

- 1) O que difere as arquiteturas de Von Neumann e de Harvard é que : Enquanto a arquitetura de Von Neumann as informações são processadas uma por vez, visto que os dados percorrem o mesmo barramento, o que torna o processo lento, a de Harvard possui duas memórias diferentes e independentes, em termos de barramento e ligação ao processador, possibilitando processar as informações de ambas as memórias(dados e instruções) ao mesmo tempo, tornando o processo mais rápido que o da arquitetura de Von Neumann.
- 2) Além de ter sido o primeiro minicomputador, permitindo seu uso em aplicações embarcadas e em linhas de produção, o PDP-8 trouxe consigo uma estrutura de interconexão em forma de barramento(contendo um barramento e um controlador para o mesmo).
- 3) Aumento da capacidade de processamento de informação, aumento da capacidade de armazenamento, aumento na quantidade de transistores presentes nos circuitos integrados e a maior capacidade de processamento(melhor desempenho).
- 4) Ocorreu uma evolução da capacidade de processamento do processador e da capacidade de armazenamento da memória, porém, as memórias não evoluíram em termos de velocidade de armazenamento e leitura de dados. Ou seja, temos um processador muito rápido e uma memória que é capaz de armazenar muita informação, porém, lenta. Isso seria o desbalanceamento entre desempenho de memória e de processadores.
- 5) Pipeline é uma técnica de implementação de processadores que permite executar as fases da execução sem precisar esperar o fim de um ciclo. Como as tarefas de busca, decodificação e execução acontecem em 3 hardwares diferentes, ao invés de esperar o fim das 3 fases para executar o próximo ciclo, a técnica de pipeline faz com que assim que for finalizada uma determinada etapa de um ciclo, a mesma etapa do próximo ciclo é iniciada, acelerando o processamento e reduzindo o tempo para execução das fases.

- 6) As funções básicas que um computador podem desempenhar são: Processamento de dados, armazenamento de dados, transferência de dados e controle (Responsável por fazer que as outras 3 funções sejam implementadas de forma correta).