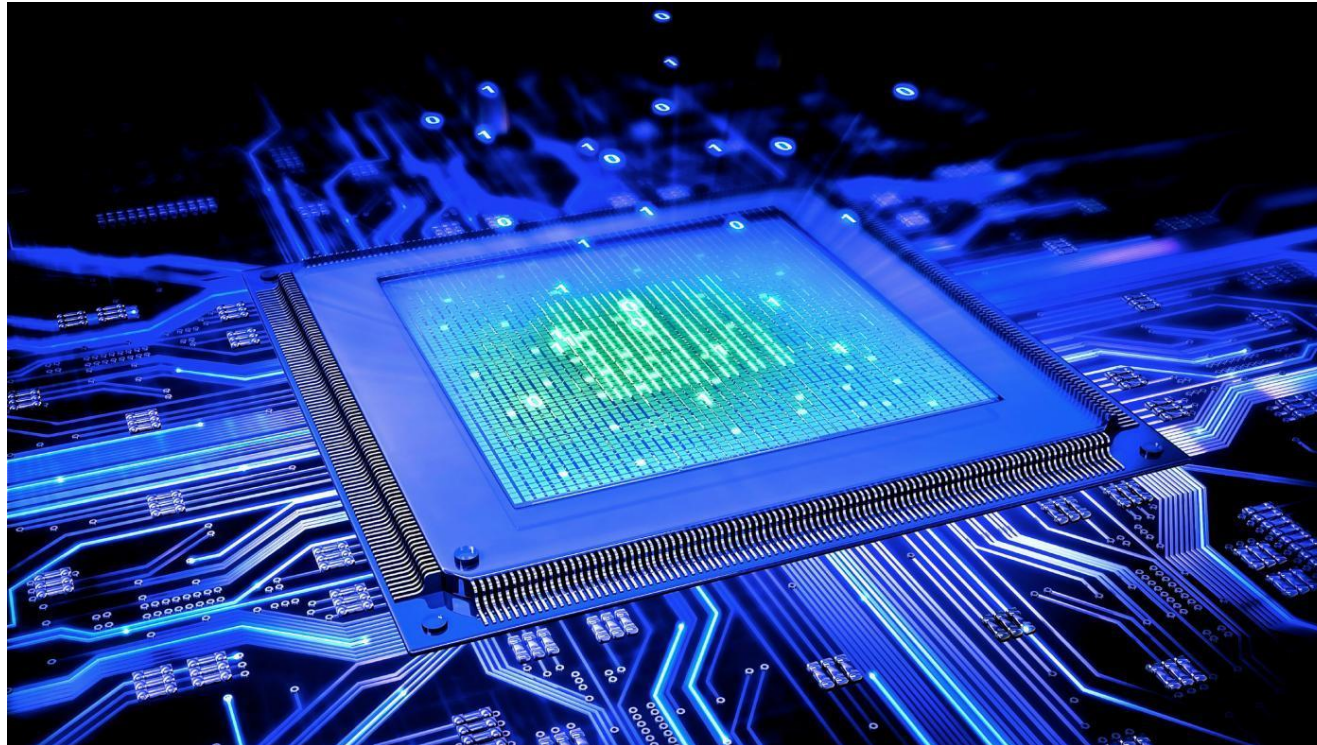


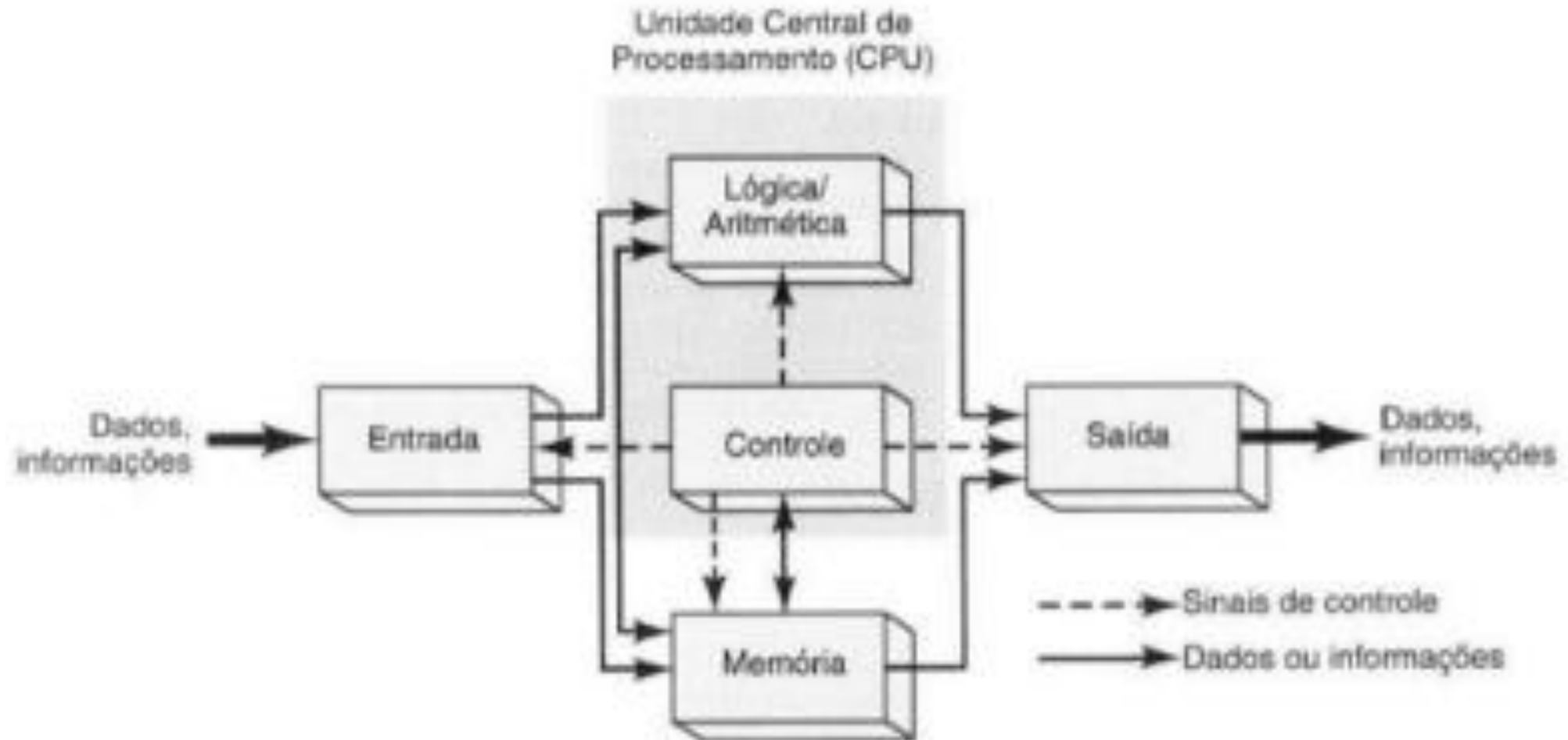
# Arquiteturas CISC e RISC

## Recursos Computacionais

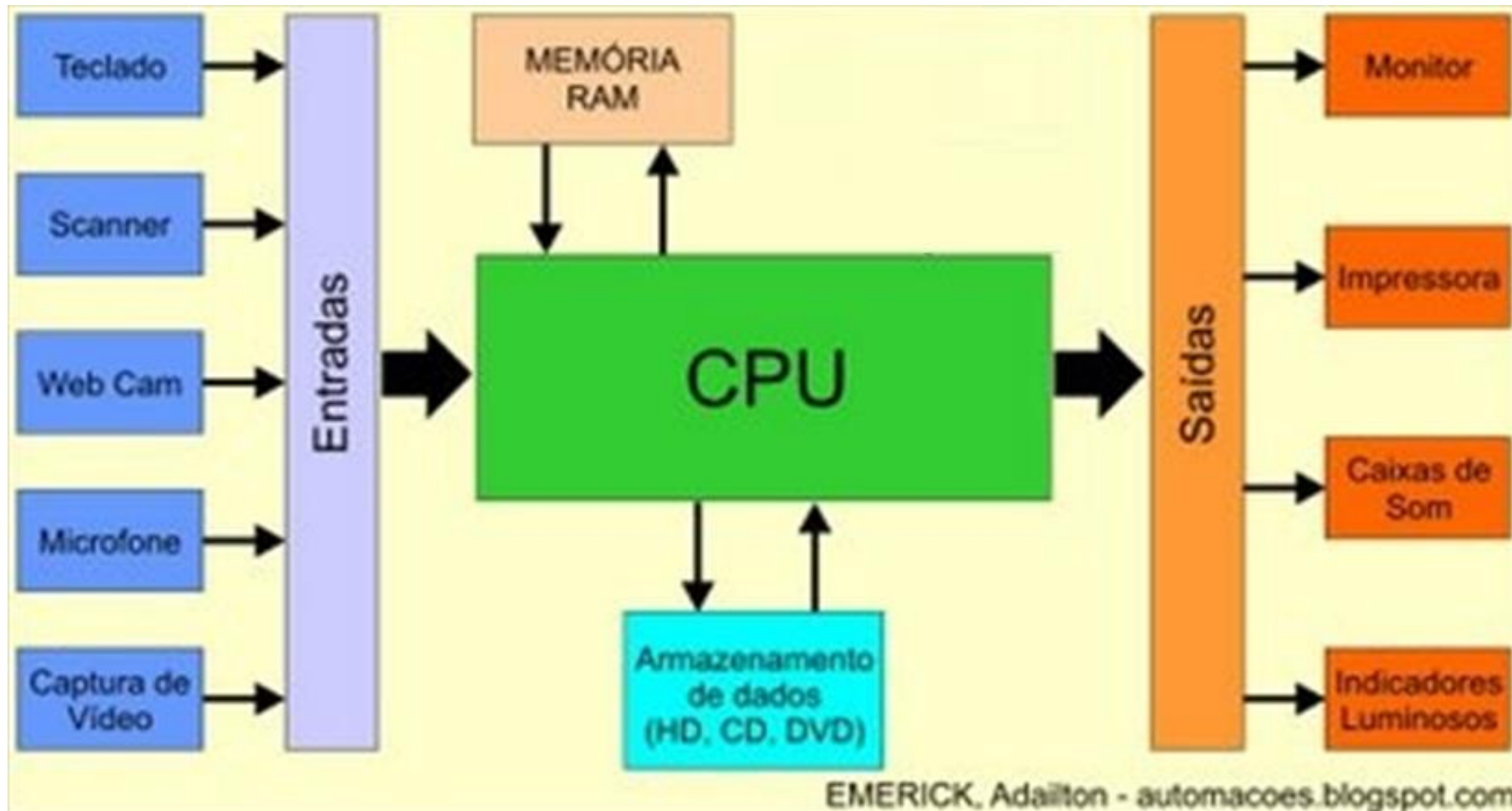
# Unidade Central de Processamento



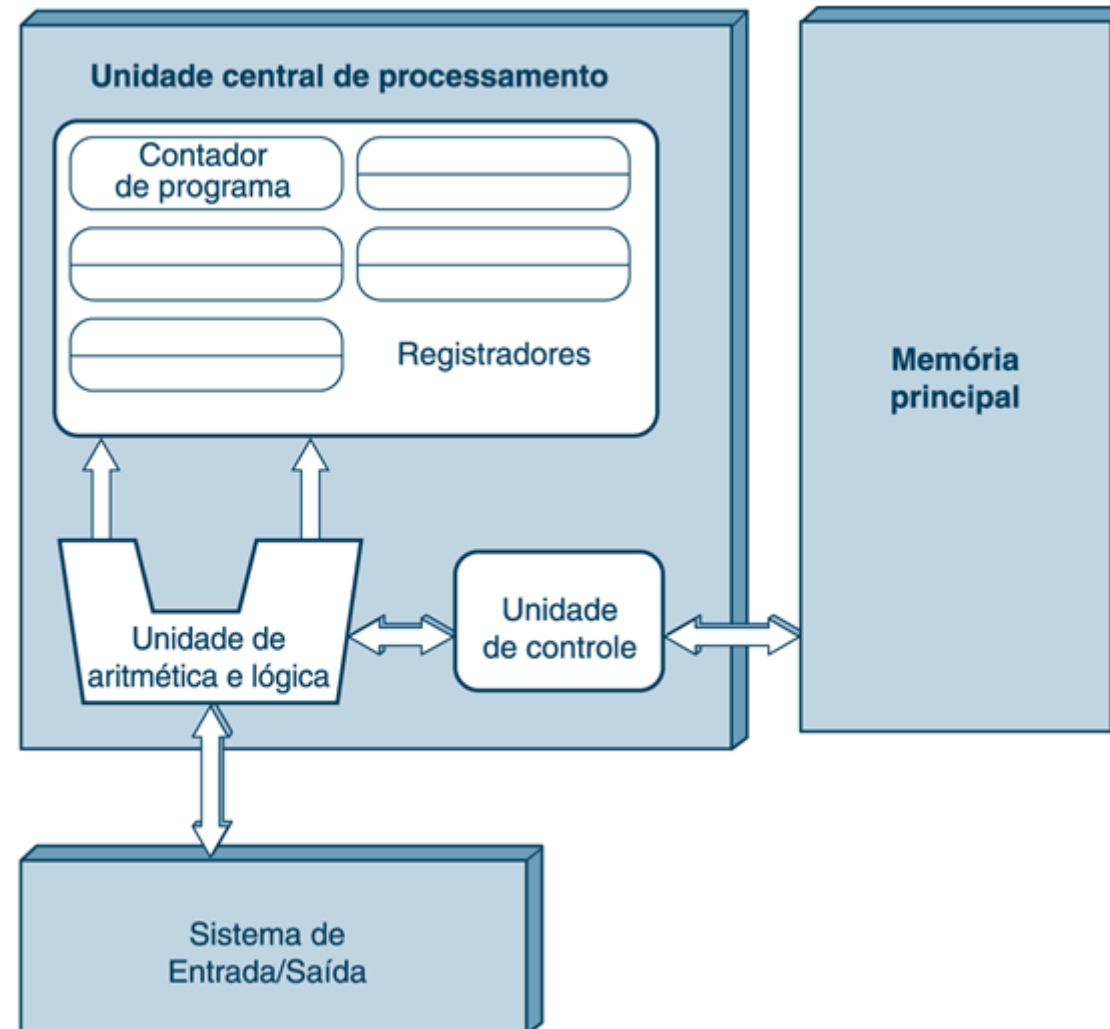
## Arquitetura Von Neumann (recordando...)



# Unidade Central de Processamento



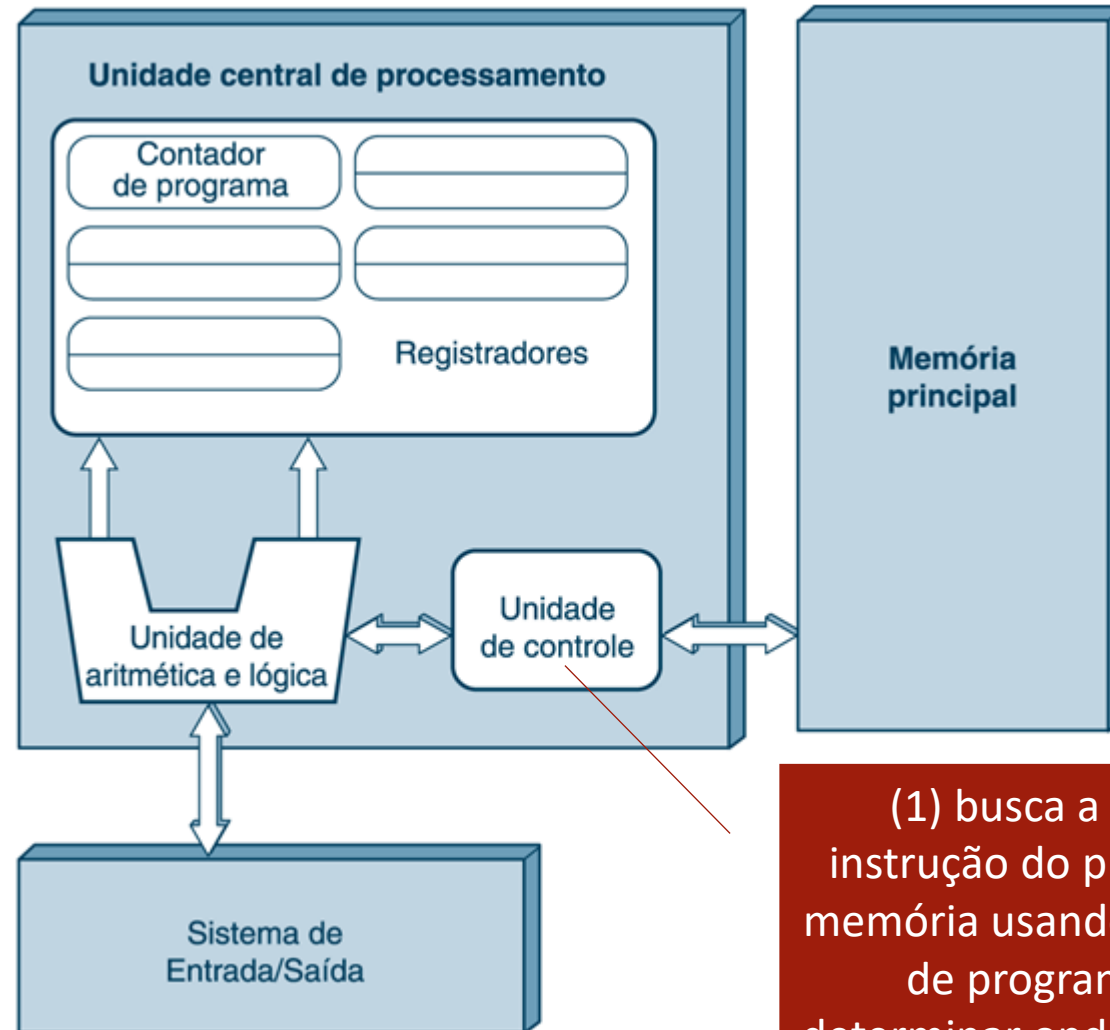
# Unidade Central de Processamento



# Unidade Central de Processamento

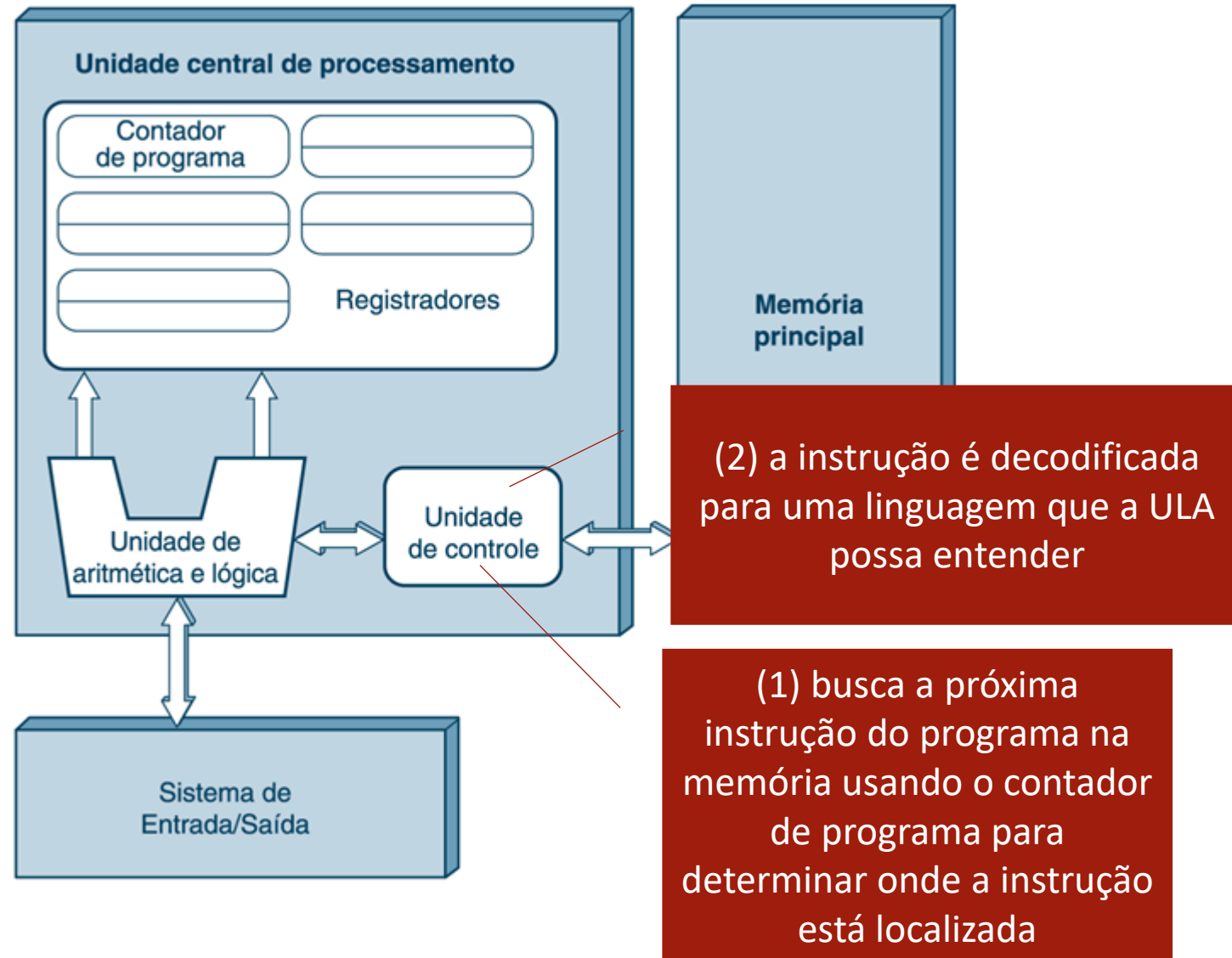
- **Unidade de Controle (UC):** busca instruções na MP e determina o tipo de cada instrução.
- **Unidade Lógica e Aritmética (ULA):** realiza um conjunto de operações necessárias à execução das instruções.
- **Conjunto (Banco) de Registradores:** pequena memória de alta velocidade, usada para armazenar resultados temporários e certas informações de controle.

# CPU



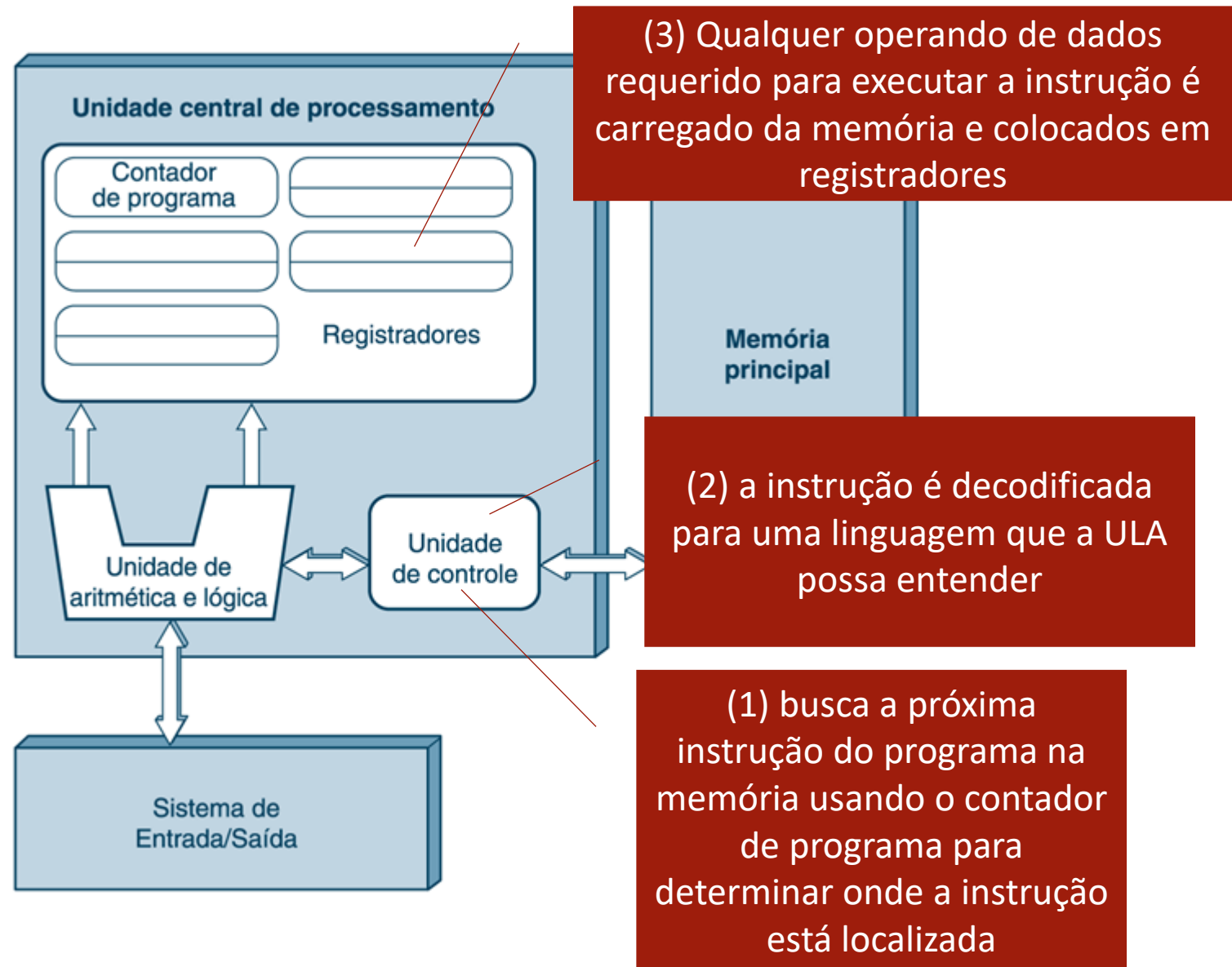
(1) busca a próxima instrução do programa na memória usando o contador de programa para determinar onde a instrução está localizada

# CPU

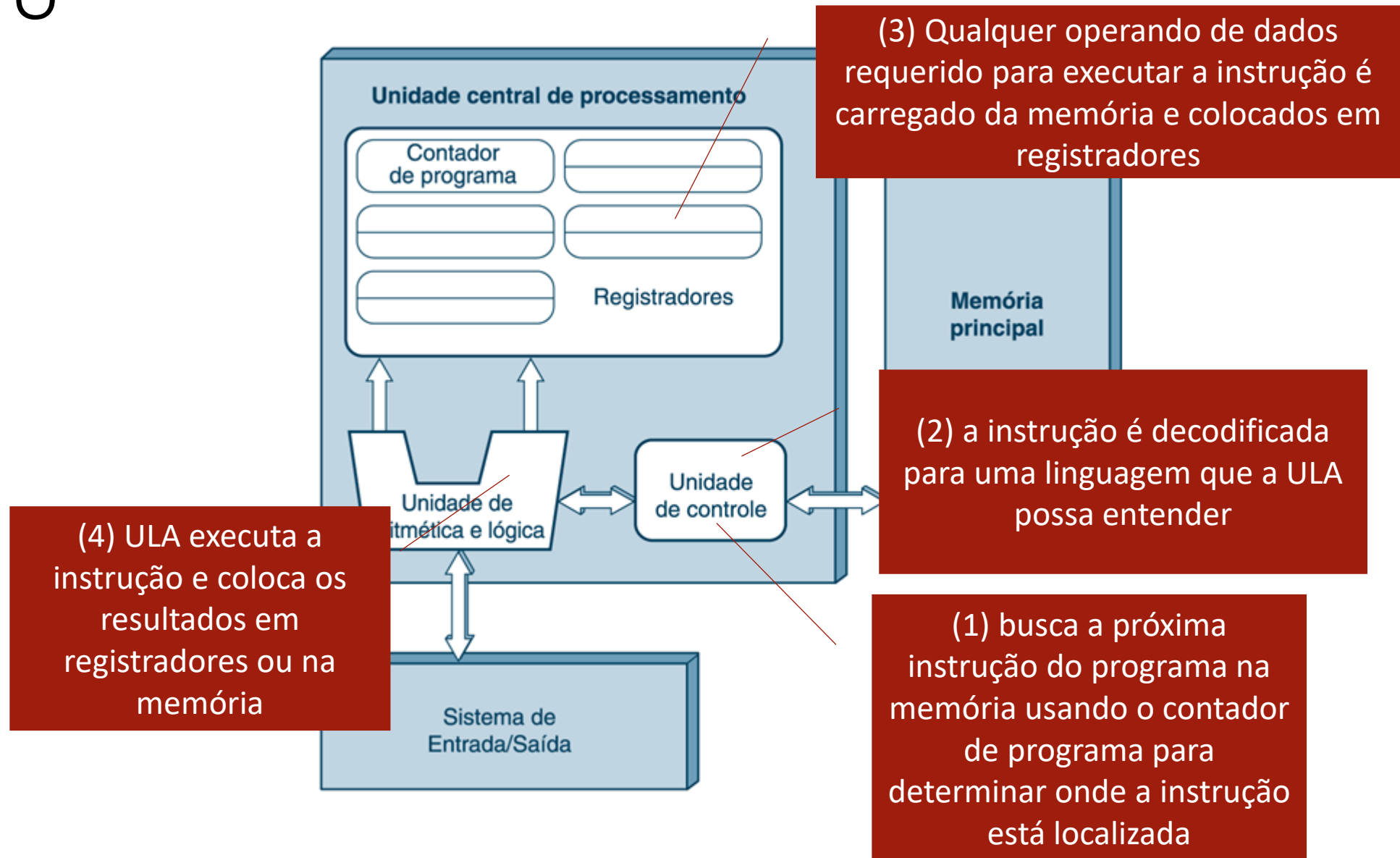




# CPU



# CPU



Uma Unidade Central de Processamento (UCP) é composta por duas partes principais:

- ULA (Unidade Lógica e Aritmética): executa as instruções realizando as operações lógicas e aritméticas;
  - UC (Unidade de Controle): tem por funções a busca, interpretação e controle de execuções das instruções, e o controle dos demais componentes do computador.
- 
- A UCP possui memórias com diversas finalidades:
  - O Registrador de Instruções (RI) armazena tudo o que será processado;
  - O Registrador de Endereços de Memória (REM) armazena os endereços dos dados que serão processados;
  - O Registrador de Dados de Memória (RDM) grava o que será processado;
  - O ACC (Acumulador) é uma memória que antecede a ULA e armazena tudo o que entra e sai da mesma.

# Formato das Instruções

As instruções possuem tamanhos limitados e normalmente seguem o tamanho máximo do Acumulador, registrador que antecede a ULA. Tais instruções são divididas em:

- **OPERANDO:** é o campo da instrução cujo valor sinaliza a localização do dado que será manipulado. Em geral, um operando identifica o endereço de memória onde está contido o dado que será manipulado, ou pode conter o endereço onde o resultado da operação será armazenado (Os operandos fornecem os dados da instrução).
- **OPCODE (código de operação):** identifica a operação a ser realizada pelo processador. (O número de bits do código de operação depende de quantas instruções tem o set de instruções da máquina).

# Ciclo de Instrução

Pode-se resumir o ciclo de instrução em estados de instrução:

- Busca do Opcode: Lê a instrução e armazena em uma área de armazenamento;
- Decodificação do Opcode: Determina o código de operação da instrução e as referências (endereços) a operandos;
- Cálculo de Operando: Determina o endereço efetivo de cada operando fonte;
- Busca de Operando: Busca cada operando localizado na memória;
- Execução da Instrução: Efetua a operação indicada e armazena o resultado;
- Escrita do Resultado: Armazena o resultado na memória.

# Arquiteturas CISC e RISC

- **RISC (*Reduced Instruction Set Computer*) (1990)**

"Computador com um conjunto reduzido de instruções" - é uma linha de arquitetura de processadores que favorece um conjunto simples e pequeno de instruções que levam aproximadamente a mesma quantidade de tempo para serem executadas.

AMD – OPTERON

INTEL – XEON

SUN – Spark

IBM - Power

- **CISC (*Complex Instruction Set Computer*) (1960)**

"Computador com um conjunto complexo de instruções" - possui um conjunto maior de instruções especializadas, algumas delas raramente usadas pelos programas.

# Arquiteturas CISC e RISC

CISC (Bloco de Instruções Complexo)	RISC (Bloco de Instruções Reduzido)
Máquina Base	Máquina Escalar
Instruções de tamanho variado Maior quantidade de modos de endereçamento	Instrução de tamanho único (fixo) /mais compactas Menos modos de endereçamento
Maior nº de transistores (esquenta mais)	Maior memória cache Poucos Registros Uso de Threads
Uso de Microprograma (Kernel)	Uso de microcódigo (hardware - Conjunto de Instruções – SSE/AVX)
<u>Uma</u> instrução por etapa de ciclo	<u>Duas ou +</u> instruções por etapa de ciclo

Arquitetura Híbrida: processadores com parte CISC e parte RISC (CRISC).

## 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., *add R1, R2, R3*)

Um ou dois operandos de registradores permitidos por instrução (por ex., *add R1, R2*)

Passagem eficiente de parâmetros por registradores no chip (processador)

Passagem de parâmetros ineficiente através da memória

Instruções de um único ciclo (ex. *load* e *store*)

Instruções de múltiplos ciclos

Controle *hardwired* (embutido no **hardware**)

Controle microprogramado

Altamente paralelizado (*pipelined*)

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

Apenas instruções *load* e *store* podem acessar a memória

Muitas instruções podem acessar a memória

Poucos modos de endereçamento

Muitos modos de endereçamento



# Vantagens

O padrão RISC do processador permite que estes processadores tenham menos transístores que processadores CISC (x86).

Essa abordagem reduz custos, liberação de calor e consumo de energia.

Essas são características desejáveis para dispositivos portáteis, como smartphones, laptops, tablets e outros dispositivos embarcados.

Uma estrutura mais simples facilita a criação de multi-core CPUs, o que impacta na redução de custos de produção.

Os processadores ARM são 90% dos processadores embarcados RISC de 32 bits ou 64 bits.

<https://www.youtube.com/watch?v=g2cbAlDwY64>

# CISC

## Máquina Base

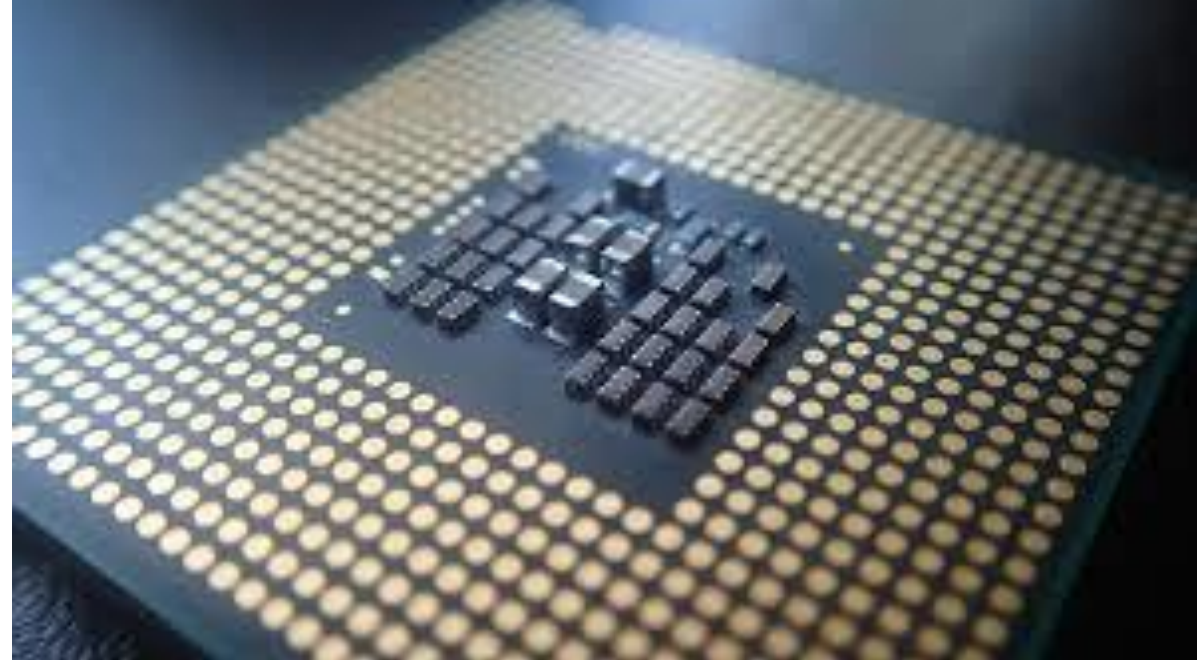
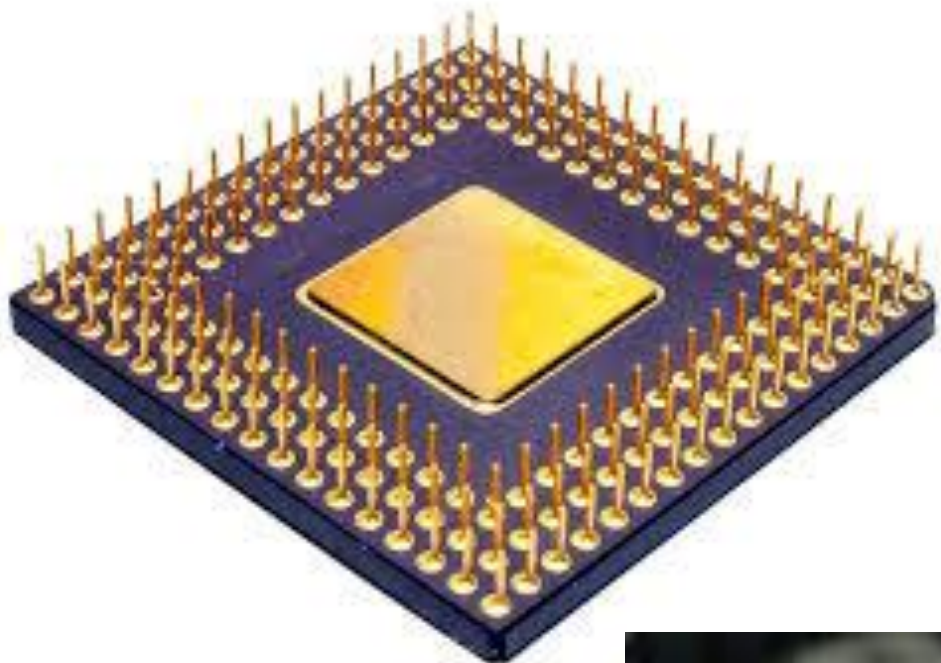
[illegible]

# RISC

## Máquina Escalar

	B	D	EX	ES
1	A			
	B			
2	C	A		
	D	B		
3	E	C	A	
	F	D	B	
4				A
				B

[illegible]



# Arquitetura ARM

**ARM**, originalmente *Acorn RISC Machine*, e depois *Advanced RISC Machine*

- Perfil-A, o perfil "Aplicativo", implementado por núcleos de 32 bits na série Cortex-A e por alguns núcleos não-ARM
- Perfil-R, o perfil "Real-time", implementado por núcleos da série Cortex-R
- Perfil-M, o perfil "Microcontrolador", implementado pela maioria dos núcleos da série Cortex-M



# Séries

CORTEX-A	Cortex-A72	Processadores para aplicações de alta performance
	Cortex-A57	
	Cortex-A53	
	Cortex-A17	
	Cortex-A15	
	Cortex-A9	
	Cortex-A7	
	Cortex-A5	
CORTEX-R	Cortex-R7	Processadores para sistemas em tempo real
	Cortex-R5	
	Cortex-R4	
CORTEX-M	Cortex-M7	Processador para micro controladores
	Cortex-M4	
	Cortex-M3	
	Cortex-M1	
	Cortex-M0+	
	Cortex-M0	
SECURCORE	SC000	Processadores para aplicações que necessitam de segurança adicional
	SC100	
	SC300	

# Modos da CPU

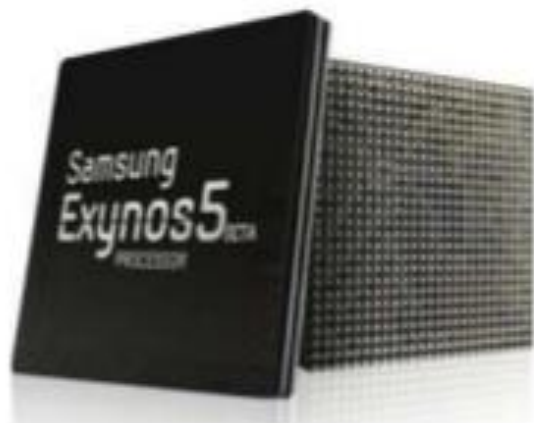
- *Modo Usuário:* O único modo não privilegiado
- *Modo FIQ:* Um modo privilegiado que é inserido sempre que o processador aceita uma solicitação de interrupção rápida (Fast Interrupt Requests)
- *Modo IRQ:* Um modo privilegiado que é inserido sempre que o processador aceita uma interrupção.
- *Modo Supervisor (SVC):* Um modo privilegiado inserido sempre que a CPU é reinicializada ou quando uma instrução SVC é executada.
- *Modo de anulação:* Um modo privilegiado que é inserido sempre que ocorre uma exceção de anulação automática ou anulação de dados.
- *Modo indefinido:* Um modo privilegiado que é inserido sempre que ocorre uma exceção de instrução indefinida.
- *Modo Sistema (ARMv4 e acima):* O único modo privilegiado que não é inserido por uma exceção. Ele só pode ser inserido executando uma instrução que grava explicitamente os bits de modo do CPSR (Current Program Status Register) de outro modo privilegiado (não do modo de usuário).



# Modos da CPU

- *Modo monitor (ARMv6 e ARMv7 Security Extensions, ARMv8 EL3):* Um modo monitor é introduzido para suportar a extensão TrustZone em núcleos ARM.
- *Modo Hyp (ARMv7 Virtualization Extensions, ARMv8 EL2):* Um modo de hipervisor que suporta os requisitos de virtualização de Popek e Goldberg para a operação não segura da CPU.
- *Modo thread (ARMv6-M, ARMv7-M, ARMv8-M):* Um modo que pode ser especificado como privilegiado ou não privilegiado, enquanto o MSP (Main Stack Pointer) ou PSP (Process Stack Pointer) é usado, também pode ser especificado em Registro de controle com acesso privilegiado. Este modo é projetado para tarefas do usuário em ambiente RTOS, mas é tipicamente usado em bare-metal para super loop.
- *Modo do manipulador (ARMv6-M, ARMv7-M, ARMv8-M):* Um modo dedicado para tratamento de exceções (exceto o RESET que é tratado no modo Thread). O modo manipulador sempre usa o MSP e funciona em nível privilegiado.

## SoCs baseados na arquitetura ARM



Exynos - Samsung



Ax - Apple



Mediatek - Mediatek



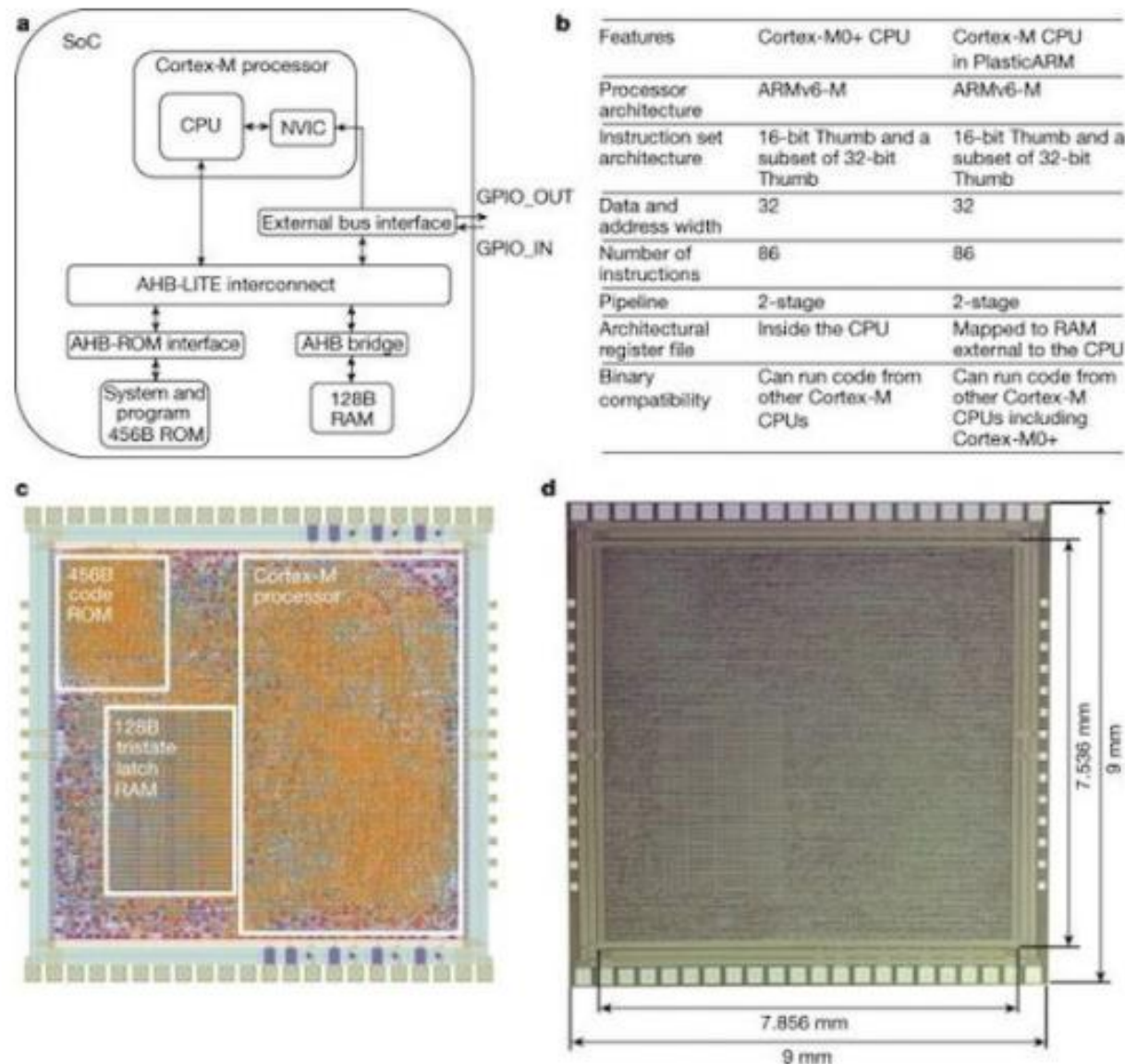
Snapdragon - Qualcomm



Tegra - Nvidia



# PlasticARM da ARM, um microprocessador flexível feito de plástico em vez de silício



PlasticARM da ARM, um microprocessador flexível feito de plástico em vez de silício

# Exemplos de aplicação





ARM	x86
Baixo consumo de energia	Alto consumo de energia
Conjunto de instruções reduzidas(RISC)	Instruções complexas (CISC)
32 e 64bits	32 e 64bits (x86-64 ou amd64)
Registrador - Registrador	Registrador - Memória

# Recursos Computacionais

- Análise do fluxo de dados: O processador analise a fila de instruções.
- Execução Especulativa: Através da Análise de Fluxo de Dados o Processador consegue realizar a decodificação antecipada.
- Previsão de Desvio: Utiliza a técnica de análise de fluxo de dados para encontrar uma dependência de dados verdadeira para criar um desvio, evitando que o processador fique ocioso.
  - Estática: avalia todo o conteúdo da memória cache e histórico de processamento.
  - Dinâmica: analisa toda fila de processamento (memórias cache e RAM) além do histórico de processamento.
- Princípio Estatístico Comprovado:
  - Análise estática
  - Análise dinâmica: exige processamento
- Pipeline: recurso computacional que acelera o processamento em até 1,5 vezes. Aumenta significativamente a eficiência de um processador.

\*\* Pipeline Vetorial: processa instruções vetoriais. Sua característica é a sua unidade escalar pipeline típica e uma unidade vetorial que realizam operações de alto nível sobre arrays lineares.

DDV (Dependência de Dados Verdadeira): depende de uma ação anterior para poder realizar a instrução seguinte.

Ex:

$$A=1+2$$

$$B=A+3$$

Quando uma informação depende do resultado da outra, o processador faz um desvio para não ficar ocioso.

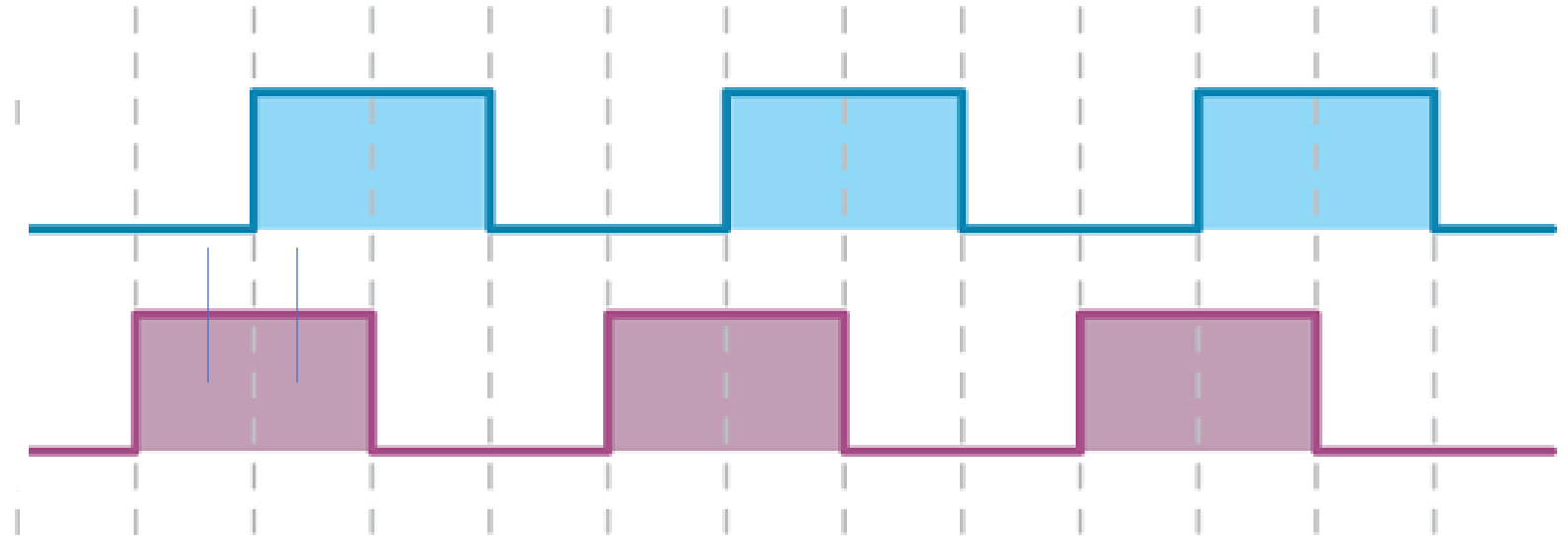
DDF (Dependência de Dados Falsa)

$$A=1+2$$

$$Z=A+4$$

CLOCK B

CLOCK A



## CISC Máquina Base

	Busca	Decod	Execuç	Escrita
1	A			
2	B	A		
3	C	B	A	
4	D	C	B	A

## RISC Máquina Escalar

	B	D	EX	ES
1	A			
	B			
2	C	A		
	D	B		
3	E	C	A	
	F	D	B	
				A
				B

## Pipeline

	B	D	EX	ES
1	A			
	/Bb			
2		A		
		Bb/Bd		
			A	
			Bd/Bex	
				A
				Bex/Bes
1	A			
	B			
	/Cb			
	/Db			

10. Calcule os períodos de clock (a-c) e as frequências (d-f) de operação dos seguintes processadores:

a) 700 MHz

b) 1.6 GHz

c) 2.8 GHz

d) 1.6 ns

e) 0.8 ns

f) 0.2 ns

a)  $F=1/T$                        $T=1/0,7 \cdot 10^9$      $T= 1,42 \cdot 10^{-9}$                        $T= 1,42 \text{ ns}$

b)  $T=1/1,6 \cdot 10^9$                        $T= 0,625 \cdot 10^{-9}$                        $T=0,625 \text{ ns}$

e)  $F=1/0,8 \cdot 10^{-9}$                        $T=1,25 \cdot 10^9$      $T= 1,25 \text{ GHz}$

$F=1/T$

$T=1/5 \cdot 10^9 \text{ Hz}$      $T=0,2 \cdot 10^{-9} \text{ s}$      $T=0,2 \text{ ns}$