



# Arquiteturas paralelas – Parte 1

---

## Processamento Paralelo

Prof. Oberlan Romão

Departamento de Computação e Eletrônica – DCEL

Centro Universitário Norte do Espírito Santo – CEUNES

Universidade Federal do Espírito Santo

# Uma breve história da computação paralela

- Foco Inicial (a partir dos anos 70): “Supercomputadores” voltados para Computação Científica



**C.mmp at CMU (1971)**  
**16 PDP-11 processors**



**Cray XMP (circa 1984)**  
**4 vector processors**



**Thinking Machines CM-2 (circa 1987)**  
**65,536 1-bit processors +**  
**2048 floating-point co-processors**



**SGI UV 1000cc-NUMA (today)**  
**4096 processor cores**

← **Blacklight at the Pittsburgh  
Supercomputer Center**

# Uma breve história da computação paralela

- Foco Inicial (a partir dos anos 70): “Supercomputadores” voltados para Computação Científica
- Outra aplicação de tendência (a partir do início dos anos 90): bancos de dados



**Sun Enterprise 10000 (circa 1997)**  
**16 UltraSPARC-II processors**



**Oracle Supercluster M6-32 (today)**  
**32 SPARC M2 processors**

# Uma breve história de desempenho do processador

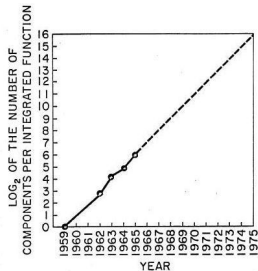
- Aumento do tamanho da palavra do processador
  - 4 bit  $\rightarrow$  8 bit  $\rightarrow$  16 bit  $\rightarrow$  32 bit  $\rightarrow$  64 bit
- Pipeline mais eficiente
- Exploração do paralelismo de nível de instrução (ILP)
- Taxas de clock mais rápidas
  - 10 MHz  $\rightarrow$  200 MHz  $\rightarrow$  3 GHz
- Durante os anos 80 e 90: grandes ganhos de desempenho exponencial
  - e depois...

# Uma breve história da computação paralela

- Foco Inicial (a partir dos anos 70): “Supercomputadores” voltados para Computação Científica
- Outra aplicação de tendência (a partir do início dos anos 90): bancos de dados
- Ponto de inflexão em 2004: Intel atinge a barreira de Densidade de Energia

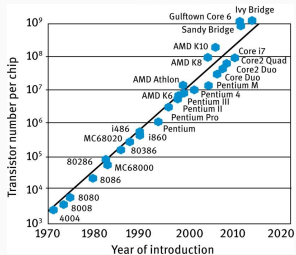
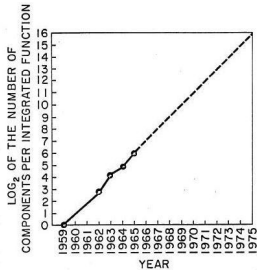
# Lei de Moore

---



**Gordon Moore (co-fundador da Intel) previu em 1965 que o número de transistores de um chips dobraria aproximadamente a cada 18 meses.**

# Lei de Moore

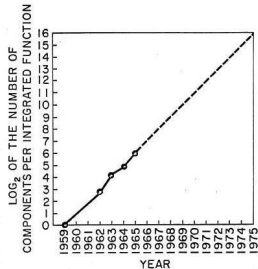


**2x transistores/chip a cada 1,5 ano**

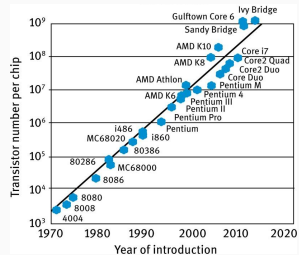
**Gordon Moore (co-fundador da Intel) previu em 1965 que o número de transistores de um chips dobraria aproximadamente a cada 18 meses.**



# Lei de Moore



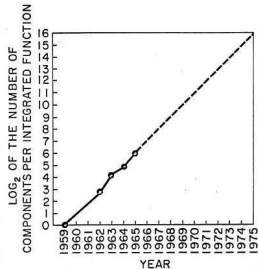
Gordon Moore (co-fundador da Intel) previu em 1965 que o número de transistores de um chips dobraria aproximadamente a cada 18 meses.



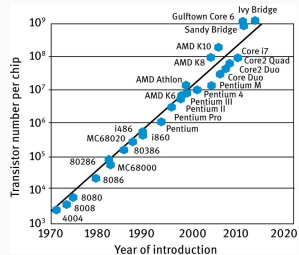
2x transistores/chip a cada 1,5 ano

Chamada de **Lei de Moore**

# Lei de Moore



Gordon Moore (co-fundador da Intel) previu em 1965 que o número de transistores de um chips dobraria aproximadamente a cada 18 meses.

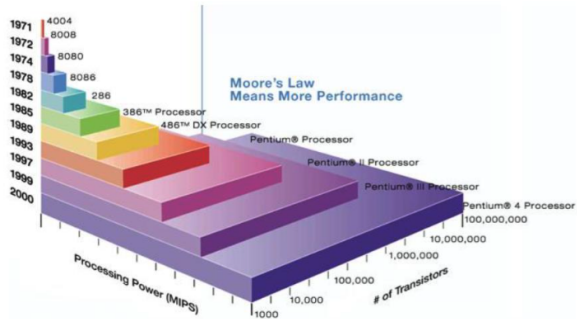


2x transistores/chip a cada 1,5 ano

Chamada de **Lei de Moore**

Os processadores tornaram-se menores, mais densos e mais poderosos.

# Lei de Moore



**1950s**

Silicon Transistor



1 Transistor

**1960s**

TTL Quad Gate



16 Transistores

**1970s**

8-bit Microprocessor



4.500 Transistores

**1980s**

32-bit Microprocessor



275.000 Transistores

**1990s**

32-bit Microprocessor



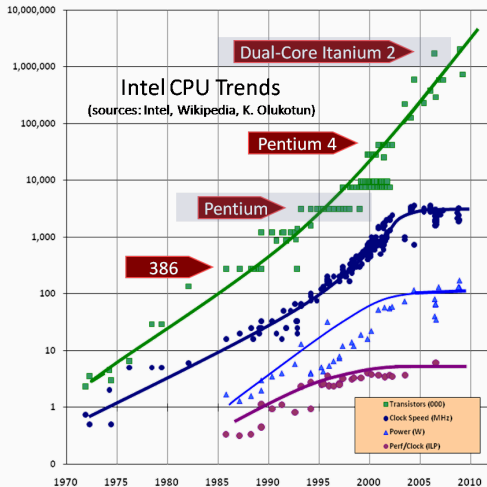
3.100.000 Transistores

**2000s**

64-bit Microprocessor

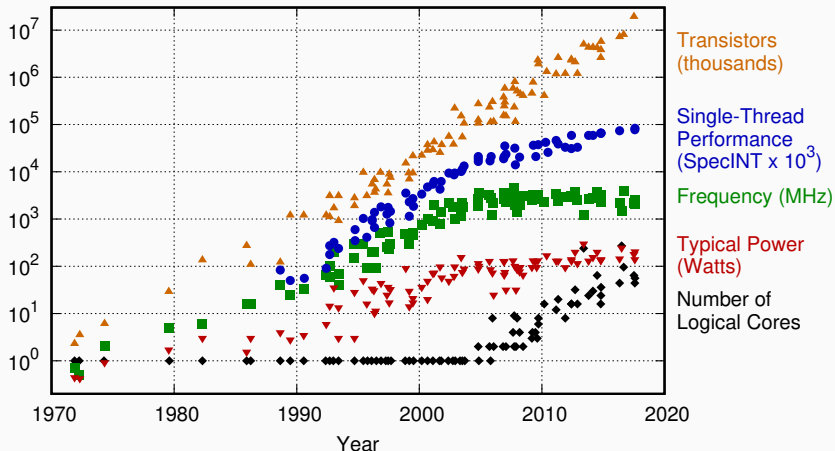


592.000.000 Transistores



- Densidade do chip continuou a dobrar a cada 1,5 ano
- O clock não

## 42 Years of Microprocessor Trend Data



Original data up to the year 2010 collected and plotted by M. Horowitz, F. Labonte, O. Shacham, K. Olukotun, L. Hammond, and C. Batten  
New plot and data collected for 2010-2017 by K. Rupp

# Uma pequena aula de física

Quais as dificuldades em aumentar o clock de processadores de um único núcleo?

- Problemas de temperatura

# Uma pequena aula de física

Quais as dificuldades em aumentar o clock de processadores de um único núcleo?

- Problemas de temperatura
  - Transistores menores  $\Rightarrow$  processadores mais rápidos

# Uma pequena aula de física

Quais as dificuldades em aumentar o clock de processadores de um único núcleo?

- Problemas de temperatura
  - Transistores menores  $\Rightarrow$  processadores mais rápidos
  - Processadores mais rápidos  $\Rightarrow$  aumento do consumo de energia



# Uma pequena aula de física

Quais as dificuldades em aumentar o clock de processadores de um único núcleo?

- Problemas de temperatura
  - Transistores menores  $\Rightarrow$  processadores mais rápidos
  - Processadores mais rápidos  $\Rightarrow$  aumento do consumo de energia
  - Aumento do consumo de energia  $\Rightarrow$  aumento do calor

# Uma pequena aula de física

Quais as dificuldades em aumentar o clock de processadores de um único núcleo?

- Problemas de temperatura
  - Transistores menores  $\Rightarrow$  processadores mais rápidos
  - Processadores mais rápidos  $\Rightarrow$  aumento do consumo de energia
  - Aumento do consumo de energia  $\Rightarrow$  aumento do calor
  - Aumento do calor  $\Rightarrow$  processadores não confiáveis

# Uma pequena aula de física

Quais as dificuldades em aumentar o clock de processadores de um único núcleo?

- Problemas de temperatura
  - Transistores menores  $\Rightarrow$  processadores mais rápidos
  - Processadores mais rápidos  $\Rightarrow$  aumento do consumo de energia
  - Aumento do consumo de energia  $\Rightarrow$  aumento do calor
  - Aumento do calor  $\Rightarrow$  processadores não confiáveis
- Problemas de velocidade da luz

# Uma pequena aula de física

Quais as dificuldades em aumentar o clock de processadores de um único núcleo?

- Problemas de temperatura
  - Transistores menores  $\Rightarrow$  processadores mais rápidos
  - Processadores mais rápidos  $\Rightarrow$  aumento do consumo de energia
  - Aumento do consumo de energia  $\Rightarrow$  aumento do calor
  - Aumento do calor  $\Rightarrow$  processadores não confiáveis
- Problemas de velocidade da luz
- Limitações de litografia (tamanho dos transistores)

# Uma pequena aula de física

Quais as dificuldades em aumentar o clock de processadores de um único núcleo?

- Problemas de temperatura
  - Transistores menores  $\Rightarrow$  processadores mais rápidos
  - Processadores mais rápidos  $\Rightarrow$  aumento do consumo de energia
  - Aumento do consumo de energia  $\Rightarrow$  aumento do calor
  - Aumento do calor  $\Rightarrow$  processadores não confiáveis
- Problemas de velocidade da luz
- Limitações de litografia (tamanho dos transistores)
- Produção e verificação difíceis

# Uma pequena aula de física

Quais as dificuldades em aumentar o clock de processadores de um único núcleo?

- Problemas de temperatura
  - Transistores menores  $\Rightarrow$  processadores mais rápidos
  - Processadores mais rápidos  $\Rightarrow$  aumento do consumo de energia
  - Aumento do consumo de energia  $\Rightarrow$  aumento do calor
  - Aumento do calor  $\Rightarrow$  processadores não confiáveis
- Problemas de velocidade da luz
- Limitações de litografia (tamanho dos transistores)
- Produção e verificação difíceis

# Uma pequena aula de física

Quais as dificuldades em aumentar o clock de processadores de um único núcleo?

- Problemas de temperatura
  - Transistores menores  $\Rightarrow$  processadores mais rápidos
  - Processadores mais rápidos  $\Rightarrow$  aumento do consumo de energia
  - Aumento do consumo de energia  $\Rightarrow$  aumento do calor
  - Aumento do calor  $\Rightarrow$  processadores não confiáveis
- Problemas de velocidade da luz
- Limitações de litografia (tamanho dos transistores)
- Produção e verificação difíceis

**Como continuar aumentando o poder de processamento dos processadores?**

# Uma pequena aula de física

Quais as dificuldades em aumentar o clock de processadores de um único núcleo?

- Problemas de temperatura
  - Transistores menores  $\Rightarrow$  processadores mais rápidos
  - Processadores mais rápidos  $\Rightarrow$  aumento do consumo de energia
  - Aumento do consumo de energia  $\Rightarrow$  aumento do calor
  - Aumento do calor  $\Rightarrow$  processadores não confiáveis
- Problemas de velocidade da luz
- Limitações de litografia (tamanho dos transistores)
- Produção e verificação difíceis

Como continuar aumentando o poder de processamento dos processadores?

- Processadores multicore

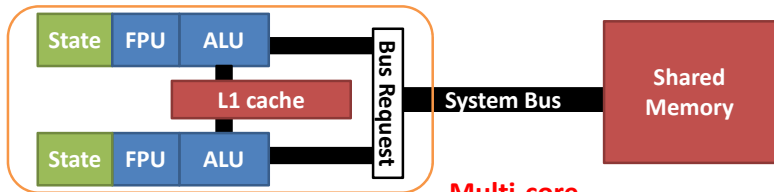
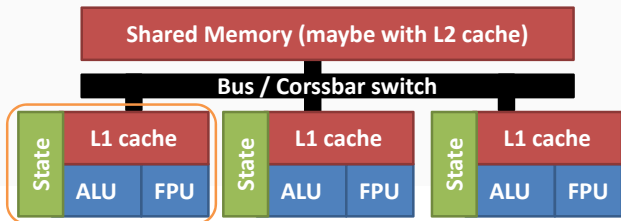


# Multiprocessadores

---

# Multiprocessadores modernos

## Multi-CPU



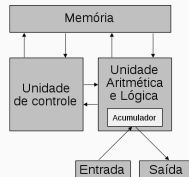
## Multi-core

- FPU: Floating-point unit
- ALU: Arithmetic logic unit

# Arquitetura de von Neumann

- Composta por quatro componentes principais:

- Memória
- Unidade de controle
- Unidade aritmética e lógica
- Entrada/Saída



- Memória de acesso aleatório é usada para armazenar as instruções e os dados do programa
  - Instruções: dados codificados que dizem ao computador o que fazer
  - Dados: informações a serem usadas pelo programa
- A unidade de controle obtém instruções/dados da memória, decodifica as instruções e, em seguida, coordena sequencialmente as operações para realizar a tarefa programada
- Unidade aritmética e lógica executa operações aritméticas básicas
- Entrada/Saída é a interface para o operador humano

So what? Who cares?

## So what? Who cares?

- Computadores paralelos ainda seguem este design básico, apenas multiplicado em unidades
- A arquitetura básica e fundamental continua a ser a mesma

# O que é um computador paralelo?

## Uma definição comum

- Um computador paralelo é uma coleção de elementos de processamento que cooperam para resolver problemas rapidamente

## Por que precisamos saber sobre hardware?

- Porque as características da máquina realmente importam
- Porque você se preocupa com a eficiência e o desempenho (você está escrevendo programas paralelos afinal de contas!)

- Existem diversas classificações para as arquiteturas paralelas
- Devido a constante evolução, nenhuma classificação consegue abranger todas as arquiteturas existentes

## Classificação de Flynn (1972)

- “Some computer organizations and their effectiveness”, IEEE Transactions on Computers, vol. C-21, pp. 948-960, 1972
- Baseia-se na unicidade e multiplicidade do fluxo de dados e instruções
- Mais conhecida

## Classificação de Duncan (1990)

- “A survey of parallel computer architectures”, IEEE Computer, pp. 5-16, Fevereiro, 1990
- Classificação mais recente e abrangente
- Menos conhecida



## Taxonomia de Flynn

- Proposta por Michael J. Flynn em 1972
- Distingue arquiteturas de computadores de acordo a quantidade de instruções e dados processados em um determinado momento
- Uma das metodologias mais conhecidas e utilizadas para classificar a arquitetura de um computador ou conjunto de computadores
- Classifica a arquitetura dos computadores segundo duas dimensões independentes: **instruções** e **dados**, em que cada dimensão pode tomar apenas um de dois valores distintos: **simples** ou **múltiplo**.

## Fluxo de dados

Fluxo de instruções

	Único	Múltiplo
Único	<b>SISD</b> <i>Single Instruction Single Data</i> Instrução Simples de Dados Simples	<b>SIMD</b> <i>Single Instruction Multiple Data</i> Instrução Simples de Múltiplos Dados
Múltiplo	<b>MISD</b> <i>Multiple Instruction Single Data</i> Instrução Múltipla de Dados Simples	<b>MIMD</b> <i>Multiple Instruction Multiple Data</i> Instrução Múltipla de Dados Múltiplos

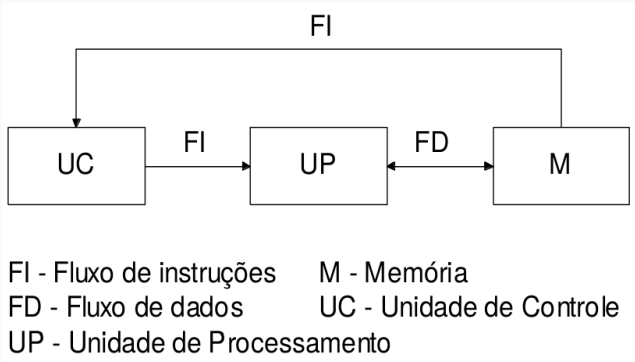
## SISD: único fluxo de instrução, único fluxo de dados

- Classe que representa os computadores convencionais (seriais), a arquitetura básica de von Neumann
  - **Fluxo Simples de Instruções:** Apenas um fluxo de instruções e processado de cada vez
  - **Fluxo Simples de Dados:** Um único fluxo de dados existe entre processador e memória
  - Execução determinística
- As instruções são executadas serialmente, porém os estágios (busca da instrução, decodificação, busca do operando e execução) podem ser sobrepostos (*pipeline*)
- Pode-se saber o que está ocorrendo exatamente em cada instante de tempo e reproduzir o processo passo a passo mais tarde

	Single Instruction	Multiple Instruction
Single Data	SISD	MISD
Multiple Data	SIMD	MIMD

# Taxonomia de Flynn

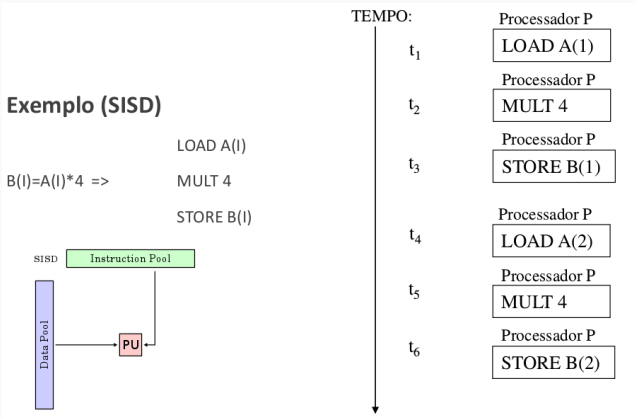
**SISD:** único fluxo de instrução, único fluxo de dados



	Single Instruction	Multiple Instruction
Single Data	SISD	MISD
Multiple Data	SIMD	MIMD

# Taxonomia de Flynn

**SISD:** único fluxo de instrução, único fluxo de dados



	Single Instruction	Multiple Instruction
Single Data	SISD	MISD
Multiple Data	SIMD	MIMD

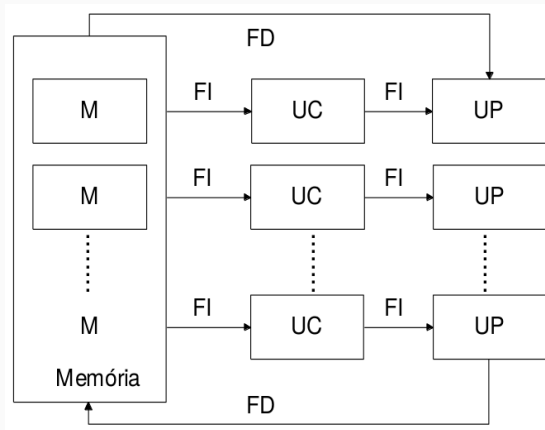
## MISD: múltiplo fluxo de instruções, único fluxo de dados

- Vários processadores, onde cada um recebe instruções distintas, mas operam sobre o mesmo conjunto de dados
- Poucos exemplos
  - Múltiplos filtros de frequência operando sobre um único fluxo de sinal
  - Múltiplos algoritmos de criptografia para decodificar uma mensagem

	Single Instruction	Multiple Instruction
Single Data	SISD	MISD
Multiple Data	SIMD	MIMD

# Taxonomia de Flynn

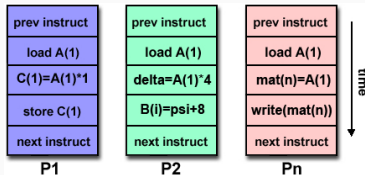
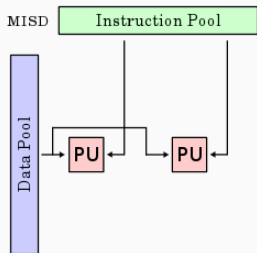
**MISD:** múltiplo fluxo de instruções, único fluxo de dados



	Single Instruction	Multiple Instruction
Single Data	SISD	<b>MISD</b>
Multiple Data	SIMD	MIMD

# Taxonomia de Flynn

**MISD:** múltiplo fluxo de instruções, único fluxo de dados



	Single Instruction	Multiple Instruction
Single Data	SISD	<b>MISD</b>
Multiple Data	SIMD	MIMD



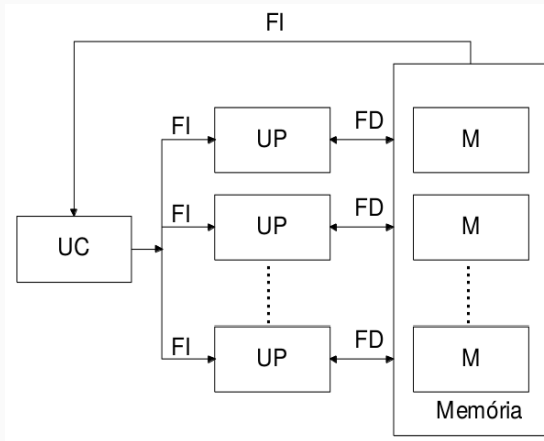
## **SIMD:** único fluxo de instruções, múltiplo fluxo de dados

- Classe que representa os processadores matriciais, paralelos e associativos
  - **Única instrução:** Todas as unidades de processamento executam a mesma instrução
  - **Vários dados:** Cada unidade de processamento pode operar em um elemento de dados diferentes
- Uma única unidade de controle que envia um fluxo de instruções para vários processadores
- Os processadores recebem a mesma instrução ao mesmo tempo e atuam sobre diferentes fluxos de dados
- GPUs são normalmente implementações SIMD

	Single Instruction	Multiple Instruction
Single Data	SISD	MISD
Multiple Data	<b>SIMD</b>	MIMD

# Taxonomia de Flynn

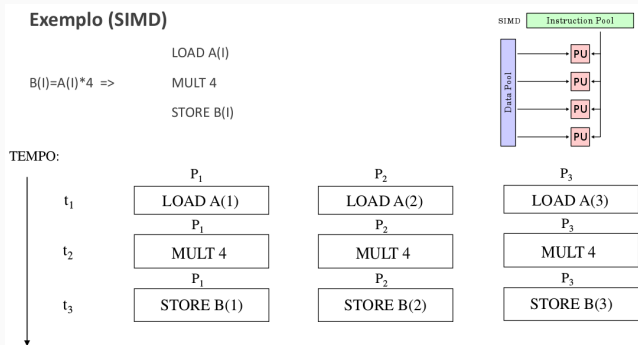
**SIMD:** único fluxo de instruções, múltiplo fluxo de dados



	Single Instruction	Multiple Instruction
Single Data	SISD	MISD
Multiple Data	<b>SIMD</b>	MIMD

# Taxonomia de Flynn

**SIMD:** único fluxo de instruções, múltiplo fluxo de dados



	Single Instruction	Multiple Instruction
Single Data	SISD	MISD
Multiple Data	<b>SIMD</b>	MIMD

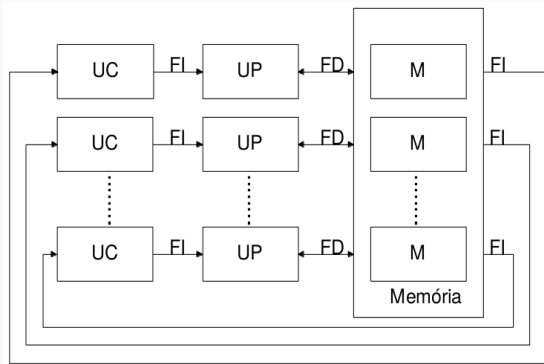
## MIMD: múltiplo fluxo de instruções, múltiplo fluxo de dados

- Cada processador pode executar um **fluxo de instruções diferentes**
- Cada processador pode trabalhar com **diferentes fluxos de dados**
- Podem ser síncronos ou assíncronos
- Atualmente o tipo mais comum de computador paralelo

	Single Instruction	Multiple Instruction
Single Data	SISD	MISD
Multiple Data	SIMD	<b>MIMD</b>

# Taxonomia de Flynn

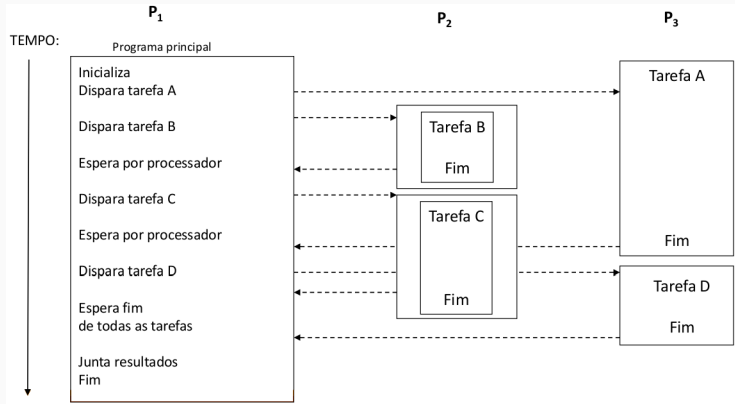
**MIMD:** múltiplo fluxo de instruções, múltiplo fluxo de dados



	Single Instruction	Multiple Instruction
Single Data	SISD	MISD
Multiple Data	SIMD	<b>MIMD</b>

# Taxonomia de Flynn

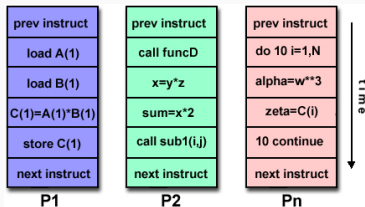
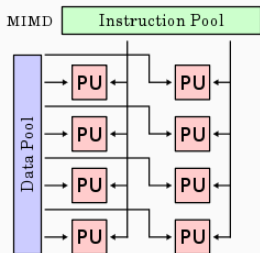
**MIMD:** múltiplo fluxo de instruções, múltiplo fluxo de dados



	Single Instruction	Multiple Instruction
Single Data	SISD	MISD
Multiple Data	SIMD	<b>MIMD</b>

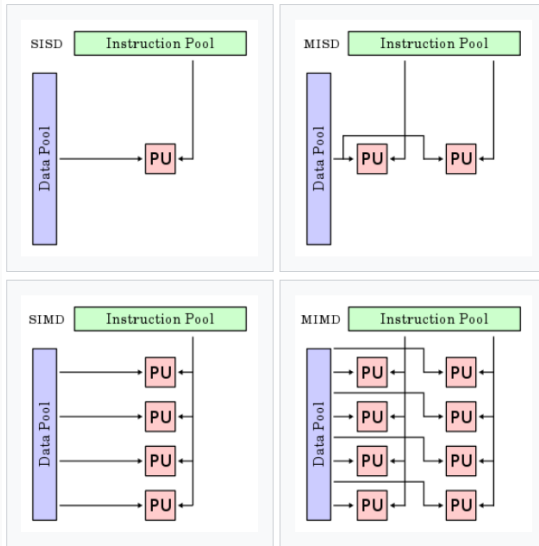
# Taxonomia de Flynn

**MIMD:** múltiplo fluxo de instruções, múltiplo fluxo de dados



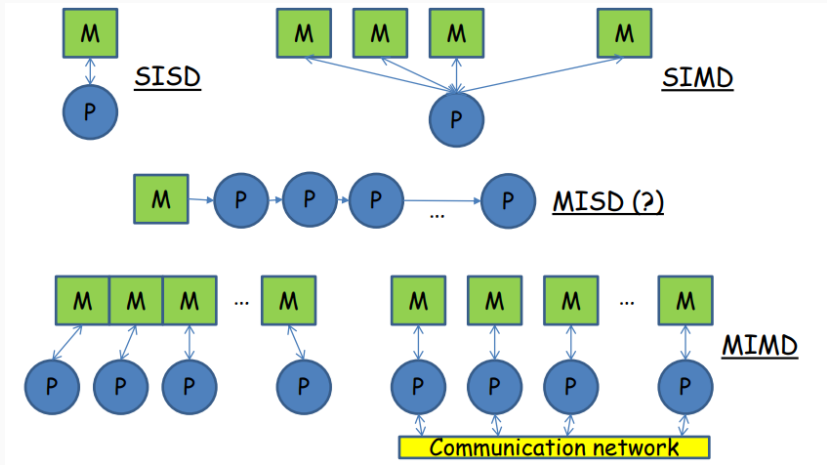
	Single Instruction	Multiple Instruction
Single Data	SISD	MISD
Multiple Data	SIMD	<b>MIMD</b>

# Taxonomia de Flynn

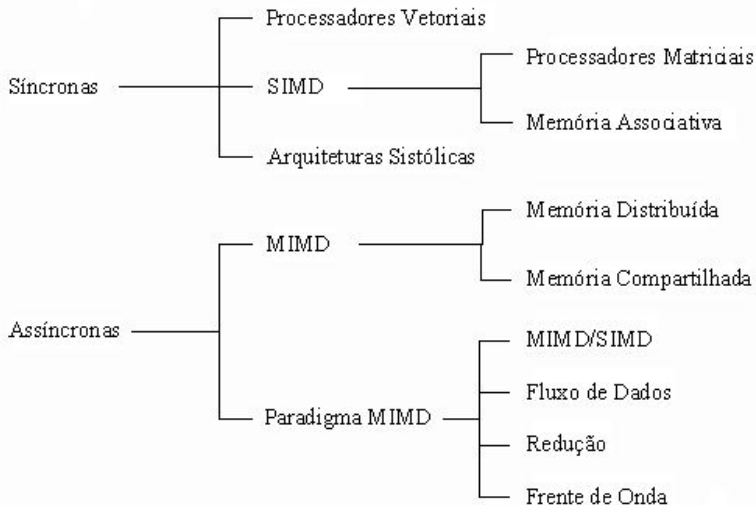




# Taxonomia de Flynn

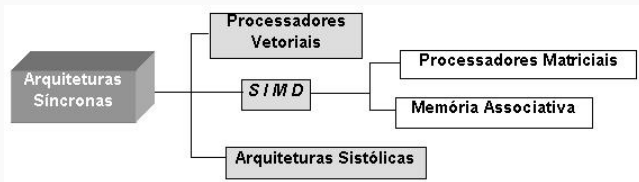


# Taxonomia de Duncan



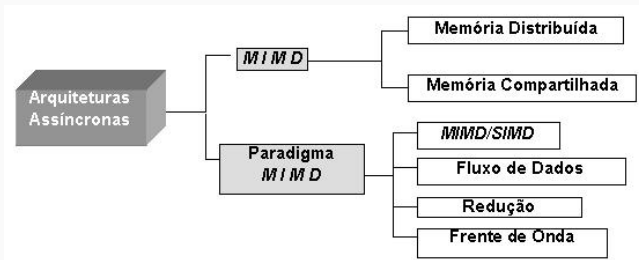
## Arquiteturas Síncronas

- As operações são coordenadas através de um relógio global, dessa forma os processadores permanecem síncronos
- Pouca flexibilidade para desenvolvimento de algoritmos paralelos



## Arquiteturas Assíncronas

- Caracterizam-se pelo controle descentralizado de hardware, de maneira que os processadores sejam independentes entre si
- Formada basicamente pelas arquiteturas MIMD, sejam convencionais ou não



1. Por que é tão difícil/complicado construir processadores com transistores menores?
2. “Se um único processador consegue resolver um problema em  $n$  segundos,  $n$  processadores conseguem resolver o mesmo problema em 1 segundo?”

- [https://en.wikipedia.org/wiki/Flynn%27s\\_taxonomy](https://en.wikipedia.org/wiki/Flynn%27s_taxonomy)
- De-Mystifying Software Performance Optimization  
<https://software.intel.com/en-us/articles/de-mystifying-software-performance-optimization>
- Introduction to Parallel Computing  
[https://computing.llnl.gov/tutorials/parallel\\_comp/](https://computing.llnl.gov/tutorials/parallel_comp/)
- Material do Prof. Fernando Santos Osório  
<http://conteudo.icmc.usp.br/pessoas/fosorio/fosorio-usp.html>