HW #5: Optimization of the Pipelined CPU

國立陽明交通大學
NATIONAL YANG MING CHIAO TUNG UNIVERSITY

Chun-Jen Tsai
NYCU

05/26/2024

HW #5: Optimization of Pipelined CPU

- ❑ Goal: optimize the pipelined CPU in HW#4
  - ■ You should try to reduce hazard induced pipeline stalls
    - – Data hazard can be resolved by forwarding
      - – Load-use hazard cannot be avoided by in-order processor, so you do not have to handle this
    - – Control hazard can be resolved by a branch predictor
  - ■ Your grade will be based on your CPI running a sample program

- ❑ The deadline of the HW is on 6/13, by 5:00pm.

Guidelines for HW#5

- ❑ You probably should start with adding the forwarding unit to resolve data hazard first (see textbook section 4.8)
- ❑ For the control hazard removal, you can try the bimodal branch predictor unit (BPU) (see textbook section 4.9)

Some Suggestions on BPU (1/2)

- A BPU records the previous branch decision of a `beq` instruction in the branch history table (BHT)
  - The BHT can be implemented as a tag-less memory
    - Think of a cache that discard all the TAG information
    - Collisions in BHT look-up does not cause failure
  - Each BHT entry contains the valid bit, the previous decision bits (taken/not taken), and the target address of the branch
  - The decision bits of the BHT can all be initialized to 1's (taken)

- For simplicity, you do not have to handle jump instructions (e.g. `j`, `jal`, and `jr`), the test program will not contain these instructions

Some Suggestions on BPU (2/2)

- The Fetch stage checks the BHT table to see if the current PC points to a branch instruction
  - The index to the BHT is calculated by PC % n, where n is the total # of entries of the BHT
  - If the valid bit of the indexed entry is 1, update the PC using the previous branch decision:
    - Decision is taken: PC ← PC target
    - Decision is not taken: PC ← PC + 4
  - If the valid bit is 0, PC ← PC + 4
- The Decode stage checks the BHT for each branch instruction, if the valid bit is 0, add the entry to the BHT
  - Set the valid bit to 1
- The Execute stage updates the BHT (the decision bits and the target PC) using the real branch results

HW #5 Grading Guide

- You should extend the size of the instruction and data memory to 64 words each
  - The TAs will use a larger program to test your CPU this time
  - There will be both data and control hazards in the program
  - The fewer cycles your CPU uses to run the program, the better grade you will get

- You should upload your code to E3 by the deadline