Assignment - 3

The last assignment is to implement one of the last few topics discussed in this course. This can be done in groups of 2 or 3.

You can choose from one of the following options for this assignment.

a. If you had a working (non-plagiarised) Tomasulo code, you could extend this to build a branch predictor

b. If you had a working (non-plagiarised) Tomasulo code, you could extend this to build a 2 level cache with prefetcher

c. If you had a working (non-plagiarised) Tomasulo code, you can extend it to a multi-core Tomasulo (You need to have L1 cache in each core. Then just replicate the single-core module, and integrate a common cache protocol like MESI)

d. If you did not have a working Tomasulo or if you would like to build one of the above independently, you can build it separately without integrating it with Tomasulo. If you choose this option, it is going to be easier than the above. So, you should demonstrate more work. For eg., you could build 2 or 3 types of branch predictors, 2 types of prefetchers etc

e. Build a vector processor with few supported RISC-V vector instructions (for eg., Vector load, Vector store, Vadd, Vmul, Vdotproduct.

f. If you would like to build an entirely independent project, by implementing a part of some published paper, pls mail me

The configuration and specifications of the above can be chosen by the individual groups.

Groups should clearly demonstrate the contribution of individual members for evaluation.

What to submit: Code

Deadline: April 25 , 11:59pm

Marks: 25

Demo sessions will be announced (mostly April 26, 27)

Discussion between groups is allowed, but you cannot submit the same code.

Codes across groups which are similar, will be given 0 marks.