Arquitectura de Computadores

Hierarquia de Memória; Memória Cache (13.2 e 13.3)

José Monteiro

Licenciatura em Engenharia Informática e de Computadores

Departamento de Engenharia Informática (DEI)
Instituto Superior Técnico

11 de Maio, 2009

Sumário da Aula

- hierarquia de memória
  - objectivos
  - princípio da localidade
- memória cache
  - funcionamento básico
  - tipos de memória cache
  - bloco da cache
  - política de substituição

Memória Ideal

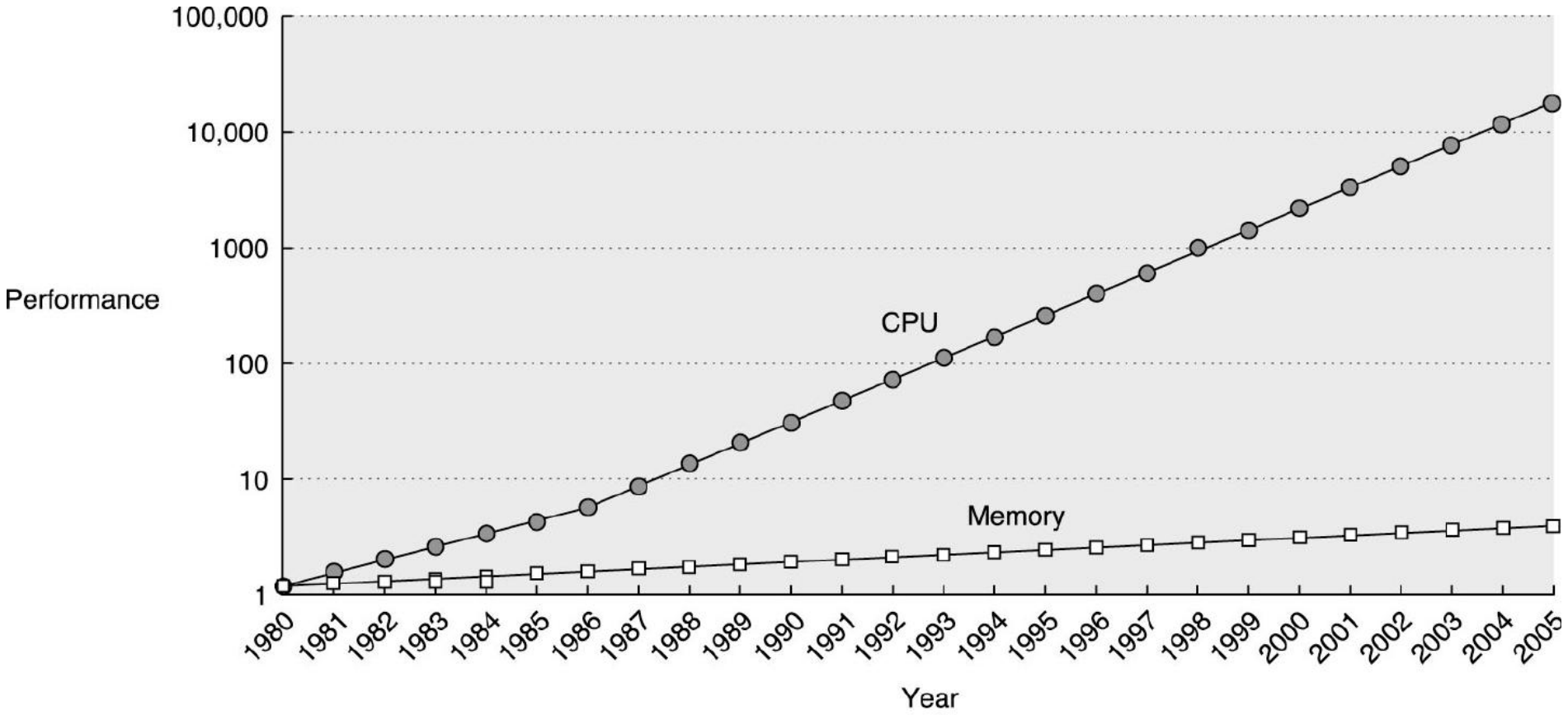
Características desejáveis para a memória:

Memória Ideal

Características desejáveis para a memória:

- barata
- grande capacidade
- rápida (tempo de acesso reduzido)
- largura de banda elevada

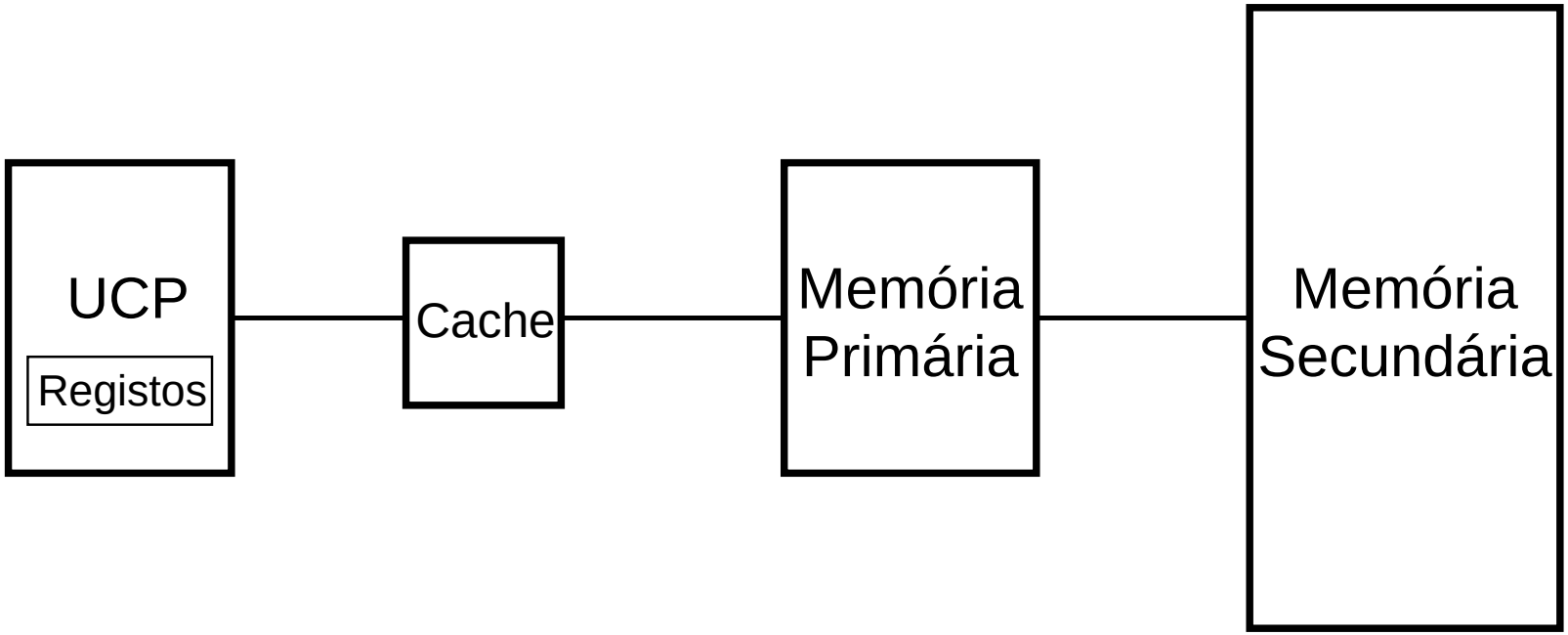
Evolução do Desempenho: CPU vs Memória



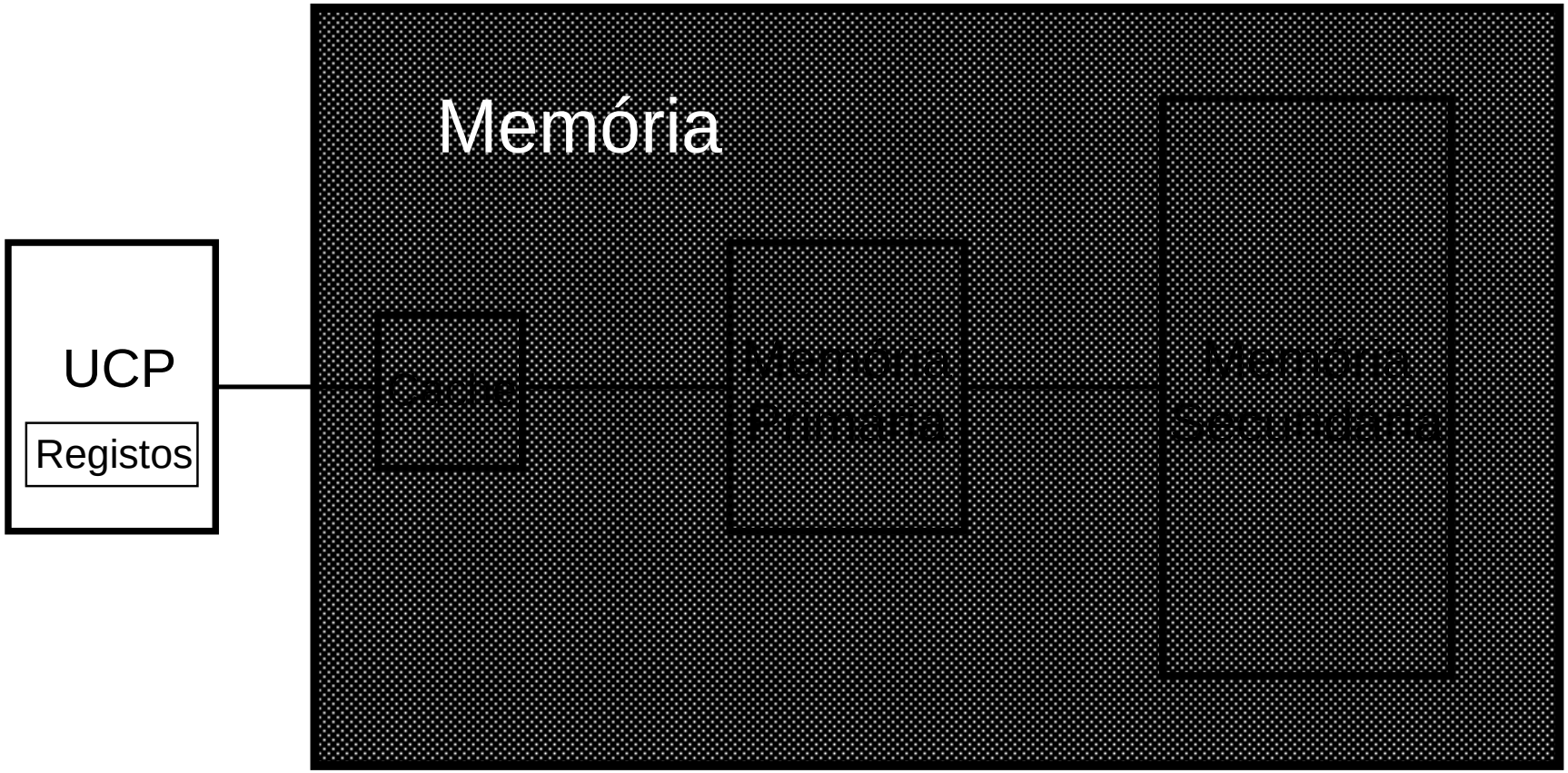
Memória: tempo de acesso diminui 7% / ano

Processador: 35% / ano de aumento de desempenho até 1986, 55% depois disso

Hierarquia de Memória



Hierarquia de Memória



Características dos Níveis de Memória

Nível	**1**	**2**	**3**	**4**
Nome	registos	cache	memória	disco
Capacidade	$<$ 1kB	$<$ 16MB	$<$ 16GB	$>$ 100GB
Tecnologia	CMOS	CMOS SRAM	CMOS DRAM	disco magnético
Acesso (ns)	0,25-0,5	0,5-25	80-250	5.000.000

Comportamento dos Programas

A caracterização do comportamento dos programas resulta da análise dos seus rastos de execução (*traces*).

Tipo de Acesso		Endereço
		⋮
fetch	2	408ed4
leitura	0	10019d94
	2	408ed8
escrita	1	10019d88
	2	408edc
	0	10013220
	2	408ee0
	2	408ee4
		⋮

Regra 90/10

Um programa gasta tipicamente 90% do seu tempo a executar 10% das instruções.

Princípio da Localidade

Regra 90/10 $\Rightarrow$ Princípio da Localidade

Princípio da Localidade

Regra 90/10 $\Rightarrow$ Princípio da Localidade

Localidade Temporal

Se um endereço é referenciado, tenderá a sê-lo de novo em breve.

Princípio da Localidade

Regra 90/10 $\Rightarrow$ Princípio da Localidade

Localidade Temporal

Se um endereço é referenciado, tenderá a sê-lo de novo em breve.

Localidade Espacial

Se um endereço é referenciado, os endereços contíguos tenderão a ser referenciados em breve.

Indicadores Estatísticos das Caches

Sucesso (*hit*): endereço a que se pretende aceder está presente na cache, sendo o acesso servido por esta.

- t_h: tempo de acesso com sucesso
- p_h: fracção de acessos com sucesso (taxa de sucesso, *hit rate*)

Falta (*miss*): endereço a que se pretende aceder não se encontra na cache, sendo necessário um acesso à memória primária.

- t_m: tempo de acesso com falta
- p_m: fracção de acessos com falta, $p_m = 1 - p_h$ (taxa de faltas, *miss rate*)
- t_p: penalidade de falta, $t_p = t_m - t_h$

Indicadores Estatísticos das Caches

Sucesso (*hit*): endereço a que se pretende aceder está presente na cache, sendo o acesso servido por esta.

- t_h: tempo de acesso com sucesso
- p_h: fracção de acessos com sucesso (taxa de sucesso, *hit rate*)

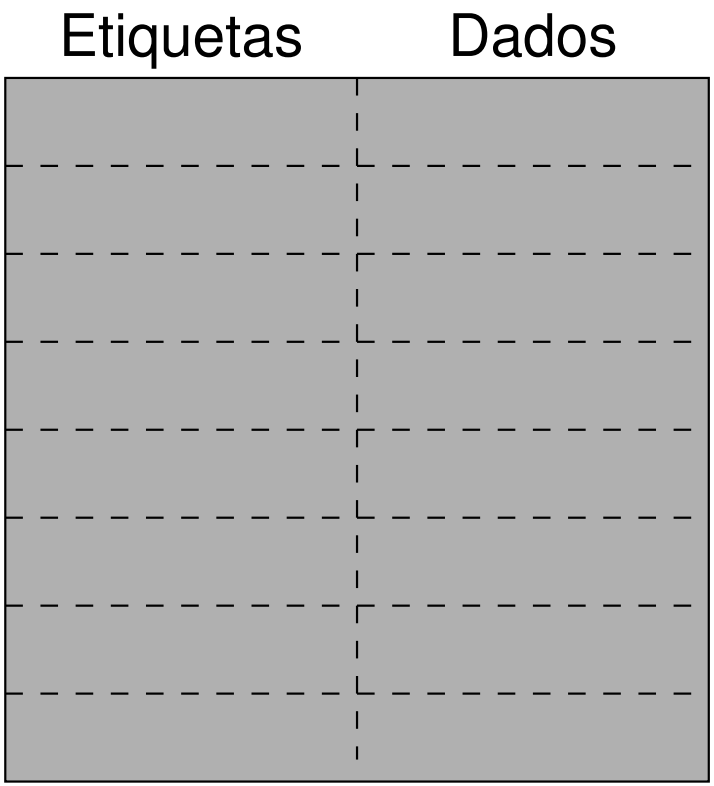
Falta (*miss*): endereço a que se pretende aceder não se encontra na cache, sendo necessário um acesso à memória primária.

- t_m: tempo de acesso com falta
- p_m: fracção de acessos com falta, $p_m = 1 - p_h$ (taxa de faltas, *miss rate*)
- t_p: penalidade de falta, $t_p = t_m - t_h$

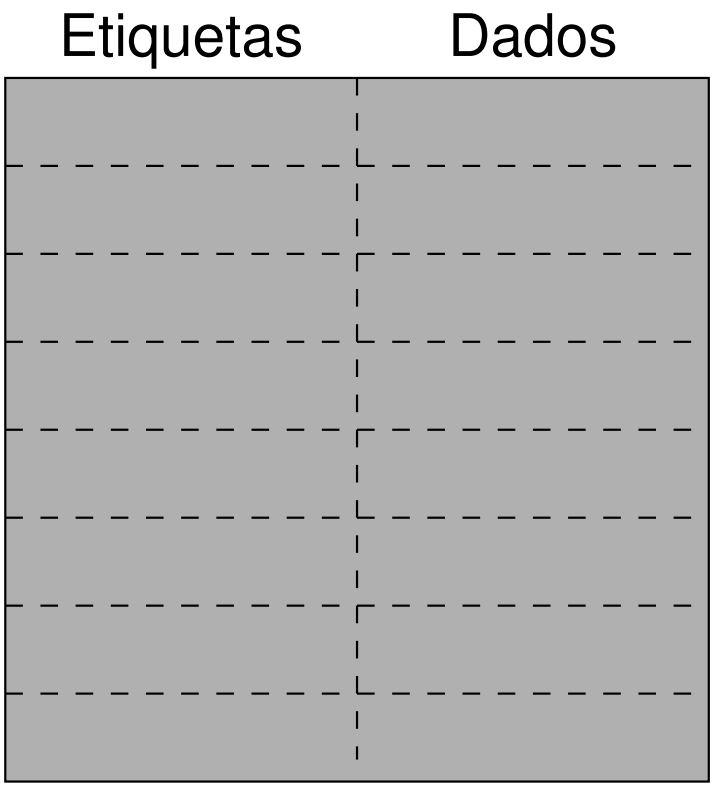
Tempo médio de acesso:

$$\begin{aligned} t_{\text{acesso}} &= p_h \times t_h + p_m \times t_m \\ &= t_h + p_m \times t_p \end{aligned}$$

Organização da Cache



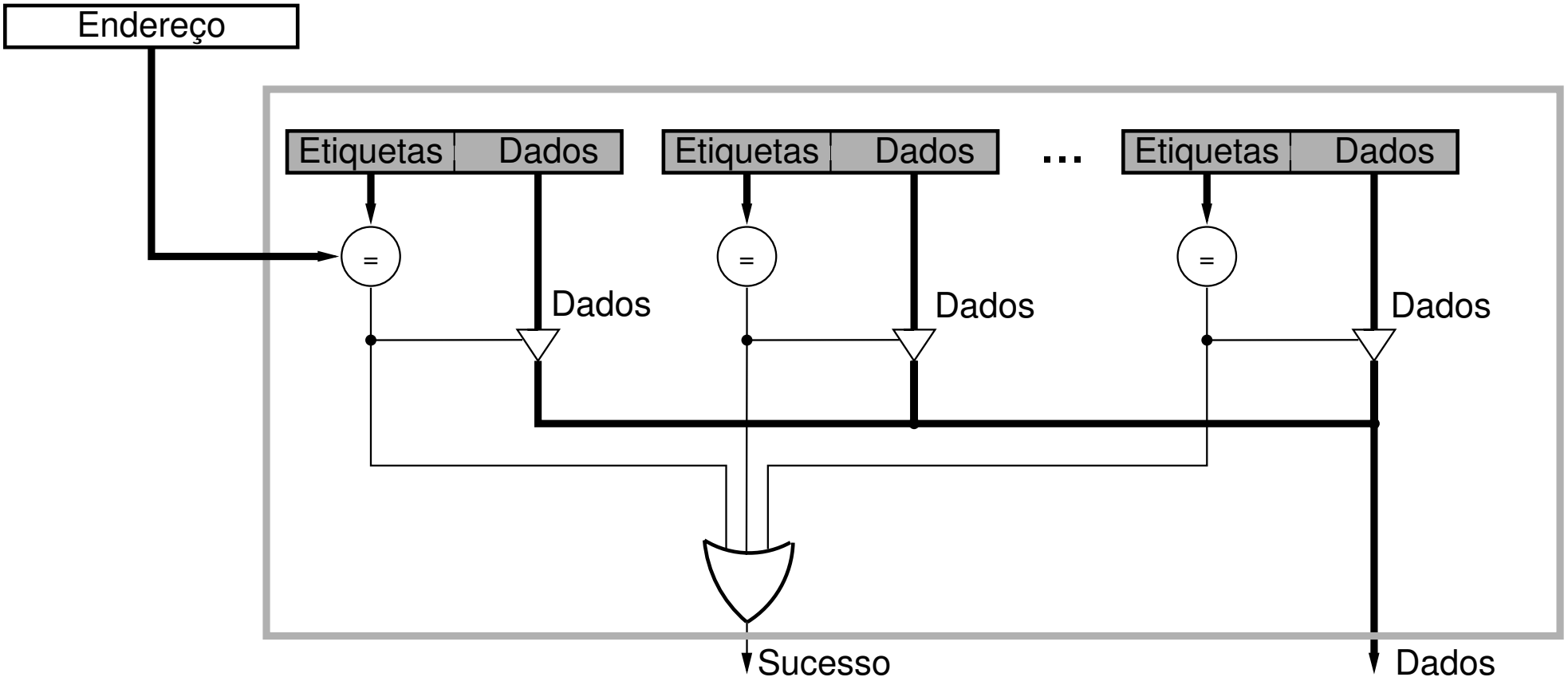
Organização da Cache



Cache Completamente Associativa

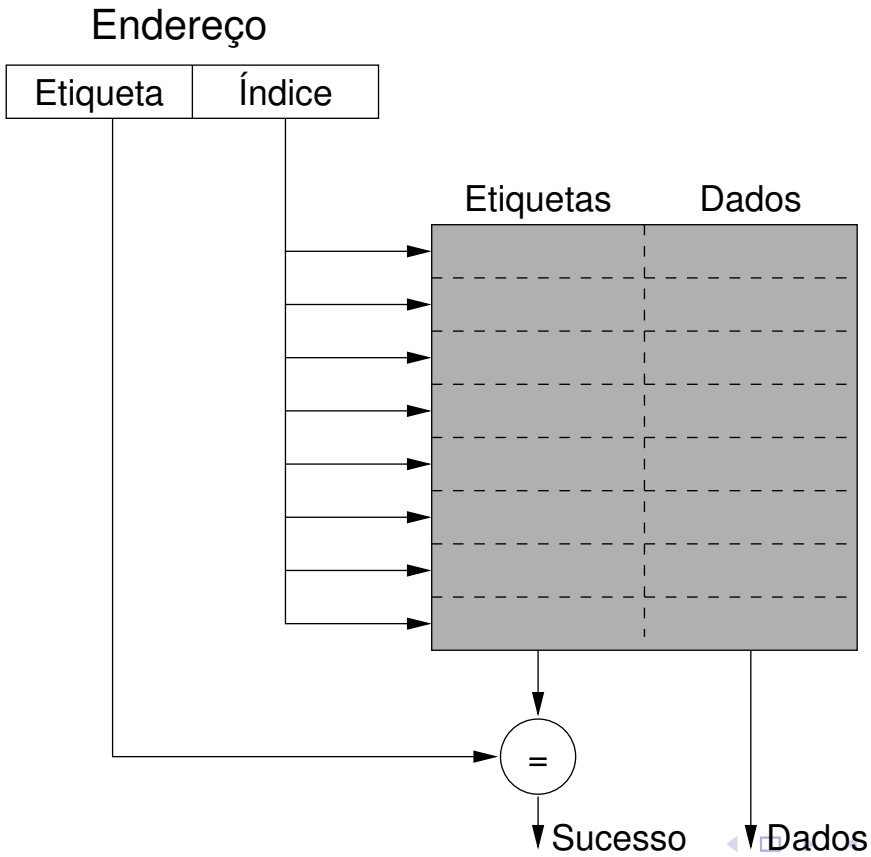
Todas as linhas da cache são testadas em paralelo, pela comparação do endereço pretendido com o campo etiqueta de cada linha.

Cache Completamente Associativa



Cache de Mapeamento Directo

Apenas uma das linhas da cache é pesquisada. O endereço é interpretado em termos de 2 campos, Índice e Etiqueta, em que o primeiro define a linha de cache com a qual o campo Etiqueta vai ser comparado.

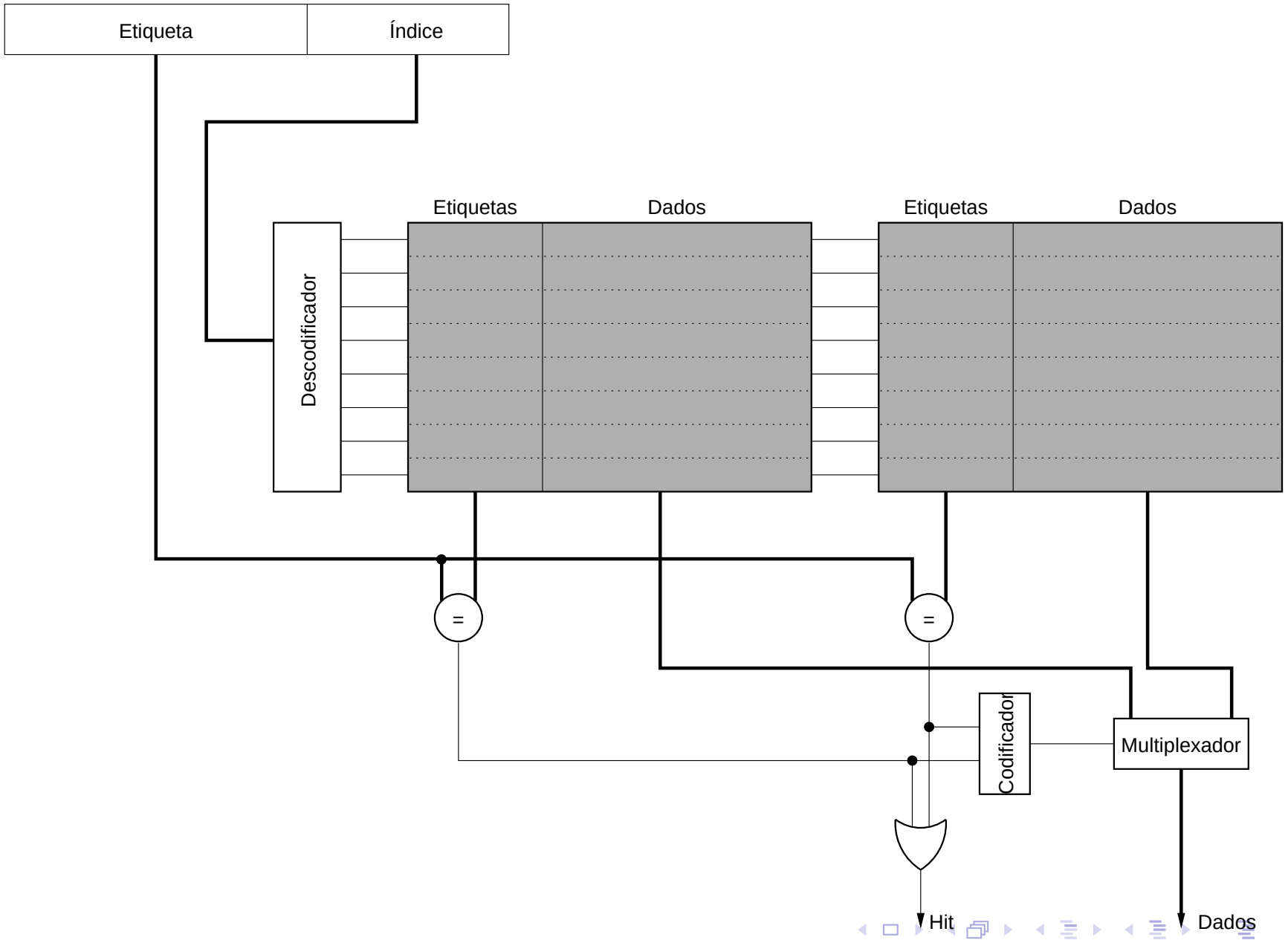


Cache Associativa com *n* Vias

Cache de Associativa com *n* Vias

São pesquisadas *n* vias (*sets*) em paralelo. O endereço é interpretado também interpretado em termos de 2 campos, Índice e Etiqueta, em que o primeiro define *n* linhas de cache a comparar com o campo Índice.

Cache Associativa com *n* Vias



Como Tirar Partido do Princípio da Localidade?

Localidade Temporal:

Como Tirar Partido do Princípio da Localidade?

Localidade Temporal:

Manter na cache os últimos endereços acedidos.

Como Tirar Partido do Princípio da Localidade?

Localidade Temporal:

Manter na cache os últimos endereços acedidos.

Localidade Espacial:

Como Tirar Partido do Princípio da Localidade?

Localidade Temporal:

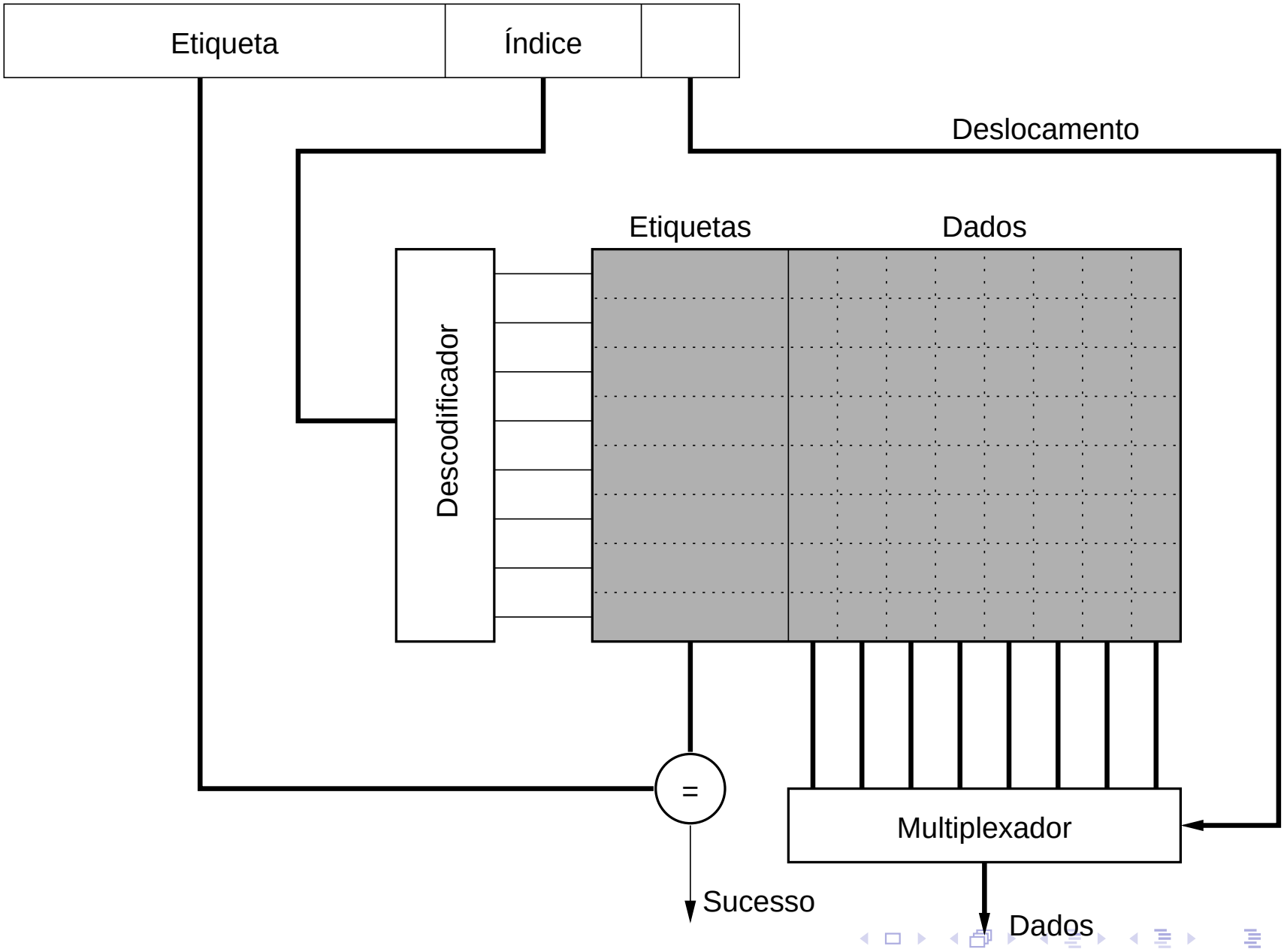
Manter na cache os últimos endereços acedidos.

Localidade Espacial:

Carregar para a cache um conjunto de posições contíguas ao endereço acedido.

Cada linha da cache corresponde não a uma posição de memória, mas a um conjunto.

Organização da Linha da Cache em Blocos



Posição de um Bloco na Cache

Onde colocar um bloco na cache?

Cache Completamente Associativa

Posição de um Bloco na Cache

Onde colocar um bloco na cache?

Cache Completamente Associativa

o bloco pode ficar em qualquer posição da cache.

Cache de Mapeamento Directo

Posição de um Bloco na Cache

Onde colocar um bloco na cache?

Cache Completamente Associativa

o bloco pode ficar em qualquer posição da cache.

Cache de Mapeamento Directo

cada bloco apenas pode ficar numa posição da cache, determinada pelos bits do campo índice.

Endereço		
Etiqueta	Índice	Bloco

Associativa de *n* vias

Posição de um Bloco na Cache

Onde colocar um bloco na cache?

Cache Completamente Associativa

o bloco pode ficar em qualquer posição da cache.

Cache de Mapeamento Directo

cada bloco apenas pode ficar numa posição da cache, determinada pelos bits do campo índice.

Endereço		
Etiqueta	Índice	Bloco

Associativa de *n* vias

o bloco tem *n* posições possíveis de colocação, uma por cada via, sendo a posição numa dada via determinada pelos bits do campo índice.

Política de Substituição

Qual o bloco a retirar da cache, se for caso disso?

Política de Substituição

Qual o bloco a retirar da cache, se for caso disso?

⇒ LRU (Least Recently Used):

retirar o que não é usado há mais tempo.

⇒ FIFO (First-in First-out):

retirar o que foi carregado para a cache há mais tempo.

⇒ Aleatório

Política de Substituição

Qual o bloco a retirar da cache, se for caso disso?

⇒ LRU (Least Recently Used):

retirar o que não é usado há mais tempo.

⇒ FIFO (First-in First-out):

retirar o que foi carregado para a cache há mais tempo.

⇒ Aleatório

	2 Vias			4 Vias			8 Vias		
Capacidade	**LRU**	**RND**	**FIFO**	**LRU**	**RND**	**FIFO**	**LRU**	**RND**	**FIFO**
16 kB	11,4	11,7	11,6	11,2	11,5	11,3	10,9	11,2	11,0
64 kB	10,3	10,4	10,4	10,2	10,2	10,3	10,0	10,1	10,0
256 kB	9,2	9,2	9,3	9,2	9,2	9,3	9,2	9,2	9,3