# Arquitetura e Organização de Sistemas Computadorizados - Processador

Osmar de Oliveira Braz Junior Márcia Cargnin Martins Giraldi



# **Objetivos**

 Apresentar os conceitos e tipos de processadores de computadores



## **Processador**

- Peça fundamental dos computadores
- Responsável direto pela movimentação e manipulação de dados
- Componente mais complexo e mais importante
- Circuito Integrado que realiza as funções de cálculo e tomada de decisão do computador
- Frequência de processamento, L1-Cache, Core





# Arquitetura do Processador

- Descreve o processador que está sendo usado no computador
- Grande parte dos computadores vem com identificação descrevendo o processador que contém dentro de si, arquitetura RISC ou CISC.





# Arquitetura do Processador

 CISC (Complex Instruction Set Computing) – computador com um conjunto complexo de instruções: usado em processadores Intel e AMD; suporta mais instruções porém mais lenta a execução delas







# Arquitetura do Processador

RISC (Reduced Instruction Set Computing) – usada em processadores PowerPC (da Apple, Motorola e IBM) e SPARC (SUN). Suporta menos instruções executando com maior rapidez o conjunto de instruções que são combinadas.









# CISC (complex instructions)

10100100

# RISC (simple instructions)

0100101110010011

0111100100100111

0111100111110100

01011101010111111 110011110011111

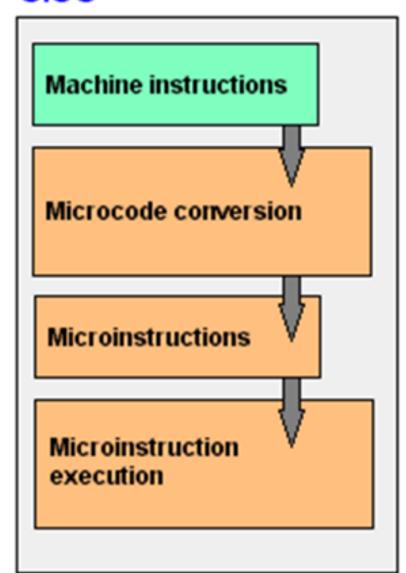
011111110011101



## CISC

- Este processadores contêm uma microprogramação, ou seja, um conjunto de códigos de instruções que são gravados no processador
- permite receber as instruções dos programas e executá-las, utilizando as instruções contidas na sua microprogramação
- reduz o tamanho do código executável por já possuírem muito do código comum em vários programas, em forma de uma única instrução

#### CISC





## CISC

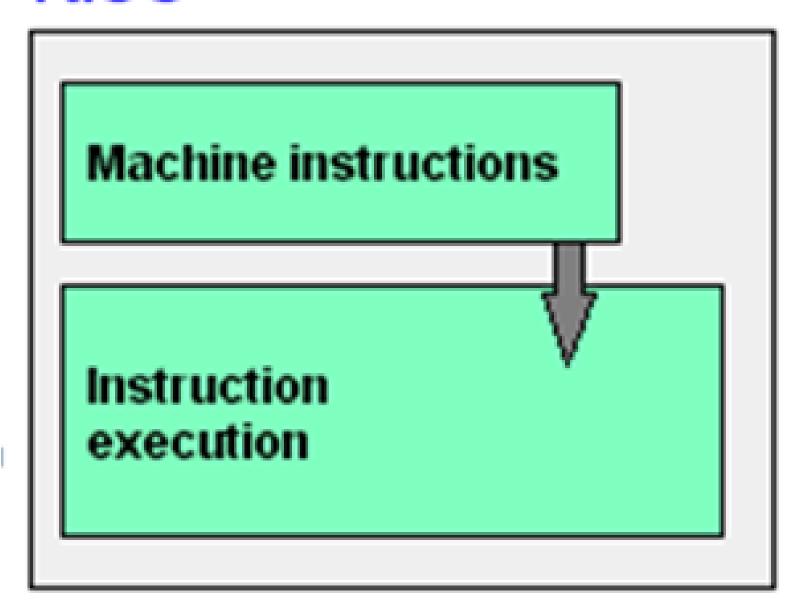
- Operação com o "a = a + b" descrita como "add a,b"
- podem simplesmente utilizar dois operandos para uma única instrução, sendo um deles fonte e destino (acumulador) e permite um ou mais operadores em memória para a realização das instruções



## **RISC**

- No começo da década de 80, a tendência era construir chips com conjuntos de instruções cada vez mais complexos. Alguns fabricantes porém, resolveram seguir o caminho oposto, criando o padrão RISC
- Suporta menos instruções, e com isso executa com mais rapidez o conjunto de instruções que são combinadas
- Os chips são mais simples e muito mais baratos por terem um menor número de circuitos internos, podem trabalhar a frequências mais altas

# RISC





## **RISC**

- Operação com o "a = b + c" descrita como "add a,b,c"
- Podem especificar três operandos para uma única instrução, mas exclusivamente se estes forem registradores.
- Cada célula de memória contém apenas 8 bits, um inteiro (32 bits) ocuparia mais de uma célula

RISC	CISC		
Conjunto de instruções semanticamente simples de tamanho fixo reduzido	Conjunto de instruções semanticamente complexas de tamanho variável extenso		
Decodificação simplificada (tabela)	Decodificação complexa (microcódigo)		
Execução regular, interpretada pelo próprio programa	Cada instrução executa à sua maneira, pelo hardware		
Instruções requerem o mesmo número de ciclos de clock para executar	Grande variação no número de ciclos de clock por instrução		
Possibilita o uso de pipeline (intenso)	Extremamente difícil/impossível o uso de pipeline		
Apenas algumas operações (load/store) em memória	Qualquer instrução pode referenciar a memória		
Poucas instruções e modos de endereçamento	Várias instruções e modos de endereçamento		
Múltiplos conjuntos de registradores	Conjunto de registradores únicos (operadores aritméticos, apontadores para a memória e memória segmentada)		
não possuem instruções para multiplicação ou divisão	possuem instruções para multiplicação ou divisão		
	13		



## RISC x CISC

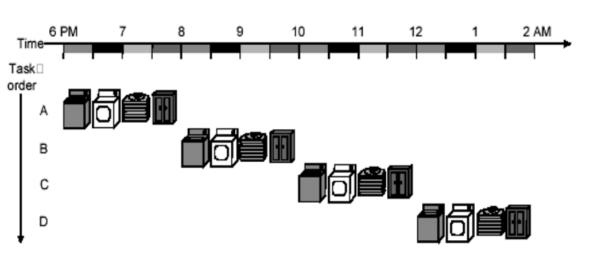
- Atualmente vemos processadores híbridos, que são essencialmente processadores CISC, mas incorporam muitos recursos encontrados nos processadores RISC (ou vice-versa);
- Apesar de por questões de Marketing, muitos fabricantes ainda venderem seus chips, como sendo "Processadores RISC", não existe praticamente nenhum processador atualmente que siga estritamente uma das duas filosofias



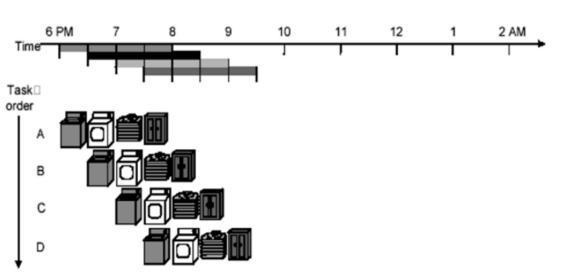
# **Pipeline**

- técnica de hardware que permite que a CPU realize a busca de uma ou mais instruções além da próxima a ser executada
- Estas instruções são colocadas em uma fila de memória dentro do processador (CPU) onde aguardam o momento de serem executadas: assim que uma instrução termina o primeiro estágio e parte para o segundo, a próxima instrução já ocupa o primeiro estágio
- Em resumo, é o processo pelo qual uma instrução de processamento é subdividido em etapas, uma vez que cada uma destas etapas é executada por uma porção especializada da CPU, podendo colocar mais de uma instrução em execução simultânea.

#### **Exemplo sem Pipeline**



#### **Exemplo com Pipeline**





# **Pipeline**

	Estágio do pipeline onde a instrução se encontra							
Ciclo de Clock	0	1	2	3	4	5	6	7
Instrução 1	BI	DI	EX	AM	GR			
Instrução 2		BI	DI	EX	AM	GR		
Instrução 3			BI	DI	EX	AM	GR	
Instrução 4				BI	DI	EX	MAME	GR



## Níveis de Paralelismo

- INSTRUÇÃO/ INSTRUCTION (granulosidade fina)
  - □ Paralelismo entre as instruções
  - Arquiteturas Pipeline, Superescalar, VLIW
- TAREFAS/THREADS (granulosidade média)
  - Paralelismo entre as threads
  - Arquiteturas SMT (Simultaneous MultiThreading)
- PROCESSOS/PROCESS (granulosidade grossa)
  - □ Paralelismo entre os processos
  - Computação Paralela
  - □ Arquiteturas multiprocessadores e multicomputadores



# **Definições**

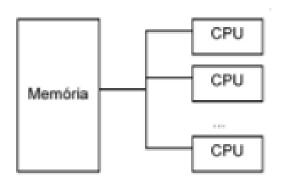
- A execução pode ser:
- Concorrência (Concurrent Execution): é a trocar de execução de recursos entre threads em um processo multithread em um único processador
- Paralelismo (*Parallel Execution*): quando cada thread de um processo pode rodar separadamente em processadores diferentes



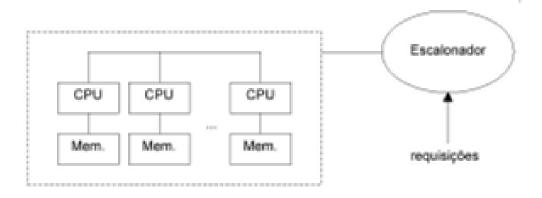
- Permite a execução das tarefas em menor tempo, através da execução em paralelo de diversas tarefas
- O paralelismo pode ser obtido em diversos níveis, com ou sem o uso de linguagens de programação paralela
- Arquiteturas de diversos tipos, elaboradas para aplicações específicas, podem ser utilizadas para acelerar a execução dessas aplicações

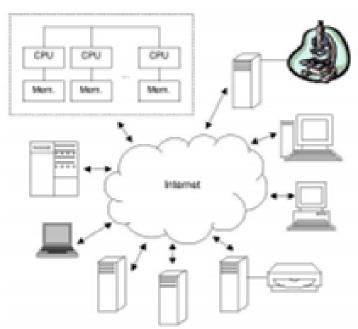


- Programação Sequencial
- Programação Concorrente
- Um servidor, atendendo vários clientes através de um política de escalonamento no tempo
- Programação Paralela
- Vários servidores, atendendo vários clientes simultaneamente no tempo

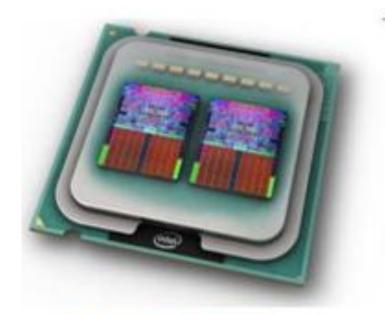


#### Arquiteturas multiprocessada

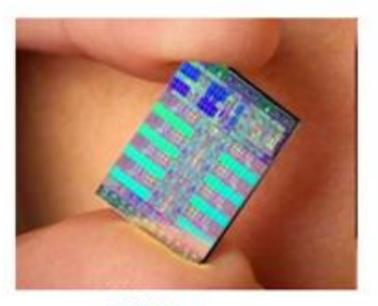




Arquiteturas multiprocessada Arquitetura de um Grid Computacional



Multicore Processor Intel



Cell Processor PlayStation 3

Mas também...
Pipeline, Pré-Fetch, Multi-Level Caches, ...

Parallel Distributed Processing (PDP) ;^)
MultiCore, MultiProcessador, Clusters => Processamento Neural





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



# Taxonomia de Flynn

- A taxonomia de Flynn baseia-se no fato de um computador executar uma sequência de instruções de dados, diferenciase o fluxo de instruções e o fluxo de dados.
- Abrange quatro classes de arquiteturas de computadores:
  - SISD (Single Instruction Single Data): Fluxo único de instruções sobre um único conjunto de dados.
  - □ SIMD (Single Instruction Multiple Data): Fluxo único de instruções em múltiplos conjuntos de dados.
  - MISD (Multiple Instruction Single Data): Fluxo múltiplo de instruções em um único conjunto de dados.
  - MIMD (Multiple Instruction Multiple Data): Fluxo múltiplo de instruções sobre múltiplos conjuntos de dados.



	<b>3</b>						
		FLUXO DE DADOS					
		SINGLE DATA (único)	<b>MULTIPLE DATA</b> (múltiplo)				
FLUXO DE INSTRUÇ ÕES	SINGLE INSTRUCTION (único)	SISD Single Instruction Single Data Instrução Simples de Dados Simples	SIMD Single Instruction Multiple Data Instrução Simples de Dados Múltiplos				
	MULTIPLE INSTRUCTION (múltiplo)	MISD  Multiple Instruction Single Data Múltiplas Instruções de Dados Simples	MIMD  Multiple Instruction Multiple  Data  Múltiplas Instruções da  Dados Múltiplos				

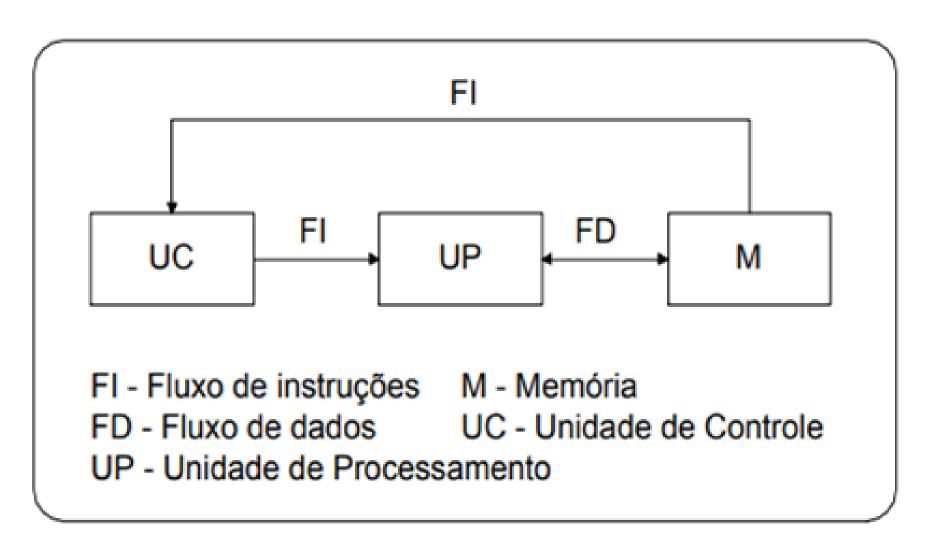


## SISD

- Classe que representa os computadores convencionais (seriais)
- 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

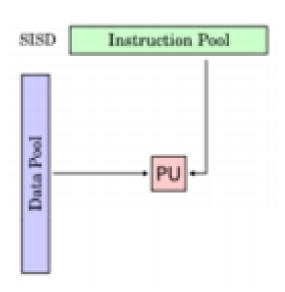


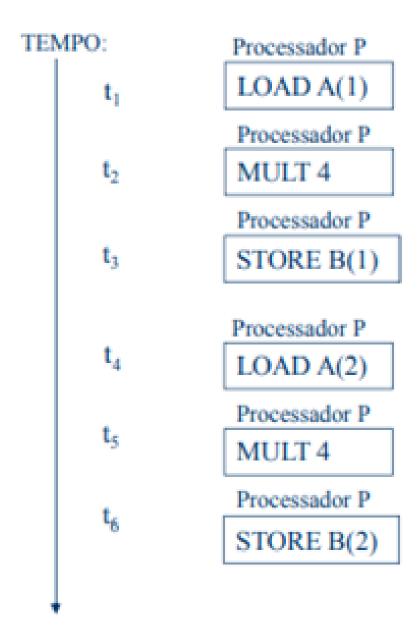
## SISD



## Exemplo (SISD)

For I = 1 to N





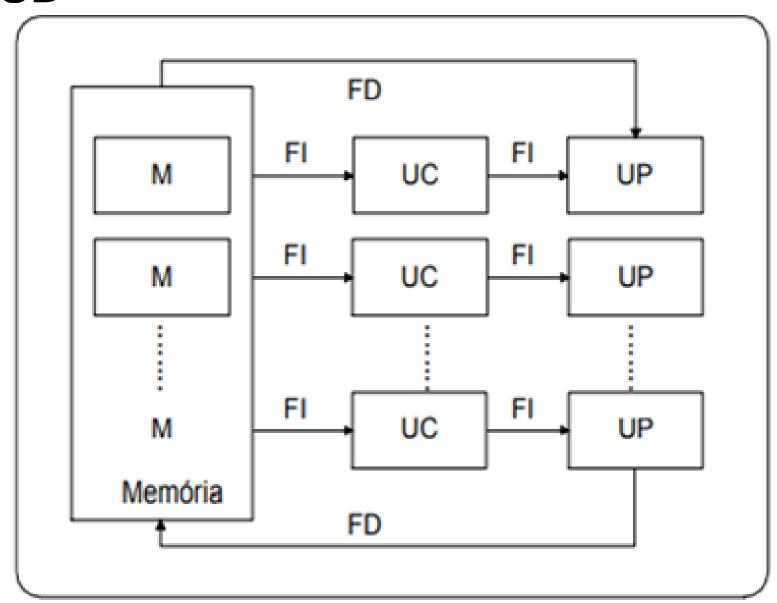


## **MISD**

- 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



# **MISD**

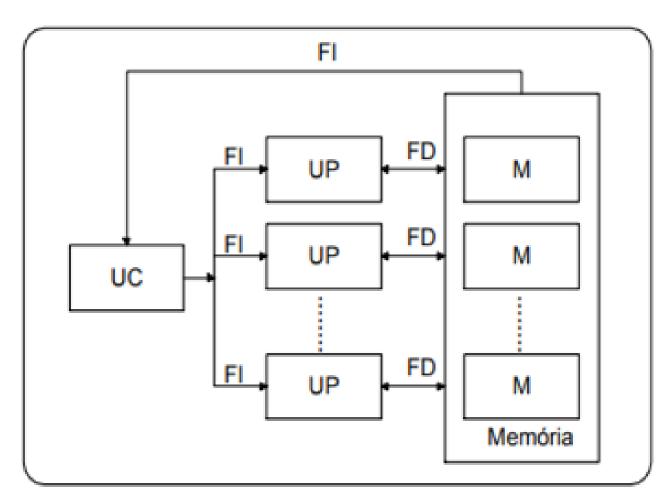




## SIMD

- Classe que representa os processadores matriciais, paralelos e associativos
- 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

## SIMD



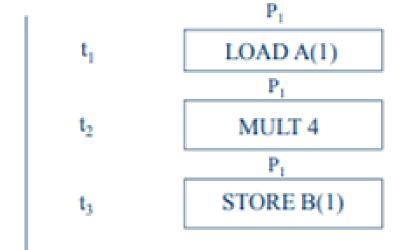
- •1024 processadores em um array 32x32, ou 4096 em 64x64
- processador de controle armazena instruções e dados são armazenados na memória de cada processador

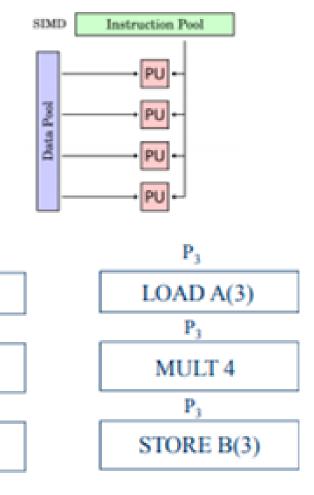
## SIMD

#### Exemplo (SIMD)



#### TEMPO:





 $P_2$ 

LOAD A(2)

Ρ,

MULT 4

STORE B(2)

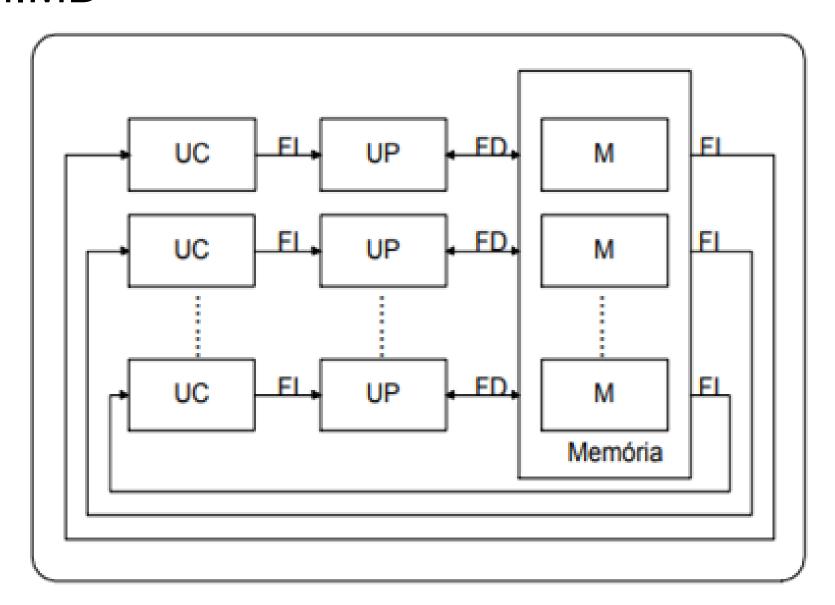


## **MIMD**

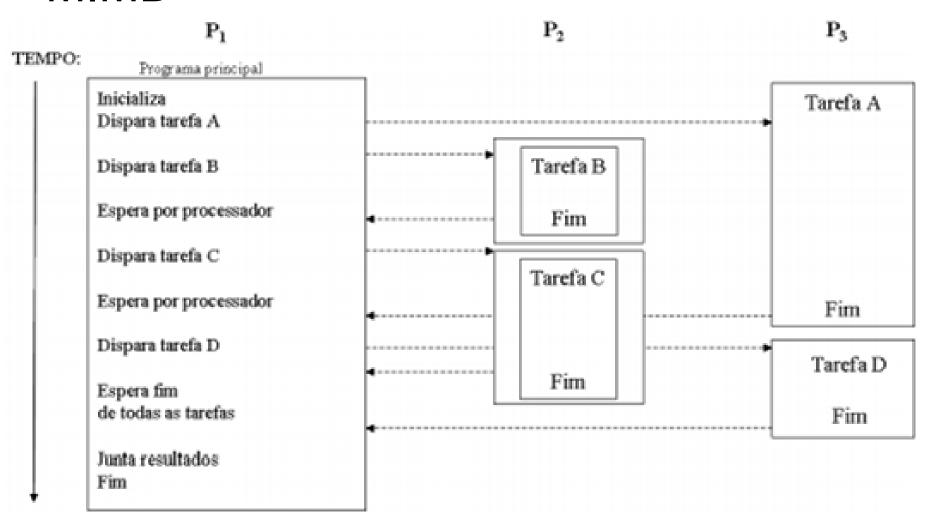
- Vários processadores, cada um controlado por uma unidade de controle
- Processadores recebem instruções diferentes e operam sob fluxo de dados diferentes
- Podem ser síncronos ou assíncronos



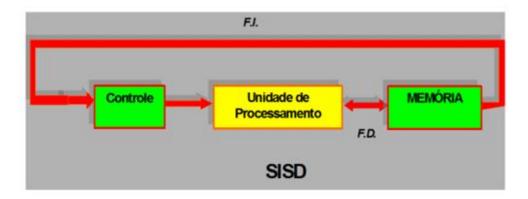
# **MIMD**



# **MIMD**

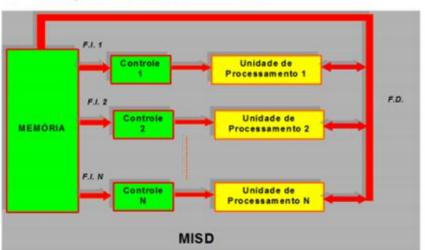


SISD: o computador consiste de uma unidade de processamento que recebe um fluxo simples de instruções e opera sobre um simples fluxo de dados;



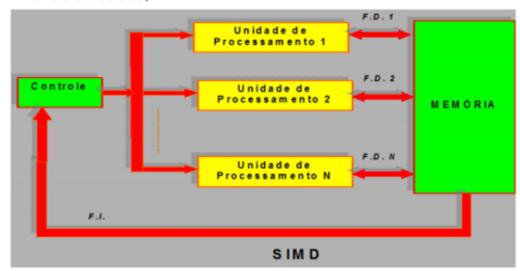
Ex.: computadores von Neumann (o usual)

MISD: N processadores, cada um com sua unidade de controle e unidade de processamento, dividem uma mesma memória e executam diferentes instruções sobre o mesmo dado



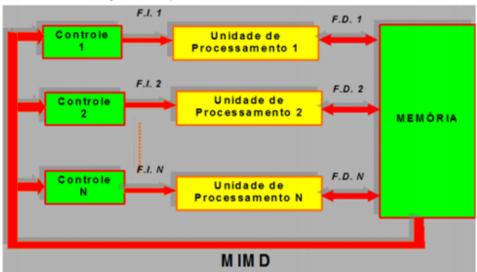
Ex.: arquitetura difícil de ser encontrada. Poderia ser utilizada para aplicar diferentes algorítmos em um mesmo dado.

SIMD: o computador consiste de uma unidade de controle e N unidades de processamento. Todo o processamento esta sobre o controle de um único fluxo de instruções mas opera sobre N fluxos de dados:



Ex.: processadores vetoriais.

MIMD: consiste de N processadores distintos, controlados por N fluxos de instruções e operando sobre N fluxos de dados.



Ex.: multiprocessadores e multicomputadores.



## **Exercícios**

- O que são as instruções VLIW, e qual a vantagem oferecida pelo uso deste tipo de instruções?
- O que é um processador super-escalar? Qual(is) sua(s) principal(is) característica(s)? Qual(is) sua(s) principal(is) limitação?
- Classifique os seguintes processadores entre arquiteturas: SISD, SIMD, MISD e MIMD: a) Zilog Z80 b) Intel 8086 c) Intel Pentium MMX d) Intel Core 2 Duo e) Cell Processor (PS3) f) Cray T90
- O que é uma arquitetura multi-core? Qual o reflexo/impacto de termos um processador multi-core em termos de Sistema Operacional e de programação de aplicativos?



## Conclusão

- Conhecemos um pouco sobre os tipos de processadores e sua arquitetura em computadores.
- A tecnologia continua a evoluir, portanto o estudo n\u00e3o para aqui.



## Referências

- WEBER, Raul Fernando. Fundamentos de arquitetura de computadores. 4. ed. Porto Alegre: Bookman, 2012. E-book. Disponível em: https://integrada.minhabiblioteca.com.br/books/9788540701434
- STALLINGS, William. Arquitetura e organização de computadores. 8.ed. São Paulo: Pearson, 2010. E-book. Disponível em: https://plataforma.bvirtual.com.br/Leitor/Publicacao/459/epub/0
- HOGLUND, Greg. Como quebrar códigos: a arte de explorar (e proteger) software. São Paulo: Pearson, 2006. E-book. Disponível em: https://plataforma.bvirtual.com.br/Leitor/Publicacao/179934/epub/0

