CS4110 : COMPUTER SYSTEMS DESIGN LAB

Assignment #3

Deadline : 29th Sept. 11:55PM

This assignment is about designing a simulator for out-of-order execution core to simulate the behaviour of superscalar processor

Specifications of the processor :

- Integer ALU unit - 2
- Load Unit - 1, Store unit - 1
- ALU operations- ADD, SUB, MUL, DIV, AND, OR, XOR

The simulator should consist of **register-renaming**, **centralized reservation station**, **re-order buffer**. It should support out-of-order execution with the help of Tomasulo Algorithm.

As soon as the execution of an instruction is completed, the result can be forwarded to all pending operations (i.e *operand forwarding*). The simulator can issue one ready instruction for each functional unit per cycle. All *ALU* and *load* instructions update their architectural register at the end of **complete** stage while *store* instructions complete their execution only at the end of **retire** stage. For *load instruction*, consider *load bypassing* and *load forwarding* optimizations.

Assumptions:

- L1 data cache is perfect cache, i.e all load/store to the cache will be a hit.
- L1 data cache has a single port through which read/write operation is performed. When both *load* and *store* requests comes to the cache at the same time, *load* request will be given priority.
- architectural register file consist of 8 registers, whose values can be initialized at the run time.

Design your simulator in such a way that user can specify the number of entries in reservation-station, re-order buffer, store buffer, and latency for each ALU operation.

Given a program code, your simulator should compute the total number of cycles required to complete the execution of the code,the final result of the computation, and the content of the architectural register file.