**Computer Architecture  
CSCE 3301-01  
Fall 2019  
Project 2 Report  
December 11th, 2019  
For  
Dr. Cherif Salama**

**Background:**

Project: Tomasulo Simulator

Author(s): Abdelhakim Badawy - [abdelhakimbadawy@aucegypt.edu](mailto:abdelhakimbadawy@aucegypt.edu)

Marwan Eid - [marwanadel99@aucegypt.edu](mailto:marwanadel99@aucegypt.edu)

Mohammed Abuelwafa - [mohammedabuelwafa@aucegypt.edu](mailto:mohammedabuelwafa@aucegypt.edu)

Ahmed Ehab - [botta633@aucegypt.edu](mailto:botta633@aucegypt.edu)

*Description: Tomasulo Processor Simulator with speculation.*

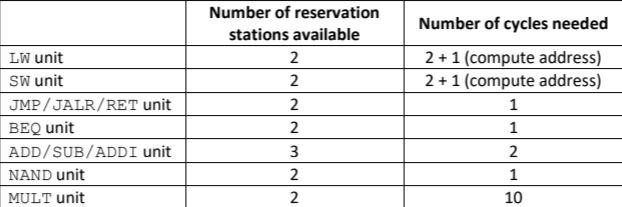
**Sections:**

1. Hardware definition.
2. Algorithm definition and description.
3. Bonus implementation description.
4. Assumptions.

***Section 1: Hardware definition.***

The hardware components of the processor are:

1. Instruction memory
2. Data memory
3. Reorder order buffer.
4. 15 reservation stations:



1. Registers: R0 🡪 R7

***Section 2: Algorithm definition.***

The simulator is a runtime environment for a processor based on the Tomasulo algorithm with speculation. The main modification is expanding the width of the processor to issue and commit two instructions at a time.

The stages of the algorithm are:

1. Issuing:

* Instructions are issued in order.
* The condition for issuing an instruction is:

1. Availability of a suitable free reservation station
2. Availability of a free Reorder Buffer slot.
3. Execution:

* Availability of the operands either written in their corresponding register(s) or as a value in their corresponding reservation(s) station.

1. Writing:

* The availability of the common data bus, *CDB*.

1. Committing:

* Committing is done in order. [condition of the speculation]
* Once an instruction is written it is then processed whether it should be committing or not. It is committed if and only if it is in the order of committing.

For modifying the processor to dual issue and dual commit instructions the following modifications and assumptions were made:

1. The main function calls the issue module twice and increments the clock cycle by 1.
2. In case of having one instruction meeting the requirements of the issue and the second one does not, the second instruction is dual issued with the third instruction, and so on.
3. Committing is also dual-necked. Two instructions are committed per clock cycle.
4. The stages modules: issue, execute, write and commit return a Boolean value reflecting whether the process was completed successfully or not.

***Section 3: Bonus implementation description:***

Test cases:

Provided with the submission are twelve test programs with the description of each and the correct expected outcome of the tests. The design of the test cases was based on the following criteria:

1. Existence of WAR, RAR, use after load and all typed of data dependencies.
2. Existence of loops and conditional statements.
3. Existence of test branches that act opposite to the prediction assumed in the processor’s implementation.

Parser:

Provided with the submission is a parsing program that takes assembly language code and translates it into machine language (binary format instructions). The instructions are then saved in an executable file that is loaded in the instruction memory and they are processed, then, one by one.