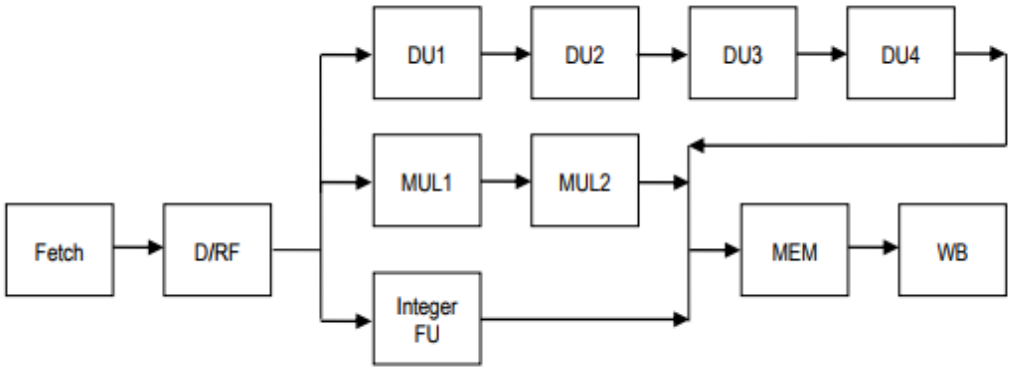
Design Document

Programming Project 2

Pooja Upadhyay
B Number – B00715200



Representation -

Register File -

public static HashMap<String, Integer> registerFile = new HashMap();

Memory -

public static HashMap<Integer, Integer> _mem = new HashMap();

Instruction List-

public static ArrayList<Instruction> _inst = new ArrayList<Instruction>();
This has list of instruction read from input file with the memory address starting from 4000 and incremented by +4 for next instruction.

Instruction Set-

1) Register-to-register instructions: ADD, SUB, MOVC, MUL, AND, OR, EX-OR
   ADD → `ADD <dest> <src1> <src2>`
   SUB → `SUB <dest> <src1> <src2>`
   MUL → `MUL <dest> <src1> <src2>`
   AND → `AND <dest> <src1> <src2>`
   MOVC → `MOVC <dest> <src1> <src2>`
   OR → `OR <dest> <src1> <src2>`
   EX-OR → `EX-OR <dest> <src1> <src2>`

2) BZ, BNZ, JUMP, BAL, HALT
   BZ → `BZ #literal` (checks the z(zero) flag and Branch when zero)
   BNZ → `BNZ #literal`(Branch when non zero)
   JUMP → JUMP <src> @#literal (This will jump to the new instruction)
   JAL →
   HALT → `Stops execution`

3) LOAD, STORE
   LOAD → LOAD <dest> <src1> <literal>
   STORE → STORE <src1> <src2> <literal>

Class Details-

// Checks whether the register is valid or not ,
//if it is true then that instruction will move to next stage.
//if it is false then it will stall.
class Register{
//refer code file “Simulate.java”
}

//Saves all the instruction data
class Instruction {
//refer code file “APEXSimulator.java”
}

//Sets the Flag for Branching
class PSWFlag{
//refer code file “Simulate.java”
}