Computer Architecture 2023 Fall Lab 3

Report

Modules Explanation

Most of the modules are same as Lab 2, so I will only mention the different parts.

ALU and ALU Control

If the instruction is beq, you the operation ALU should do should be same as **sub**.

Branch Predictor

This module has two tasks, verify the branch prediction in EX stage and predict the branch result of ID stage.

Verify: Input is EX.branch and EX.ALUResult, and the module should determine whether the prediction of this branch is correct. If not, than we should flush IF_ID and ID_EX since both of the instructions in the stage register are wrong, we will also set predictWrong to true.

Predict: Input is ID.branch, we should determine whether to branch base on the state of branch predictor. If strongly taken or weakly taken, then we should branch, and flush IF_ID since the instruction is not correct. Otherwise, we shouldn't branch, so do nothing.

Predict_o: if should taken, 1, otherwise 0.

PredictPC: When verifying, if we predict it should taken but it actually shouldn't, set to 1, else if we predict it should non-taken but it actually should, set to 2. Otherwise, set to 0.

Branch Flush

This module is aimed to determine whether to flush IF_ID and what PC should pass to PC_MUX. If (predictWrong || ID.branch && Predict_o), then we should flush IF_ID. We also need to determine the value passed to PC_MUX. If PredictPC == 1, EX.PC+4, else if PredictPC == 2, EX.PC_Branch, else, ID_PC_Branch.

IF_ID

Flush when Branch Flush send flush signal to it.

ID_EX

Need to store PC and PC_Branch to enable branch flush.

Should be flushed when PredictWrong.

Difficulties Encountered and Solutions in This Lab

Forgot to add 4 to EX.PC. = =

Development Environment

MacOS, iverilog.