

中位數計算

題目

- 輸入九個8 bit unsigned integer，輸出這九個數的中位數

算法

- odd-even transposition sort

Unsorted array: 2, 1, 4, 9, 5, 3, 6, 10

Step 1(odd): 2 1 4 9 5 3 6 10

Step 2(even): 1 2 4 9 3 5 6 10

Step 3(odd): 1 2 4 3 9 5 6 10

Step 4(even): 1 2 3 4 5 9 6 10

Step 5(odd): 1 2 3 4 5 6 9 10

Step 6(even): 1 2 3 4 5 6 9 10

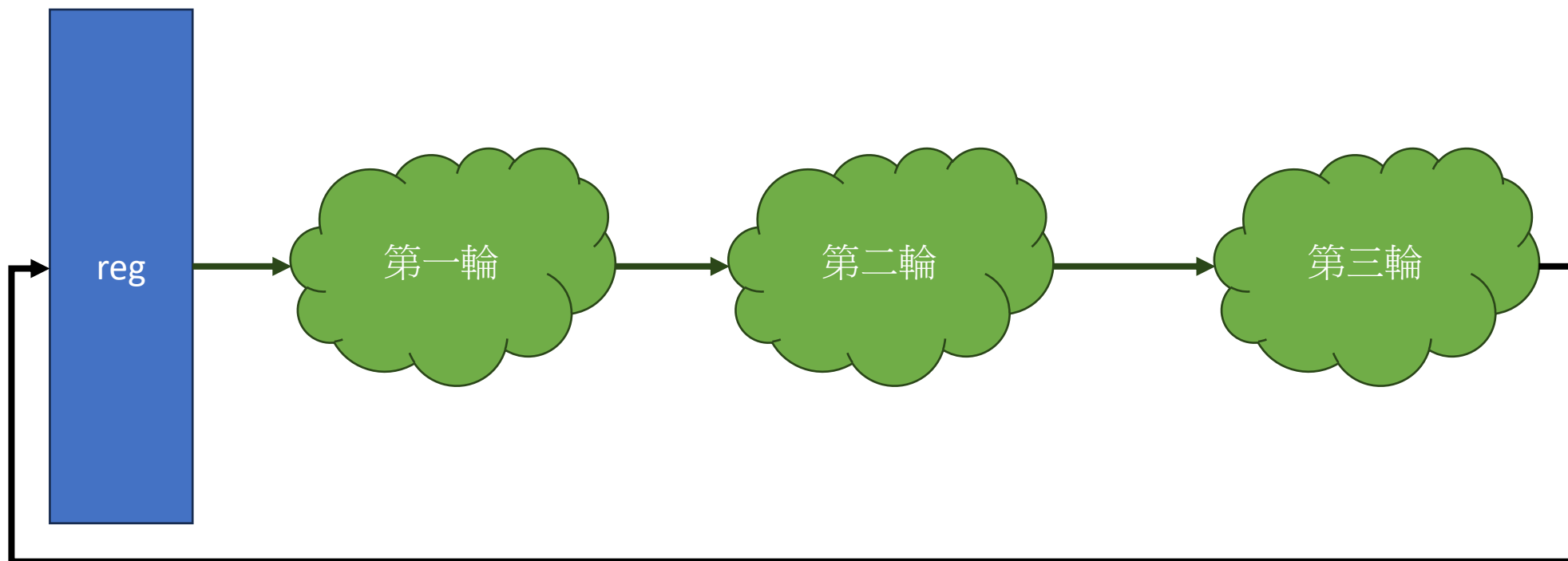
Step 7(odd): 1 2 3 4 5 6 9 10

Step 8(even): 1 2 3 4 5 6 9 10

Sorted array: 1, 2, 3, 4, 5, 6, 9, 10|

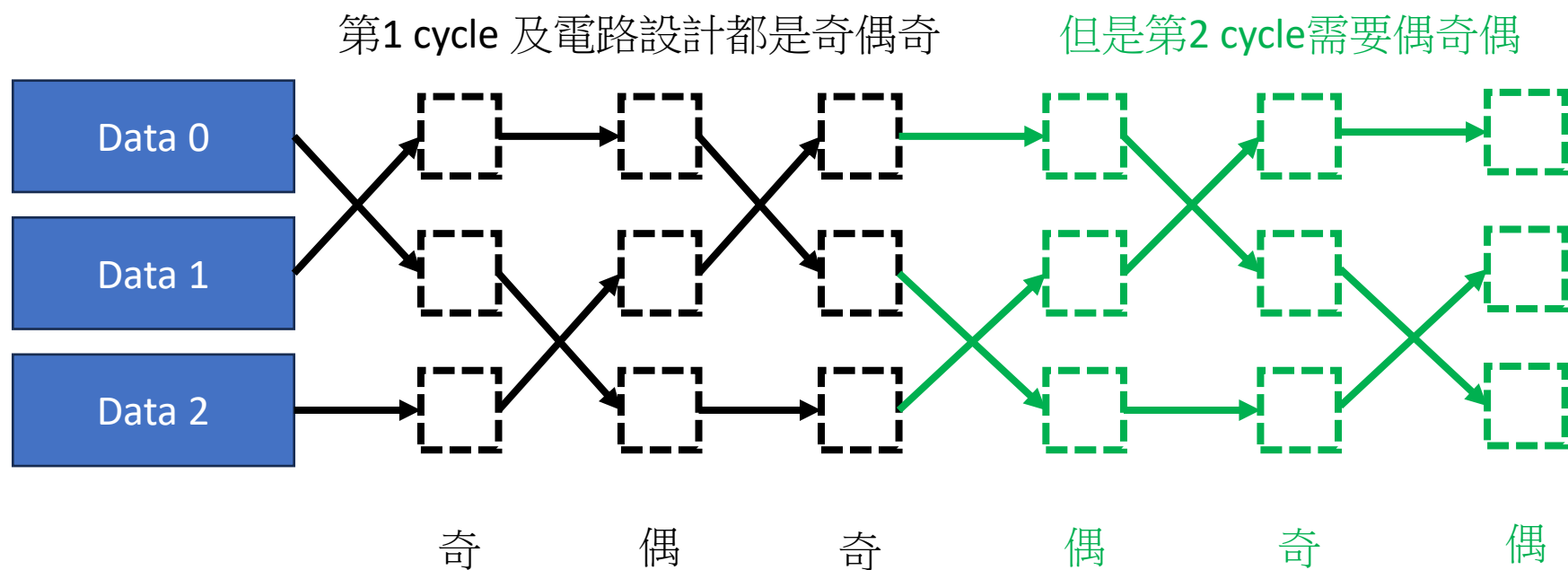
初步想法

- 九個數要做九輪，拆成3 cycle做，每cycle做3輪



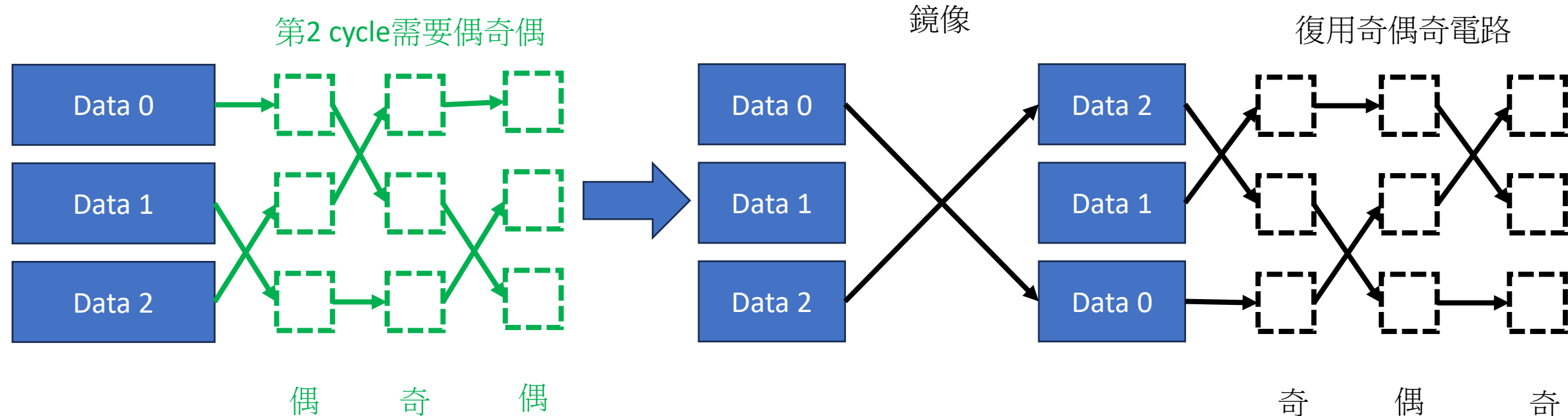
問題1:奇偶數對不上

- 電路是奇->偶->奇，第一cycle沒問題
- 第二 cycle需要偶->奇->偶，與電路不同



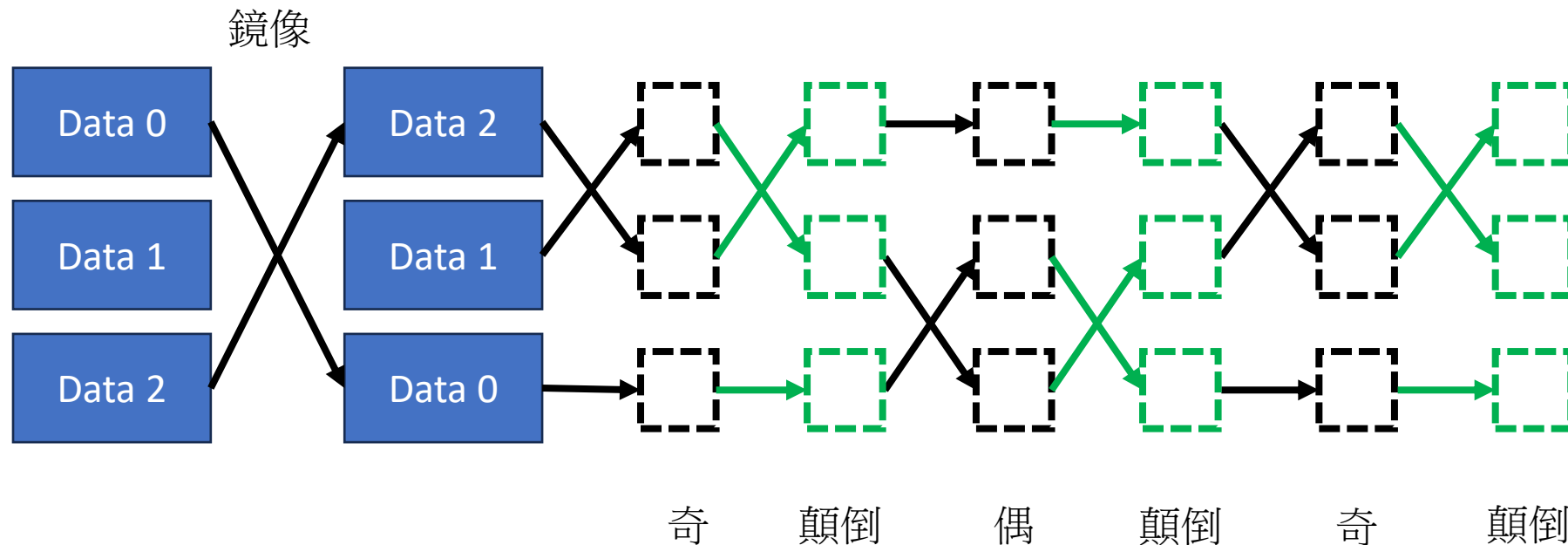
解決方法:鏡像

- 在這個題目裡，奇偶數的電路其實在input鏡像後是類似的
- 加一組mux，在需要的時候鏡像，就可以復用同一組電路



問題2: sorting方向相反

- 原本的**sorting**是小的往上，大的往下。鏡像後，為了延續之前的中間結果，**sorting**方向也要鏡像反過來，變成大的往上。
- 解決方法: 每層交叉之後，加一組**mux**，在適當的時候顛倒排序結果



波型

- 用iverilog/gtkwave進行simulation

Input資料

```
probe_0[7:0] = 7
probe_1[7:0] = 21
probe_2[7:0] = 29
probe_3[7:0] = 219
probe_4[7:0] = 88
probe_5[7:0] = 163
probe_6[7:0] = 28
probe_7[7:0] = 213
probe_8[7:0] = 0
```

7				219		0
21				213		7
29				163		21
219		88				28
88		28		29		
163		219		0		88
28		0		28		163
213		163		21		213
0		213		7		219

中間結果:
鏡像後大的向上, 小的向下

3 cycle後
排序完成