

行政院國家科學委員會專題研究計畫 成果報告

具有即時效能/功率監控功能的高效率可程式化三維電腦繪圖晶片系統：軟硬體開發及整合--子計畫八：三維電腦繪圖晶片系統之可測試性設計與系統階層測試(3/3)
研究成果報告(完整版)

計畫類別：整合型
計畫編號：NSC 98-2220-E-390-001-
執行期間：98年08月01日至99年12月31日
執行單位：國立高雄大學電機工程學系

計畫主持人：洪進華

計畫參與人員：碩士班研究生-兼任助理人員：蘇湘涵
碩士班研究生-兼任助理人員：李政緯
碩士班研究生-兼任助理人員：黃子銘
碩士班研究生-兼任助理人員：蔡佳男

報告附件：出席國際會議研究心得報告及發表論文

處理方式：本計畫涉及專利或其他智慧財產權，2年後可公開查詢

中華民國 100 年 03 月 31 日

行政院國家科學委員會補助專題研究計畫

成果報告
 期中進度報告

具有即時效能/功率監控功能的高效率可程式化三維電腦繪圖晶片系統：軟硬體開發及整合-子計畫八：三維電腦繪圖晶片系統之可測試性設計與系統階層

測試

計畫類別： 個別型計畫 整合型計畫

計畫編號：NSC 98 -2200 -E -390 -001

執行期間： 98 年 8 月 1 日 至 99 年 12 月 31 日

計畫主持人：洪進華 教授

共同主持人：

計畫參與人員：蘇湘涵 蔡佳男 黃子銘 李政緯

成果報告類型(依經費核定清單規定繳交)： 精簡報告 完整報告

本成果報告包括以下應繳交之附件：

赴國外出差或研習心得報告一份

赴大陸地區出差或研習心得報告一份

出席國際學術會議心得報告及發表之論文各一份

國際合作研究計畫國外研究報告書一份

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

涉及專利或其他智慧財產權， 一年 二年後可公開查詢

執行單位：國立高雄大學電機工程學系

中 華 民 國 99 年 12 月 31 日

中文摘要

隨著生活品質提高，各種電子資訊技術發達，各種電子產品出現讓我們的生活更加方便，像是可以隨時隨地接收到最新資訊的PDA，捷運站方便簡單的購票系統，7-11門市提供各種福利的ibon，不管去到哪邊，都會有相關服務系統介面為我們解決需求。為了可以適應各種的年齡層，不管大人小孩都可以簡單的操作這些電子產品，其系統介面都是使用圖形化使用者介面 (Graphical User Interface)的設計，利用簡單的圖示或說明來讓使用者明白其功用，並能夠依照其指示來使用介面上的功能。

而SoC (System-on-Chip)近來發展蓬勃，其相關技術如軟硬體協同設定、SoC (System-on-Chip)驗證技術、低功率消耗、IP的可重用性、與嵌入式軟體移植與開發也逐漸興起，使他原有的面積小、速度快、耗電低等等優點更加進步。

傳統的scan-based電路設計花費了很多的測試時間在測試向量的輸入上，所以測試成本提高了很多。在這計劃中，利用了IEEE-1500標準測試電路來針對系統進行測試，並結合BIST (Built-In Self-Testing)技術。在這樣的測試環境下，除了原本IEEE-1500標準測試電路機制所具備的功能外更進一步的使用BIST (Built-In Self-Testing)來對待測電路進行測試，並整合運用到3D繪圖系統晶片上。

關鍵字：圖形化使用者介面 (Graphical User Interface)、可測試性設計 (Design For Testability)、內建自我測試 (Built-In Self-Testing)、IEEE 1500 標準、3D 繪圖系統晶片

英文摘要

As the quality of the life improves, various electronic information technologies are developed and various electronic products make our life more convenient, such as PDA which can receive the latest message any time wherever we go, the ticket purchasing system, the ibon system of 7-11 which offers the different welfare. No matter where we go, there are many relevant service system interfaces to solve the requirement of us. In order to let electronic products could be easily handled by people of any age bracket, all the system interfaces are designed to be the Graphical User Interface (GUI). The user can understand the system functions by simple icon or instruction. Following the GUI instructions, user can use the system functions.

Recently, the developments of SoC techniques are popular, such as Hardware and Software Association set, SoC (System-on-Chip) check technology, low power consumption, the reuse of IP, and embedded software development and so on, so that the area can be smaller, speed of process can be faster and the power consumption can be lower, etc.

Traditional IEEE-1500-based circuit needs many test time to scan input the test pattern. It raises the testing cost a lot. In this plan, we combine the IEEE-1500 standard and Built-In Self-Testing (BIST) technique to test the system. Under such test environment, not only the functions of IEEE-1500 standard, but also the fast testing advantage of BIST (Built-In Self-Testing) can be achieved when test the circuit. We apply it to test many benchmark circuits and 3D Graphics SoC as well as.

Keywords: Graphical User Interface 、 DFT (Design For Testability) 、 BIST (Built-In Self-Testing) 、 IEEE 1500 Standard Test Circuit 、 3D Graphics SoC

目錄

中文摘要.....	II
英文摘要.....	III
目錄.....	IV
圖目錄.....	VI
表目錄.....	IX
前言.....	1
1. 研究目的.....	2
2. 研究方法.....	3
2.1. IEEE 1500 測試架構.....	3
2.1.1. IEEE-1500 測試界面.....	3
2.1.2. IEEE 1500 測試界面指令暫存器(WIR)	6
2.1.3. 旁路暫存器(Bypass Register).....	7
2.1.4. IEEE 1500 測試界面單元(Wrapper Cell).....	7
2.1.5. IEEE 1500 核心指令集	9
2.2. IEEE 1500 腳位簡化.....	13
2.3. 測試存取埠控制器(TAP controller)與邊界掃描測試流程之控制	15
2.4. 轉換 IEEE-1500 的 Test Pattern.....	16
2.4.1. 測試向量擴展(Test Pattern Expansion)	16
2.4.2. 演算法.....	17
2.4.3. 流程圖說明.....	18
2.5. BIST 架構介紹	18
2.5.1. 內建自我測試.....	19
2.5.2. 線性回饋移位暫存器.....	20
2.5.3. MISR (Multiple Input Signature Register)	21
2.6. BIST 與邊界掃描電路結合	23
2.6.1. 輸入 Boundary Scan Cell 合併到 LFSR.....	23
2.6.2. 輸出 Boundary Scan Cell 合併到 MISR.....	25
2.6.3. 修改 WRCK 信號設計.....	27
2.6.4. RUNBIST 指令	28
2.6.5. IEEE-1500 與 BIST 架構整合說明	29
3. 運作機制.....	31
3.1. 3DG 繪圖晶片架構與測試機制.....	31
3.2. JTAG 外部腳位規劃	32
3.3. VS 和 FS 測試向量擴展	33
4. 結果.....	34

4.1. IEEE-1500 without BIST 之測試機制驗證.....	34
4.1.1. 指令與資料時序模擬圖.....	34
4.1.2. 測試模式和正常模式控制.....	35
4.1.3. 傳送資料路徑的模式設定.....	36
4.1.4. VS 和 FS 以 IEEE-1500 without BIST 結果.....	37
4.1.4.1. Normal mode 結果.....	37
4.1.4.2. Serial testing mode 結果比對.....	39
4.2. IEEE-1500 with BIST 之測試機制驗證.....	44
4.2.1. VS 和 FS & IEEE-1500 with BIST 架構圖.....	44
4.2.2. Hardware Overhead.....	45
4.2.3. VS 和 FS 以 IEEE 1500 with BIST 結果比對.....	48
5. 結果與未來工作.....	56
參考文獻.....	57

圖目錄

圖 1-1. 3D 繪圖系統晶片	2
圖 2-1. IEEE-1500 測試介面	3
圖 2-2. Wrapper Interface Port (WIP)	4
圖 2-3. 指令暫存器(Wrapper Instruction Register)	6
圖 2-4. 指令暫存器時脈動作	7
圖 2-5. Wrapper Bypass Register	7
圖 2-6. 測試介面單元的架構	8
圖 2-7. 邊界掃描暫存器(Wrapper Boundary Cell)	9
圖 2-8. WS_Bypass 指令	9
圖 2-9. WS_EXTEST 指令	10
圖 2-10. WP_EXTEST 指令	10
圖 2-11. WS_SAFE 指令	11
圖 2-12. WS_PRELOAD 指令	11
圖 2-13. WP_PRELOAD 指令	12
圖 2-14. WS_INTEST_RING 指令	12
圖 2-15. WS_INTEST_SCAN 指令	13
圖 2-16. IEEE-1149.1 TAP to IEEE-1500 WSP interface logic	14
圖 2-17. IEEE Std.1149.1 TAP 控制器	15
圖 2-18. IEEE-1149.1 邊界掃描	16
圖 2-19. 測試向量擴展公式	16
圖 2-20. 測試向量擴展	17
圖 2-21. 測試向量擴展演算法	17
圖 2-22. 測試向量擴展流程圖	18
圖 2-23. 內建式自我測試架構圖	19
圖 2-24. External-XOR LFSR	20
圖 2-25. Internal-XOR LFSR	20
圖 2-26. 4-Stage External-XOR LFSR	20
圖 2-27. 4-Stage External-XOR LFSR 特徵多項式	21
圖 2-28. 4-Stage External-XOR LFSR 狀態	21
圖 2-29. Internal-XOR MISR	22
圖 2-30. 第 i 個時脈的連續掃描電路	22
圖 2-31. 第 i+1 個時脈 MISR 架構	22
圖 2-32. 輸入 Boundary Scan Cell 合併 LFSR	23
圖 2-33. 輸入 Boundary Scan Cell 編號 20	24
圖 2-34. 輸入 Boundary Scan Cell 編號 30	24
圖 2-35. 輸入 Boundary Scan Cell 編號 40	25

圖 2-36. 輸出 Boundary Scan Cell 合併 MISR	25
圖 2-37. 輸入 Boundary Scan Cell 編號 50	26
圖 2-38. 輸入 Boundary Scan Cell 編號 60	26
圖 2-39. 輸入 Boundary Scan Cell 編號 70	27
圖 2-40. 修改電路設計的 WRCK 信號	27
圖 2-41. BIST 邊界掃瞄電路.....	29
圖 3-1. 整體 3DG 架構.....	31
圖 3-2. 繪圖晶片導入 IEEE-1500 with BIST 測試架構	32
圖 3-3. IEEE-1149.1 和 IEEE-1500 外部腳位切換	33
圖 4-1.指令暫存器的運作模擬圖.....	34
圖 4-2.資料暫存器的運作模擬圖	34
圖 4-3.執行正常模式	35
圖 4-4.執行測試模式	35
圖 4-5. VS 作測試時之路徑	36
圖 4-6. FS 作測試時之路徑.....	37
圖 4-7. Normal Mode 測試架構圖.....	38
圖 4-8. VS 的 Boundary Scan Cell 包覆前 Pattern 模擬.....	38
圖 4-9. VS 的 Boundary Scan Cell 包覆後 Pattern 模擬.....	38
圖 4-10. FS 的 Boundary Scan Cell 包覆前 Pattern 模擬	39
圖 4-11. FS 的 Boundary Scan Cell 包覆後 Pattern 模擬	39
圖 4-12. Benchmark c6288 在 Serial testing mode 測試結果	40
圖 4-13. VS 在 Serial Testing mode 測試結果	41
圖 4-14. FS 在 Serial testing mode 測試結果.....	41
圖 4-15. Serial Testing Mode 測試架構圖	42
圖 4-16. VS 的 Serial Testing mode 結果比對	43
圖 4-17. FS 的 Serial Testing mode 結果比對.....	43
圖 4-18. VS+ IEEE-1500.....	44
圖 4-19. FS + IEEE-1500	45
圖 4-20. VS 的 TAP Controller 的面積.....	45
圖 4-21. VS 的 IEEE-1500 Wrapper 的面積	46
圖 4-22. IEEE-1500 佔 VS 的硬體負擔	46
圖 4-23. FS 的 TAP Controller 的面積	47
圖 4-24. FS 的 IEEE-1500 Wrapper 的面積	47
圖 4-25. IEEE-1500 with BIST 佔 FS 的硬體負擔	48
圖 4-26. Benchmark c6288 使用 LFSR 測試結果	48
圖 4-27. Benchmark c6288 的 Pattern 個數以及 TC 和 FC.....	49
圖 4-28. VS 在 Serial Testing mode 測試結果	50
圖 4-29. FS 在 RUNBIST 測試結果.....	50

圖 4-30. Characteristic Polynomial(特徵多項式).....	51
圖 4-31. Test Coverage 和 Fault Coverage 計算公式.....	51
圖 4-32. RUNBIST 測試架構圖	52
圖 4-33. 32 Bits 的 LFSR 架構	52
圖 4-34. 23 個時脈週期後 LFSR 內部的值.....	52
圖 4-35. 23 個時脈週期後 LFSR 結果.....	53
圖 4-36. 40 個時脈週期後 MISR 內部的值結果比對(VS).....	53
圖 4-37. 40 個時脈週期後 MISR 結果(VS).....	54
圖 4-38. 40 個時脈週期後 MISR 內部的值(FS)	54
圖 4-39. 40 個時脈週期後 MISR 結果比對(FS)	55

表目錄

表格 2-1	IEEE-1500 測試介面腳位功能及用途.....	5
表格 2-2	IEEE-1500 與 IEEE 1149.1 對應的關係.....	14
表格 2-3	TAP 控制器的狀態分配.....	28
表格 4-1	Benchmark 在 Serial Testing mode 的結果.....	40
表格 4-2	VS 在 Serial Testing mode 測試結果	41
表格 4-3	FS 在 Serial Testing mode 測試結果	42
表格 4-4	使用 Benchmark 在 BUNBIST 的結果.....	49
表格 4-5	VS 在 Serial Testing mode 測試結果	50
表格 4-6	FS 在 RUNBIST 測試結果.....	51

前言

由於 SoC 的測試為一相當複雜的問題，在這次計畫中我們主要是針對 IEEE-1500 機制來作整合應用，IEEE-1500 標準是將電路測試標準化，並提供即插即用的特性，而這個標準是針對電路隔離機制與存取機制做測試。因此，IEEE-1500 標準這種機制它的運作方式是從外部控制器接收測試資料與測試命令，並將所接收到的訊號傳送到相對應的電路中來對電路做測試。

因為 IEEE-1500 在掃描設計時需要長時間的將測試資料 Shift 到掃描暫存器內部，所以我們把 IEEE-1500 的邊界掃描暫存器(Boundary Scan Register)加以改良同時符合 IEEE-1500 的標準，將 IEEE-1500 和 BIST(Built-In Self-Testing)兩種機制加以結合，以下我們會對此方法做更詳盡的介紹，包括演算法和整合機制...等等。

這次研究是參與國科會計畫「具有即時效能/功率監控功能的高效率可程式化三維電腦繪圖晶片系統」所開發出來的自動化輔助設計平台，它具有 SoC 系統晶片的開發與應用，藉由此計畫的晶片參與測試應用階層進行完善的測試執行，以及單晶片的開發時程與開發成本部分，以提高執行的效能為目的，協助其發展及設計。而在此研究在計畫中則負責系統單晶片測試的開發與設計。

1. 研究目的

嵌入式系統雖非嶄新的技術，但因現今 IC 設計和網路技術的蓬勃發展，其應用的領域亦趨廣泛。舉凡如醫療儀器、數位影音系統、手機、電信和網路設備、遊戲等，皆為嵌入式系統的應用範疇。此計劃 3DG SoC 三維電腦圖學的發展方向相當熱門，並加入 DFT 架構以利 3DG SoC 之三維電腦圖學內部來做測試。

3D 電腦繪圖是為了模擬出真實世界的物體，讓三度空間的物體可以顯現在二度空間平面的螢幕上，透過許多複雜的演算法或模型來模擬真實物體在三度空間中的狀況，最後投射在二維平面的螢幕上。經過演算法運算後還有許多因素會影響到最後此影像的真實度，包含投射的方法、物件模型的細緻程度、光源模型的精確度等等；如圖1-1 3D繪圖系統晶片所示。

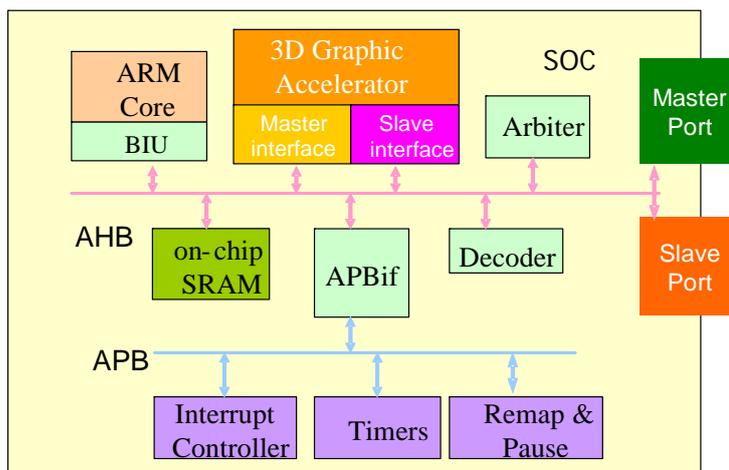


圖 1-1. 3D 繪圖系統晶片

2. 研究方法

SoC 不斷的進步，IC 的測試整合以及標準的建立相對地顯示出其所賦予的重要性，1995 年九月，IEEE Test Technology Technical Council(TTTC)創立一個 Technical Activity Committee(TAC)以促進這領域的發展。1996 年六月，IEEE Standard Activity Board 創立了 IEEE 標準工作小組(Standard Working Group)，其後就命名為 IEEE-1500。在此將對 IEEE 1500 作介紹。

2.1. IEEE 1500 測試架構

IEEE-1500 的核心是由設計核心者自己自訂的，只要它能符合 IEEE-1500 的介面環境就可以。在本節中我們將建立一個符合 IEEE-1500 標準的核心測試架構環境，並以隨插即用(plug-and-play)的方式整合於系統晶片中[1]。

2.1.1. IEEE-1500 測試界面

如圖 2-1 所示，此介面包含並列式機制的存取介面(TAM-in 及 TAM-out)和串列式機制的存取介面(WSI、WSO)。並列式存取機制為系統整合者自行定義，而串列式存取機制為強制規範[2]。

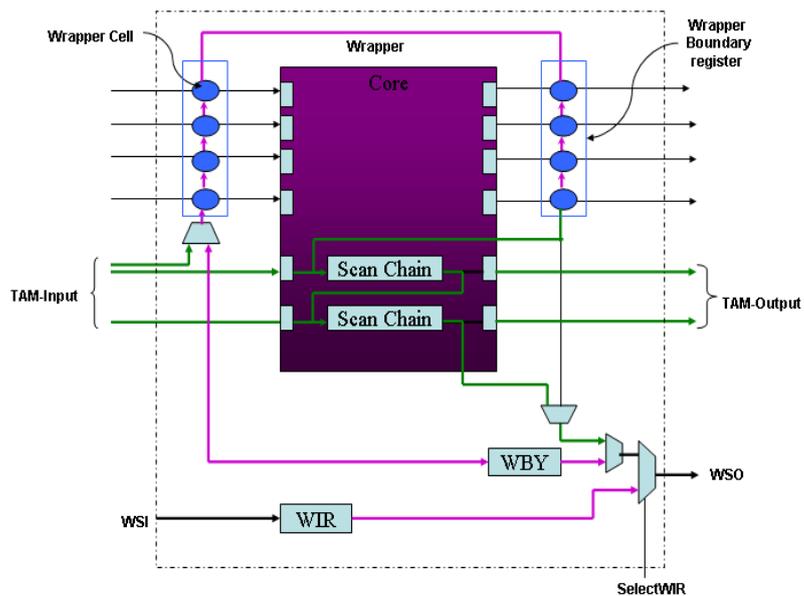


圖 2-1. IEEE-1500 測試介面

以下為 IEEE-1500 所包含測試介面的元件：

- 【1】 Wrapper Boundary Cells (WBC)：測試介面邊界掃描暫存器
- 【2】 Wrapper Instruction Register (WIR)：測試介面指令暫存器
- 【3】 Wrapper Cell：在 Core 的輸入端和輸出端都加上 Wrapper Boundary Cells(WBC)，對核心做測試和驗證。
- 【4】 Wrapper Bypass (WBY)：測試介面旁路暫存器
- 【5】 Wrapper Interface Port(WIP)：測試介面埠，如圖 2-2 所示
- 【6】 TAM-Input / TAM-Output(TAM-In / TAM-Out)：TAM 輸出/ TAM 輸入，為平行傳輸方式，為可選用訊號，用以加速測試資料的傳輸。

如圖 2-2 所示，IEEE-1500 有六條控制線以及兩條資料線，表格 2-1 為其測試介面腳位功能及用途簡介，並在以下各小章節，我將對 IEEE-1500 裡面的每個元件功能做詳細的介紹。

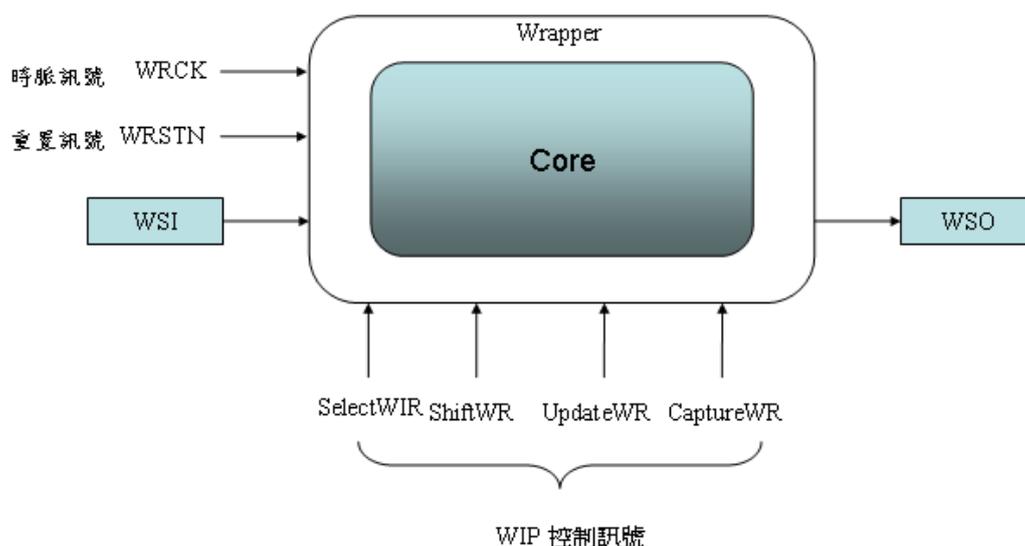


圖 2-2. Wrapper Interface Port (WIP)[3]

表格 2-1 IEEE-1500 測試介面腳位功能及用途[4]

訊號名稱	功能	用途
WRCK	時脈訊號	為 IEEE-1500 測試介面專用，用於 IEEE-1500 所有電路上，例如：WBR、WIR、WBY…等等的暫存器。
WRSTN	重置訊號	為 IEEE-1500 測試介面專用，用於 IEEE-1500 所有電路上，用途與 WRCK 相同。
WSI、 WSO	輸入訊號、 輸出訊號	為 IEEE-1500 的串列輸入端和串列輸出端，利用 WSI 把指令送到 WIR 中，並根據 WIR (Wrapper Instruction Register) 所接收到的指令，和 WIP 的控制訊號，去決定 IEEE-1500 中的哪一個資料暫存器，應該被接到 WSI 和 WSO 之間來傳送資料。
SelectWIR	選擇訊號	為 IEEE-1500 選擇訊號，用途為選擇 WIR 或是選擇其他暫存器(例如：WBR、WIR、WBY…等等的暫存器)。用來決定哪一個來接到 WSI 和 WSO 之間來傳送資料。
CaptureWR、 ShiftWR、 UpdateWR、	控制訊號	為 IEEE-1500 控制信號，用來控制 WIR 或是 WBR 的動作。配合 WRCK、SelectWIR 和 WIR 的指令，來決定全部的資料暫存器的行為。

2.1.2. IEEE 1500 測試界面指令暫存器(WIR)

WIR(Wrapper Instruction Register) 它是一個 4 Bit 的指令暫存器，其作用是用來存取 IEEE 1500 測試介面的指令。如圖 2-3 所示，由兩階段組成的 WIR 暫存器，它的兩階段長度為相等的，第一階段是移位暫存器(Shift Register)，第二階段是更新暫存器(Update Register)。WIR 收到指令後並經由解碼器解碼，並依據這些解碼信號來控制相對應的資料暫存器(例如：WBY、WBR...等)。

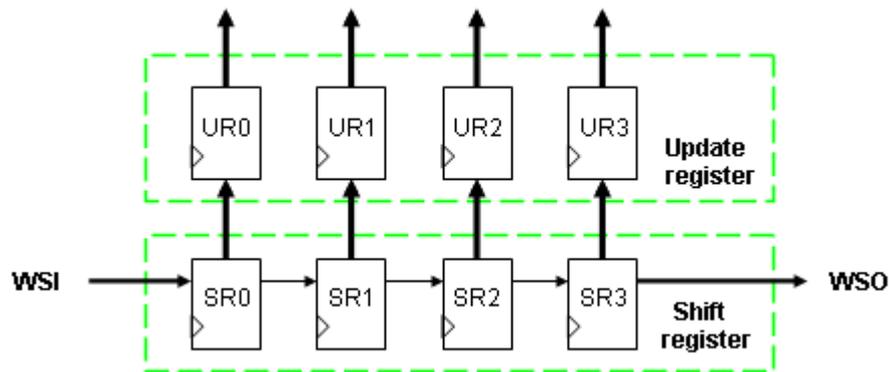


圖 2-3. 指令暫存器(Wrapper Instruction Register)

如圖 2-4 所示，將指令載進去 WIR，它的對應動作方式說明如下：

Step 1. 設定【SelectWIR = 1】：

先將把 WIR 連接到 WSI 和 WSO 之間，準備開始要將新的指令讀入 WIR 中。

Step 2. 設定【SelectWIR = 1 ， ShiftWIR = 1】：

新的指令將會在 WRCK 的正緣訊號以串列(Serial)的方式進入 WIR 中。

Step 3. 設定【ShiftWR = 0】：

當新的指令已經進入 WIR，將此指令留在 WIR 中。

Step 4. 等到【UpdateWR = 1】：

當 WRCK 負緣觸發時，WIR 就會解碼此指令，控制相對應的資料暫存器。

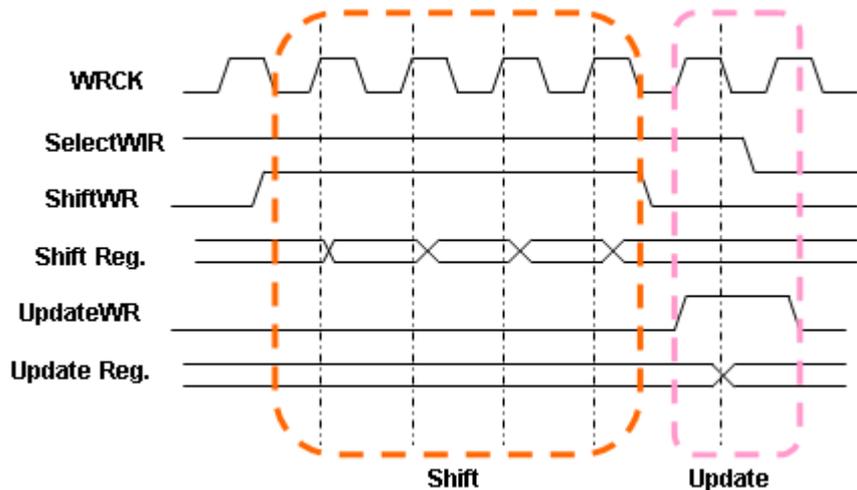


圖 2-4. 指令暫存器時脈動作

2.1.3. 旁路暫存器(Bypass Register)

旁路暫存器(Bypass Register)它是只有一個位元的暫存器，是WSI到WSO中最短的距離，當有兩個以上待測電路時，就可以將要送到下一個待測電路的資料使用旁路暫存器(Bypass Register)快速的傳送過去，來加速其測試速度，不用經過Shift來送資料給WBR(Wrapper Boundary Register)的時間，如圖2-5所示。

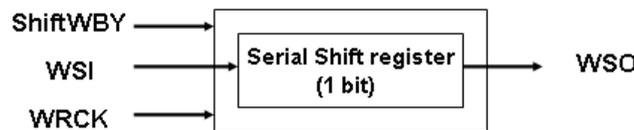


圖 2-5. Wrapper Bypass Register

2.1.4. IEEE 1500 測試界面單元(Wrapper Cell)

測試介面單元的架構如圖 2-6 所示，在核心電路的每個輸入端與輸出端都會加上測試介面單元(Wrapper Cell)，原先的 Core Input 和 Core Output 會被 Wrapper Function Input(WFI)和 Wrapper Function Output(WFO)取代。

- [1] 在測試模式(Test Mode)時：測試介面單元(Wrapper Cell)會發揮它的功能。
- [2] 在正常模式(Normal Mode)時：測試介面單元(Wrapper Cell)會完全被忽略。

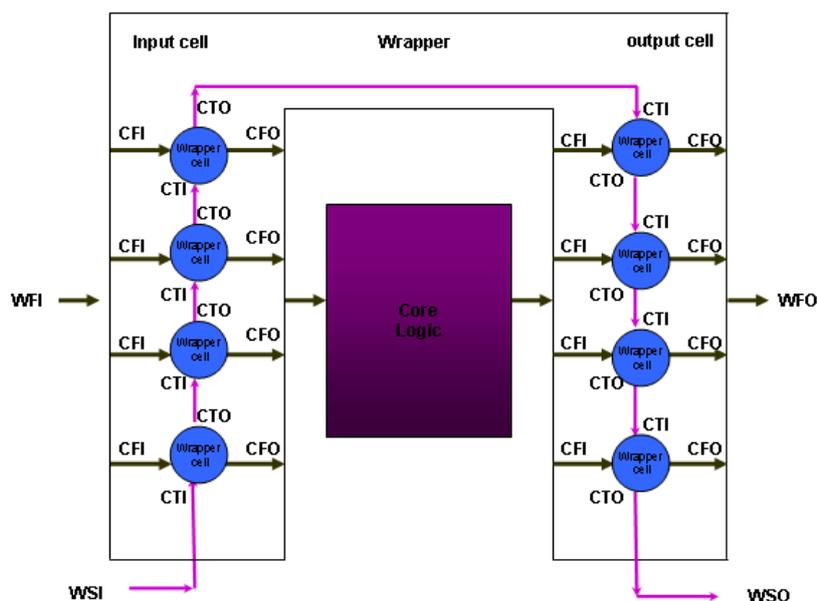


圖 2-6. 測試介面單元的架構

減少接腳個數：在 IEEE-1149.1 標準邊界掃描暫存器使用兩個掃描細胞(Scan Cell)，而 IEEE-1500 標準邊界掃描暫存器在移位(shift)和擷取(capture)運作期間，我們僅使用一個 D 型正反器(D flip-flop)，所以沒有更新(update)正反器，如圖 2-7。我們將對此邊界掃描暫存器的運作模式做說明：正常模式(normal mode)，更新模式(update mode)，擷取模式(capture mode)，移位模式(shift mode)。

【1】在正常模式(normal mode)期間：

$SC = X$ ， $AC = 0$ ，CFI 的值會傳遞到 CFO。

【2】在移位模式(shift mode)期間：

$SC = 1$ ，掃描細胞(Scan Cell)的 CTO 的值被傳遞到下一個掃描細胞(Scan Cell)的 CTI。

【3】在更新模式(update mode)期間：

$AC = 1$ ，掃描細胞(Scan Cell)的值被傳遞到 CFO。

【4】在擷取模式(capture mode)期間：

$SC = 1$ ， $AC = 0$ ，將 CFI 的值會傳遞到 CFO，而 CFO 的值會傳遞到掃描細胞(Scan Cell)，並在下一個 WRCK 時再將值抓取出來。

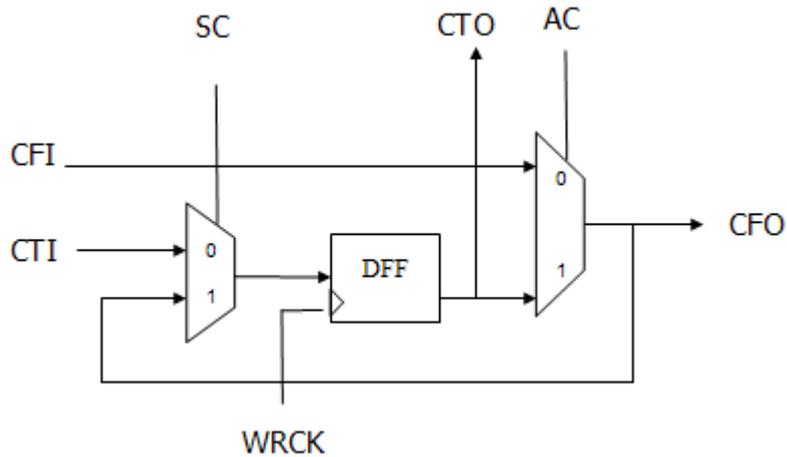


圖 2-7. 邊界掃描暫存器(Wrapper Boundary Cell)[5]

2.1.5. IEEE 1500 核心指令集

在這個章節，我們將介紹 IEEE 1500 測試架構所使用的指令集，以及每個設定的詳細內容。WIR 會根據所解讀到的指令下去做路徑設定，以下為每個指令的詳細說明[6]：

1. WS_Bypass：如圖 2-8 所示，用途為將測試資料通過，經由旁路暫存器(WBY)的資料可以很快地通過 WSI(Wrapper serial input)與 WSO(Wrapper serial output)。

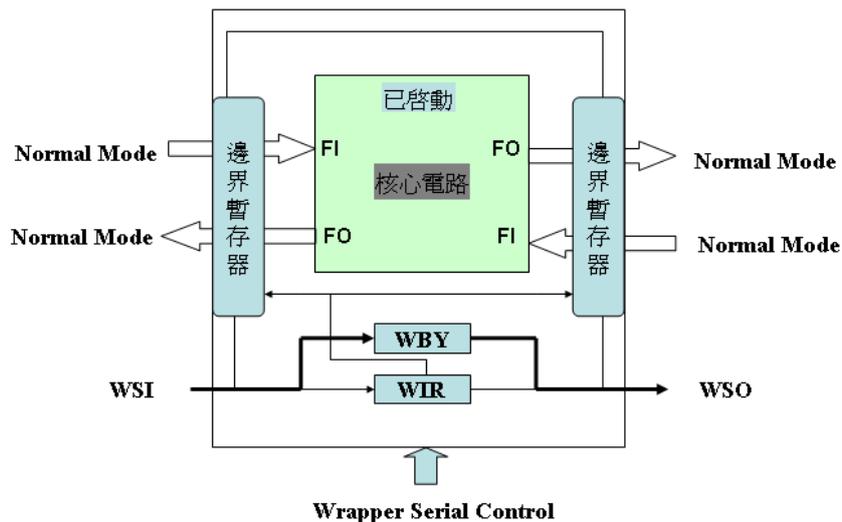


圖 2-8. WS_Bypass 指令

2. WS_EXTEST：如圖 2-9 所示，這個指令會 Wrapper Cell 連成一串，輸入為 WSI (Wrapper

serial input) 輸出為 WSO (Wrapper serial output)。用於將核心電路與電路之間做連接，會將測試資料傳送到 WSO (Wrapper serial output)，然後 Update，並在下一顆核心電路的輸入端擷取(Capture)輸出結果來作測試。

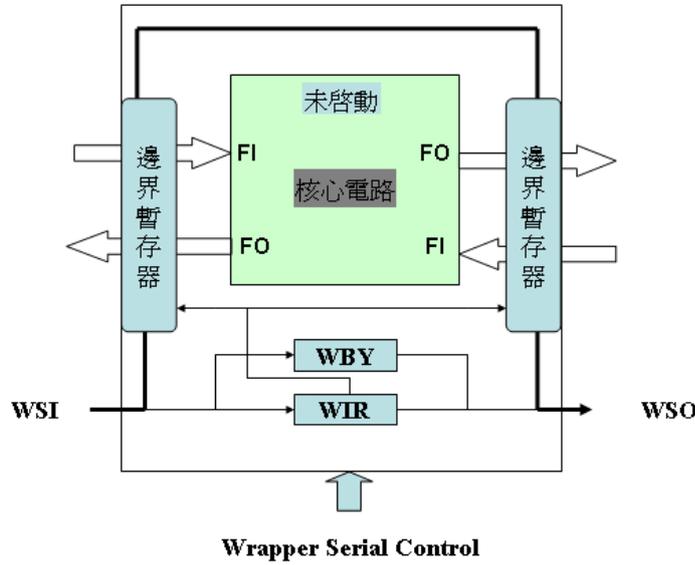


圖 2-9. WS_EXTEST 指令

3. WP_EXTEST：如圖 2-10 所示，邊界暫存器(Wrapper Boundary Register)可以被切割為多個部份，除了資料是以並列(Parallel)方式進行之外，此指令與 WS_EXTEST 相同，主要用於核心電路與電路之間的連接線，而輸入與輸出是利用 TAM。

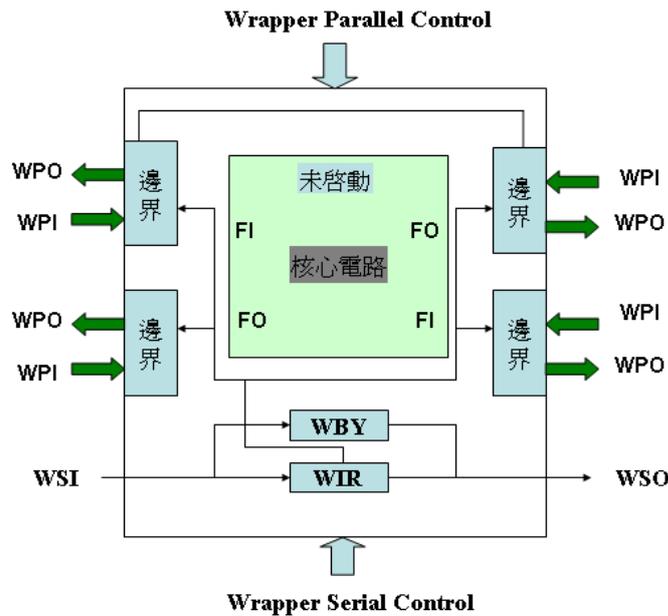


圖 2-10. WP_EXTEST 指令

4. WS_SAFE：如圖 2-11 所示，用於將測試資料會由輸入端 WSI (Wrapper serial input)輸

入，經由旁路暫存器(Bypass Register)，從 WSO (Wrapper serial output)輸出到下一個核心。此模式會將 Wrapper 切換到安全狀態，而核心輸出的值並不會影響核心以外的電路，所以邊界暫存器(Wrapper Boundary Register)的 Wrapper 功能輸出端將被設定為安全值，該核心也被設定在安全模式。

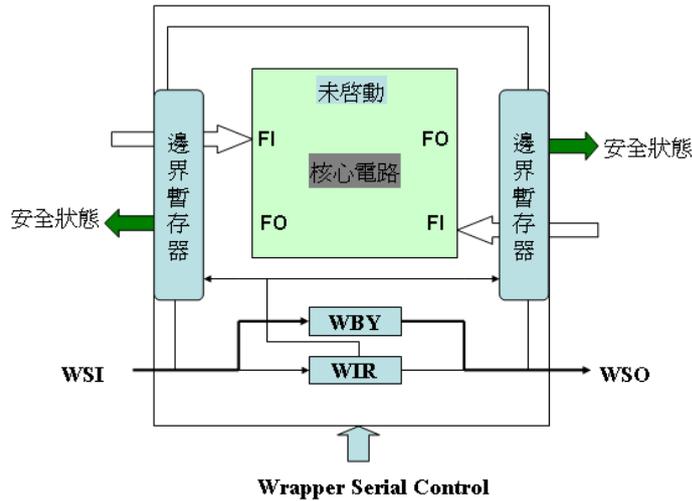


圖 2-11. WS_SAFE 指令

5. WS_PRELOAD: 如圖 2-12 所示，此指令不會干擾連接在邊界暫存器(Wrapper Boundary Register)的電路，它會將所有的 Wrapper Cell 連成一串，測試資料會以串列(Serial)方式載入邊界暫存器(Wrapper Boundary Register)。

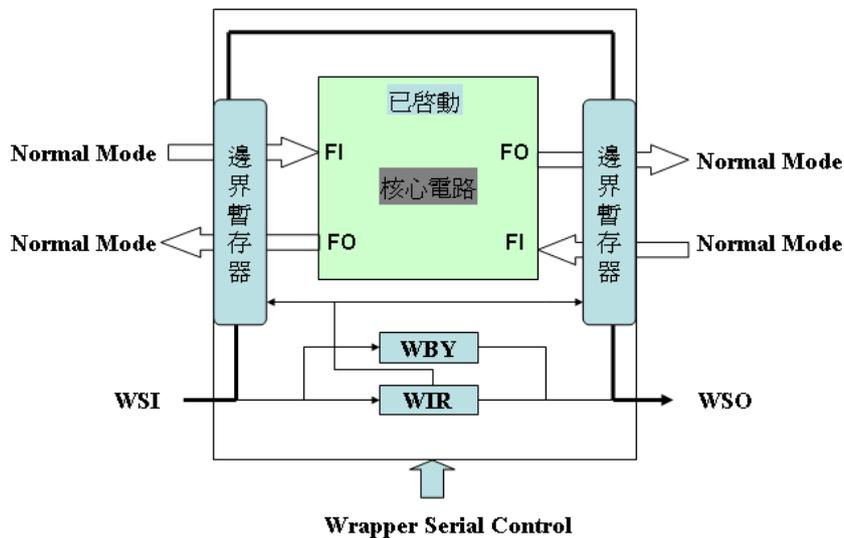


圖 2-12. WS_PRELOAD 指令

6. WP_PRELOAD: 如圖 2-13 所示，測試資料以並列(Parallel)方式載入，並將邊界暫存器(Wrapper Boundary Register)分割成多個部份。

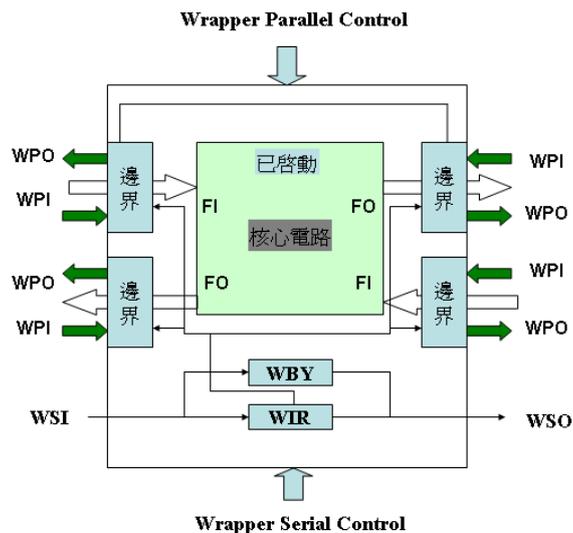


圖 2-13. WP_PRELOAD 指令

7. WS_CLAMP：邊界暫存器(Wrapper Boundary Register)將本來要載入的測試資料傳送出去，此指令會由 Wrapper serial Control 來的控制，測試資料由 WSI (Wrapper serial input) 輸入，經由旁路暫存器(Bypass Register)，從 WSO (Wrapper serial output)輸出到下一個核心。
8. WS_INTEST_RING：如圖 2-14 所示，使用於內部測試，全部的 Wrapper Cell 會連成串。其測試方法為：將測試資料從 WSI (Wrapper serial input)輸入經過 Shift 到適當位置，到 Update 時將值送入核心測試，Capture 時再將核心反應結果抓取到 Wrapper Cell 上，從 WSO (Wrapper serial output)輸出。

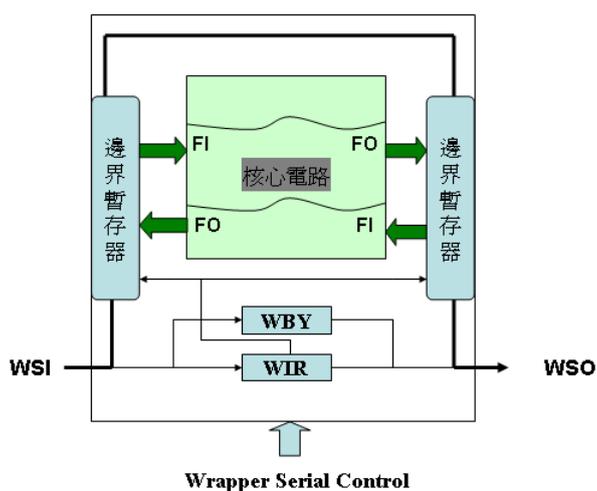


圖 2-14. WS_INTEST_RING 指令

9. WS_INTEST_SCAN：，如圖 2-15 所示，此指令會將內部的 Scan Chain 連接在邊界暫

存器(Wrapper Boundary Register)之間。

其測試方法為：將資料由 WSI (Wrapper serial input)輸入經過 Shift 將資料傳進 Scan chain 上，由 Update 時將值送入核心測試，Capture 時再將核心反應結果抓取到 Wrapper Cell 上，從 WSO (Wrapper serial output)輸出。

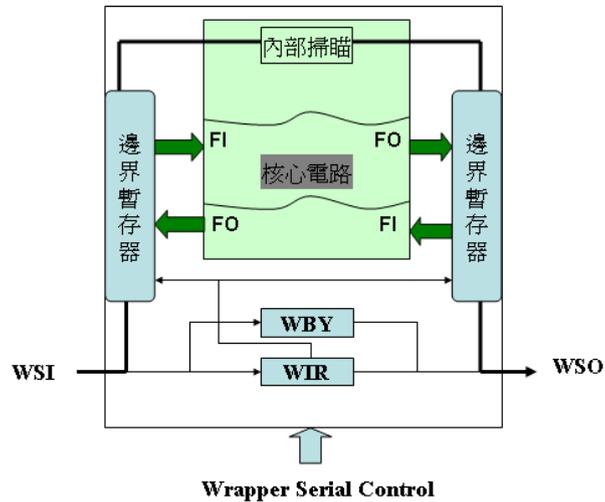


圖 2-15. WS_INTEST_SCAN 指令

2.2. IEEE 1500 腳位簡化

IEEE 1500 有八支接腳而 IEEE 1149.1 有五隻接腳，我們使用 IEEE 1149.1 的 TAP 來控制 IEEE 1500 的動作。由圖 2-16 所示，我們將 IEEE 1500 的八支接腳做簡化腳位的功能以符合 IEEE 1149.1 的 TAP 動作方式。

- (1) IEEE 1500: WSI、WSO、WRCK、WRSTN、ShiftWR、UpdateWR、CaptureWR、SelectWR
- (2) IEEE 1149.1: TMS、TCK、TDI、TDO、TRST*

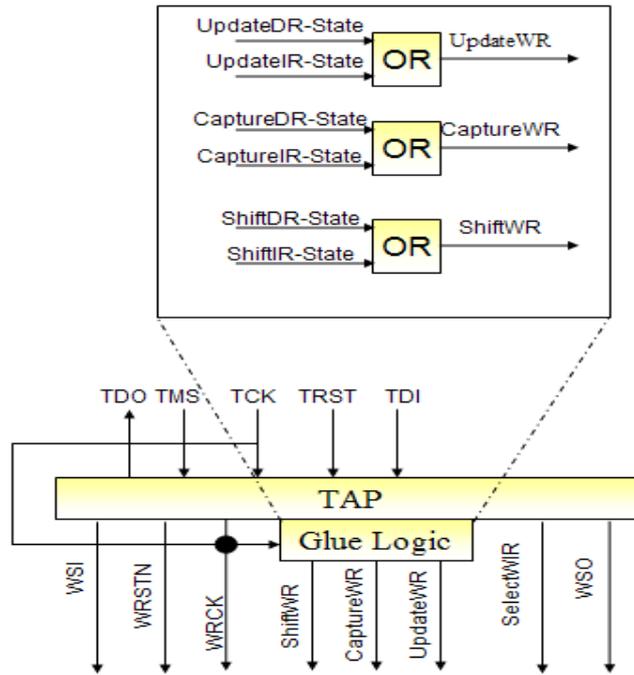


圖 2-16. IEEE-1149.1 TAP to IEEE-1500 WSP interface logic[5]

其 IEEE-1500 的八支接腳控制訊號與 IEEE 1149.1 四支接腳控制訊號可由表格 2-2 看出對應的關係：

表格 2-2 IEEE-1500 與 IEEE 1149.1 對應的關係

IEEE 1149.1	IEEE-1500
TCK	WRCK
TRST	WRSTN
TDI	WSI
TDO	WSO
ShiftIR State	SelectWIR = 1 ShiftWR = 1
ShiftDR State	SelectWIR = 0 ShiftWR = 1
UpdateIR State	SelectWIR = 1 UpdateWR = 1
UpdateDR State	SelectWIR = 0 UpdateWR = 1
CaptureIR State	SelectWIR = 1 CaptureWR = 1
CaptureDR State	SelectWIR = 0 CaptureWR = 1

2.3. 測試存取埠控制器(TAP controller)與邊界掃描測試流程之控制

如圖 2-17、2-18 所示，利用 TMS 的 0/1 變化和 TCK 來控制 TAP 控制器中的有限狀態機(FSM)，其邊界掃描測試主要流程為[7]：

Step 1. EXTEST 指令被讀取和解碼，其 TAP 控制器的狀態轉換為下：

Test-logic-reset → Run-test/idle → Select-DR-scan → Select-IR-scan → Capture-IR → Shift-IR → ... → Exit-IR → Update-IR →

Step 2. 指令解碼器將解碼後，TAP 控制器會設定好 TDI 到 TDO 的路徑並對應到該資料暫存器。測試樣本(test pattern)以串列(Serial)方式通過邊界掃描暫存器，其 TAP 控制器的狀態轉換為下：

Select-DR-scan → Capture-DR → Shift-DR → ... → Exit-DR →

Step 3. 將測試樣本(Test pattern)以並列(Parallel)方式送至 Update 暫存器，其 TAP 控制器的狀態轉換為下：

Update-DR → Select-DR-scan → Capture-DR →

Step 4. 抓取測試反應(Test response)結果並以串列(Serial)方式輸出通過邊界掃描暫存器到 TDO 來做比較，其 TAP 控制器的狀態轉換為下：

Capture-DR → Shift-DR → ... → Exit-DR →

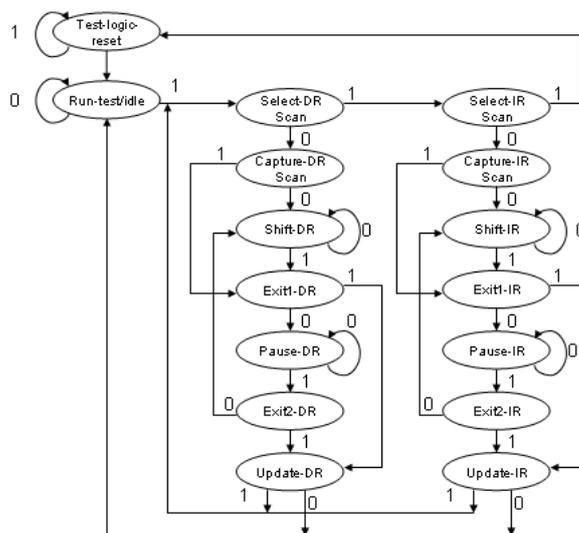


圖 2-17. IEEE Std.1149.1 TAP 控制器

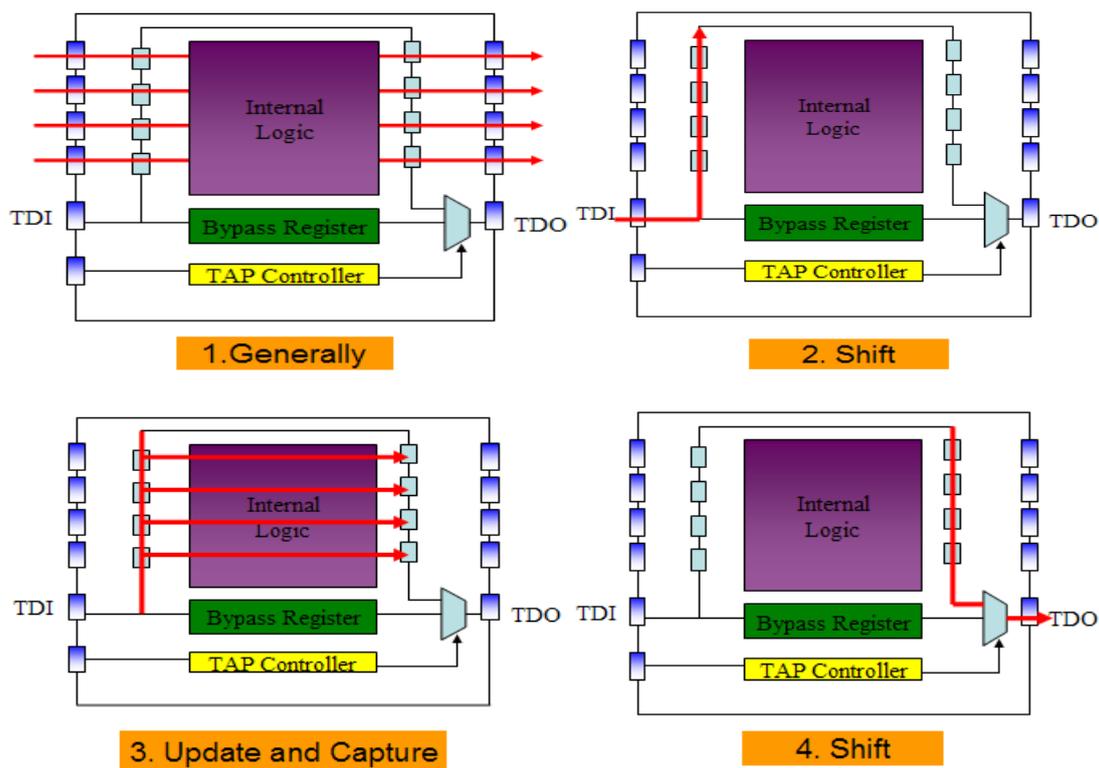


圖 2-18. IEEE-1149.1 邊界掃描

2.4. 轉換 IEEE-1500 的 Test Pattern

由於 test pattern 數量通常非常的龐大，為了保證 IEEE-1500 在轉換 Test Pattern 的過程中，能夠完全正確無誤，所以我們使用 testbench 中 fdisplay 指令，將正常模式送出的輸入測試資料 Dump 出來，並利用 C 語言來轉換以符合 IEEE-1500 格式的 Test Pattern。

2.4.1. 測試向量擴展 (Test Pattern Expansion)

要確保所有輸入的部分都要輸出，在讀入檔案之後，我們計算輸入接腳個數(Input Pin Number)和輸出接腳個數(Output Pin Number)，並依照圖 2-19 公式來做測試向量擴展。

$$\text{Shift times} = \text{MAX} \{ \text{Input Pin Number} , \text{Output Pin Number} \}$$

圖 2-19. 測試向量擴展公式

在這邊我們舉個例子，如圖 2-20 顯示出 Input Pin Number = 10 Pin，Output Pin Number = 13 Pin，所以由圖 2-19 測試向量擴展公式中得知：

$$\text{Shift times} = \text{MAX} \{10, 13\} = 13$$

故我們需要再輸入的測試向量前面做測試向量擴展(Test Pattern Expansion)3 bits。

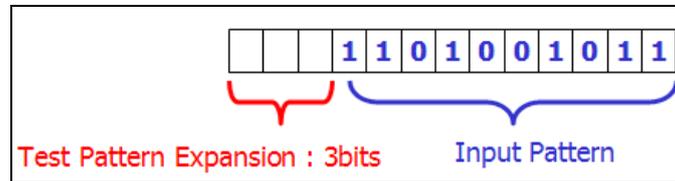


圖 2-20. 測試向量擴展

2.4.2. 演算法

Test Pattern 轉換演算法說明如下：

```

Begin
  While (Read test_pattern)
  {
    OutCount=OutPattern_bits; //calculate the bit number in OutPattern
    InCount = InPattern_bits; //calculate the bit number in In Pattern
    If(OutCount>InCount)
    {
      For(i= InCount; i<OutCount; i++)
      {
        print test_pattern[i+1]; // Expansion test pattern
      }
      print InPattern
    }
    Else // If InPattern > OutPattern
      print InPattern
  }
End

```

圖 2-21. 測試向量擴展演算法

2.4.3. 流程圖說明

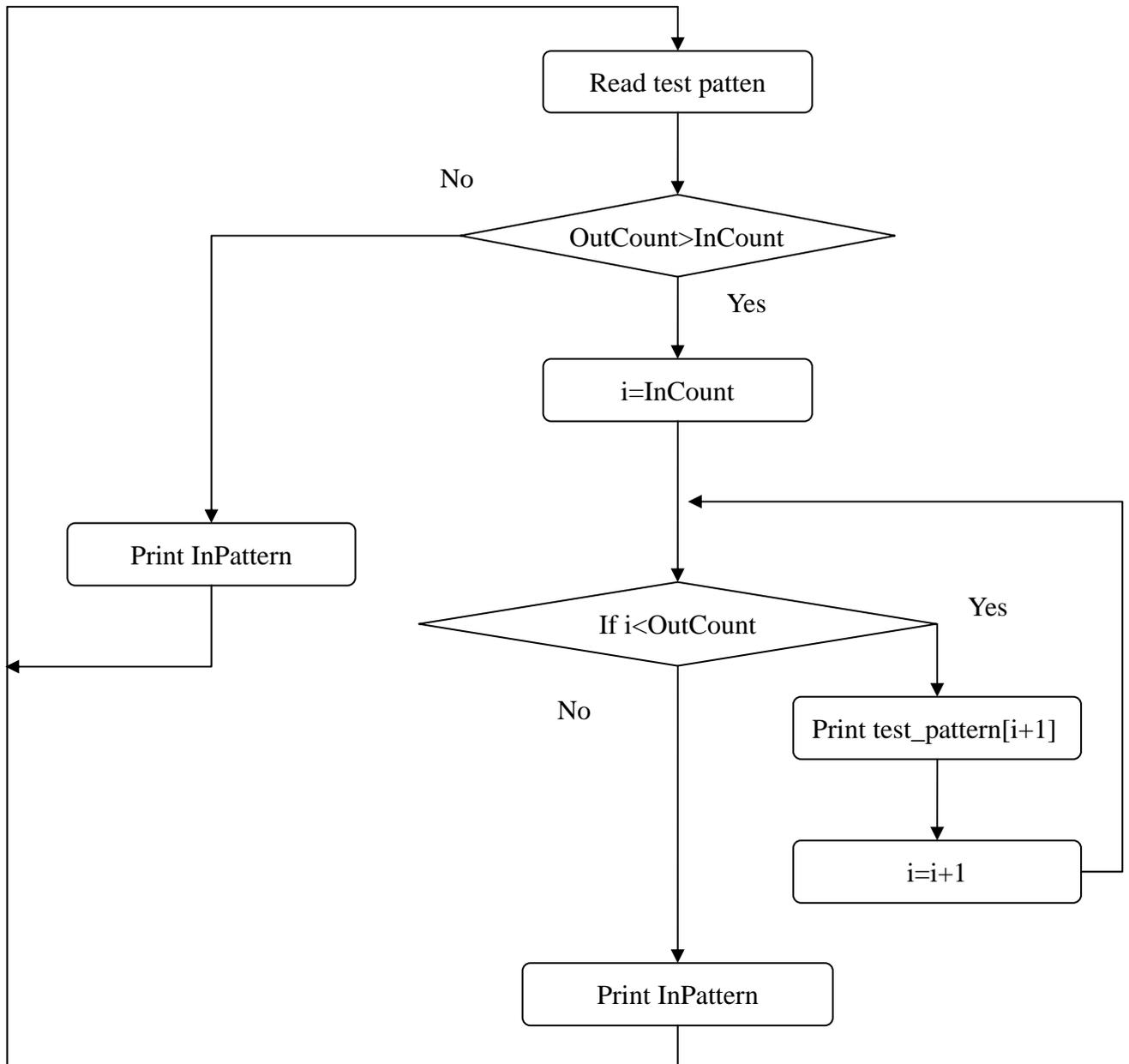


圖 2-22. 測試向量擴展流程圖

2.5. BIST 架構介紹

本節首先會先介紹內建自我測試架構，下一章節我們會將 IEEE-1500 與 BIST 架構結合做一個完整的說明。

2.5.1. 內建自我測試

內建自我測試，是指在原始電路功能以外，在加上其他的電路來測試電路是否正確。而內建自我測試架構如圖 2-23 所示，它主要可分為：

1. 測試向量產生器(Test Pattern Generator)：用來產生測試向量，設計者可依據自己所需要的功能去設計它。
2. 待測電路(Circuit Under Test)：由 Test Pattern Generator 產生測試向量送到 Circuit Under Test 來做測試。
3. 資料壓縮器和比較器(Data Compressor 和 Comparator)：經由 Data Compressor 將 Circuit Under Test 的輸出結果做資料壓縮。最後再經由原先儲存在 ROM 或其他儲存裝置內的正確結果和壓縮過的測試結果輸入至比較器做比對，若是比對結果不相同則表示電路功能有錯誤，若是比對結果相同則表示電路功能正常。

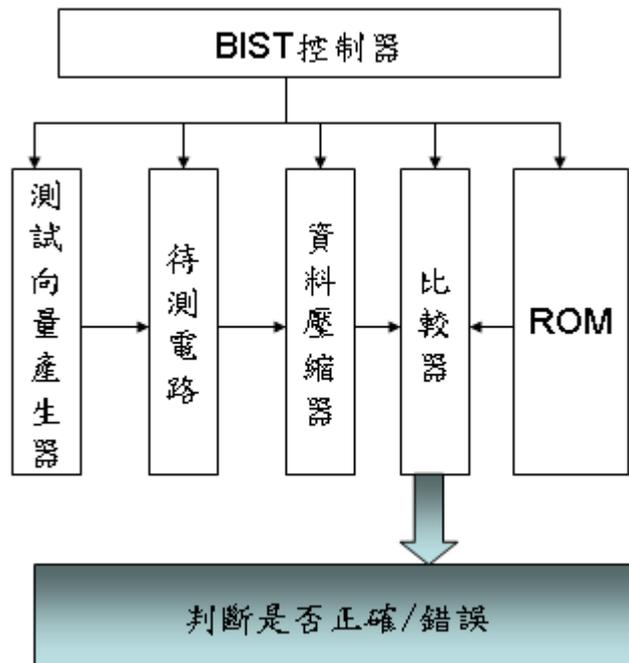


圖 2-23. 內建式自我測試架構圖

2.5.2. 線性回饋移位暫存器

兩種線性回饋移位暫存器(LFSR)主要都是由 D Flip-Flop 和 Exclusive-OR Gate 組成。[5] 圖 2-24 為第一型為 External-XOR LFSR，而圖 2-25 為第二型為 Internal-XOR LFSR，其特徵多項式 Characteristic Polynomial: $g(x) = g_n x^n + g_{n-1} x^{n-1} + \dots + g_0 x^0$

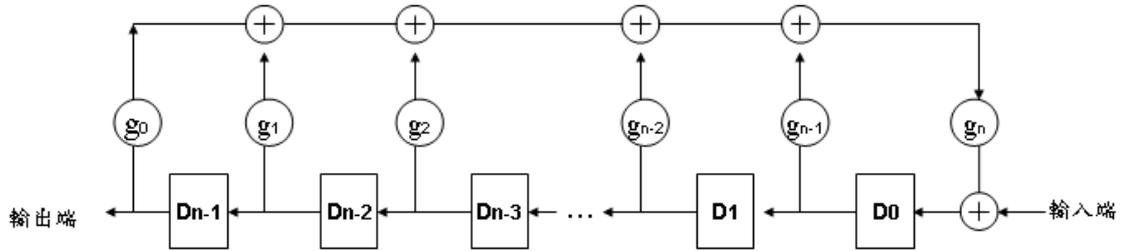


圖 2-24. External-XOR LFSR

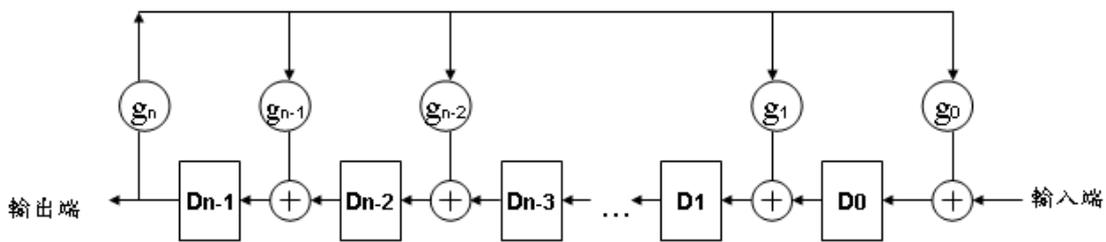


圖 2-25. Internal-XOR LFSR

在這多項式中， x 是延遲運算式(Delay Operator)， D_n 是正反器， g_n 是迴授連接訊號(Feedback Connections)。 g_n 有連接正反器則為 1，否則為 0。

N bits 的線性移位暫存器可產生最大的測試向量長度為 $2^n - 1$ 。

以圖 2-26 為例，此為一個四位元的 External-XOR LFSR，暫存器初始的 Seed 值為 0001，由圖得知 $g_0 = 1$ ， $g_1 = 1$ ， $g_4 = 1$ ，故特徵多項式(Characteristic Polynomial)計算如圖 2-27 所示，而圖 2-28 顯示 15 種 LFSR 暫存器內部序列狀態。

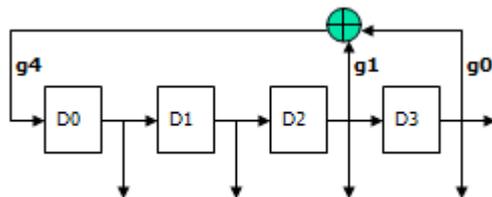


圖 2-26. 4-Stage External-XOR LFSR

$$\begin{aligned}
 \text{Characteristic Polynomial} &= g_0 + g_1x + g_2x^2 + g_3x^3 + g_4x^4 \\
 &= 1 + 1 \times x + 0 \times x^2 + 0 \times x^3 + 1 \times x^4 \\
 &= 1 + x + x^4
 \end{aligned}$$

圖 2-27. 4-Stage External-XOR LFSR 特徵多項式

D0	D1	D2	D3	Decimal
0	0	0	1	1
1	1	0	0	12
0	1	1	0	6
0	0	1	1	3
1	1	0	1	13
1	0	1	0	10
0	1	0	1	5
1	1	1	0	14
0	1	1	1	7
1	1	1	1	15
1	0	1	1	11
1	0	0	1	9
1	0	0	0	8
0	1	0	0	4
0	0	1	0	2
0	0	0	1	1

圖 2-28. 4-Stage External-XOR LFSR 狀態

2.5.3. MISR (Multiple Input Signature Register)

MISR(Multiple Input Signature Register)為多重訊號輸入的簽章分析法，由兩種圖 2-29 為 Internal-XOR MISR，其特徵多項式(Characteristic Polynomial)為：

$$g(x) = g_n x^n + g_{n-1} x^{n-1} + \dots + g_0 x^0。$$

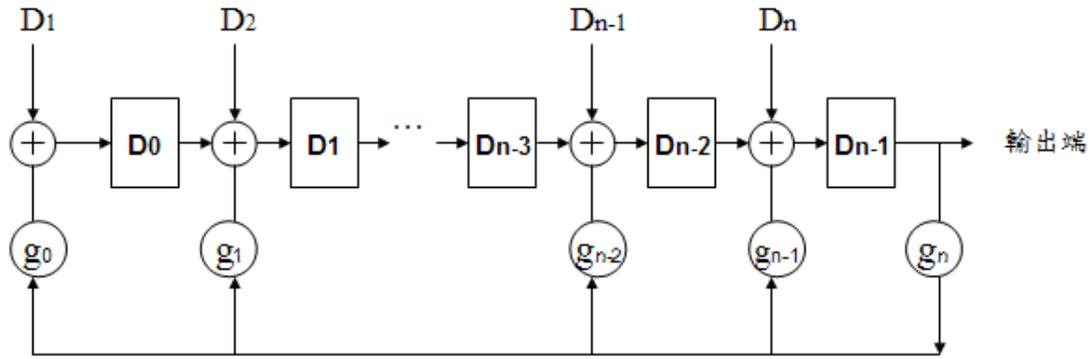


圖 2-29. Internal-XOR MISR

以圖 2-30 為例，此為一個四位元的 MISR，暫存器內部目前的狀態值(Initially State Value) 為 1111，而輸入訊號的值為 1010，如圖 2-31 得知，經由運算後的結果為 0101，所以新的簽章(Signature)結果為 0101。

第 i 個時脈

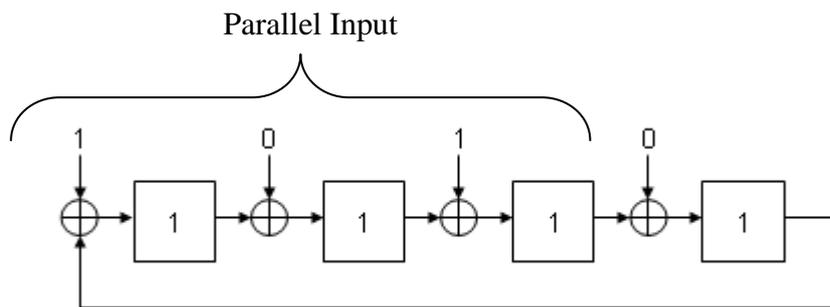


圖 2-30. 第 i 個時脈的連續掃瞄電路

第 i+1 個時脈

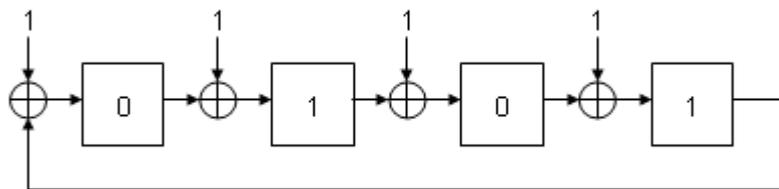


圖 2-31. 第 i+1 個時脈 MISR 架構

將每一次產生的測試結果送到 MISR 去做簽章分析(Signature Analysis)，在全部的測試向量都做測試以後就將在 MISR 裡的簽章傳送出來並判斷此待測電路是否有錯誤，如果簽章結果是一樣的，我們稱此電路為 Fault-Free 電路，否則此電路就為有 Faulty 的電路，以此方式就可以減少測試時間。

但是此方式也有缺點，因為簽章分析的技術是一種壓縮的方法，所以就會產生失真問題，

我們稱為 Alias 現象，所以就有學者針對此現象做一些 Aliasing-Free 的探討[8][9][10]。若是 n 位元的測試結果暫存器，可對印到 2^n 個可能結果，顧名思義，對印不到之可能結果的機率為 $1/2^n$ 。但是這種機率非常小，無法偵測到錯誤的機率為 $1/2^n\%$ 。

2.6. BIST 與邊界掃描電路結合

在這節中，我們會介紹輸入的 Boundary Scan Cell 如何合併到 LFSR 和輸出的 Boundary Scan Cell 如何合併到 MISR，並詳細說明 IEEE-1500 與 BIST 架構整合的運作機制[11]。

2.6.1. 輸入 Boundary Scan Cell 合併到 LFSR

如圖 2-32 為輸入 Boundary Scan Cell 合併到 LFSR，編號為 20, 30, 40，其 Characteristic Polynomial 為 $1 + X^3 + X^4$ 。

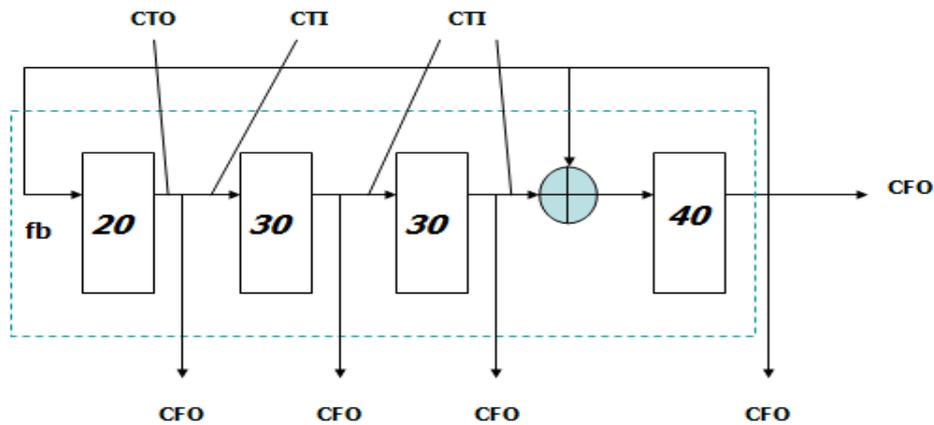


圖 2-32. 輸入 Boundary Scan Cell 合併 LFSR

圖 2-33 為輸入 Boundary Scan Cell 編號 20，它是由兩個多工器和一個 D 型正反器所組成。多工器編號 21 的第一個和第三個資料輸入為 CTI，第二個資料輸入為 fb(feedback)，第四個資料輸入連到 CFO。D 型正反器編號 22，clock 輸入為 mWRCK，其中 m 表示 modify 是指為修改後的 WRCK，資料輸入為多工器編號 21 的輸出，資料輸出為 CTO。第二個多工器編號 23，有一輸入為 CFI，另一輸入為 D 型正反器編號 22 的輸出，由 AC 來控制輸出到 CFO 的值。

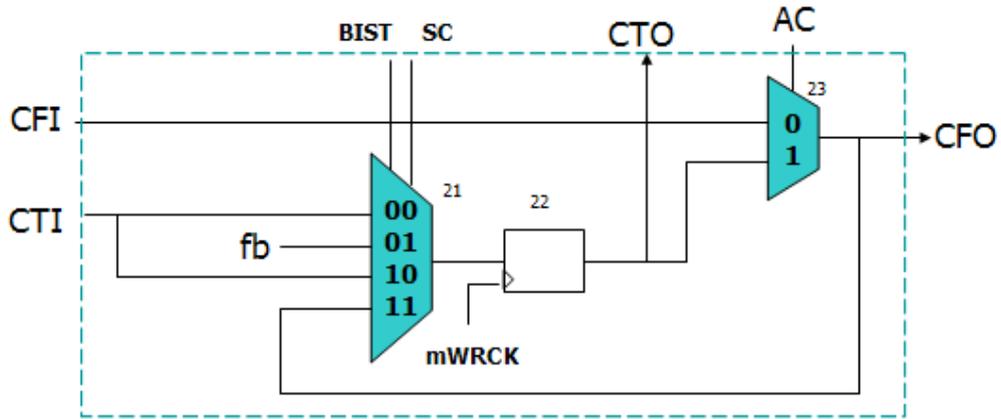


圖 2-33. 輸入 Boundary Scan Cell 編號 20

圖 2-34 為輸入 Boundary Scan Cell 編號 30，多工器編號 31 的第一、二、三個資料輸入為 CTI，連通道前一個 Cell 的 CTO，第四個資料輸入連到 CFO。

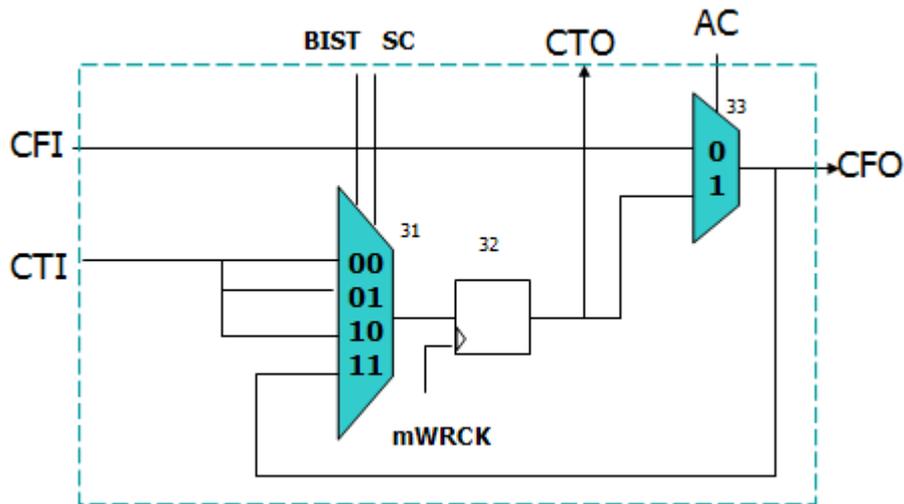


圖 2-34. 輸入 Boundary Scan Cell 編號 30

圖 2-35 為輸入 Boundary Scan Cell 編號 40，多工器編號 41 的第一、三個資料輸入為 CTI，連通道前一個 Cell 的 CTO，同時為 XOR 閘編號 44 的輸入，第二個資料輸入為 XOR 閘編號 44 的輸出，第四個資料輸入連到 CFO。

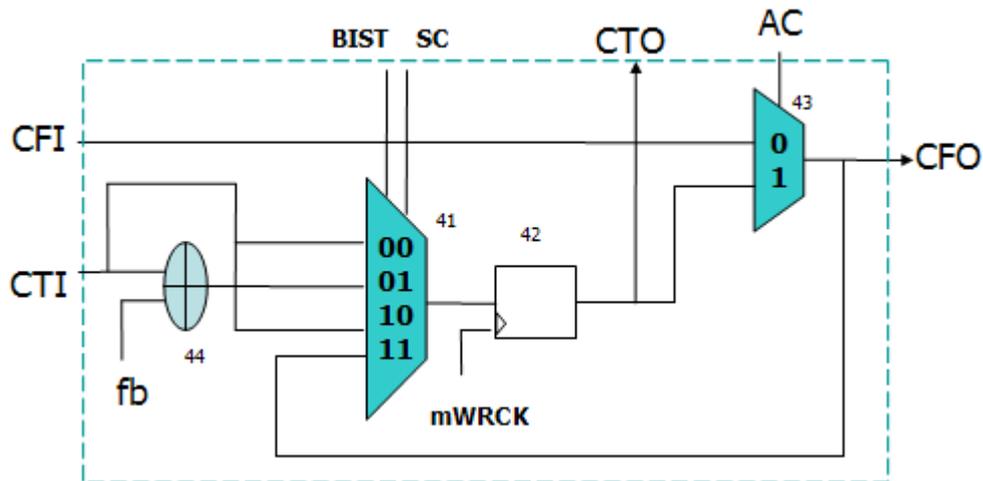


圖 2-35. 輸入 Boundary Scan Cell 編號 40

2.6.2. 輸出 Boundary Scan Cell 合併到 MISR

如圖 2-36 為輸入 Boundary Scan Cell 合併到 MISR，編號為 50, 60, 70，其 Characteristic Polynomial 為 $1 + X^3 + X^4$ 。

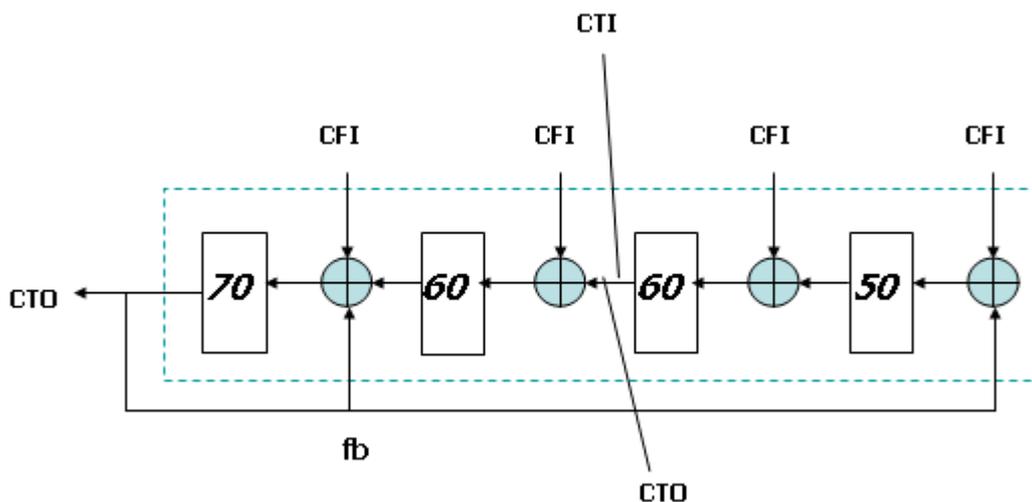


圖 2-36. 輸出 Boundary Scan Cell 合併 MISR

圖 2-37 為輸出 Boundary Scan Cell 編號 50，它是由兩個多工器和一個 D 型正反器所組成。多工器編號 51 的第一個和第三個資料輸入為 CTI，第二個資料輸入連通到 XOR 閘編號 54 的輸出，而 XOR 閘編號 54 的輸入為 CFI 和 fb (feed-back)，第四個資料輸入連到 CFO。

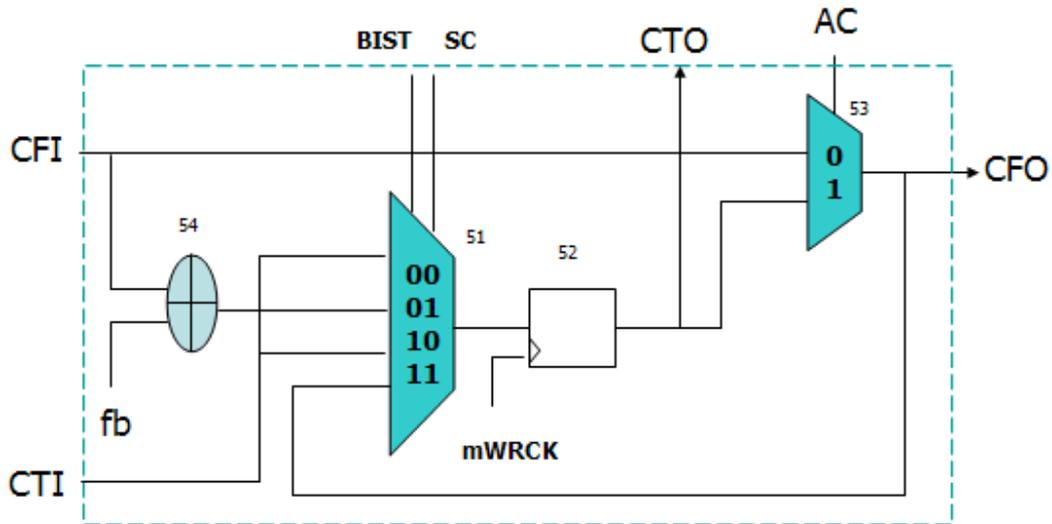


圖 2-37. 輸入 Boundary Scan Cell 編號 50

圖 2-38 為輸入 Boundary Scan Cell 編號 60，多工器編號 61 的第一、三個資料輸入為 CTI，連通道前一個 Cell 的 CTO，第二個資料輸入連通到 XOR 閘編號 64 的輸出，而 XOR 閘編號 64 的輸入為 CFI 和 CTI，第四個資料輸入連到 CFO。

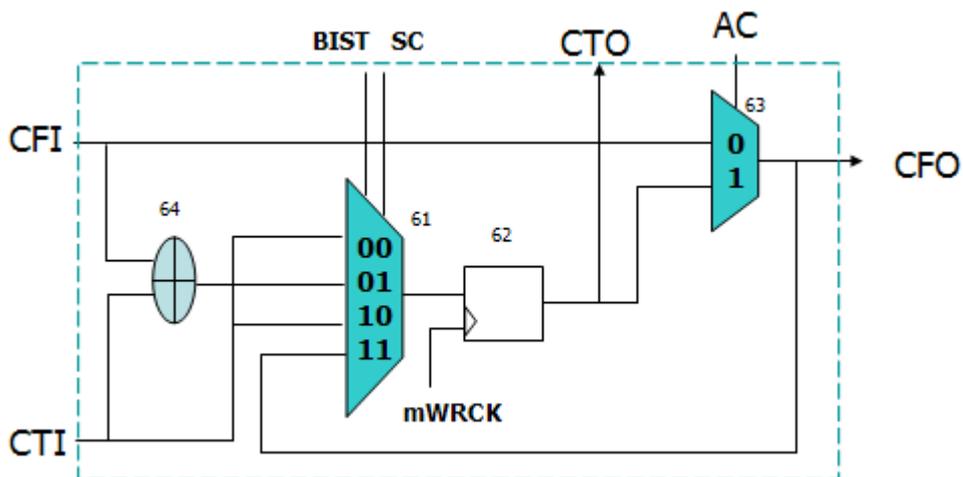


圖 2-38. 輸入 Boundary Scan Cell 編號 60

圖 2-39 為輸入 Boundary Scan Cell 編號 70，多工器編號 71 的第一、三個資料輸入為 CTI，連通道前一個 Cell 的 CTO，同時為 XOR 閘編號 74 的輸入，第二個資料輸入為 XOR 閘編號 75 的輸出，第四個資料輸入連到 CFO。

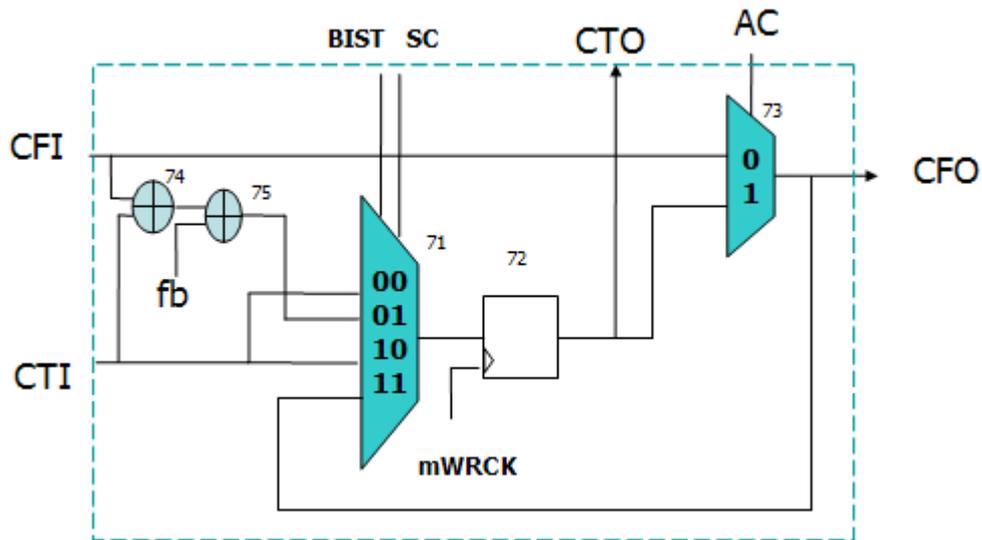
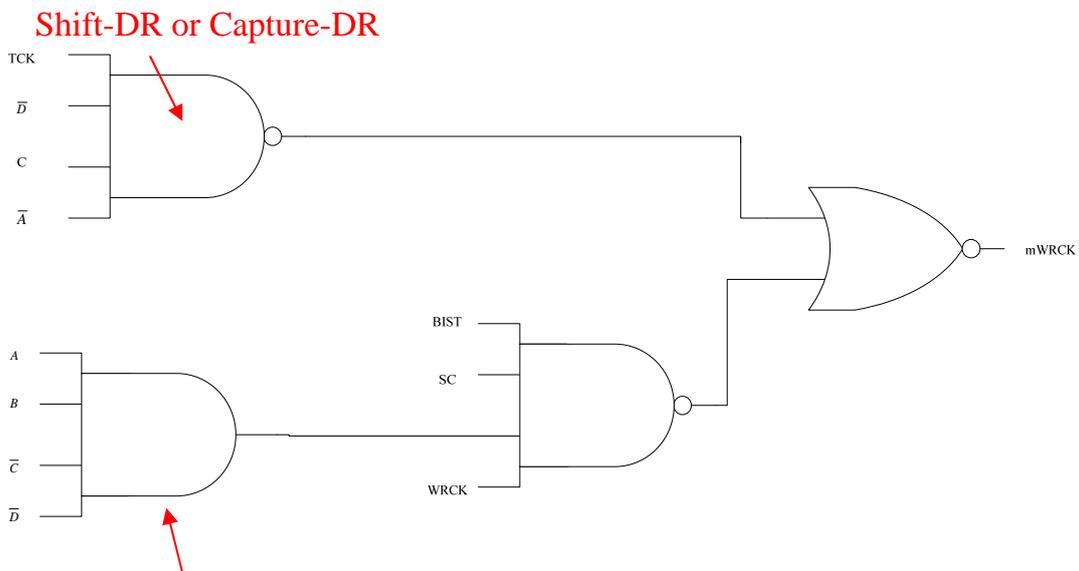


圖 2-39. 輸入 Boundary Scan Cell 編號 70

2.6.3. 修改 WRCK 信號設計

如圖2-40為電路設計中邊界掃描暫存器(Wrapper Boundary Cell)的WRCK修改其信號命名為mWRCK信號。當RUNBIST指令以並行(Parallel)方式載入更新暫存器(Update Register)，然後將指令加以解碼，並對應到該指令去執行和傳送資料到要使用的資料暫存器。而TAP控制器會在Run-Test/Idle控制的狀態。表格3-1為TAP控制器的狀態分配表。



Run-Test/Idle 圖 2-40. 修改電路設計的 WRCK 信號

表格 2-3 TAP 控制器的狀態分配

Controller state	State assignment ABCD	Controller state	State assignment ABCD
Exit2_DR	0 0 0 0	Exit2_IR	1 0 0 0
Exit1_DR	0 0 0 1	Exit1_IR	1 0 0 1
Shift_DR	0 0 1 0	Shift_IR	1 0 1 0
Pause_DR	0 0 1 1	Pause_IR	1 0 1 1
Select_IR_Scan	0 1 0 0	Run_Test_Idle	1 1 0 0
Update_DR	0 1 0 1	Update_IR	1 1 0 1
Capture_DR	0 1 1 0	Capture_IR	1 1 1 0
Select_DR_Scan	0 1 1 1	Test_Logic_Reset	1 1 1 1

2.6.4. RUNBIST 指令

當 RUNBIST 指令被載入指令暫存器並解碼，控制信號 BIST 為邏輯 0，SC 為邏輯 1。如圖 2-41 所示，輸入 Boundary Scan Register IBS，Boundary Scan Cell 編號 20 和 40 的多工器編號 21 和 41 選擇 feedback input fb，而 Boundary Scan Cell 編號 30 的多工器編號 31 選擇 CTI。輸出 Boundary Scan Register OBS，Boundary Scan Cell 編號 50、60 和 70 的多工器編號 51、61 和 71 選擇 XOR 閘 54、64 和 75。

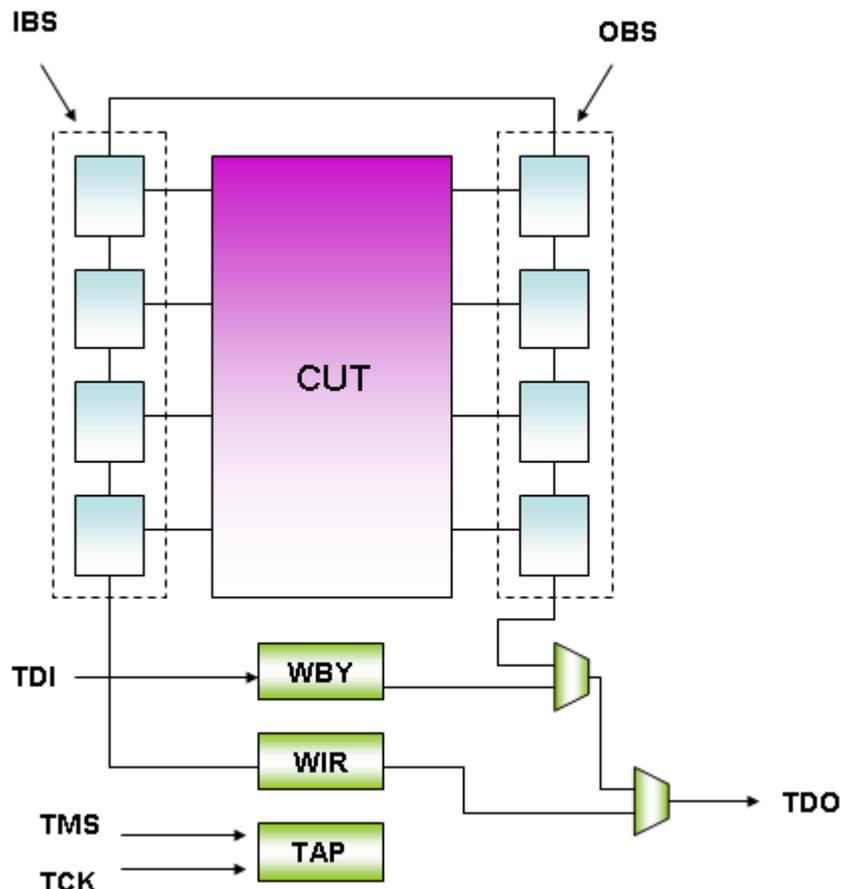


圖 2-41. BIST 邊界掃瞄電路

2.6.5. IEEE-1500 與 BIST 架構整合說明

如圖 2-41 輸入 Boundary Scan Register IBS 為測試向量產生器(Test Pattern Generator)提供測試向量到電路的輸入端，另一方面輸出 Boundary Scan Register OBS 為一個輸出反應分析器(Output Response Analyzer)，其運作方式如下：

- 【1】 當 RUNBIST 指令被載入並解碼，並設定 LFSR 的 Seed 值，例如:0001；和 MISR 的 Seed 值，例如:0000。
- 【2】 當對應的指令為 RUNBIST 指令時，會同時產生對應的控制信號。此時，輸入 Boundary Scan Register IBS 運作 LFSR，輸出 Boundary Scan Register OBS 運作 MISR。
- 【3】 TAP 控制器運作在 Run-Test/Idle 狀態，並在 15 clock cycle 完成 BIST 運作。
- 【4】 當 BIST 運作結束，將 Scan out 結果經由輸出 Boundary Scan Register OBS(BIST=1，SC=0)，並比較正確的 Signature，來偵測是否有錯誤存在。

[LFSR 的 Pattern Test 分析]:

我們使用 IEEE-1500 和 BIST 架構整合是採用 Test-Per-Clock BIST 的方式，在每一個 Clock Cycle 動作時就會有一筆新的測試向量送到電路去做測試，用以減少測試時間。

使用 LFSR 產生 Test Pattern 來對電路做測試的架構有兩種方式：

一、Deterministic Pattern Test(決定性的向量測試)，此種方式可以先計算所需要的測試向量來對電路做測試，例如[12][13][14][15]的方法，但此種方式也有其優缺點所在。

優點：Test Coverage 較高。

缺點：電路設計較為困難。

二、Random Pattern Test(亂數向量測試)：

優點：電路設計較為簡單。

缺點：若需要達到較高的 Test Coverage 需要較長的測試時間。

3. 運作機制

3.1.3DG 繪圖晶片架構與測試機制

圖 3-1 為整體 3DG 架構圖，由圖可以得知為了測試如此複雜的架構需要非常完善的測試機制。在此次的 3DG 計畫中，我們以 Serial Testing mode 與 RUNBIST 兩種測試機制，來驗證 VS 與 FS 兩繪圖核心是否正確。

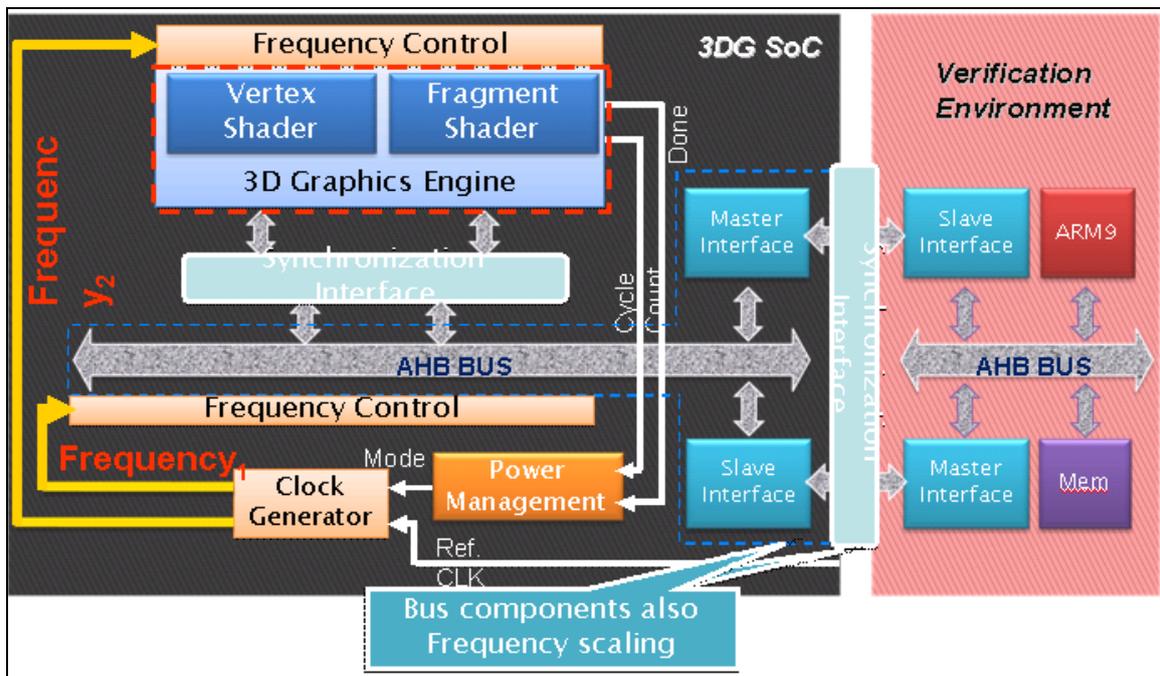


圖 3-1. 整體 3DG 架構圖

[測試機制說明]：

由於 Serial Testing mode 與 RUNBIST 兩種測試機制的 Boundary Scan Cell 都是直接與 IP 相連接，在這次計畫中，加入了 AHB BUS 的關係，故我們必須將 Boundary Scan Cell 安插在 IP 與 AHB Wrapper 中間，使得 IEEE-1500 with BIST 可以正常運作。圖 3-2 為繪圖晶片導入 IEEE-1500 with BIST 測試架構，可觀察到透過 IEEE-1500 with BIST 介面控制 Boundary Scan Cell，已達到 IEEE-1500 協定本身所定義的直接對 IP 作測試資料的存取。在設計上可利用 IEEE-1149.1 的標準，使得五個信號 (TDI、TDO、TMS、TRST、TCK)，可相容於 IEEE-1500 介面操作。可減少外部 IO 腳位的需求。

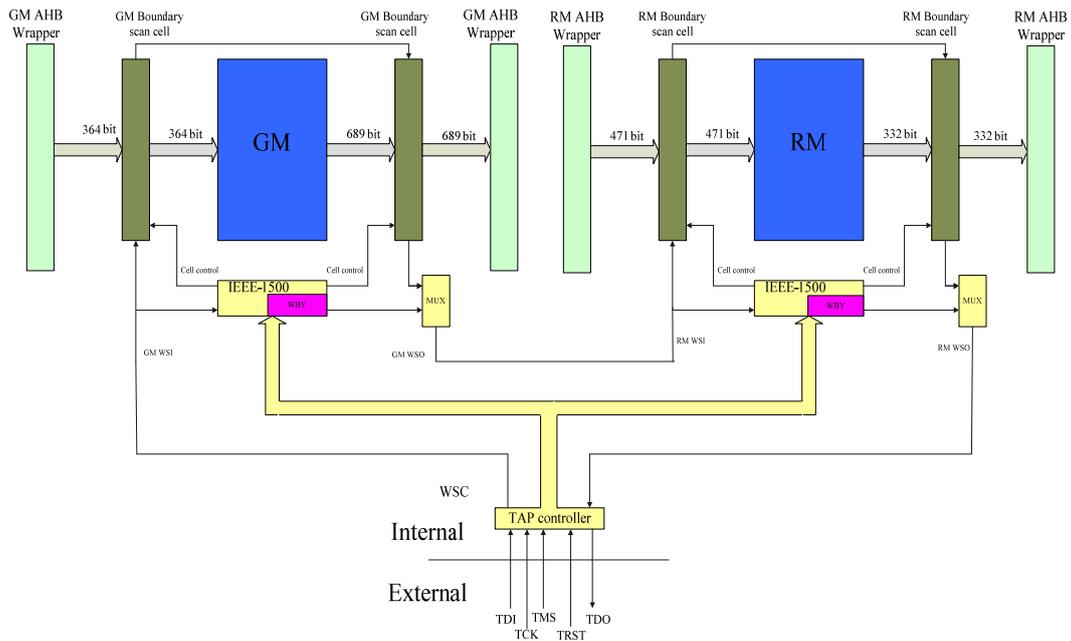


圖 3-2. 繪圖晶片導入 IEEE-1500 with BIST 測試架構

Serial Testing mode 與 RUNBIST 兩種測試機制在正常模式及測試模式的動作說明：

在正常模式時：Boundary Scan Cell 對即將要被測試的 IP 會呈現透明的狀態，也就是說 IP 並不知道有 Boundary Scan Cell 的存在。

對 IP 內部測試時：當我們選擇某個 IP 要對它作測試時，Boundary Scan Cell 會被連接到該 IP 的輸入輸出接腳，並經由輸出訊號作比對來判斷此 IP 功能是否正確。

3.2. JTAG 外部腳位規劃

IEEE-1500 測試機制在執行時有五支接腳 TCK、TRST、TMS，TDI 與 TDO，藉由這五支接腳來控制整個 IEEE-1500 的動作。如圖 3-3，可以利用 SEL 訊號這根訊號腳來切換 IEEE-1149.1 和 IEEE-1500 的訊號控制去傳送該訊號到適當的位子。

當 SEL 訊號 = 1，為 IEEE-1149.1 使用。

當 SEL 訊號 = 0，為 IEEE-1500 使用。

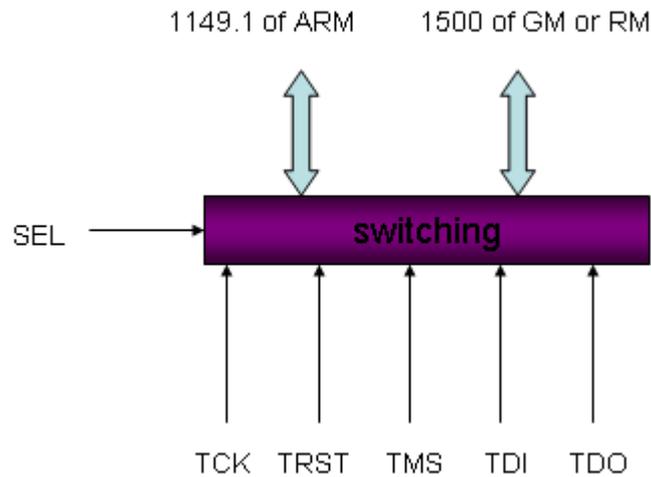


圖 3-3. IEEE-1149.1 和 IEEE-1500 外部腳位切換

3.3. VS 和 FS 測試向量擴展

我們將 Vertex Shader 與 Fragment Shader 的移位次數計算(Shift Times Calculate)分別做說明如下：

Vertex Shader：

在 Vertex Shader 中，Input Pin Number = 364 Pin，Output Pin Number = 689 Pin，移位次數計算(Shift Times Calculate):

$$\text{Shift times} = \text{MAX} \{364, 689\} = 689$$

故我們需要再輸入的測試向量前面做測試向量擴展(Test Pattern Expansion)。

Fragment Shader：

在 Fragment Shader 中，Input Pin Number = 471 Pin，Output Pin Number = 332 Pin，所以由圖 2-19 移位次數計算(Shift Times Calculate)中得知:

$$\text{Shift times} = \text{MAX} \{471, 332\} = 471$$

故我們只需要將輸入部分的 Test Pattern 全部都輸出即可。

4. 結果

本段將探討 IEEE-1500 Serial Testing mode 和 RUNBIST 其驗證的方法與過程，包括指令部份和資料部份的時序模擬圖和結果比對。

4.1. IEEE-1500 without BIST 之測試機制驗證

本段將探討 IEEE-1500 Serial Testing mode 驗證的方法與過程，包括指令部份和資料部份的時序模擬圖，並使用 Debussy&看架構圖和使用 nWave&看其模擬訊號結果。

4.1.1. 指令與資料時序模擬圖

如圖 4-1 所示，當 SelectWIR=1 時，會根據 State 模式的運作 Shift_IR、Update_IR 與 Capture_IR 指令，將指令部份傳送到指令暫存器內部。



圖 4-1.指令暫存器的運作模擬圖

如圖 4-2 所示，當 SelectWIR=0 時，會根據 State 模式的運作來 Shift_DR、Update_DR 與 Capture_DR 指令，將資料部份傳送到資料暫存器內部。

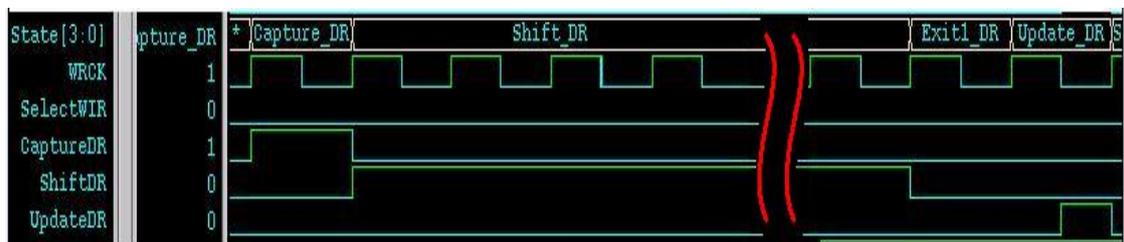


圖 4-2.資料暫存器的運作模擬圖

4.1.2. 測試模式和正常模式控制

如圖 4-3，在正常模式控制時我們將 $WRSTN = 1'b0$ ，此時 Boundary Scan Cell 為透明狀態，資料皆可傳遞過去。

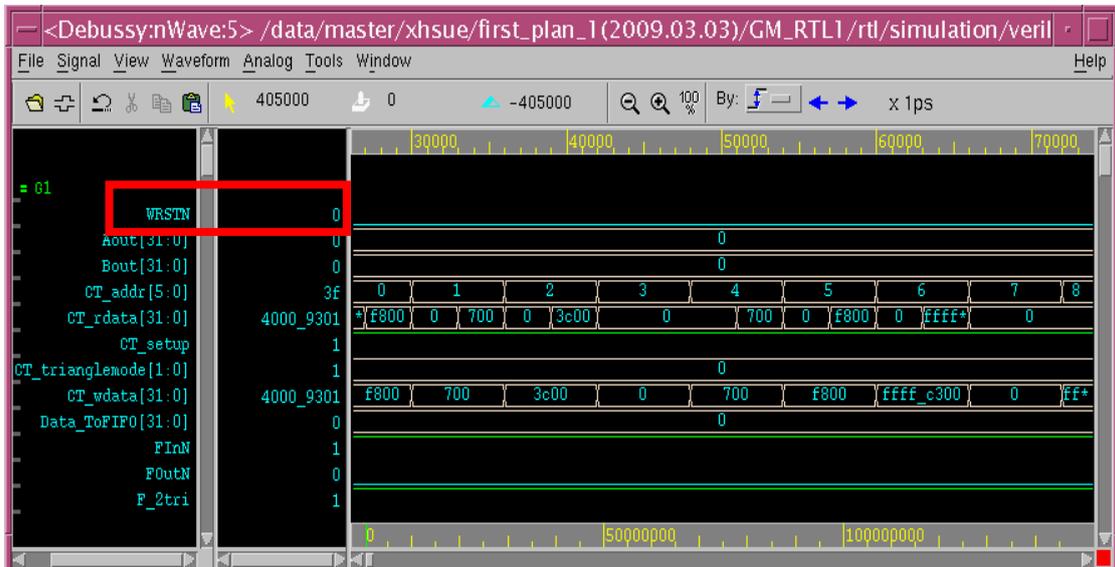


圖 4-3.執行正常模式

如圖 4-4，在測試模式控制時我們將 $WRSTN = 1'b1$ ，此時會依照 TAP 的狀態去執行對應的動作。

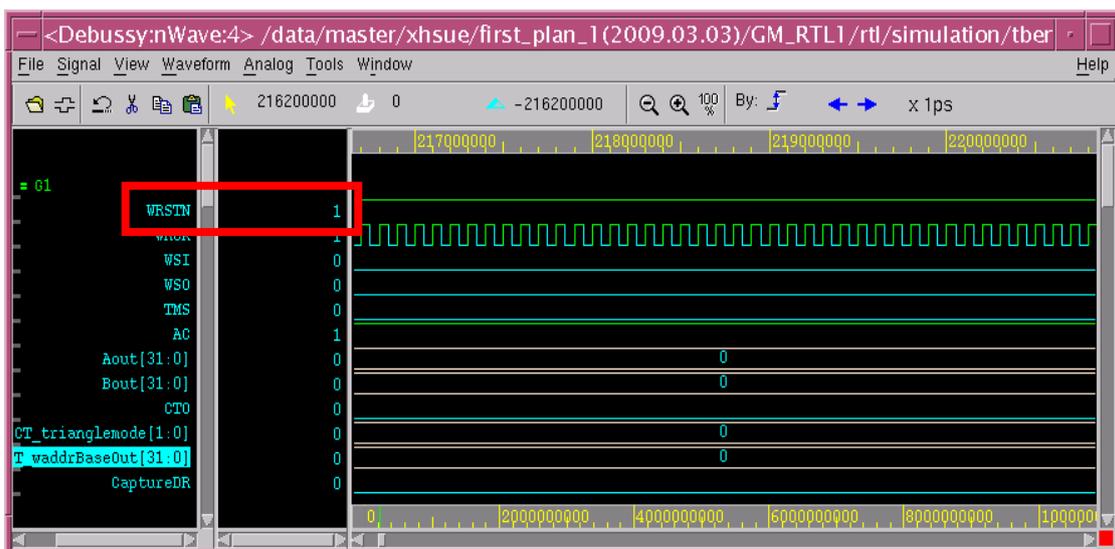


圖 4-4.執行測試模式

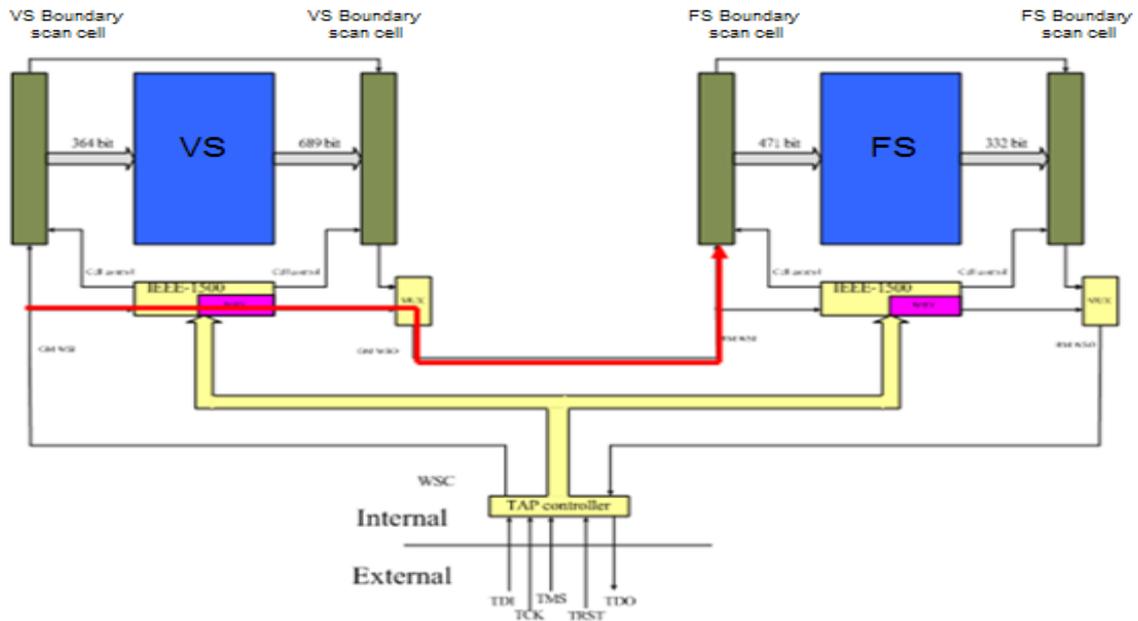


圖 4-6. FS 作測試時之路徑

4.1.4. VS 和 FS 以 IEEE-1500 without BIST 結果

以下各小節我們分別驗證出 Normal mode 和 Serial Testing mode 結果。

4.1.4.1. Normal mode 結果

如圖 4-7 為 Normal Mode 測試架構圖，我們將 VS 和 FS 的 Test Pattern 送進待測電路做測試，並將結果輸出與正確的輸出結果做比對，判斷是否正確無誤。

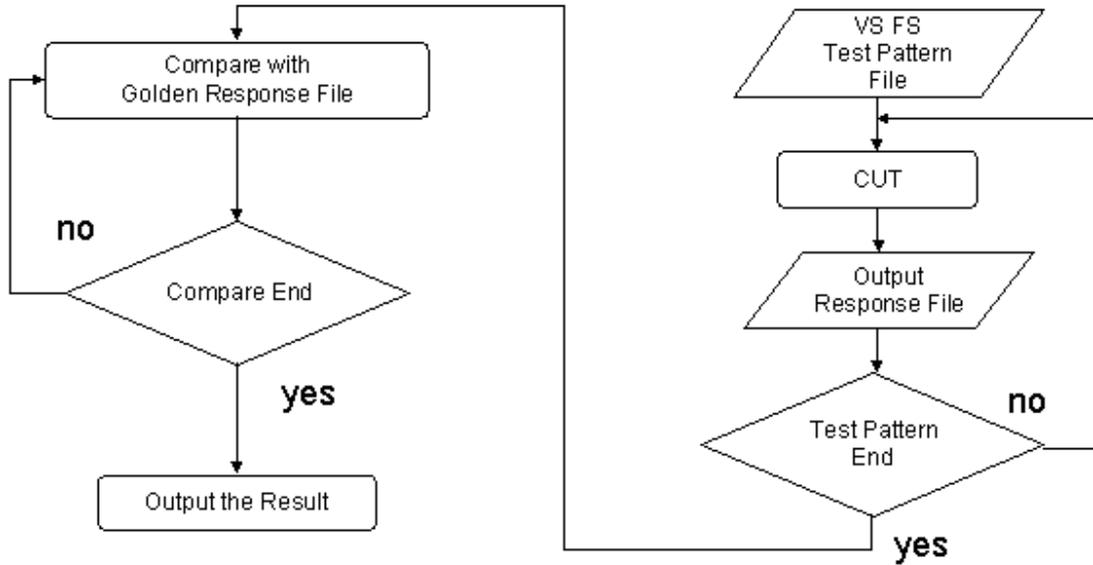


圖 4-7. Normal Mode 測試架構圖

為了確保 IEEE-1500 在包覆 Boundary Scan Cell 之前和之後兩者在 Normal Mode 環境下其結果相同，Boundary Scan Cell 皆可相訊號傳遞過去。圖 4-8 為 VS 的 Boundary Scan Cell 包覆前 Pattern 模擬，圖 4-9 為 VS 的 Boundary Scan Cell 包覆後 Pattern 模擬。



圖 4-8. VS 的 Boundary Scan Cell 包覆前 Pattern 模擬



圖 4-9. VS 的 Boundary Scan Cell 包覆後 Pattern 模擬

圖 4-10 為 FS 的 Boundary Scan Cell 包覆前 Pattern 模擬，圖 4-11 為 FS 的 Boundary Scan Cell 包覆後 Pattern 模擬。

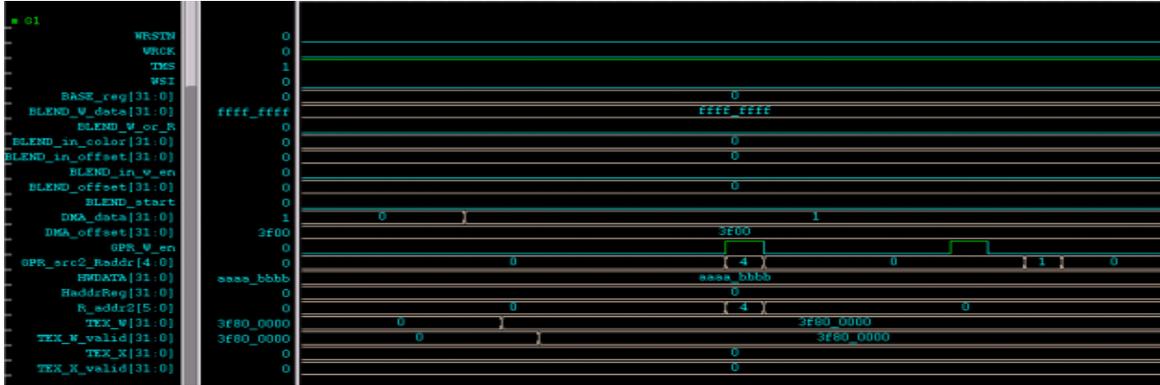


圖 4-10. FS 的 Boundary Scan Cell 包覆前 Pattern 模擬

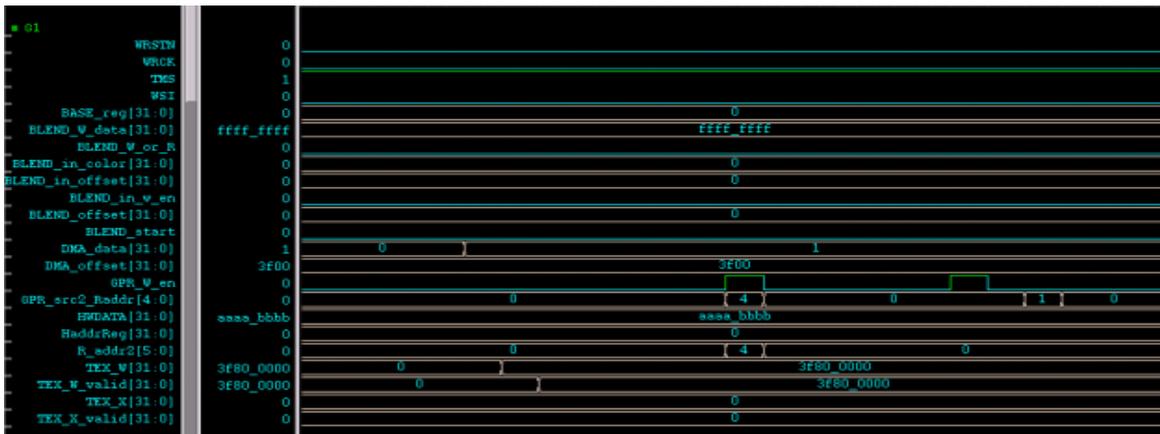


圖 4-11. FS 的 Boundary Scan Cell 包覆後 Pattern 模擬

4.1.4.2. Serial testing mode 結果比對

我們將電路串 Internal Scan Chain 用使用 Teramax 工具跑出 ATPG 的 Pattern 個數，所使用的 Fault Model 為 Stuck-at-Fault，以下為使用 Benchmark ISCAS85 電路[16]和 ISCAS89 電路[17]以及 VS 和 FS 來分析其 Test Coverage 為何，並將結果做比對。

1. Test Coverage 分析:

[Benchmark]

圖 4-12 顯示出使用 Benchmark c6288 在 Serial Testing mode 作測試所產生的 Pattern 個數為 58 筆和所達到的 Test Coverage 為 99.95%。

```

Uncollapsed Stuck Fault Summary Report
-----
fault class                code    #faults
-----
Detected                   DT      10631
  detected_by_simulation   DS      (10631)
Possibly detected         PT        0
Undetectable              UD        4
  undetectable-unused     UU      (4)
ATPG untestable          AU        0
Not detected              ND        5
  not-observed            NO      (5)
-----
total faults                10640
test coverage              99.95%
fault coverage             99.92%
-----

Pattern Summary Report
-----
#internal patterns        58
#full_sequential patterns 58
-----

```

圖 4-12. Benchmark c6288 在 Serial testing mode 測試結果

實驗結果整理如表格 4-1 為使用 Benchmark 電路作測試的結果，顯示出輸入個數和 Test Pattern 個數以及所需的 clk 時間並分析 Test Coverage 和 Fault Coverage 為何。

表格 4-1 Benchmark 在 Serial Testing mode 的結果

Circuit	#Input	#Test pattern	#Total clk	Test Coverage(%)	Fault Coverage(%)
s298	17	36	612	100	99.32
s344	24	29	696	100	99.31
s349	24	28	672	100	100
s510	25	78	1950	100	99.61
c432	36	80	1980	99.2	98.75
c499	41	59	2747	99.4	99.74
c1355	41	122	5002	99.76	100
c5315	178	113	20114	100	99.82
c6288	32	58	1632	100	99.95

[VS]

我們將 VS 串 Internal Scan Chain 並使用 Teramax 工具跑出其 ATPG 所得到的結果如圖 4-13 並分析如表格 4-2 所示。

Uncollapsed Stuck Fault Summary Report		
fault class	code	#faults
Detected	DT	61130
Possibly detected	PT	15119
Undetectable	UD	33467
ATPG untestable	AU	854
Not detected	ND	559348
total faults		669918
test coverage		10.79%
Pattern Summary Report		
#internal patterns		571
#full_sequential patterns		571

圖 4-13. VS 在 Serial Testing mode 測試結果

表格 4-2 VS 在 Serial Testing mode 測試結果

Circuit	#Input	#Output	#Test Pattern	#Total Clk	Test Coverage(%)
VS	364	689	571	393419	10.79

[RM]

我們將 RM 串 Internal Scan Chain 並使用 Teramax 工具跑出其 ATPG 所得到的結果如圖 4-14 所示並分析如表格 4-3 所示。

Uncollapsed Stuck Fault Summary Report		
fault class	code	#faults
Detected	DT	7180
Possibly detected	PT	1375
Undetectable	UD	32496
ATPG untestable	AU	2744
Not detected	ND	641043
total faults		684838
test coverage		1.21%
Pattern Summary Report		
#internal patterns		10
#full_sequential patterns		10

圖 4-14. FS 在 Serial testing mode 測試結果

表格 4-3 FS 在 Serial Testing mode 測試結果

Circuit	#Input	#Output	#Test Pattern	#Total Clk	Test Coverage(%)
RM	471	332	10	4710	1.21

2. 結果比對:

如圖 4-15 為 Serial Testing mode 測試架構圖，我們使用 Serial 掃描方式將 Test Pattern 送進待測電路做測試，並將輸出結果 Scan Out 出來與正確的輸出結果做比對，判斷是否正確無誤。

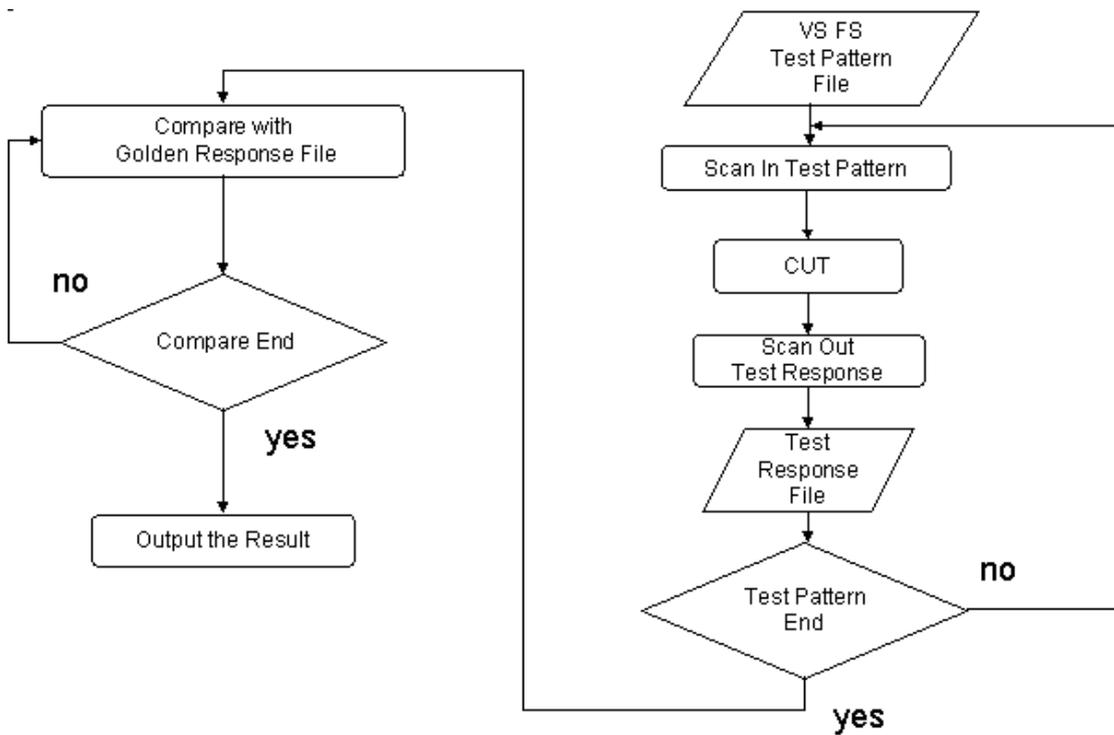


圖 4-15. Serial Testing Mode 測試架構圖

我們使用 Verilog 中 Display 指令打出輸出腳位的訊號值，並使用 Visual C++ 來比對其結果是否正確。圖 4-16 為 VS 的 Serial Testing mode 比對結果，圖 4-17 為 FS 的 Serial Testing mode 比對結果。

```
C:\Documents and Settings\hung\桌面\1)程式整理\GM\比對\Debug\compare.exe
GM第108筆pattern，共找到0個錯誤!!
GM第109筆pattern，共找到0個錯誤!!
GM第110筆pattern，共找到0個錯誤!!
GM第111筆pattern，共找到0個錯誤!!
GM第112筆pattern，共找到0個錯誤!!
GM第113筆pattern，共找到0個錯誤!!
GM第114筆pattern，共找到0個錯誤!!
GM第115筆pattern，共找到0個錯誤!!
GM第116筆pattern，共找到0個錯誤!!
GM第117筆pattern，共找到0個錯誤!!
GM第118筆pattern，共找到0個錯誤!!

比對完成，共找到0個錯誤
請按任意鍵繼續 . . .
Press any key to continue
```

圖 4-16. VS 的 Serial Testing mode 結果比對

```
C:\Documents and Settings\hung\桌面\1)程式整理\RM\比對\Debug\compare.exe
RM第81筆pattern，共找到0個錯誤!!
RM第82筆pattern，共找到0個錯誤!!
RM第83筆pattern，共找到0個錯誤!!
RM第84筆pattern，共找到0個錯誤!!
RM第85筆pattern，共找到0個錯誤!!
RM第86筆pattern，共找到0個錯誤!!
RM第87筆pattern，共找到0個錯誤!!
RM第88筆pattern，共找到0個錯誤!!
RM第89筆pattern，共找到0個錯誤!!
RM第90筆pattern，共找到0個錯誤!!
RM第91筆pattern，共找到0個錯誤!!

比對完成，共找到0個錯誤
請按任意鍵繼續 . . .
Press any key to continue
```

圖 4-17. FS 的 Serial Testing mode 結果比對

針對 Test Coverage 較低的原因作分析:

由於此 3D 繪圖晶片中我們對 Vertex Shader 來做 test coverage 評估。Vertex Shader 包含

基本的座標轉換(Transformation)、光源照射(Lighting)以及刪除多餘運算的 Backface Culling 及 Clipping 等等，而 Fragment Shader 是為了達到更真實的視覺效果，材質貼圖在三維圖學應用上是相當重要且常被使用的技術。在許多進階的著色效果，包含陰影、環境與凹凸貼圖全都仰賴多樣化的材質貼圖功能應用。由此我們可以知道在這龐大的計畫中需要較多的 Pin 腳個數，當 Pin 腳個數較多時，則會影響 Test Coverage，所以此計劃的電路較大而造成 Test Coverage 較低。

4.2.IEEE-1500 with BIST 之測試機制驗證

以下我們將介紹 VS 和 FS 在 RUNBIST 驗證的方法與過程，和 Hardware Overhead 分析，並使用 Debussy&看架構圖和使用 nWave&看其模擬訊號結果。

4.2.1. VS 和 FS & IEEE-1500 with BIST 架構圖

我們使用 Debussy& 看 IEEE-1500 在 RUNBIST 機制下包上 VS 和 FS 的階層圖，圖 4-18 為 VS Top 模組和 VS 的 RUNBIST 之 Wrapper Cell 以及 VS 的 Modify TAP Controller，而圖 4-19 為 FS Top 模組和 FS 的 RUNBIST 以及 FS 的 Modify TAP Controller。

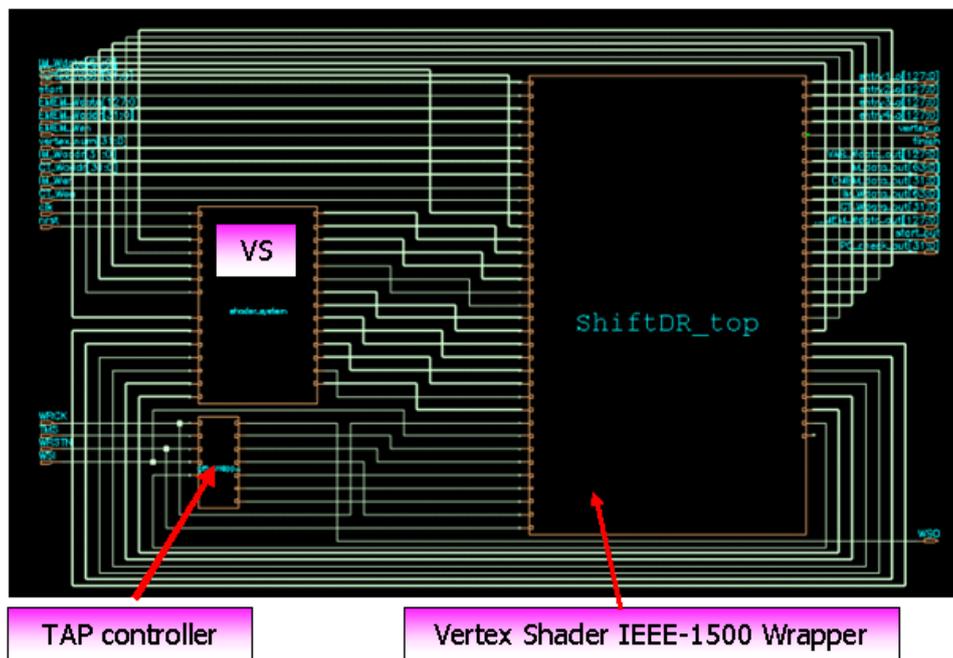


圖 4-18. VS+ IEEE-1500

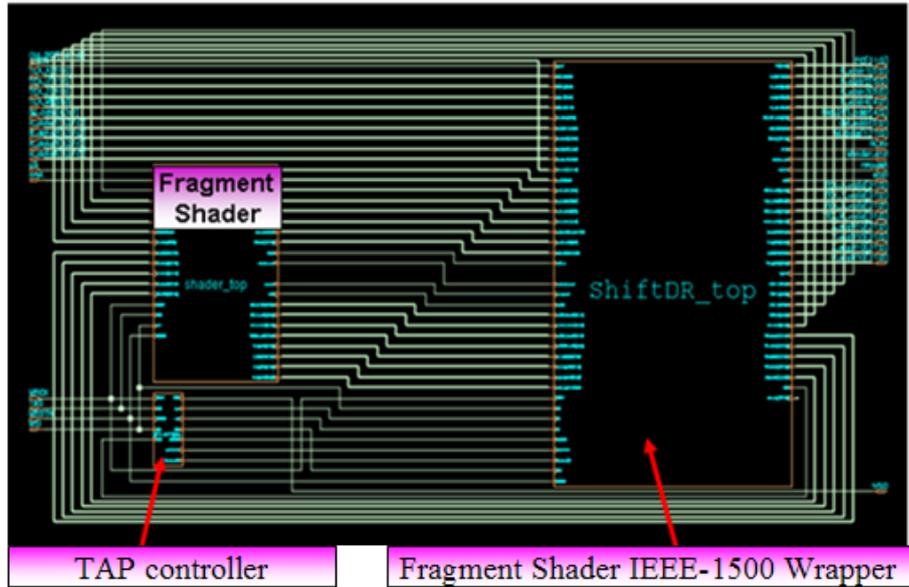


圖 4-19. FS + IEEE-1500

4.2.2. Hardware Overhead

在 VS 的部份其 Gate count 為 3481K，在圖 4-20，為 VS 的 TAP Controller 其 Gate count 為 1.362K，而圖 4-21，為 VS 的 IEEE-1500 其 Gate count 為 161.986K，所以如圖 4-22 得知所以 IEEE-1500 佔 VS 的硬體負擔為 4.69%。

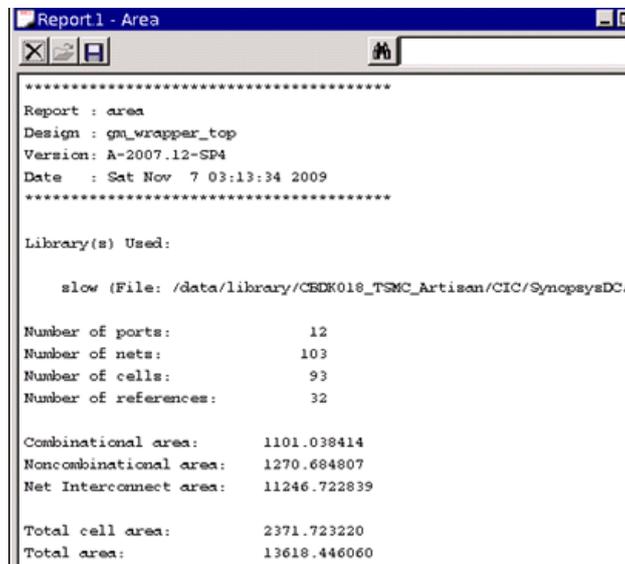


圖 4-20. VS 的 TAP Controller 的面積

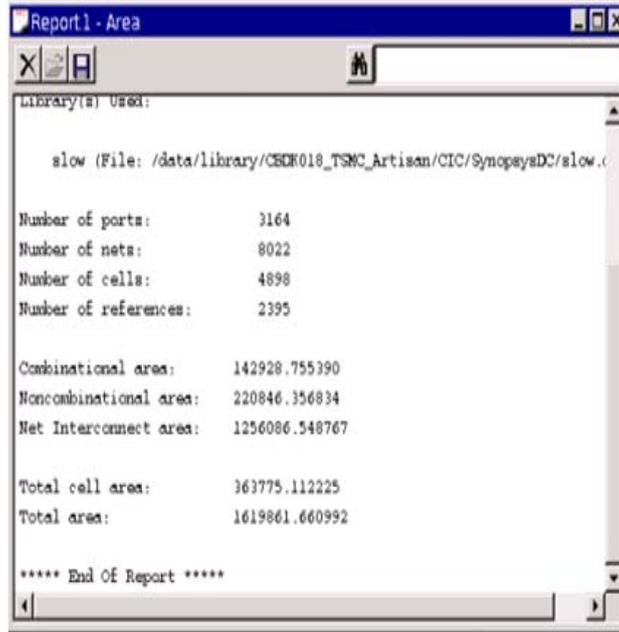


圖 4-21. VS 的 IEEE-1500 Wrapper 的面積

$$\begin{aligned}
 \text{Hardware overhead} &= [\text{DFT area} / \text{Vertex Shader area}] * 100\% \\
 &= [(1.362 \text{ K} + 161.986 \text{ K}) / 3481 \text{ K}] * 100\% \\
 &= 4.69\%
 \end{aligned}$$

圖 4-22. IEEE-1500 佔 VS 的硬體負擔

在 FS 的部份其 Gate count 為 1415.5K，在圖 4-23，為 FS 的 TAP Controller 其 Gate count 為 1.362K，而圖 4-24，為 FS 的 IEEE-1500 其 Gate count 為 152.934K，所以如圖 4-25 得知所以 IEEE-1500 佔 FS 的硬體負擔為 10.9%。

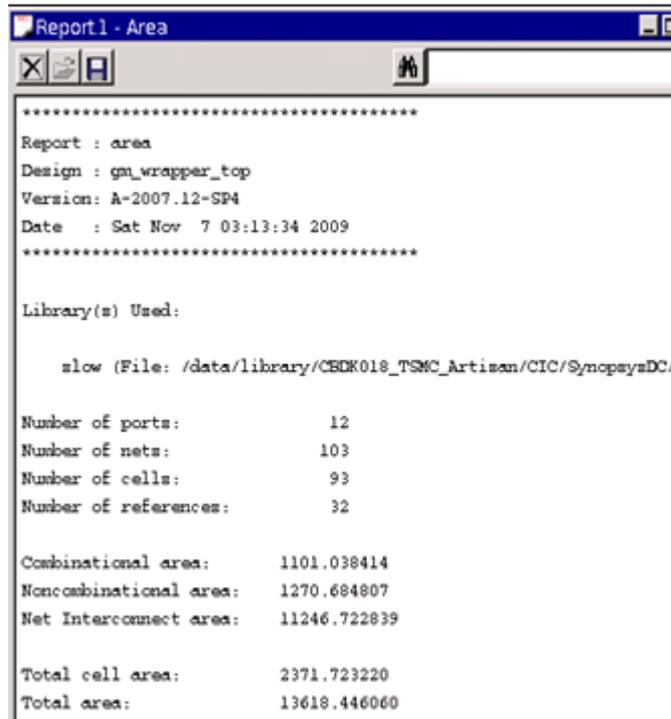


圖 4-23. FS 的 TAP Controller 的面積

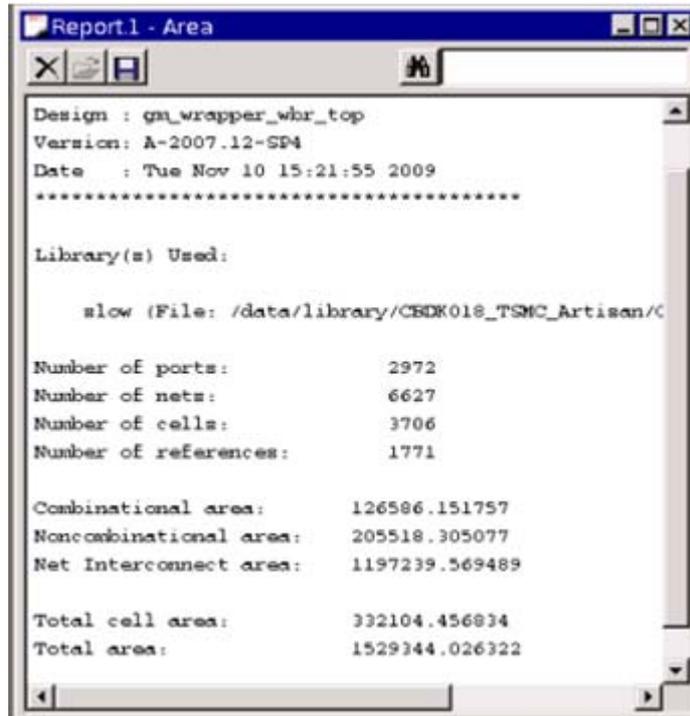


圖 4-24. FS 的 IEEE-1500 Wrapper 的面積

$$\begin{aligned} \text{Hardware overhead} &= [\text{DFT area} / \text{Fragment Shader area}] * 100\% \\ &= [(1.362 \text{ K} + 152.9436 \text{ K}) / 1415.5 \text{ K}] * 100\% \\ &= 10.9\% \end{aligned}$$

圖 4-25. IEEE-1500 with BIST 佔 FS 的硬體負擔

4.2.3. VS 和 FS 以 IEEE 1500 with BIST 結果比對

我們採用 Random Pattern Test 的方式，所使用的 Fault Model 為 Stuck-at-Fault，LFSR Seed 值為 01 交替，以下為使用 Benchmark ISCAS85 電路[16]和 ISCAS89 電路[17]以及 VS 和 FS 來分析其 Test coverage 為何，並將結果做比對。

1. Test Coverage 分析:

[Benchmark]

圖 4-26 和圖 4-27 顯示出使用 LFSR 對 Benchmark c6288 作測試所產生的 Pattern 個數為 119 筆和所達到的 Test Coverage 為 99.96%。

fault class	code	#faults
Detected	DT	10632
detected_by_simulation	DS	(10632)
Possibly detected	PT	0
Undetectable	UD	4
undetectable-unused	UU	(4)
ATPG untestable	AU	0
Not detected	ND	4
not-observed	NO	(4)
total faults		10640
test coverage		99.96%
fault coverage		99.92%

Pattern Summary Report		

#internal patterns		0
#external patterns (filename119.evcd)		119
#full_sequential patterns		119

圖 4-26. Benchmark c6288 使用 LFSR 測試結果

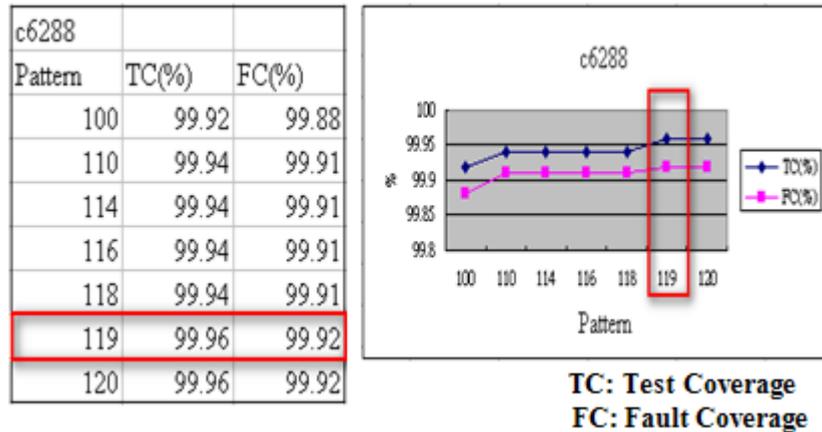


圖 4-27. Benchmark c6288 的 Pattern 個數以及 TC 和 FC

實驗結果整理如表 4-4 為 LFSR 使用 Benchmark 電路作測試的結果，我們將 LFSR 的 Test Coverage 和 Fault Coverage 測試到和 ATPG 所產生的 Test Pattern 個數相同或更高的情況。

表格 4-4 使用 Benchmark 在 BUNBIST 的結果

Circuit	#Input	#Test pattern	#CLK(RUNBIST)	Test Coverage(%)	Fault Coverage(%)
s298	17	468	468	100	99.32
s344	24	536	536	100	99.31
s349	24	349	349	100	100
s510	25	1352	1352	100	99.61
c432	36	533	533	99.2	98.75
c499	41	701	701	99.4	99.74
c1355	41	2478	2478	99.76	100
c5315	178	3550	3550	100	99.82
c6288	32	119	119	100	99.96

[VS]

圖 4-28 和表 4-5 顯示出使用 LFSR 對 VS 作測試所產生的 Pattern 個數為 500 筆和所達到的 Test Coverage 為 10.32%。

Uncollapsed Stuck Fault Summary Report		
fault class	code	#Faults
Detected	DT	46213
detected_by_simulation	DS	(46213)
Possibly detected	PT	47953
not_analyzed-pos_detected	NP	(47953)
Undetectable	UD	33096
undetectable-unused	UU	(8)
undetectable-tied	UT	(11348)
undetectable-blocked	UB	(21740)
ATPG untestable	AU	0
Not detected	ND	585650
not-controlled	NC	(280396)
not-observed	NO	(305254)
total faults		712912
test coverage		10.32%
fault coverage		9.85%
Pattern Summary Report		
#internal patterns		0
#external patterns (filename_GM_IN130_gate_scan_chain_500times.evcd)		500
#full_sequential patterns		500

圖 4-28. VS 在 Serial Testing mode 測試結果

表格 4-5 VS 在 Serial Testing mode 測試結果

Circuit	#Input	#Output	#Test pattern	#Total clk	Test Coverage(%)
VS	364	689	500	344500	10.32

[FS]

圖 4-29 和表 4-6 顯示出使用 LFSR 對 FS 作測試所產生的 Pattern 個數為 500 筆和所達到的 Test Coverage 為 0.52%。

Uncollapsed Stuck Fault Summary Report		
fault class	code	#faults
Detected	DT	2725
detected_by_simulation	DS	(2725)
Possibly detected	PT	1280
not_analyzed-pos_detected	NP	(1280)
Undetectable	UD	32445
undetectable-unused	UU	(46)
undetectable-tied	UT	(10892)
undetectable-blocked	UB	(21507)
ATPG untestable	AU	0
Not detected	ND	648388
not-controlled	NC	(255481)
not-observed	NO	(392907)
total faults		684838
test coverage		0.52%
fault coverage		0.49%
Pattern Summary Report		
#internal patterns		0
#external patterns (filename_RM_IN130.evcd)		500
#full_sequential patterns		500

圖 4-29. FS 在 RUNBIST 測試結果

表格 4-6 FS 在 RUNBIST 測試結果

Circuit	#Input	#Output	#Test pattern	#Total clk	Test Coverage(%)
FS	471	332	500	235500	0.52

圖 4-30，為我們使用的 LFSR Characteristic Polynomial(特徵多項式)，圖中 1:0 代表特徵多項式為 $x+1$ ，我們將 LFSR Seed 值設定為 0101...0101，圖 4-31 為 Test Coverage 和 Fault Coverage 計算公式。

1: 0	2: 1 0	3: 1 0	4: 1 0
5: 2 0	6: 1 0	7: 1 0	8: 6 5 1 0
9: 4 0	10: 2 0	11: 2 0	12: 7 4 3 0
13: 4 3 1 0	14: 12 11 1 0	15: 1 0	16: 5 3 2 0
17: 3 0	18: 7 0	19: 6 5 1 0	20: 3 0
21: 2 0	22: 1 0	23: 5 0	24: 4 3 1 0
25: 3 0	26: 8 7 1 0	27: 8 7 1 0	28: 3 0
29: 2 0	30: 16 15 1 0	31: 3 0	32: 28 27 1 0
33: 13 0	34: 15 14 1 0	35: 2 0	36: 11 0

圖 4-30. Characteristic Polynomial(特徵多項式)

$\text{Test coverage} = \text{detected faults} / \text{detectable faults}$ $\text{Fault coverage} = \text{detected faults} / \text{all faults}$

圖 4-31. Test Coverage 和 Fault Coverage 計算公式

2. 結果比對:

如圖 4-32 為 RUNBIST 驗證架構圖，首先設定 LFSR 的 Seed 值，然後計算此筆 LFSR 所產生的 Pattern 對待測電路而言所產生的 Test Coverage 為何，並比對結果是否正確。

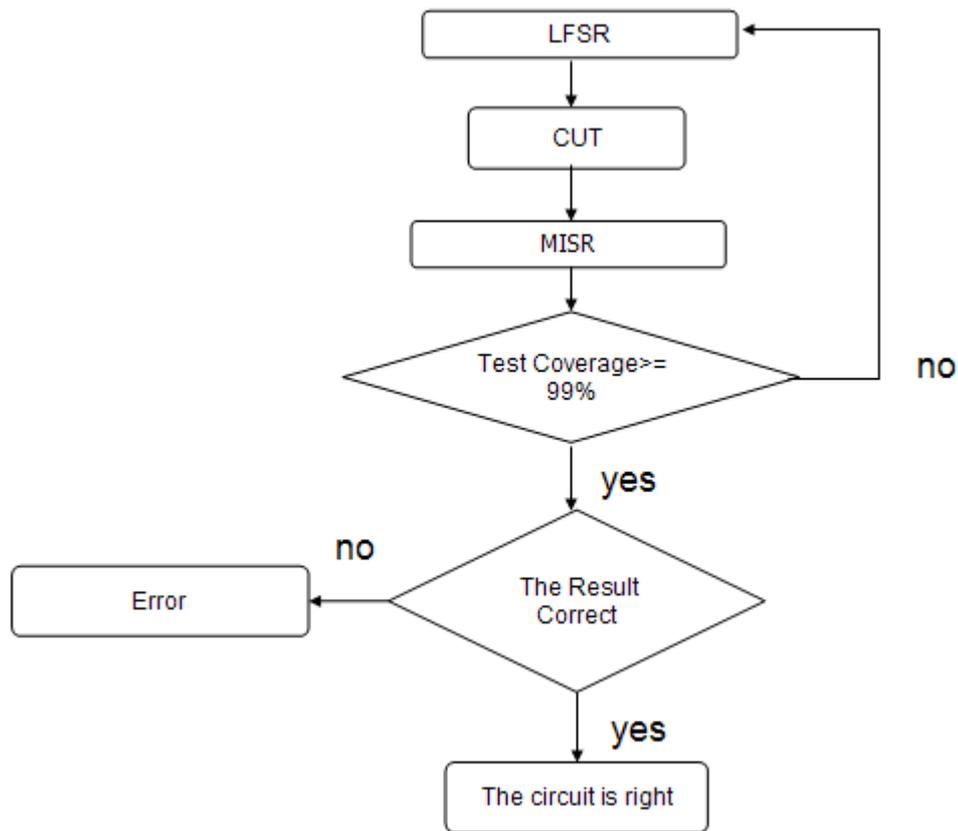


圖 4-32. RUNBIST 測試架構圖

[LFSR 結果比對]:

圖 4-33 32 Bits 的 LFSR 架構，初始 Seed Value 為 5555_5555(16)，經過 23 個時脈週期後其 LFSR 內部的值為 5541_ee6a(16)，如圖 4-34 所示。

Characteristic Polynomial : $x^{32} + x^{28} + x^{27} + x^1 + 1$



圖 4-33. 32 Bits 的 LFSR 架構

i_TEX_Z[31:0]	5541_ee6a	XXXX_XXXX	5541_ee6a
i_TEX_X[31:0]	5541_ee6a	XXXX_XXXX	5541_ee6a
i_TEX_W[31:0]	5541_ee6a	XXXX_XXXX	5541_ee6a
i_R_data4[127:0]	5541_ee6a_5541_ee6a_5541_ee6a_5541_ee6a	XXXX_XXXX_X*	5541_ee6a_5541_ee6a_5541_ee6a_5541_ee6a
i_R_data3[127:0]	5541_ee6a_5541_ee6a_5541_ee6a_5541_ee6a	XXXX_XXXX_X*	5541_ee6a_5541_ee6a_5541_ee6a_5541_ee6a
i_R_data2[127:0]	5541_ee6a_5540_7b9a_5541_ee6a_5541_ee6a	XXXX_XXXX_X*	5541_ee6a_5540_7b9a_5541_ee6a_5541_ee6a
i_R_data1[127:0]	5541_ee6a_5541_ee6a_5541_ee6a_5541_ee6a	XXXX_XXXX_X*	5541_ee6a_5541_ee6a_5541_ee6a_5541_ee6a
i_0ld_DST[127:0]	5541_ee6a_5541_ee6a_5541_ee6a_5541_ee6a	XXXX_XXXX_X*	5541_ee6a_5541_ee6a_5541_ee6a_5541_ee6a
i_IM_data[63:0]	5541_ee6a_5541_ee6a	XXXX_XXXX_X*	5541_ee6a_5541_ee6a

圖 4-34. 23 個時脈週期後 LFSR 內部的值

我們使用 C++ 去比對其 MISR 結果，圖 4-37 和圖 4-39 為第 40 個時脈週期後 VS 和 FS 的 MISR 結果比對。

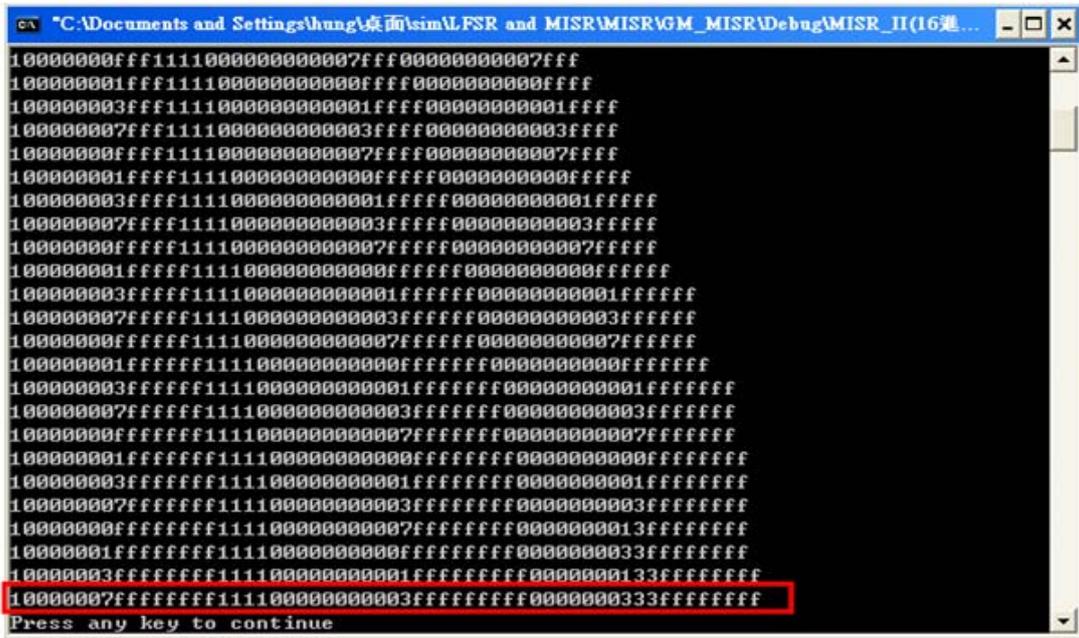


圖 4-37. 40 個時脈週期後 MISR 結果(VS)



圖 4-38. 40 個時脈週期後 MISR 內部的值(FS)

```
*C:\Documents and Settings\hung\桌面\sim\FSR and MISR\MISR\ARM_MISR\Debug\MISR_II(16進...
1 0 1 0 6 0 0 0 0 0 851 0 851 5d04 0 851
0 1 0 0 c 0 0 0 0 0 10a3 0 10a3 ba09 0 10a3
1 0 0 0 18 0 0 0 0 0 2146 0 2146 17413 0 2146
0 0 0 1 10 0 0 0 0 0 428d 0 428d 2e827 0 428d
0 0 1 1 0 0 0 0 0 0 851a 0 851a 5d04f 0 851a
0 1 1 0 1 0 0 0 0 0 10a34 0 10a34 ba09e 0 10a34
1 1 0 1 3 0 0 0 0 0 21468 0 21468 17413d 0 21468
1 0 1 0 6 0 0 0 0 0 428d1 0 428d1 2e827b 0 428d1
0 1 0 0 c 0 0 0 0 0 851a3 0 851a3 5d04f6 0 851a3
1 0 0 0 18 0 0 0 0 0 10a346 0 10a346 ba09ec 0 10a346
0 0 0 1 10 0 0 0 0 0 21468d 0 21468d 17413d8 0 21468d
0 0 1 1 0 0 0 0 0 0 428d1a 0 428d1a 2e827b0 0 428d1a
0 1 1 0 1 0 0 0 0 0 851a34 0 851a34 5d04f61 0 851a34
1 1 0 1 3 0 0 0 0 0 10a3468 0 10a3468 ba09ec2 0 10a3468
1 0 1 0 6 0 0 0 0 0 21468d1 0 21468d1 17413d84 0 21468d1
0 1 0 0 c 0 0 0 0 0 428d1a3 0 428d1a3 2e827b09 0 428d1a3
1 0 0 0 18 0 0 0 0 0 851a346 0 851a346 5d04f613 0 851a346
0 0 0 1 10 0 0 0 0 0 10a3468d 0 10a3468d ba09ec27 0 10a3468d
0 0 1 1 0 0 0 0 0 0 21468d1a 0 21468d1b 7413d84f 0 21468d1a
0 1 1 0 1 0 0 0 0 0 428d1a34 0 428d1a36 e827b09e 0 428d1a34
1 1 0 1 3 0 0 0 0 0 851a3468 0 851a346d d04f613d 0 851a3468
1 0 1 0 6 0 0 0 0 1 a3468d1 1 a3468da a09ec27b 1 a3468d1
0 1 0 0 c 0 0 0 0 2 1468d1a3 2 1468d1b4 413d84f6 2 1468d1a3
1 0 0 0 18 0 0 0 0 4 28d1a346 4 28d1a368 827b09ec 4 28d1a346
Press any key to continue
```

圖 4-39. 40 個時脈週期後 MISR 結果比對(FS)

5. 結果與未來工作

在這篇報告裡我們將 IEEE-1500 和 BIST 機制作結合，所以在測試機制驗證之下使用兩種測試模式來做驗證，除了 IEEE-1500 標準的串列掃描測試機制之外，我們還加入了 BIST 的機制來增加其測試的考量，藉此改善 IEEE-1500 在串列機制中緩慢的測試速度。由於 IEEE-1500 在串列掃描時，每一時脈僅送一筆資料到邊界掃描暫存器內部，當有龐大的掃描資料要送到待測電路內部做測試時，就需要更多的時間，若使用 BIST 來對待測電路進行測試，就能大大的降低測試時間的考量，同時也更嚴謹的對待測電路來做驗證。

首先我們先使用 BIST 機制再次對 3D 繪圖系統晶片進行測試，同時也將 LFSR 送到 Benchmark 電路的測試向量來評估 Test Coverage 和 Fault Coverage。在 MISR 部分我們也使用 C 語言將電路輸出的結果來當 MISR 的輸入向量做比對判定結果是否正確無誤。再使用 IEEE-1500 串列機制來對 3D 繪圖系統晶片來進行測試，並將結果掃描出來做比對判定結果是否正確無誤，由於此 Vertex Shader 和 Fragment Shader 電路較為龐大和 Pin 腳個數較多造成 Test Coverage 較低。

總之在此，我們將 IEEE-1500 和 BIST 機制作結合，並更進一步的對待測電路來做驗證，以減少在測試時間上的花費。在之後的測試機制發展，將會再以改善 LFSR 的測試覆蓋率[18]，和擁有更小的測試電路面積為重點或者加入其他壓縮機制[19][20]以達到更好的測試結果為目標。

參考文獻

- [1] *IEEE Standard Test Interface Language (STIL) for Digital Test Vector Data—Core Test Language (CTL)*, Sponsor Test Technology Standards Committee of the IEEE Computer Society, 2005.
- [2] DaSilva, F. Zorian, Y. Whetsel, L. Arabi, K. Kapur and R. Synopsys, “Overview of the IEEE P1500 Standard,” in *Proc IEEE International Test Conf.*, Charlotte, NC, Sept. 2003, pp. 988-997.
- [3] CTAG Working Group, “Overview of Proposed IEEE P1500 Scaleable Architecture for Testing Embedded Cores,” in *Proc. DATE*, Mar. 2001.
- [4] E.J. Marinissen etc., “On IEEE P1500’s standard for embedded core test,” *IEEE Journal of Electronic Testing*, vol. 18, no. 4-5, pp. 365-383, Aug. 2002.
- [5] H. Yi, J. Song and S. Park, “Low-Cost Scan Test for IEEE-1500-Based SoC”, *IEEE Transactions on Instrumentation and measurement*, vol. 57, no. 5, pp.1071-1078, May 2008
- [6] E.J. McCluskey, “Built-in Self-Test Techniques,” in *Proc. IEEE Design Test Computers*, vol. 2, no. 2, pp.21-28, Apr. 1985.
- [7] H. Yi, J. Song and S. Park, “Interconnect Delay Fault Test on Boards and SoCs with Multiple Clock” *ETRI Journal*, vol. 30, no. 3, pp.403-411, Jun. 2008.
- [8] S. R. Das, A. Hossain, S. Biswas and E.M. Petriu, “Aliasing-free Compaction Revisited,” *IET Circuits Devices Systems*, vol. 2, no. 1, pp. 166 – 178, Feb.2008.
- [9] S. R. Das, S. Mukherjee, E. M. Petriu, M. H. Assaf and A. Hossain, “Space Compaction for Embedded Cores-Based System-on-Chips (SOCs) Using Fault Graded Output Merger”, in *Proc IMTC 2007 - Instrumentation and Measurement Technology Conf.*, vol. 3 of 5, Warsaw, Poland, 1 May -3 May 2007, pp.1-5.
- [10] S. Biswas, S. R. Das and E. M. Petriu, “Space Compactor Design in VLSI Circuits Based on Graph Theoretic Concepts”, in *Proc IMTC 2005 – Instrumentation and Measurement Technology Conference Ottawa*, Canada, vol. 1, no.16-19, May 2005, pp.178-183.
- [11] C. H. Tsai, F. D. Guo, J. H. Hong and C. W. Wu, “IEEE-1149.1 Boundary Scan Circuit Capable of Built-In Self-Testing”, United States Patent Number: 5,570,375 , Oct.29,1996.
- [12] Jutman, A. Tsertov and A. Ubar, “Calculation of LFSR Seed and Polynomial Pair for

- BIST Applications”, *IEEE Design and Diagnostics of Electronic Circuits and Systems 11th IEEE Workshop, DC, USA*, April 2008, pp.1-4.
- [13] X. Z. Chen, C. I. H. Chakravarthy, “Structure Design and Optimization of 2-D LFSR-Based Multisequence Test Generator in Built-In Self-Test”, *IEEE Trans. Instrumentation and Measurement*, vol. 57, no. 3, pp.651-663, Mar. 2008.
- [14] X. Y. Chen, C. I. H. Chen, “Automated Synthesis of a Multiple-Sequence Test Generator using 2-D LFSR”, in *Proc. IEEE International ASIC Conf.*, pp.75-79, 13 Sep.-16 Sep. 1998
- [15] C. I. H. Chen, K. George, “Configurable Two-Dimensional Linear Feedback Shifter Registers for Parallel and Serial Built-In Self-Test”, *IEEE Trans. on instrumentation and measurement*, vol. 53, no.4, pp.1005-1014, Aug. 2004.
- [16] F. Brglez, D. Bryan and K. Kozminski, “Combinational Profiles of Sequential Benchmark Circuits”, in *Proc. IEEE International Symposium on Circuits and System*, vol.3, 8-11 May 1989, pp.1929-1934.
- [17] Kajihara, S. Kinoshita, K. Pomeranz, I. Reddy, S.M., “Combinationally Irredundant ISCAS-89 Benchmark Circuits”, *IEEE International Symposium Circuit and System*, vol. 4, pp.632 – 634, 12 May -15 May 1996.
- [18] J. Gu, C. Zhaohui and Y. Xiaoyang, “Research and implementation of the hardware/software co-design based on structure test model of SoC”, in *Proc International Symposium on Computational Intelligence and Design, China*, vol. 1, 14 Dec.-12 Dec. 2009, pp.526-530
- [19] C. Su, K. Hwang, “A Serial Scan Test Vector Compression Methodology.” in *Proc. IEEE Test Conf.*, Baltimore, MD, USA, Oct.1993, pp. 981-988.
- [20] 邱升南, 2000.08, ”Generating Efficient Tests for Continuous Scan”, 國立中興大學資訊科學研究所

出席國際學術會議心得報告

計畫編號	NSC 98-2200-E-390 -001
計畫名稱	三維電腦繪圖晶片系統之可測試性設計與系統階層測試
出國人員姓名	洪進華
服務機關及職稱	國立高雄大學電機系助理教授
會議時間地點	2010年10月15-17日德國 (Darmstadt, Germany)
會議名稱	2010 Sixth International Conference on Intelligent Information Hiding and Multimedia Signal Processing

一、參加會議經過

10/14 晚上搭機前往德國，10/15 早上抵達飯店休息，瞭解會議議程與場地，研究會場路線圖。

10/15 早上 8:00 至會場報到，領取精簡論文集，議程表、光碟片、與精緻小禮物及背包，參加開幕，開始聆聽第一場 Keynote speech I and II，並且參加 Technical session 並與與會學者專家交換意見。:

Keynote speech I:

演講題目：“Recent Advances in Face Biometrics and Security Surveillance”

主講人： Stan Z. Li (Chinese Academy of Sciences, China)。

Keynote speech II:

演講題目：“Information Theory and Biometrics”

主講人： Dr. ir. Frans M.J. Willem (Technische Universiteit Eindhoven, Netherlands)。

10/16 聆聽 Keynote speech III and IV，參加 Technical session、發表論文並與與會學者專家交換意見。

Keynote speech III:

演講題目：“Steganography vs. Steganalysis: Progress and Challenge”

主講人： Jiwu Huang (Sun Yat-Sen University, China)。

Keynote speech IV:

演講題目：“Privacy and Transparency: Google and Financial Markets challenge Internet”

主講人： Günter Müller (University of Freiburg, Germany)。

10/17 聆聽 Keynote speech V，參加 Technical session 並與與會學者專家交換意見。

Keynote speech V:

演講題目：“Emotional Intelligence with Machine Intelligence”

主講人： Pau-Choo (Julia) Chung (National Cheng Kung University (NCKU), Taiwan)。

10/18~10/22 學術交流觀摩、10/22 搭機返台。

IEEE 智慧資訊隱藏與多媒體訊號處理 (Intelligent Information Hiding and Multimedia Signal Processing) 是有關計算機、加解系統、超大型積體電路與多媒體訊號處理設計理論與應用方面相當重要與吸引人之國際性研討會。每次都會吸百餘位在此專門領域之學者專家與工業界人士與會，其中當然不乏各個特別領域高水準論文的作者。IIH-MSP 向來都是由國際電機電子工程學會 (IEEE) 轄下的訊號處理學會 (Signal Processing Society) 與智慧型計算學會 (Computational Intelligence Society) 主辦。本會議每年舉行一次，本次會議地點在德國，會議時間從九十九年十月十五日起到九十九年十月十七日止總共三天。整個會議期間總共有約有180篇論文發表。值得一提的是這180篇論文是從將近300篇投稿論文中精挑細選的。IIH-MSP目前在訊號處理、加解密系統、多媒體與VLSI設計領域最活躍的研討會之一。台灣學術界今年參加 IIH-MSP 的教授有將近20位左右，發表的論文數目近百篇篇。本人與會後皆與幾位與會的學者專家交換了一些意見與心得，相信對彼此都有助益。會場的空間與動流流暢，大家都可以仔細聆聽所有的論文發表並參與討論。除了論文發表外，尚有五場專家主題演講，內容非常豐富。

本人於抵達德國達姆斯特丹大學研討會會場，報到後隨即參加會議。會議每天從上午九點開始到下午五~六點，晚上則為一些慶典與宴會等活動。大會有五場正式演講與論文發表分別安排在第一天、第二天、第三天。其間每個 Section 都有多位學者專家對演講者的論文興趣濃厚，提出各種見解與看法，大家無拘束的討論，彼此互相認識。會議全程發表的論文議題包括下列各項：

- (1).Watermarking: techniques, attacks, protocols, applications
- (2).Steganography and steganalysis: techniques, protocols, applications
- (3).Cryptography and cryptanalysis: techniques, protocols, applications
- (4).Data authentication issues and access control themes
- (5).Broadcast and public-key encryption
- (6).Forensic analysis and tracing traitors
- (7).Digital rights management and legal aspect
- (8).RFID security and home network privacy
- (9).Platform integrity and trusted computing
- (10).Systems engineering and development for information hiding & security
- (11).VLSI/ASIC/FPGA/SOC design and implementation for information hiding & security
- (12).Enabling technologies and emerging standards for information hiding & security
- (13).Multimedia sensing and sensory systems
- (14).Multimedia source coding and channel coding
- (15).Multimedia signal analysis and visualization
- (16).Multimedia signal mining and data fusion
- (17).Multimedia networking and communication techniques (wired & wireless)
- (18).Multimedia/multimodal signals interpretation and automatic recognition
- (19).Multimedia databases and retrievals

- (20).Multimedia hyperlink techniques and applications
- (21).Advances in multimedia content description interface (e.g. MPEG-7)
- (22).Advances in multimedia framework: (e.g. MPEG-21)
- (23).Systems engineering and development for multimedia systems
- (24).Enabling technologies and emerging standards for multimedia systems

二、與會心得

與會人士涵蓋全球各地，有美國、加拿大、瑞典、德國、英國、法國、義大利、澳洲、日本、台灣、中國、印度等國家的專家學者與業界人士。其中台灣與會師生有將近 20 人。

以會議本身所涵蓋的內容而言，我們可從中看出訊號處理、加解密系統、多媒體與 VLSI 這個領域的進展，及學術界與工業界個別與共同所關心的問題與研究發展方向。工業界參與者包含通訊、計算機、多媒體、IC 設計公司以及系統設計開發者。然而本會議還是以學術界參加者為絕大多數。學術界的研究自然偏向理論性，前瞻性，尚待改進，及較新發現或發展出的問題上，其範圍相當廣泛。而目前大多數人所共同關心的電路與系統設計有關的研究包括下列數項：

- (1) Watermarking: techniques, attacks, protocols, applications
- (2) Cryptography and cryptanalysis: techniques, protocols, applications
- (3) RFID security and home network privacy
- (4) Multimedia networking and communication techniques (wired & wireless)
- (5) Multimedia hyperlink techniques and applications
- (6) Systems engineering and development for information hiding & security
- (7) VLSI/ASIC/FPGA/SOC design and implementation for information hiding & security

以會議過程而言，主辦單位還算用心，並未出大差錯。議程安排合理，且絕大多數論文均能依原時間發表。一般好的會議均嚴格要求論文宣讀者出席，而這次會議即做得相當好。『IEEE 智慧資訊隱藏與多媒體訊號處理』(IIH-MSP) 重學術而且相當沒有商業氣息，本人對此深表贊同。它不但水準高且領域廣泛，同行學者出席比率極高，故鼓勵同仁多參加。這次本人帶回 2010 年『IEEE 智慧資訊隱藏與多媒體訊號處理年度會議』(IIH-MSP) 會議論文集光碟片一份。

國科會補助計畫衍生研發成果推廣資料表

日期:2011/02/23

國科會補助計畫	計畫名稱: 子計畫八: 三維電腦繪圖晶片系統之可測試性設計與系統階層測試(3/3)
	計畫主持人: 洪進華
	計畫編號: 98-2220-E-390-001- 學門領域: 晶片科技計畫--整合型學術研究計畫

無研發成果推廣資料

98 年度專題研究計畫研究成果彙整表

計畫主持人：洪進華		計畫編號：98-2220-E-390-001-					
計畫名稱：具有即時效能/功率監控功能的高效率可程式化三維電腦繪圖晶片系統：軟硬體開發及整合							
--子計畫八：三維電腦繪圖晶片系統之可測試性設計與系統階層測試(3/3)							
成果項目		量化			單位	備註（質化說明：如數個計畫共同成果、成果列為該期刊之封面故事...等）	
		實際已達成數（被接受或已發表）	預期總達成數(含實際已達成數)	本計畫實際貢獻百分比			
國內	論文著作	期刊論文	0	0	100%	篇	
		研究報告/技術報告	0	0	100%		
		研討會論文	1	1	100%		
		專書	0	0	100%		
	專利	申請中件數	0	0	100%	件	
		已獲得件數	0	0	100%		
	技術移轉	件數	0	0	100%	件	
		權利金	0	0	100%	千元	
	參與計畫人力（本國籍）	碩士生	4	4	100%	人次	
		博士生	0	0	100%		
		博士後研究員	0	0	100%		
		專任助理	0	0	100%		
國外	論文著作	期刊論文	0	0	100%	篇	
		研究報告/技術報告	0	0	100%		
		研討會論文	2	2	100%		
		專書	0	0	100%		章/本
	專利	申請中件數	0	0	100%	件	
		已獲得件數	0	0	100%		
	技術移轉	件數	0	0	100%	件	
		權利金	0	0	100%	千元	
	參與計畫人力（外國籍）	碩士生	0	0	100%	人次	
		博士生	0	0	100%		
		博士後研究員	0	0	100%		
		專任助理	0	0	100%		

<p>其他成果 (無法以量化表達之成果如辦理學術活動、獲得獎項、重要國際合作、研究成果國際影響力及其他協助產業技術發展之具體效益事項等，請以文字敘述填列。)</p>	<p>無</p>
--	----------

科 教 處 計 畫 加 填 項 目	成果項目	量化	名稱或內容性質簡述
	測驗工具(含質性與量性)	0	
	課程/模組	0	
	電腦及網路系統或工具	0	
	教材	0	
	舉辦之活動/競賽	0	
	研討會/工作坊	0	
	電子報、網站	0	
	計畫成果推廣之參與(閱聽)人數	0	

國科會補助專題研究計畫成果報告自評表

請就研究內容與原計畫相符程度、達成預期目標情況、研究成果之學術或應用價值（簡要敘述成果所代表之意義、價值、影響或進一步發展之可能性）、是否適合在學術期刊發表或申請專利、主要發現或其他有關價值等，作一綜合評估。

1. 請就研究內容與原計畫相符程度、達成預期目標情況作一綜合評估

達成目標

未達成目標（請說明，以 100 字為限）

實驗失敗

因故實驗中斷

其他原因

說明：

2. 研究成果在學術期刊發表或申請專利等情形：

論文： 已發表 未發表之文稿 撰寫中 無

專利： 已獲得 申請中 無

技轉： 已技轉 洽談中 無

其他：（以 100 字為限）

3. 請依學術成就、技術創新、社會影響等方面，評估研究成果之學術或應用價值（簡要敘述成果所代表之意義、價值、影響或進一步發展之可能性）（以 500 字為限）

我們將 IEEE-1500 和 BIST 機制作結合，所以在測試機制驗證之下使用兩種測試模式來做驗證，除了 IEEE-1500 標準的串列掃描測試機制之外，我們還加入了 BIST 的機制來增加其測試的考量，藉此改善 IEEE-1500 在串列機制中緩慢的測試速度。由於 IEEE-1500 在串列掃描時，每一時脈僅送一筆資料到邊界掃描暫存器內部，當有龐大的掃描資料要送到待測電路內部做測試時，就需要更多的時間，若使用 BIST 來對待測電路進行測試，就能大大的降低測試時間。不過目前還有測試函蓋率有待提升，若能解決此一問題，則對 IC 測試領域是一大創新成就，未來不論是學術界或是業界都有很好的應用價值。