COMPUTER SCIENCE

Computer Organization and Architecture

Instruction Pipelining

Lecture_03

Vijay Agarwal sir


TOPICS
TO BE
COVERED
01 Pipelining Concepts
01 Pipelining Hazards

PIPELINE CONCEPT

Execution time in Pipeline & Non pipeline.

SPEED UP FACTOR

UNIFORM & NON Uniform Delay Pipeline.

Pipelining Strategy

To apply this concept
To instruction
Execution we must
Recognize that an
Instruction has a
Number of stages

Similar to the use of
An assembly line in a
Manufacturing plant

New inputs are
Accepted at one end
Before previously
Accepted inputs
Appear as output at
The other end

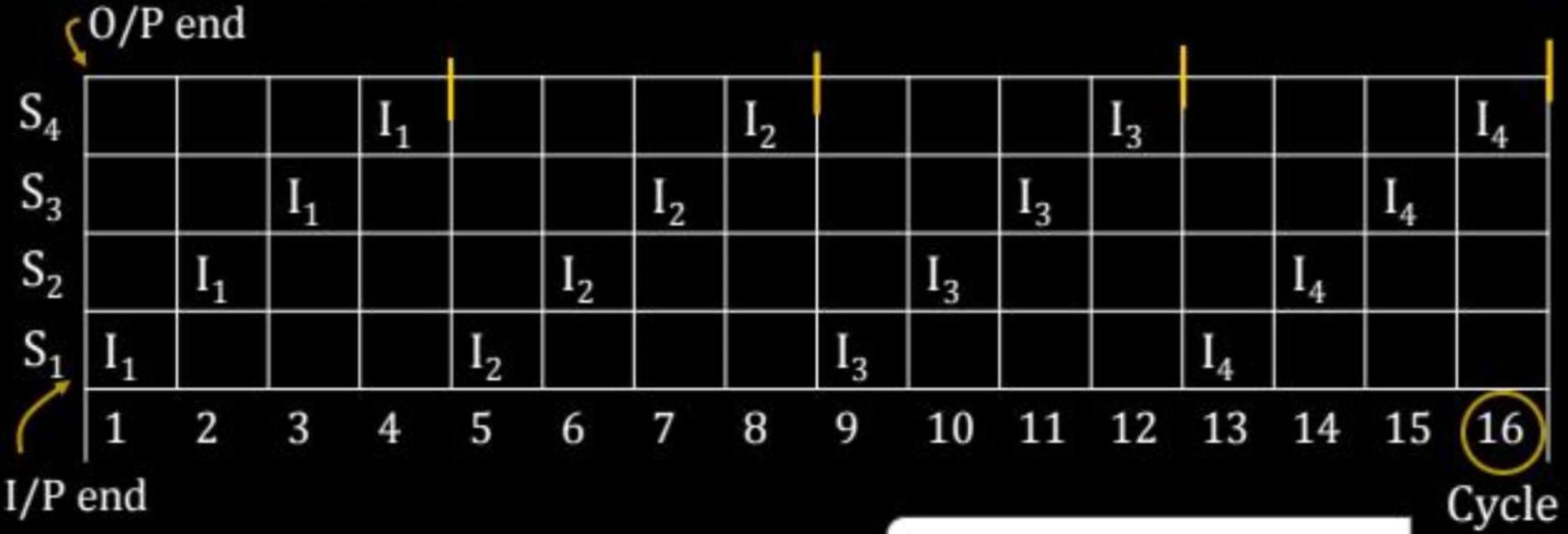
PIPELINE

- Pipelining is a mechanism which is used to improve the performance of the system in which task (Instruction) are executed in overlapping manner.
- Pipelining is a decomposition technique that means the problem is divided into sub problem & Assign the sub problem to the pipes then operate the pipe under the same clock.

Let us consider 4 segment pipeline used to execute 4 instruction the execution sequence as:

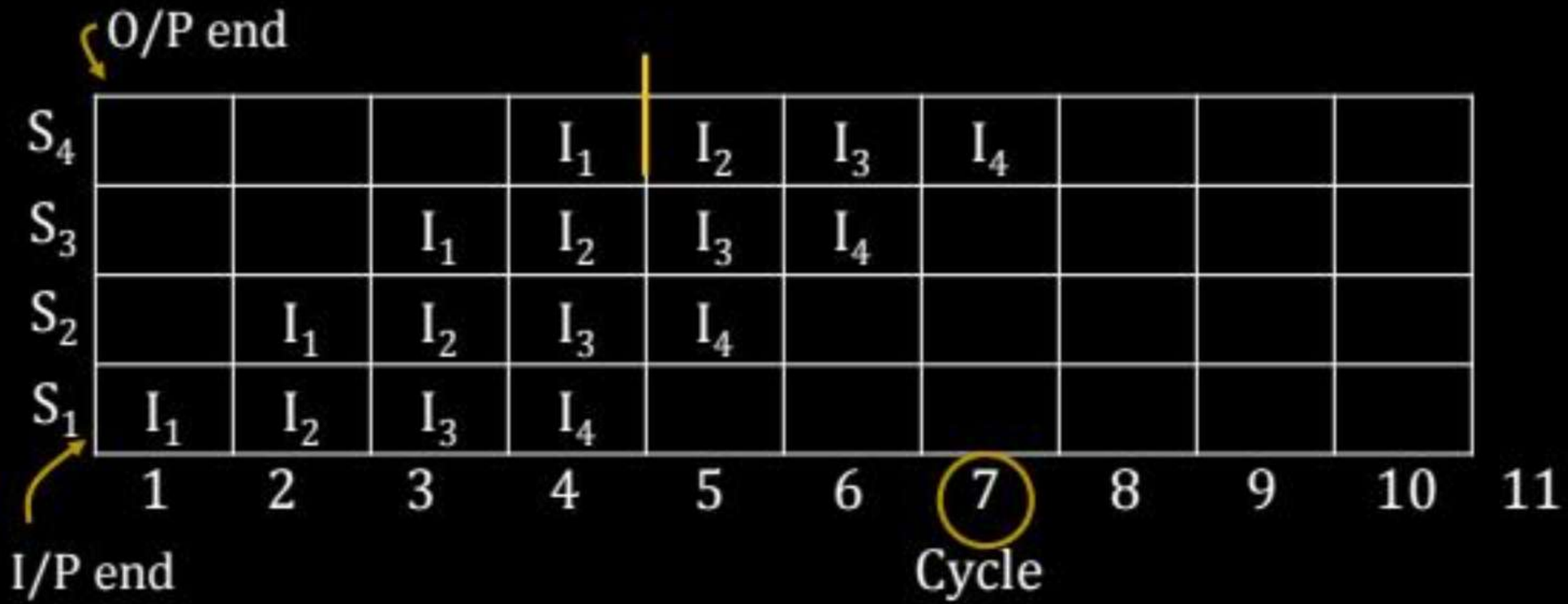
Segment/stages = $[S_1\ S_2\ S_3\ S_4]$

Instruction: $[I_1\ I_2\ I_3\ I_4]$



n = 4, t_n = 4, Non pipeline

Non-PIPELINE



PIPELINE

$k = 4$

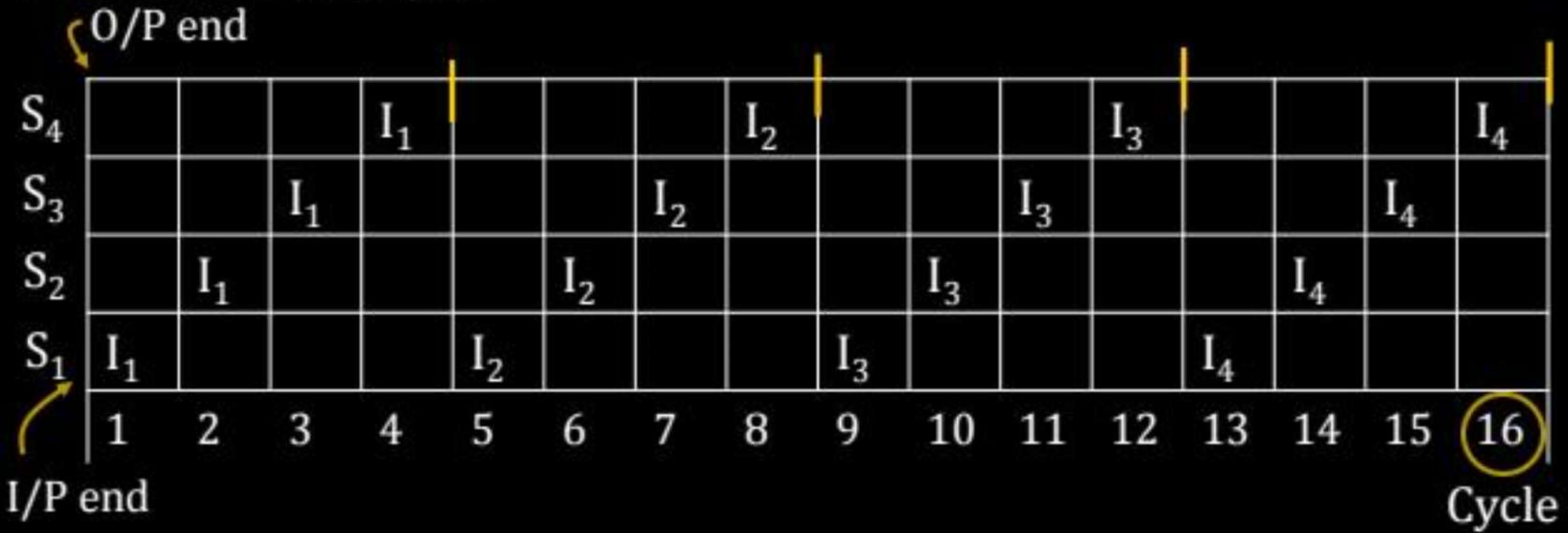
$n = 4$

PIPELINE

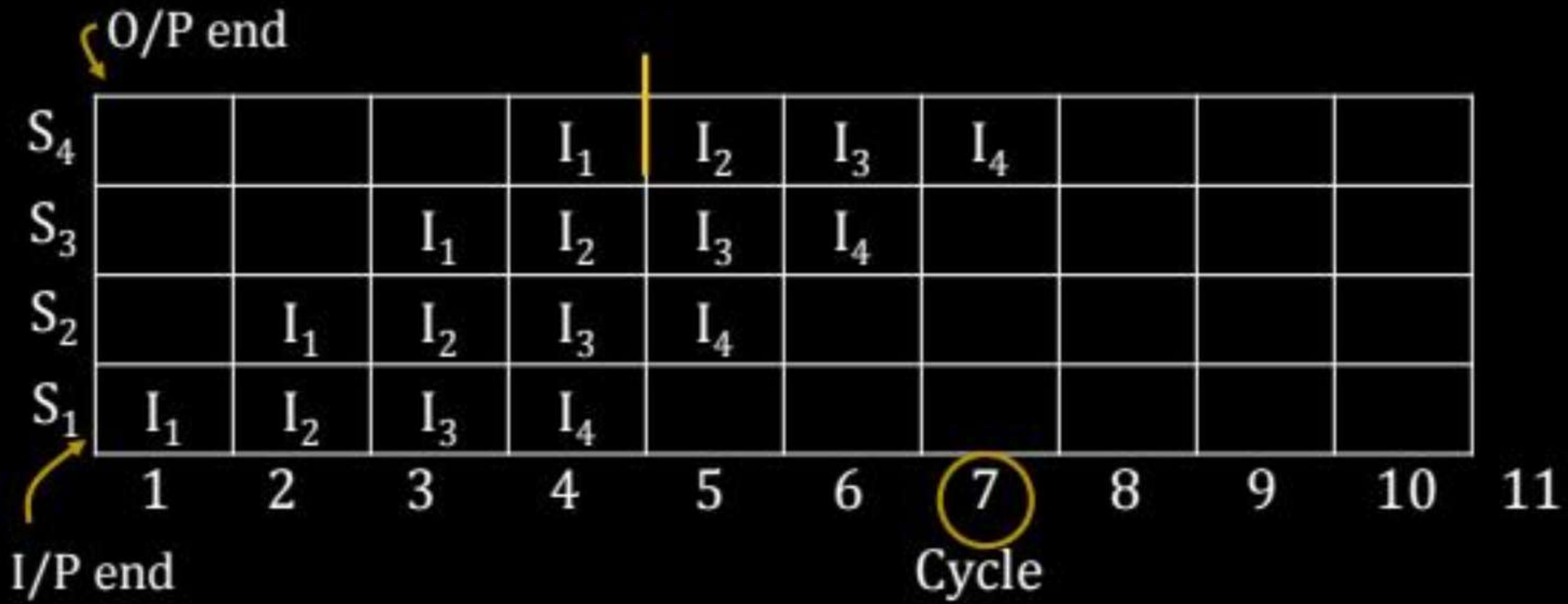
Let us consider 4 segment pipeline used to execute 4 instruction the execution sequence as: PIPELINING

Segment/stages = $[S_1\ S_2\ S_3\ S_4]$

Instruction: $[I_1\ I_2\ I_3\ I_4]$



n = 4, t_n = 4, Non pipeline



PIPELINE

$k = 4$

$n = 4$

In the Pipeline.

$$CPI = 1$$

I. Q.1 How Design [Construct] the Pipeline. ?

II. Q.2 WHAT is the Meaning of CPI = 1 ?

III. Q.3 How to Set this CPI in Uniform Delay ? ?

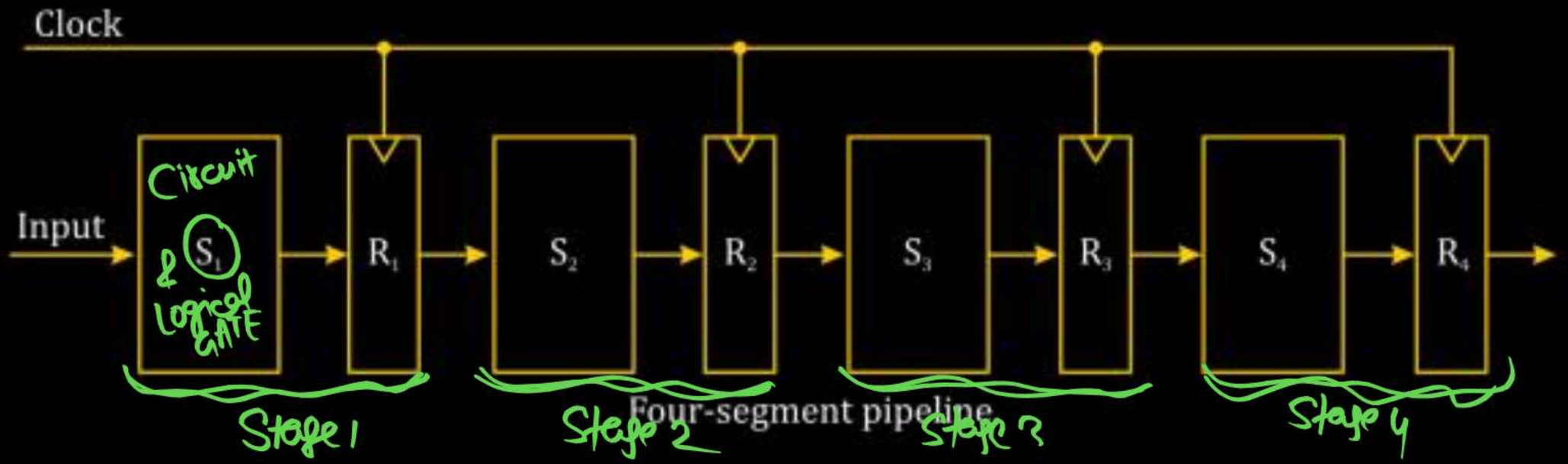
IV. Q.4 How to Set this CPI in Non uniform Delay. ?

PIPELINE Design

@ RISC : 5 Stage

'5'

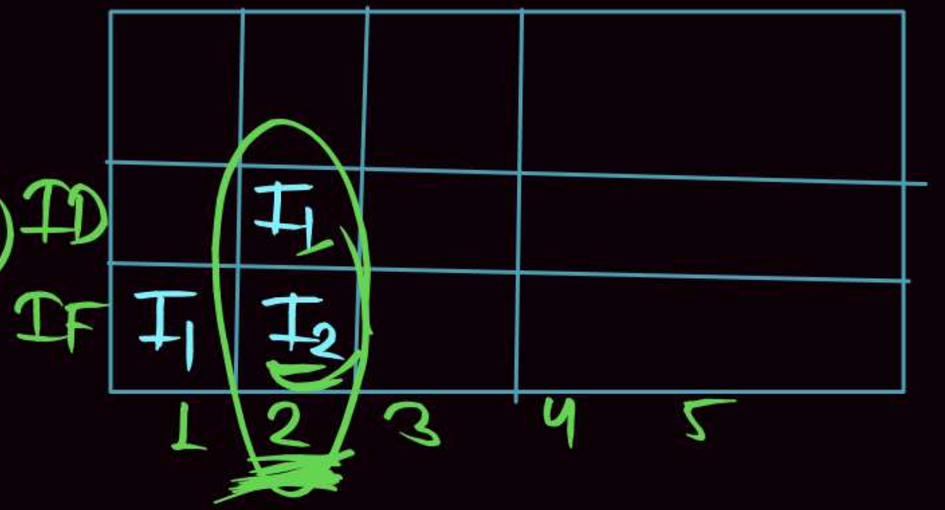
(I) If we Want to Construct 'N' Stage Pipeline, then entire CPU [entire Control Unit] is Divided into 'N' functional Unit [Independent Stages functional Unit] which is independent from each other.



Independent functional Unit means, when One functional Unit Perform the Task, in the same time other functional Unit Performing the other Task (operation).

(eg) Instruction Pipeline

I_1 is in Decode Stage & I_2 is in Fetch Stage. in the Clock cycle No 2.



Note

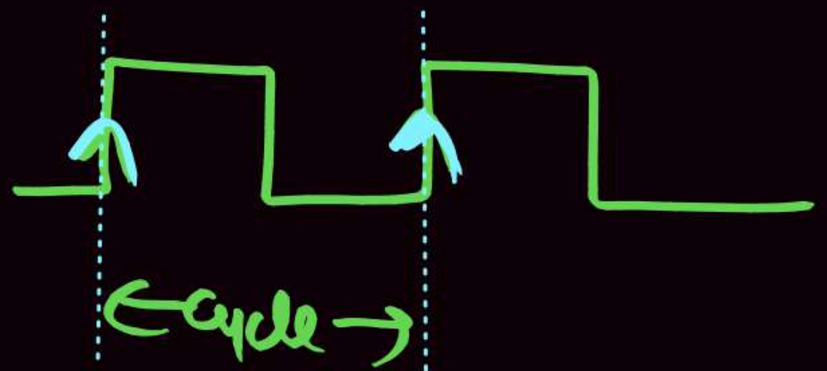
Characteristic of the PIPELINE is, in Every New Cycle New Input must be inserted into the Pipeline.

CPI = 1

Cycle Per Instn = 1

⇓

One Clock Pulse Transition.



In PIPELINE

RISC ⇓ Successful Pipeline

CPI = 1

CPI = 1

Clock cycle per Instruction = 1.

Q. How to set this CPI & WHY clock is Required ?

Q. Why Clock is Required ?

Soln (i) Because whenever, We Enable the clock, the operation are Performed (Data is Moved from One Register to Other Register or etc ~~other task~~)

& (ii) Provide the Synchronization between the Stage.

PIPELINE Design

UNIFORM Delay.

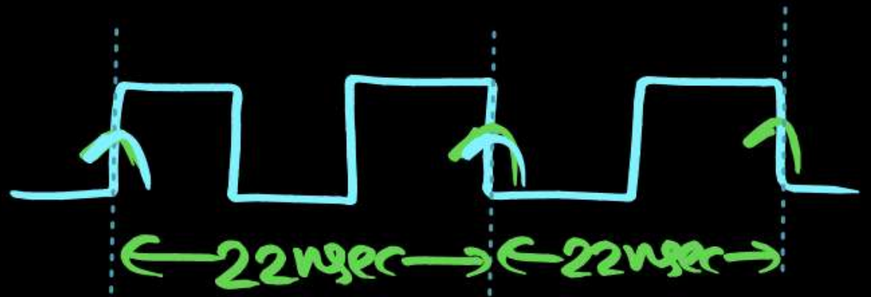
Time = 22nsec.

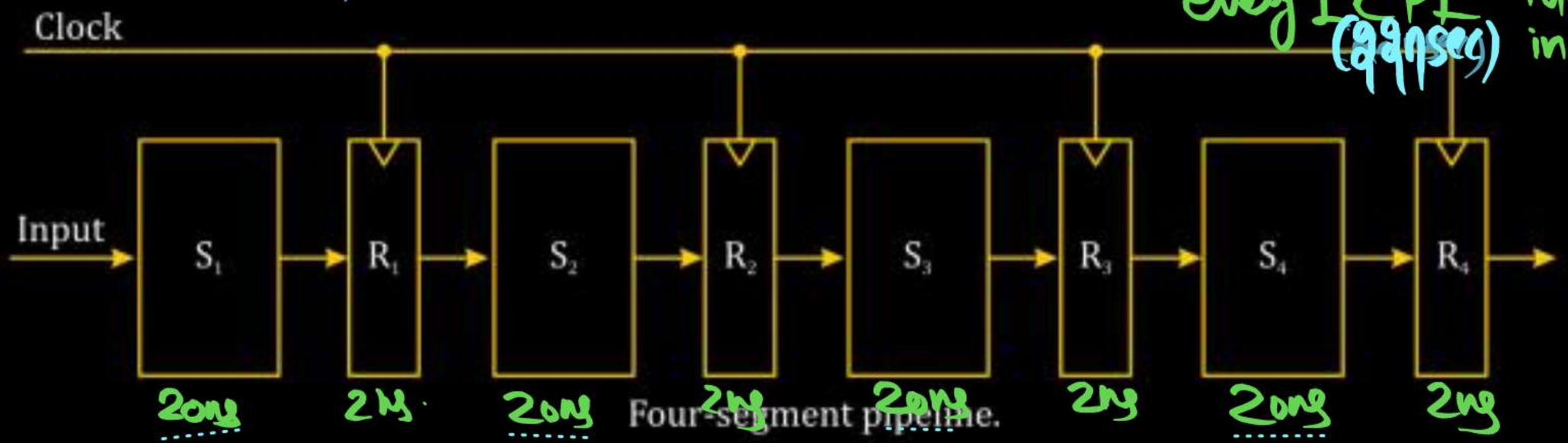
$F = \frac{1}{22\text{nsec}}$ Hz.

CPI = 1

Cycle Per Instⁿ = 1 Cycle

Cycletime = 22nsec





Four-segment pipeline.

every 1 CPI (22nsec) new i/p insert in the Pipeline

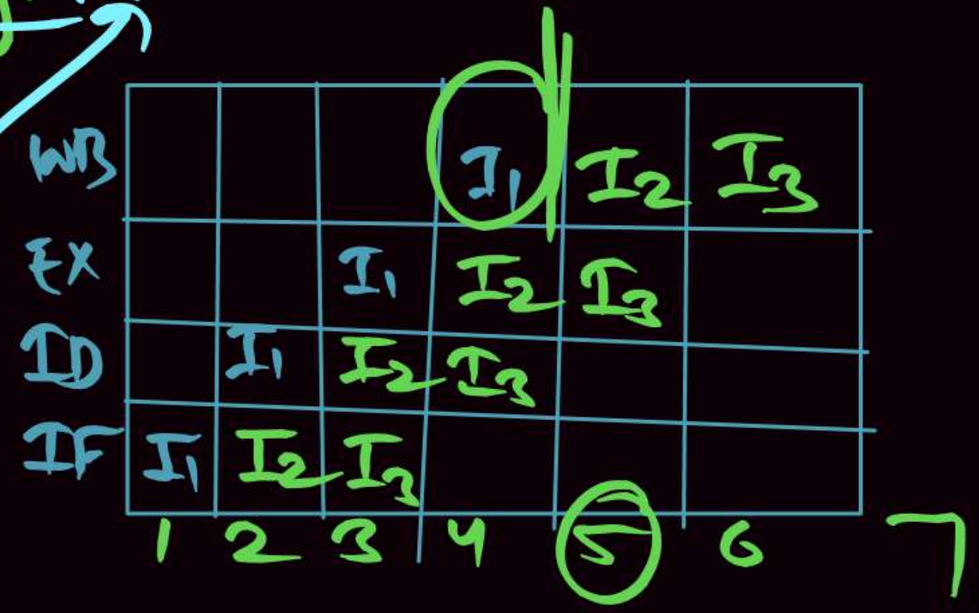
In Uniform Delay PIPELINE

First Instn takes → k * tp → 4×22

Remaining (n-1) Instn → $(n-1)$ 22 nsec

After every 1 cycle we getting O/P.

$(n-1)$ Instn



at 5: I_2

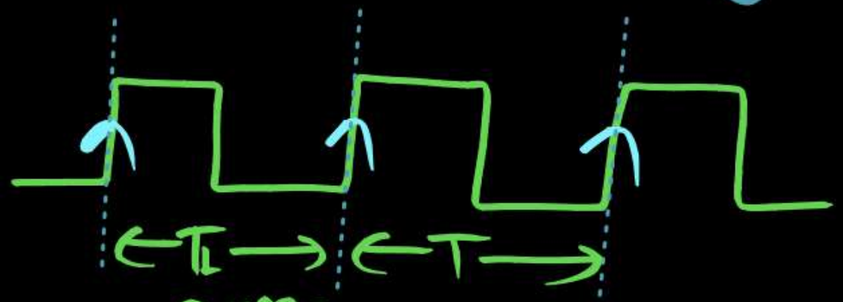
at 6: I_3

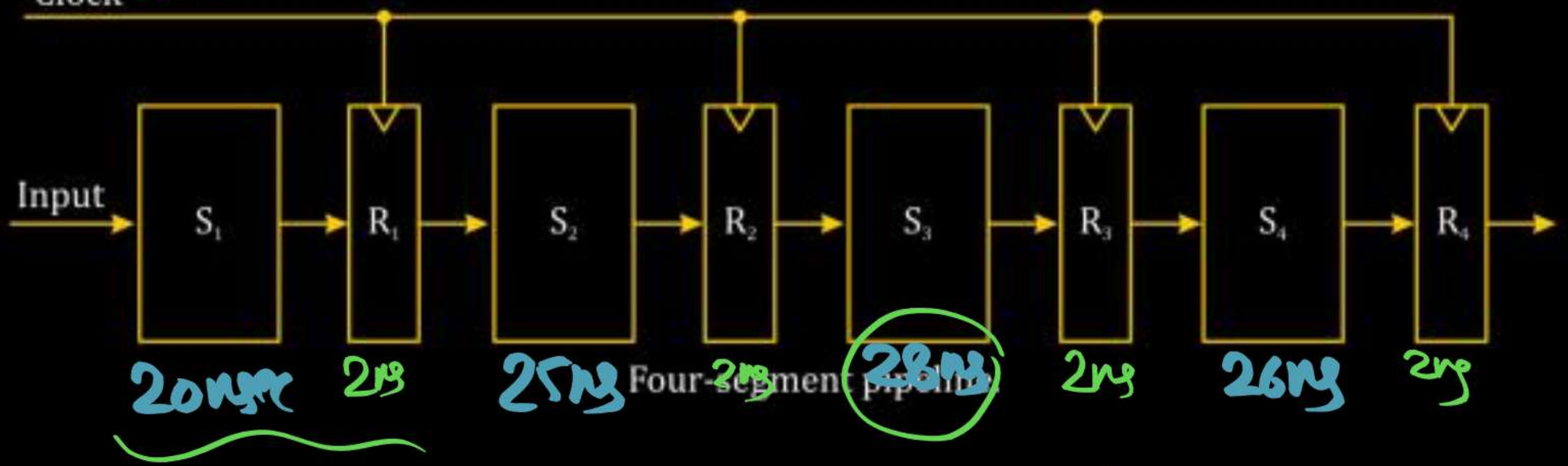
PIPELINE Design

Non Uniform Delay:

$t_p = \max\left[\left(\text{Stage Delay} + \text{Bubbled Delay}\right)\right]$

$t_p = 30\text{nsec}$.





Note

If we take Minimum means (20+2) = 22nsec.

$$\text{Clock} = \frac{1}{22\text{nsec}}\ \text{Hz}.$$

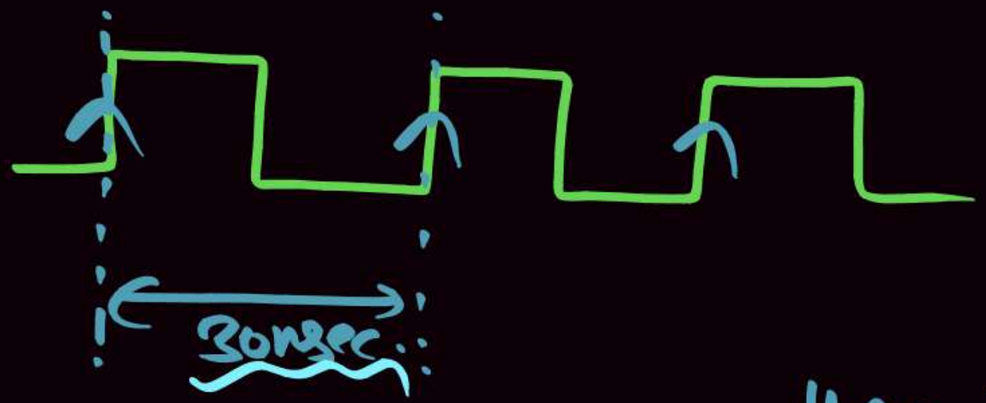
In 22nsec Stage 1 Work finish & Data Movement from Stage 1 to Stage 2.

But in 22ns [In this Clock] Stage 2, Stage 3, Stage 4 (in this example) Task Not Complete [Still Processing]

So Proper Working functioning of the Pipeline max Stage Delay is Taken.
So always take maximum of the (Stage + Buffer Delay)

Maximum.

So t_p = 30nsec.



One Instruction coming out every 30nsec.

Synchronization.

If we take Maximum = 30ns.

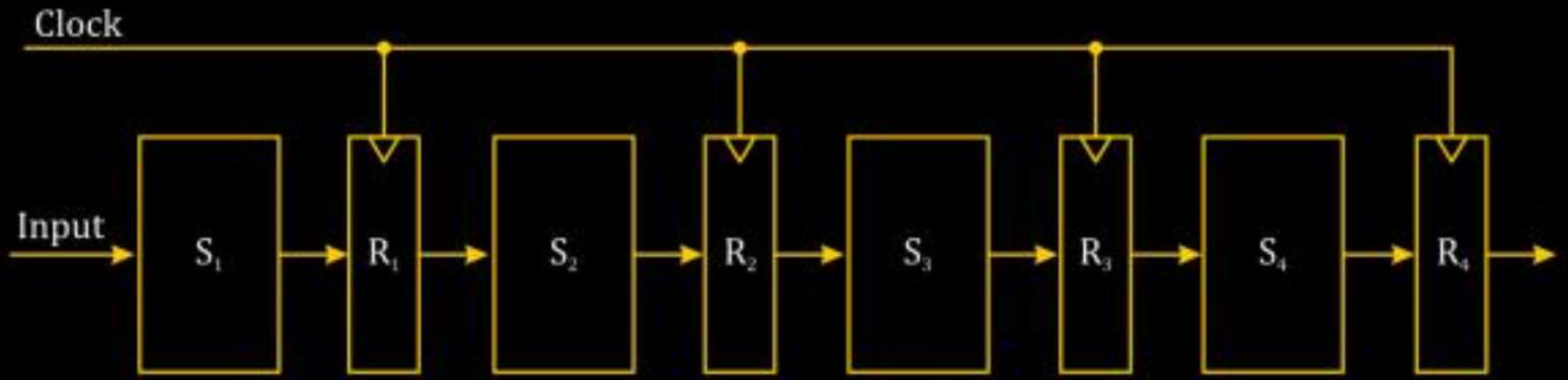
then in S_2 Task finish in 22nsec.

but clock is set to be 30nsec.

Every Phase output will be available at 30nsec [When 1 cycle(30ns) then stage 1 O/P given to S_2 & so on]

So synchronization.

PIPELINE Design



Four-segment pipeline.

PIPELINE Design

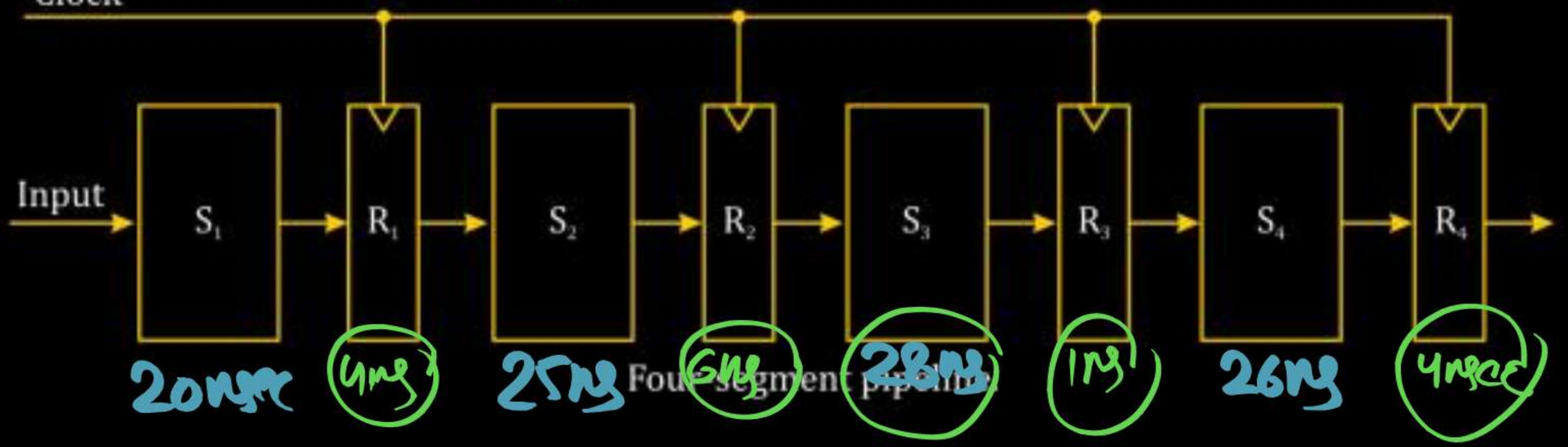
Non Uniform Delay:
& Register Delay is Different

$$t_p = \max\left[\left(\text{Stage Delay} + \text{Buffered Delay}\right)\right]$$

$t_p = 31$

max(20+4, 25+6, 28+1, 26+4)

max(24, 31, 29, 30)



PIPELINE Performance Evaluation

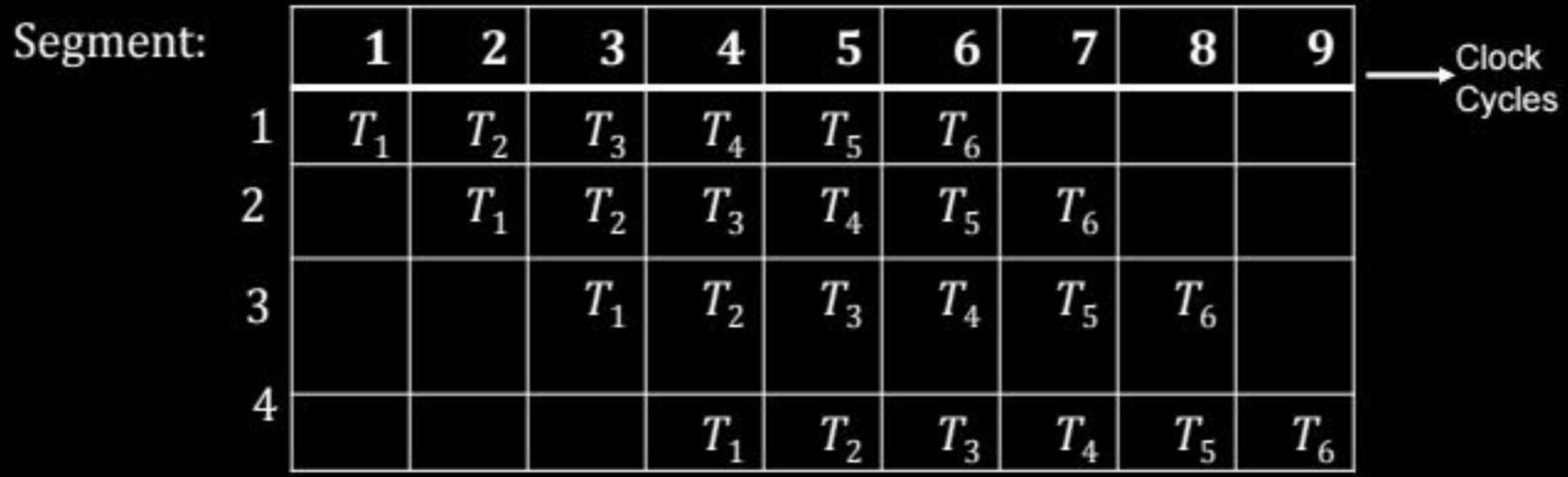
Now consider the case where a k-segment pipeline with a clock cycle time t_p, is used to execute n tasks.

The first task T_1 requires a time equal to kt_p, to complete its operation since there are k segments in the pipe.

The remaining n – 1 tasks emerge from the pipe at the rate of one task per clock cycle and they will be completed after a time equal to $(n - 1)t_p$.

Therefore, to complete n tasks using a k-segment pipeline requires $k + (n - 1)$ clock cycles.

Space-time diagram for pipeline.



For example, the diagram of Fig. 9-4 shows four segments and six tasks. The time required to complete all the operations is 4 + (6 – 1) = 9 clock cycles, as indicated in the diagram.

So, to complete n tasks using a *k*-segment pipeline is:

ET pipe= $k + (n - 1)$ clock cycles

So, to complete n tasks in Non- pipeline is:

ET Non-pipeline= $n * tn$

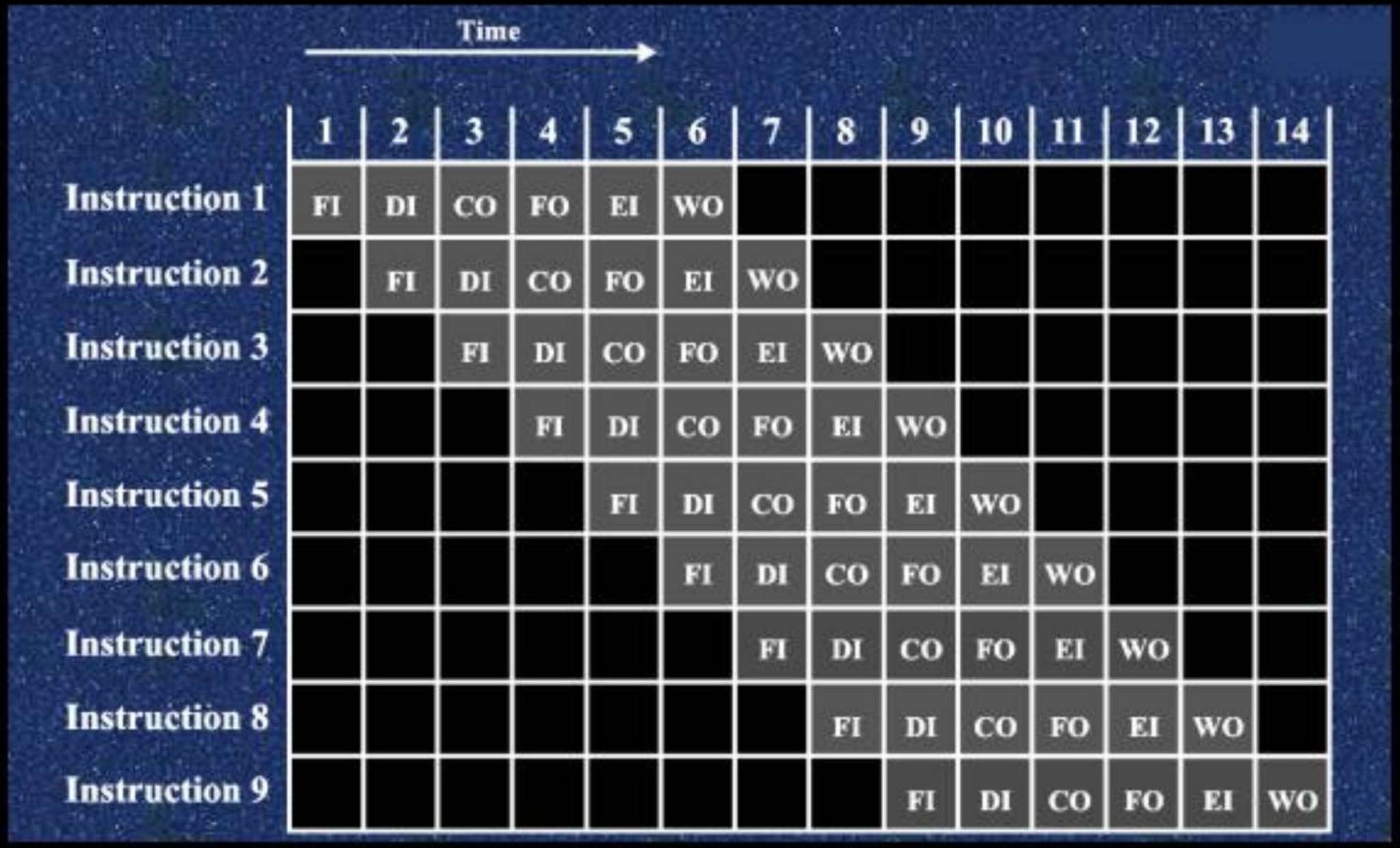
$$S = \frac{nt_n}{(k+n-1)t_p}$$

As the number of tasks increases, n becomes much larger than $k - 1$, and $k + n - 1$ approaches the value of n. Under this condition, the speedup becomes

$$S = \frac{t_n}{t_p}$$

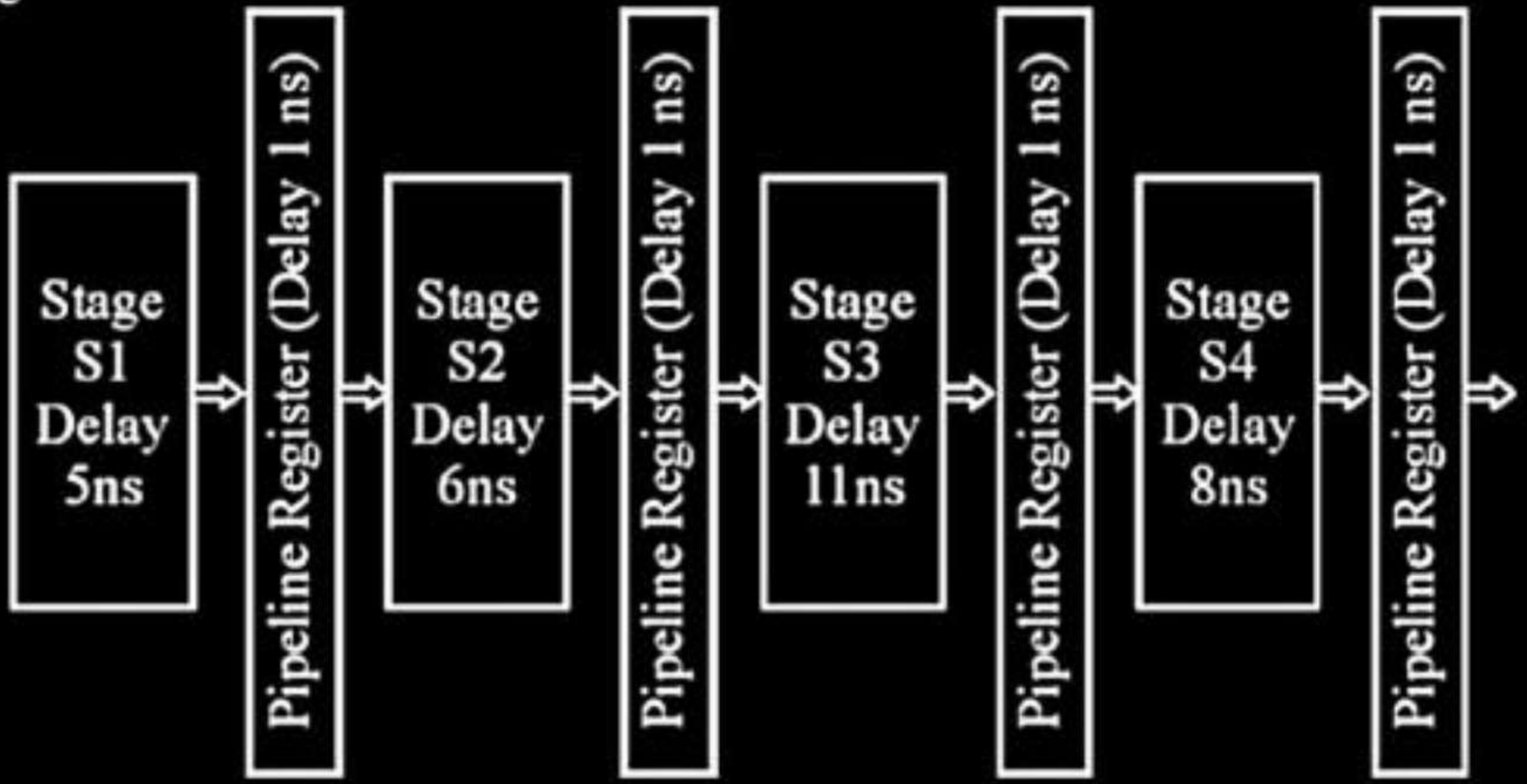
Additional Stages

- **Fetch Instruction (FI)**
  - Read the next expected Instruction into a buffer.
- **Decode Instruction (DI)**
  - Determine the opcode and the operand specifiers.
- **Calculate operands(CO)**
  - Calculate the effective address of each source operand.
  - This may involve displacement, register indirect or other forms of address calculations.
- **Fetch Operands(FO)**
  - Fetch each operand from memory.
  - Operands in register need not be fetched.
- **Executed Instruction(EI)**
  - Perform the indicated operation and store the result, if any, in the specified destination operand location
- **Write Operand(WO)**
  - Store the result in memory



Timing Diagram for Instruction pipeline operation

Q. 4 Consider an instruction pipeline with four stages (S1, S2, S3 and S4) each with combinational circuit only. The pipeline registers are required between each stage and at the end of the last stage. Delays for the stages and for the pipeline registers are as given in the figure



When Motivated Uniform Delay & ideal Case then S=4

What is the approximate speed up of the pipeline in steady state under ideal conditions when compared to the corresponding non-pipeline implementation?

[GATE-2011: 2 Marks]

In Ideal Case : $S = \frac{t_n}{t_p} = \frac{30}{12} = 2.5$

A 4.0

B 2.5

C 1.1

D 3.0

But we Discussed In Ideal Case

Speed up Factor = k (#Stages) = 4

But Answer is 2.5 Why?

Sol'n Bcz its Non uniform Delay Pipeline.

When Speed Factor = 4 { if uniform Delay & Register Delay is Not given
or No Time is given for any Stage only theory (Ideal given)

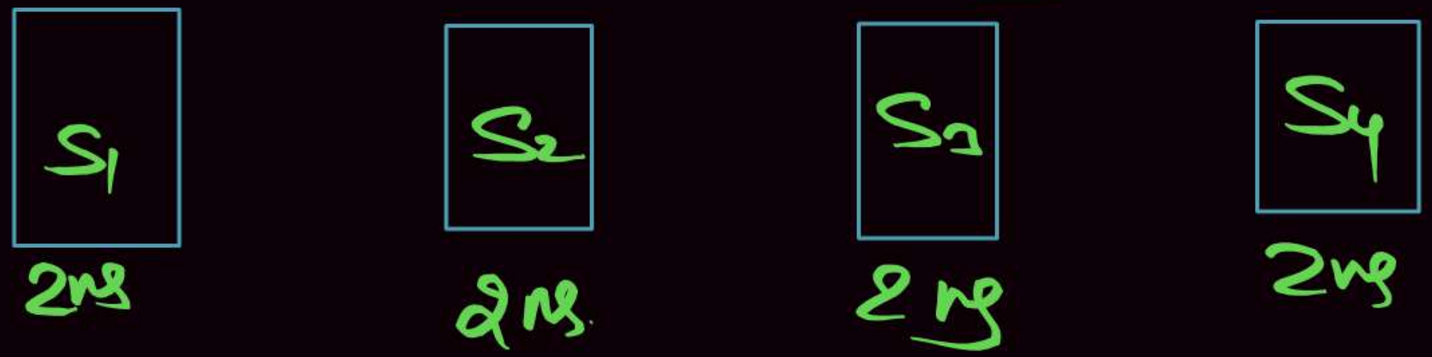
Uniform Delay Pipeline

$t_p = 2ns$

$t_n = 2+2+2+2 \Rightarrow k * t_p \Rightarrow 4 \times 2 = 8ns$

$t_n = k * t_p$

8ns



$S = \frac{t_n}{t_p}$

$= \frac{k * t_p}{t_p} \Rightarrow \boxed{S = k}$

Uniform Delay Pipeline.

NAT Q. 9

The stage delays in a 4-stage pipeline are 800, 500, 400 and 300 picoseconds. The first stage (with delay 800 picoseconds) is replaced with a functionally equivalent design involving two stages with respective delays 600 and 350 picoseconds. The throughput increase of the pipeline is ____ percent.

[GATE-2016(Set-1)-CS: 2M]

NAT Q. 10

Consider a non-pipelined processor with a clock rate of 2.5 gigahertz and average cycles per instruction of four. 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 gigahertz. Assume that there are no stalls in the pipeline. The speed up achieved in this pipelined processor is 3.2 Ans

[GATE-2015(Set-1)-CS: 2M]

Ans (3.2)

2.5GHz Cycle time $= \frac{1}{2.5 \times 10^9}$ sec $= 0.4 \times 10^{-9}$ sec $= 0.4$ nsec

Non Pipeline

Each Instruction takes : 4 cycle

Cycle time $= \frac{1}{2.5 \times 10^9}$ sec $= 0.4$ nsec

ET in Non Pipeline $(t_n) = 4 \times 0.4$

$= 1.6$ nsec

2GHz time $= \frac{1}{2 \times 10^9} = 0.5$ ns

In Pipeline

1 Instn takes = 1 Cycle

CPI = 1 Cycle

Cycle time $= \frac{1}{2 \times 10^9}$ sec $= 0.5$ nsec

ET in Pipeline = 0.5 nsec

$S = \frac{ET_{NP}}{ET_P} \Rightarrow \frac{1.6}{0.5} = 3.2$ Ans

MCQ Q. 11

Consider a 4-stage pipeline processor. We want to execute a loop: For(i=1;i<=1000;i++){ I1, I2, I3, I4} where the time taken (in ns) by instruction I1 to I4 for stages S1, S2, S3, S4 is shown below:

	S1	S2	S3	S4
I1	1	2	1	2
I2	2	1	2	1
I3	1	1	2	1
I4	2	1	2	1

The Output of I1 for i=2 will be available after ?

[GATE-2004-CS: 2M]

A 11ns

Ans (c).

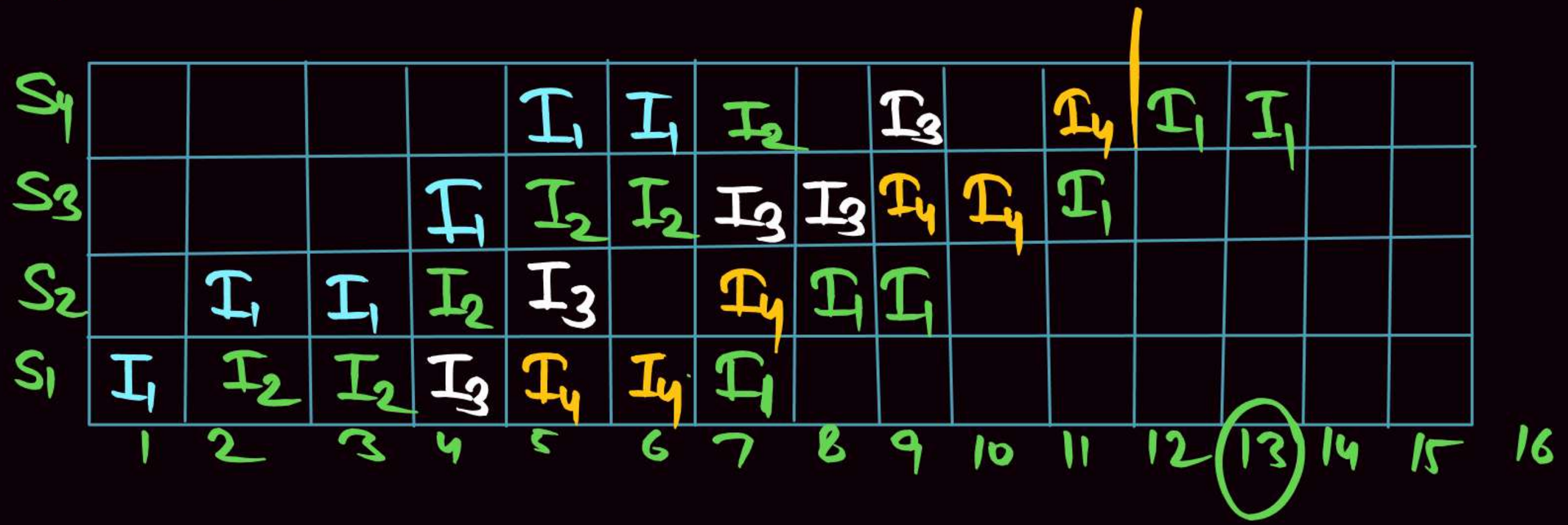
B 12ns

C 13ns

D 28ns

	S_1	S_2	S_3	S_4
I_1	1	2	1	2
I_2	2	1	2	1
I_3	1	1	2	1
I_4	2	1	2	1

$i=1$ $[I_1\ I_2\ I_3\ I_4]$.



. ∴

MCQ Q. 12

Consider a 4-stage pipeline processor. The number of cycles needed by the four instructions I1, I2, I3, I4 in stages S1, S2, S3, S4 is shown below:

	S1	S2	S3	S4
I1	2	1	1	1
I2	1	3	2	2
I3	2	1	1	3
I4	1	2	2	2

What is the number of cycles needed to execute the following loop?
for (i = 1 to 2) {I1; I2; I3; I4;}

[GATE-2009-CS: 2M]

A 16

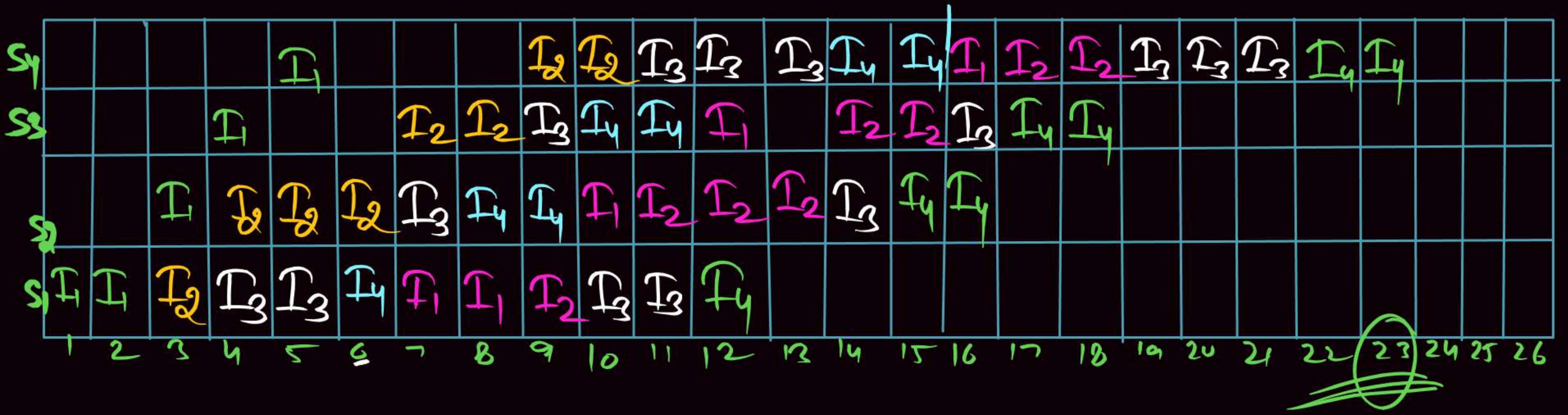
Ans (B).

B 23

C 28

D 30

	S_1	S_2	S_3	S_4
I_1	2	1	1	1
I_2	1	3	2	2
I_3	2	1	1	3
I_4	1	2	2	2



NAT Q. 13

Consider a 3 GH_z (gigahertz) processor with a three-stage pipeline and stage latencies τ_1, τ_2, and τ_3 such that $\tau_1 = 3\tau_2/4 = 2\tau_3$. If the longest pipeline stage is split into two pipeline stages of equal latency, the new frequency is _____ GH_z, ignoring delays in the pipeline registers.

Ans (4 GHz)

[GATE-2016(Set-2)-CS: 2M]

$$\tau_1 = \frac{3\tau_2}{4} = 2\tau_3$$

$T_1 : T_2 : T_3 = 6 : 8 : 3$

$$\left(\tau_1 = \frac{3\tau_2}{4}\right) \& \left(\frac{3\tau_2}{4} = 2\tau_3\right)$$

$$\frac{\tau_1}{\tau_2} = \frac{3}{4} \quad \frac{6}{8} \qquad \frac{\tau_2}{\tau_3} = \frac{8}{3}$$

$\therefore$

$T_1 : T_2 : T_3 = 6 : 8 : 3$

Assume time x

$T_1 : 6x$, $T_2 : 8x$ $T_3 = 3x$.

$t_p : \max(6x, 8x, 3x)$

$$t_p = 8x$$

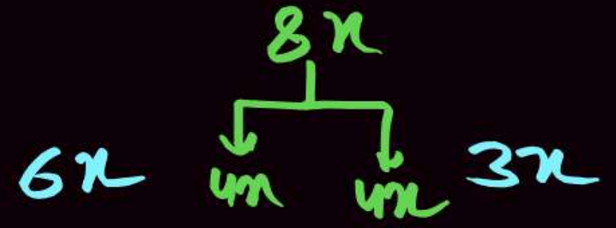
$$\text{Frequency} = \frac{1}{\text{Time}}$$

$$3\text{GHz} = \frac{1}{8x}$$

$$\frac{1}{x} = 24\text{GHz} \quad \text{— eq ①}$$

New Design

Largest Stage Split into 2 equal Stage Delay



$t_{p_{new}} = \max(6x, 4x, 4x, 3x)$

$$t_{p_{new}} = 6x$$

$$\text{Frequency}_{new} = \frac{1}{t_{p_{new}}}$$

$$F_{new} = \frac{1}{6x} \Rightarrow \frac{1}{6} \times \frac{1}{x}$$

$$F_{new} = 4\text{GHz} \quad \text{Ans}$$

$\frac{1}{6} \times 24\text{GHz}$

NAT Q. 14

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 $1GH_Z$, then the clock frequency of P_2 (in GH_Z) is _____.

[GATE-2014(Set-1)-CS: 2M]

$ET_{P_1} = \#Instn \times Cycle\ time \times CPI$

$\#Instn$ Not given in P_1 & P_2 (& Same Instn)

$ET_{P_1} = CPI \times Cycle\ time.$

$ET_{P_2} = 1.2\ CPI \times Cycle\ time_2$

$ET_{P_2} = 0.75\ ET_{P_1}$

MCQ Q. 15

Consider the following processor design characteristics:

I. Register-to-register arithmetic operations only.

II. Fixed-length instruction format.

III. Hardwired control unit.

Which of the characteristics above are used in the design of a RISC processor?

[GATE-2018-CS: 1M]

A I and II only

B II and III only

C I and III only

Ans (D)

D I, II and III

Q. **COA 2022: CS**

NIC PYQ SERIES

Consider a machine with 40 MHz processor which has run a benchmark program. The executed program consists of 100,000 (1 Lakh) instruction executions, with the following instruction mix and clock cycle count. What will be the effective CPI, MIPS rate, and execution time.

Instruction Type	Instruction Count	Cycles/ Instructions
Integer arithmetic	45000 (45%)	1
Data Transfer	32000 (32%)	2
Floating point	15000 (15%)	2
Control transfer	8000 (8%)	2

$$ET = \sum [IC \times CPI] \times \text{Cycle time}$$

$$.45 \times 1 + .32 \times 2 + .15 \times 2 + 0.08 \times 2$$

$$T \propto \frac{1}{F}$$

$$= 0.45 + 0.64 + 0.30 + 0.16$$

$$\text{Cycle time} = \frac{1}{40 \times 10^6}$$

$$\text{Avg CPI} = 1.55 \text{ Cycle}$$ Avg

$$\text{Avg Instn ET} = 1.55 \times \frac{1}{40} \times 10^{-6}$$

$$\text{Avg Instn}_{ET} = 0.03875 \times 10^{-6}$$

(ii) 1 Instn ET = 0.03875×10^{-6} Sec

In 1 Sec = $\frac{1}{0.03875} \times 106$

$\Rightarrow$ **25.8 MIPS.** Ans

10^5 Instn 1000 00

Total Prog ET = $1.55 \times \frac{1}{40 \times 10^6} \times 10^5$

$\frac{1.55}{40} \times 10^{-1}$

$\Rightarrow \frac{1.55}{4} \times 10^{-2}$

$= 0.387 \times 10^{-2}$

Prog ET $= 3.87 \times 10^{-3}$ Ans 3.87 msec

A CPI:3.55; MIPS: 30; Execution time:1.87 ms

B CPI:1.55; MIPS: 25.8; Execution time:3.87 ms

C CPI:5.60; MIPS: 45.8; Execution time:2.87 ms

D CPI:2.55; MIPS: 35.8; Execution time:4.87 ms

Q.

COA 2017: IT

NIC PYQ SERIES

Which of the following is not a form of main memory ?

A. Instruction cache

B. Instruction register

C. Instruction opcode

D. Translation look-aside buffer

Q. COA 2017: CS

NIC PYQ SERIES

In a 10-bit computer instruction format, the size of address field is 3-bits. The computer uses expanding OP code technique and has 4 two-address instructions and 16 one-address instructions. The number of zero address instructions it can support is

A 256

C 640 ✓

B 356

D 756

$\#\text{Opcode in AF} = 12 \times 2^{7-4}$

$\Rightarrow 12 \times 2^3$

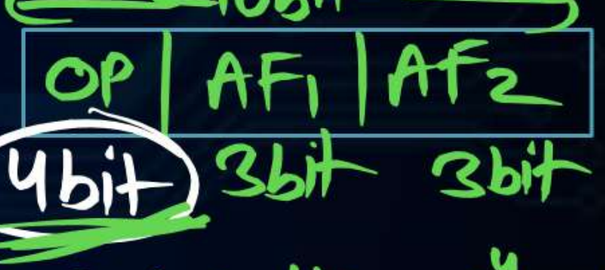
$= 96$

Given = 16

Free = 96 − 16 = 80

$80 \times 2^{10-7} \Rightarrow 80 \times 2^3 = 80 \times 8 = 640$

Primitive [lowest opcode bit]



Total operation $= 2^4 = 16$

Given = 4

Free opcode = 16 − 4

= 12

Q.

COA 2017: CS

NIC PYQ SERIES

A micro programmed control unit

A. is faster than a hardwired unit

B. Facilitates easy implementation of a new instruction

C. is useful when small programs are to be run

D. All of the above

THANK YOU!