# pipelining

Presented By
**Dr. Banchhanidhi Dash**

**School of Computer Engineering**
**KIIT University**

# Pipeline Basic

Pipelining is one of the most importantand popular technique that is used to enhance the performance of a processor.
it is an implementation technique, which is done in the hardware and it exploits different types of parallelism.

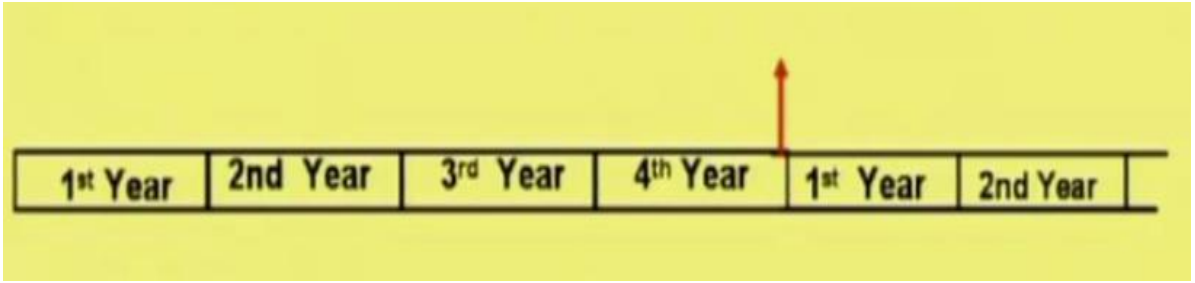
example of pipelining in our day to day life.
:putting the fire off is a pipelining technique.

5 buckets of water is moving from the pond to the fire simultaneously.

Consider the two ways the engineering college works

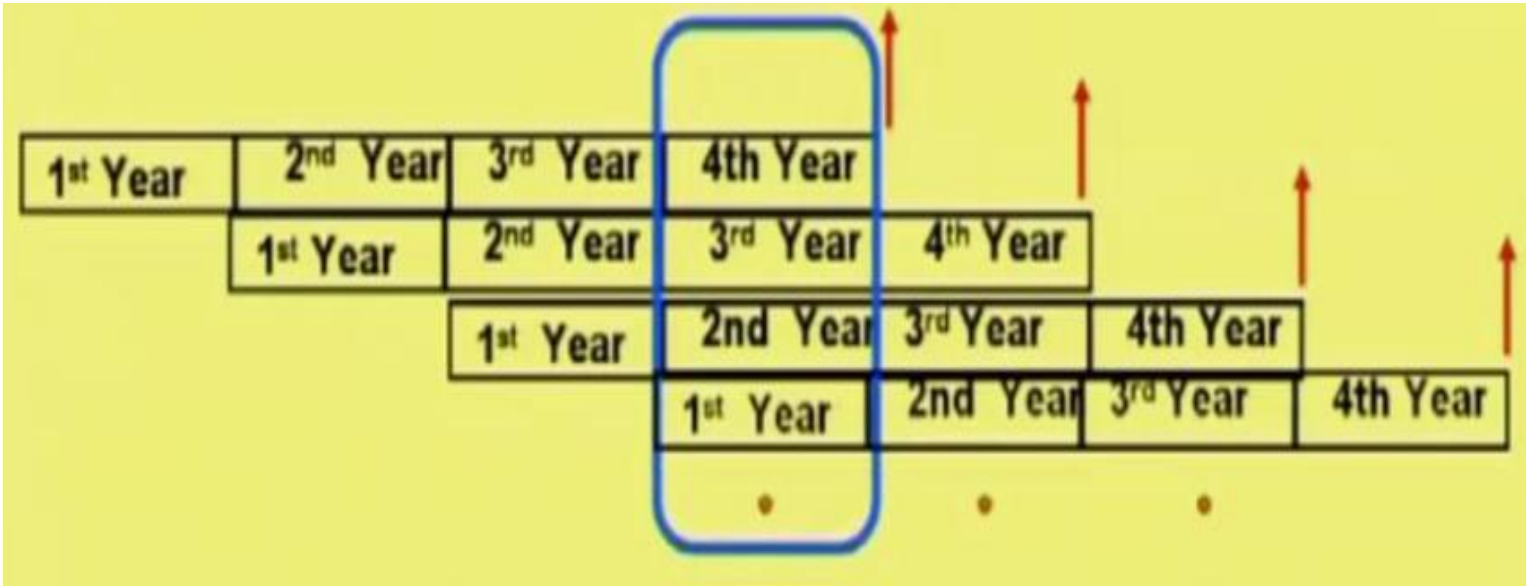
**approach-1**, in which admission take place only when the batch of student passes out from the college.

in every four year, one batch is passing out, so throughput is 1 by 4.




**approach-2 Admits students every year**

**Average no. of students graduating per year increases 4 times.**



# Pipelining in Computer Architecture

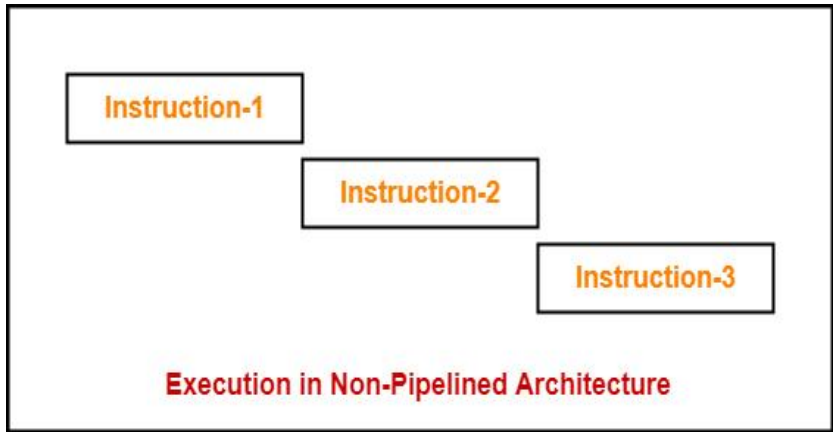
- ✓ A program consists of several number of instructions.
- ✓ These instructions may be executed in the following two ways-


## Non-Pipelined Execution

All the instructions of a program are executed sequentially one after the other.

A new instruction executes only after the previous instruction has executed completely.

This style of executing the instructions is highly inefficient.




If time taken for executing one instruction = t, then-

Time taken for executing 'n' instructions = n x t

# What is Pipelining

- Pipelining : Pipelining is a process of arrangement of hardware elements of the CPU such that its overall performance is increased. Simultaneous execution of more than one instruction takes place in a pipelined processor.
- Pipelining incorporates the concept of overlapped execution:
  - Used in many everyday applications without our notice.
- Has proved to be a very popular and successful way to exploit ILP:
  - Instruction pipes are being used in almost all modern processors.

**Pipelined Execution-**

- In pipelined architecture, Multiple instructions are executed parallelly.This style of executing the instructions is highly efficient
- A pipelined processor does not wait until the previous instruction has executed completely.
- Rather, it fetches the next instruction and begins its execution..

To introduce pipelining in a processor P, the following steps must be followed:

- Sub-divide the input process into a sequence of subtasks. These subtasks will make stages of pipeline
- Each stage Si of the pipeline according to the subtask will perform some operation on a distinct set of operands.
- When stage Si has completed its operation, results are passed to the next stage Si+1 for the next operation.
- The stage Si receives a new set of input from previous stage Si-1 .

**Five -Stage Pipeline-**

In 5 stage pipelined architecture, the execution of each instruction is completed in following 5 stages-

- Instruction fetch (IF)
- Instruction decode (ID)
- Instruction Execute (Ex)
- Memory operation(MEM)
- Write back (WB)

To implement five stage pipeline,

✓ The hardware of the CPU is divided into five functional units.

✓ Each functional unit performs a dedicated task.

## Pipeline Stages

**RISC processor** has 5 stage instruction pipeline to execute all the instructions in the RISC instruction set. Following are the 5 stages of RISC pipeline with their respective operations:

**Stage 1 (Instruction Fetch)**

In this stage the CPU reads instructions from the address in the memory whose value is present in the program counter.

**Stage 2 (Instruction Decode)**

In this stage, instruction is decoded and the register file is accessed to get the values from the registers used in the instruction.

**Stage 3 (Instruction Execute)**

In this stage, ALU operations are performed.

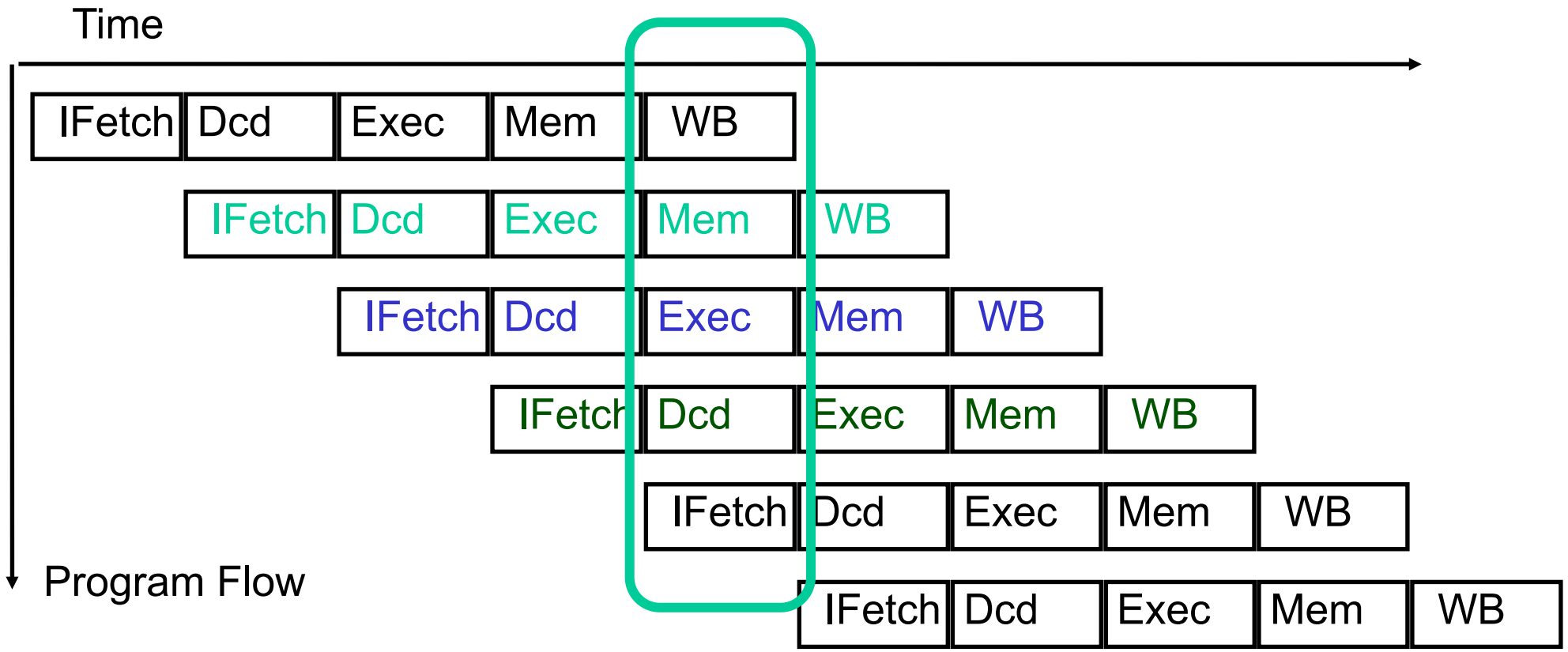
**Stage 4 (Memory Access)**

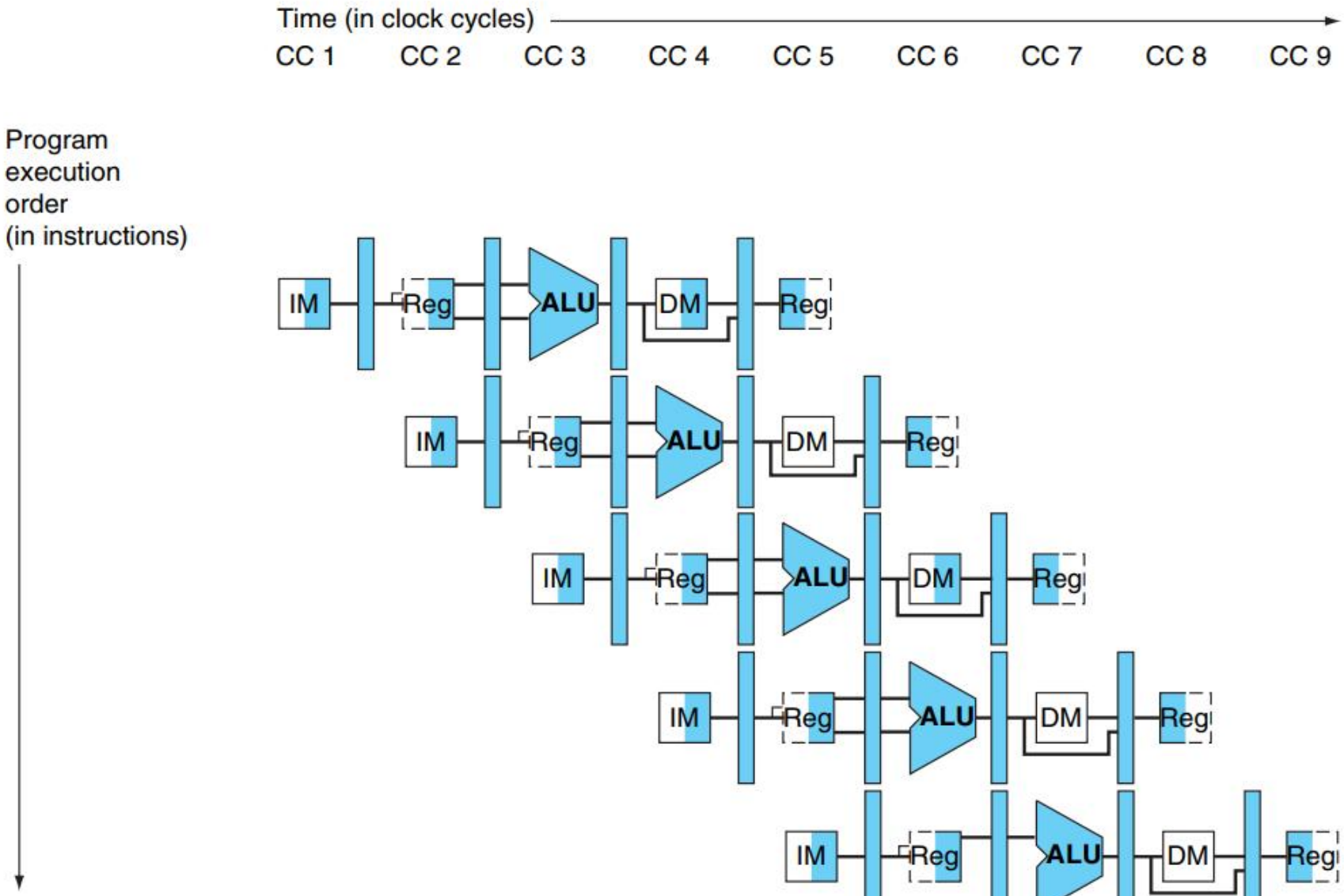
In this stage, memory operands are read and written from/to the memory that is present in the instruction.

**Stage 5 (Write Back)**

In this stage, computed/fetched value is written back to the register present in the instructions.

# Pipelined Execution






**multiple cycle pipeline diagram for 5 instruction**

## Execution-

In pipelined architecture,Instructions of the program execute parallely.

When one instruction goes from nth stage to (n+1)th stage, another instruction goes from (n-1)th stage to nth stage.

## Phase-Time Diagram-

- ✓ Phase-time diagram shows the execution of instructions in the pipelined architecture.
- ✓ The following diagram shows the execution of five instructions in five stage pipeline architecture.

| | | CC 1 | CC 2 | CC 3 | CC 4 | CC 5 | CC 6 | CC 7 | CC 8 | CC 9 |
|---|---|---|---|---|---|---|---|---|---|---|
| IF | s1 | I1 | I2 | I3 | I4 | I5 | | | | |
| ID | s2 | | I1 | I2 | I3 | I4 | I5 | | | |
| EX | s3 | | | I1 | I2 | I3 | I4 | I5 | | |
| MEM | s4 | | | | I1 | I2 | I3 | I4 | I5 | |
| WB | s5 | | | | | I1 | I2 | I3 | I4 | I5 |

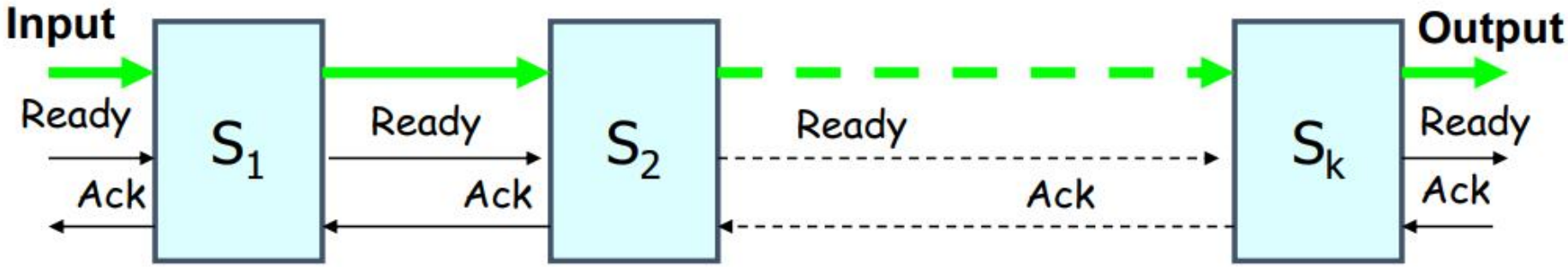
Time taken to execute 5 instructions in five stage pipelined architecture = 9 clock cycles.

## Linear Pipeline

• A linear pipeline processor is a cascade of Processing Stages which are linearly connected to perform fixed function over a stream of data flowing from one end to the other.
• Linear pipeline are static pipeline because they are used to perform fixed functions.

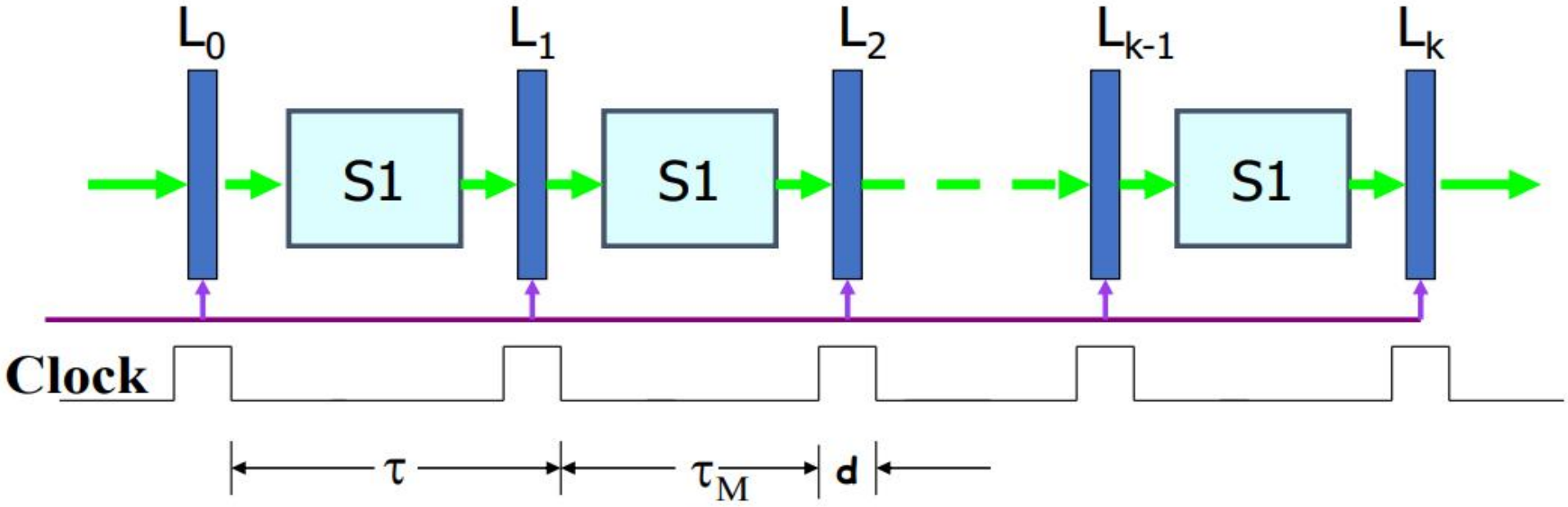
On the basis of the control of data flow along the pipeline. we model linear pipelines in two categories:
• **Synchronous Pipeline**
• • **Asynchronous Pipeline (Handshaking)**

# *Asynchronous Pipeline*




- Data flow between adjacent stages in an asynchronous pipeline is controlled by a handshaking protocol.
- Asynchronous pipelines are useful in designing communication channels in massage passing multicomputer
- Asynchronous pipelines may have variable throughput rate. Different amount of delay may be experienced in different stages.

# Synchronous Pipeline

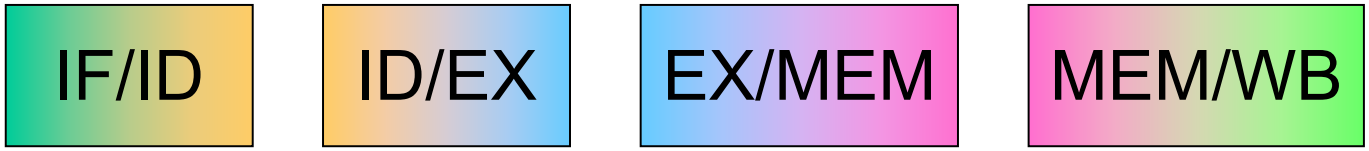



- Clocked latches are used to interface between stages. On the arrival of a clock pulse, all latches transfer data to the next stage simultaneously.
- The pipeline stages are combinational logic circuits. It is desired to have approximately equal delays in all stages.
- These delays determine the clock period and thus the speed of the pipeline.

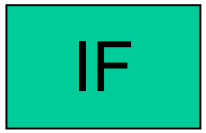
# Pipeline Registers

- Pipeline registers are essential part of pipelines:
  - There are 4 groups of pipeline registers in 5 stage pipeline.
- Each group saves output from one stage and passes it as input to the next stage:
  - IF/ID
  - ID/EX
  - EX/MEM
  - MEM/WB
- This way, each time "something is computed"...
  - Effective address, Immediate value, Register content, etc.
  - It is saved safely in the context of the instruction that needs it.

# Pipeline Registers



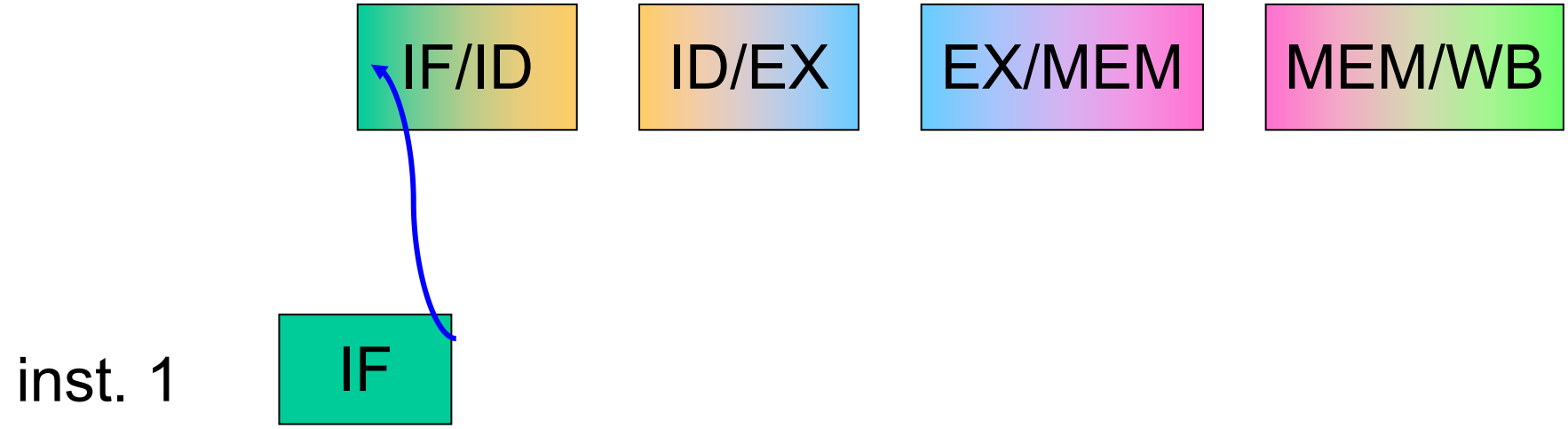

inst. 1



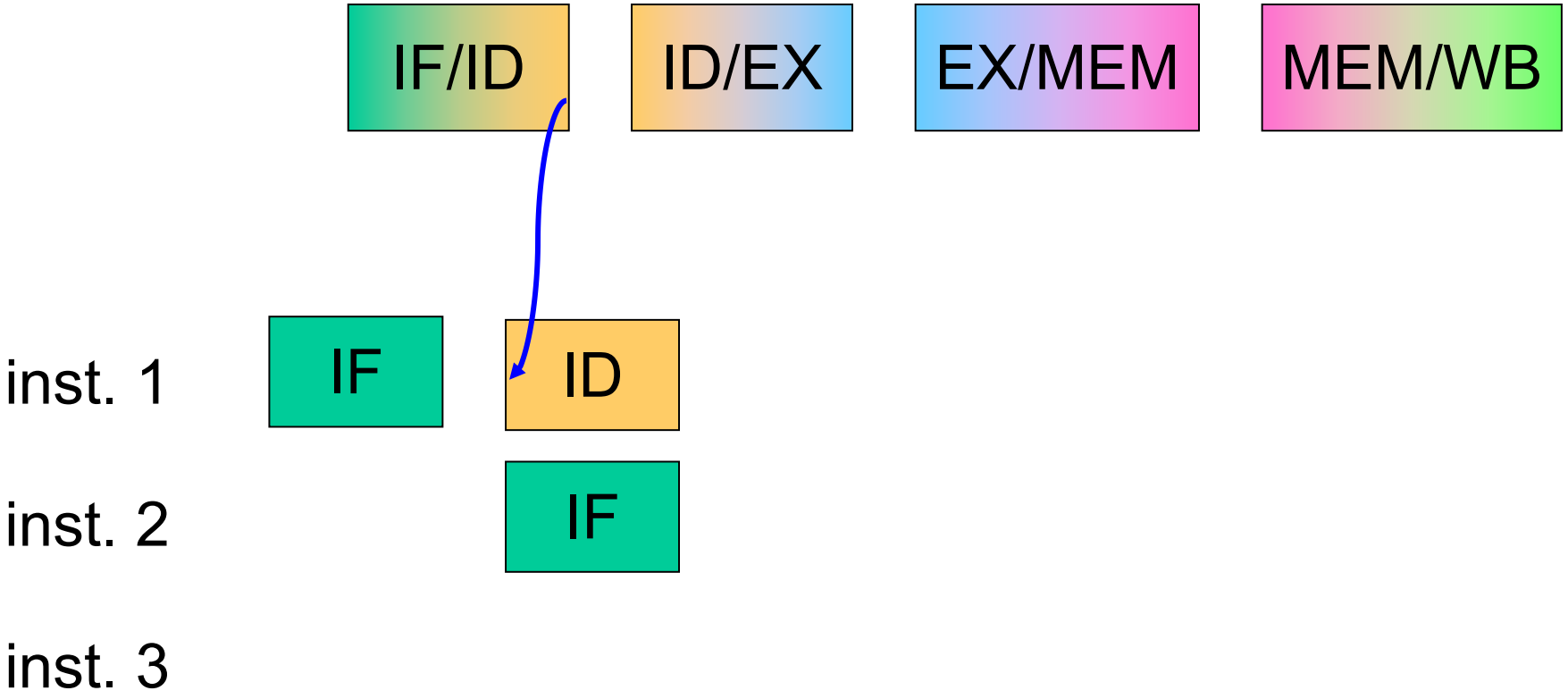

inst. 2

inst. 3

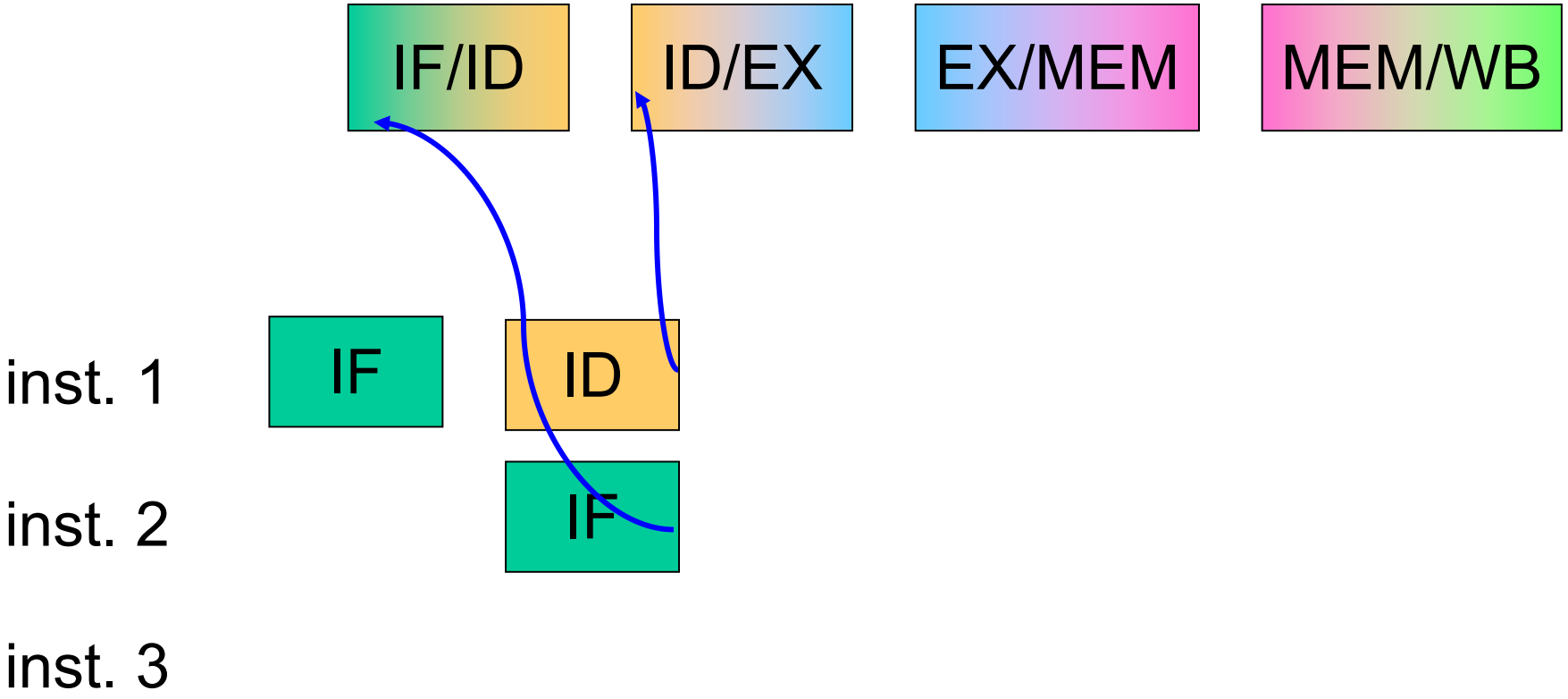
# Pipeline Registers



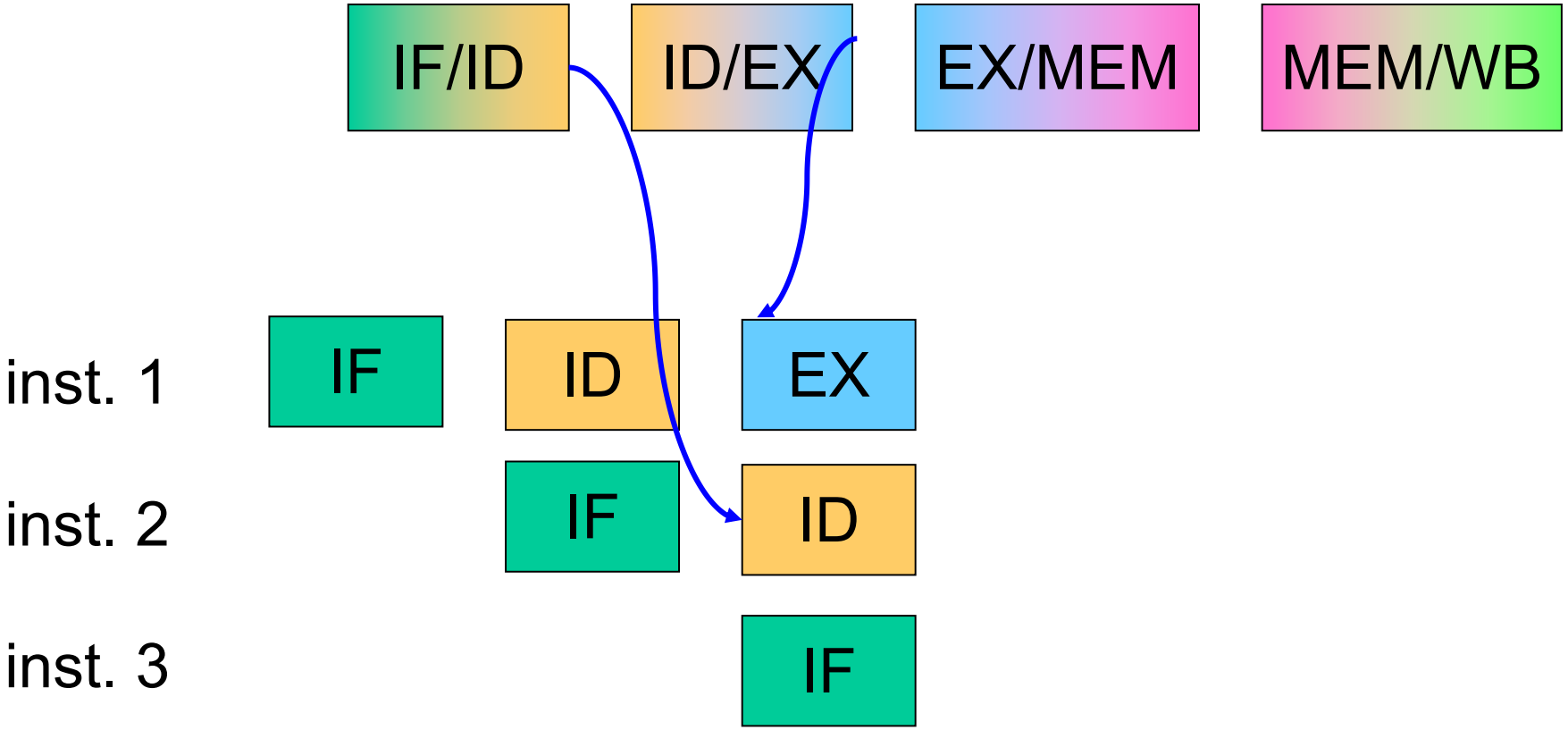
# Pipeline Registers



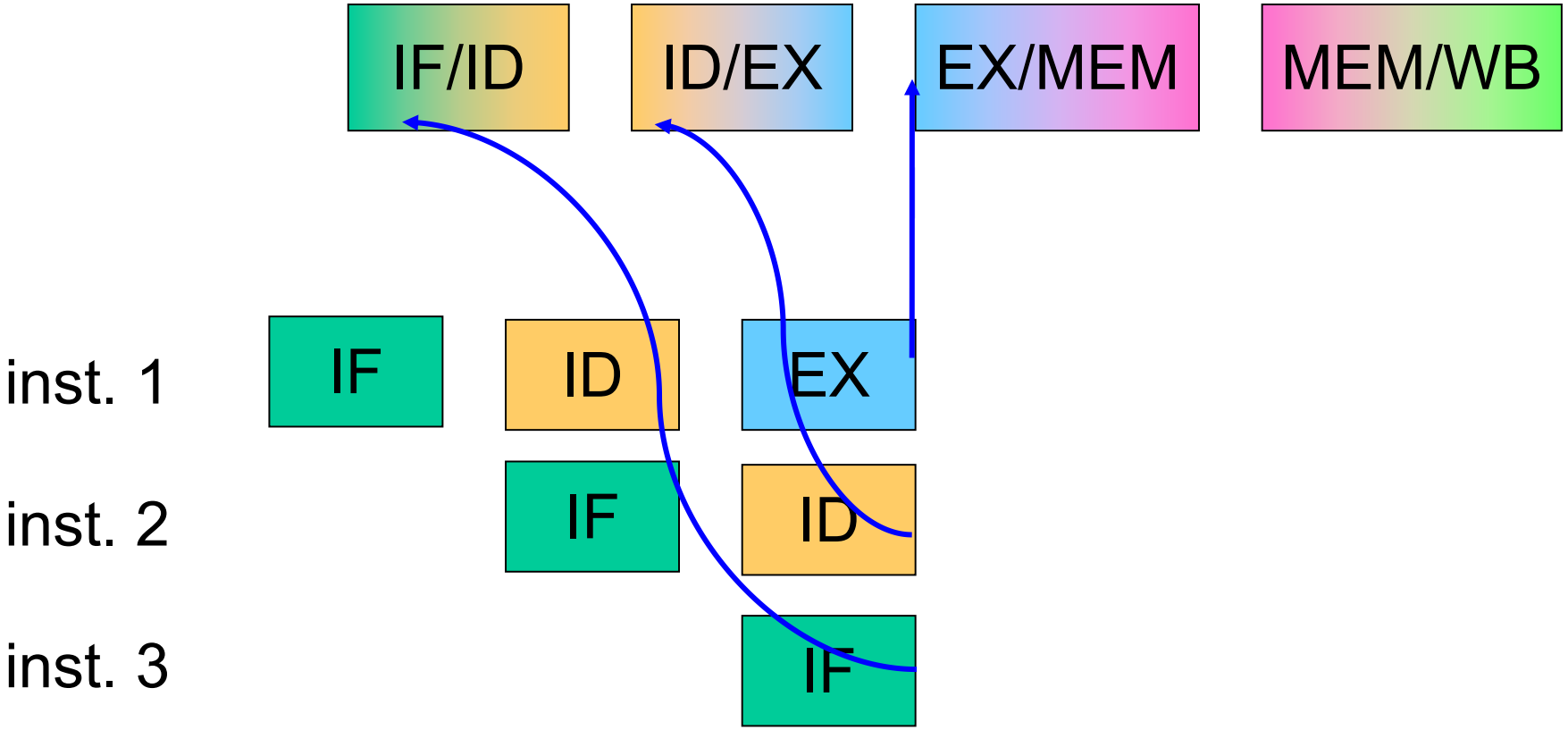
# Pipeline Registers



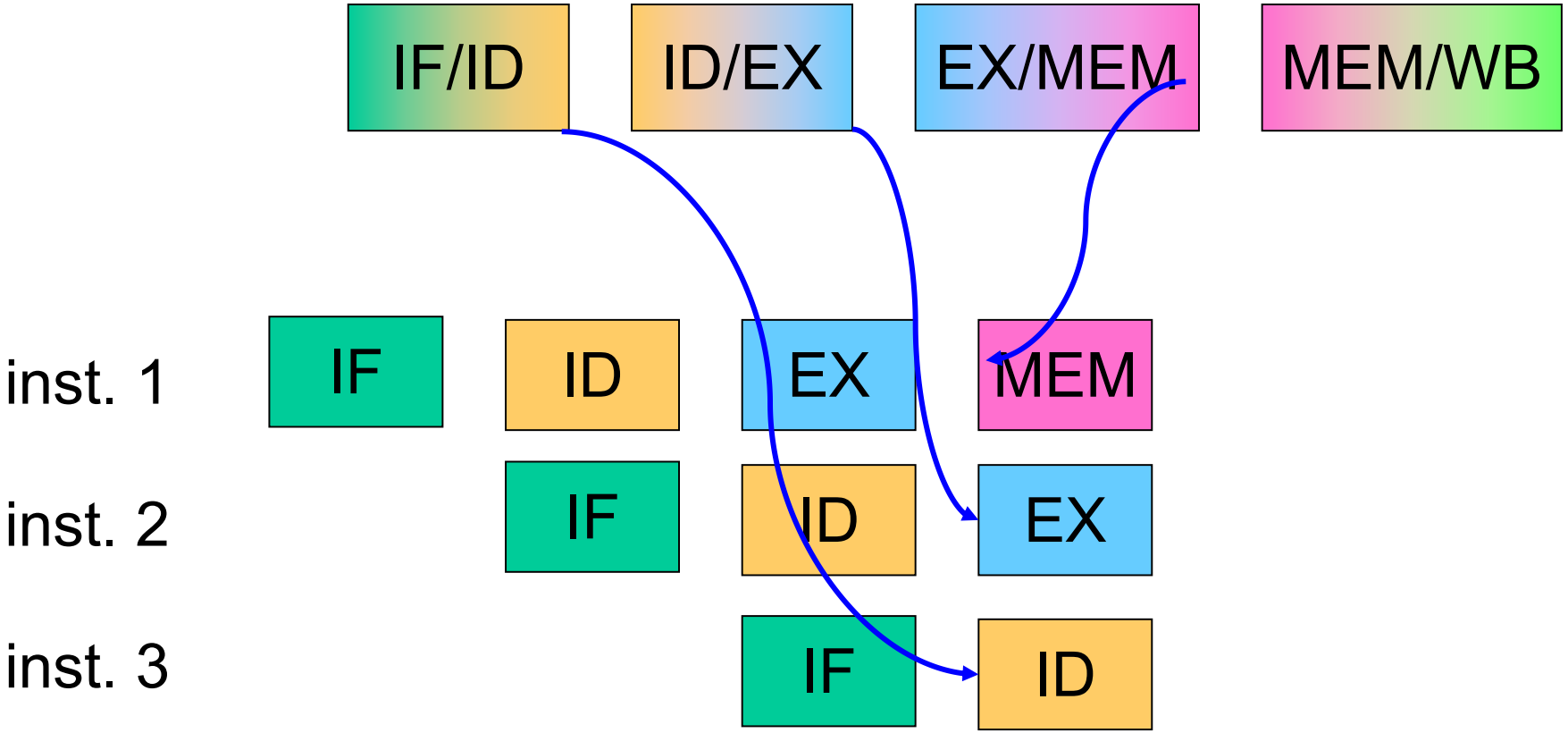
# Pipeline Registers



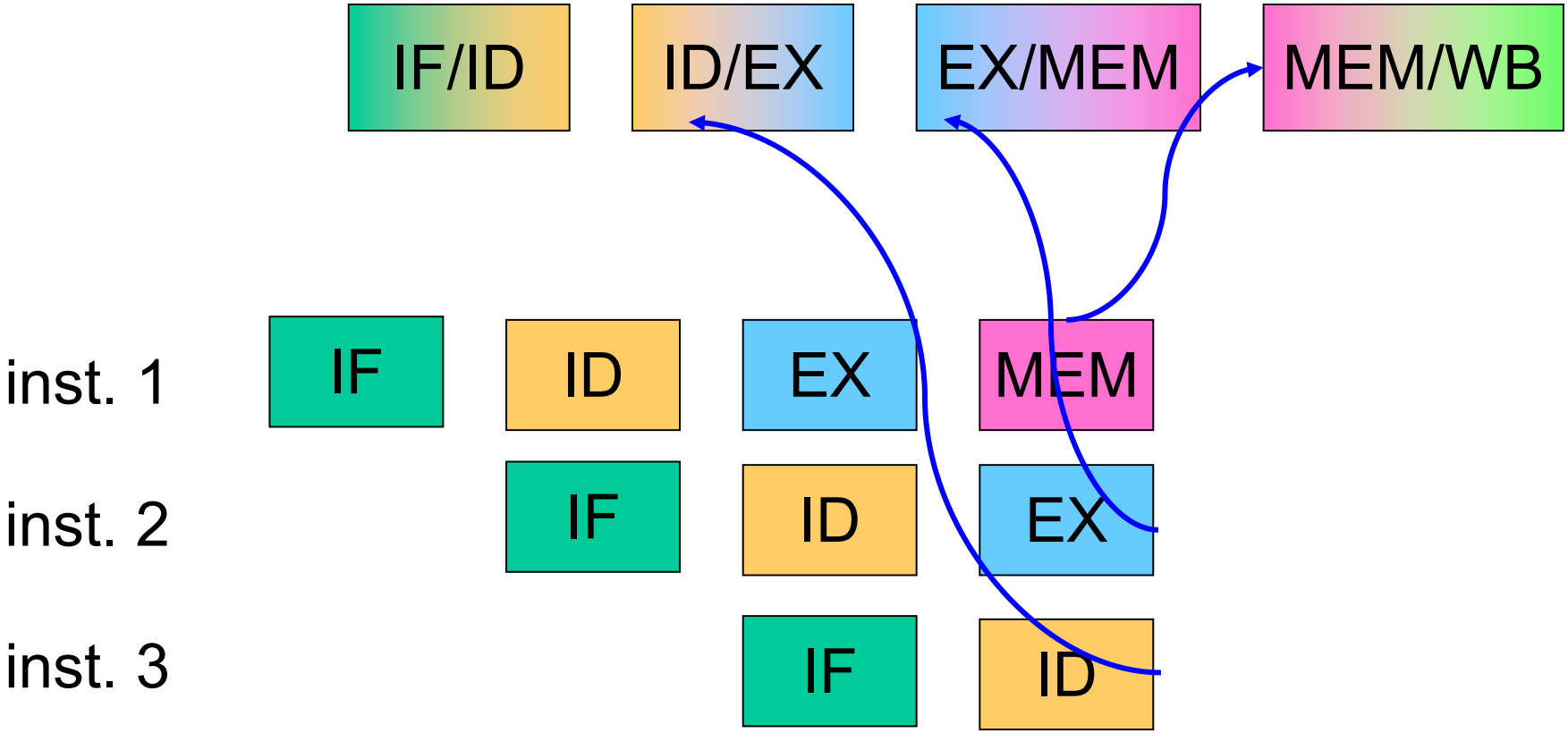
# Pipeline Registers



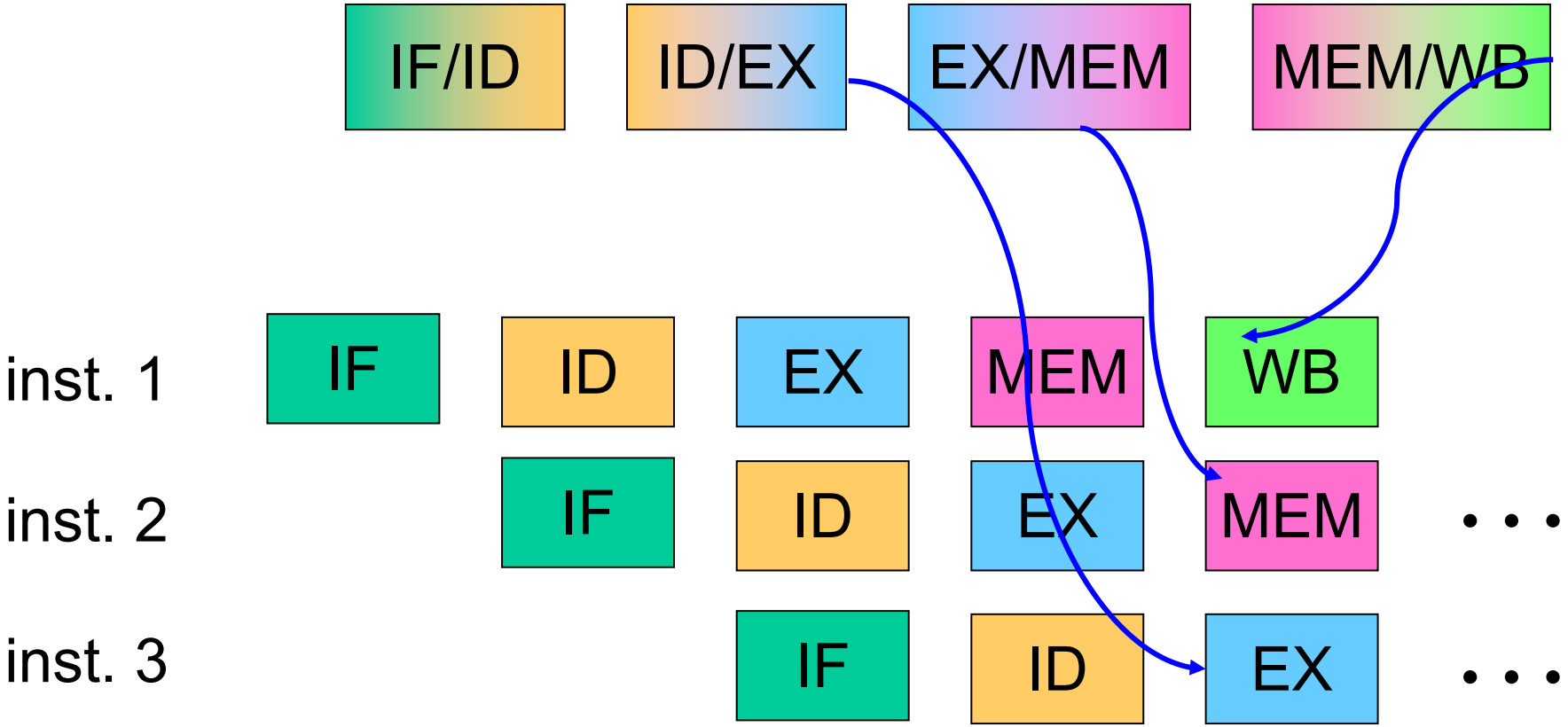
# Pipeline Registers



# Pipeline Registers

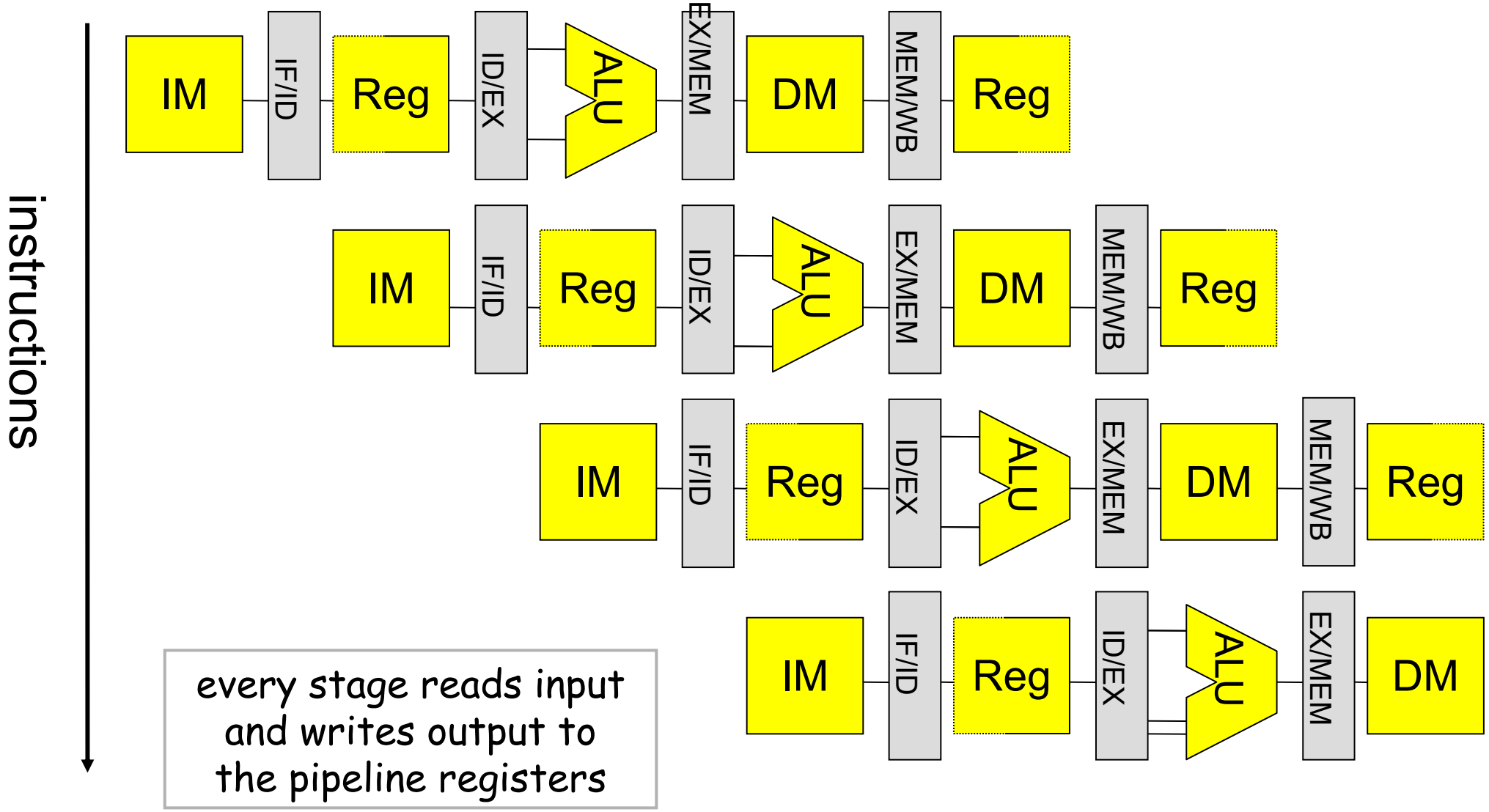


# Pipeline Registers




- Typically, we will not think too much about pipeline registers and one just assumes that values are passed "magically" down stages of the pipeline.

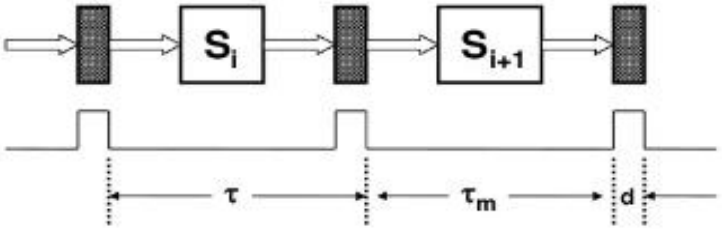
# Pipeline Register Depiction



# Basic Pipelining Terminologies

- Pipeline cycle (or Processor cycle):
  - The time required to move an instruction one step further in the pipeline.
  - Not to be confused with clock cycle.
- Synchronous pipeline:
  - Pipeline cycle is constant (clock-driven).
- Asynchronous pipeline:
  - Time for moving from stage to stage varies
  - Handshaking communication between stages

# A Few Pipeline Concepts




Pipeline cycle : $\tau$

Latch delay : $d$

$$\tau = \max\{\tau_m\} + d$$

Pipeline frequency : $f$

$$f = 1 / \tau$$

# Calculation of Important Parameters-

Consider-
A pipelined architecture consisting of k-stage pipeline
Total number of instructions to be executed = n

## Calculating pipeline Cycle Time-

### Case-01: All the stages offer same delay-

If all the stages offer same delay, then-
Cycle time = Delay offered by one stage including the delay due to its register

### Case-02: All the stages do not offer same delay-

If all the stages do not offer same delay, then-
Cycle time = Maximum delay offered by any stage including the delay due to its register

## Calculating Frequency Of Clock-

Frequency of the clock (f) = 1 / pipeline Cycle time

# Example

Suppose the time delays of the 5 stages are 60ns,50ns, 90ns, 80ns ,50ns respectively& the interface latch has a delay of d = 10ns.find the pipeline cycle time,non-pipeline cycle time,clock frequency of the pipeline and clock frequency of non pipeline:

- Hence the cycle time of this pipeline can be granted to be like :- $\tau$ = 90 + 10 =100ns
- If it is non-pipeline then = 60 + 50 + 90 + 80+50 =330ns
- Clock frequency of the pipeline
  - (f) = $1/(100*10^{-9})$ =10Mhz
- Clock frequency of the non pipeline
  - (f) = $1/(330*10^{-9})$ =3.003Mhz

## Example

A six stage pipeline processor has the stage delays as 150ns ,120ns,190ns ,130ns 140ns, and 120ns respectively. Pipeline Registers are used between the stages and have a delay of 5ns each. Calculate the cycle time for pipeline processor and non pipeline processor.

- solutions
- Hence the cycle time of this pipeline can be granted to be like :- = 190 + 5 =195ns
- If it is non-pipeline then = 150ns+ 120ns,190ns +130ns +140ns+120ns =850ns

# Why Pipelining RISC Processors is Easy

- Recall the main principles of RISC:
  - All operands are in registers
  - The only operations that affect memory are loads and stores.
  - Few instruction formats, fixed encoding (i.e., all instructions are the same size)
- Although pipelining could conceivably be implemented for any architecture,
  - It would be inefficient.
  - Pentium has characteristics of RISC and CISC --- CISC Instructions are internally converted to RISC-like instructions.

## Advantages of Pipelining

Instruction throughput increases.
Increase in the number of pipeline stages increases the number of instructions executed simultaneously.
Faster ALU can be designed when pipelining is used.
Pipelined CPU's works at higher clock frequencies than the RAM.
Pipelining increases the overall performance of the CPU.

## Disadvantages of Pipelining

Designing of the pipelined processor is complex.
Instruction latency increases in pipelined processors.
The throughput of a pipelined processor is difficult to predict.
The longer the pipeline, worse the problem of hazard for branch instructions.