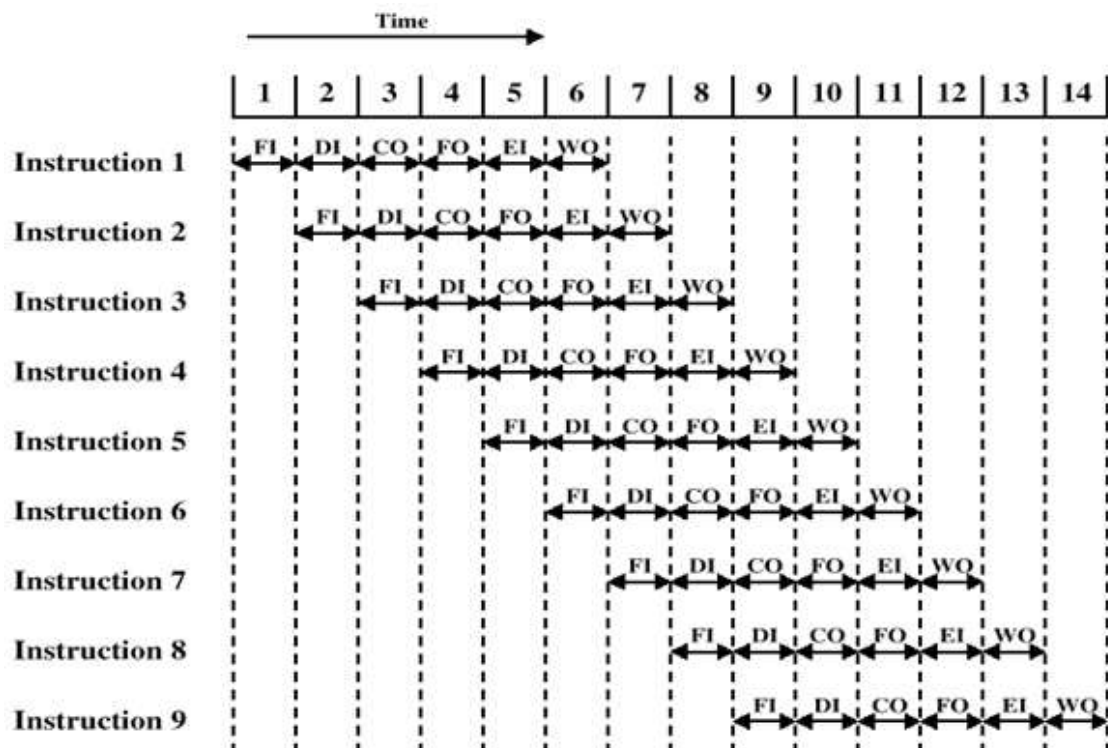


Pipeline

Pipeline, no contexto da computação, é um processo usado em processadores para executar instruções de forma mais eficiente, dividindo-as em etapas como em uma linha de montagem. Enquanto uma instrução está sendo finalizada, outra já pode começar, aumentando o desempenho do sistema.

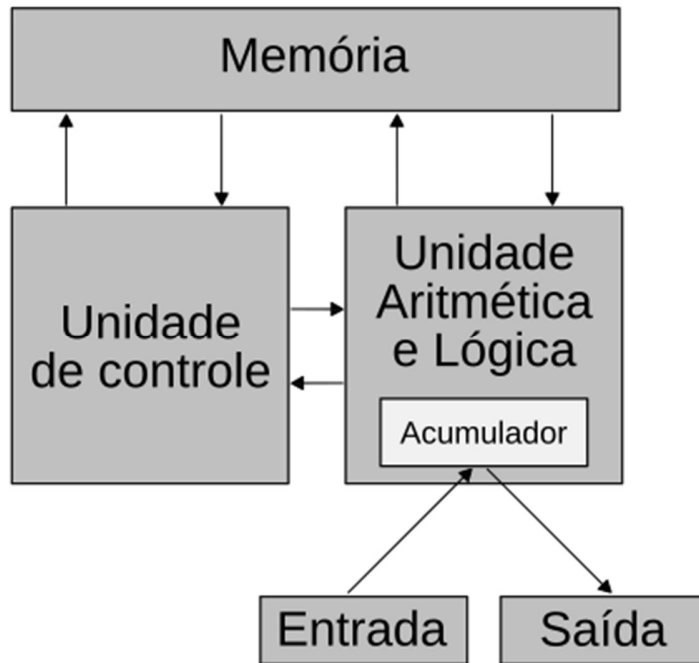
Um exemplo comum é a divisão em cinco etapas: buscar a instrução (Fetch), decodificar (Decode), executar (Execute), acessar a memória (Memory) e escrever o resultado (Write-back). Cada etapa trabalha simultaneamente em diferentes instruções, o que caracteriza o pipeline.



Como um Programa é Executado no Computador?

Quando um programa é executado, o sistema operacional primeiro o carrega da memória secundária (como HD ou SSD) para a memória RAM. Em seguida, o processador interpreta e executa suas instruções, utilizando recursos como registradores, a unidade lógica e aritmética (ALU) e acesso à memória.

Durante esse processo, o processador busca os dados necessários na RAM, executa os cálculos e armazena os resultados. Esses resultados podem ser exibidos na tela, enviados para dispositivos ou armazenados novamente em disco.



Pipeline na Prática

O uso de pipeline traz benefícios como maior velocidade na execução e melhor aproveitamento da CPU. No entanto, também apresenta desafios. Entre eles estão os hazards, que ocorrem quando instruções dependem de resultados anteriores, e os saltos condicionais, que interrompem o fluxo contínuo do pipeline.

Processadores modernos, como os da Intel e AMD, utilizam pipelines com muitos estágios. Além disso, usam técnicas como execução fora de ordem e especulação de ramificação para manter o pipeline funcionando de forma contínua e eficiente.