

# Organização e Arquitetura de Computadores

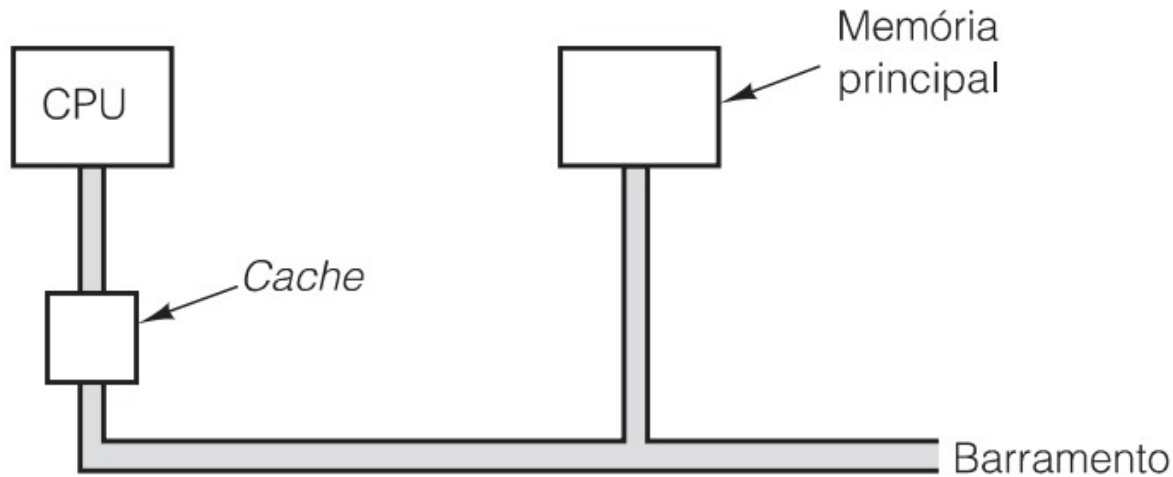
---

Julio Cesar Goldner  
Vendramini

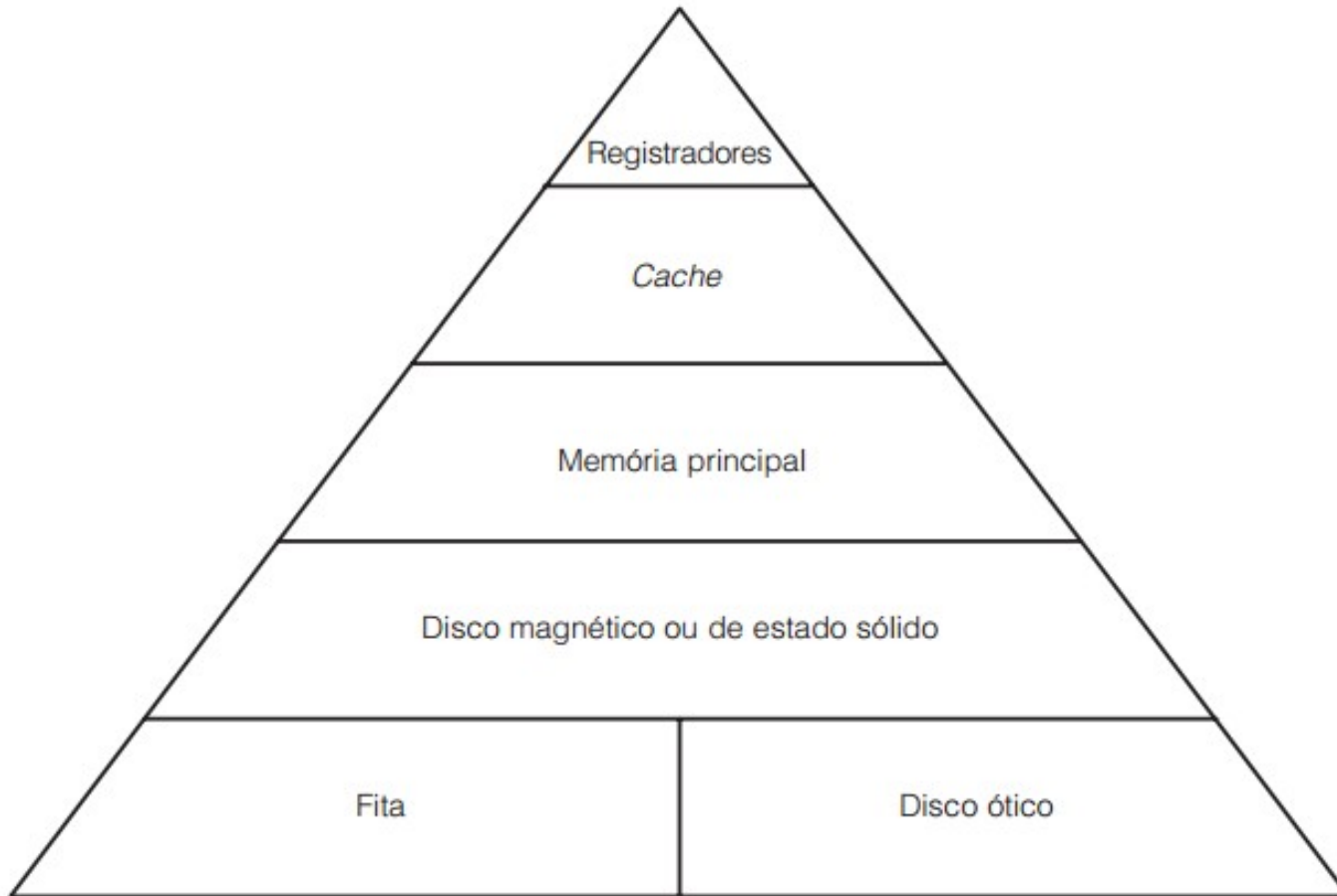
# Memória Cache

Julio Cesar Goldner  
Vendramini

A localização lógica da *cache* é entre a CPU e a memória principal. Em termos físicos, há diversos lugares em que ela poderia estar localizada.



# Hierarquia de memória de cinco níveis

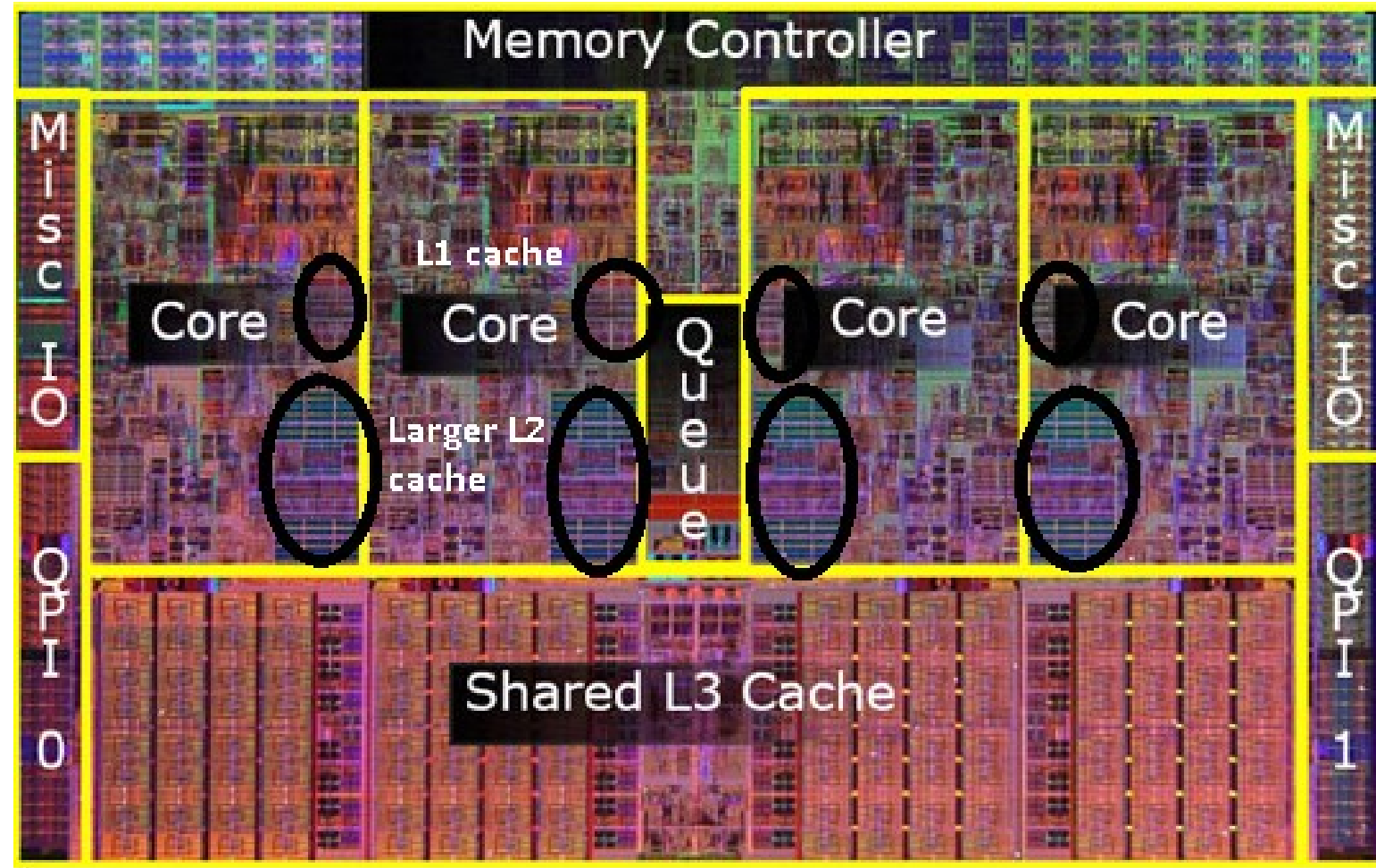


## O que é?

- É um tipo de memória SRAM muito rápida que fica entre o processador e a memória RAM.
- Sua latência é muito baixa comparada a memória Ram
- E sua taxa de transferência também é muito maior

## Como é dividida?




- A memória cache atualmente é dividida em 3
- L1 ~64KB por núcleo (dividida em duas – Cache pra dados e pra instruções)
- L2 ~256KB por núcleo
- L3 - 2MB até 64MB (compartilhado entre os núcleos)




- Custo de acesso em ciclos de clock do processador:
- L1: 4 ciclos
  - L2: 11 ciclos
  - L3: 39 ciclos
  - Memória RAM: 107 ciclos

Fonte: <https://medium.com/software-design/why-software-developers-should-care-about-cpu-caches-8da04355bb8a>

# AIDA64 Cache & Memory Benchmark

	Read 	Write 	Copy 	Latency
Memory	39803 MB/s	39256 MB/s	34922 MB/s	93.3 ns
L1 Cache	421.86 GB/s	211.54 GB/s	422.09 GB/s	1.2 ns
L2 Cache	330.50 GB/s	208.79 GB/s	354.89 GB/s	5.0 ns
L3 Cache	230.36 GB/s	188.19 GB/s	237.57 GB/s	13.7 ns
CPU Type	QuadCore AMD Ryzen 5 1600 [Summit Ridge, Socket AM4]			
CPU Stepping	ZP-B1			
CPU Clock	3392.8 MHz			
CPU FSB	99.8 MHz [original: 100 MHz]			
CPU Multiplier	34x	North Bridge Clock		1330.5 MHz
Memory Bus	1330.5 MHz	DRAM:FSB Ratio		40:3
Memory Type	Dual Channel DDR4-2667 SDRAM [16-18-18-38 CR1]			
Chipset	AMD B350, AMD Taishan, AMD K17 IMC			
Motherboard	Gigabyte GA-AB350M-D53H V2			
BIOS Version	F30 [AGESA: Combo-AM4 0.0.7.2]			



 CPU-Z

CPU

Caches

Mainboard

Memory

SPD

Graphics

Bench

About

Processor

Name

AMD Ryzen 5 1600

Code Name

Summit Ridge

Max TDP

65.0 W

Package


Socket AM4 (1331)

Technology

14 nm

Core Voltage

0.912 V



Specification

AMD Ryzen 5 1600 Six-Core Processor

Family

F

Model

1

Stepping

1

Ext. Family

17

Ext. Model

1

Revision

ZP-B1

Instructions

MMX(+), SSE, SSE2, SSE3, SSE4.1, SSE4.2, SSE4A, x86-64, AMD-V, AES, AVX, AVX2, FMA3, SHA

Clocks (Core #0)

Core Speed

1546.31 MHz

Multiplier

x 15.5

Bus Speed

99.76 MHz

Rated FSB

Cache

L1 Data

4 x 32 KBytes

8-way

L1 Inst.

4 x 64 KBytes

4-way

Level 2

4 x 512 KBytes

8-way

Level 3

2 x 8 MBytes

16-way

Selection

Socket #1

Cores

4

Threads

4

CPU-Z

Ver. 1.92.0.x64

Tools

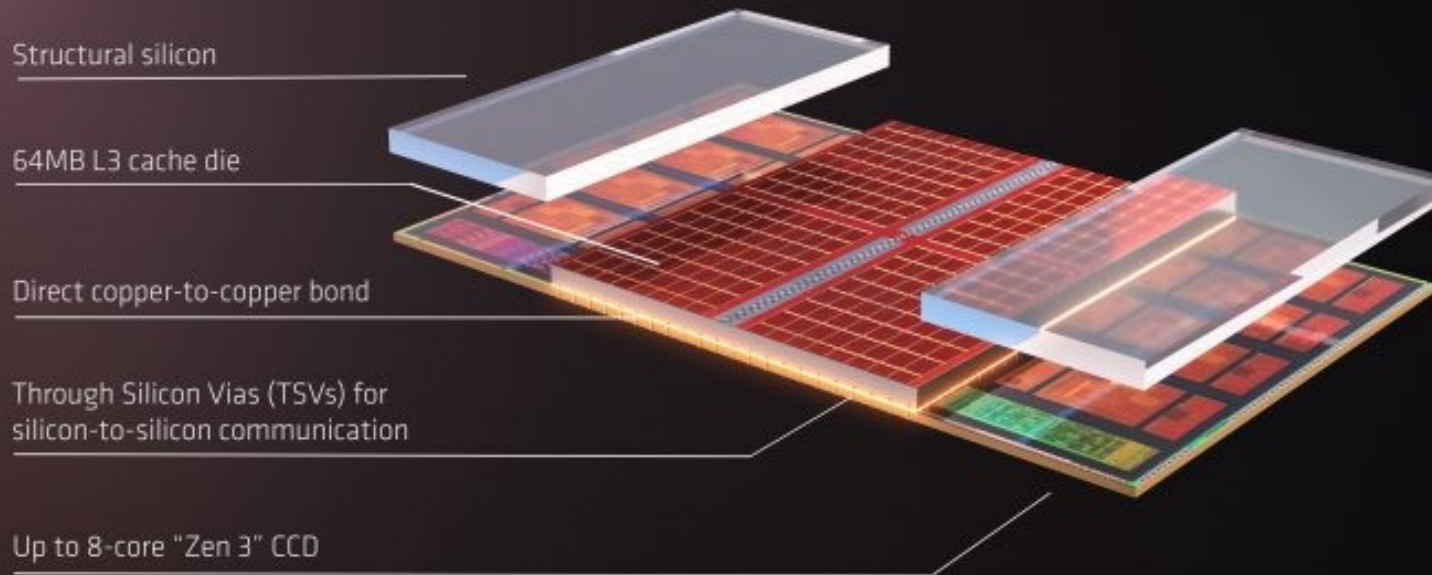
Validate

Close

Foto em alta qualidade do die do Ryzen série 5000

<https://wccftech.com/amd-ryzen-5000-zen-3-vermeer-undressed-high-res-die-shots-close-ups-pictured-detailed/>

# AMD 3D CHIPLET TECHNOLOGY



**A PACKAGING BREAKTHROUGH FOR HIGH-PERFORMANCE COMPUTING**

AMD 3D V-CACHE PROTOTYPE PICTURED

# 3D V-CACHE TECHNOLOGY

## PERFORMANCE UPLIFT ACROSS GAMING



15% FASTER GAMING ON AVERAGE

TUTO FEDERAL

■ ■ Espírito Santo

Fonte: <https://gadgettendency.com/imagine-a-ryzen-9-5950x-with-192mb-of-l3-cache-and-showed-an-easy-way-to-increase-cache-for-their-processors/>

## Como a memória cache funciona?

- Por nossa sorte, a execução e consequentemente o carregamento das instruções dos programas é em sua maioria sequencial ou repetitiva.
  - Isso não acontece com os desvios e chamadas de procedimentos(funções)
- Com isso, a ideia da memória cache é carregar os valores dos próximos endereços de memória, pois geralmente é o que é utilizado (**Localidade espacial**). Com isso, não precisamos esperar o tempo de acesso da memória RAM.
- Na repetição de código(loop), reutilizamos várias vezes as mesmas instruções e dados(**Localidade temporal**).

# Técnicas pra ajudar no desempenho da memória RAM (paralelo e sequencial)

- Pré busca ajudar diminuir o tempo de acesso

Tecnologia	Tamanho da Pré-busca
DDR	2
DDR2	4
DDR3	8
DDR4	8
DDR5	16

- Largura em paralelo:
  - Cada módulo de memória possui largura de 64 bits
  - Existem atualmente 3 modos configuração de largura de memória pelos processadores:
    - Single Chanel (64 bits)
    - Dual Channel (128 bits)
    - Quad Channel (256 bits)

## Exemplo:

- Cycles por instrução (CPI) = 2
- Penalidade (acesso a memória) = 40 ciclos(clocks)
- Taxa de instruções que acessam a memória (Load , Store) = 36%
- Qual a degradação de desempenho devido aos acessos à memória?

## Exemplo:

- Qual a degradação de desempenho devido aos acessos à memória? (Vamos supor a falta de cache no sistema primeiro:)
- Supondo um programa com 1000 instruções;
  - Possui  $1000 \times 36\%$  de instruções LOAD/STORE = 360
  - Teremos 640 instruções de acesso a registradores (execução rápida)
- Qual tempo total em ciclos?
  - $640 \times 2(\text{CPI}) + 360 \times 40(\text{penalidade}) = 1280 + 14400 = 15680$  Ciclos



## Exemplo (melhorando a arquitetura do processador):

- $CPI = 1$
- Qual a degradação de desempenho devido aos acessos à memória? (Vamos supor a falta de cache no sistema primeiro:)
- Supondo um programa com 1000 instruções;
  - Possui  $1000 \times 36\%$  de instruções LOAD/STORE = 360
  - Teremos 640 instruções de acesso a registradores (execução rápida)
- Qual tempo total em ciclos?
  - $640 \times 1(CPI) + 360 \times 40(\text{penalidade}) = 640 + 14400 = 15040$  Ciclos
- Ganho de desempenho entre os dois processadores :
- $(15680 - 15040) / 15680 = 4\%$  (4% mais rápido apenas)

## Exemplo:

- Taxa de erro: 5%
- Cycles por instrução (CPI) = 2
- Penalidade (acesso a memória) = 40 ciclos(clocks)
- Penalidade (acesso a memória cache) = 4 ciclos (clocks)
- Taxa de instruções que acessam a memória (Load , Store) = 36%
- Qual a degradação de desempenho devido aos acessos à memória?

## Exemplo:

- Qual a degradação de desempenho devido aos acessos à memória? (Agora possuímos cache com taxa de erro de 5%)
- Supondo um programa com 1000 instruções;
  - Possui  $1000 \times 36\%$  de instruções LOAD/STORE = 360
  - Teremos 640 instruções de acesso a registradores (execução rápida)
- Qual tempo total em ciclos?
  - $640 \times 2(\text{CPI}) + 360 \times 95\% \times 4(\text{penalidade cache}) + 360 \times 5\% \times 40(\text{penalidade memória})$
  - $1280 + 1368 + 720 = 3368$  ciclos

# Como a memória cache funciona?

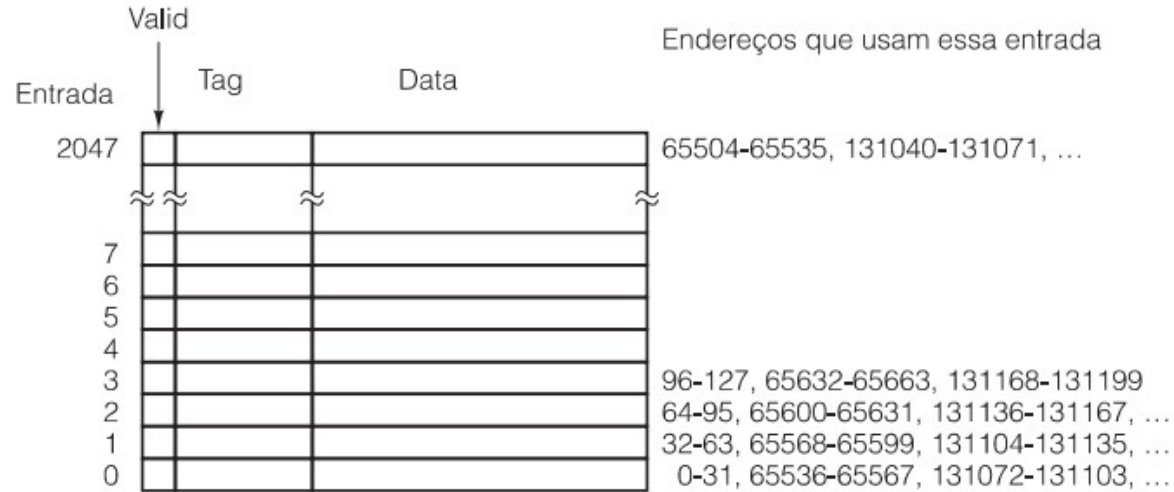
- Cache de mapeamento direto
- Cache associativa
- Cache associativa de conjunto

# Cache de mapeamento direto

- Mapeamento direto e Mapeamento direto em conjunto
- Implementação mais simples. Porém uma linha da memória pode ocupar apenas uma linha da cache.
- Apresentação dos professores Alexandre Amory e Edson Moreno

# Cache de mapeamento direto

(a) Cache de mapeamento direto. (b) Endereço virtual de 32 bits.



(a)



(b)

# Cache Mapeamento Associativo

Cada linha da cache pode armazenar qualquer linha da memória

## Memória Associativa



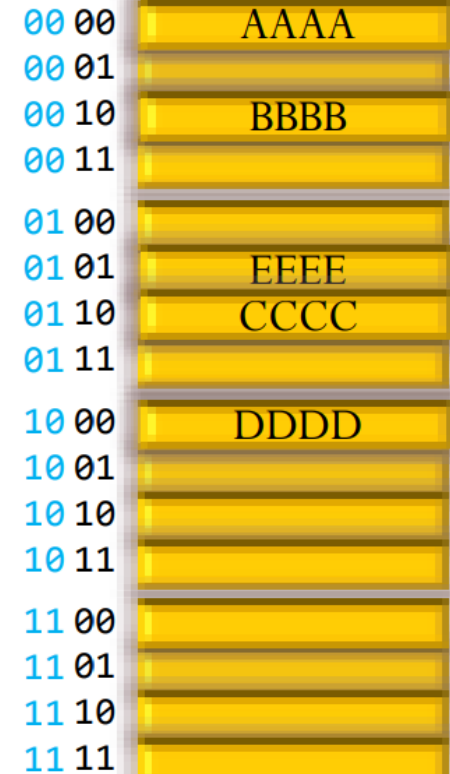
## CACHE



endereço 4 bits (16 posições)

tag (4 bits)

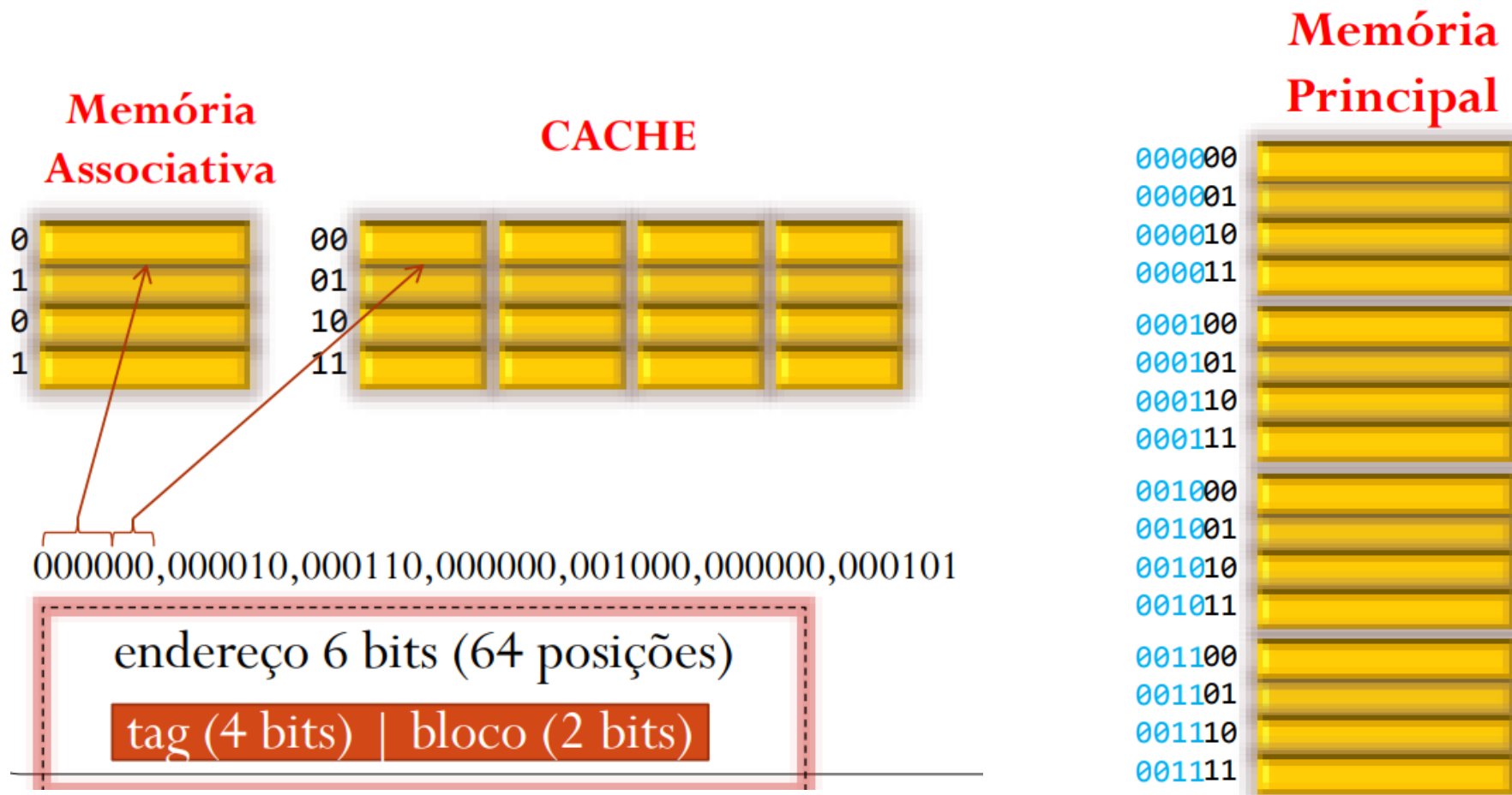
## Memória Principal



UTO FEDERAL

Espírito Santo

# Cache Mapeamento Associativo em blocos



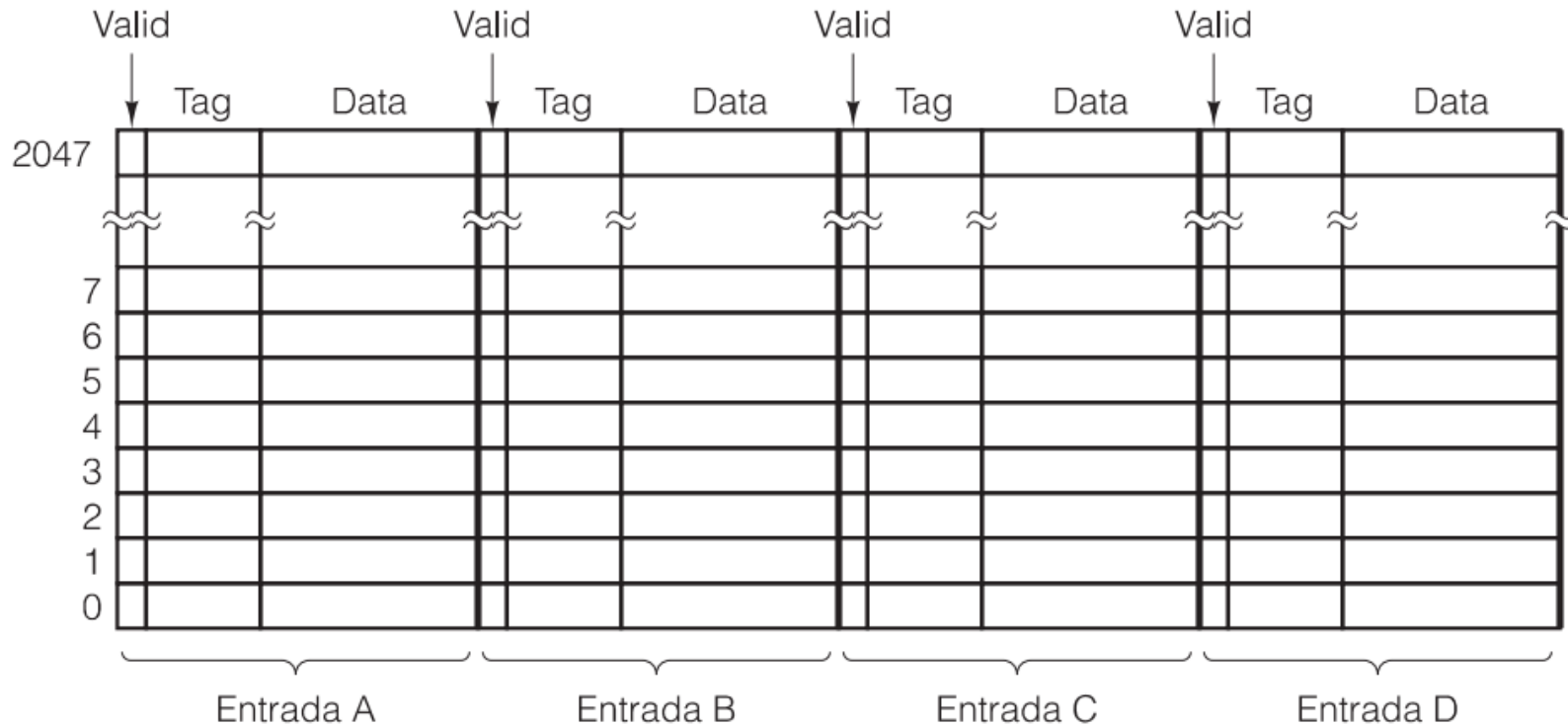


# Cache associativa de conjunto

- Une os dois conceitos de cache anteriores
- Utilizado nos processadores atualmente
- Uma linha da memória pode utilizar mais de um local da cache (n-vias)
- Técnicas de substituição na cache
  - Geralmente é utilizada a LRU (Menos utilizada recentemente)

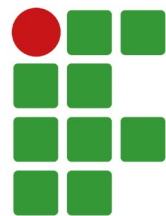
# Cache associativa de conjunto

Cache associativa de conjunto de quatro vias.



# Gravação de dados de cache na memória

- Escrita Direta (Write Through)
- Escrita Retardada (Write deferred) ou Escrita Retroativa (Write Back)



**INSTITUTO FEDERAL**  
Espírito Santo

Educação pública, gratuita e de qualidade