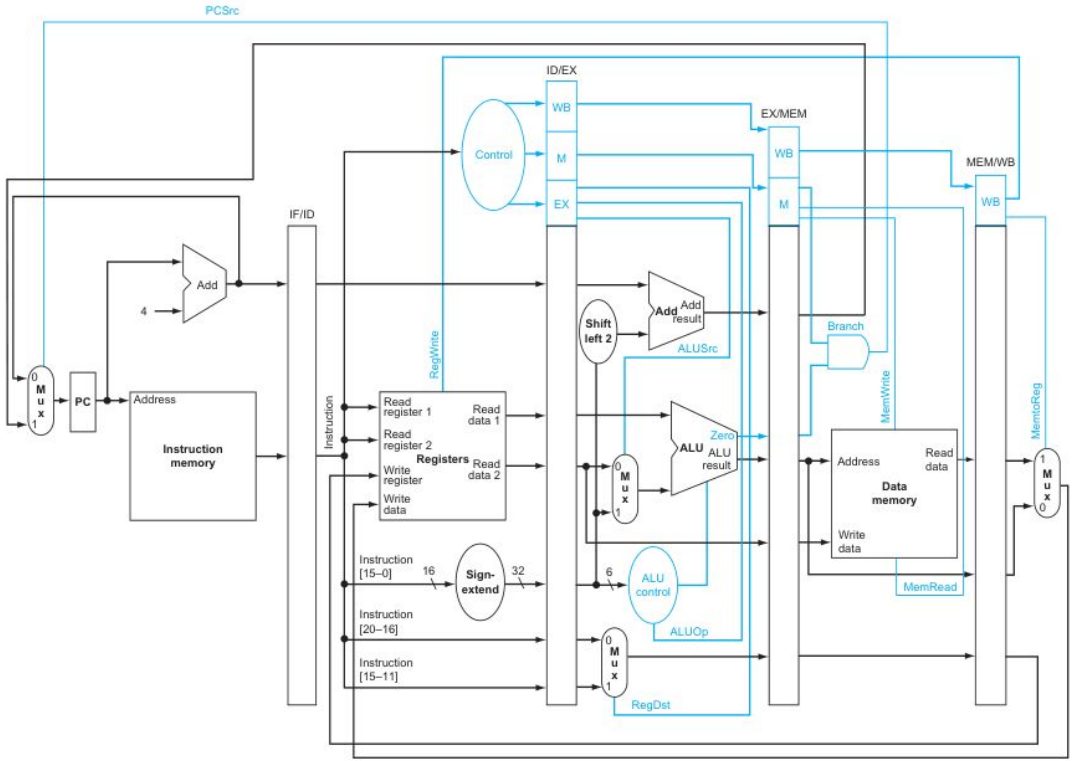
Computer Organization Lab 4

教授:蔡文錦
TAs:林浩君、薛乃仁、吳年茵

Objectives

In this lab, we will transform the single-cycle processor designed in the previous lab into a pipelined processor.

Pipelined CPU Architecture Diagram (1/2)



Pipelined CPU Architecture Diagram (2/2)

- Based on the previous diagram, we will implement a five-stage pipelined processor comprising the **IF, ID, EX, MEM, and WB** stages.
- A pipeline register must be inserted between each pair of consecutive stages.
- Each pipeline register should hold fields for data and control signals, and they should be written on the **positive clock edge**.
- **There should be no operation when a NOP instruction (32'b0) is read.**
- **Rename Simple_Single_CPU.v to Pipeline_CPU.v.**

Pipeline Register Description

- Please design four pipeline registers, ensuring each is **positive edge-triggered** and has a default value of 0.
- Integrate these pipeline registers into the single-cycle CPU you designed in Lab 3 to create the pipelined CPU required for this lab.
- **Do not set any delay time for the sequential circuits of the pipeline registers you design.**

Demands

- Please use the provided modules and the modules from Lab 3 to complete the design of your CPU.
- Each pipeline register should contain fields for both data and control signals.

Task Description

- We will test only the following instructions:
  - add, sub, and, or, nor, slt, sll, srl, addi, lw, sw
- There are 2 test files, test_1.txt and test_2.txt, with each test case worth 50 points.
- The file Pipe_Reg.v and Testbench.v are provided.

```
Pipe_Reg #(
    .size(size_of_reg)
) FOO (
    .clk_i(clk_i),
    .rst_n(rst_n),
    .data_i({data_in_1, data_in_2, data_in_3}),
    .data_o({data_out_1, data_out_2, data_out_3})
);
```

To initialize the Pipe_Reg *FOO* with a 32 bits register, set *size_of_reg* to 32.

Testing Method

Command:

```
> ./a.out
test 1
test 2
Score: 100/100

Testbench.v:298: $finish called at 740000 (1ps)
```

- **Ensure the program compiles successfully by executing the command `iverilog Testbench.v`**

Submission

- Compress all the ***.v files** into one zip file, and name your zip file as **HW4_{studentID}.zip (e.g. HW4_0811510.zip)**
- **No additional directory!**
- **Wrong format will have 10% penalty.**
- **Any assignment work by fraud will get a zero point.**
- **No late submission (deadline: 8/16 23:55)!**

```
> zipinfo -1 HW4_0987654321.zip
Adder.v
ALU_Ctrl.v
ALU.v
Data_Memory.v
Decoder.v
Instr_Memory.v
Mux2to1.v
Mux3to1.v
Pipeline_CPU.v
Pipe_Reg.v
Program_Counter.v
Reg_File.v
Shifter.v
Sign_Extend.v
Testbench.v
Zero_Filled.v
```