Milestone Overview

You will augment your pipelined MIPSlite microprocessor so that it detects and correctly handles data hazards.

This project milestone is worth 40 points. For full credit, your solution must be completed no later than Thursday, April 13. A 10% penalty will be applied for each day that your solution is late (based on the date when your files are submitted via the "handin" system).

Milestone Deliverables

The deliverables for this milestone are:

- DataHazardUnit.c
- MemoryUnit.c
- BranchUnit.c
- machine.c

Be sure to use the specified file names, and to submit your files for grading via the "handin" system.

Milestone Specifications

Due to the interaction between consecutive instructions, the MIPSlite microprocessor must detect and handle data hazards. When a data hazard is detected, correct execution results can still be obtained by forwarding the appropriate value from the third pipeline stage.

For this milestone, you will add the Data Hazard Unit to your pipelined machine from Milestone #8 so that it detects and correctly processes data hazards.

Create a new project subdirectory under your account and copy the appropriate files, including your "MemoryUnit.c", "BranchUnit.c" and "machine.c", as well as the following instructor-supplied files:

/user/cse420/Project/Milestone09/machine*

The following instructor-supplied files are available but must not be copied into your account:

/user/cse420/Project/Milestone09/ClockUnit*
/user/cse420/Project/Milestone09/FetchUnit*
/user/cse420/Project/Milestone09/ControlUnit*
/user/cse420/Project/Milestone09/RegisterUnit*
/user/cse420/Project/Milestone09/DisplayUnit*
/user/cse420/Project/Milestone09/ShiftUnit*
/user/cse420/Project/Milestone09/MathUnit*
/user/cse420/Project/Milestone09/BranchUnit.h
/user/cse420/Project/Milestone09/MemoryUnit.h
/user/cse420/Project/Milestone09/DataHazardUnit.h

The first nine components behave as specified in the previous milestones.
The following diagram gives the structure of a pipelined implementation of the MIPSLite microprocessor.

A data hazard occurs when the instruction in the third stage of the pipeline updates a register (other than \$0\), and that register number matches the register number in the "rs" field and/or the "rt" field of the instruction in the second stage of the pipeline (when those fields represent source operands).

When a data hazard is detected, the appropriate value from the third stage of the pipeline should be selected instead of a value from the Register File.

The inputs to the Data Hazard Unit are:

a) asserted when instruction in 3rd stage updates a register (1 bit)
b) register to be updated (5 bits)
c) asserted when rs field in 2nd stage is a source operand (1 bit)
d) register number from rs field (5 bits)
e) asserted when rt field in 2nd stage is a source operand (1 bit)
f) register number from rt field (5 bits)

The outputs from the Data Hazard Unit are:

a) asserted when data hazard detected (1 bit)
b) asserted when intermediate result should be forwarded for rs (1 bit)
c) asserted when intermediate result should be forwarded for rt (1 bit)

Use "HexOut" components and user run-time flag 3 within your "machine.c" to display the following values from the second pipeline stage: PC (32 bits), instruction (32 bits), DataHazard signal (1 bit), ForwardRs signal (1 bit), and ForwardRt signal (1 bit). The five signal values must be displayed on one line, with one blank character between signal values.