作與效能分析。

我們針對圖形顯像之標竿程式進行資料流程圖、迴圈特性及算術特性上的研究分析，並設計相關的編譯技術。在第一年度中

具有以下四個研究成果：(1)軟硬體程式切割技術；(2)資料流程圖切割技術；(3)程式分割排程技術；(4)高繞線效率的硬體配置架構。

在第二年的計畫執行中，我們進一步設計可重組式架構及其算術運算單元。利用此架構，我們將 3D 繪圖顯像的應用在此架構上進行模擬並且發現這種設計可以有效提升運算的效率與邏輯電路使用率。除此之外，我們採用具有類似運算結構的 MPEG-4 編碼流程，配合有效的排程演算法執行於本架構上發現該應用亦可有效提升編碼執行效率。在這年度的計畫執行中，我們展現出可重組式架構在應用上的彈性。

### 二、英文摘要

Rapidly evolution of silicon technology allows more than hundred millions transistors being integrated into single chip. This advanced technology leads the design of chip into new level of system integration - the whole system can be integrated into a single chip. Commercial products have the features of shorter product lifecycle, smaller amount of total products, and higher profit. Hence, the

rapid prototyping, product differentiation and easy bug fixing have become very important issue for system design of SoC. In this project, we focus on the applications of 3D rendering and propose architecture of reconfigurable computation. By using this architecture, the product engineer can develop next generation of products rapidly and fix bugs while they were occurred in current product faster. At the mean time, the cost of production can be incredible reduced by the flexible the reconfigurable architecture.

We choose 3D rendering as an example. It has the features of longer computation time, large amount of data with highly parallelism, and pipelining computation structures. If we develop a dedicated hardware for it, it can achieve the benefit of highly parallel computations. We also need the technology of compiler that can cooperate with the reconfigurable hardware tightly. In the three-year project, we spilt the jobs as following. In the first year, we analyze the benchmark and design a technology compiler. In the second year, we design the reconfigurable architecture and processing element. And in the last year of the project, we implement the reconfigurable architecture and analyze its performance.

We analyze the data flow graphs, loop features and computation features of the benchmark of 3D rendering application. According to these features we design associated compile

technology for them. In the first year, we have four results as follows. (1) Hardware/software program partition (2) Data flow graph partition (3) program scheduling and (4) highly routing utilization for placement and route.

Furthermore, we design the architecture of reconfigurable computation and processing element in the second year of the project. By using this architecture, we simulate the application of 3D rendering and find out that the performance of execution is improved and the utilization of hardware circuit is also better. In addition, we use the application of MPEG-4 encoding on our reconfigurable architecture. The MPEG-4 has almost the same features as 3D rendering on computation time and large amount of data. To cooperate with high efficient scheduling technology, the architecture can also achieve high performance of execution. In this year, we demonstrate that the reconfigurable architecture have higher flexible on multi applications products.

**關鍵詞：**可重組式計算、系統單晶片、繪圖顯像, MPEG4

**Keywords :** Reconfigurable Computing, SoC, Rendering, MPEG4

### 三、研究目的

因應現今嵌入式系統與消費性電子的迅速發展，產品週期愈來愈短，而效能需求卻愈來愈高，可重組式計算(Reconfigurable Computing)透過重設硬體功能與資料路徑，可重組式架構具備下列功用：(1)縮短產品研發時間；(2)使硬體架構

直接切合應用程式的需要；(3)以有限的晶片面積進行大規模運算。在研發時程與效能需求的雙重壓力下，可重組式架構提供快速且高效率的設計方法目前已成功的應用於實現即時影音播放[1]、Combinatorial Search[2]、與 Stream Processing[3]等多項應用中。

可重組式計算的研究，包含硬體架構設計與軟體編譯技術二方面。硬體架構設計方面，目前最常見的系統是以 FPGA (Field Programmable Gate Array) 作為硬體架構 [1][2][4][5][6]，另一種則為運算單元陣列(Array of Processing Element)。在軟體設計方面，利用合成(Synthesis)技術[7]將軟體或硬體描述語言的應用程式編譯為應用專屬電路，再以硬體佈局(Placement and Route) [4]方法將電路對應至可重組式架構的處理單元與資料路徑上。

儘管已有諸多技術被提出，當可重組式計算方式要被應用到更複雜的對象上時，卻遇到下列困難：(1)於硬體設計時應用對象尚不明確，以至於繞線佔用過多晶片面積 [6]，(2)缺乏成熟的編譯技術來規劃有限的硬體資源在不同時間點如何利用，造成許多硬體資源閒置，計畫中將探討上述困難的因應之道，並為可重組式計算方式開發更複雜的應用對象—以 3D 繪圖程式為代表的複雜算術系統。

本計畫將針對以繪圖顯像(Rendering)程式為代表的複雜算術系統，探討可重組式計算系統的軟硬體之設計方法。研究的最佳化為提高每單位晶片面積能增加的執行效能。欲達成此目標，在硬體與軟體方面分別條列如下：

(1)在硬體方面：設計可重組式管線系統(Reconfigurable Pipelined System)架構，以硬體實現多種不同形式的繪圖顯像管線。[8][9]

(2)軟體方面：提出編譯技術設計方法，以

充分利用有限的硬體資源，並發揮硬體管線高資料處理率的潛能。

在第二年度的計畫執行中，我們的工作主要為可重組式架構之設計、可重組式運算單元之設計與應用於 3D 繪圖顯像的排程方式。其中，可重組式架構之設計與可重組式運算單元之設計的主要目的為建構一個可重組式系統單晶片，並解決可重組式架構與微處理器、記憶體間介面的設計。另外，則必須處理其大量資料存取需求所造成的資料瓶頸。而 3D 繪圖顯像的應用則是利用排程技術將應用程式執行於該系統單晶片並解決其資源配置的問題，以期提高硬體使用率。

另外，我們亦將此系統用於 MPEG-4 影像壓縮與編碼的應用上。此應用具有大量資料運算的特性，因此可以利用可重組式架構的高運算能力解決其效能上的瓶頸。同時，更可以展現出此架構在應用上的彈性。

#### 四、結果與討論

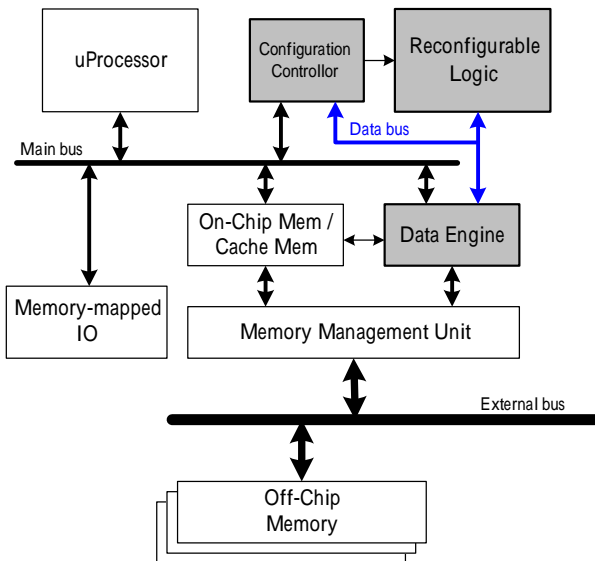
本計畫第二年度之重要成果如下：

##### 1. 可重組式架構之設計

如圖一所示，可重組式系統單晶片由以下單元所組成：(1)32 位元微處理器，如 MISP, ARM7 等一般用途微處理器(2)快取記憶體/晶片內記憶體(Cache Memory/On-chip Memory)及記憶體控制單元(Memory Management Unit) (3)記憶體映像輸出/輸入介面 (4)可重組式架構則包括：硬體配置控制器(Configuration Controller)、可重組式邏輯(Reconfigurable Logic)與資料引擎(Data Engine)。

應用程式由編譯器在編譯階段決定軟體與硬體執行程序的分割，決定那些運算需由可重組式架構執行。應用程式執行時，由微處理器將程式由晶片外記憶體載入並開始執行。當應用程式執行到硬體執行的程序之前，微處理器會向硬體配置控制器要求配

置可重組式邏輯，並在配置完成後視需要提出執行的要求。



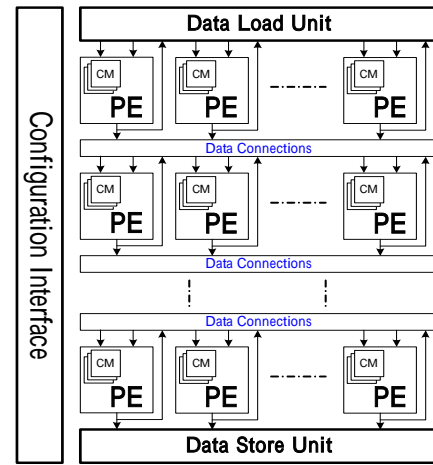
圖一、可重組式系統單晶片

硬體配置控制器 (Configuration Controllor) 負責接收 CPU 所發出的新的硬體配置需求，根據此需求向資料引擎要求所需的資料並將這些資料配置到可重組式邏輯中準備執行。直到微處理器發出執行要求後，控制可重組式邏輯開始執行。

可重組式邏輯 (Reconfigurable Logic) 由運算單元陣列為主要組成，加上相對應的資料儲存/讀取單元輔助其資料的需求。由於此架構以 32 位元 CPU 為核心因此其運算單元的設計重點為執行一個 32 位元運算、兩個 16 位元運算以及四個 8 位元運算等三種模式以符合原始應用程式的編譯流程。其目標為取代一連串的 32 位元指令的執行。

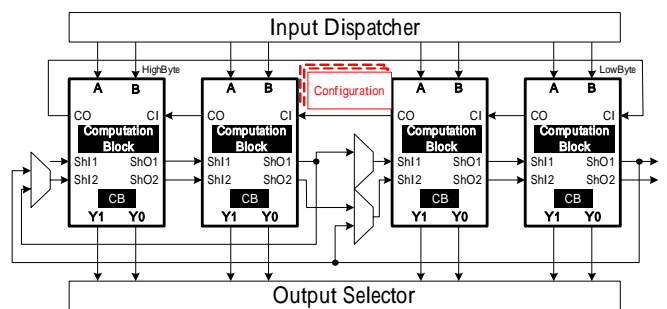
資料引擎 (Data Engine) 則是負責存取硬體配置資料以及計算資料，提供硬體配置控制器與可重組式邏輯的資料需求。同時，維持 CPU 的快取記憶體與可重組式邏輯間資料的一致性與相依性。

## 2. 可重組式運算單元之設計



圖二、可重組式邏輯架構圖

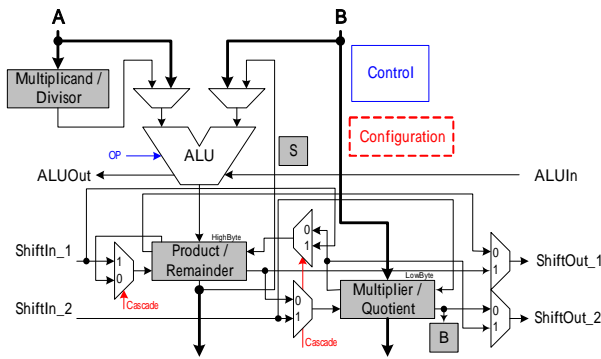
可重組式邏輯由 (1) 可重組式運算單元 (Processing Element, PE) 陣列 (2) 硬體配置介面，及 (3) 資料儲存/載入單元所組成。陣列中的基本運算單元經由資料連結網路形成運算單元陣列，編譯程式可以根據實際的陣列配置產生相對應的資料流 (Data Flow)。在執行階段時，硬體配置介面 (Configuration Interface) 根據硬體配置控制器的指定將配置資料寫入相對應的運算單元以及連結網路。透過改變運算單元的功能執行與連結網路的連接方式達成不同的運算。



圖三、可重組式運算單元

在可重組式運算單元設計方面，為了配合 32 位元微處理的架構，一個運算單元是由四個可串接的 8 位元計算方塊 (Computation Block, CB) 所組成。各個方塊具有獨立運算的能力並且可經由兩個為一組或四個為一

組執行 16 位元或 32 位元的運算，如圖三所示。同時亦可將兩個或以上的可重組式運算單元串接為 64 位元或以上的運算，在這種高位元的運算情況之下則必須適度的降低執行頻率以配合高位元運算所產生的訊號延遲。



圖四、八位元計算方塊架構圖

如圖四所示，八位元計算方塊由一個算數邏輯單元(ALU)、數個八位元暫存器及控制邏輯所組成。當硬體配置的需求被寫入到計算方塊中，ALU 及控制單元透過解碼瞭解應用程式所要求的功能，並執行指定的運算。此單元可以執行加、減、乘、除及位移等運算，根據運算的複雜度執行時間為一個或多個時脈週期。

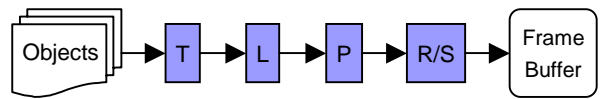
### 3. 利用運算排程提高 3D 繪圖顯像執行效率

3D 圖像是由大量的物件存在於不同的向量與位置所組成，因此 3D 繪圖顯像程式需要大量的數學運算且資料之間的平行性極高。由於每筆資料處理的時間不盡相同，使用微處理器來執行的程式效能無法有效提升，因此通常必須設計一個特定的硬體來加速運算效能。

為了增加硬體應用的彈性與硬體資源的有效利用，我們使用可重組式運算架構進行大規模的運算，使硬體結構直接切合實際的需求，達到高效能的目的。在探討 3D 顯像程式如何在可重組式系統上執行，透過：

1. 程式分割：提高最大平行性的前提下，將程式分割成大量的程式片段。
2. 工作排程：利用平行特性來重用硬體和並且減少重組的次數進而降低總執行時間。

傳統的 3D 繪圖顯像程式包括以下的流程：Transformation, Lighting, Projection, Rasterization and Shading。每一個 3D 物件必須透過這五種運算產生影像配置到視框緩衝器(Frame Buffer)，且每個物件間的運算為互相獨立。



圖五、3D 繪圖顯像管線

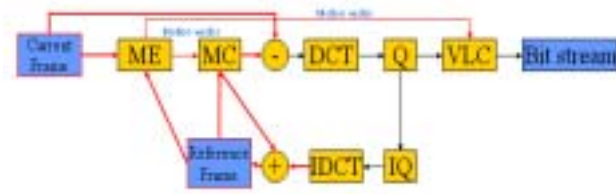
工作排程的方法可分為 FCFS (First-Come-First-Serve) 與優先權 (Priority) 等方式。FCFS 即是以工作指派的時間為優先順序的依據，這種方法也是最直接簡單的方法。優先權排程方式，則是以五種運算工作間的佇列(Queue)目前的需要等待運算的個數為優先權的依據。

根據我們的實驗結果，優先權排程方式可以得到較好的系統效能，而且當硬體配置時間較長時，此種排程方式可以隱藏配置時間對效能的影響。在相同可重組式硬體的前提下，這種排程方式可以減少 30%至 80%的執行時間。而在相同的執行時間的條件下，硬體的需求則只需原來的 75%。

### 4. 應用在 MPEG-4 的排程方式

可重組式架構提供一個彈性的硬體運算，因此本架構並不侷限在 3D 的應用。更進一步地，我們可以將可重組式架構延伸到各種需要大量運算的應用之上。例如：Digital Signal Process, 語音應用及影像應用等等。在本計畫，我們嘗試利用架構執行具有類似運算結構的應用上，以驗證可重

組式架構在多元應用的可行性。



圖六、MPEG-4 執行流程

如圖六所示，MPEG-4 的壓縮過程必須經過 Motion Estimation (ME)[10][11][12][13][14][15][16][17]、Motion Compensation (MC)[18]、Discrete Co-Sin Transform (DCT)[19][20][21][22]、Quantization (Q)、Variable Length Coding (VLC)、Inverse Quantization (IQ)、Inverse DCT (IDCT)等步驟。其中有大量的矩陣運算與 3D 顯像運算有相當的同質性。因此，我們將這個應用也納入我們的研究範圍。

我們的研究將工作區分為以上七種步驟並以這些為基本運算單元。我們發現其中大部分的運算時間為 ME，如果 ME 的演算法為 Full Search 的話，則 ME 的運算時間佔總時間的百分之九十以上。因此，我們根據此特性將排程的方法設計為以下兩種方式：(1)Partial Reconfigurable scheduling: 不論何種工作均以分時多工的方式排程。(2)ME Dedicated scheduling: 由於 ME 的執行時間為整體時間的關鍵，因此我們規劃一部分的可重組式邏輯專供 ME 使用，其餘的工作以分時多工的方式共同利用可重組式邏輯。

根據我們的實驗結果，當程式在 ME 步驟採用演算法 Three-step search (TSS) 時，則 Partial Reconfigurable scheduling；而 Full Search 演算法時，ME Dedicated scheduling 可以有較大的好處。在資料會流排寬度為 64 位元的情況下，TSS 演算法需要 69 個 8bit-CB，Full search 演

算法需要 37 個 8bit-CB。在資料匯流排寬度為 64 位元的情況下，TSS 演算法需要 69 個 PE，Full search 演算法需要 37 個 PE。

## 五、計畫成果自評

在第一年的研究中，我們針對編譯的技術上的研究，針對大量資料運算的應用研究相關的技術並發表三份碩士論文：(1)在可動態重組系統中利用預先抓取及合併的技術降低重組負擔[23]、(2)多 FPGA 之可重組式系統中資料流處理的資料流程圖切割方法設計[24]、(3)在應用執行過程中可重新設定組態系統中重新設定時間的減少與隱藏[25]。這些碩士畢業論文針對標竿程式進行分析並設計適用編譯技術對後續的計畫研究有長足貢獻。

在第二年的計畫時程中，我們更進一步地設計可重組式架構及運算單元，使其符合現今消費性產品的需求。這樣的硬體架構可以有效的減短產品的研發時程及降低因為重新設計新晶片所需的風險，可大量提高產品的可行性及獲利率。3D 繪圖顯像應用是未來消費性產品的主流，它讓電腦的應用從平面的限制擴展到三度空間的擬真環境。我們藉由可重組式架構來實做 3D 加速引擎不僅可以加速應用程式的執行效率，更進一步地當此應用未來有所更新規格時，我們亦可同時更新應用程式與編譯程式達到相同的硬體即可適用於未來相同領域產品的相容性。對於快速的發展原形機有非常大的助益。

綜觀前兩年的計畫執行，我們根據初始的規劃發展編譯技術及基礎硬體架構。在編譯技術部分已經發表三篇碩士畢業論文，而在今年也預計有二到四篇論文的發表。在計畫達成程度上，已經充分達成預期的要求。

除了 3D 繪圖顯像的應用之外，我們對於影像應用的 MPEG-4 編碼流程進行應用的分析與架構設計。我們發現這些需要大量資



料運算的應用可以有效的利用此可重組式架構來達成，由這一點可以展現出可重組式架構除了在研發時程的優越性更擁有多樣化的應用，對於系統單晶片的設計上，是一個非常適用的架構。

在第三年的計畫中，我們將會根據今年對於可重組式架構系統單晶片及可重組式運算單元之設計，進行硬體實做的工作。預計將以硬體描述語言(HDL)如：Verilog，VHDL，描述各部分硬體的實際動作。預計將整個系統製作完成，並在硬體電路模擬環境下，驗證系統的正確性。此外，在 3D 繪圖顯像的應用上，將其應用程式編譯為執行程序以相同的模擬環境驗證其演算法及功能的正確性。

## 六、參考文獻

- [1] S. D. Haynes, et. al., "Video image processing with the Sonic architecture," *IEEE Computer*, Vol. 33, No. 4, pp. 50-57, 2001.
- [2] M. Platzner, "Reconfigurable accelerators for combinatorial problems," *IEEE Computer*, Vol. 33, No. 4, pp. 58-60, 2001.
- [3] E. Caspi, et. al., "Stream computation organized for reconfigurable execution (SCORE): Introduction and Tutorial," *Proceeding of 10th Int' l Conf. Field-Programmable Logic and Applications*, 2000
- [4] V. Bets and J. Rose, *Architecture and CAD for Deep Submicron FPGAs*, 1999.
- [5] J. Hauser, and J. Wawrzynek, "Garp: A MIPS Processor with a Reconfigurable Coprocessor," *Proc. IEEE Symp. Field-Programmable Custom Computing Machines*, 1997.
- [6] Andr' e DeHon: Reconfigurable Architectures for General-Purpose Computing, *A. I. Technical Report No. 1586, Massachusetts Institute of Technology*. 1996
- [7] Michael I. Gordon, William Thies, Michal Karczmarek, Jasper Lin, Ali S. Meli, Andrew A. Lamb, Chris Leger, Jeremy Wong, Henry Hoffmann, David Maze, and Saman Amarasinghe, "A Stream Compiler for Communication-Exposed Architectures", *ASPLOS*, 2002
- [8] Katherine Compton, Scott Hauck, "Reconfigurable Computing: A Survey of Systems and Software", *ACM Computing Surveys*, Vol. 34, No. 2, June 2002, pp. 171 - 210.
- [9] Vinoo Srinivasan, Sriram Govindarajan, and Ranga Vemuri "Fine-grained and coarse-grained behavioral partitioning with effective utilization of memory and design space exploration for multi-FPGA architectures", *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, February 2001
- [10] Sau-Gee Chen, "An Area/Time-efficient motion estimation micro core", *IEEE*, 1993.
- [11] Maleeha Abbas, Batool Talha, Shoab Khan, and Adeel Abbas "A motion estimation chip for block based MPEG-4 Video Applications", *IEEE*, 2000.
- [12] Xiao-Dong Zhang, Chi-Ying Tsui "An efficient and reconfigurable VLSI architecture for different block matching motion estimation algorithms", *IEEE*, 1997.
- [13] D Xu, JM Noras and W Booth "A Simple and efficient VLSI Architecture for a very fast high performance three step search algorithm", *IEE*, 1998.
- [14] Santanu Dutta, Wayna Wolf "A

- Flexible Parallel Architecture Adapted to Block-Matching Motion-Estimation Algorithms” , IEEE, 1996.
- [15] Yeu-Shen Jehng, Liang-Gee Chen, and Tzi-Chiueh, “An Efficient and Simple VLSI Tree Architecture for Motion Estimation Algorithms” , IEEE, 1993
- [16] Li-Hsun Chen, Wei-Lung Liu, Oscar T.-C. Chen, Ruey-Liang Ma “A reconfigurable Digital Signal Processor Architecture for High-Efficiency MPEG-4 Video Encoding” , IEEE, 2003
- [17] Jinkui CHOI, Nozomu TOGAWA, Masao Yangisawa, and Tatsuo Ohtsuki “VLSI Architecture for a Flexible Motion Estimation with Parameters” , IEEE, 2002
- [18] Chang L. Lee , “Parallel Implementation of motion-compensation for HDTV Video decoder ” , IEEE, 1998.
- [19] Luciano Volcan Agostini, Ivan Saraiva, Sergio Bampi, “Pipelined Fast 2-D DCT Architecture for JPEG Image Compression” , IEEE, 2001.
- [20] Jarmo Takala, Jari Nikara, David Akopian, and Jukka Saarinen “Pipeline Architecture for 8x8 Discrete Cosine Transform” , IEEE, 2000.
- [21] A. Aggoun and I. Jalloh , “Two-dimensional DCT/IDCT architecture” , IEE, 2003.
- [22] Mario Kovac and Ranganathan, “JARUAR : A Fully Pipelined VLSI Architecture for JPEG Image Compression Standard” , IEEE, 1995.
- [23] 任軒, “在可動態重組系統中利用預先抓取及合併的技術降低重組負擔 /Reconfiguration Overhead Reduction Using Prefetch and Merge Techniques in Run-Time Reconfigurable System” , 碩士論文, 2003
- [24] 陳美璇, “多 FPGA 之可重組式系統中資料流處理的資料流程圖切割方法設計 /Data Flow Graph Partitioning for Stream Processing in Multi-FPGA Reconfigurable System” , 碩士論文, 2003
- [25] 曾啟華, “在應用執行過程中可重新設定組態系統中重新設定時間的減少與隱藏 /Reconfiguration Overhead Reduction and Hiding of Run-Time Reconfigurable System” , 碩士論文, 2003