

CSci 4203 Spring 2017 Lab Assignment 2 (Issued 03/07/2017, due 04/06/2017) :

The purpose of this lab assignment is to learn how the MIPS pipeline works by modifying a given incomplete behavioral implementation of the MIPS pipeline from section 4.13 of the textbook

(http://booksite.elsevier.com/9780124077263/downloads/advance_contents_and_appendices/section_4.13.pdf) and running various assembly codes on the pipeline.

Problem 0 (20 points):

Augment the given MIPS model so that it can run OR, AND, SLT instructions.

Problem 1 (20 points):

Augment the given MIPS model so that it can run the “conditional move” instruction(opcode 0, funct 29), which uses the ‘R’ instruction format and is defined as follows :

```
if (R[rs]<R[rt]) R[rd]=R[shamt]
else do nothing
```

Problem 2 (20 points):

Augment the given MIPS model so that it can run the “jrt” instruction(opcode 30) which uses the ‘I’ instruction format and is defined as follows :

```
if (R[rs]==0) R[rs]=R[rs]+1 and PC=PC+4+loop
else do nothing
```

Problem 3 (20 points):

Augment the given MIPS model so that it can run the “lwcab” (which is short for load word check array bounds (opcode 31)) instruction, which uses the ‘R’ instruction format and is defined as follows :

```
if (R[rt]<R[rd]) R[rs]=M[rt]
else R[rs]=0
```

Problem 4 (20 points):

Implement a ‘taken’ branch predictor for the MIPS pipeline. It will be tested on the ‘BEQ’ instruction.

Handout :

You are provided with an incomplete MIPS behavioral model, an example benchmark and this pdf. The benchmark folder has a “README_benchmarks.md” text file which briefly describes the benchmark.

Handin :

Please turn in your MIPS module, other modules that you may have written, in a single tar.gz file, which should have the name StudentName_Lab2.tar.gz. Your code should be well commented and any documentation you wish to include should appear at the top of each module file.

Grading Criteria :

I will test your code using the provided benchmark as well as others not available to you. Your final grade will depend on the fraction of testcases which are successfully passed by the code as well as the quality of your implementation. There will be partial credit if your code does not work, if it is readable and well documented.