

# Memórias do Computador

## *Principal, Cache e Externa*

Givanaldo Rocha de Souza

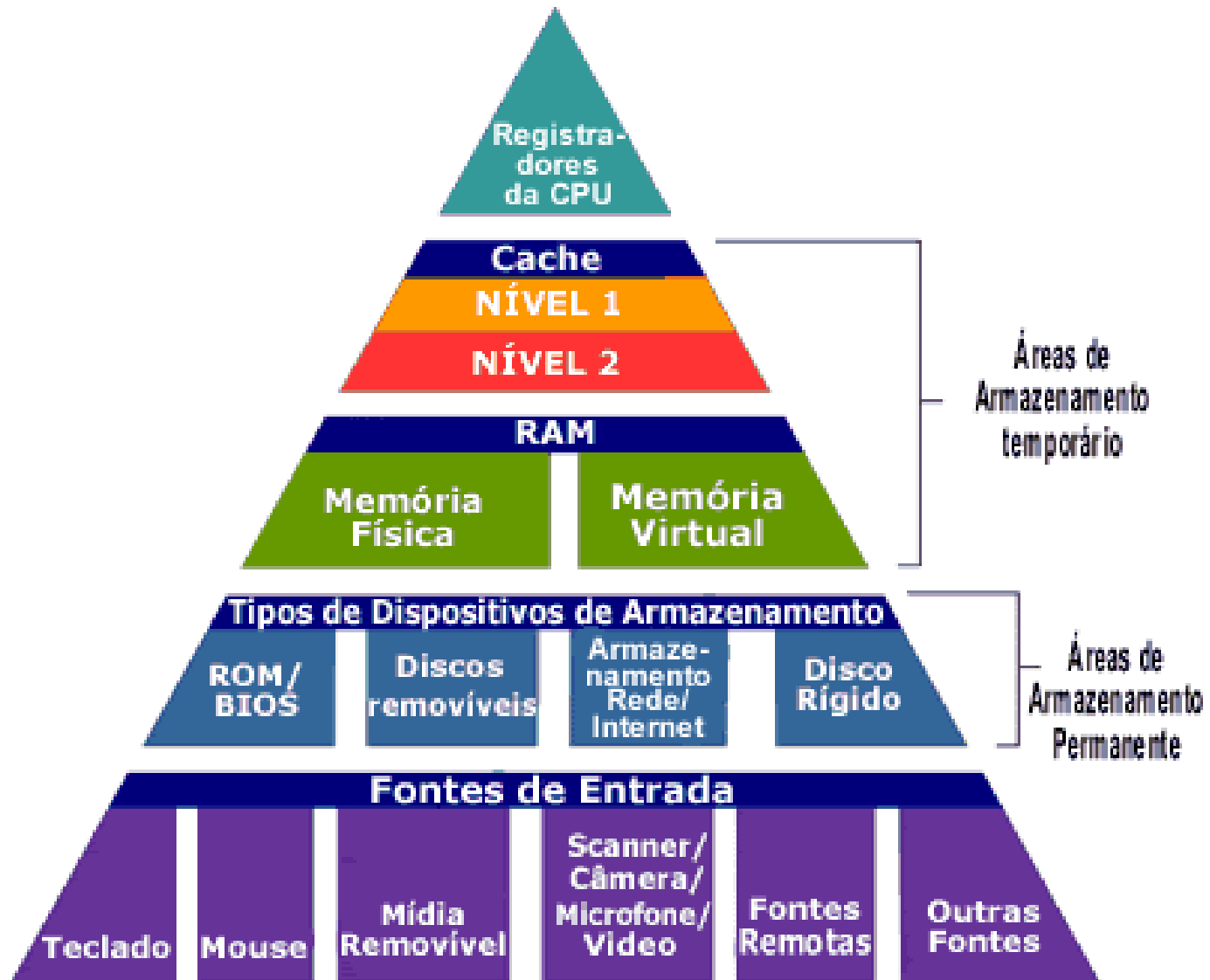
<http://docente.ifrn.edu.br/givanaldorochoa>

[givanaldo.rocha@ifrn.edu.br](mailto:givanaldo.rocha@ifrn.edu.br)

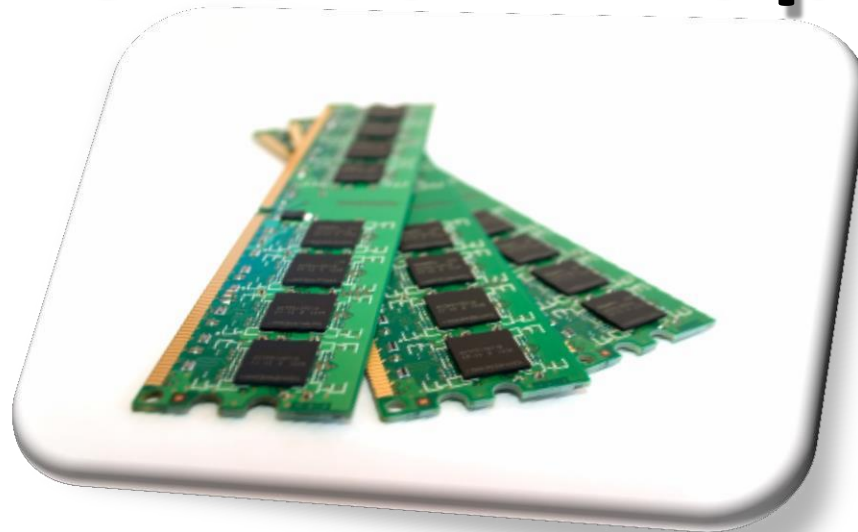
# Unidades de Medidas

Unidade	Sigla	Múltiplo	Quantidade de bytes
Byte	B	$2^0$	1 = 8 bits
KiloByte	KB	$2^{10}$	1024
MegaByte	MB	$2^{20}$	1 048 576
GigaByte	GB	$2^{30}$	1 073 741 824
TeraBytes	TB	$2^{40}$	1 099 511 627 777
PetaByte	PB	$2^{50}$	1 125 899 906 842 624
ExaByte	EB	$2^{60}$	1 152 921 504 606 846 976
ZettaByte	ZB	$2^{70}$	1 180 591 620 717 411 303 424
YotaByte	YB	$2^{80}$	1 208 925 819 614 629 174 706 176

# Hierarquia das memórias

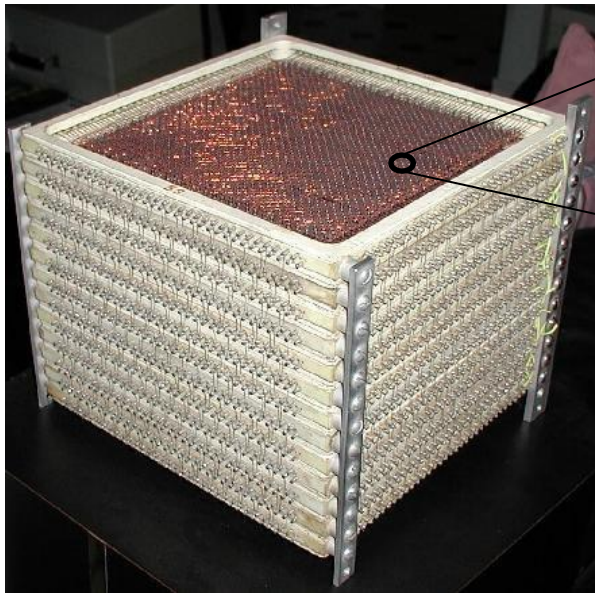


## Memória Principal

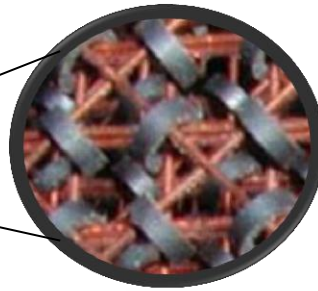


# Memória Principal

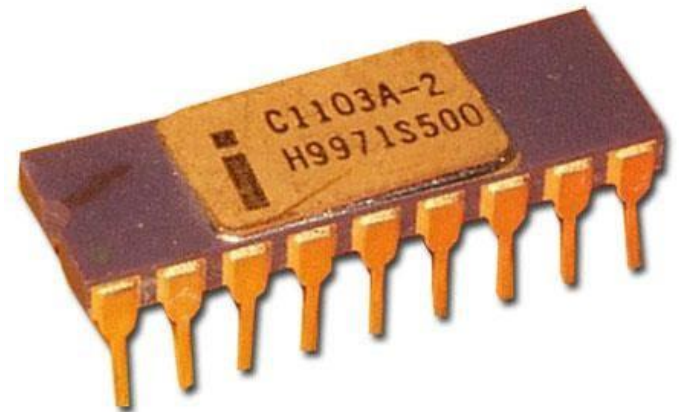
- ❑ Nos primeiros computadores, as memórias de acesso aleatório eram uma matriz de loops ferromagnéticos em forma de anel (núcleos).
- ❑ Hoje, o uso de chips semicondutores para memória principal é praticamente universal.



*Um rack com memórias de ferrite.  
No detalhe, as “rosquinhas” da memória.*



Fonte: <http://www.museudocomputador.com.br>



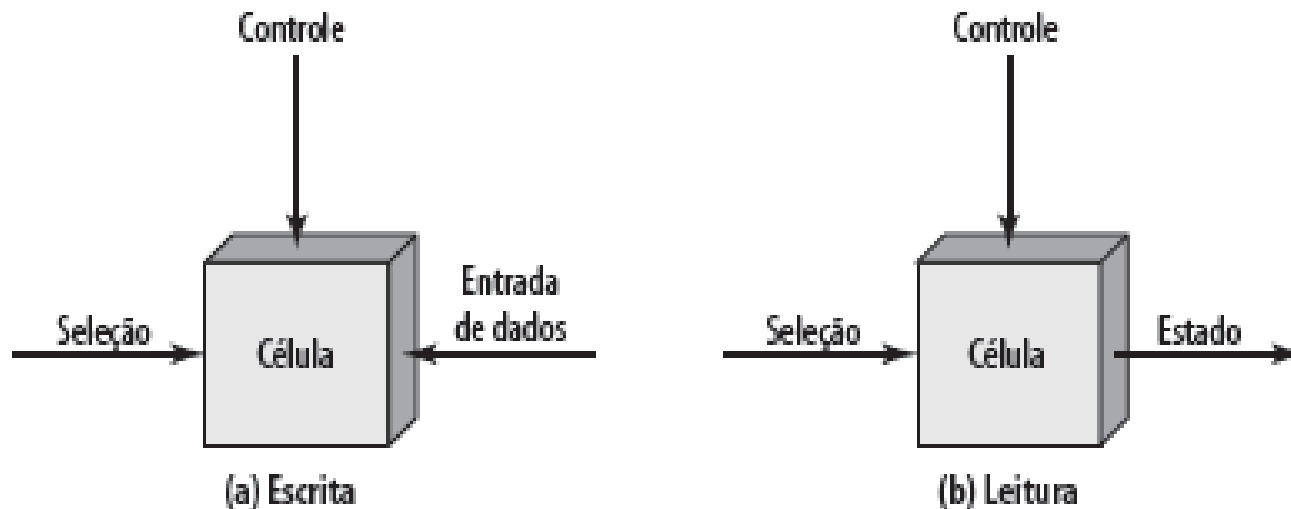
*O primeiro chip DRAM (Dynamic RAM).  
Em 1972 já era o melhor chip de memória  
semicondutor no mundo, acabando com a  
memória de ferrite.*

# Memória Principal

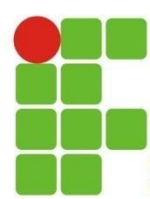
## Organização

### Elemento básico: **célula de memória**

- ✓ Apresentam dois estados estáveis (ou semiestáveis)
- ✓ Capazes de serem escritas, para definir o estado
- ✓ Capazes de serem lidas, para verificar o estado



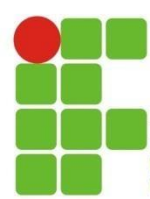
*Operação de uma célula de memória*



# Memória Principal

## ☐ Tipos de memória de semicondutores

Tipo de memória	Categoria	Mecanismo de apagamento	Mecanismo de escrita	Volatilidade
Memória de Acesso Aleatório (RAM)	Memória de Leitura e escrita	Eletricamente, em Nível de Byte	Eletricamente	Volátil
Memória Apenas de Leitura (ROM)	Memória apenas de leitura	Não é possível	Máscaras	Não-volátil
ROM Programável (PROM)				
PROM Apagável (EPROM)	Memória Principalmente de leitura	Luz UV, em nível de pastilha	Eletricamente	
Memória Flash		Eletricamente, em nível de Blocos		
PROM Eletricamente Apagável (EEPROM)		Eletricamente, em nível de Bytes		



# Memórias RAM

---

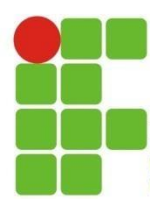
- ❑ RAM Dinâmica (DRAM)
  - Células armazenam dados com a carga de capacitores
  - É necessário um circuito de regeneração (*refresh*)
  - Usada na **Memória Principal**
  
- ❑ RAM Estática (SRAM)
  - Valores são armazenados usando configurações de *flip-flops* com portas lógicas
  - Não é necessário o circuito de regeneração
  - Usada na **Memória Cache**



# Memórias ROM

---

- ❑ Memória somente para leitura .
  
- ❑ Não volátil.
  
- ❑ Aplicações:
  - Bibliotecas de funções de uso frequente.
  - Programas do sistema.
  - Tabelas de função.
  
- ❑ Tamanho moderado, pois os dados não precisam ser armazenados em dispositivos secundários.



# Tipos de Memória ROM

---

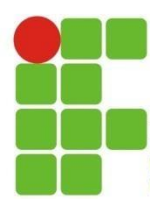
- ❑ ROM programável (PROM)
  - Mais barata que a ROM.
  - Pode ser escrita (eletricamente) apenas uma vez.
  - Necessário um equipamento especial para o processo de escrita ou “programação”.
  
- ❑ ROM programável e apagável (EPROM)
  - Lida e escrita eletricamente.
  - Antes da escrita todas as células de armazenamento são apagados através da exposição à luz ultravioleta intensa.
  - Mais cara que a PROM.



# Tipos de Memória ROM

---

- ❑ ROM programável e apagável eletronicamente (EEPROM)
  - Escrita pode ser feita somente nos bytes endereçados, sem modificar os demais.
  - Mais cara que a EPROM e menos densa.
  
- ❑ Flash
  - Intermediária entre a EPROM e EEPROM tanto no custo quanto na funcionalidade.
  - Usa tecnologia elétrica de apagamento.
  - Alta densidade.



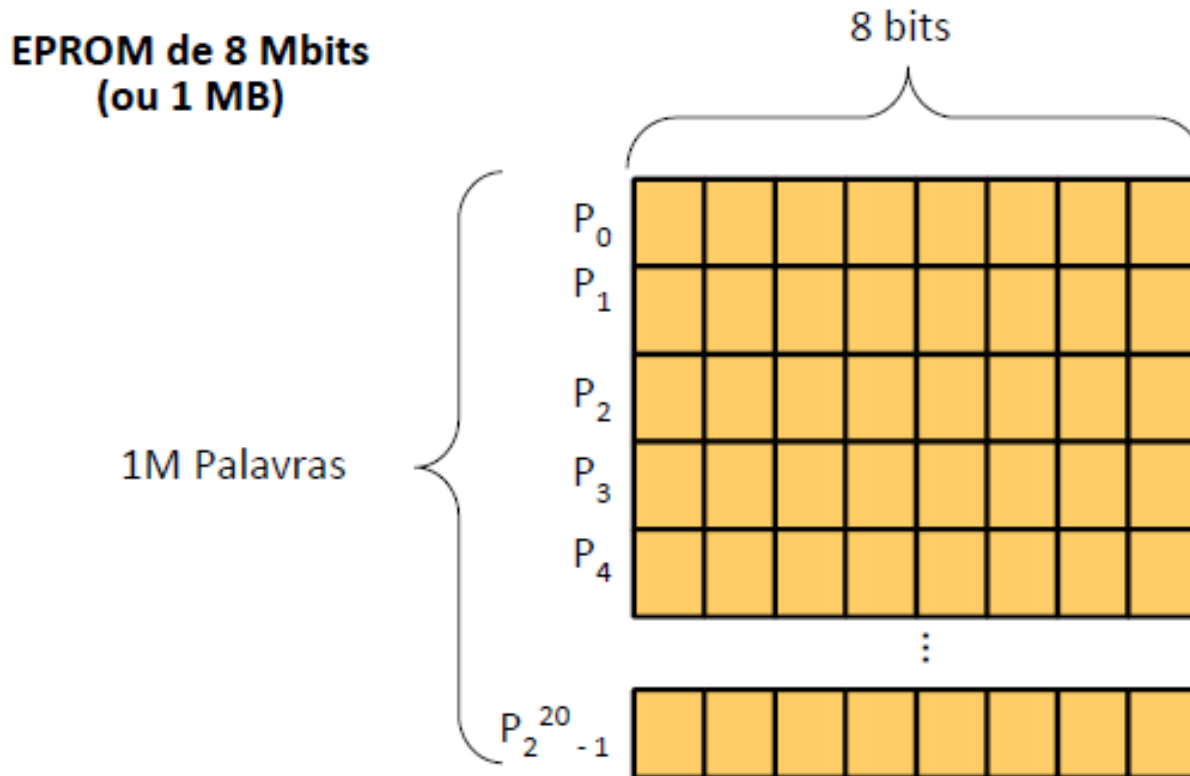
# Lógica do chip

---

- ❑ Memórias semicondutoras vem em chips empacotados que contêm um vetor de células de memória.
- ❑ Para as memórias semicondutoras, uma das principais questões de projeto é o *número de bits de dados que podem ser lidos/escritos de cada vez*.
- ❑ Em um extremo, o arranjo físico de células no vetor é o mesmo que o arranjo lógico de palavras de memória. O vetor é organizado em  $W$  palavras de  $B$  bits cada.
- ❑ No outro extremo está a organização de 1 bit por chip, em que os dados são lidos/escritos 1 bit de cada vez.

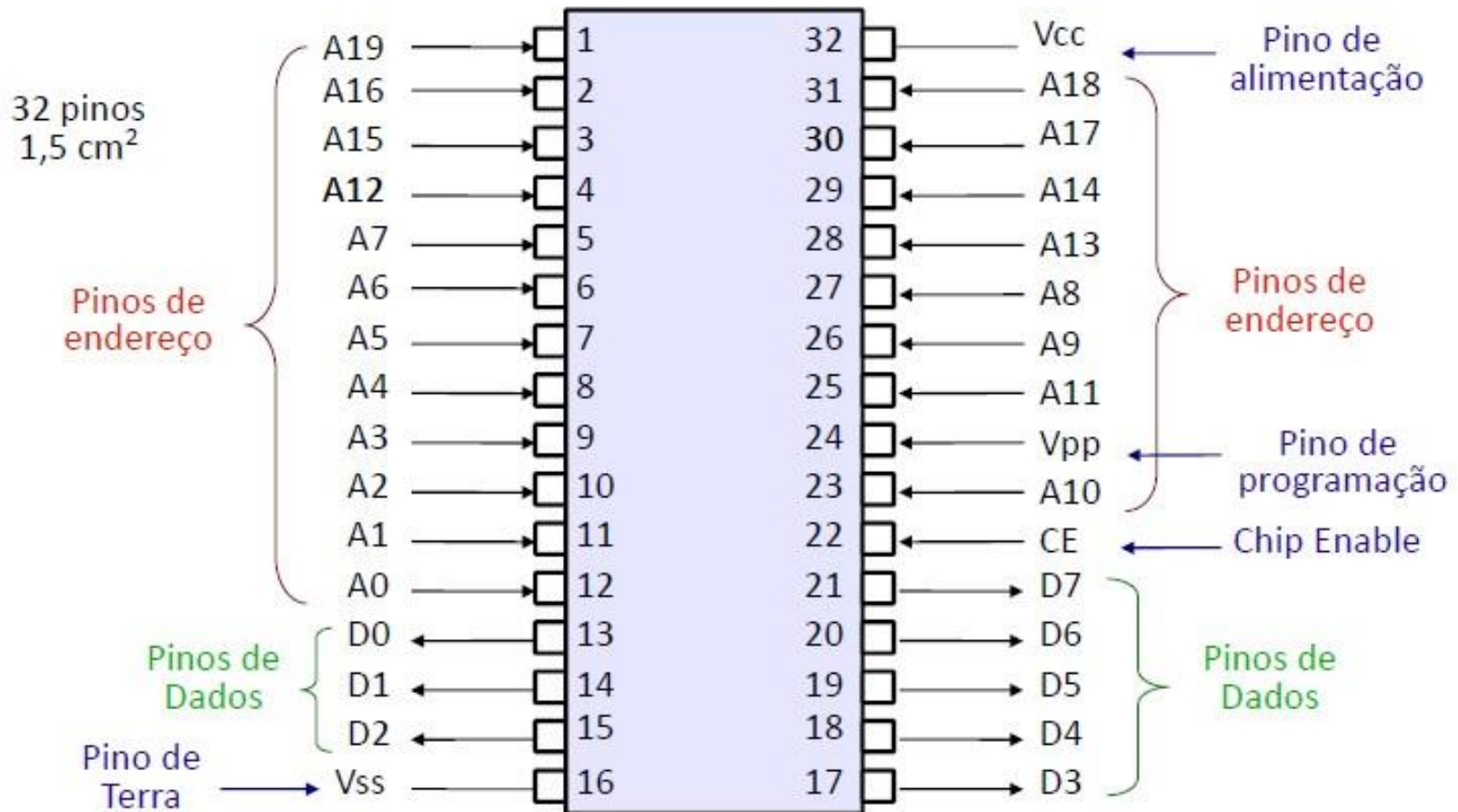
# Arranjos físicos das células

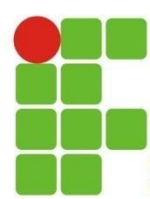
- ❑ **Exemplo 01:** arranjo físico é igual ao arranjo lógico das palavras na memória, tal como é percebido pelo processador.



# Arranjos físicos das células

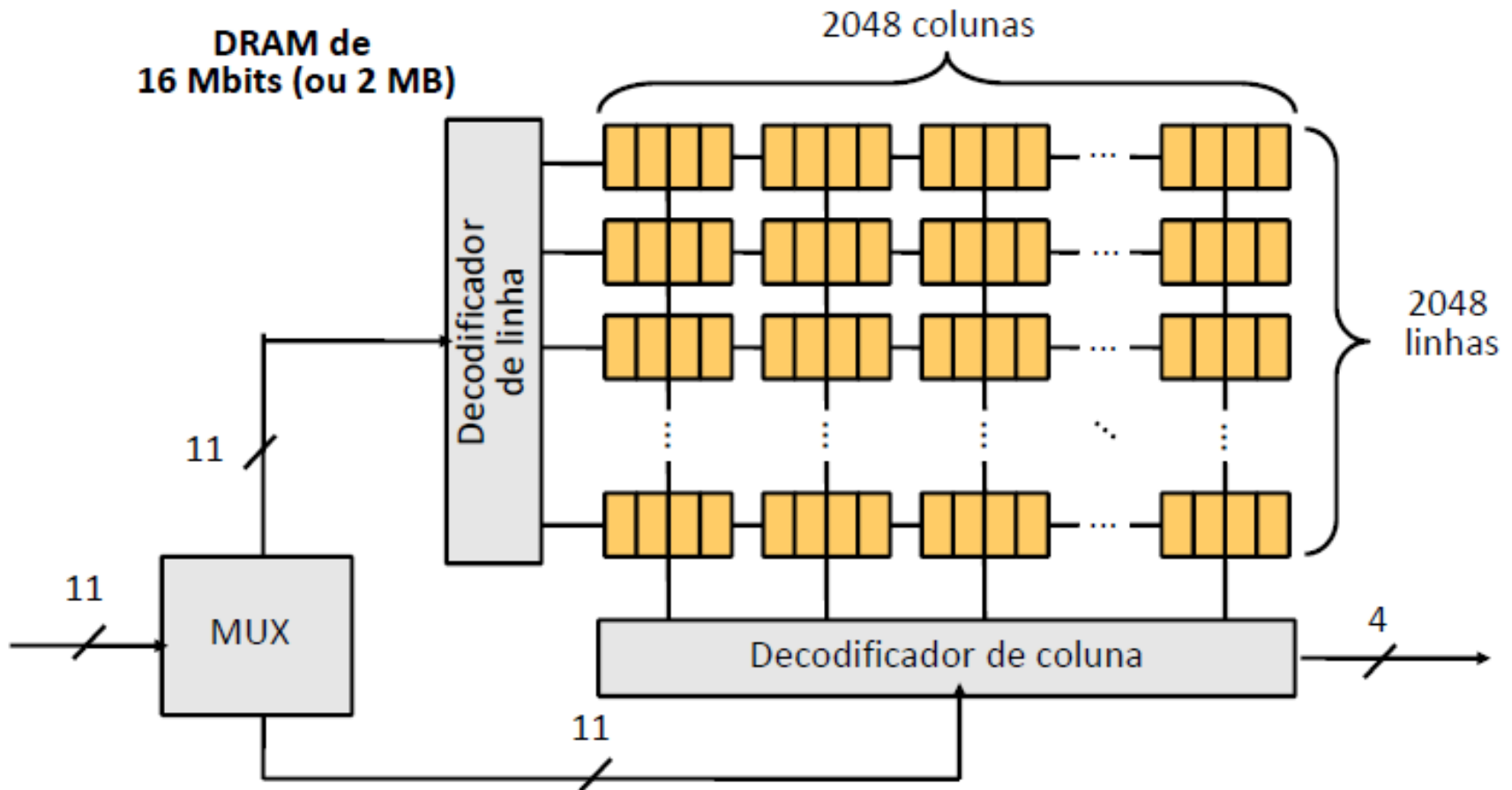
## Formato da pastilha do Exemplo 01:





# Arranjos físicos das células

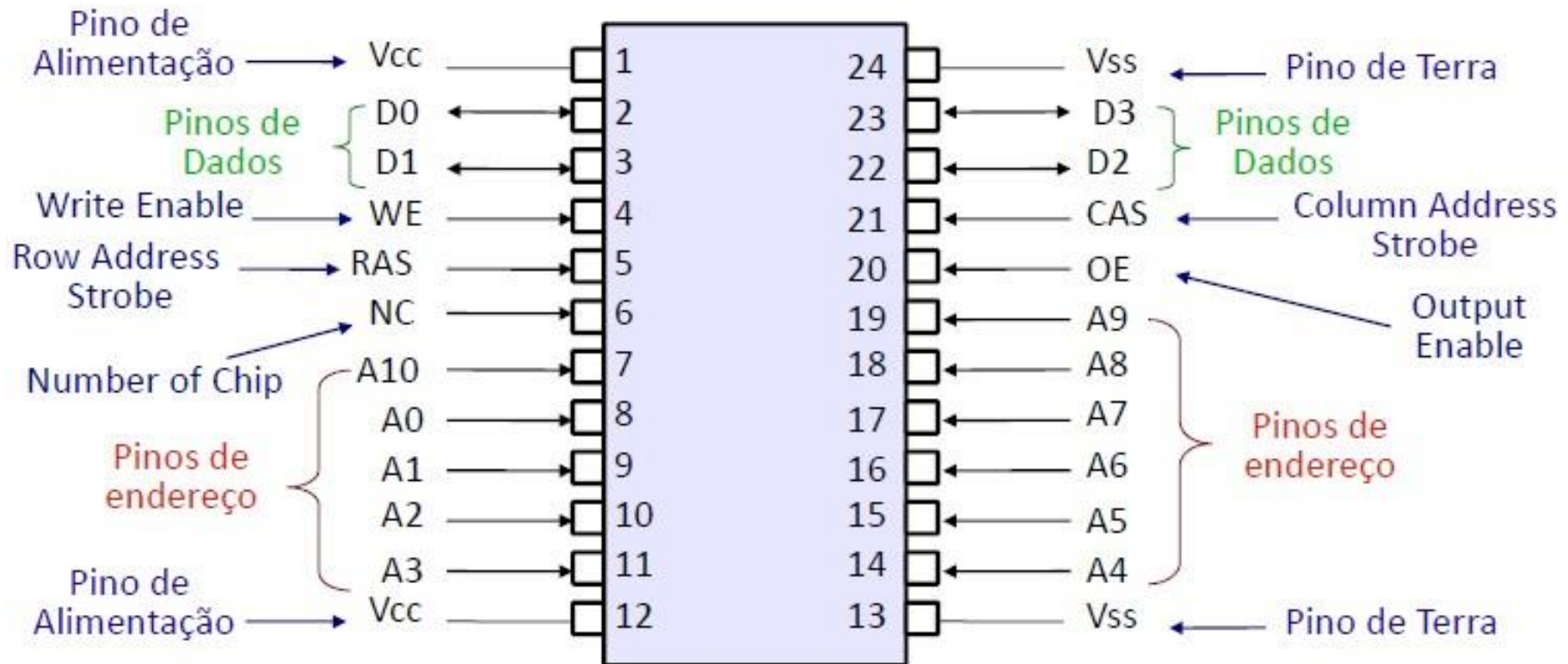
- Exemplo 02: arranjo em matrizes quadradas contendo grupos de células (16 Mbits = 4 matrizes de 2048 elementos)



# Arranjos físicos das células

## Formato da pastilha do Exemplo 02:

16 pinos



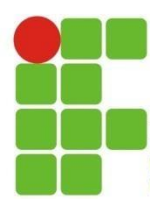




# Arranjos físicos das células

---

- O arranjo em matrizes quadradas de grupos de células possibilita pastilhas mais densas.
- A adição de uma linha de endereço faz com que a quantidade de linhas e colunas da matriz seja duplicada.
- A capacidade total de memória da pastilha é, portanto, quadruplicada.



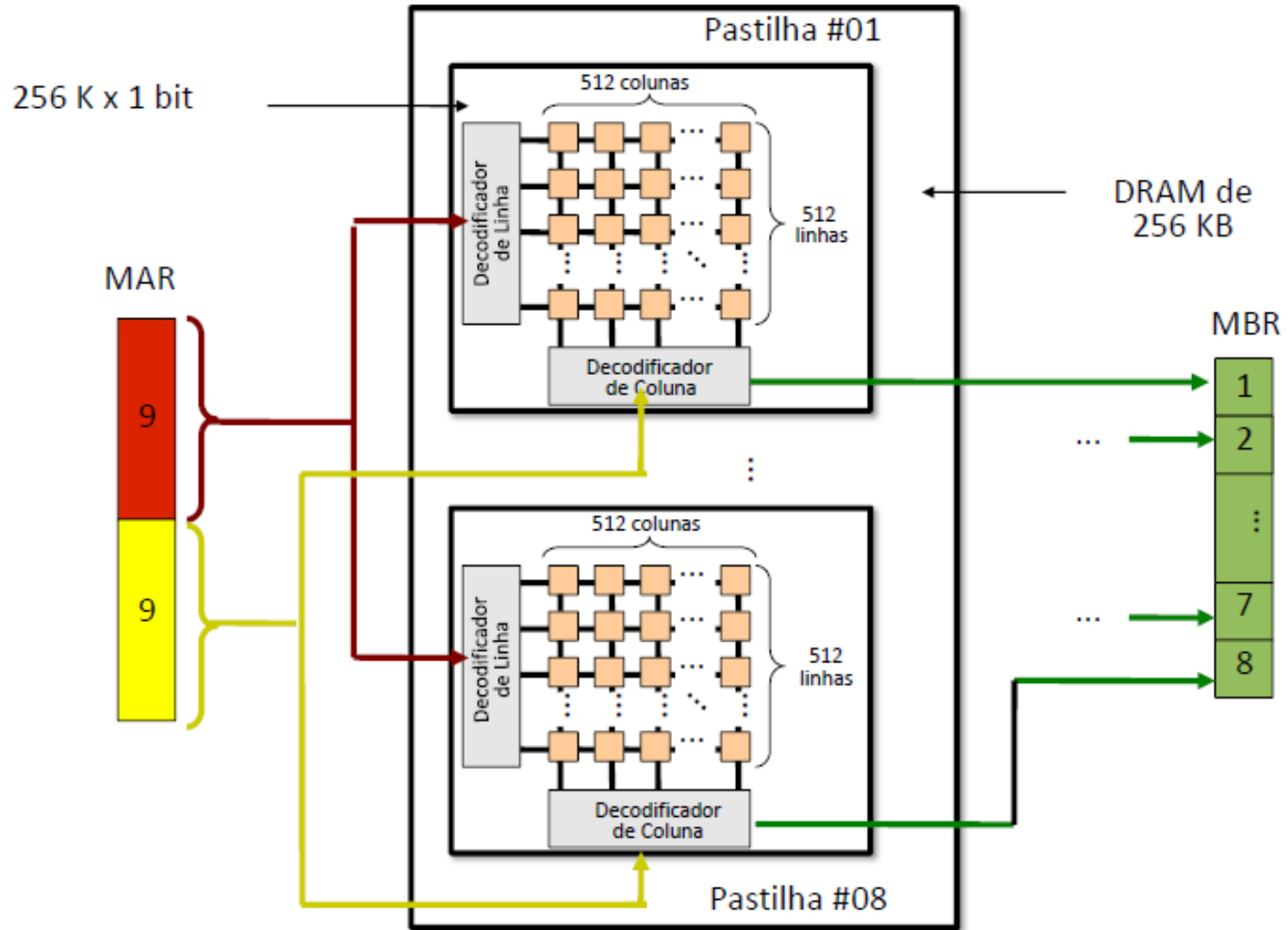
# Organização em Módulos

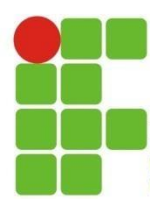
---

- Se um chip de RAM contém apenas 1 bit por palavra, claramente precisamos de pelo menos um número de chips igual ao número de bits por palavra.
  - A seguir, um módulo de memória consistindo em 256K palavras de 8 bits.
  - Para 256K, um endereço de 18 bits é necessário.
  - O endereço é apresentado a 8 chips de 256K x 1 bit, cada um oferecendo a entrada/saída de 1 bit.
-



# Organização em Módulos

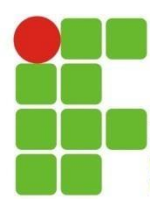




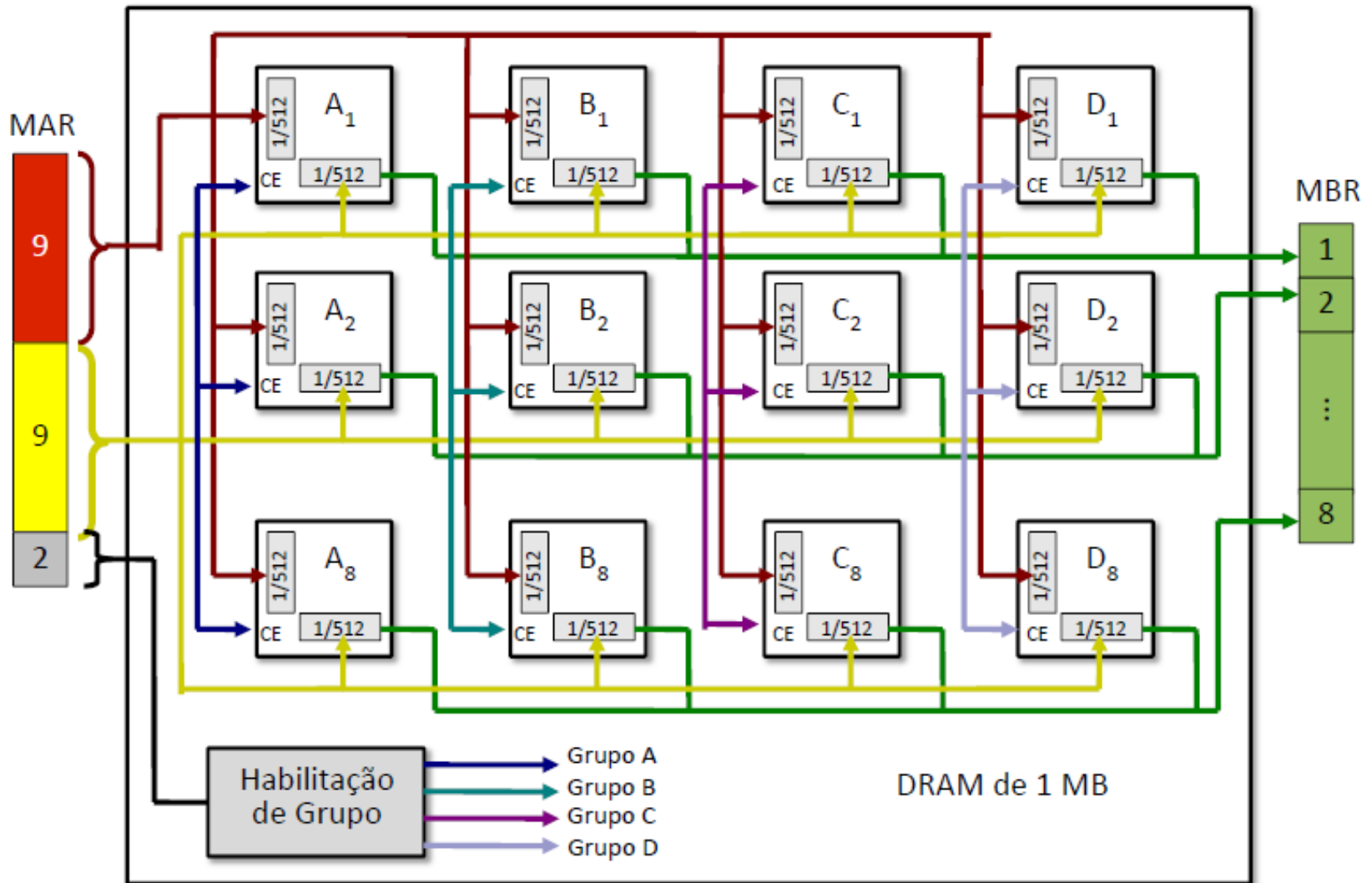
# Organização em Módulos

---

- Essa organização funciona desde que o tamanho da memória seja igual ao número de bits por chip.
- No caso em que um memória maior é necessária, um vetor de chips é necessário.
  
- A seguir, uma memória consistindo em 1M palavras por 8 bits por palavra.
- Temos 4 colunas de chips, cada coluna contendo 256 K palavras.
- Para 1M palavra, 20 linhas de endereços são necessários.



# Organização em Módulos

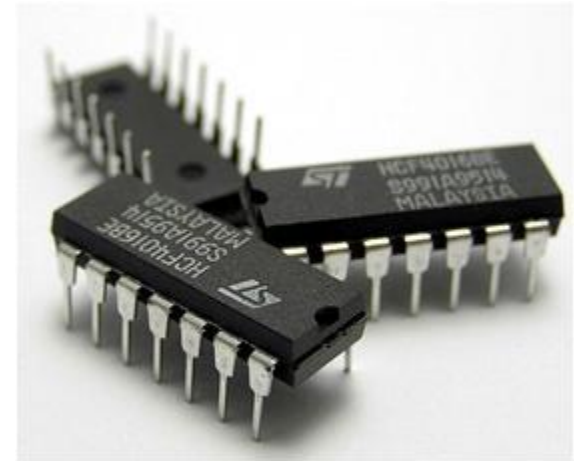




# Formas de encapsulamento

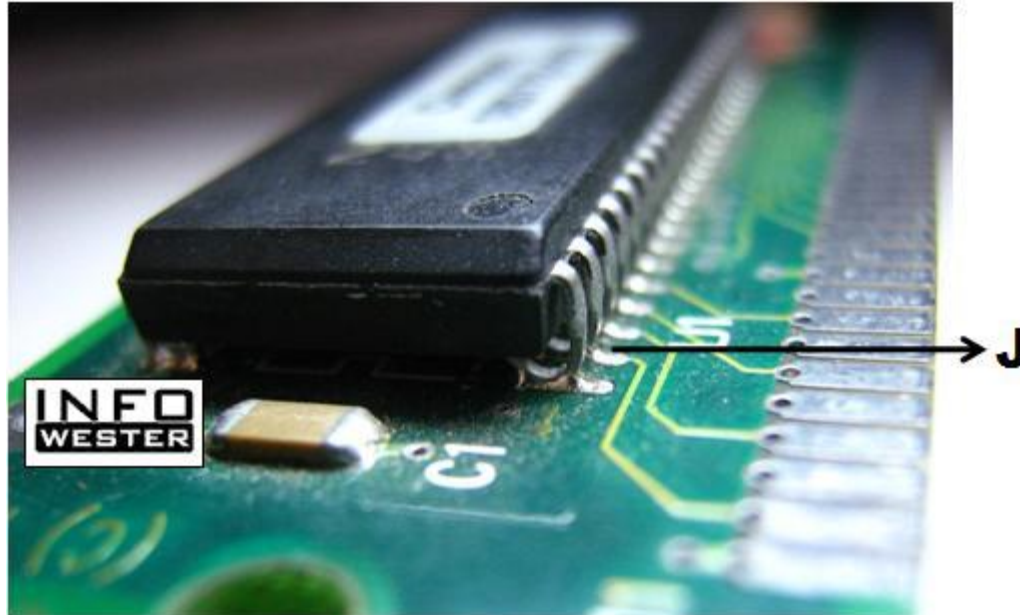
## ❑ Módulos DIP (Dual Inline Package)

- Usados em PCs antigos (XT, 286 e os primeiros 386)
- Soldados diretamente na placa mãe ou encaixados individualmente em soquetes disponíveis na placa
- Upgrade de memória ou substituição de módulos era difícil



# Formas de encapsulamento

- **SOJ** (*Small Outline J-Lead*): esse encapsulamento recebe este nome porque seus terminais de contato lembram a letra 'J'. Foi bastante utilizado em módulos SIMM (vistos mais à frente) e sua forma de fixação em placas é feita através de solda, não requerendo furos na superfície do dispositivo;



# Formas de encapsulamento

- **TSOP** (*Thin Small Outline Package*): tipo de encapsulamento cuja espessura é bastante reduzida em relação aos padrões citados anteriormente (cerca de 1/3 menor que o SOJ). Por conta disso, seus terminais de contato são menores, além de mais finos, diminuindo a incidência de interferência na comunicação. É um tipo aplicado em módulos de memória SDRAM e DDR (que serão abordados adiante). Há uma variação desse encapsulamento chamado **STSOP** (*Shrink Thin Small Outline Package*) que é ainda mais fino;





# Formas de encapsulamento

- **CSP** (*Chip Scale Package*): mais recente, o encapsulamento CSP se destaca por ser "fino" e por não utilizar pinos de contato que lembram as tradicionais "perninhas". Ao invés disso, utiliza um tipo de encaixe chamado **BGA** (*Ball Grid Array*). Esse tipo é utilizado em módulos como DDR2 e DDR3 (que serão vistos à frente).



# Formas de encapsulamento

## ❑ Módulos SIMM (Single Inline Memory Module)

- 30 vias



- 72 vias

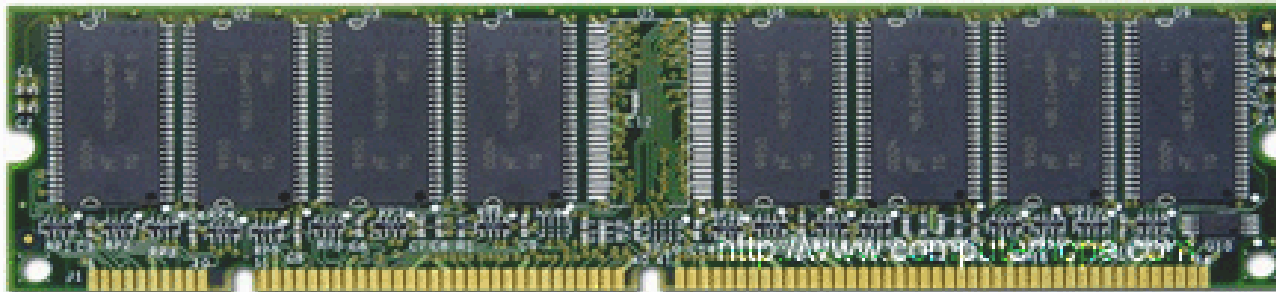




# Formas de encapsulamento

- ❑ Módulos DIMM (Dual Inline Memory Module)
  - Possuem contatos em ambos os lados do módulo
  - Possuem 168 vias
  - Trabalham com palavras de 64 bits

512MB DIMM

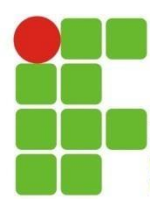


<http://www.computerhope.com>

# Formas de encapsulamento

- ☐ Comparação entre o tamanhos físicos:





# Evolução das DRAMs

---

## ☐ Memórias Regulares

- Primeiro tipo de memória usado em PCs.
- Módulos DIP.
- Usadas em computadores XT, 286 e nos primeiros 386.

## ☐ Memórias FPM (Fast Page Mode)

- Baseadas na ideia de que os dados são gravados sequencialmente na memória.
- Módulos SIMM de 30/72 vias, assíncronas (ciclo independente da placa-mãe).
- Usadas nos 386, 486 e nos primeiros Pentium.

## ☐ Memórias EDO (Extended Data Output)

- Mais rápidas que as memórias FPM.
  - Módulos SIMM de 72 vias, assíncronas (ciclo independente da placa-mãe).
  - Usada em alguns 486 com slots PCI na placa-mãe e Pentium.
-



# Evolução das DRAMs

---

## ☐ Memórias SDRAM (Synchronous DRAM)

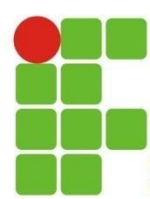
- Sincronizadas com os ciclos da placa-mãe (borda de subida do *clock*).
- Divisão dos módulos de memória em vários bancos.
- Também especificadas de acordo com a frequência nominal do barramento: PC66 (66MHz), PC100 (100MHz), PC133 (133MHz).

## ☐ Memórias DDR (Double Data Rate)

- Também conhecida como DDR SDRAM.
- Transferência de dados se dá na borda de subida e na borda de descida do *clock* da placa-mãe. Ou seja, duplica a velocidade com relação às SDRAM.

## ☐ Memórias DDR2, DDR3 e DDR4 (*novo*)

- Evoluções das memórias DDR.



# Diferenças entre DDRs

---

## ☐ Versões encontradas

- DDR: 266 MHz, 333 MHz e 400 MHz
- DDR2: 533 MHz, 677 MHz e 800 MHz
- DDR3: 1066 MHz e 1333 MHz
- DDR4 (projeção): 1600MHz, 2133MHz e 3200 MHz

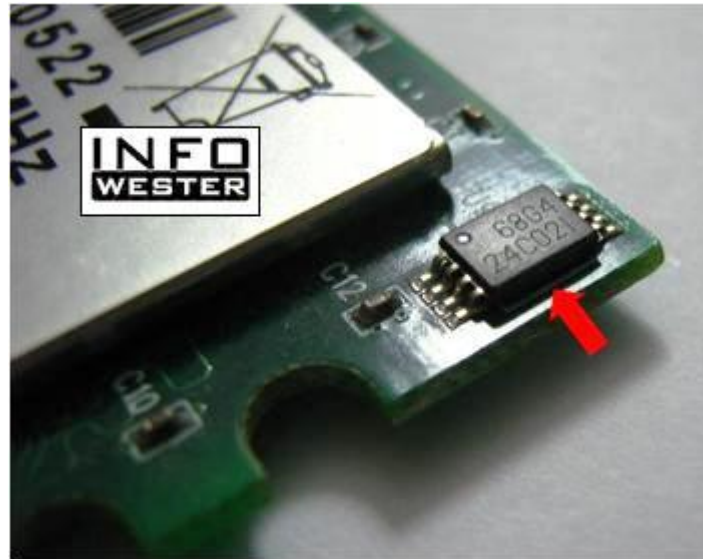
## ☐ Alimentação:

- DDR: 2,5 V
  - DDR2: 1,8V
  - DDR3: 1,5 v
  - DDR4: 1,2V
-



# SPD (Serial Presence Detect)

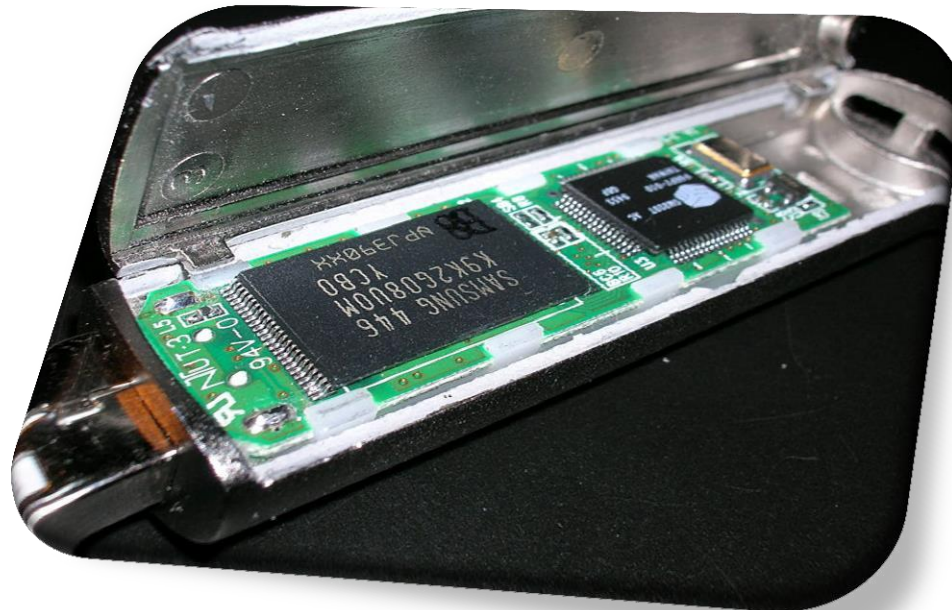
O SPD é um pequeno chip (geralmente do tipo EEPROM) inserido nos módulos de memória que contém diversas informações sobre as especificações do dispositivo, como tipo (DDR, DDR2, etc), voltagem, temporização/latência, fabricante, número de série, etc.

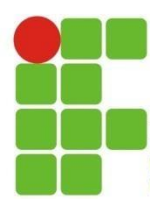




# Voltando à memória flash

- ❑ Memória do tipo EEPROM (Electrically-Erasable Programmable ROM).
- ❑ Desenvolvida na década de 1980 pela Toshiba.
- ❑ Chips semelhantes aos da memória RAM, permitindo que múltiplos endereços sejam apagados ou escritos numa só operação.
- ❑ Comumente usada em cartões de memória, pen drives, MP3 Players, câmeras digitais e celulares.





# Memória flash

---

- Oferece um tempo de acesso rápido e melhor resistência do que os tradicionais discos rígidos.
- Solid-State Drive: Conhecido pela sua sigla, SSD, é um dispositivo sem partes móveis para armazenamento não volátil de dados digitais.
- O disco SSD usa memória flash para armazenamento, o que difere dos sistemas magnéticos (como HDs e fitas) ou óticos (discos como CDs e DVDs). São discos mais caros que os HDD.
- Estratégias de uso: SSD + HDD = SSHD (adequação do custo/benefício).
- Quando empacotado em cartões de memória, são extremamente duráveis, suportando até a imersão em água.

# Exemplos de uso da memória flash

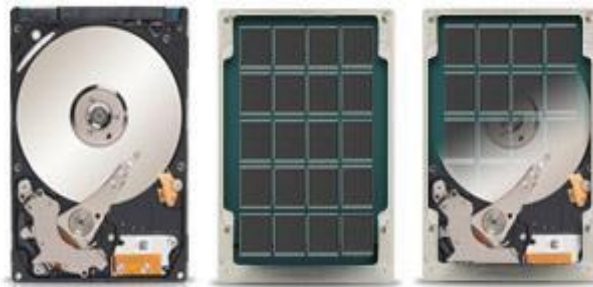


*Discos SSD*



*Cartões de memória*

Fonte: Seagate



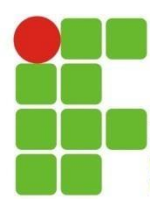
=



*Os SSHDs combinam os pontos fortes da SSD e do HDD em só dispositivo.*

## Memória Cache





# Memória Cache

---

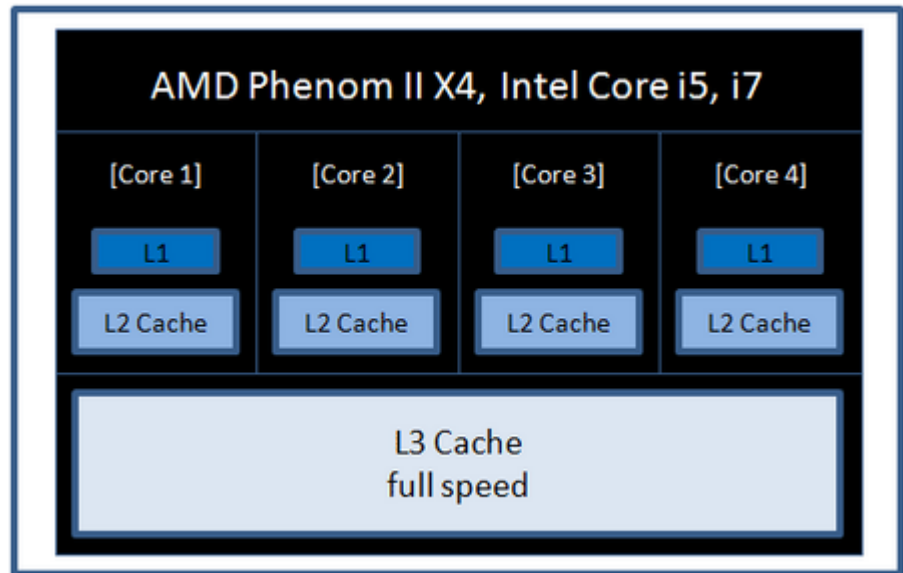
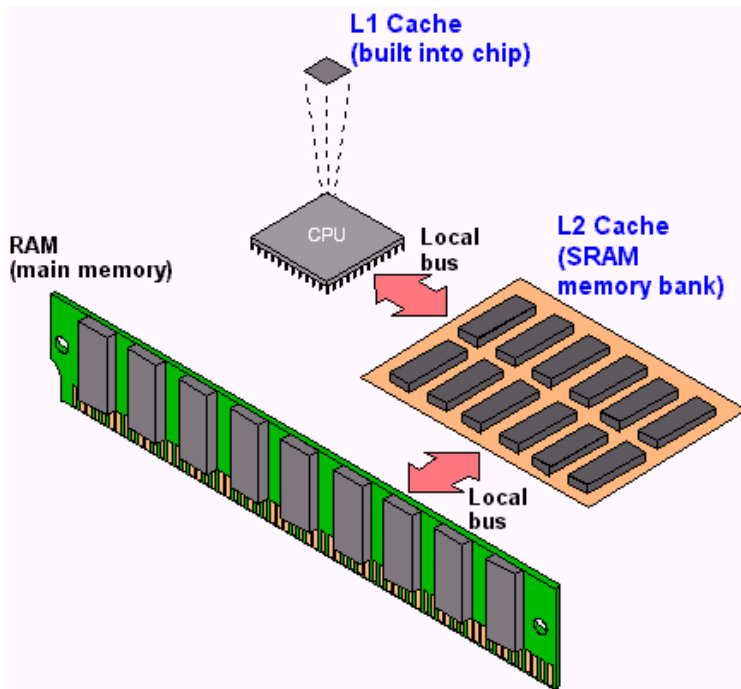
- ❑ Apesar dos avanços da memória, ainda demorava muito para os dados irem da memória principal para o processador.
  - ❑ Uma memória principal grande e lenta é combinada com uma memória cache pequena e rápida.
  - ❑ A memória cache foi projetada para acumular momentaneamente os dados a serem enviados do ou para o processador e seus periféricos.
  - ❑ O princípio é duplicar parte dos dados da memória principal em um módulo menor e mais rápido.
-

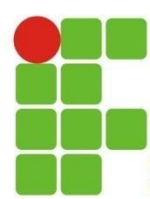


# Níveis de Memória Cache

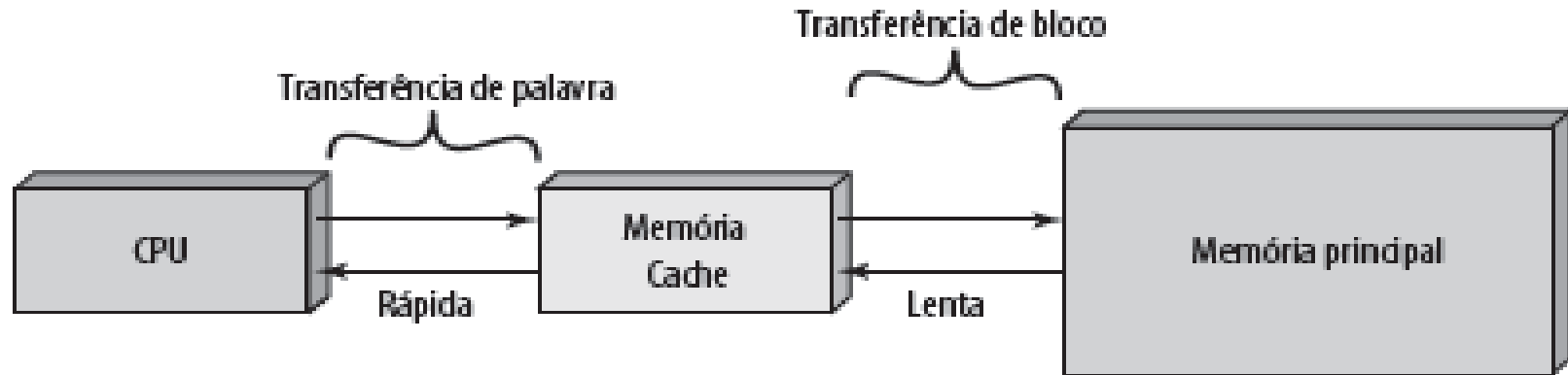
- ❑ Nível 1 (L1): no próprio chip do processador.
- ❑ Nível 2 (L2): chip separado, acoplada ao processador.
- ❑ Nível 3 (L3): chip na placa-mãe.

From Computer Desktop Encyclopedia  
© 1999 The Computer Language Co., Inc.

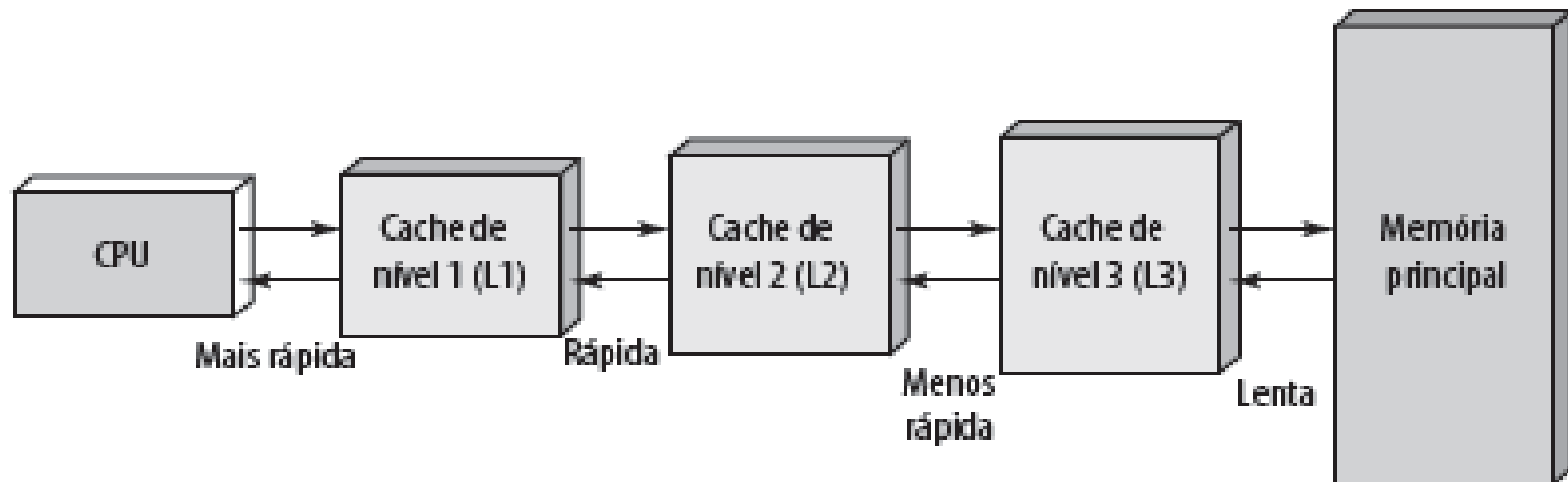




# Cache e Memória Principal



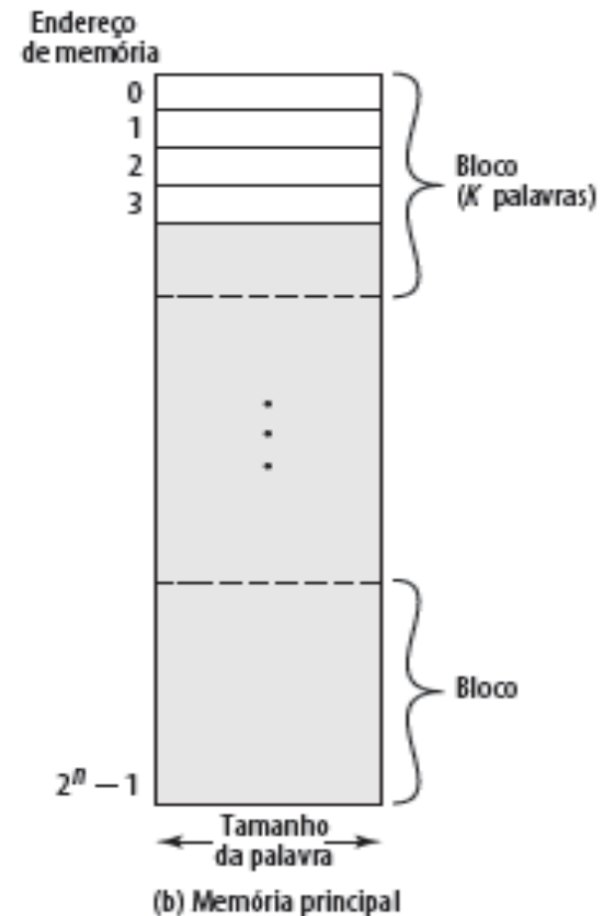
(a) Cache única



(b) Organização de cache em três níveis

# Cache e Memória Principal

- Em qualquer instante, um subconjunto dos blocos da memória principal reside na cache.



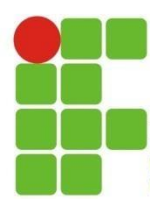




# Cache e Memória Principal

---

- ❑ A memória principal consiste em até  $2^n$  palavras endereçáveis, com cada palavra tendo um endereço distinto de  $n$  bits.
  
- ❑ Essa memória é considerada como sendo uma série de blocos de tamanho fixo com  $K$  palavras cada ( $M$  blocos na memória principal):
  - $M = 2^n / K$
  
- ❑ A cache consiste em  $m$  blocos, chamados linhas, cada uma contendo  $K$  palavras, mais um tag de alguns bits de controle.
  
- ❑ A largura de uma linha, sem incluir tag e bits de controle, é o tamanho da linha.



# Operação da Cache

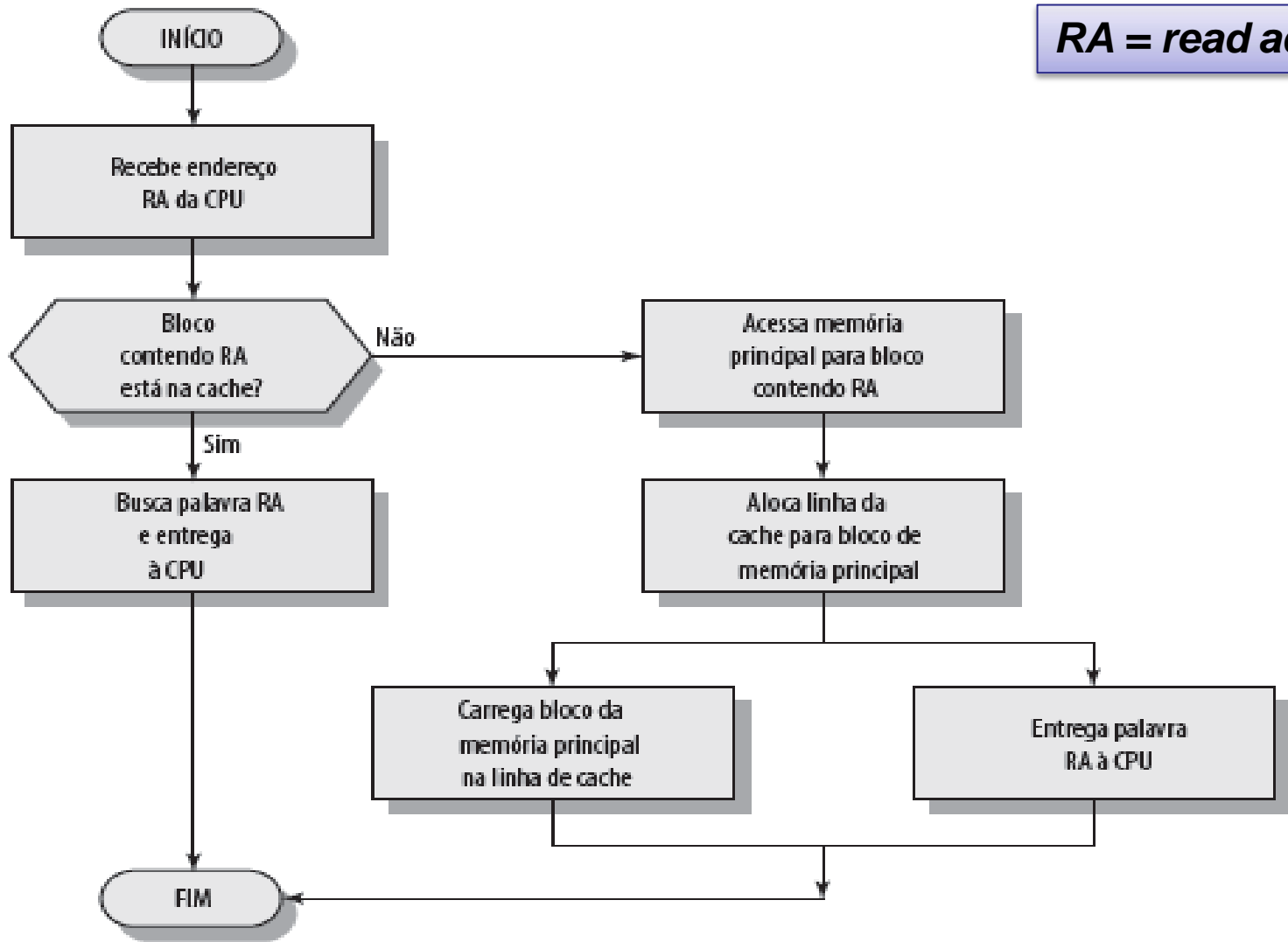
---

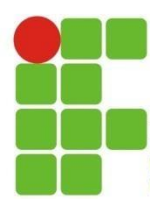
- CPU requisita conteúdo do local de memória.
  - Verifica se os dados estão em cache.
  - Se estiverem, copia da cache (rápido).
  - Se não, lê bloco solicitado da memória principal para a cache.
  - Depois, entrega da cache à CPU.
  - Cache inclui tags para identificar qual bloco da memória principal está em cada slot da cache.
-



# Operação de Leitura

**RA = read address**





# Operação de Leitura

---

- O processador solicita dados da memória:
  - Se os dados estiverem na cache (**hit**), os buffers de dados e endereço são desativados e a comunicação é apenas entre o processador e a memória cache, sem tráfego no barramento do sistema.
  - Se os dados não estiver na cache (**miss**), o endereço desejado é carregado no barramento do sistema e os dados são transferidos através do buffer de dados para a cache e para o processador.
  - Quanto mais presença de dados na cache, melhor é o desempenho do sistema.

# Projeto de Cache

---

- Endereçamento: *lógicos e físicos.*
  - Tamanho.
  - Função de mapeamento: *direta, associativa, associativa em conjunto.*
  - Algoritmo de substituição: *LRU, FIFO, LFU, aleatório.*
  - Política de escrita: *write-through, write-back, write once.*
  - Tamanho de bloco.
  - Número de caches: *um ou dois níveis, unificada ou separada.*
-



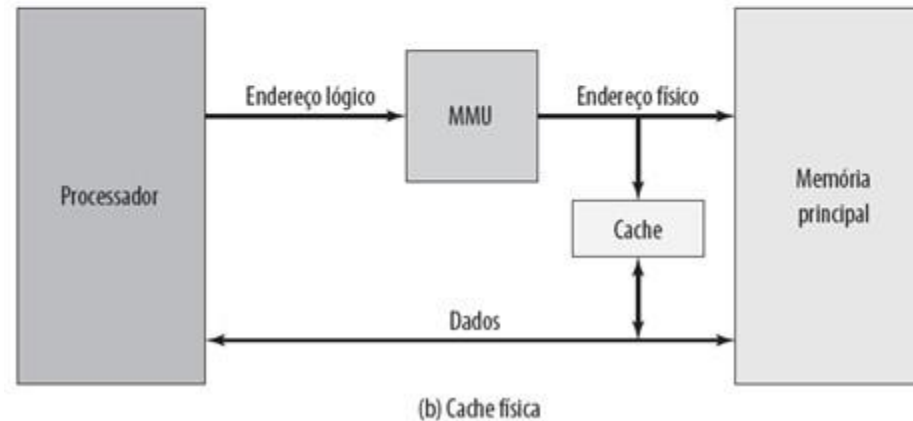
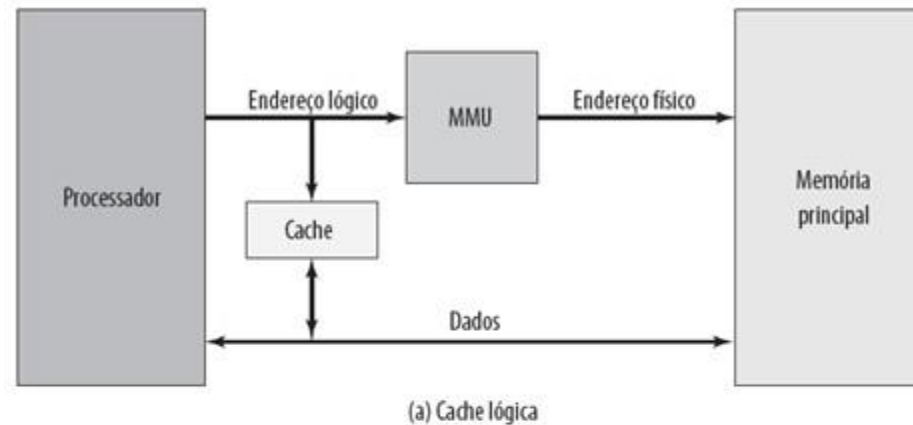
# Endereçamento

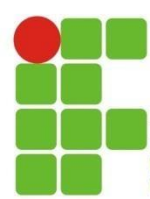
---

- ❑ Quase todos os processadores admitem memória virtual.
  - Permitem que programas enderecem a memória a partir de um ponto de vista lógico, sem considerar a quantidade de memória principal disponível fisicamente.
- ❑ Os campos das instruções contém endereços virtuais.
- ❑ Para leitura e escrita, uma unidade de gerenciamento da memória (MMU – *memory management unit*) traduz cada endereço virtual para um endereço físico na memória principal.
- ❑ O projetista escolhe colocar a cache **entre o processador e a MMU** ou **entre a MMU e a memória principal**.

# Endereçamento

- ❑ **Cache lógica** (cache virtual): armazena dados usando endereços virtuais. O processador acessa a cache sem passar pela MMU.
- ❑ **Cache física:** armazena dados usando endereços físicos da memória principal.
- ❑ Cache lógica tem vantagem na velocidade de acesso.
- ❑ A desvantagem é que os sistemas de memória virtual oferecem o mesmo espaço de endereçamento de memória virtual para cada aplicação.





# Tamanho

---

- ❑ Deve ser grande para que o tempo médio de acesso à memória total seja próximo ao tempo de acesso da memória cache.
- ❑ Deve ser pequena para que o custo total por bit seja próximo do custo por bit da memória principal.
- ❑ Outros motivos para minimização da cache:
  - Quanto maior a cache, maior o número de pinos – e mais lento o endereçamento.
  - O espaço limitado na placa de circuitos

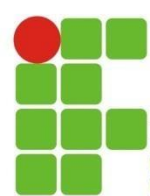


# Tamanho

**Tabela 4.3** Tamanhos de memória cache de alguns processadores

Processador	Tipo	Ano de introdução	Cache L1 <sup>a</sup>	Cache L2	Cache L3
IBM 360/85	Mainframe	1968	16 a 32 KB	—	—
PDP-11/70	Minicomputador	1975	1 KB	—	—
VAX 11/780	Minicomputador	1978	16 KB	—	—
IBM 3033	Mainframe	1978	64 KB	—	—
IBM 3090	Mainframe	1985	128 a 256 KB	—	—
Intel 80486	PC	1989	8 KB	—	—
Pentium	PC	1993	8 KB/8 KB	256 a 512 KB	—
PowerPC 601	PC	1993	32 KB	—	—
PowerPC 620	PC	1996	32 KB/32 KB	—	—

(Continua)



# Tamanho

(Continuação)

**Tabela 4.3** Tamanhos de memória cache de alguns processadores

Processador	Tipo	Ano de introdução	Cache L1 <sup>a</sup>	Cache L2	Cache L3
PowerPC G4	PC/servidor	1999	32 KB/32 KB	256 KB a 1 MB	2 MB
IBM S/390 G4	Mainframe	1997	32 KB	256 KB	2 MB
IBM S/390 G6	Mainframe	1999	256 KB	8 MB	—
Pentium 4	PC/servidor	2000	8 KB/8 KB	256 KB	—
IBM SP	Servidor avançado/ Supercomputador	2000	64 KB/32 KB	8 MB	—
CRAY MTA <sup>b</sup>	Supercomputador	2000	8 KB	2 MB	—
Itanium	PC/servidor	2001	16 KB/16 KB	96 KB	4 MB
SGI Origin 2001	Servidor avançado	2001	32 KB/32 KB	4 MB	—
Itanium 2	PC/servidor	2002	32 KB	256 KB	6 MB
IBM POWER5	Servidor avançado	2003	64 KB	1,9 MB	36 MB
CRAY XD-1	Supercomputador	2004	64 KB/64 KB	1 MB	—
IBM POWER6	PC/servidor	2007	64 KB/64 KB	4 MB	32 MB
IBM z10	Mainframe	2008	64 KB/128 KB	3 MB	24 a 48 MB

**a** Dois valores separados por uma barra referem-se a caches de instrução e dados.

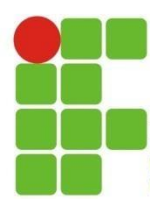
**b** As duas caches são apenas de instrução; não há caches de dados.



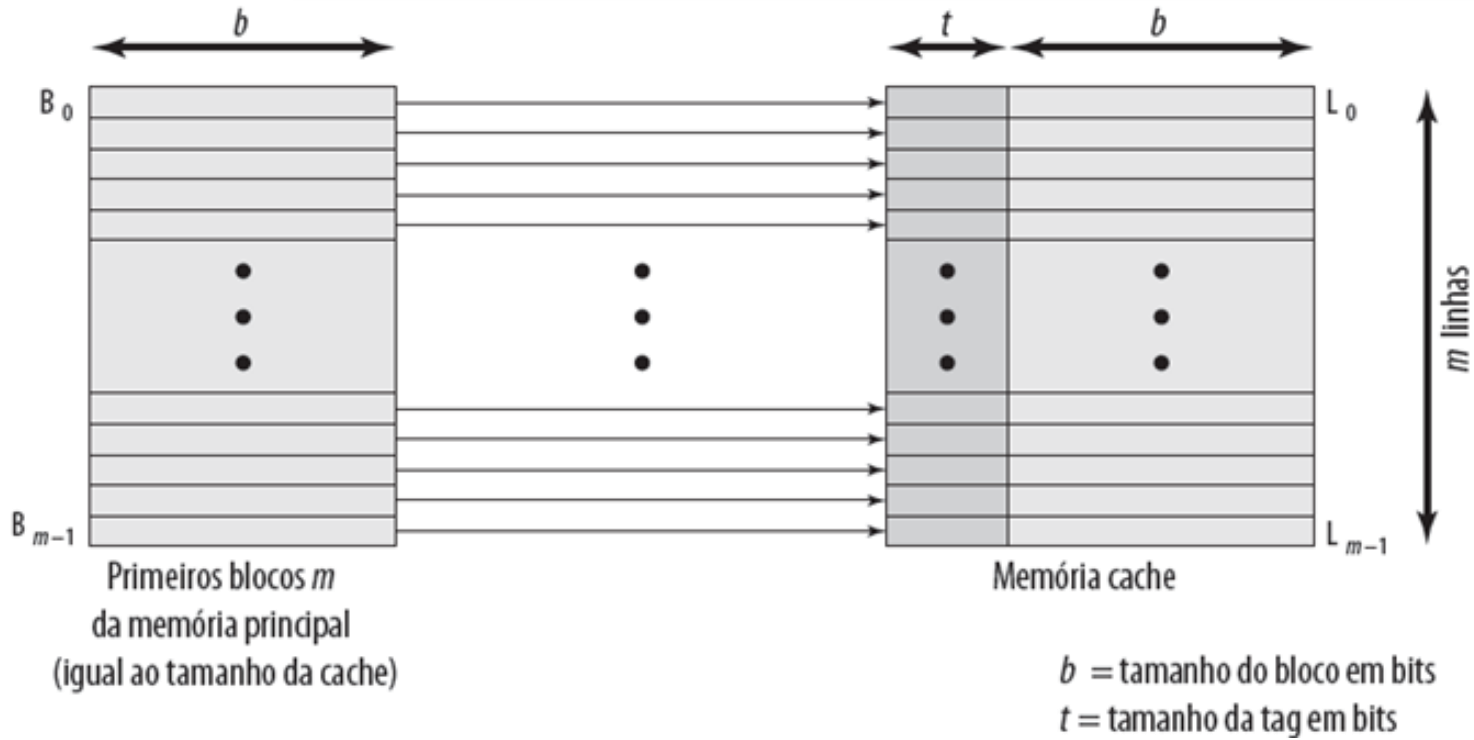
# Função de Mapeamento

---

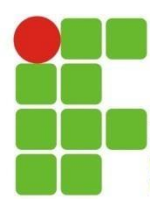
- ❑ Como existem menos linhas de cache do que blocos da memória principal, é necessário haver um algoritmo para mapear os blocos da memória principal às linhas da cache.
  
- ❑ Três técnicas:
  - Mapeamento direto  
*Mapeia cada bloco da memória principal a apenas uma linha de cache possível.*
  
  - Mapeamento associativo  
*Permite que cada bloco da memória principal seja carregado em qualquer linha da cache.*
  
  - Mapeamento associativo em conjunto  
*Um meio termo que realça os pontos fortes das técnicas direta e associativa, reduzindo as desvantagens.*



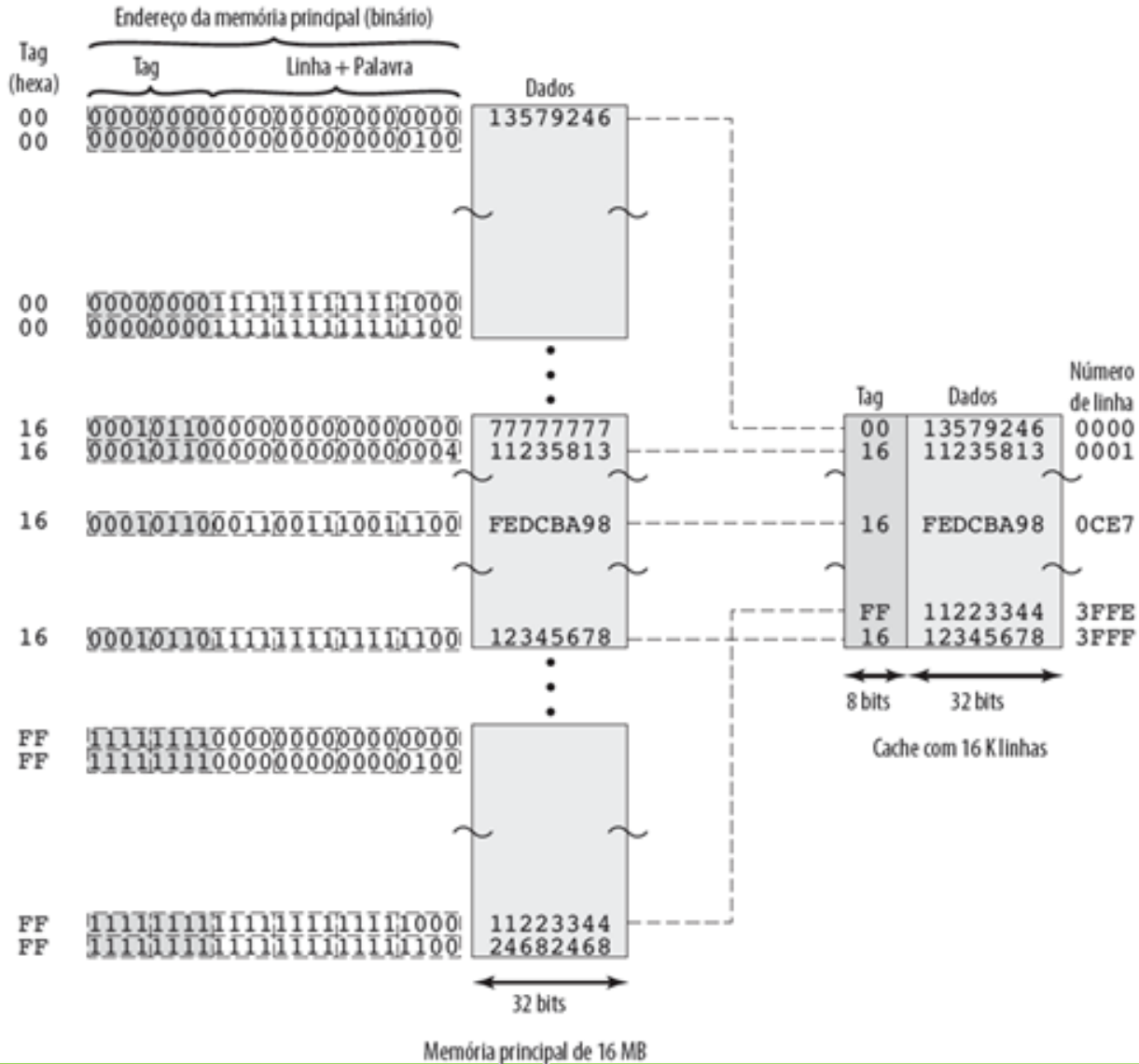
# Mapeamento direto



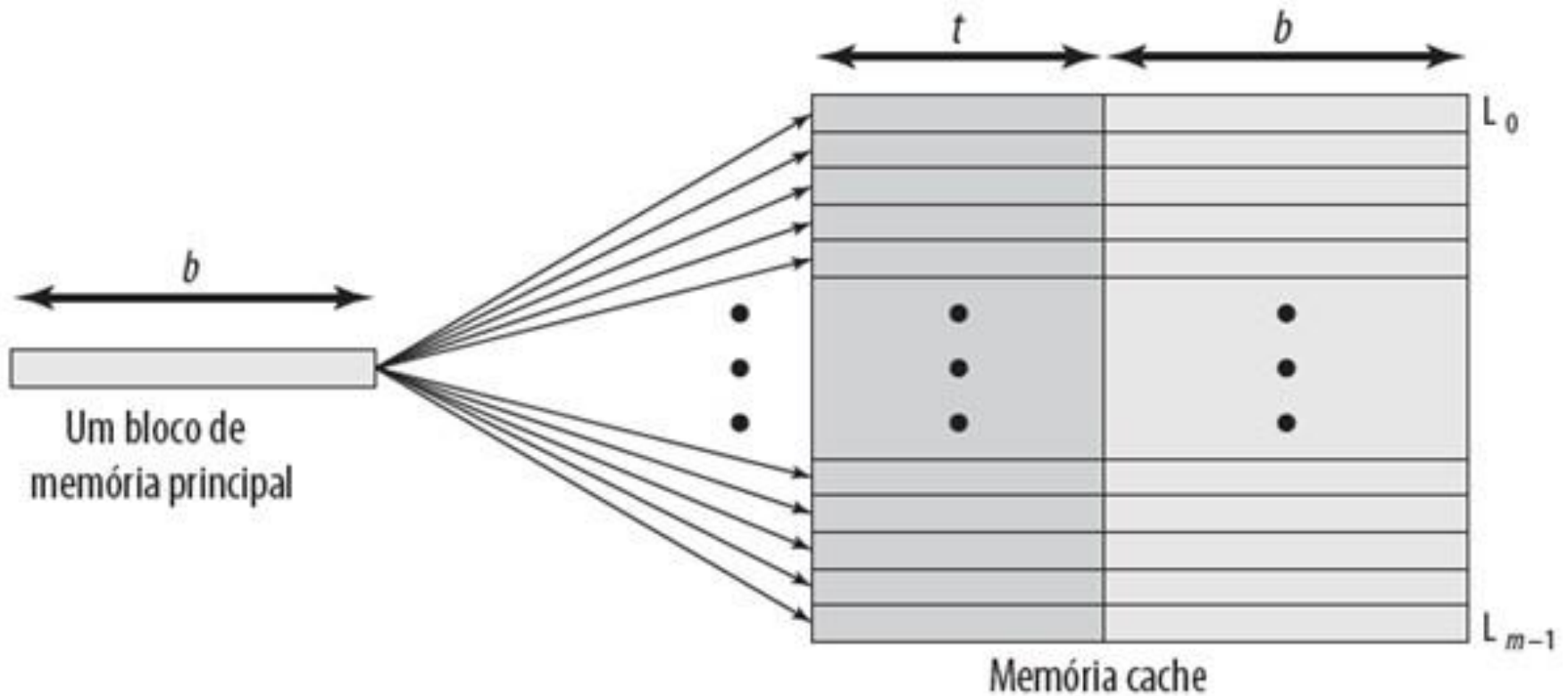
Linha de cache	Blocos de memória principal mapeados
0	$0, m, 2m, \dots, 2^s - m$
1	$1, m + 1, 2m + 1, \dots, 2^s - m + 1$
$\vdots$	$\vdots$
$m - 1$	$m - 1, 2m - 1, 3m - 1, \dots, 2^s - 1$

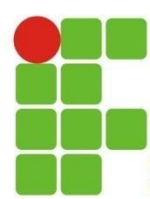


# Mapeamento direto (exemplo)

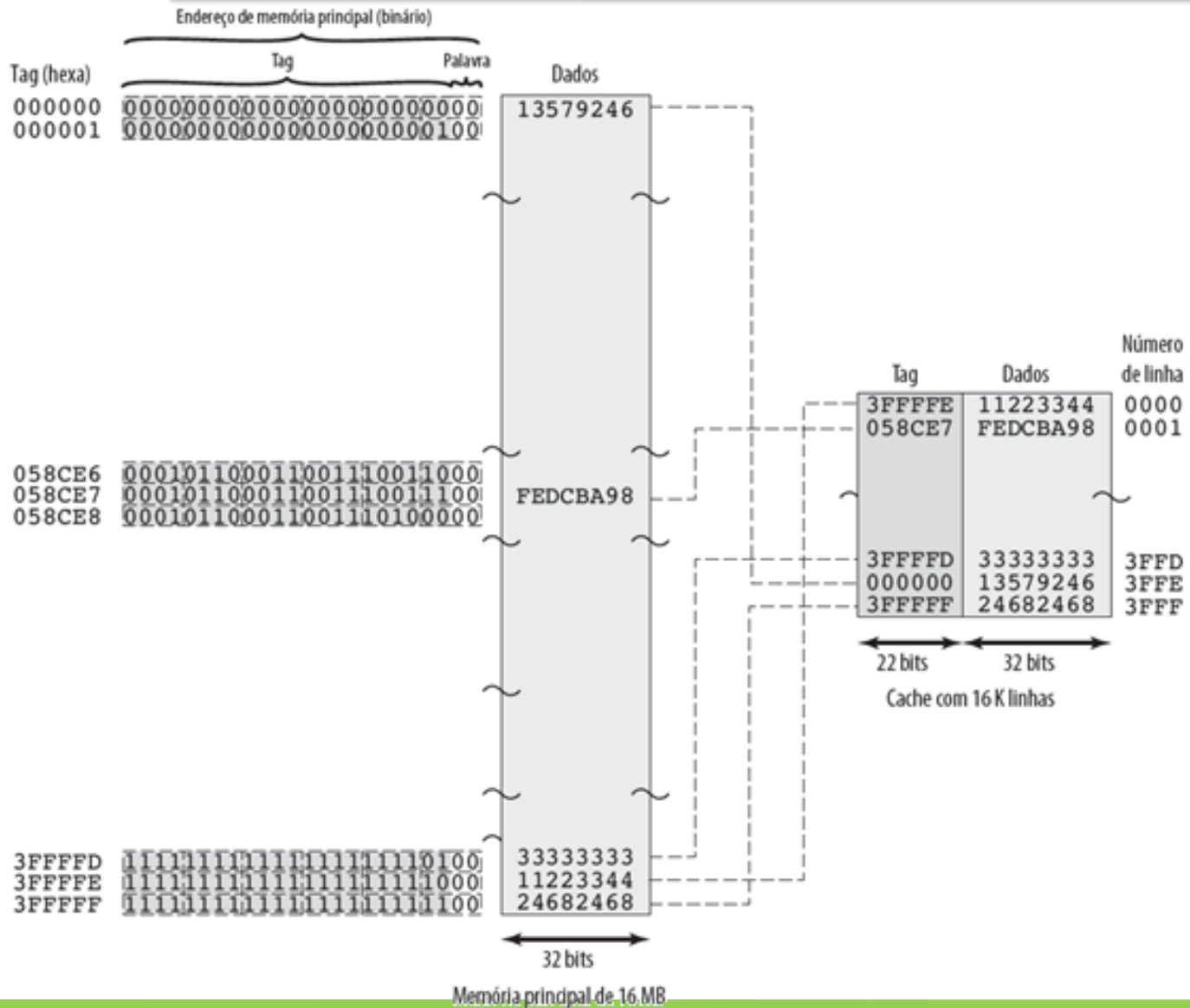


# Mapeamento associativo

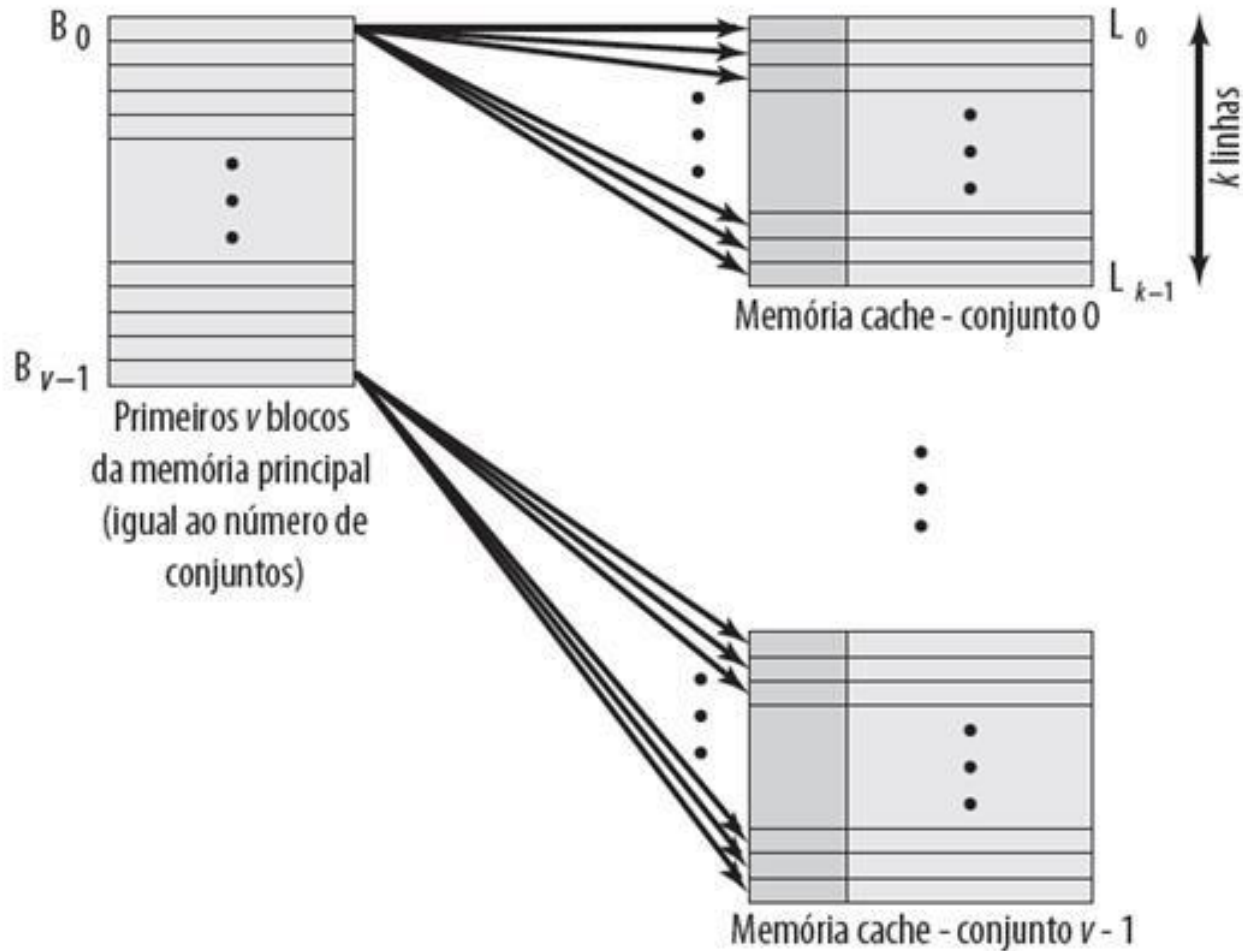




# Mapeamento associativo (exemplo)



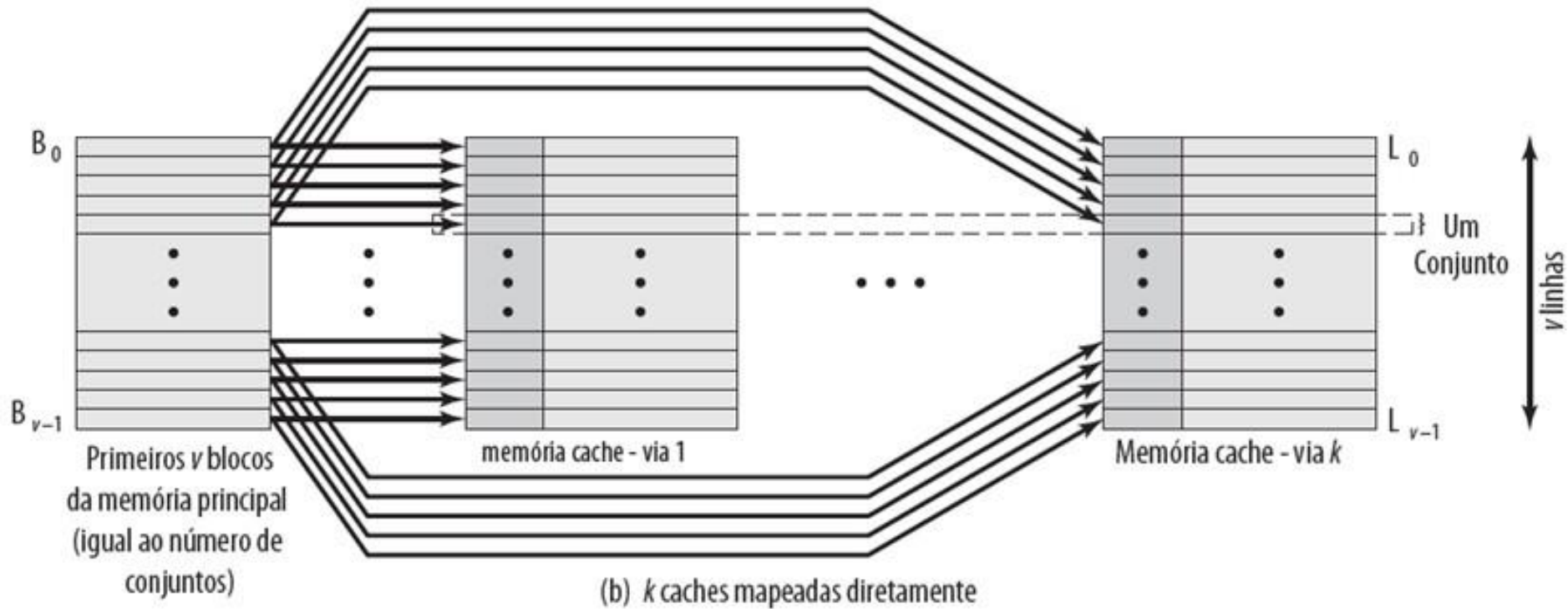
# Mapeamento associativo em conjunto

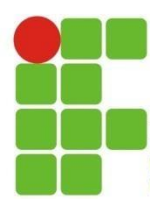


(a)  $v$  caches mapeadas associativas

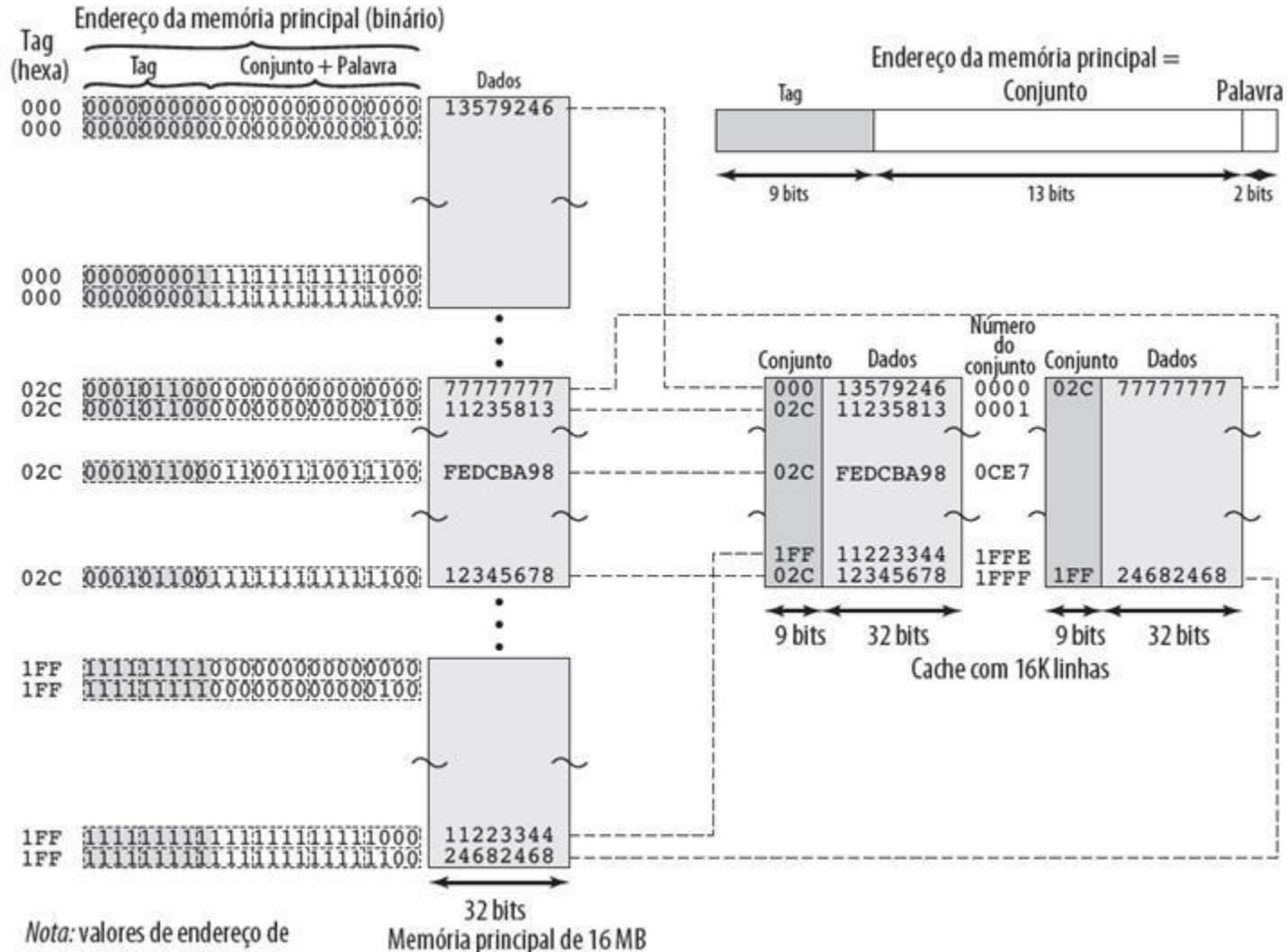


# Mapeamento associativo em conjunto





# Mapeamento associativo em conjunto (exemplo)



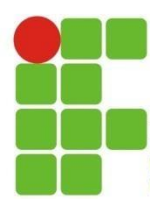
Nota: valores de endereço de memória em binário; outros, em hexadecimal.



# Algoritmos de Substituição

---

- Quando um novo bloco é trazido para a cache, um dos blocos existentes deve ser substituído.
  - No mapeamento direto, não há alternativa, pois cada bloco é mapeado em uma única linha.
  - Para os mapeamentos associativo e associativo por conjuntos, é necessário um algoritmo de substituição.
  - Recomenda-se a implementação em hardware, por questões de desempenho.
  - A seguir, os quatro algoritmos de substituição mais comuns.
-



# Algoritmos de Substituição

---

## LRU (Menos Recentemente Usado)

- *Substitui aquele bloco no conjunto que permaneceu na cache por mais tempo sem qualquer referência a ele.*
- *Implementação com bits de uso (USE).*

## FIFO (First In First Out)

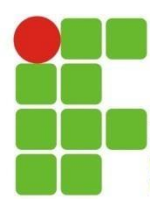
- *Substitui o bloco do conjunto que esteve na cache por mais tempo.*
- *Implementação com áreas de armazenamento circular.*

## LFU (Menos Frequentemente Usado)

- *Substitui aquele bloco no conjunto que teve menos referências.*
- *Implementação com contadores.*

## Aleatório

- *Escolhe uma linha aleatória dentre as candidatas.*
- *Segundo simulações, este método apresenta um desempenho levemente inferior aos demais.*



# Políticas de Escrita

---

- Quando um bloco que está residente na cache estiver para ser substituído, existem dois casos a considerar:
  - Se o bloco antigo na cache não tiver sido alterado, ele pode ser substituído por um novo bloco sem primeiro atualizar o bloco antigo.
  - Se pelo menos uma operação de escrita foi realizada em uma palavra nessa linha da cache, então a memória principal precisa ser atualizada antes de trazer o novo bloco.
  
- Problema encontrado:
  - A memória principal pode ser utilizada tanto por **outros processadores** quanto por **dispositivos de E/S**.

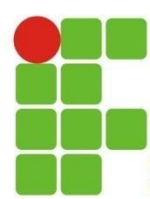


# Políticas de Escrita

---

## ❑ Escrita direta (*write through*)

- Todas as operações de escrita são feitas tanto na memória principal quanto na cache.
- Vantagem: a memória principal está sempre atualizada.
- Desvantagem: geração de tráfego de memória considerável

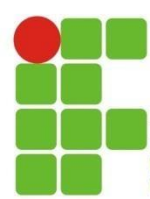


# Políticas de Escrita

---

## ❑ Escrita de volta (*write back*)

- Escritas são feitas apenas na cache.
- Quando uma linha da cache é atualizada, um bit de atualização associado a ela é setado em 1.
- Quando um bloco for substituído, ele é escrito de volta na memória apenas se o seu bit de atualização for 1.
- Vantagem: minimiza o número de operações de escrita na memória.
- Desvantagem: partes da memória principal podem ficar inválidas.



# Políticas de Escrita

---

## ❑ Escrita uma vez (write once)

- Ideal para sistemas multiprocessados com memória principal compartilhada.
  - É uma mistura de *write through* e *write back*.
  - Cada microprocessador escreve na memória principal sempre que o bloco correspondente na cache foi atualizado pela primeira vez (*write through*).
  - Os demais microprocessadores são alertados da alteração.
  - Outras alterações naquele bloco são realizadas apenas na cache local e o bloco da memória só será atualizado quando o bloco for substituído na cache (*write back*).
-

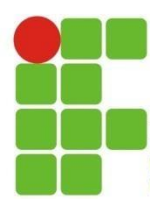




# Tamanho de bloco (linha)

---

- À medida em que esse número aumenta, aumenta inicialmente a taxa de acertos.
- Entretanto, se esse número aumentar muito, a taxa de acertos diminuirá.
- Porque não usar blocos muito grandes:
  - Para uma dada capacidade, o número de linhas diminui.
  - Cada palavra adicional estará mais distante da usada – e a chance de uso será menor.
- Tamanho ideal: 2 – 8 palavras



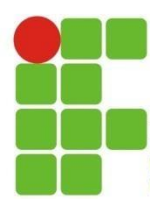
# Número de caches

---

- ❑ Anteriormente, a cache era externa ao processador.
  
- ❑ Com o avanço da eletrônica, tem-se:
  - Cache L1: interna ao processador
  - Cache L2: externa ao processador
    - ✓ Também podem ser internas e incluir uma cache L3
  
- ❑ Unificadas ou Separadas?
  - Unificadas (instruções + dados)
  - Separadas (uma para instruções e outra para os dados)
  - Projetos atuais baseiam-se em caches separadas, por motivos de desempenho junto ao pipeline

## Memórias Externas





# Tipos de Memória Externa

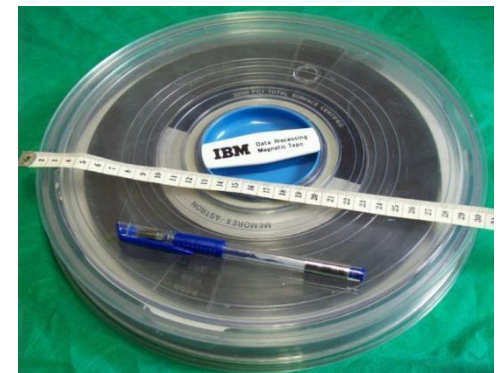
## ❑ Disco magnético:

- HD
- RAID
- Disco Flexível (Disquete)

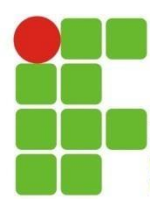


## ❑ Ótica:

- CD-ROM
- CD-Recordable (CD-R)
- CD-R/W
- DVD
- Blue-ray

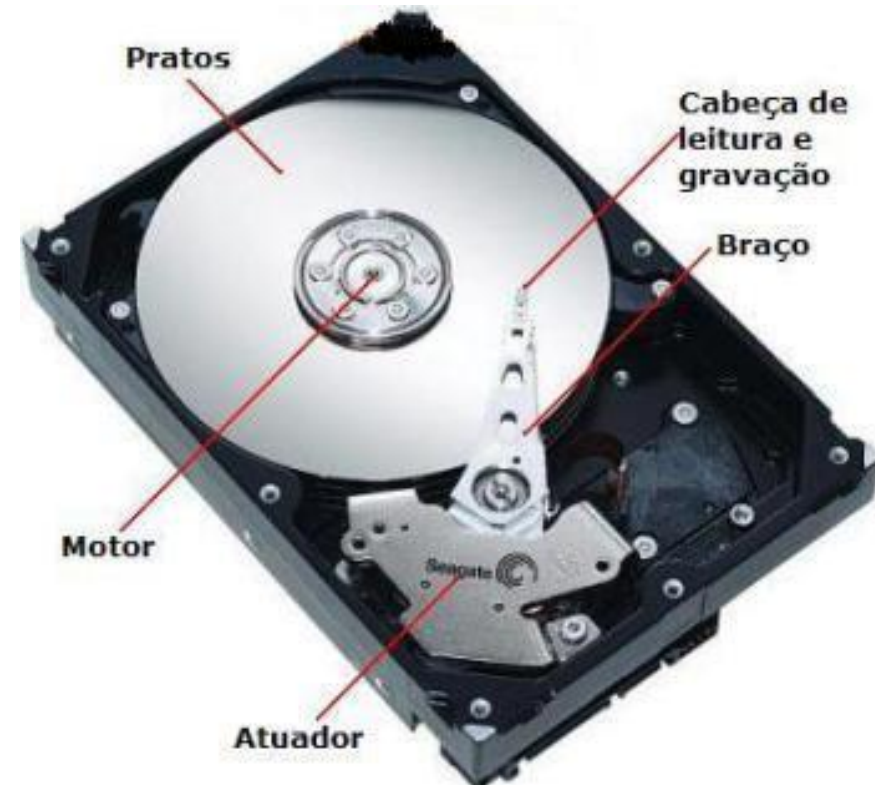


## ❑ Fita magnética



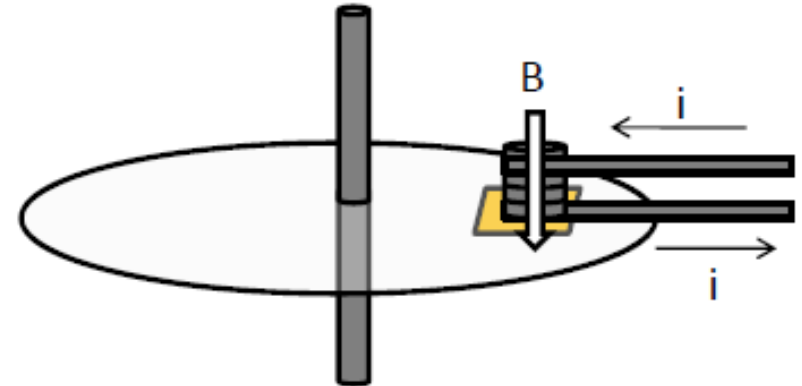
# Disco Magnético - HD

- ❑ Prato circular construído de material não magnético (substrato), coberto com um material magnetizável.
- ❑ Substrato feito de alumínio, liga de alumínio ou, mais recentemente, de vidro.
- ❑ Alguns benefícios do substrato de vidro:
  - Maior uniformidade da superfície  
*Aumenta confiabilidade*
  - Redução nos defeitos da superfície  
*Erros reduzidos de leitura/gravação*
  - Melhor rigidez.
  - Maior resistência a choques e danos.

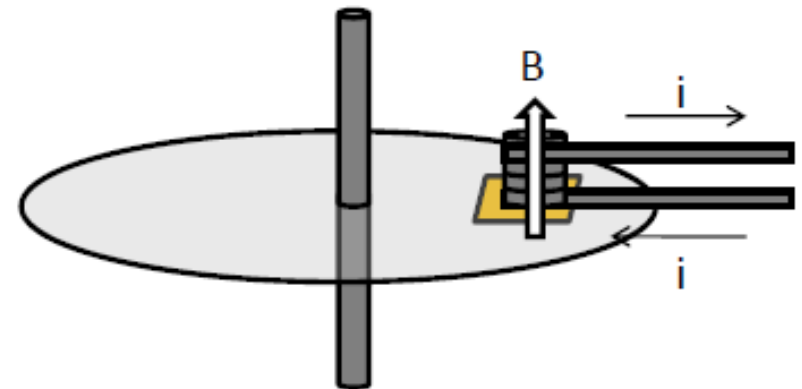


# Disco Magnético - HD

- ❑ Os discos são feitos de material plástico ou metálico, coberto por material magnetizante.
- ❑ Os cabeçotes são, na verdade, bobinas.
- ❑ Escrita:
  - Correntes em sentidos opostos criam padrões magnéticos distintos.
  - Esses padrões são usados para se representar "1"s e "0"s.



Bit "1" gravado

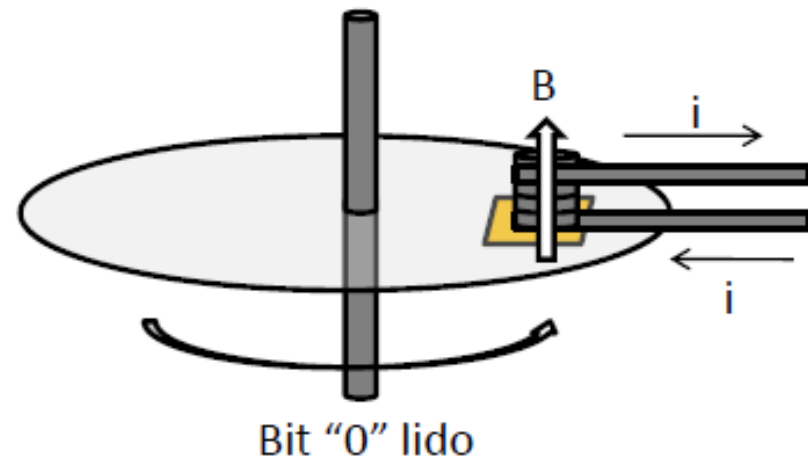
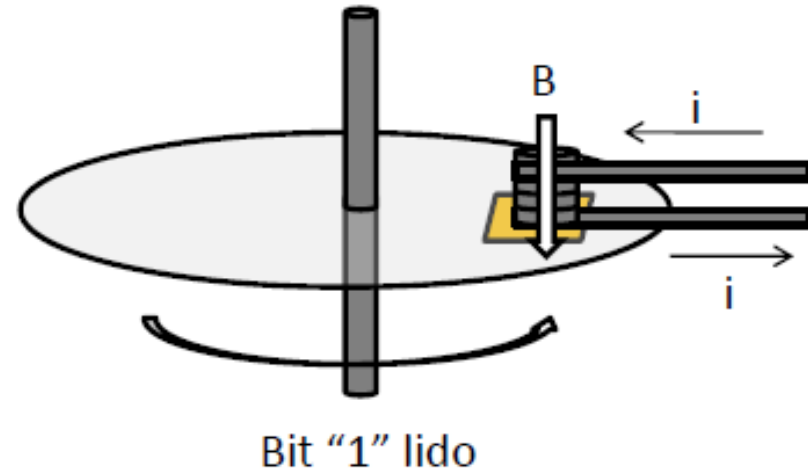


Bit "0" gravado

# Disco Magnético - HD

## Leitura:

- Baseia-se no fato de que um campo magnético que se move em relação a uma bobina induz uma corrente nessa bobina.

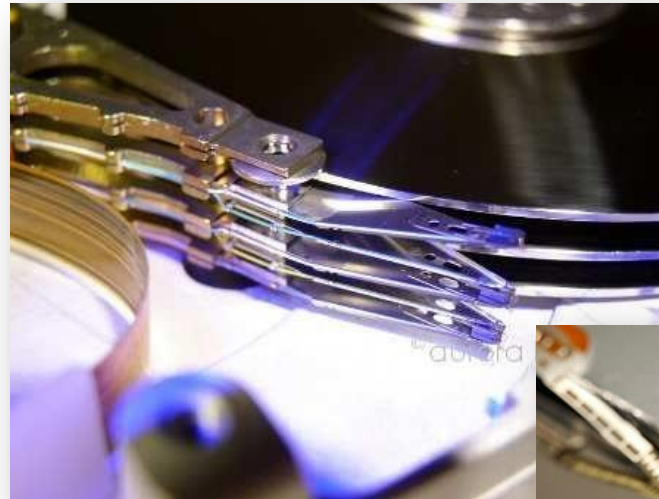
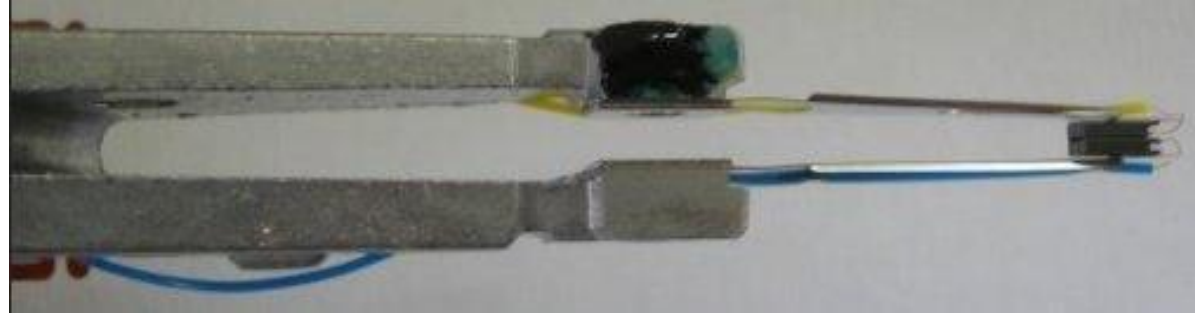






# Disco Magnético - Escrita

- ❑ A cabeça de gravação é feita de material facilmente magnetizável, em forma de anel com um espaço (gap) de um lado e um fio condutor (com algumas voltas) do lado oposto.
- ❑ Uma corrente elétrica no fio induz um campo magnético no espaço, que magnetiza uma pequena área do meio de gravação.





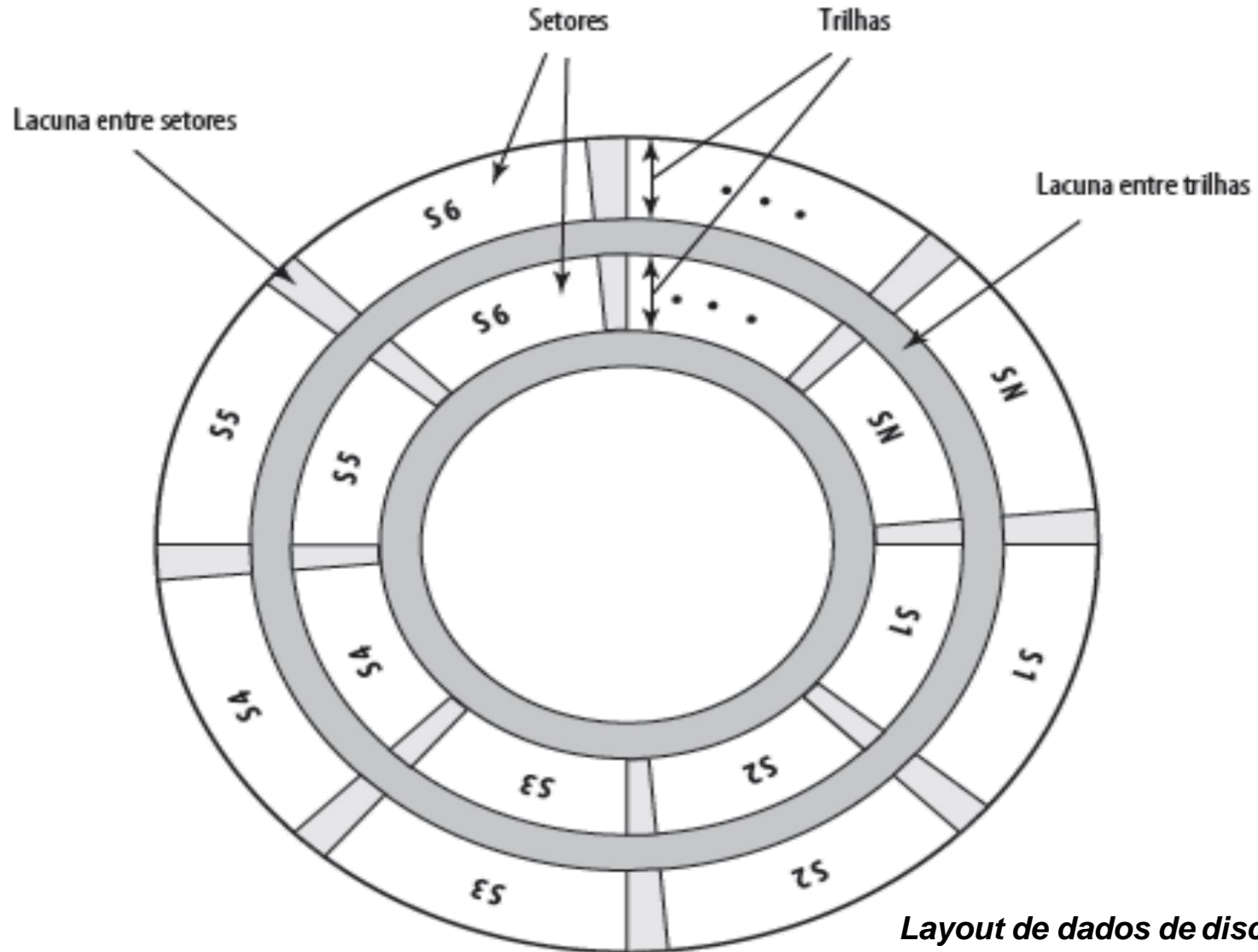


# Organização e Formatação de Dados

---

- A cabeça é um dispositivo pequeno capaz de ler e escrever em uma parte do prato girando por baixo dela.
- Organização dos dados no prato em um conjunto concêntricos de anéis (**trilhas**).
- As trilhas adjacentes são separadas por **lacunas** e existem milhares de trilhas por superfície.
- Os dados são transferidos de e para o disco em **setores**, que pode ser variável, mas normalmente é fixo em 512 bytes.
- Existem centenas de setores por trilhas.
- Para evitar requisitos de precisão excessivos no sistema, setores adjacentes são separados por **lacunas**.

# Organização e Formatação de Dados



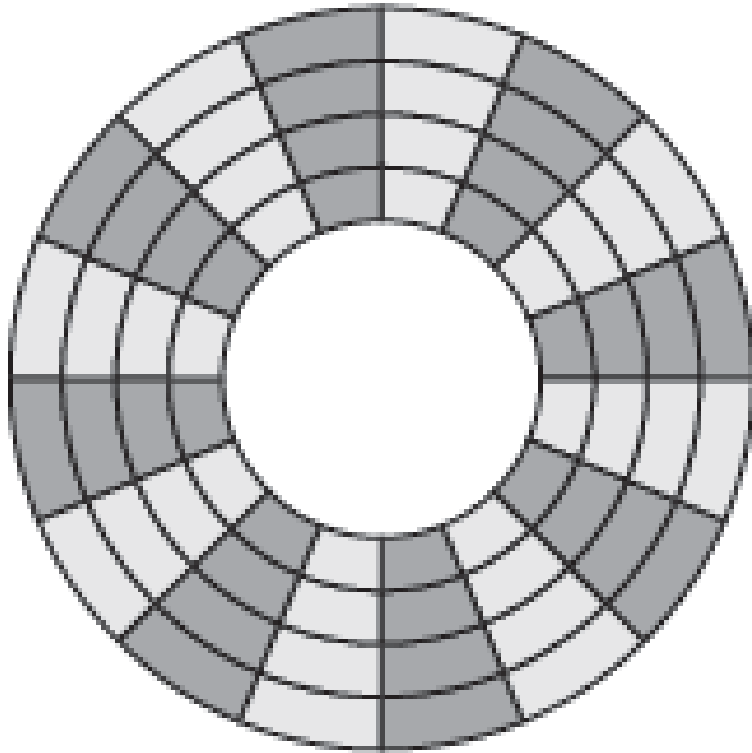


# Organização e Formatação de Dados

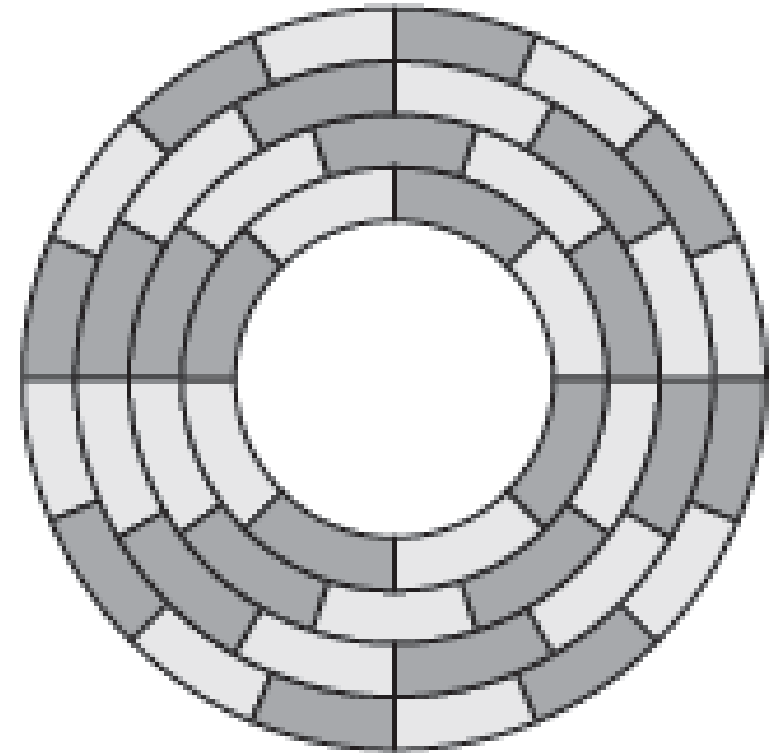
---

- Para que a cabeça de leitura possa ler todos os bits (de qualquer trilha) na mesma velocidade, pode-se aumentar o espaçamento entre os bits de informação gravados nos segmentos do disco.
  
- A informação pode ser varrida com a mesma taxa, girando o disco em uma velocidade fixa (**velocidade angular constante** ou CAV).
  
- Velocidade angular constante
  - Setores em forma de fatia de torta e trilhas concêntricas
  - Trilhas e setores individuais endereçáveis
  - Perda de espaço nas trilhas externas – menor densidade de dados
  
- Pode usar **gravação em múltiplas zonas** para aumentar capacidade
  - Cada zona tem número fixo de bits por trilha
  - Circuito mais complexo

# Organização e Formatação de Dados



**(a) Velocidade angular constante**

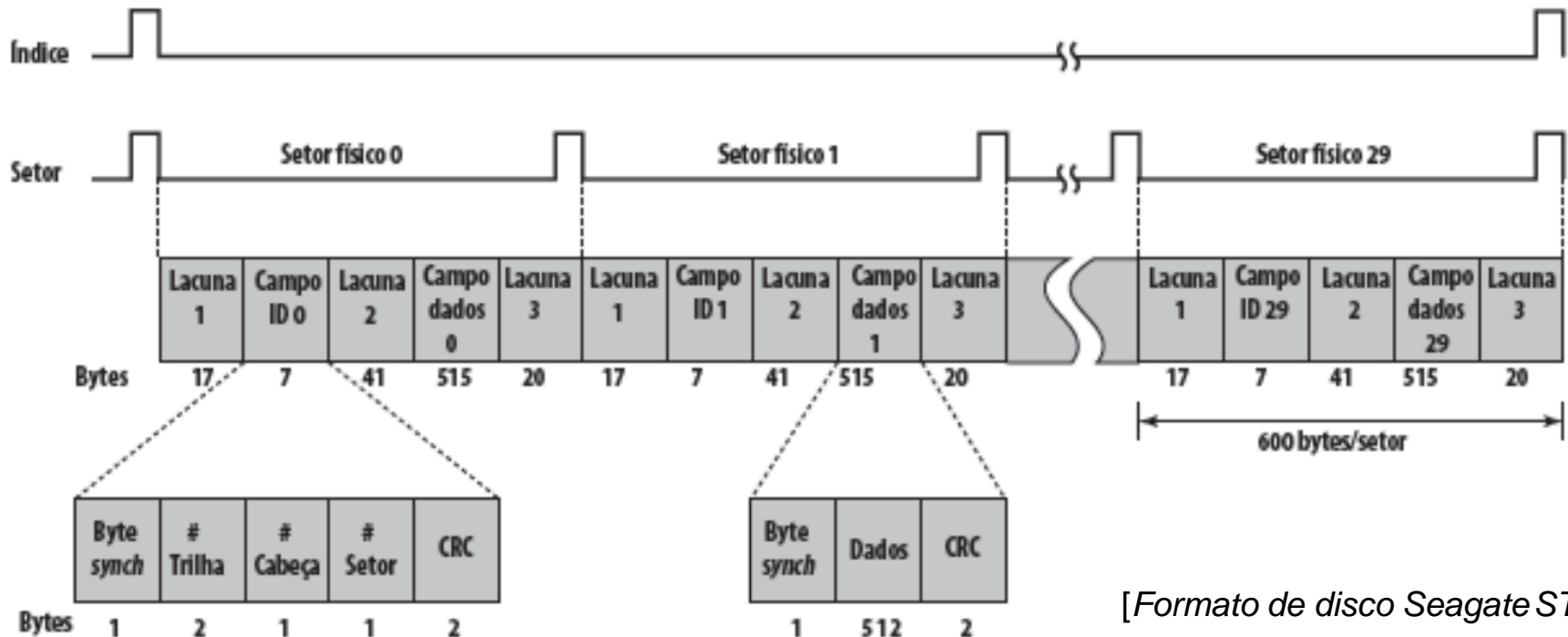


**(b) Gravação em múltiplas zonas**

***Comparação de métodos de layout de disco***

# Localizando setores

- ❑ Deve ser capaz de identificar início da trilha e setor.
- ❑ Formatar disco:
  - Informações adicionais não disponíveis ao usuário
  - Marca trilhas e setores

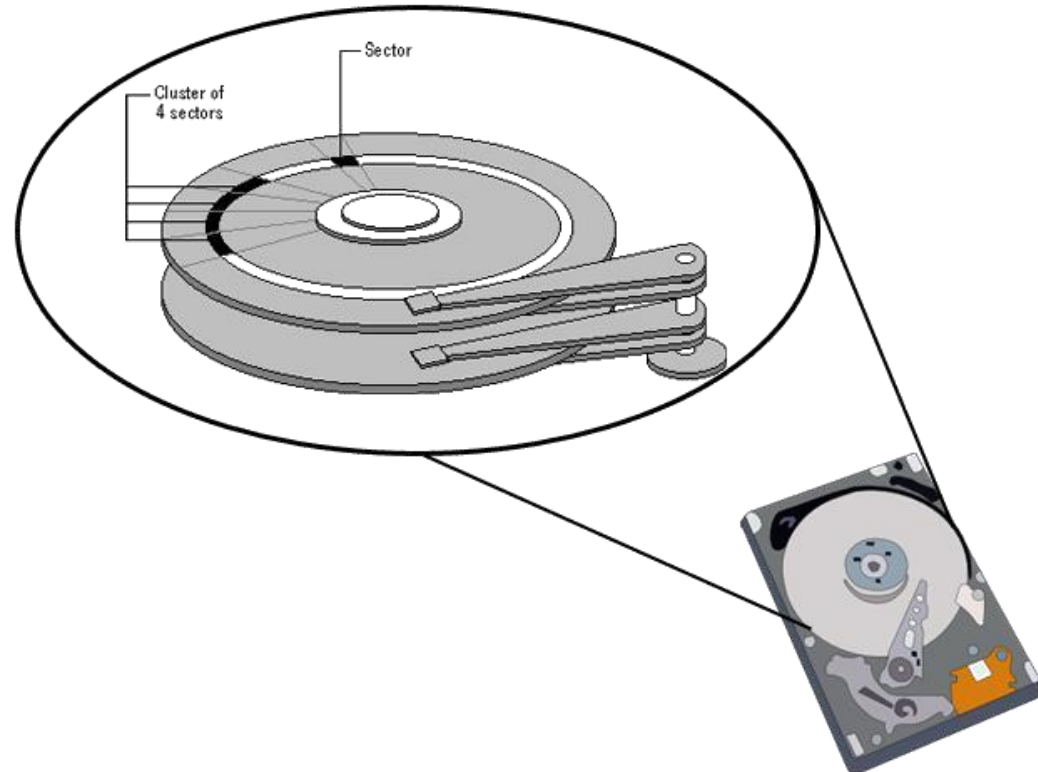


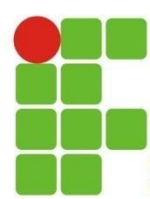
[Formato de disco Seagate ST506]



# Clusters

- ❑ O disco agrupa os setores numa entidade chamada cluster.
- ❑ O cluster é a menor unidade de espaço alocável em disco.
- ❑ Quanto maior a capacidade do disco, maior será o tamanho do cluster, isto é, mais setores serão chamados a formá-lo.
- ❑ Não importa quão pequeno seja um arquivo, ele sempre usará um cluster inteiro para ser armazenado.

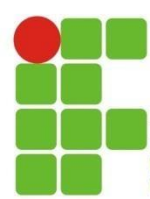




# Clusters

- ❑ **Exemplo:** arquivo de 290 bytes armazenado num disco que possui clusters formados por 16 setores.
  - Tamanho do cluster (em bytes):  $16 \times 512 \text{ bytes} = 8.182 \text{ bytes}$
  - Assim, 7.892 bytes não serão utilizados nem estarão disponíveis para armazenar outro arquivo.

ESPAÇO MÍNIMO OCUPADO POR UM ARQUIVO NO DISCO			
Capacidade do Disco (MB)	Bytes por setor	Nº de setores por cluster	Espaço mínimo ocupado por um arquivo (KB)
1,44 (disquete)	512	1	0,5
Até 256	512	8	4
De 256 a 512	512	16	8
De 512 a 1024	512	32	16
Acima de 1024	512	64	32



# Características físicas

<b>Movimento da Cabeça</b>	<b>Pratos</b>
Cabeça fixa (uma por trilha)	Único prato
Cabeça móvel (uma por superfície)	Múltiplos pratos
<b>Portabilidade do disco</b>	<b>Mecanismo da cabeça</b>
Disco não removível	Contato (disquete)
Disco removível	Lacuna fixa
	Lacuna aerodinâmica (Winchester)
<b>Faces</b>	
Única face	
Dupla face	

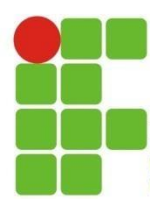




# Discos RAID

---

- ❑ RAID = Redundant Array of Independent Disks
- ❑ Subsistema de armazenamento utilizado em *servidores* e *storages de dados* que utiliza dois ou mais discos rígidos para construir um conjunto lógico, abstraído da aplicação ou do usuário, visando a segurança e/ou performance do sistema.
- ❑ Estratégia criada por pesquisadores de Berkeley
- ❑ Há vários tipos de RAID, promovendo vantagens como:
  - Redundância
  - Aumento na *taxa de transferência*
  - Aumento na *taxa de processamento de solicitações*



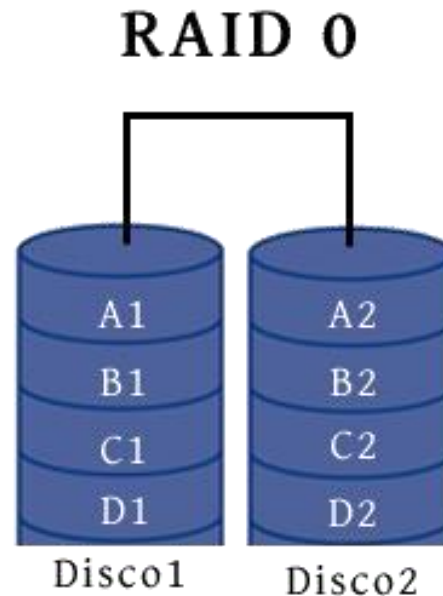
# Níveis RAID

---

- Cada diferente combinação de instruções é denominada “nível RAID”
  - Cada nível emprega as operações de I/O nos discos de acordo com a necessidade do equipamento.
  - Alguns níveis RAID tem como objetivo prover uma camada de segurança ao sistema.
  - Outros são utilizados para aumentar performance de leitura e escrita dos discos,
  - Existem níveis RAID que podem somar ambas as opções anteriores.
-

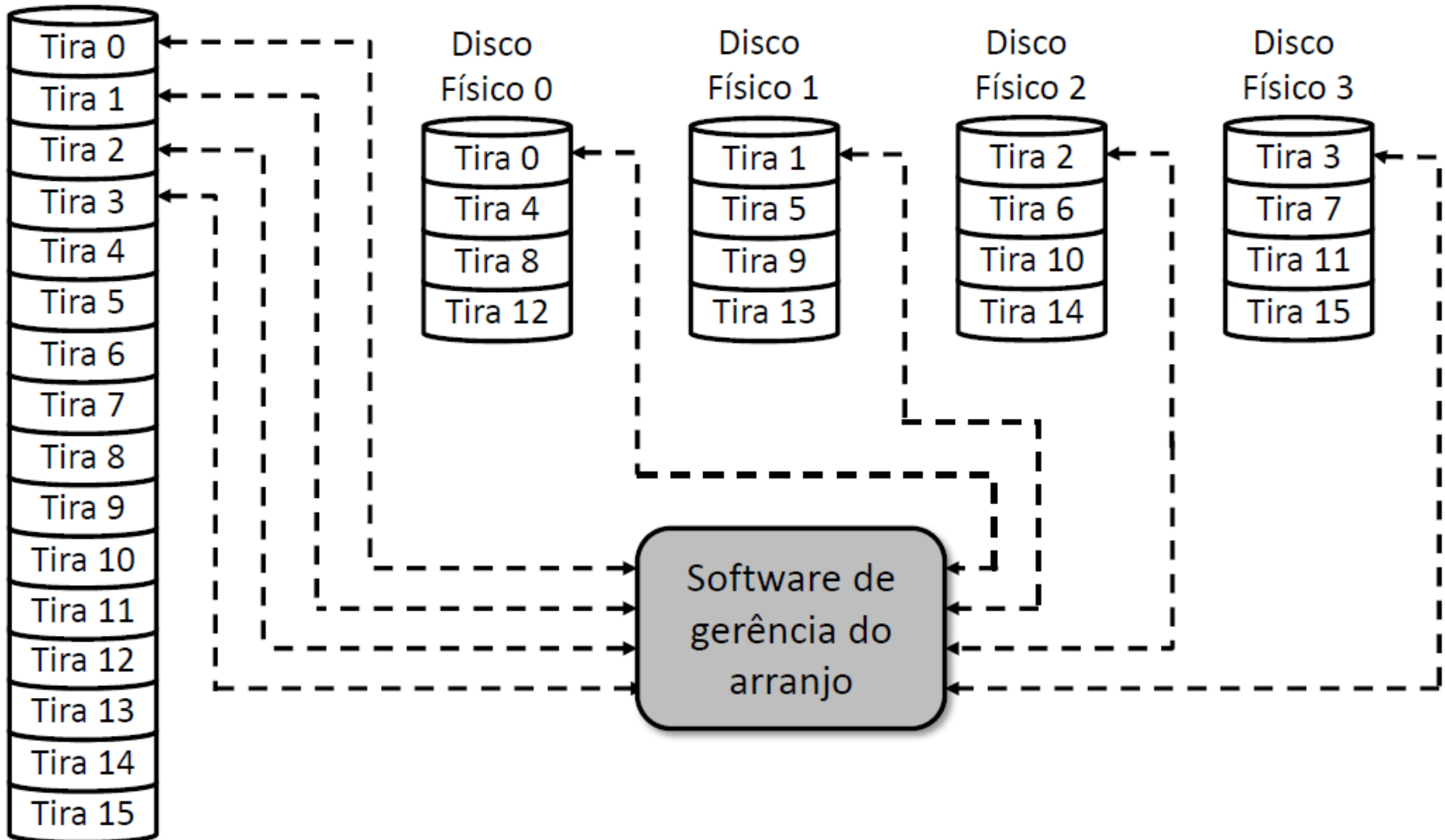
# RAID 0 (espalhamento de dados)

- ❑ Não redundante.
- ❑ Necessita de dois ou mais discos.
- ❑ Os blocos de dados são divididos entre os discos, aumentando consideravelmente a performance de leitura e escrita do conjunto RAID.
- ❑ Contudo, em caso de perda de um dos discos, haverá perda total dos dados.



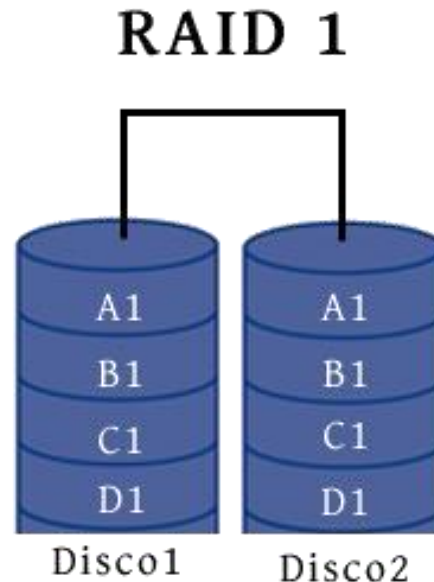
# RAID 0 (espalhamento de dados)

Disco lógico



# RAID 1 (replicação de dados)

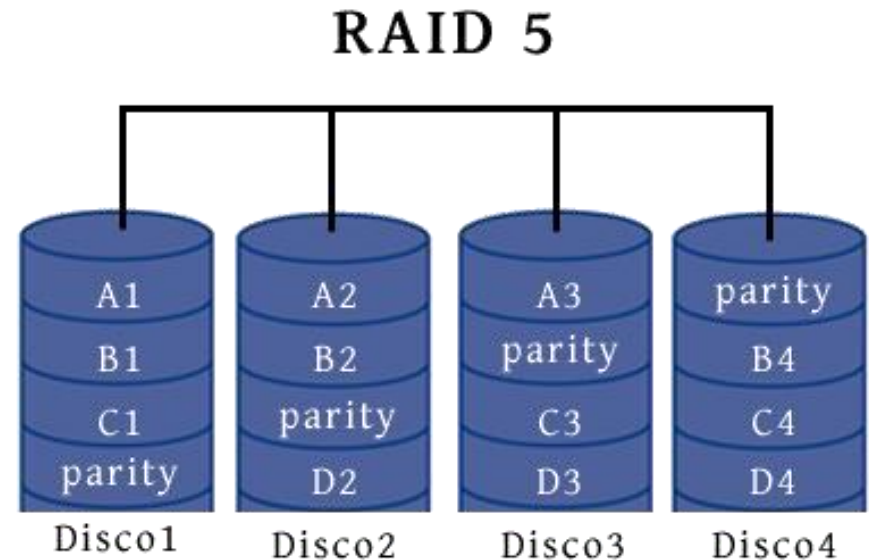
- ❑ Necessita de no mínimo 2 discos.
- ❑ É basicamente o espelhamento dos discos: os dados gravados no disco 1 serão automaticamente gravados no disco 2.
- ❑ Em caso de perda de um dos discos, os dados permanecem íntegros.
- ❑ Contudo, esta solução consome o dobro de discos para armazenamento.





# RAID 5 (paridade distribuída)

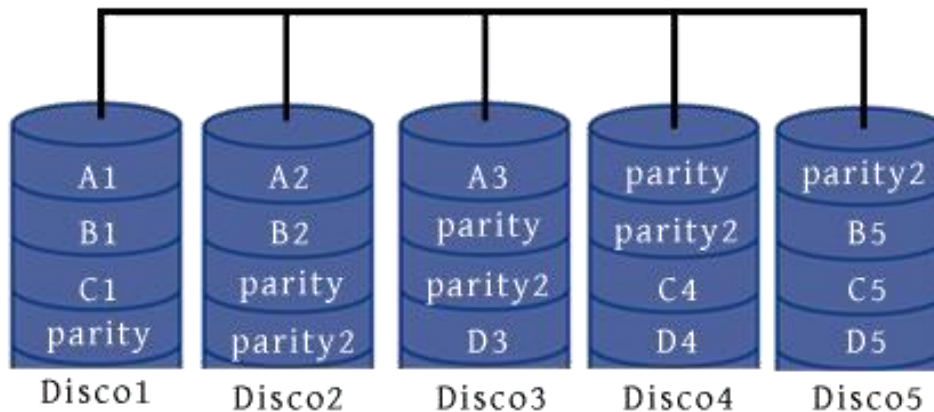
- ❑ Necessita de no mínimo 3 discos.
- ❑ Os blocos de dados são distribuídos entre os discos, aumentando a performance de leitura e escrita.
- ❑ Há ainda dados de paridade distribuídos também entre os discos, gerados por um algoritmo que permite em caso de perda de um dos discos, que os dados permaneçam íntegros.
- ❑ É uma opção que soma segurança e performance. Porém, todo arranjo RAID 5 com 3 ou até 21 discos irá consumir o equivalente a um dos discos para paridade.



# RAID 6 (redundância dupla)

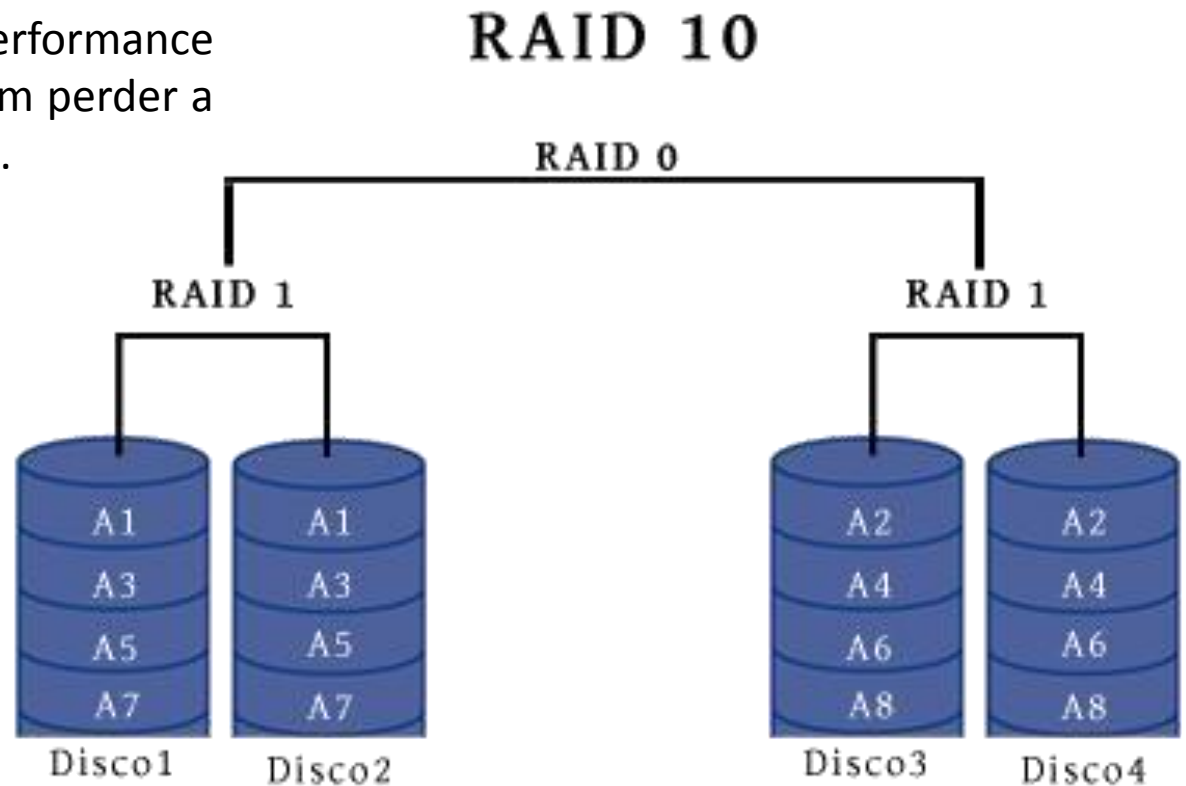
- ❑ Necessita de no mínimo 4 discos.
- ❑ Similar ao RAID 5, contudo os dados de paridade são duplicados.
- ❑ Este nível RAID permite a perda de até dois discos por arranjo, sem perda de dados.
- ❑ Assim como o nível RAID 5, o nível RAID 6 soma segurança e performance, porém, consumindo 2 discos para paridade por arranjo.

## RAID 6



# RAID 10 (RAID 1 + RAID 0)

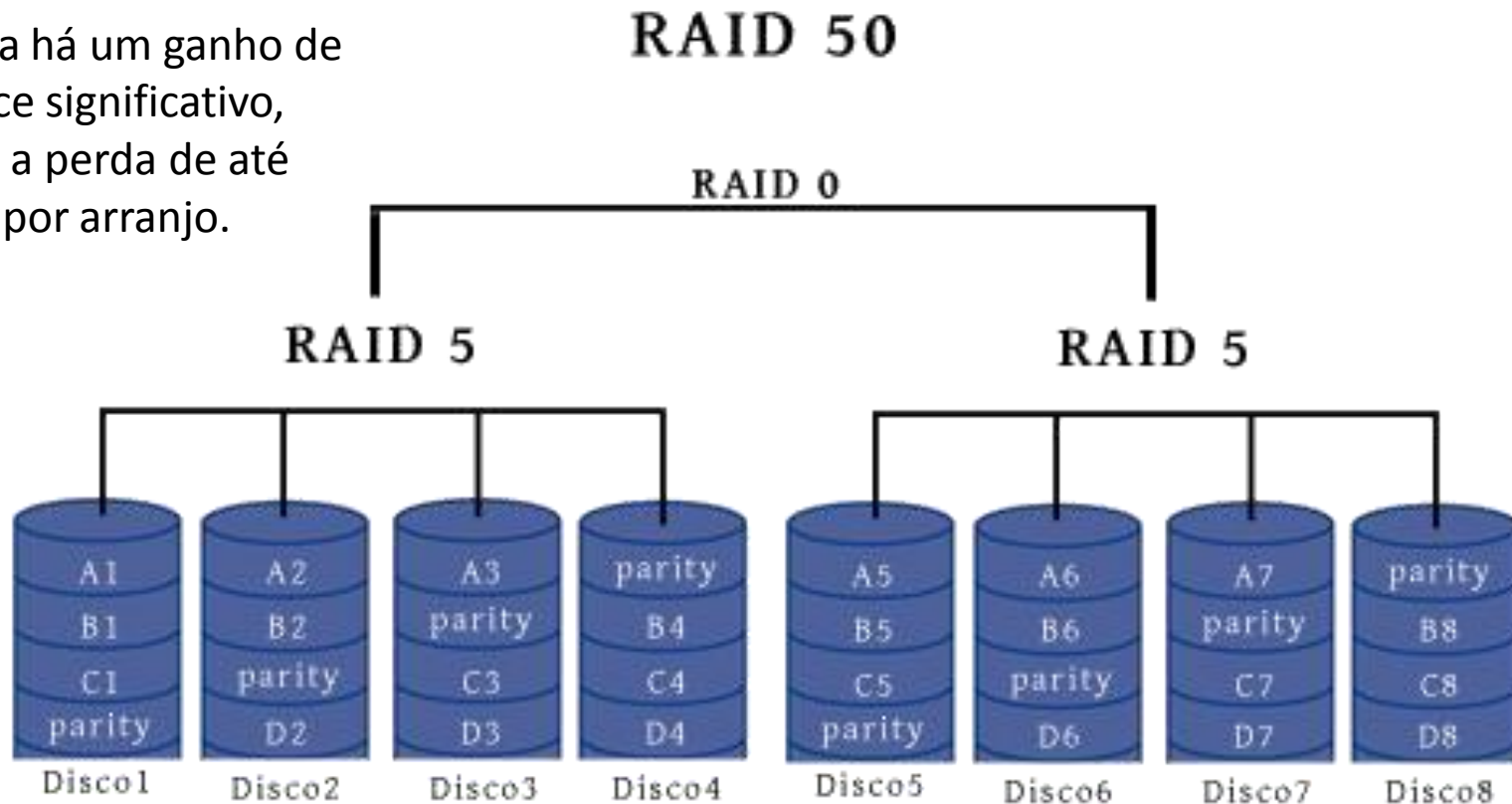
- ❑ Necessita de no mínimo 4 discos.
- ❑ Os blocos de dados são divididos (RAID 0) em dois pares de discos realizando o RAID 1 entre si.
- ❑ Com isto, aumenta-se a performance do servidor ou storage, sem perder a segurança do arranjo RAID.





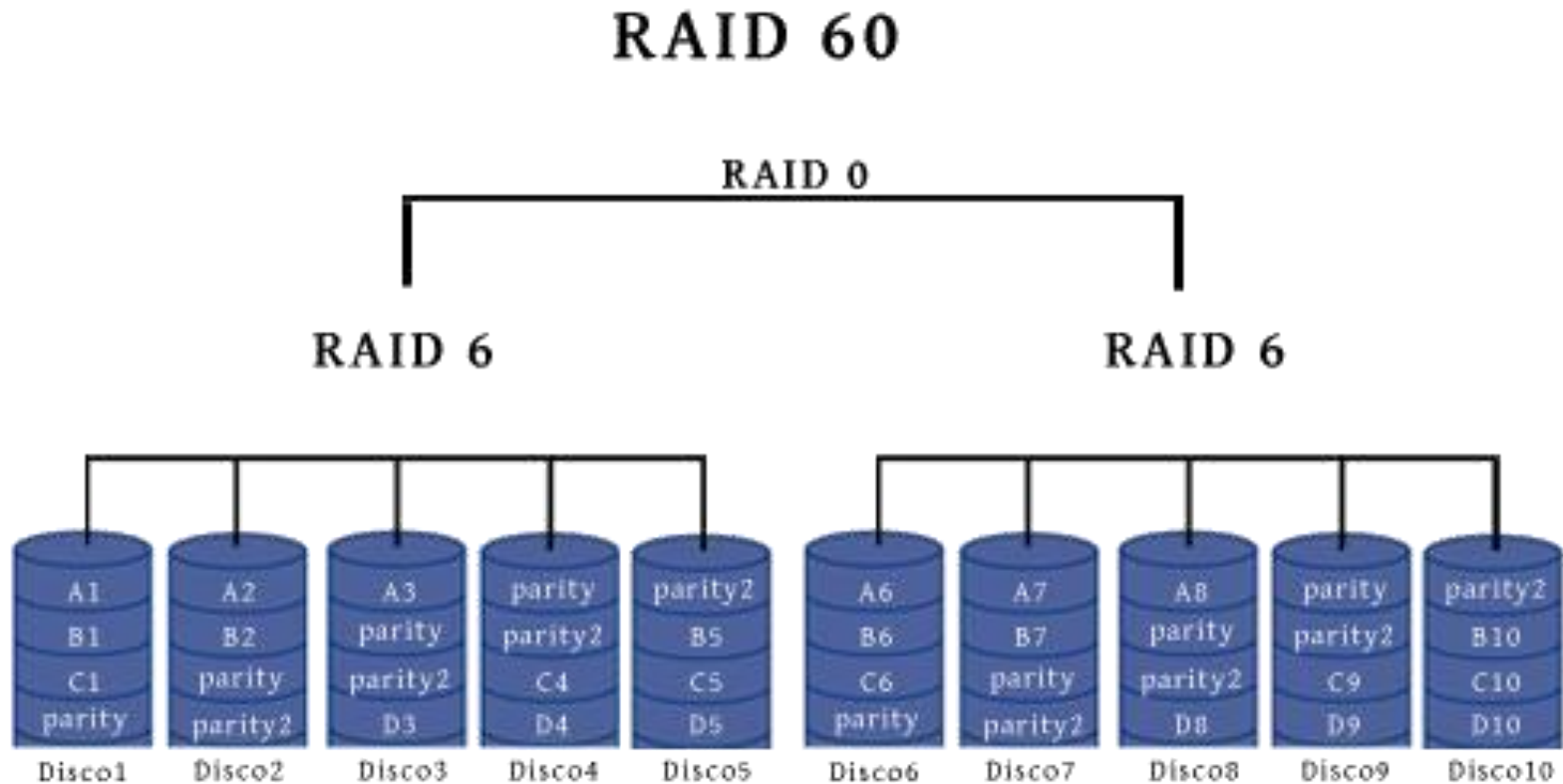
# RAID 50 (RAID 5 + RAID 0)

- ❑ Necessita de no mínimo 6 discos, mas geralmente é utilizado em arranjos com muitos discos.
- ❑ O RAID 50 divide os dados em dois (ou mais) conjuntos em RAID 5.
- ❑ Desta forma há um ganho de performance significativo, permitindo a perda de até dois discos por arranjo.



# RAID 60 (RAID 6 + RAID 0)

- Similar ao RAID 50, porém, os dados após serem divididos são armazenados em arranjos RAID 6.

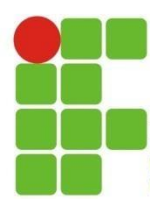




# Discos Óticos

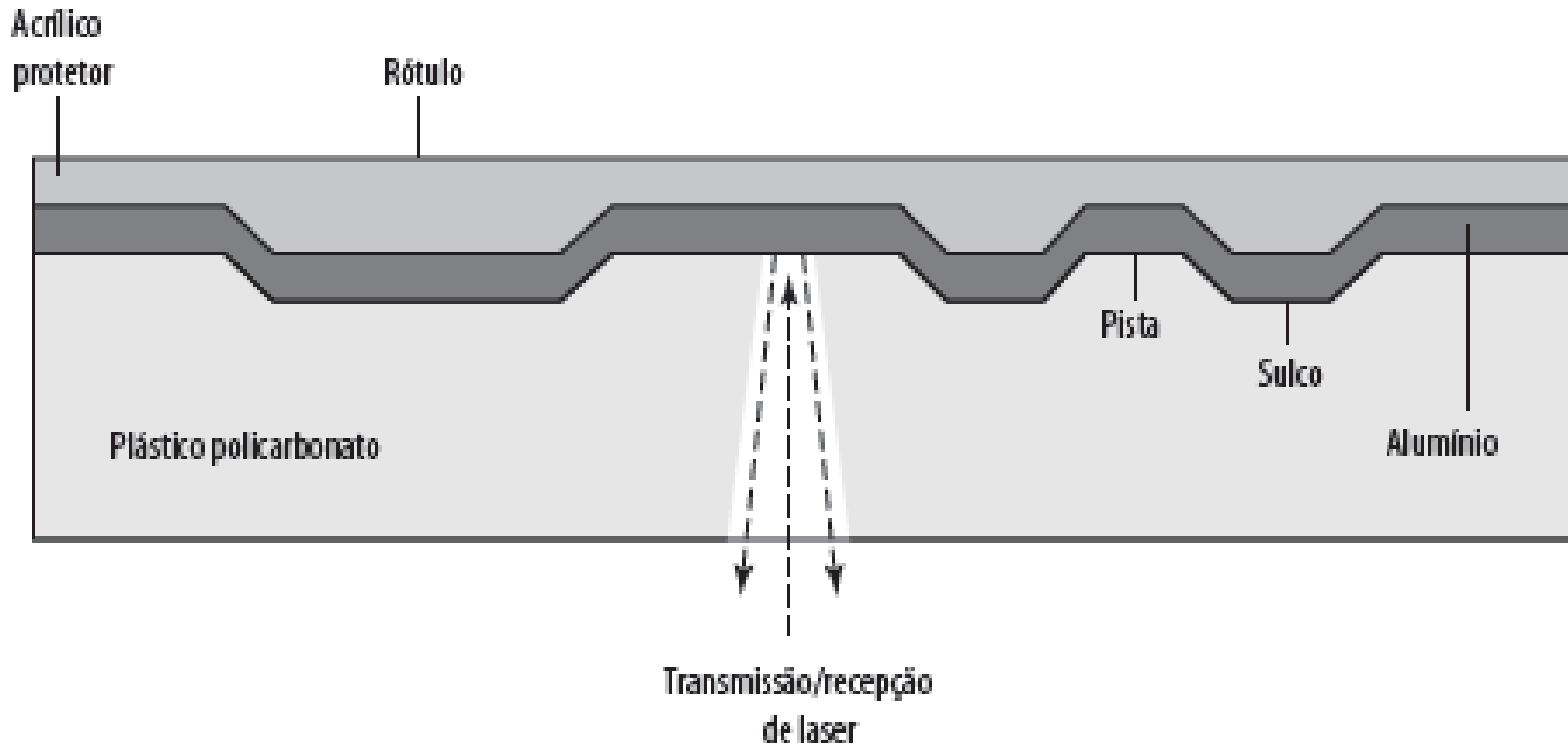
---

- ❑ 1983 → sistema de áudio digital de disco compacto.
  
- ❑ CD-ROM:
  - 650 Mb, gerando mais de 70 minutos de áudio.
  
  - Mesmo processo de fabricação do CD de áudio.
    - ✓ Disco fabricado em policarbonato, revestido com uma superfície de alto índice de reflexão (alumínio, tipicamente).
  
    - ✓ Informações (dados ou músicas) gravadas na superfície como uma série de sulcos microscópicos.
  
    - ✓ Superfície sulcada é protegida contra pó e arranhões por uma cobertura de laca ou verniz clara .
  
  - Dispositivo de leitura mais resistente e possui mecanismos de correção de erro.

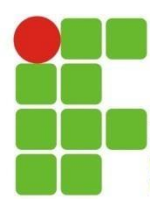


# Mecanismo de leitura

- ❑ Fotosensor examina a superfície em intervalos regulares.



- ❑ Começo ou fim de sulco indicam “1”.
- ❑ Nenhuma mudança na elevação indica “0”.



# Layout do disco ótico

---

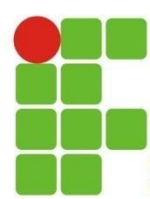
- ❑ **Premissa básica:** os setores são lidos a uma taxa constante.
  
- ❑ CLV (Velocidade Linear Constante)
  - Dados distribuídos no disco em setores de mesmo tamanho
  - Capacidade e atraso rotacional são maiores para as trilhas mais externas
  - Armazenamento mais eficiente
  - Necessário ajuste na velocidade



# CD-RW

---

- Usa um material que tem dois índices de reflexão diferentes, em dois estados distintos de fase:
    - **Estado amorfo:** moléculas exibem uma orientação aleatória e que reflete pouca luz;
    - **Estado cristalino:** possui uma superfície suave com boa reflexão de luz.
  
  - Um feixe de luz laser pode alterar o material de um estado para outro.
  
  - Material eventualmente perde de maneira permanente essa propriedade.
-

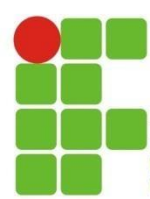


# DVD

- ❑ Bits são mais próximos aos outros, quando comparado a um CD.

	CD	DVD
Espaço entre loops na espiral	1,6 $\mu\text{m}$	0,74 $\mu\text{m}$
Distância mínima entre buracos	0,834 $\mu\text{m}$	0,4 $\mu\text{m}$

- ❑ Laser possui comprimento de onda mais curto.
- ❑ Aumento de aproximadamente 7 vezes na capacidade de armazenamento ( $\approx 4,7$  GB) quando comparado a um CD ( $\approx 700$  MB).



# DVD de Camada Dupla

---

- Utiliza uma segunda camada (semi-refletora) de pistas e sulcos, acima da primeira camada (completamente refletora).
- Laser pode ajustar o foco e atingir cada camada separadamente.
- Baixa refletividade da segunda camada faz com que a capacidade total ( $\approx 8,5$  GB) não seja rigorosamente o dobro da anterior.
- O DVD pode ainda ter dois lados.
  - Enquanto isso, os dados são gravados em apenas um lado do CD.



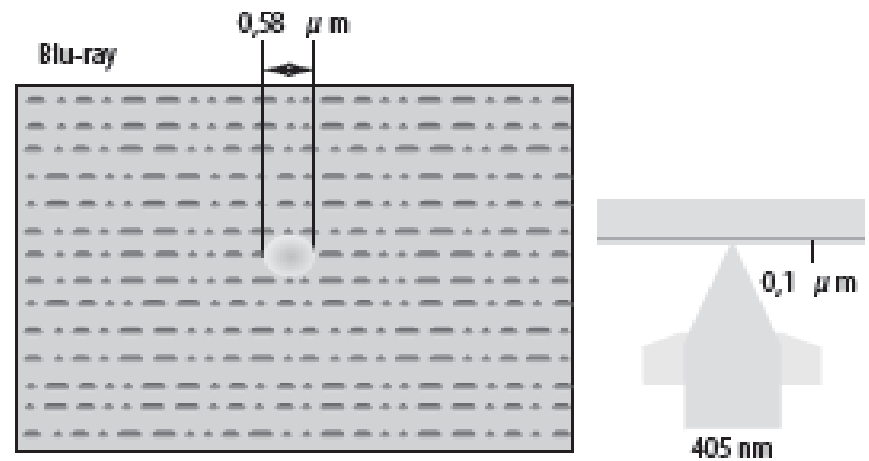
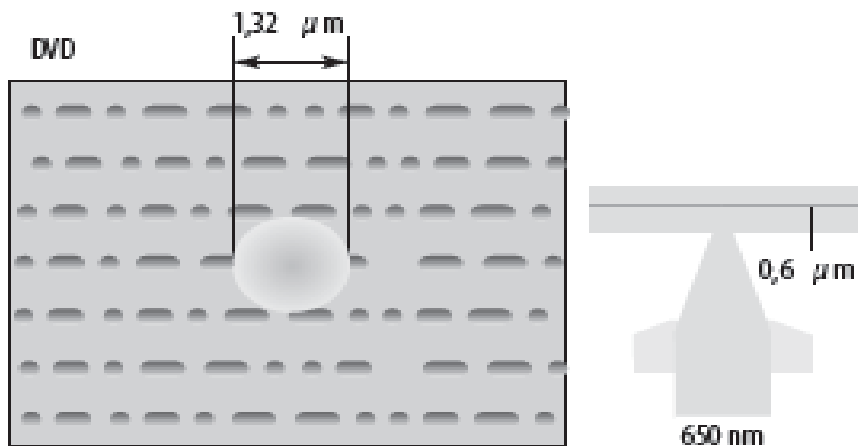
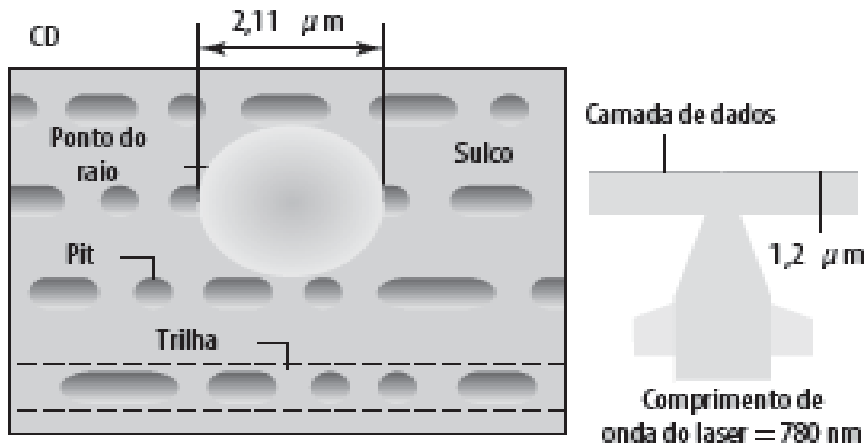


# Discos Óticos de alta definição

---

- ❑ Projetados para vídeos de alta definição e com capacidade muito mais alta que o DVD.
  - Laser com comprimento de onda mais curto
  - Sulcos menores.
  
- ❑ HD-DVD:
  - 15 GB de único lado, única camada.
  
- ❑ Blue-ray:
  - Camada de dados mais próxima do laser.
    - ✓ Foco mais estreito, menos distorção, sulcos menores.
  - 25 GB em única camada.
  - Disponível para apenas leitura (BD-ROM), regravável uma vez (BR-R) e re-regravável (BR-RE).

# Características da memória ótica





# Comparação dos discos óticos

---

## CD (Compact Disk)

- *Disco não apagável que armazena informações de áudio digitalizadas.*
- *O sistema padrão utiliza discos de 12 cm e pode gravar mais de 60 minutos de tempo de execução sem interrupções.*

## CD-ROM (CD Read-Only Memory)

- *Disco não apagável para armazenar dados de computador.*
- *O sistema padrão utiliza discos de 12 cm e pode manter mais de 650 Mb.*

## CD-R (CD Recordable)

- *Semelhante a um CD-ROM.*
  - *O usuário pode gravar no disco apenas uma vez.*
-

# Comparação dos discos óticos

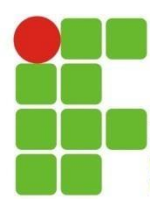
---

## CD-RW (CD Recordable / Rewriteable)

- *Semelhante a um CD-ROM.*
- *O usuário pode apagar e regravar no disco várias vezes.*

## DVD (Digital Versatile Disk)

- *Tecnologia para produzir representação digitalizada e compactada de informação de vídeo, além de grandes volumes de outros dados digitais.*
- *São usados diâmetros de 8 a 12 cm, com uma capacidade de dupla face chegando até 17 Gb.*
- *O DVD básico é somente de leitura (DVD-ROM).*



# Comparação dos discos óticos

---

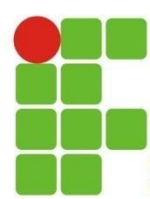
- DVD-R (DVD Recordable)
  - *Semelhante a um DVD-ROM.*
  - *O usuário pode gravar no disco apenas uma vez.*
  - *Só podem ser usados discos de uma face.*
  
- DVD-RW (DVD Recordable / Writeable)
  - *Semelhante a um DVD-ROM*
  - *O usuário pode apagar e regravar no disco várias vezes.*
  - *Só podem ser usados discos de uma face.*
  
- Blue-ray DVD
  - *Disco de vídeo de alta definição.*
  - *Oferece densidade de armazenamento de dados muito maior que o DVD.*
  - *Uma única camada em uma única face pode armazenar 25 Gb.*



# Fitas Magnéticas

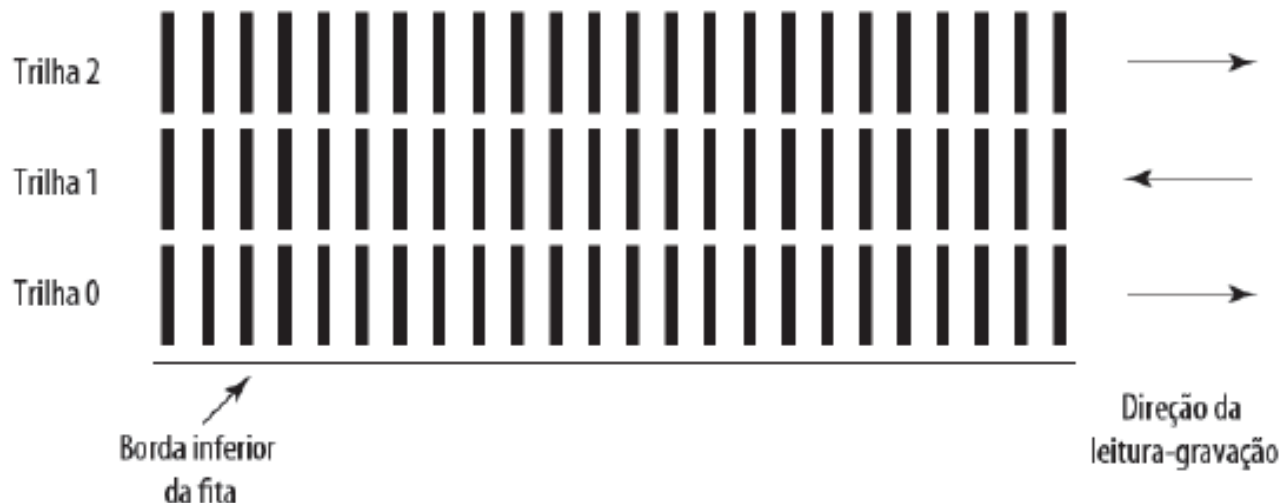
- ❑ Fita coberta com óxido magnético, sendo utilizada para registro de informações analógicas ou digitais, incluindo áudio, vídeo e dados de computador.
- ❑ Foi o primeiro tipo de memória secundária.
- ❑ Ainda é utilizada como o elemento de menor custo e menor velocidade da hierarquia de memória.





# Fitas Magnéticas

- ❑ A maioria dos sistemas modernos usa a gravação serial em que os dados são dispostos com uma sequência de bits ao longo de cada trilha.
- ❑ Gravação serpentina
  - O primeiro conjunto de bits é gravado ao longo de toda a extensão da fita.
  - Quando o final é alcançado, as cabeças são reposicionadas para gravar uma nova trilha, e a fita é novamente gravada em sua extensão, agora na direção oposta.



# Bibliografia

---

Esta apostila foi desenvolvida pelo Prof. Givanaldo Rocha de Souza do Instituto Federal do Rio Grande do Norte, tendo em partes, acréscimo de conteúdo feita pelo Prof. Denis Diniz do curso Técnico em Informática do Instituto Federal do Sul de Minas, campus Muzambinho Centro de Referencia Capetinga MG.

<[www.infowester.com.br](http://www.infowester.com.br)>