**Digital IC Design**

**Homework-4**

**Image scale-down**

**學號：N26991784**

**姓名：林宏達**

**１.設計概念與技巧：**

作業4的做法跟作業3很像，資料也是由左到右、由上到下依序進入，所以我先設一個大小為256的暫存器，在前面256個ｃｙｃｌｅ先把值存起來，之後再利用６４個ｃｙｃｌｅ把值算出來。

作業４也是設計一個一維［２５５：０］大小的暫存器，利用兩個ｃｏｕｎｔｅｒ數ｃｙｃｌｅ讓狀態執行正確的動作。這次的除４也是利用拉線的方式來表示。

ＥＸ： Ａ＝｛２‘ｄ０，ａｄｄｒ１［７：２｝｝

（其中，ａｄｄｒ１為暫存器內的植）

所以每個暫存器的值＋上除４就變成：

assign addr1 = (scaledown)?data[({counter1,1'd0}+8'd0)]:8'd0;

assign addr2 = (scaledown)?data[({counter1,1'd0}+8'd1)]:8'd0;

assign addr3 = (scaledown)?data[({counter1,1'd0}+8'd128)]:8'd0;

assign addr4 = (scaledown)?data[({counter1,1'd0}+8'd129)]:8'd0;

always @ (posedge clk)

begin

if (reset) begin

A = 8'd0;

B = 8'd0;

C = 8'd0;

D = 8'd0;

end

else if (state==S2)

begin

A = {2'd0,addr1[7:2]};

B = {2'd0,addr2[7:2]};

C = {2'd0,addr3[7:2]};

D = {2'd0,addr4[7:2]};

end

end

這次的作業４原本也是想跟作業３一樣，利用同步下去設計ｒｅｓｅｔ，但後來的ｐｒｅ－ｓｉｍ跟ｐｏｓｔ－ｓｉｍ的ｔｅｓｔｂｅｎｃｈ會有時間誤差，導致我的ａｓｋ拉起來的時間不正確，所以就改成非同步ｒｅｓｅｔ來設計。

狀態機如下圖：

reset

Save

算位置

除4

輸出

Ｓ０狀態為：ｒｅｓｅｔ。

Ｓ１狀態為：將２５６筆資料存入暫存器。

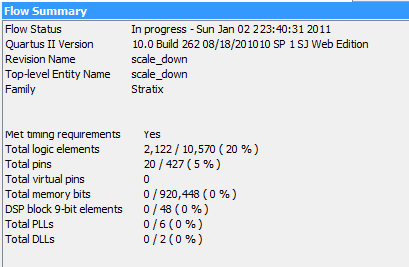
Ｓ２狀態為：算出要運算的位置。

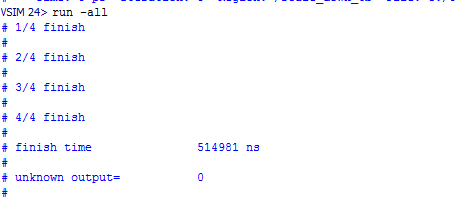
Ｓ３狀態為：將算出的位置存的值除以４。

Ｓ４狀態為：將除以４的值相加後直接輸出。

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２１２２電晶體

完成時間為５１４９８４ｎｓ

電路的操作速度為８．９