Habib University

Dhanani School of Science and Engineering

Computer Architecture

(CS-330 / EE-471)

Lab Project Task

Lab 11-12 – Project
A Pipelined RISC Processor

Objectives

In the next two labs, you will be employing all learnt skills to extend the single-cycle RISC-V processor to a subset of pipelined processor.

Project task

We have studied pipeline implementation of a RISC-V processor with data forwarding techniques to overcome data hazards.

Implement the pipeline version of RISC-V processor shown in Figure 1. Initialize all the pipeline registers to an appropriate size. The control values for the forwarding multiplexers are shown in Table 1.

Table 1. The control values for forwarding multiplexers.

Mux control	Source	Explanation
ForwardA = 00	ID/EX	The first ALU operand comes from the register file.
ForwardA = 10	EX/MEM	The first ALU operand is forwarded from the prior ALU result.
ForwardA = 01	MEM/WB	The first ALU operand is forwarded from data memory or an earlier ALU result.
ForwardB = 00	ID/EX	The second ALU operand comes from the register file.
ForwardB = 10	EX/MEM	The second ALU operand is forwarded from the prior ALU result.
ForwardB = 01	MEM/WB	The second ALU operand is forwarded from data memory or an earlier ALU result.

Refer to lecture slides in order to understand the behavior of *forwarding unit*,

Verify the functionality of forwarding by introducing data dependencies in R-format instructions. Do not check the dependency of a load instruction result on the next instruction, as the architecture shown in Figure 1 does not support *stalling* to overcome certain type of data hazard.

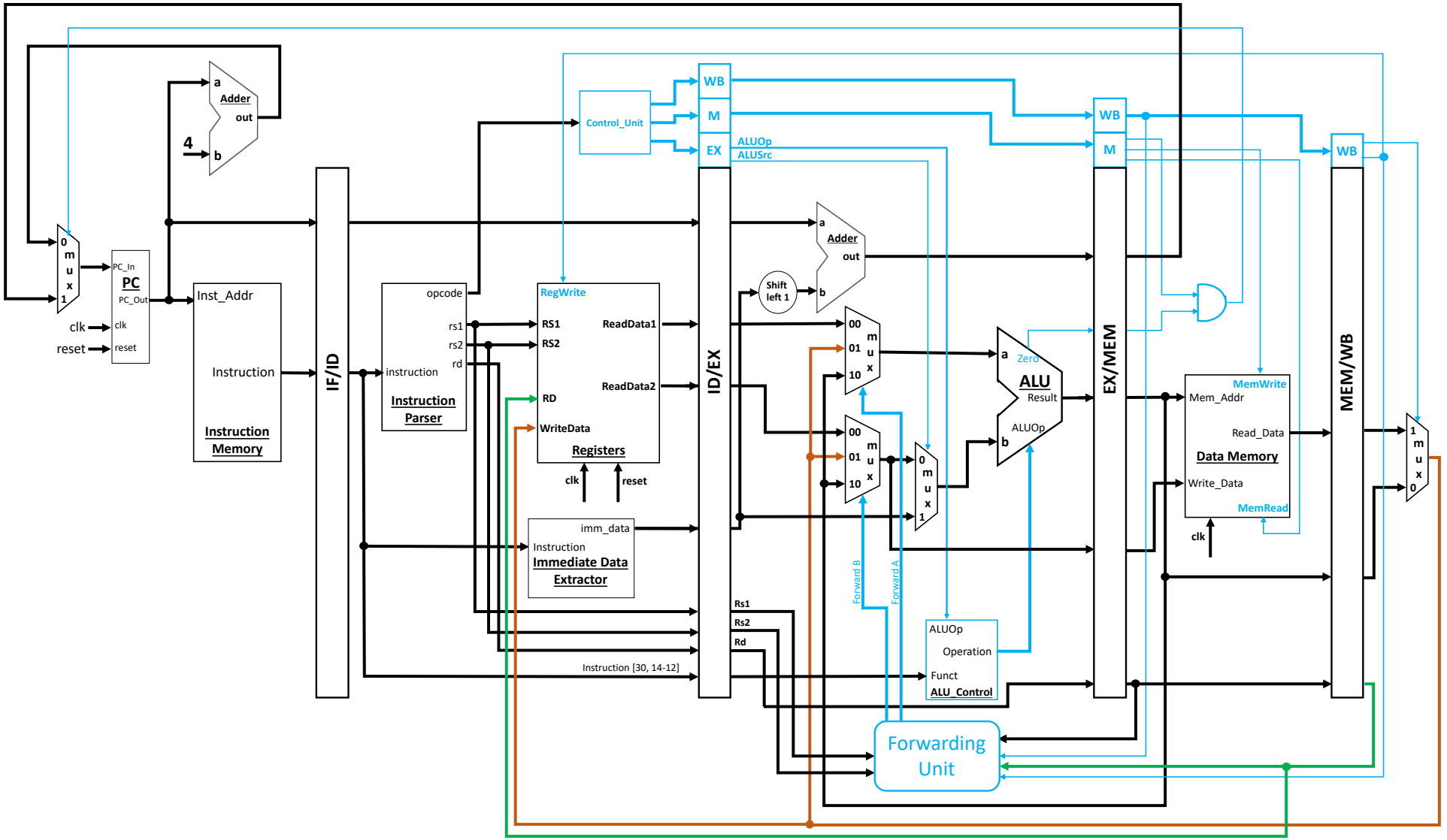

Figure 1. A subset of Pipeline RISC-V processor with forwarding functionality only. Some wires are shown in color just to help a user to distinguish wires easily (specifically between IF/ID and ID/EX stages).