Lecture 12 (Replay)

1 Replay

Instead of flushing the entire pipeline, we only replay the instructions which are affected because of misprediction

2 Forward Slice

The tree of preducer-consumer relation

3 Non-Selective Replay

1. Define a window of vulnerability (WV) for n cycles
2. Load should complete within these n cycles (expected)
3. If it doesn't complete, then we replay

3.1 Squashing and Reissuing

1. If there is a misprediction, all instructions in WV of dependent instruction are squashed
2. Their operands' ready bit is set to zero
3. They are reissued in order of forward slice
4. Issue remains with orphan instructions (those not in forward slice but in WV)

3.2 Implementation

1. There exists a kill wire which is set to 1 on a misspeculation
2. If the timer of the operand is non-zero, then we reset its ready bit to zero
3. Otherwise, we know that this operand will not be squashed

3.2.1 Replaying Instructions

3.2.1.1 Approach 1

1. Maintain an issue queue
2. Remove from IW if the instruction has been verified

3.2.1.2 Approach 2

1. Move the instruction to a replay queue
2. Remove it from the queue if it is verified

3.3 Orphan Instructions

1. There might exist operands that are squashed but were not in the forward slice
2. We can keep a track of squashed instructions and rebroadcast the tag of orphan instructions
3. Alternatively, we execute them when they reach the head of ROB

4 Delayed Selective Replay

1. Extend non-selective replay mechanism
2. At time of asserting kill signal, plant poison bit in destination register of load
3. Propagate the bit along bypass paths and register file
4. When instruction finishes execution, check if poison bit is set
    - if yes, squash it
    - else remove it from IW