Computer Architecture

Fall, 2024
Week 11
2024.11.10

組別：_____ 簽名：_____

[group 1]

1. Qustion:

Describe how the stall technique is used to handle load-use data hazards in a pipelined processor and why it is necessary.

Ans:

The stall technique pauses the pipeline when an instruction depends on a previous load instruction's result that isn't available yet. This ensures that the pipeline waits until the data is ready, preventing errors from using incomplete information.

[group 2]

2. Question:

1. Give an example of what may cause the following hazards and whether or not it can happen in MIPS 5-stage pipeline:

a. WAR Data Hazard

b. Structural hazard

c. WAW Data Hazard

d. RAW Data Hazard

Ans:

a. Cannot happen, an instruction J tries to write an operand before the preceding instruction I reads it.

b. Can happen, 2 instructions try to use the same resources, e.g. instruction X tries to access memory for data access (load/store) while instruction Y tries to access memory for Instruction Fetch.

c. Cannot happen, instruction Y tries to write operand before instruction X writes it.

d. Can happen, instruction Y tries to read an operand before instruction X writes it.

[group 4]

3. Question:

Identify all of the data dependencies in the following code and show which dependencies are data hazards.

1. add $2, $5, $4
2. add $4, $2, $5
3. sw $5, 110($2)
4. add $3, $2, $4

Ans:

	Data dependency	Data hazard
$2	(1, 2) (1, 3) (1, 4)	(1, 2) (1, 3)
$4	(2, 4)	(2, 4)

[group 6]

4. Question:

sub $2, $1, $3

and $12, $2, $5

add $14, $2, $2

sw $15, 100($2)

(1) 說明以上指令有甚麼樣的 hazard。

(2) 修改指令以解決這個問題。

Ans:

(1)Data hazard

(2)

sub $2, $1, $3

nop

nop

and $12, $2, $5

add $14, $2, $2

sw $15, 100($2)

[group 7]

5. Question:

Here’ s a sequence of MIPS instruction:

1. sub $2, $1, $3
2. and $12, $2, $5
3. or $13, $6, $2
4. add $14, $2, $2
5. sw $15, 100($2)

How many clock cycle can we save if we use forwarding instead of inserting NOPs to solve data hazards?

Ans:

We can save two clock cycles by forwarding. These instructions need two nops to make sure the data is written before the following instructions read it, which means there will be two wasted clock cycles.

[group 8]

6. Question:

Fill in the blanks! (Ignore the red box)

Hazard Name	Definition	Solution
Structural Hazard		
Data Hazard		
Control Hazard		

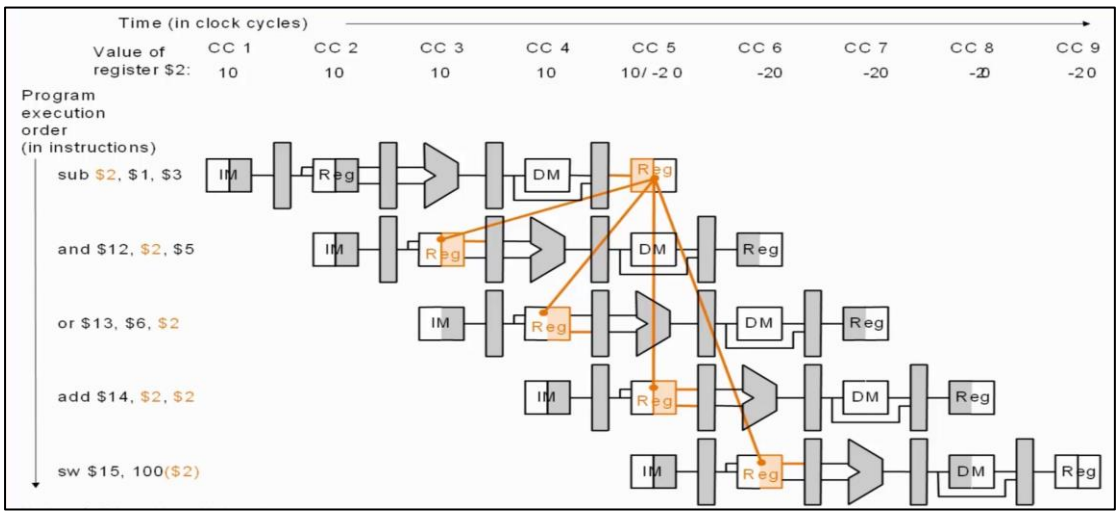
Ans:

Hazard Name	Definition	How to Handle
Structural Hazard	Attempt to use the same resource in two different ways at the same time	- Waiting - Use 2 memory (Data and Instruction Memory)
Data Hazard	Attempt to use item before ready	- Inserting NOP - Forwarding - Stalls
Control Hazard	Attempt to make decision before condition is evaluated	

[group 9]

7. Question:

下圖哪幾個 instructions 會出現 hazard? (Register 有 internal forwarding)



Ans:

and $12, $2, $5

or $13, $6, $2

[group 11]

8. Question:

假設以下兩條指令依序執行：

1. ADD $t0, $t1, $t2
2. SUB $t3, $t0, $t4

這兩個指令在流水線中會出現什麼 hazard？如果不使用 forwarding 的話如何處理？這種處理方法又有甚麼缺點？

Ans:

RAW hazard,可以透過插入一些 NOP 指令，但同時會增加額外的 clock cycles，延長指令的執行時間以及降低流水線的效率。

[group 10]

9. Question:

是非簡答題

a. Forwarding (data forwarding) alone can fully resolve all types of data hazards, including load-use hazards.(F/T)

b. In the MIPS five-stage pipeline, the write-back stage always occurs in the fifth cycle for all instructions, which helps prevent WAW (Write After Write) hazards.(F/T)

c. Data hazards only include RAW (Read After Write) hazards(F/T)

d. What are the three primary types of pipeline hazards?

Ans:

a. False,Forwarding can resolve many data hazards but is insufficient for load-use hazards. When a load instruction is immediately followed by an instruction that needs its result, a stall cycle may still be necessary.

b. True

c. False,Data hazards include not only RAW (Read After Write) hazards but also WAR (Write After Read) and WAW (Write After Write) hazards.

d. Structural hazards: occur when two instructions require the same hardware resource simultaneously.

Data hazards: occur when an instruction depends on the result of a previous instruction that is still in the pipeline.

Control hazards: occur due to the pipeline’ s inability to predict the outcome of branch instructionsestion:

[group 13]

10. Question:

There is a instruction add.

add $1, $2, $3 and which clockcycle time is 100ps, ALU latency is 80ps, Mux Latency is 10ps.

For each stage of the pipeline, what are the values of control signals asserted by add in that pipeline stage?

Ans:

for EX:

ALUSrc=0, ALUop=10, RegDst-1

for MEM:

Branch=0, MemWrite =0, MenRead=0

for WB:

MemtoReg=0, RegWrite=1

[group 14]

11. Question:

Consider the following sequence of instructions in a 5-stage MIPS pipeline (with stages: IF, ID, EX, MEM, WB):

1. lw $t0, 0($s1)
2. add $t2, $t0, $s3
3. sub $t3, $t2, $s4
4. sw $t3, 4($s5)

Answer the following questions:

1. Identify the types of hazards that occur in this instruction sequence.
2. Explain why each hazard occurs and specify between which stages and instructions the hazards exist.

Ans:

Types of Hazards:

- Data Hazard: Exists between lw and add, as add depends on the value loaded into $t0 by lw.
- Data Hazard: Exists between add and sub, as sub depends on the result of add stored in $t2.

Explanation of Hazards:

- The lw instruction loads data into $t0 in the MEM stage, but the add instruction needs $t0 in the EX stage. This creates a Read After Write (RAW) hazard.
- The add instruction writes to $t2 in the WB stage, while sub needs $t2 in the EX stage. This also creates a RAW hazard.