Review Fall 2004 Final Exam Problem 2, which was discussed in class on Monday, 13 March 2006.

Problem 1: Using the solution to Fall 2004 Final Exam problem 2 parts a, b, and d (but not c) as a starting point, make changes to implement a new two-source register MM instruction `add.mmr` which operates as shown in the example below. *Hint: The solution requires a register file modification.*

```
 add.mmr   (r1), (r2), r3      # Mem[r1] = Mem[r2] + r3
```

Problem 2: Your boss, a stuck-in-the-twentieth-century RISC true believer who only grudgingly agreed to include `add.mm`, `add.mr`, `add.rm`, and `add.mmr` in MMMIPS, flies into an incoherent rage when you suggest also adding `add.mmm` to MMMIPS. What pushed your boss over the edge? (That is, why is `add.mmm` much harder to add to the implementation in the Fall 2004 exam than `add.mmr`.) Instruction `add.mmm` operates as shown below:

```
 add.mmm   (r1), (r2), (r3)      # Mem[r1] = Mem[r2] + Mem[r3]
```

Problem 3: Write a pair of programs intended to show the benefit of MMMIPS. Both programs should do the same thing, program *A* should use ordinary MIPS instructions and run on the MIPS pipeline shown below. Program *B* should use MMMIPS instructions and run on the implementation shown in the exam solution. Reasonable bypass connections may be added, including those needed for branches.

(*a*) Show the programs.

(*b*) Compute the execution time (in cycles) of each program. The comparison should be fair so each program should be producing the same result.
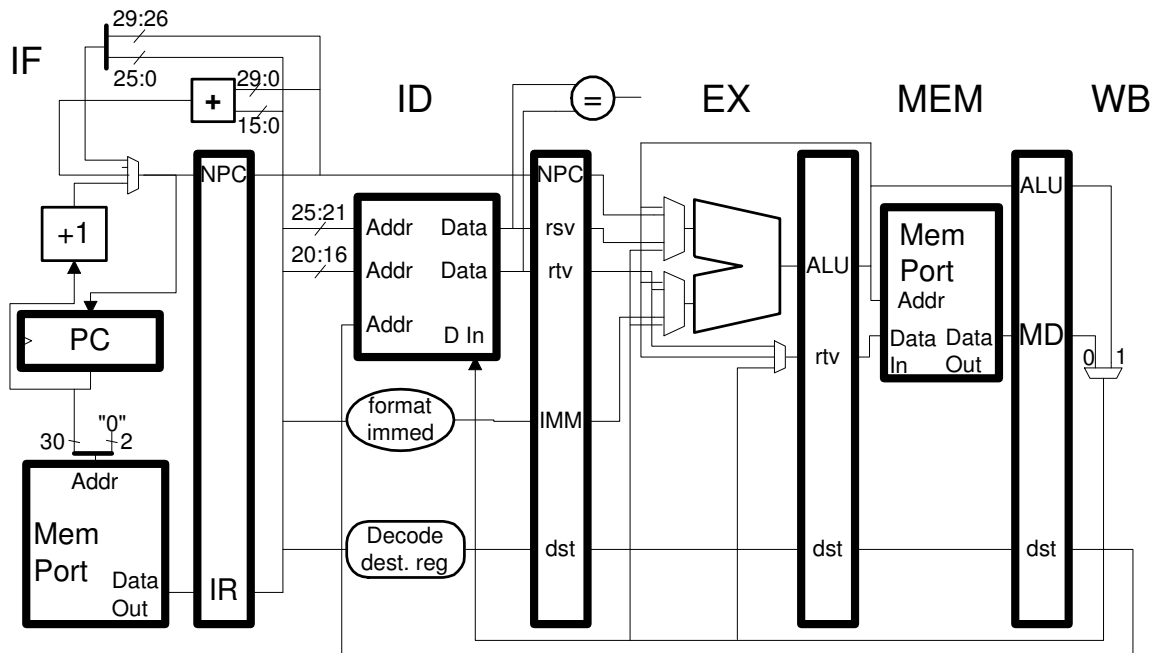
Problem 4: Show a program that will run slower on the MMMIPS implementation that the ordinary MIPS implementation. That program, of course, should not use MMMIPS instructions. Reasonable bypass connections can be added, including those needed for branches. *Hint: Branches are important.*