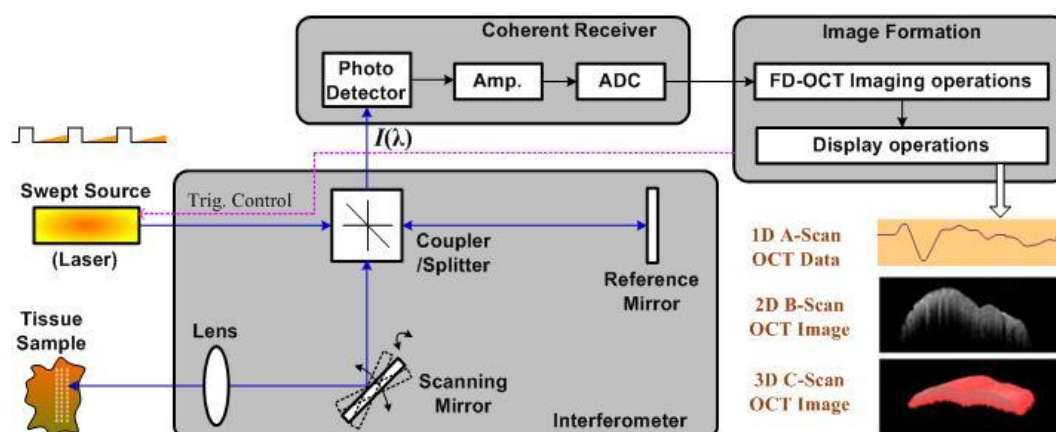


(一) 摘要

過去幾年間，光學同調斷層掃描(Optical Coherence Tomography, OCT)[1]技術有很大的進展並廣泛的運用在醫療上面，一開始做為醫療成項用途推出時採用的是 PC 平台，到後來經過二次、三次修改之後採嵌入式平台較為優勢。OCT 又可以分為時域式(TD)[2]、空間域式(SD)[2]與掃頻式(SS)[2]三種系統架構，而大多數 FD-OCT 系統使用掃頻源激光(SS-OCT)，包含三個功能塊：干涉儀，相干接收器和圖像組成。在干涉儀中，來自激光源的光束經過耦合器產生干涉信號 $I(\lambda)$ ，再藉由相干接收器中的類比數位轉換器(ADC)採樣。將採樣的 $I(\lambda)$ 干涉信號發送到圖像形成方塊，以執行 OCT 成像操作並產生數據。由於光學機械前端(干涉儀和相干接收器)和需要分離的 PC 的設備需求，使得系統變得笨重。對於面積小且能量效率高的處理器(FDOCT 成像在小型手持式設備)具有相當實際的價值。

隨著行動醫療的發展，可攜式(小型化)OCT 系統的開發也成為趨勢。小型化 OCT 裝置的成像運算平台開發也會是一個重要課題，其內容包括了 DC removal、Re-Sampling、FFT、Magnitude Compression 以及 Dynamic Range Mapping 等運作。為了讓 OCT 成像處理能達到更高的效能並有效地結合攜帶式裝置，我們會採用軟硬體協同設計(software/hardware co-design)[3]的方式，針對各個成像運算功能的特性與其適用於軟硬體做實現的優缺點，分成硬體(HW)及軟體(SW)兩部分來合作完成。運算較複雜且較有速度考量的部分(如 DC removal、Re-Sampling 與 FFT)由硬體來執行，其他較需運作彈性且與 display 端有關的部分(如 Magnitude Compression 與 Dynamic Range Mapping)則由軟體來完成。兩部份的設計除了各自所屬的開發外，並會以協同的方式完成整體 OCT 成像運算，其中涵蓋了軟硬體之間的整合流程、運作控制、資料傳遞等，以期達到最有效率的系統呈現。



圖一、使用FDOCT 系統處理方塊掃頻式激光(SS-OCT)

(二) 研究動機與研究問題

目前 OCT 成像運算利用 PC 為平台來執行的居多。因應可攜式 OCT 的應用，本專題計畫以軟硬體協同設計 (software/hardware co-design)[3] 整合 FPGA 的硬體設計及行動裝置(如手機或平板電腦)的軟體開發，發展小型化 OCT 裝置的成像處理平台，達到方便性、立即性及普遍性。

軟硬體協同設計利用軟硬體分別有的各項優點解決過去只用單一硬體或軟體的缺點，不僅降低硬體設計的風險也加速整體開發的時間。而 OCT 系統也得考量軟體的運作及軟硬體間的整合與協同設計。如圖三，FD-OCT 的成像處理，開發者可以事先規劃軟硬體的分工以進行協同設計。

在做軟硬體協同設計時任何需要用軟體或硬體去實現的功能都必須要考慮到系統的效率與成本，通常會將常用的部分做成硬體，因為硬體處理速度較快但是成本會比較高，我們的目標是為了得到最大的效率。像是圖三中，「Dynamic Range Mapping」[9]和「Magnitude Compression」[10]，因為此部分運算需要多一點彈性和調整，所以將其規劃至軟體端執行，以產生最快的速度與最多的產出，而其之前的運算較為複雜，考慮到效率問題，我們安排由硬體端做實現。最後，軟硬體間互相溝通，實現軟硬體協同設計，以此來降低整體成本而達到最高效率取得平衡。不過，在協同的部分可能會遇到軟硬體間無法互相溝通或無法正確讀取與寫入等的情況，需要檢查一下硬體的規格，或是開發語言不支持接受硬體的資料。硬體部份我們將用 Verilog，軟體部分會用 C 語言來實現本次專題。

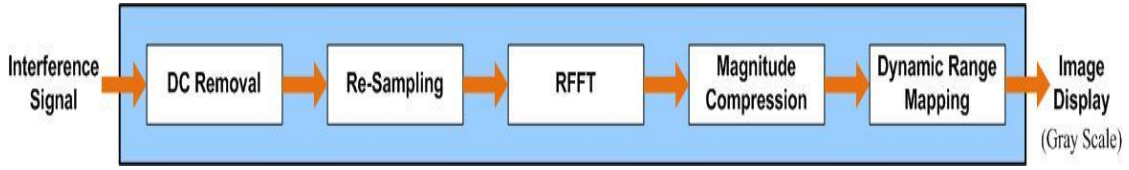
(三) 文獻回顧與探討

1. FD-OCT 之成像運算

如下圖，FD-OCT 的成像運算包括 DC 移除或過濾(DC Removal or Filter)、重新取樣(Re-Sampling)、快速傅立葉轉換(Fast Fourier Transform; FFT)、數值壓縮(Magnitude Compression)以及動態範圍映射(Dynamic Range Mapping)等主要運算，我們將會利用 FPGA-SoC 來實作 OCT 成像運算。

干涉信號 $I(\lambda)$ (圖一)常常會包含 DC 背景雜訊，這也意味著直流項在第一階段就應該被移除[4]。直流項去除後的干涉信號 $I'(\lambda)$ 存在於波長域(λ -space)之中。因此，重新取樣(Re-Sampling)的運算[5]-[6]是為了轉換 $I'(\lambda)$ 變成 $I'(w)$ ，而 $I'(w)$ 是一個表示 A-line 訊號頻譜的線性訊號並存在於頻域(k-space)之中。一個深度解析的圖形(Depth-resolved profile)(對應於時間)可以通過對 $I'(w)$ 做 FFT(或是逆 IFFT)運算[1], [5]-[7]來獲得。隨後 FFT 的結果會為了產生顯示數據而被送出，其中 FFT 輸出的數值會被壓縮成適當的動態範圍[4], [8]。干擾訊號是一系列的實值數據，因此在沒有訊號處理的情況下，圖二中的 FFT 輸入為實數。這意味著，相應的 FFT 執行是一個實值 FFT(RFFT)運算[4]-[5], [7]。當把 RFFT 應用於一個

實際的 N 點序列 $x(n)$ 時，在 FFT 的結果 $X(k)$ 中， $X(k)$ 和 $X(-k)$ 在理論上表現為埃爾米特對稱(Hermitian symmetry)。因此，在 A-scan 顯示數據中只使用一半(即 $N/2$)的 FFT 輸出取樣。



圖二、FD-OCT 系統的成像處理方塊圖

因在經過 RFFT 功能方塊運算後的 A-Scan data 有可能產生差異較大的值，導致在成像的時候會浪費額外的 bits 來顯示不重要的資訊。而為了解決這個問題，Magnitude Compression 的功能即是系統會將 FFT 這個運算後的值的大小以 $\log[.]$ 的計算方式進行「壓縮」，其用意是為了可以降低數值範圍。

對應到 Dynamic Range Mapping，其運作則是為了可以更有效地呈現合適對比度的 OCT 影像，每一條 A-Scan(或每組 frame)的壓縮後的最大值會被系統會正規化至「1」($\log[.]$ 的最大值訂為 0 dB 基準)，且將 0 dB 以下至若干 dB 的區間規定為動態範圍(dynamic range)，同時規定此範圍內的值對應至 0~255 的 8-bit 灰階值(gray scale)可以顯示 OCT 影像。

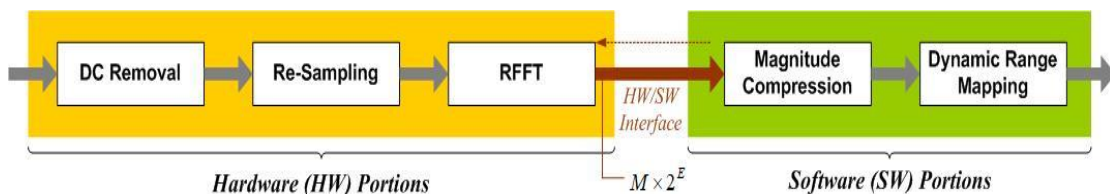
上段內容可由 eq.(1)的運算式來描述， nor_MC 是指 Magnitude Compression 輸出之 $\log[.]$ 值正規化後的結果，而 DR_{dB} 是上述所設定之 0-dB 以下的動態範圍(dynamic range)； $\lfloor \cdot \rfloor$ 與 $rnd\{\cdot\}$ 則各自表示負值的 clipping 處理與四捨五入的運算。

$$Gray\ Scale = rnd\left\{\left\lfloor nor_MC \times \left(\frac{255}{DR_{dB}}\right) + 255 \right\rfloor\right\} \quad (1)$$

(四) 研究方法及步驟

1. 整體架構

本專題計畫所開發的 FPGA-SoC 在執行 FD-OCT 成像運算時，其運作架構如圖三所示。



圖三、整體計劃架構的處理方塊

DC Removal、Re-Sampling 及 RFFT 用硬體去實現，而 Magnitude Compression 及 Dynamic Range Mapping 則用軟體去實現，而在 RFFT 功能方塊要傳送資料結果給 Magnitude Compression 中間傳送複數(例如： $a+bj$)有些比較困難的問題。因電腦使用二進位制的運算[11]，所以在計算機科學中會使用浮點數(floating-point number)做運算，浮點(floating point, FP)是一種對於實數的近似數值的表現法，而利用其進行浮點計算。而由於二進位制也無法完全準確的換算成十進位制的部分小數，如：0.1，因此只能使用近似值的表達方式來呈現結果。eq.(2)為十進位制浮點數的表示方式。

$$3.1415 = \underbrace{31415}_{\text{mantissa}} \times 10^{-4} \quad (2) \quad \text{exponent}$$

假如 a 有 20bits，但因電腦幾乎是以二進位制作傳送資料的動作，所以我們利用以 2 為基數的冪數的方法，將 a 由 M 和 E 來表示： $a = M \times 2^E$ ，M 為尾數(Mantissa)，E 為指數(Exponent)。令 M + E 有 16bits 的方式做傳送，如此以來用硬體實現的部分做出的結果可以更容易的由軟體接手做運算。

2. 硬體設計內容

(1) DC 移除(DC Removal):

DC 噪音的測量結果(DC 項) 會被預先儲存在 RAM，並直接從干擾信號中被減去。

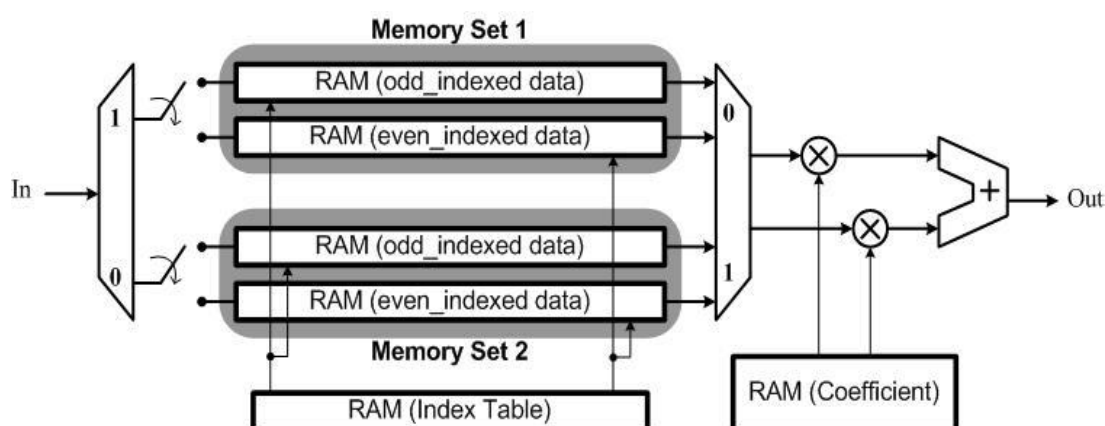
(2) 重新取樣(Re-Sampling):

插值被應用於兩個相鄰的數據取樣，這兩個數據取樣由最接近重新取樣紀錄的兩個整數去索引，來取得重新取樣的數據。圖四說明了用於線性插值(包括前端 DC 移除)的所提出的 Re-Sampler 的架構。兩組記憶體模組輪流儲存通過數據流的 DC 移除的輸入，由此每個數據流中的奇數或偶數索引數據被會被分開做儲存的動作。基於重新取樣索引(預先存儲在 RAM 中)的值，我們可以確定兩個最接近的整數索引(奇數或偶數)及其對應的插值係數，以便訪問兩個相鄰的數據流進行線性插值運算。

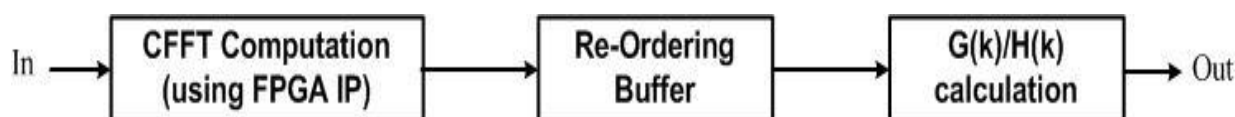
(3) 實數型快速傅立葉轉換(Real Fast Fourier Transform；RFFT)

執行 RFFT 背後的想法是使用複數的 FFT(即 CFFT)，其中所輸入的虛部被設置為零。這個方法很簡單，但它不是有效率的利用 CFFT 硬體資源。現存一種的方法為透過使用 CFFT 來同時計算兩個 RFFT 來考慮硬體效率。例如，兩個實數序列 $g(n)$ 和 $h(n)$ ($n = 0, 1, \dots, N-1$) 的兩個原始 N 點的 RFFT 可以通過應用 N 點的 CFFT 到一個複數序列 $f(n)=g(n)+j^* h(n)$ 。如果我們令 $g(k)$ 和 $H(k)$ 分別是 $g(n)$

和 $h(n)$ 的 FFT 的正 $N/2$ 共軛對稱部分，而 $F(k)$ 是 N 點的 FFT 的 $F(N)$ 輸出。 $G(k)$ 跟 $H(k)$ 的結果通過 $G(k) = 1/2 * [F(k) + F'(-k)]$ 和 $H(k) = (-j/2) * [F(k) - F'(-k)]$ ，其中 $F'(\cdot)$ 指複數的共軛項。上述策略使得可以使用 N 點 RFFT 的「two streams」代替與後處理共同的單個「one-stream」 N 點 CFFT，這適用於基於 CFFT 的硬體且較高效率的 RFFT 計算。



圖四、用於線性插值的 Re-Sampler 的架構



圖五、RFFT 的運作流程

在我們的設計裡，顯示處理包含以下三個操作：

- (1) 數值計算：假設 RFFT 的輸出值是 $a + j * b$ ，則 $a^2 + b^2$ (或其平方根) 被計算為一個數值。
- (2) 對數壓縮：數值被標準化 (即 $(a^2 + b^2)$) 到 0 和 1 之內，然後通過 $10 \times \log_{10}[(a^2 + b^2)]$ 操作進行壓縮以獲得以分貝 (dB) 為單位的對數壓縮值。
- (3) 增益和動態範圍映射：對於給定的增益偏移量和動態範圍，對數壓縮值被轉換 (映射) 為 8-bit gray scale，用於圖像顯示。這裡我們以 open CV 做為例子，因人眼對綠色的敏感度會比較大，對藍色最小，因此將綠色權重擺得比較大，藍色較小，我們要把 RGB 變灰階，其公式為 $Y = 0.299R + 0.587G + 0.114 * B$ ，而把灰階轉為 RGB 就變成 $B=Y, G=Y, R=Y$ 。[12]

3. 軟體設計內容

通過軟硬體協同的方法[13]，來實現 OpenCV 演算法向高性能處理平台 Zynq SOC 系統的移植和加速。Xilinx Zynq SOC 是具有 ARM+FPGA 架構的全可編程處理器，集成了雙核 Cortex-A9 處理系統、FPGA 邏輯和一些關鍵外設，提供了軟硬體和 I/O 可編程性。其邏輯和外設在 Zynq SOC 中的高度集成性確保了相對於採用分離式組建設計的系統而言提高數據傳輸速率，高度的軟硬體集成性能幫助設計人員實現極為高效的嵌入式視覺系統。

Vivado HLS 高級語言綜合工具開發 OpenCV 時[14]，編寫源碼和測試文件，並且採用 C++ 編譯器進行編譯、仿真和調試，產生可執行文件，之後用 I/O 函數提取 FPGA 實現部分，如圖三，且使用可綜合的 Vivado HLS 庫函數代碼代替 OpenCV 函數的調用。

Survey & Study :

研究時間：2 個月

收集有關 FD-OCT 成像及增益和動態範圍映射資料的文獻並加以研究，研究其原理及效能之後設計軟硬體架構。且透過 Survey & Study 我們初步估測各種方法的優劣及效率差異，並希望做優化以讓效率提升。

階段一：FD-OCT 成像應用

研究時間：1 個月

執行方式：前面已經有熟悉 FD-OCT 成像原理，此階段找到合適的干涉信號 $I(\lambda)$ ，並使用 C 語言來試著實現 FD-OCT 的成像運算。

階段二：FPGA 成像硬體設計

研究時間：2 個月

執行方式：使用上一階段的成像運算結果，此階段要設計合適的軟硬體協同設計部份，我們使用 Verilog 語言及電路合成的方式來實現硬體的部份。

階段三：FPGA 成像軟體設計

研究時間：3 個月

執行方式：基於第一階段的成像運算結果，此階段要設計合適的軟硬體協同設計部份，熟悉 FPGA-SoC 的 SDK，並使用 C 語言與 OpenCV 讓資料所應對的圖形顯示於輸出設備，設備使用單獨的電腦做為顯示設備。

階段四：成像系統整合與測試

研究時間：1.5 個月

執行方式：整合第二、三階段完成的成像硬體與軟體，並測試成果。

階段五：成像與系統整合與測試

研究時間：2 個月

執行方式：將完成的成像系統與 OCT 系統整合起來，並測試成果。

階段六：製作成果報告與相關資料

研究時間： 1.5 個月

執行方式： 撰寫出本計畫的成果報告以及其他相關資料。

持續學習(計畫結束後)：將成果拿去參加各種論文比賽或申請專利

執行時間：計畫結束後仍持續進行

執行方式：請老師指導如何撰寫論文，並將此專題撰寫成期刊或會議論文進行發表。並持續在老師的指導下參與國內相關資訊專題競賽，甚至可以到國外，之後與 ITRI 進行後續技術合作與推廣。長遠規劃可以建立一個雲端系統，透過大量在雲端資料庫 累積樣本，把大量數據交由人工智慧(AI)來做分析、辨別，以期可透過雲端系統輔助醫療人員做症狀辨識。除此之外，此計畫的設計成果(包含基於雲端的症狀辨識方法)如有創新的方案，我們會進一步配合學校申請專利。

執行時間 執行項目	2018 年												2019 年		
	2 月	3 月	4 月	5 月	6 月	7 月	8 月	9 月	10 月	11 月	12 月	1 月	2 月	3 月	
Survey&Study															
階段一															
階段二															
階段三															
階段四															
階段五															
階段六															
計畫結束															

圖六、本專題計畫執行時間之甘特圖

(五) 預期結果

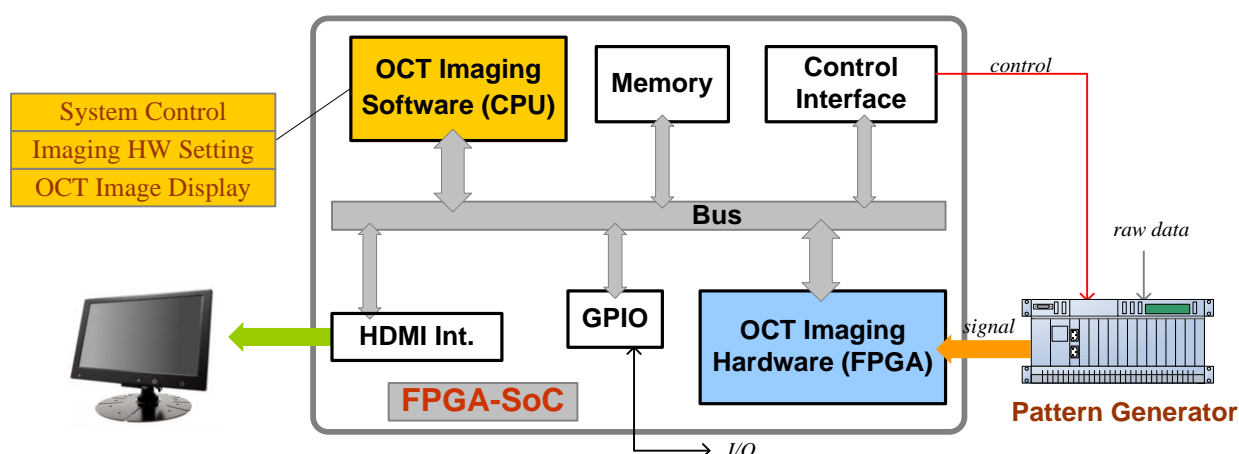
因 FD-OCT 成像處理需要前級光機所產生的干涉信號作為輸入，所以在剛開始設計的階段，我們會先在現有的系統端擷取干涉信號來作為成像運算暨系統開發的 raw data，這樣在執行後續的設計流程會較有幫助。針對此部分，本計畫會與工研院(ITRI)/生醫中心進行合作，藉由 ITRI 所開發的 SS-OCT 或 SD-OCT 平台，我們可取得實際干涉信號的 raw data 樣本。所擷取之樣本可依屬性儲存於樣本產生器(pattern generator)內，而後續的開發可分為以下兩個階段。

第一，我們會在電腦平台上設計軟體程式去執行 FD-OCT 成像運算與校準處理計算，以作為整體開發的基礎。接著以軟體的模擬結果，評估出 FD-OCT 成像硬體的定點數(fixed-point)規劃並發展出與硬體設計結果做比對的環境。

在第一個階段可重置 FD-OCT 成像硬體的設計完成後，接下來我們會藉由內建 CPU 的 FPGA-SoC 平台作為基礎去進行軟硬體協同設計與 OCT 系統的開發，其整體 OCT 系統開發的架構與輪廓見圖七所描述。而其中關於軟體(SW)與硬體

(HW)的分工，成像硬體的運算模式以及軟體的功能可對應於前一段的內容。

一開始會將成像硬體加入前端訊號擷取的介面電路，並於 FPGA-SoC 平台上去實現與驗證。如圖七所示，FD-OCT 成像硬體可從前級的樣本產生器擷取干涉信號的 raw data 進行成像運算，並透過匯流排(Bus)與 CPU 進行數據傳遞與溝通；而 CPU 所負責的是，執行相關的軟體處理或計算，並建構起整體的系統操控。在系統運作時，CPU/SW 可透過控制介面對前級光機模組或圖形產生器進行控制，並對 FD-OCT 成像器執行設定與數據接收，而經過 CPU/SW 處理後的影像訊息透過 HDMI 介面送至顯示器做顯像。



圖七、整體 OCT 系統與軟硬體協同設計的開發架構

(六)參考文獻

- [1] J. M. Schmitt, "Optical Coherence Tomography (OCT): a Review," *IEEE J. Sel. Topics Quantum Electron.*, vol. 5, no. 4, pp. 1205-1215, July/Aug. 1999.
- [2] 材料世界網 OCT 技術概述與生醫應用
url:<https://www.materialsnet.com.tw/DocView.aspx?id=27579OCT>
- [3] SoC 軟硬體協同設計與實例
url:<http://speed.cis.nctu.edu.tw/~ydlin/miscpub/soc-2005.pdf>
- [4] M. Ali and R. Parlapalli, "Signal processing overview of optical coherence tomography systems for medical imaging" Texas Instruments, Inc., White Paper, 2010. Available: <http://www.ti.com/lit/wp/sprabb9/sprabb9.pdf>
- [5] R. Leitgeb, C. K. Hitzenberger and A. F. Fercher, "Performance of fourier domain vs. time domain optical coherence tomography," *Opt. Express*, vol. 11, no. 8, pp. 889–894, Apr. 2003.

- [6] Y. Yasuno, V. D. Madjarova, S. Makita, M. Akiba, A. Morosawa, C. Chong, T. Sakai, K.-P. Chan M. Itoh and T. Yatagai, “Three-Dimensional and High-Speed Swept-Source Optical Coherence Tomography for in vivo Investigation of Human Anterior Eye Segments,” *Optics Express*. vol. 13, no. 26, pp. 10652–10664, Dec. 2005.
- [7] P. H. Tomlins and R. K. Wang, “Theory, development and applications of optical coherence tomography,” *J. of Physics D: Applied Physics*, vol. 38, pp. 2519-2535, 2005.
- [8] Kang Zhang, and Jin U. Kang, “Graphics Processing Unit-Based Ultrahigh Speed Real-Time Fourier Domain Optical Coherence Tomography,” *IEEE J. Sel. Topics Quantum Electron.*, vol. 18, no. 4, pp. 1270-1279, July/Aug. 2012.
- [9] Teoman E. Ustun, Nicusor V. Iftimia, R. Daniel Ferguson, and Daniel X. Hammer, “Real-time processing for Fourier domain optical coherence tomography using a field programmable gate array,” *Review of Scientific Instruments*, vol. 79, no. 11, 114301, 2008.
- [10] A. E. Desjardins, B. J. Vakoc, M. J. Suter, S. -H. Yun, G. J. Tearney, and B. E. Bouma, “Real-time FPGA processing for high-speed optical frequency domain imaging,” *IEEE Trans. Med. Imag.* vol. 28, no. 9, pp. 1468–1472, Sep. 2009.
- [11] 浮點數 wiki
url:<https://zh.wikipedia.org/wiki/%E6%B5%AE%E7%82%B9%E6%95%B0>
- [12] [OpenCV] 轉換影像為灰階 (Transform Image to Gray Level)
url:<https://cg2010studio.com/2011/06/06/opencv-%E8%BD%89%E6%8F%9B%E5%BD%B1%E5%83%8F%E7%82%BA%E7%81%B0%E9%9A%8E-transform-image-to-gray-level/>
- [13] 一種硬體加速 OpenCV 的圖像處理方法研究
url:<https://kknews.cc/zh-tw/tech/p89myz.html>
- [14] 何賓，張艷輝. Xilinx FPGA 數位訊號處理權威指南[M].北京：清華大學出版社，2014.

(七)需要指導教授指導內容

1. FD-OCT 系統運作概念的建立
2. FD-OCT 成像運算
3. 相關文獻 Survey 與討論
4. FPGA 硬體架構設計與 Verilog 語言的撰寫
5. SDK 結合 open CV 的處理
6. FPGA-SoC 開發介面指導
7. 成果報告的撰寫技巧

8. 期刊或會議論文的撰寫指導
9. 與工研院的技術合作之指導
10. 如何規劃專利申請