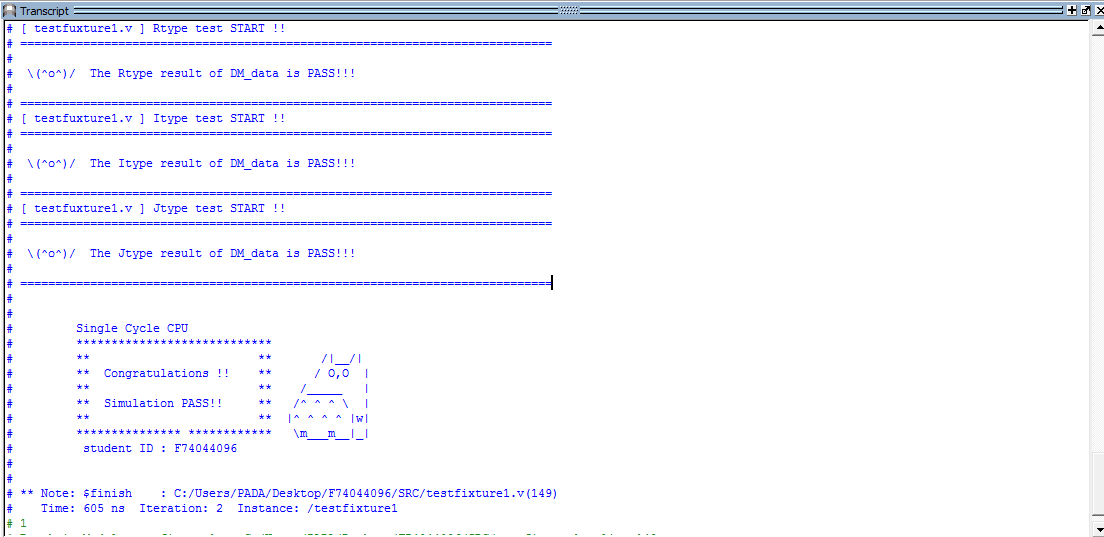
**Computer Organization 2017**

**HOMEWORK II**

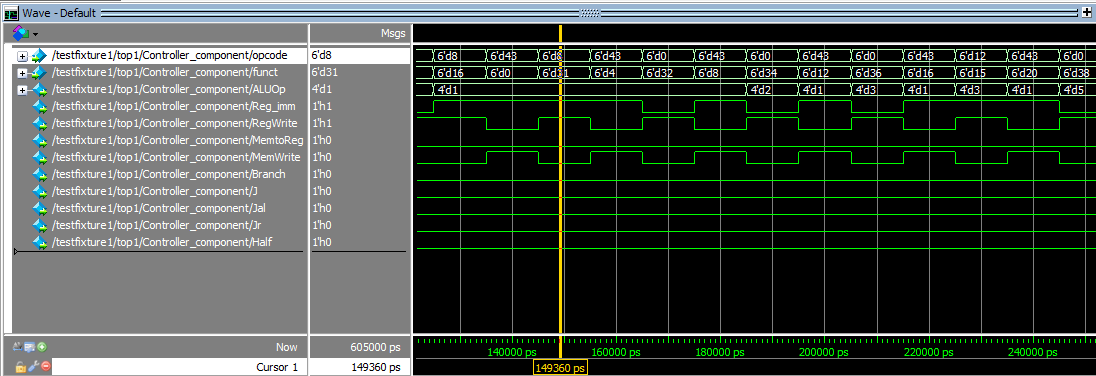
系級: 108 學號: F74044096 姓名: 吳元智

**實驗結果圖(snapshot of result)**

****

**指令波型圖( Instructions waveform )**

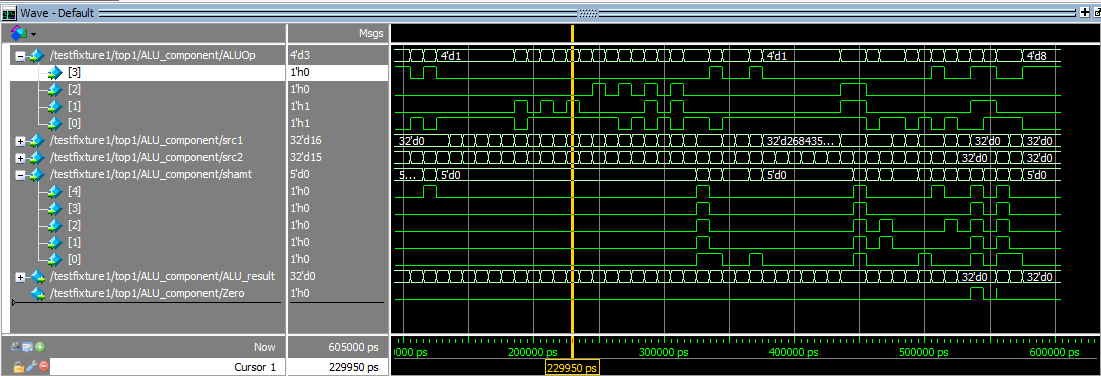
(Please explain why your snapshot is correct, including the wires, signals.)

Controller:

Reason:

由上圖可看出opcode是001000，funct是011111，由此可以推斷出是執行addi指令，因此ALUOp回傳值 = 1，並將其值丟到ALU裡面進行add(op\_add = 1)，另外Reg\_imm = 1代表src2是要讀取imm而非Rt，Regwrite = 1則代表要從Register讀取資料，並將這個訊號丟給Regfile去做讀取的動作

至於MemtoReg(是否從Memory 寫入Register)、MemWrite(是否寫入Memory)、Branch(進行Branch的判斷)、J (是否Jump至指定Register)、Jal(是否讀取PC)、Jr(是否Jump至指定PC)、Half(判斷讀取32bit完整資料或是後16bit資料並做sign\_extend)，因為並不會執行以上的動作(ex : 從Memory 寫入Register，Branch…)，所以其訊號為0

ALU

Reason:

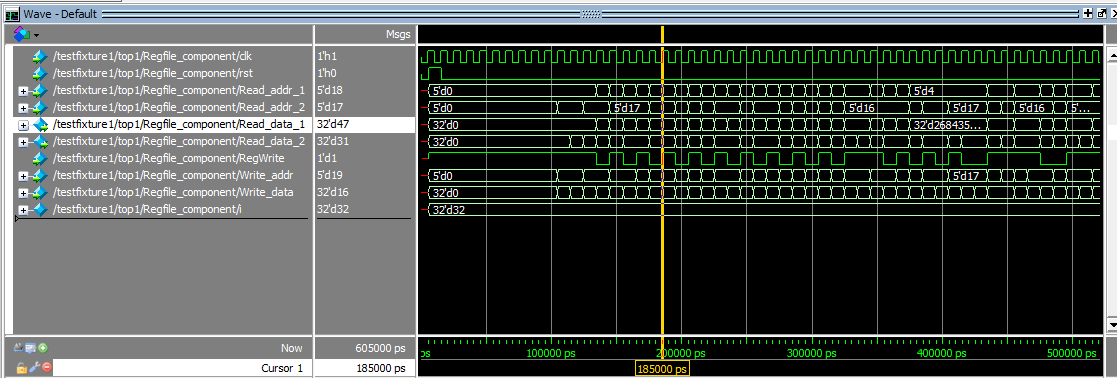
由上圖可看出ALUOp = 3，也就是執行and指令(op\_and = 3)，

src1 = 16 -> 00…010000

src2 = 15 -> 00…001111

經過計算後(src1 & src2)得到ALU\_result = 0 -> 00…000000

而and指令並不需要用到shamt(左移或右移幾bit)、Zero(回傳是beq或bne)所以這些wire及訊號為0

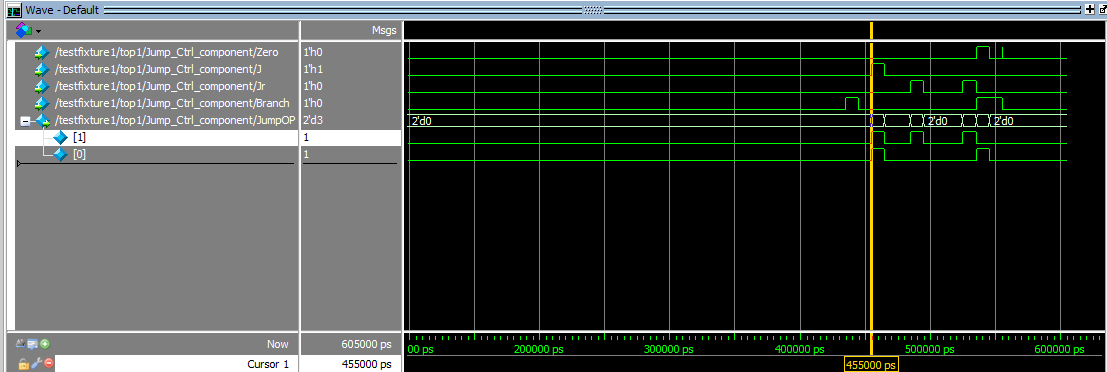
Regfile

Reason:

根據上圖可看出第1個Register是$18，第2個Register是$17，而最終輸出Register則是$19，由此對應IM\_data，其正在執行

02519822 //sub $19, $18, $17 ; 15: sub $s3, $s2, $s1 # $s3 = 47 - 31 = 16 (10000)

而$18的資料確實是47，$17為31，$19 = $18 - $17 = 16，而RegWrite = 1代表要從指定Register讀取資料，rst = 0，則代表要把資料輸入Write\_data，clk為計時用，至於i則是迴圈所用到的變數

Jump\_Ctrl

Reason:

由上圖來看，Zero、J、Jr、Branch皆為訊號，而JumpOP則是傳值給Multiplexer決定跳至哪一個Register，而在code裡面，我設定若是(Brranch && Zero) -> JumpOP = 1(進行Branch)，若無(Brranch && Zero)，則判斷Jr -> JumpOP = 2(進行Jr)，若以上兩個訊號都沒有才判斷J -> JumpOP = 3(進行Jump)，若三個訊號都沒有，則JumpOP = 0(不Jump)

CPU datapath (If your CPU datapath is different from HW2 Fig.2.)

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)

關於這次的作業，我對於CPU的運作結構，及其讀取資料並執行的路徑也更加了解，像是Controller掌管著大部分MIPS指令的輸入，再傳出相對應的訊號給各個元件，並開始依序執行資料的讀入，運算，最後再將其傳回給對應的Register，另外Sign\_extend也是藉由之前上課所學到的東西來時做出來，這次的作業不僅僅只是實作一顆Single\_Cycle\_CPU，也有複習到之前所學的東西

至於這次我遇到最大的難題就是如何處理lh及sh指令讀取資料，因為lh是先讀取資料的後16bit，之後再做Sign\_extend，這樣出來的答案才是正確，所以在Controller裡面我多加了一個訊號來確認是lw還是lh，再在top裡面加入multiplxer來判斷