# CS & IT ENGINEERING

## Computer Organization & Architecture

1500 Series

Lecture No.- 05

By- Vijay Agarwal Sir

# Recap of Previous Lecture

Topic Expand Opcode Techniue

Topic ALU Data Path

Topic Micro Operation & Micro Program

Topic Control Unit

# Topics to be Covered

**Topic** Pipeline

**Topic** Pipeline Hazards

[NAT]

#Q. An 7-stages perfectly balanced instruction pipeline has [redacted]-cycle-time overhead. If 40% of the instruction in 4 pipeline stall cycles, the speedup achieved with respect to non-piped execution when an application is executing on this 7-stage pipeline is ______

B.I.F = 40%

#stalls = 4

#stalls/Inst$^n$ = $.40 \times 4$

$= 1.6$

$$S = \frac{7}{(1+1.6)}$$

$$= 7/2.6 = \boxed{2.692} \ \underline{\text{Ans}}$$

[NAT]

Consider the pipeline processor have instruction fetch, instruction decode, execute, memory and write back have the latencies 300ρs, 180ρs, 170ρs, 200ρs and 140ρs respectively & Each pipeline have registers between the stages having delay of 20ps. For the enhancement purpose the largest pipeline stages are split into 2 equal delays. The new cycle time value is x (in pico sec.) & the latency of a instruction in a new pipeline is y (in pico sec). The value of x + y is 1540 (in pico sec.).

Ans (1540).

OLD Design

5 Stage      300   180   170   200   140.

New Design

6 Stages      150   150   180   170   200   140.

Buffer Delay = 20

$t_{p\,new} = (200 + 20) = 220\,\text{nsec}$.

$X = 220\,\text{nsec}$

Latency of One Instn = $6 \times 220$

$y = 1320\,\text{ns}$

$X + y = 220 + 1320$

$= 1540$ Ans

[NAT]

Consider a 4 stage pipeline with IF, ID and WB stages taking 1 clock cycle each. The EX stage takes 2 clock cycle for any arithmetic operation and 4 clock cycle for store operation. Operand forwarding from the Ex to ID stage is used for the below set of instruction sequence.

$I_1$ ADD, $R_2 \leftarrow R_0, R_1,$ $R_2 \leftarrow R_0 + R_1,$

$I_2$ MUL, $R_4 \leftarrow R_2, R_3,$ $R_4 \leftarrow R_2 \times R_3$

$I_3$ SUB, $R_5 \leftarrow R_2, R_4,$ $R_5 \leftarrow R_2 - R_4$

$I_4$ STORE $R_5$, x, store the content of M [X] to register $R_5$ 13 Ans

The number of clock cycles required to complete the sequence of instruction is?

Ans(13).

| | | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| WB | | | | | I | | $I_2$ | | $I_3$ | | | | $I_4$ | | | | | |
| EX | | | $I_1$ | $I_1$ | $I_2$ | $I_2$ | $I_3$ | $I_3$ | $I_4$ | $I_4$ | $I_4$ | $I_4$ | | | | | | |
| ID | | $I_1$ | | $I_2$ | | $I_3$ | | $I_4$ | | | | | | | | | | |
| IF | $I_1$ | $I_2$ | $I_3$ | $I_4$ | | | | | | | | | | | | | | |
| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 |

[NAT]

Consider a 5-stage pipeline processor. The number of cycles needed by four instructions $I_1$, $I_2$, $I_3$, $I_4$ in stage $S_1$, $S_2$, $S_3$, $S_4$ and $S_5$ is shown below:

| | $S_1$ | $S_2$ | $S_3$ | $S_4$ | $S_5$ |
|---|---|---|---|---|---|
| $I_1$ | 2 | 2 | 1 | 2 | 3 |
| $I_2$ | 1 | 1 | 2 | 1 | 2 |
| $I_3$ | 1 | 3 | 1 | 1 | 1 |
| $I_4$ | 1 | 1 | 1 | 1 | 1 |

What is the number of cycles needed to execute instruction i=1 completely for first iteration, for the below loop?

```
for(i = 1 to 2)
      {
                  I1;
                  I2;
                  I3;
                  I4;
      };
```

|  | $S_1$ | $S_2$ | $S_3$ | $S_4$ | $S_5$ |
|---|---|---|---|---|---|
| $I_1$ | 2 | 2 | 1 | 2 | 3 |
| $I_2$ | 1 | 1 | 2 | 1 | 2 |
| $I_3$ | 1 | 3 | 1 | 1 | 1 |
| $I_4$ | 1 | 1 | 1 | 1 | 1 |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| $S_5$ |  |  |  |  |  |  |  | $I_1$ | $I_1$ | $I_1$ | $I_2$ | $I_2$ | $I_3$ | $I_4$ |  |  |  |  |  |  |  |  |
| $S_4$ |  |  |  |  |  | $I_1$ | $I_1$ | $I_2$ | ⊖ | ⊖ | $I_3$ | ⊖ | $I_4$ |  |  |  |  |  |  |  |  |  |
| $S_3$ |  |  |  |  | $I_1$ | $I_2$ | $I_2$ |  | $I_3$ | ⊖ | $I_4$ |  |  |  |  |  |  |  |  |  |  |  |
| $S_2$ |  |  | $I_1$ | $I_1$ | $I_2$ | $I_3$ | $I_3$ | $I_3$ | $I_4$ |  |  |  |  |  |  |  |  |  |  |  |  |  |
| $S_1$ | $I_1$ | $I_1$ | $I_2$ | ⊖ | $I_3$ | $I_4$ | ⊖ | ⊖ |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 |

|  | $S_1$ | $S_2$ | $S_3$ | $S_4$ | $S_5$ |
|---|---|---|---|---|---|
| $I_1$ | 2 | 2 | 1 | 2 | 3 |
| $I_2$ | 1 | 1 | 2 | 1 | 2 |
| $I_3$ | 1 | 3 | 1 | 1 | 1 |
| $I_4$ | 1 | 1 | 1 | 1 | 1 |

Assume Data is stored in Buffer.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| $S_5$ |  |  |  |  |  |  |  | $I_1$ | $I_1$ | $I_1$ | $I_2$ | $I_2$ | $I_3$ | $I_4$ |  |  |  |  |  |  |  |  |
| $S_4$ |  |  |  |  |  | $I_1$ | $I_1$ | $I_2$ |  | $I_3$ | $I_4$ |  |  |  |  |  |  |  |  |  |  |  |
| $S_3$ |  |  |  |  | $I_1$ | $I_2$ | $I_2$ |  | $I_3$ | $I_4$ |  |  |  |  |  |  |  |  |  |  |  |  |
| $S_2$ |  |  | $I_1$ | $I_1$ | $I_2$ | $I_3$ | $I_3$ | $I_3$ | $I_4$ |  |  |  |  |  |  |  |  |  |  |  |  |  |
| $S_1$ | $I_1$ | $I_1$ | $I_2$ | $I_3$ | $I_4$ |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 |

## [MCQ]

Consider a hypothetical processor with ~~five~~ Seven pipeline stages (IF, ID, EX, MEM, WB) perfectly balanced & clock cycle time 11 nsec. With the following branch frequencies:

Jump and calls-25% — 4

Conditional branches -30%

Taken conditional branch – 60% = 5

$$\#\text{Stalls}/\text{Instn} = .25 \times 4 + .30 \times .60 \times 5$$

$$= 1 + .9$$

$$= \boxed{1.9}$$

Un-conditional & conditional branches are resolved at the end of fifth and sixth stage respectively processor always executes the branch successor regradless of target and flushes the pipeline if branch is taken. What is the throughput (in million instructions per second) of the system?

A 34.4

B 31.3

C 29.2

D None of these

Unconditional = B.P = 5 − 1 = 4

Conditional = 6 − 1 = 5.

Avg Instⁿ = (1 + # Stalls/Instⁿ) × cycle time

$\Rightarrow (1+1.9) \times 11 \Rightarrow 2.9 \times 11$ nsec

$= 31.9$ nsec

Throughput $= \dfrac{1}{31.9 \times 10^{-9}} \Rightarrow \dfrac{1}{31.9} \times 10^{9} \Rightarrow \dfrac{1000 \times 10^{6}}{31.9}$

$= 31.3 \times 10^{6}$

$= 31.3$ MIPS.

## NAT

H.W

Consider two processors $P_1$ and $P_2$ executing the same instructions set. Assume that under identical conditions, for the same input, a program running on $P_2$ takes 25% less time but incurs 20% more CPI (clock cycles per instruction) as compared to the program running on $P_1$. If the clock frequency of $P_1$ is $1 GH_Z$, then the clock frequency of $P_2$ (in $GH_Z$) is

________.

# NAT

Consider a non-pipelined processor with a clock rate of 4 gigahertz and average cycles per instruction of six. The same processor is upgraded to a pipelined processor with five stages; but due to the internal pipeline delay, the clock speed is reduced to 2.5 gigahertz. Assume that there are no stalls in the pipeline. The speed up achieved in this pipelined processor is 3.75 Ans

## Non Pipeline Processor

Cycle time $= \frac{1}{4GHz} \Rightarrow 0.25$ nsec

1 Instn takes = 6 cycle.

## In Nonpipeline

$ET = 0.25 \times 6$

$= 1.5$ nsec

## Pipeline Processor

$\boxed{CPI = 1}$

Cycle time $= \frac{1}{2.5GHz} = 0.4$ nsec.

$ET_{PIPE} = 0.4$ ns

$S = \frac{ET_{NP}}{ET_P}$

$= \frac{1.5}{0.4} = \boxed{3.75}$ Ans

## MCQ

Delayed branching can help in the handling of control hazards.

The following code is to run on a pipelined processor with one branch delay slot:

I1: ADD R2 ← R7 + R8

I2: SUB R4 ← R5 - R6x

I3: ADD R1 ← R2 + R3

I4: STORE Memory [R4] ← R1

BRANCH to Label if /R1 = = 0

Which of the instruction I1, I2, I3 or I4 can legitimately occupy the delay slot without any other program modification?

A I1

B I2

C I3

[✓ D] I4

$I_1$: ADD $R_2 \leftarrow R_7 + R_8$.

$I_2$: SUB $R_4 \leftarrow R_5 - R_6$.

$I_3$: ADD $R_1 \leftarrow \underline{R_2} + R_3$

$I_4$ STORE $\underline{M[R_4]} \leftarrow R_1$

Branch to Label if $R_1 == 0$.

(a) $\underline{\boxed{I_1}}$ $R_2 \leftarrow R_7 + R_8$.

↳ Used in $I_3$. (Result of $I_1$ as operand Using in $I_3$)

✗

(b) $I_2$ $\boxed{R_4} \leftarrow R_5 - R_6$.

↳ Used in $I_4$ for $M[R_4]$ storing.

(c) $I_3$: $R_1 \leftarrow R_2 + R_3$

↳ used in $I_4$. Not working if we used Den Properly.

(d) $\underline{I_4}$ $\underline{M[R_4] \leftarrow R_1}$

$\underline{M[R_4]} \leftarrow R_1$

[NAT]

Assume a processor that has 5-segment in order-pipelining

| Instruction | Meaning of instruction |
|---|---|
| ✗ $I_0$: X: LOAD $R_2$, 8($R_3$) | //$R_2 \leftarrow$ MEM [8 + $R_3$] |
| ✓ $I_1$: ADD $R_4$, $R_2$, $R_9$ | // $R_4 \leftarrow R_2 + R_9$ |
| ✗ $I_2$: SUB $R_3$, $R_2$, $R_5$ | // $R_3 \leftarrow R_2 - R_5$ |
| ✓ $I_3$: STORE $R_8$, 12($R_3$) | //MEM [12 + $R_3$] = $R_8$ |
| ✗ $I_4$: ADD $R_7$, $R_3$, $R_6$ | // $R_7 \leftarrow R_3 + R_6$ |
| $I_5$: BEQ $R_7$, $R_8$, X | //If $R_7$ == $R_8$ then jump to X |

Which instruction(s) in the above assembly sequence would you place in the delay slot? Assume that the number of available delay slots is 2?

$I_1$, $I_3$.

[NAT]

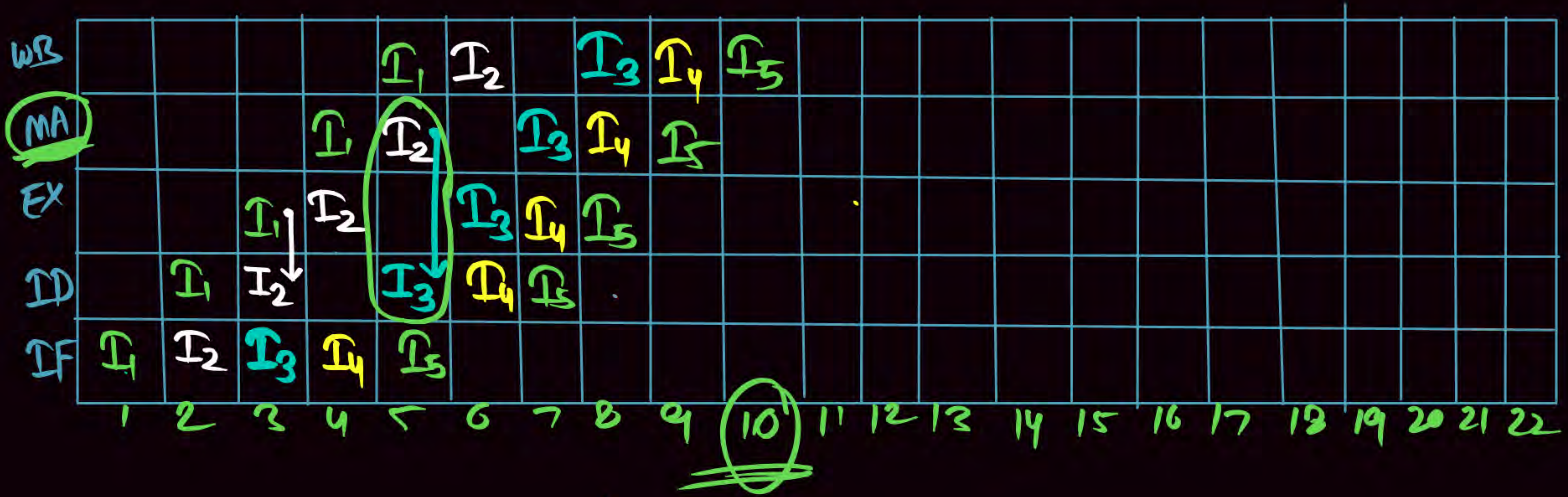
#Q. A pipelined processor uses a 5 stage instruction pipeline with the following stages: **instruction fetch (IF) instruction decode & Operand Fetch (ID), Execute(EX), Memory Access(MA) and write back (WB).**

Consider the following sequence of instruction:

| Instruction | Meaning of instruction |
|---|---|
| $I_1$: ADD $R_0, R_1, R_2$ | //$R_0 \leftarrow R_1 + R_2$ |
| $I_2$:LOAD $R_3$ 100($R_0$) | //$R_3 \leftarrow$ MEM [100 + $R_0$] |
| $I_3$:ADDI $R_4, R_3$, 20 | //$R_4 \leftarrow R_3$ +20 |
| $I_4$:ADD $R_3, R_2, R_1$ | //$R_3 \leftarrow R_2 + R_1$ |
| $I_5$:STORE $R_3$ 100($R_0$) | //MEM [100 + $R_0$] $\leftarrow R_3$ |
| | |

The number of Clock cycle Required in the sequence of instructions Using Operand forwarding 10 Ans

$I_2$ : LOAD ⇒ MA Stage.



| | | | | | | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| WB | | | | | $I_1$ | $I_2$ | | $I_3$ | $I_4$ | $I_5$ | | | | | | | | | | | | |
| MA | | | | $I_1$ | $I_2$ | | $I_3$ | $I_4$ | $I_5$ | | | | | | | | | | | | | |
| EX | | | $I_1$ | $I_2$ | | $I_3$ | $I_4$ | $I_5$ | | | | | | | | | | | | | | |
| ID | | $I_1$ | $I_2$ | | $I_3$ | $I_4$ | $I_5$ | | | | | | | | | | | | | | | |
| IF | $I_1$ | $I_2$ | $I_3$ | $I_4$ | $I_5$ | | | | | | | | | | | | | | | | | |
| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 |

| | | | | | | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| WB | | | | | | | | | | | | | | | | | | | | | | |
| MA | | | | | | | | | | | | | | | | | | | | | | |
| EX | | | | | | | | | | | | | | | | | | | | | | |
| ID | | | | | | | | | | | | | | | | | | | | | | |
| IF | | | | | | | | | | | | | | | | | | | | | | |
| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 |

Now Change in Question

& Change in Stage working

without operand forwarding.

[NAT]

#Q. A pipelined processor uses a 5 stage instruction pipeline with the following stages: **instruction fetch (IF) instruction decode** (ID), **Operand Fetch** (OF) Perform operation (PO) & Memory Access **and write back (WB).**

Consider the following sequence of instruction:

| Instruction | Meaning of instruction |
|---|---|
| $I_1$: ADD $R_0$, $R_1$, $R_2$ | //$R_0 \leftarrow R_1 + R_2$ |
| $I_2$:LOAD $R_3$ 100($R_0$) | //$R_3 \leftarrow$ MEM [100 + $R_0$] |
| $I_3$:ADDI $R_4$, $R_3$, 20 | // $R_4 \leftarrow R_3$ +20 |
| Not Depend $I_4$:ADD $R_3$, $R_2$, $R_1$ | //$R_3 \leftarrow R_2 + R_1$ |
| $I_5$:STORE $R_3$ 100($R_0$) | //MEM [100 + $R_0$] $\leftarrow R_3$ |
| | |

Without The number of Clock cycle Required in the sequence of instructions ~~Using~~ Operand forwarding ____.

|  | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| WB |  |  |  |  | $I_1$ |  |  | $I_2$ |  |  | $I_3$ | $I_4$ |  |  | $I_5$ |  |  |  |  |  |  |  |
| PO |  |  |  | $I_1$ |  |  | $I_2$ |  |  | $I_3$ | $I_4$ |  |  | $I_5$ |  |  |  |  |  |  |  |  |
| OF |  |  | $I_1$ | – | – | $I_2$ |  |  | $I_3$ | $I_4$ |  |  | $I_5$ |  |  |  |  |  |  |  |  |  |
| ID |  | $I_1$ | $I_2$ | $I_3$ | $I_4$ | $I_5$ |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| IF | $I_1$ | $I_2$ | $I_3$ | $I_4$ | $I_5$ |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| WB |  |  |  |  | $I_1$ |  |  | $I_2$ |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| PO |  |  |  | $I_1$ |  |  | $I_2$ |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| OF |  |  | $I_1$ | $I_2$ | $I_2$ | $I_2$ ✓ |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| ID |  | $I_1$ | $I_2$ | Wrong Value |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| IF | $I_1$ | $I_2$ |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 |

# NAT

Which of the following are NOT true in a pipelined processor?

1. Bypassing can handle all RAW hazards. → False [Load & Store (MA Stage)]
2. Register renaming can eliminate all register carried WAR hazards.

Anti
o/p } ⇒ Register Renaming.

[✓ A 1 only]

B 2 only

C 1 and 2 Both

D None of these

# NAT

H.w

An instruction pipeline has five stages, namely, instruction fetch (IF), instruction decode and register fetch (ID/RF), instruction execution (EX), memory access (MEM), and register write back (WB) with stage latencies 1 ns, 2.2 ns, 2ns ,1 ns, and 0.75 ns, respectively (ns stands for nanoseconds). To gain in terms of frequency, the designers have decided to split the ID/RF stage into three stages (ID, RF1, RF2) each of latency 2.2/3 ns. Also, the EX stage is split into two stages (EX1, EX2) each of latency 1 ns. The new design has a total of eight pipeline stages. A program has 20% branch instructions which execute in the EX stage and produce the next instruction pointer at the end of the EX stage in the old design and at the end of the EX2 stage in the new design. The IF stage stalls after fetching a branch instruction until the next instruction pointer is computed. All instructions other than the branch instruction have an average CPI of one in both the designs. The execution times of this program on the old and the new design are P and Q nanoseconds, respectively. The value of P/Q is _____.

[NAT]

Consider a Pipelined processor Operating at 2.5GHZ. Pipeline have 6-segment pipeline, IF, ID, EX1, EX2, MA, and WB. The branch is resolved at the end of 3rd cycle for unconditional branches and at the end of 4th Stage cycle for conditional branches. Assume 30% of all instructions are conditional branches out of which 60% are taken and 5% are unconditional branches or procedure calls. Assume that only the branches instructions result in stall. then what is Average instruction Execution time of this processor? (in ~~Micro~~ Nano Second)

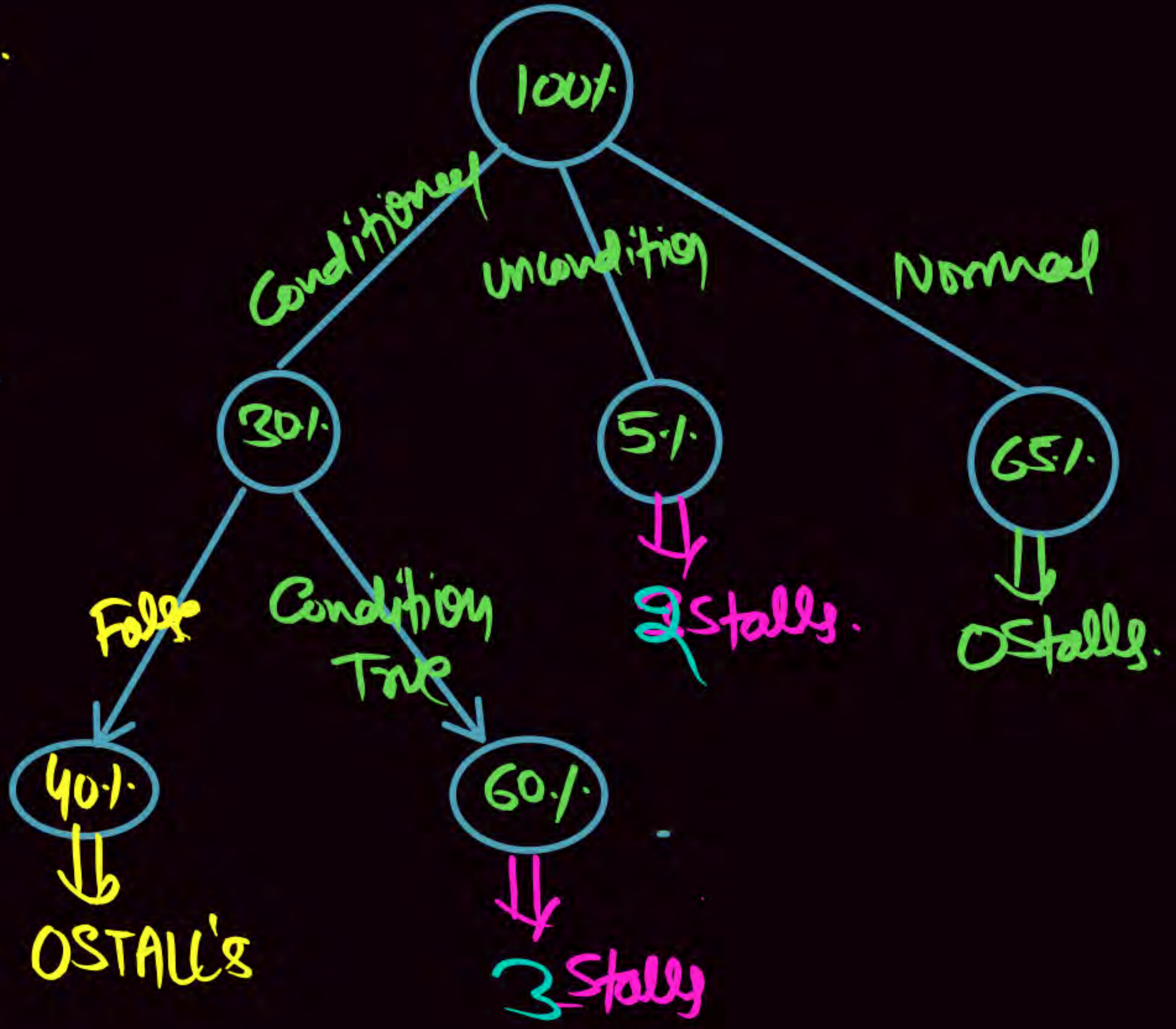
Uncondition ⇒ Stall = 3 − 1 = 2.

Conditional = 4 − 1 = 3

$$\#\text{Stalls}/\text{Instn} = .30 \times .40 \times 0 + .30 \times .60 \times 3 + .05 \times 2 + .65 \times 0.$$

$$= 0.64.$$

$$\text{Cycle time} = \frac{1}{2.54}\text{ sec} = \underline{\underline{0.4\text{ nsec}}}$$



Avg Instn ET = (1 + #stalls/instn) × Cycle time.

$\Rightarrow (1 + 0.64) \times 0.4$ nse

$\Rightarrow 1.64 \times 0.4$ nsec

$\Rightarrow 0.656$ nsec

# NAT

Consider a non-pipelined processor operating at 4 GHz. It takes 7 clock cycles to complete an instruction. You are going to make a 6-stage pipeline out of this processor. Overheads associated with pipelining force you to operate the pipelined processor at 2 GHz. In a given program, assume that 30% are memory instructions, 60% are ALU instructions and the rest are branch instructions. 5% of the memory instructions cause stalls of 50 clock cycles each due to cache misses and 50% of the branch instructions cause stalls of 2 cycles each. Assume that there are no stalls associated with the execution of ALU instructions. For this program, the speedup achieved by the pipelined processor over the non-pipelined processor (round off to 2 decimal places) is ________

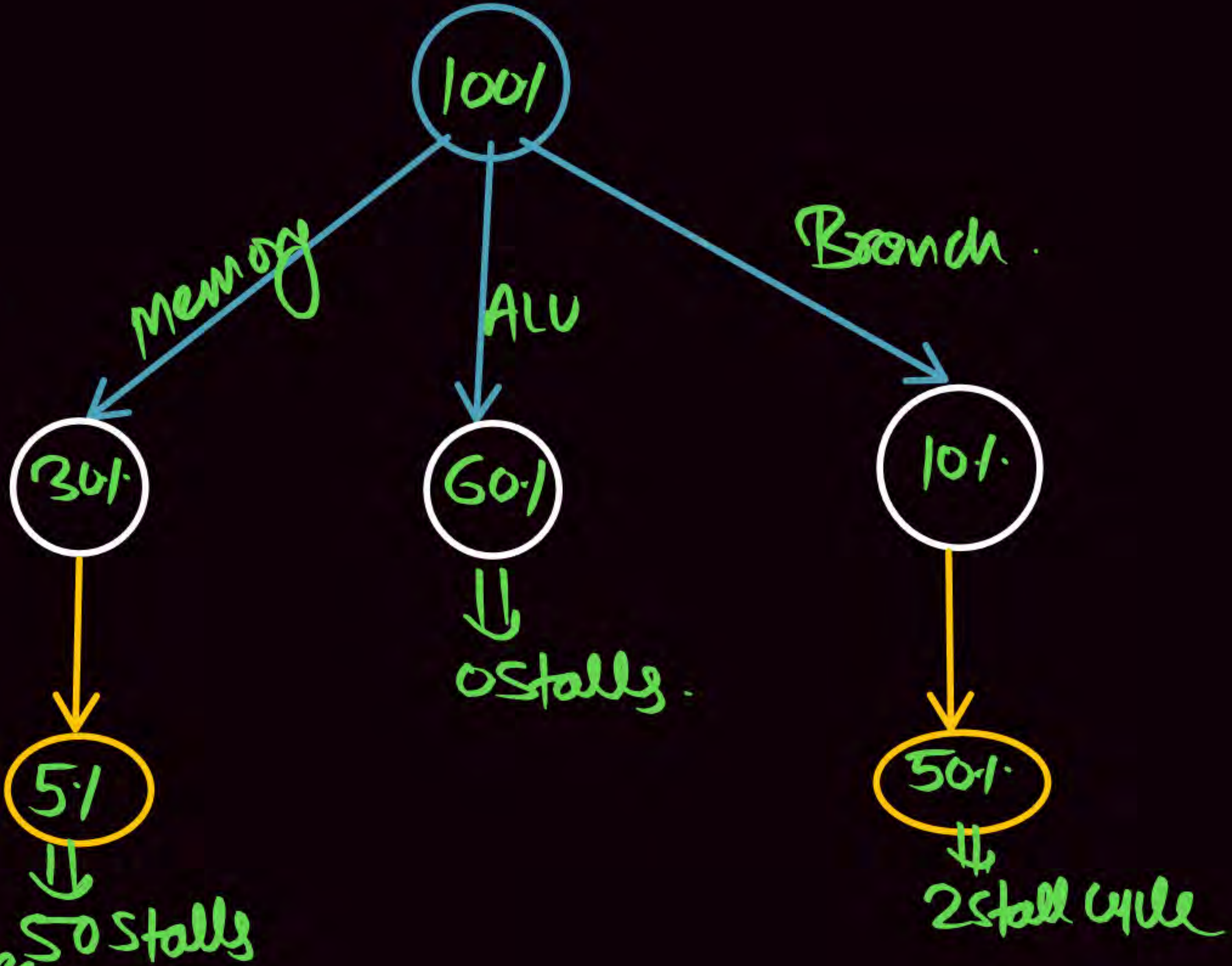
#Stalls/Instn =

$.30 \times 0.05 \times 50 +$

$.60 \times 0 +$

$.10 \times .50 \times 2$

$= 0.85$



Avg Instn ET $= (1 + \frac{\#\text{Stall}}{\text{Ins}})$ cycle

$\Rightarrow (1 + 0.85) \times \frac{1}{2}$ nsec

$= \frac{1.85}{2}$ nsec $= \boxed{0.925 \text{ nsec}}$

Non Pipeline = 4GHz

Cycle time $= \frac{1}{4} = 0.25$ nsec.

7 cycle

CPI x Cycle time.

Non Pipeline $= 7 \times 0.25$

$= 1.75$ nsec.

$$S = \frac{ET_{NP}}{ET_{PIPE}}$$

$$= \frac{1.75}{0.925}$$

$S = 1.89$ Ans

## [NAT]

Consider a pipeline which is operating with a 1.8 GHz frequency contain 8 stages. Pipeline allow overlapping of all the instruction except branch instruction. Processor stop fetching of a sequential instruction after the branch instruction until the outcome is known. All the instruction output is available at the end of execution [Last 8th Stage]. Program contain 40% branch instruction, among them 60% are conditional branch in which 40% instruction does not satisfy the condition (when condition is false) then the following instruction are overlapped then what is the average instruction execution time (in nano seconds) ____? (upto 2 decimal point)

$B.P = 8 - 1 = 7$ Stalls.

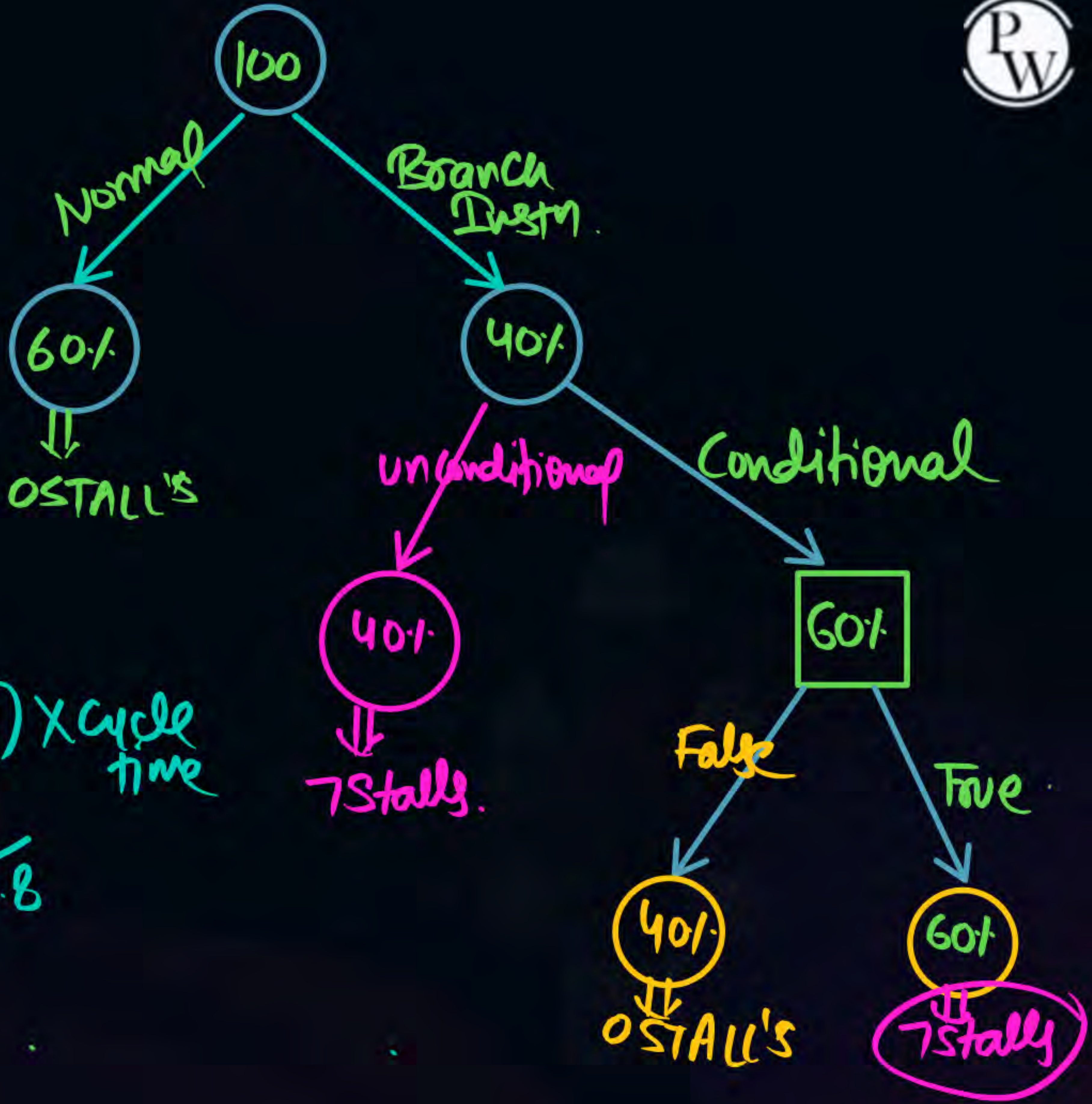
$$\#\text{Stalls/Inst}^n = .60 \times 0 + .40 \times .40 \times 7 + .40 \times .60 \times .40 \times 0 + .40 \times .60 \times .60 \times 7$$

$$\boxed{\#\text{Stalls/Inst}^n = 2.128.}$$

$$\text{Avg Inst}^n\ \text{ET} = (1 + \#\text{Stalls/Inst}^n) \times \text{cycle time}$$

$$\Rightarrow (1 + 2.128) \times \frac{1}{1.8}$$

$$= 1.73\ \text{nsec.}$$



[NAT] H.W

The width of the physical address on a machine is 32 bits, 4 way set associative cache each block hold 2k byte (2 kilo byte) & having word of size 4 bytes. There are 512 blocks in the cache. Initially cache is empty then how many data words are brough into the cache is__________.

[NAT] H.W

If a 4-way set associative cache is made up of 64 bit words. 8 words per line and 8192 sets. What is the size of cache memory (in MB(Mega Bytes) byte)______

# 2 mins Summary

Topic Pipeline

Topic Pipeline Hazards

# THANK - YOU