

## Unidade 13: Paralelismo: SMP e Processamento Vetorial

Prof. Daniel Caetano

**Objetivo:** Apresentar os conceitos fundamentais da arquitetura SMP e alguns detalhes do processamento Vetorial.

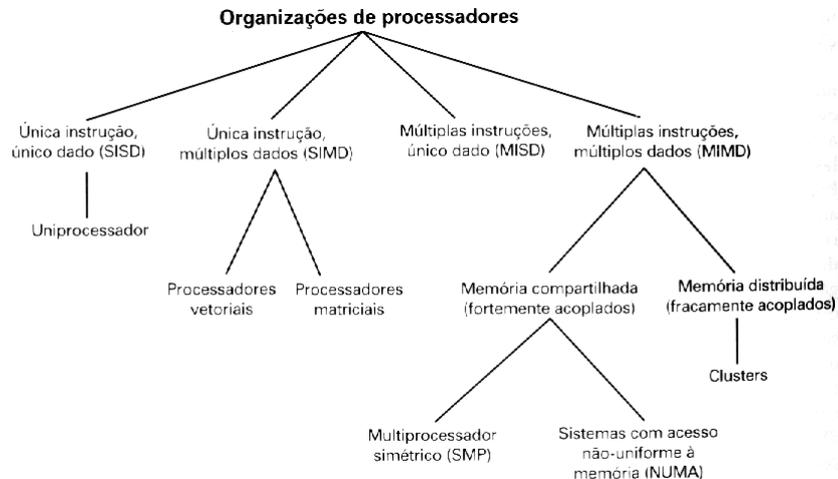
### Bibliografia:

- STALLINGS, W. **Arquitetura e organização de computadores**. 5ed. São Paulo: Ed. Pearson Prentice Hall, 2003.

- MURDOCCA, M. J; HEURING, V.P. **Introdução à arquitetura de computadores**. S.I.: Ed. Campus, 2000.

## INTRODUÇÃO

Como visto anteriormente, há diversas formas de se realizar processamentos paralelos, como indicado na figura a seguir. Nesta aula serão apresentados alguns detalhes sobre a arquitetura SMP e mais algumas informações sobre os tipos de cálculos realizados pelos processadores vetoriais.



## 1. ARQUITETURA SMP (SYMMETRIC MULTI PROCESSING)

Até bem pouco tempo atrás, a arquitetura mais comum para os computadores pessoais tinha um único processador. Com a queda dos custos e a sempre crescente demanda, foi introduzida nos computadores pessoais a arquitetura SMP, que pode ser definida com as seguintes características:

1. Existência de dois ou mais processadores com capacidades comparáveis.
2. Todos os processadores compartilham a memória um barramento ou outro esquema de conexão qualquer, de forma que o tempo de acesso à memória é praticamente o mesmo para cada processador.
3. Os dispositivos de e/s são compartilhados, podendo ser acessados por canais comuns ou por canais distintos que levem aos mesmos dispositivos.
4. Todos os processadores podem executar as mesmas funções (simetria!)
5. O sistema é controlado por um sistema operacional integrado, que permite a interação entre os processadores e seus programas - em nível de tarefas, arquivos e dados.

Ou seja, como decorrência do item 5, é o sistema operacional quem deve escalonar os processos e threads entre os diversos processadores.

As potenciais vantagens dos sistemas SMP são:

- \* **Desempenho:** desempenho nitidamente maior que o de máquinas monoprocessadas.
- \* **Disponibilidade:** como todos os processadores são equivalentes, a falência de 1 dos processadores não impede a máquina de continuar trabalhando, ainda que mais lentamente.
- \* **Crescimento Incremental:** é possível aumentar o desempenho do sistema adicionando processadores.
- \* **Escalabilidade:** os fabricantes podem oferecer uma larga faixa de produtos, com custos e desempenhos diferentes.

Mais recentemente, é possível até mesmo falar em maior economia de energia, visto que os equipamentos mais modernos são capazes de **desligar** as CPUs inativas, bem como **reduzir sua velocidade de processamento** quando a capacidade não é exigida.

## 2. ORGANIZAÇÃO DA ARQUITETURA SMP

O principal desafio da arquitetura SMP é reduzir os conflitos dos diversos processadores no seu acesso à memória. Uma vez que cada um dos processadores é independente e todos eles precisam acessar os dados, a organização para coordenar os acessos à memória é determinante para um bom desempenho.

Existem diversas estruturas para implementar essa coordenação na arquitetura SMP. Elas podem ser classificadas da seguinte forma: barramento comum, memória com múltiplas portas ou unidade de controle central.

### 2.1. Unidade de Controle Central

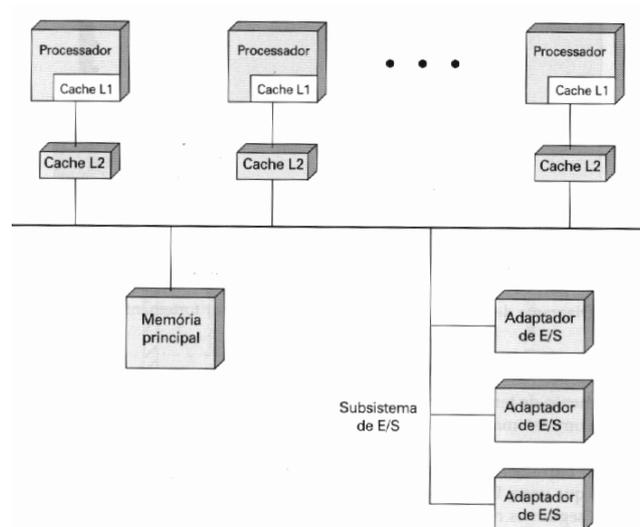
Uma vez que a grande dificuldade da arquitetura SMP é coordenar o acesso à memória pelas diferentes CPUs, uma das primeiras estratégias foi centralizar a Unidade de Controle: assim, existia uma unidade de controle única, mas várias unidades de processamento (ULA/FPU).

Uma das vantagens desta estrutura é uma coordenação completa dos acessos, já que a única unidade de controle tem capacidade de organizar todo o processo. Por outro lado, a Unidade de Controle, por si só, já é um circuito demasiadamente complexo e as unidades de controle centralizadas para vários processadores eram ainda mais complexas, em especial se fosse desejado um número dinâmico de unidades de processamento.

Este era o tipo de organização mais usado pela IBM na década de 1960 e 1970, mas, devido a alta complexidade, hoje ele praticamente caiu em desuso.

## 2.2. Barramento Comum (ou Tempo Compartilhado)

Na estrutura de barramento comum, também conhecida como "de tempo compartilhado", existem várias CPUs completas, todas elas conectadas ao mesmo barramento. As interconexões são praticamente as mesmas de um sistema monoprocessoado, mas o gerenciamento do barramento é um pouco mais complexo (apesar de ele se manter essencialmente o mesmo), pois passa a ser necessário **distinguir os diferentes módulos** (e processadores), mantendo o **controle de arbitragem** e o **compartilhamento de tempo**.



Há diversas vantagens nessa estrutura:

- \* **Simplicidade:** é a forma mais simples de organizar um sistema multiprocessador, fica tudo muito parecido com um sistema monoprocessoador.

- \* **Flexibilidade:** geralmente é fácil expandir o número de processadores no barramento.

- \* **Confiabilidade:** Como o barramento é passivo, a falha em um dispositivo/cpu não precisa, necessariamente, comprometer todo o funcionamento do sistema.

Há desvantagens, entretanto: há um grande gargalo no acesso a memória, já que todas as CPUs precisam acessar pelo mesmo barramento. Esse gargalo é, normalmente, mitigado

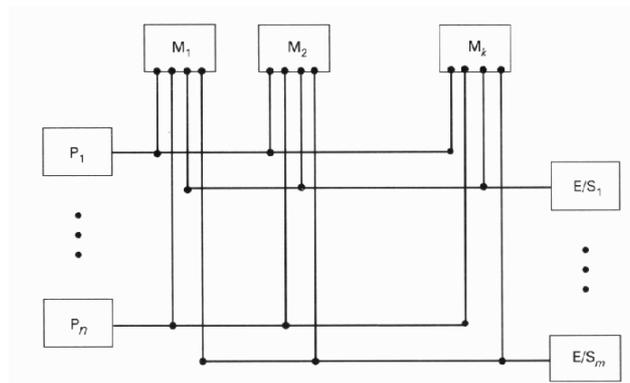
por uma memória Cache para cada processador. Infelizmente, isso introduz outras complexidades.

Como vários processadores podem estar trabalhando nas mesmas áreas da memória, se um processador escreve na memória, ele invalida os dados do cache de outro processador. Assim, é preciso haver também uma comunicação direta entre os processadores, para que os ajustes de cache sejam feitos, quando necessário.

Esses ajustes são feitos seguindo um protocolo de **coerência de cache**. Existem diversos mecanismos conhecidos para isso, mas foge ao escopo do curso detalhá-los.

### 2.3. Memória com Múltiplas Portas

Esse tipo de estruturação fornece um barramento de acesso à memória para cada componente do sistema, permitindo que, em tese, todos os dispositivos possam solicitar acesso a uma mesma memória.



A grande vantagem é a redução dos conflitos de arbitragem de uso de barramento mas, por outro lado, exige o uso de memória com múltiplas portas (multiport memory) que são, usualmente, mais caras. A razão para que elas sejam mais caras é que a própria memória é responsável por resolver os conflitos que surgem.

Finalmente, esse tipo de estrutura dificulta o controle da coerência do cache, sendo necessário um mecanismo direto de controle por parte das CPUs, pois não existe uma forma conveniente dos outros processadores detectarem esses problemas pelo próprio barramento.

## 3. PROCESSAMENTO VETORIAL

Na aula passada foi comentada a existência de processadores vetoriais e matriciais. Esses processadores são usados em sistemas chamados **supercomputadores**.

Foge ao escopo do curso tratar detalhes sobre o funcionamento interno destes processadores, mas cabem alguns esclarecimentos sobre sua utilidade.

Há uma série de problemas, nas áreas de aerodinâmica, sismologia, meteorologia, física atômica - dentre outras - que exigem uma alta precisão numérica em cálculos de ponto flutuante, executando programas que aplicam repetidamente complexos conjuntos de operações em grandes vetores de números.

A maioria destes problemas pertence à categoria dos problemas de simulação de campos contínuos, isto é, um problema em que a situação física pode ser descrita por uma superfície ou uma região em 3 dimensões. Exemplos são a evolução climática na atmosfera terrestre, o campo de temperaturas e velocidades no jato de propulsão de um foguete ou mesmo as linhas de corrente de deslocamento de ar ao redor de uma asa de avião.

Um simples cálculo de soma destes problemas é expresso na forma de matrizes:

$$\begin{bmatrix} 1,5 \\ 7,1 \\ 6,9 \\ 100,5 \\ 0 \\ 59,7 \end{bmatrix} + \begin{bmatrix} 2,0 \\ 39,7 \\ 1000,003 \\ 11 \\ 21,1 \\ 19,7 \end{bmatrix} = \begin{bmatrix} 3,5 \\ 46,8 \\ 1006,903 \\ 111,5 \\ 21,1 \\ 79,4 \end{bmatrix}$$

$A + B = C$

É fácil imaginar porque o paralelismo de instruções simples sobre múltiplos dados (SIMD) ajuda neste tipo de cálculo. Quando o procedimento envolve multiplicação de matrizes, alguns truques são necessários, mas o paralelismo também ajuda.

Computadores capazes de resolver estes problemas com multiprocessamento vetorial/matricial são poucos no mundo, cada unidade custando alguns "poucos" milhões de dólares. Basicamente eles só existem nos grandes centros de pesquisa, onde são realmente necessários.

Como, mesmo com estes computadores, este tipo de problema requer, para resolver um problema simples, da ordem de  $10^{13}$  operações aritméticas, tomando cerca de 2 a 3 dias para resolver alguns problemas não tão complexos, esta é, ainda, uma área em muito estudo e que, certamente, terá avanços num futuro próximo.

#### **4. BIBLIOGRAFIA**

STALLINGS, W. **Arquitetura e organização de computadores**. 5ed. São Paulo: Ed. Pearson Prentice Hall, 2003.

MURDOCCA, M. J; HEURING, V.P. **Introdução à Arquitetura de Computadores**. S.I.: Ed. Campus, 2000.