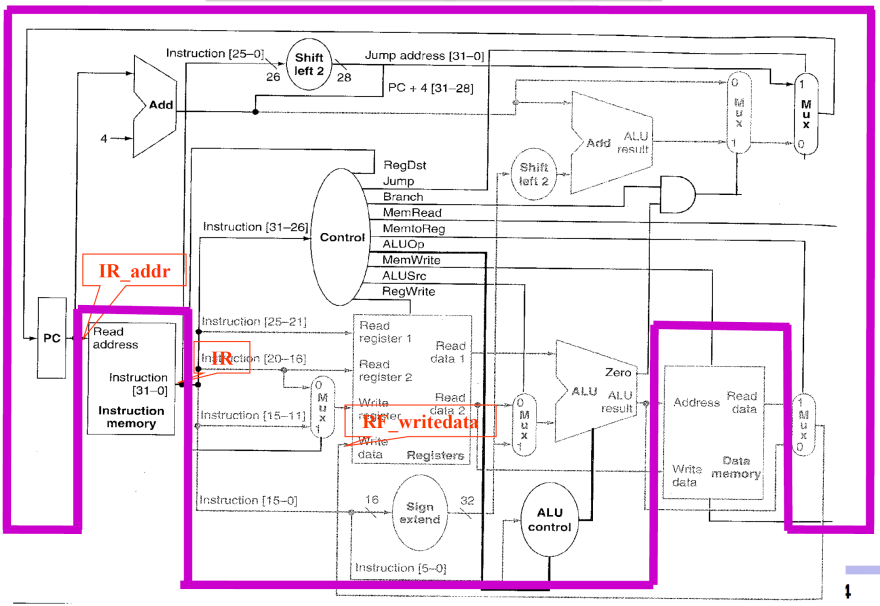
Single MIPS Report

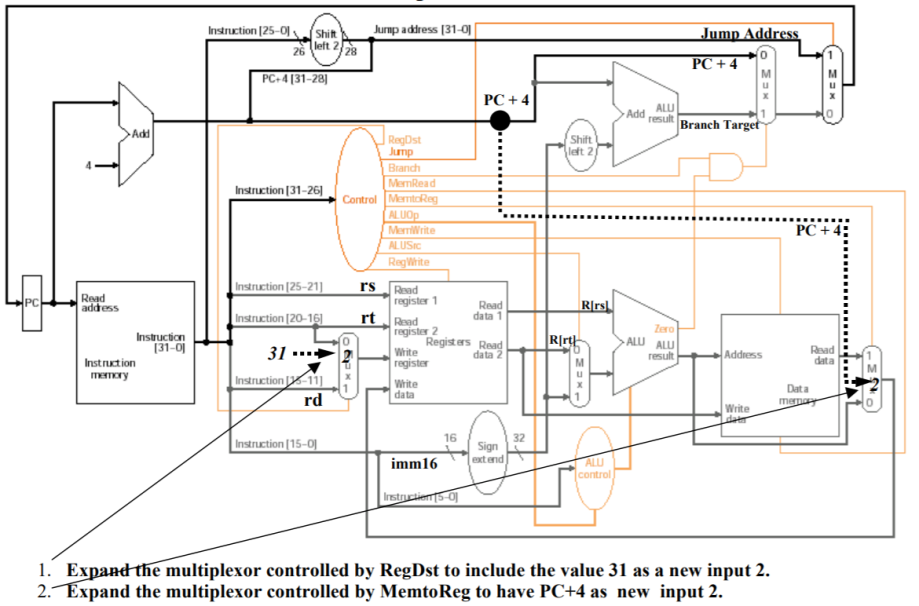
姓名：鐘民憲 學號：B06901017 系級：電機二

一、硬體設計

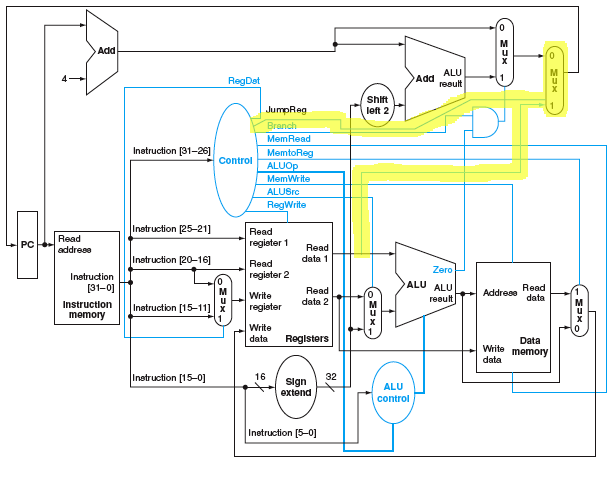


硬體基本上就是按照此圖設計，不過由於作業要求support的instructions包含R-type(add, sub, and, or, slt)、lw, sw、beg以及j, jal, jr，而此設計並沒有辦法執行jal與jr，故做了以下兩個修改：

jal：(control signal also need to consider this condition)



jr：(control signal also need to consider this condition)



與圖片稍有不同，我設計的方式是將MUX\_Jr與MUX\_Jump合併在一起，變成一個3-to-1的MUX，然後control signal “Jump”也改成2-bits

二、程式設計

Verilog程式的部分我分成兩個module去寫，一個是SingleCycle\_MIPS，基本上囊括了所有東西，另一個是register\_file，原本的寫法是直接將register\_file另成變數reg [31:0] register [31:0](reg的陣列)，但是simulation發現一直會出現error: virtual memory limit exceeded，於是就將它獨立出來寫成一個block，雖然基本上是相同的東西，但是這樣就不會出現error了。

此外，我一開始在寫的時候將所有的變數都令為reg並把所有combination circuit都寫在always @(\*)裡面，想說這樣code看起來比較一致，感覺也沒什麼問題，但是我發現這樣做程式執行會出錯，在看nWave時明明就是直接連接的兩個reg，理論上在同個時刻顯現出來的數值要相等，但是結果卻會有所不同，所以我將許多變數改成了wire，只要不牽涉到if…else或case，都一律改成wire，這樣就能保證數值會確實傳遞。

三、執行結果

