**Digital IC Design**

**Homework-3**

**Mask**

**學號：N26991784**

**姓名：林宏達**

**１.設計概念與技巧：**

因為資料是由左到右、由上到下依序進入，再加上運算的關係，所以至少存48筆資料在暫存器才可以運算，所以在電路內部設計一個一維［４７：０］大小的暫存器，讓資料可以儲存後再做運算。

由於我是實現是以存４８筆資料再作運算的形式，所以在狀態方面會比之前的ＹｃＢｃＲ還要來的多。其中我在運算方面的除法也就是位移，我是利用拉線的方式來完成。

ＥＸ：ｄａｔａ［ａ］＞＞４→｛４‘ｄ０，ｄａｔａ［ａ｝［７：４］｝

（假設我的ｄａｔａ［ａ］為８ｂｉｔ）

所以每個位移就變成：

mask\_result = {4'd0,data[count4][7:4]} + {4'd0,data[count4+2'd2][7:4]}

+ {4'd0,data[count4+6'd32][7:4]} + {4'd0,data[count4+6'd34][7:4]}

+ {3'd0,data[count4+1'd1][7:3]} + {3'd0,data[count4+5'd16][7:3]}

+ {3'd0,data[count4+5'd18][7:3]} + {3'd0,data[count4+6'd33][7:3]}

+ {2'd0,data[count4+5'd17][7:2]} ;

然後依序的運算為： １－２－３

２－３－１

３－１－２

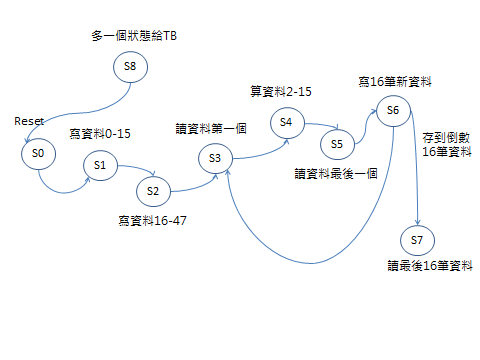
另外，在第２－１５行的開頭跟結尾是不用運算的，所以我就是直接從暫存器讀出值來輸出，所以就寫成：

assign temp1 = data[{read1count[1:0],4'd0}];

assign temp2 = data[6'd15+{read15count[1:0],4'd0}];

這邊因為讀的位置也是連續一直加１６，所以也是用直接拉線的方式來完成。

狀態機如下圖：

****

其中，因為ＴＢ的ｒｓｔ是在第２個ｃｙｃｌｅ才出現，這樣我的狀態已經開始跳了，所以再多一個Ｓ８狀態再Ｓ０前面最後的結果就不會有錯誤。

Ｓ１狀態為：讀０－１５筆資料，輸入直接輸出，順便寫入暫存器。

Ｓ２狀態為：寫入第１６－４７筆資料。

Ｓ３狀態為：讀第２行第一筆資料。

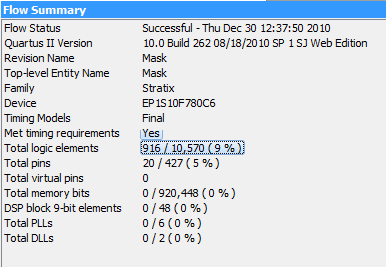
Ｓ４狀態為：運算中間的２－１５筆資料。

Ｓ５狀態為：讀第２行最後一筆資料。

Ｓ６狀態為：寫入新的１６筆資料，這邊會設一個ｃｏｕｎｔｅｒ如果存到倒數１６筆資料的話，下一個狀態就跳Ｓ７，否則就回到Ｓ３繼續讀第一筆資料依此類推。

Ｓ７狀態為：讀出最後１６筆資料直接輸出。

２.Ｐｏｓｔｓｉｍ的電路速度與面積



總面積為９１６個電晶體，電路的操作速度為３７．１