

Arquitetura de Computadores

PROF. ISAAC

Processadores

Arquitetura de Computadores

No desenvolvimento de um projeto, como você escolhe:

1. Uso de CPU, GPU, TPU ou FPGA?
2. ARM ou x86?
3. Threads?
4. Multicomputador ou Multiprocessador?

Arquitetura de processadores

CISC:

Computador com um conjunto complexo de instruções.

RISC:

Computador com um conjunto reduzido de instruções.

RISC - Reduced Instruction Set Computer

Principais Características RISC:

- Possui apenas instruções simples.
- Não possuem micro-programação.
- Instruções de baixo nível de complexidade.
- Instruções com largura fixa.

CISC - Complex Instruction Set Compute

Principais Características CISC:

- Um grande conjunto de instruções.
- Instruções com elevado grau de complexidade.
- Disputa entre tamanho do código x desempenho.
- Redução do tamanho do código executável de um programa.

Comparação entre Arquiteturas RISC e CISC

RISC	CISC
Múltiplos conjuntos de registradores, muitas vezes superando 256	Único conjunto de registradores, tipicamente entre 6 e 16 registradores
Três operandos de registradores permitidos por instrução (por ex., <i>add R1, R2, R3</i>)	Um ou dois operandos de registradores permitidos por instrução (por ex., <i>add R1, R2</i>)
Passagem eficiente de parâmetros por registradores no chip (processador)	Passagem de parâmetros ineficiente através da memória
Instruções de um único ciclo (ex. <i>load</i> e <i>store</i>)	Instruções de múltiplos ciclos
Controle <i>hardwired</i> (embutido no hardware)	Controle microprogramado
Altamente paralelizado (<i>pipelined</i>)	Fracamente paralelizado
Instruções simples e em número reduzido	Muitas instruções complexas
Instruções de tamanho fixo	Instruções de tamanho variável
Complexidade no compilador	Complexidade no código

Arquitetura de processadores

ARM:

A arquitetura ARM é indiscutivelmente o processador mais utilizado em telefones celulares, iPods, equipamentos de sensor remoto e muitos outros dispositivos.

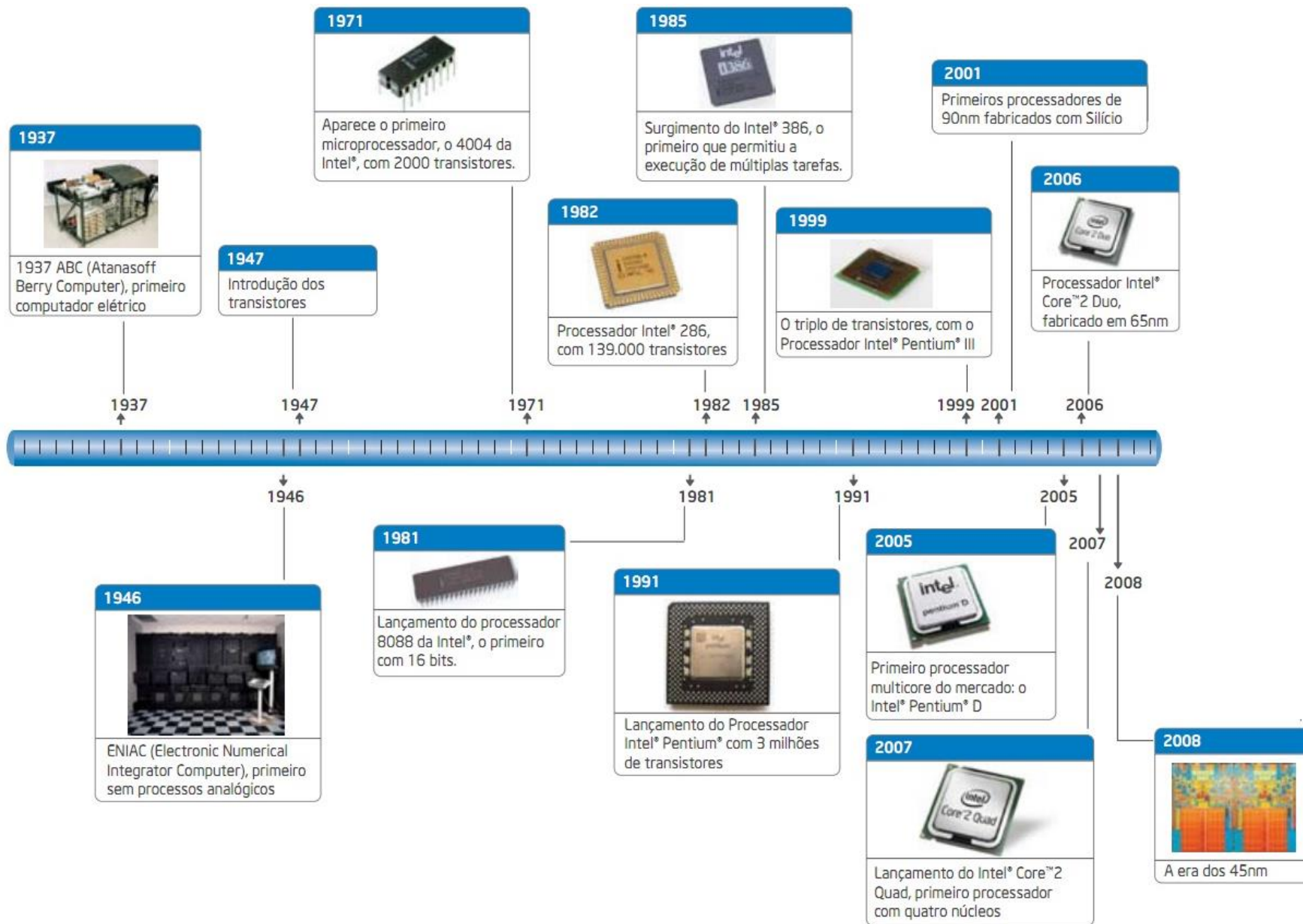
A ARM é essencialmente um computador com conjunto de instruções reduzido (RISC). Membros recentes da família ARM fazem uso de princípios de projeto superescalar e multicore.

Intel x86:

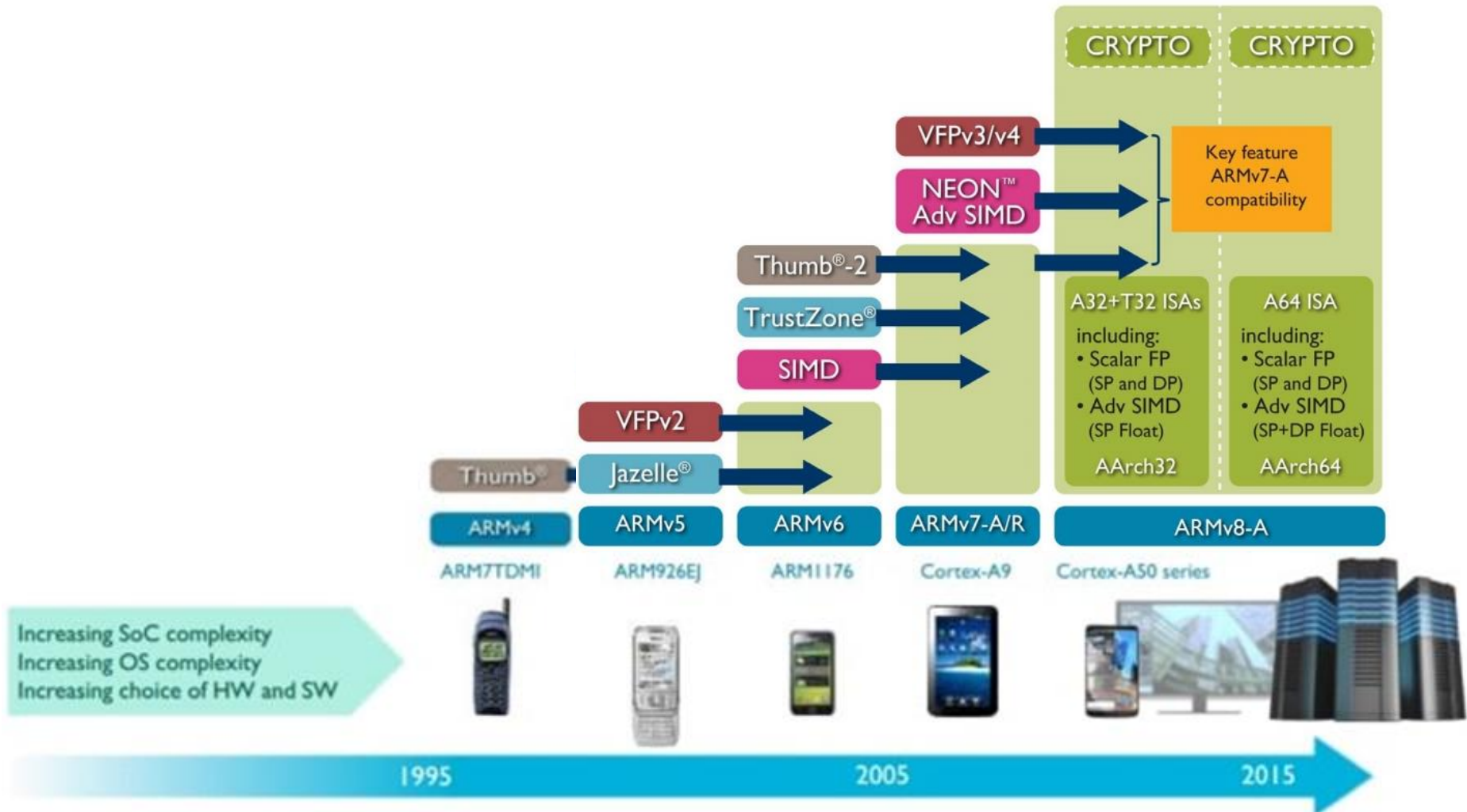
A arquitetura x86 é a mais utilizada para sistemas de computador não embarcados. O x86 é essencialmente um computador CISC com algumas características RISC. Membros recentes da família x86 usam princípios de projeto superescalar e multicore.

A evolução dos recursos na arquitetura x86 fornece um estudo de caso único da evolução da maioria dos princípios de projeto na arquitetura do computador.

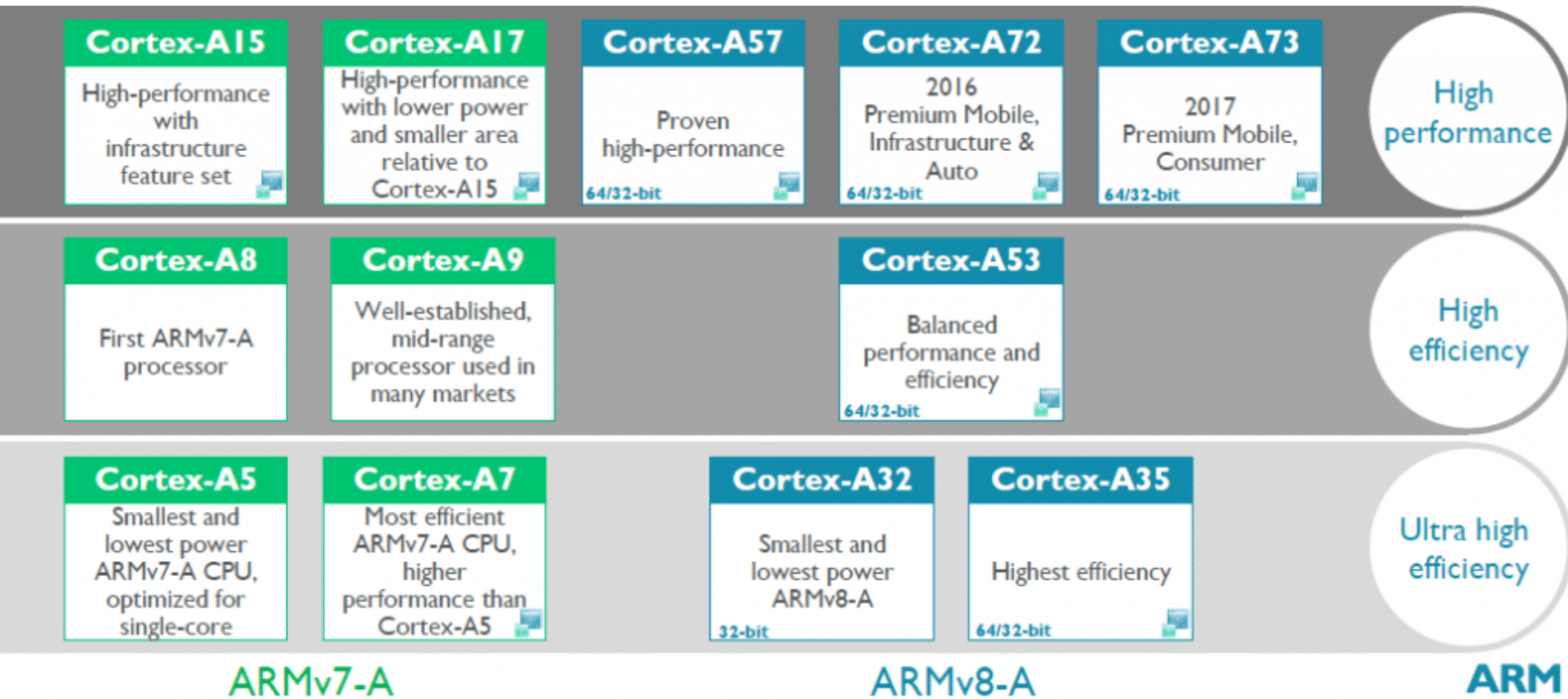
Evolução da arquitetura Intel x86



Evolução da arquitetura ARM



Arquitectura ARM



Exemplos

- Celulares e tablets...
- Arduino Due: 32-bits ARM Cortex-M3.
- Amazon EC2 A1: 64-bit ARM Neoverse.
- Nvidia Jetson AGX Xavier Series:
 - CPU: ARM v8.2 64-bit 8 núcleos (16GB de RAM).
 - GPU: NVIDIA Volta 512 núcleos CUDA e 64 Tensor cores.

Exercícios de Fixação

1. Possuí um conjunto pequeno e simples de instruções, que levam uma quantidade de tempo muito próxima para serem executadas. É muito utilizada em celulares.
 - a. RISC
 - b. CISC

2. O objetivo da arquitetura RISC foi simplificar a vida do programador, reduzindo o tamanho do código executável de um programa.
 - a. Certo
 - b. Errado

Exercícios de Fixação

3. Acerca das arquiteturas RISC e CISC, analise as alternativas e assinale a INCORRETA.
 - a. A arquitetura RISC apresenta uma simplicidade de hardware quando comparada a arquitetura CISC.
 - b. Os processadores RISC são capazes de executar várias centenas de instruções complexas.
 - c. CISC possuem instruções de múltiplos ciclos.
 - d. A arquitetura RISC possuem instruções de tamanho fixo.

Exercícios de Fixação

4. Em relação a arquitetura RISC, assinale a CORRETA.
- a. Possui poucos registradores, tipicamente entre 6 e 16.
 - b. Possui muitas instruções complexas.
 - c. Possui instruções de múltiplos ciclos.
 - d. Instruções otimizadas, sendo executadas geralmente em um único ciclo.

Exercícios de Fixação

5. Assinale, das alternativas abaixo, a única que NÃO identifica corretamente e especificamente as características típicas de um processador RISC clássico.
- a. Execução sem micro-programação
 - b. Instruções de baixo nível de complexidade.
 - c. Complexidade no compilador.
 - d. Poucos registradores.

Exercícios de Fixação

6. Assinale verdadeiro ou Falso para as afirmações.

- () Arquiteturas do tipo CISC possuem instruções de tamanho fixo
- () RISC possui menor quantidades de instruções se comparado a arquitetura CISC.
- () Arquiteturas do tipo RISC possuem um conjunto de instruções longo e complexo.
- () ARM é essencialmente um processador com conjunto de instruções reduzido (RISC).

Exercícios de Fixação

7. Assinale a alternativa correta, em relação a arquitetura RISC.
- a. Executam as instruções com menos rapidez do que os processadores CISC
 - b. Têm a vantagem de reduzir o tamanho do código executável, já que possuem muitas das instruções no processador.
 - c. Suportam menos instruções do que processadores com arquitetura CISC.
 - d. Têm um auto consumo de energia, porque possuem muitos transistores, mais transistores que processadores CISC.

Exercícios de Fixação

2. O objetivo da arquitetura RISC foi simplificar a vida do programador, reduzindo o tamanho do código executável de um programa.
 - a. Certo
 - b. Errado

Resposta:

Assembly, também conhecida como Linguagem de Máquina, significa programar exatamente da mesma forma que uma CPU ou um microcontrolador executa suas instruções. Ou seja, RISC dificulta a vida do programador assembly, mas caso o programador esteja usando outra linguagem de programação, essa tarefa passa a ser do compilador.

Paralelismo

Formas de Paralelismo

- Paralelismo de instrução
 - Pipeline
 - Arquiteturas superescalares
- Multithread
- Adição de CPUs

Paralelismo de Instrução

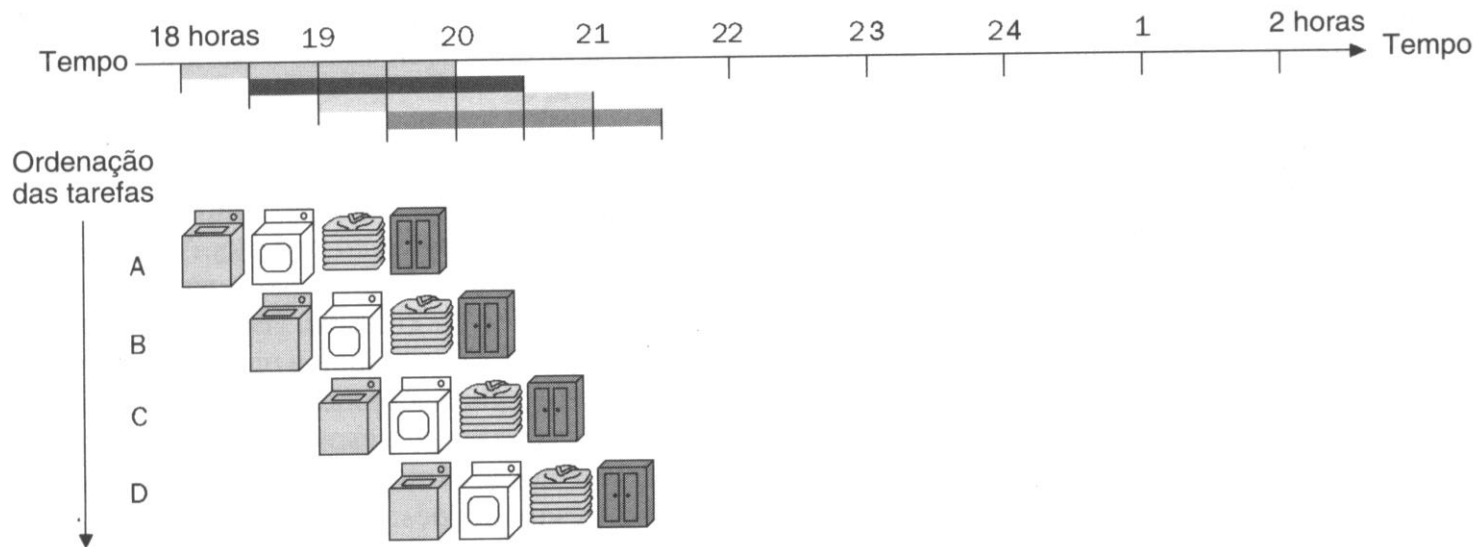
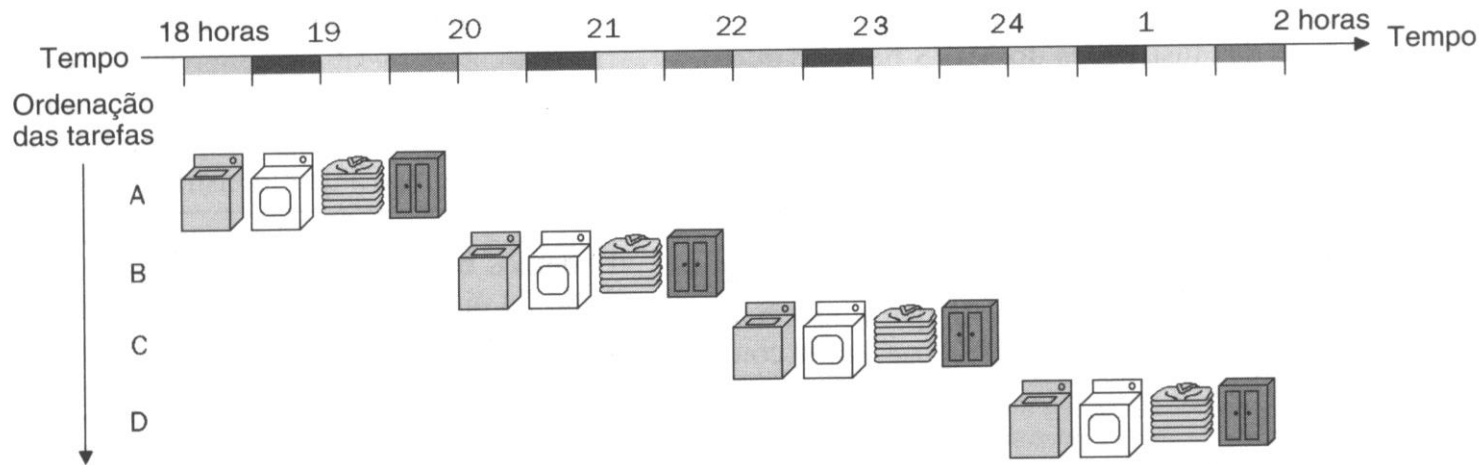
- Feito em hardware
- As instruções já são formadas para serem executadas ao mesmo tempo.
- Exemplos: Pipelines, Superescalares e Very Long Instruction Word (VLIW)

Pipeline

Pipeline

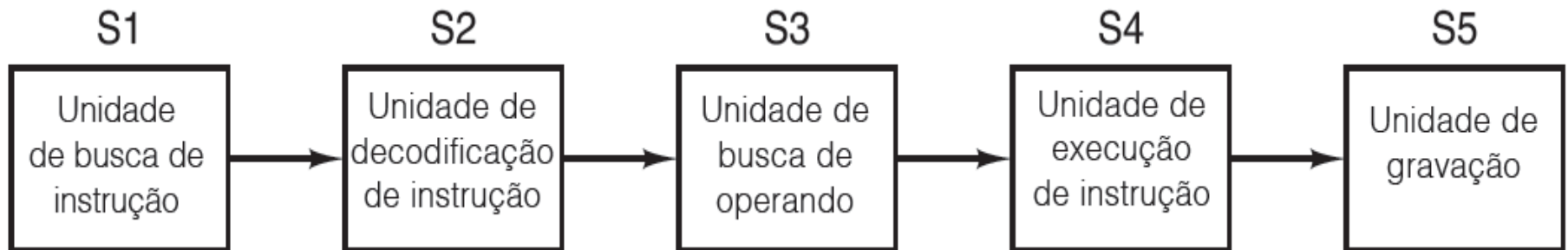
- Técnica onde várias instruções são sobrepostas na execução do programa.
- Uma forma de tentar aproveitar recursos livres em uma CPU.
- Executa partes do programa em paralelo.
- É utilizada para tornar as CPUs mais rápidas.

Analógia do Pipeline

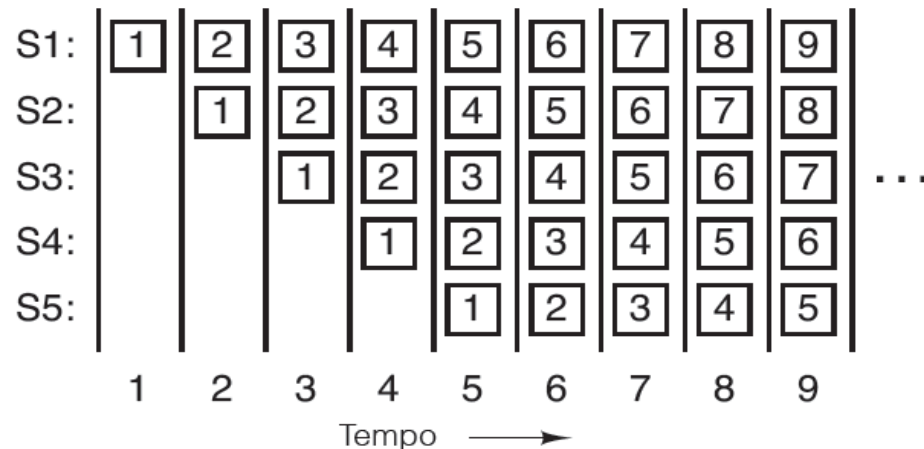


Pipeline

Pipeline de cinco estágios.



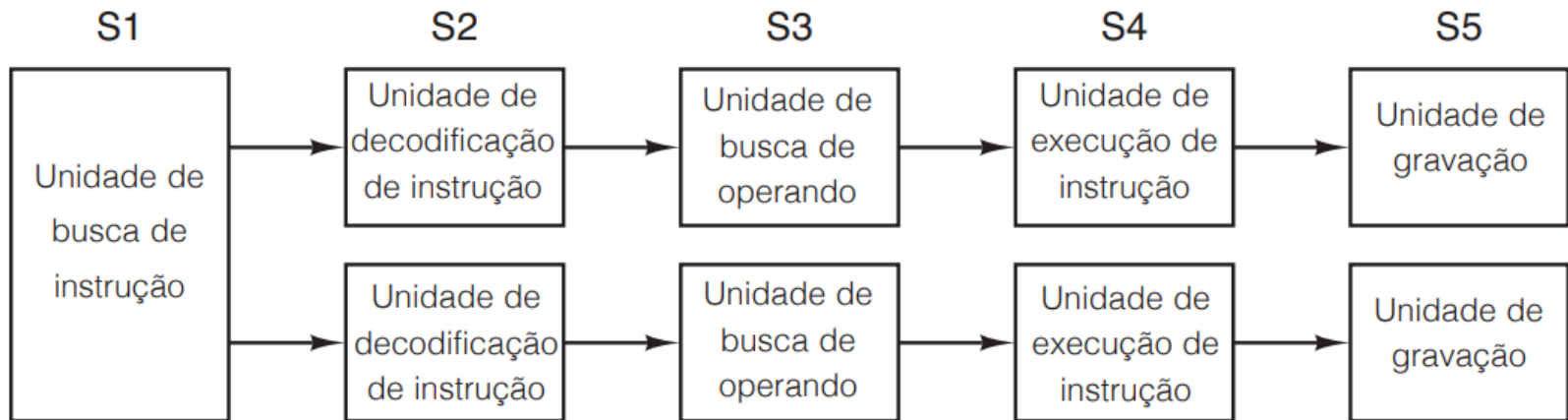
Estado de cada estágio como uma função do tempo.



Arquiteturas Superescalares

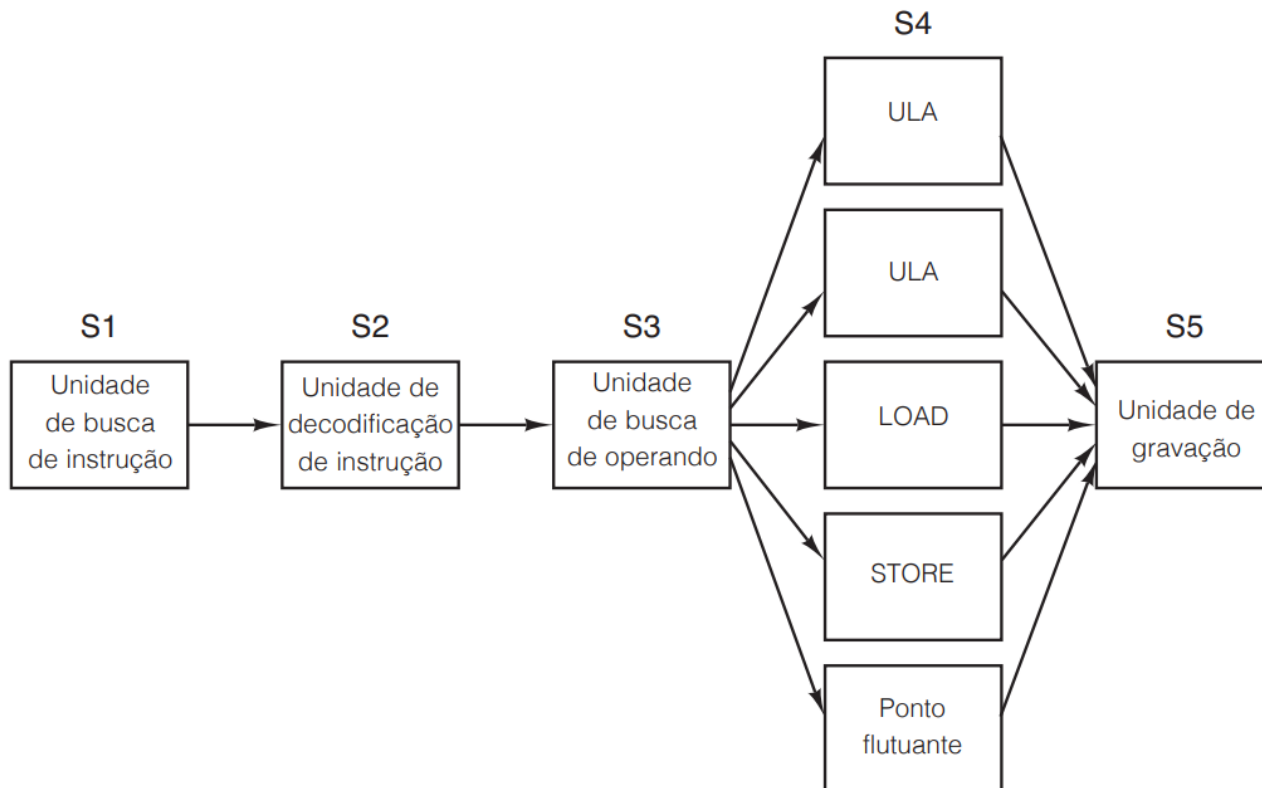
Arquiteturas Superescalar

- Se um pipeline é bom, então certamente dois pipelines são ainda melhores. A figura abaixo apresenta uma CPU com dois pipelines.



Arquiteturas Superescalar

- Uma outra abordagem também é utilizada em CPUs. A ideia básica é ter apenas um único pipeline, mas lhe dar várias unidades funcionais, conforme mostra a Figura:



Multithreads

Multithreads

- Uma thread T1 está em execução
- Uma thread T2 está em espera
- T1 precisa de um dado que vai demorar para chegar...
- Neste instante, o processador fica ocioso esperando o dado para T1
- Por que não usar este tempo para processar T2?

Multithreads

Duas formas de troca de threads:

1. Granulação fina: troca de thread a cada nova instrução.
2. Granulação grossa: troca entre as threads quando é necessário parar para esperar uma informação.

Exemplo

T1

T1.1 MOV A,#10

T1.2 MOV B,#20

T1.3 MUL AB

T1.4 MOV R0,A

T1.5 MOV R1,B

T2

T2.1 MOV A,#10

T2.2 MOV B,#20

T2.3 DIV AB

T2.4 MOV R0,A

T2.5 MOV R1,B

Exemplo

Tempo	T+0	T+1	T+2	T+3	T+4	T+5	T+6	T+7	T+8	T+9
Serial	T1.1	T1.2	T1.3	T1.4	T1.5	T2.1	T2.2	T2.3	T2.4	T2.5
MT G Fina	T1.1	T2.1	T1.2	T2.2	T1.3	T2.3	T1.4	T2.4	T1.5	T2.5
MT G Grossa	T1.1	T1.2	T1.3	T2.1	T2.2	T2.3	T1.4	T1.5	T2.4	T2.5

Multiprocessador e Multicomputador

Processador vs Computador

“Maneira de organizar o acesso aos dados”.

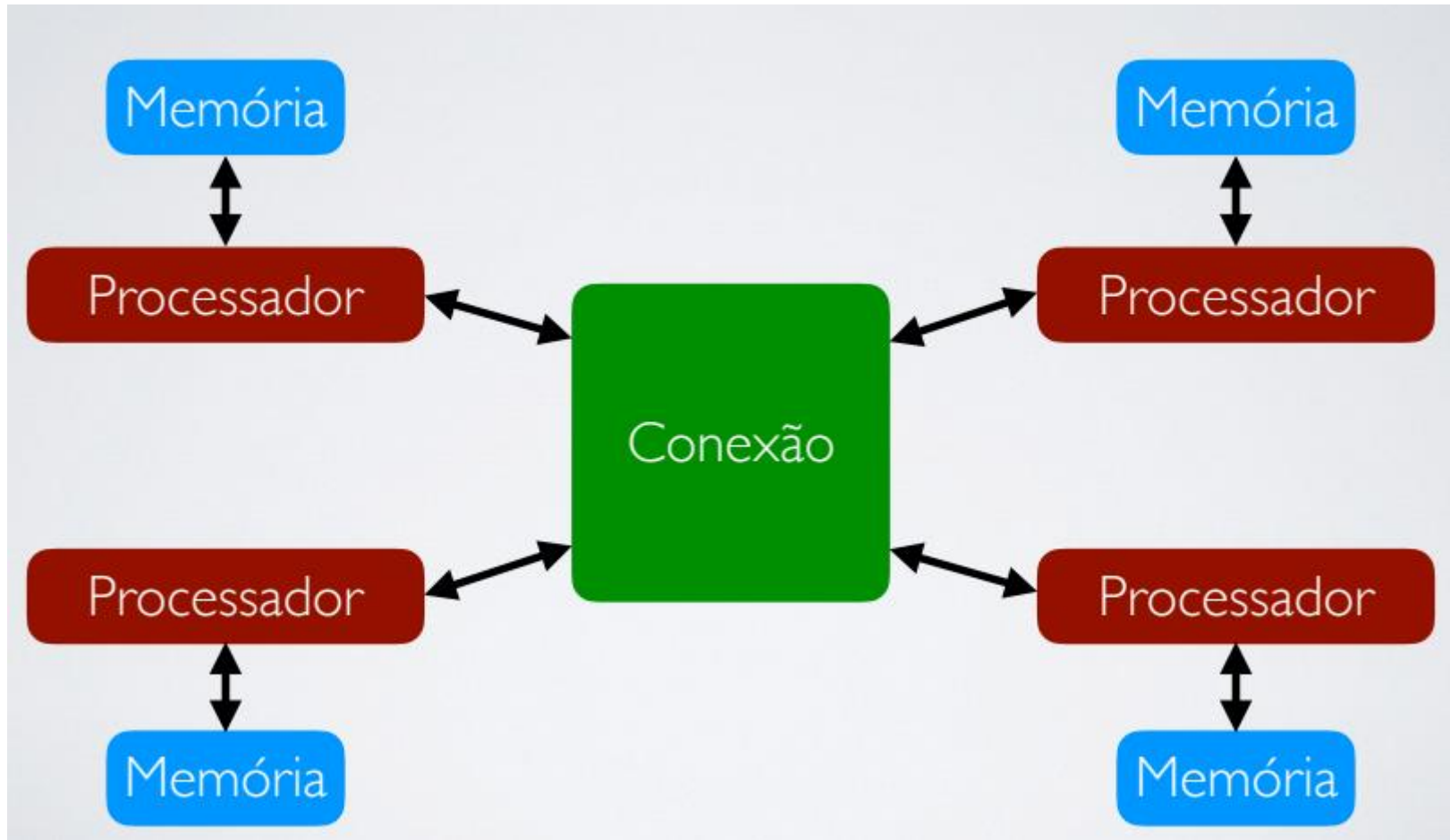
Multiprocessador :

Vários processadores acessam o mesmo espaço de memória.

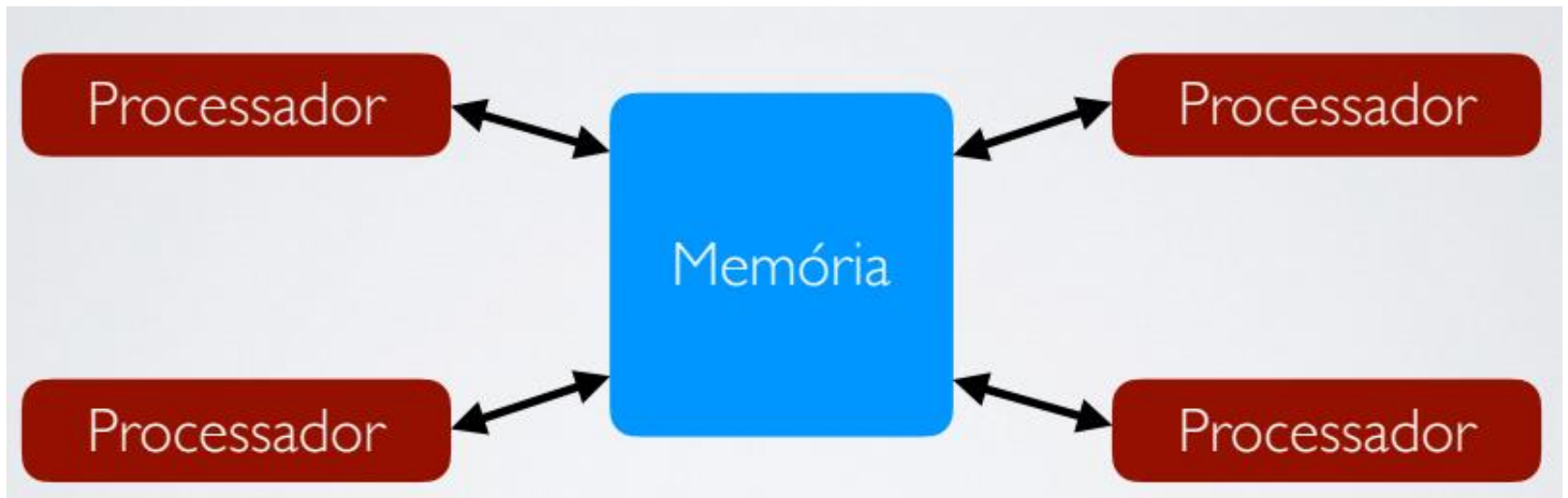
Multicomputador :

Vários processadores com memórias privadas acessa a mesma rede para troca de informações.

Multicomputador



Multiprocessador



Multiprocessadores

Homogêneo:

Todos os núcleos são iguais.

Heterogêneo:

Núcleos especializados em tarefas específicas .

Coprocessador

Coprocessador

- Um processador é desenvolvido seguindo uma arquitetura específica.
- Arquiteturas diferentes podem executar tarefas diferentes com maior desempenho.

Coprocessador

- Um processador pode ter coprocessadores para tarefas diferentes.
- Por exemplo, colocar um coprocessador de ponto flutuante no 8051?

Exemplos

- Coprocessador de ponto flutuante.
- Pentium MMX: coprocessador multimídia.
- GPU, placas de som, rede podem ser considerada coprocessador.

Um processador atual:

- Coprocessador para tarefas específicas.
- Vários núcleos.
- Que executam várias threads.
- Usam pipeline de vários estágios.
- E podem estar ligados a vários computadores.

Tipos de processadores

ASIC

Application Specific Integrated Circuits

- Circuitos especializados/otimizados em fazer uma tarefa em domínios bem definidos.

CPU

Unidade de Processamento Central

- 1 operação realizada em “1 dado” por vez.
- Paralelismo é realizado pela adição de estágios de pipeline, threads, adição de núcleos...

CPU

- Por serem sequenciais, demoram para fazer a mesma operação em diversos valores
- Permite fazer diversas operações em diversos valores

GPU

Unidade de Processamento Gráfico

- 1 operação realizada em N dados.
- Paralelismo com vários núcleos e dentro do núcleo são diferentes.

GPU

- Uma GPU com vários núcleos pode executar operações diferentes por núcleo, mas não dentro do núcleo.
- Bom para operações em vetores ou matrizes (imagens, redes neurais, etc)

GPU

Embarrassingly parallel: fácil para paralelizar e que os dados não dependem um do outro:

1. Renderização em computação gráfica.
2. Simulação de partículas sub-atômicas.
3. Métodos de Monte Carlo.
4. Redes Neurais Artificiais.

GPGPU

General-Purpose GPU

- Nome dado depois do uso genérico.
- Continuam processando da mesma forma, mas o foco é em deixar mais genérico sem perder desempenho.

CUDA

Compute Unified Device Architecture

- Núcleo CUDA é um processador simples dentro de um núcleo da GPU.
- Uma GPU (da Nvidia) é feita de vários núcleos CUDA.
- Processa inteiros e float de precisão simples

TPU

Unidade de Processamento de Tensores

- Produzida pelo Google para aumentar o desempenho de processamento de redes neurais.
- Não está disponível para a venda...
- Não confundir com os Tensor Cores da Nvidia!

TPU

- ASIC CISC em 700MHz com 24MB de SRAM.
- Quantização de ponto flutuante para inteiro de 8-bits.
- Primeira geração possui 65.536 8-bit Matrix Multiplier Unit (MXU).

Outros processadores

- Field-Programmable Gate Array (FPGA).
- Vision Processing Unit (VPU).
- AI Processing in Memory Architecture (APiM)

FPGA

Field Programmable Gate Array

- Circuitos programáveis para propósito geral, circuitos para aplicação específica (ASIC) ou mesmo circuitos não digitais podem ser utilizados para aumentar o desempenho ou a segurança.

FPGA

Field Programmable Gate Array



Exemplo de utilização da FPGA

Robô do time RoboFEI SSL



VPU - Vision Processing Unit

Intel comprou a Movidius:

- Myriad X
- Myriad 2
- Neural Compute Stick



APIM - AI Processing in Memory Architecture

Orange Pi AI Stick 2801



Exercícios de Fixação

1. Divide a execução da instrução em várias partes, cada uma manipulada por uma parte dedicada do hardware, sendo que todas essas partes podem ser executadas em paralelo. Assinale a alternativa correta.
 - a) Cluster.
 - b) pipeline.
 - c) multithread.
 - d) execução especulativa.
 - e) Thread.

Exercícios de Fixação

2. Com relação à hierarquia de memória em máquinas de vários núcleos (multicores), assinale a alternativa INCORRETA.
- a) Os núcleos podem possuir memória cache de alto nível (L1) internamente e compartilhar memórias cache de menor nível (L2 ou L3).
 - b) Os processadores atuais possuem vários núcleos e possuem vários estágios de pipeline.
 - c) As sincronizações das memórias cache acabam piorando o desempenho do processo paralelo.
 - d) Caches de diferentes núcleos obrigatoriamente trocam mensagens entre si com o intuito de trocarem informações.

Exercícios de Fixação

3. De acordo com a definição de sistemas paralelos, afirmativas seguintes,

I. É possível organizar o hardware em sistemas com várias CPUs através de vários processadores tipicamente homogêneos e localizados em um mesmo computador.

II. Sistemas paralelos multicomputadores dependem de uma arquitetura onde cada processador possui sua própria memória local.

III. Sistemas paralelos multiprocessadores podem compartilhar memória de forma homogênea ou heterogênea.

IV. Não é possível obter uma arquitetura que apresente multiprocessadores em barramento.

verifica-se que está(ão) correta(s)

- a) I, II, III e IV.
- b) I, apenas.
- c) II, III e IV, apenas.
- d) II e III, apenas.
- e) I, II e III, apenas.

Respostas

Resposta dos exercícios:

1. RISC
2. b
3. b
4. d
5. d
6. F V F V
7. c

Paralelismo

- 1 - b
- 2 - d
- 3 - e

Bibliografia

Stallings, Willian. Arquitetura e Organização de Computadores. 10ª Ed, Pearson, 2017.

Murdocca, Miles J., and Vincent P. Heuring. Introdução à arquitetura de computadores. Elsevier, 2001.

David A.Patterson & John Hennessy. Organização e projeto de computadores: A interface de Hardware e Software. 4ª Ed. Elsevier. 2014.

ZELENOVSKY, R.; MENDONÇA, A. Microcontroladores Programação e Projeto com a Família 8051. MZ Editora, RJ, 2005.

Gimenez, Salvador P. Microcontroladores 8051 - Teoria e Prática, Editora Érica, 2010.