LSU EE 4720 **Homework 7** **Due: 28 April 2025 7:30**

Formatted 9:20, 25 April 2025

Collaboration Rules

Each student is expected to complete his or her own assignment. It is okay to work with other students and to ask questions in order to get ideas on how to solve the problems or how to overcome some obstacle (be it a question of MIPS or assembler syntax, interpreting error messages, how a part of the problem might be solved, etc.) It is also acceptable to seek out assembly language resources for help on MIPS, etc. It is okay to make use of AI LLM tools such as ChatGPT and Copilot to generate sample code. (Do not assume LLM output is correct. Treat LLM output the same way one might treat legal advice given by a lawyer character in a movie: it may sound impressive, but it can range from sage advice to utter nonsense.)

After availing oneself to these resources **each student is expected to be able to complete the assignment alone**. Test questions will be based on homework questions and **the assumed time needed to complete the question will be for a student who had solved the homework assignment on which it was based**.

Student Expectations

Some of the problems require thought, and students are expected to persevere until they find a solution. It is each student's duty to him or herself to resolve frustrations and roadblocks quickly, hopefully helped along by the satisfaction of making progress. There are plenty of old problems and solutions to look at. One way to resolve issues is to ask Dr. Koppelman or others for help.

Resources

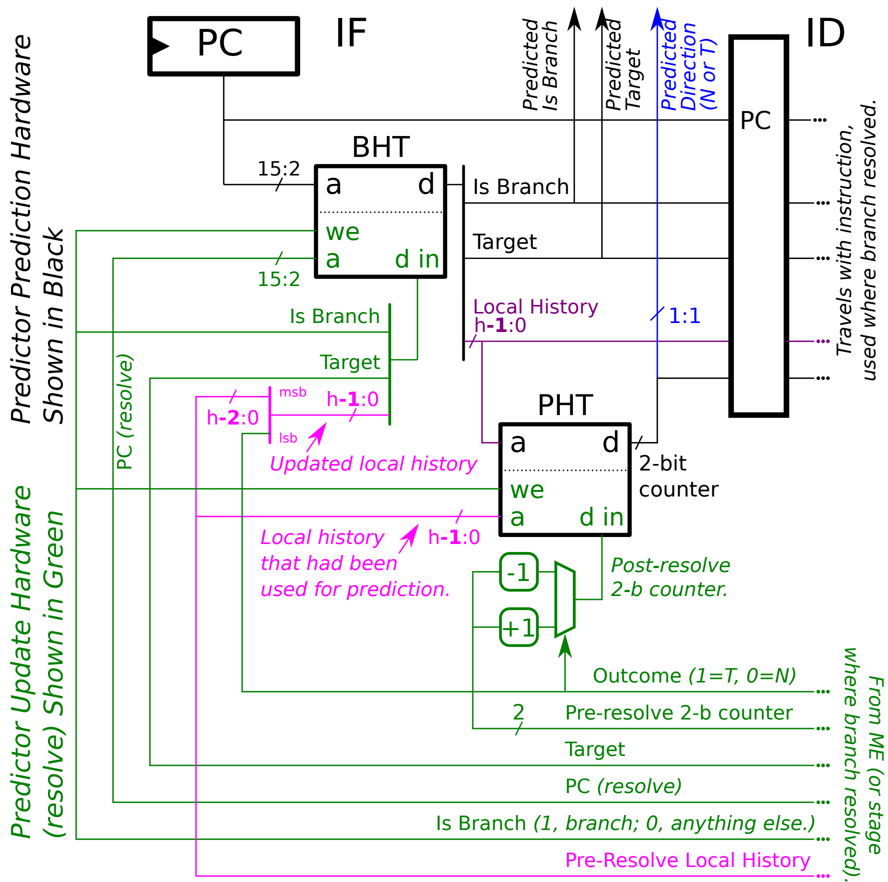
Questions about superscalar MIPS implementations can be found in most final exams.

Problem 1: Locate 2024 Final Exam Problem 4, which asks for analysis of two patterns on bimodal and local predictors.

(*a*) Solve 2024 Final Exam Problem 4(a).

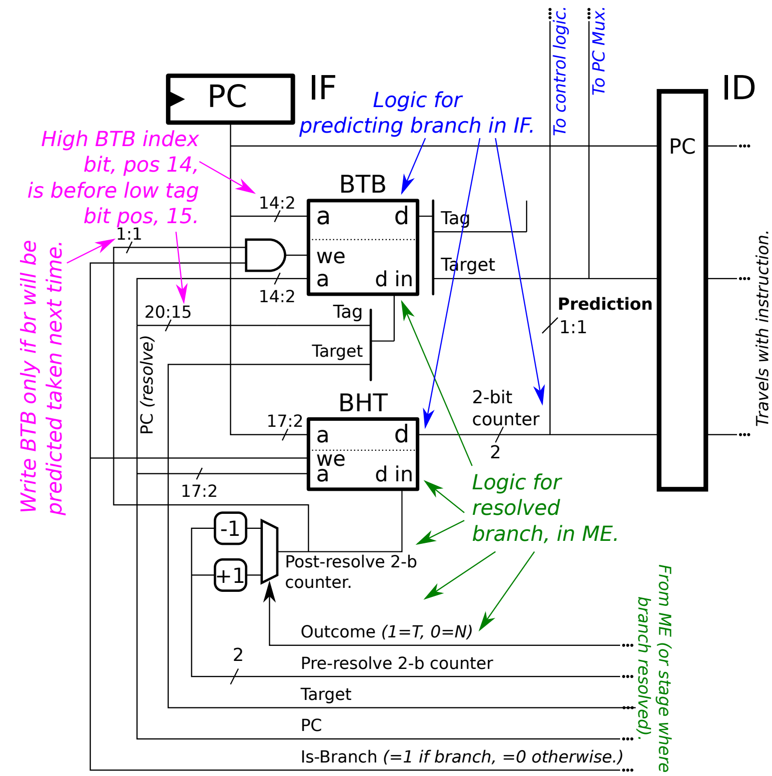
(*b*) What is the smallest local history size for which branches B1 and B2 (from 2024 Final Exam Problem 4) are each predicted at 100% accuracy? *This could have been part b on the final exam question.*

(*c*) Someone foolishly argues that limiting the local history to only three outcomes keeps the cost of the branch prediction hardware low. Explain why that argument is foolish based on the diagram below. Use the sizes, in bits, of the BHT and PHT in your explanation.



There is another problem on the next page.

Problem 2: The diagram below is of a bimodal predictor in which the BHT (branch history table) can keep track of eight times as many branches as the BTB (branch target buffer). Some purple text on the left explains that the BTB should only be updated for a branch that will be predicted taken next time.



Show two sets of branches. In the first set heeding the advice of that text improves performance. In the other set a branch predictor that updates every time would do as well as one that updates only on a misprediction. The solution should look something like the sample below, but with the branch addresses, such as `0x12340`, and outcome patterns changed. Additional branches can be added to each set.

```
# Set 1: (Need to modify these.)

0x12340 B11:  N  T  N  T  N  T  N  T  N  T  N  T  N  T  N  T


0x56784 B12:   T  N  T  N  T  N  T  N  T  N  T  N  T  N  T  N  T


# Set 2:  (Need to modify these.)

0x12340 B21:  N  T  N  T  N  T  N  T  N  T  N  T  N  T  N  T


0x56784 B22:   T  N  T  N  T  N  T  N  T  N  T  N  T  N  T  N  T
```