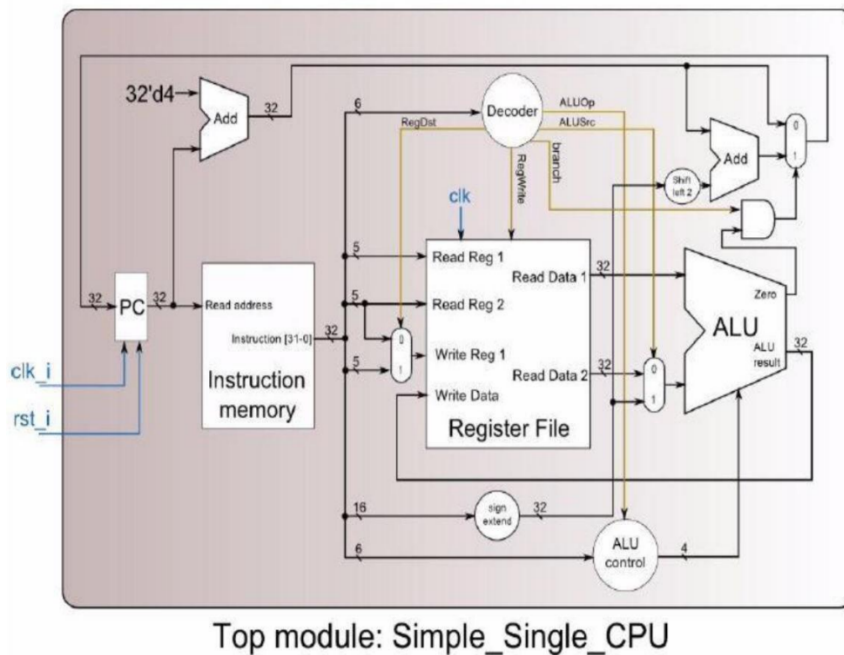


Architecture diagrams:



Hardware module analysis:

- (1) PC: 要執行程式的下一個指令位址
- (2) Full_adder.v: 1bit的加法。
- (3) Adder.v: 利用Full_adder.v, 拼出32bits的adder。
- (4) ALU.v: 根據ctrl_i的值來決定要做哪類基本運算。
- (5) ALU_Ctrl.v: 處理ALUOp_i和funct_i這兩個output
- (6) Decoder.v: 藉由instr_op_i, 判斷其他的輸出應該要為多少。
- (7) Shifter_left_Two_32: 往左shift兩個bits。
- (8) Sign_Extend.v: 將16bits轉32bits。
- (9) Simple_Single_CPU.v: 把需要用到的線接上。

Finished part:

執行結果兩筆測資皆正確。

# run 1000ns		# run 1000ns	
	2r0= 0		2r0= 0
r1= 10		r1= 1	
r2= 4		r2= 0	
r3= 0		r3= 0	
r4= 0		r4= 0	
r5= 6		r5= 0	
r6= 0		r6= 0	
r7= 0		r7= 14	
r8= 0		r8= 0	
r9= 0		r9= 15	
r10= 0		r10= 0	
r11= 0		r11= 0	
r12= 0		r12= 0	
\$stop called at time : 260 ns		\$stop called at time : 260 ns	

Problems you met and solutions:

- (1) 要仔細看圖中CPU的線如何接
- (2) debug 很多問題, 有時候稍微改一點就錯了
- (3) 寫作業的途中, 很常不確定自己所寫的部分是否正確。

Summary:

雖然離繳交日很久了, 但原本很害怕寫verilog, 花了很久的時間才慢慢完成, 但對於CPU的運作更有一點概念了。