Computer Organization
Lab 3 - Single Cycle CPU

教授：蔡文錦

TAs：單宇晟 吳年茵 曾偉杰

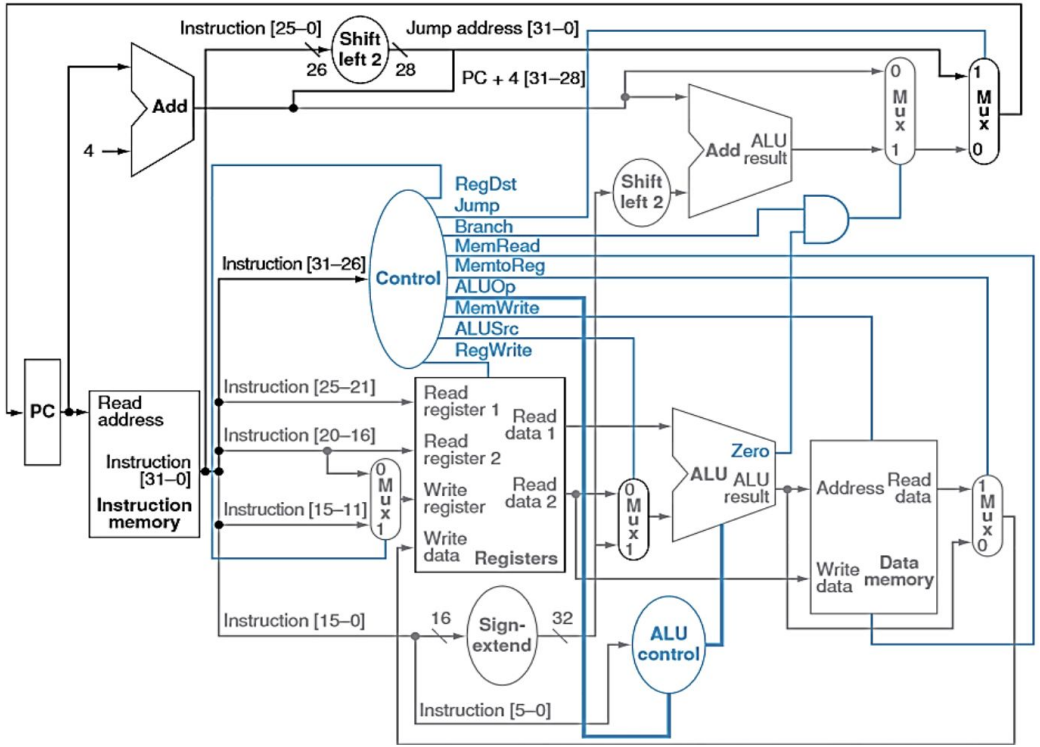
Objectives

In this lab, we are going to implement a **single cycle CPU** with memory unit, which can run **R-type**, **I-type** and **J-type** instructions.

- To realize how to set the control signal in different instruction type.
- To learn how to follow the datapath to form a single cycle CPU

Overview

- Implement the datapath of single cycle CPU as belows



Attached Files

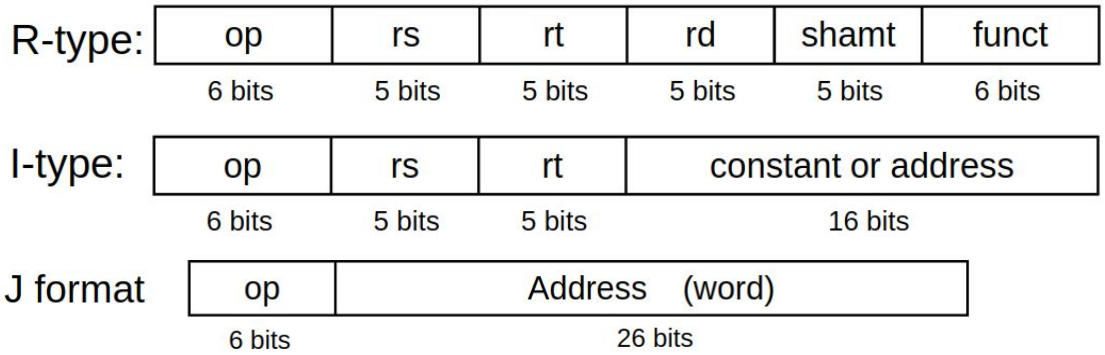
- **TO DO**
  - Adder.v
  - ALU_Ctrl.v
  - ALU.v (You can use the code from Lab 2 or rewrite one)
  - MUX_2to1.v, MUX_3to1.v
  - Decoder.v
  - Sign_Extend.v
  - Simple_Single_CPU.v
  - Shift_Left_Two_32.v
- **DO NOT modify**
  - ProgramCounter.v
  - Reg_File.v,
  - Instr_Memory.v
  - Data_Memory.v
- **For validation - DO NOT modify**
  - testbench.v
- **Testcase - YOU CAN modify the instructions in it.**
  - *.txt

Instruction Set

You are going to implement these instructions :

- **R-type :** add, sub, AND, OR, NOR, slt, sll, srl, sllv, srlv, jr
- **I-type :** lw, sw, beq, bne, addi
- **J-type :** j, jal

Instruction Format:



Instruction Set

R-type		
Function	**Op Field**	**Function Field**
add	6’b000000	6’b100010
sub	6’b000000	6’b100000
AND	6’b000000	6’b100101
OR	6’b000000	6’b100100
NOR	6’b000000	6’b101010
slt	6’b000000	6’b100111
sll	6’b000000	6’b000000
srl	6’b000000	6’b000010
sllv	6’b000000	6’b000100
srlv	6’b000000	6’b000110
jr	6’b000000	6’b001000

I-type	
Function	**Op Field**
addi	6b’001000
lw	6b’101011
sw	6b’100011
beq	6b’000101
bne	6b’000100

J-type	
Function	**Op Field**
j	6b’000011
jal	6b’000010

You can make your own design for sll, srl, sllv, srlv

- Such as
  1. adding new modules and data paths
  2. extending existing ALU operations
  3. or any others …

Compile & Run

- **Compile**
  - $ iverilog -o lab3 testbench.v
- **Run**
  - $ ./lab3

Wrong results:

```
************************************************
* Register Error! [Register 31]                *
* Correct result: 00000050                     *
* Your result:    00000000                     *
************************************************
************************************************
*               You have 19 error !            *
************************************************
```

Correct results:

```
************************************************
*            Congratulation. ALL PASS !        *
************************************************
```

Grading Policy

- There are **5 hidden cases with serial several instructions**, and you will get **20** points for each correct testcase, totally 100 points.
- **Any assignment work by fraud will get a zero point！**
- **No late submission！**

Submission

- **Please attach student IDs as comments at the top of each TO DO file.**
- The files you should hand in include:
  - **all *.v files excluding testbench.v**
- Compress all file ***.v** into one zip file **without any extra folder layer**, and name your zip file as **HW3_{studentID}.zip**
  - **e.g.**
    - **HW3_123456789.zip**
      - **{*.v}**
- **Wrong format will have 20% penalty !**
- **Deadline: 2025/4/27 23:55 (No late submission) !**