

# Computer Organization 2021

## Lab 4: Pipeline CPU I

**Due : 2021/08/22 23:55**

### 1. Goal

In this lab, please modify the single cycle processor designed in lab3 to a pipelined processor. And you **don't** have to consider the hazard issue.

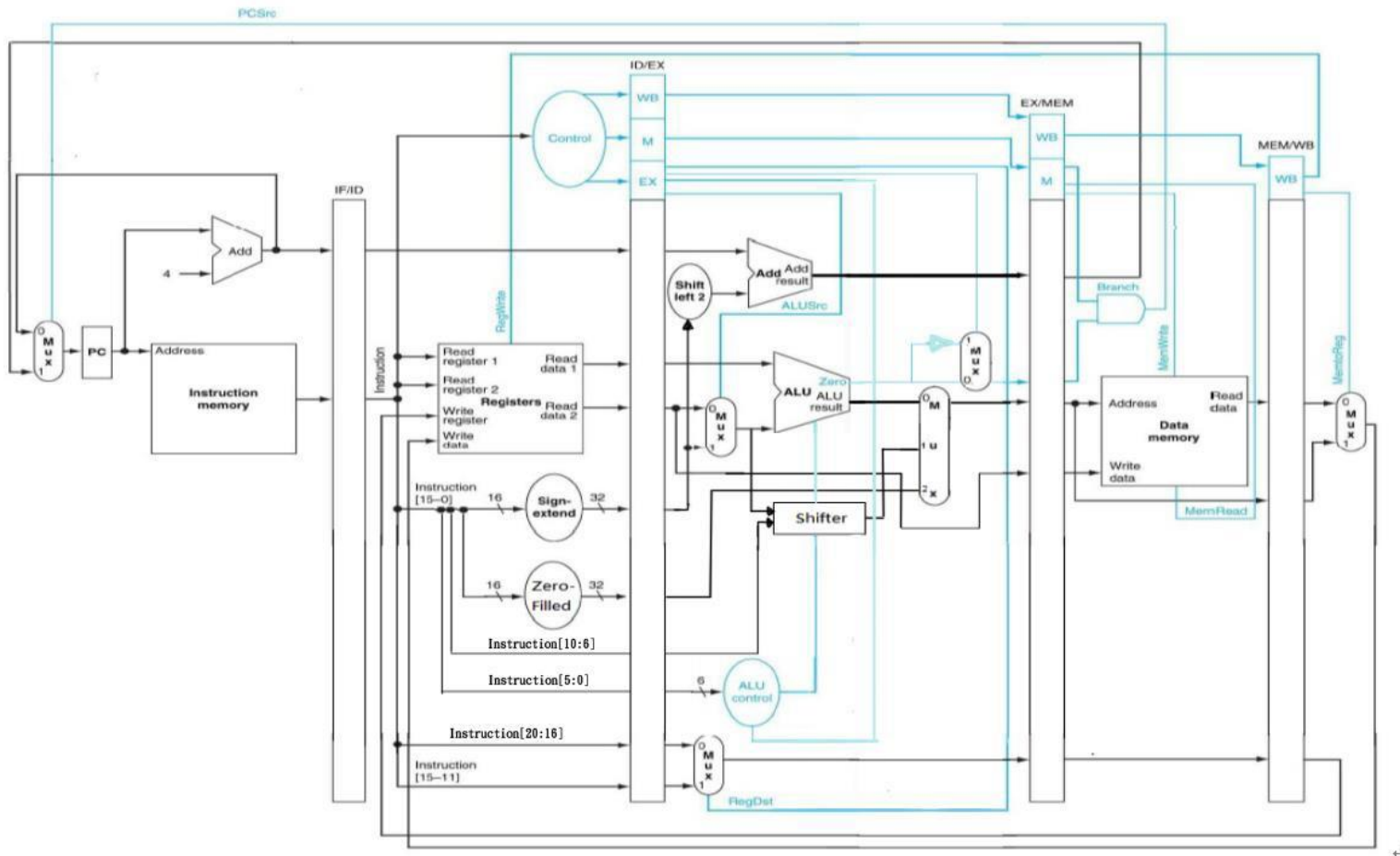
### 2. Demands

- A. Please use **iverilog** as your HDL simulator.
- B. Submit all \*.v source files and report( pdf ) on E3.  
The type of compressed file must be "zip" (Ex. Lab4\_0816000.zip)  
**Other form of file will get -10%.**
- C. Reg\_file(negative-edge triggered), Program\_Counter, and TestBench are supplied.  
Please use these modules and modules in Lab 3 to accomplish the design of your CPU.  
**Remember to change the ``include" region in "TestBench.v"**
- D. For each pipeline register, it should contain the fields for **data** and **control signals**.
- E. Instruction set: we will test part of the instructions which have been implemented in previous lab :  
**add, sub, and, or, nor, slt, sll, srl, addi, lw, sw.**  
We will **NOT** test "beq", "bne", "jump", "jal", and "jr" instruction.  
You may remove the circuits for jump, jr and jal in your design.

### 3. Pipelined CPU

#### A. Architecture diagram

### A. Architecture diagram



According to the above diagram, in this lab you should implement a five stage pipelined processor with IF, ID, EX, MEM, and WB stages.

You should insert a pipeline register between each two stages.

Each pipeline register should contain the fields for **data** and **control signals**.

The pipeline registers are written when the positive clock edge occurs.

## B. The description of pipeline stage

The function of each stage is described as follows:

- **IF stage:**  
In this stage, the processor fetches an instruction from the instruction memory and performs  $PC + 4$ .
- **ID stage:**  
In this stage, the processor decodes the instruction to generate the control signals, reads two source registers, and generates the sign extended immediate value.
- **EX stage:**  
In this stage, ALU\_Ctrl generates control signals for function units according to ALUOp. At the same time, Register Write ID and branch target are also determined in this stage.
- **MEM stage:**  
In this stage, the processor accesses data memory according to the control signals. The modification of PC from branch taken instruction is also performed in this stage.
- **WB stage:**  
In this stage, the processor will write the value into register file according to the control signal when negative clock edge occurs.

## C. Description of pipeline register

Please design four pipeline registers. Each pipeline register must be "positive-edge triggered", has default value 0.

Then, insert these pipeline registers into your single-cycle CPU designed in Lab3 to accomplish the pipelined CPU required in this lab.

**DO NOT** set any delay time for the sequential circuits of the pipelined registers designed by you.

#### 4. Test

Modify [line 43 of TestBench.v](#) to read different test data.

There are 2 test files, [CO\\_P4\\_test\\_data1.txt](#) and [CO\\_P4\\_test\\_data2.txt](#).

Corresponding instructions and output answer are in [data1\\_result.txt](#) and [data2\\_result.txt](#)

#### 5. Grade

- A. Total score: 100pts. **COPY WILL GET A 0 POINT!**
- B. Instruction score: 80 pts.
- C. Report: 20 pts - format is in [StudentID\\_report.docx](#).

#### 6. Hand in your assignment

Please upload the assignment to the E3.

Put all **\*.v source files** and **report( pdf )** into same compressed file.

(Use **Lab4\_student ID.zip** to be the name of your compressed file, **please make sure your files are correct**)

#### 7. Q&A

If you have any question, just send email to all TAs via E3 platform.