

Pipelining (III)

Real Computer Pipelining

- In real computers, the general four steps of instruction execution, Fetch, Decode, Execute, and Write-back, are often divided further to better balance the length of each step.
- MIPS pipelining
 - MIPS is a RISC developed by MIPS technologies, a classic RISC pipelining
 - Five steps to execute an instruction
 - Fetch instruction from memory
 - Read registers while decoding the instruction. The regular format of MIPS instructions allows reading and decoding to occur simultaneously.
 - Access an operand in data memory
 - Execute the operation or calculate an address
 - Write the result into a register
- Morden Pipelining
 - E.g. Pentium / PowerPC
 - Seven steps to execute an instruction
 - Instruction fetch unit: get instructions
 - Decode the instructions
 - Identify where the operands are
 - Fetch the operands to the right place
 - Execute the operation
 - Store the operands
 - Operation Commit Unit: update the architectural state of the CPU
- In this course, we use the four steps for simplicity without losing the generality of this topic.

Hazards

- Pipelining can improve the throughput then speed up the instruction execution. However, there are some situations that prevent the next instruction from starting in the next cycle. We call these situations “hazards.”
- Hazards can generally be grouped into three types: structural hazards, data hazards, and control hazards.

Structural Hazards

- Structural hazards occur when a **required resource is busy**. For example, in dirty laundry example, if the load B starts using the washer once the load A is done with washing, the dryer is still drying the load A and is unavailable to the load B when it is done with washing.
- To address the structural hazards, **more stages** are usually helpful. More stages need more control hardware, but it also lowers the bound on cycle time of each stage. Modern CPU divides the four stages we discussed above into even smaller steps, so the time spent on each stage will be more balanced.
- Another way to address the structural hazards is **waiting**. Wait till the required resource becomes available. Actually, all hazards can be resolved by waiting. We call the waiting “stall.”