



SNS COLLEGE OF TECHNOLOGY

(An Autonomous Institution)



COIMBATORE-35

**Accredited by NBA-AICTE and Accredited by NAAC – UGC with A++ Grade
Approved by AICTE, New Delhi & Affiliated to Anna University, Chennai**

DEPARTMENT OF ELECTRICAL AND ELECTRONICS ENGINEERING

COURSE NAME: 19EEB303 / Microcontroller and its Applications

III YEAR / VI SEMESTER

Unit III – IOT - ARCHITECTURE REFERENCE MODEL

Topic: Pipeline



Pipeline

Pipelining in ARM processors can be described as a situation in which many stages of instructions such as the fetch, the decode, the execution, among others, are enhanced or overlapped in order to improve the performance of the CPU. Each stage of the pipeline at work on a different instruction at the same time thereby enhancing throughput.

- A **Pipelining** is the mechanism used by **RISC(Reduced instruction set computer)** processors to execute instructions,
- by speeding up the execution by **fetching** the instruction, while other instructions are being **decoded and executed** simultaneously.
- Which in turn allows the memory system and processor to work continuously.
- The pipeline design for each ARM family is different.



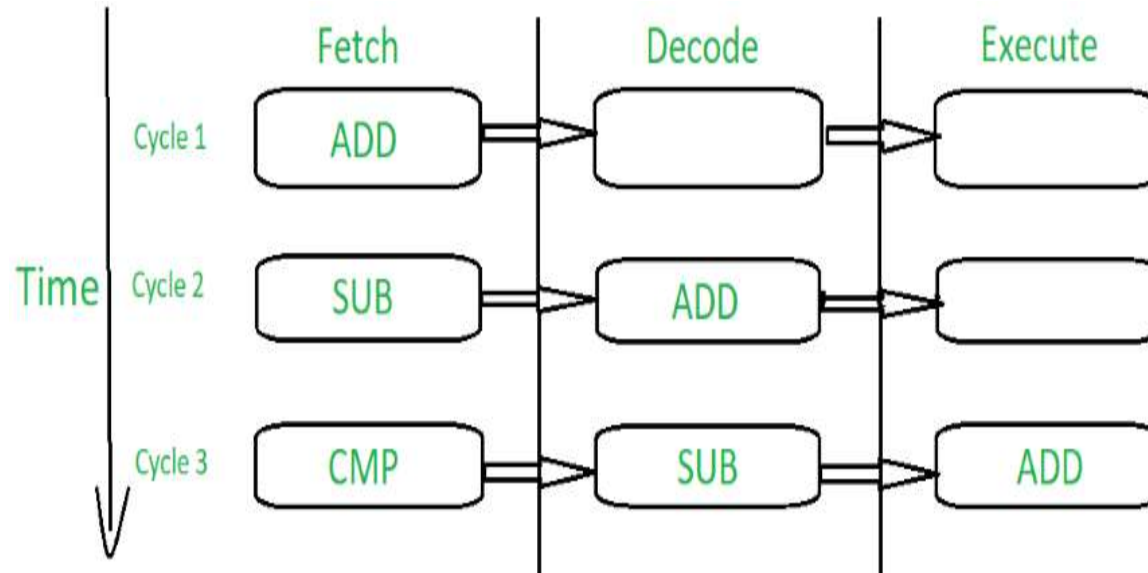
Pipeline

Pipelining is a design technique or a process which plays an important role in increasing the efficiency of data processing in the processor of a computer and [microcontroller](#). By keeping the processor in a continuous process of fetching, decoding and executing called (F&E cycle).

ARM devices need pipelining because of [RISC](#) as it emphasizes on compiler complexity. Each stage is equivalent to 1 cycle, that is $n \text{ stages} = n \text{ cycles}$



Pipeline





Pipeline

- Fetch loads an instruction from memory.
- Decode identifies the instruction to be executed.
- Execute processes the instruction and writes the result back to the register.
- By over lapping the above stages of execution of different instructions, the speed of execution is increased.
- The pipelining allows the core to execute an instruction every cycle, which results in increased throughput.



Pipeline

ARM Pipeline Characteristics

- The ARM pipeline doesn't process an instruction until it passes completely through the execution stage.
- In the execution stage, the PC always points to the instruction address + 8 bytes.
- When the processor is in thumb state, PC always points to the instruction address + 4 bytes.
- While executing branch instructions or branching by direct modification of PC causes the ARM core to **flush** its pipeline.
- As instruction in the execution stage will complete its execution even though an interrupt has been raised.



Pipeline

ARM 7

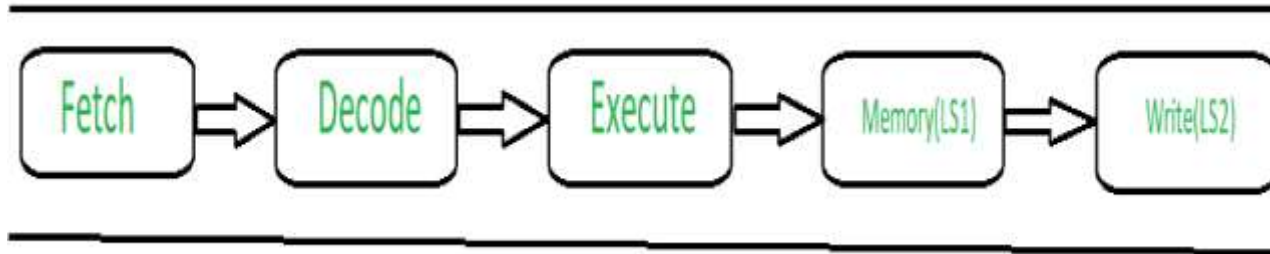
- It has 3 stage pipelining as shown in the figure.
- It can complete it's process in 3 cycles.
- It has the basic F&E cycle leading to optimum throughput.
- This is why the [ARM 7](#) has the lowest throughput as compared to that of it's other family members.
- It processes 32bit data.

ARM 9

- Pipelining in ARM 9 is similar to ARM 7 but with 5 stages.
- It takes 5 cycles to complete the process.



Pipeline



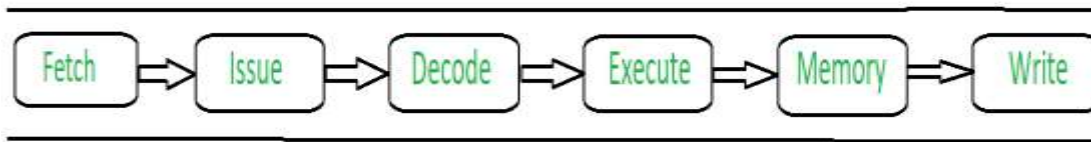
- Fetch- It will fetch instructions from memory.
- Decode- It decodes the instructions that were fetched in the first cycle.
- ALU – It executes the instruction that has been decoded in the previous stage.
- LS1(Memory) Loads/Stores the data specified by load or store instructions.
- LS2(Write) Extracts (zero or sign) extends the data loaded by byte or half word load instruction.
- Because of an increase in stages and efficiency, the throughput is 10%-13% higher than ARM 7.
- Core frequency of ARM 9 is slightly higher than that of ARM 7.



Pipeline

ARM 10

- It is a six stage pipeline. Which in turn takes 6 cycles to complete the process.
- Same as that of ARM 9 but with an **issue** stage which checks whether the instruction is ready to get decoded in the current stage or not.
- It nearly doubles the throughput than that of ARM 7.
- The core frequency is higher than that of ARM 9.



6 stage pipelining

The stages of pipelining may increase or decrease on the basis of the instruction sets processed per cycle (In maximum situations, stages tend to increase to increase efficiency).



Pipeline

3-Stage Pipelining

Pipelining is a form of instruction counterpart or execution in stages and a 3-stage pipeline is a basic version of the same. It splits the instruction cycle into three stages:

It splits the instruction cycle into three stages:

- **Fetch:** It is to note that the instruction is retrieved from the memory.
- **Decode:** This instruction is fetched and then the instruction is decoded where it is understood on which operation is required to be performed.
- **Execute:** The instruction, in turn, gets decoded and the resultant calculated value is stored.

Advantages of 3-Stage Pipelining

- **Reduced Complexity:** This is so because when one designs for fewer stages are unlike to possess as complex a design as one that has many stages.
- **Lower Power Consumption:** Smaller numbers would also seem to be associated with smaller power demands read smaller power stages.
- **Reduced Latency:** Instructions can move from fetch to execution in few steps so that they are executed quickly.



Pipeline

5 Stage Pipelining

A 5- stage pipeline is an enhanced form of pipelining used in today's [CPU](#) or computer processors. It breaks the instruction cycle into five stages:

It breaks the instruction cycle into five stages:

- **Fetch:** Thus the instruction is fetched from memory.
- **Decode:** Information is fetched using cache control instructions; the fetched instruction is in turn decoded.
- **Execute:** As instructed the operation described in the instruction is performed.
- **Memory Access:** Read any registers needed by the instruction.
- **Write-back:** To the register the result of the operation is then written back.

Advantages of 5-Stage Pipelining

- **Increased Instruction Throughput:** For more instruction, they work at the same time hence enhancing performance.
- **Better Resource Utilization:** It is used on the components of the CPU in a manner of less wastage.
- **Scalability:** It has become a regular pipeline structure incorporated in advanced processors to warrant scalability of performance.



Pipeline

Advantages of Pipelining in ARM

- **Increased Efficiency:** This way the ARM [processors](#) can handle multiple instructions in parallel and thus have a higher throughput.
- **Improved Power Efficiency:** ARM has a streamlined pipeline design which again consumes little of power thus making it very efficient in mobiles and other embedded systems.
- **Reduced Latency:** As mentioned above, pipelining helps in saving the time it takes to execute instructions, thus making [ARM](#) architecture to be fast and responsive.



Pipeline

Disadvantages of Pipelining in ARM

- **Pipeline Hazards:** These are the conditions that result in interruption of the pipelines leading to delay in the execution of instructions.
- **Increased Complexity:** Mention that with addition of the pipelines, the total design of this processor escalates as well.
- **Stalling:** Data dependences may require instructions to wait and hence result in pipeline stalls.