CS 352 – Prof. Mark **Assignment #4** **Due March 10, 2005**

Problems #1-11:

Problem #1: see back side of this sheet
(this is a heavily corrected version of P&H 6.13)

P&H 6.14

P&H 6.17

P&H 6.18

P&H 6.19

P&H 6.21

P&H 6.22

P&H 6.31

> 6.31 [10] <§6.6> One extension of the MIPS instruction set architecture has two new instructions called movn (move if not zero) and movz (move if zero). For example, the instruction
>
> ```
> movn $8, $11, $4
> ```
>
> copies the contents of register 11 into register 8, provided that the value in register 4 is nonzero (otherwise it does nothing). The movz instruction is similar but copying takes place only if the register's value is zero. Show how to use the new instructions to put whichever is larger, register 8's value or register 9's value, into register 10. If the values are equal, copy either into register 10. You may use register 1 as an extra register for temporary use. Do not use any conditional branches.

P&H 6.32

> 6.32 [10] <§6.6> {Ex. 6.31} The solution to Exercise 6.31 should involve the execution of fewer instructions than would be required using conditional branches. Sometimes, however, rewriting code to use movn and movz rather than conditional branches doesn't reduce the number of instructions executed. Nonetheless, even if the use of movn and movz doesn't reduce the number of instructions executed, it can still make the program faster if it is being executed on a pipelined datapath. Explain why.

P&H 6.33

P&H 6.35

Problem #1

The following figure shows the values of various signals within the 5-stage pipelined machine during a particular cycle. The goal of this exercise is to use the information provided by the control signals to determine as much as possible about the instruction residing within each pipeline stage. If you cannot fill in a field of an instruction (e.g. source register #1), specify why. For some instructions, the information inside the front cover of the textbook (or Appendix A) will be useful. For other instructions, it will be useful to look at the control-signal information provided in Figures 6.23 through 6.25 on pg. 401.

If you are having trouble with this problem, it may be helpful to look at the example shown in Figures 6.14.5 through 6.14.9 in the Chapter 6 section of the For More Practice portion of the companion CD. (The figures are on pages FMP-6.14-7 through FMP-6.14-11).

Note also that left-to-right values like “1100” sometimes become several distinct top-to-bottom values like “1 / 10 / 0” in the next pipeline stage. This happens in particular as instructions move from the ID stage into the EX stage.

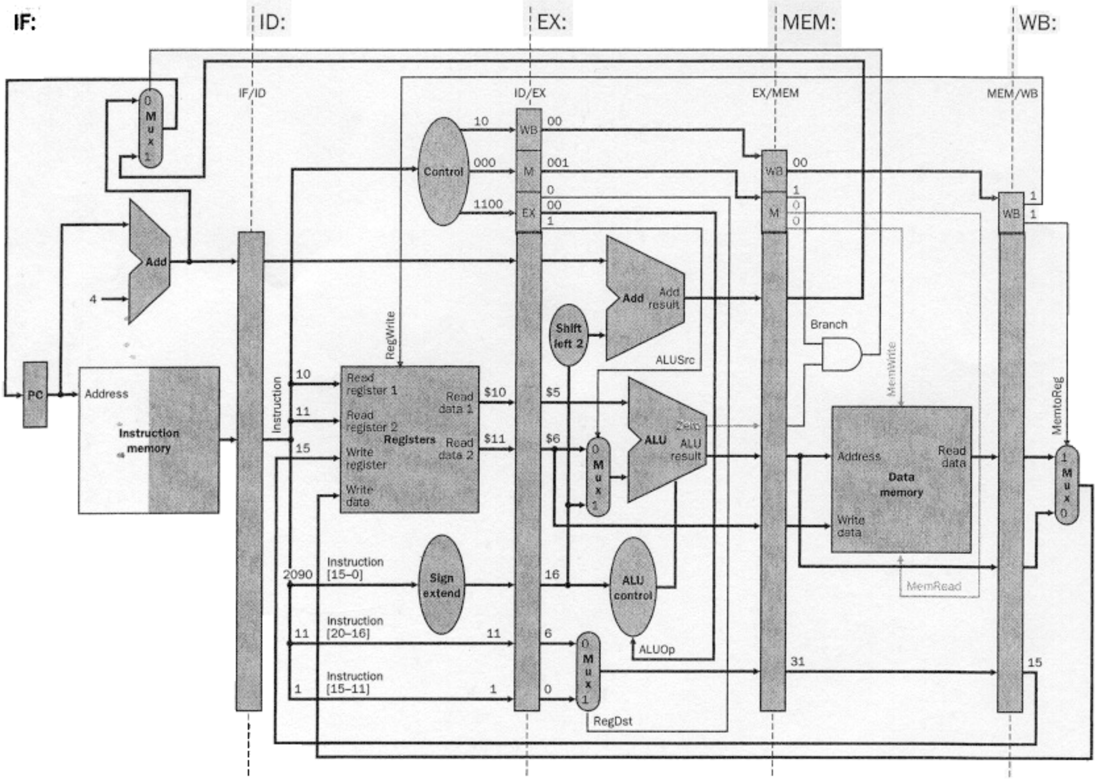

FIGURE 6.71 The pipelined datapath for Exercise 6.9. Use the numeric labels to determine as many fields of each of the five instructions in the pipeline as possible.