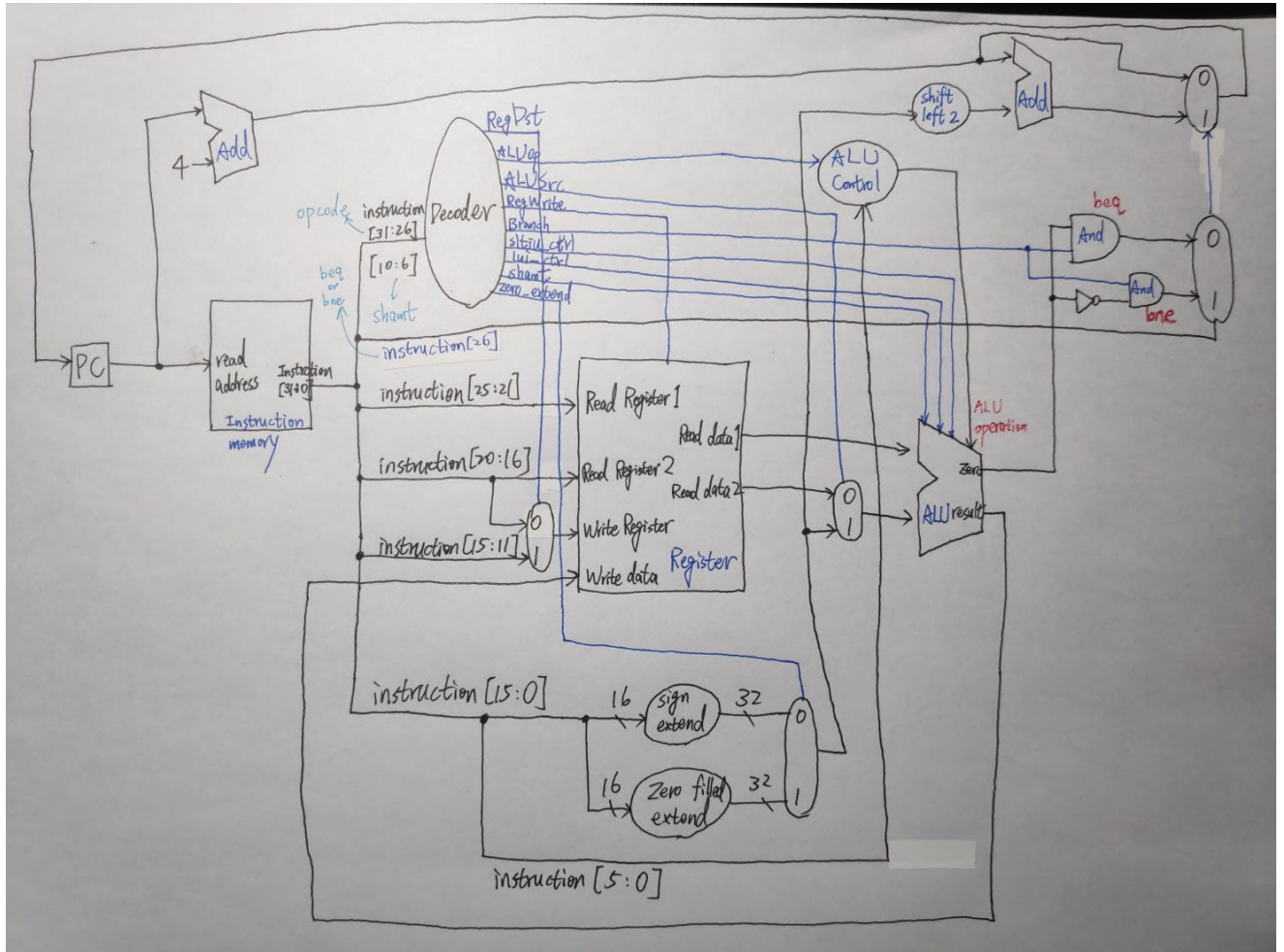


# Term Project - Lab 2(Simplified single cycle CPU) Report

Architecture diagram:



### Module description:

**Adder:** 給兩個 input 值，並相加出一個 output 值。用於兩個地方。1. PC 每次+4。2. 算出 Branch 要跳到的 address。

ALU (include alu\_top, alu8bits, alu32bits, CLA4bits, CLA8bits): 執行 and, or, add, sub, slt, shift 等主要運算。主要有兩個 input 值，Data1 和 Data2(R-type)或 Data1 和 signed-extension 過後的 instruction[15:0](I-type)。利用 Decoder 的各種 output 決定要執行哪種運算(包含 sltiu\_ctrl, lui\_ctrl, shamt)。若是其他 R-type instructions 則用 ALU\_Control 的 output 來決定。

ALU\_Ctrl: 輸入 ALUop 和 instruction[5:0](function field)並輸出 ALU control input 給 ALU，讓 ALU 知道它該做甚麼。

Decoder: 輸入 instruction[31:26](opcode)和 shamt(right shift bits, sra, srav 才會用到)，輸出各個指令需要用到的值。包含 RegDst, ALUop, ALUSrc, RegWrite, Branch, sltiu\_ctrl, lui\_ctrl, shamt, zero\_extend。並接到需要用到的地方。

Instr\_Memory: 讀進 PC address，輸出 32bits instruction。

MUX\_2to1: 兩個 input 值，透過一個 control 值來決定要選哪個 input 值來 output。需要選擇時就需要 MUX。此 lab 有五個地方會用到。

ProgramCounter: 輸出 PC 值。

Reg\_File: 讀取 Rs, Rt 的值，寫入 Rd(有需要的話)。

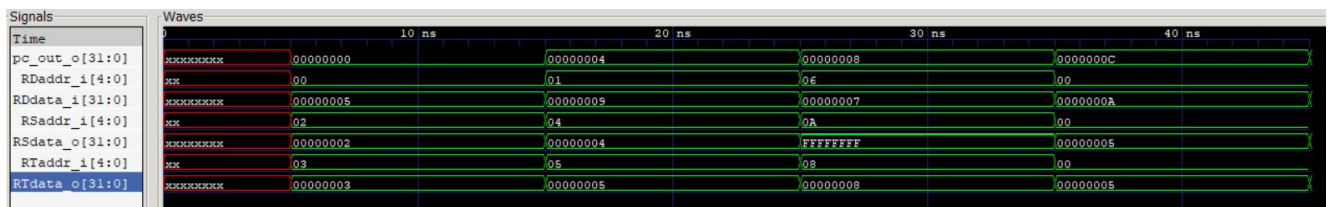
Shift\_Left\_Two\_32: 32bits 的 input 值左移兩位(x4)。

Sign\_Extend: 把 16 bit 的 input 做 sign extension 成 32 bit 。

Zero\_Filled\_Extend: 16 bit 的 input 前面填 16 個 0 擴充成 32 bit 。

Simple\_Single\_CPU: 將上述 module 依照上圖 diagram 接線 。

Waveform:



Questions:

- What is the difference between "input [15:0] input\_0" and "input [0:15] input\_0" inside the module?

Ans: 前者最左邊為 Most significant bit ，後者最右邊為Most significant bit 。例如:

output [3:0] A; // A為4bit , A[3] , A[2] , A[1] , A[0]

output [0:3] B; // B為4 bit , B[0] , B[1] , B[2] , B[3]

- What is the meaning of "always" block in Verilog?

Ans: 以always為主的程式區塊，只有每當觸發條件成立時，執行一次，執行完後需要等待下一次的觸發條件成立才會再次執行。

- What are the advantages and disadvantages of port connection by order and port connection by name in Verilog?

Ans:

### Port connection by order:

**Advantages:** 不需要列出全部 port name，只要確認 port 的順序。

**Disadvantage:** 可讀性較差，port 順序有變化就容易出錯。

### Port connection by name:

**Advantage:** 可讀性較佳，不需要確認順序，只需確認 port name 是否正確。

**Disadvantage:** 需要花費大量時間打出每個 port name。

### Contribution:

討論各種實作方法、各個 module 實作、CPU 接線、debug ... ...。每個 module 都是共同討論並實作，無明顯分工。

### Discussion:

建議老師或助教先教一點基本的 verilog 語法與實作範例。否則有些對 verilog 不熟的同學會做得蠻辛苦的。另外，實作上我們用 decoder 輸出額外的控制元(sltiu\_ctrl, lui\_ctrl, shamt)來執行 sltiu, lui, sra, srav。