Outline for the Simulation of Tomasulo

Central to this will be the **Processor class**



* The processor will indicates the specification of the processor type
  + Some fields
    - Number of arithmetic logical units( ex. 1 add/sub, 1 mul/div)
    - Issue+dipatch same cc (no, Boolean)
    - Reg free + Reg alloc in same cc (no, Boolean)
    - Number of cycle for exe of add, div etc( const int)
    - # of Reservation stations(THINK this should go to **RS class)**
    - **Mult unit has precedence for broadcast.**

**Queue should be generic**

**Queue<Instruction> InstructionQueue(queueSize);**

**Processor** might need to support the following **method**

Load instruction to queue method called readFile(File f)

* For this I’m thinking having an utility method that return an variable of object type Queue
* So once we get the number of instruction from the line, use a for loop ,Get the line, enqueue it;
* After this

Initialize registerfile()

**THE Instruction class**

* Opcode
  + For the getopcode() func, will need to return the type of instr
* Dest operand
* Src1, src2 (source operands)
* Get the instruction method should be able to read a file as outline

**Will need** an Execute function(opcode, dest, src1, src2)

{

Switch(opcode)

Case1: mult

R1 = src1 \* src2, etc

}

**For each clock cycle, might need to check**

If queue not empty

* Which instruction ready to issueich instruction
* If (isOperandReady( instruction ready to broadcast.
  + Qj and Qk == null; then instruction is ready to dispatch;

For broadcast function it shoud return a tag (like an index) for the instruction being broadcast.

Some of the class might have to extend the Processor Class;

Maybe I should have a method that say **getStatus**; that will get the status of the state of processor, which can the is there an instruction that is ready to dipatch, issue, if yes which one.



Get that instruction and do some action. The get status might need to be call each time we get an instance of a new clock cycle.

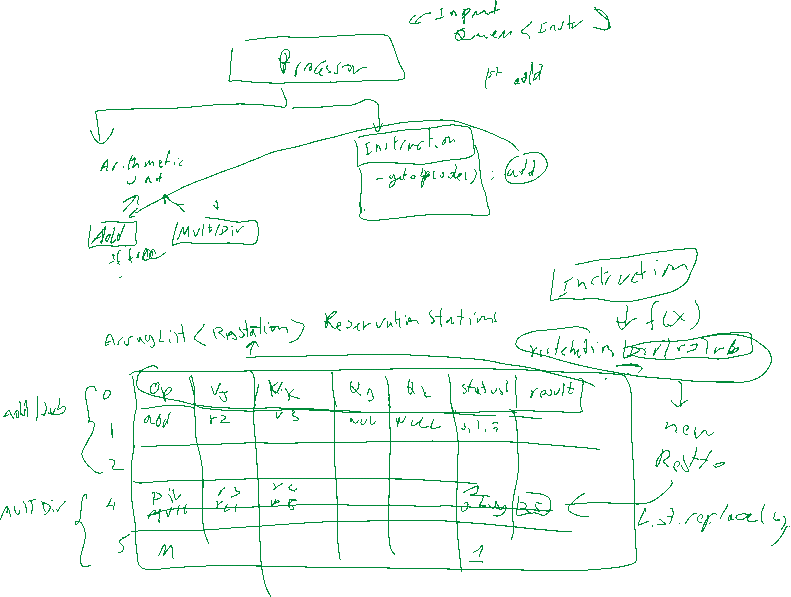
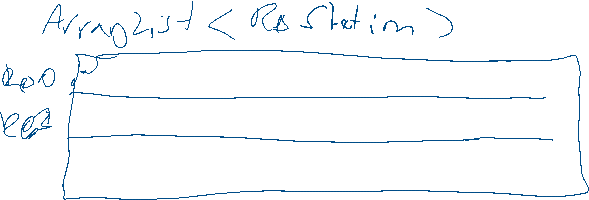
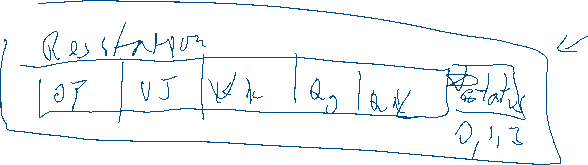
Think we should also have a **ExecutionUnit Class** That will serve as the parent class for other subunit

Like the adderunit and the mult/Div unit.(those separate classes should extend the ExectutionUnit Class)

This



The adderUnit and the MulUnit subclass should have a getStatus method as well to check the if the unit



is busy for a particular clock cycle.