

# 106-2 CVSD – HW3

## *Local Median Filter Engine*

### 1. 問題描述

請完成一Local Median Filter Engine(後文以**LMFE** 表示)的電路設計。此電路可以將任意輸入的二維(2D)灰階(Gray Level)影像訊號，做中間值濾波器(Median Filter)的運算，並將運算後影像的每一像素(Pixel)之結果，循序輸出。有關LMFE 詳細規格將描述於後。

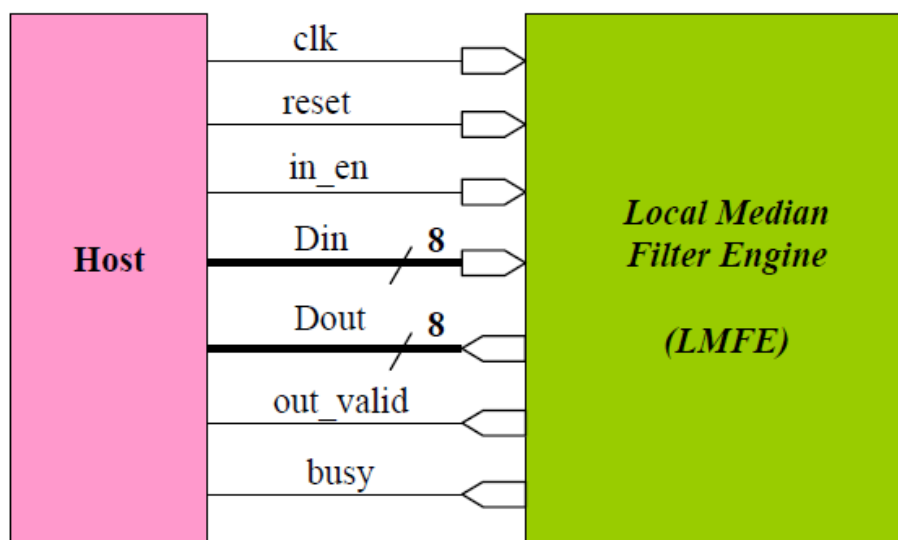
本電路各輸入輸出信號的功能說明，請參考表一。同學必須根據下一節所給的設計規格及附錄A 中的測試樣本完成設計驗證。



圖一、Local Median Filter Engine 之方塊圖

### 2. 設計規格

#### 2.1 系統方塊圖

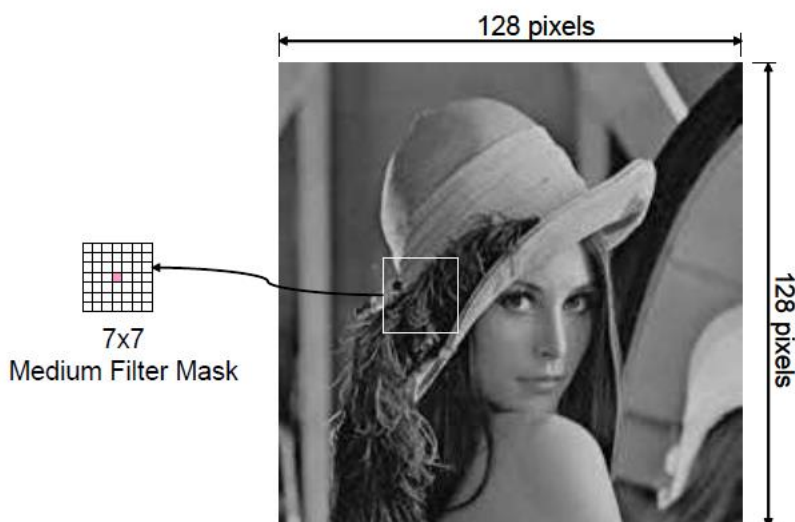


圖二、系統方塊圖

## 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。 (註: Host 端採clk”負”緣時送資料。)
reset	I	1	高位準”非”同步(active high asynchronous)之系統重置信號。
in_en	I	1	資料輸入致能控制訊號。當Host 端有資料要輸入時，時busy 訊號為Low，in_en 就會為High，反之，busy訊號為High 時，in_en 就會為Low。當Host 端所有資料送完後，該訊號到模擬結束前將永遠維持為Low。
Din	I	8	LMFE 資料輸入的匯流排。Host 端會透過此匯流排將整張完整影像的訊號進行輸入。每一個週期僅能輸入一個Pixel 值，且輸入過的Pixel 值無法再重複輸入一次。註：輸入順序請參照2.3.1。
busy	O	1	LMFE 忙碌之控制訊號。當為High 時，表示系統正處於忙碌階段，告知Host 端，暫時停止Din 資料的輸入；反之，當為Low 時，表示告知Host 端可繼續由Din輸入資料。
out_valid	O	1	輸出資料有效之控制訊號。當為High 時，表示目前輸出的資料為有效的；反之，當為Low 時，表示目前輸出資料為無效的，即不被採用。
Dout	O	8	LMFE 資料輸出的匯流排。當LMFE 計算完畢後，可透過此匯流排將運算完畢的Pixel 值，輸出至Host 端。 注意：每一個週期僅能輸出一筆Pixel 值。



圖三、灰階影像資料

## 2.3 系統描述

### 2.3.1 輸入端灰階影像訊號

LMFE 電路只針對灰階影像訊號做處理，如圖三所示，助教所提供之每張灰階影像尺寸固定為128x128，共計16384 個Pixels，每個Pixel 皆以8 位元表示，故其訊號範圍為0 ~255。影像輸入順序，固定為由左而右，由上而下的順序(即如圖四所示，影像訊號輸入順序按照方格中之編號 0,1,2,3,4,.....,16382,16383)，循序輸入至同學的LMFE 電路中。

注意：

1. 任何一個Pixel 之影像訊號只能讀取一次，並無反覆讀取之功能。
2. 讀取過程中，同學若想暫停影像訊號輸入，可以透過Busy 控制訊號拉為High 即可，處理適宜後，再將Busy 控制訊號拉為Low 即可繼續進行影像資料之讀取。

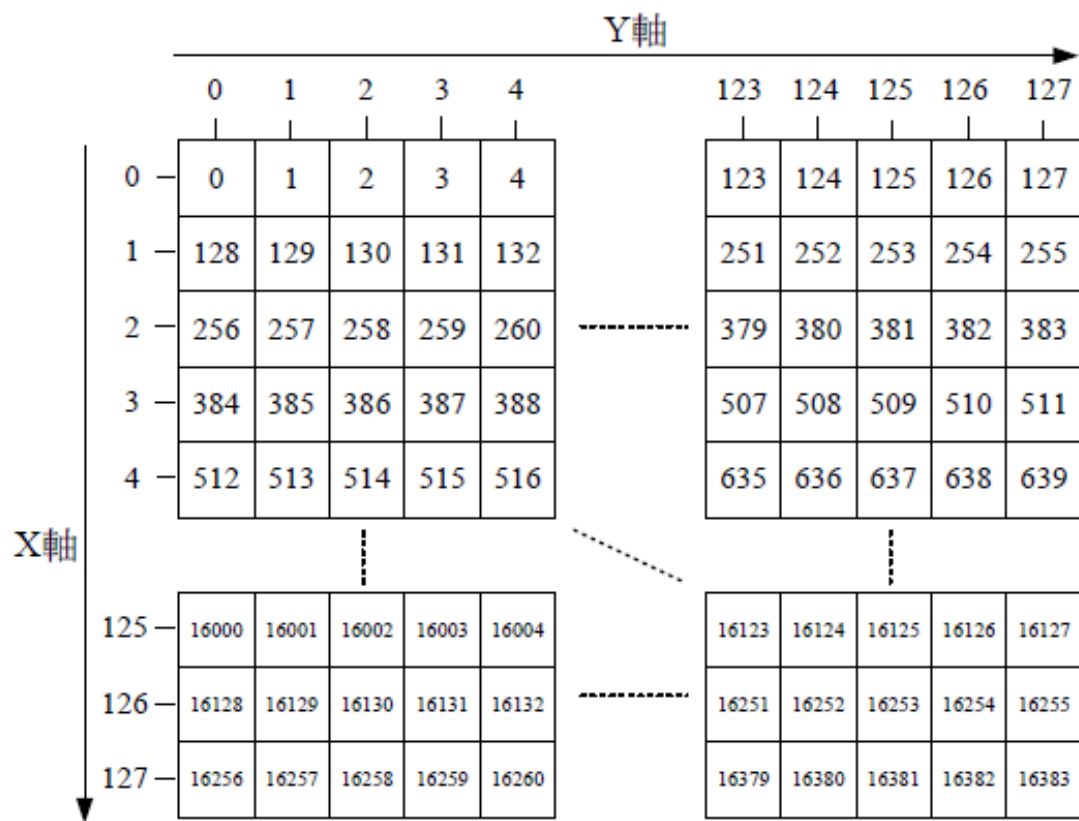
### 2.3.2 LMFE 電路運算方法

LMFE 電路運算方法係使用NxN 遮罩(Mask)來框住影像資料之局部區域(本題Mask 為固定7x7 之大小)，將被框住的範圍內NxN 個Pixels 中，找出其中間值來取代原影像點。如圖五所示，以N=3 為例，3x3 Mask 座標軸中心(X 軸, Y 軸)=(2, 2)，被框住的九個Pixels 影像訊號值經排列後可找出中間值為"147"，所以新的影像座標(2, 2)這點的值由"147"取代之。倘若被NxN Mask 框住的影像訊號超出影像邊界範圍外，請同學自行補0，再行計算。如圖六所示，以N=3 為例，3x3 Mask 座標軸中心(X 軸, Y 軸)=(0, 0)，共有五個Pixels 的訊號超出範圍外，將這些Pixel 補上0 後再行排列後，可找出中間值為"0"。

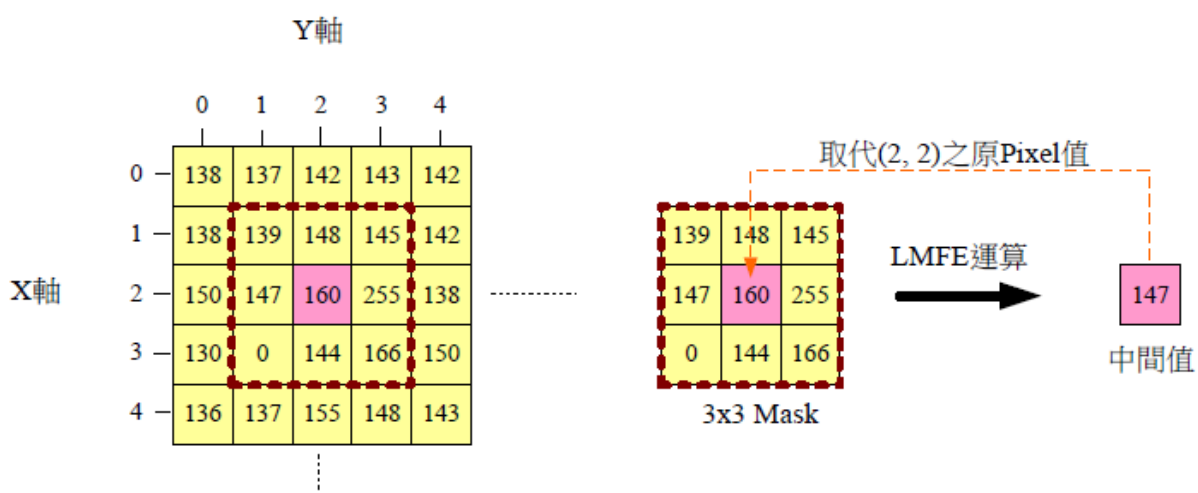
運算過程中，NxN Mask 中心之座標點可以由同學自行決定最恰當之順序，直到整張影像計算完畢。

注意：

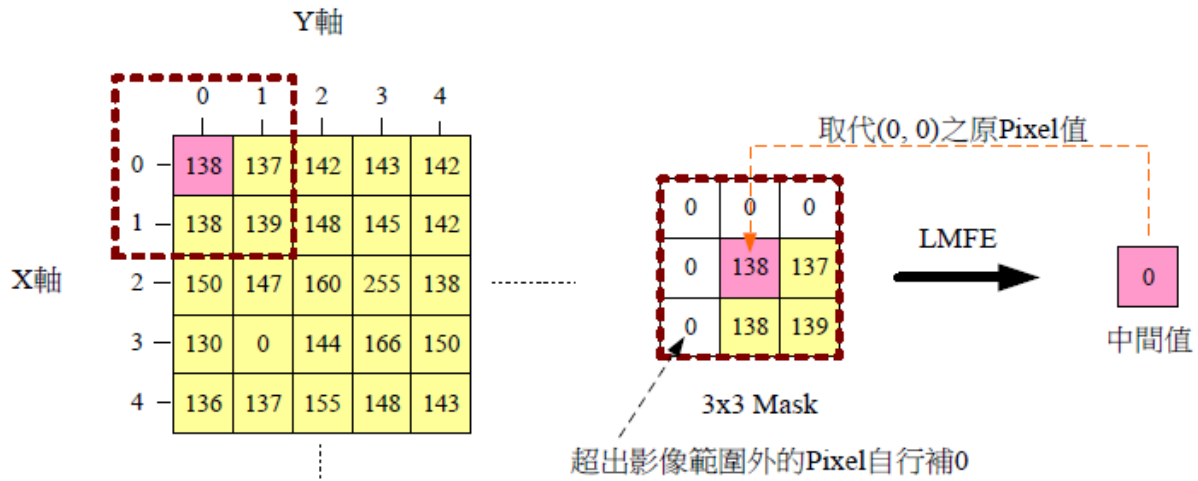
1. 本題Mask 為固定7x7 之大小，即固定N=7。



圖四、影像座標與影像輸入/輸出掃描順序圖



圖五、3x3 Mask 座標軸中心(2, 2)之範例



圖六、3x3 Mask 座標軸中心(0,0)之範例

### 2.3.3 LMFE 電路輸出順序

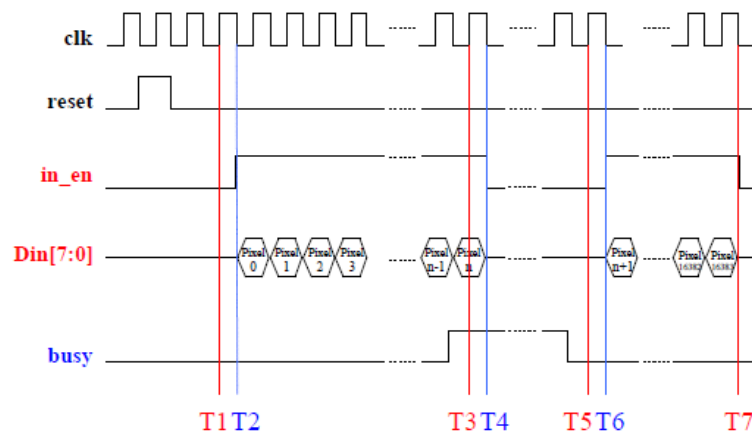
LMFE 輸出順序：請同學按照如圖四方格中之編號 0, 1, 2, ..., 16383 依序送出。當該 Pixel 值輸出是有效的，可將 "out\_valid" 訊號拉為 High，反之則為 Low。

注意：

1. 可自行決定，每運算出一個 Pixel 結果值立即輸出或整張影像都完成後再一併輸出。
2. 每一個週期只能輸出一個 Pixel 值。
3. 同一個座標點之 Pixel 值不可重覆送第二次以上。

## 2.4 電路時序規格

### 2.4.1 LMFE 電路輸入時序規格



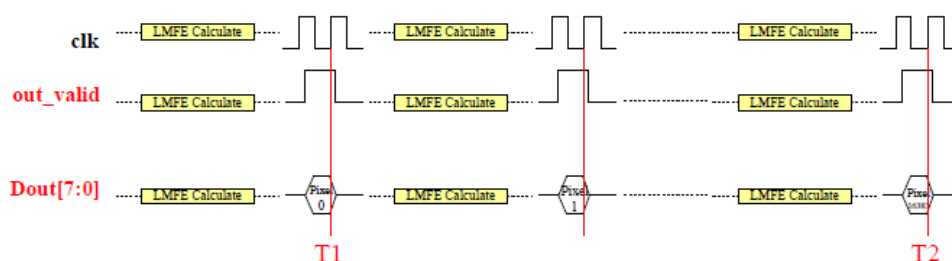
圖七(a)、LMFE 電路輸入之時序圖

1. LMFE 電路初始化，Reset 一個Cycle 的時間。
2. T1 時間點，Reset 後等待兩個Cycle 的時間，Host 於T1 時間點判斷busy 訊號為Low，因此T2 時間點，開始送出第一筆Pixel 訊號。
3. T3 時間點，是假設同學欲暫停影像訊號輸入，可將busy 拉為High，Host 於T3 時間點判斷busy 為High，Host 於T4 時間點開始暫停影像訊號輸入。**Host 沒有送訊號期間，Din 會一直維持在高阻抗 (High Impedance)，in\_en 維持在Low。**
4. T5 時間點，Host 端判斷busy 為Low，於T6 時間點繼續輸入剩餘的影像Pixel 訊號。

註: T3~T6時間點，僅提供給需要暫停影像訊號輸入之同學參考範例，並非強制要中斷讀取。

5. T7 時間點，Host 已完成輸入16384 筆影像訊號， T7 時間點以後，**Din 會一直維持在高阻抗 (High Impedance)，in\_en 維持在Low。**

## 2.4.2 LMFE 電路輸出之時序規格



圖七(b)、LMFE 電路輸出之時序圖

同學可讀取局部或全部影像訊號後，開始進行LMFE 運算。在輸出部分，同學可自行決定，每運算出一個Pixel 結果值立即輸出或整張影像都完成後再一併輸出。

1. T1 時間點，同學已完成Pixel 0 (即如圖四，輸出順序編號0)之計算，輸出至Host，並控制 out\_valid 訊號為High，剩餘之影像訊號依此類推，**請參照2.3.3 之規定順序輸出。**
2. T2 點，同學已經將該影像最後一筆Pixel16383 計算完畢後輸出，本模擬立即結束。

### 2.4.3 SRAM 記憶體規格與時序規格

製作LMFE 電路，同學應該會用到Single Port SRAM 記憶體。在此提供兩種SRAM 資源 (Resource)，分別是1KB (1024 x 8 bits)、8KB (8192 x 8 bits)記憶體，同學可依自己需求挑選最適當數量。例如：某同學需用到16KB SRAM，可以在硬體描述語言描述中，將8KB SRAM 呼叫兩顆或1KB SRAM 呼叫16 顆使用。

有關1KB 及8KB 記憶體細節規格與記憶體讀寫時序圖，詳如記憶體附件中。

注意：

1. 假如同學有需要用到1Kbyte 以上記憶體，建議”不要使用Register (Flip-Flop)來製作記憶體”，以免造成合成時，恐怕會跑很久甚至有如當機的現象發生，請注意!

## 附錄

### 附錄A 設計檔(For Verilog)

#### 1. 下表為設計檔

表2、設計檔案說明

檔名	說明
<b>LMFE.v</b>	同學所使用的設計檔，已包含系統輸/出入埠之宣告
pattern1.dat	提供16384 個Pixels 的影像訊號，即一張完整影像訊號。 <b>欲模擬 pattern1 的結果，請務必搭配testfixture1.v 檔案</b> <b>註：樣本皆以十六進制表示。</b>
golden1.dat	提供一張影像經過LMFE運算後的影像訊號值，共計16384 筆。 <b>註：Golden Pattern 以十六進制表示。</b>
<b>testfixture1.v</b>	Test Bench 已自動加入對應的pattern 檔、golden 檔。
.synopsys_dc.setup	使用Design Compiler (DC)作合成之初始化設定檔。同學請依Library 實際擺放位置， <b>自行填上Search Path 的設定。注意：無論合成或APR，只需使用worst case library</b> ；例如：slow.db、sram_1024x8_t13_slow_syn.db、sram_8192x8_t13_slow_syn.db 等library。
LMFE_DC.sdc	使用DC 作合成之sdc 檔。 <b>同學可自行調整cycle值。</b>
sram_1024x8_t13.v	1KB SRAM Verilog 模擬用之檔案。
sram_8192x8_t13.v	8KB SRAM Verilog 模擬用之檔案。
main.tcl	助教提供之合成基本指令集。
tsmc13.v	TSMC 0.13um製程檔，僅提供同學進行gatelevel模擬用，請勿外傳於網路上。

請使用**LMFE.v**，進行LMFE 電路之設計。其模組名稱、輸出/入埠宣告如下所示：

```

module LMFE (clk, reset, Din, in_en, busy, out_valid, Dout);
input    clk;
input    reset;
input    in_en;
output   busy;
output   out_valid;
input    [7:0] Din;
output   [7:0] Dout;
endmodule

```



2. 提供的一個Test Bench 檔案testfixture1.v 對應到 pattern1.dat 與 golden1.dat、，這些都已加入了，同學只要注意pattern 檔案的路徑即可。

例如：

Test Bench 模擬，使用testfixture1.v：

```
`define PAT "./pattern1.dat"
```

```
`define EXP "./golden1.dat"
```

註：同學無須作修改，只需注意pattern1.dat的檔案位置即可，預設為目前目錄。

3. 提供之Test Bench 檔案，多加敘述如下：

```
`define End_CYCLE 10000000
```

```
`define SDFFILE "./LMFE_syn.sdf"
```

```
`ifdef SDF
```

```
initial $sdf_annotate(`SDFFILE, LMFE);
```

```
`endif
```

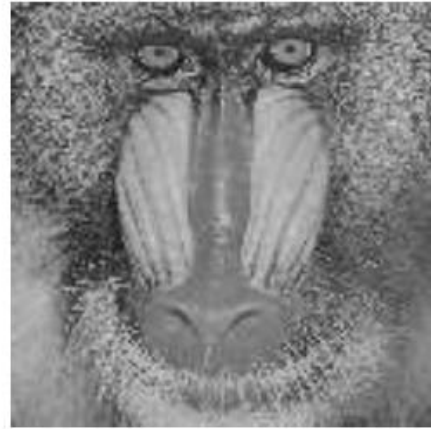
註：

1. End\_CYCLE 預設1000 萬個Cycles，其目的可以防止同學因電路有錯，模擬陷入無窮迴圈之境，倘若同學確定模擬需超過1000 萬個Cycles 以上，可自行再加大此Cycle 數。
2. SDF 檔案，請自行修改SDF 實際檔名後模擬。
3. 在Test Bench 中，提供`ifdef SDF 的描述，其目的是讓本Test Bench 可以作為RTL模擬、合成後模擬用。注意：當同學在合成後模擬，請務必多加一個參數”+define+SDF”，方可順利模擬。

註：使用NC-Verilog 執行Gate-level 模擬時間甚久，為方便同學辨識目前模擬狀況，這邊提供的testbench於模擬時，比對每1000 個Pixels 資料正確，即秀出一次正確的訊息，最後模擬到出現”Congratulations! All data have been generated successfully!”訊息表示該影像檔經LMFE 運算比對結果完全正確。



(a)



(b)

圖八、兩張真實 128x128 灰階影像之測試樣本(a) Lena 圖，(b) Baboon 圖

## 附錄B 測試樣本

兩組真實影像之測試樣本，如圖八(a)、(b)所示。為了讓同學看完題目後，更能明確題意，在此以Lena 影像訊號之測試樣本為例，如圖九(a)所示，為原始Lena影像訊號(由於篇幅有限，在此僅秀出Lena 最左上方影像7x7 之大小作範例)，經過7x7 Mask做LMFE 運算後可以得到新Lena 影像訊號如圖九(b)所示。

	0	1	2	3	4	5	6		0	1	2	3	4	5	6
0	138	138	140	131	134	130	132	7X7 Mask LMFE運算後	0	0	0	130	130	130	130
1	137	136	136	132	131	134	131		1	0	131	132	132	131	131
2	133	133	134	135	133	134	129		2	0	132	133	133	132	132
3	134	132	132	133	131	129	130		3	131	133	134	134	133	134
4	134	134	134	134	135	132	132		4	132	133	134	134	134	134
5	135	130	134	134	135	134	126		5	132	134	134	134	134	135
6	129	130	134	135	135	135	131		6	132	134	134	135	135	137

(a)

(b)

圖九、Lena 128x128 灰階影像訊號(a)原始影像訊號，(b) LMFE 運算後新影像訊號值

## 附錄 C 繳交檔案

RTL Category		
Design Stage	File	Description
N/A	StudentID.pdf	Design Report Form
RTL Simulation	LMFE.v	Verilog synthesizable RTL code
Gate-level Category		
Design Stage	File	Description
Pre-layout Gate-level Simulation	LMFE_syn.v	Verilog gate-level netlist generated by Synopsys Design Compiler
	LMFE_syn.sdf	Pre-layout gate-level sdf
	LMFE_syn.ddc	Design database generated by Synopsys Design Compiler

Naming convention: **StudentID\_HW3\_vk.zip**  
**(k is number of version, k = 1,2,...)**

FTP: 只允許上傳/ 無權限下載或刪除

Deadline	2018/5/1 中午 12:00
IP	140.112.20.128
Port	1232
Account	CVSD_STUDENT
Password	cvsd2018

## 附錄 D 助教聯絡方式

王勝輝 harry@access.ee.ntu.edu.tw

李懷霆 wesli@access.ee.ntu.edu.tw

## 附錄 E 評分標準

(1) 依是否完成設計第一階段評分(60%):

即完成下列三項要求。請按照 A、B、C 的要求順序完成。

◇ “完成設計”的三項要求:

A、RTL 通過作業提供的測試樣本模擬。(20%)

B、RTL 在助教的測試資料之下通過模擬。(20%)

C、完成 Synthesis，且 FAS\_syn.v 的 GateLevel Pre-layout Simulation 有達到 A、B 項之要求。(20%)

<Note1> 請確認助教在 RTL 模擬上可以直接跑下列指令。

```
ncverilog testfixture1.v LMFE.v sram_1024x8_t13.v  
sram_8192x8_t13.v +notimingcheck
```

如果有額外的 module 在其它.v 檔，請利用`include 檔案在 FAS.v 之中，並且確認檔案命名。  
在 gate-level 模擬，助教會用以下指令：

```
ncverilog testfixture1.v LMFE_syn.v sram_1024x8_t13.v sram_8192x8_t13.v  
tsmc13.v +define+SDF
```

對於 gate-level simulation，請記得更改 SDF 檔案的名稱以及確認檔案位置，與 testbench 中的變數”SDDFILE”一致。

<Note2> 合成環境的設置：請根據 LAB 中 Synthesis 部分所提供的”.synopsys\_dc.setup”設置。

target_library	slow.db fast.db
link_library	* \$target_library dw_foundation.sldb
symbol_library	generic.sdb
synthetic_library	dw_foundation.sldb

<Note3> 關於合成 **sdc constraint** 除 **clock cycle** 之外不可更動。

(2) 依設計的 Cost 做第二階評分(40%):

助教會將符合第一階段完成設計(三項要求都完成)者，依據下式作第二階段的評分，其中 Cost 越低成績越佳。

$$\text{Cost} = \text{Area}(\mu\text{m}^2) * \text{Total simulation time}(\text{ns})$$

1. Area = synthesis report 中的 Total Cell Area (不含 wire load model 面積)
2. Total simulation time = clock period\*total clock cycle (ncverilog 總模擬時間)
3. 請將 design compiler 關於 Area 以及 ncverilog 模擬截圖附在報告中以證明，否則此部分無法獲得分數。

A+ Class: ~5% (40 points)  
 A Class: 6%~10% (35 points)  
 B Class: 11%~25% (30 points)  
 C Class: 26%~70% (20 points)  
 D Class: 71%~100% (10 points)  
 F Class: 在 gate-level 模擬時有錯誤 (0 points)

- ◇ 注意，因助教測資對於 Stage Analysis 之精準度考量請不要 truncate 此部分的運算。
- ◇ 在此次作業中助教會嚴厲抓抄襲，實際看.v 檔以及合成時的.ddc 檔(工作檔，可知道 constraint 等)，並且檢查報告內容是否與程式相符以確認公平性。任何作弊行為將會導致無法通過這門課。

### (3) 作業報告:

1.

附上數據表:

Area( $\mu\text{m}^2$ )	
Clock cycle(ns)	
Total simulation time(ns)	
Cost( $\mu\text{m}^2 \cdot \text{ns}$ )	

2.

Area (total cell area) in Design compiler 截圖

RTL 以及 gate-level ncverilog simulation 截圖

3.

架構實現方法