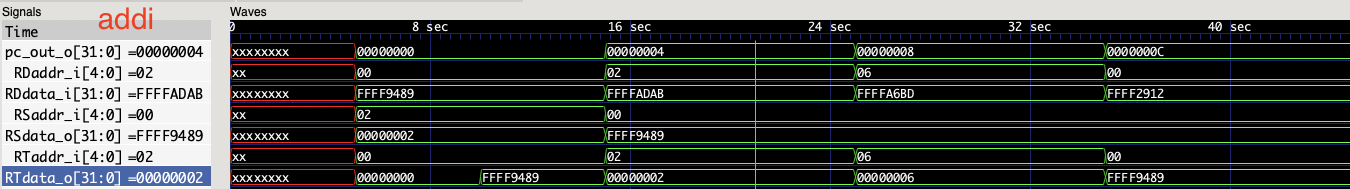
**Computer Organization Lab 2**

Simplified Single-cycle CPU

0716055 王耀德

* Architecture diagram:



* Module description:
  1. Adder : 對PC作加法 (PC+4 (Adder1) 或branch (Adder2) )。
  2. ALU\_Ctrl : 控制ALU要做什麼運算。
  3. ALU : 對傳入的資料執行各種運算(or, add, shift …)。
  4. Decoder : 透過Instr\_memory傳來的op\_code控制Reg\_write、Reg\_dst、ALU\_src與ALU\_op，決定要執行的操作。
  5. Instr\_Memory : 將32-bit Instruction分別傳進各個module。
  6. Mux\_2to1(write\_reg) : 判斷src1是否要write。
  7. Mux\_2to1(ALU\_src) : 判斷是reg2或sign-extend的部分進入ALU。
  8. Mux\_2to1(PC\_sourse) : 判斷PC接下來是PC+4或是Branch的結果。
  9. ProgramCounter : 紀錄現在指令在Instr\_Memory的位置。
  10. Shift\_Left\_two\_32 : 對Sign\_Extend後的32-bit左移2位達到x4的效果。
  11. Sign\_Extend : 把Instr[15-0]做sign extention到32-bit。
  12. Single\_Simple\_CPU : 把上述的mudule依照目標連接起來。
* Waveform:
* Questions:
  1. What is the difference between "input [15:0] input\_0" and "input [0:15] input\_0" inside the module?

input [15:0] input\_0 的index是降冪排列，依序由input\_0[15] 排列至input\_0[0]，而input [0:15] input\_0 則相反，是升冪排列。

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

"always" block表示當()內的訊號改變時，會執行block內部的指令。

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

by order：優點是可以打比較少字，缺點是如果module port很多或是做過修改，不容易閱讀且比較難找出新增或移除的參數。

by name : 優點是清楚知道參數對應的值是什麼，缺點是要打比較多的字。

* Contribution:

我負責寫Decoder還有測試testbench與debug，我的組員敍溶負責寫其餘module，並一起完成ALU、連接各module。

* Discussions:

一開始時有點無從下手，尤其是有些指令沒有看過，仔細了解過後可以發現只要分清楚是r-type還是i-type、要branch或是load之後，就比較知道怎麼下手了。測試時在sra和srav的r6結果卡比較久。後來發現會被辨識成unsigned的表示，所以就改成放進陣列，就成功表示出shift right arithmetic了。這次寫起來因為不用像lab1用1bit ALU慢慢組合，寫起來比較不會有憤怒的感覺。