

行政院原子能委員會  
委託研究計畫研究報告

基於 FPGA 之高解析度 PET/SPECT 訊號擷取系統設計  
FPGA-Based High Resolution PET/SPECT Signal Acquisition  
System Design

計畫編號：992001INER073

受委託機關(構)：國立臺灣海洋大學

計畫主持人：吳宗達

核研所聯絡人員：張鍾鴻

聯絡電話：02-24622192 EXT：6251

E-mail address：tdwu@mail.ntou.edu.tw

報告日期：99.12.01

## 目錄

目錄.....	I
中文摘要.....	1
英文摘要.....	2
壹、計畫緣起與目的.....	3
一、計畫緣起.....	3
二、計畫目的.....	5
三、正子斷層造影原理.....	6
四、正子乳房專用造影儀系統架構.....	7
五、論文整體架構.....	9
貳、研究方法與過程.....	10
一、文獻探討.....	10
二、硬體原型架構.....	11
(一) 脈衝產生器.....	13
(二) 數位訊號處理發展板.....	14
(三) 數位 I/O 介面卡.....	14
三、系統概述.....	15
四、研究相關原理.....	17

(一) CFD 原理.....	17
(二) 線性擬合.....	18
五、模組設計與說明.....	19
(一) 同步時脈單元.....	19
(二) 系統時間單元.....	21
(三) 事件擷取模組.....	23
(四) 數位 CFD 模組.....	26
(五) 觸發信號產生模組.....	31
(六) 符合事件計算系統模組.....	32
(七) 測試訊號產生單元.....	38
<b>參、主要發現與結論.....</b>	<b>40</b>
一、時間模組電路之測試.....	40
(一) 事件擷取模組測試.....	41
(二) 數位 CFD 模組測試.....	43
二、符合事件計算系統模組電路之測試.....	48
(一) 總符合事件的輸出測試.....	48
三、時間模組 1 對 1 系統整合測試.....	55
(一) 數位訊號處理模組與 PCIe 介面整合測試.....	57

四、4 對 4 信號擷取系統整合測試.....	61
五、結論.....	78
肆、參考文獻.....	<b>80</b>

## 中文摘要

### 基於 FPGA 之高解析度 PET/SPECT 訊號擷取系統設計

時間準確性是影響正子斷層造影儀的效能之關鍵因素。本計劃的目標是設計適用於正子斷層造影儀的全數位時間鑑別系統，主要是使用數位定比鑑別器以及數位內插技術，以萃取事件時間並且將時間解析度最佳化。硬體原型部份包含脈衝產生器、Xilinx Virtex-4 現場可程式化閘陣列以及 105-MHz, 14 位元之類比數位轉換器。而電路設計將以 Verilog 硬體描述語言來實現。本論文提出了兩種數位內插演算法，它們都是由線性擬合方法所推演而來，分別為權重法以及二分法，使用該方法可得到 32 倍的時間準確度，在文中將會比較權重法以及二分法之間的差別。最後，經實驗結果得知，於 1 對 1 的偵檢器系統當中可以有效地達到 550ps 的符合事件時間解析度。此外本系統亦與後級之事件符合計算系統整合成功，符合事件對之資料可存入電腦中等待成像處理。

## 英文摘要

### **FPGA-Based High Resolution PET/SPECT Signal Acquisition**

#### **System Design**

Time accuracy is a key factor affecting the performance of a positron emission tomography (PET). A fully digital time discrimination system for PET is presented in this thesis. The digital constant fraction discriminator (dCFD) and digital interpolated techniques are used to extract event time and optimize timing resolution. Two kinds of digital interpolated algorithms deduced from linear fit method are proposed, which are respectively weighted and bisection methods. These algorithms were designed by Verilog hardware description language (HDL) and implemented on a single FPGA. In our design, 5 bits of timing-mark are used in order to obtain 32 times of original time precision.

The hardware prototype for the time discrimination experiments includes a pulse generator, a Xilinx Virtex-4 field programmable gate array (FPGA) and 105-MHz, 14-bit analog to digital converters (ADC). The experiment results show that the timing resolution of 550ps Full Width Half Maximum (FWHM) can be effectively achieve for both the weighted method and bisection method.

## 壹、計畫緣起與目的

### 一、計畫緣起

正子斷層造影儀(Positron Emission Tomography，簡稱 PET)是使用在臨床上的核醫檢測技術之一，比起 X 光、電腦斷層掃描(Computed Tomography)、磁振造影 (Magnetic Resonance Imaging)等傳統檢查，更利於偵測人體中的病變細胞位置，使得癌症能夠提早篩檢出，因此若能提升 PET 的判斷準確率，則對醫療界是很大的貢獻。

影響準確率的內在因素源自於 PET 的系統效能，包括感測晶體材料的或是前端偵檢器(detector)的靈敏度，而在電機相關領域裡所關注的是——電子系統的訊號品質，若能有效地提升它的解析度，則 PET 的影像品質更佳，誤判率也能改善，而傳統上使用類比電路來進行訊號處理，基於特殊應用積體電路(Application-Specific Integrated Circuit，簡稱 ASIC)的設計所挾帶而來的缺點是造價昂貴且開發時間較長，故近年來的趨勢是選擇採用全數位式電路的技術來取代 ASIC，意即在處理正子訊號的能量、位置以及時間資訊皆使用數位電路來實現，全數位式設計具有成本低、高彈性、擴充性佳等特性，它不需要等待 IC 下線(Tape-out)，馬上就能進行功能與時序

的除錯驗證，也不用因硬體升級或功能的修改，而要額外增加電路面積，僅需更動其軟體設定即可。

然而，使用全數位的電路又會面臨到另一個問題——類比數位轉換器(Analog to Digital Converter，簡稱 ADC)取樣率的限制，而判定符合事件(coincidence)是 PET 的核心工作，它必須比較三個資訊有無符合，包含位置、能量、時間，才能決定它為正子事件對，一般而言，正子脈衝訊號上升時間極短，產生頻率也很快，ADC 取樣到的資料並無法表示事件的全貌，特別是精準時間資料的取得最具難度，若能提出有效的方法來取代高速硬體，便可兼顧了成本和整體效率。



## 二、計畫目的

本研究主要基於現場可程式閘陣列(Field Programmable Gate Array, 簡稱 FPGA)技術, 以開發適用於正子乳房專用造影儀的資料擷取系統以及數位訊號處理系統, 並且設計適合於硬體的演算法, 以得到最佳化的訊號品質。重點項目包含: 實現高速資料擷取 I/O 介面模組, 以提高系統的處理速度, 並實現數位式的定比鑑別器(Constant Fraction Discriminator, 簡稱 CFD)來處理粗略的時間資料, 於數位 CFD 後端則分別實現「權重內插」和「二分內插」兩種不同電路架構來有效地提升時間解析度, 演算完成的結果會透過 PCI-Express(PCIe)傳輸介面匯入至電腦端並且存檔, 以利資料的後處理與分析。

### 三、正子斷層造影原理

在進行正子斷層造影之前，病患需要注射一種具微量放射性的顯影藥劑，在臨床上最常使用的是 FDG (氟代去氧葡萄糖)，藥劑進入體內後會被器官組織所吸收，在代謝過程中，FDG 開始衰變，則放射出正子與體內細胞的電子撞擊而進行質能轉換，這道過程稱之為「互毀作用」(annihilation)，如圖 1.1 所示，所衍生的結果是一道反向約  $180^\circ$  的  $\gamma$  射線，稱之為光子對(photon pairs)事件，而它的能量為 511k eV。



圖 1.1 互毀作用示意圖

PET 即是偵測這群  $\gamma$  射線來做為篩檢癌細胞的依據，其中由於癌細胞的代謝速度遠比正常細胞來得快上許多，故它同時也會吸收較多的藥劑，正子事件將集中在病灶可能的位置，而在 PET 的螢幕端呈現深色的影像資料，以幫助醫師做進一步的診斷。

#### 四、正子乳房專用造影儀系統架構

以正子乳房專用造影儀為例，如圖 1.2 所示，其整體架構大致上可以分為五個部分：偵檢器、類比前端電路、數位訊號處理模組、資料傳輸介面以及伺服器端，以下將簡述其各部份的主要功能。

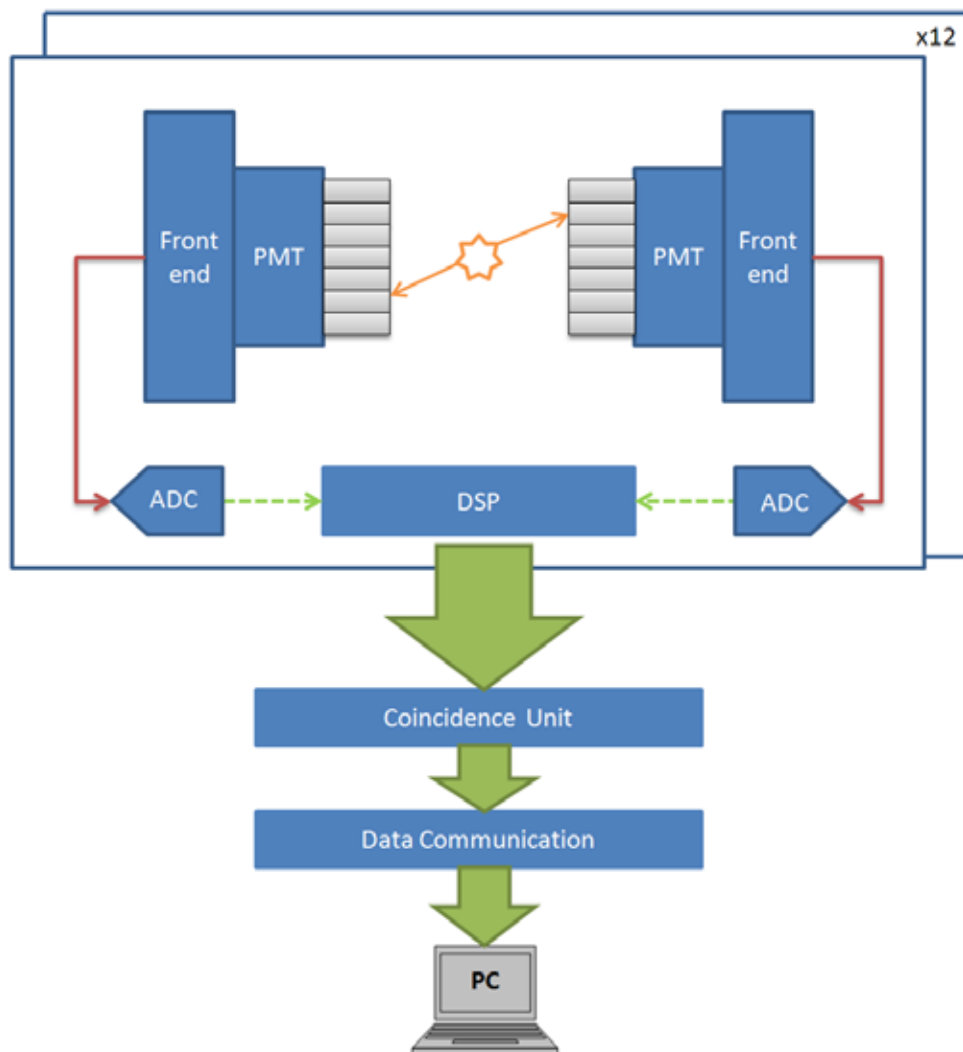


圖 1.2 正子乳房專用造影儀系統架構圖

第一部分為由晶體陣列和光電倍增管(Photomultiplier, 簡稱PMT)所組成的偵檢器，為 12 對 12 平行相向的架構，其中晶體陣列將正子事件的  $\gamma$  射線轉換成可見光，再由 PMT 把光訊號收集並且轉換成電流訊號，以便處理。

第二部分為類比前端電路，主要包含讀出(readout)電路、前置放大器(pre-amplifier)、權重運算器、積分器、加法器以及 ADC 等類比電路；讀出電路把第一部分輸出的電流訊號轉成代表能量資訊的電壓訊號，並且產生表示事件時間資訊的脈衝觸發訊號，還有晶體陣列座標的位置資訊，分別是 X 方向以及 Y 方向，接著藉由前端放大器將此訊號提升，再送入權重運算器、積分器、加法器進行類比訊號處理，最後由 ADC 將預處理後的類比訊號轉換成數位訊號送入下一級。

第三部分為數位訊號處理(Digital Signal Processing, 簡稱 DSP)模組，負責處理正子事件的數位資訊以及判定所偵測到的事件是否為總符合事件(total coincidence event)的符合事件單元。

第四部分為資料傳輸介面，用來做數位資料的擷取。而第五部分，伺服器端以軟體將代表正子事件的特性資料做分析並加以重建，最後在螢幕上呈現出可解讀的圖像。

本研究的原型架構將以正子乳房專用造影儀為基礎，建構出 1 對 1 的偵檢器系統架構，第一部份以及第二部份的硬體將以脈衝產生器來取代，而第三部份正子事件的數位資訊處理工作，重點將放在時間資訊的取得，該模組主要包含同步時脈單元、系統時間單元、事件擷取模組、數位 CFD 模組、數位內插模組、觸發信號模組以及符合事件計算模組電路，最後整合第四部份的資料傳輸介面及第五部份的伺服電腦端，實現數位資料的擷取與分析。

## 五、論文整體架構

本論文總共五個章節，第一章首先陳述本研究計畫的動機與目的，再介紹 PET 的基礎原理以及相關的系統架構。第二章探討既有的相關研究，還有本研究所使用的硬體原型架構。第三章說明本數位系統整體架構，再深入細說各個模組的設計與實現的過程，並針對使用不同演算核心的電路加以做比較。第四章將呈現子模組電路、系統整合的驗證成果，以及資料後處理的分析討論。第五章為結論與相關建議。

## 貳、研究方法與過程

### 一、文獻探討

隨著電路技術的演進，訊號處理的架構已漸漸演變成數位電路取代類比電路，以目前 PET 的領域為例，為了達到提升空間解析度的需求，則需要更多的訊號處理通道，然而以傳統的類比模組電路來因應隨時在變化的系統規格，需要相當多的修改費用，而且組裝出來的體積也因功能擴充而變得十分龐大，因此近年來國外研發團隊已經將趨勢轉向全數位式的電路架構做發展，以降低成本和提升效率[1][2][3][4]。

而影響 PET 的影像輸出品質，其中包括許多因素，像是靈敏度、空間及時間解析度、事件處理率。其中，準確的正子時間資訊，對於 PET 進行符合事件的計算尤其重要，傳統的方法常採用類比式的 CFD 串接 TDC(Time-to-Digital Converter)做為時間標記的產生[5]，然而已有文獻提出數位式 CFD 擁有比類比式 CFD 更佳的性能表現[6]，於是一部分團隊紛紛投向數位式 CFD 的懷抱[7][8]，而在時間最佳化的技術上則引進數位訊號處理的概念或數值理論方法[9][10]，如廣泛使用的線性內插或者是強調準確的擬合演算[6] [8]

[11][12]，甚至發展出類神經網路(Artificial Neural Network)、多次項內插等新穎的研究方法[13][14]。

正如許多文獻所呈現的實驗結果，對於事件時間產生的效率及準確性，將取決於所使用的演算技術，本研究將以此為參考依據，來設計適合本系統架構的方法。

## 二、硬體原型架構

硬體原型的設計包含脈衝產生器、數位訊號處理發展板、數位 I/O 板以及個人電腦端等四個部份，如圖 2.1 所示。

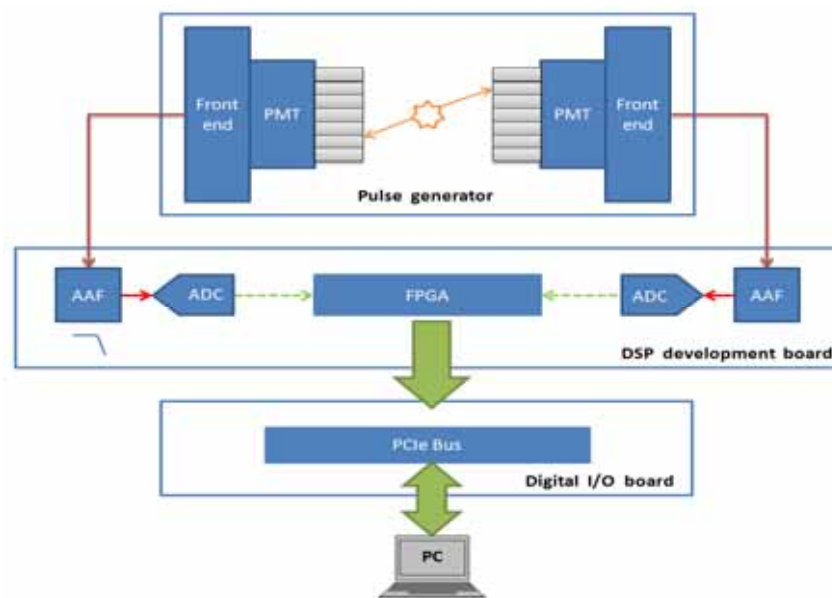


圖 2.1 硬體架構示意圖

第一部份，脈衝產生器為此原型的訊號來源，與原本的 PET 系統架構相比，脈衝產生器取代了放射源、晶體陣列、光電倍增管以

及使用 ASIC 製作的類比前端電路。

第二部份，數位訊號處理板用來連接脈衝產生器的輸出，而在訊號處理的路徑上包含三種裝置電路：抗失真濾波器(Anti-Aliasing Filter)、類比數位轉換器(ADC)以及 FPGA 晶片。

在類比訊號進入 ADC 之前，必須做初步的濾波處理。根據奈奎斯特取樣定理(Nyquist sampling theorem)所述，取樣頻率至少大於欲取樣信號之最大頻率的兩倍，因此，通常在 ADC 前端會增加低通濾波器，在取樣前適當地限制輸入訊號的頻率，將訊號品質最佳化，以防止高頻雜訊造成訊號失真，而此低通濾波器即為一抗失真濾波器。然後，經由抗失真濾波器調節後的訊號，再送入 ADC 做數位取樣，將原本的類比訊號轉換成數位訊號的數位值。

數位化後的訊號，則透過數位訊號處理板的 FPGA 來負責主要的計算，其中包括兩部分：訊號擷取系統以及符合事件計算系統，而本論文的研究重心即是完成訊號擷取系統的設計與實現，來因應 PET 實際應用上的快速脈衝訊號。

第三部份，經 FPGA 處理完畢所得到代表正子事件的屬性資料，藉由支援 PCIe(PCI-express)匯流排的高速數位 I/O 介面卡進行資料傳輸。



最後，採用 LabVIEW 圖控軟體所開發的資料擷取程式[15]，加以設定數位 I/O 介面卡的組態，並且將資料存取在個人電腦端的硬碟裡，以提供影像重建所需的資訊，以下各小節將深入介紹硬體原型中各個主要設備的功能與特性。

### (一) 脈衝產生器

在訊號源的設定上，如圖 2.2 所示，主要是為了建構 PET 系統的正子事件，正子事件波形實際上是從同位素放射源衰變產生的  $\gamma$  射線，經過偵檢器模組以及類比前端電路等眾多裝置所得來的類比訊號，為一電壓型態。而在原型的設計中，使用 Berkeley Nucleonics Corporation 的 BL-2 脈衝產生器來簡化複雜的配置，以加速本研究在開發設計時的驗證、測試與除錯。



圖 2.2 BL-2 脈衝產生器

BL-2 脈衝產生器的特性在於它提供了相當寬的調變範圍，其波形上升及下降時間、循環頻率、振幅以及脈衝寬度等參數可供控制，

而且具有波形品質穩定和快速的產生速度等特性，它的高彈性及可靠性十分適用於本研究的原型。

## (二) 數位訊號處理發展板

在數位訊號處理系統的設計上，選用 Xilinx Incorporation 的 XtremeDSP 發展板來輔助實現，如圖2.3所示，板子上具有兩個ADC通道，各自由截止頻率為58MHz的抗失真濾波器及14位元ADC晶片所組成，負責訊號源的偵測與擷取；而訊號的核心計算由一主要FPGA (XC4VSX35-10FF668) 搭配一管理時脈的FPGA(XC2V80-4CS144)來共同運作，FPGA和ADC的工作時脈都源自於105MHz的晶體震盪器。發展板亦具有40餘個數位I/O接腳，提供本研究在電路設計上的應用。



圖 2.3 XtremeDSP 發展板

## (三) 數位 I/O 介面卡

最後階段的資料傳輸，使用 National Instruments Corporation 的 PCIe-6537 數位 I/O 介面卡，如圖 2.4 所示，它提供了通用型的 32 位

元平行傳輸 I/O，並且支援高速 PCIe 介面，最快可至 50MHz 時脈，高達 200MB/s 的連續數據吞吐量，由於它的高頻寬及高速率的特性，滿足了本數位系統龐大的資料流，故選擇該介面卡整合至數位訊號處理板與電腦端之間的傳輸介面。



圖 2.4 PCIe-6537 數位 I/O 介面卡

### 三、系統概述

在，探討訊號擷取系統的運作之前先來檢視各個子模組電路的分布情形，如圖 3.1 所示，整體系統的架構包含：同步時脈單元、系統時間單元、事件擷取模組、數位 CFD 模組、事件位置與能量計算模組、觸發信號產生模組以及符合事件計算模組。

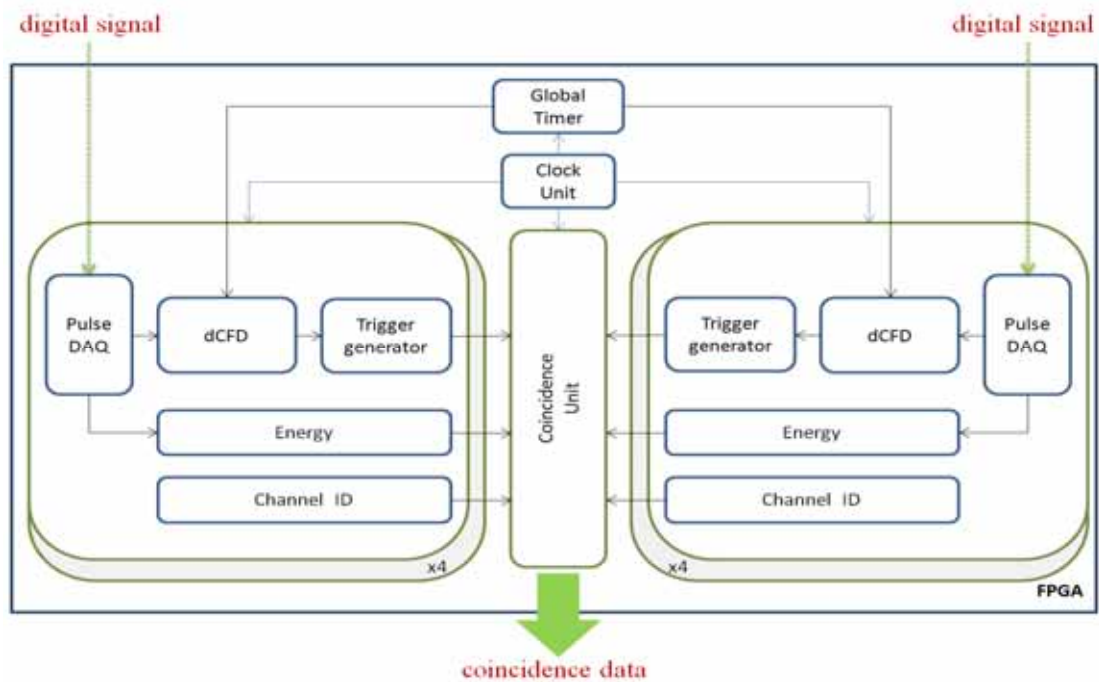


圖 3.1 系統架構圖

關於資料處理的核心部份，首先以事件擷取模組做 ADC 資料接收的連結動作，之後擷取到的事件能量資料送入數位 CFD 模組執行時間演算，並由觸發信號產生模組獲得表示時間資訊的脈衝信號，該訊號擷取系統將各自實現在每一個 ADC 通道上。最後，各個 ADC 通道所得的位置、能量和時間將共同透過符合事件計算模組來處理，符合事件的資料包含配對事件的位置與能量，再由資料輸出處理電路完成非同步的資料傳輸工作，使終端的數位 I/O 板與 FPGA 介面做整合。

除此之外，為了確保不同通道之間的時訊同步，就需要同步時脈單元來分配給系統內所有同步電路的工作時脈；系統時間單元則

負責提供各個時間模組共同的參考時間，以下各小節將針對各個子模組深入做說明。

#### 四、研究相關原理

##### (一) CFD 原理

CFD 為目前廣泛使用的時間萃取技術，而其數位形式如方程式 (3.1) 所示，可視為一延遲項與一衰減項的相減運算，其中 D 為延遲量(delay)，CF 為定比例(constant fraction)。CFD 所產生的結果是一個雙極性(bipolar)的波形，如圖 3.2 所示，它與橫軸會交於一零交越點(zero crossing point)，該點代表著事件時間資訊，而且它不受振幅大小影響而偏移，故本系統將應用 FPGA 來完成數位 CFD 的設計，以因應 PET 系統常常接收到振幅不一的脈衝事件之問題。

$$\text{CFD}[k] = \text{pulse}[k - D] - \text{CF} * \text{pulse}[k] \quad k = 0, 1, 2, \dots,$$

(3.1)

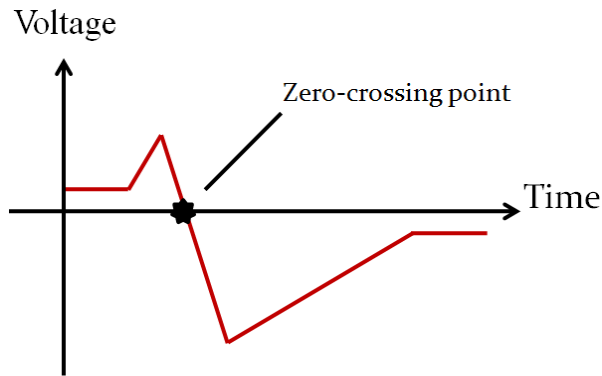


圖 3.2 CFD 波形示意圖

## (二) 線性擬合

於數位系統裡，時間精度代表著時間準確性，它主要是由取樣頻率的大小而定，硬體的取樣頻率愈高，則時間精度愈好，取樣得到的波形愈完整，取樣頻率愈低則反之。除了提高取樣頻率之外，還可以使用一些數值理論方法來突破目前硬體的規格限制，最普遍的方法是使用內插技術，其中又以線性擬合最為簡單可行，如圖 3.3 所示，橫軸為時間，縱軸為振幅能量，已知 A、B 兩個取樣資料的能量值，假設之間為線性變化，則可用方程式(3.2)來表示直線上任一點的時間資料，本系統的數位內插模組即是基於該原理來進行設計。

$$T' = T_1 + \frac{E' - E_1}{E_2 - E_1} (T_2 - T_1) ,$$

(3.2)

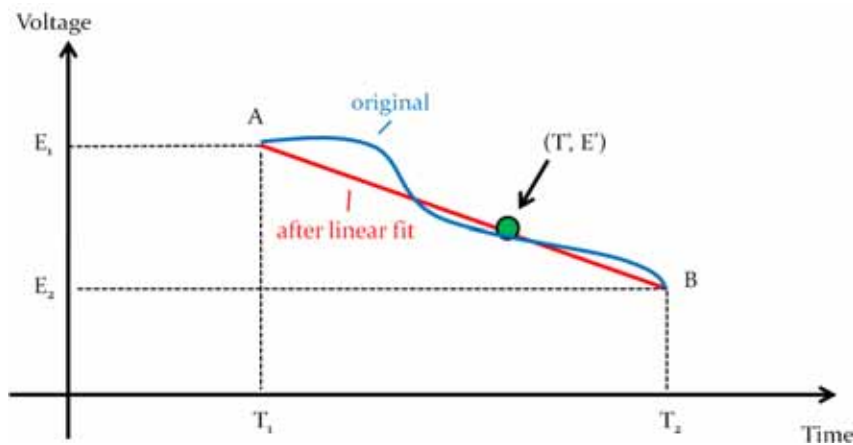


圖 3.3 線性擬合示意圖

## 五、模組設計與說明

### (一) 同步時脈單元

時脈是一個週期性訊號，用做控制同步電路的工作頻率。數位訊號處理發展板提供了各種時脈工作模式，例如各自使用獨立的時脈源或是共用相同的時脈源，以控制不同的硬體設備進行取樣或產生訊號；在本研究中，時脈源自於數位訊號處理板上的晶體震盪器，它的震盪頻率為 105MHz。在硬體時脈的設定方面，從晶體震盪器的時脈訊號進入管理時脈的 FPGA 之後，將經由相同長度的實體路徑，適當地分配至操作類比數位轉換的兩個 ADC 以及主要計算的

FPGA，以完成不同硬體之間的同步，如圖 3.4 所示；另一方面，在 FPGA 內部的時脈通道有限，因此若需要對多個同步數位電路進行時脈控制以及同步化處理，則要有一個共同的同步時脈單元來提供各自的工作時脈。

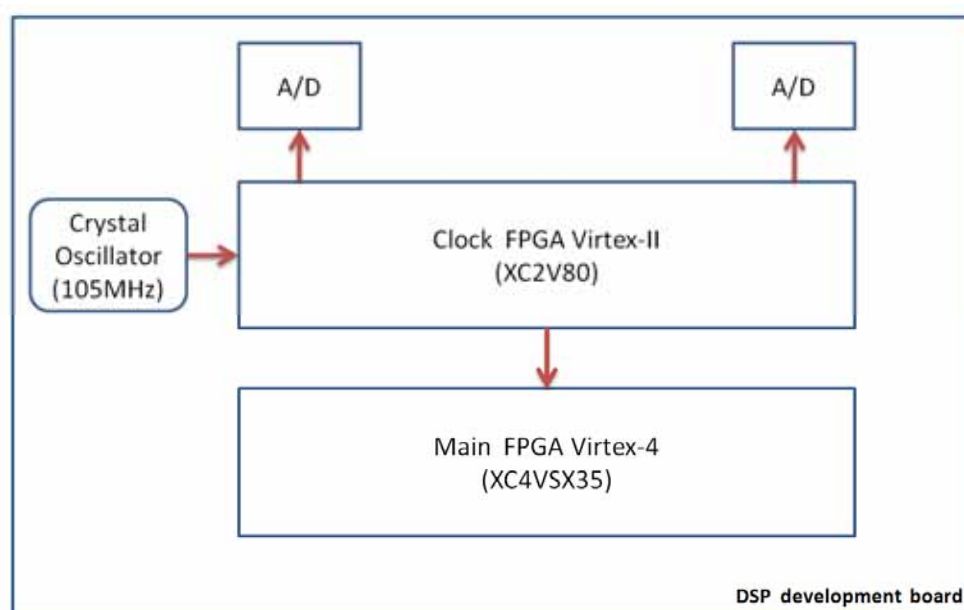


圖 3.4 硬體時脈配置示意圖

同步時脈單元應用了內建的 DCM(Digital Clock Manager) 來加速本研究的電路設計，它是 Xilinx Inc.所提供的 IP(Intellectual Property)模組，含 DLL(Delay Locked Loop)、DFS(Digital Frequency Synthesizer)以及 DPS(Digital Phase Shifter)等功能。

在進行系統時脈分配時，由於訊號線延遲不一，進而造成各個電路資料取樣時的錯誤，此現象稱作 Clock Skew，為了消除這個問題，如圖 3.5 所示，於 DCM 輸入端額外加入了全域延遲緩衝器(Input



Global Buffer，簡稱 IBUFG)，輸出端增加延遲緩衝器(Output Buffer，簡稱 OBUF)來達到延遲一致，以維持時脈的品質。另外，因為各個數位電路可能需要不同頻率的工作時脈，所以利用 DFS 技術將時脈源進行倍頻，即時脈頻率乘以整數倍，或是除頻，即時脈頻率除以整數倍，亦可將時脈頻率做分數倍的輸出，以合成出所需的工作頻率。而 DPS 則用來控制時脈源的相位延遲 90 度、180 度及 270 度的工作週期，亦可進一步於 DFS 處理後再做相位延遲，以產生出更精細的相對時脈。

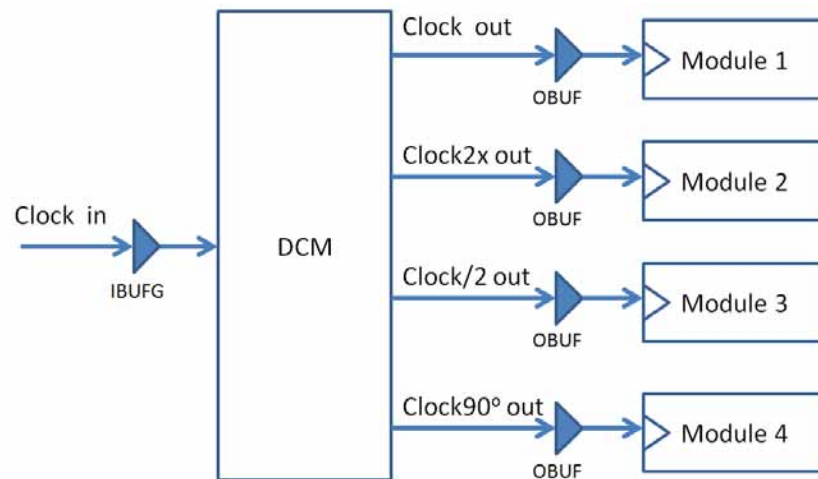


圖 3.5 同步時脈單元的功能示意圖

## (二) 系統時間單元

不同通道的時間模組必須要有共同的參考時間點，因為假使不同模組採用自己的參考時間點，可能會因為所接收到的脈衝振幅大

小不一，進而影響參考時間點也有所不同，舉例來說，兩個時間模組各自以準位觸發做為參考時間，如圖 3.6 所示，振幅較大的脈衝會較快觸發，其參考時間  $t_1$  亦較早，而振幅較小的脈衝需要較多的樣本才能超過能量準位門檻，它的參考時間  $t_2$  就相對地來得晚，因此，儘管事件同時發生，其產生的粗略時間標記卻不相同。

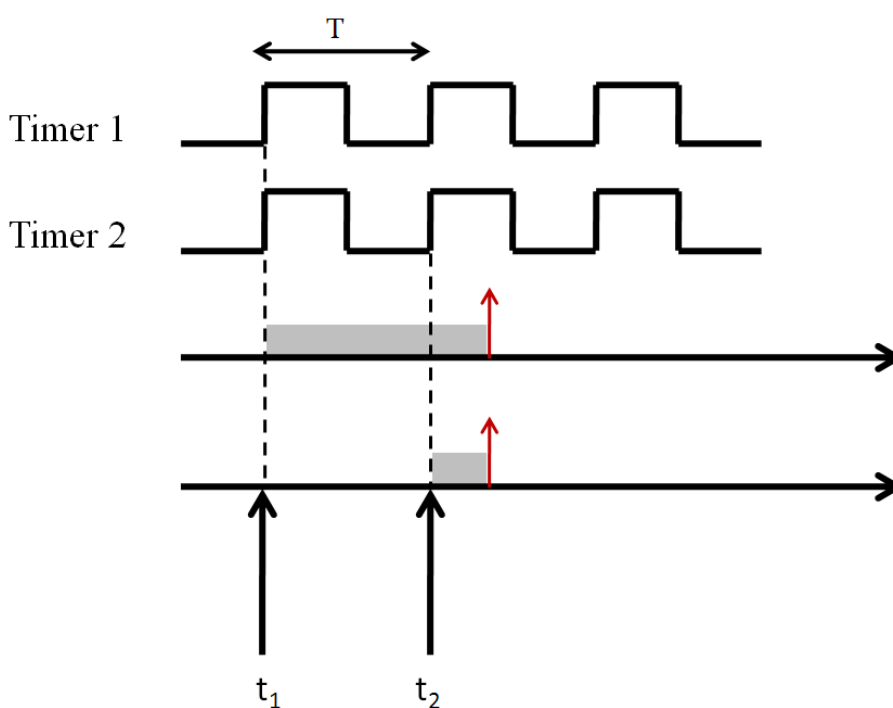


圖 3.6 準位參考時間示意圖

為了排除這個問題，在此設計系統時間單元，它是由系統時脈驅動而計數，所有的時間模組將共用單一系統時間做為參考標準，如圖 3.7 所示，以確保時間標記能維持一致性。

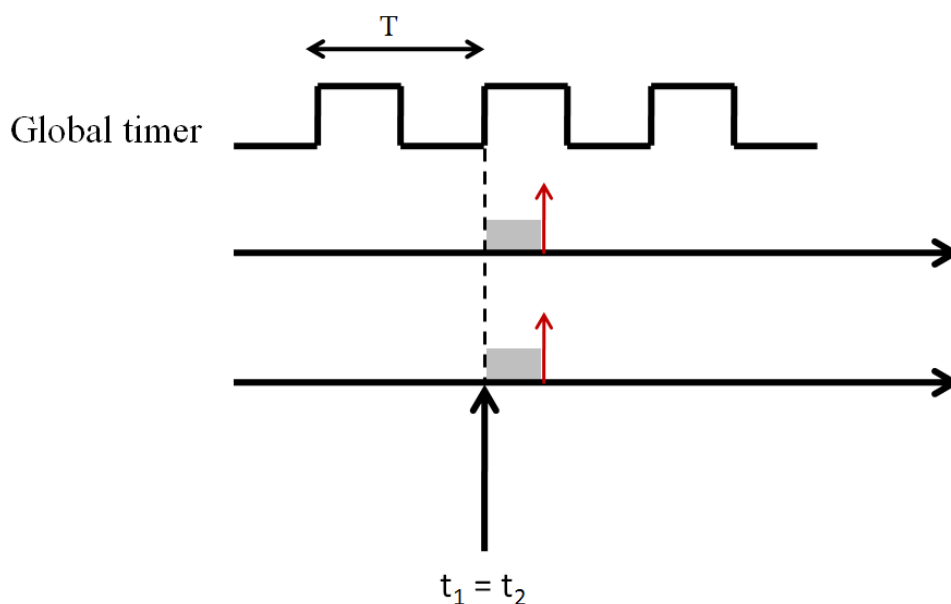


圖 3.7 系統參考時間示意圖

### (三) 事件擷取模組

事件擷取模組功能具有四個主要功能：ADC 與 FPGA 介面做滿足時序的連結動作、事件的動態偵測、擷取事件資料流以及事件資料的回復校準。

首先，為了滿足 ADC 資料的輸出與 FPGA 介面的時序條件，進行取樣的時脈可以是延遲後的 FPGA 工作時脈或者是 ADC 提供的資料準備(DRY)訊號[16]，本模組使用後者的方法。做法詳如圖 3.8 所示，採用兩級不同時脈驅動的架構，當 DRY 訊號改變狀態(low→high)的瞬間，隨即觸發第一級暫存器進行取樣，該電路行為稱之為正緣觸發(positive edge triggered)，本系統將一律採用正緣觸發來控制所有

的同步電路。而下一級之後的電路則皆使用 FPGA 內部時脈，以保證 ADC 資料能不失真地傳輸至 FPGA。

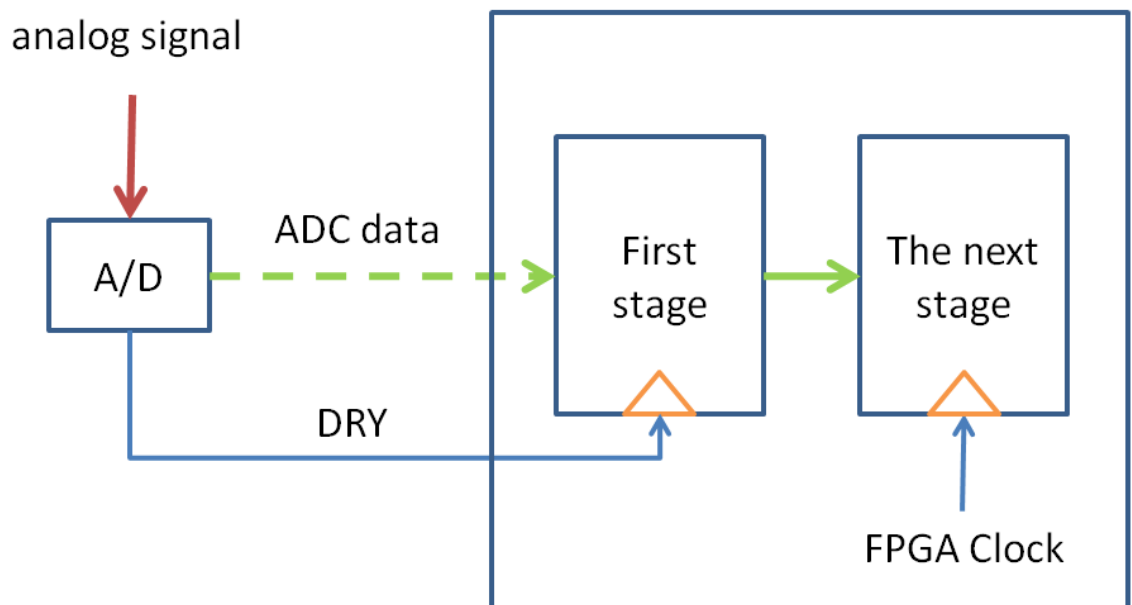


圖 3.8 ADC 與 FPGA 同步工作示意圖

FPGA 正確接收到數位資料後，其原始資料是無論有無發生事件皆全盤取樣進來，然而，事件才是對萃取能量、位置及時間資訊有用的資料，後級電路是直到事件發生，始擷取並進一步計算該事件的訊號流，沒有事件時，後級電路則是保持閒置的狀態。

在此採取動態偵測的方式來分離背景與事件，意即從連續變化的資料流中，快速取得可能為事件的波形資料。一方面，以移位暫存器實現資料的即時存取，每隔一個工作時脈就移動暫存器內部的資料至下級暫存器，做持續地更新；另一方面，即時地運算移位暫存

器局部的資料總和(即能量積分)，如圖 3.9 所示，包含移位暫存器的最前端，稱為 pre-data 暫存區；還有移位暫存器的尾端，稱為 baseline 暫存區，由於環境因素會導致背景能量偏移(offset)的現象，baseline 暫存區的目的之一也是為了計算此偏移量。這兩個暫存區總和之間的差是否超過準位門檻(level threshold)，決定了事件與非事件的區別。

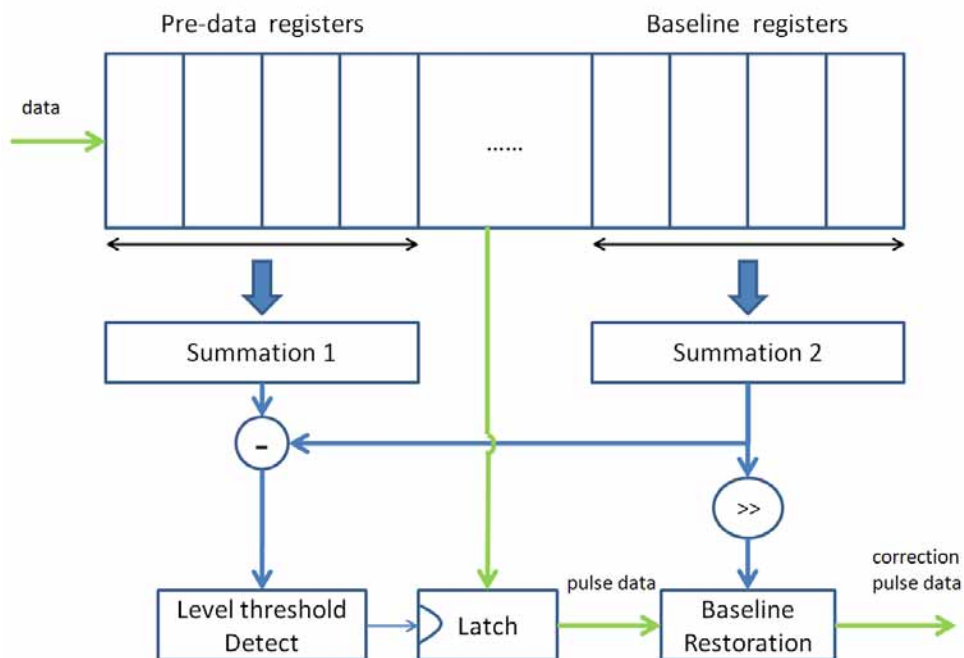


圖 3.9 事件擷取模組工作示意圖

當沒有事件發生，pre-data 暫存區與 baseline 暫存區各自的總和差並未超過準位門檻，資料將持續送入移位暫存器的尾端而捨棄。而當事件發生，pre-data 暫存區的總和逐漸增大，直至大過 baseline 暫存區預設的準位門檻，即開始進行擷取適當長度的事件資料。

接著，事件資料將經過 baseline 校準單元，其中，本模組的參考 baseline 值為了達到降低雜訊的效果，是源自於 baseline 暫存區的平均值，而後完成了 baseline 的減法運算後，再輸出正確的能量值，以及用來進行交握(handshaking) 的致能(enable)訊號，以要求後級電路準備接受來自於本模組的資料。

#### (四) 數位 CFD 模組

數位 CFD 模組包含 ROI(Region of Interest)、CFD 訊號產生、零交越點搜尋等三個主要功能。

前級事件擷取模組所得來的完整事件能量資訊，實際上僅有波形的前沿(leading edge)對時間資料的萃取有用，故使用 ROI 電路選擇性地將事件的局部資訊送入數位 CFD，以減少後級電路不必要的運算。

ROI 單元如圖 3.10 所示，內部使用一個計數器來顯示目前狀態，並控制拴鎖電路的輸出狀態，拴鎖電路的真值表如表 3.1 所示，其中 CE(Clock Enable)表示致能訊號，D 為輸入資料，CLK 為同步時脈，Q 為輸出資料。當沒有事件資料輸入，計數器不做變動，而拴鎖電路(CE=0)也不會有任何輸出變動。當偵測到代表資料傳輸的致能訊號，計數器即開始記錄目前輸入資料為波形的第幾筆，且控制拴鎖

電路(CE=1)持續輸出資料，直到計數器超過定義的 ROI 長度，則控制控鎖電路(CE=0)停止輸出之後的波形資料，當最後一筆資料輸入完成，計數器的數值將重置為零。附帶一提，前級事件擷取模組在擷取固定長度的波形時，也是應用同樣的工作原理。

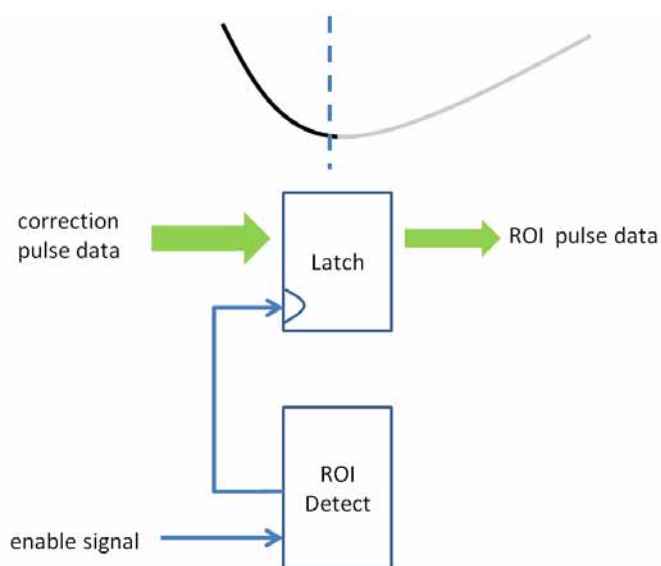


圖 3.10 ROI 單元工作示意圖

表 3.1 控鎖電路真值表

Inputs			Output
CE	D	CLK	Q
0	X	X	No change
1	0	↑	0
1	1	↑	1

完成了事件資料的過濾後，剩下的資料將進入數位 CFD 訊號產

生單元，該數位 CFD 演算原理如同類比 CFD，如圖 3.11 所示，輸入的波形將經過兩個資料路徑，分別做延遲以及乘上定比例的處理，之後執行減法運算即得到 CFD 的波形。

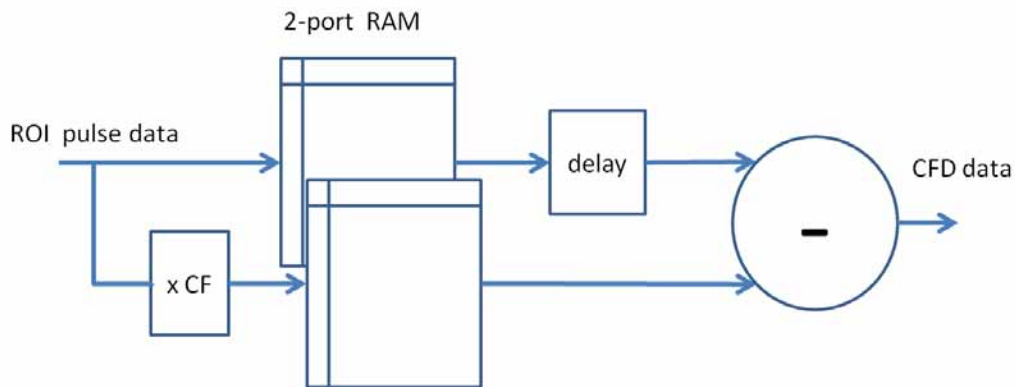


圖 3.11 數位 CFD 訊號產生單元示意圖

在數位 CFD 的電路設計上，以 2-port 的記憶體做為資料路徑的分離，此記憶體為同步先寫後讀的時序模式，第一條路徑以暫存器的層級數目來控制資料延遲的時間；在另一條路徑上，資料與定比例做相乘運算，該定比例為介於 0 和 1 之間的數值，因為考慮了電路的複雜度，在此不直接實現浮點數(floating point)乘法運算，而是使用簡易的右移位運算來替代，例如乘上 0.25 的定比例，改以右移位 2 位元來完成。最後兩筆結果送入減法器，以延遲的資料減去乘上定比例的資料。

事件波形調變為 CFD 訊號後，可進一步從零交越點(以下簡稱零點)決定粗略的事件時間，而零點的位置會位於一個正值資料樣本與



一個負值資料樣本之間的單元區域，於本文中稱之為「零交越區」；在此設計零點搜尋單元來執行零交越區的搜尋，並針對零交越區兩端的資料樣本進行零點檢測。

零點搜尋單元如圖 3.12 所示，主要包含異號偵測單元、拴鎖電路、資料緩衝區、斜率比較單元、零點檢測單元以及外部的系統時間單元。

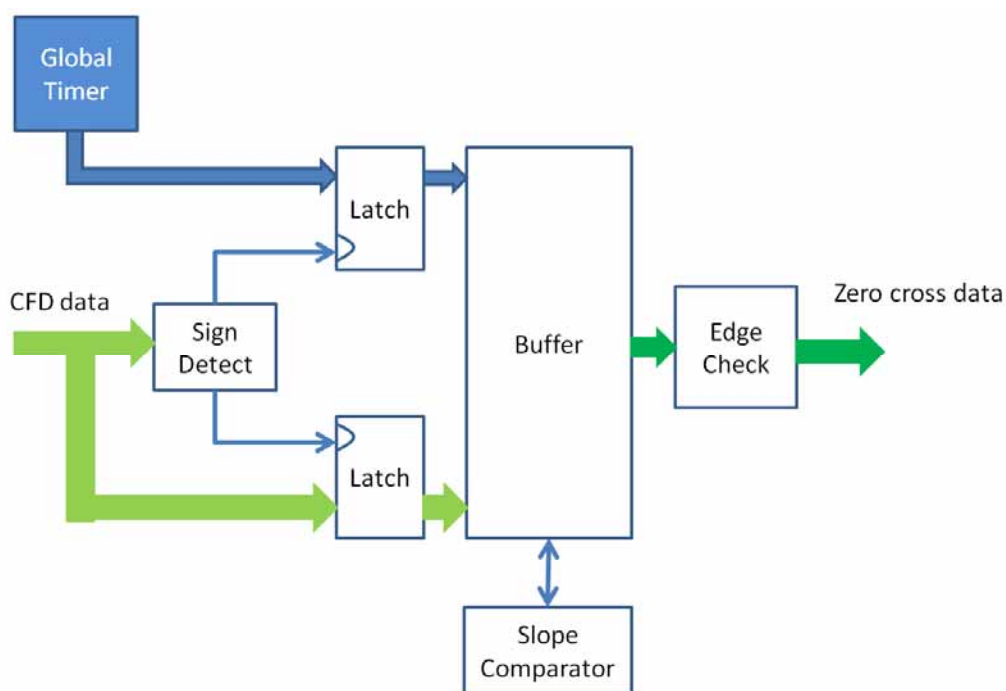


圖 3.12 零點搜尋單元示意圖

在本系統中，採用 2 的補數格式來表示資料型態，而其中代表正負性質的符號為資料的最高有效位元(Most Significant Bit，簡稱 MSB)，以 14 位元的資料為例，則它的 MSB 即第 14 位元數。MSB 若為 1 則表示負數，MSB 若為 0 則表示正數(包含零值)。根據零點

鄰近資料的特性，零點搜尋單元所要做的第一步即是比較每兩筆資料的正負，在此設計異號偵測單元，以求得零交越區。

異號偵測單元以 2 位元的移位暫存器連續擷取輸入資料的 MSB，當移位暫存器內部的值為 00 或 11，則表示同號，相反地，若內部數值是 10 或 01，即代表異號。而根據本論文所要產生的 CFD 特性，該零點會存在於正訊號劇烈轉變成負訊號的波段之間，故當輸入資料的 MSB 為 01 時，才代表偵測到可能的零交越區。

經由異號偵測單元來判定資料是否滿足零點交越區的第一個條件，若異號條件不成立，則拴鎖電路不做該筆資料的輸出動作；若異號條件判定成立，則拴鎖電路進行輸出代表零交越區的兩筆樣本資料，並且由後級的緩衝區做儲存，於此同時，代表中央標準時間的時間資訊也會一起存入緩衝區。

但是，以上所描述的資料處理流程有一個前提，擷取進來的波形資料只能有一個零點，換句話說，它沒有辦法處理多個零點的情況。在本研究的環境中，是有可能因為受背景雜訊影響，而在正確零點之前的位置隨機產生造成誤判的零點，因此在此設計斜率比較單元來解決這個潛在的問題。

對於零點搜尋的準則重新加以定義：零點會存在於正訊號劇烈

轉變成負訊號的波段之間，意即零點鄰近的兩個能量資料，它們的斜率會比其他波段的斜率來得大，而且這斜率值必定為負。異號偵測單元送出候選資料的同時，斜率比較單元將計算它的斜率值，一併與能量和時間資訊暫存在緩衝區，同一事件的 CFD 波形又偵測到下一筆的異號資料時，也再重複一樣的工作流程，等到異號偵測單元處理完所有資料，將緩衝區的斜率值做比較並排序，斜率小的候選資料被淘汰，而斜率大的即為搜尋到的零點交越區，再送入零點檢測單元完成最後的處理工作。

將緩衝區中的資料送入零點檢測單元後，便檢查這兩筆資料之一是否恰為零點，若檢查出零點，則輸出零點旗標(zero flag)的控制訊號，告知後級的內插電路不做進一步的內插運算；若檢查無零點，則輸出這兩筆樣本資料，還有通知後級電路準備接收資料的致能(enable)訊號。

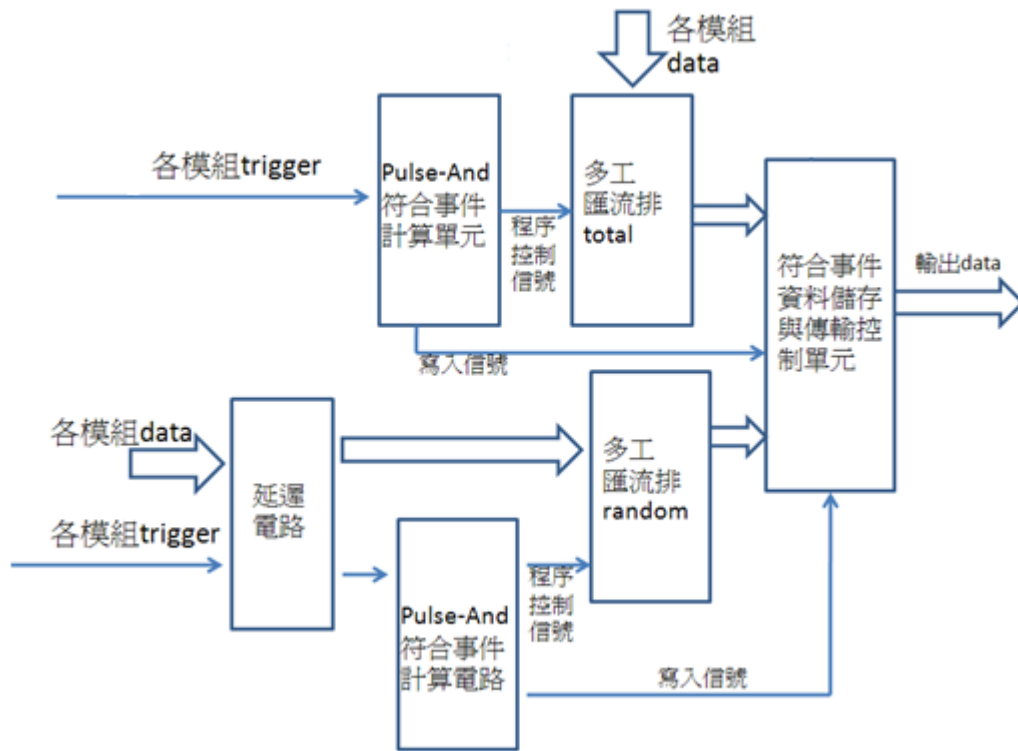
#### (五) 觸發信號產生模組

後級的符合事件計算系統為 pulse-and 電路架構，則本訊號擷取系統的輸出訊號必須做出對應的配合，才能滿足兩系統不同的資料型態，因此設計觸發信號產生模組，以產生 pulse-and 電路所需的微小 pulse 信號，pulse 寬度設定為 2 個時脈週期，以保證相差 1 單位

的時間標記也能判定為符合事件，而 pulse 信號的時序關係是根據數位 CFD 模組所演算獲得的時間標記來決定。

#### (六) 符合事件計算系統模組

符合事件計算系統上，我們使用 Pulse-And 架構來計算是否為符合事件，而當有兩個事件被所配對的偵檢器相對的模組偵測到且能量皆超過所要求的標準時，只要兩事件的時間資訊信號(Pulse trigger)差距在系統所定義的誤差值內，事件符合計算電路將會判定此兩事件為一組符合事件(coincidence event)，然而實際上此兩事件並不一定是真實符合事件(true coincidence event)，有可能是因為隨機背景雜訊被偵檢器所偵測到，這些非真實符合事件都被定義成隨機符合事件(accidental coincidence event)。所以在事件符合計算電路中將分為兩個部份來計算符合事件；一個為總符合事件(total coincidence event)計算單元，另一個為隨機符合事件(accidental coincidence event)計算單元。用來判定符合事件是否成立的時間資訊信號若為 Pulse trigger，我們稱其判定方式為 Pulse-And 架構。



Pulse-And 組合架構事件符合計算電路

總符合事件計算單元與隨機符合事件計算單元的電路架構完全相同，差別在於前級的事件資訊(時間資訊信號以及總能量信號)將先經過延遲電路再進入隨機 Pulse-And 符合事件計算單元中，而進入總 Pulse-And 符合事件計算單元的信號則不需經過延遲電路。

送入 Pulse-And 符合事件計算單元的信號，將分成時間資訊信號以及事件總能量信號兩個部份來進行信號處理，各模組的時間資訊信號會送入計算單元進行判定，而各模組的事件總能量信號則是送入多工匯流排中等待；首先，藉由檢查各模組的時間資訊信號來

判定是否有符合事件，若有符合事件成立，則馬上產生符合事件的模組配對信號至多工匯流排，將相對應兩個模組的事件總能量信號移至 FIFO 記憶體中儲存。

計算單元的信號處理流程如圖 3.4 及圖 3.5 所示；前級各模組的 Pulse trigger 皆將送入至判定邏輯中進行判定，當有任一個模組配對成立，同時會送出判定信號至時序控制電路來啟動程序控制，以及將各模組對信號送至編碼電路進行編碼；而時序控制電路將判斷其接收到的判定信號，距離前一次收到的判定信號是否相隔超過電路工作所需的 Dead-Time(Pulse-And 架構 Dead-Time 小於 20ns)。若超過則送出程序控制信號來偵測是否有誤判的發生，以及控制編碼電路送模組配對信號。並將位於多工匯流排入口處的事件總能量信號拴鎖於拴鎖器，完成之後接著送出重置信號將編碼電路重置，以便進行下一次的工作。反之，若低於電路工作所需的 Dead-Time，則不會送出程序控制信號。

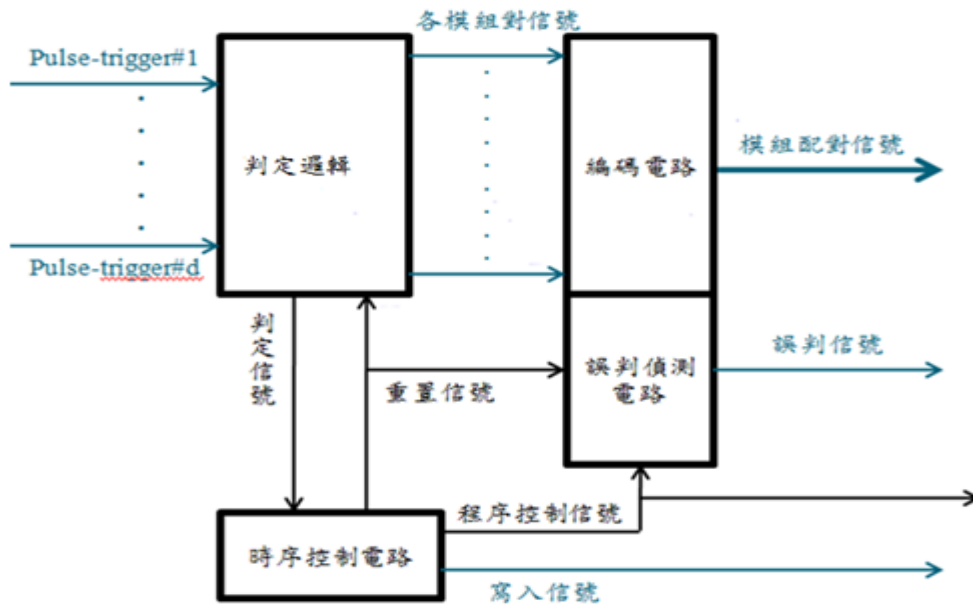


圖 3.4 Pulse-And 符合事件計算單元之信號處理流程圖

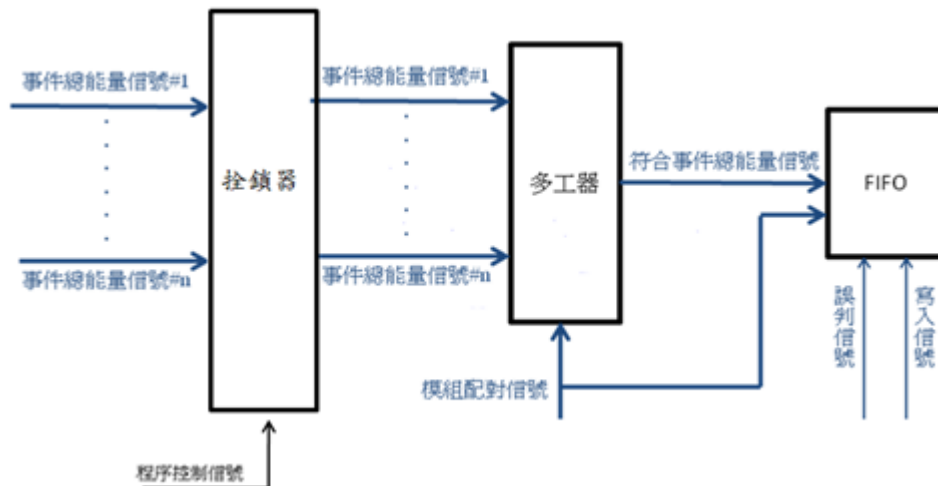
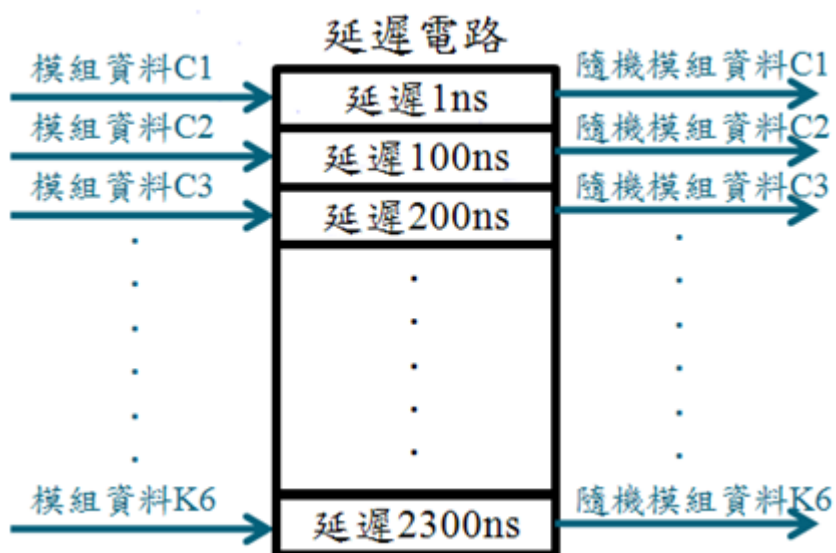


圖 3.5 選擇符合事件配對中相對應模組的總能量信號

計算單元所送出的模組配對信號將用來控制多工匯流排，選擇所判定之符合事件配對相對應的兩個模組，並將配對的兩個事件總能量信號與模組配對信號合併送至 FIFO 記憶體；而計算單元所送出的寫入信號，將配合誤判信號一同控制 FIFO 記憶體的寫入端。若誤

判信號成立時，當筆資料將不寫入 FIFO 記憶體而直接捨棄，若誤判信號不成立時，當筆資料將依據寫入信號寫入 FIFO 記憶體中。

設計隨機符合計算單元的目的，在於建構模擬偵檢器偵測到背景雜訊的隨機模型。輸入總 Pulse-And 符合事件計算單元的時間資訊信號以及事件總能量信號，將先經過延遲電路再進入隨 Pulse-And 符合事件計算單元中，進行隨機符合事件的計算，如事件編碼及選取總能量信號並將其存入 FIFO 等工作。延遲電路將各模組的資料(包含時間資訊信號與總能量信號)分別延遲 100ns 以上，造成每個模組間的時間差距都在 100ns，再送入計算單元中來判定隨機符合事件。

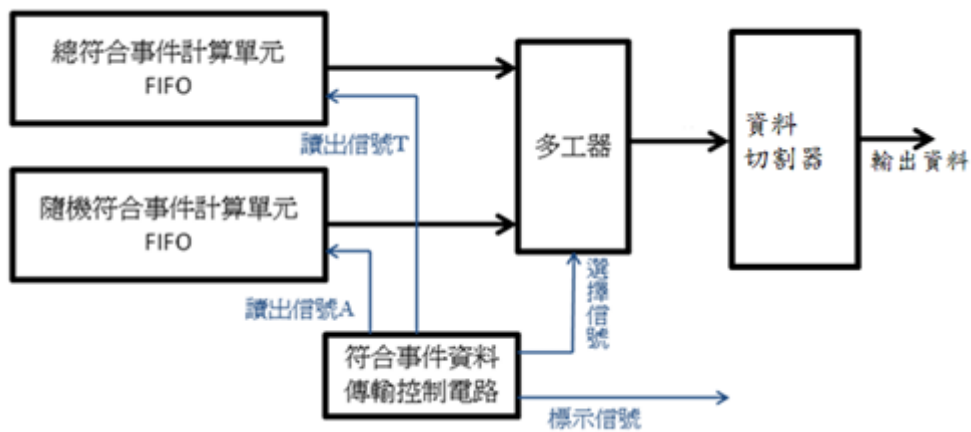


延遲電路示意圖



而由於後級資料傳輸介面僅提供一個傳輸管道(channel)，也就是兩個計算單元的 FIFO，在同一時間只有其中一個可以將資料讀出，再透過多工匯流排將讀出的資料移至傳輸管道。所以在此設計一個符合事件資料傳輸控制電路，將送出選擇信號來控制多工器。而用於儲存隨機符合事件資料的 FIFO 內，儲存的資料達 128 筆時，資料傳輸控制電路將連續送出 128 次讀出信號 A，將隨機 FIFO 內的資料讀出。反之，若用於儲存隨機事件資料的 FIFO 內，儲存的資料未達 128 筆。則當有資料存入用於儲存總符合事件資料 FIFO，資料傳輸控制電路將送出讀出信號 T 將總符合 FIFO 內的資料讀出。

另外，配合選擇信號控制的多工器將兩個計算單元 FIFO 的資料正確的移至資料切割器；由於後級的數位 I/O 資料擷取卡可接收的資料寬度為 32bits，而 FIFO 輸出的資料寬度為 64bits，所以必須透過資料切割器將 64bits 的資料切割成兩筆 32bits 的資料分兩次傳輸。傳輸控制電路也將送出一個標示信號，來表示傳輸管道上是否有資料在傳輸，以便於後級數位 I/O 資料擷取卡擷取事件符合計算電路的輸出資料。



符合事件資料傳輸控制示意圖

### (七) 測試訊號產生單元

在數位系統進行最後階段的測試時，通常會加入一組已知的測試訊號，也稱作測試圖樣(Test Pattern)，以建構出整個數位電路的模擬驗證環境，並藉由檢測系統輸出訊號的過程，來進行驗證及除錯的動作，如圖 3.22 所示。本論文的電路測試將以脈衝產生器的輸入訊號為主，測試訊號產生單元為輔，確保數位電路在各種測試條件下，皆能夠正確而無瑕疵地運作。

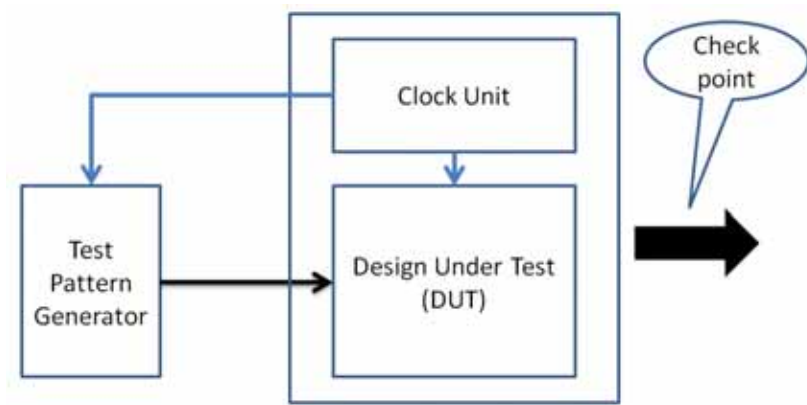


圖 3.22 測試訊號產生單元示意圖

## 參、主要發現與結論

### 一、時間模組電路之測試

本節將呈現時間模組應用於單一 ADC 通道的成果，它的重點測試有三個部分，分別為事件擷取模組、數位 CFD 模組以及數位內插模組，並針對其中負責核心演算的子單元電路來做功能檢驗與分析。

在硬體實測方面，如圖 4.1 所示，包含脈衝產生器、數位訊號處理板、下載線、邏輯分析儀等裝置。將脈衝產生器輸出的負脈衝訊號，以 Xilinx Inc. 原廠傳輸線連結到數位訊號處理板上的單一 ADC 端，建構出輸入訊號的環境。



圖 4.1 硬體實測配置圖

而在進行電路測試前，必須先對脈衝產生器做相關設置，其類比波形如圖 4.2 所示，因為光電倍增管之 dynode 端電流經過讀出電路所輸出的訊號為負變化，所以在此也設定脈衝產生器輸出為負脈

衝，而它的上升時間(rise time)設定為 30ns，以符合真實的正子事件訊號；振幅設定為-1.1V，以充分使用 ADC 的輸入範圍；頻率刻度調為 1MHz，但實際量測結果約為 1.8MHz。

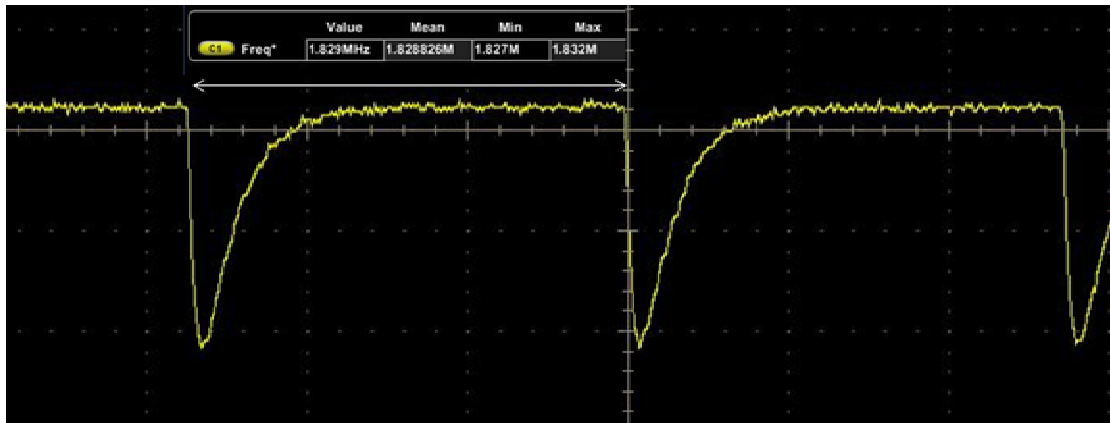


圖 4.2 脈衝產生器輸入波形圖

接著針對輸出訊號方面做架設，規劃訊號想要輸出的 I/O 腳位，以邏輯分析儀的彩色線依照其所屬編號做連接。最後，將下載線連接到處理板上的專用腳位，以 ISE iMPACT 軟體準備燒錄代表 FPGA 電路設定的 bit-stream。附註一提，下載線與邏輯分析儀的另一端都接上電腦的 USB 介面，以載入燒錄檔和擷取 FPGA 的輸出結果。

### (一) 事件擷取模組測試

本小節使用脈衝產生器做為訊號源，輸入至數位訊號處理板的單一 ADC 通道，來執行模組電路的驗證。測試結果可以分為三個檢查重點：第一，設計上的功能是否正確無誤；第二，資料時序是否穩

定；第三，各模組整合的相容性，以下所有測試皆依據這三點進行說明。

事件擷取模組的輸出結果包含兩筆訊號：與後級電路做資料交握的致能訊號，以及經過 baseline 減算校正的能量資料，如圖 4.3 所示，當輸入訊號的能量積分未達到門檻值，則致能訊號與能量資料致能訊號皆為 0，表示沒偵測到事件，不執行資料擷取行為；而當輸入訊號的能量積分超過門檻值，致能訊號為 1，並輸出代表完整事件的 15 筆能量資料，資料內容以 16 進制格式表示，如圖 4.4 所示，其資料波形以 2 補數格式表示，如圖 4.5 所示，其結果可以確定本電路功能正確。

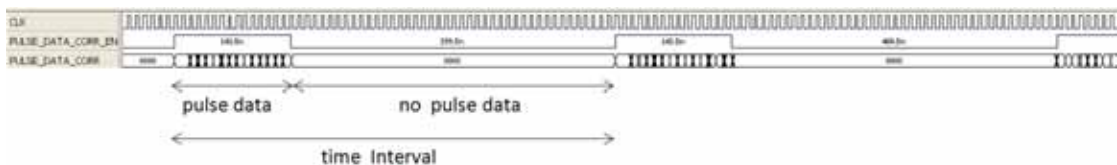


圖 4.3 事件擷取模組測試結果



圖 4.4 事件擷取模組測試結果資料

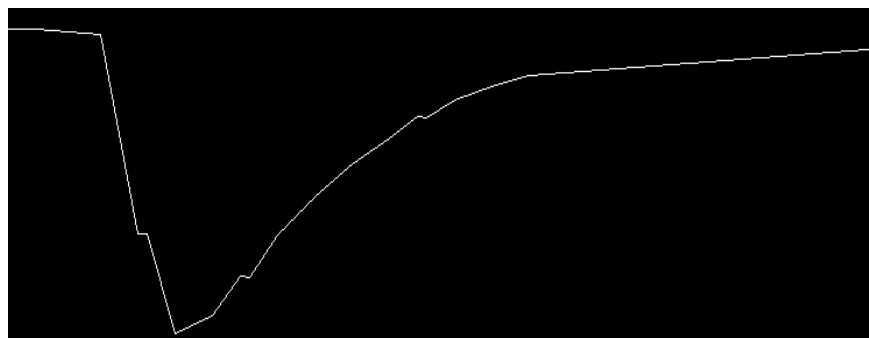


圖 4.5 事件擷取模組輸出波形

## (二) 數位 CFD 模組測試

本小節將對數位 CFD 模組分做三個階段測試：第一，ROI 單元與數位 CFD 單元測試；第二，零點搜尋單元的獨立測試；第三，數位 CFD 模組整合測試。

第一階段的測試，將數位 CFD 單元與前級的事件擷取模組、ROI 單元做連結，同樣由脈衝產生器來提供輸入訊號。它的輸出結果包含數位 CFD 單元輸入端的 ROI 資料，致能訊號、CFD 訊號，如圖 4.6 所示，ROI 單元正確地將原本 15 筆的完整事件能量資料，取出代表感興趣區域的前 9 筆能量資料，依序為 0006、0003、3E08、26A8、203E、22CC、2714、2B81、2F5E，並且輸出到數位 CFD 單元做處理。



圖 4.6 數位 CFD 單元測試結果

參照本論文 3.1 小節所述的 CFD 演算公式：

$$\text{CFD}[k] = \text{pulse}[k + D] - \text{CF} * \text{pulse}[k] \quad k = 0, 1, 2, \dots$$

其中 CF 設定為 1/4，D 設定為 1 單位時脈週期，所計算出的結果依

序為 0006、0081、045E、2E99、278B、2907、2C34 共 8 筆 CFD 資料，與電路輸出結果一致，其資料以 2 補數格式表示，如圖 4.7 所示，亦為正確的 CFD 波形。此外，致能訊號與 CFD 資料的時序上維持同步。

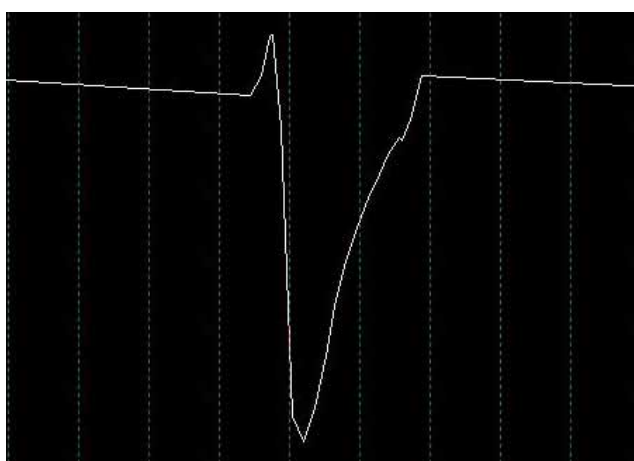


圖 4.7 數位 CFD 單元輸出波形

改變脈衝參數設定而對應輸出的 CFD 波形如圖 4.8 所示，其中編號 1 的波形圖為事件頻率 100kHz，編號 2 為頻率 1MHz，編號 3 為頻率 10MHz，編號 4 為振幅 0.4V，編號 5 為振幅 0.8V，編號 6 為振幅 1.1V，在即時的測試中皆可得到正確的波形輸出。

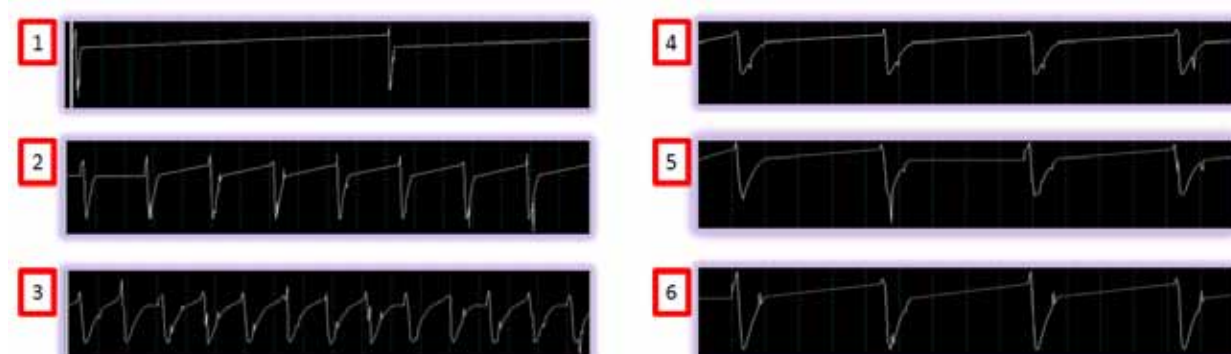




圖 4.8 數位 CFD 單元對應不同環境的輸出波形

第二階段針對零點搜尋單元做獨立的驗證，使用測試資料產生單元做為輸入訊號，其中測試資料的設定如圖 4.9 所示，每筆資料寬度皆為一個時脈週期，其中包含四組不同情況的 CFD 訊號，彼此又間隔一個時脈週期，第一組為源自於數位 CFD 單元的輸出，用做測試電路功能的參考波形，而它的零交越區位在第三筆資料 0339 與第四筆資料 3A86 之間；第二組將參考波形的第二筆資料修改為負值 FFFE，相當於在第一筆與第二筆之間加入誤判的零交越區，用來測試斜率比較機制；第三組將參考資料的第三筆資料修改為零值，即在零交越區的其中一側資料設為零點，主要目的是測試零點檢查功能；第四組是第二組與第三組的情況同時成立，以做整體的交叉測試。

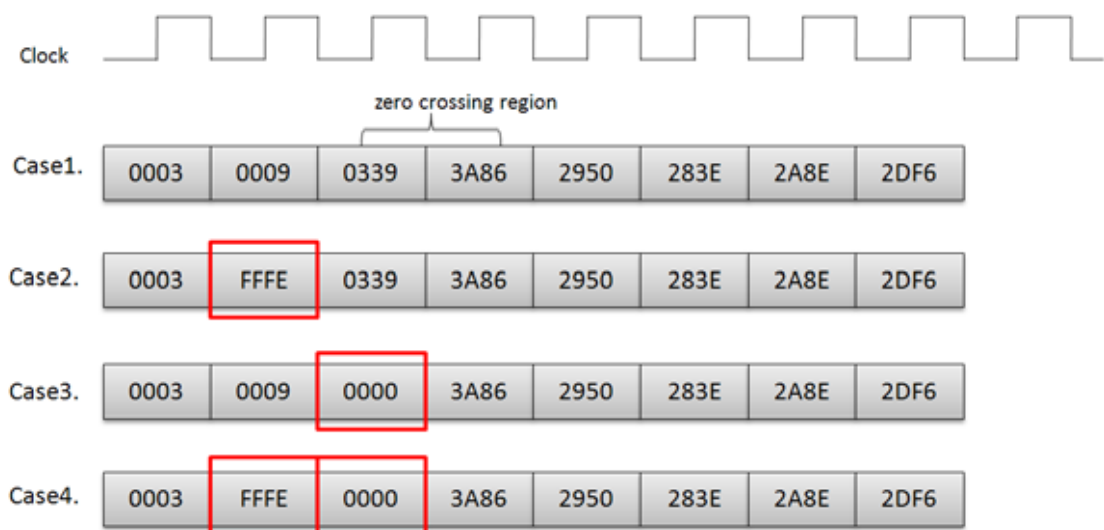


圖 4.9 零點搜尋單元的輸入測試資料

輸出結果包含致能訊號、正值能量資料、負值能量資料、零點旗標、粗略時間標記(本文稱之為 index time)以及系統時間，如圖 4.10 所示，第一組測試資料所對應的輸出為代表零交越區的正值能量資料 0339 和負值能量資料 3A86，以及時序同步的致能訊號，還有粗略時間標記 0，表示當時搜尋到該零交越區的系統時間；第二組測試資料所對應的輸出同樣也為正值能量資料 0339 和負值能量資料 3A86，以及時序同步的致能訊號，還有粗略時間標記 1，結果表示斜率比較機制確實有將誤判的零交越區捨棄掉，而輸出負變化率較大的正確資料；第三組測試資料對應的輸出則為零點旗標與粗略時間標記 2，結果證實零點檢查單元正確地運作；第四組測試資料所對應的輸出為零點旗標與粗略時間標記 3，結果表示斜率比較機制以及零點檢查單元彼此獨立，互不影響地正常工作。

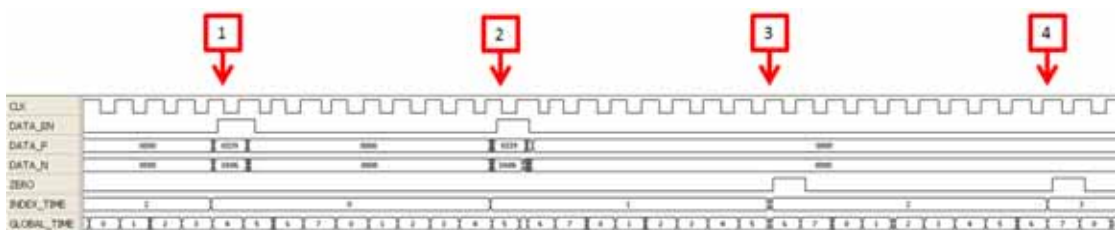


圖 4.10 零點搜尋單元測試結果

第三階段是將以上測試成功的事件擷取模組、ROI 單元、數位 CFD 單元以及零點搜尋單元合併做整合測試，輸入訊號使用脈衝產生器，局部結果如圖 4.11 所示，第一筆輸出資料內容 01FE、2C71

以及粗略時間標記 7，第二筆輸出資料內容 04A7、32AC 以及粗略時間標記 0，由結果得知資料與致能訊號正確輸出且時序同步，另外，零點旗標在即時的測試中總是維持 0，儘管如此，在 PET 應用上還是有可能在本模組中得到零點，故仍保有零點檢查單元電路，以防萬一。



圖 4.11 數位 CFD 模組整合測試結果

## 二、符合事件計算系統模組電路之測試

本節將針對 Pulse-And 架構符合事件計算電路做整體測試，測試包含總符合事件與隨機符合事件的輸出測試。

### (一) 總符合事件的輸出測試

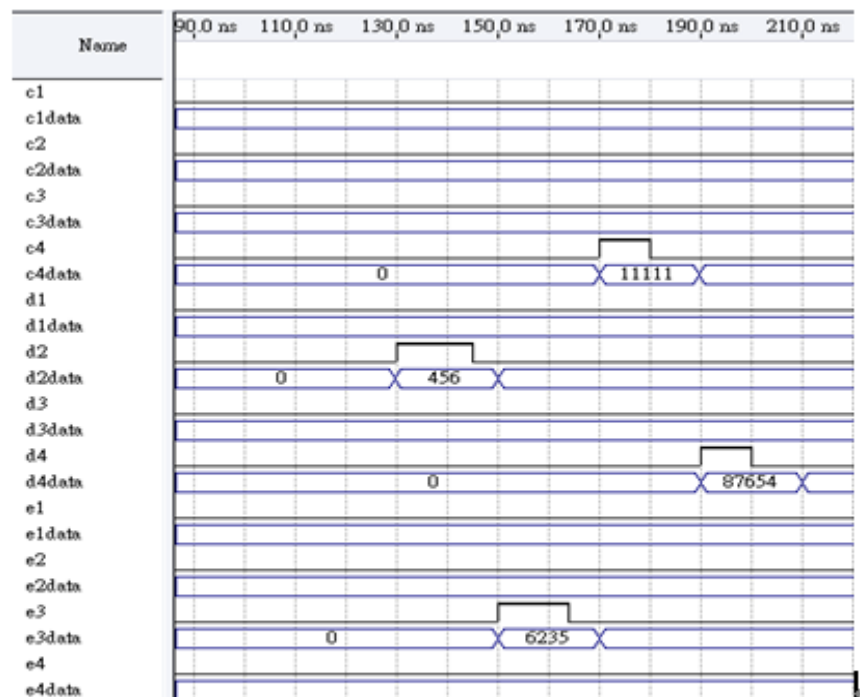


圖 4.30 前模組輸入之測試資料

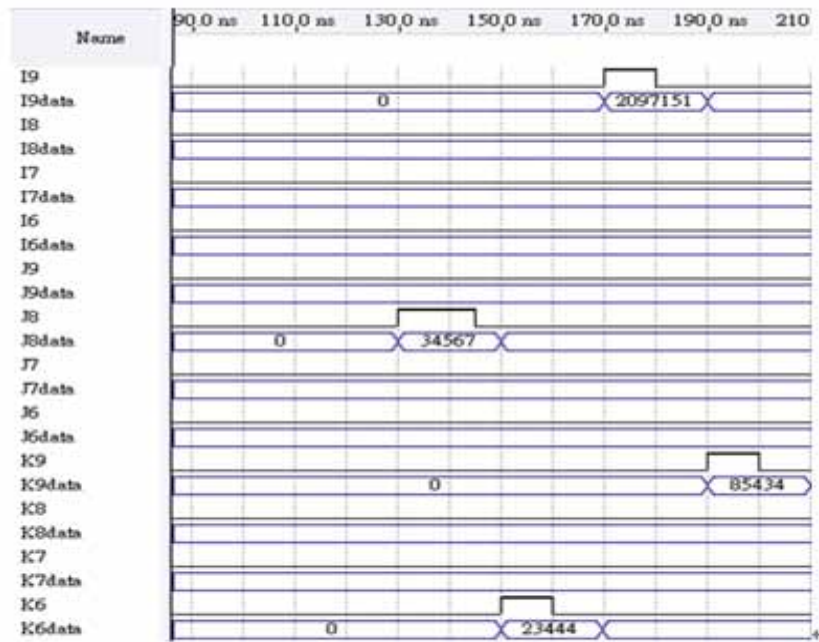
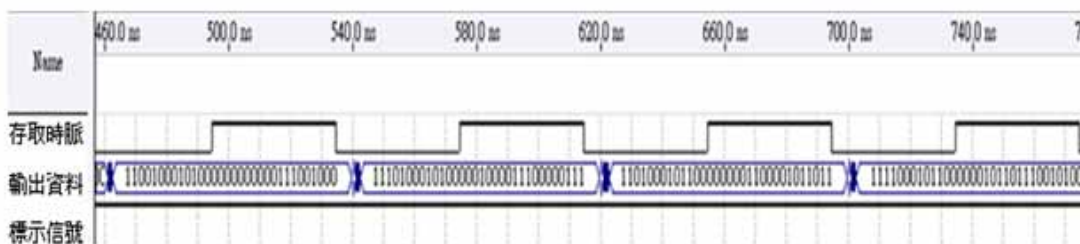
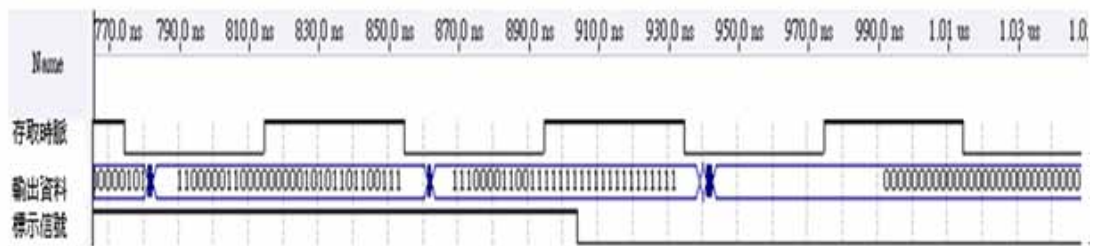


圖 4.31 後模組輸入之測試資料

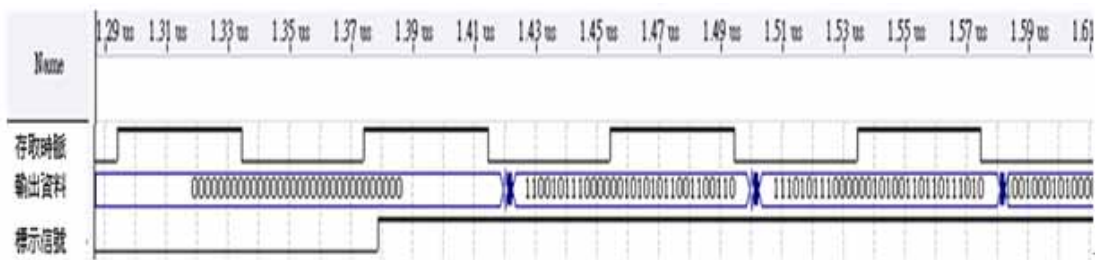
使用分組式 Pulse-And 架構的事件符合計算電路其 Dead-Time 為 20ns，因此在時間軸 130ns、150ns、170ns 及 190ns 處分別安排 DJ-28、EK-36、CI-49 及 DK-49 的符合事件以形成一週期 20ns 的測試事件流，並每隔 1  $\mu$ s 就重複輸入進行測試。



DJ-28 與 EK-36 符合事件資料輸出



CI-49 符合事件資料輸出



DK-49 符合事件資料輸出

由 DJ\_28 與 DK\_36 符合事件資料輸出第一筆輸出的 32bits 資料可知其第 31~29bits 為 110 代表此為總符合事件且屬於前模組的資料，第 28~21bits 為 01000101 帶表事件 DJ-28 的編碼，第 20~0bits 為 0000000000000111001000，其為 D2 模組資料 456 的二進位形式。第二筆輸出的 32bits 資料可知其第 31~29bits 為 111 代表此為總符合事件且屬於後模組的資料，第 28~21bits 為 01000101 帶表事件 DJ-28 的編碼，第 20~0bits 為 000001000011100000111，其為 J8 模組資料 34567 的二進位形式。第三筆輸出的 32bits 資料可知其第 31~29bits 為 110 代表此為總符合事件且屬於前模組的資料，第 28~21bits 為 10001011 帶表事件 EK-36 的編碼，第 20~0bits 為

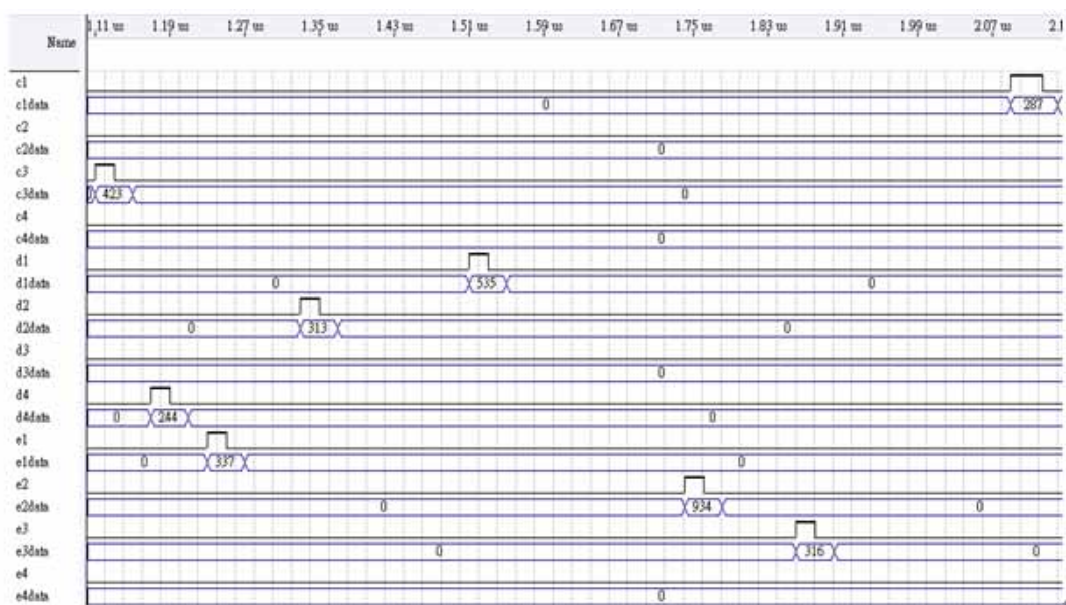
000000001100001011011，其為 E3 模組資料 6235 的二進位形式。第四筆輸出的 32bits 資料可知其第 31~29bits 為 111 代表此為總符合事件且屬於後模組的資料，第 28~21bits 為 01000101 帶表事件 EK-36 的編碼，第 20~0bits 為 000000101101110010100，其為 K6 模組資料 23444 的二進位形示。而 CI-49 及 DK-49 事件的資料亦可由相同的觀察方法觀察 CI-49 及 DK-49 的測試結果得知被順利輸出。

由測試結果得知，時間符合計算單元亦送出 12.5MHz 的存取時脈，以供應後端的數位 I/O 卡使用，標示信號則作為其寫入資料所需的 Write enable 信號。而由 CI-49 發現時間符合計算單元在輸出 CI-49 事件的資料後停止輸出，標示信號也轉為 Low level，這是因為輸出 CI-49 事件的資料後，FIFO 內存放的事件資料只剩一筆 DK-49 事件的資料，所以 FIFO 停止輸出。當事件 DJ-28 在經過  $1\mu\text{s}$  又被再次輸入事件符合計算電路並存如 FIFO 後，DK-49 才被讀出，標示信號轉為 High level。

## (二) 隨機符合事件的輸出測試

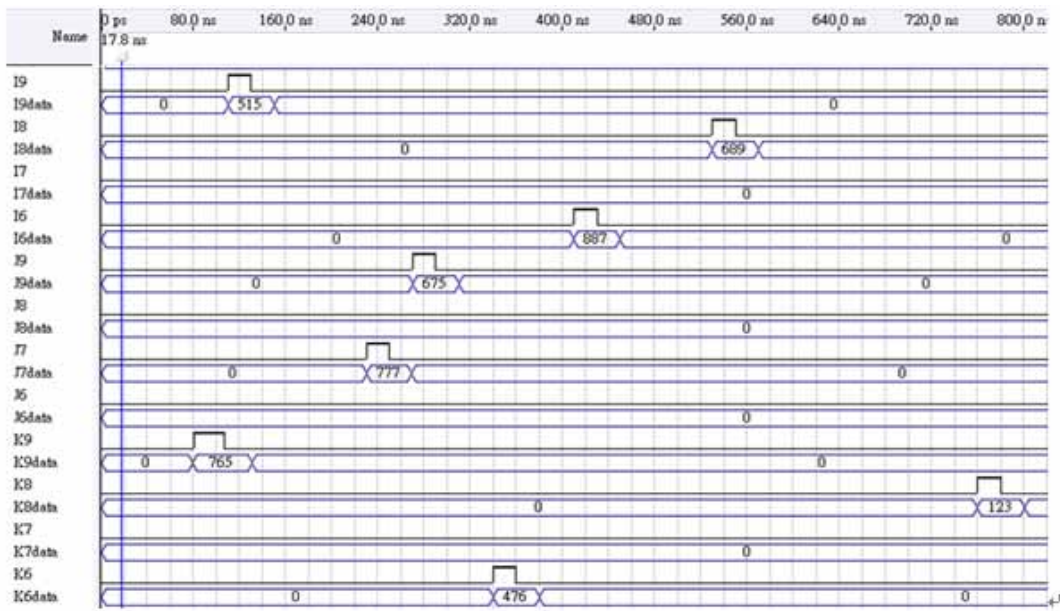
隨機符合事件必須在 FIFO 內累積到 128 筆資料的時候才會輸出，所以我們輸入的前、後模組測試資料其會在時間軸  $1310\mu\text{s}$ 、 $1830$

$\mu s$ 、 $1870 \mu s$ 、 $1910 \mu s$ 、 $2030 \mu s$ 、 $2080 \mu s$ 、 $2640 \mu s$ 、 $2860 \mu s$   
 處，依序發生 CI-39、DI-28、DJ-49、DI-16、EJ-17、CK-19、EK-26、  
 EK-38 等八筆隨機符合事件配對，並將此事件群以  $3 \mu s$  的週期重複  
 輸入 16 次即可達到 128 筆的隨機符合事件。



前模組輸入之測試資料

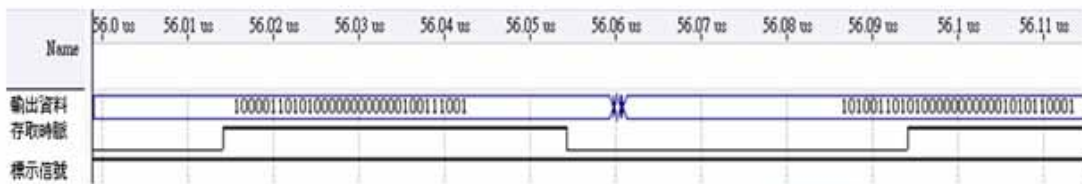




後模組輸入之測試資料



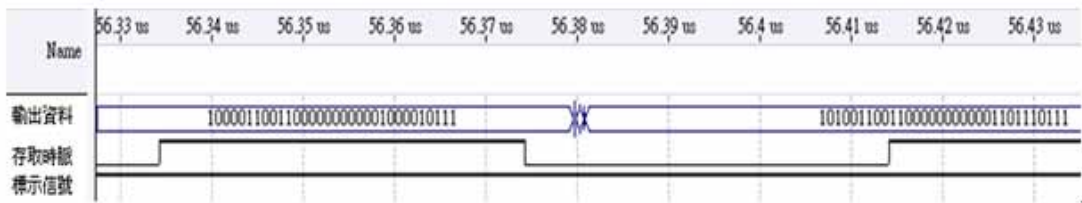
CI-39 隨機符合事件資料輸出



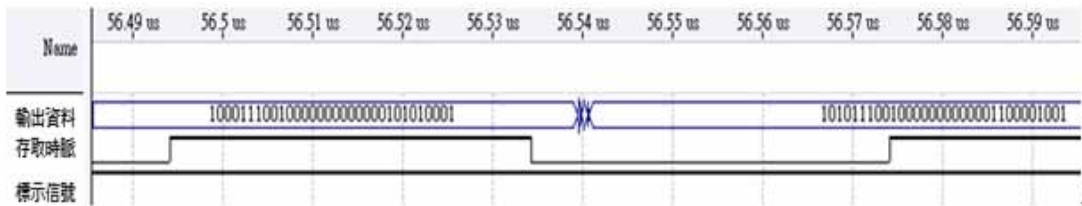
DI-28 隨機符合事件資料輸出



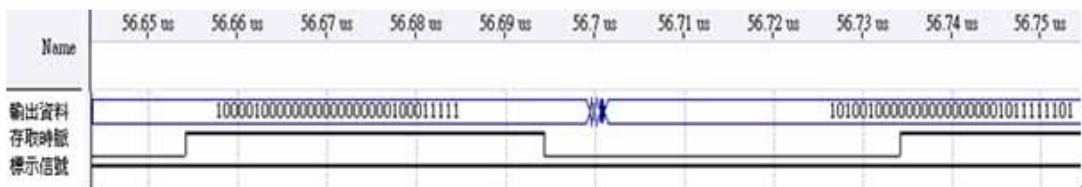
DJ-49 隨機符合事件資料輸出



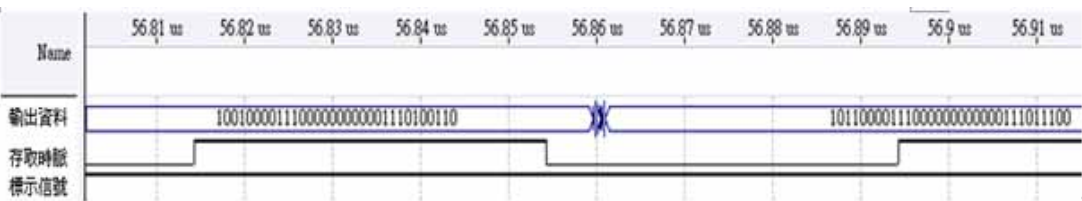
DI-16 隨機符合事件資料輸出 ↵



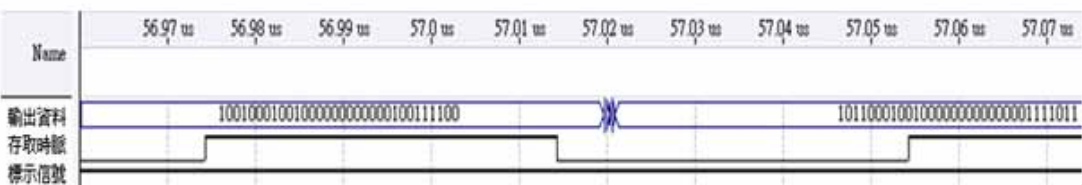
EJ-17 隨機符合事件資料輸出 ↵



CK-19 隨機符合事件資料輸出 ↵



EK-26 隨機符合事件資料輸出 ↵



EK-38 隨機符合事件資料輸出 ↵

由 CI\_39 第一筆輸出的 32bits 資料可知其第 31~29bits 為 100

代表此為隨機符合事件且屬於前模組的資料，第 28~21bits 為 00001000 帶表事件 CI-39 的編碼，第 20~0bits 為 000000000000110100111，其為 C3 模組資料 423 的二進位形式。第二筆輸出的 32bits 資料可知其第 31~29bits 為 101 代表此為總符合事件且屬於後模組的資料，第 28~21bits 為 00001000 帶表事件 CI-39 的編碼，第 20~0bits 為 000000000001000000011，其為 I9 模組資料 515 的二進位形式。其餘的 DI-28、DI-49、DI-16、EJ-17、CK-19、EK-26、EK-38 隨機符合事件，皆可以此觀察方法從圖 4.38~圖 4.44 得知，被隨機符合計算單元正確計算並由 FIFO 記憶體順利讀出。

### 三、時間模組 1 對 1 系統整合測試

本節將實現時間模組應用於 ADC 雙通道的整體測試，兩個 ADC 通道後端會連接各自的時間模組，演算完成的時間資料再送到共用的 FIFO 匯流排模組做資料組合、儲存以及完成最後的傳輸行為。在硬體實測方面，如圖 4.16 所示，與前一章節測試唯一不同之處是，脈衝產生器輸出的負脈衝訊號將透過 T 型接頭做一對二的分接，之後同樣使用 Xilinx Inc. 原廠傳輸線連結到數位訊號處理板上的兩個 ADC 通道，以建構 1 對 1 偵檢器的系統環境。

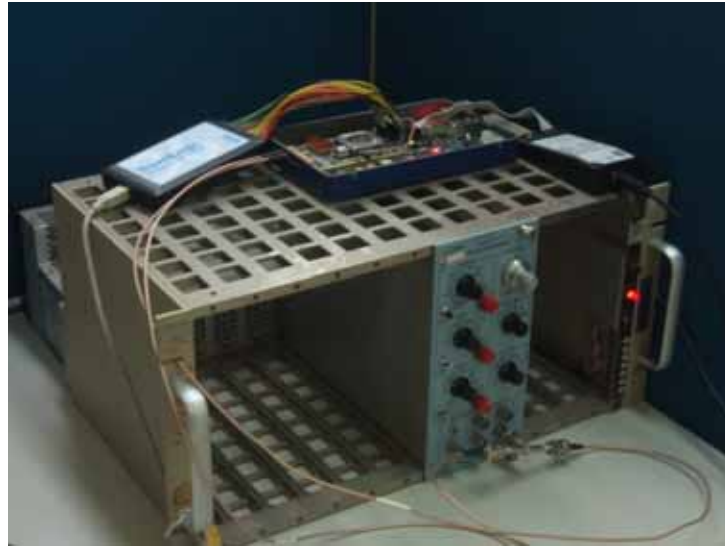


圖 4.16 硬體實測配置圖

時間模組與 FIFO 匯流排模組整合測試的輸出結果包含 3 種訊號：分別是提供 PCIe-6537 數位介面卡的時脈訊號、請求(Request)訊號以及 16 位元的時間標記資料，在此將該 16 位元資料依照不同 ADC 所得到的粗略與精細時間標記做分類呈現，如圖 4.17 所示，請求訊號為 1 時，則表示資料正在傳輸中，請求訊號為 0 時，則表示閒置。附註一提，由於以權重內插為核心的時間模組與二分內插所得到的結果是相同的，在此就不再重複陳述。

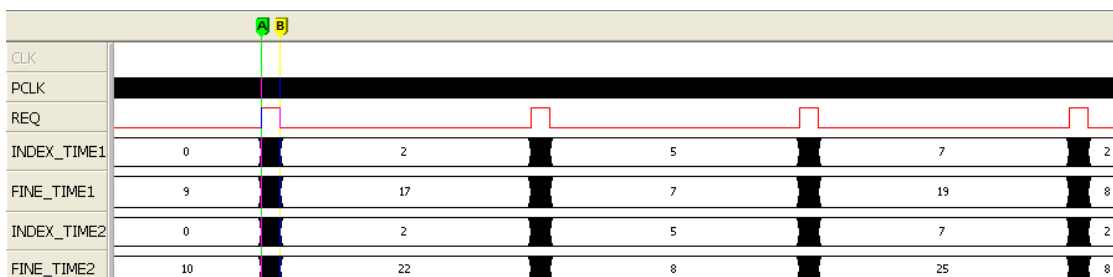


圖 4.17 整合測試結果

將圖 4.17 的測試結果局部放大，如圖 4.18 所示，可觀察到每次傳輸的資料量為 512 筆，也就是說，FIFO 匯流排模組正確地控制 FIFO 等待到深度為半滿，始連續輸出這 512 筆資料，另外，請求訊號亦與資料維持同步。

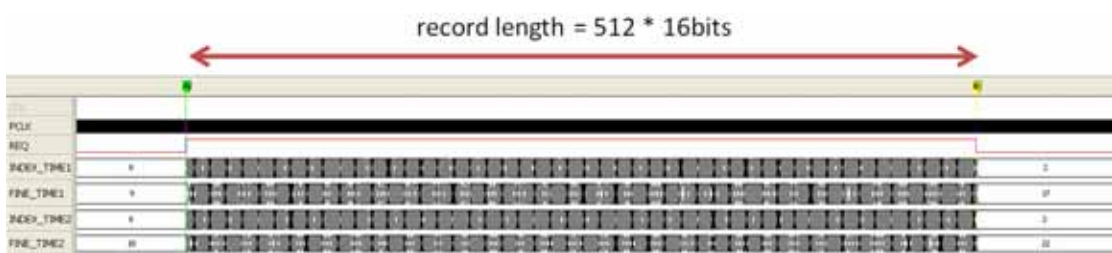


圖 4.18 整合測試結果放大圖

其詳細的資料內容如圖 4.19 所示，給予 PCIe-6537 的時脈確實為系統時脈的 1/4 頻率，而且 PCIe-6537 的時脈正緣處也恰於資料的中央位置，滿足了資料傳輸時的穩定性。

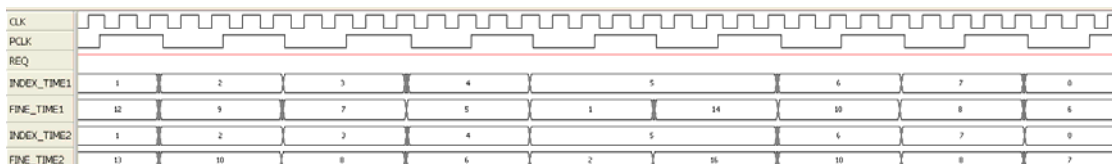


圖 4.19 整合測試結果資料內容

### (一) 數位訊號處理模組與 PCIe 介面整合測試

前兩節已經完成了數位時間系統的邏輯驗證，接下來將實現該系統模組與 PCIe 介面的整合，主要的成果包含資料擷取、資料後處理以及其分析結果。

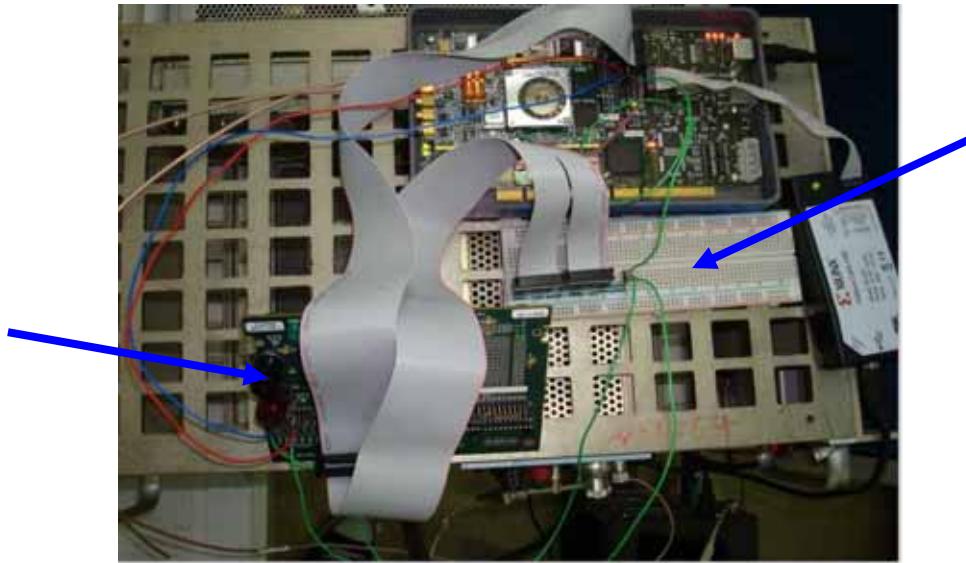


圖 4.20 DSP 模組與 PCIe 介面整合配置

如圖 4.20 所示，與前一節的測試相比，少了邏輯分析儀，而多了 3 個硬體：數位 I/O 連接板、麵包板以及 PCIe 數位 I/O 介面卡。數位 I/O 連接板主要功能為 GPIO(General Purpose I/O)與 PCIe 的介面轉接，板上具有數位 I/O 以及對地接腳，而數位訊號處理板的輸出共有 18 位元的訊號，在此以匯流排線做 16 位元資料的傳輸連接，2 條杜邦線(藍線、紅線)分別連接取樣時脈、請求訊號。至於每一個位元的訊號都有一個對地接腳，在這使用麵包板來當作短路片，將數位訊號處理板上的參考地引到麵包板上後，再把對地接腳(分別為 16 位元的匯流排線與 2 條綠色杜邦線)也連接到麵包板已規劃好的路徑上，來達到 18 位元的訊號與數位訊號處理板的全共地。最後，透過數位同軸電纜連接到電腦端的 PCIe-6537 介面卡，以完成資料通訊介

面的配置。

在資料擷取方面，將藉由本實驗室以 LabVIEW 軟體所開發的介面輔助實現[15]，如圖 4.21 所示，該介面主要用來控制 PCIe-6537 介面卡的組態設定，並且將擷取到的數位資料進行建檔與儲存。

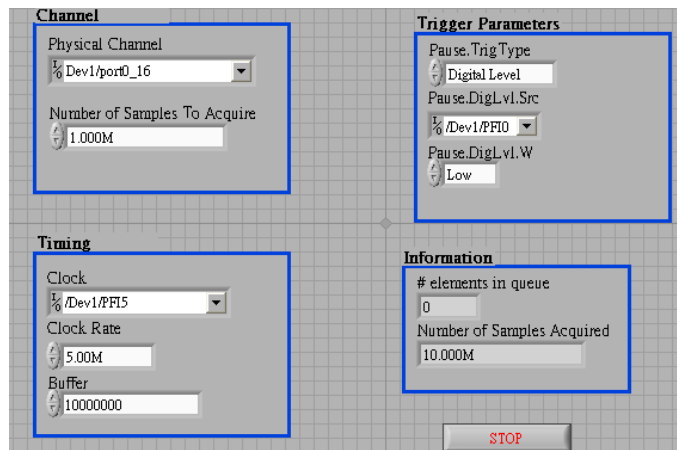


圖 4.21 PCIe 傳輸之使用者介面

擷取結果如圖 4.22 所示，每筆資料長度為 16 位元，第一筆資料為 8080，第二筆資料為 C0C0，依此類推，資料是由左至右，再由上至下地依照先後順序記錄。

	0001	0203	0405	0607	0809	0A0B	0C0D	0E0F
000000	0A0A	0506	2828	2122	4546	2D2F	6162	4A4A
000010	7074	6868	8A8B	8686	A8A8	A2A3	C6C6	AEB1
000020	E1E2	CACA	EEF1	E8E8	0A0A	150E	2828	2122
000030	4546	2D2F	6162	4A4A	6E70	6868	898A	8585
000040	A8A8	A1A2	C6C6	AFB3	E1E2	CACA	EEF1	E8E8
000050	2223	0A0A	2E30	2828	4A4A	4646	6868	6162
000060	8686	6F72	A2A3	8A8B	AFB2	A8A8	CACB	C6C6
000070	E8E9	E2E3	0A0B	EFF3	2829	2223	4647	3035
000080	6364	4A4B	7277	6869	8A8B	8687	A8A9	A3A4
000090	C6C7	B1B5	E3E4	CACB	F2F7	E8E9	0B0B	0E0F
0000A0	2829	2324	4647	3136	6364	4B4C	747C	6969

圖 4.22 原始資料內容

接著，將上述所擷取到的原始資料執行後處理，後處理介面如圖 4.23 所示，而其局部結果如圖 4.24 所示，資料表示兩個脈衝訊號的時間差，而單位為一個時脈週期，以小數的方式呈現，其結果可以分為 3 種情況：第 1 筆結果為時間資料 0A0A 所算得，表示 ADC1 脈衝與 ADC2 脈衝的時間完全一致；第 2 筆為時間資料 0506 所算得，表示 ADC1 脈衝領先 ADC2 脈衝 0.03125 單位(即 1/32 個時脈週期)；第 22 筆資料為時間資料 150E 所算得，表示 ADC1 脈衝落後 ADC2 脈衝 0.21875 單位 (即 7/32 個時脈週期)。

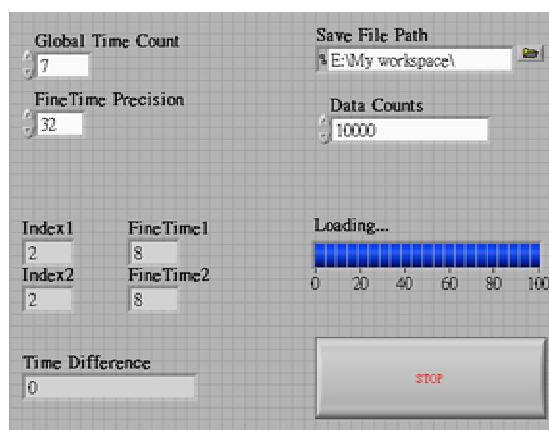


圖 4.23 後處理介面

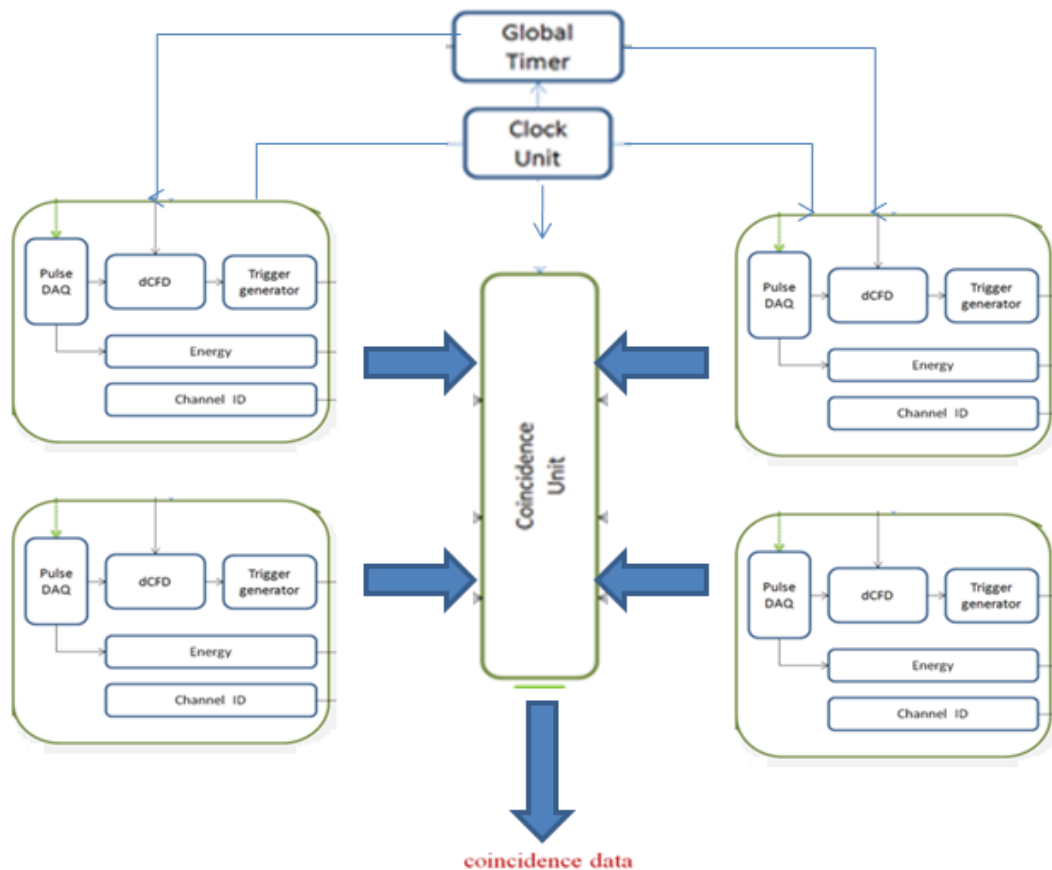
1	0	9	-0.125	17	-0.03125	25	-0.03125
2	-0.03125	10	0	18	0	26	-0.0625
3	0	11	-0.03125	19	-0.09375	27	-0.03125
4	-0.03125	12	0	20	0	28	0
5	-0.03125	13	0	21	0	29	-0.0625
6	-0.0625	14	-0.03125	22	0.21875	30	0
7	-0.03125	15	0	23	0	31	-0.03125
8	0	16	-0.09375	24	-0.03125	32	0

圖 4.24 資料後處理結果



#### 四、4 對 4 信號擷取系統整合測試

本節將測試 4 對 4 信號擷取系統運作與輸出資料，如上節所提時間電路模組以測試完畢且能達到 550ps 的時間解析度，對於 4 對 4 信號擷取系統就是整合 4 組 1 對 1 的時間電路模組並連接符合事件系統電路模組，架構圖如下所示。



4 對 4 信號擷取系統示意圖

如同 4.3 節脈敘述衝產生器輸出的負脈衝訊號將透過 T 型接頭做一對二的分接，之後同樣使用 Xilinx Inc. 原廠傳輸線連結到數位

訊號處理板上的兩個 ADC 通道，以建構 1 對 1 偵檢器的系統環境，而 4 對 4 共有 16 種組合，我們依序做 16 種不同組合之測試，藉由 PCIe-6537 與 PCIe 傳輸之使用者介面如圖 4.21 將資料存入硬碟並且用 PSpad 以 16 進制格式開啟輸出的符合事件資料檢查是否正確無誤。

輸出的資料型式是 16 進制的資料，而且前模組需先輸出後，後模組再輸出，以此順序循環一百萬筆。以下是 16 不同組合的配對代號表，來檢查輸出資料是否正確。

Title配對	配對信號	模組配對	配對信號
D-J	0100	2-8	0101
E-K	1000	2-7	0110
E-J	0111	3-8	1001
D-K	0101	3-7	1010

我們設定的模組名稱為前模組 D2, D3, E2, E3 後模組 K7, K8, J7, J8 共 16 種組合。

DK\_28 寫入硬碟資料

	0001	0203	0405	0607	0809	0A0B	0C0D	0E0F
000000	CAA0	EAA0	CAA0	EAA0	CAA0	EAA0	CAA0	EAA0
000010	CAA0	EAA0	CAA0	EAA0	CAA0	EAA0	CAA0	EAA0
000020	CAA0	EAA0	CAA0	EAA0	CAA0	EAA0	CAA0	EAA0
000030	CAA0	EAA0	CAA0	EAA0	CAA0	EAA0	CAA0	EAA0
000040	CAA0	EAA0	CAA0	EAA0	CAA0	EAA0	CAA0	EAA0
000050	CAA0	EAA0	CAA0	EAA0	CAA0	EAA0	CAA0	EAA0
000060	CAA0	EAA0	CAA0	EAA0	CAA0	EAA0	CAA0	EAA0
000070	CAA0	EAA0	CAA0	EAA0	CAA0	EAA0	CAA0	EAA0
000080	CAA0	EAA0	CAA0	EAA0	CAA0	EAA0	CAA0	EAA0
000090	CAA0	EAA0	CAA0	EAA0	CAA0	EAA0	CAA0	EAA0
0000A0	CAA0	EAA0	CAA0	EAA0	CAA0	EAA0	CAA0	EAA0
0000B0	CAA0	EAA0	CAA0	EAA0	CAA0	EAA0	CAA0	EAA0
0000C0	CAA0	EAA0	CAA0	EAA0	CAA0	EAA0	CAA0	EAA0
0000D0	CAA0	EAA0	CAA0	EAA0	CAA0	EAA0	CAA0	EAA0
0000E0	CAA0	EAA0	CAA0	EAA0	CAA0	EAA0	CAA0	EAA0
0000F0	CAA0	EAA0	CAA0	EAA0	CAA0	EAA0	CAA0	EAA0

輸出資料前模組代碼為 CAA0 後模組代碼為 EAA0，且按照前後順序輸出並且循環，輸出資料如我們預期一樣。

DK\_27 寫入硬碟資料

	0001	0203	0405	0607	0809	0A0B	0C0D	0E0F
000000	CAC0	EAC0	CAC0	EAC0	CAC0	EAC0	CAC0	EAC0
000010	CAC0	EAC0	CAC0	EAC0	CAC0	EAC0	CAC0	EAC0
000020	CAC0	EAC0	CAC0	EAC0	CAC0	EAC0	CAC0	EAC0
000030	CAC0	EAC0	CAC0	EAC0	CAC0	EAC0	CAC0	EAC0
000040	CAC0	EAC0	CAC0	EAC0	CAC0	EAC0	CAC0	EAC0
000050	CAC0	EAC0	CAC0	EAC0	CAC0	EAC0	CAC0	EAC0
000060	CAC0	EAC0	CAC0	EAC0	CAC0	EAC0	CAC0	EAC0
000070	CAC0	EAC0	CAC0	EAC0	CAC0	EAC0	CAC0	EAC0
000080	CAC0	EAC0	CAC0	EAC0	CAC0	EAC0	CAC0	EAC0
000090	CAC0	EAC0	CAC0	EAC0	CAC0	EAC0	CAC0	EAC0
0000A0	CAC0	EAC0	CAC0	EAC0	CAC0	EAC0	CAC0	EAC0
0000B0	CAC0	EAC0	CAC0	EAC0	CAC0	EAC0	CAC0	EAC0
0000C0	CAC0	EAC0	CAC0	EAC0	CAC0	EAC0	CAC0	EAC0
0000D0	CAC0	EAC0	CAC0	EAC0	CAC0	EAC0	CAC0	EAC0
0000E0	CAC0	EAC0	CAC0	EAC0	CAC0	EAC0	CAC0	EAC0
0000F0	CAC0	EAC0	CAC0	EAC0	CAC0	EAC0	CAC0	EAC0

輸出資料依序為前模組代碼 CAC0 後模組代碼 EAC0。

DJ\_28 寫入硬碟資料

	0001	0203	0405	0607	0809	0A0B	0C0D	0E0F
000000	C8A0	E8A0	C8A0	E8A0	C8A0	E8A0	C8A0	E8A0
000010	C8A0	E8A0	C8A0	E8A0	C8A0	E8A0	C8A0	E8A0
000020	C8A0	E8A0	C8A0	E8A0	C8A0	E8A0	C8A0	E8A0
000030	C8A0	E8A0	C8A0	E8A0	C8A0	E8A0	C8A0	E8A0
000040	C8A0	E8A0	C8A0	E8A0	C8A0	E8A0	C8A0	E8A0
000050	C8A0	E8A0	C8A0	E8A0	C8A0	E8A0	C8A0	E8A0
000060	C8A0	E8A0	C8A0	E8A0	C8A0	E8A0	C8A0	E8A0
000070	C8A0	E8A0	C8A0	E8A0	C8A0	E8A0	C8A0	E8A0
000080	C8A0	E8A0	C8A0	E8A0	C8A0	E8A0	C8A0	E8A0
000090	C8A0	E8A0	C8A0	E8A0	C8A0	E8A0	C8A0	E8A0
0000A0	C8A0	E8A0	C8A0	E8A0	C8A0	E8A0	C8A0	E8A0
0000B0	C8A0	E8A0	C8A0	E8A0	C8A0	E8A0	C8A0	E8A0
0000C0	C8A0	E8A0	C8A0	E8A0	C8A0	E8A0	C8A0	E8A0
0000D0	C8A0	E8A0	C8A0	E8A0	C8A0	E8A0	C8A0	E8A0
0000E0	C8A0	E8A0	C8A0	E8A0	C8A0	E8A0	C8A0	E8A0
0000F0	C8A0	E8A0	C8A0	E8A0	C8A0	E8A0	C8A0	E8A0

輸出資料依序為前模組代碼 C8A0 後模組代碼 E8A0。

DJ\_27 寫入硬碟資料

	0001	0203	0405	0607	0809	0A0B	0C0D	0E0F
000000	C8C0	E8C0	C8C0	E8C0	C8C0	E8C0	C8C0	E8C0
000010	C8C0	E8C0	C8C0	E8C0	C8C0	E8C0	C8C0	E8C0
000020	C8C0	E8C0	C8C0	E8C0	C8C0	E8C0	C8C0	E8C0
000030	C8C0	E8C0	C8C0	E8C0	C8C0	E8C0	C8C0	E8C0
000040	C8C0	E8C0	C8C0	E8C0	C8C0	E8C0	C8C0	E8C0
000050	C8C0	E8C0	C8C0	E8C0	C8C0	E8C0	C8C0	E8C0
000060	C8C0	E8C0	C8C0	E8C0	C8C0	E8C0	C8C0	E8C0
000070	C8C0	E8C0	C8C0	E8C0	C8C0	E8C0	C8C0	E8C0
000080	C8C0	E8C0	C8C0	E8C0	C8C0	E8C0	C8C0	E8C0
000090	C8C0	E8C0	C8C0	E8C0	C8C0	E8C0	C8C0	E8C0
0000A0	C8C0	E8C0	C8C0	E8C0	C8C0	E8C0	C8C0	E8C0
0000B0	C8C0	E8C0	C8C0	E8C0	C8C0	E8C0	C8C0	E8C0
0000C0	C8C0	E8C0	C8C0	E8C0	C8C0	E8C0	C8C0	E8C0
0000D0	C8C0	E8C0	C8C0	E8C0	C8C0	E8C0	C8C0	E8C0
0000E0	C8C0	E8C0	C8C0	E8C0	C8C0	E8C0	C8C0	E8C0
0000F0	C8C0	E8C0	C8C0	E8C0	C8C0	E8C0	C8C0	E8C0

輸出資料依序為前模組代碼 C8C0 後模組代碼 E8C0。

DK\_38 寫入硬碟資料

	0001	0203	0405	0607	0809	0A0B	0C0D	0E0F
000000	CB20	EB20	CB20	EB20	CB20	EB20	CB20	EB20
000010	CB20	EB20	CB20	EB20	CB20	EB20	CB20	EB20
000020	CB20	EB20	CB20	EB20	CB20	EB20	CB20	EB20
000030	CB20	EB20	CB20	EB20	CB20	EB20	CB20	EB20
000040	CB20	EB20	CB20	EB20	CB20	EB20	CB20	EB20
000050	CB20	EB20	CB20	EB20	CB20	EB20	CB20	EB20
000060	CB20	EB20	CB20	EB20	CB20	EB20	CB20	EB20
000070	CB20	EB20	CB20	EB20	CB20	EB20	CB20	EB20
000080	CB20	EB20	CB20	EB20	CB20	EB20	CB20	EB20
000090	CB20	EB20	CB20	EB20	CB20	EB20	CB20	EB20
0000A0	CB20	EB20	CB20	EB20	CB20	EB20	CB20	EB20
0000B0	CB20	EB20	CB20	EB20	CB20	EB20	CB20	EB20
0000C0	CB20	EB20	CB20	EB20	CB20	EB20	CB20	EB20
0000D0	CB20	EB20	CB20	EB20	CB20	EB20	CB20	EB20
0000E0	CB20	EB20	CB20	EB20	CB20	EB20	CB20	EB20
0000F0	CB20	EB20	CB20	EB20	CB20	EB20	CB20	EB20

輸出資料依序為前模組代碼 CB20 後模組代碼 E820。

DK\_37 寫入硬碟資料

	0001	0203	0405	0607	0809	0A0B	0C0D	0EOF
000000	CB40	EB40	CB40	EB40	CB40	EB40	CB40	EB40
000010	CB40	EB40	CB40	EB40	CB40	EB40	CB40	EB40
000020	CB40	EB40	CB40	EB40	CB40	EB40	CB40	EB40
000030	CB40	EB40	CB40	EB40	CB40	EB40	CB40	EB40
000040	CB40	EB40	CB40	EB40	CB40	EB40	CB40	EB40
000050	CB40	EB40	CB40	EB40	CB40	EB40	CB40	EB40
000060	CB40	EB40	CB40	EB40	CB40	EB40	CB40	EB40
000070	CB40	EB40	CB40	EB40	CB40	EB40	CB40	EB40
000080	CB40	EB40	CB40	EB40	CB40	EB40	CB40	EB40
000090	CB40	EB40	CB40	EB40	CB40	EB40	CB40	EB40
0000A0	CB40	EB40	CB40	EB40	CB40	EB40	CB40	EB40
0000B0	CB40	EB40	CB40	EB40	CB40	EB40	CB40	EB40
0000C0	CB40	EB40	CB40	EB40	CB40	EB40	CB40	EB40
0000D0	CB40	EB40	CB40	EB40	CB40	EB40	CB40	EB40
0000E0	CB40	EB40	CB40	EB40	CB40	EB40	CB40	EB40
0000F0	CB40	EB40	CB40	EB40	CB40	EB40	CB40	EB40
000100	CB40	EB40	CB40	EB40	CB40	EB40	CB40	EB40

輸出資料依序為前模組代碼 CB40 後模組代碼 EB40。

DJ\_38 寫入硬碟資料



	0001	0203	0405	0607	0809	0A0B	0C0D	0E0F
000000	C920	E920	C920	E920	C920	E920	C920	E920
000010	C920	E920	C920	E920	C920	E920	C920	E920
000020	C920	E920	C920	E920	C920	E920	C920	E920
000030	C920	E920	C920	E920	C920	E920	C920	E920
000040	C920	E920	C920	E920	C920	E920	C920	E920
000050	C920	E920	C920	E920	C920	E920	C920	E920
000060	C920	E920	C920	E920	C920	E920	C920	E920
000070	C920	E920	C920	E920	C920	E920	C920	E920
000080	C920	E920	C920	E920	C920	E920	C920	E920
000090	C920	E920	C920	E920	C920	E920	C920	E920
0000A0	C920	E920	C920	E920	C920	E920	C920	E920
0000B0	C920	E920	C920	E920	C920	E920	C920	E920
0000C0	C920	E920	C920	E920	C920	E920	C920	E920
0000D0	C920	E920	C920	E920	C920	E920	C920	E920
0000E0	C920	E920	C920	E920	C920	E920	C920	E920
0000F0	C920	E920	C920	E920	C920	E920	C920	E920

輸出資料依序為前模組代碼 C920 後模組代碼 E920。

DJ\_37 寫入硬碟資料

	0001	0203	0405	0607	0809	0A0B	0C0D	0E0F
000000	C940	E940	C940	E940	C940	E940	C940	E940
000010	C940	E940	C940	E940	C940	E940	C940	E940
000020	C940	E940	C940	E940	C940	E940	C940	E940
000030	C940	E940	C940	E940	C940	E940	C940	E940
000040	C940	E940	C940	E940	C940	E940	C940	E940
000050	C940	E940	C940	E940	C940	E940	C940	E940
000060	C940	E940	C940	E940	C940	E940	C940	E940
000070	C940	E940	C940	E940	C940	E940	C940	E940
000080	C940	E940	C940	E940	C940	E940	C940	E940
000090	C940	E940	C940	E940	C940	E940	C940	E940
0000A0	C940	E940	C940	E940	C940	E940	C940	E940
0000B0	C940	E940	C940	E940	C940	E940	C940	E940
0000C0	C940	E940	C940	E940	C940	E940	C940	E940
0000D0	C940	E940	C940	E940	C940	E940	C940	E940
0000E0	C940	E940	C940	E940	C940	E940	C940	E940
0000F0	C940	E940	C940	E940	C940	E940	C940	E940

輸出資料依序為前模組代碼 C940 後模組代碼 E940。

EK\_28 寫入硬碟資料

	0001	0203	0405	0607	0809	0A0B	0C0D	0E0F
000000	DOAO	FOAO	DOAO	FOAO	DOAO	FOAO	DOAO	FOAO
000010	DOAO	FOAO	DOAO	FOAO	DOAO	FOAO	DOAO	FOAO
000020	DOAO	FOAO	DOAO	FOAO	DOAO	FOAO	DOAO	FOAO
000030	DOAO	FOAO	DOAO	FOAO	DOAO	FOAO	DOAO	FOAO
000040	DOAO	FOAO	DOAO	FOAO	DOAO	FOAO	DOAO	FOAO
000050	DOAO	FOAO	DOAO	FOAO	DOAO	FOAO	DOAO	FOAO
000060	DOAO	FOAO	DOAO	FOAO	DOAO	FOAO	DOAO	FOAO
000070	DOAO	FOAO	DOAO	FOAO	DOAO	FOAO	DOAO	FOAO
000080	DOAO	FOAO	DOAO	FOAO	DOAO	FOAO	DOAO	FOAO
000090	DOAO	FOAO	DOAO	FOAO	DOAO	FOAO	DOAO	FOAO
0000A0	DOAO	FOAO	DOAO	FOAO	DOAO	FOAO	DOAO	FOAO
0000B0	DOAO	FOAO	DOAO	FOAO	DOAO	FOAO	DOAO	FOAO
0000C0	DOAO	FOAO	DOAO	FOAO	DOAO	FOAO	DOAO	FOAO
0000D0	DOAO	FOAO	DOAO	FOAO	DOAO	FOAO	DOAO	FOAO
0000E0	DOAO	FOAO	DOAO	FOAO	DOAO	FOAO	DOAO	FOAO
0000F0	DOAO	FOAO	DOAO	FOAO	DOAO	FOAO	DOAO	FOAO
000100	DOAO	FOAO	DOAO	FOAO	DOAO	FOAO	DOAO	FOAO

輸出資料依序為前模組代碼 DOA0 後模組代碼 FOA0。

EK\_27 寫入硬碟資料

	0001	0203	0405	0607	0809	0A0B	0C0D	0E0F
000000	DOC0	FOC0	DOC0	FOC0	DOC0	FOC0	DOC0	FOC0
000010	DOC0	FOC0	DOC0	FOC0	DOC0	FOC0	DOC0	FOC0
000020	DOC0	FOC0	DOC0	FOC0	DOC0	FOC0	DOC0	FOC0
000030	DOC0	FOC0	DOC0	FOC0	DOC0	FOC0	DOC0	FOC0
000040	DOC0	FOC0	DOC0	FOC0	DOC0	FOC0	DOC0	FOC0
000050	DOC0	FOC0	DOC0	FOC0	DOC0	FOC0	DOC0	FOC0
000060	DOC0	FOC0	DOC0	FOC0	DOC0	FOC0	DOC0	FOC0
000070	DOC0	FOC0	DOC0	FOC0	DOC0	FOC0	DOC0	FOC0
000080	DOC0	FOC0	DOC0	FOC0	DOC0	FOC0	DOC0	FOC0
000090	DOC0	FOC0	DOC0	FOC0	DOC0	FOC0	DOC0	FOC0
0000A0	DOC0	FOC0	DOC0	FOC0	DOC0	FOC0	DOC0	FOC0
0000B0	DOC0	FOC0	DOC0	FOC0	DOC0	FOC0	DOC0	FOC0
0000C0	DOC0	FOC0	DOC0	FOC0	DOC0	FOC0	DOC0	FOC0
0000D0	DOC0	FOC0	DOC0	FOC0	DOC0	FOC0	DOC0	FOC0
0000E0	DOC0	FOC0	DOC0	FOC0	DOC0	FOC0	DOC0	FOC0
0000F0	DOC0	FOC0	DOC0	FOC0	DOC0	FOC0	DOC0	FOC0
000100	DOC0	FOC0	DOC0	FOC0	DOC0	FOC0	DOC0	FOC0

輸出資料依序為前模組代碼 DOC0 後模組代碼 FOC0。

EJ\_28 寫入硬碟資料

	0001	0203	0405	0607	0809	0A0B	0C0D	0E0F
000000	CEA0	EFA0	CEA0	EFA0	CEA0	EFA0	CEA0	EFA0
000010	CEA0	EFA0	CEA0	EFA0	CEA0	EFA0	CEA0	EFA0
000020	CEA0	EFA0	CEA0	EFA0	CEA0	EFA0	CEA0	EFA0
000030	CEA0	EFA0	CEA0	EFA0	CEA0	EFA0	CEA0	EFA0
000040	CEA0	EFA0	CEA0	EFA0	CEA0	EFA0	CEA0	EFA0
000050	CEA0	EFA0	CEA0	EFA0	CEA0	EFA0	CEA0	EFA0
000060	CEA0	EFA0	CEA0	EFA0	CEA0	EFA0	CEA0	EFA0
000070	CEA0	EFA0	CEA0	EFA0	CEA0	EFA0	CEA0	EFA0
000080	CEA0	EFA0	CEA0	EFA0	CEA0	EFA0	CEA0	EFA0
000090	CEA0	EFA0	CEA0	EFA0	CEA0	EFA0	CEA0	EFA0
0000A0	CEA0	EFA0	CEA0	EFA0	CEA0	EFA0	CEA0	EFA0
0000B0	CEA0	EFA0	CEA0	EFA0	CEA0	EFA0	CEA0	EFA0
0000C0	CEA0	EFA0	CEA0	EFA0	CEA0	EFA0	CEA0	EFA0
0000D0	CEA0	EFA0	CEA0	EFA0	CEA0	EFA0	CEA0	EFA0
0000E0	CEA0	EFA0	CEA0	EFA0	CEA0	EFA0	CEA0	EFA0
0000F0	CEA0	EFA0	CEA0	EFA0	CEA0	EFA0	CEA0	EFA0
000100	CEA0	EFA0	CEA0	EFA0	CEA0	EFA0	CEA0	EFA0

輸出資料依序為前模組代碼 CEA0 後模組代碼 EFA0。

EJ\_27 寫入硬碟資料

	0001	0203	0405	0607	0809	0A0B	0C0D	0E0F
000000	CEC0	EEC0	CEC0	EEC0	CEC0	EEC0	CEC0	EEC0
000010	CEC0	EEC0	CEC0	EEC0	CEC0	EEC0	CEC0	EEC0
000020	CEC0	EEC0	CEC0	EEC0	CEC0	EEC0	CEC0	EEC0
000030	CEC0	EEC0	CEC0	EEC0	CEC0	EEC0	CEC0	EEC0
000040	CEC0	EEC0	CEC0	EEC0	CEC0	EEC0	CEC0	EEC0
000050	CEC0	EEC0	CEC0	EEC0	CEC0	EEC0	CEC0	EEC0
000060	CEC0	EEC0	CEC0	EEC0	CEC0	EEC0	CEC0	EEC0
000070	CEC0	EEC0	CEC0	EEC0	CEC0	EEC0	CEC0	EEC0
000080	CEC0	EEC0	CEC0	EEC0	CEC0	EEC0	CEC0	EEC0
000090	CEC0	EEC0	CEC0	EEC0	CEC0	EEC0	CEC0	EEC0
0000A0	CEC0	EEC0	CEC0	EEC0	CEC0	EEC0	CEC0	EEC0
0000B0	CEC0	EEC0	CEC0	EEC0	CEC0	EEC0	CEC0	EEC0
0000C0	CEC0	EEC0	CEC0	EEC0	CEC0	EEC0	CEC0	EEC0
0000D0	CEC0	EEC0	CEC0	EEC0	CEC0	EEC0	CEC0	EEC0
0000E0	CEC0	EEC0	CEC0	EEC0	CEC0	EEC0	CEC0	EEC0
0000F0	CEC0	EEC0	CEC0	EEC0	CEC0	EEC0	CEC0	EEC0
000100	CEC0	EEC0	CEC0	EEC0	CEC0	EEC0	CEC0	EEC0

輸出資料依序為前模組代碼 CEC0 後模組代碼 EEC0。

EK\_38 寫入硬碟資料

	0001	0203	0405	0607	0809	0A0B	0C0D	0E0F
000000	D120	F120	D120	F120	D120	F120	D120	F120
000010	D120	F120	D120	F120	D120	F120	D120	F120
000020	D120	F120	D120	F120	D120	F120	D120	F120
000030	D120	F120	D120	F120	D120	F120	D120	F120
000040	D120	F120	D120	F120	D120	F120	D120	F120
000050	D120	F120	D120	F120	D120	F120	D120	F120
000060	D120	F120	D120	F120	D120	F120	D120	F120
000070	D120	F120	D120	F120	D120	F120	D120	F120
000080	D120	F120	D120	F120	D120	F120	D120	F120
000090	D120	F120	D120	F120	D120	F120	D120	F120
0000A0	D120	F120	D120	F120	D120	F120	D120	F120
0000B0	D120	F120	D120	F120	D120	F120	D120	F120
0000C0	D120	F120	D120	F120	D120	F120	D120	F120
0000D0	D120	F120	D120	F120	D120	F120	D120	F120
0000E0	D120	F120	D120	F120	D120	F120	D120	F120
0000F0	D120	F120	D120	F120	D120	F120	D120	F120
000100	D120	F120	D120	F120	D120	F120	D120	F120

輸出資料依序為前模組代碼 D120 後模組代碼 F120。

EK\_37 寫入硬碟資料

	0001	0203	0405	0607	0809	0A0B	0C0D	0E0F
000000	D140	F140	D140	F140	D140	F140	D140	F140
000010	D140	F140	D140	F140	D140	F140	D140	F140
000020	D140	F140	D140	F140	D140	F140	D140	F140
000030	D140	F140	D140	F140	D140	F140	D140	F140
000040	D140	F140	D140	F140	D140	F140	D140	F140
000050	D140	F140	D140	F140	D140	F140	D140	F140
000060	D140	F140	D140	F140	D140	F140	D140	F140
000070	D140	F140	D140	F140	D140	F140	D140	F140
000080	D140	F140	D140	F140	D140	F140	D140	F140
000090	D140	F140	D140	F140	D140	F140	D140	F140
0000A0	D140	F140	D140	F140	D140	F140	D140	F140
0000B0	D140	F140	D140	F140	D140	F140	D140	F140
0000C0	D140	F140	D140	F140	D140	F140	D140	F140
0000D0	D140	F140	D140	F140	D140	F140	D140	F140
0000E0	D140	F140	D140	F140	D140	F140	D140	F140
0000F0	D140	F140	D140	F140	D140	F140	D140	F140
000100	D140	F140	D140	F140	D140	F140	D140	F140

輸出資料依序為前模組代碼 D140 後模組代碼 F140。

EJ\_38 寫入硬碟資料



000000	CF20	EF20	CF20	EF20	CF20	EF20	CF20	EF20
000010	CF20	EF20	CF20	EF20	CF20	EF20	CF20	EF20
000020	CF20	EF20	CF20	EF20	CF20	EF20	CF20	EF20
000030	CF20	EF20	CF20	EF20	CF20	EF20	CF20	EF20
000040	CF20	EF20	CF20	EF20	CF20	EF20	CF20	EF20
000050	CF20	EF20	CF20	EF20	CF20	EF20	CF20	EF20
000060	CF20	EF20	CF20	EF20	CF20	EF20	CF20	EF20
000070	CF20	EF20	CF20	EF20	CF20	EF20	CF20	EF20
000080	CF20	EF20	CF20	EF20	CF20	EF20	CF20	EF20
000090	CF20	EF20	CF20	EF20	CF20	EF20	CF20	EF20
0000A0	CF20	EF20	CF20	EF20	CF20	EF20	CF20	EF20
0000B0	CF20	EF20	CF20	EF20	CF20	EF20	CF20	EF20
0000C0	CF20	EF20	CF20	EF20	CF20	EF20	CF20	EF20
0000D0	CF20	EF20	CF20	EF20	CF20	EF20	CF20	EF20
0000E0	CF20	EF20	CF20	EF20	CF20	EF20	CF20	EF20
0000F0	CF20	EF20	CF20	EF20	CF20	EF20	CF20	EF20
000100	CF20	EF20	CF20	EF20	CF20	EF20	CF20	EF20

輸出資料依序為前模組代碼 CF20 後模組代碼 EF20。

EJ\_37 寫入硬碟資料

	0001	0203	0405	0607	0809	0A0B	0C0D	0E0F
000000	CF40	EF40	CF40	EF40	CF40	EF40	CF40	EF40
000010	CF40	EF40	CF40	EF40	CF40	EF40	CF40	EF40
000020	CF40	EF40	CF40	EF40	CF40	EF40	CF40	EF40
000030	CF40	EF40	CF40	EF40	CF40	EF40	CF40	EF40
000040	CF40	EF40	CF40	EF40	CF40	EF40	CF40	EF40
000050	CF40	EF40	CF40	EF40	CF40	EF40	CF40	EF40
000060	CF40	EF40	CF40	EF40	CF40	EF40	CF40	EF40
000070	CF40	EF40	CF40	EF40	CF40	EF40	CF40	EF40
000080	CF40	EF40	CF40	EF40	CF40	EF40	CF40	EF40
000090	CF40	EF40	CF40	EF40	CF40	EF40	CF40	EF40
0000A0	CF40	EF40	CF40	EF40	CF40	EF40	CF40	EF40
0000B0	CF40	EF40	CF40	EF40	CF40	EF40	CF40	EF40
0000C0	CF40	EF40	CF40	EF40	CF40	EF40	CF40	EF40
0000D0	CF40	EF40	CF40	EF40	CF40	EF40	CF40	EF40
0000E0	CF40	EF40	CF40	EF40	CF40	EF40	CF40	EF40
0000F0	CF40	EF40	CF40	EF40	CF40	EF40	CF40	EF40
000100	CF40	EF40	CF40	EF40	CF40	EF40	CF40	EF40

輸出資料依序為前模組代碼 CF40 後模組代碼 EF40。

## 五、結論

本論文基於全數位式硬體實現了高解析度時間鑑別系統，它不但可以即時偵測與擷取脈衝波形，並轉換成粗略以及高解析的數位時間資訊，也能夠隨時因應需求來進行功能的修改或者擴充。

時間鑑別系統一方面以「權重內插」和「二分內插」兩種不同電

路架構來使單一通道的時間精度有效地提升到 297.6ps，為原先未經過內插演算的 32 倍；另一方面以 1 對 1 雙通道進行整合，經 FPGA 計算完畢的資料透過 PCIe 介面送至電腦端，以程式軟體執行後處理，其統計結果可知本系統的平均時間解析度為 550ps。

比較權重法與二分法的電路特性，權重法所使用的資源較多，優點是運算時間較快，為 38ns；然而，採用二分法的電路資源較小，約為權重內插的一半，但必須付出的代價是速度變慢，運算時間延長到 104.5ns；而它們都具備共同的優點——dead time 很小，由於兩種演算電路皆使用管線化的設計，在工作頻率為 105MHz 的環境下，它們都可以處理連續而來的資料。

資料分析結果顯示在 0.2V~1.1V 的範圍內，本系統並不受脈衝振幅大小的影響，皆能達到 550ps 的時間解析度；除此之外，分析結果亦呈現出脈衝的上升時間與時間解析度具有關聯性，當上升時間增加，則時間解析度愈接近理論解析度 297.6ps。

本時間鑑別系統與後級之事件時間符合計算電路整合後，可成功測試將符合事件傳送至電腦端儲存，等待後續成像處理。

## 肆、参考文献

- [1] M.-A. Tétrault, et. al., “System Architecture of the LabPET Small Animal PET Scanner,” *IEEE Trans. Nucl. Sci.*, vol. 55, no. 5, pp.2546-2550, 2008.
- [2] R. Fontaine, et. al., “The Hardware and Signal Processing Architecture of LabPET™, a Small Animal APD-Based Digital PET Scanner,” *IEEE Trans. Nucl. Sci.*, vol.56, no.1, pp.3-9, 2009.
- [3] M. Bergeron, et. al., “Performance Evaluation of the LabPET APD-Based Digital PET Scanner,” *IEEE Trans. Nucl. Sci.*, vol.56, no.1, pp.10-16, 2009.
- [4] Chao Wang, et. al., “A Real Time Coincidence System for High Count-Rate TOF or Non-TOF PET Cameras Using Hybrid Method Combining AND-Logic and Time-Mark Technology,” *IEEE Trans. Nucl. Sci.*, vol.57, no.2, pp.708-714, 2010.
- [5] B. K. Swann, et. al., “A Custom Mixed-Signal CMOS Integrated Circuit for High Performance PET Tomograph Front-End Applications,” *IEEE Trans. Nucl. Sci.*, vol.50, no.4, pp.909-914, 2003.
- [6] J.-D. Leroux, et. al., “Time determination of BGO-APD detectors by digital signal processing for positron emission tomography,” *2003 IEEE Nuclear Science Symposium Conference Record*, vol.3, pp. 1723-1727, 2003.
- [7] A. Fallu-Labruyere, H. Tan, W. Hennig and W. K. Warburton, “Time resolution studies using digital constant fraction discriminator,” *Nucl. Instrum. Meth. Phys. Res. A*, vol. 579, pp.247-251, 2007 .
- [8] G. Hegyesi, et. al., “Ethernet Based Distributed Data Acquisition System for a Small Animal PET,” *IEEE Trans. Nucl. Sci.*, vol.53, no.4, pp.2112-2117, 2006.
- [9] P. Guerra , J. E. Ortuo , G. Kontaxakis , M. J. Ledesma-Carbayo , J. J. Vaquero , M. Desco and A. Santo “Real-time digital timing in positron emission tomography”, *IEEE Trans. Nucl. Sci.*, vol. 55, pp. 2531-2540, 2008.

- [10] A. Mann, et al., "A sampling ADC data acquisition system for positron emission tomography," *IEEE Trans. Nucl. Sci.*, vol.53, no.1, pp.297-300, 2006.
- [11] M. Streun, et al., "Timemark correction for the ClearPET™ scanners," *2005 IEEE Nuclear Science Symposium Conference Record*, vol.4, pp.2057-2060, 2005.
- [12] R. Abbiati, L. Bertossi, A. Geraci, E. Gatti, G. Ripamonti, "High-resolution digital online linear procedure for timing of detected events," *IEEE Trans. Nucl. Sci.*, vol.53, no.3, pp.1270-1274, 2006.
- [13] J.-D. Leroux, et al., "Time Discrimination Techniques Using Artificial Neural Networks for Positron Emission Tomography," *IEEE Trans. Nucl. Sci.*, vol.56, no.3, pp.588-595, 2009.
- [14] Hicham Semmaoui, Marc-André Tétrault, Roger Lecomte, and Réjean Fontaine, "Signal Deconvolution Concept Combined With Cubic Spline Interpolation to Improve Timing With Phoswich PET Detectors," *IEEE Trans. Nucl. Sci.*, vol.56, no.3, pp.581-587, 2009.
- [15] 黃耀慶，正子斷層造影儀資料傳輸與控制介面之實現，國立臺灣海洋大學電機工程學系碩士論文，2008。
- [16] Analog Devices Inc., AD6645 (Rev. D), <http://www.analog.com/>.
- [17] 劉士維，應用於正子斷層造影儀之數位符合偵測處理與資料擷取系統，國立台灣海洋大學電機工程學系碩士論文，2009。