

# CSE 141: Introduction to Computer Architecture

## Homework 2, Winter 2014

### Single Cycle Datapath Vanilla ISA Processor

A. Draw the single cycle data path for the processor based on the Vanilla ISA as described in the code given here. (svn co svn://[parallel.ucsd.edu/cse141L/core/tag/lab2part2](http://parallel.ucsd.edu/cse141L/core/tag/lab2part2))

*Note: Use a drawing tool such as PowerPoint, draw the datapath described in the module **core** in the code. You should label all signals and give all widths. You can represent the data memory as an external module. Control signals should be clearly indicated but control logic (e.g. and gates etc) should not be drawn. Clearly show all the data paths and mark the control signals on module interfaces including control between two modules as shown in the reference diagram. Markup a separate copy for each of the following questions. We will provide a sample .ppt file that shows the expected style; you can scavenge registers and other elements from it.*

B. For the following instructions add necessary data paths and control signals to the implementation above. You should indicate the necessary settings of the control signals when the particular instruction is being executed.

(i) **LX - Load Extended**

*Example usage of instruction*

*assembly: LX \$2, \$4*

*rtl:  $RF[(rd \wedge 5'b10000)] \leftarrow MEM[RF[rd]+RF[rs]]$ ;*

In the above instruction the value in  $MEM[2 + 4]$  is stored in a destination register that is a function of \$2. The destination register is determined by  $f(r) = ((r) \text{ xor } 16)$ . Thus, if the instruction was LX \$4, \$6 the destination register defined implicitly is \$20.

(ii) **SX - Store Extended**

*Example usage of instruction*

*assembly: SX \$2, \$4*

*rtl:  $MEM[RF[rd]+RF[rs]] \leftarrow RF[(rd \wedge 5'b10000)]$*

In the above instruction, the value in the register defined as a function of the source register is stored in the location defined by  $MEM[2 + 4]$ . The register from where the value is to be stored is determined by  $f(rs) = ((rs) \text{ xor } 16)$ . Thus, if the instruction was SX \$4, \$6 the register defined would be \$20.

Explain briefly, what are the advantages and disadvantages of defining the destination register as a function of the register as opposed to assigning a fixed destination register?

(iii) **GTM**

*Example usage of instruction*

*assembly: GTM \$2, \$4*

*rtl:  $RF[rd] \leftarrow RF[rs] \wedge PC$*

In the above instruction, the value from \$4 is XORed with the PC and stored in \$2.

(iv) **EID - Execute Instruction from Memory**

*Example usage of instruction*

*EID \$2, \$4*

*instruction =  $MEM[RF[rs] + RF[rd]]$*

*execute(instruction)*

In EID the data from the MEM[\$2+\$4] is loaded as the next instruction to be executed. If this loaded instruction is not a jump instruction, the execution continues at PC+4 after the loaded instruction is executed.