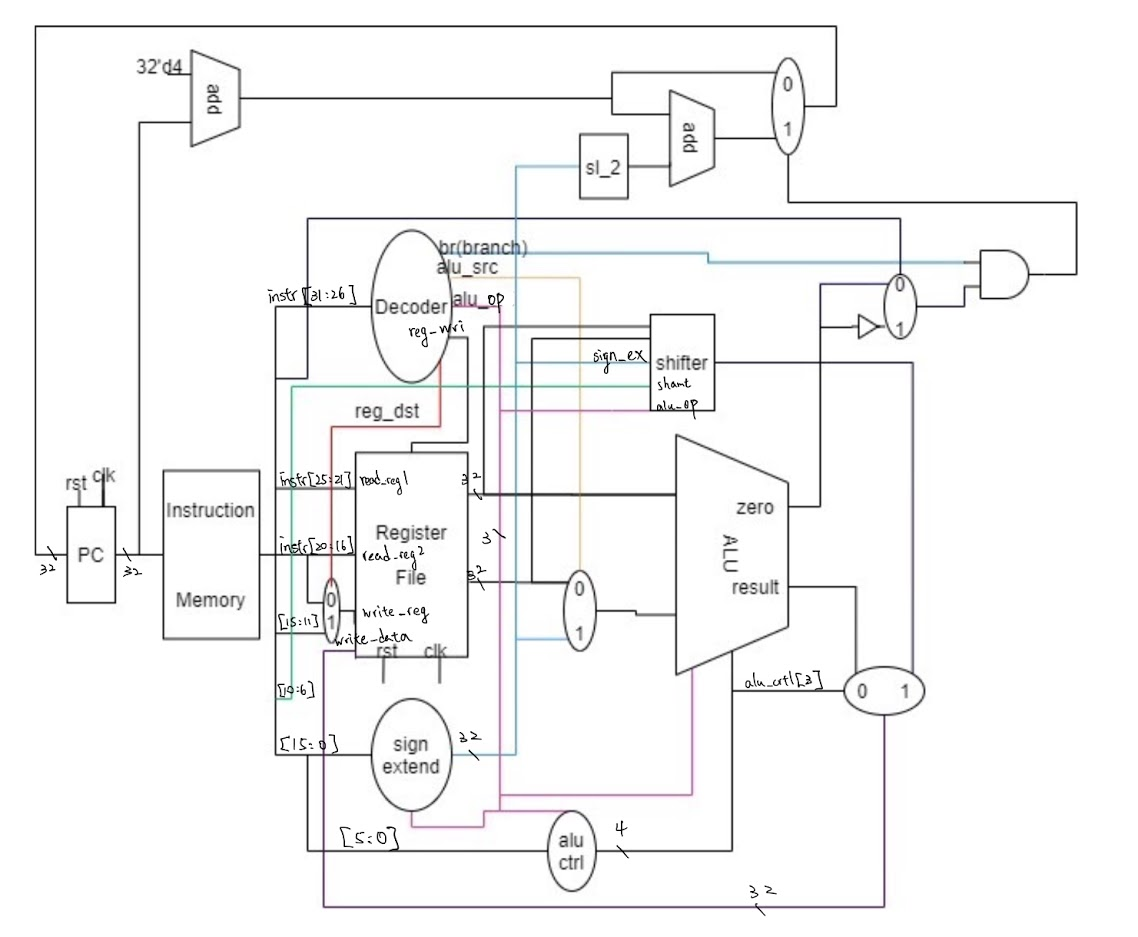
**Lab2\_Report**

0710028陳敬諺

* Architecture diagram:



* Module description:

Adder1:

執行pc+=4

Mux\_Write\_Reg:

由reg\_dst決定write\_reg的位置

Decoder:

將opcode轉換至對應的{RegWrite , ALU\_op , ALU\_src , Reg\_dst , Branch}

AC(ALU\_Crtl):

由ALU\_op , function(instructions[5:0])決定alu\_ctrl

SE(Sign\_Extend):

將instructions[15:0]變更為32bits，並用alu\_op判斷是sign ext./zero ext.

Mux\_ALUsrc:

用ALU\_src決定ALU的src2，0為rt輸入，1為sign\_ex輸入

ALU:

32bits ALU，原則上沿用lab1的架構，運算結果由alu\_ctrl控制，alu\_ctrl[3]=>A\_invert，alu\_ctrl[2]=>B\_invert，alu\_ctrl[1:0]=00為and、01為or、10為add、11為slt。另外利用alu\_op判斷sltiu的情形，當遇到sltiu運算，取第32個bit的cout跟1再做一次add運算才是真正的less\_set結果。

shifter32:

以shift\_ctrl(alu\_ctrl)決定shift的型態，shift\_ctrl=1000為lui，1001為sra、1010為srav

Shifter\_1\_2:

執行 ( n << 2 )

Adder2:

執行pc += ( 4+ ( n << 2 ) )

equal\_confirm:

比較bne、beq的opcode，在opcode[0](instruction[26])會有差異，所以用opcode[0]作為select，用zero\_result作為alu\_src1與alu\_src2是否相同的依據。

and\_gate to pointer:

以branch控制要不要輸出equal\_confirm的結果

Mux\_PC\_Source:

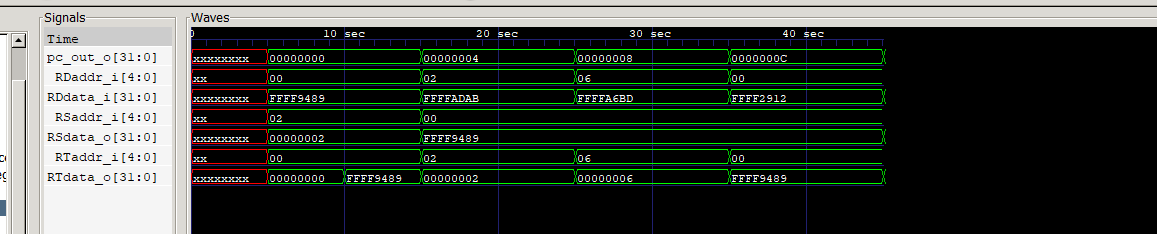
當pointer結果輸入會辨別要不要以adder2的結果回傳給PC

result\_confirm:

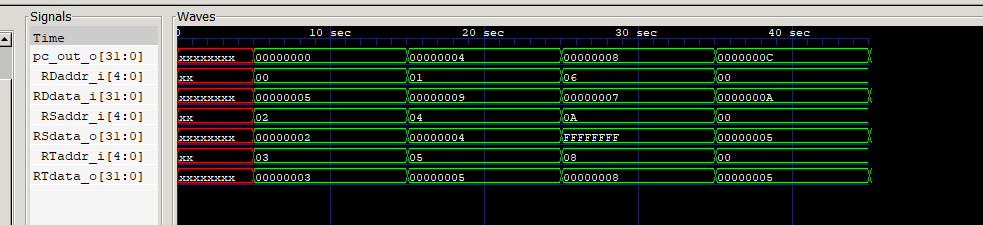
用alu\_ctrl[3]控制，1為回傳shifter\_result到write\_data、0為回傳alu\_result到write\_data

* Waveform:

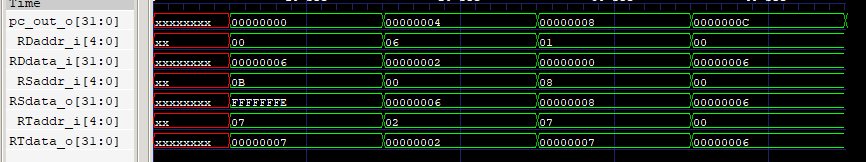
addi



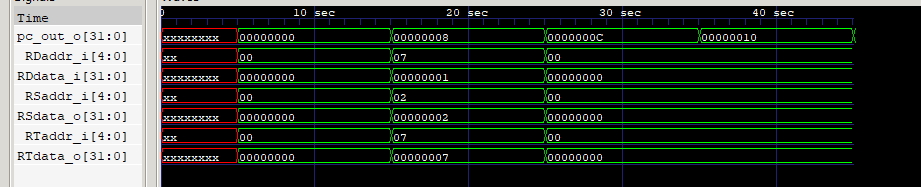
Addu



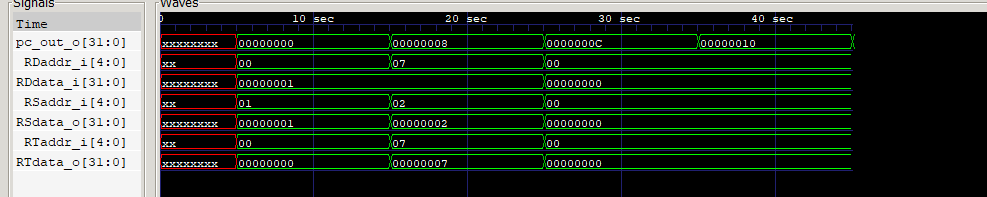
and



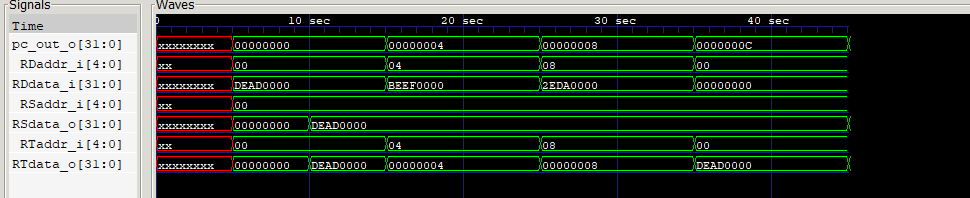
Beq



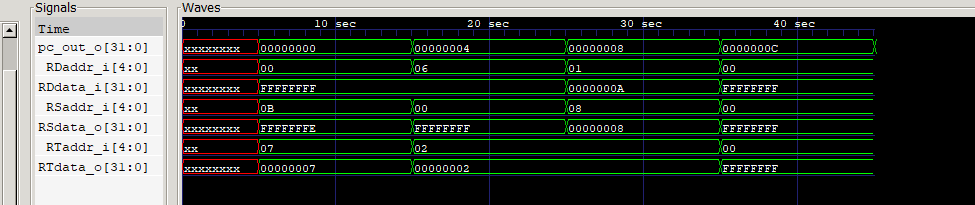
Bne



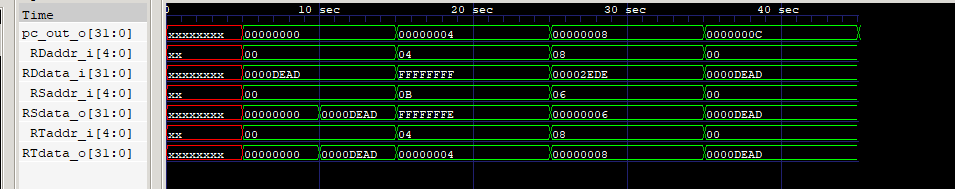
Lui



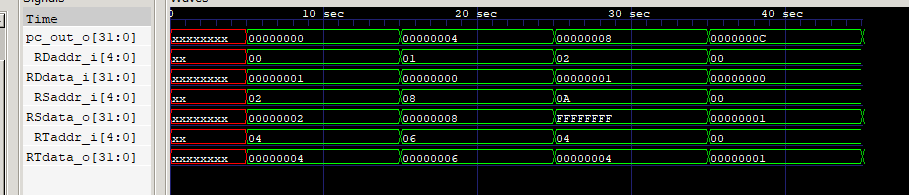
or



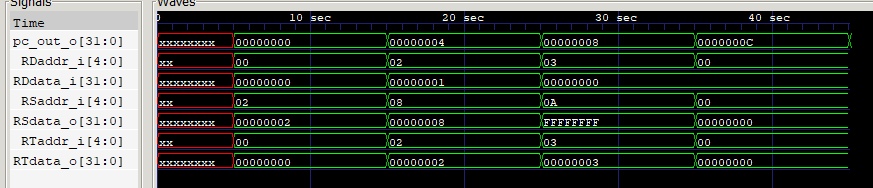
Ori



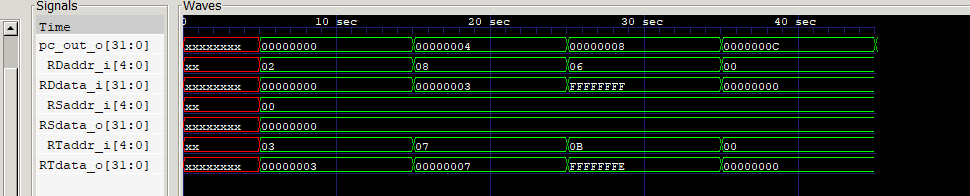
Slt



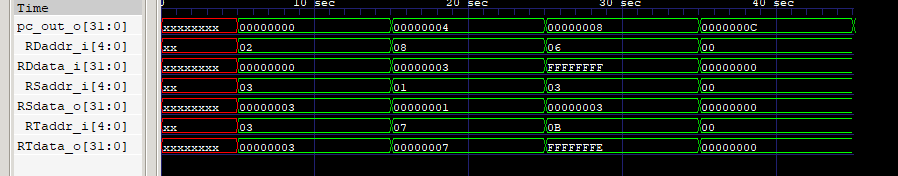
Sltiu



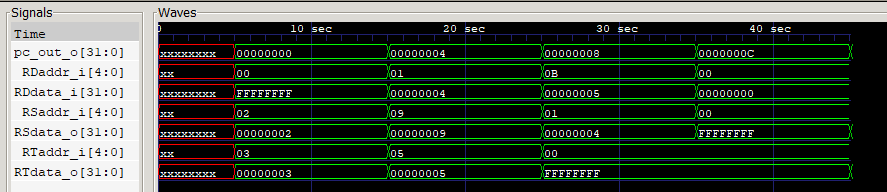
Sra



Srav



Subu



* Questions:

1. What is the difference between "input [15:0] input\_0" and "input [0:15] input\_0" inside the module?

若是前者 input\_0[0] 會是最右邊的位數，而後者 input\_0[0]會最左邊的位數

1. What is the meaning of "always" block in Verilog?

當指定參數條件符合或是設定任何在always block的參數改變，即執行一次always block裡的程序

1. What are the advantages and disadvantages of port connection by order and port connection by name in Verilog?

Port connection by order:

Advantage: 較簡潔，當module的參數與外部參數命名一樣時，可直接套用

Disadvantage: 容易混淆，debug時可能會造成不便

Port connection by name:

Advantage: 架構清楚，且不用對應原先module設定的順序，在trace和debug上較有優勢。

Disadvantage: 若module參數名稱與原先建立的不相符(錯字)，執行程序就會出錯。

* Contribution:

Most of architecture、debug

* Anything to share:

這次lab我發現自己有個壞毛病，在實行alu\_ctrl之類的功能時，我會習慣直接輸入數字代碼，而非利用localparam之類的狀態機參數定義，導致後面在debug的時候，trace不易，要看數字去對應到功能花了不少時間。