實用數位系統設計：

Final Project

Geofence

組別：窩不知道

組長：陳晉毅 B093011056

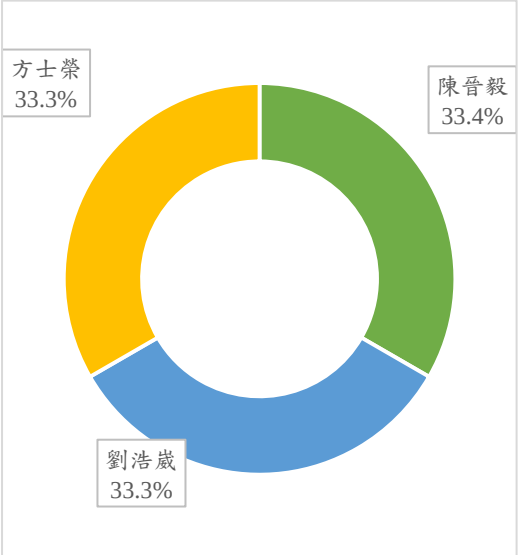
組員：劉浩崴 B093011055

組員：方士榮 B093011008

設計實作摘要

是否通過 RTL (合成前, pre-synthesis) 驗證?	通過
是否通過 gate-level(post-synthesis)驗證?	通過
Clock period (ns)	4
Area (um^2)	31907.72478
Power (mW)	2.5683
Simulation time (ns) (合成後)	28628
是否引用其他組的設計構想或架構?	No
Coding style check and enhancement?	yes
Code coverage evaluation and enhancement?	No
其他重要特色	部分外積不用進乘法器

工作分配與貢獻



陳晉毅 33.4%：

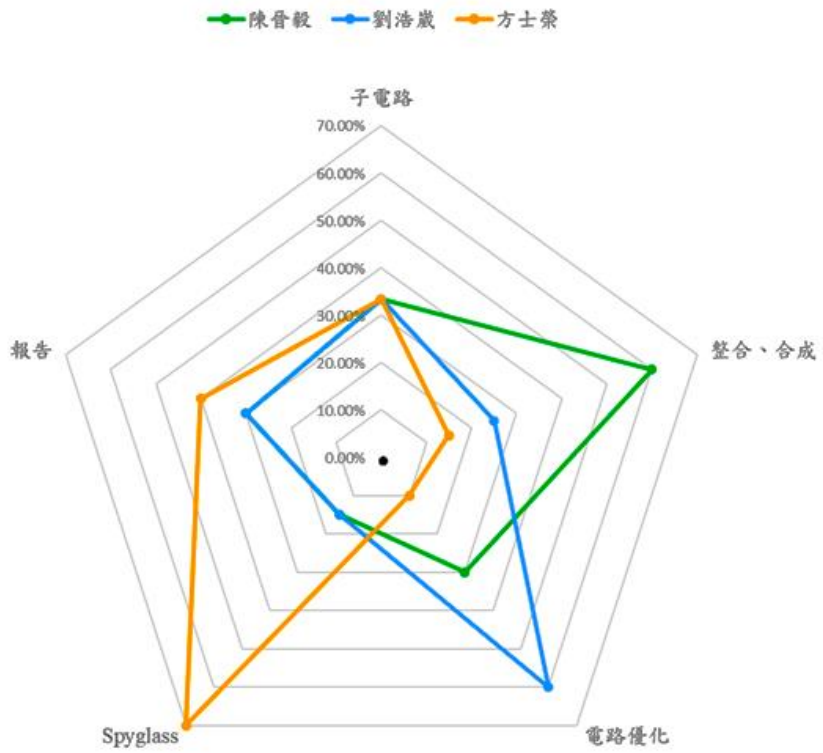
電路設計、撰寫、電路優化、電路整合、
驗證與架構圖繪製(Address Controllor、FSM&Output)、報告撰寫

劉浩崴 33.3%：

電路設計、撰寫、驗證與架構圖繪製(Vector & Cross)、
電路優化、報告撰寫

方士榮 33.3%：

電路設計、撰寫、驗證與架構圖繪製(RF、RF I/O)、Spyglass、
報告撰寫、報告優化

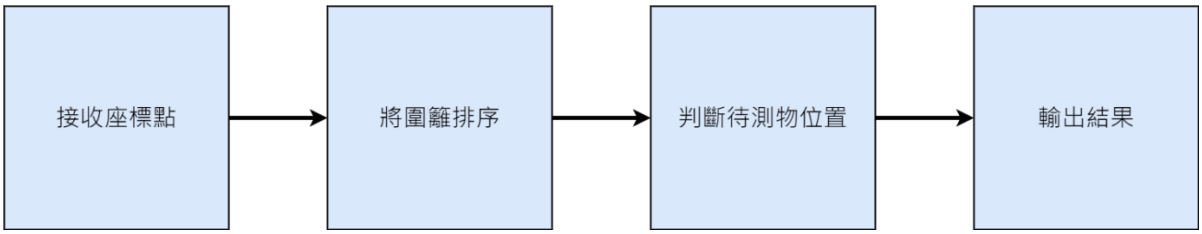


組長：陳晉毅 33.4% 陳晉毅

組員：劉浩崴 33.3% 劉浩崴

方士榮 33.3% 方士榮

一、演算法介紹與說明



在演算法的部分，整體尋找待測物的演算法是參考講義的內容，而其中在排序的部分，我們有用了兩種演算法下去比較與評估，分別是 Bubble sort 與 Quick sort，在評估後則選了 Bubble sort 作為我們的排序演算法。

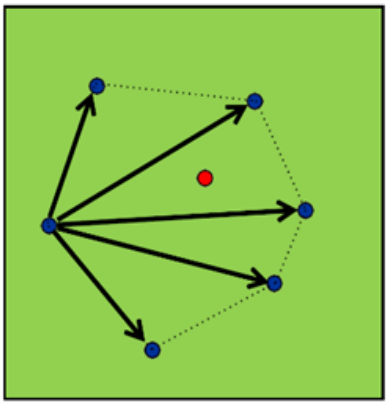
(一) 整體演算法

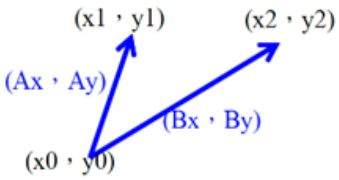
一、 運作方式

整體以外積為核心，可以分成兩個部分

第一部分：排柵欄

這一部分是要將柵欄點排序，讓柵欄點統一順時針排列或逆時針排列，此時利用外積來判斷柵欄點的相對位置。取其中一個柵欄點作為原點，分別和其他 5 個柵欄點形成向量，將兩向量兩兩做外積後，根據外積結果的正負號與右手定則，可以判斷兩向量(兩個點)的相對位置。

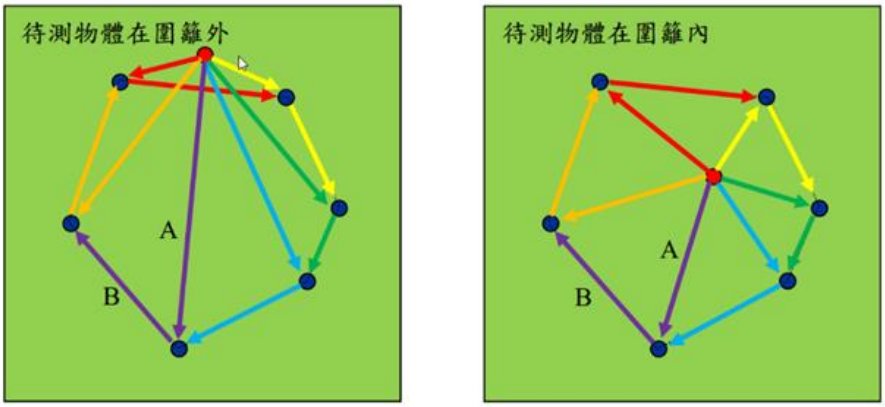




在了解兩個點的相對位置後，可以利用排序演算法依照其相對位置做排序，讓點的排列統一順逆方向。

第二部分：觀測待測物

這個部分是觀察待測物與柵欄點之間的相對位置，具體觀測方式如同第一步，是使用外積然後觀察正負號，待測物在柵欄內與外的相對關係如下圖:



左圖中，如果待測物在柵欄之外，那待測物和兩柵欄之間形成的向量外積會有正有負。而在右圖中，如果待測物在柵欄之內，那向量外積會同號，因此可以利用此關係來判斷待測物在柵欄內還是在外。

二. 特色、優缺點

這個演算法的特色是使用外積，整個流程都圍繞著向量與外積在運算。

優點：

優點	原因
精確度高	數學上，向量之間的關係明確
大架構不會太複雜	主要運算是向量外積

缺點：

缺點	原因
面積較大	會使用到乘法器
運算時間稍長	既排序也要比較

我們使用這種演算法的主因是在於優點內的精確度高。

(二) 排序演算法

一、 Bubble sort

這是一種比較型的排序法，比較誰大誰小，在這裡的話則是比較誰左誰右，在 Bubble sort 中，運作時會分成幾輪，每一輪會針對序列中的成員進行比較，以此抓出最大或最右邊的成員並放到序列的最右邊，下一輪則對剩下的成員重複以上的動作直到排序完。

二、 Quick sort

這也是一種比較型的排序法，執行的方式是一種遞迴，方法是在序列中隨機挑選一個成員和其他成員比較，比被挑選成員大或在它左邊的放在它序列中的左邊，反之放右邊，此時被挑選的成員的左側和右側會分別形成兩個子序列，再分別對兩個子序列執行遞迴，直到任何子序列剩下兩個元素並判斷完關係後結束遞迴。

三、 比較與評估

以下列出這兩種排序法常見的優缺點:

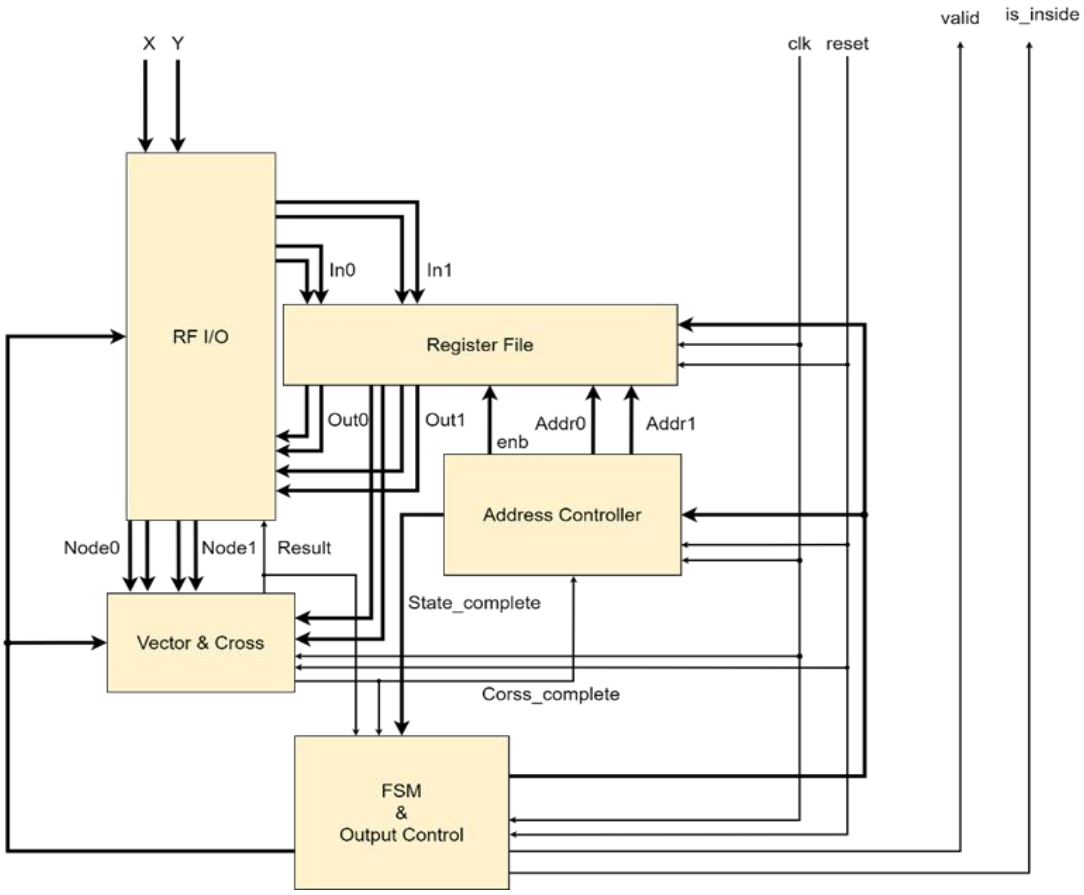
	優點	缺點
Quick Sort	排序快 O (nlog(n))	需要額外記憶體
Bubble Sort	簡單	排序慢 O (n*n)

我們以這次的情況與評分標準下去評估，首先是我們這次排序的數量不多，只有 5 個而已，因此雖然 Quick Sort 排得比較快，但這項優點並不凸出。再者 Quick Sort 在硬體上不好實現，且這種方法會使用到額外的記憶體，在 IC 電路上則是會多出 Flip-Flop，估算後會用上約 50 個 Flip-Flop，這會導致我們的面積上升許多，因此最後選擇了 Bubble Sort，以下統整出我們放棄 Quick Sort 選擇 Bubble Sort 的原因：

1. 使用 Quick Sort 會多出 50 多個 Flip-Flop
2. Quick Sort 在排序成員少的情況下，優點不凸出。
3. 相對起來 Bubble Sort 好實現、面積小且時間和比 Quick Sort 不會多太多。

二、架構介紹與說明

- 我們整體的大架構如下圖所示：

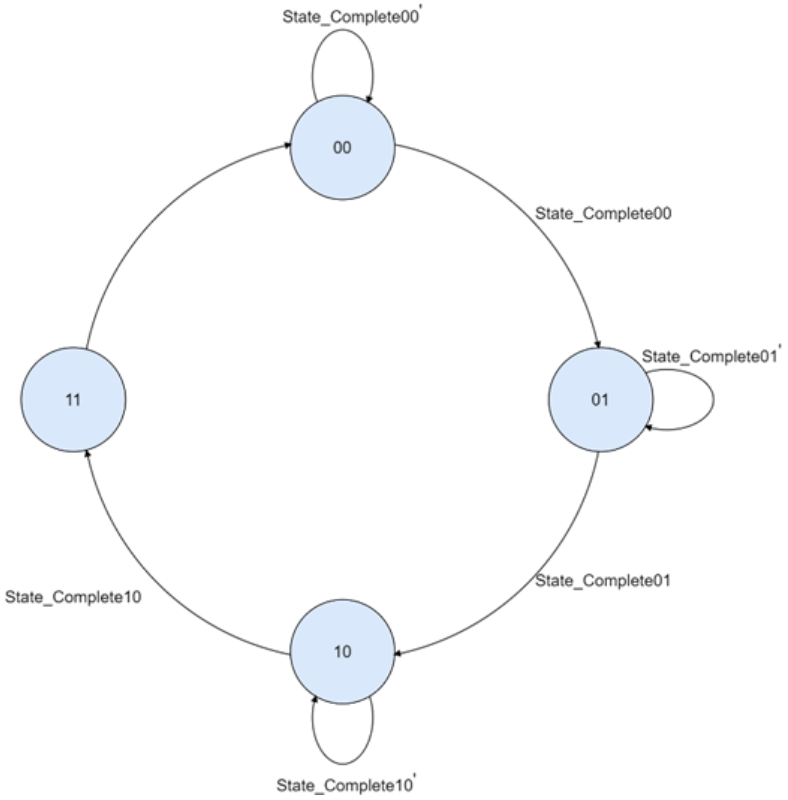


電路名稱	功能
FSM & Output Control	Global 有限狀態機、輸出結果
Register File (RF)	儲存座標點
RF I/O	控制 RF 的資料路徑
Address Controller	控制 RF 當前的位址
Vector & Cross	將輸入的座標點形成向量並做外積

這樣的設計是以功能為分區，優點及特色是電路架構明確、並且可以重複利用同功能的部分；缺點是在資料路徑的控制上會有許多 MUX，這會導致面積較大。對於重複利用的例子以外積為例，當要做外積的話就透過資料路徑的管理，統一送入向量外積電路做運算，再把結果送回需要外積結果的電路，這樣就只要一個向量外積電路就好。

在向量外積電路中，會有多種架構的分析與比較，會在下列做說明。以下分別介紹各子電路的詳細規格及功能。

- 我們的 Global_FSM 的狀態如下圖所示：



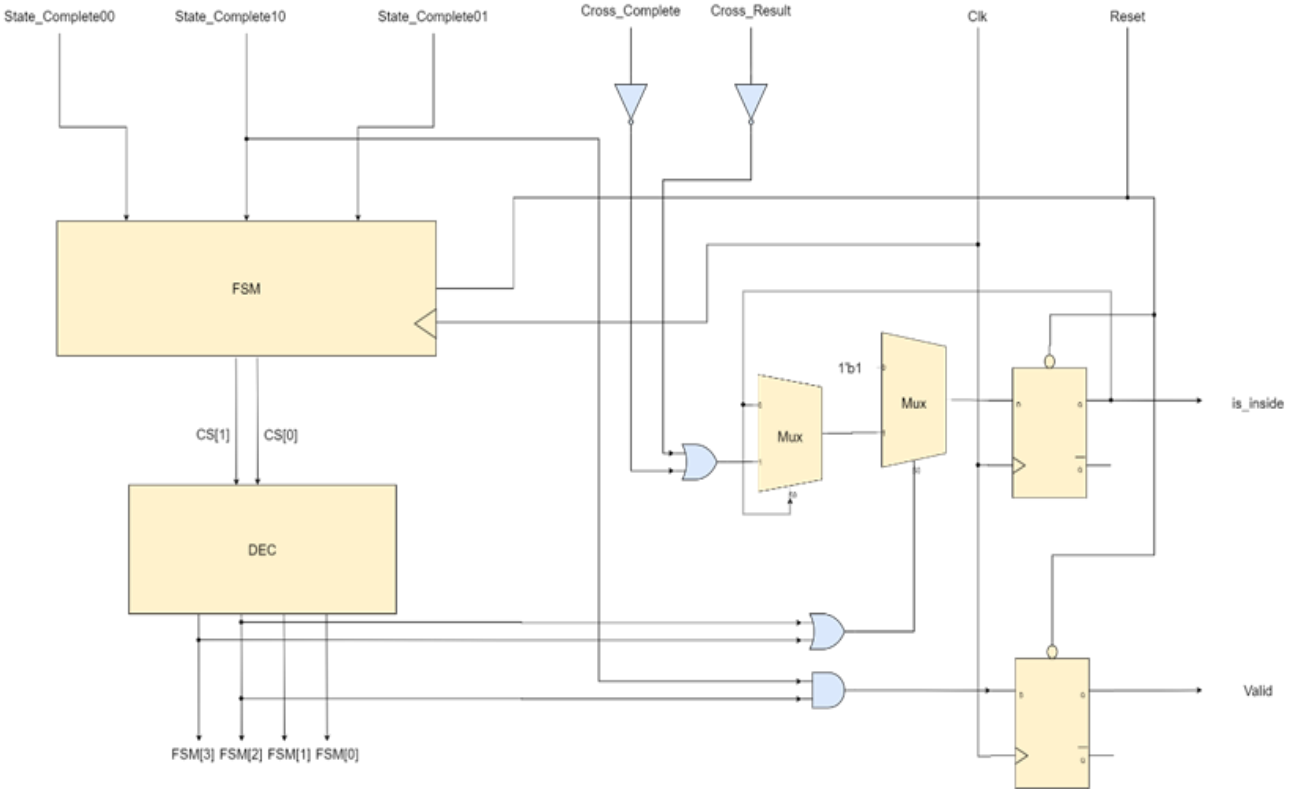
00：儲存外界給予的座標點

01：排序柵欄

10：判斷待測物

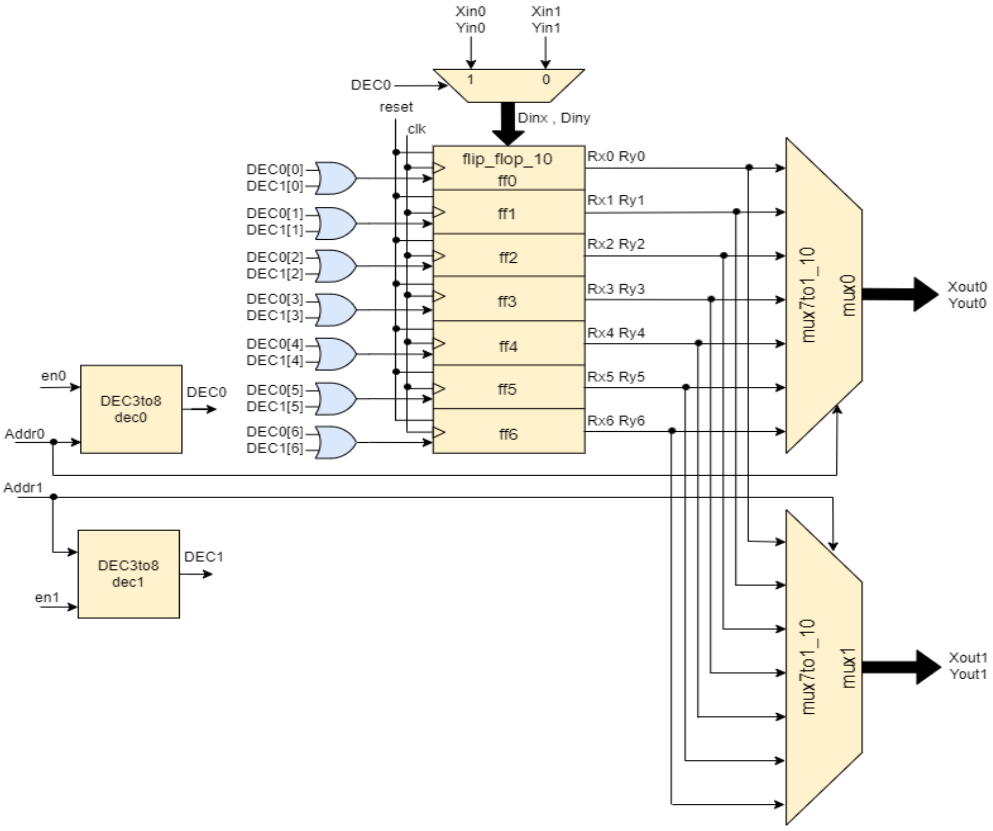
11：輸出判斷結果

- FSM & Output Control



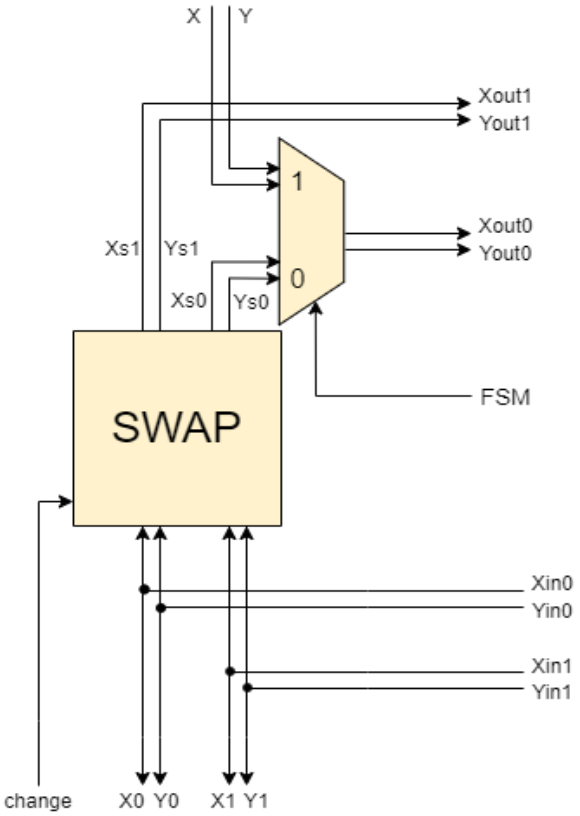
說明：有限狀態機的部分，我們會將目前的狀態做解碼，將四種狀態變成四條線的訊號，這樣有利於其他電路取用目前狀態的訊號。而 is_inside 的部分是在狀態 10 的階段，初始值會是 1，但如果偵測到有外積結果異號的情況就會維持 0，而 valid 則會在狀態 11 的時候將值拉起。

- RF



說明： 此 register file (RF)為兩組寫入兩組讀出，因此會有兩個位址訊號（Addr0, Addr1）控制兩個輸出分別要讀取哪一組 address 的內容。而此電路架構中，可存取 7 個 address 的內容，每一個 address 可存取一組座標，而每組座標皆由兩個 10 bits 的內容組成，因此總共會用到 140 個 flip-flop。而輸入的資料會由 Addr0 及 Addr1 經過兩個解碼器來控制要存在哪一個位址。

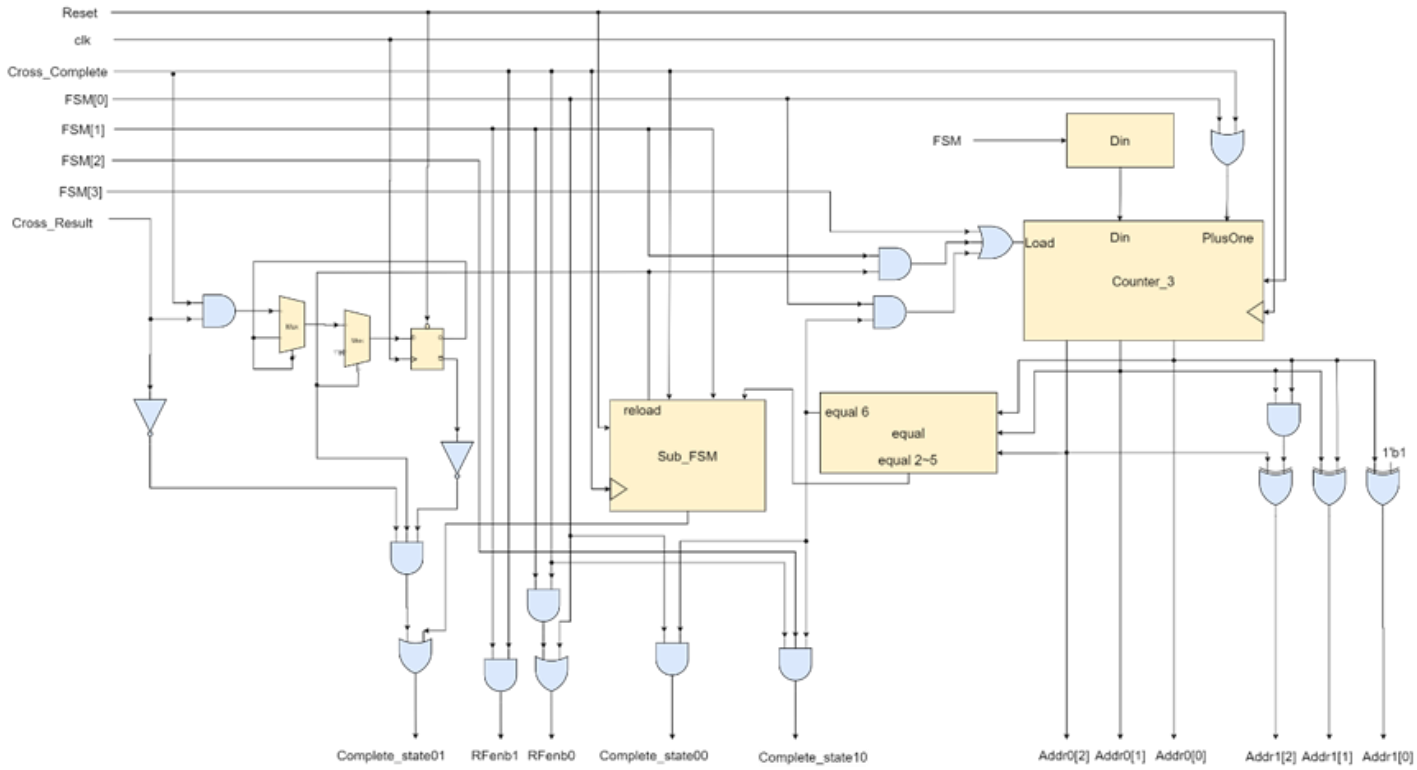
- RF_IO



說明： 當 finite_state_machine (FSM)輸出 1 時控制資料被寫入 RF 中，反之，則寫入 SWAP 電路輸出的其中一組座標 (Xs0,Ys0)。

SWAP 電路接收到外積電路(vector_cross)輸出 change 為 1 時，SWAP 電路會將從 RF 讀取的兩組座標做交換，若外積電路輸出 change 為 0 則不交換。

- Address Controller



說明:

主要由 Counter、Sub_FSM、Flag(電路左側)、判斷當前數數內容、加 1 電路(右下角的四個邏輯閘)所構成。

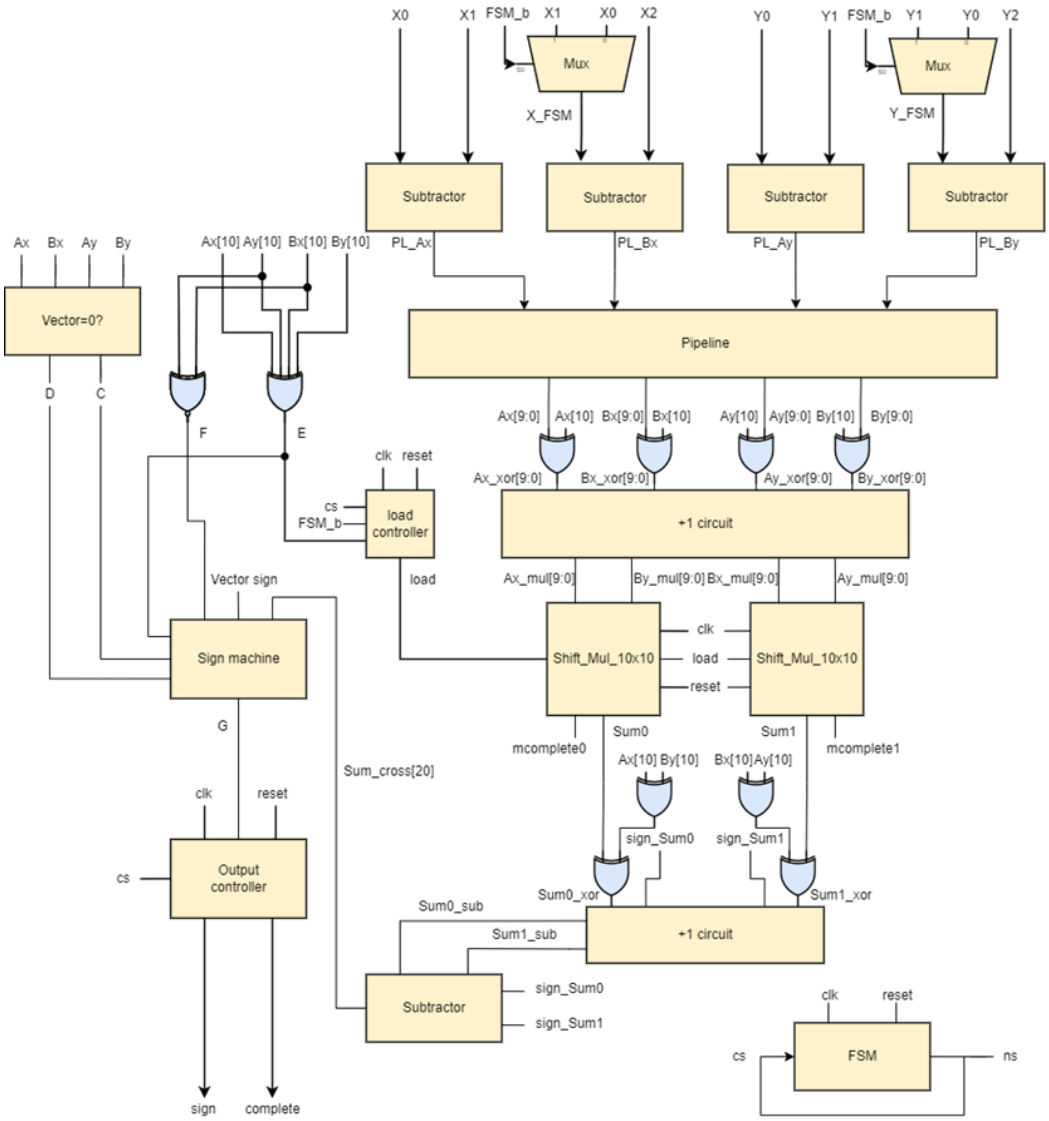
此電路的主要功能有兩項，第一項是控制 RF 內的位址；第二項是給予 Global_FSM 轉態的訊號。

對於第一項的功能，Counter 會依照 Global_FSM 做調變，狀態 00 的時候由 0 數到 6，把座標位置分別存入 RF 內的空間；在 01 的時候，根據 Bubble Sort 的排序演算法，指向需要做比較與排序的座標點；在 10 的時候則是指向需要觀察的座標點；在 11 的時候則重製。由於 Bubble Sort 演算法的特性以及觀察待測物的演算法，會需要取序列中相鄰兩個點的成員，因此使用加 1 電路來實現。

第二項是以 Counter 以及 Global_FSM 等等做判斷，完成時則輸出訊號給予有限狀態機做轉態，其中在狀態 01(排序)的時候，由 Flag 電路來偵測柵欄是否已排序好，如果已經排序完成則輸出完成訊號，這樣的好處是不用等待 Bubble Sort 完成，可以省下排序的時間。

- Vector & Cross

1. 電路架構 1 (pipeline)



- **電路演算法(正常情況)**

a. 將基準點(待測物)及兩組座標根據 Geofence 的主 FSM 狀態決定要進行排柵欄或測量待測物後擴位 (sign bit) 並輸入減法器產生兩組向量。

b. 由於產生的向量會有正負符號的問題，因此在輸入乘法器前會先根據向量的符號進行 2 補數的轉換 (XOR 和+1circuit)，並將轉為正數的兩組向量輸入兩組 sequential 的 10x10 移位乘法器。

c. 當乘法器完成運算後會先根據原先向量的正負號關係將乘法器輸出的兩組乘積進行 2 補數的轉換 (XOR 和+1 circuit)，接著擴位加上 sign bit 後輸入減法器產生向量外積結果。

d. 得到外積結果後取其最高位元 (sign bit) 存給 sign 並輸出外積符號和完成電路運算的 complete 訊號。

- **電路演算法 (特例)**

a. 由於電路的乘法器使用 sequential 的寫法會花費較多時間運算，因此利用了向量的符號及是否為零的特例提供可以提早輸出結果並不用進入乘法器的路徑。

b. 特例 1: 向量為零

AxBy-AyBx		
AxBy	AyBx	Sign
0	+	-
0	-	+
+	0	+
-	0	-

c. 特例 2: 向量符號

AxBy-AyBx				
Ax	By	Ay	Bx	Sign
+	+	+	-	+
+	+	-	+	+
-	-	+	-	+
-	-	-	+	+
+	-	+	+	-
+	-	-	-	-
-	+	+	+	-
-	+	-	-	-

- **架構優缺點**

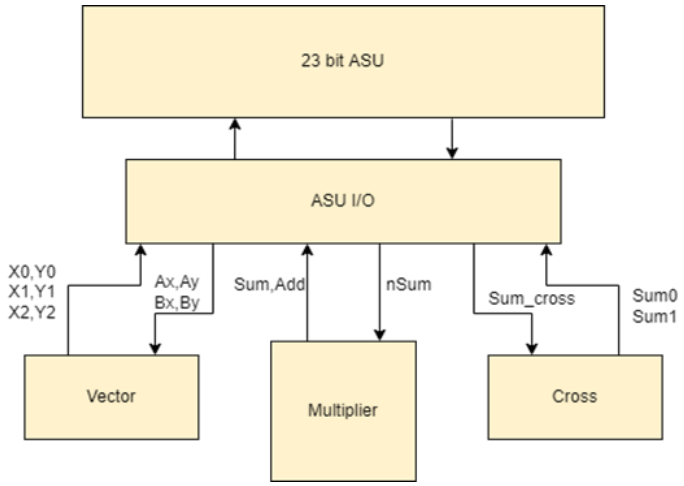
優點:

1. 乘法器使用 sequential 的寫法面積較小，利於並行運算。
2. 使用 pipeline 提升操作頻率。
3. 可使用特例的情況讓整體運算時間壓至 6 個 clk

缺點:

1. 乘法器使用 sequential 的寫法運算所費時間較長 (worst case=10 clk)，但在乘法器內加上提早完成的 flag 後可減少 0~9 個 clk 的運算時間。
2. 向量要為正數才能進入乘法器運算，因此會花費額外面積在加一電路上。

2. 電路架構 2 (pipeline+共用加減法器)



說明: 此架構與前一架構大致相同，但為了節省面積，將所有使用到的加法器和減法器整合為 2 個 23 bit 的加減法器，並規畫路徑根據 FSM 的狀態進行相對應的加法或減法。

3. 架構 1 和架構 2 比較

架構 1: 4 個 11 bit 減法器、2 個 20 bit 加法器、1 個 21 bit 減法器

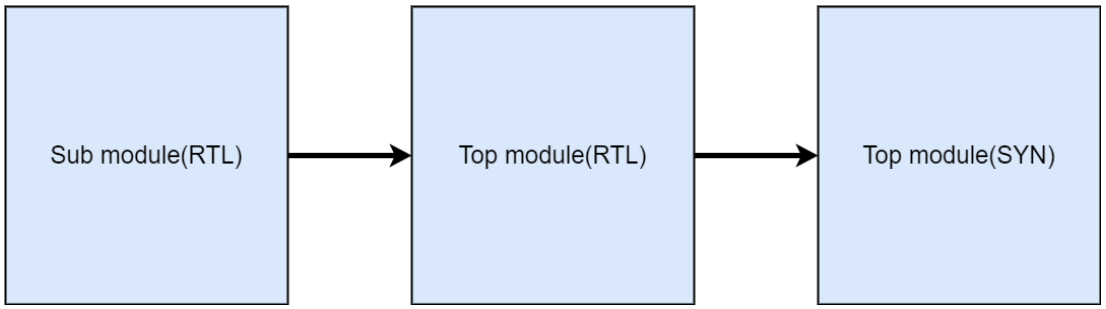
架構 2: 2 個 23 bit 加減法器、規劃路徑的 decoder

若各個加法器以 full adder 為單位且不考慮 decoder 面積的話，架構 2 的電路最多可省下 59 個 full adder 的面積。

4. 最終採用架構

架構 1 在邏輯合成後最佳效能為在操作頻率=4 (ns)時，此時電路面積為 17201 (um^2)，架構 2 在合成前原先預計可以省下 6000 的面積，但在合成後發現操作頻率壓低的話 23 bit 的加減法器會被 EDA 合成為 2 個 23 bit 加法器和減法器，且加上 decoder 的面積後總電路面積不減反增，而架構 2 最佳效能為在操作頻率=4.3(ns)時，此時電路面積為 32914 (um^2)。根據兩個架構在 Geofence 整個電路合成後的結果，最終向量外積電路所採用的架構為架構 1。

三、驗證流程說明

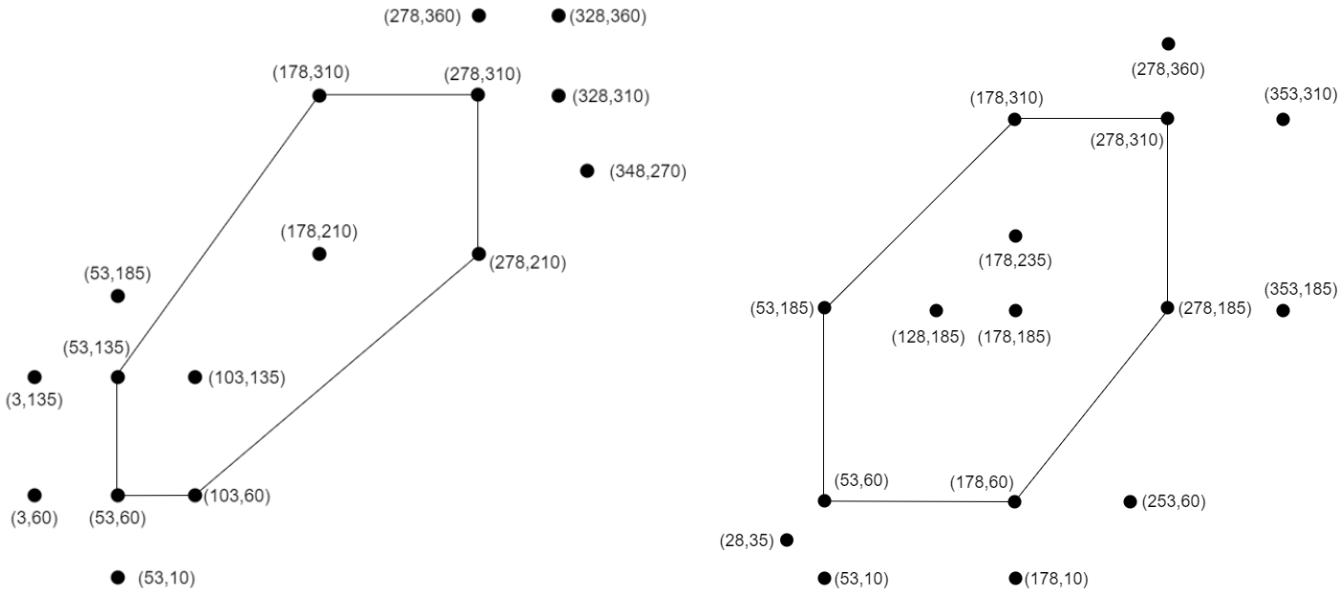


我們驗證電路的流程是先驗證 sub module，當 sub module 皆驗證成功後再去驗證 top module，這樣的好處是在整合後出問題時，可以更精確地抓出錯誤在哪。

首先在驗證 sub module 的部分，我們會先設計一個自動化的 TB，並且根據不同的電路來設計 Pattern 和 Expected，其中會盡可能的考慮各種 corner case，而這部分是為了要確認 RTL 的功能完整性，因此當 sub module 皆通過各自的 TB 後，就直接整合並且跑助教的 TB。

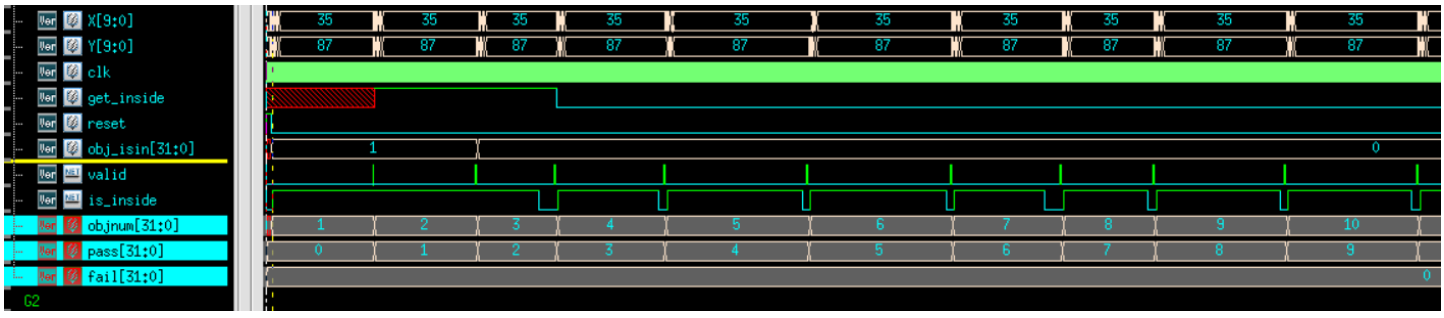
在驗證整合後電路的部分，為了避免合成後會增加更多錯誤的可能性，因此會先以 RTL 檔來做驗證，找出有問題的點，這一部分的問題大多都出在各模組之間的溝通與整合，待 RTL 模擬皆正確後，進行邏輯合成後的模擬並持續除錯直到通過全部的 Pattern。

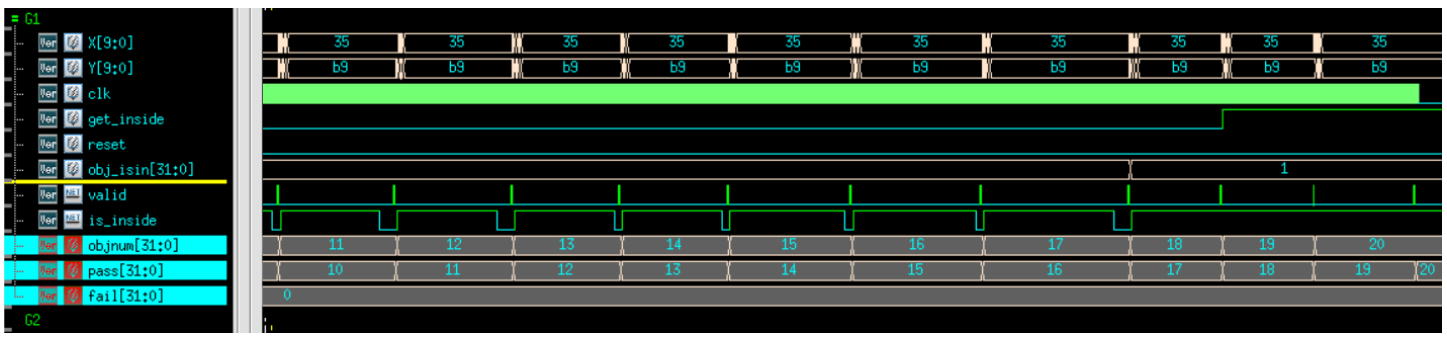
通過之後我們有再自己額外設計 Pattern 來做模擬與驗證，TB 是採用助教給的 TB，然後將 univ.data 的內容改為自己設計的數據，以此實現自動化的驗證，以下展示我們設計的 Pattern 與驗證的結果和波型。



```
Scenario1(in):     X     Y
       Object:   103,  135
          AP1:    53,   60
          AP2:   103,   60
          AP3:   278,  210
          AP4:   278,  310
          AP5:   178,  310
          AP6:    53,  135
Object1: Golde/Return => 1/1, PASS

Scenario2(in):     X     Y
       Object:   178,  210
          AP1:    53,   60
          AP2:   103,   60
          AP3:   278,  210
          AP4:   278,  310
          AP5:   178,  310
          AP6:    53,  135
Object2: Golde/Return => 1/1, PASS

Scenario3(out):    X     Y
       Object:    53,   10
          AP1:    53,   60
          AP2:   103,   60
          AP3:   278,  210
          AP4:   278,  310
          AP5:   178,  310
          AP6:    53,  135
Object3: Golde/Return => 0/0, PASS

Scenario4(out):    X     Y
       Object:     3,   60
          AP1:    53,   60
          AP2:   103,   60
          AP3:   278,  210
          AP4:   278,  310
          AP5:   178,  310
          AP6:    53,  135
Object4: Golde/Return => 0/0, PASS

Scenario5(out):    X     Y
       Object:     3,  135
          AP1:    53,   60
          AP2:   103,   60
          AP3:   278,  210
          AP4:   278,  310
          AP5:   178,  310
          AP6:    53,  135
Object5: Golde/Return => 0/0, PASS

Scenario6(out):    X     Y
       Object:    53,  185
          AP1:    53,   60
          AP2:   103,   60
          AP3:   278,  210
          AP4:   278,  310
          AP5:   178,  310
          AP6:    53,  135
Object6: Golde/Return => 0/0, PASS

Scenario7(out):    X     Y
       Object:   278,  360
          AP1:    53,   60
          AP2:   103,   60
          AP3:   278,  210
          AP4:   278,  310
          AP5:   178,  310
          AP6:    53,  135
Object7: Golde/Return => 0/0, PASS

Scenario8(out):    X     Y
       Object:   328,  360
          AP1:    53,   60
          AP2:   103,   60
          AP3:   278,  210
          AP4:   278,  310
          AP5:   178,  310
          AP6:    53,  135
Object8: Golde/Return => 0/0, PASS

Scenario9(out):    X     Y
       Object:   328,  310
          AP1:    53,   60
          AP2:   103,   60
          AP3:   278,  210
          AP4:   278,  310
          AP5:   178,  310
          AP6:    53,  135
Object9: Golde/Return => 0/0, PASS

Scenario10(out):   X     Y
       Object:   348,  270
          AP1:    53,   60
          AP2:   103,   60
          AP3:   278,  210
          AP4:   278,  310
          AP5:   178,  310
          AP6:    53,  135
Object10: Golde/Return => 0/0, PASS

Scenario11(out):   X     Y
       Object:    28,   35
          AP1:    53,   60
          AP2:   178,   60
          AP3:   278,  185
          AP4:   278,  310
          AP5:   178,  310
          AP6:    53,  185
Object11: Golde/Return => 0/0, PASS

Scenario12(out):   X     Y
       Object:    53,   10
          AP1:    53,   60
          AP2:   178,   60
          AP3:   278,  185
          AP4:   278,  310
          AP5:   178,  310
          AP6:    53,  185
Object12: Golde/Return => 0/0, PASS

Scenario13(out):   X     Y
       Object:   178,   10
          AP1:    53,   60
          AP2:   178,   60
          AP3:   278,  185
          AP4:   278,  310
          AP5:   178,  310
          AP6:    53,  185
Object13: Golde/Return => 0/0, PASS

Scenario14(out):   X     Y
       Object:   253,   60
          AP1:    53,   60
          AP2:   178,   60
          AP3:   278,  185
          AP4:   278,  310
          AP5:   178,  310
          AP6:    53,  185
Object14: Golde/Return => 0/0, PASS

Scenario15(out):   X     Y
       Object:   353,  185
          AP1:    53,   60
          AP2:   178,   60
          AP3:   278,  185
          AP4:   278,  310
          AP5:   178,  310
          AP6:    53,  185
Object15: Golde/Return => 0/0, PASS

Scenario16(out):   X     Y
       Object:   353,  310
          AP1:    53,   60
          AP2:   178,   60
          AP3:   278,  185
          AP4:   278,  310
          AP5:   178,  310
          AP6:    53,  185
Object16: Golde/Return => 0/0, PASS

Scenario17(out):   X     Y
       Object:   278,  360
          AP1:    53,   60
          AP2:   178,   60
          AP3:   278,  185
          AP4:   278,  310
          AP5:   178,  310
          AP6:    53,  185
Object17: Golde/Return => 0/0, PASS

Scenario18(in):    X     Y
       Object:   178,  235
          AP1:    53,   60
          AP2:   178,   60
          AP3:   278,  185
          AP4:   278,  310
          AP5:   178,  310
          AP6:    53,  185
Object18: Golde/Return => 1/1, PASS

Scenario19(in):    X     Y
       Object:   178,  185
          AP1:    53,   60
          AP2:   178,   60
          AP3:   278,  185
          AP4:   278,  310
          AP5:   178,  310
          AP6:    53,  185
Object19: Golde/Return => 1/1, PASS

Scenario20(in):    X     Y
       Object:   128,  185
          AP1:    53,   60
          AP2:   178,   60
          AP3:   278,  185
          AP4:   278,  310
          AP5:   178,  310
          AP6:    53,  185
Object20: Golde/Return => 1/1, PASS

          AP7:    53,  185
-------------------------------------------------
-- Simulation finish,  Pass = 20 , Fail =  0   --
-------------------------------------------------
Simulation complete via $finish(1) at time 7552 NS + 0
```

```
-------------------------------------------------
-- Simulation finish,  Pass = 20 , Fail =  0   --
-------------------------------------------------
Simulation complete via $finish(1) at time 7552 NS + 0
./tb.sv:147              $finish;
```





由上面幾張圖可以發現，電路皆有通過我們自己設計的 pattern，由波型圖也可以看出我們的 fail 為 0，並且皆有正確輸出 valid 和 is_inside 值。

當這些模擬都結束後，我們進一步去進行 Spyglass 對 Coding Style 做驗證，稍後會做詳細的說明。

◆ Spyglass coding style check and enhancement

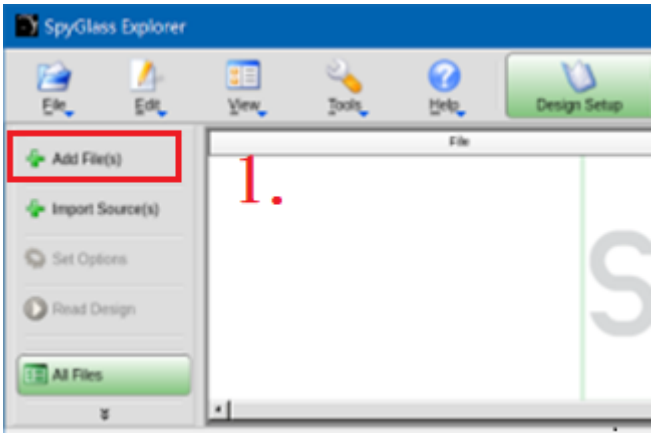
操作流程

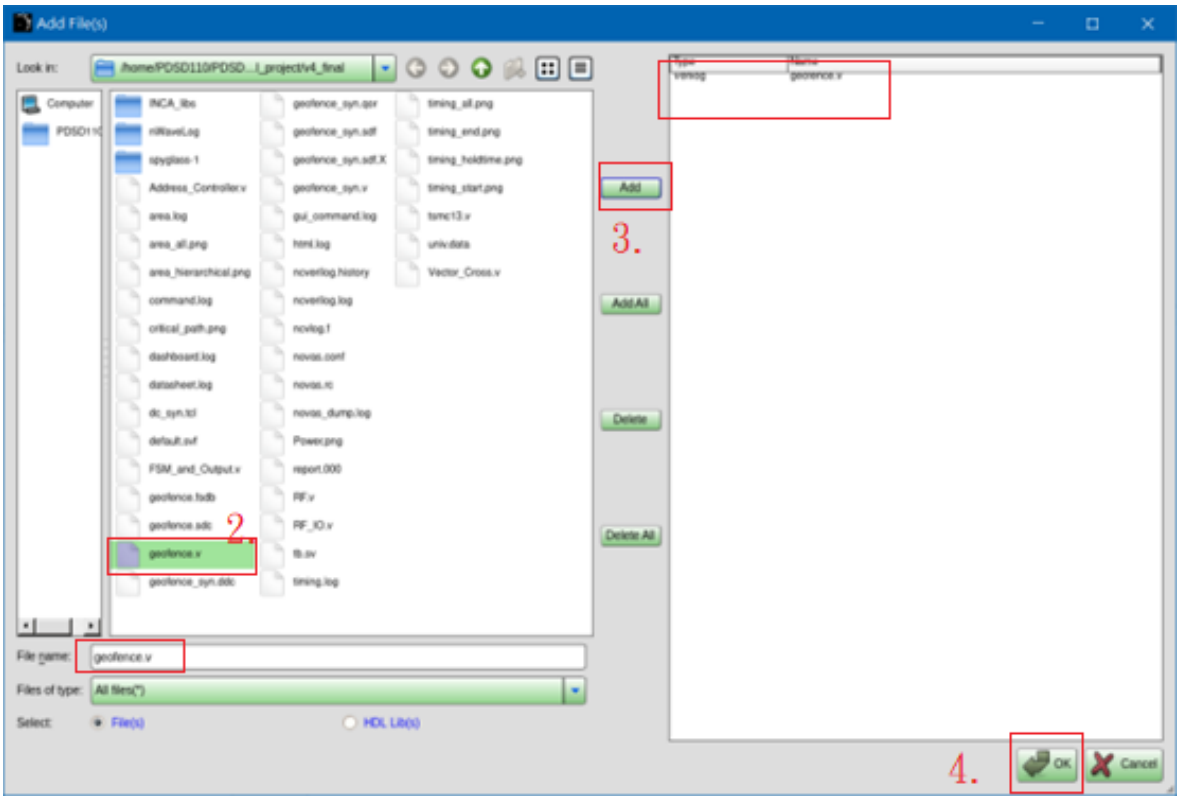
Step1: 開啟 spyglass

開啟欲執行的資料夾

→ 輸入 source /usr/cad/synopsys/CIC/spyglass.cshrc → 輸入 spyglass

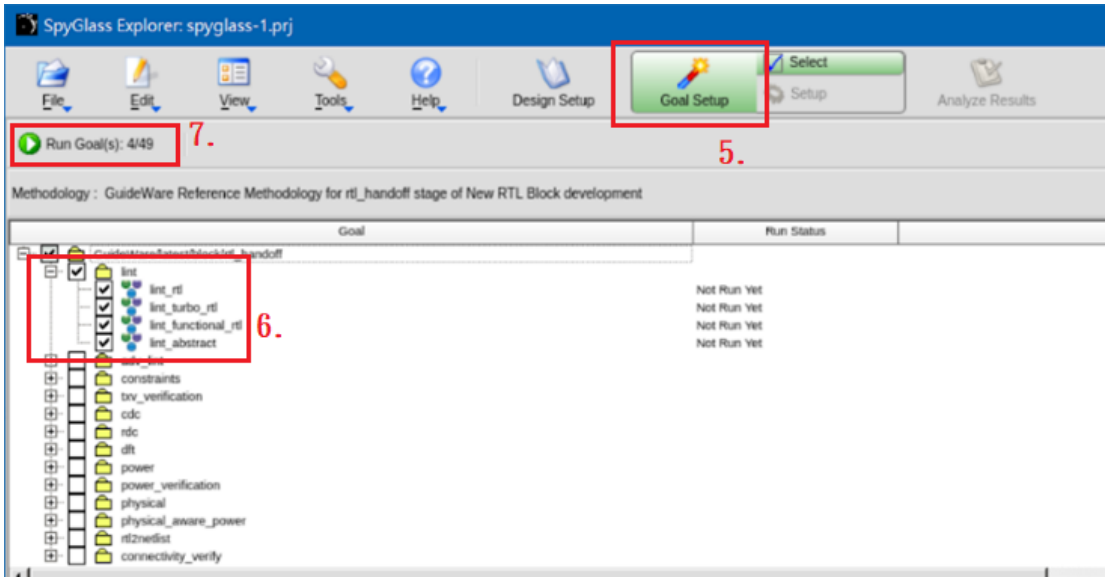
Step2: 加入欲執行的.v 檔





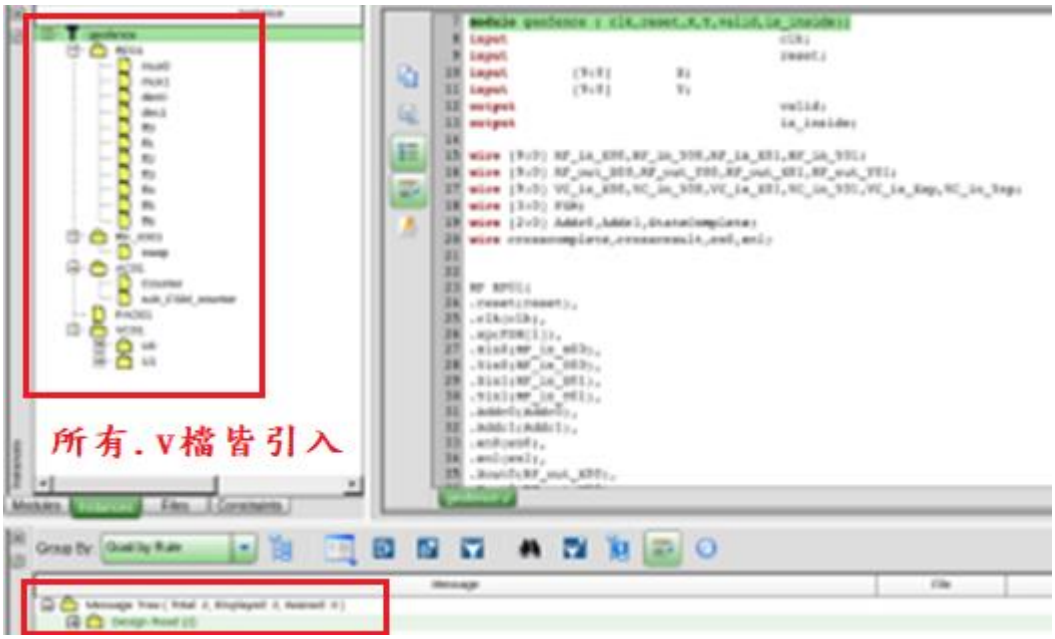
Step3: goal set up（lint 選項全勾）

Step4: run goals



Step5: check result

```
-------------------------------------------------------------------------------------------------
  Goal Violation Summary:
     Waived   Messages:                    0 Errors,      0 Warnings,      0 Infos
     Reported Messages:        0 Fatals,   0 Errors,      0 Warnings,      3 Infos
-------------------------------------------------------------------------------------------------
```



說明: 沒有出現任何 error 和 warning，但一開始將所有.v 檔都引入會出現 15 個 fatal，原因是我們在 top module（Geofence.v）中已將其他 .v 檔引入，因此這 15 個 fatal 的原因皆為 re-declaration of a module with the same name.

四、邏輯合成流程說明與合成參數

1. 邏輯合成流程說明

除了 sdc 檔與 TB 的 cycle 外，我們全部使用助教給予的數據

Step1：

調整 sdc 檔的時間限制

```
3  set cycle 4.0
```

同時調整 TB 檔內的運作週期，使其和時間限制相同

```
2  `define CYCLE      4.0
```

Step2：

開啟 design vision 並 source tcl 檔

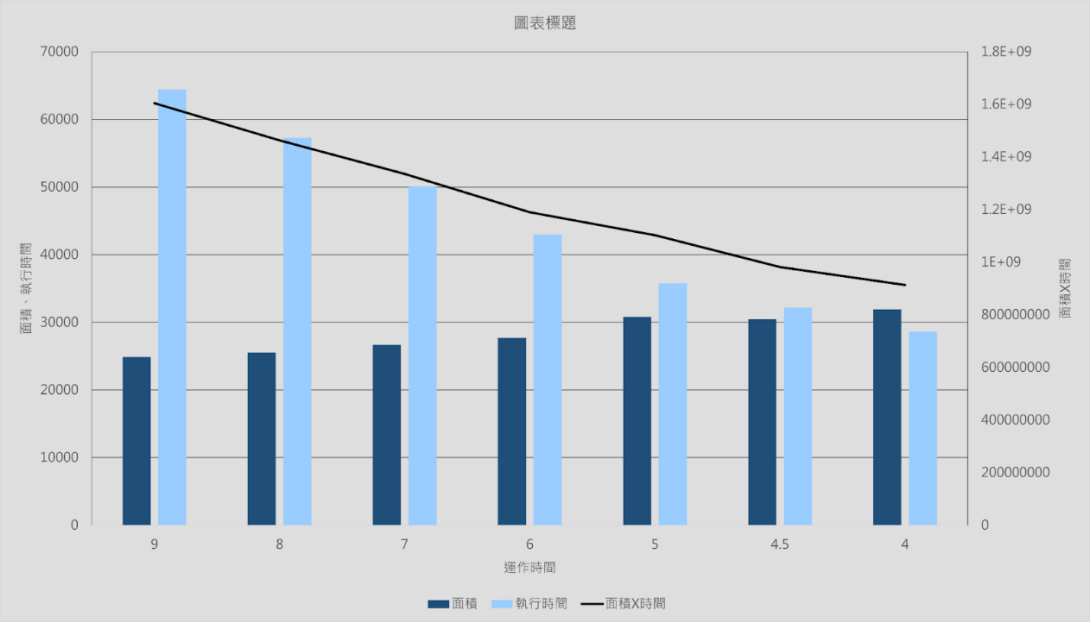
Step3： 等待程式跑完即完成合成

合成後的數據整理：

以下分成兩個部分做整理，首先是我們採用的電路架構，其次是向量外積電路中，有共用加減法器的情況。

A. 採用的電路

- 我們對不同的運作週期下去分析，如下圖：



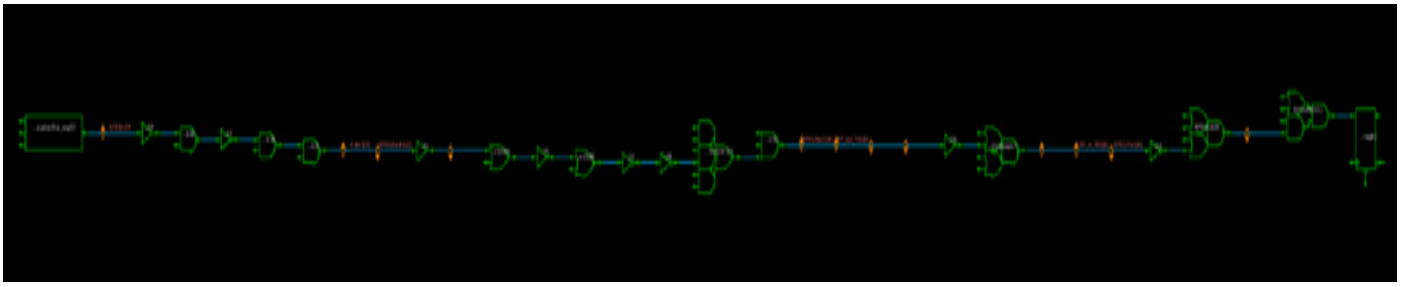
最後發現在運作週期為 4 (ns) 下的時間 X 面積表現最好，因此以下都是在運作週期為 4 (ns) 下得出的結果。

- 電路總面積以及各電路的占比：

```
Hierarchical area distribution
------------------------------

                                Global cell area          Local cell area
                                -----------------  ----------------------------
Hierarchical cell               Absolute    Percent  Combi-      Noncombi-   Black-
                                Total       Total    national    national    boxes   Design
------------------------------  ----------  -------  ----------  ----------  ------  ------------------------
geofence                      31907.7248    100.0      95.0544      0.0000  0.0000  geofence
AC01                            789.2910      2.5     354.7566     32.2506  0.0000  Address_Controller
AC01/Counter                    278.3736      0.9     120.5154    157.8582  0.0000  counter_3
AC01/sub_FSM_counter            123.9102      0.4      59.4090     64.5012  0.0000  counter_2
FA001                           293.6502      0.9     134.0946    159.5556  0.0000  FSM_and_Output
RF01                          12197.5162     38.2    2182.8564      0.0000  0.0000  RF
RF01/dec0                       112.0284      0.4     112.0284      0.0000  0.0000  DEC3to8_1
RF01/dec1                        88.2648      0.3      88.2648      0.0000  0.0000  DEC3to8_0
RF01/ff0                        899.6220      2.8     254.6100    645.0120  0.0000  flip_flop_10_6
RF01/ff1                        892.8324      2.8     247.8204    645.0120  0.0000  flip_flop_10_5
RF01/ff2                        877.5558      2.8     232.5438    645.0120  0.0000  flip_flop_10_4
RF01/ff3                        894.5298      2.8     249.5178    645.0120  0.0000  flip_flop_10_3
RF01/ff4                        896.2272      2.8     251.2152    645.0120  0.0000  flip_flop_10_2
RF01/ff5                        855.4896      2.7     210.4776    645.0120  0.0000  flip_flop_10_1
RF01/ff6                        879.2532      2.8     234.2412    645.0120  0.0000  flip_flop_10_0
RF01/mux0                      1649.8728      5.2    1649.8728      0.0000  0.0000  MUX7to1_10_1
RF01/mux1                      1968.9840      6.2    1968.9840      0.0000  0.0000  MUX7to1_10_0
RF_IO01                        1330.7616      4.2     687.4470      0.0000  0.0000  RF_IO
RF_IO01/swap                    643.3146      2.0     643.3146      0.0000  0.0000  SWAP
VC01                          17201.4514     53.9    4406.4504   1738.1376  0.0000  Vector_Cross
VC01/U0                        3364.2467     10.5      40.7376      0.0000  0.0000  Shift_Mul_10x10_1
VC01/U0/U0                      850.3974      2.7     205.3854    645.0120  0.0000  SLreg_1
VC01/U0/U1                      454.9032      1.4     117.1206    337.7826  0.0000  SRreg_1
VC01/U0/U3                     2018.2086      6.3     308.9268    680.6574  0.0000  ACCreg20_1
VC01/U0/U3/add_364             1028.6244      3.2    1028.6244      0.0000  0.0000  ACCreg20_1_DW01_add_1
VC01/U1                        3438.9323     10.8      40.7376      0.0000  0.0000  Shift_Mul_10x10_0
VC01/U1/U0                      848.7000      2.7     203.6880    645.0120  0.0000  SLreg_0
VC01/U1/U1                      459.9954      1.4     122.2128    337.7826  0.0000  SRreg_0
VC01/U1/U3                     2089.4994      6.5     314.0190    699.3288  0.0000  ACCreg20_0
VC01/U1/U3/add_364             1076.1516      3.4    1076.1516      0.0000  0.0000  ACCreg20_0_DW01_add_1
VC01/sub_276                    794.3832      2.5     794.3832      0.0000  0.0000  Vector_Cross_DW01_sub_7
VC01/sub_58                     919.9908      2.9     919.9908      0.0000  0.0000  Vector_Cross_DW01_sub_4
VC01/sub_59                     760.4352      2.4     760.4352      0.0000  0.0000  Vector_Cross_DW01_sub_3
VC01/sub_60                     869.0688      2.7     869.0688      0.0000  0.0000  Vector_Cross_DW01_sub_2
VC01/sub_61                     909.8064      2.9     909.8064      0.0000  0.0000  Vector_Cross_DW01_sub_1
------------------------------  ----------  -------  ----------  ----------  ------  ------------------------
Total                                                 21894.7625  10012.9623  0.0000

***** End Of Report *****
```

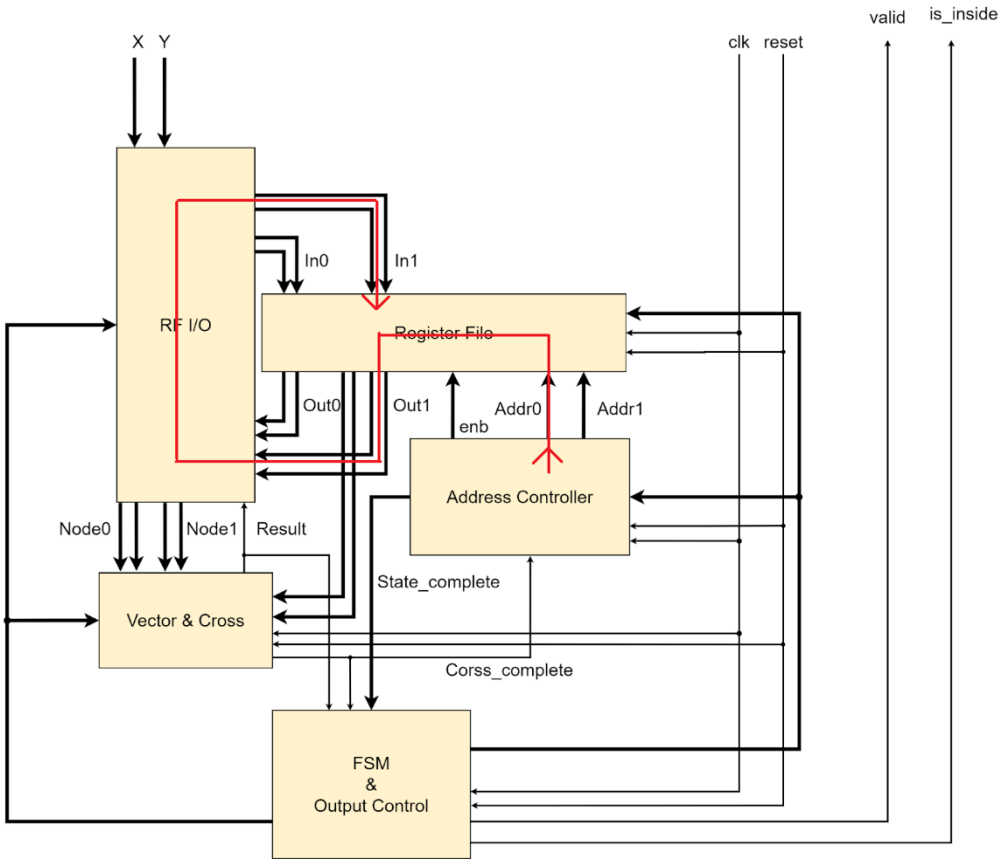
名稱	面積 (um^2)	占比 (%)	邏輯閘數量 (NAND2)
Geofence	31908	100	6381.6
FSM & output	294	0.9	58.8
RF	12198	38.2	2439.6
RF I/O	1331	4.2	266.2
Vector & cross	17201	53.9	3440.2
Address Control	789	2.5	157.8

- 最長路徑：



```
Des/Clust/Port     Wire Load Model       Library
------------------------------------------------
geofence           tsmc13_wl10           slow

Point                                                   Incr       Path
-----------------------------------------------------------------------------
clock clk (rise edge)                                   0.00       0.00
clock network delay (ideal)                             0.50       0.50
AC01/Counter/Out_reg[1]/CK (DFFRHQX8)                   0.00       0.50 r
AC01/Counter/Out_reg[1]/Q (DFFRHQX8)                    0.26       0.76 r
AC01/Counter/Out[1] (counter_3)                         0.00       0.76 r
AC01/U19/Y (BUFX20)                                     0.14       0.90 r
AC01/U20/Y (AND2X8)                                     0.15       1.05 r
AC01/U17/Y (CLKINVX8)                                   0.07       1.12 f
AC01/U18/Y (NAND2X8)                                    0.07       1.19 r
AC01/U21/Y (NAND2X8)                                    0.06       1.25 f
AC01/Addr1[2] (Address_Controller)                      0.00       1.25 f
RF01/Addr1[2] (RF)                                      0.00       1.25 f
RF01/U241/Y (BUFX20)                                    0.13       1.38 f
RF01/mux1/addr[2] (MUX7to1_10_0)                        0.00       1.38 f
RF01/mux1/U86/Y (NOR2X8)                                0.09       1.48 r
RF01/mux1/U55/Y (INVX12)                                0.05       1.53 f
RF01/mux1/U80/Y (NOR2X8)                                0.14       1.67 r
RF01/mux1/U22/Y (INVX16)                                0.08       1.75 f
RF01/mux1/U48/Y (INVX20)                                0.11       1.85 r
RF01/mux1/U123/Y (AOI222XL)                             0.26       2.12 f
RF01/mux1/U50/Y (NAND3X1)                               0.39       2.51 r
RF01/mux1/Douty[6] (MUX7to1_10_0)                       0.00       2.51 r
RF01/Yout1[6] (RF)                                      0.00       2.51 r
RF_IO01/Yin1[6] (RF_IO)                                 0.00       2.51 r
RF_IO01/swap/Yin1[6] (SWAP)                             0.00       2.51 r
RF_IO01/swap/U16/Y (CLKINVX3)                           0.22       2.74 f
RF_IO01/swap/U7/Y (OAI22X1)                             0.30       3.03 r
RF_IO01/swap/Yout1[6] (SWAP)                            0.00       3.03 r
RF_IO01/Yout1[6] (RF_IO)                                0.00       3.03 r
RF01/Yin1[6] (RF)                                       0.00       3.03 r
RF01/U102/Y (INVX3)                                     0.29       3.32 f
RF01/U225/Y (OAI22XL)                                   0.51       3.84 r
RF01/ff0/Yin[6] (flip_flop_10_6)                        0.00       3.84 r
RF01/ff0/U11/Y (OAI2BB2X1)                              0.30       4.13 r
RF01/ff0/Yout_reg[6]/D (DFFRX1)                         0.00       4.13 r
data arrival time                                                  4.13

clock clk (rise edge)                                   4.00       4.00
clock network delay (ideal)                             0.50       4.50
clock uncertainty                                      -0.10       4.40
RF01/ff0/Yout_reg[6]/CK (DFFRX1)                        0.00       4.40 r
library setup time                                     -0.27       4.13
data required time                                                 4.13
-----------------------------------------------------------------------------
data required time                                                 4.13
data arrival time                                                 -4.13
-----------------------------------------------------------------------------
slack (MET)                                                        0.00
```

➢ 在電路中的位置如下圖：



- 根據 qor 檔的資料：

```
Timing Path Group 'clk'
-----------------------------------
Levels of Logic:                18.00
Critical Path Length:            3.63
Critical Path Slack:             0.00
Critical Path Clk Period:        4.00
Total Negative Slack:            0.00
No. of Violating Paths:          0.00
Worst Hold Violation:            0.00
Total Hold Violation:            0.00
No. of Hold Violations:          0.00
-----------------------------------
```

我們最長路徑的邏輯閘層數為 18 個邏輯閘。

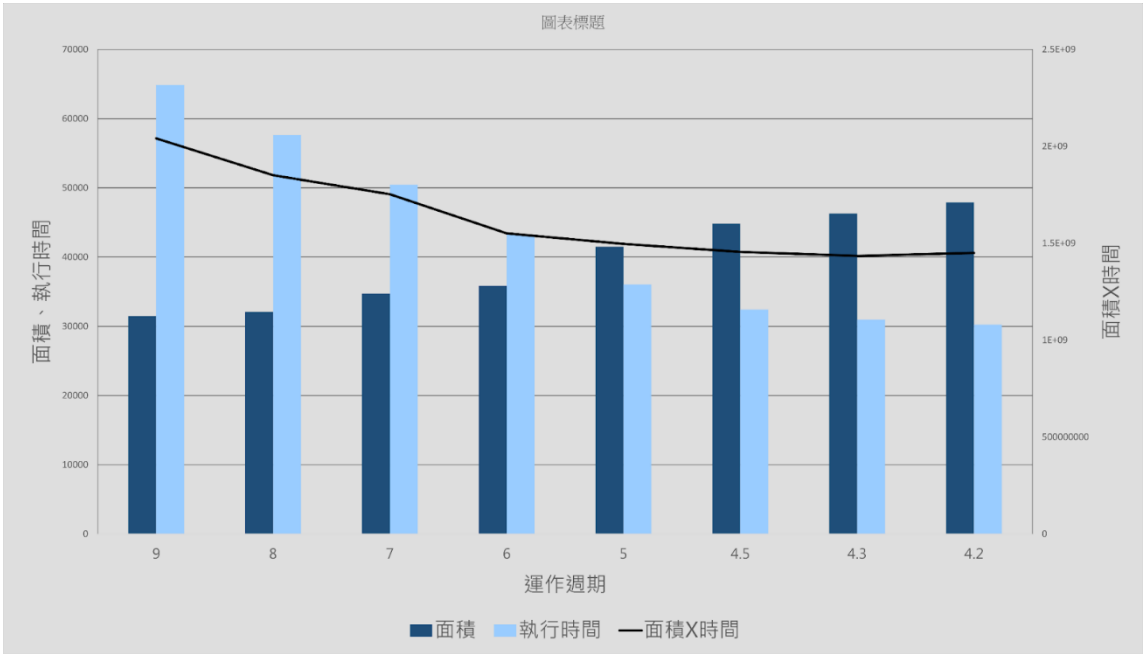
- 下圖是功率以及最短路徑：

```
****************************************
Report : power
        -analysis_effort low
Design : geofence
Version: P-2019.03
Date   : Sun Jun 12 15:36:47 2022
****************************************


Library(s) Used:

    slow (File: /mnt3/CBDK_IC_Contest_v2.1/SynopsysDC/db/slow.db)


Operating Conditions: slow   Library: slow
Wire Load Model Mode: top

Design        Wire Load Model            Library
------------------------------------------------
geofence              tsmc13_wl10       slow


Global Operating Voltage = 1.08
Power-specific unit information :
    Voltage Units = 1V
    Capacitance Units = 1.000000pf
    Time Units = 1ns
    Dynamic Power Units = 1mW    (derived from V,C,T units)
    Leakage Power Units = 1pW


  Cell Internal Power  =   1.8798 mW   (74%)
  Net Switching Power  = 659.8390 uW   (26%)
                         ---------
Total Dynamic Power    =   2.5397 mW  (100%)

Cell Leakage Power     =  28.6664 uW


                 Internal         Switching           Leakage            Total
Power Group      Power            Power               Power              Power   (   %    )  Attrs
--------------------------------------------------------------------------------------------------
io_pad             0.0000           0.0000              0.0000           0.0000  (   0.00%)
memory             0.0000           0.0000              0.0000           0.0000  (   0.00%)
black_box          0.0000           0.0000              0.0000           0.0000  (   0.00%)
clock_network      0.0000           0.0000              0.0000           0.0000  (   0.00%)
register           1.6835        7.5181e-02          9.2102e+06           1.7679  (  68.83%)
sequential         0.0000           0.0000              0.0000           0.0000  (   0.00%)
combinational      0.1964           0.5847          1.9456e+07           0.8005  (  31.17%)
--------------------------------------------------------------------------------------------------
Total              1.8798 mW        0.6598 mW       2.8666e+07 pW        2.5683 mW

***** End Of Report *****
```

```
Report : timing
        -path full
        -delay min
        -max_paths 1
        -sort_by group
Design : geofence
Version: P-2019.03
Date   : Sun Jun 12 15:39:19 2022
****************************************

Operating Conditions: slow   Library: slow
Wire Load Model Mode: top

  Startpoint: RF01/ff1/Xout_reg[6]
              (rising edge-triggered flip-flop clocked by clk)
  Endpoint: RF01/ff1/Xout_reg[6]
            (rising edge-triggered flip-flop clocked by clk)
  Path Group: clk
  Path Type: min

  Des/Clust/Port     Wire Load Model       Library
  ------------------------------------------------
  geofence           tsmc13_wl10           slow

  Point                                    Incr       Path
  -----------------------------------------------------------
  clock clk (rise edge)                    0.00       0.00
  clock network delay (ideal)              0.50       0.50
  RF01/ff1/Xout_reg[6]/CK (DFFRX1)         0.00       0.50 r
  RF01/ff1/Xout_reg[6]/QN (DFFRX1)         0.45       0.95 r
  RF01/ff1/U20/Y (OAI2BB2X2)               0.12       1.08 f
  RF01/ff1/Xout_reg[6]/D (DFFRX1)          0.00       1.08 f
  data arrival time                                   1.08

  clock clk (rise edge)                    0.00       0.00
  clock network delay (ideal)              0.50       0.50
  clock uncertainty                        0.10       0.60
  RF01/ff1/Xout_reg[6]/CK (DFFRX1)         0.00       0.60 r
  library hold time                       -0.03       0.57
  data required time                                  0.57
  -----------------------------------------------------------
  data required time                                  0.57
  data arrival time                                  -1.08
  -----------------------------------------------------------
  slack (MET)                                        0.51


***** End Of Report *****
```

B. 向量外積電路中使用共用的加減法器

- 我們對不同的運作週期下去分析，如下圖：



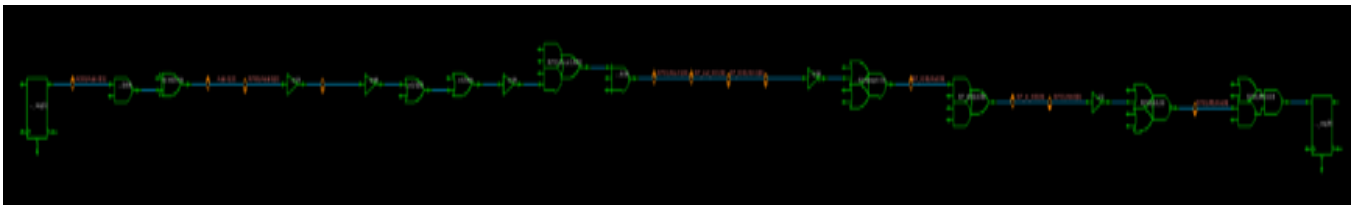
最後發現在運作週期為 4.3 (ns) 下的時間 X 面積表現最好，因此以下都是在運作週期為 4.3 (ns) 下得出的結果。

- 電路總面積以及各電路的占比：

```
Hierarchical area distribution
------------------------------

                                Global cell area           Local cell area
                                ----------------  -----------------------------
Hierarchical cell               Absolute    Percent  Combi-      Noncombi-   Black-
                                Total       Total    national    national    boxes   Design
------------------------------  ----------  -------  ----------  ----------  ------  ----------------------
geofence                       46294.8871    100.0     28.8558      0.0000  0.0000  geofence
AC01                             719.6976      1.6    342.8748     32.2506  0.0000  Address_Controller
AC01/Counter                     227.4516      0.5     84.8700    142.5816  0.0000  counter_3
AC01/sub_FSM_counter             117.1206      0.3     52.6194     64.5012  0.0000  counter_2
FA001                            269.8866      0.6    110.3310    159.5556  0.0000  FSM_and_Output
RF01                           10980.4804     23.7   2079.3150      0.0000  0.0000  RF
RF01/dec0                        105.2388      0.2    105.2388      0.0000  0.0000  DEC3to8_1
RF01/dec1                         79.7778      0.2     79.7778      0.0000  0.0000  DEC3to8_0
RF01/ff0                         979.3998      2.1    334.3878    645.0120  0.0000  flip_flop_10_6
RF01/ff1                         882.6480      1.9    237.6360    645.0120  0.0000  flip_flop_10_5
RF01/ff2                         887.7402      1.9    242.7282    645.0120  0.0000  flip_flop_10_4
RF01/ff3                         903.0168      2.0    258.0048    645.0120  0.0000  flip_flop_10_3
RF01/ff4                         908.1090      2.0    263.0970    645.0120  0.0000  flip_flop_10_2
RF01/ff5                         894.5298      1.9    244.4256    650.1042  0.0000  flip_flop_10_1
RF01/ff6                         889.4376      1.9    244.4256    645.0120  0.0000  flip_flop_10_0
RF01/mux0                       1060.8750      2.3   1060.8750      0.0000  0.0000  MUX7to1_10_1
RF01/mux1                       1310.3928      2.8   1310.3928      0.0000  0.0000  MUX7to1_10_0
RF_IO01                         1381.6836      3.0    541.4706      0.0000  0.0000  RF_IO
RF_IO01/swap                     840.2130      1.8    840.2130      0.0000  0.0000  SWAP
VC01                           32914.2832     71.1   9704.0358   2705.6555  0.0000  Vector_Cross
VC01/U0                         2727.7218      5.9     54.3168      0.0000  0.0000  Shift_Mul_10x10_1
VC01/U0/U0                      1030.3218      2.2    317.4138    712.9080  0.0000  SLreg_1
VC01/U0/U1                       504.1278      1.1    166.3452    337.7826  0.0000  SRreg_1
VC01/U0/U3                      1138.9554      2.5    325.9008    813.0546  0.0000  ACCreg20_1
VC01/U1                         2714.1426      5.9     64.5012      0.0000  0.0000  Shift_Mul_10x10_0
VC01/U1/U0                       977.7024      2.1    295.3476    682.3548  0.0000  SLreg_0
VC01/U1/U1                       594.0900      1.3    178.2270    415.8630  0.0000  SRreg_0
VC01/U1/U3                      1077.8490      2.3    288.5580    789.2910  0.0000  ACCreg20_0
VC01/U2                         6791.2973     14.7   1198.3644      0.0000  0.0000  ASU_1
VC01/U2/add_351                 2715.8400      5.9   2715.8400      0.0000  0.0000  ASU_1_DW01_add_3
VC01/U2/sub_351                 2877.0930      6.2   2877.0930      0.0000  0.0000  ASU_1_DW01_sub_8
VC01/U3                         5762.6730     12.4    857.1870      0.0000  0.0000  ASU_0
VC01/U3/add_351                 2283.0030      4.9   2283.0030      0.0000  0.0000  ASU_0_DW01_add_3
VC01/U3/sub_351                 2622.4830      5.7   2622.4830      0.0000  0.0000  ASU_0_DW01_sub_1
VC01/add_244                    1256.0760      2.7   1256.0760      0.0000  0.0000  Vector_Cross_DW01_inc_3
VC01/add_245                    1252.6812      2.7   1252.6812      0.0000  0.0000  Vector_Cross_DW01_inc_2
------------------------------  ----------  -------  ----------  ----------  ------  ----------------------
Total                                                34918.9126  11375.9745  0.0000

***** End Of Report *****
```

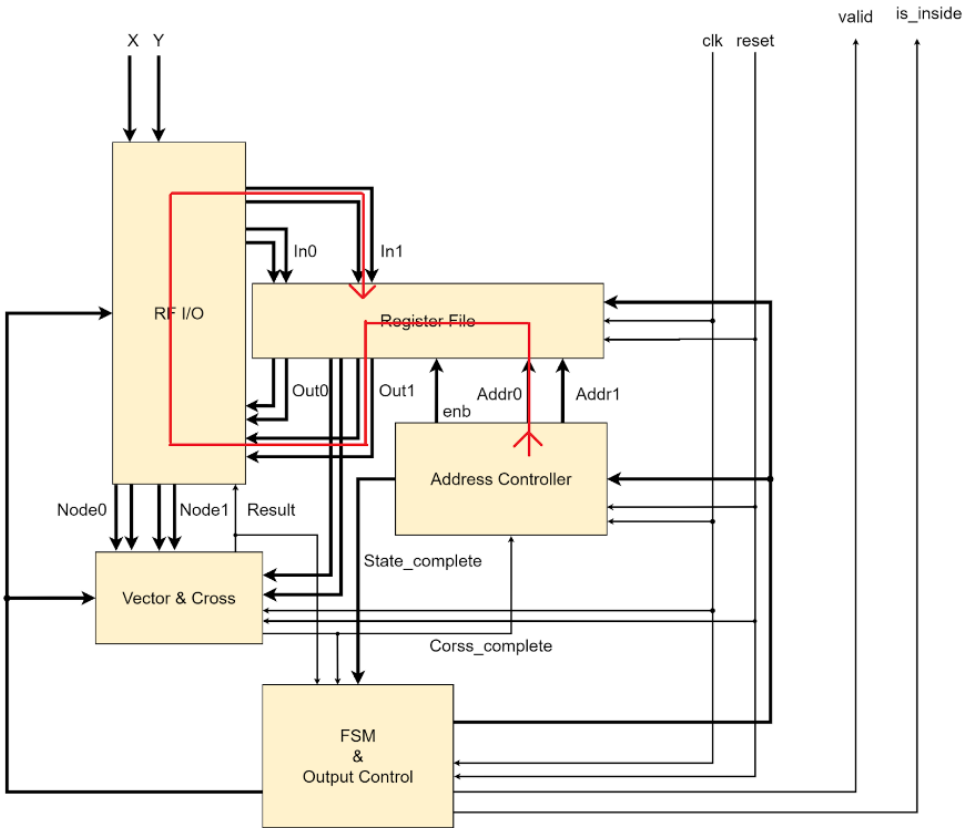
名稱	面積(um^2)	占比(%)	邏輯閘數量(NAND2)
Geofence	46295	100	9259.0
FSM & output	270	0.6	54.0
RF	10980	23.7	2196.0
RF I/O	1382	3.0	276.4
Vector & cross	32914	71.1	6582.8
Address Control	720	1.6	144.0

➢ 最長路徑：



Point	Incr	Path
clock clk (rise edge)	0.00	0.00
clock network delay (ideal)	0.50	0.50
AC01/Counter/Out_reg[1]/CK (DFFRX4)	0.00	0.50 r
AC01/Counter/Out_reg[1]/Q (DFFRX4)	0.71	1.21 f
AC01/Counter/Out[1] (counter_3)	0.00	1.21 f
AC01/U6/Y (NAND2X6)	0.14	1.34 r
AC01/U19/Y (XNOR2X4)	0.18	1.52 f
AC01/Addr1[2] (Address_Controller)	0.00	1.52 f
RF01/Addr1[2] (RF)	0.00	1.52 f
RF01/U23/Y (BUFX16)	0.16	1.69 f
RF01/mux1/addr[2] (MUX7to1_10_0)	0.00	1.69 f
RF01/mux1/U48/Y (INVX8)	0.08	1.77 r
RF01/mux1/U22/Y (OR2X2)	0.16	1.93 r
RF01/mux1/U44/Y (NOR2X4)	0.09	2.03 f
RF01/mux1/U28/Y (BUFX20)	0.17	2.19 f
RF01/mux1/U62/Y (AOI22XL)	0.49	2.68 r
RF01/mux1/U26/Y (NAND3X1)	0.39	3.07 f
RF01/mux1/Doutx[9] (MUX7to1_10_0)	0.00	3.07 f
RF01/Xout1[9] (RF)	0.00	3.07 f
RF_IO01/Xin1[9] (RF_IO)	0.00	3.07 f
RF_IO01/swap/Xin1[9] (SWAP)	0.00	3.07 f
RF_IO01/swap/U16/Y (CLKINVX4)	0.17	3.24 r
RF_IO01/swap/U10/Y (OAI22X2)	0.13	3.36 f
RF_IO01/swap/Xout0[9] (SWAP)	0.00	3.36 f
RF_IO01/U60/Y (AO22X2)	0.40	3.76 f
RF_IO01/Xout0[9] (RF_IO)	0.00	3.76 f
RF01/Xin0[9] (RF)	0.00	3.76 f
RF01/U122/Y (INVX6)	0.16	3.92 r
RF01/U130/Y (OAI22X1)	0.17	4.09 f
RF01/ff5/Xin[9] (flip_flop_10_1)	0.00	4.09 f
RF01/ff5/U18/Y (OAI2BB2XL)	0.35	4.44 f
RF01/ff5/Xout_reg[9]/D (DFFRX1)	0.00	4.44 f
data arrival time		4.44
clock clk (rise edge)	4.30	4.30
clock network delay (ideal)	0.50	4.80
clock uncertainty	-0.10	4.70
RF01/ff5/Xout_reg[9]/CK (DFFRX1)	0.00	4.70 r
library setup time	-0.26	4.44
data required time		4.44
data required time		4.44
data arrival time		-4.44
slack (MET)		0.00

➢ 在電路中的位置如下圖：



➢ 根據 qor 檔的資料：

```
Timing Path Group 'clk'
-----------------------------------
Levels of Logic:                15.00
Critical Path Length:            3.94
Critical Path Slack:             0.00
Critical Path Clk Period:        4.30
Total Negative Slack:            0.00
No. of Violating Paths:          0.00
Worst Hold Violation:            0.00
Total Hold Violation:            0.00
No. of Hold Violations:          0.00
-----------------------------------
```

最長路徑的邏輯閘層數為 15 個邏輯閘。

- 下圖是功率以及最短路徑：

```
****************************************
Report : power
        -analysis_effort low
Design : geofence
Version: P-2019.03
Date   : Sun Jun 12 23:18:22 2022
****************************************


Library(s) Used:

    slow (File: /mnt3/CBDK_IC_Contest_v2.1/SynopsysDC/db/slow.db)


Operating Conditions: slow   Library: slow
Wire Load Model Mode: top

Design        Wire Load Model            Library
------------------------------------------------
geofence               tsmc13_wl10       slow


Global Operating Voltage = 1.08
Power-specific unit information :
    Voltage Units = 1V
    Capacitance Units = 1.000000pf
    Time Units = 1ns
    Dynamic Power Units = 1mW    (derived from V,C,T units)
    Leakage Power Units = 1pW


  Cell Internal Power  =   2.1159 mW   (73%)
  Net Switching Power  = 774.6110 uW   (27%)
                         ---------
Total Dynamic Power    =   2.8905 mW  (100%)

Cell Leakage Power     =  39.3185 uW


                 Internal         Switching           Leakage            Total
Power Group      Power            Power               Power              Power   (   %    )  Attrs
--------------------------------------------------------------------------------------------------
io_pad             0.0000           0.0000            0.0000            0.0000  (   0.00%)
memory             0.0000           0.0000            0.0000            0.0000  (   0.00%)
black_box          0.0000           0.0000            0.0000            0.0000  (   0.00%)
clock_network      0.0000           0.0000            0.0000            0.0000  (   0.00%)
register           1.8922        5.7390e-02        1.0423e+07            1.9600  (  66.90%)
sequential         0.0000           0.0000            0.0000            0.0000  (   0.00%)
combinational      0.2237           0.7172        2.8895e+07            0.9698  (  33.10%)
--------------------------------------------------------------------------------------------------
Total              2.1159 mW        0.7746 mW     3.9318e+07 pW         2.9298 mW

***** End Of Report *****
```

```
****************************************
Report : timing
         -path full
         -delay min
         -max_paths 1
         -sort_by group
Design : geofence
Version: P-2019.03
Date   : Sun Jun 12 23:21:03 2022
****************************************

Operating Conditions: slow   Library: slow
Wire Load Model Mode: top

  Startpoint: VC01/U1/U0/Dout_reg[7]
              (rising edge-triggered flip-flop clocked by clk)
  Endpoint: VC01/U1/U0/Dout_reg[8]
            (rising edge-triggered flip-flop clocked by clk)
  Path Group: clk
  Path Type: min

  Des/Clust/Port     Wire Load Model       Library
  ------------------------------------------------
  geofence           tsmc13_wl10           slow

  Point                                    Incr       Path
  -----------------------------------------------------------
  clock clk (rise edge)                    0.00       0.00
  clock network delay (ideal)              0.50       0.50
  VC01/U1/U0/Dout_reg[7]/CK (DFFRX2)       0.00       0.50 r
  VC01/U1/U0/Dout_reg[7]/QN (DFFRX2)       0.30       0.80 f
  VC01/U1/U0/U14/Y (OAI2BB2X4)             0.13       0.93 r
  VC01/U1/U0/Dout_reg[8]/D (DFFRHQX4)      0.00       0.93 r
  data arrival time                                   0.93

  clock clk (rise edge)                    0.00       0.00
  clock network delay (ideal)              0.50       0.50
  clock uncertainty                        0.10       0.60
  VC01/U1/U0/Dout_reg[8]/CK (DFFRHQX4)     0.00       0.60 r
  library hold time                       -0.02       0.58
  data required time                                  0.58
  -----------------------------------------------------------
  data required time                                  0.58
  data arrival time                                  -0.93
  -----------------------------------------------------------
  slack (MET)                                         0.35


***** End Of Report *****
```

五、邏輯合成與驗證結果(與比較)

A. 我們採用的版本

(1) 電路面積

```
****************************************
Report : area
Design : geofence
Version: P-2019.03
Date   : Sun Jun 12 15:34:18 2022
****************************************

Library(s) Used:

    slow (File: /mnt3/CBDK_IC_Contest_v2.1/SynopsysDC/db/slow.db)

Number of ports:                         1745
Number of nets:                          4129
Number of cells:                         2280
Number of combinational cells:           1936
Number of sequential cells:               301
Number of macros/black boxes:               0
Number of buf/inv:                        481
Number of references:                      11

Combinational area:              21894.762488
Buf/Inv area:                     3646.015198
Noncombinational area:           10012.962292
Macro/Black Box area:                0.000000
Net Interconnect area:          266188.009857

Total cell area:                 31907.724780
Total area:                     298095.734637

***** End Of Report *****
```

(2) 電路運算時間

```
--------------------------------------------------
--    Simulation finish,  ALL PASS              --
--------------------------------------------------
Simulation complete via $finish(1) at time 28628 NS + 0
./tb.sv:147             $finish;
ncsim> exit
soc08 [~/final_project/SYN/Geofence]
```

(3) 電路面積*總運算時間

電路面積：31907.72478 (um^2)

總運算時間：28628 (ns)

電路面積*總運算時間：31907.72478*28628 = 913,454,345.00184

B. 加減法器共用的版本

(1) 電路面積

```
****************************************
Report : area
Design : geofence
Version: P-2019.03
Date   : Sun Jun 12 23:16:36 2022
****************************************

Library(s) Used:

    slow (File: /mnt3/CBDK_IC_Contest_v2.1/SynopsysDC/db/slow.db)

Number of ports:                         2080
Number of nets:                          5475
Number of cells:                         3286
Number of combinational cells:           2914
Number of sequential cells:               327
Number of macros/black boxes:               0
Number of buf/inv:                        786
Number of references:                       7

Combinational area:              34918.912637
Buf/Inv area:                     6787.902582
Noncombinational area:           11375.974506
Macro/Black Box area:                0.000000
Net Interconnect area:          361575.154724

Total cell area:                 46294.887143
Total area:                     407870.041867

***** End Of Report *****
```

(2) 電路運算時間

```
----------------------------------------------------
--    Simulation finish,  ALL PASS                --
----------------------------------------------------
Simulation complete via $finish(1) at time 30990100 PS + 0
./tb.sv:147             $finish;
ncsim> exit
soc08 [~/final project/SYN/Geofence]
```

(3) 電路面積*總運算時間

電路面積：46294.887143 (um^2)

總運算時間：30990.1 (ns)

電路面積*總運算時間：46294.887143*30990.1 = 1,434,683,182.0502843

六、心得與討論

(1) 陳晉毅

問題與討論:

Q1. 為什麼合成後的電路面積比預期的大?

A1. 原本我對自己電路的面積沒有頭緒,經助教提示後發現軟體可以分別報出不同子模組的面積以及占比，後來發現我們有些預料之外的面積在於資料路徑控制的電路，也就是 MUX 等等的，估計占比有達 1/5 到 1/4，這是蠻大的面積，但這在大架構制定時就已經定了,當時在設計架構時沒有特別去考慮這一塊,因此根據這個經驗,未來在想架構的時候，也要把資料路徑控制電路的面積考慮進來，並且不能低估其面積的占比；另外還有一項是當在觀察電路面積時，也都要勾選 Area Report 中的 hierarchical，這樣可以更好的掌握各子模組間的面積占比。

Q2. 整合後的電路不會動

A2. 原本認為我們設計的電路架構在整合後不會出錯，但後來還是發現有問題，解決方法是到 nWave 查看各個訊號，起初不了解的時候只能把每個訊號都拉出來看，而後來發現這時的問題大多都出在各模組之間的溝通完整度，例如沒有遵照 Global FSM 的狀態等等的，因此有了這次的經驗後，往後在整合後的模擬出錯時，會先以模組之間的溝通下手來找問題。

心得:

我認為我在這次的專題內學到了很多東西，由演算法開始設計架構，到整合並優化電路，經歷了一次由自己的小組來設計電路的過程，其中我認為將演算法變成電路與整合電路的過程是比較吃力一點的,前者會需要考慮各種可以共用電路的情況,讓面積越小越好,並且要完整實現演算法內的所有功能;後者則是會因為電路間的溝通不完整而失敗，這時要叫出 nWave 一個一個將線路訊號抓出來看，這個過程我認為是要耗費耐心的。對於這次的專題，我們這組沒有特別去引用別組的算法與架構，其中一個原因是在於我們組的時間不夠，改演算法與架構會耗費許多時間，而另一個原因是我想知道如果只靠自己組的力量可以完成到什麼程度，雖然最後的效能感覺不到全班的前 50%，但我還是很滿意，因為在優化自己電路的過程中,我們試了很多種想法和方法,例如 Bubble Sort 的 Flag、pipeline、向量外積電路共用加減法器等等，雖然不是每個都真的讓電路變得更好，但重要的是我們透過實現了這些方法來了解到更多寫電路的技巧以及需要注意的事項。當然，在別組報告的同時我們也有去觀察別組為什麼效能比我們好，其中我發現當乘法器使用 combinational 的方式寫時，合成軟體會找出很棒的電路，並且可以兩個有號數相乘；又或是使用不同的排序法，例如 Merge Sort 或水平 bubble sort，在我們透過自己的方法來優化電路的同時，也了解到其他人設計電路的方法，所以我認為我在這次的期末專題學到了許多東西。

(2) 劉浩崴

問題與討論:

Q1. 乘法器的載入和完成訊號會因為暫存器的關係比預期的晚 1 個 clk 存取導致乘法器讀不到數值。

A1. 調整內部電路 FSM 轉態的條件，使完成的訊號提早結束，這樣載入訊號便可讓提前讀取要進入乘法器的數值。

Q2. 在共用加減法器時會因為借位的關係讓同時做兩組數值的運算時得出不正確的數值。

A2. 在兩組數值之間額外擴位，並將擴位的數值設為 1，這樣遇到借位時便不會影響到兩組數值運算的結果。

心得:

這次的 Geofence 期末專題電路中我是負責設計向量外積電路的部分，儘管 Geofence 整體電路的演算法已經出來,但向量外積這塊電路卻是最獨立也是最影響電路效能的部分，因此花了不少心思在設計電路架構上，而在架構中，我認為最成功的部份就是找出不用進入乘法器就可以得出外積符號的特例及路徑，雖然很吃 tb 提供的數據，但結果的確為電路的效能提升了不少。至於在合成電路的部分，令我感到意外的是，先前在架構中為了減少面積及延遲，所以裡面所有加法器和減法器都是自己一個一個刻出來的，但在合成時發現因為把內容寫得太細，反而導致 EDA 無法找出最適合電路架構的加法器和減法器，進而讓電路面積表現得不甚理想，結果最後在試著用 data flow 的方式撰寫後，這問題便改善了許多,這經驗也讓我認知到在設計一個大架構時,並不是所有部件都要靠自己寫出來，有時利用 EDA 工具反而能夠得出更適合電路的架構，而且效率也好上許多。而在這次期末專題中，花費最多時間且最讓我頭痛的就是整合和 debug 的部分，因為每個人撰寫的部分不同，所以儘管在測試自己電路時沒問題，整合起來後那些自以為不會出問題的地方仍然會跑出一堆奇奇怪怪的 bug 出來，為了找出電路的問題，我們藉著 nWave 把訊號一個一個抓出來看，在經過一連串不斷地修改及模擬後，終於通過了所有的數據，儘管 debug 過程很辛苦且煩躁，但在看到電腦螢幕上 ALL PASS 那幾個字後，整個人又好起來了。最後在電路優化的部分，因為在外積電路中使用最多的就是加法器和減法器，所以便把優化項目放在共用加減法器上，但在共用後合成軟體為了讓運作頻率壓低，反而讓電路的整體面積變大，在經過多次測試後效能還是比不過最一開始的架構，這大概是這次專題最讓我感到可惜的地方。總結這次的期末專題，我認為自己在設計架構和 coding 的基本功上仍然有許多可以改進和補足的地方，像是路徑規劃、時序上各個訊號及模組間的配合和一些能不浪費額外硬體空間的寫法等等,不過這次也是第一次能夠成功完成這個不小的電路設計還是蠻有成就感的,也很感謝各個隊友的 carry 讓專題能順利地完成,儘管過程不輕鬆，但還是獲得了許多寶貴的經驗及 ic 設計上的知識。

(3) 方士榮

問題與討論:

Q1. 在使用 spyglass 時，結果雖然沒有出現 error 或 warning，但卻一直出現 15 個 fatal，內容大致為 The re-declaration of a module with the same name is not allowed.

A1. 因為在 top module (Geofence.v) 中已有引入其他.v 檔，所以在 spyglass 中只需要加入 Geofence.v 就好。若把全部的.v 檔都加入則會出現 re-declaration

Q2. 打 7 對 1 多工器時，一直編譯錯誤。

A2. always block 中@後的括號必須將等號右邊的參數都寫進去，最簡單的方法就是養成習慣直接打一個*號，如此一來便能減少錯誤發生的機會。

心得:

一開始看到題目的時候，跟上次參加 IC 設計競賽的情況一樣，完全不知道如何實現，腦中出現的想法也比較偏軟體，因此，我們沒有想到其他比較特別的實現方法，就參考題目提供的方法，利用外積來實現這次期末專題的地理圍籬系統。至於排序演算法的部分，我們一開始有想到使用 bubble sort 和 quick sort 來比較，但後來發現 quick sort 占用的硬體面積太高了，最後決定用 bubble sort 來實現我們的排序。

經過這次的分組合作，我學到了非常多東西，像是如何和組員溝通，以及如何表達自己的看法，還有如何做好時間分配，因為還要考慮到其他科目期末考的時間。除此之外，在開始做期末專題以前，我做作業每次都是用手動的 testbench，但這次做專題時，我們組長有教我打自動化的 testbench，學會之後發現真的方便很多，光是這點我就覺得能和高手合作真的可以學到很多東西。在開始做期末專題以前總覺得設計電路是一件很難的事，但在這次的實作過程中，我發現，真正難的是驗證，我們的電路是否對於某些 corner case 無法順利運作，這部分其實也花了不少時間在討論。

在這次實作當中，我遇到讓我印象比較深刻的問題就是，我的 RTL code 編譯過了，但波型卻完全跑不出來，想了很久都找不到我的 testbench 哪裡出了問題，我把我的 testbench 丟給其他組長後，他的波型卻沒有問題，最後才發現，原來是我忘記把我 testbench 需要引用的資料放到工作站中，所以波型檔跑得出來，但卻沒有讀到任何資料，有了這次經驗後，我學到，養成良好的習慣可以減少很多不必要的麻煩。

在準備這次專題時，我們這組有遇到一些問題有去找助教討論，在討論過程中，也和助教學到很多關於 coding style 的部分，例如: always @後面的括號中可以直接用”*”星號取代一大串的參數，這樣的好處是可以避免因為少打一個參數而產生的錯誤。此外，乘法器和加法器的部分也不見得要全部自己打，因為以我們現在的程度很難打出比合成軟體找到的乘法器還要更好的架構，要自己打出一個更好的乘法器真的會花很多時間。

七、參考資料

參考題目講義、課堂講義、謝東佑老師上課內容、

邱日清老師上課內容、助教解惑