

DCP3362 Computer Organization Lab 2

* Name: 石育璋 ID: A073708 Email: stoneoneto1203@gmail.com

1 Architecture diagrams

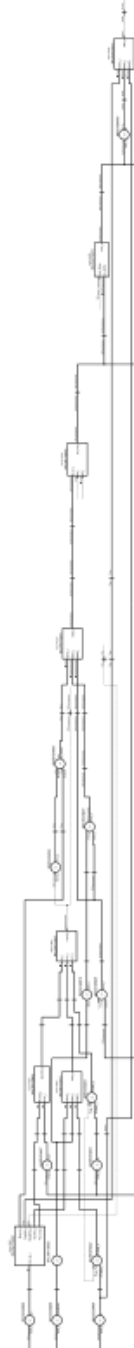


Fig. 1.1: Data flow of CPU

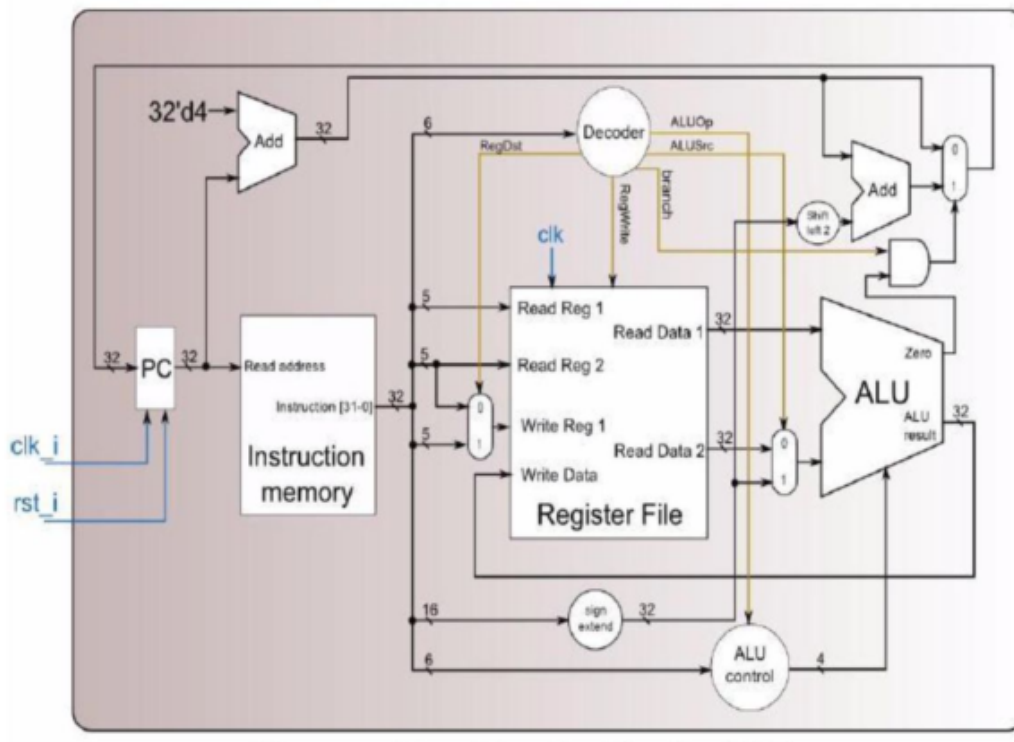


Fig. 1.2: Architecture

2 Hardware module analysis

1. 在 ProgramCounter 的設計中:

2 個 CPU 控制信號:

- (input) clk_i
- (input) rst_i

rst_i 為重置信號; clk_i 為 clock cycle 的控制信號, 當 clk_i 上沿時, (output) pc_out_o 進行更新。

```

1 module ProgramCounter(
2   clk_i ,
3   rst_i ,
4   pc_in_i ,
5   pc_out_o
6 );

```

Fig. 2.1: ProgramCounter

2. 在 Instr_Memory 的設計中: pc_addr_i 為輸入指令地址, 透過 \$readmemb 讀入我們設定的指令檔案, (output) instr_o 輸出指令。

```

1 module Instr_Memory(
2     pc_addr_i,
3     instr_o
4 );

```

Fig. 2.2: Instr_Memory

3. 在 Decoder 的設計中:

- (input) instr_op_i: 為 instruction[31:26]
- (output) RedWrite_o: 控制是否寫入寄存器
- (output) ALU_op_o: 提供 ALU control
- (output) ALUSrc_o: 控制 ALU source 的 2 – 1 MUX
- (output) RegDst_o: 控制選取 RD 的 2 – 1 MUX
- (output) Branch_o: 控制分支信號

```

1 module Decoder(
2     instr_op_i,
3     RegWrite_o,
4     ALU_op_o,
5     ALUSrc_o,
6     RegDst_o,
7     Branch_o
8 );

```

Fig. 2.3: Decoder

4. 在 Sign_Extend 的設計中:

- (input) data_i: 為 16-bit 的 signed number
- (output) data_o: 輸出經 signed extension 之 data_i 的值

```

1 module Sign_Extend(
2     data_i,
3     data_o
4 );

```

Fig. 2.4: Sign_Extend

5. 在 Shift_Left_Two_32 的設計中:

- (input) data_i: 為 32-bit 的 number
- (output) data_o: 輸出左移 2-bit 之 data_i 的值

```

1 module Shift_Left_Two_32(
2     data_i,
3     data_o
4 );

```

Fig. 2.5: Shift_Left_Two_32

6. 在 Reg_File 的設計中:

- (input) RSaddr_i: RS 地址
- (input) RTaddr_i: RT 地址
- (input) RDaddr_i: RD 地址
- (input) RDdata_i: 要輸入 RD 的資料
- (input) RegWrite_i: 是否寫入 RD 控制信號
- (output) RSdata_o: RS 資料
- (output) RTdata_o: RT 資料

```
1 module Reg_File(  
2     clk_i ,  
3     rst_i ,  
4     RSaddr_i ,  
5     RTaddr_i ,  
6     RDaddr_i ,  
7     RDdata_i ,  
8     RegWrite_i ,  
9     RSdata_o ,  
10    RTdata_o  
11 );
```

Fig. 2.6: Reg_File

3 Finished part

除了以上 modules 的設計之外，還包括連接所有 module 之間的連線，以及 3-bit ALUop 自定義。下面是 testbench 測試出來的波形圖。



Fig. 3.1: wave1

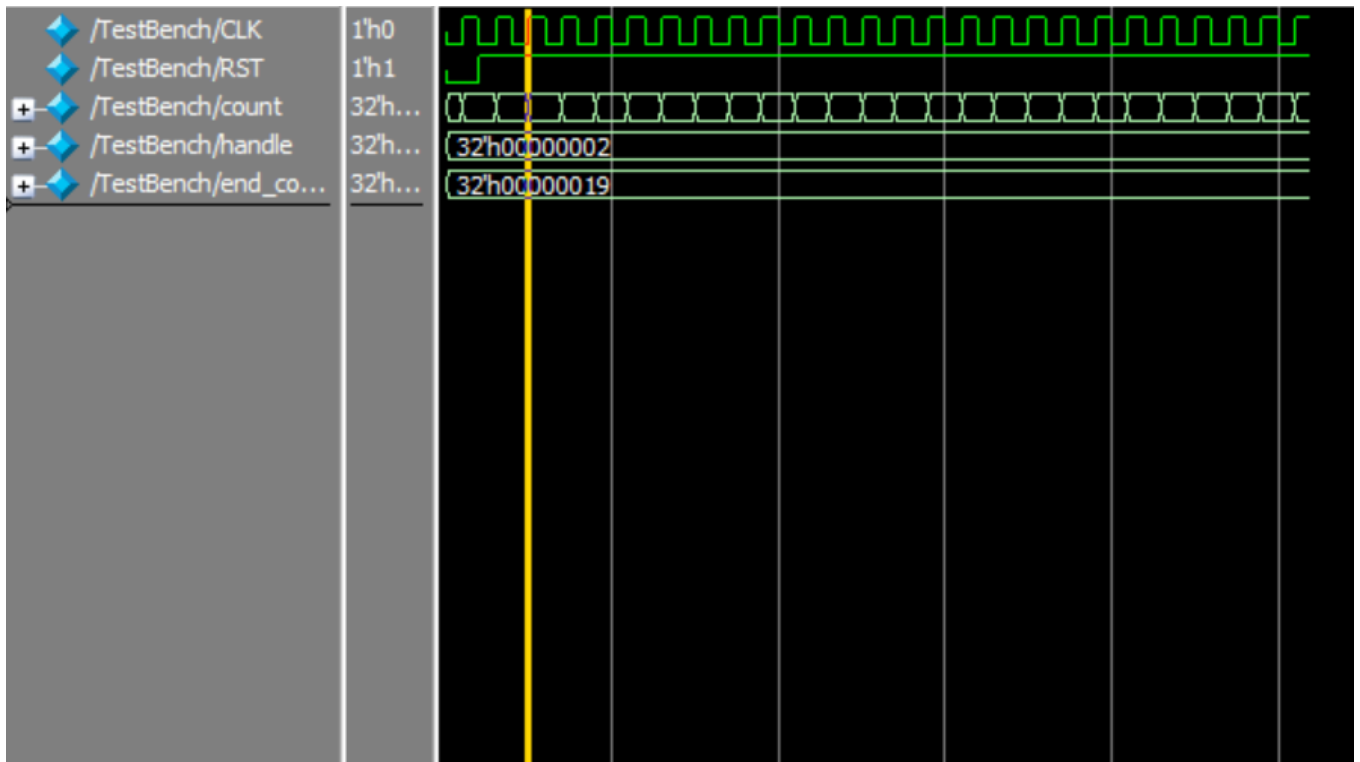


Fig. 3.2: wave2

4 Problems you met and solutions

1. 在組裝所有 modules 時 wire 常常連錯，最後參考老師 ppt 上的架構圖進行連接。
2. 教材裡找不到 3-bit 的 ALUop 定義，後面自定義 addi 等指令。

5 Summary

這個 Lab 很考驗看波形圖 debug 的能力，才能找出哪裡的線接錯，以及 modlue 是否有誤。