**CO project 4 – Pipelined CPU I**

*ID: 0340249*

*Name: 孫聖*

* **Introduction**

這個實驗中我們要完成簡單的Pipelined CPU，具有基本的R-type指令及lw,sw功能。本次實驗添加了Pipe\_Reg來記錄每個階段的所需要保存的值。同時需要調整Test2指令的順序，使之不會發生衝突。

* **Procedure**

先是填充Pipe\_Reg的Port，首先要確定每個階段需要保存的bit的數目，從上到下依次填充，避免錯漏。之後，要將整合了所有資料的wire重新譯碼成能夠使用資料。唯一需要注意的是，在ID階段Register中WriteRegister和Writedata都是從WB階段獲得的，而不能直接使用ID階段的資料。

調整指令順序時，只需要將有衝突的指令間隔兩個指令即可。

* **Results**

能通過第一個測資，且第二個調整完順序後與原始所算出的答案相同。

* **Conclusion**

這次實驗所需要添加新的模組較少，但是調試各個Pipe\_Reg所需的過程十分繁瑣，而且容易出錯。由於是Pipelined CPU, Decoder, ALUCtrl和ALU的Default值也要做相應的修改。在調整指令順序時發現這個CPU只需要將衝突的指令間隔2個指令即可。但老師上課所說這樣需要更加高級的機制，普通的FF要間隔3個指令才能避免衝突。在這方面有所不解。