PIPELINING

Slides prepared by:
Dr. Zubair Ahmad Shah
Department of Computer Science and Engineering
Islamic University of Science and Technology, Awantipora

Pipelining

- Pipelining is a technique of decomposing a sequential process into suboperations, with each subprocess being executed in a special dedicated segment that operates concurrently with all other segments.
- The result obtained from computation in each segment is transferred to the next segment in the pipeline.
- The final result is obtained when data have passed through all segments.
- The overlapping of computation is made possible by associating a register with each segment in the pipeline.

Example

We want to perform combined multiply and add operations with a stream of numbers:

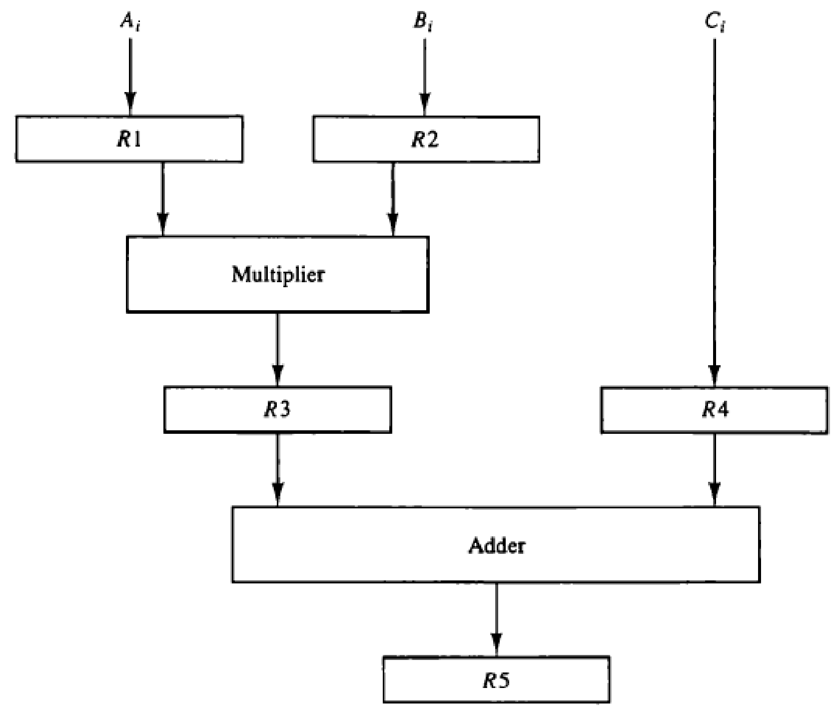
$A_i * B_i + C_i \quad \text{for } i = 1, 2, 3, \ldots, 7$

$R1 \leftarrow A_i, \quad R2 \leftarrow B_i$	Input A_i and B_i
$R3 \leftarrow R1 * R2, \quad R4 \leftarrow C_i$	Multiply and input C_i
$R5 \leftarrow R3 + R4$	Add C_i to product

Suboperations performed in each segment of the pipeline

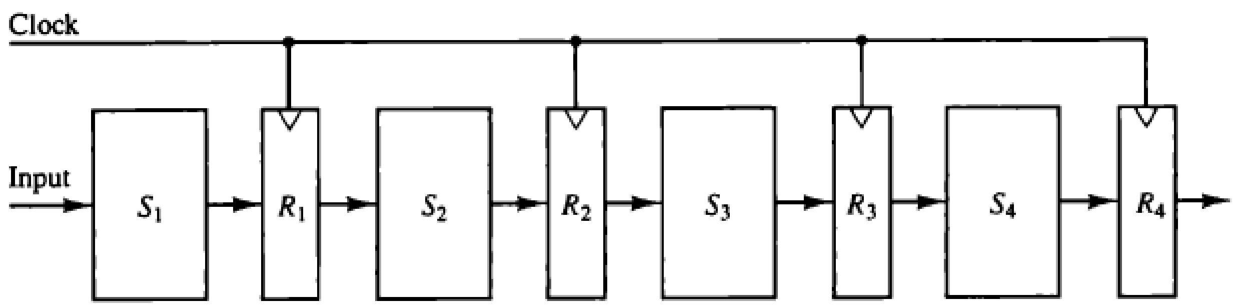
Clock Pulse Number	Segment 1		Segment 2		Segment 3
	$R1$	$R2$	$R3$	$R4$	$R5$
1	A_1	B_1	—	—	—
2	A_2	B_2	$A_1 * B_1$	C_1	—
3	A_3	B_3	$A_2 * B_2$	C_2	$A_1 * B_1 + C_1$
4	A_4	B_4	$A_3 * B_3$	C_3	$A_2 * B_2 + C_2$
5	A_5	B_5	$A_4 * B_4$	C_4	$A_3 * B_3 + C_3$
6	A_6	B_6	$A_5 * B_5$	C_5	$A_4 * B_4 + C_4$
7	A_7	B_7	$A_6 * B_6$	C_6	$A_5 * B_5 + C_5$
8	—	—	$A_7 * B_7$	C_7	$A_6 * B_6 + C_6$
9	—	—	—	—	$A_7 * B_7 + C_7$

Content of Registers in Pipeline Example



Example of pipeline processing.

General Structure of a Pipeline and *Space-time* Diagram



General structure of a four-segment pipeline

Segment:	1	2	3	4	5	6	7	8	9
1	T_1	T_2	T_3	T_4	T_5	T_6			
2		T_1	T_2	T_3	T_4	T_5	T_6		
3			T_1	T_2	T_3	T_4	T_5	T_6	
4				T_1	T_2	T_3	T_4	T_5	T_6

Clock cycles

Spacetime diagram for a four-segment pipeline

Space-time diagram shows segment utilization as a function of time.

QUESTION:

a) A nonpipeline unit takes t_n nsec to process each task. What will be the time taken by it to process n tasks?

b) A k-segment pipeline has a a clock cycle time of t_p nsec. What will be the time taken by it to process the same n tasks?

c) What is the speedup of the second system over the first?

SOLUTION:

a) Total time taken by a nonpipeline unit to process n tasks = $n * t_n$ nsec

b) Segments in the pipe = k

$\Rightarrow$ First task will complete in k clock cycles.

Each of the remaining (n - 1) tasks will complete in 1 clock cycle

=> The remaining (n - 1) tasks will complete in (n - 1) clock cycles.

$\Rightarrow$ Total time taken by the pipeline to complete n tasks = k + (n - 1) clock cycles

= $(k + n - 1) * t_p$ nsec

c) Speedup = $\dfrac{n * t_n}{(k + n - 1) * t_p}$

Speedup of pipeline processing over equivalent nonpipeline processing = $\dfrac{n * t_n}{(k + n - 1) * t_p}$

As the number of tasks (n) increase,

k + n -1 approaches the value of n.

=> Speedup (S) = $\dfrac{t_n}{t_p}$

Assuming that the time it takes to process a task is the same in pipeline and nonpipeline circuits,

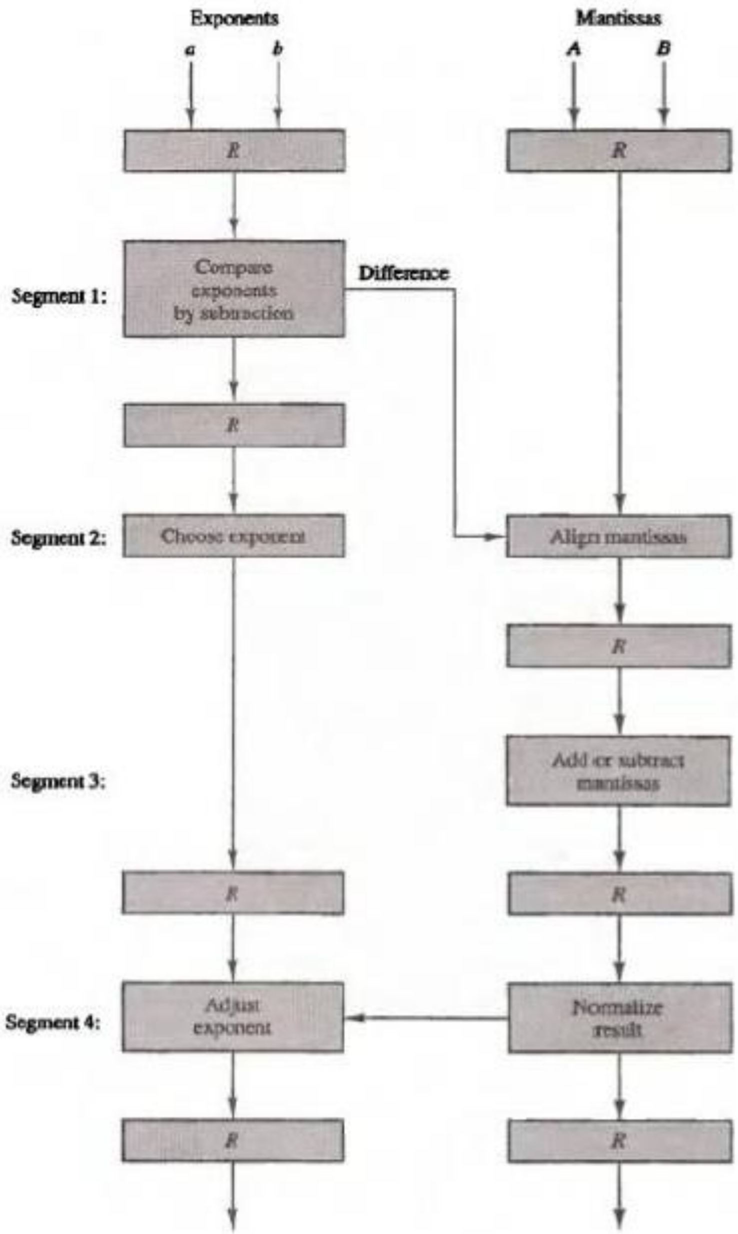
$$S = \frac{k * t_p}{t_p} = k$$

=> The theoretical maximum speedup that a pipeline can provide = k

i.e. number of segments in the pipeline.

Arithmetic Pipeline

- Pipeline arithmetic units are usually found in very high speed computers.
- They are used to implement floating-point operations, multiplication of fixed point numbers and similar computations.



Pipeline for floating-point addition and subtraction.

$$X = 0.9504 \times 10^3$$
$$Y = 0.8200 \times 10^2$$

The two exponents are subtracted in the first segment to obtain $3 - 2 = 1$. The larger exponent 3 is chosen as the exponent of the result. The next segment shifts the mantissa of Y to the right to obtain

$$X = 0.9504 \times 10^3$$
$$Y = 0.0820 \times 10^3$$

This aligns the two mantissas under the same exponent. The addition of the two mantissas in segment 3 produces the sum

$$Z = 1.0324 \times 10^3$$

The sum is adjusted by normalizing the result so that it has a fraction with a nonzero first digit. This is done by shifting the mantissa once to the right and incrementing the exponent by one to obtain the normalized sum.

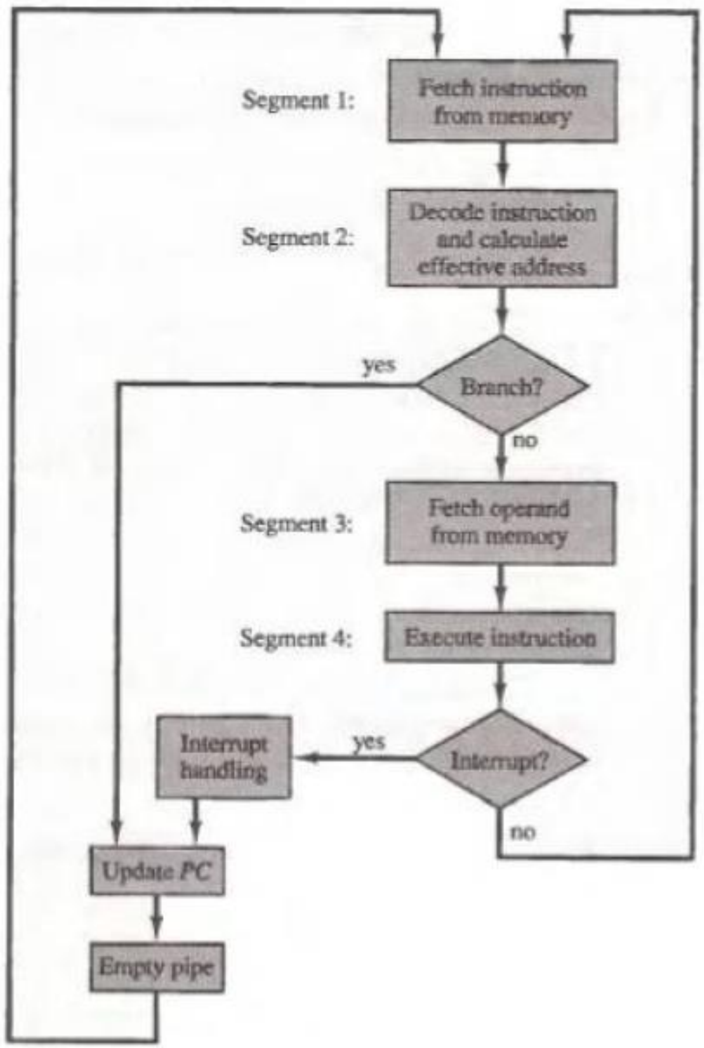
$$Z = 0.10324 \times 10^4$$

Instruction Pipeline

- Pipeline processing can occur not only in data stream but in the instruction stream as well.

- **Instruction Cycle:**

1. Fetch the instruction from memory.
2. Decode the instruction.
3. Calculate the effective address.
4. Fetch the operands from memory.
5. Execute the instruction.
6. Store the result in the proper place.



Four-segment CPU pipeline.

Step:		1	2	3	4	5	6	7	8	9	10	11	12	13
Instruction:	1	FI	DA	FO	EX									
	2		FI	DA	FO	EX								
(Branch)	3			FI	DA	FO	EX							
	4				FI	–	–	FI	DA	FO	EX			
	5					–	–	–	FI	DA	FO	EX		
	6									FI	DA	FO	EX	
	7										FI	DA	FO	EX

Timing of instruction pipeline.

- For reference see your textbook:
  - Mano, M. Morris. *Computer system architecture*. Prentice-Hall of India, 2003.