

國立臺灣大學電機資訊學院電子工程學研究所



碩士論文

Graduate Institute of Electronics Engineering

College of Electrical Engineering and Computer Science

National Taiwan University

Master Thesis

基於現場可程式化邏輯閘陣列之

高解析度與高精確度的格式器原型

A High Resolution and High Accuracy

FPGA Formatter Prototype

舒柏騏

Po-Chi Shu

指導教授：黃俊郎 博士

Advisor: Jiun-Lang Huang, Ph.D.

中華民國 103 年 11 月

November, 2014

國立臺灣大學碩士學位論文
口試委員會審定書



基於現場可程式化邏輯閘陣列之
高解析度與高精確度的格式器原型

A High Resolution and High Accuracy
FPGA Formatter Prototype

本論文係舒柏騏君 (R01943099) 在國立臺灣大學電子工程學研究所完成之碩士學位論文，於民國 103 年 11 月 28 日承下列考試委員審查通過及口試及格，特此證明

口試委員：

(指導教授)

系主任、所長：

誌謝



首先感謝我的指導教授黃俊郎博士，自入學以來無數耐心且仔細的指導，不僅為本論文及研究一路指引方向並給予建議，更將對待人處事及作事方法之獨到見解不吝傳授，讓我一生受用不盡。

感謝口試委員洪浩喬教授、陳中平教授百忙之中仍抽空協助審查，並給予許多實用的建議，且叮嚀作研究該有之精神。

感謝同組的宇奕、仲昀、俊宏、家安、齡芸每週和我開會討論彼此研究，互相指導與合作，有你們的幫助才讓研究得以開花結果。

也感謝實驗室的所有學長姐、同學和學弟妹，在我的研究生涯中一起努力，共同琢磨許多知識，生活上也給我很多照顧和幫助，讓我能順利的完成論文及研究，並充實的度過這段時光。

最後感謝我的家人和珮君，謝謝你們長久以來的照顧，讓我有機會能夠進行研究並完成論文。無論遇到任何難關始終都陪在我身邊，協助我解決所有的問題，在背後給我最大的力量，支持我走到最後。

中文摘要



格式器在自動測試機台中是一個關鍵的核心模組，將測試向量、邊緣時序及訊號格式等資訊組合成格式化的測試訊號。目前格式器以 ASIC 設計為主流，FPGA 設計較為少見。一般來說，ASIC 格式器擁有較高的效能，FPGA 格式器則有相對成本較低、設計靈活度較高及不需下線的優勢，但在效能方面普遍不及 ASIC 格式器，仍有發展和進步空間。

本論文提出的格式器原型是以 Xilinx Spartan-6 FPGA 實現，利用時間多工的測試符號生成達成 100 Mbps 的測試符號頻率。透過特別設計的混合式延遲線作為時序電路，精細的控制邊緣放置的時間點，並發展對應之校正機制來提升邊緣放置的精確度。實驗結果顯示，我們提出的 FPGA 格式器原型擁有 20 ps 的邊緣放置解析度，以及 28 ps 的邊緣放置精確度。

關鍵字：自動測試機台、格式器、時序電路、延遲線、現場可程式化邏輯閘陣列

Abstract



The formatter is a key module in automatic test equipment. It formats the input data including test vectors, edge timings and signal formats to the desired test signal. ASIC design is the mainstream for formatter rather than FPGA design. In general, ASIC formatters achieve higher performance while FPGA formatters take advantage of lower cost, higher flexibility and no need for tape-out. Compared to ASIC formatters, there's room for improvement on performance for FPGA formatters.

In this work, the proposed formatter prototype is implemented by Xilinx Spartan-6 FPGA. It achieves 100 Mbps symbol rate by time-multiplexed symbol generation. We design special hybrid delay lines for timing circuits to help placing the timing edges, and develop corresponding calibration mechanism for edge placement resolution and accuracy improvement. Measurement results show that the edge placement resolution and accuracy of this prototype are 20 ps and 28 ps, respectively.

Keywords-test equipment, formatter, timing circuits, delay lines, FPGA

目錄



口試委員會審定書.....	#
誌謝.....	i
中文摘要.....	ii
Abstract.....	iii
目錄.....	iv
圖目錄.....	vii
表目錄.....	ix
第一章 緒論.....	1
1.1 研究動機與目標.....	1
1.2 相關研究.....	2
1.3 研究貢獻.....	4
1.4 論文架構.....	4
第二章 格式器介紹與挑戰.....	5
2.1 自動測試機台.....	5
2.2 格式器.....	6
2.3 可程式化延遲線.....	10

2.4 現場可程式化邏輯閘陣列.....	13
第三章 提出之 FPGA 格式器	18
3.1 架構設計.....	18
3.1.1 系統架構.....	18
3.1.2 格式器架構.....	19
3.2 時間多工與多相位時脈產生器.....	23
3.3 符號產生器與邊緣產生器.....	26
3.4 波形記憶體與編碼方式.....	29
3.5 特性量測與校正機制	32
3.5.1 延遲線特性量測	33
3.5.2 校正機制	34
第四章 以 FPGA 實現之延遲線.....	36
4.1 混合式延遲線	36
4.2 以進位鏈實現粗調節延遲線.....	38
4.3 以查找表實現細調節延遲線.....	40
第五章 實驗與量測結果.....	44
5.1 系統規格與實驗配置	44



5.2 延遲線特性量測與校正結果.....	45
5.2.1 延遲線特性量測結果	45
5.2.2 校正結果.....	47
5.3 邊緣放置解析度分析	51
第六章 結論.....	53
6.1 結論.....	53
6.2 未來研究方向	53
參考文獻.....	55



圖目錄



圖 2-1	自動測試機台方塊圖.....	5
圖 2-2	輸入資訊之格式化.....	7
圖 2-3	常見訊號格式之波形範例.....	9
圖 2-4	可程式化延遲線.....	11
圖 2-5	延遲線輸入與輸出關係.....	11
圖 2-6	FPGA 之設計流程.....	14
圖 2-7	CLB 與 Slice 架構圖.....	15
圖 2-8	CLB 陣列和繞線資源.....	16
圖 3-1	系統架構圖.....	19
圖 3-2	格式器架構圖.....	20
圖 3-3	符號產生器之控制訊號週期與有效輸出區間.....	23
圖 3-4	時間多工之測試符號生成.....	24
圖 3-5	多相位時脈產生器架構圖與生成時脈.....	25
圖 3-6	邊緣放置涵蓋範圍.....	26
圖 3-7	邊緣產生器架構圖.....	27
圖 3-8	波形記憶體之編碼方式.....	30

圖 3-9	波形記憶體架構圖.....	31
圖 3-10	延遲碼對應表示意圖.....	35
圖 4-1	混合式延遲線.....	36
圖 4-2	混合式延遲線之延遲碼.....	37
圖 4-3	進位單元架構圖.....	38
圖 4-4	為粗調節延遲線修正後之設計流程.....	40
圖 4-5	三輸入查找表架構圖.....	41
圖 4-6	以查找表實現之游標尺延遲線.....	42
圖 4-7	為細調節延遲線修正後之設計流程.....	43
圖 5-1	延遲線特性曲線.....	47
圖 5-2	延遲線 A1 經校正後之特性曲線.....	49



表目錄



表 2-1	常見訊號格式表	8
表 2-2	延遲線重要參數表.....	12
表 5-1	延遲線特性量測動態範圍表	46
表 5-2	校正後延遲線參數表.....	48
表 5-3	校正後之延遲線精確度表	50
表 5-4	校正後之延遲線參數表(50 ps 解析度)	52

第一章 緒論



1.1 研究動機與目標

自動測試機台(automatic test equipment, ATE)在現在的半導體產業中是不可或缺的一環。隨著製程不斷演進，積體電路測試的重要性日益俱增，而自動測試機台之功能即為對積體電路進行自動測試，以確保其各項參數有達到設計時的目標規格。

自動測試機台主要是由測試向量記憶體(vector memory)、控制器、格式器(formatter)、接收單元(receiving unit)和接腳介面電路(pin electronics, PE)來執行自動測試的功能。測試向量記憶體儲存需要的資訊，控制器控制機台輸出測試訊號和回收測試結果的時間。格式器本質上是一種波形產生器，負責產生測試時需要的波形，並透過接腳介面電路輸出至待測物(device under test, DUT)。接收單元則同樣透過接腳介面電路回收並比較測試結果，判斷待測物之參數是否滿足設計規格。

格式器在自動測試機台中是一個關鍵的核心模組，負責產生時序精確的測試波形提供測試機台進行交流測試(AC test)。市面上現成的格式器資源稀少，以 ASIC 格式器為主流，FPGA 格式器較為少見。一般來說，ASIC 格式器效能較高但開發成本也較高，FPGA 格式器

則是以較低的成本與其高靈活度，加上不需下線(tape-out)之優點著稱，但在效能方面普遍不及 ASIC 格式器，且仍有進步空間。

不同種類的積體電路需要不同類型的測試機台，以針對測試上的特殊需求加以特殊設計。若加入成本考量的觀點，不同規格的積體電路也需要不同等級的測試機台。中低階規格的積體電路在測試時並不需要太高規格的測試機台，測試成本較低且規格適當的測試機台是更加適合的選擇。

FPGA 格式器即能符合這樣的需求。其較低的開發成本能降低使用測試機台的測試成本，高設計靈活度和不需下線的特性更讓測試機台為測試目標變更設計成為可能。雖然效能不如 ASIC 格式器般頂尖但仍能符合中低階積體電路的測試需求，在效能方面的進步空間也讓 FPGA 格式器值得發展。

因此，本研究之目標為發展針對中低階應用而設計的 FPGA 格式器，希望在現成的 FPGA 上實現擁有格式器基本功能的 FPGA 格式器原型(prototype)。

1.2 相關研究

在格式器的相關研究中，可以先粗略的分成產業界和學術界兩大

類別。業界中大多數的選擇是使用自行研發的 ASIC 格式器，可自行訂定所需之規格，並達到較佳的效能。但需要較高的研發成本和較長的研發時間。




而若想選用現成的格式器晶片，美國模擬器件公司(Analog Devices, ADI)的產品 ADATE207[1]是目前最為廣泛使用的選項。它擁有 100 Mbps 的測試符號頻率(symbol rate)和 39.06 ps 的邊緣放置解析度(resolution)，以及大於 100 ps 的邊緣放置精確度(accuracy)。

在學術界的相關文獻裡，又可以分為以 ASIC 或是使用 FPGA 來實現格式器。一般來說以 ASIC 實現之格式器規格較高，如在[2]中之格式器規格為 800 Mbps 的測試符號頻率及 81 ps 的邊緣放置解析度，而[3]中之格式器規格為 250 Mbps 的測試符號頻率和 20 ps 的邊緣放置解析度。

至於使用 FPGA 來實現格式器則是較為稀少的選擇。[4]中雖然規格較低，只擁有 6 Mbps 的測試符號頻率和 15 ns 的邊緣放置解析度，但實現了完整的自動測試機台之功能。而[5]為目前的研究成果中規格最好的一篇，可達到 100 Mbps 的測試符號頻率、20 ps 的邊緣放置解析度及 74 ps 的邊緣放置精確度，也是本論文最主要之參考對象。

1.3 研究貢獻



本論文根據 FPGA 之特性與構造，設計出合適的格式器架構，並實現一個測試符號頻率 100 Mbps、邊緣放置解析度 20 ps 且精確度 28 ps 之 FPGA 格式器原型。針對此架構所需之規格，以及 FPGA 內部可用之資源，設計其中之關鍵元件延遲線，再搭配修正後的設計流程，將兩種延遲線架構實現於 FPGA 之上。同時配合此架構發展對應之校正機制，克服 FPGA 帶來的規格限制，提升格式器邊緣放置之精確度。

1.4 論文架構

本章說明本論文的研究動機、目標及貢獻，並簡介相關之研究。第二章介紹研究主題，由自動測試機台至格式器和延遲線，並指出面臨的挑戰。第三章為本論文提出之 FPGA 格式器，其中之關鍵元件延遲線則在第四章中詳細介紹。第五章是實驗及量測的結果，而最後一章第六章提出未來的研究方向，並針對本論文作出結論。

第二章 格式器介紹與挑戰



2.1 自動測試機台

自動測試機台係指針對積體電路進行自動測試的一套系統，由電子電路和機械結構組成，並由電腦擔任控制端。為了確認待測物的各項參數是否有達到設計時的目標規格，自動測試機台會對待測物送出測試訊號，再回收測試結果並分析。目的是區分出功能不正常的晶片，或是對不同效能範圍的晶片進行分類(binning)。

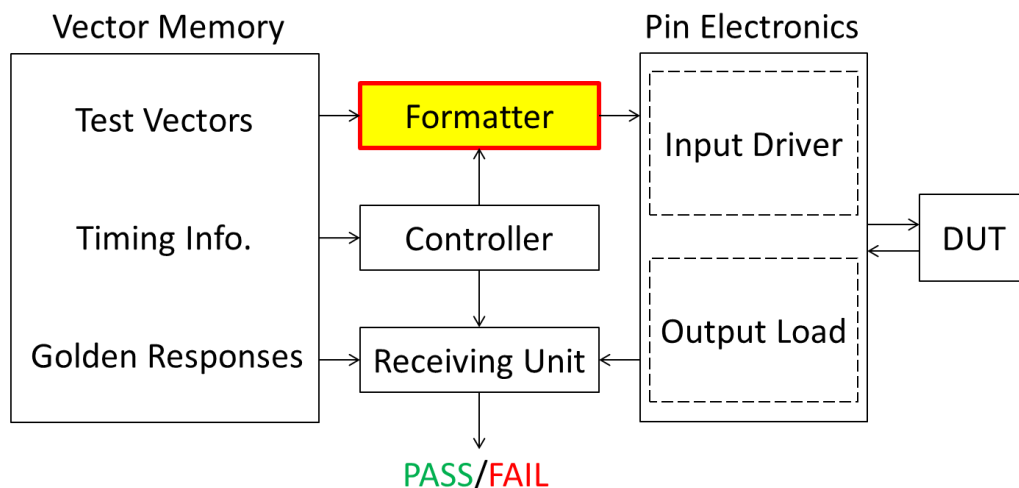


圖 2-1 自動測試機台方塊圖

自動測試機台主要是由測試向量記憶體(vector memory)、控制器、格式器(formatter)、接收單元(receiving unit)和接腳介面電路(pin electronics, PE)來執行自動測試的功能。測試向量記憶體儲存測試訊號的資訊和預期的測試結果，通常越高規格的測試機台會有越大的記

記憶體容量，以減少重新載入的次數。控制器則安排格式器和接收單元運作的時間，在正確的時間回收測試結果。



格式器負責產生時序精確的測試波形，將記憶體中預存的測試向量(test vector)、時序設定(timing set)及訊號格式(signal format)組合成一組格式化的測試符號(symbol)，透過接腳介面電路中的驅動電路(driver)傳送到待測物，以進行測試。

接收單元能在控制器指定的時間點判斷測試結果，並與預期的測試結果比較，分析待測物的各項參數是否達到設計時的目標規格。接腳介面電路則是作為測試機台和待測物間的連接介面，負責包括轉換訊號準位(signal level)、電流負載(current load)及比較電壓等功能。

介紹完自動測試機台後，可以發現格式器是其中關鍵的核心模組，在下一節中將會詳細介紹格式器的基本功能以及其重要的規格。

2.2 格式器

格式器(formatter)在自動測試機台中負責產生時序精確的測試波形，提供格式化的訊號讓自動測試機台進行交流測試，驗證待測物是否能在正確的時間點運作所設計的功能。

格式器本質上是一種波形產生器，會將輸入的資訊組合成一組格



式化的訊號來輸出。這些輸入的資訊包括測試向量(test vector)、邊緣
時序(edge timing)及訊號格式(signal format)，而輸出的波形則由一連
串的測試符號(symbol)所組成。

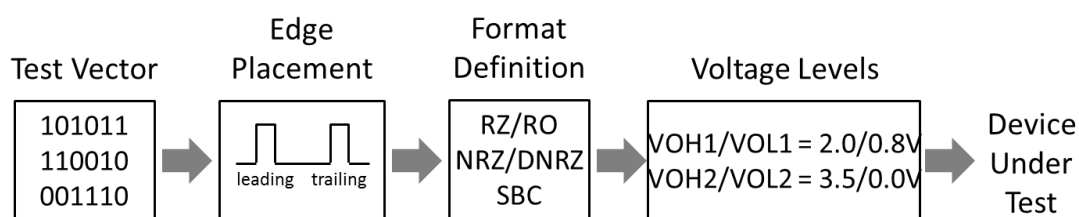


圖 2-2 輸入資訊之格式化

測試向量指每個測試週期(cycle)中要輸出給待測物之邏輯值，通常儲存在測試向量記憶體中。邊緣時序則是每個測試週期中訊號轉態(transition)的時間點，大部分的格式器都支援每個測試週期兩個以上的邊緣時序。訊號格式針對不同的測試目的而設計，每個測試週期中可以使用不同的訊號格式。表 2-1 即為一些常見的訊號格式。

表 2-1 常見訊號格式表

不歸零 (NRZ)	不歸零(non-return-to-zero)格式在每個測試週期中直接輸出測試向量，僅在測試週期之起始點需要一個邊緣時序。
延遲不歸零 (DNRZ)	延遲不歸零(delayed-non-return-to-zero)格式將不歸零格式在每個測試週期中延遲一段時間輸出，同樣需要一個邊緣時序。
回歸零 (RZ)	回歸零(return-to-zero)格式僅在測試向量為邏輯一的測試週期輸出一個正脈衝，其餘時間皆輸出邏輯零。在一個測試週期中需要兩個邊緣時序。
回歸一 (RO)	回歸一(return-to-one)格式僅在測試向量為邏輯零的測試週期輸出一個負脈衝，其餘時間皆輸出邏輯一。和回歸零格式一樣需要兩個邊緣時序。
補數環繞 (SBC)	補數環繞(surrounded-by-compliment)格式在每個測試週期中以其補數包圍測試向量來輸出，加上測試週期的起始點共需要三個邊緣時序。

不同的訊號格式適合不同特性的測試目的，需要的邊緣時序數量也不盡相同。一般來說測試時脈腳位時會選擇回歸零格式，回歸一格式適合測試負緣觸發的控制腳位，而資料腳位通常會以補數環繞格式產生符合時序最低標準的波形，用來測試建立時間(setup time)及保持時間(hold time)等交流參數(AC parameter)。

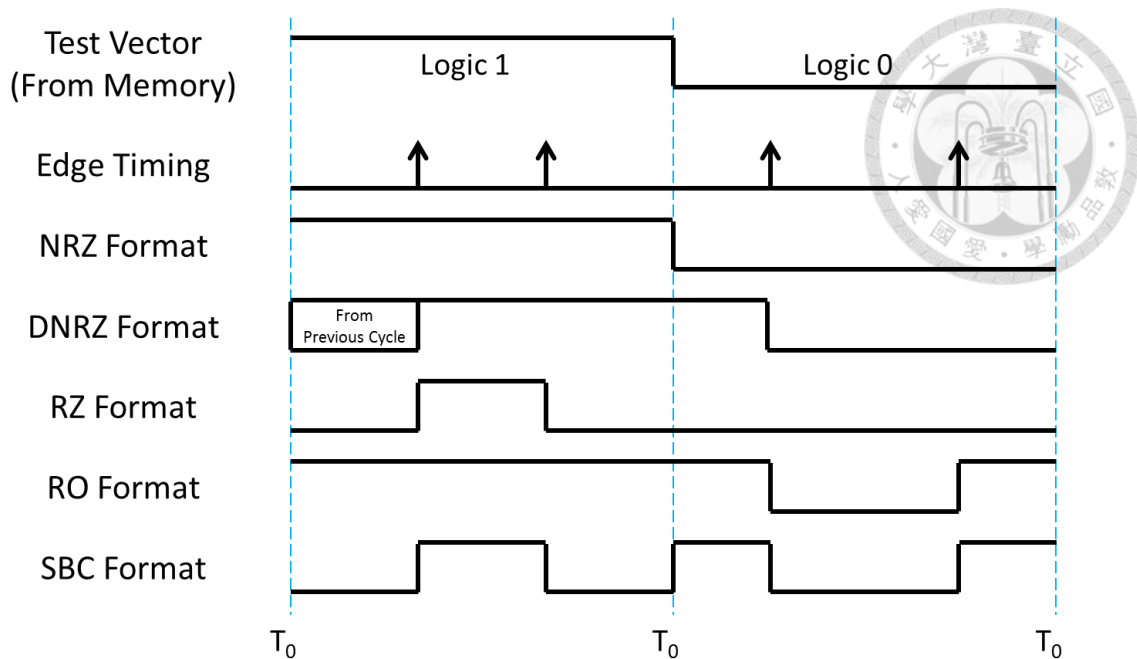



圖 2-3 常見訊號格式之波形範例

圖 2-3 為常見訊號格式的波形範例，從圖中可以發現補數環繞格式在一個測試週期中需要最多的邊緣時序，加上測試週期的起始點一共需要三個邊緣時序。若格式器欲支援補數環繞訊號格式，設計時必須要能在一個測試週期中產生至少三個轉態邊緣(transition edge)。

在一個測試週期中由測試向量、邊緣時序及訊號格式所組合而成的格式化訊號稱為測試符號(symbol)，格式器的輸出即為一連串的測試符號構成之測試波形。這些測試符號的輸出頻率稱為測試符號頻率(symbol rate)，是格式器的重要規格之一，其倒數即為測試週期的長度，又稱為測試符號長度(symbol length)。

為了在測試符號中依照指定的邊緣時序產生轉態邊緣，格式器必



須要有邊緣放置(edge placement)的功能。邊緣放置的最小刻度稱為邊緣放置解析度(edge placement resolution)，而最大的偏移量則稱為邊緣放置精確度(edge placement accuracy)，在格式器中都是最重要的規格。

格式器的測試符號頻率(即測試週期的長度)是由系統時脈的頻率來決定，若以 FPGA 實現通常會被時脈頻率的上下限制住。而提升邊緣放置的解析度和精確度是很大的挑戰，需要在格式器中有精密的時序電路。如[5]中所提到的，單純以同步數位電路來實現的話，即使高如 1 GHz 的時脈頻率也僅能達到 1 ns 的解析度。若想要達到次奈秒(sub-nanosecond)等級的解析度，使用延遲線(delay line)來控制邊緣放置的時間點是一個聰明的做法，而其中最困難之處即在於將延遲線實現於 FPGA 之上。

在下一節中即要介紹可程式化延遲線(programmable delay line)，以及將其實現於 FPGA 上所要面臨的挑戰。

2.3 可程式化延遲線

延遲線(delay line)是一種利用傳遞延遲(propagation delay)將輸入訊號延後一段時間後再輸出的電路，通過延遲線的訊號將只在時間上有變化而邏輯值不變。若一條延遲線其傳遞延遲的時間長度是可以控

制的，就稱之為可程式化延遲線(programmable delay line)。

一般來說，可程式化延遲線擁有三個埠(port)，分別為輸入埠、輸出埠和控制埠。控制埠負責決定延遲線目前的延遲值(delay value)，而訊號由輸入埠輸入後經過此延遲值再由輸出埠輸出。

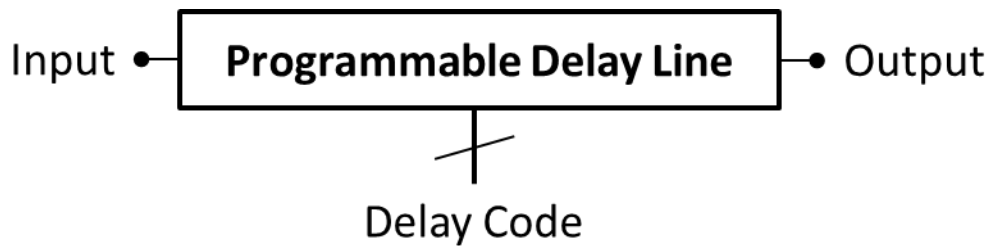


圖 2-4 可程式化延遲線

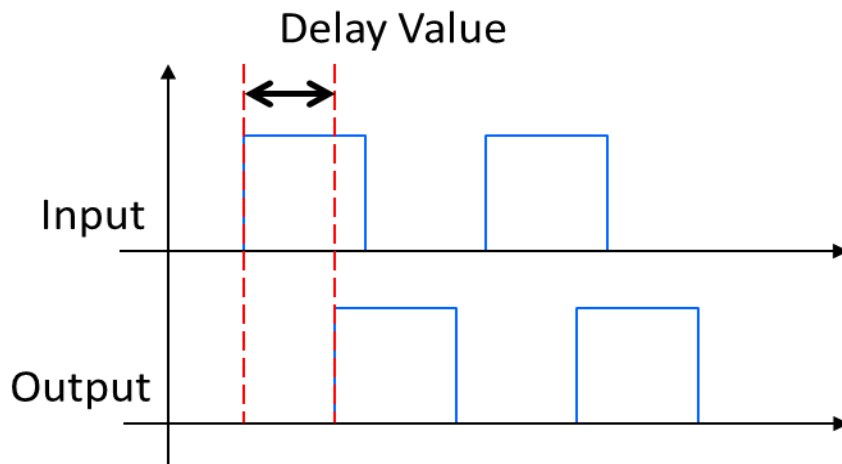


圖 2-5 延遲線輸入與輸出關係

在這裡我們定義由控制埠輸入的控制訊號為延遲碼(delay code)，每一個延遲碼將會對應到一個延遲值。一條 N 位元的延遲線代表它擁有 N 位元的延遲碼，也就是擁有二的 N 次方個延遲值。要評估一條延遲線的效能，有一些重要的參數可以做為參考，以下用表 2-2 來

列出並解釋這些參數的意義：



表 2-2 延遲線重要參數表

單調性 (Monotonicity)	隨著延遲碼的遞增，延遲值是否隨之遞增。
最小延遲 (Latency)	延遲線最小的傳遞延遲，一般來說越小越好。
動態範圍 (Dynamic range)	延遲線的可程式化範圍，又稱為跨距(span)。即最大延遲和最小延遲之差值。
解析度 (Resolution)	延遲值的最小刻度，又稱為步距(step size)。
線性度 (Linearity)	延遲線線性的程度，通常以積分/差分非線性誤差(INL/DNL)來呈現。

大部分的延遲線在設計時都希望具有單調性，讓延遲碼調升時延遲值也能隨之增加。最小延遲是延遲線的一個重要規格，因訊號需通過延遲線故傳遞延遲不可能為零，但一般來說希望能越小越好。動態範圍和最小延遲之間是一種接近權衡(trade-off)的關係，透過串接延遲線來增加動態範圍時也會造成其最小延遲的增加。線性度是延遲線線性的程度，通常在評估時會以積分/差分非線性誤差(integrated non-linearity, INL/differential non-linearity, DNL)來呈現。

格式器在邊緣放置時需要精密的時序電路，高解析度的延遲線是一種適合的解答。在格式器中延遲線負責精確的控制邊緣放置的位置，故動態範圍代表邊緣放置的範圍、解析度主宰邊緣放置解析度，而線

性度的優劣則影響邊緣放置的精確度。



若以 FPGA 來實現延遲線，受到製程偏移(process variation)及繞線不確定性(uncertainty)的影響，其線性度甚至單調性皆不易控制。

由於設計工具在佈局(layout)時最佳化之目標通常為關鍵路徑延遲(critical path delay)或面積，故在 FPGA 上實現高線性度的延遲線是很大的挑戰。

2.4 現場可程式化邏輯閘陣列

現場可程式化邏輯閘陣列(field programmable gate array, FPGA)是一種被設計為可以不斷重新組態(reconfigure)的數位積體電路。不同於 ASIC 設計需要經過下線製造，FPGA 透過每一次的重新組態即可立即實現不同功能的電路，提供一種研發成本較低的設計流程(design flow)，以及修改設計或增加電路功能的靈活性。

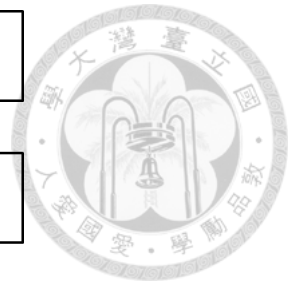
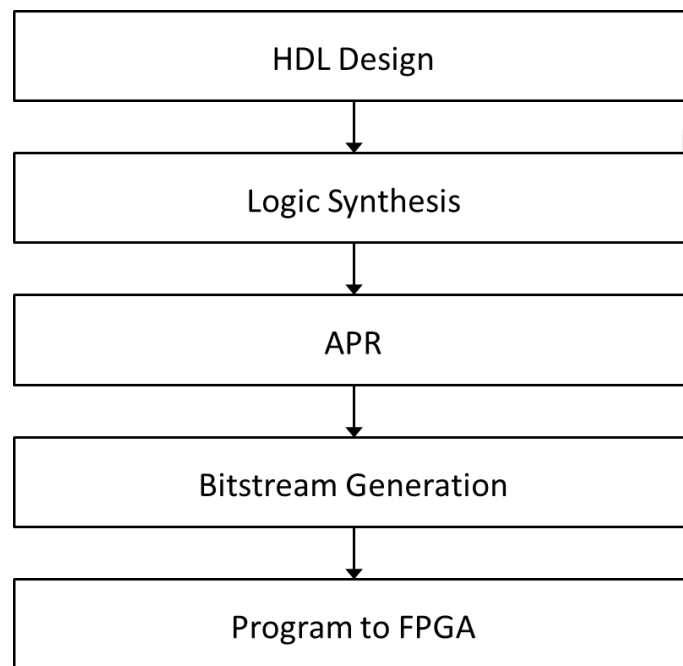


圖 2-6 FPGA 之設計流程

FPGA 內部的構造主要是由基本邏輯元件、繞線資源和記憶體所組成。基本邏輯元件實現各種邏輯電路，再透過繞線資源決定邏輯元件間的連接方式，以實現設計之電路。這些邏輯元件實現的邏輯電路和繞線的方式即為所謂的組態(configuration)，會以位元串(bitstream)的形式被儲存在記憶體中。

以本論文使用的 FPGA – Xilinx Spartan-6 XC6SLX75 為例，其基本邏輯元件以 CLB (configurable logic block) 為單位，一個 CLB 內有兩個 Slice，由組合邏輯(combinational logic)和儲存元件組成。組合邏輯的部分包含六輸入(6-input)的查找表(look-up table, LUT)和進位鏈(carry chain)，儲存元件則是使用 D 型正反器(D-type flip-flop)。

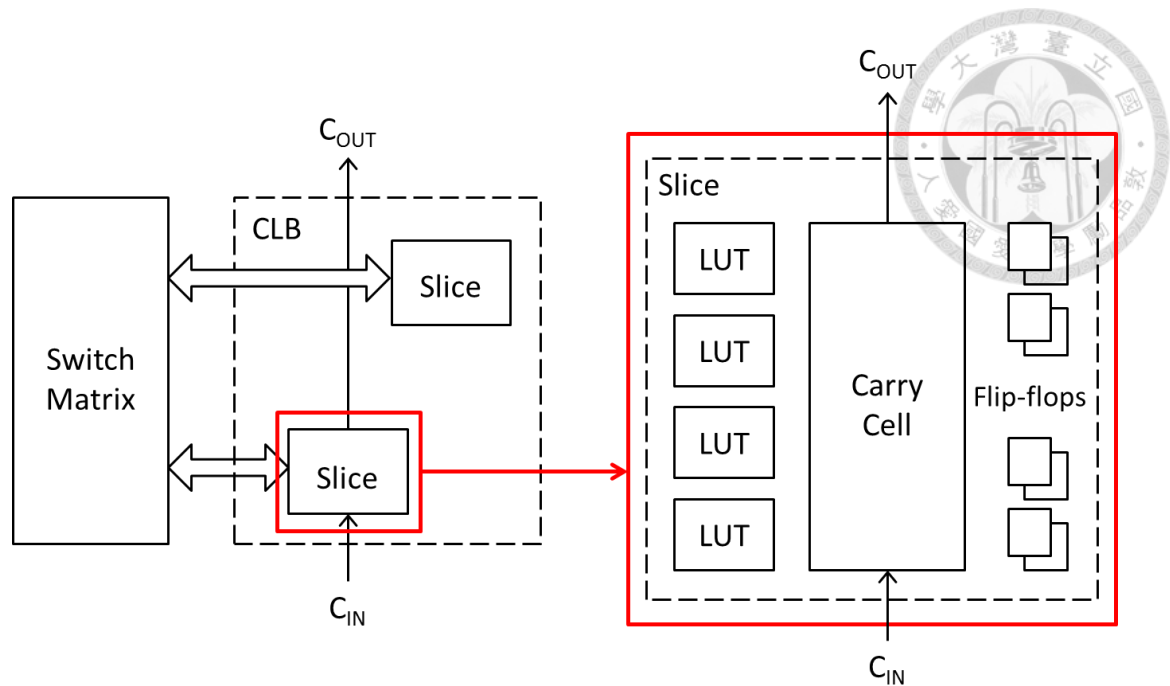


圖 2-7 CLB 與 Slice 架構圖

每個 Slice 中有四個查找表，透過組態設定其真值表(truth table)後可以實現任意的布林函式。而在每個 CLB 的兩個 Slice 中會有一個擁有進位單元(carry cell)，並以專屬的路徑和上下相鄰的進位單元串接在一起形成進位鏈。透過進位鏈而不經過外部繞線來傳遞訊號，可以節省實現加法器時傳遞進位的時間。另外，一個 Slice 中會有八個 D 型正反器，其中四個和查找表的輸出端相連接，用來實現循序邏輯(sequential logic)，另外四個則可以連接到許多訊號，可做為較自由的用途。

其繞線資源則是以 Switch Matrix 為單位，負責連接 Slice 和 CLB 外部的訊號路徑。CLB 在 FPGA 中是以二維陣列的方式排列，每一

個 CLB 旁皆配置有一個 Switch Matrix。在 CLB 陣列之間有格狀的、縱橫交錯的訊號路徑，每一個 CLB 皆透過自己的 Switch Matrix 連接上外部的訊號路徑。透過選擇 Switch Matrix 內不同的路徑，可以連接到對應的外部訊號路徑，再和不同的 Slice 及 CLB 連接。

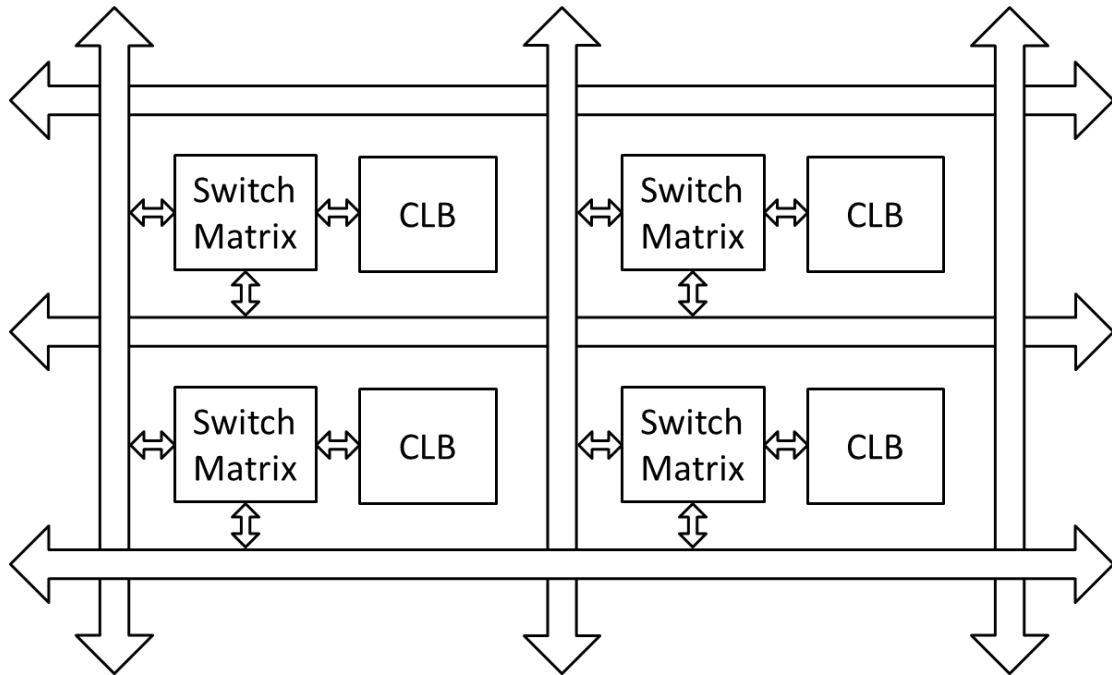


圖 2-8 CLB 陣列和繞線資源

雖然 FPGA 內部擁有大量重複且規則排列的邏輯元件和繞線資源，但和 ASIC 設計一樣，仍然存在著製程偏移的問題。當相同的電路實現在不同的位置或不同的 FPGA 上時，會呈現出不同的特徵，對於想在 FPGA 上實現精確的時序電路是非常大的挑戰。

另一個挑戰則是繞線的不確定性(uncertainty)。以 FPGA 實現時序電路時，元件間的繞線所造成的延遲是設計時無法準確預估的，卻

會嚴重影響電路實際上的效能。這些因 FPGA 而造成的基本問題，都是在設計 FPGA 格式器時需要考慮和解決的，在下一章中將會介紹本論文提出的 FPGA 格式器，以及解決問題的方法。



第三章 提出之 FPGA 格式器



3.1 架構設計

除了格式器本身之外，系統中還包含電腦端的特性量測與校正機制，以及同樣實現在 FPGA 上的 PXI 控制器和匯流排。本節將會先介紹系統中格式器以外的部分，再針對格式器架構的設計進行說明。

3.1.1 系統架構

在 FPGA 格式器原型的完整系統架構中，分成電腦端和 FPGA 端兩個部分，兩端之間使用 PXI 做為溝通介面。電腦端係指電腦及其執行的軟體，負責驅動 PXI 與控制存取匯流排上掛載的暫存器、控制特性量測和執行校正機制、並提供以 LabView 架設的圖形化介面作為溝通平台。FPGA 端則包含實現於其上的格式器、PXI 控制器和匯流排，以及掛載在匯流排上可供存取的暫存器。

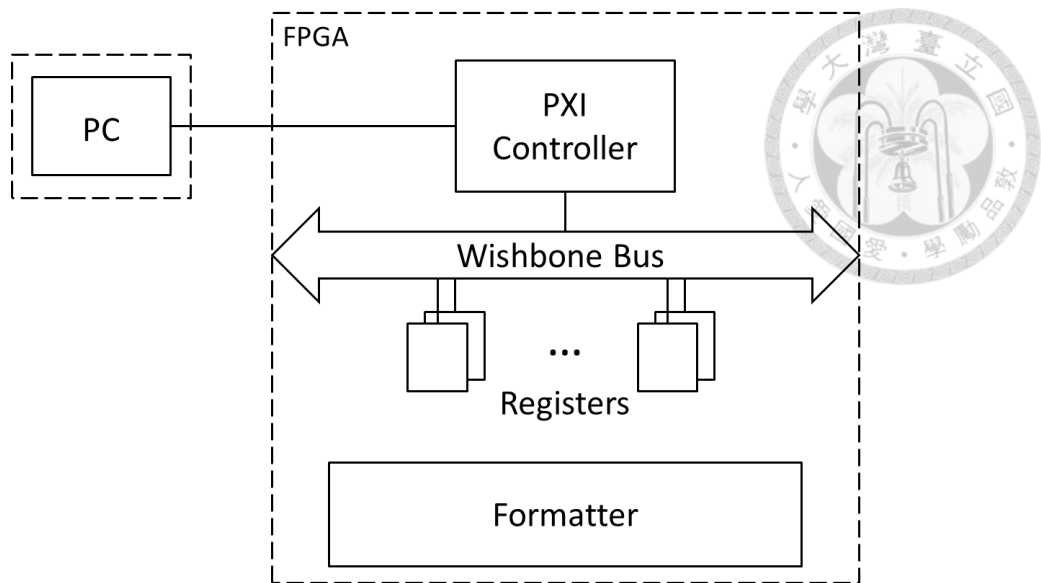


圖 3-1 系統架構圖

匯流排上掛載的暫存器以每 32 個位元為單位，每個單位配有一個位址，透過電腦端的圖形化介面指定位址即可直接進行讀寫。這些儲存在暫存器內的資料，經過格式器中的波形記憶體下載後會成為格式器的控制訊號，其編碼方式將會在[3.4 波形記憶體與編碼方式]中介紹。如[5]中所提到，此架構最大的優點為其擴充性，當需要控制的硬體增加時，只需要掛載更多的暫存器並配置新的位址即可。

介紹完格式器以外的部分後，接下來將針對格式器架構的設計進行說明。

3.1.2 格式器架構

格式器是這個系統中的主角，同時也是自動測試機台中不可或缺的核心模組。在本節中將針對格式器架構的設計概念來說明，每一塊

模組詳細的實作方法和技術將在接下來的各節中再陸續詳述。

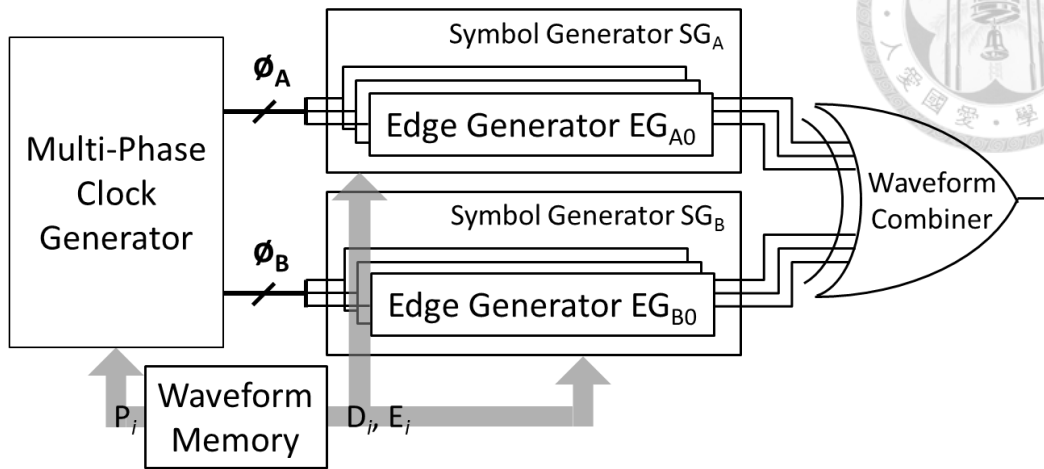


圖 3-2 格式器架構圖

如圖 3-2 所示，本論文提出的 FPGA 格式器係由波形記憶體 (waveform memory)、多相位時脈產生器 (multi-phase clock generator)、兩組符號產生器 (symbol generator, SG) 及波形合成器 (waveform combiner) 所構成。

波形記憶體儲存使用者輸入的波形，並控制時脈產生器和符號產生器去產生需要的測試波形。多相位時脈產生器負責產生不同相位的時脈送給符號產生器，支援兩組符號產生器交錯的工作週期，並與符號產生器內的延遲線合力完成邊緣放置的任務。

兩組符號產生器的架構相同，為支援在一個測試週期中提供三個邊緣時序的需求，各由三個邊緣產生器 (edge generator, EG) 組成。符號產生器在一次的工作週期中會產生一個測試符號，並以時間多工

(time-multiplexed)的方式輪流運作。產生出的測試符號再經由波形合成器合成後即成為需要的測試波形，也就是格式器之輸出。



在實現格式器的主要功能 – 邊緣放置時有三大核心目標：提高解析度、擴大可放置範圍和提升精確度。提高邊緣放置解析度意即縮小邊緣可移動的最小步距，需要擁有可以細微調整邊緣時序的能力。而擴大邊緣可放置的範圍則提升了實現的難度，使得需求進階為在大範圍內都要能夠細微的調整邊緣時序。

為達成這樣的需求，設計中使用了分段式(segmented)的概念，將邊緣放置的步距分成三種不同的大小。最大的步距負責延展邊緣放置的範圍，需要的高解析度則由最小的步距提供。這樣的設計在達成邊緣放置需求的同時，能夠有效降低延遲線之最小延遲及使用的硬體資源，更可以和提出的校正機制互相配合，完成提升精確度的目標。

多相位時脈產生器和符號產生器在此架構中是最關鍵的核心模組，負責實現邊緣放置，是格式器架構設計中的重點。在邊緣放置的三種步距中，最大的步距是由多相位時脈產生器提供的時脈切換而成。除了供給兩組符號產生器相位相反的時脈之外，時脈產生器同時也在每個測試週期內產生四種不同相位的時脈，將測試週期平均的分成四等份。透過輸出不同相位的時脈給邊緣產生器作為輸入，切換長度為

四分之一個測試符號長度(symbol length)的最大步距。



另外兩種步距則是由邊緣產生器中的延遲線所提供，在延遲線的設計上同樣也使用了分段式的技巧。兩組符號產生器中各有三個邊緣產生器，各自在輪到自己所屬的工作週期時決定是否產生邊緣以及邊緣產生的時間。關於多相位時脈產生器和符號產生器更詳細的設計概念與實作技術，將會分別在[3.2 時間多工與多相位時脈產生器]和[3.3 符號產生器與邊緣產生器]中介紹。

波形合成器在格式器的架構中是一個有趣的設計，事實上它僅是一個由 FPGA 內之查找表(LUT)實現的互斥或(XOR)邏輯閘。其輸入是兩組符號產生器的輸出，也就是時間上交錯的測試符號，其輸出則直接連接到格式器的輸出。由於互斥或邏輯閘的特性，輸出會隨著任一輸入的變化而變化，讓其適合用來作為將時間上錯開的訊號合成之用途。這個簡單的設計會將因時間多工而分工產生的測試符號合成，形成格式器產生的測試波形。

介紹完格式器架構的設計概念之後，在接下來的各節中將陸續介紹每一塊模組詳細的實作方法和技術。

3.2 時間多工與多相位時脈產生器



格式器的輸出波形是由連續的測試符號所組成，對於符號產生器之需求為不間斷的產生測試符號。因一組符號產生器無法不間斷的產生測試符號，需要由多組符號產生器分工輪流產生連續的測試符號，這樣的運作方式稱為時間多工(time-multiplexed)的符號生成。

若符號產生器之控制訊號週期和有效輸出區間的長度相同，一組符號產生器就能不間斷的產生測試符號。控制訊號至少需在經過延遲線之最小延遲加上動態範圍的時間後才能再次更新，而延遲線之動態範圍即為有效輸出區間。因延遲線存在不為零的最小延遲(latency)，故控制訊號週期必定大於有效輸出區間，一組符號產生器在產生測試符號後需經過一段時間準備，才能再產生下一個測試符號。

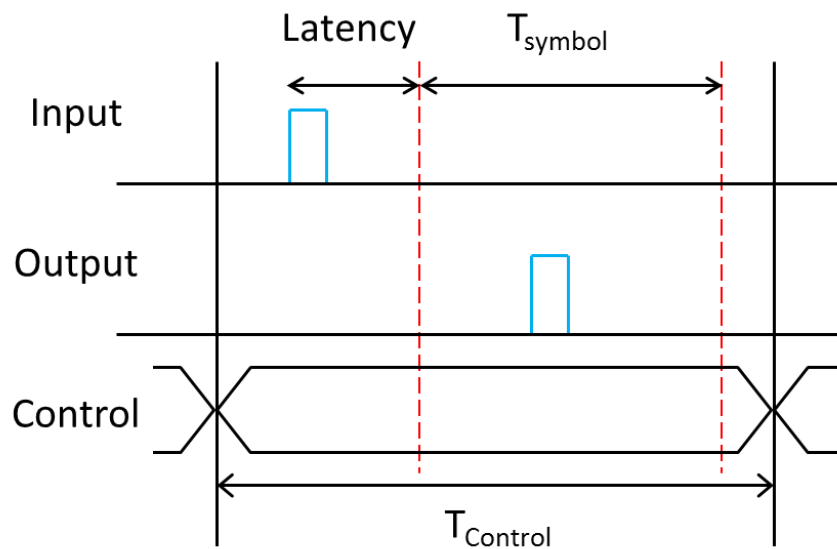


圖 3-3 符號產生器之控制訊號週期與有效輸出區間

為了使用最低需求的兩組符號產生器，我們以降低最小延遲為目標設計延遲線。在提出的格式器中，兩組符號產生器輪流運作產生連續的測試符號，並在對方產生測試符號時更新控制訊號。

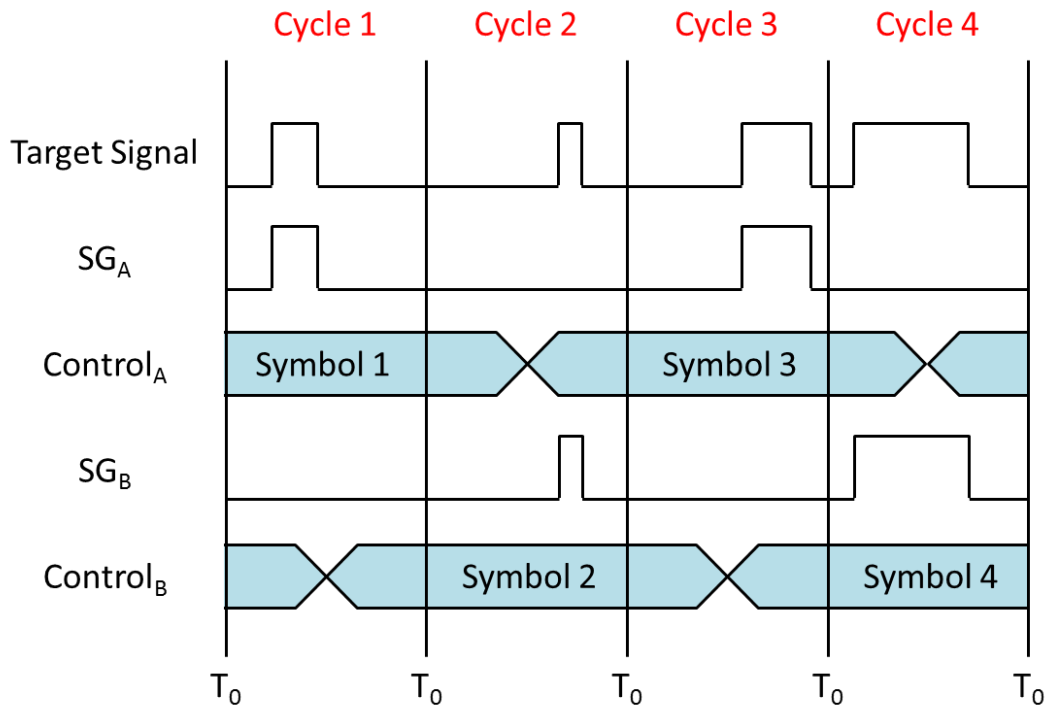


圖 3-4 時間多工之測試符號生成

多相位時脈產生器的存在主要有兩個目的：支援時間多工的符號生成以及提供邊緣放置的最大步距。時脈產生器提供兩組符號產生器兩套頻率為測試符號頻率(symbol rate)的一半、相位相反的時脈，其中每套包含四種相位差各四十五度之時脈，將所屬之測試週期分成四等份，提供四分之一測試符號長度(symbol length)的最大步距。

兩套相位相反的時脈讓兩組符號產生器能交替運作產生連續的



測試符號，而每套四種時脈的設計讓延遲線只需涵蓋四分之一測試符號長度的動態範圍，降低延遲線的最小延遲，以達成使用最低需求兩組符號產生器的目標。

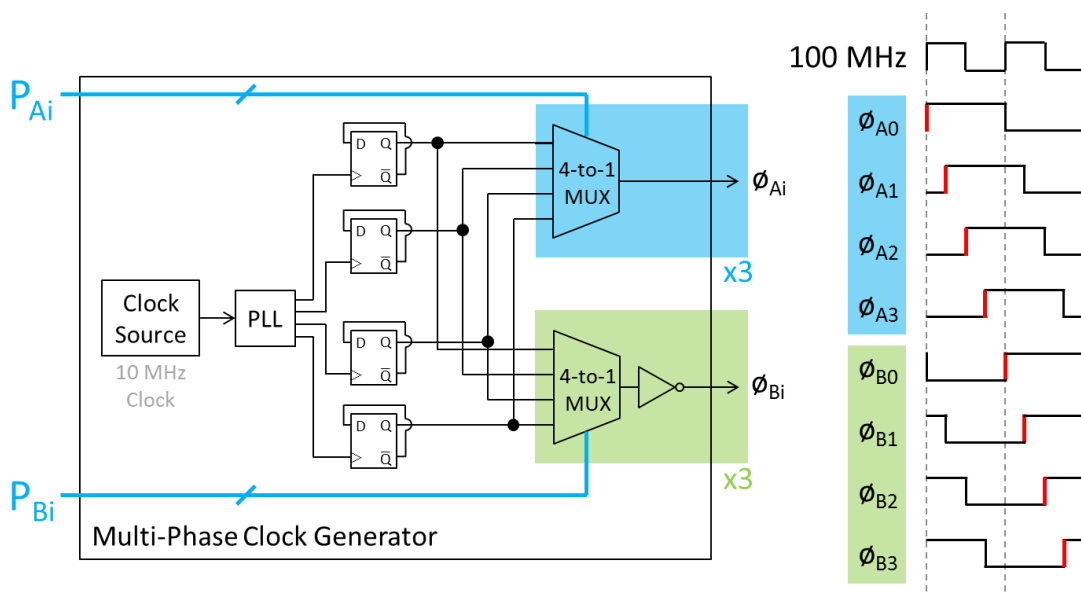


圖 3-5 多相位時脈產生器架構圖與生成時脈

如圖 3-5 中所示，時脈產生器之輸入為其控制訊號－相位選擇碼 P，輸出為六組可調相位時脈和兩組固定相位時脈(圖中省略)。六組可調相位時脈分成兩種，分別可以輸出兩套相位相反的時脈，每組可調相位時脈都有四種不同相位的時脈可選擇，由相位選擇碼 P 決定輸出之時脈。兩組固定相位時脈則是提供波形記憶體運作之用。

多相位時脈產生器所有的時脈皆來自 PXI 提供的 10 MHz 時脈源 (clock source)。經由鎖相迴路(phase-locked loop, PLL)將其升頻至測試符號頻率 100 MHz 並生成四種相位差各九十度的時脈後，送至四顆



由 D 型正反器實作之除頻器，產生四種頻率為測試符號頻率一半、相位差各四十五度之時脈。

這四種時脈由六組四對一多工器(4-to-1 multiplexer)提供給相位選擇碼 P 選擇，其中一半的多工器輸出端接有反相器(inverter)讓時脈反相，產生兩套一共八種不同相位之時脈，作為邊緣產生器之輸入。

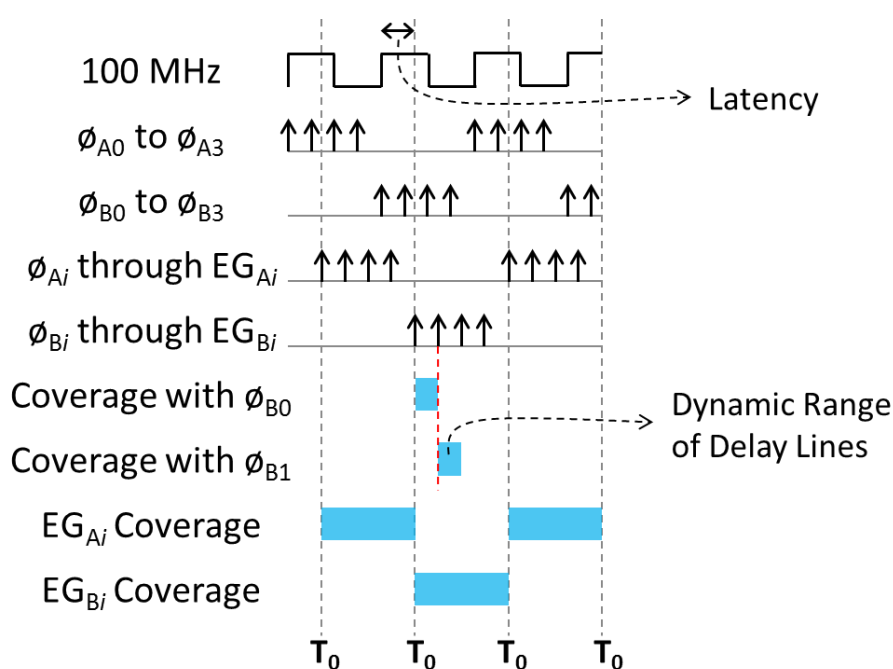


圖 3-6 邊緣放置涵蓋範圍

3.3 符號產生器與邊緣產生器

高解析度和高精確度的邊緣放置是研究中最大的挑戰，而實現的關鍵就在於符號產生器(symbol generator, SG)與邊緣產生器(edge generator, EG)的設計。符號產生器的功能就如同它的名字，是產生需

要的測試符號。為滿足在一個測試週期中提供三個邊緣時序的需求，一組符號產生器是設計由三個邊緣產生器所組成。



邊緣產生器的功能則是在指定的時間點產生轉態邊緣(transition edge)，也就是實際的去實現邊緣放置。為達成次奈秒(sub-nanosecond)等級甚至更高的邊緣放置解析度，我們在邊緣產生器的設計中使用了延遲線來控制邊緣時序。

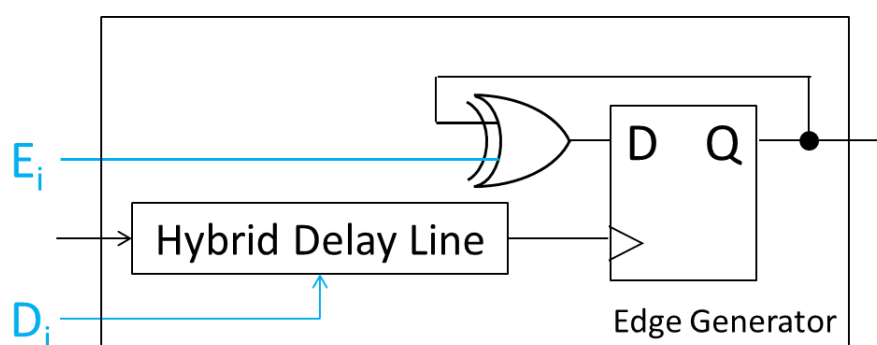


圖 3-7 邊緣產生器架構圖

如圖 3-7 所示，邊緣產生器是由延遲線、互斥或邏輯閘和 D 型正反器構成。輸入訊號包括由時脈產生器的可調相位時脈提供之時脈、邊緣產生器的開關 E 及延遲線的延遲碼 D，而輸出則為指定之時間點上的轉態邊緣，或是維持和前一個週期相同的狀態(state)。

我們將 D 型正反器之輸出經過一個互斥或邏輯閘後接回輸入形成回授(feedback loop)，在每一次正反器被觸發(trigger)時，若互斥或邏輯閘的另一個輸入為邏輯一，正反器之輸出即會翻轉(flip)，反之則

會維持不變。此輸入即為邊緣產生器之開關 E，控制邊緣產生器在這個測試週期中是否產生邊緣。



產生邊緣的時間點則是由觸發正反器之時脈來決定。我們讓時脈產生器提供之時脈通過延遲線後再送至正反器作為觸發時脈，由時脈相位和延遲線提供邊緣放置的三種步距。延遲線之延遲碼 D 控制延遲線之延遲值，和相位選擇碼 P 共同決定邊緣產生器產生邊緣的時間，也就是邊緣放置的位置。

在我們的設計中，每個邊緣產生器能進行邊緣放置的範圍都涵蓋整個測試週期，因符號產生器是由三個邊緣產生器組成，意即在每個測試符號中都支援三個全範圍的邊緣時序。這樣的架構除了滿足訊號格式的需求之外，也為邊緣放置的精確度帶來了意想不到的優勢。

在大多數的訊號格式裡，一個測試週期並不需要多達三個邊緣時序，在需要三個邊緣時序的訊號格式裡，其中一個又固定存在於測試週期之起始點。故在每個測試符號中，我們皆以三個全範圍的邊緣時序去達成最多兩個動態的邊緣放置目標，有充裕的條件可以選擇精確度較高之邊緣產生器來產生邊緣，提升邊緣放置之精確度。

雖然邊緣產生器需要涵蓋整個測試符號長度的範圍，因採用分段式(segmented)的設計概念之故，邊緣產生器中的延遲線只需要涵蓋四

分之一測試符號長度的動態範圍即可。這樣的設計節省了延遲線的串接長度，降低最小延遲以達到減少符號產生器組數的目標。



為提供邊緣放置需要的高解析度，延遲線本身同樣也採用分段式設計概念，由兩種不同步距大小的延遲線組合而成。關於延遲線之設計及其面臨的問題與解決方式，將在[3.5 特性量測與校正機制]以及第四章中介紹。

在[5]的設計中，符號產生器在三個邊緣產生器的輸出端會先利用互斥或邏輯閘將邊緣合併成為測試符號，作為符號產生器之輸出。故其符號產生器是由三個邊緣產生器和一個三輸入之互斥或邏輯閘組成。而在本論文的設計中，將此互斥或邏輯閘併入波形合成器，僅使用一個六輸入查找表(LUT)即可完成實作。

3.4 波形記憶體與編碼方式

波形記憶體(waveform memory)在格式器中負責儲存待測物需要或使用者輸入的測試波形。在本論文提出的設計中，波形記憶體是以循環移位暫存器(circular shift register)來實現。選用循環移位暫存器的目的是為了讓送出的測試波形能夠重複出現形成週期波，方便示波器擷取與測量。

波形記憶體內共有六串循環移位暫存器，以三串為一組，分別儲存兩組符號產生器內共六個邊緣產生器、以及對應的時脈產生器之控制訊號。這些控制訊號讓時脈產生器與符號產生器去產生需要的測試符號，包括決定每個邊緣是否產生以及這些邊緣產生的時間。

測試波形在格式器的原型中被設計成以控制訊號的方式來儲存。先將波形拆解成每個測試符號中需要的轉態邊緣及其邊緣時序，再把這兩種資訊以邊緣是否產生及產生之時間的形式化為一組控制訊號，以每個邊緣產生器為單位去分配並儲存至波形記憶體中。

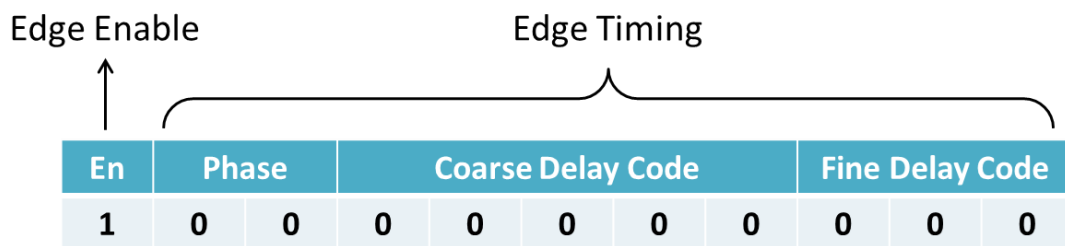


圖 3-8 波形記憶體之編碼方式

圖 3-8 為波形記憶體中使用的編碼方式，以邊緣之存在與否及其時間位置來描述一個邊緣。從最高有效位(most significant bit, MSB)開始依序為邊緣產生器的開關 E、相位選擇碼 P 及延遲碼 D。由最高有效位決定邊緣產生器在這個測試週期中是否產生邊緣，其餘的十個位元則可視為代表時間位置的一組延遲碼，以三種步距的組合來描述邊緣產生的時間。



這樣的編碼方式使用 11 個位元來描述一個邊緣，因設計中是由三個可移動的邊緣(不一定要被產生)來構成一個測試符號，意即使用了 33 個位元來描述一個測試符號。

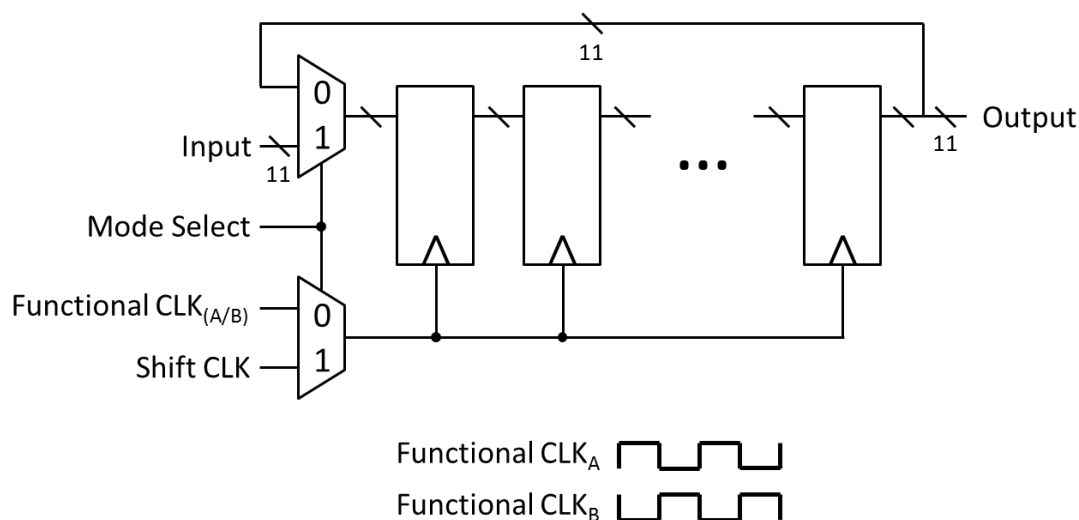


圖 3-9 波形記憶體架構圖

波形記憶體有兩種操作模式，分別為移位模式(shift mode)和工作模式(functional mode)。在移位模式時，波形記憶體從電腦端下載波形，將控制訊號一組組存入。而在工作模式時，將儲存之控制訊號依序送出，以產生儲存之波形。

若將波形記憶體視為一個三維的儲存空間，其長度為一串循環移位暫存器中暫存器的數量，寬度為每一級暫存器的位元數，高度則為循環移位暫存器的數量。

移位暫存器的串接長度取決於波形記憶體欲儲存的測試符號數

量，想儲存多少數量的測試符號就串接多少暫存器即可。儲存的測試符號會在串接長度個測試週期後重複循環出現，形成週期波。



而暫存器的位元寬度則是由時脈產生器和符號產生器需要的控制訊號數量決定，每一級暫存器中儲存的是一個邊緣的資訊。目前的編碼方式是以 11 個位元來描述一個邊緣，位元寬度即為 11 個位元。

和對應的邊緣產生器相同，六串移位暫存器同樣分為兩組，每三串為一組。兩組移位暫存器由相位相反的時脈推動，在非工作週期時將控制訊號送入所屬的時脈產生器與邊緣產生器，以設定下一個工作週期需要的組態。

3.5 特性量測與校正機制

製程偏移和繞線的不確定性是 FPGA 延遲線無法消除的誤差來源，而延遲線的不完美會影響邊緣放置的精確度。為提升邊緣放置精確度，本論文提出邊緣產生器中特別設計之混合式延遲線(hybrid delay line)，並針對此設計發展搭配的校正機制，將無法消除之基本問題轉換為可利用的特性。本節將會先介紹延遲線之特性量測，再說明所發展之校正機制。



3.5.1 延遲線特性量測

延遲線的效能在格式器中主宰著邊緣放置的解析度和精確度。為了解系統中延遲線之特性，並提供給校正機制作校正之用，我們必須對延遲線進行特性量測(characterization)。

FPGA 延遲線有許多無法消除的誤差來源，包括 FPGA 之製程偏移(process variation)、繞線的不確定性(uncertainty)以及佈局時最佳化目標的不相容。這些誤差來源帶來延遲線的非線性度，造成邊緣放置的不精準，降低格式器邊緣放置的精確度。

除了延遲線本身的不完美之外，時脈產生器到各個邊緣產生器之間的歪斜(skew)及多相位時脈的不精確也會影響邊緣放置的精確度。這兩種誤差會造成測試符號的起始點偏移(offset)和延遲線的動態範圍間不完美的連接。若將四種相位選擇和八個位元的延遲線視為一組十個位元的數位對時間轉換器(digital-to-time converter)，則等效於一條涵蓋完整測試符號長度、十個位元之延遲線。其特性即為格式器邊緣放置之特性，故以其作為特性量測之對象。

特性量測意即透過量測將特性曲線建立出來，實際上的作法即為量測並記錄延遲碼所對應之延遲值。我們透過電腦端執行以 LabView 架設之自動化量測工具，同時控制 FPGA 端及量測工具示波器，將所

有延遲碼對應之延遲值逐一量測，並建立延遲線特性表來記錄特性量測之結果。



3.5.2 校正機制

高精確度的邊緣放置是本研究中重要的目標，而發展校正機制 (calibration mechanism) 之目的即為提升格式器邊緣放置之精確度。我們提出之校正機制係針對特別設計之混合式延遲線 (hybrid delay line) 發展而成，故需先介紹特別設計之混合式延遲線。

由於上述的各種誤差來源，延遲線特性曲線中之延遲值並不會均勻的分布。我們設計讓較小步距之延遲線的動態範圍略大於較大之步距，製造帶有重疊 (overlap) 之特性曲線，希望藉由散佈數量過多的延遲值進入特性曲線，避免特性曲線中缺乏某些範圍之延遲值。

在建立帶有重疊之特性曲線後，校正機制即可從中選出想要的延遲值，並將選中之延遲值標記出來，建立另一條擁有高線性度的特性曲線。由於由誤差造成的延遲值分布不均同時會有讓延遲值間隔變大及變小之情形，這樣的特性曲線在經過校正機制之後將有可能得到更高的解析度。故我們提出之校正機制不僅能提升邊緣放置精確度，更利用了 FPGA 延遲線不精準的特性，創造出超越 FPGA 規格限制的邊緣放置解析度。

校正機制的校正目標為消除時脈產生器到邊緣產生器的歪斜 (skew)、多相位時脈的不精確以及混合式延遲線的非線性度造成之影響，也就是測試符號的起始點偏移(offset)和不均勻分布之特性曲線。其具體的方法為先選擇起始點以建立理想之特性曲線，再選出最接近理想曲線之延遲值，並建立延遲碼對應表以標記選中之延遲值。

User Code	Delay Code
0	401 _(hex)
1	402 _(hex)
2	403 _(hex)
3	408 _(hex)

圖 3-10 延遲碼對應表示意圖

第四章 以 FPGA 實現之延遲線



4.1 混合式延遲線

為了能精準且細微的控制邊緣放置的時間點，達成高解析度的邊緣放置，我們在格式器中使用了延遲線作為時序電路。根據格式器中的設計，對於延遲線的設計需求包括略大於四分之一測試符號長度的動態範圍及與目標解析度同等級的最小步距，同時希望能盡量降低延遲線之最小延遲，以降低符號產生器之組數。

在延遲線的一些重要參數中，彼此會有互相牽制的狀況，不同架構的延遲線通常擁有不同的優點，但也有各自之問題。單一架構的延遲線難以同時達成我們對延遲線的所有需求，故我們將兩種不同架構之延遲線組合在一起，形成混合式延遲線(hybrid delay line)，嘗試以複合式的設計來綜合兩者之優點。

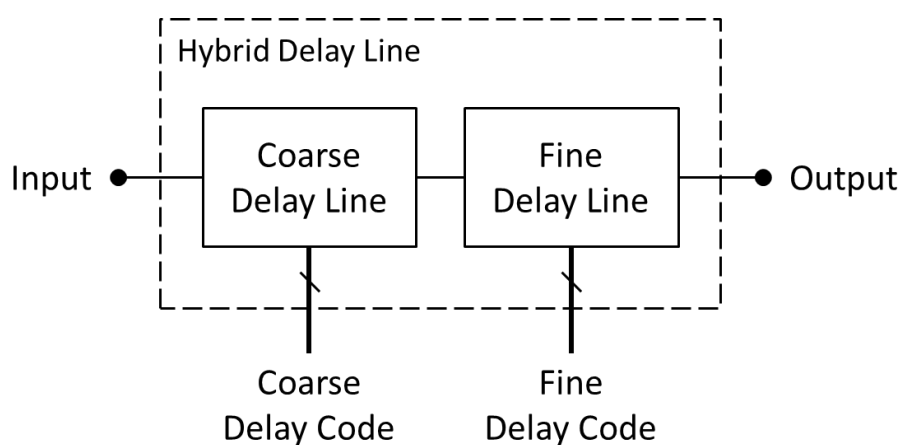


圖 4-1 混合式延遲線

混合式延遲線組合步距較大之粗調節延遲線(coarse delay line)及步距較小之細調節延遲線(fine delay line)，希望能同時達到大動態範圍及高解析度之需求。由粗調節延遲線擴展動態範圍，細調節延遲線則提供高解析度。這樣的組合方式讓最小延遲會是兩者之總和，線性度則是受到較差的一方主宰。

這種分段式概念的設計一般來說會讓細調節延遲線之動態範圍剛好與粗調節延遲線的解析度相等，但為了和校正機制相配合，我們設計讓細調節延遲線之動態範圍略大於粗調節延遲線之解析度，並讓粗調節延遲線之動態範圍同時也略大於多相位時脈之解析度，藉以製造帶有重疊之特性曲線。雖然最小延遲會是兩種延遲線之總和，但若和全以細調節延遲線實現相同的動態範圍及解析度的狀況相比，仍大大節省了延遲線之最小延遲。

Coarse Delay Code					Fine Delay Code		
0	0	0	0	0	0	0	0

圖 4-2 混合式延遲線之延遲碼

以下兩節將分別針對粗調節延遲線及細調節延遲線的架構和實作技術詳細說明。



4.2 以進位鏈實現粗調節延遲線

粗調節延遲線(coarse delay line)的設計目標為提供足夠的動態範圍及保持盡可能低的最小延遲。為增加和細調節延遲線組合時的相容性，單調性和不錯的線性度同時也是我們設計時考量之目標。

在 FPGA 內部的資源中我們選用進位鏈(carry chain)來實現粗調節延遲線。進位鏈是由上下相鄰的進位單元(carry cell)串接而成，以專屬之訊號路徑傳遞進位訊號而不需透過外部繞線路徑，能加速實現加法器時進位訊號之傳遞。其訊號傳遞方向為單向性的由下而上，並擁有訊號路徑低延遲的特性。

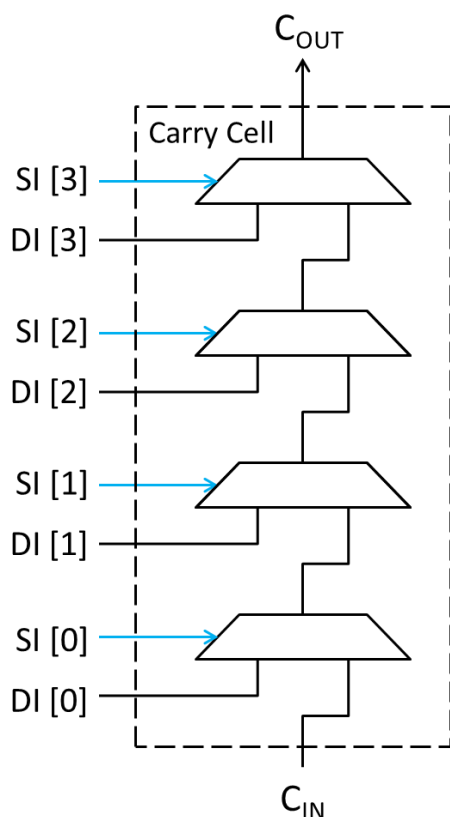


圖 4-3 進位單元架構圖




圖 4-3 為進位單元之架構圖。一個進位單元基本上是由四個多工器組合而成，每一級多工器的兩個輸入皆分別為內部訊號路徑及外部訊號路徑，並由多工器之選擇訊號決定選用之輸入，用來實現前瞻式加法器(carry-lookahead adder, CLA)之進位。

進位鏈的低延遲特性和簡單直接的串接方式是我們選用實現粗調節延遲線之原因，能在增加延遲線串接長度擴大動態範圍的同時降低疊加的最小延遲。我們以進位單元作為延遲單元(delay cell)串接成粗調節延遲線藉以保持單調性的特性，並企圖利用內部訊號路徑而不透過外部繞線傳遞訊號來降低繞線不確定性，提高延遲線之線性度。

在利用進位鏈實現粗調節延遲線時存在著一些需要解決的問題，本研究提出了一些實作技術用以克服這些問題。延遲線在邏輯上是多餘的(redundant)電路，在邏輯合成時合成器會最佳化(optimize)電路並將其消除。我們在硬體描述語言設計時使用了呼叫低階原始物件(low-level primitive)的技術，使延遲單元不會在邏輯合成時被消除。

進位鏈在傳遞訊號時是單方向的由下而上，故延遲線的輸入訊號必須由延遲線上方向下輸入，讓訊號在通過更多延遲單元時需要經過更長的多餘訊號路徑，以製造訊號之傳遞延遲。使用控制擺放(placement)之規範指令(constraint)控制多相位時脈之可調相位輸出

(即延遲線之輸入)擺放的位置即可滿足此需求。



延遲線在進位鏈中之訊號路徑相當規律，但延遲線輸入訊號的路徑規律性仍舊影響著延遲線之線性度。我們使用控制繞線(routing)之規範指令規定延遲線輸入訊號的繞線路徑，令其規律以提升粗調節延遲線之線性度。

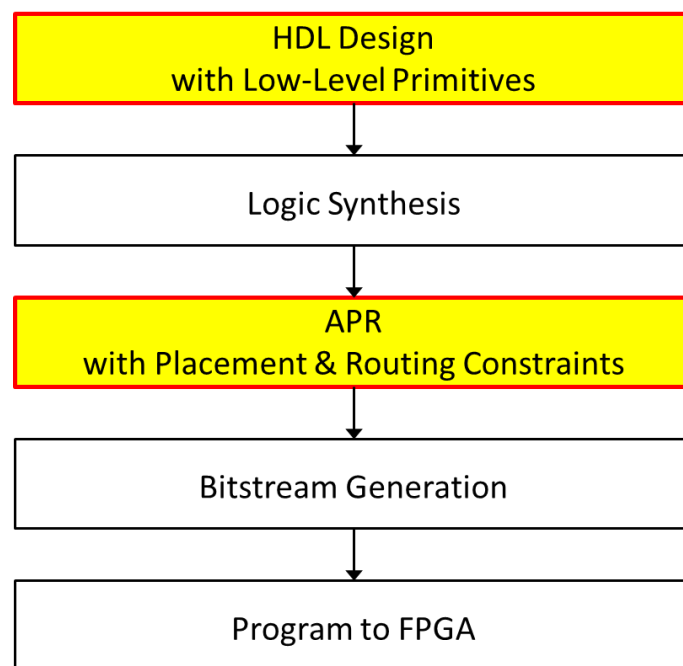


圖 4-4 為粗調節延遲線修正後之設計流程

4.3 以查找表實現細調節延遲線

細調節延遲線(fine delay line)的設計目標為提供高解析度及略大於粗調節延遲線解析度之動態範圍，線性度的部分則交由校正機制校正後來達成。

為達到次奈秒(sub-nanosecond)等級且超越 FPGA 內部資源規格之高解析度，細調節延遲線是採用游標尺延遲線(vernier delay line)的概念設計而成。游標尺延遲線之精神為運用兩種刻度之間的差距產生小於規格刻度之解析度，在 FPGA 內部的資源中我們選用六輸入之查找表(look-up table, LUT)來實現此細調節延遲線。

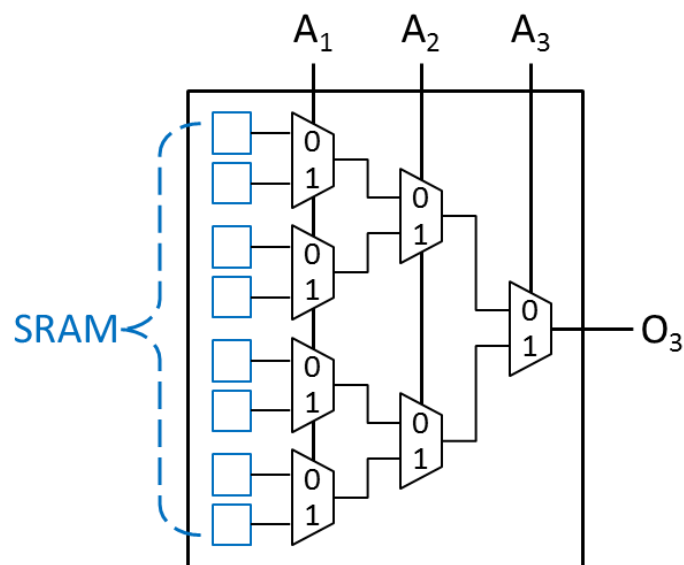


圖 4-5 三輸入查找表架構圖

圖 4-5 為一個三輸入查找表之架構圖。查找表在 FPGA 中負責實現任意的組合邏輯(combinational logic)，將欲實現之真值表儲存在其內部之靜態隨機存取記憶體(static random-access memory, SRAM)中。以輸入的多工器選擇訊號當作位址，透過一層層樹狀的多工器結構讀出預存之邏輯值。

本研究參考[6]中提出的以查找表實現游標尺延遲線之技術，在

查找表之靜態隨機存取記憶體中儲存交錯的邏輯零和邏輯一。此時第一層多工器的選擇訊號會決定查找表輸出之邏輯值，其餘選擇訊號則決定訊號傳遞之路徑。將不同路徑之傳遞延遲視為不同刻度，其差值即為游標尺延遲線創造出之解析度。

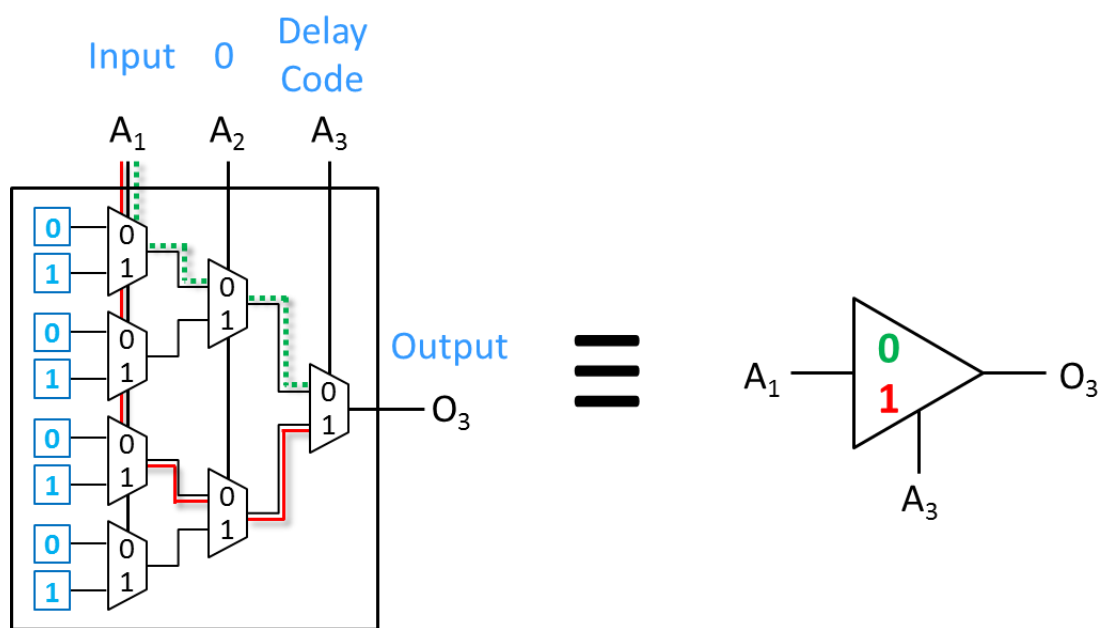


圖 4-6 以查找表實現之游標尺延遲線

如圖 4-6 所示，我們將查找表組態成帶有可程式化延遲值之緩衝器(buffer)，將第一層多工器的選擇訊號輸入視為延遲線輸入、查找表輸出視為延遲線輸出、其餘選擇訊號輸入視為延遲碼，即完成了細調節延遲線的延遲單元(delay cell)之實作。

在利用查找表實現細調節延遲線時同樣也存在需要克服的問題。在呼叫查找表之低階原始物件時需預先寫入真值表，若在此時寫入交

錯的邏輯零和邏輯一，邏輯合成時合成器會將查找表最佳化為一輸入一輸出之緩衝器，消去其餘的控制訊號，使我們無法調整延遲值。

故本研究提出了針對此問題修正之設計流程。在呼叫查找表時先將其組態為需用到多輸入之邏輯閘，保留住欲使用之輸入埠，讓合成器不會將其消除。完成邏輯合成和佈局之流程後再修改靜態隨機存取記憶體內之真值表，將查找表重新組態為可調延遲之緩衝器。此時繼續執行位元串生成並燒錄，即可在 FPGA 上實現設計之細調節延遲線。

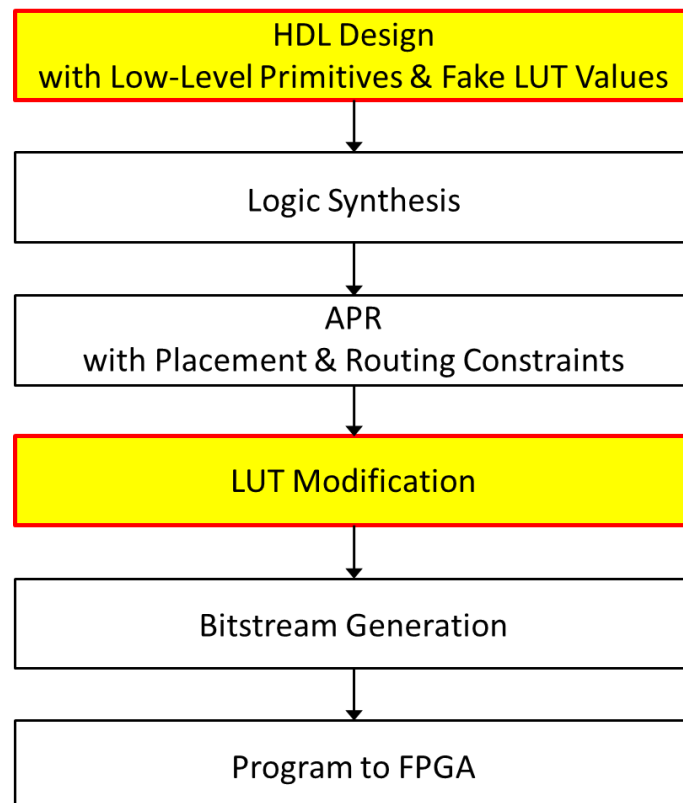


圖 4-7 為細調節延遲線修正後之設計流程

第五章 實驗與量測結果



5.1 系統規格與實驗配置

在本研究開始時所設定的設計目標規格為 100 Mbps 的測試符號頻率、50 ps 的邊緣放置解析度及高精確度，並希望能以最低需求之兩組符號產生器完成格式器的設計。因細調節延遲線之平均步距僅略高於 20 ps，故嘗試變更目標規格為 20 ps 的邊緣放置解析度。

系統中電腦端的電腦並無特殊的規格需求，僅需要支援 PXI 溝通介面即可。FPGA 端使用國內廠商能高電子(OpenATE)的產品 PXI FPGA 母板(PXI FPGA Carrier)，其上搭載之 FPGA 晶片為 Xilinx Spartan-6 XC6SLX75。此母板可透過通用型輸入輸出(GPIO)連接具有不同轉接頭之子板，插入 PXI 機箱(chassis)內即可運作。

PXI 機箱的部分使用的是國家儀器(National Instruments, NI)的 PXI-1036，負責電腦端與 FPGA 之間的溝通，以及 FPGA 的供電。子板則是自行製作的印刷電路板(printed circuit board, PCB)，將訊號由 FPGA 送至 SMA 轉接頭，以便使用同軸電纜(coaxial cable)連接至示波器。示波器使用安捷倫(Agilent)的 DSO-X 3052A，為所有實驗的量測工具，透過 USB 2.0 將量測結果傳送回電腦端。

5.2 延遲線特性量測與校正結果



延遲線的特性量測是透過電腦端執行以 LabView 架設的自動化量測工具來完成，主要功能是透過 PXI 控制 FPGA 延遲線之延遲碼，同時控制示波器進行特性量測，再接收示波器之量測結果並建檔。校正機制則是利用 MATLAB 撰寫而成，同樣由電腦端執行，分別建立延遲線特性表及校正後之延遲碼對應表，並計算積分/差分非線性誤差(INL/DNL)。本節將分別列出並分析延遲線特性量測結果和校正之結果。

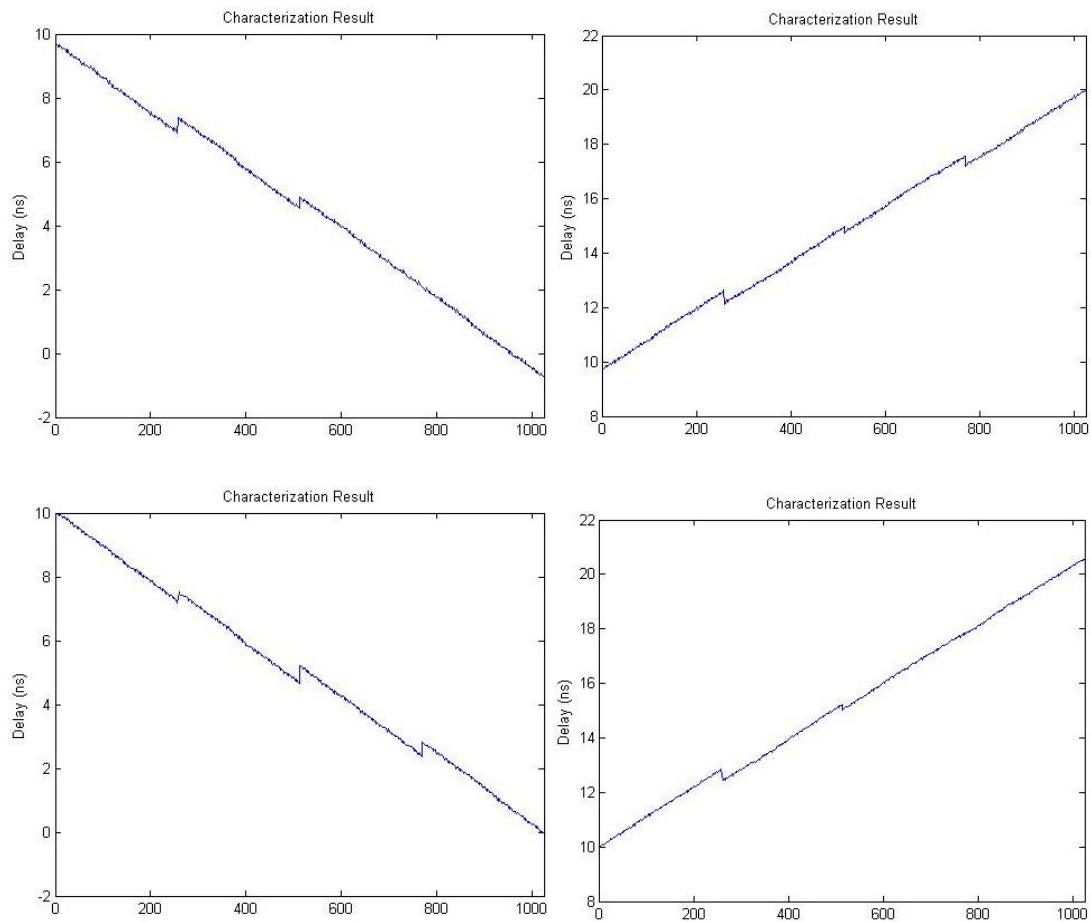
5.2.1 延遲線特性量測結果

在我們提出的格式器架構中一共有六條混合式延遲線，分別存在於六個邊緣產生器中。混合式延遲線由五個位元的粗調節延遲線及三個位元的細調節延遲線組成，加上兩個位元的多相位時脈，可視為一組十個位元的數位對時間轉換器(digital-to-time converter)，擁有 1024 組延遲碼和延遲值。特性量測即是指將這 1024 組延遲碼和延遲值量測出來並建表記錄，以提供校正機制作選擇。

表 5-1 延遲線特性量測動態範圍表

	延遲線 A0	延遲線 A1	延遲線 A2	延遲線 B0	延遲線 B1	延遲線 B2
延遲值數量	1024 (10 bits)					
動態範圍 (ns)	10.478	10.153	10.443	10.289	10.565	10.382

表 5-1 為六條延遲線特性量測後得到的動態範圍，經由四種相位之時脈延展後皆有超過測試符號長度 10 ns。為提供校正機制足夠的搜尋空間(search space)，延遲值的分布必須以些微重疊(overlap)的方式填滿整個測試符號長度。圖 5-1 即為六條延遲線之特性曲線。



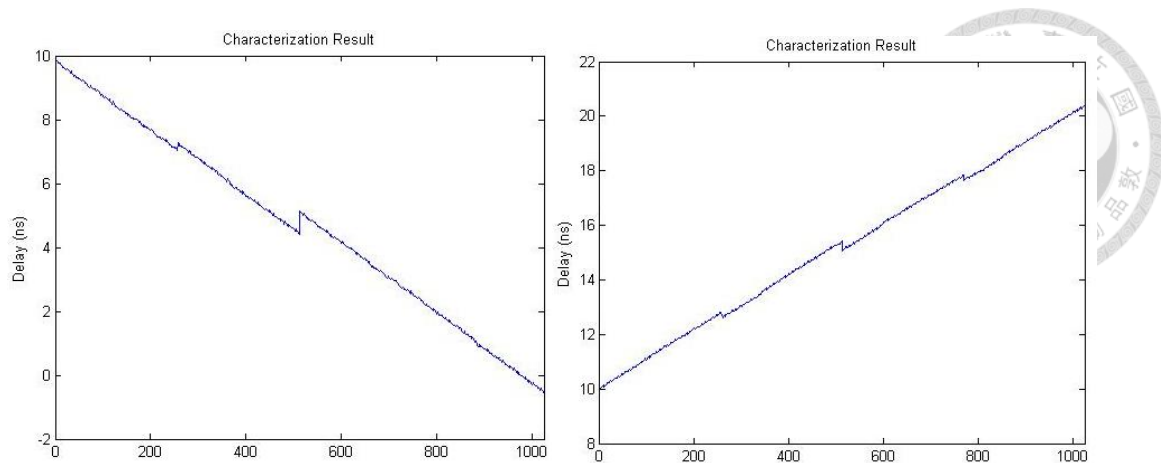


圖 5-1 延遲線特性曲線

如圖 5-1 中可以看到，特性曲線係由數段小曲線重疊而成。每一小段曲線有八個延遲值，是三位元的細調節延遲線之特性曲線，而每一小段曲線的起始點即為粗調節延遲線之延遲值。

在大約每四分之一個測試符號長度的地方會有較大較明顯的重疊，是切換多相位時脈所造成，透過這些大大小小的重疊，提供了校正機制對於接近理想特性曲線的延遲值更多的選擇。

5.2.2 校正結果

為了提高邊緣放置的精確度，我們提出特別設計的混合式延遲線與搭配之校正機制。校正機制有兩個主要的校正目標，分別為抵消起始點偏移(offset cancellation)和提升線性度(linearity improvement)。而校正方式則是由 1024 個延遲值中，萃取出最理想的一組延遲值，建表記錄以供使用，並計算積分/差分非線性誤差(INL/DNL)。

我們針對 20 ps 的目標邊緣放置解析度進行了校正，並將結果整

理如下表：



表 5-2 校正後延遲線參數表

	延遲線 A0	延遲線 A1	延遲線 A2	延遲線 B0	延遲線 B1	延遲線 B2
延遲值數量	500	489	500	500	500	500
動態範圍 (ns)	10.000	9.780	10.000	10.000	10.000	10.000
解析度(ps)	20 (t_{LSB})					
最大積分非 線性誤差 (LSB)	1.20	1.10	1.90	1.40	1.00	1.85
精確度(ps)	24	22	38	28	20	37

表 5-2 為校正後延遲線的重要參數，由於目標邊緣放置解析度為 20 ps 且測試符號長度為 10 ns，故校正機制會從 1024 個延遲值中萃取出最接近理想曲線的 500 個延遲值，構成動態範圍 10 ns 之特性曲線。我們定義邊緣放置的最大誤差為邊緣放置精確度，由解析度的最低有效位(LSB)乘上最大的積分非線性誤差(INL)而得出。

比較特別的地方是在六條延遲線中，位於邊緣產生器 A1 中之延遲線 A1 有動態範圍短缺的問題。這個問題的原因是由於時脈產生器到邊緣產生器的歪斜(skew)和多相位時脈的不精確疊加後超過了混

合式延遲線動態範圍可以補償的範圍，故造成延遲線 A1 會在測試符號的最尾端大約 220 ps 的位置無法放置邊緣，約占整個測試符號長度的 2.2%。

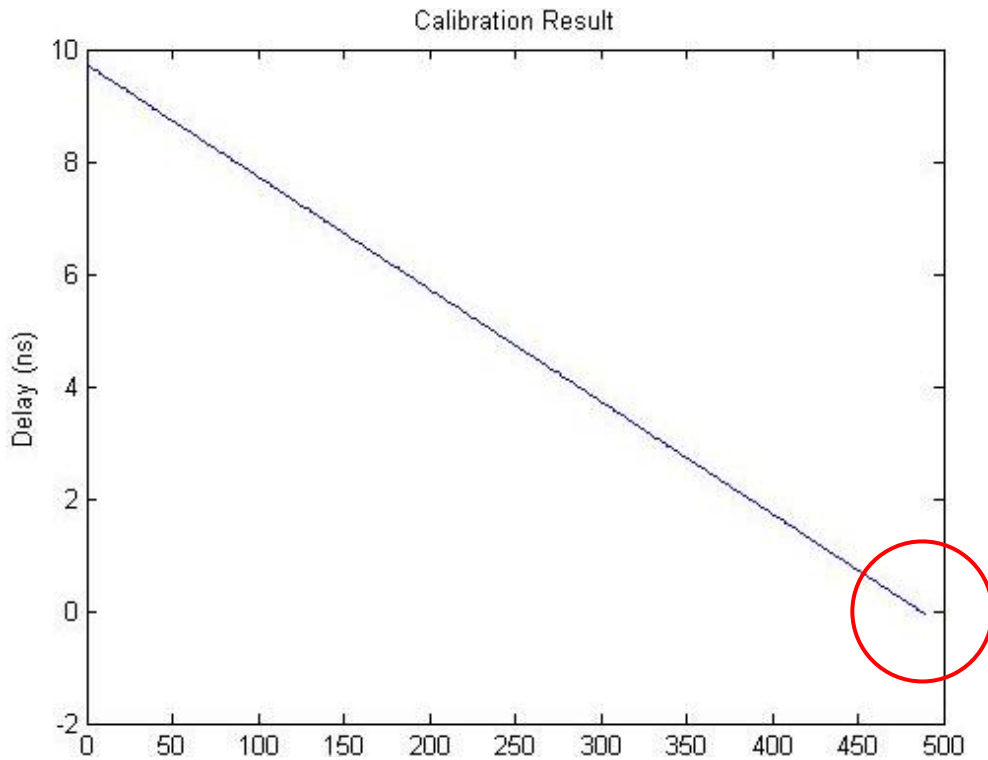


圖 5-2 延遲線 A1 經校正後之特性曲線

但因為提出之格式器在構造上的優勢，此問題並不會造成格式器的功能損失。由於符號產生器中有三個邊緣產生器，每一個邊緣產生器都支援完整的測試符號長度，在每個測試符號中，三個邊緣產生器可以被任意的選擇來產生需要的邊緣。因此，若需要在邊緣產生器 A1 無法放置邊緣的區間放置邊緣時，僅需選用其他邊緣產生器即可。

事實上，在大多數的訊號格式裡，一個測試週期並不需要多達三個邊緣時序，在需要三個邊緣時序的訊號格式裡，其中一個又固定存在於測試週期之起始點。也就是說，一個測試符號中需要產生的動態邊緣時序最多只有兩個。

因此，運用上述提到的符號產生器之架構優勢，在每一個測試符號的產生中，我們都能針對需要放置邊緣的位置選擇精確度較高的兩個邊緣產生器。在最糟的情況下也至少能避開精確度最低的一個邊緣產生器。

表 5-3 校正後之延遲線精確度表

	延遲線 A0	延遲線 A1	延遲線 A2	延遲線 B0	延遲線 B1	延遲線 B2
最大積分非 線性誤差 (LSB)	1.20	1.10	1.90	1.40	1.00	1.85
精確度(ps)	24	22	38	28	20	37

表 5-3 為校正後六條延遲線的最大積分非線性誤差(INL)及精確度。在將符號產生器 A 及符號產生器 B 中各自最大的積分非線性誤差去掉後，格式器邊緣放置精確度的期望值可進步到 28 ps (1.40 LSB)。在設計之架構和校正機制相輔相成的情況下，我們提出之 FPGA 格式器能夠使用精確度 38 ps 之延遲線實現精確度 28 ps 的邊緣放置。

5.3 邊緣放置解析度分析

一般來說，低於 0.5 個最低有效位(least significant bit, LSB)的最大積分非線性誤差(INL)被視為線性度優異的一個標準。所謂的邊緣放置解析度分析，意即找出在這個標準之下，系統可以達到的最高解析度為何。

在我們提出的校正機制中，希望的解析度是使用者可以輸入的參數之一，事實上理想曲線即是根據希望的解析度來建立的。故只要輸入不同的解析度需求，就可以得到該解析度下的校正結果，包含延遲碼對應表及積分/差分非線性誤差(INL/DNL)。

在研究開始時，格式器設計目標規格為 50 ps 的邊緣放置解析度與高精確度。因此，我們針對 50 ps 的邊緣放置解析度進行校正，整理結果如下表：

表 5-4 校正後之延遲線參數表(50 ps 解析度)

	延遲線 A0	延遲線 A1	延遲線 A2	延遲線 B0	延遲線 B1	延遲線 B2
延遲值數量	200	196	200	200	200	200
動態範圍 (ns)	10.000	9.800	10.000	10.000	10.000	10.000
解析度(ps)	50 (t_{LSB})					
最大積分非 線性誤差 (LSB)	0.38	0.36	0.36	0.28	0.28	0.24
精確度(ps)	19	18	18	14	14	12

由表 5-5 中可以發現，所有延遲線最大的積分非線性誤差(INL)都小於 0.5 個最低有效位(LSB)，表示若以 50 ps 的邊緣放置解析度作為系統規格，可以達到相當優異的邊緣放置精確度。事實上，根據校正結果，我們能以精確度 19 ps 的延遲線實現精確度 18 ps 的邊緣放置。

第六章 結論



6.1 結論


本論文設計並實現一個擁有高解析度及高精確度的邊緣放置功能之 FPGA 格式器原型。利用特別設計之混合式延遲線做為時序電路，精確的控制邊緣時序，並將 FPGA 時序電路無法消除的基本問題轉換為可利用的特性，配合所發展之校正機制，達成超越 FPGA 規格限制的高解析度及高精確度。

實驗結果顯示，本論文設計的格式器架構能夠在解析度為 20 ps 的情況下使用精確度 38 ps 的時序電路完成精確度 28 ps 的邊緣放置，並在解析度為 50 ps 的情況下達到高達 18 ps 的邊緣放置精確度。

透過修正設計流程後實現之 FPGA 延遲線，本論文實作出一個測試符號頻率 100 Mbps、邊緣放置解析度 20 ps、精確度 28 ps 之格式器原型，並支援在一個測試週期中需要三個邊緣時序以內之常見訊號格式。

6.2 未來研究方向

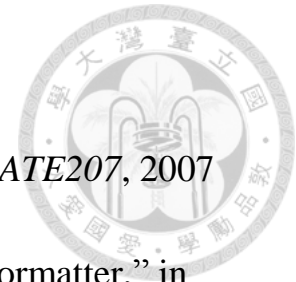
本論文提出的 FPGA 格式器原型擁有高解析度及高精確度的邊緣放置功能，但僅支援 100Mbps 的測試符號頻率。可調整的測試符



號頻率是目前的發展趨勢，針對高階的待測物進行測試時也需要更好的測試環境，故提高測試符號頻率並開發其可調性是未來研究之方向。透過改變多相位時脈產生器中鎖相迴路輸出之時脈頻率，再搭配延遲線動態範圍的配合，即有機會能提高及實現可調整之測試符號頻率。

在本論文中，自動量測工具和校正機制建立之延遲線特性表和延遲碼對應表目前都是存放在電腦端，尚未和溝通介面及整個系統作整合。將其整合入系統內是必要之發展方向，將能使系統完整性更臻成熟。

參考文獻



- [1] Analog Devices Inc. *Quad Pin Timing Formatter ADATE207*, 2007
- [2] A. R. Syed, “RIC/DICMOS-Multi-Channel CMOS Formatter,” in *International Test Conference*, 2003, pp. 175-184.
- [3] J. Park, et al. “Integration of Dual Channel Timing Formatter System for High Speed Memory Test Equipment,” in *International SoC Design Conference*, 2012, pp. 185-187.
- [4] L. Mostardini, L. Bacciarelli, L. Fanucci, L. Bertini, M. Tonarelli and M. D. Marinis, “FPGA-based Low-cost Automatic Test Equipment for Digital Integrated Circuits,” in *International Workshop on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications*, 2009, pp. 32-37.
- [5] Y.-Y. Chen, “An FPGA-based Sub-nanosecond Low-cost Timing Generator and Formatter,” M.S. thesis, National Taiwan University, Taipei, Taiwan, 2013.
- [6] Y.-Y. Chen, J.-L. Huang and T. Kuo, “Implementation of Programmable Delay Lines on Off-the-Shelf FPGAs,” in *AUTOTESTCON*, 2013, pp. 1-4.