

# **NTHU CS 516000**

## **FPGA Architecture & CAD**

### **Final Project Report**

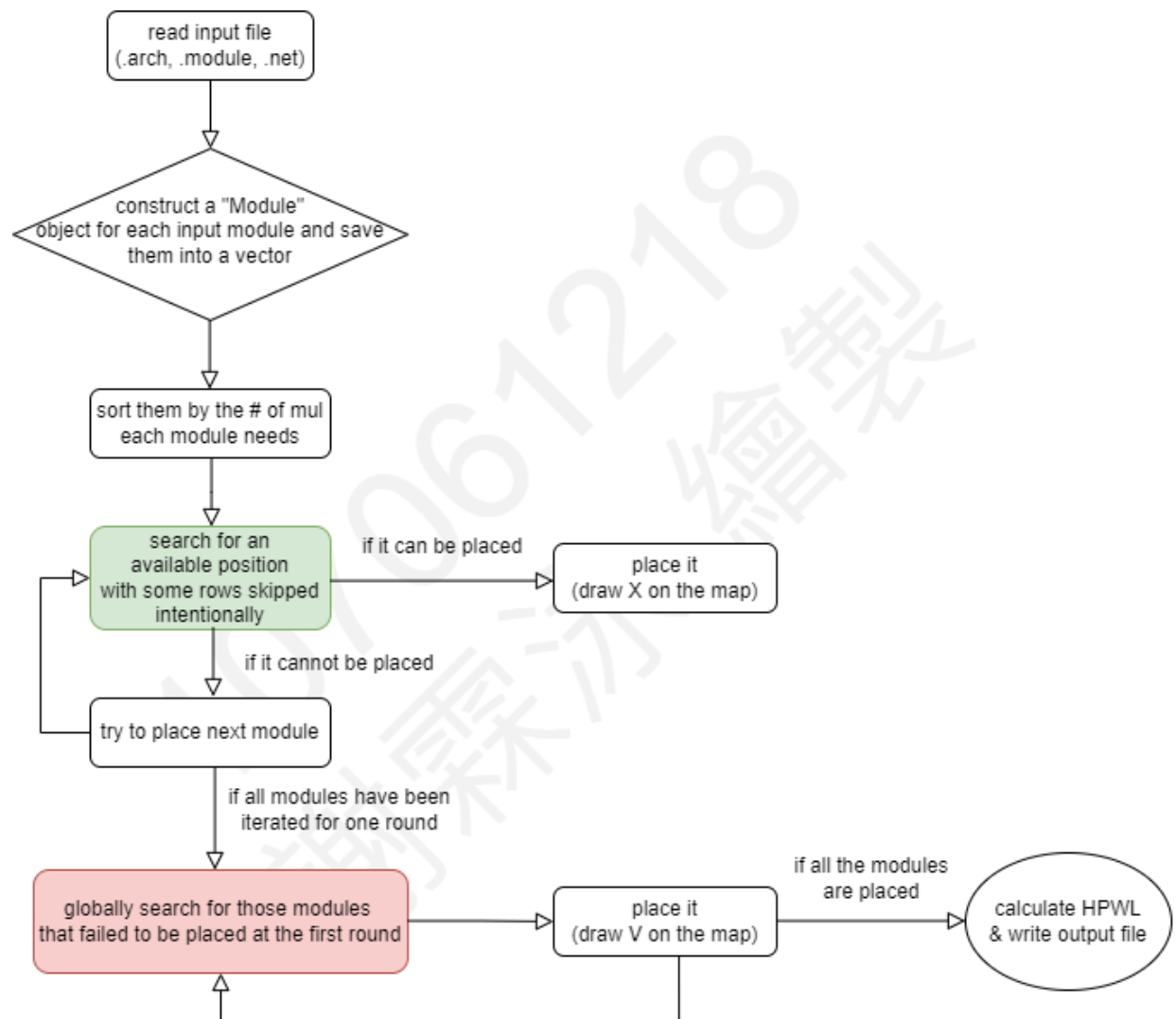
Title: Floorplanning for FPGAs with Heterogeneous  
Resources

Student ID: 107061218

Name: 謝霖泳

Member's name: X (finish on my own)

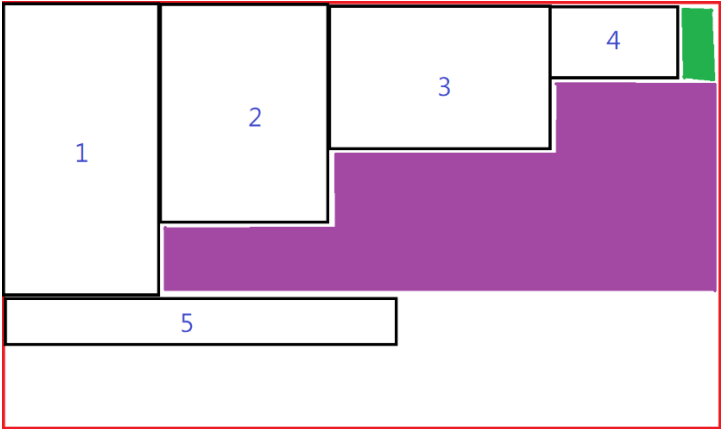
## 一、系統架構及流程圖



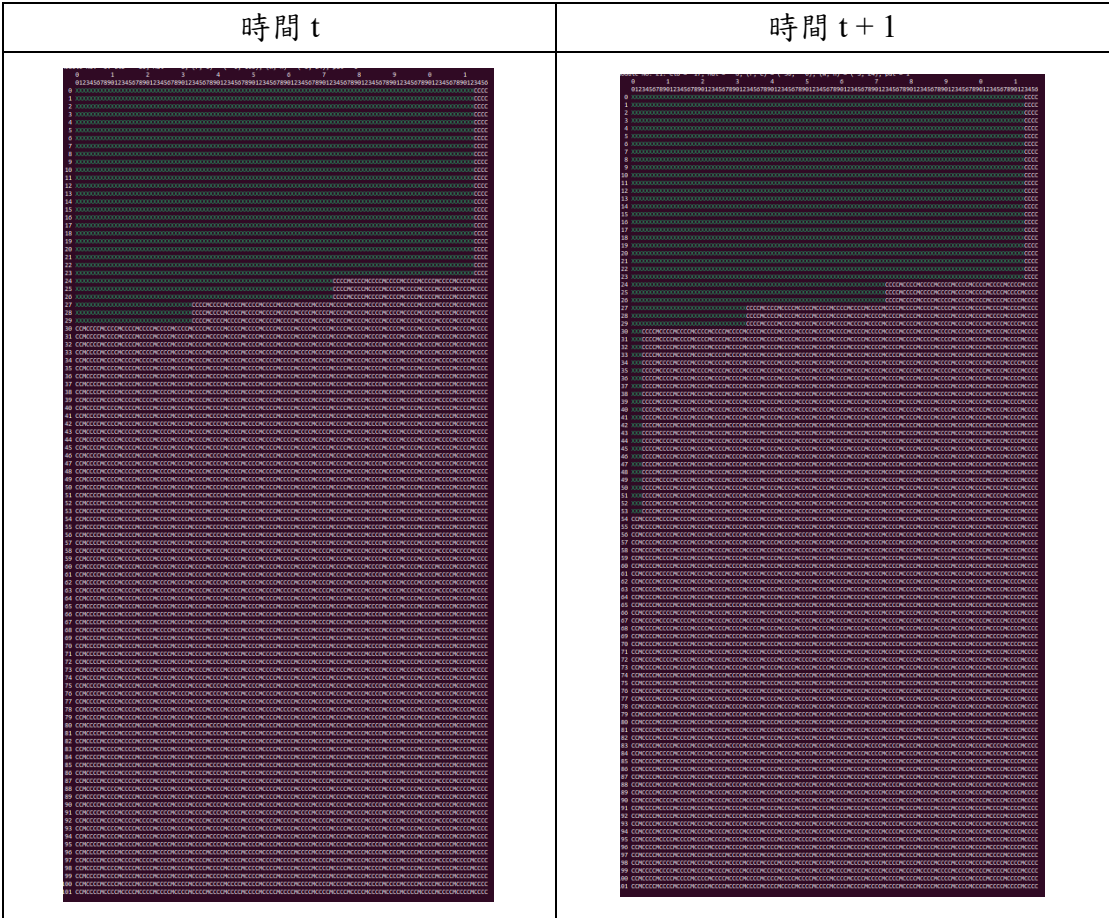
我的設計方法主要分成了幾個部分，先將 input 讀進來並存好我要的資料後，我會將所有的 module 進行排序，排序的 key 是先依照 MUL 數量，再依照 CLB 數量進行排序，所以需要較多 multiplier 的就會先放，因為佔的高度比較高。

排序完之後，就會進行第一次擺放，高度我都直接決定好就是 multiplier 的數量\*3，接著開始往右擴增，增加 width 直到 MUL 及 CLB 的數量均滿足要求，就在相應的位置畫 X 做記號並進行下一個 module 的擺放。反之，若寬度已經擴增到這個 row 的盡頭仍無法放進來的話，為了維持較佳的空間利用性，接下來並不會直接從下一個 row 開始嘗試，而是會從這個橫排最高的那一塊所佔到的 row 的下一個下去進行排序，也就是 flow chart 中的綠色區塊寫的：有些 row 會被跳過，並不會將全部的 row 都 iterate 過。

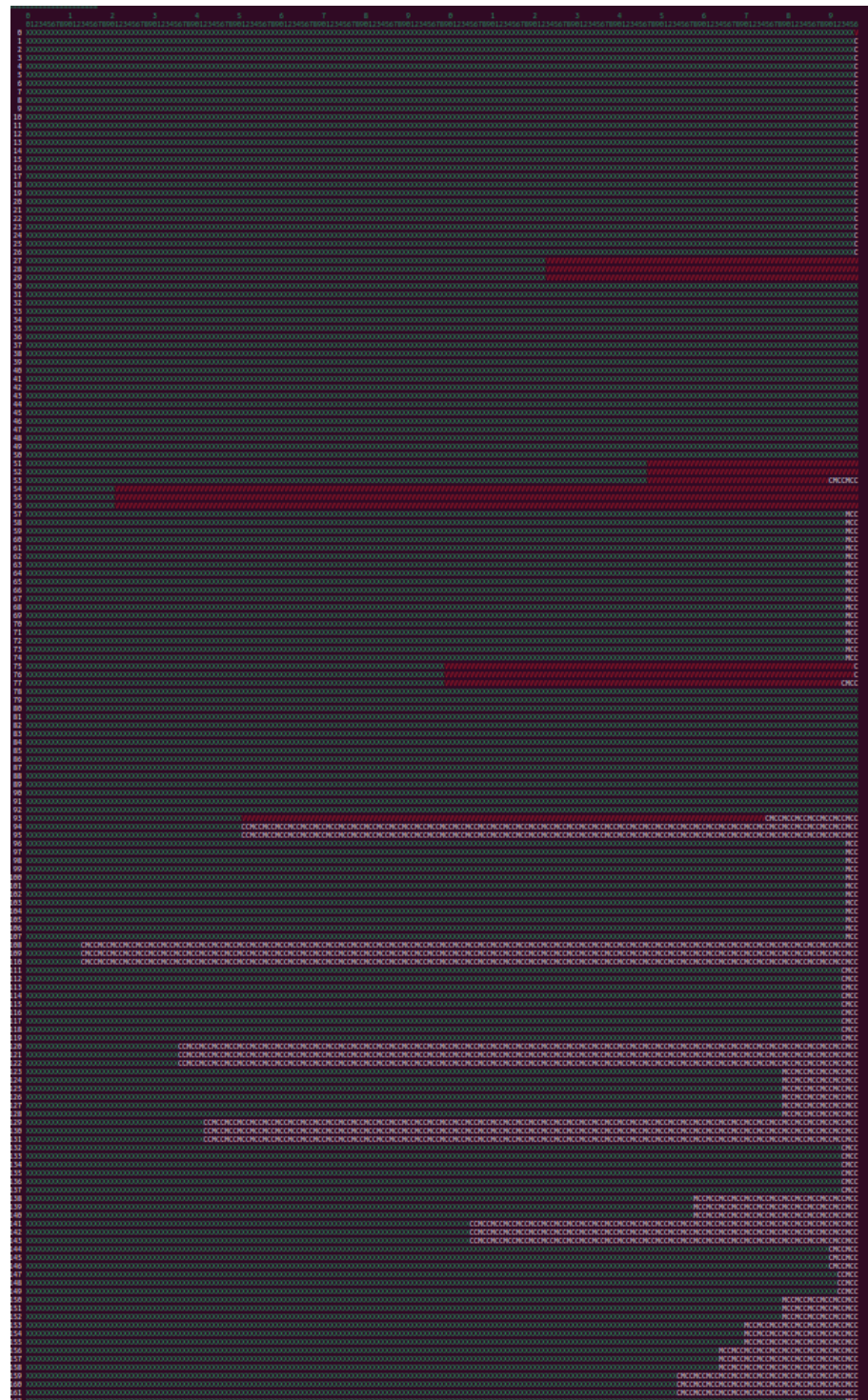
以下是示意圖。當我要放第五塊的時候，發現第四塊右邊的綠色區域已經塞不下了，我下一個 iterate 的 row 並不是 row1，而是 block1 的高度+1 那個 row，所以並不會有東西放進紫色區。會這樣做的目的是防止這塊如果被放進紫色區，疊加起來的高度可能比第一塊還高，這樣會造成空間的浪費。



以下是真實情況，我用綠色顯示的 X 就是已經被 occupied 的位置。可以發現第二張圖左邊相較於第一張圖，多了一條很長的綠色，這條就是因為它在原本的 row 往右擴增的時候，高度限制不能超過該 row 最高的高度，所以放不進去，但如果沒有這個限制是可以放進去的，但往下就會超過原本的高度，導致空間浪費。



經過第一輪擺放之後，大部分的 module 都可以完美擺入，接著便要對剛剛放不進去的 module 進行 global search 的**第二次擺放**，逐一 iterate 過每個 row 跟 column，只要找到可以的位置就插進去，因為大部分需要 multiplier 的大 module 都已經在第一輪擺放中擺進去了，所以需要動用到第二輪擺放的大多是小型 module，幾乎都不會用到 multiplier，所以應該都可以輕鬆找到屬於自己的窩。找到之後，我便將該位置畫紅色的 V，來與第一輪擺放的 module 作分別。目前僅有 test case 6 動用到第二輪搜尋，其最終結果長這樣。



以下是我的 pseudo code。

```
1. read_file
2. construct the map and save all the modules
3. # R, C denotes the # of row and column of the global map
4. for module in all_module:
5.     can_place = False
6.     for r = last_row, last_row + height, last_row + 2 * height:
7.         for c = 0: C - 1:
8.             if mul is enough and clb is enough:
9.                 place
10.                break
11.            width++ until the space meets the requirement
12. # global search
13. for module that has not been placed:
14.     for r = 0: R - 1:
15.         for c = 0: C - 1:
16.             if mul is enough and clb is enough:
17.                 place
18.                 break
19. calc HPWL
20. write file
```

## 二、 程式的編譯及執行(以 case6 為例)

### 1. 編譯

***g++ main.cpp***

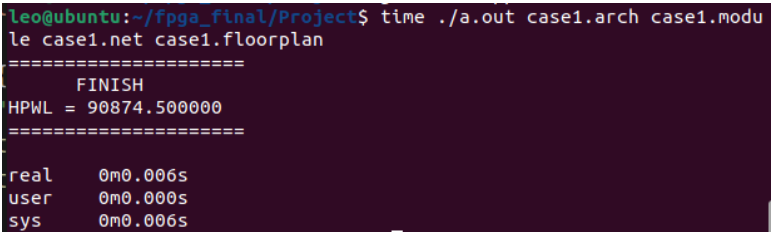
### 2. 執行

***./a.out case6.arch case6.module case6.net case6.floorplan***

### 3. 驗證

***./verifier benchmarks/case6.arch benchmarks/case6.module  
benchmarks/case6.net outputs/case6.floorplan***

## 三、 各個 benchmark 之執行情況

Case #	HPWL	runtime (ms)	執行圖片
1	90874.5	6	

2	92897.5	7	<pre> leo@ubuntu:~/fpga_final/Project\$ time ./a.out case2.arch case2.modu le case2.net case2.floorplan ===== FINISH HPWL = 92897.500000 =====  real    0m0.007s user    0m0.000s sys     0m0.007s </pre>
3	446172.5	15	<pre> leo@ubuntu:~/fpga_final/Project\$ time ./a.out case3.arch case3.modu le case3.net case3.floorplan ===== FINISH HPWL = 446172.500000 =====  real    0m0.015s user    0m0.014s sys     0m0.000s </pre>
4	327991.5	16	<pre> leo@ubuntu:~/fpga_final/Project\$ time ./a.out case4.arch case4.modu le case4.net case4.floorplan ===== FINISH HPWL = 327991.500000 =====  real    0m0.016s user    0m0.014s sys     0m0.000s </pre>
5	562491	23	<pre> leo@ubuntu:~/fpga_final/Project\$ time ./a.out case5.arch case5.modu le case5.net case5.floorplan ===== FINISH HPWL = 562491.000000 =====  real    0m0.023s user    0m0.014s sys     0m0.008s </pre>
6	531168	26	<pre> leo@ubuntu:~/fpga_final/Project\$ time ./a.out case6.arch case6.modu le case6.net case6.floorplan ===== FINISH HPWL = 531168.000000 =====  real    0m0.026s user    0m0.025s sys     0m0.000s </pre>

以上的 HPWL 計算結果均與助教提供之 verifier 計算結果無異。

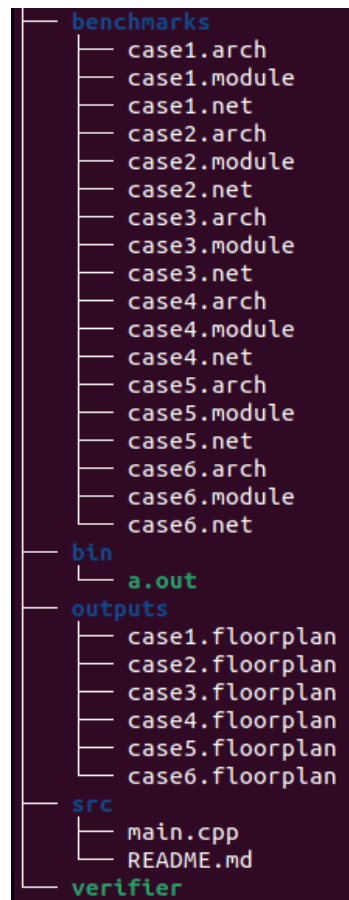
```

leo@ubuntu:~/fpga_final/Project$ ./verifier benchmarks/case1.arch benchmarks/case1.module benchmarks/case1.net outputs/case1.floorplan
Total WireLength: 90874.5
leo@ubuntu:~/fpga_final/Project$ ./verifier benchmarks/case2.arch benchmarks/case2.module benchmarks/case2.net outputs/case2.floorplan
Total WireLength: 92897.5
leo@ubuntu:~/fpga_final/Project$ ./verifier benchmarks/case3.arch benchmarks/case3.module benchmarks/case3.net outputs/case3.floorplan
Total WireLength: 446172.5
leo@ubuntu:~/fpga_final/Project$ ./verifier benchmarks/case4.arch benchmarks/case4.module benchmarks/case4.net outputs/case4.floorplan
Total WireLength: 327991.5
leo@ubuntu:~/fpga_final/Project$ ./verifier benchmarks/case5.arch benchmarks/case5.module benchmarks/case5.net outputs/case5.floorplan
Total WireLength: 562491
leo@ubuntu:~/fpga_final/Project$ ./verifier benchmarks/case6.arch benchmarks/case6.module benchmarks/case6.net outputs/case6.floorplan
Total WireLength: 531168

```



#### 四、 檔案架構



#### 五、 GitHub 連結

[https://github.com/LeoTheBestCoder/fpga\\_final](https://github.com/LeoTheBestCoder/fpga_final)