

行政院國家科學委員會專題研究計畫 成果報告

以 FPGA 實現電腦斷層掃描重建演算法之硬體設計 研究成果報告(精簡版)

計畫類別：個別型
計畫編號：NSC 94-2213-E-009-130-
執行期間：94年08月01日至95年07月31日
執行單位：國立交通大學資訊科學學系(所)

計畫主持人：荊宇泰

計畫參與人員：碩士班研究生-兼任助理：吳俊瑋、林坤世

處理方式：本計畫可公開查詢

中華民國 96 年 01 月 04 日

前言

自從 1895 年德國物理學家 Roentgen 發現 X 射線以來，其在醫學影像領域的應用就受到世人的關注。然而，由於受到電腦技術的局限，真正的臨床應用時期直到二十世紀後半期才突顯出來。數學家 Radon 早在 1917 年和 1919 年分別提出 Radon Transformation 及 Inverse Radon Transformation 公式，表現在影像處理上意義即為後來的投影算變化（投影算子）。二十世紀 50 年代初期，美國神經外科醫生 Oldendorf 為了克服普通 X-Ray 成像組織結構重疊偽影 (artifact)，發表了第一篇 CT (Computer Tomography) 論文，此時才稱為電腦斷層掃描技術；1963 年，南非物理學家 Cormack 為了準確估計 X-Ray 在組織間的衰減率，第一次把非迭代的級數理論引入 CT 重建演算法中；1968~1972 年，英國工程師 Hounsfield 位了區別大腦的灰質和白質，製造了世界上第一台商用電腦斷層掃描機。Cormack 與 Hounsfield 在 1979 年獲得諾貝爾獎，並且在影像處理界，設置了 Hounsfield 獎。近 20 多年，CT 在理論演算法的研究一直處於極為活耀的地位，而且新進的成像方式，例如：MRI (Magnetic Resonance Image) 及 PET (Positron Emission Tomography) 等，在算法上與 CT 也很相似。

研究動機

CT 可以廣泛的應用在許多領域例如醫學及非破壞性的一些偵測應用，例如機場的行李檢查，海關的貨櫃檢查等。而 CT 用的 algorithm 有幾種，其中以 the filtered back-projection algorithm (以下簡稱 the fbp algorithm) 是較為成熟的演算法。但是此演算法需要大量且密集的計算，因此採用 serial implement 來實作 fbp，則時間上的花費將很大，而無法達成在 real time 內重建影像。對於此一缺點，有數個方式被提出，可歸類成三大類型，第一是針對演算法本身，進行改進，來減少時間的耗費。第二是改進演算法的所使用的硬體來加速計算，降低時間的需求。第三是利用 parallel processing 的技術來減少時間的耗費。在我們的設計中也會針對此三點來作一些調整以及改進。

文獻探討

國內目前對於類似的應用研究較少，國外對於利用硬體加速 the filtered back-projection algorithm 有較多的研究，例如 1981 年 C. Thompson and T. Peters 的 "A fractional address accumulator for fast backprojection"，以及 1985 年的 R. Hartz, D. Bristow, and N. Mullani 的 "A real time TOFPET slice backproject engine using dual Am 29116 microprocessors"，還有 1993 年的 Iskender Agi, P.J. Hurst, and K.W. Currente 等人提出利用 VLSI 技術完成 forward and backward Radon transform 專用的微處理器。

研究方法

(a)理論介紹

Filtered back-projection algorithm 是由 Bracewell and Riddle[9]所提出，為 CT scanner 常採用的演算法。主要的理論包含了 Radon transform、Filtered Projection 以及 Back Projection 這幾個部分，請參考[2]，下面是對所用到理論的簡單說明。

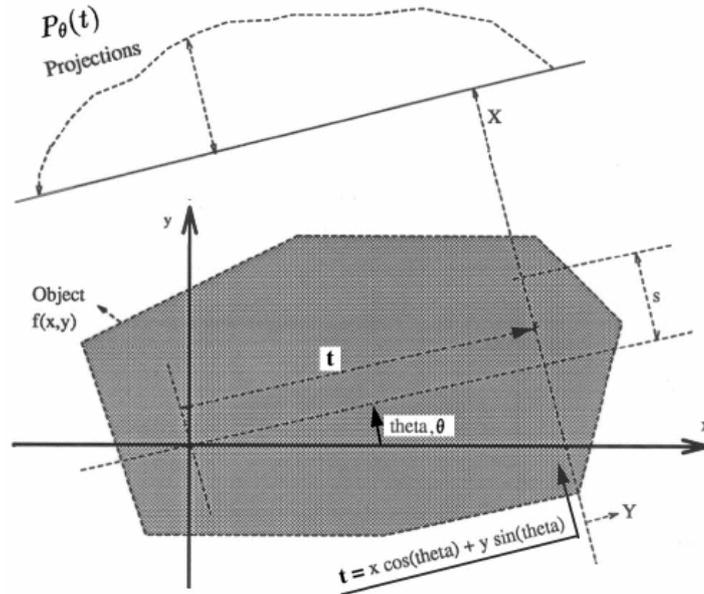


Figure 1 : Radon transform 示意圖

如上圖我們以二維函數 $f(x, y)$ 來描述物體，加上 (θ, t) 參數，以線性積分 $P_\theta(t)$ 來描述 (θ, t) 參數對應的 x-ray 射線對物體作投影而得到衰減係數總和。

$$P_\theta(t) = \int_{(\theta, t)} f(x, y) ds \quad (1)$$

使用 delta function 後可將公式(1)改寫如下

$$P_\theta(t) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(x, y) \delta(x \cos \theta + y \sin \theta - t) dx dy \quad (2)$$

公式(2)就是函數 $f(x, y)$ 的雷登轉換(Radon Transform)。若集合所有同一個角度 θ 的 $P_\theta(t)$ 值就成為該角度的投影資料。

濾波反投影演算法是利用傅立葉理論(Fourier Theory)與投影資料 $P_\theta(t)$ 來求得問題的封閉解(closed form solution)繼而重組出截面圖。所以首先定義角度為 θ 的投影資料及其傅立葉轉換如下式(3)：

$$S_\theta(\omega) = \int_{-\infty}^{\infty} P_\theta(t) e^{-j2\pi\omega t} dt \quad (3)$$

其中 ω 是 *radians/unit length*

經由公式推導[2]最終可得如下的兩式：

$$\text{濾波投影(Filtered Projection)} : Q_i(t) = \int_{-\infty}^{\infty} S_{\theta}(\omega) |\omega| e^{j2\pi\omega t} d\omega \quad (4)$$

$$\text{背投影(Back Projection)} : f(x, y) = \int_0^{\pi} Q_{\theta}(x \cos \theta + y \sin \theta) d\theta \quad (5)$$

假設 0° 到 180° 之間對待測物體共作 K 次不同角度的投影，所以演算法簡單的流程如下：

■ 公式(4)取得角度 θ_i 的投影資料 $P_{\theta_i}(t)$ ， i 從 0 到 $K-1$ ，執行下列工作共 K 次。

1. 將 $P_{\theta}(t)$ 作傅立葉轉換到頻率域，成為 $S_{\theta}(\omega)$ 。
2. 對 $S_{\theta}(\omega)$ 乘以濾波器 $|\omega|$ 在頻率域作加權，避免重建後影像剝蝕的情形。
3. 將加權後的 $S_{\theta}(\omega)$ 值作反傅立葉轉換回空間域(space domain)，成為 $Q_{\theta}(t)$ 。

■ 公式(5)從點 (x, y) 及角度 θ_i 算出 $t_j = x \cos \theta_i + y \sin \theta_i$ ， i 從 0 到 $K-1$ 。

累加每個 $Q_{\theta_i}(t)$ 中，位置 t_j 上得到的 $Q_{\theta_i}(t_j)$ 值，共累加 K 次，得到 (x, y) 點原本的線性衰減係數。

接下來利用 Figure 2 來表示整個濾波反投影演算法的重建過程，由圖中可以看出，首先經由 Radon transform 取得 Source Sinogram，接下來進入 Filtered Projection，首先對 Sinogram data 作傅立葉轉換到頻率域，之後在頻率域乘以濾波器 $|\omega|$ 做加權動作，再將加權後的值作反傅立葉轉換回空間域，成為 Filtered Sinogram，最後利用 Filtered Sinogram data 來做 Backprojection 重建回實際的影像。

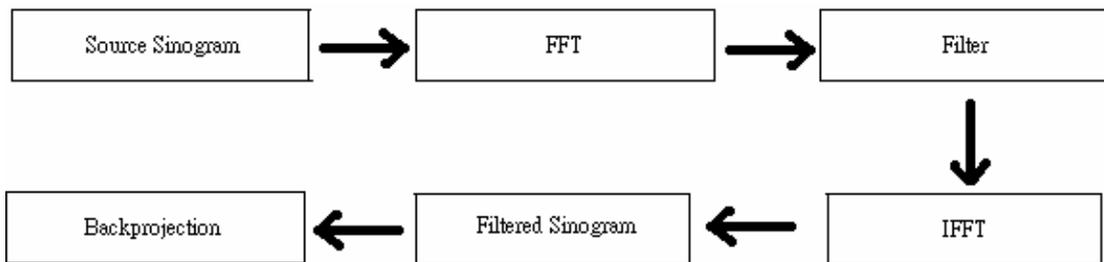


Figure 2 : Filtered back-projection algorithm 流程圖

(b)實作介紹

我們使用的平台是 Xilinx 公司的 ML310 Development Board，由於硬體資源的限制，並沒有辦法將影像一次讀入 block_ram 中作處理，因此在我們的設計中

會先對影像作一個前處理，也就是將影像切割，然後在針對每一個小區塊去作處理，最後再合併成一個完整的影像，Figure 3 是 ML310 Development Board 的系統內部架構圖。

平台的核心是一顆 Virtex- II pro FPGA(Field Programmable Gate Array)，型號是 XC2VP30。這顆 FPGA 共有 30,816 Logic Cells 提供邏輯合成的電路資源，且具有 2 顆內建的 PowerPC 405 處理器，可以當作嵌入式系統的核心處理器。主要的匯流排是 Process Local Bus(PLB)與 On-chip Peripheral Bus(OPB)，PLB bus 與 OPB bus 是 IBM CoreConnect 的標準匯流排，其中 PLB bus 是高速匯流排用來和高速設備相連接，而 OPB bus 是低速匯流排與低速設備相連接，兩個匯流排中間有個 Bridge 作連接，形成整個嵌入式系統的骨幹。

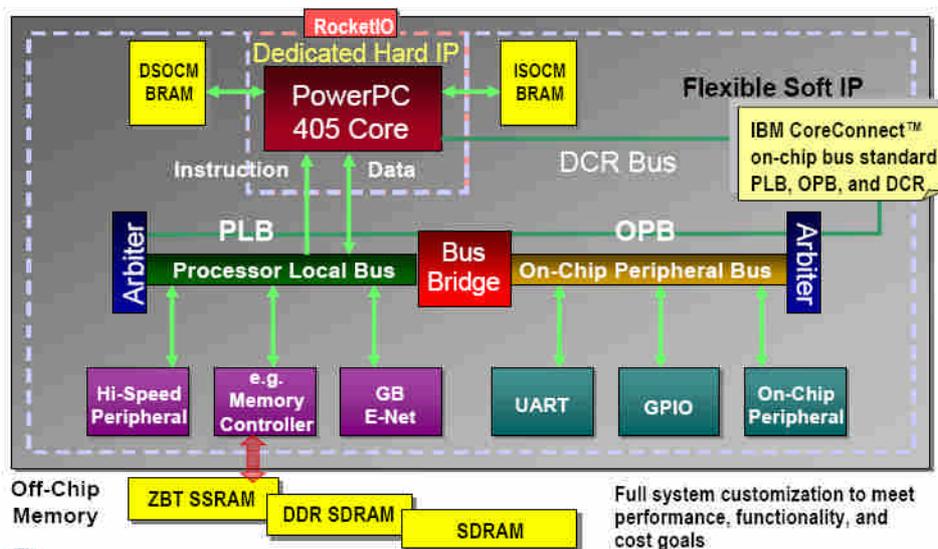


Figure 3 以 PowerPC 405 為核心的嵌入式系統架構

由於我們採用 soc 的設計方式，因此在設計剛開始便要先決定軟硬體的劃分工作，我們主要將工作劃分成四個部分，分別是：

- (a) 將影像資訊讀入以及將最後重建好的影像寫出。
- (b) 計算重建每個 block 所需資訊。
- (c) 將資料作 fft、filter、以及 ifft。
- (d) 針對每一個 block 做影像重建。

其中(a)本身是處理記憶卡(CompacFlash card)與軟體間的資料傳輸，包含一些 driver 以及 library 的使用，因此直接以軟體來實作。(b)則是因為實作時會將影像切割成小區塊，因此必須計算重建時每個 block 所需的資訊，譬如每一個區塊要對應到 Filtered Sinogram 的哪一個區段，這個部分由於運算次數以及所耗時間在可容忍範圍內，因此為增加實作的便利性，也將以軟體直接實作。至於(b)、(d)由於所需的運算量較高且亦較耗時，因此自然成為實作成硬體的重點部份。

Figure 4 是實作上所架構出來的系統架構圖，圖中包含了所有在 ML310 Development Board 上會用到的資源，以及彼此間之連結狀況，從圖上可以看出高速匯流排 Process Local Bus(PLB)主要是與 PowerPC、Sdram、Bram 以及我們自己所設計的模組 FBP 相連接。而低速匯流排 On-chip Peripheral Bus(OPB)則是與 CompactFlash 及 RS232_Uart 相連接。

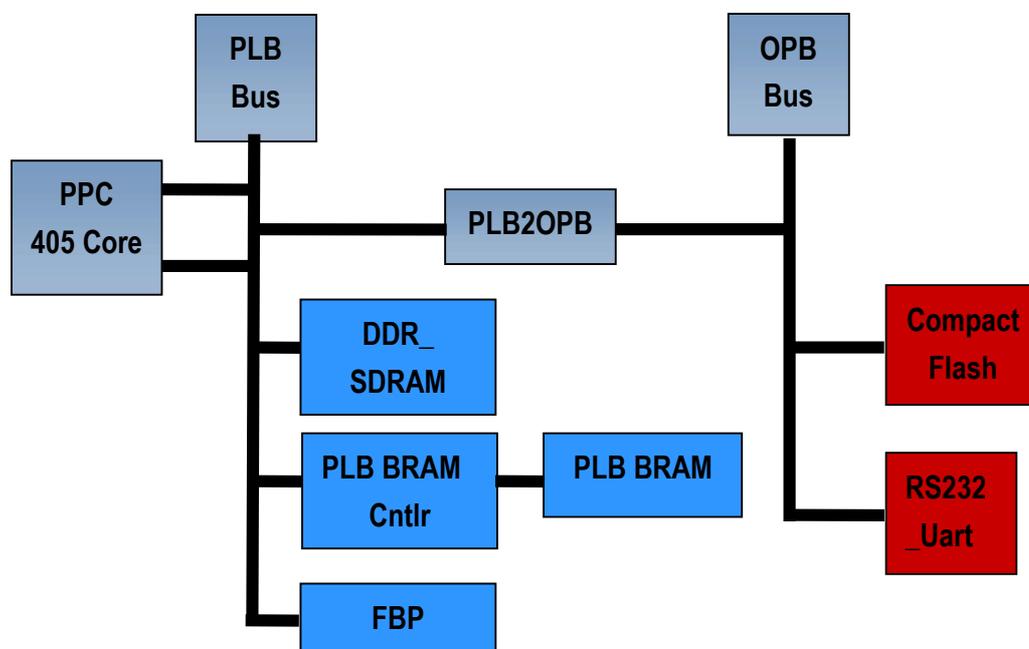


Figure 4 設定完成後之系統架構圖

在上圖中 RS232_Uart 的功用是讓使用者在 PC 上透過 terminal 程式經由 serial port 與系統溝通。而 Compact Flash 則讓使用者可以透過 System ACE CF Controller 來控制 CompactFlash card 的存取，作為 sinogram 影像及完成重建後影像的儲存設備。在這部分 Xilinx 公司有提供一個名稱為 xilfatfs 的函式庫可以幫助我們達成此功能。至於 BRAM 的部份是我們利用 Xilinx 公司提供的 Single-Port Block Memory IP 幫我們在記憶體中規劃出一塊空間供我們使用，主要的用途是當程式在硬體端執行時，可以用來暫時儲存資料。最後是 FBP 模組，這個模組主要做的工作就是讓我們在硬體裡完成整個 backprojection 的工作，接下來，我們再說明一下 backprojection 的過程。

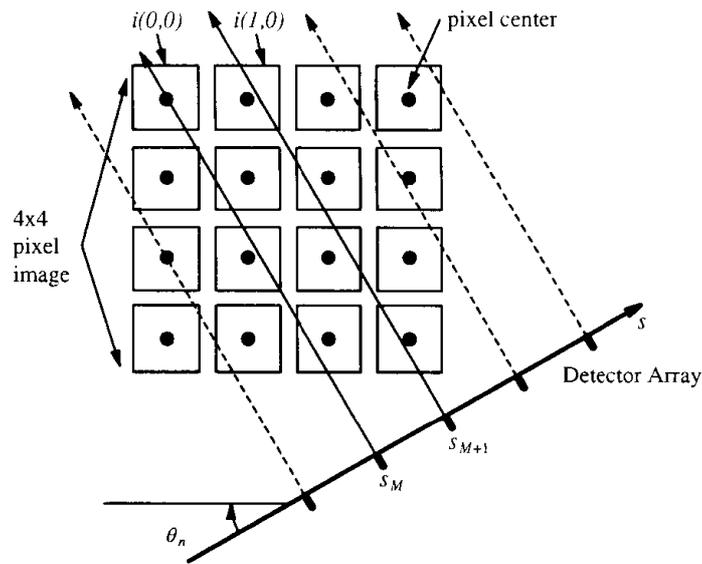


Figure 5 backprojection 示意圖

上圖 Figure 5 中的 Detector Array 指的就是相對於這個區塊的 Filtered Sinogram data，而反投影的過程就像是我們在 Detector Array 反射出一道道的光束，並紀錄光束經過的所有點，然後將 Filtered Sinogram 內的值加入此光束所經過的點。而在實際上我們令光束每次向前傳播的單位 $\Delta k=1$ ，如此計算下一點的 X 方向的變化量就是 $\Delta k \times \sin \theta$ ，而 Y 方向的變化量為 $\Delta k \times \cos \theta$ ，可輕易的算出下一個將要經過的座標。

為了加速其執行效率，我們會將 Detector Array 上的點區分成奇、偶數點，譬如 S_{M-3} 、 S_{M-1} 、 S_{M+1} 、 S_{M+3} 一組而 S_{M-2} 、 S_M 、 S_{M+2} 一組，然後奇數點與偶數點各自做平行處理，以加快整體執行效率，之所以分成奇、偶數點主要是為了確保反投影過程中不會有資料衝突的情形，如 Figure 6 所示，若不分奇、偶數點則相鄰的訊息在做反投影的時候，有可能會在同一時脈加到同一個格子點上，造成資料衝突。

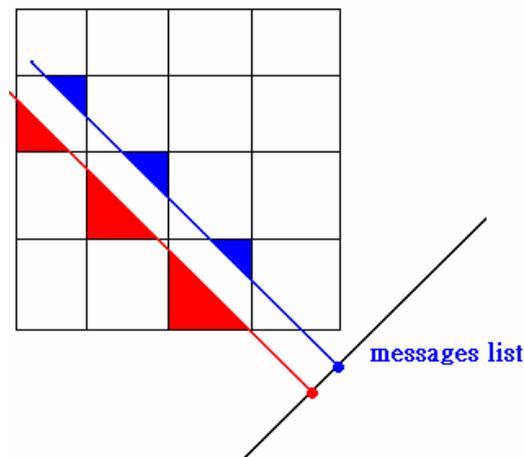


Figure 6

結果與討論

首先我們看到軟體模擬的結果，這是利用 Xilinx 所開發的一套嵌入式系統開發軟體 EDK(Embedded Development Kit)進行軟體模擬，撰寫 C 語言，搭配 ML310 中的 powerPC 進行運算，時脈為 100MHz，圖 7-1、圖 7-2 分別為 128x128 的原始圖以及其 sinogram 影像，圖 7-3 採用區塊大小為 4X4 的結果，圖 7-4 採用區塊大小為 16X16 的結果，圖 7-5 採用區塊大小為 64X64 的結果，圖 7-6 採用區塊大小為 128X128 的結果，明顯的圖 7-6 結果較好，因為不需要計算截出 sub-sinogram，可避免誤差累積，因此效果較佳。

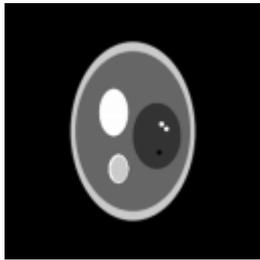


圖 7-1 原始圖

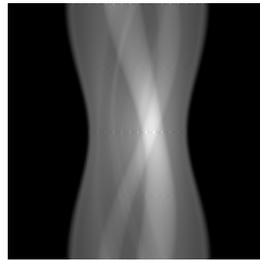


圖 7-2 sinogram 影像

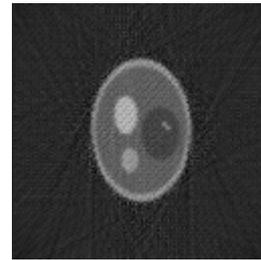


圖 7-3

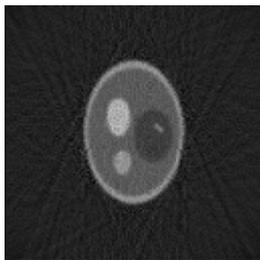


圖 7-4

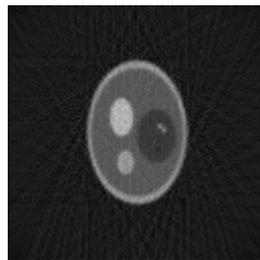


圖 7-5

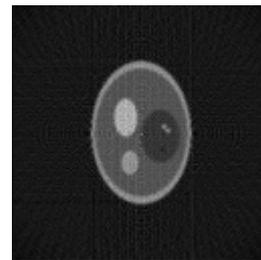


圖 7-6

同樣地，我們再影像大小 256x256 為樣本，圖 8-1、圖 8-2 分別為 256x256 的原始圖以及其 sinogram 影像，圖 8-3 採用區塊大小為 4X4 的結果，圖 8-4 採用區塊大小為 16X16 的結果，圖 8-5 採用區塊大小為 64X64 的結果，圖 8-6 採用區塊大小為 256X256 的結果，相同的道理，圖 8-6 結果較佳，因此若有更多硬體資源的話，得到較好的重建結果。

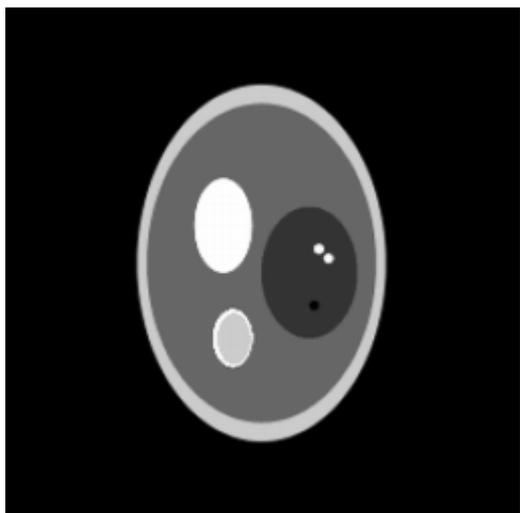


圖 8-1 原始圖

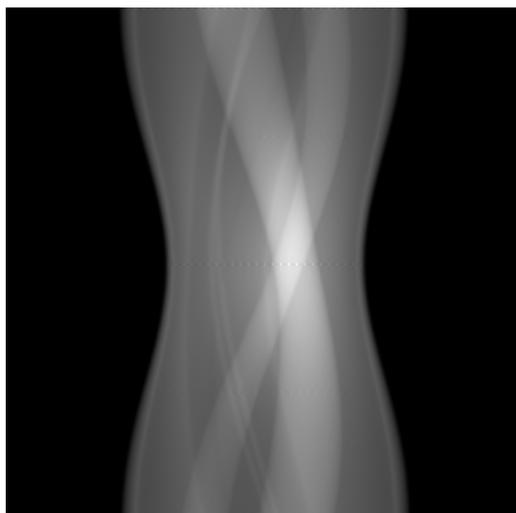


圖 8-2 sinogram 影像

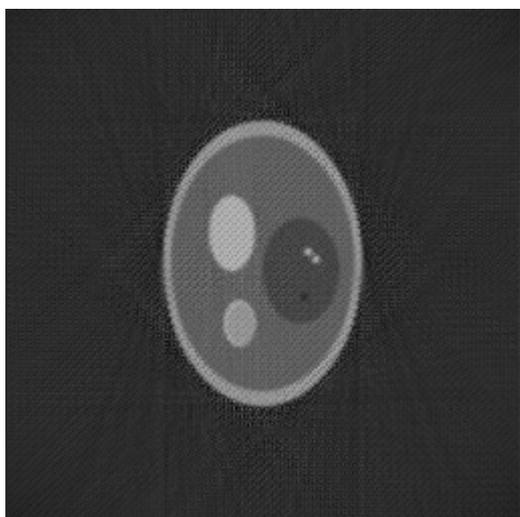


圖 8-3

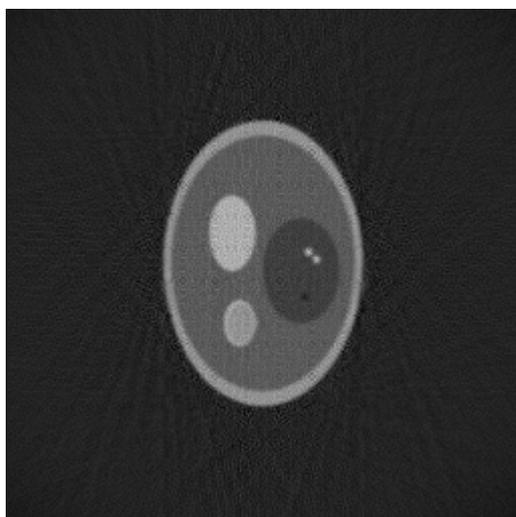


圖 8-4

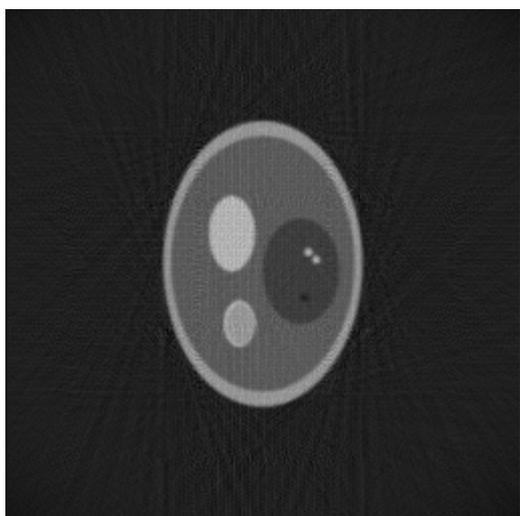


圖 8-5

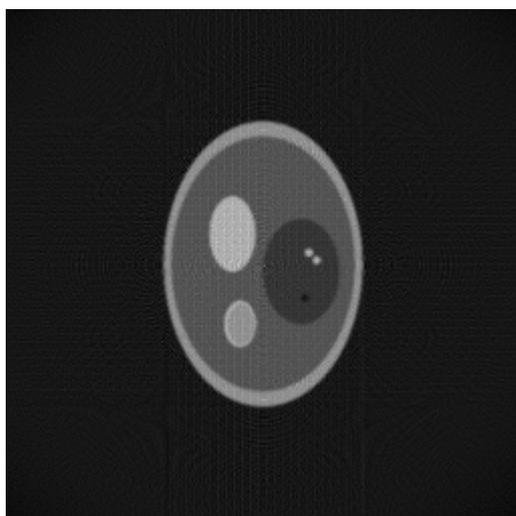


圖 8-6

接下來是我們加上硬體模組執行後的結果，圖 9-1 是以 128X128 原始圖為樣本，在硬體中採用 4X4 區塊大小重建出來的的結果，而圖 9-2 則是以 256X256 原始圖為樣本，同樣在硬體中採用 4X4 區塊大小重建出來的的結果。

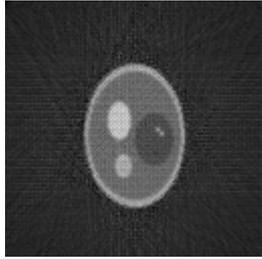


圖 9-1

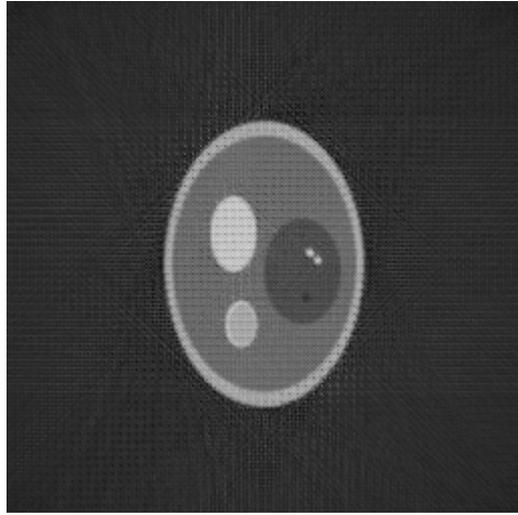


圖 9-2

表(1)是直接利用 PowerPC 執行以及加入硬體模組執行兩者間的比較結果，由表中我們可以看出，利用軟硬體共同設計的方式會比單獨使用軟體實作嵌入式系統來的有效率許多。

	比較項目	Filter 時間	Backproject 時間
128x128	軟硬體共同設計	約 1 秒	約 4 秒
	以軟體實作嵌入式系統	約 2 分鐘	約 1 分鐘
256x256	軟硬體共同設計	約 1 秒	約 30 秒
	以軟體實作嵌入式系統	約 10 分鐘	約 8 分鐘

表(1)

時脈資訊：

Minimum period : 9.951ns (Maximum frequency: 100.492MHz)

Maximum path delay from/to any node : 2.220ns

FPGA 資源使用率

Number of Block RAMs : 76 out of 136 55%

Number of occupied Slices : 11,313 out of 13,696 82%

由實驗的結果我們可以看出，利用軟硬體協同設計的方式的確可以加速影像重建的速度，此外由於在硬體的實作上，有關影像資料的儲存，我們是利用 Xilinx 公司提供的 Single-Port Block Memory IP 在記憶體中規劃出一塊空間供我們使用，因此實際上並沒有用到完整的 SDRAM 空間，未來若能直接對此 SDRAM 做操控，如此便可一次將整張影像讀入作處理，並對整張影像作 backprojection，而由軟體的經驗得知，一次對整張影像作 backprojection 將會有較好的重建品質，此外若能一次將影像讀入 memory 做處理也將會大大的減少資料傳輸的時間，若再加上良好的 pipeline 排程的設計，相信在執行效率上還會有大幅的改善。

Reference:

- [1] J. Radon, *Über die Bestimmung von Funktionen durch ihre Integralwerte längs gewisser Mannigfaltigkeiten*, Ber. Verh. Sachs. Akad. Wiss. Leipzig Math. Phys. Kl. **69**, 262-77 (1917)
- [2] A.C. Kak and M. Slaney, *Principles of Computerized Tomographic Imaging*. IEEE, Inc., New York: IEEE Press, 1988.
- [3] C. Thompson and T. Peters, "A fractional address accumulator for fast backprojection," *IEEE Transactions on Nuclear Science*, vol. NS-28, no. 4, pp. 3648-3650, 1981.
- [4] R. Hartz, D. Bristow, and N. Mullani, "A real time TOFPET slice backproject engine using dual Am 29116 microprocessors," *IEEE Transactions on Nuclear Science*, vol. NS-32, pp. 839-842, February 1985.
- [5] Iskender Agi, P.J. Hurst, and K.W. Current, "An image processing IC for backprojection and spatial histogramming in a pipelined array," *IEEE Journal of Solid State Circuits*, vol. 28, no. 3, pp. 210 - 220, 1993.
- [6] Anup B. Sharma, Keith R. Allen and Roy P. Pargas, "Some new systolic designs for two-dimensional convolution", ACM, 1988.
- [7] Bei-Chuan Chen, Yu-Tai Ching, "A new antialiased line drawing algorithm", *Computers & Graphics*, 2001.
- [8] Bertil Schmidt, Manfred Schimmler, Heiko Schroder, "Tomographic Image Reconstruction on the Instruction Systolic Array", *Computers and Artificial Intelligence*, 1995.
- [9] R.H. Bracewell and A.C. Riddle, "Inversion of fan beam scans in radio astronomy," *Astrophysics Journal*, vol. 150, pp. 427 - 434, 1967.
- [10] Alexandro M.S. Adario, Eduardo L. Roehle, Sergio Bampi, "Dynamically Reconfigurable Architecture for Image Processor Applications", DAC99, New Orleans, Louisiana, ACM, 1999.