Milestone #3

Milestone Overview

You will develop the Fetch Unit for the MIPSlite microprocessor.

This project milestone is worth 40 points. For full credit, your solution must be completed no later than Thursday, February 9. 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:

- FetchUnit.tables
- FetchUnit.c

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

Milestone Specifications

The Fetch Unit uses the value of the Program Counter to address into the Instruction Memory Unit to fetch the next machine language instruction.

For this milestone, you will:

1. Experiment with an instructor-supplied test fixture to solidify your understanding of the Clock Unit.
2. Extend the instructor-supplied Fetch Unit to implement the desired functionality.

Create a new project subdirectory under your account and copy the following files:

- /user/cse420/Project/Milestone03/FetchUnit*
- /user/cse420/Project/Milestone03/ClockUnit*
- /user/cse420/Project/Milestone03/IMemUnit*

Use the test fixture in "ClockUnit.check.c" (and the accompanying makefile) to experiment with the Clock Unit.

Note that one of three alternative circuits will be generated, based on the command-line arguments supplied when "simex" is executed. For example, "simex -u 0 20" will set user flag 0 to the value 20 and the third configuration will be selected (and the flag’s value will be used to determine the "TempAuto" value).

Edit your copy of "FetchUnit.tables" and complete the table entries to specify the requested information.

Edit your copy of "FetchUnit.c" (an incomplete version of the Fetch Unit) to implement the missing functionality.

Use the test fixture in "FetchUnit.check.c" (and the accompanying makefile) to experiment with the Fetch Unit.

To validate your Fetch Unit, remove "FetchUnit.o" and re-execute "FetchUnit.test.make". Examine the resulting output to verify that your Fetch Unit operates correctly.
The Clock Unit will provide the timing signals for the MIPSlite microprocessor, as given in the following diagram.

The "ClockPulse" and "Running" signals are the two outputs of the Clock Unit; the "TempAuto" signal is internal to the Clock Unit. Note that the length of the clock cycle is 10000 simulation time units.

The "ClockPulse" signal will be used to synchronize activities in the MIPSlite microprocessor. The "Running" signal will be used to initialize certain components of the microprocessor and will be asserted before the start of the second clock cycle. The "TempAuto" signal will span a number of clock pulses, and provides a mechanism to run the simulation in automatic clock mode for an interval before changing to manual clock mode (for debugging).

The Fetch Unit will fetch the next instruction and update the Program Counter, as given in the following diagram.

The Instruction Memory Unit (IMem Unit) is a separate module representing the memory unit holding the machine language instructions for the currently executing program (comparable to an instruction cache).

Use a Register for the Program Counter (PC). All address and instruction pathways are 32 bits wide, except for the connection of the PC to the IMem Unit (use bits 11:2 of the PC, which will provide word addressing in an address space of 1024 words).

The "TargetAddress" signal is the destination address for control transfer instructions (the Branch Unit will be implemented later).