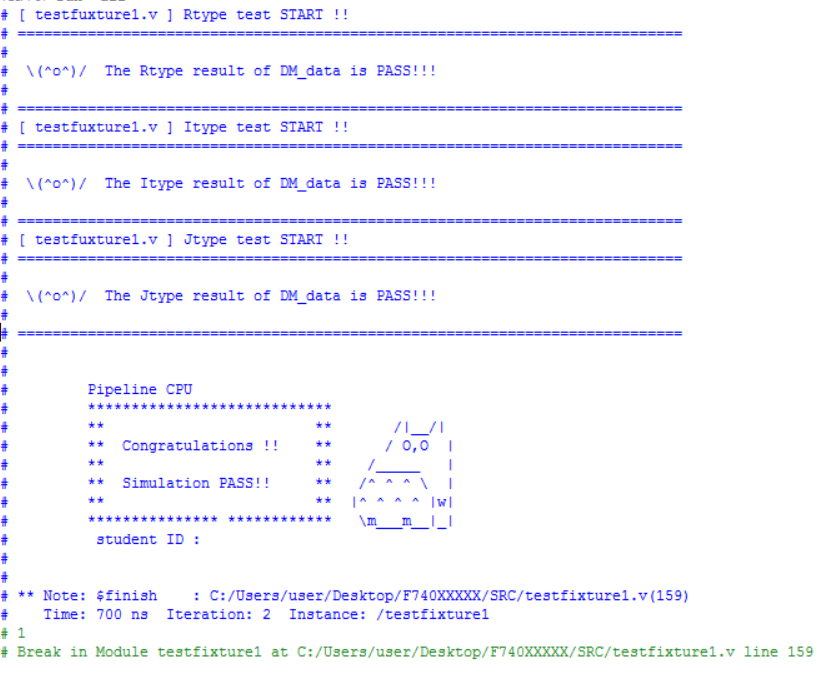
**Computer Organization 2017**

**HOMEWORK III**

系級: 資訊108 學號: F74042117 姓名: 梁定能

**實驗結果圖(snapshot of the results)**

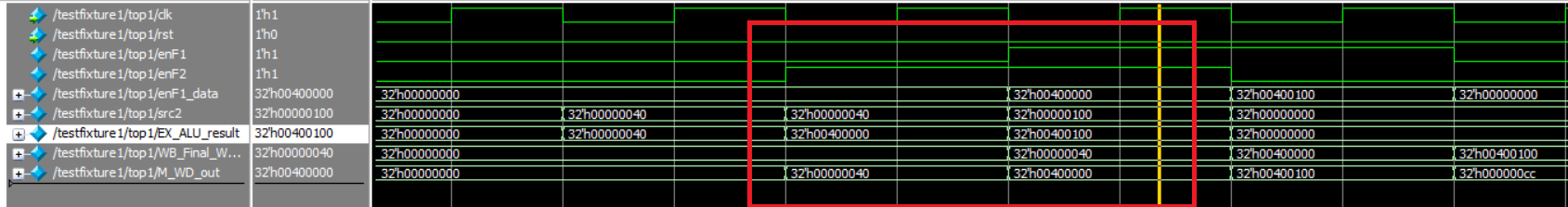


**指令波型圖(Snapshot of the instruction execution waveforms.)**

(Please explain why your snapshot is correct, including the wires, signals. The description should be as detailed as possible, e.g. why this situation occurs, and in waveform where does it occur?)

1. Instruction with Forwarding

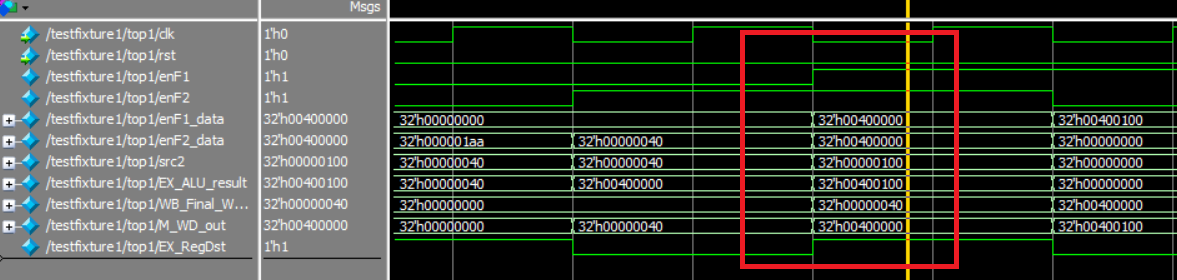
R-type:



Description:

當enF1或enF2為1時，及代表forwarding發生了，而enF1\_data的值就是從m\_wd\_out傳回的，傳回後座add的動作放入ALU\_result

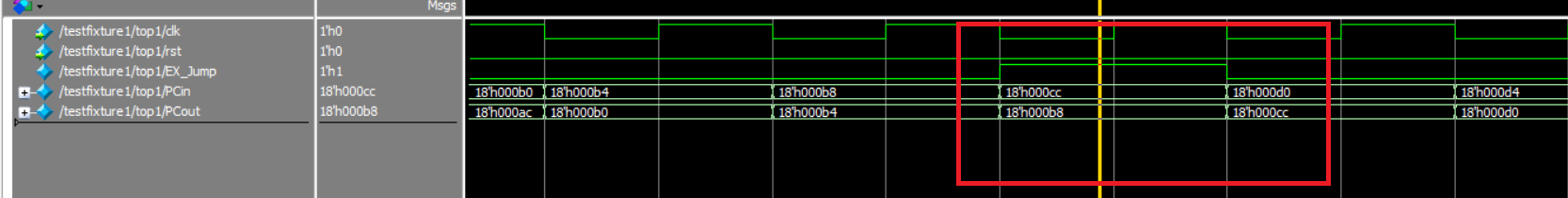
I-type:



Description:

當regdst為1時為I type，enF1 跟 src2 為ALU的input ，enF1其值為M\_WD\_out而來，而src2為imm之延伸而來，因為當enF1或enF2為1時為forwarding。

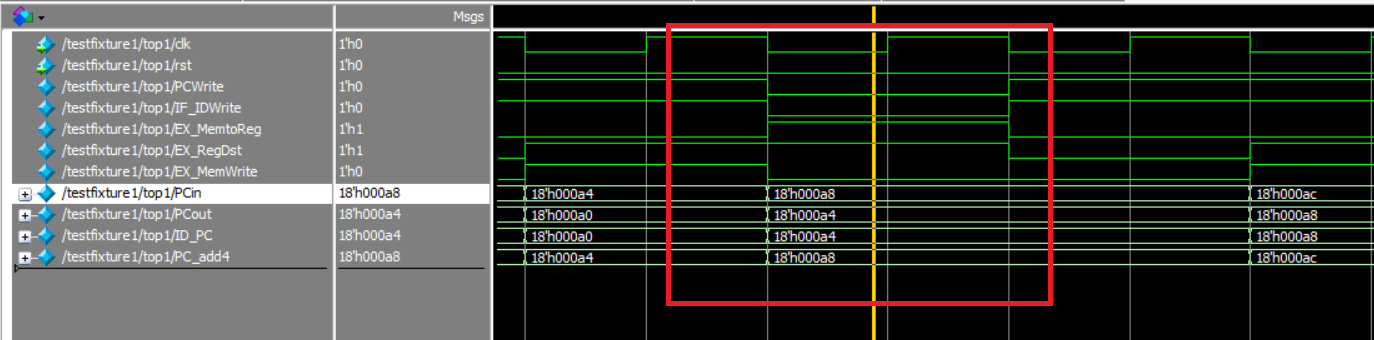
J-type:



Description:

在EX\_JUMP唯一之前PC理論上每經過一個cycle是加上4，因此如上圖在紅框之前PCOUT跟PCIN都是一值加上4，在執行JUMP後，PCIN立刻改為要JUMP的ADDRESS。

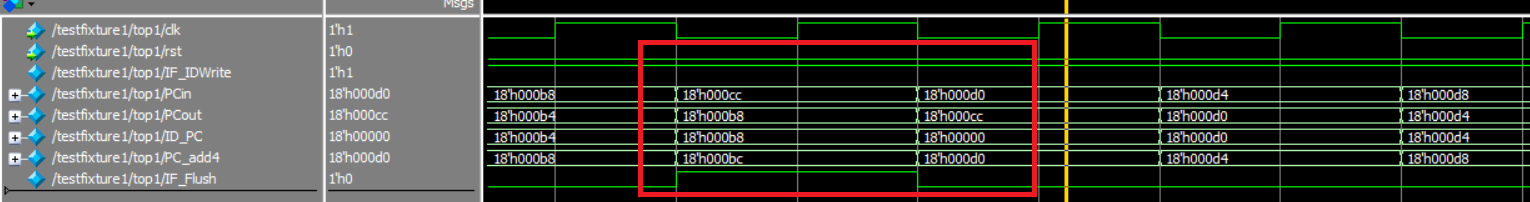
1. Load Stall:



Description:

當發生load stall時，if\_id\_write跟pcwrite皆不能被寫入(電路圖上IF\_IDWRITE跟PCWIRTE從1->0)，就是電路圖上的PCIN~PC\_ADD4的相同值將重新再跑一次CYCLE，並且ex中的control像是memwrite等等皆會歸0

1. Branch Delay (& Flush):



Description:

當發生branch時，在if-id之間的flush設為1，當其為一時，if的pc將delay一個cycle，並在ID-PC設為0，然後在下一個CYCLCE，IF-PC(PC\_add4)才將值放入ID-PC

If you CPU data is from Figure 5 in the home, show your CPU datapath and explain why you want to design your CPU this way.

Reason:

**心得(Report)**

(請寫下完成本次作業的心得、學到哪些東西、困難點的部分。大約   
 100~200字 )

(Please write your learned lesson and conclusion, and difficult point. About   
 100~200 words)

在HDU的判斷很容易寫錯，為此吃了不少的苦頭，同時因為在TOP的線太多了，花了很多的時間在接線，看到很眼花，而且很容易接錯線，造成無法simmulation，而且同時也須讓自己熟悉如何看電路圖debug，最困難的點，我認為是在要清楚知道每個指令的線之間的關西，像是何時出現了stall，這個問題就是需要熟讀講義與課本才能迎刃而解。