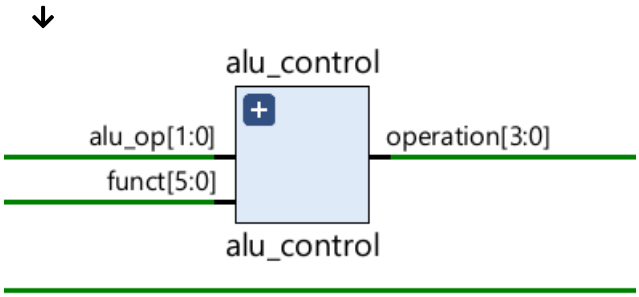
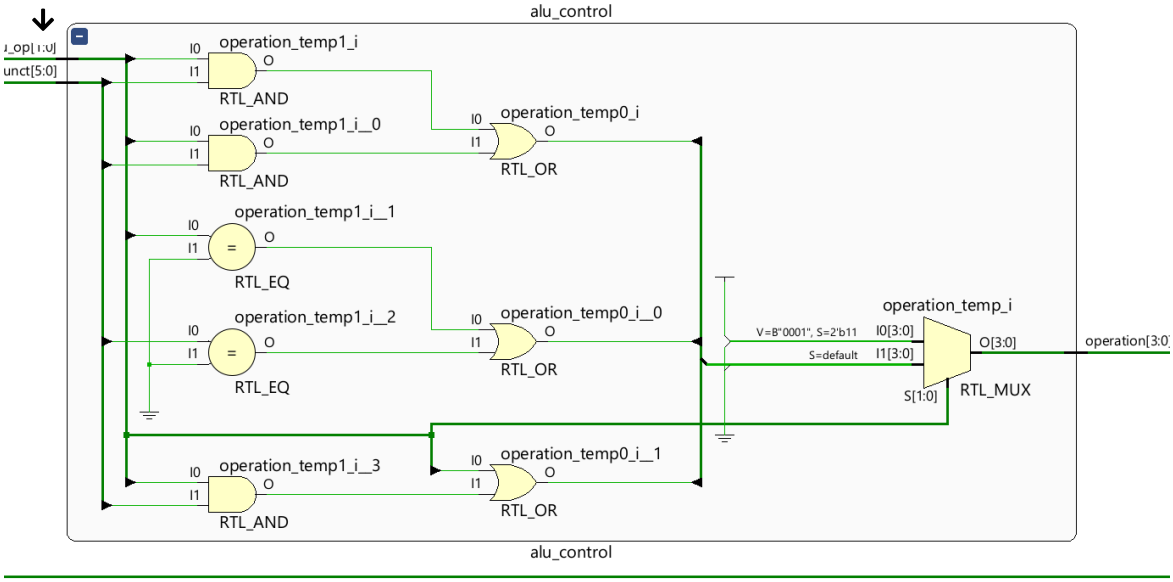
tags: `2024 年 下學期讀書計畫` `Computer Organization`

Lab2_111652017

1. Architecture Digrams

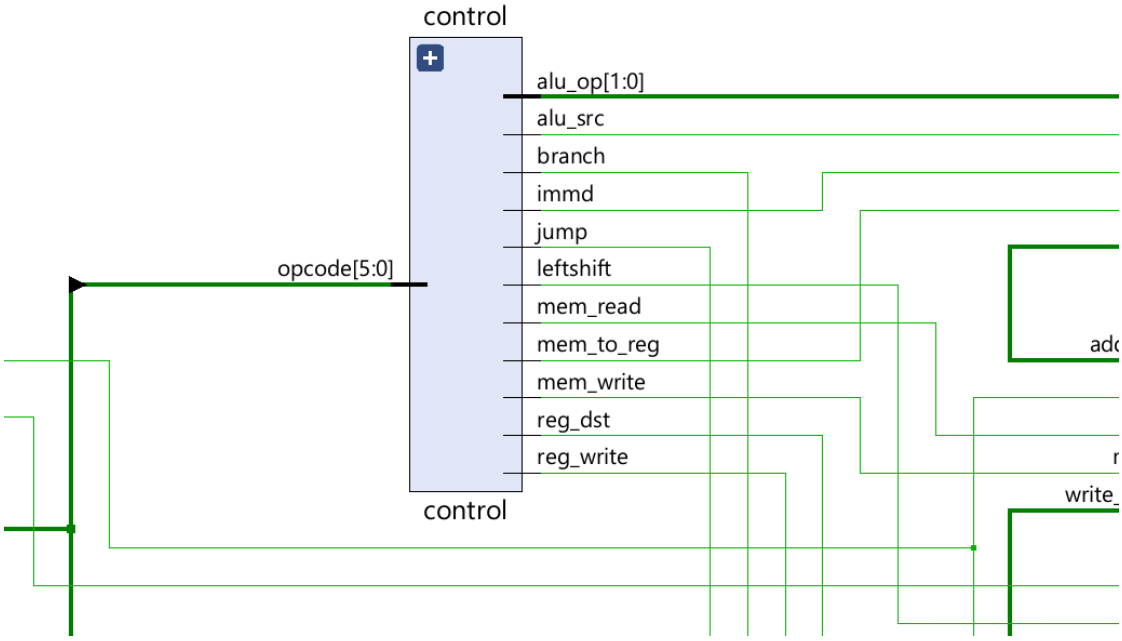
ALU control



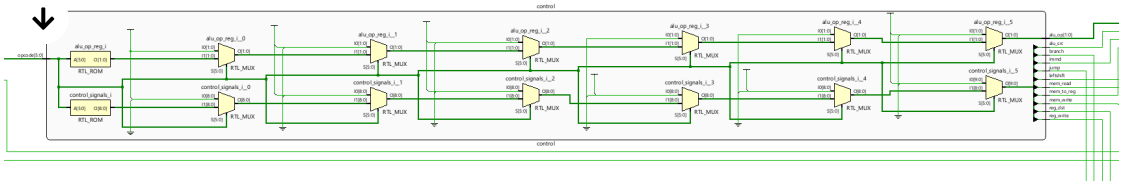


main control

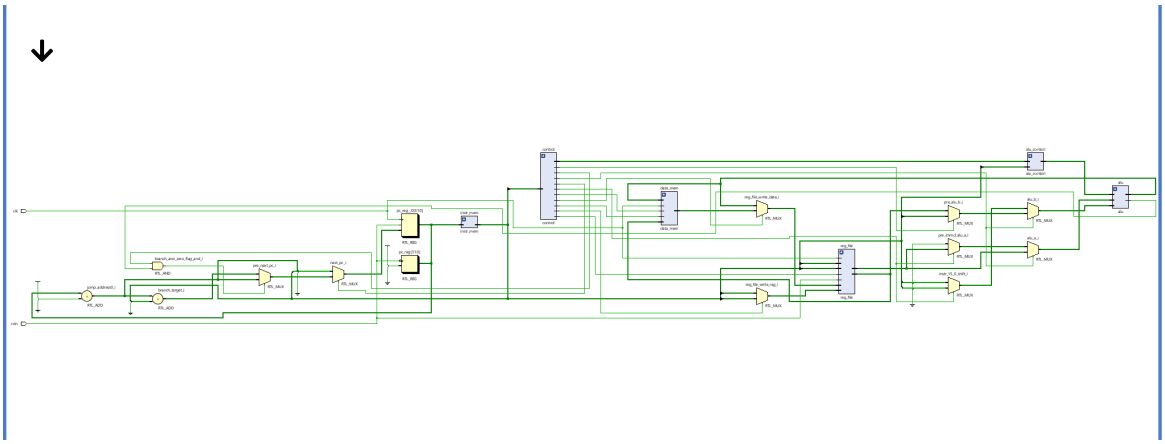
- 大架構



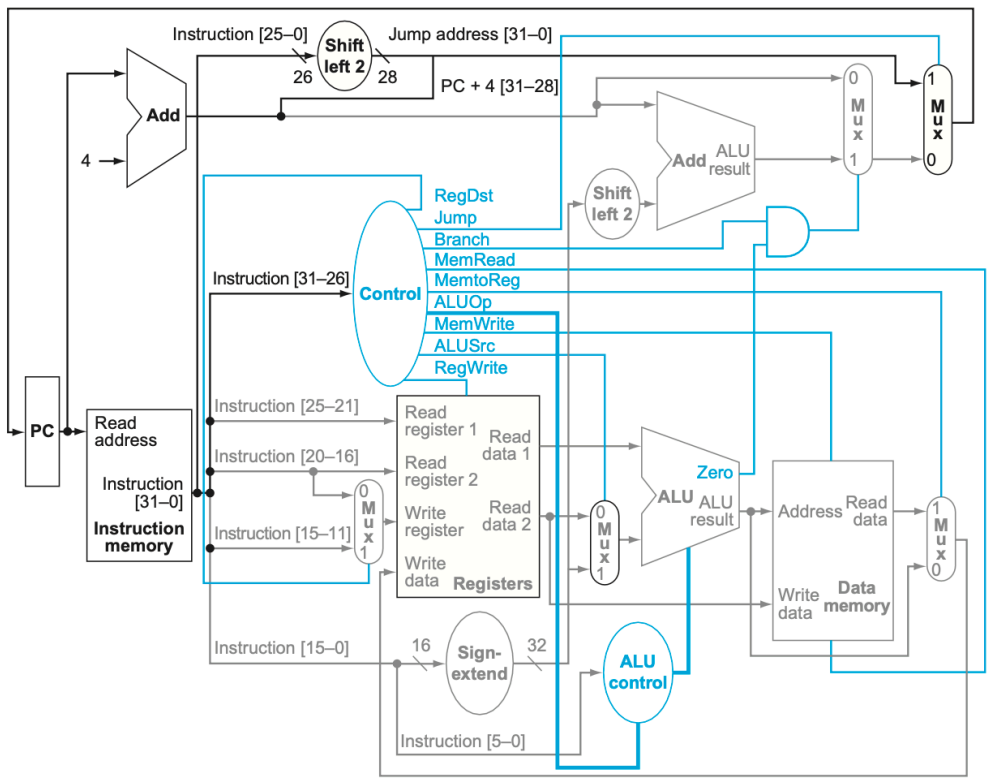
- 小架構



single-cycle processor



explain



- 大致上是按照這張圖做
- 但在實現 lui 和 ori 的時候, main control 需要多加兩個 controls, immd 和 shiftleft
- immd == 1 代表現在是 lui 或 ori operation
- immd == 1 代表現在不是 lui, 也不是 ori operation
- immd == 1, shiftleft == 1 代表上面那種情況
- immd == 1, shiftleft == 0 代表下面那種情況

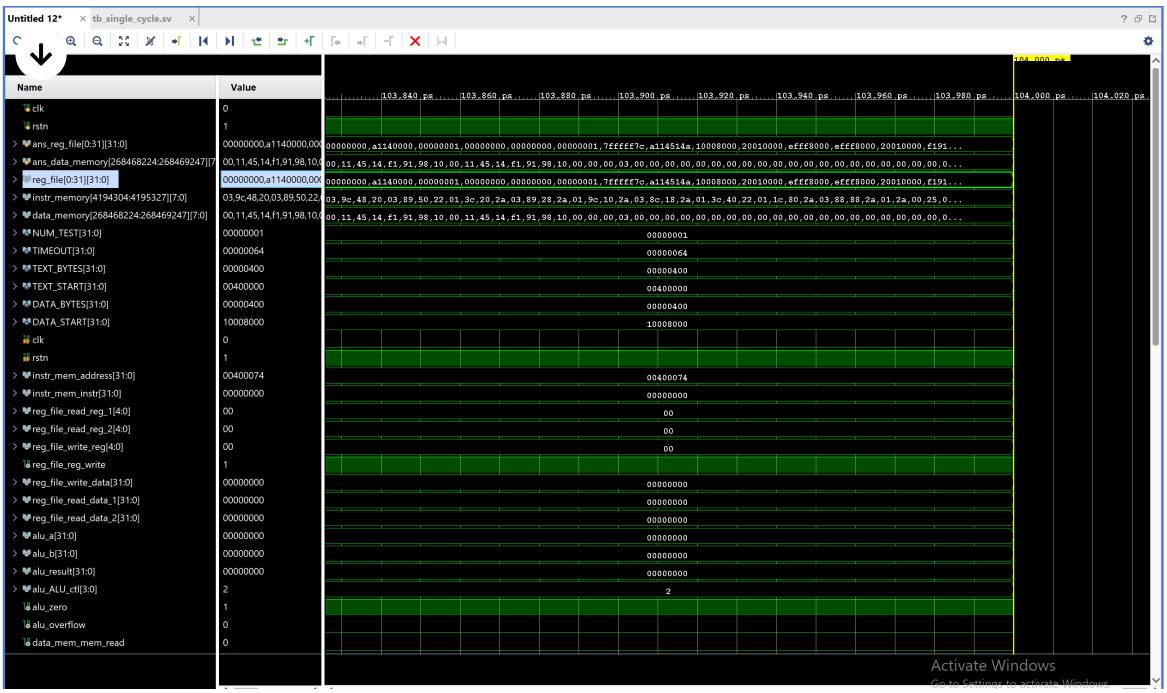
```
rt = { immd , 16'b0 }
```

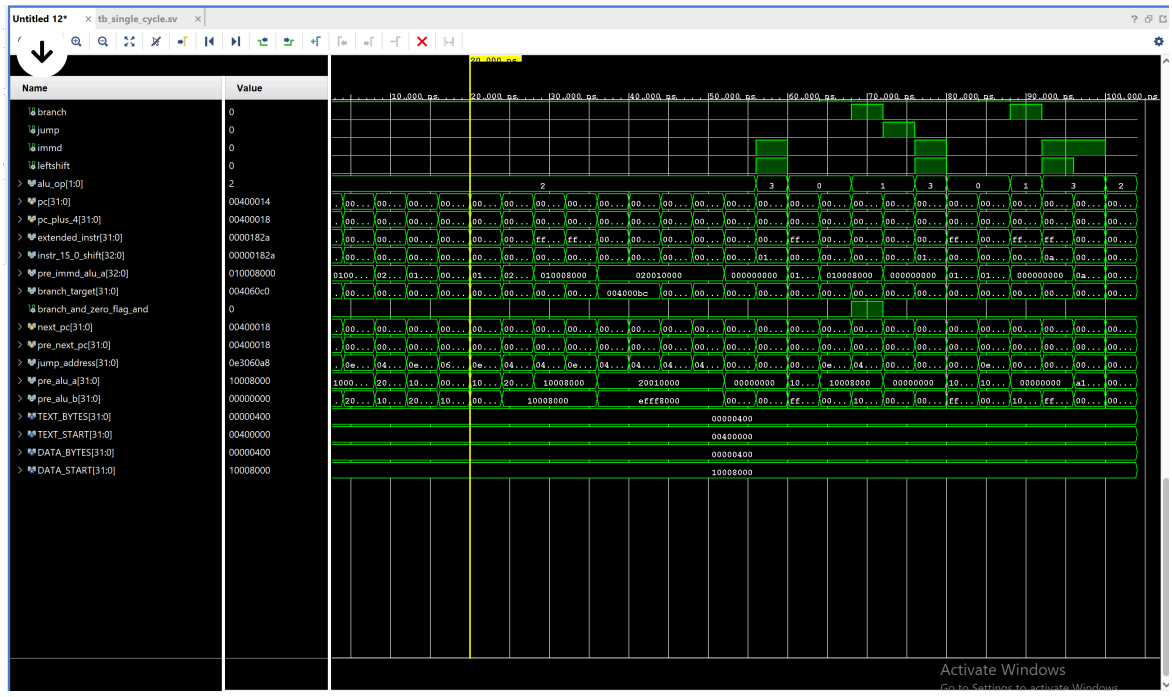
```
rt = rs | { 16'b0 , immd }
```

- 在實做 alu_a, alu_b, 會多兩個 MUX
- 因為現在 alu_a 有可能是 32'b0
- (在上面的第一種情況, 我們可以把 rt 看成 32'b0 和 {immd, 16b'0} 做 or)
- 因為現在 alu_b 有可能是 {immd, 16b'0} or {16b'0, immd}

2. Experimental Result

waveform





other test case

- infinite loop
- beq $0, $0, 8

3. Answer the following Questions

(1) When does write to register/memory happen during the clock cycle? How about read?

- write to register: (reg_write control == 1 and posedge clk), or negedge rstn == 1

- write to memory: posedge clk and mem_write control == 1
- read to register: read_reg_1 control == 1 or read_reg_2 control == 1
- read to memory: mem_read control == 1

(2) Translate the "branch" pseudo instructions (blt, bgt, ble, bge) in the Green Card into real instructions. Only at register can be modified, and other common registers should not be modified.

- blt $rs, $rt, Label;
  - slt $at, $rs, $rt # Set $at to 1 if $rs < $rt, else 0
  - bne $at, $zero, Label # Branch to Label if $at != 0
- bgt: $rs, $rt, Label;
  - slt $at, $rt, $rs # Set $at to 1 if $rt < $rs, else 0
  - bne $at, $zero, Label # Branch to Label if $at != 0
- ble: $rs, $rt, Label;
  - slt $at, $rt, $rs # Set $at to 1 if $rt < $rs, else 0
  - beq $at, $zero, Label # Branch to Label if $at == 0
- bge: $rs, $rt, Label;
  - slt $at, $rs, $rt # Set $at to 1 if $rs < $rt, else 0
  - beq $at, $zero, Label # Branch to Label if $at == 0

(3) Give a single beq assembly instruction that causes infinite loop. (consider that there's no delay slot)

- beq $0, $0, 8

(4) The j instruction can only jump to instructions within the "block" defined by "(PC+4)[31:28]". Design a method to allow j to jump to the next block (block number +1) using another j.

- 原本我們只有寫一次 jump 指令
- 改成寫 jump, nop, jump 指令
- 這樣我們會在 nop 執行 PC+4 (跳到下一個 block)

(5) Why a Single-Cycle Implementation Is Not Used Today?

- inefficient
- the clock cycle must have the same length for every instruction
- the longest possible path in the processor determines the clock cycle

4. Problems Encountered and Solution

- 波形圖遇到 xxxx
- 可以把 scope 的東西丟到波形圖的 name 測試看看
- 檢查錯誤訊息對應到的 operation, 協助 debug

5. Feedback

- No comment