

其他有助於審查之資料- 樊明勝

| | |
|--------|---|
| 目 錄 | <ul style="list-style-type: none">☞ 專題成果 (EDA 相關)☞ 專題內容 (EDA 相關)☞ 課業、Project 上的表現☞ 就學計畫☞ 其他 |
|--------|---|

各位甄選委員們您好，我的名字是“樊明勝”，目前是清華大學的資訊工程系的大四應屆畢業生。興趣的研究方向比較偏向是 GPU computing acceleration 與研究上的結合、用 GPU 去加速 CAD 的運算、或 EDA 與 Blockchain Computation 上的優化、CAD 優化 Intelligent chip 設計等等。

✚ 專題成果：

內容是獨自報名參賽 2021 的 IC/CAD contest，參加題目是 Problem F，在成果上獲得了特優，且在專題的競賽上目前進入了決賽

所附圖表上的 NFSCORE 是你的得分 normalize 化(除以該 testcase 最高分)的結果，題目總共有 20cases，其中有十七筆有 95up 的成績，其中九個拿下 100 分，代表為那九個 cases 上的參賽者中表現最好的。

在專題的過程中也學習到不少有關 EDA 對於線路 optimization 上的知識，而在實作上，因為這題目的要求在其它 paper 上沒有看到類似的研究方向，因此在過程中的優化上的演算法或技巧要自己想該如何做，而在驗證的自己的 verilog 過程中也有學操作 ABC 和 cad tool 像 VCS

| | | | | | | | | | | | | | | | | | | | | |
|------------------------------|----------------------------------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|------------|------------|------------|------------|------------|------------|------------|------------|------------|------------|------------|
| Testcase | testcase1 | testcase2 | testcase3 | testcase4 | testcase5 | testcase6 | testcase7 | testcase8 | testcase9 | testcase10 | testcase11 | testcase12 | testcase13 | testcase14 | testcase15 | testcase16 | testcase17 | testcase18 | testcase19 | testcase20 |
| FFSCORE | 1.95653 | 0.126844 | 0.178167 | 0.000000 | 0.000000 | 0.000000 | 0.000000 | 0.000000 | 0.000000 | 0.000000 | 0.000000 | 0.000000 | 0.000000 | 0.000000 | 0.000000 | 0.000000 | 0.000000 | 0.000000 | 0.000000 | 0.000000 |
| NFSCORE | 100 | 98.82049 | 100 | 100 | 100 | 100 | 96.54088 | 97.28922 | 99.68051 | 98.36168 | 100 | 100 | 100 | 100 | 100 | 99.28398 | 100 | 99.19871 | 99.71256 | 99.71256 |
| fail to generate optimized.v | | | | | | | | | | | | | | | | | | | | |
| function changed | | | | | | | | | | | | | | | | | | | | |
| optimized.v parsing fail | | | | | | | | | | | | | | | | | | | | |
| total NFSCORE: | 1699.874 | | | | | | | | | | | | | | | | | | | |
| Note: | testcase13: optimized.v is empty | | | | | | | | | | | | | | | | | | | |

※F題

| | | |
|----|----------|------------|
| 獎項 | 得獎隊伍 | 得獎學生 |
| 特優 | cadb0106 | 樊明勝 (清華大學) |

[得獎公告連結](#)

專題內容：

■ 題目要求與例子概述

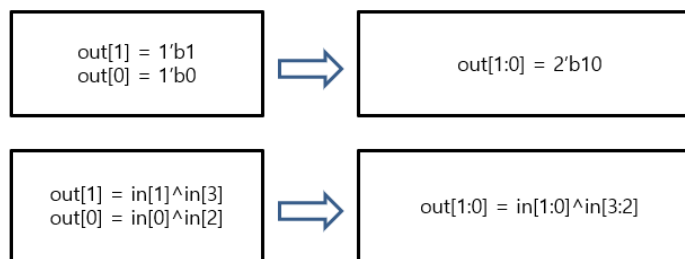
2021 的 IC/CAD contest 的 Problem F，score 上的 spec 表示希望減少他給 verilog testcase 中的 assignment 跟 bit selection 的數量，所謂的 bit selection 指說像是 in[1] 中的 [] 符號會被算是 bit selection count。

```
assign out[3:2] = in[1:0];  
(2 bit/part selects: out[3:2], in[1:0])
```

<COUNT_ASGN> + <COUNT_SEL> = 1 + 2 = 3

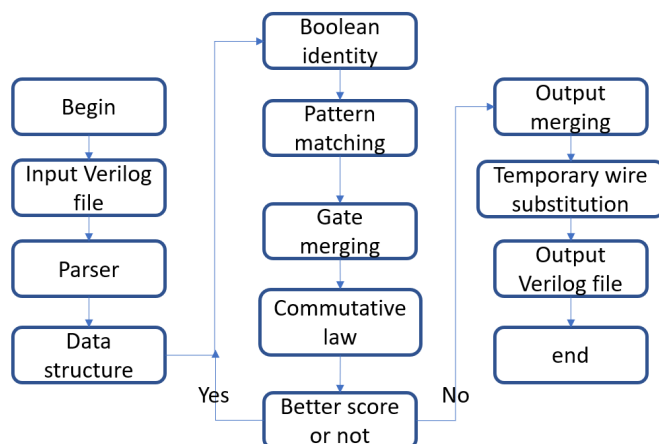
以上圖做為例子，就是一個 assignment 跟兩個 bit selections，且這兩個 bit selections 是作為連續的 selection。

而作為 input 的 verilog file 則為非連續的 bit selection，那在優化的過程中只要 circuit 的 functionality 都沒有改變的情況下，可以做線路的簡化以及合併都是允許的。



所謂的合併可以看上述的例子，而這動作被我稱做為 output merging，是在優化中蠻重要的一步，可以同時減少 assignment 跟 bit selection。

■ 程式優化步驟



上圖為整個的程式的運作流程圖，我認為自己在 pattern matching 跟 output merging 實作上是勝過他人的部分，用自己的想法結合上自己的 data structure 是關鍵。

■ 題目設計的目與程式結果

程式的目的就是為了減少 verilog file 中的 assignment 跟 bit selection 的數量，也同時等同於減少了 instructions 的數量，可以減少 simulator 花在 fornt-end 的時間，而加速 simulator 等同加速 validating functional correctness.

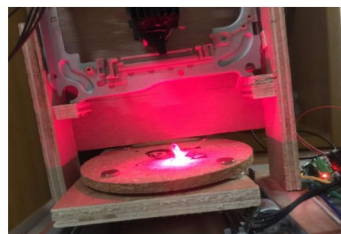
那下面的表格顯示，左側是尚未優化的得分，右側是優化後的得分，而比率的意思代表說相比於原來的 verilog file，優化後檔案的 number of instructions 只佔原本檔案中的多少數量，在幾筆數據當中差異相差快 10 倍和 75 倍，也就是成果是將 verilog 的行數減少到快原來的 1/10 更甚者 1/75。

| Testbench | original | optimized | ratio of score | Testbench | original | optimized | ratio of score |
|-----------|----------|-----------|----------------|-----------|----------|-----------|----------------|
| case 1 | 0.3658 | 0.5660 | 0.6462 | case 6 | 0.0005 | 0.0376 | 0.0132 |
| case 2 | 0.4102 | 0.6106 | 0.6717 | case 7 | 0.3977 | 0.6424 | 0.6191 |
| case 3 | 0.4042 | 0.6336 | 0.6379 | case 8 | 0.5232 | 1.9565 | 0.2674 |
| case 4 | 0.0714 | 0.5714 | 0.1250 | case 9 | 0.1069 | 0.1239 | 0.8627 |
| case 5 | 0.4444 | 4 | 0.1111 | | | | |

■ 課業、Project 上的表現:

在修課上清大的教授在大一的時候即對邏輯設計方面有清晰的教學，對 verilog 上面也有足夠紮實的練習，打下了良好的基礎，也因此在此課程和邏輯實驗上都可以取得 A++ 的成績，此外，在計算機概論和 EDA 上，資工系主任也教的非常清楚，個人在這些方面的理解都有足夠的把握，也因此有請教授幫我寫推薦信。

在邏輯實驗的期末作業上，我跟另一個組員合作，寫 verilog 用 vivado 和 FPGA(Xilinx Artix7) 合成出一台雷射列印的 CNC，獲得了綜合評比的第二名，以及人氣獎，而在這份計劃的整體構思和結構，以及機器的製作主要是由我所負責的，因此在連結中所看到的架構圖和 module 的銜接及 state 上的控制皆由我所處理。附註: [連結](#)

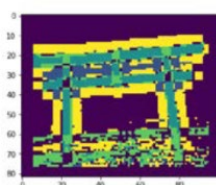


在嵌入式系統上面作業表現完整，期末作業在 **arduino 接上其它晶片 (nodemcu)**結合 IoT，用手機語音去控制一個遊戲，拿到的分數有 98 分，而在當時的時候 IoT 的結合是自學的，另一個晶片的部份也是。 附註: [連結](#)

在線性代數的 final 作業上，是對於物品的**色塊分層**，除了基本要求外，我還有提出一個的演算法，可以達到讓物體有比較明顯的分色使物品顯現出來。在 report 最後面是我的算法的部份。 附註: [連結](#)



原圖



經由我的算法

就學計畫

| | |
|--|--|
| 根據老師專業領域的不同，一是運用 GPU 作用在 CAD 的 tool 的運算上面，二是研究運用 CAD 使 Blockchain Computation 的設計上優化 | |
| GPU 作用在 CAD tool 的運算上面 | <p>將 GPU 作用在 CAD 的 tool 的運算上面，試看看能不能形成更多的平行化效果，在我所做的比賽當中，有些步驟上(像是 boolean identity 或 pattern matching)在運算的時候資料結構其實是獨立的，那如果我可以有效的做到平行化的計算，相信在 CAD 上面的電路驗證或電路合成會更有效率。</p> <p>用現在的研究趨勢的 CAD 中的資料結構 AIG，作為例子，我們知道在每個 node 的時候，當需要找 k-cut 的時候，不會是只有一個 node 是要尋找，會是一串的 sub node 先行尋找 cut，那如果將平行化的運算運用在這邊的話，AIG 的 rewriting 就會比較快，甚至是在後面找尋 FRAIG 的時候也是一樣的道理，node 的添加在 thrashing 的時候也會變快</p> |
| CAD 使 Blockchain Computation 設計上優化 | <p>研究運用 CAD 使 Blockchain Computation 的設計上優化，在現在區塊鏈、元宇宙的趨勢下，機器的運算能力與功耗是一個很重要的點，即使之後的 blockchain 從 pow 換成 pos 的趨勢，但還是有大量的交易的紀錄要處理，那如果可以在區塊鏈的計算上，找到可以用硬體加速或是比較用低功耗的邏輯運算都是蠻重要的點，而且對目前運算的 gas fee 上有所幫助。</p> <p>在功耗上，我認為說可以利用 n,p transistor 特性上的不同，會造成不一樣的耗能的點，試著進行著手看看。</p> <p>在速度上，應該根據加密貨幣的 hash 計算上面的不同，為他的計算方式，選擇應該使用的資料結構，跟計算的優化的方式不同，甚至是在排程的線路上做優化</p> |

其它經歷與興趣

學校的表現外，在暑假時有協助舉辦參加 2019 清大 ioncamp(為教學競技程式的營隊)，對於競程上在當時也有所了解; 在大三下的時候參加 leetcode 的讀書會，在其中精進自己的 coding 上的能力，讓自己可以打出更好的 code。

在學業上外的平時生活，認為自己是一個喜歡多方探索及興趣的人，且在實作上面具有自己的想法，對於財金知識、區塊鏈、資工相關的領域也有興趣，並還在了解學習中。

My github repo:[連結](#)

自傳有提到的成果跟資料可以在 github 上面看到

期許

綜上所述，期望可以有幸申請上台大電子工程學研究所 EDA 組，在過程中增進自己的能力，且在實務上可以有所表現，對台大課程跟研究上抱有很大的期待，謝謝評審、教授。