

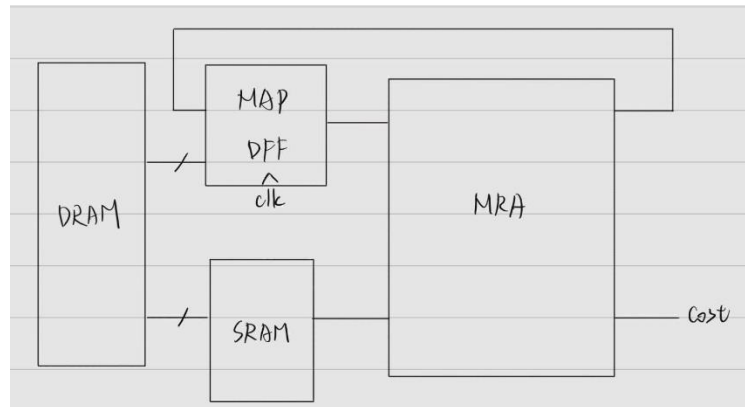
# Midterm Project

姓名：蘇柏叡

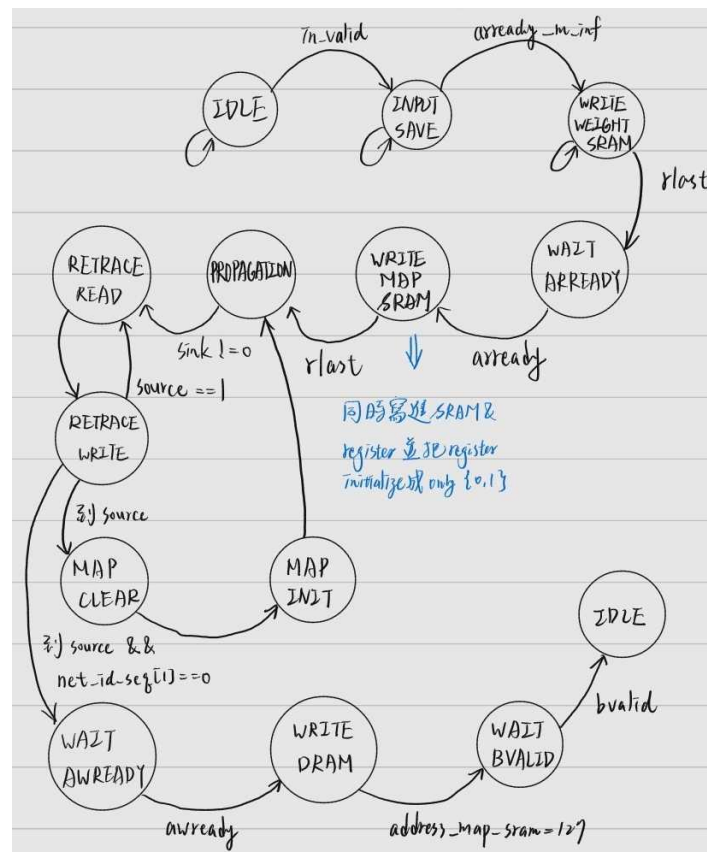
學號：109511055

iclab 帳號：iclab103

1. 電路架構：從 DRAM 把值讀出來到 DFF 跟 SRAM 裡面，並用 MAP DFF 的值去做丟到 MRA 裡面做 propagation，retrace 時同時利用 MAP DFF 跟 SRAM 裡的值去算 cost。



2. FSM 如下，看 state 的名字應該還蠻好理解的，比較特別的是 retrace 的部分分成 2 個 cycle，因為 SRAM 讀寫最少需要兩個 cycle，MAP INIT 是把 source、sink 的部分抓出來分別標成 3 跟 0。



### 3. 優化方法：

- 演算法的部分基本上就跟助教講解的一樣，差別只有在 1122 變成 2233，一開始是想說 0 當成可以走、1 當成障礙物，那就只能用 2233 了，後來在優化時才想到用 2233 的話可以只比較 MSB 就知道能不能 retrace。
- 優化面積的部分主要就是看有沒有 redundant 的邏輯，上下左右兩行在讀進 DFF 的時候可以不用判斷，還有因為是用 2233 去做 propagate 的關係，所以很多時候都可以只判斷 MSB 就好，像是判斷是否 retrace 結束了(如下圖)等等，且因為這次會用到很多變數放在 index 裡的寫法，這樣寫一次就會合出包括所有可能的大 MUX，因此減少一個 bit 的判斷一次就可以少好幾百個 MUX。

```
PROPAGATION : begin
    if(map_dff[loc_y_seq[1]][loc_x_seq[1]][1] == 1) begin // sink value != 0
        nxt_state = RETRACE_READ;
        // show_map;
    end
    else begin
        nxt_state = PROPAGATION;
        // show_map;
    end
end
```

- 速度的部分就沒有特別去優化，基本上就被 area 跟演算法決定了，因為這次的 project 實際運算其實沒有很大，大部分都只有 MUX 而已，所以 period 一定可以很低，比較少會有 slack 的問題，但要滿足 250 萬的限制，所以最後測到的甜蜜點為 8ns，面積為 247 萬。