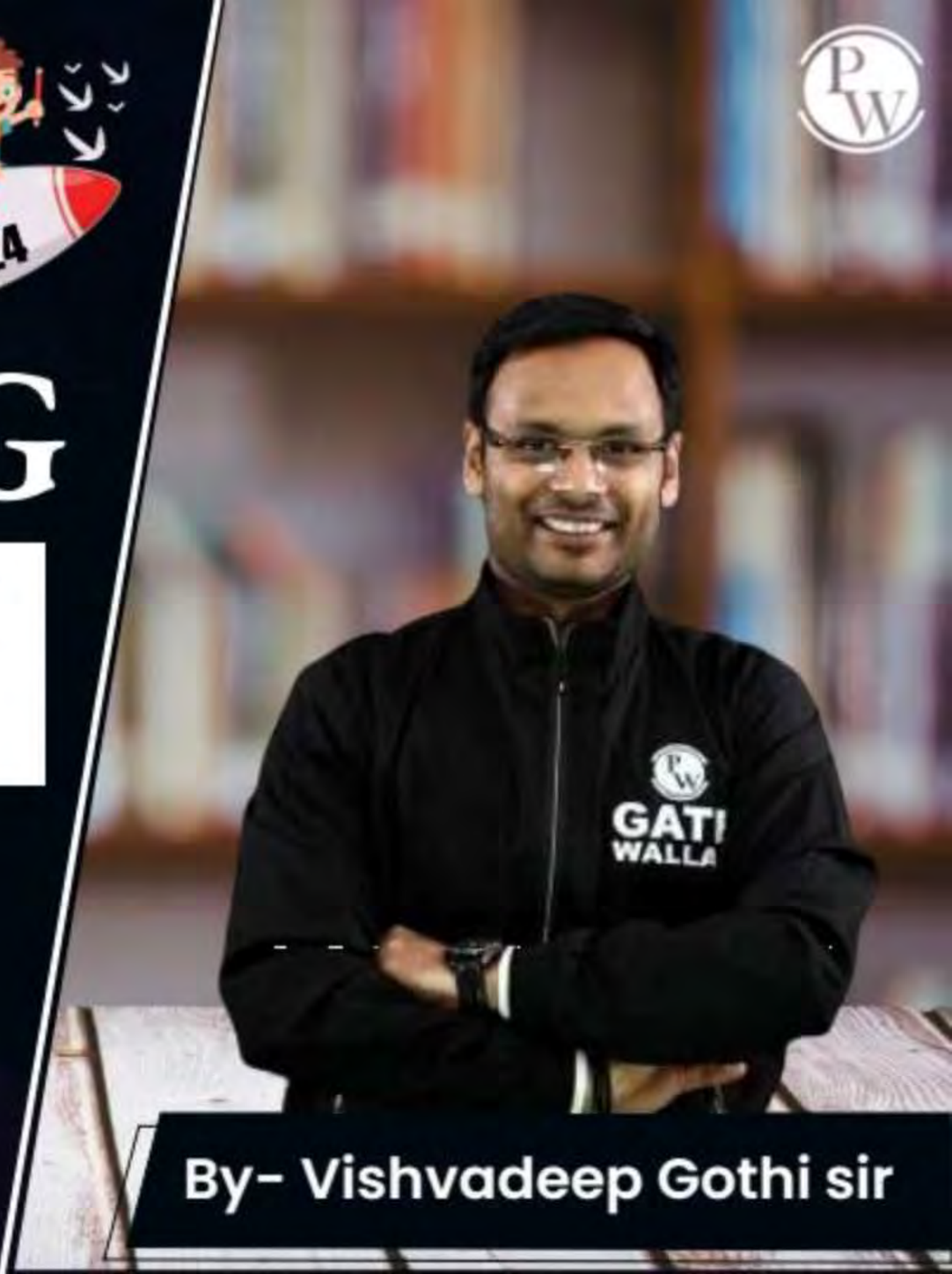
CS & IT ENGINEERING

COMPUTER ORGANIZATION AND ARCHITECTURE

Pipeline Processing

Lecture No.- 05



By- Vishvadeep Gothi sir

Recap of Previous Lecture

- **Topic** Instruction Pipeline
- **Topic** Pipeline Hazard
- **Topic** Operand Forwarding
- **Topic** Branch Prediction

Topics to be Covered

- **Topic** Instruction Pipeline
- **Topic** Data Hazard Classification
- **Topic** RAW, WAW, WAR

Topic : Instruction Pipeline

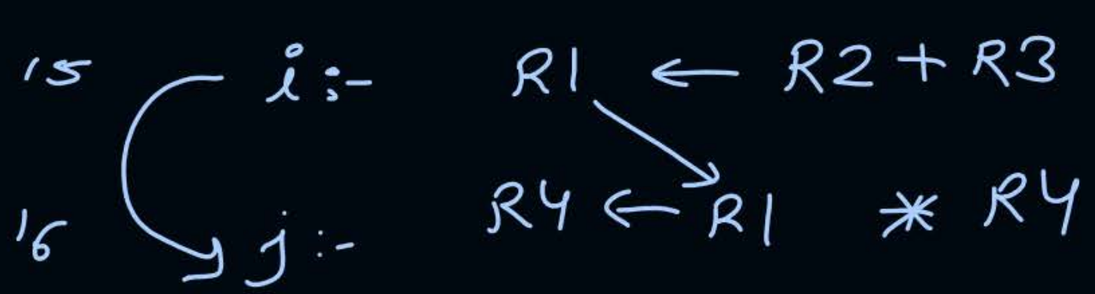
IF: Instruction Fetch
ID: Instruction Decode & Address Calculation
OF: Operand Fetch
EX: Execution
WB: Write Back

Topic : Instruction Pipeline

Clock Cycles

Instructions	1	2	3	4	5	6	7	8	9	10	11	12	13	14
I1														
I2														
I3														
I4														
I5														
I6														
I7														
I8														
I9														

for data dependency :-



after i^{th} instruction's WB, j^{th} instn should go for OF.

no. of stalls due to data dependency = WB phase no − OF phase no.

$= 5 - 3$

$= 2$

[NAT]

#Q. Consider a pipelined processor with the following four stages:

1. IF: Instruction Fetch
2. ID: Instruction Decode and Operand Fetch
3. EX: Execute
4. WB: Write Back

The IF, ID, and WB stages take one clock cycle each to complete the operation. The number of clock cycles for the EX stage depends on the instruction. The ADD and SUB instructions need 1 clock cycle and the MUL instruction needs 3 clock cycle in the EX stage. What is the number of clock cycles taken to complete the following sequence of instructions?

1. ADD R2, R1, R0 — R2←R1+R0
2. MUL R4, R3, R2 — R4←R3*R2
3. SUB R6, R5, R4 — R6←R5-R4

	ADD	SUB	MUL
EX	1	1	3
	Stalls		2

IF — 1
ID — 1
WB — 1

2 Data hazards present
- 1. from 1 to 2 for R2
- 2. from 2 to 3 for R4

no. of stalls per data hazard = 4 − 2 = 2

no. of cycles w/o hazard = 4+3−1 = 6
stalls due to structural hazard = 2
— 11 — data hazard = 2*2 = 4

Total = 12 Ans.

Ex:-

1. R1 ← R2 + R3
2. R4 ← R1 * R5
3. R9 ← R1 — R6

1	2	3	4	5	6	7	8	9
IF	ID	OF	EX	WB				
	IF	ID	—	—	OF	EX	WB	
		IF	ID	—	—	OF	EX	WB

w/o hazard = 5 + 3 − 1 = 7

stalls = 2

9

2 data dependencies
from instⁿ 1 to 2 for R1
& —11— 1 to 3 —11—

but 1 to 3 will not cause stall.

Ex:-

R1 ← R2 + R3

R4 ← R1 * R5

R1 ← R6 + R8

R9 ← R1 + R15

No. of data dependencies ⇒ ?

= 2

stall causing dependencies = 2

Ex:-

R1 ← R2 + R3

R4 ← R12 * R5

R10 ← R6 + R8

R9 ← R1 + R13

no. of data dependencies =

stall causing dependencies = 0

Data

Topic : Hazard Classification

Assume that there are two instructions i and j, and i is executed before j

- R.A.W. ⇒ Read After write
- W.A.W. ⇒ write —11—
- W.A.R. ⇒ write After Read

Topic : Read After Write (RAW)

→ True data dependency

j tries to read a source before *i* writes it. So *j* incorrectly gets the old value.

i : R1 ← R2 + R3

j : R5 ← R1 * R4

IF ID OF EX WB

IF ID OF EX WB

Soln ⇒ operand forwarding

Topic : Write After Write (WAW)

→ write dependency

↳ never causes stalls

j tries to write an operand before it is written by i

i : $R1 \leftarrow R2 + R3$

j : $R1 \leftarrow R4 * R5$

} Soln ⇒ Register Renaming

i : $R1 \leftarrow R2 + R3$

j : $R9 \leftarrow R4 * R5$

Topic : Write After Read (WAR)

→ anti-dependency

↳ no stalls

j tries to write an operand before i reads it, hence i incorrectly gets new value

i: R1 ← R2 + R3

j: R2 ← R4 * R5

Soln :- reg. renaming

i: R1 ← R2 + R3

j: R10 ← R4 * R5

RAW ⇒ True dependency

WAW ⇒ write "]

WAR ⇒ Anti - "] → False dependencies

[NAT]

#Q. Count the number of RAW, WAW and WAR dependencies?

ADD R2, R1, R0	R2←R1+R0
MUL R4, R3, R2	R4←R3*R2
SUB R6, R5, R4	R6←R5-R4
ADD R6, R7, R8	R6←R7+R8
MUL R7, R1, R2	R7←R1*R2
SUB R1, R3, R4	R1←R3-R4

RAW = 4
WAW = 1
WAR = 3

[MSQ] GATE- 24

#Q. An instruction format has the following structure

Instruction number opcode destination reg source reg-1, source reg-2

Consider the following sequence of instruction to be executed in a pipelined processor

I1: DIV R3, R1, R2 $R3 \leftarrow R1 / R2$

I2: SUB R5, R3, R4 $R5 \leftarrow R3 - R4$

I3: ADD R3, R5, R6 $R3 \leftarrow R5 + R6$

I4: MUL R7, R3, R8 $R7 \leftarrow R3 * R8$

Which of the following statement is/are TRUE?

A. There is a WAW dependency on R3 between I3 and I4

B. ✓ There is a WAW dependency on R3 between I1 and I3

C. ✗ There is a RAW dependency on R3 between I1 and I2

D. There is a RAW dependency on R3 between I2 and I3

[MSQ]

#Q. For a pipelined CPU with a single ALU, consider the following situations

I. The J + 1st instruction uses the result of the Jth instruction as an operand — data hazard

II. The execution of a conditional jump instruction → control hazard

III. The Jth and J + 1st instructions require the ALU at the same time — structural hazard

Which of the above can cause a hazard

A. I and II only
B. II and III only
C. III only
D. All the three

[NAT]

#Q. Register renaming is done in pipelined processors:

a) as an alternative to register allocation at compile time
b) for efficient access to function parameters and local variables
c) to handle certain kinds of hazards
d) as part of address translation

[NAT]

#Q. Consider the following sequence of instructions, which is to be executed on a 5 stage pipeline: IF, ID, OF, EX and WB. Each stage takes one cycle for each instruction. Total Cycles required to execute this sequence

1. When pipeline uses operand forwarding ⇒ 10
2. When pipeline does not use operand forwarding ⇒ 14

ADD R2, R1, R0	R2←R1+R0
MUL R4, R3, R2	R4←R3*R2
SUB R6, R5, R4	R6←R5-R4
ADD R6, R7, R8	R6←R7+R8
MUL R7, R1, R2	R7←R1*R2
SUB R1, R3, R4	R1←R3-R4

1. no stalls due to data dependencies.

$n = 6$ inst^{ns}

$k = 5$

no. of cycles $= 5 + 6 - 1 = 10$ cycles

2. stalls due to data dependency $= (5-3) * 2 = 4$

$$\begin{array}{r} \text{Total cycles} = 10 \\ +4 \\ \hline 14 \text{ cycles} \end{array}$$

[NAT]

#Q. Consider the following sequence of instructions, which is to be executed on a 5 stage pipeline: IF, ID, OF, EX and WB. Each stage takes one cycle for each instruction. Total Cycles required to execute this sequence

1. When pipeline uses operand forwarding
2. When pipeline does not use operand forwarding

ADD R2, R1, R0	R2←R1+R0
MUL R4, R3, R2	R4←R3*R2
LOAD R7, (1000)	R7←M[1000]
ADD R6, R7, R8	R6←R7+R8
MUL R7, R1, R2	R7←R1*R2
SUB R1, R3, R4	R1←R3-R4

1. 2 stall cycles due to data dependency on Load instⁿ.

cycles w/o hazard = 5 + 6 − 1 = 10

stalls = 2

Total = 12

---

2. stalls due to 2 dependencies = 2 + 2 = 4

Total = 5 + 6 − 1 + 4 = 14 cycles

Topic : CPI Calculation in Pipeline

w/o hazard no. of cycles for n no. of inst^{ns} = $k+n-1$

with ——||—— ||—— ||— = $k+n-1+$ stall cycles

$$CPI = \frac{\text{total no. of cycles for n inst}^{ns}}{n}$$

$$\text{w/o hazard CPI} = \frac{k+n-1}{n}$$

$$\text{with hazard CPI} = \frac{k+n-1+\text{stalls}}{n}$$

In ideal conditions :-

w/o hazard CPI = 1

with hazard CPI = $\frac{n + \text{stalls}}{n}$ or $1 + \frac{\text{stalls}}{n}$

$$\text{Avg. inst}^{n} \text{ execution time} = CPT_{avg} * t_p$$

[NAT]

#Q. Consider a 5-stage pipeline which is executing a program of 1000 instructions.

Among all instructions 200 instructions cause 2 stall cycles each.

1. Calculate CPI of pipeline?
2. If pipeline cycle time is 3ns then what is average instruction execution time?
3. Calculate CPI of pipeline in ideal conditions?
4. If pipeline cycle time is 3ns then what is average instruction execution time in ideal conditions?

Total stall cycles = 200 * 2 = 400

1. $CPI = \frac{5 + 1000 - 1 + 400}{1000} = 1.404$

2. Avg inst^{n} Execut^{n} time (AIET) = 1.404 * 3 = 4.212 ns

3. $CPI = 1 + \frac{400}{1000} = 1.4 \Rightarrow 1 + \frac{200 * 2}{1000}$

4. AIET = 1.4 * 3 = 4.2 ns

$1 + 0.2 * 2$

In prev. Quest^{n} :-

200 out of 1000 inst^{ns} cause stalls

⇓

$\frac{200}{1000}$ ⇒ 0.2 or 20% of all inst^{ns} cause stalls

$$CPI = 1 + 0.2 * 2$$

#Q.) Consider a 5-stage pipeline in which 30% instns cause 1 stall each.
CPI of pipeline is ?

Ans :-

Soln :- Here value of n is not given, hence use ideal condition

$$CPI = 1 + 0.3 * 1$$

$$= 1.3$$

#q.] 5-stage pipeline

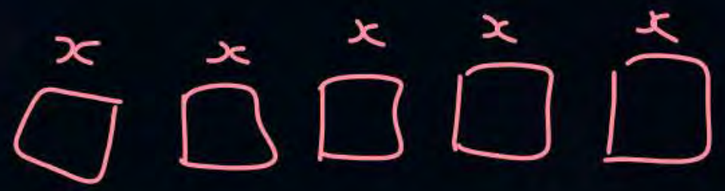
20% instns ⇒ 2 stalls each

15% instns ⇒ 1 stall each

$$CPI = 1 + (0.2 * 2) + (0.15 * 1)$$

$$= 1.55$$

[NAT]



#Q. Consider a 5-stage instruction pipeline, where all stages take equal delay. When an application is executing on this 5-stage pipeline, consider 20% of the instructions incur 3 pipeline stall cycles.

1. Average CPI for pipeline? $1 + 0.2 * 3 = 1.6$
2. Average instruction execution time for pipeline? $1.6 * t_p$
3. The speedup of pipeline achieved with respect to non-pipeline?

with hazard $S_{ideal} = \frac{t_n}{CPI_{avg} * t_p} = \frac{5x}{1.6 * x} = 3.125 = 3.13$

$t_n = 5x$

$t_p = x$

[NAT]

Homework

#Q. Consider a 5-stage instruction pipeline, where stages take delays 5ns, 4ns, 6ns, 4ns and 5ns respectively. The pipeline is used to execute a program in which 25% instructions cause 4 stalls due to hazard. The average instruction execution time in the pipeline is _____ns?

[NAT]

Homework

#Q. A CPU has a five-stage pipeline and runs at 1 GHz frequency. Instruction fetch happens in the first stage of the pipeline. A conditional branch instruction computes the target address and evaluates the condition in the third stage of the pipeline. The processor stops fetching new instructions following a conditional branch until the branch outcome is known. A program executes 10^9 instructions out of which 20% are conditional branches. If each instruction takes one cycle to complete on average, the total execution time of the program is _____seconds?

2 mins Summary

Topic Instruction Pipeline

Topic Data Hazard Classification

Topic RAW, WAW, WAR

Happy Learning

THANK - YOU