

積體電路設計實驗 Integrated Circuit Design Laboratory

2023 FALL Midterm Project - Maze Router Accelerator

姓名:李品賢 帳號:iclab021 系級:電子碩 112 學號:312510151

一、 電路架構

硬體使用：

1. 128 words – 128 bits SRAM
儲存指定的 frame_id 所對應的 Location map
2. 128 words – 128 bits SRAM
儲存指定的 frame_id 所對應的 Weight map
3. 64 x 64 2 bits Register
實作 Lee's algorithm 之 Fill 和 Retrace

演算法 Lee's algorithm:

1. 初始設定
 - 在 64 x 64 Register path map 中，以 2'd0 代表 empty，2'd1 代表 blocked，2'd2 及 2'd3 代表 Lee's algorithm 在 fill 的過程中填入的「1」、「2」。
 - 初始狀態的 path map，只有在 macro 處為 2'd1 (blocked)，其餘為 2'd0 (empty)。
 - 將 source 處填入 2'd2，sink 處填入 2'd0，作為 fill 及 retrace 的終止條件。
2. Fill
 - 以一個 2 bits 的計數器 step_cnt 在 fill 的過程中不停往上數，計數器的數字 2'd0, 2'd1, 2'd2, 2'd3，分別對應填入 path map 值為 2'd2, 2'd2, 2'd3, 2'd3。
 - 從 source 開始，以 BFS 的方式往外擴散，依序填入 2'd2, 2'd2, 2'd3, 2'd3，其中 source 代表「第一個」2'd2，source 往外的第一步代表「第二個」2'd2，若 sink 被填入 2'd2 或 2'd3 時，結束 fill。
3. Retrace
 - 計數器於此時往下數，可以往回推在 fill 所填入的 2'd2, 2'd2, 2'd3, 2'd3 的順序。

- 每 trace 一格的時間為 2 cycle，第一個 cycle，將此格填入 2'd1(blocked)，將 Location map SRAM 的資料讀出，也將 Weight map SRAM 的資料讀出，並將 weight 累加至 cost register，第二個 cycle，將 net_id 寫回 Location map SRAM，並 retrace 下一格所在位置。

二、FSM (流程圖)

狀態使用：

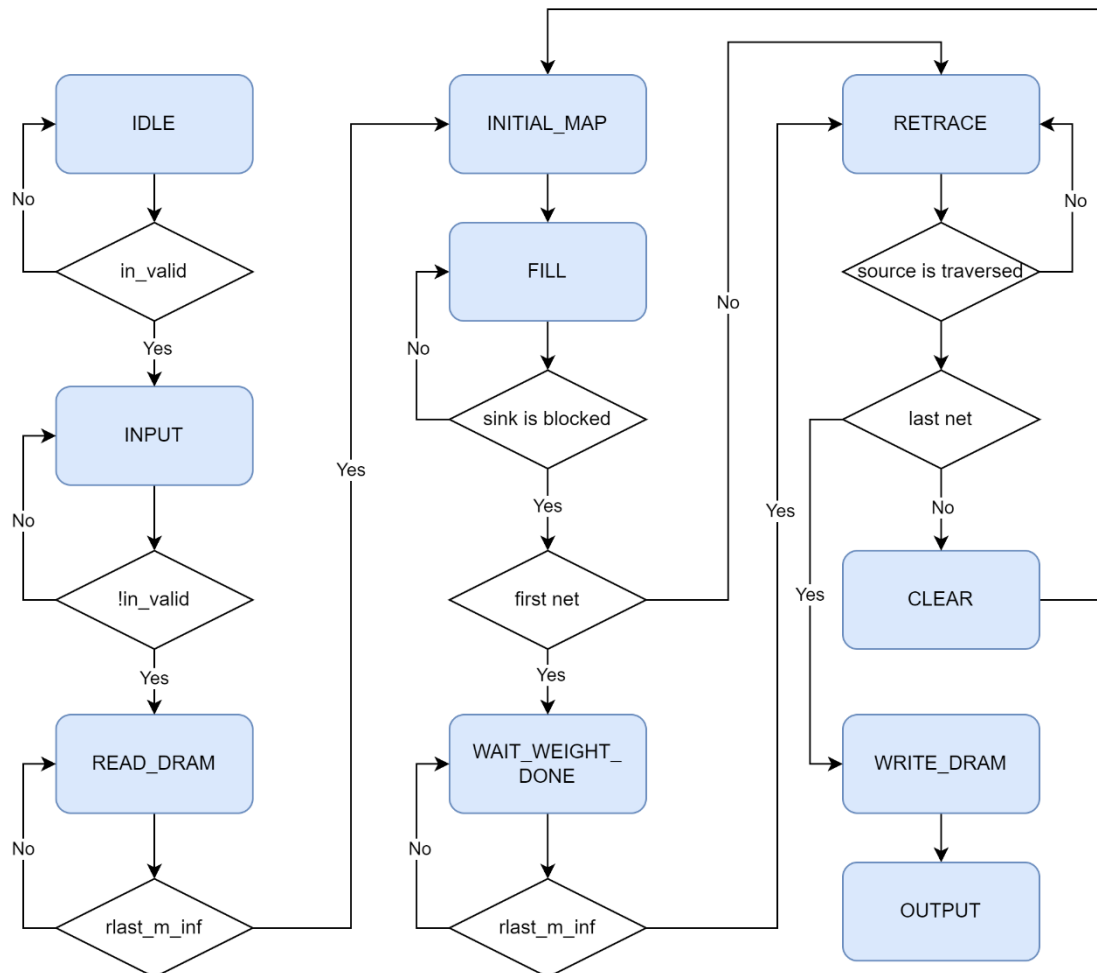
1. IDLE
等待 in_valid 訊號拉起，跳至 INPUT state。
2. INPUT
將 frame_id 及 net_id 及所對應的 source, sink 儲存，當 in_valid 降下，跳至 READ_DRAM state。
3. READ_DRAM
和 DRAM 溝通，將 frame_id 對應的 Location map 存進 SRAM，也順便初始化 path map，當 rlast_m_inf 拉起時，跳至 INITIAL_MAP state。
4. INITIAL_MAP
將選中的 net_id 所對應的 source, sink 填入特殊值，以利 Lee's algorithm 進行，source 填入 2'd2，sink 填入 2'd0，1 cycle 後，跳至 FILL state。
5. FILL
 - 實作 Lee's algorithm，從 source 開始，以 BFS 的方式往外擴散，若 sink 被填入 2'd2 或 2'd3 時，進行 state 轉換，若此時為第一條 net，跳至 WAIT_WEIGHT_DONE state，若非，跳至 RETRACE state。
 - 若此時為第一條 net，和 DRAM 溝通，將 frame_id 對應的 Weight map 存進 SRAM。
6. WAIT_WEIGHT_DONE
做第一條 net 才會進此 state，當 rlast_m_inf 拉起時，跳至 RETRACE state。
7. RETRACE
從 sink 一格一格往 source trace，當 trace 到 source 時，進行 state 轉換，若此時為最後一條 net，跳至 WRITE_DRAM state，若非，跳至 CLEAR state。
8. CLEAR
將 path map 中，除了 2'd1 外的格子都填入 2'd0，1 cycle 後，跳至 INITIAL_MAP state。
9. WRITE_DRAM

將 Location map SRAM 的值存回 DRAM，當 bvalid_m_inf 拉起時，跳至 OUTPUT state。

10. OUTPUT

將 busy 訊號拉下，1cycle 後，跳至 IDLE state。

Finite State Machine:



圖一 FSM 流程圖

三、 優化方法

1. 以 counter 初始化 path map

以 counter 將 path map 從 DRAM 依序讀入，會比以 shifter register 讀入節省面積，約少 6-8 萬。

2. 共用判斷式

將常用到的判斷式，以額外的一條線接著(舉例:assign c_FILL = (c_state == FILL))，能夠節省面積，約少 2-3 萬。

3. 以 2'd2, 2'd3 代表 Lee's algorithm 的「1」、「2」

因為 $2'd2 == 2'b10$ ，且 $2'd3 == 2'b11$ ，在 fill 時，若周圍四格的 MSB 為 "1"，代表此格需要被填入當時對應的值，以此方法只需判斷一個 bit (MSB)，若以較直覺的 $2'd1, 2'd2$ 代表 Lee's algorithm 的「1」、「2」，則每次都必須判斷兩個 bits。