

2018 IC Design Contest

Cell-Based IC Design Category for Graduate Level

RF Indoor Localization Engine

1.問題描述

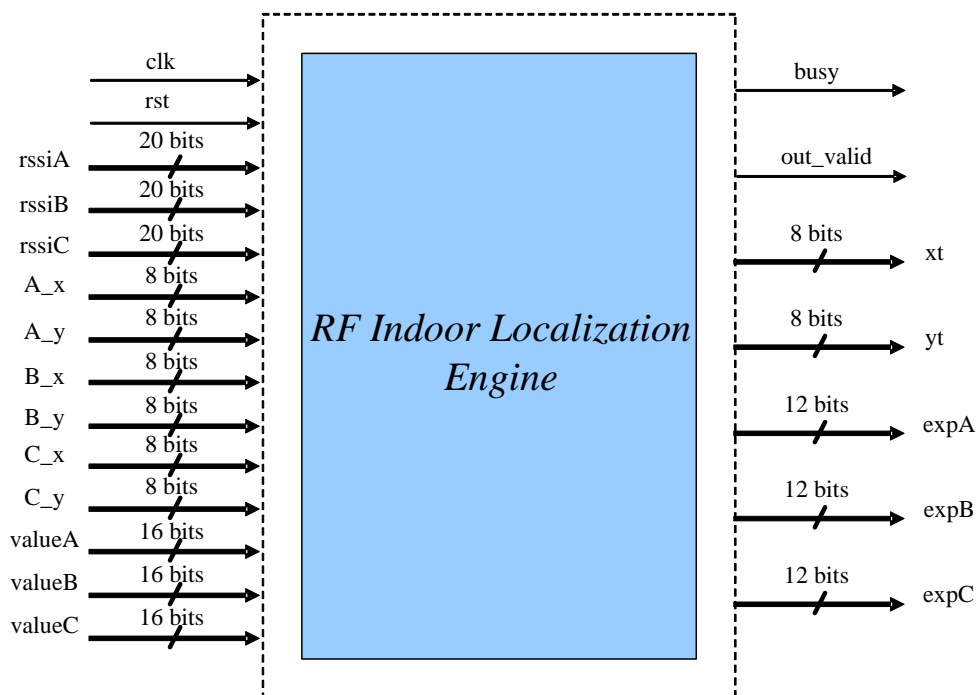
請完成一 RF Indoor Localization Engine(後文以 **RFILE** 表示)的電路設計。本電路可依據固定安裝於室內 A、B、C 三點的 RF Sensor 輸出之 RSSIA、RSSIB、RSSIC 訊號強度來定位，將人在室內空間的實際位置 (後文以 **T 點** 表示)，以 x_t 、 y_t 座標輸出，即為所求!有關 RFILE 詳細規格將描述於後。表一為本電路各輸入輸出信號的功能說明。各參賽隊伍必須依照下一節所指定的設計規格及附錄 A 中的測試樣本完成設計驗證。

本次 IC 設計競賽比賽時間為上午 08:30 到下午 20:30。當 IC 設計競賽結束後，CIC 會根據第三節中的評分標準進行評分。為了評分作業的方便，各參賽隊伍應參考附錄 E 中所列的要求，附上評分所需要的檔案。

本題目之測試樣本置於 [/usr/cad/icc2018/bgc/icc2018cb.tar](#)，請執行以下指令取得測試樣本：

```
tar xvf /usr/cad/icc2018/bgc/icc2018cb.tar
```

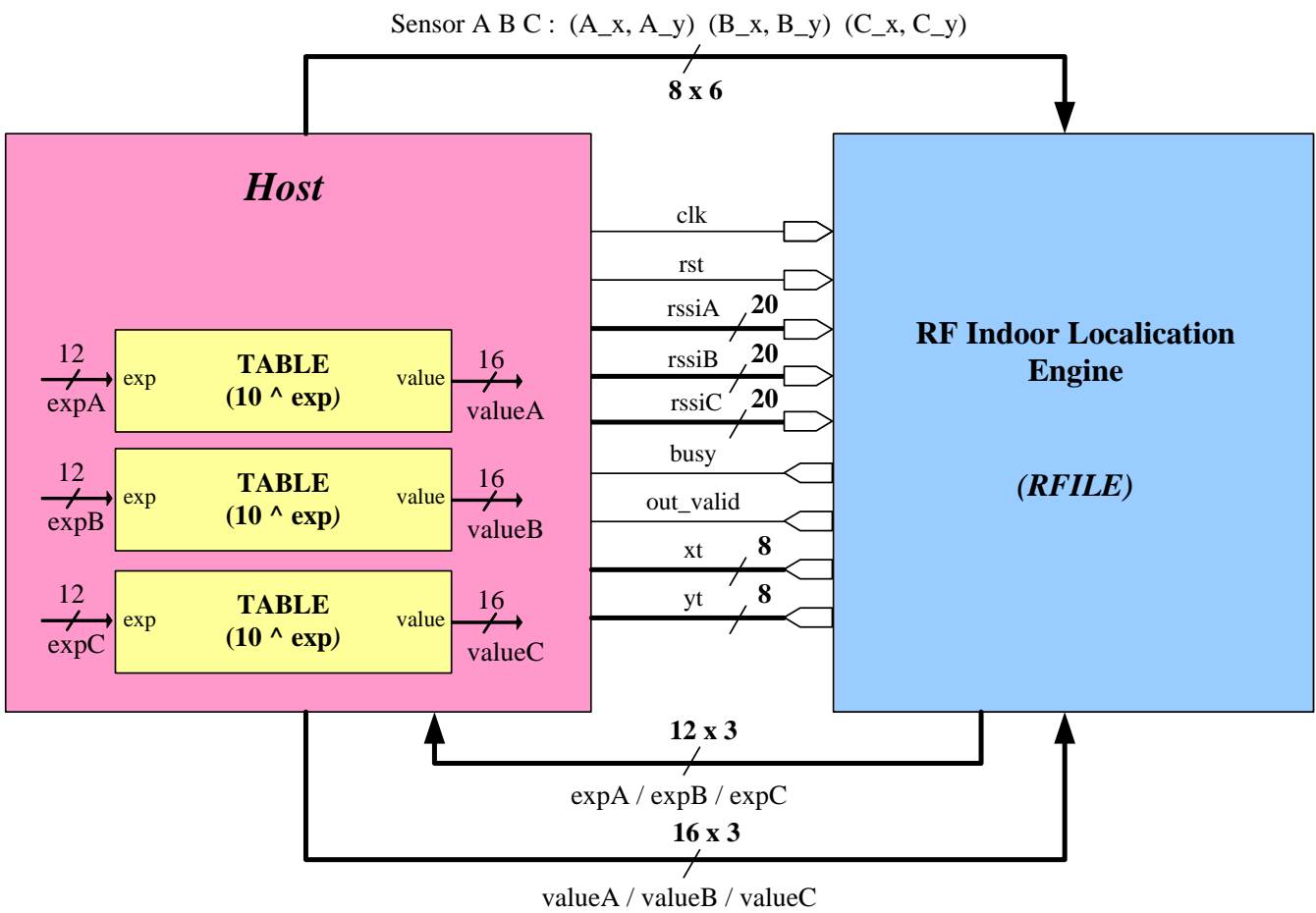
軟體環境及設計資料庫說明請參考附錄 F 與附錄 G。



圖一、RF Indoor Localization Engine 之方塊圖

2.設計規格

2.1 系統方塊圖



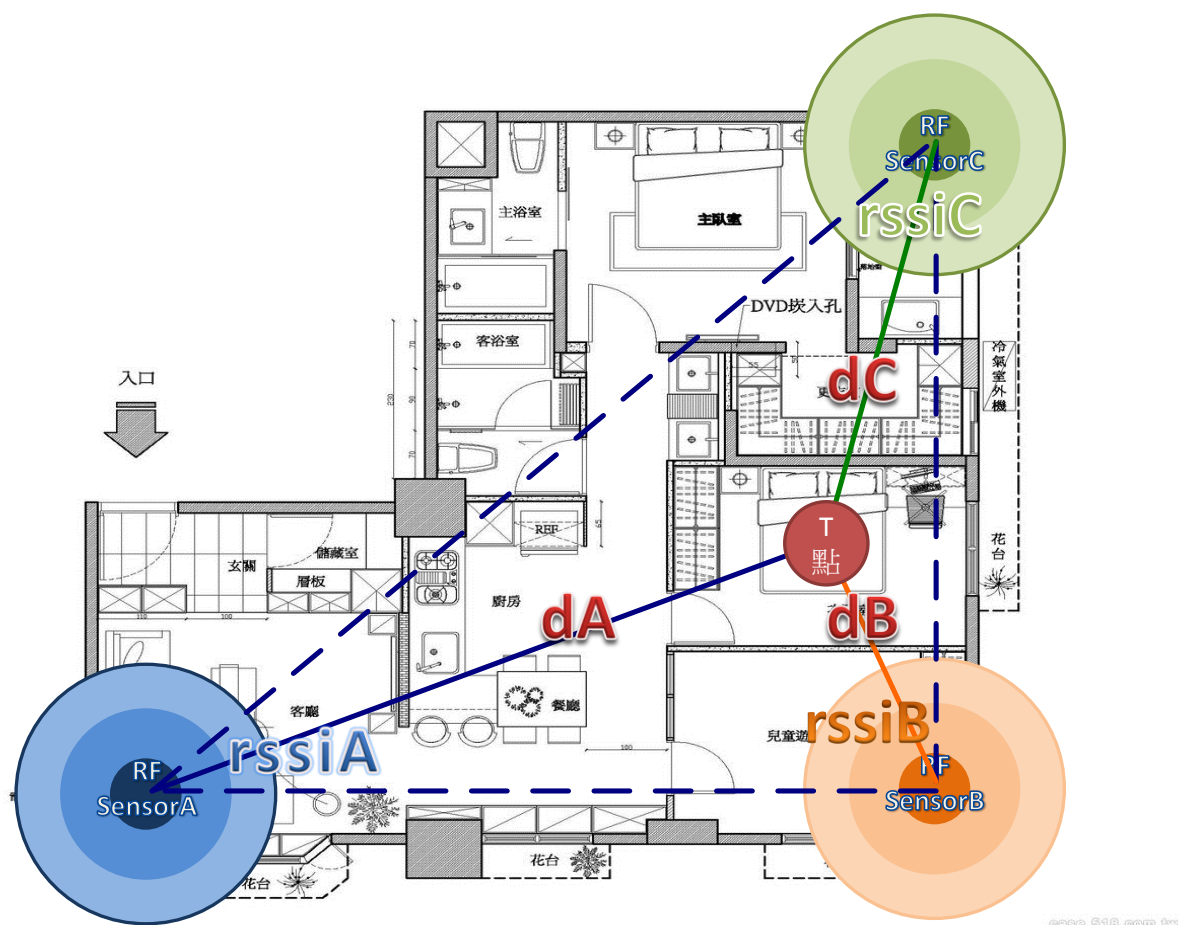
圖二、系統方塊圖

2.2 輸入/輸出介面

表 1 -輸入/輸出訊號

Signal Name	I/O	Width	Simple Description
clk	I	1	本系統為同步於時脈正緣之同步設計。
rst	I	1	高位準非同步(active high asynchronous)之系統重置信號。
rssiA	I	20	將 RF Sensor A 偵測的 RSSI 訊號值輸入至 RFILE 電路，其整數部分為 8bits，小數部分為 12bits。（該值必為負值，取 2'sC 表示之）
rssiB	I	20	將 RF Sensor B 偵測的 RSSI 訊號值輸入至 RFILE 電路，其整數部分為 8bits，小數部分為 12bits。（該值必為負值，取 2'sC 表示之）

rssIC	I	20	將 RF Sensor C 偵測的 RSSI 訊號值輸入至 RFILE 電路，其整數部分為 8bits，小數部分為 12bits。（該值必為負值，取 2'sC 表示之）
A_x	I	8	RF Sensor A 架設在室內的座標點，其 x 軸的值。在同一個測試樣本，該值為 固定值，且為正整數 。
A_y	I	8	RF Sensor A 架設在室內的座標點，其 y 軸的值。在同一個測試樣本，該值為 固定值，且為正整數 。
B_x	I	8	RF Sensor B 架設在室內的座標點，其 x 軸的值。在同一個測試樣本，該值為 固定值，且為正整數 。
B_y	I	8	RF Sensor B 架設在室內的座標點，其 y 軸的值。在同一個測試樣本，該值為 固定值，且為正整數 。
C_x	I	8	RF Sensor C 架設在室內的座標點，其 x 軸的值。在同一個測試樣本，該值為 固定值，且為正整數 。
C_y	I	8	RF Sensor C 架設在室內的座標點，其 y 軸的值。在同一個測試樣本，該值為 固定值，且為正整數 。
busy	O	1	RFILE 忙碌之控制訊號。當 busy 為 Low 時，Host 端會輸入一筆新的 rssIA、rssIB、rssIC 的訊號值，反之，當 為 High 時，表示系統正處於忙碌階段，此時 Host 端提供的 rssIA、rssIB、rssIC 的資料保持不變。
out_valid	O	1	RFILE 有效輸出資料之通知訊號。當為 High，表示目前輸出的 x_t 、 y_t 座標值為有效的輸出，反之則輸出為 Low。
xt	O	8	偵測到 T 點的 x 軸座標值輸出。 註：該值 必為正整數 。
yt	O	8	偵測到 T 點的 y 軸座標值輸出。 註：該值 必為正整數 。
expA expB expC	O	12	使用”10 的次方運算查表電路”之 待查詢 的輸入值。該查表前輸入的數值，以 小數位數 12bits ，從 RFILE 電路輸出至 Host 端的查表電路，本題 查表電路所耗時間為 0ns 。 註：詳細運算方法請參照 2.3.2。
valueA valueB valueC	I	16	使用”10 的次方運算查表電路”之 查詢後 的輸出值。該查表後獲得的數值，以 整數位數 4bits ， 小數位數 12bits 的結構，輸入至 RFILE 電路，本題 查表電路所耗時間為 0ns 。註：詳細運算方法請參照 2.3.2。



圖三、室內定位系統示意圖

2.3 系統描述

圖三為室內定位系統示意圖，裝有多組 RF Sensor 用來定位人類位於室內空間的實際方位，例如冷氣的風可集中吹在 T 點有人的區域，以達到節電之效用。當然 RF Sensor 裝越多組可定位的空間也就越廣，但考量到複雜度問題，**本題規定 RF Sensor 僅固定 A、B、C 三組**，偵測的 T 點只會在三角型區域移位，**T 點不會超出三角形之外，也不會在三角形的線上**。

RF Sensor 係透過接收訊號強度測量法(Received Signal Strength Indicator – RSSI)的數值大小來計算發射端和接收端間的距離，**RSSI 必為負值**，其絕對值的數字越大表示 Sensor 到 T 點的距離越遠，反之其絕對值的數字越小表示 Sensor 到 T 點的距離越近，例如：**-90 與 -70 相比，-90 距離較遠，-70 距離較近**。使用 RSSI 值來計算距離遠近之公式如(1)式所示

$$d = 10^{((\text{abs}(\text{RSSI}) - \alpha) / (10 * n))} \quad (1)$$

註：

d - 計算所得距離 ((1)式中 ^ 表示為 10 的次方)

RSSI - 接收信號強度 (必為負值，**abs** 表示對 RSSI 取絕對值)

α - 發射端和接收端相隔 1 米時的信號強度 (**本題規定 α 固定為 59**)

n - 環境衰減因子 (**本題規定 n 固定為 2**)

因此 RFILE 電路可根據 A、B、C 三個 Sensor 的 RSSI 數值，在此以 rssiA、rssiB、rssiC 表示，將 RSSI 值套入(1)式，分別計算出 d_A 、 d_B 、 d_C ，接著帶入距離公式如(2)式，即可計算出 T 點之座標，將 T 點之座標 x_t 、 y_t 輸出，即為所求。

$$f(x_t, y_t) = \begin{cases} (x_A - x_t)^2 + (y_A - y_t)^2 = d_A^2 \\ (x_B - x_t)^2 + (y_B - y_t)^2 = d_B^2 \\ (x_C - x_t)^2 + (y_C - y_t)^2 \leq d_C^2 \end{cases} \quad (2)$$

註 1：

Sensor A 架設於室內座標 – 以(x_A 、 y_A)表示，該值為**已知的固定值**。

Sensor B 架設於室內座標 – 以(x_B 、 y_B)表示，該值為**已知的固定值**。

Sensor C 架設於室內座標 – 以(x_C 、 y_C)表示，該值為**已知的固定值**。

T 點會隨時變動，其座標 – 以(x_t 、 y_t)表示，該值本題規定**必為正整數**。

註 2：

由於 RF 本身就不夠準確， d_C^2 在此容許自行乘上一個倍率例如 1.01，即 $(1.01*d_C)^2$ 。

2.3.1 RFILE 電路的輸入

RFILE 電路輸入主要有兩個項目：

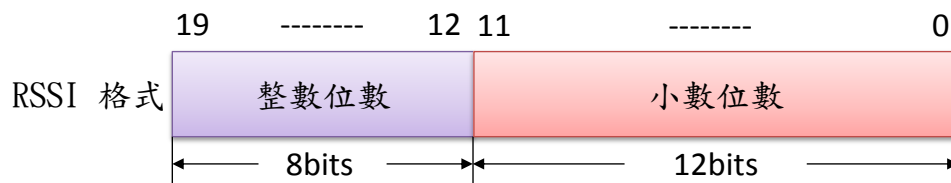
A. RF Sensor A、B、C 三個固定點架設的座標值

在同一個測試樣本中，**A、B、C 三點的座標值**，分別以(A_x , A_y)、(B_x , B_y)、(C_x , C_y)這六組腳位做輸入，從模擬的開始至最後，該值**永遠為固定值且為正整數**，實際座標值，詳見如附錄 B。

註：A、B、C 三點座標的排列組合，**未必是直角三角形**，鈍角或銳角三角形皆有可能。

B. RSSI 訊號格式

A、B、C 三個 Sensor 的 RSSI 數值，分別以 rssiA、rssiB、rssiC 輸入。由於考量該數值的小數部分對偵測 T 點實際座標影響甚遠，在此定義 RSSI 數值格式如下圖：



由於 RSSI 數值必為負值，因此 rssiA、rssiB、rssiC 收到的數值，要以 2'sC 解讀，範例如下：

rssIA 收到	A2FA4
取2' sC	<u>5D</u> <u>05C</u>
整數位數	$(93)_{10}$
小數位數	$(05C)_{16} / 4096 \Rightarrow 0.0224609375$
十進制解讀	-93.0224609375

2.3.2 RFILE 電路的運算方法

RFILE 電路運算方法詳見(1)、(2)式，在此不再贅述。不過(1)式運算過程中，會使用到 10 的次方運算，**本題有提供 10 的幕次方查表**，已定義在 table.v，輸入(expA/expB/expC)為 12bits 小數，輸出(valueA/ valueB/valueC)為 4bits 整數加 12bits 小數，其計算方法，舉例說明如下：

次方範例	$10 ^ (3.4022)$	
等於	$(10^3) \times (10^{\underline{0.4022}})$	
小數 (精確度12bits)	$0.4022 \times 4096 = (1647)_{10}$	
1647 查表 (table.v)	$(2862)_{16} = (10338)_{10}$	請自行開啟table.v檔
$10^{\underline{0.4022}}$ 相當於	$10338 / 4096 = 2.524$	
expA 送出查表	$(66F)_{16}$	所以expA需要12bits
valueA 查表獲得	$(2862)_{16}$	所以valueA需要16bits
因此	$(10^3) \times (10^{\underline{0.4022}})$ $= 1000 \times 2.524 = 2524$	使用查表法順利算出

註 1：table.v 查表電路已製作於 TestBench 當中，參賽者無需 Coding 查表電路。

註 2：table.v 查表電路**運算時間 0ns**。

註 3：使用 table.v 查表電路前，**請自行先將指數取出小數部份，再進行查表，整數部分自行計算**。

2.3.3 RFILE 電路的輸出

當參賽者使用(1)、(2)式定位出 T 點座標後，此時將 out_valid 訊號拉為 High，並且以 x_t 、 y_t 輸出 T 點座標值，即完成 RFILE 電路功能。由於考量運算過程中的量化誤差與 RF 本身就不夠準確，T 點標準答案與參賽者使用 RFILE 電路定位出的 x_t 、 y_t ，允許低於 10 的誤差值都算是正確，其算法如(3)式所示，因此在模擬時每一組 Pattern 都會秀出個別 difference 數值，只要低於 10，該組 Pattern 就算是 PASS，如圖四所示。

$$((x_{\text{golden}} - x_t)^2 + (y_{\text{golden}} - y_t)^2) < 10 \quad (3)$$

註 1：T 點的標準答案以 x_{golden} 、 y_{golden} 表示。

註 2： x_t 、 y_t 必為正整數。

```
ncsim> run
FSDB Dumper for IUS, Release Verdi3_L-2016.06-SP1-1, Linux, 09/27/2016
(C) 1996 - 2016 by Synopsys, Inc.
*Verdi3* FSDB WARNING: The FSDB file already exists. Overwriting the FSDB file may
*Verdi3* : Create FSDB file 'RFILE.fsdb'
*Verdi3* : Begin traversing the scopes, layer (0).
*Verdi3* : End of traversing.
*Verdi3* : Begin traversing the MDAs, layer (0).
*Verdi3* : Enable +mda and +packedmda dumping.
*Verdi3* : End of traversing the MDAs.
-----
Start to Send RSSI & Compare ...

T00: your xt=34 yt=07 == expect xt=34 yt=07 => difference= 0 => PASS
T01: your xt=22 yt=11 == expect xt=22 yt=11 => difference= 0 => PASS
T02: your xt=34 yt=1b == expect xt=34 yt=1b => difference= 0 => PASS
T03: your xt=22 yt=0f == expect xt=22 yt=0f => difference= 0 => PASS
T04: your xt=2e yt=2d == expect xt=2e yt=2d => difference= 0 => PASS
T05: your xt=37 yt=15 == expect xt=37 yt=15 => difference= 0 => PASS
      :
      :
T91: your xt=37 yt=21 == expect xt=37 yt=21 => difference= 0 => PASS
T92: your xt=16 yt=0f == expect xt=16 yt=0f => difference= 0 => PASS
T93: your xt=31 yt=17 == expect xt=31 yt=17 => difference= 0 => PASS
T94: your xt=2e yt=19 == expect xt=2e yt=19 => difference= 0 => PASS
T95: your xt=16 yt=13 == expect xt=16 yt=13 => difference= 0 => PASS
T96: your xt=19 yt=11 == expect xt=19 yt=11 => difference= 0 => PASS
T97: your xt=37 yt=2b == expect xt=37 yt=2b => difference= 0 => PASS
T98: your xt=2b yt=09 == expect xt=2b yt=09 => difference= 0 => PASS
T99: your xt=37 yt=2f == expect xt=37 yt=2f => difference= 0 => PASS

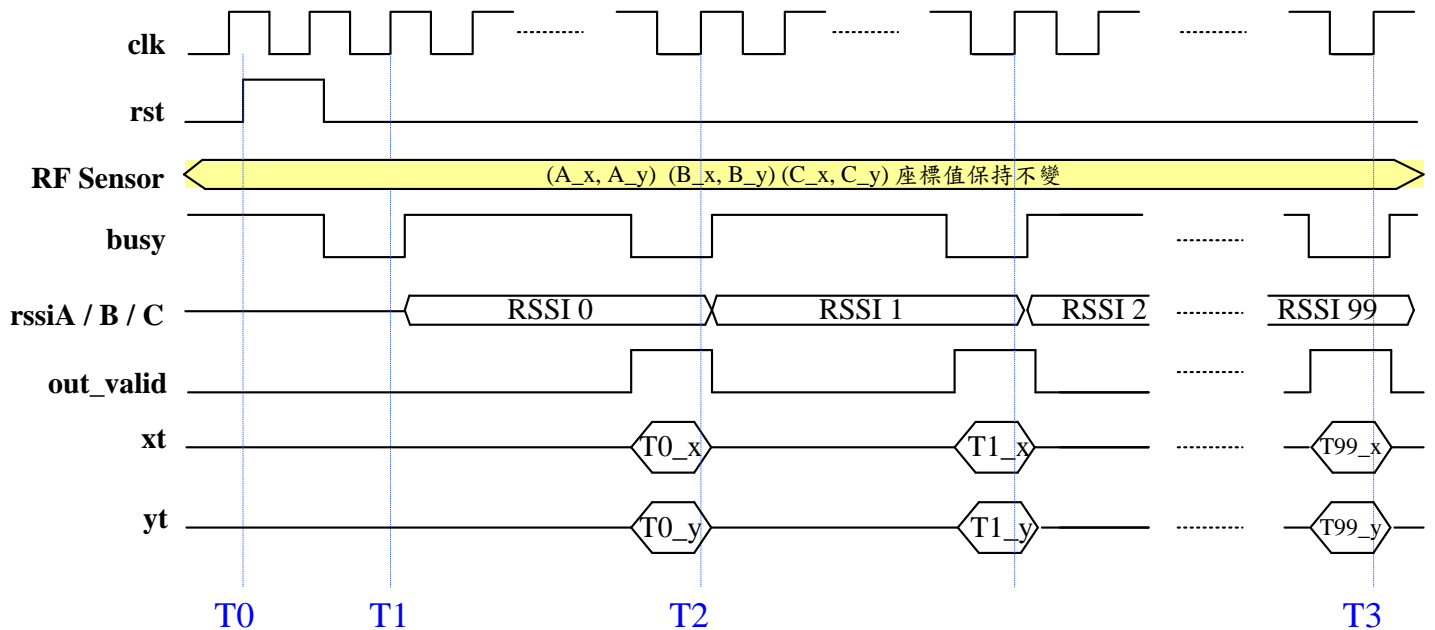
Send RSSI & Compare Over!
-----

Congratulations! All data have been generated successfully!
-----PASS-----

Simulation complete via $finish(1) at time 49992 NS + 0
./testfixture.v:183      #(`CYCLE/2); $finish;
ncsim> exit
```

圖四、RFILE 電路模擬範例

2.4 RFILE 電路時序規格



圖五、RFILE 電路時序圖

RFILE 電路時序圖說明如下：

1. T0 ~ T1 時間點，RFILE 電路初始化。
2. T1 時間點，Host 端判斷 busy 訊號為 Low，隨即送出第 0 筆 RSSI 訊號。
3. T1 ~ T2 時間點，RFILE 電路運算過程中，rssiA、rssiB、rssiC 輸入的第 0 筆 RSSI 訊號會一直維持住。
4. T2 時間點，RFILE 電路定位出 T 點所在處後，此時發送 out_valid 訊號為 High，用以告知 Host 端讀取第 0 筆 T 點 x_t 、 y_t 座標值，此時 busy 訊號也可以在此時間點設為 Low，以告知 Host 端準備輸入下一筆 RSSI 訊號，即 RSSI 1。
5. 重覆 T1 ~ T2 動作之行為。
6. T3 時間點，最後一筆 T 點被定位出，此時發送最後一次 out_valid 訊號為 High，用以告知 Host 端讀取第 99 筆 T 點 x_t 、 y_t 座標值後，電路運作模擬結束。

註：圖五中 RF Sensor 係指 A、B、C 三組 Sensor，在本次的測試樣本中，給定的 (A_x, A_y) 、 (B_x, B_y) 、 (C_x, C_y) 為何，在整個模擬期間，這些座標值恆為固定值。

註：圖五未提供 exp、value 之時序圖，原因是 exp A、B、C 輸出到 Host 端查表電路後，以 value A、B、C 回傳查表結果，這段期間的延遲時間為 0ns。

3. 評分標準

評分方式會依設計完成程度，分成 A、B、C、D 四種等級，排名順序為 A>B>C>D，評分項目有兩個，分別為**模擬時間**、**功率消耗**，主辦單位會依此兩項目做為同等級之評分。另外，**請參賽者提供一組正確的週期時間(CYCLE TIME)**給評分人員驗證本電路之正確性。

◇ 評分項目一：依”模擬時間”(Time)長短評分

各參賽隊伍將 APR 完成後，執行 Gate-level Post-layout Simulation 模擬完後，會出現模擬時間，評分人員會以此模擬時間如圖四範例，紀錄成 **Time =49992ns** 做評分。

註：三組測試樣本若模擬時間不同，Simulation Time 以最大值為準。

◇ 評分項目二：依”功率消耗”(Power)大小評分

各參賽隊伍將 APR 完成後，請用 PrimeTime-PX (後文以 **PT-PX** 表示)作 Power 分析，本題主辦單位已提供 PT-PX 的自動化執行程式：**pt_script.tcl**，參賽者只要將 pt_script.tcl 檔案會使用到的相關檔案(即 **RFILE_pr.v**、**RFILE_APR.sdc**、**RFILE.fsdb**、**.synopsys_pt.setup**)準備好，執行下述指令即可作 Power 分析，執行後之 log 訊息如下所示，**功率消耗請自行轉換單位為毫瓦(mW)**，以此例讀作 1.275mW，紀錄成 **Power=1.275** 作評分。

```
unix% pt_shell -f ./pt_script.tcl
```

Power Group	Internal Power	Switching Power	Leakage Power	Total Power	(%)	Attrs
clock_network	8.441e-04	7.640e-05	1.104e-07	9.206e-04	(72.19%)	i
register	5.459e-05	1.344e-05	1.258e-05	8.060e-05	(6.32%)	
combinational	1.743e-04	6.189e-05	3.793e-05	2.741e-04	(21.49%)	
sequential	0.0000	0.0000	0.0000	0.0000	(0.00%)	
memory	0.0000	0.0000	0.0000	0.0000	(0.00%)	
io_pad	0.0000	0.0000	0.0000	0.0000	(0.00%)	
black_box	0.0000	0.0000	0.0000	0.0000	(0.00%)	

Net Switching Power	= 1.517e-04	(11.90%)				
Cell Internal Power	= 1.073e-03	(84.13%)				
Cell Leakage Power	= 5.062e-05	(3.97%)				

Total Power	= 1.275e-03	(100.00%)				

X Transition Power	= 2.347e-09					
Glitching Power	= 8.350e-08					

Peak Power	= 0.0530					
Peak Time	= 113590.281					

註：三組測試樣本若 Power 值不同，Power 以最大值為準。

註：未完成 APR 者，請以合成後結果作為等級 C 之評分依據。

設計完成程度四種等級，如下：

☆ **等級 A：** 達成”完成設計”之三項要求

- a、 功能正確，RTL 模擬與正確解答比對完全正確。
- b、 完成 Synthesis，且 Gate-Level Pre-layout Simulation 結果正確。
- c、 **完成 APR，並達成 APR 必要項目**，Gate-Level Post-layout Simulation 結果正確。

註：完成 APR 之必要項目

- i. 只需做 Marco layout (即不用包含 IO Pad、Bonding Pad)。
- ii. VDD 與 VSS Power Ring 寬度請各設定為 **2um**，**只須做一組**。
- iii. **不需加 Dummy Metal**。
- iv. **Power Stripe 務必至少加一組**，其 VDD、VSS 寬度各設定為 **2um**。
(Power Stripe 垂直方向至少一組，水平方向可不加)
- v. **務必要加 Power Rail (follow pin)**。
- vi. **Core Filler 務必要加**。
- vii. APR 後之 GDSII 檔案務必產生。
- viii. 完成 APR，DRC/LVS **完全無誤(見附錄 C 說明)**。

等級 A 之評分方法：

$$\text{Score} = \text{Power} \times \text{Time}$$

例如：

$$\text{Score} = \text{Power} \times \text{Time} = 1.275 \times 49992 = 63739.8$$

註：Score 越小者，同級名次越好！

☆ **等級 B：** 已做到 APR，但等級 A 之”APR 必要項目”有部分不符合，DRC/LVS 錯誤總數量容許 **5 個(含)以下**

此等級之成績計算方式如下：

$$\text{Score} = \text{Power} \times \text{Time} \times (\text{DRC} + \text{LVS 的錯誤總數量})$$

註：Score 越小者，同級名次越好！

- ☆ **等級 C**：僅完成合成，或做到 APR，但 DRC/LVS 錯誤總數量**超過 5 個以上**
此等級之成績計算方式如下：

$$\text{Score} = \text{Power} \times \text{Time}$$

註：

1. Score 越小者，同級名次越好！
2. 等級 C，視 APR 為 Fail，Time 以 Gate-level Pre-layout Simulation 為主。
3. 等級 C，視 APR 為 Fail，Power 以合成後的相關檔案使用 PT-PX 作 Power 分析。

- ☆ **等級 D**：未達成前三等級者，成績計算方式為 All RTL Simulation，比對結果之 error 總數量
越少者，分數越高。

$$\text{Score} = \text{Total error of All RTL Simulations}$$

註：

1. 等級 D，Score 評分方式為所有模擬的 error 總數作相加。
2. 等級 D，視合成與 APR 皆為 Fail，Power、Time 將不予考慮。
3. 等級 D，只以 RTL Simulation 正確率為主，Score 越小者(即 error 越少)，同級名次越好。

附錄

附錄 A 為主辦單位所提供各參賽者的設計檔案說明；附錄 B 為主辦單位提供的測試樣本說明；附錄 C 為設計驗證說明；附錄 D 為評分用檔案，亦即參賽者必須繳交的檔案資料；附錄 E 則為設計檔案壓縮整理步驟說明；附錄 F 中說明本次競賽之軟體環境；附錄 G 中說明本次競賽使用之設計資料庫；附錄 H 一元二次方程式求根。

附錄 A 設計檔

1. 下表為主辦單位所提供各參賽者的設計檔

表 2、設計檔案說明

檔名	說明
RFILE.v	本題之設計檔，已包含系統 Input/Output Port 之宣告，請以此檔案作為 RFILE 電路之設計。
table.v	10 的次方運算查表電路(僅針對小數部份作查表)，該 table 已在 TestBench 加入。注意：模擬時，請務必在目前目錄下，準備 table.v 檔案，方可順利模擬。
testfixture.v	本題僅有一個 TestBench，卻有三種測試樣本需模擬，因此請在模擬期間，自行使用+define+PAT1、+define+PAT2、+define+PAT3 參數，作三種測試樣本的切換。
pattern1.dat pattern2.dat pattern3.dat	作為 RFILE 電路模擬時，三種測試樣本的輸入訊號。註：這些檔案已加入至 TestBench，無需額外設定。
golden1.dat golden2.dat golden3.dat	作為 RFILE 電路模擬時，三種測試樣本的輸出比對。註：這些檔案已加入至 TestBench，無需額外設定。
RFILE_DC.sdc	Design Compiler 作合成之 Constraint 檔案，請自行設定 period 的期望值，但環境相關參數請勿更改。
RFILE_APR.sdc	Encounter、IC Compiler 作 APR 之 Constraint 檔案，請自行設定 period 的期望值，但環境相關參數請勿更改。

.synopsys_dc.setup	使用 Design Compiler 作合成或 IC Compiler Layout 之初始化設定檔。參賽者請依 Library 實際擺放位置，自行修改 Search Path 的設定。 註：無論合成或 APR，只需使用 worst case library。
.synopsys_pt.setup	使用 PT-PX 作 Power 量測之初始化設定檔。參賽者請依 Library 實際擺放位置，自行修改 Search Path 的設定。 註：作 Power 量測，只需使用 worst case library。
pt_script.tcl	使用 PT-PX 作 Power 量測之自動化執行 Script，參賽者使用前，請自行將會用到的相關檔案擺在目前目錄下，方可正常執行。

2. 本次比賽中，若沒有任何舞弊之行為，主辦單位便不會有隱藏的 Pattern 作測試。

3. 使用 Encounter 作 APR 請注意，模擬時請務必自行加上+ncmaxdelays 參數。

例如：

```
> ncverilog +ncmaxdelays testfixture.v RFILE_pr.v tsmc13_neg.v +define+PAT1+SDF
+access+r
```

註：本次比賽中有三種測試樣本，請自行使用+define+PAT1、+define+PAT2、+define+PAT3 參數，三選一作模擬。

附錄 B 測試樣本

本題有三種測試樣本，RF Sensor A、B、C 座標，已設定於 TestBench 中，如下。在 testfixture.v 檔案之 31 ~ 52 行，已清楚記載 RF Sensor 架設之座標值。

```

29 RFILE u_RFILE( .clk      (clk      ),
30                .rst      (rst      ),
31 `ifdef PAT2
32     .A_x        (8'd5      ),
33     .A_y        (8'd5      ),
34     .B_x        (8'd100    ),
35     .B_y        (8'd20     ),
36     .C_x        (8'd50     ),
37     .C_y        (8'd100    ),
38 `elsif PAT3
39     .A_x        (8'd5      ),
40     .A_y        (8'd21     ),
41     .B_x        (8'd80     ),
42     .B_y        (8'd91     ),
43     .C_x        (8'd95     ),
44     .C_y        (8'd1      ),
45 `else //PAT1
46     .A_x        (8'd2      ),
47     .A_y        (8'd2      ),
48     .B_x        (8'd62     ),
49     .B_y        (8'd2      ),
50     .C_x        (8'd62     ),
51     .C_y        (8'd62     ),
52 `endif

```

RFILE 電路的另一個重要輸入訊號 – RSSI，其內容如下，TestBench 在每單一週期自動讀取三筆資料，由 rssiA、rssiB、rssiC 三組輸入腳位作輸入。

```
A2FA4      //Pattern000: RSSIA=-93.0225
B007F      //Pattern000: RSSIB=-79.969
A20D3      //Pattern000: RSSIC=-93.9485
A608D      //Pattern001: RSSIA=-89.9656
A6F61      //Pattern001: RSSIB=-89.0388
A283C      //Pattern001: RSSIC=-93.4854
A20D3      //Pattern002: RSSIA=-93.9485
A8659      //Pattern002: RSSIB=-87.6033
A5C73      //Pattern002: RSSIC=-90.2219
A63BD      //Pattern003: RSSIA=-89.7664
A7359      //Pattern003: RSSIB=-88.7908
A23D3      //Pattern003: RSSIC=-93.761
...

```

pattern1.dat

註：資料左側為十六進制，資料右側以註解方式說明該筆 Pattern 為哪個 T 點偵測到的 RSSI 數值，該數值以十進制表示，以方便參賽者作 Debug。

註：在此只秀出第一組測試樣本 RSSI 數值，其他組測試樣本，請詳見 pattern2.dat、pattern3.dat。

使用 RFILE 電路定位出 T 點座標之標準答案，內容如下：

```
34      //T000: Xt=052
07      //T000: Yt=007
22      //T001: Xt=034
11      //T001: Yt=017
34      //T002: Xt=052
1B      //T002: Yt=027
22      //T003: Xt=034
0F      //T003: Yt=015
...

```

golden1.dat

註：資料左側為十六進制，資料右側以註解方式說明該筆資料為第幾筆 T 點的標準解答，在此以十進制表示，以方便參賽者作 Debug。

註：在此只秀出第一組測試樣本 T 點的標準解答，其他組請自行參閱 golden2.dat、golden3.dat。

附錄 C 設計驗證說明

參賽者繳交資料前應完成 RTL，Gate-Level 與 Physical 三種階段驗證，以確保設計正確性。**注意：**每組限定只能使用 1 license，勿使用 Multi-CPU。

- RTL 與 Gate-Level 階段：參賽者必須進行 RTL simulation 及 Gate-Level simulation，模擬結果必須滿足本題指定之 Period 下，功能完全正確。
- Physical 階段，包含三項驗證重點：
 1. 依主辦單位各項要求，實現完整且正確的 layout (詳細之各項要求，請見評分標準)。
 2. 完成 post-layout simulation：參賽者必須使用 **P&R 軟體寫出之 Netlist 檔與 SDF 檔** 完成 post-layout gate-level simulation，以下分為 IC Compiler、Encounter 兩種軟體說明 netlist 與 sdf 寫出步驟。

- i. 使用 Synopsys IC Compiler 者，執行步驟如下：

在 IC Compiler 主視窗底下點選

“ File > Export > Write SDF...”

Specify Version	Version 2.1
Instance	空白即可
File name	RFILE_pr.sdf
Significant digits	2

按 **OK**。

對應指令： write_sdf -version 2.1 RFILE_pr.sdf

“ File > Export > Write Verilog...”

先按 **Default**

Output verilog file name	RFILE_pr.v
Output physical only cells	disable
Wire declaration	enable
Backslash before Hierarchy Separator	Enable
All other options	Default value

按 **OK**。

- ii. 使用 Cadence Encounter 者，執行步驟如下：

在 Encounter 視窗下點選：

“ File → Save → Netlist...”

Netlist File	RFILE_pr.v
All other options	Default value

按 **OK**。

“ Timing → Extract RC...”，按 **OK**。

“Timing → Write SDF...”

Ideal Clock	Disable
SDF Output File:	RFILE_pr.sdf

按 **OK**。

3. 完成 DRC 與 LVS 驗證：參賽者必須以其所使用之 **P&R 軟體內含之 DRC 與 LVS 驗證功能完成 DRC 與 LVS 驗證**，以下分為 IC Compiler、Encounter 兩種軟體說明執行步驟。

- i. 使用 Synopsys IC Compiler 者，驗證 DRC 與 LVS 步驟如下：

在 IC Compiler Layout 視窗底下點選

“Route > Verification > DRC ...”

Read child cell from	Cell view
All other options	Default value

按 **OK**。

將跳出 Error Browser 視窗，請參賽者自行查看是否有錯，若有請自行修改 Layout 到 0 個 Violation 為止。

“Route > Verification > LVS ...”

Pins not connected to a wire segment(Floating port)	disable
All other options	Default value

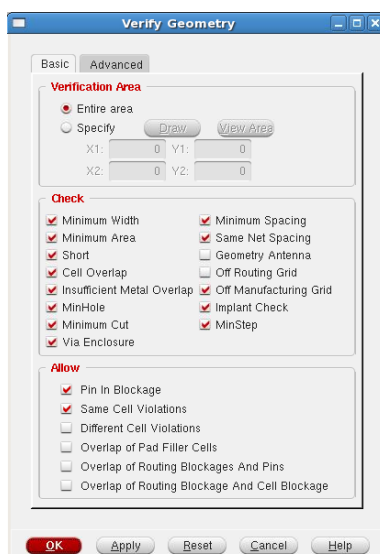
按 **OK**。

將跳出 Error Browser 視窗，檢查看看是否有錯，若有請自行修正到 0 個 Violation 為止。

- ii. 使用 Cadence Encounter 者，驗證 DRC 與 LVS 步驟如下：

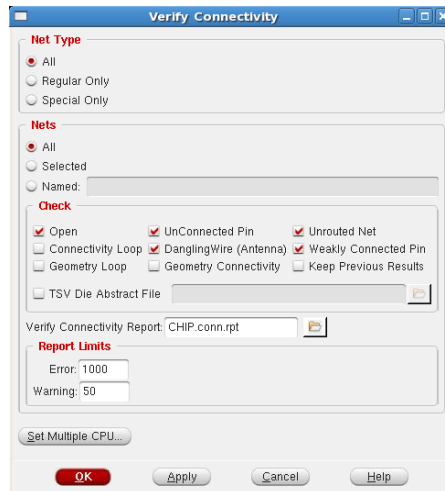
在 Encounter 視窗下點選

1. DRC 驗證：請選“Verify → Verify Geometry...” Default 值，按 **OK**。



註：若 DRC 有發生錯誤，請選“Tools → Violation Browser...”查明原因。

2. LVS 驗證：請選“Verify → Verify Connectivity...” Default 值，按 **OK**。



註：若 LVS 有發生錯誤，請選“Tools → Violation Browser...”查明原因。

附錄 D 評分用檔案

評分所須檔案可以下幾個部份：(1)RTL design，即各參賽隊伍對該次競賽設計的 RTL code，若設計採模組化而有多個設計檔，請務必將合成所要用的各 module 檔放進來，以免評審進行評分時，無法進行模擬；(2)Gate-Level design，即由合成軟體所產生的 gate-level netlist，以及對應的 SDF 檔；(3)Physical design，使用 **Synopsys IC Compiler** 者，請記得將整個 **Milkyway Library** 等相關的 design database，壓縮成一個檔案。使用 **Cadence Encounter** 者，請將 Encounter 相關的 design database (包含 **.enc** 檔案與 **and .enc.dat** 目錄)，壓縮成一個檔案。壓縮的檔案格式如下：假設參賽者的 design database 目錄名稱爲“your_lib”，請執行底下的 UNIX 指令，最後可以得到“your_name.tar”的檔案。

```
> tar cvf your_name.tar your_lib
```

在執行以上的指令之前，請確定將你使用的 P&R Tool 儲存後關閉，再執行上述的指令，否則在壓縮的過程會出現錯誤。

表 3

RTL category		
<i>Design Stage</i>	<i>File</i>	<i>Description</i>
N/A	N/A	Design Report Form
RTL Simulation	*.v or *.sv	Verilog 或 System Verilog ...
Gate-Level category		
<i>Design Stage</i>	<i>File</i>	<i>Description</i>
Pre-layout Gate-level Simulation	*_syn.v	Verilog gate-level netlist generated by Synopsys Design Compiler
	*_syn.sdf	Pre-layout gate-level sdf
Physical category		
<i>Design Stage</i>	<i>File</i>	<i>Description</i>
P&R	*.tar	archive of the design database directory
	*.gds	GDSII layout
	DRC/LVS report	不用儲存 DRC/LVS Report 檔案!只需在 Design Report Form 上填寫 DRC/LVS 錯誤總數量即可。(目標要做到 0 個錯誤!)
Post-layout Gate-level Simulation	*_pr.v	Verilog gate-level netlist generated by Cadence Encounter or Synopsys IC Compiler
	*_pr.sdf	Post-layout gate-level sdf

附錄 E 檔案整理步驟

當所有的文件準備齊全如表 3 所列，請按照以下的步驟指令，提交相關設計檔案，將所有檔案複製至同一個資料夾下，步驟如下：

1. 在自己的 home directory 建立一個新目錄，名稱叫做“**result**” 例如：
 - > **mkdir ~/result**
2. 將附錄 D 要求的檔案複製到 result 這個目錄。例如：
 - > **cp RFILE.v ~/result/**
 - > **cp RFILE_syn.v ~/result/**
 -
3. 在 Design Report Form 中，填入所需的相關資訊。

附錄 F 軟體環境

1. 使用者登入後自動會設定好以下軟體環境：

Vendor	Tool	Executable
Cadence	Virtuoso *1	icfb
	Composer	icfb
	NC-Verilog	ncverilog
	SOC Encounter	encounter
Synopsys	Design Compiler	dv, dc_shell
	VCS-MX	vcs
	IC Compiler	icc_shell -gui
	Hspice	hspice
	Cosmos Scope *1	cscope
	Custom Explorer *1	wv
	Laker *1	laker
	Laker ADP*1	adp
	Verdi *1	verdi, nWave, nLint
Mentor	Calibre *3	calibre
	QuestaSim	vsim
Utility	vi	vi, vim
	gedit	gedit
	nedit	nedit
	pdf reader	acroread
	calculate	gnome-calculator, bc -l
	gcc	gcc
	Matlab	matlab

EDA 軟體所須使用的 license 皆已設定完成，不須額外設定

*1 該軟體限定使用 1 套 license

*3 該軟體限定使用 3 套 license

附錄 G 設計資料庫

設計資料庫位置： /usr/cad/icc2018/CBDK_IC_Contest_v2.1

目錄架構

ICC/		
	tsmc13gfsg_fram/	ICC core library
	tsmc13_CIC.tf	ICC technology
	macro.map	layer mapping file
	tluplus/	
	t013s8mg_fsg_typical.tluplus	t13 tluplus file
	t013s8mg_fsg.map	t13 tluplus mapping file
SOCE/		
	lef/	
	tsmc13fsg_8lm_cic.lef	LEF for core cell
	lib/	
	slow.lib	worst case for core cell
	streamOut.map	Layout map for GDSII out
SynopsysDC/		
	db/	
	slow.db	Synthesis model (slow)
	lib/	
	slow.lib	timing and power model
Verilog/		
	tsmc13_neg.v	Verilog simulation model
Phantom/		
	tsmc13gfsg_fram.gds	Standard Cell GDSII file

附錄 H 一元二次方程式求根

若解題中需解一元二次方程式，在此提供公式解

$$a y^2 + b y + c = 0$$

其根為

$$y_{1,2} = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

本題依解題方法不同，未必會使用到此式。

Design Report Form

登入帳號(login-id)		
RTL category		
<i>Design Stage</i>	<i>Description</i>	<i>File Name</i>
RTL Simulation	使用之 HDL 名稱 (例如：Verilog、System Verilog)	
RTL Simulation	RTL 檔案名稱 (RTL file name)	
Gate-Level category		
<i>Design Stage</i>	<i>Description</i>	<i>File Name</i>
Pre-layout Gate-level Simulation	Gate-Level 檔案名稱 (Gate-Level Netlist file name)	
	Pre-layout sdf 檔案名稱	
	Gate-Level simulation, 所使用的 CYCLE Time (請確定模擬功能正確)	() ns
Physical category		
<i>Design Stage</i>	<i>Description</i>	<i>File Name or Value</i>
P&R	使用之 P&R Tool (請填入 IC compiler 或 Encounter)	
	設計資料庫檔案名稱 (Library name) (ICC: Milkyway Library Name, Encounter: xxx.enc.dat)	
	DRC 錯誤總數量 (ex: 0 個)	
	LVS 錯誤總數量 (ex: 0 個)	
Post-layout Gate-level Simulation	Gate-Level 檔案名稱 (Gate-Level Netlist file name)	
	Post-layout sdf 檔案名稱	
	Post-layout Simulation 所使用的 CYCLE Time (請確定模擬功能正確) (請寫出三組模擬中的最大值) Ex: 3ns	
	Post-layout Simulation Time (Simulation Time, ex: 49992 ns) Time = ? (請寫出三組模擬中的 最大值)	
	Power Measure using PT-PX Power = ? (ex: 1.275 mW) (請寫出三組測試樣本中的 最大值)	
Over All	最後完成之等級? (ex: 等級 A)	
其他說明事項 (Any other information you want to specify: (如設計特點 ...) 如寫不下可寫於背面		