

# Endereços

# Endereço

A **memória** pode ser vista como uma sequência de *bytes*

A **posição** de cada *byte* dentro dessa sequência é indicada por um **número** (entre 0 e a **dimensão da sequência** - 1)

## Memória

Bytes																
Posição	0	1	2	3	4	5	6	7	8	9	10	11	12	13	...	

Um **endereço** é um número que corresponde à posição de um *byte* nessa sequência

# Palavra (1)

A maioria dos **acessos** à memória são acessos a **palavras**

Uma **palavra** é constituída por **bytes** (normalmente 4 ou 8)

As palavras são numeradas sequencialmente a partir de 0

Calculando o quociente entre um **endereço** e o **número de bytes por palavra** obtém-se o **número da palavra** a que o *byte* pertence

$$\text{n}^{\circ} \text{ da palavra} = \frac{\text{endereço}}{\text{bytes por palavra}}$$

O **número da palavra** é por vezes chamado **endereço da palavra**

## Palavra (2)

### Exemplo

Num sistema com palavras de 32 bits, o *byte* com endereço 268435474 pertence à palavra

$$67108868 = \frac{268435474}{4}$$

Os restantes *bytes* da palavra 67108868 têm os endereços

$$268435472 = 67108868 \times 4$$

$$268435473 = 67108868 \times 4 + 1$$

$$268435475 = 67108868 \times 4 + 3$$

O endereço usado para aceder a uma *palavra* é o endereço do seu *primeiro byte*

No caso da palavra 67108868, será o endereço 268435472

Informalmente, a palavra num dado endereço é a palavra que *começa* no *byte* com aquele endereço

## Palavra (3)

### Exemplo

Num sistema com palavras de 64 bits, o *byte* com endereço 268435474 pertence à palavra

$$33554434 = \frac{268435474}{8}$$

Os restantes *bytes* da palavra 33554434 têm os endereços

$$268435472 = 33554434 \times 8$$

$$268435473 = 33554434 \times 8 + 1$$

$$268435475 = 33554434 \times 8 + 3$$

$$268435476 = 33554434 \times 8 + 4$$

$$268435477 = 33554434 \times 8 + 5$$

$$268435478 = 33554434 \times 8 + 6$$

$$268435479 = 33554434 \times 8 + 7$$

# Bloco (1)

A unidade de transferência de informação entre os vários níveis de memória é o **bloco**

Um **bloco** é constituído por **palavras** (1, 2, 4, 8, ...)

Os blocos são numerados sequencialmente a partir de 0

O **número do bloco** a que uma palavra pertence obtém-se calculando o quociente entre o **número da palavra** e o **número de palavras por bloco**

$$\text{n}^{\circ} \text{ do bloco} = \frac{\text{n}^{\circ} \text{ da palavra}}{\text{palavras por bloco}}$$

## Bloco (2)

Calculando o quociente entre um endereço e o número de bytes por bloco obtém-se o número do bloco a que o byte pertence

$$\text{n}^{\circ} \text{ do bloco} = \frac{\text{endereço}}{\text{bytes por bloco}}$$

### Exemplo

Se um bloco tiver 2 palavras, a palavra número 67108868 pertence ao bloco

$$33554434 = \frac{67108868}{2}$$

A outra palavra do bloco 33554434 é a palavra

$$67108869 = 33554434 \times 2 + 1$$

## Bloco (3)

### Exemplo

Se um bloco tiver 2 palavras de 32 bits, o *byte* com endereço 268435474 pertence ao bloco

$$33554434 = \frac{268435474}{2 \times 4}$$

### Exemplo

Se um bloco tiver 8 palavras, a palavra 67108868 pertence ao bloco

$$8388608 = \frac{67108868}{8}$$

As outras palavras do bloco 8388608 são as palavras

$$67108864 = 8388608 \times 8 \quad \text{a} \quad 67108867 = 8388608 \times 8 + 3 \quad \text{e}$$

$$67108869 = 8388608 \times 8 + 5 \quad \text{a} \quad 67108871 = 8388608 \times 8 + 7$$



# Página

Página é o nome que se dá aos blocos quando se fala de memória virtual

$$\text{n}^{\circ} \text{ da página} = \frac{\text{endereço}}{\text{bytes por página}}$$

A página é física ou virtual consoante o endereço seja físico ou virtual

## Exemplo

Se uma página tiver 4 KB, o endereço 268435474 pertence à página

$$65536 = \frac{268435474}{4096}$$

Os bytes da página 65536 são os com endereços entre

$$268435456 = 65536 \times 4096 \quad \text{e} \quad 268439551 = 65536 \times 4096 + 4095$$

# Memória

Bytes, palavras, blocos

## Memória

Exemplo com palavras de 32 bits e blocos com 2 palavras

Bytes	<table><tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr></table>																													
Endereço	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	...														
Palavra		0			1			2			...																			
Bloco				0							...																			

# Cache

Uma **cache** contém (**conjuntos de**) **blocos**

O **conjunto**, ou o **índice** da posição, onde um **bloco** poderá residir na cache é determinado pelo resto da divisão entre o **número do bloco** e o **número de conjuntos** da cache

$$\text{índice} = \text{n}^{\circ} \text{ do bloco} \% \text{n}^{\circ} \text{ de conjuntos}$$

O **resto** do número do bloco constitui o **tag**, que ajuda a identificar o bloco presente em cada posição da cache

$$\text{tag} = \frac{\text{n}^{\circ} \text{ do bloco}}{\text{n}^{\circ} \text{ de conjuntos}}$$

Donde:  $\text{n}^{\circ} \text{ do bloco} = \text{tag} \times \text{n}^{\circ} \text{ de conjuntos} + \text{índice}$

# TLB

O TLB é uma *cache* indexada a partir do *nº da página*

O *índice* e o *tag* são calculados como para as restantes caches, com o *nº da página* a substituir o *nº do bloco*

$$\text{índice} = \text{nº da página} \% \text{nº de conjuntos}$$

$$\text{tag} = \frac{\text{nº da página}}{\text{nº de conjuntos}}$$

# Endereços e aritmética binária (1)

Como o computador só conhece os valores 0 e 1, trabalhando com potências de 2 faz com que as operações aritméticas anteriores se reduzam a **manipulações de bits**

## Exemplos

Se um endereço tiver 32 bits, a representação binária do endereço 268435474 é

0001 0000 0000 0000 0000 0000 0001 0010

Se uma palavra tiver 32 bits =  $2^4$  bytes, o byte 268435474 pertence à palavra

$$\begin{aligned} & \frac{0001\ 0000\ 0000\ 0000\ 0000\ 0000\ 0001\ 0010_2}{100_2} = \\ & = 1000000000000000000000000000100_2 \\ & = 67108868 \end{aligned}$$

## Endereços e aritmética binária (2)

## Exemplos (cont.)

Os *bytes* da palavra 67108868 têm os endereços

0001 0000 0000 0000 0000 0000 0001 0000<sub>2</sub>

0001 0000 0000 0000 0000 0000 0001 0001<sub>2</sub>

0001 0000 0000 0000 0000 0000 0001 0010<sub>2</sub>

0001 0000 0000 0000 0000 0000 0001 0011<sub>2</sub>

Se um bloco tiver  $8 = 2^3$  palavras, a palavra 67108868 pertence ao bloco

[illegible]



## Endereços e aritmética binária (4)

### Exemplos (cont. 3)

Se a cache tiver  $16 = 2^4$  conjuntos, o conjunto em que o bloco 8388608 poderá residir é o

$$\begin{aligned} 100000000000000000000000_2 \% 10000_2 &= 0000_2 \\ &= 0 \end{aligned}$$

O tag para este bloco será

$$\begin{aligned} &\frac{100000000000000000000000_2}{10000_2} = \\ &= 100000000000000000000_2 \\ &= 524288 \end{aligned}$$



## Endereços e aritmética binária (5)

### Exemplos (cont. 4)

Se uma página tiver  $4\text{ KB} = 2^{12}$  bytes, o endereço 268435474 pertence à página

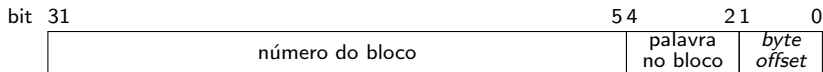
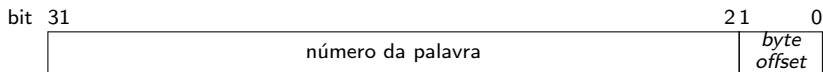
$$\begin{aligned} & \frac{0001\ 0000\ 0000\ 0000\ 0000\ 0000\ 0001\ 0010_2}{1000000000000_2} = \\ &= 10000000000000000_2 \\ &= 65536 \end{aligned}$$

Os bytes da página 65536 são os com endereços de

$$\begin{aligned} & 0001\ 0000\ 0000\ 0000\ 0000\ 0000\ 0000\ 0000_2 \quad \text{a} \\ & 0001\ 0000\ 0000\ 0000\ 0000\ 1111\ 1111\ 1111_2 \end{aligned}$$

## Endereços e aritmética binária (6)

Estas operações dão origem a várias **visões** sobre os endereços



Válido para endereços e palavras de **32 bits**, blocos com **8 palavras**, cache com **16 conjuntos** e páginas com **4 KB**

## Endereços e aritmética binária (7)

## Exemplo

Para o endereço  $268435474 = 0001\ 0000\ 0000\ 0000\ 0000\ 0000\ 0001\ 0010_2$

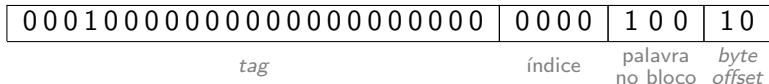
bit 31 21 0



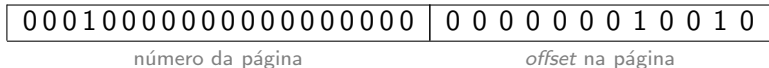
bit 31 54 21 0



bit 31 98 54 21 0



bit 31 12 11 0



Endereços e palavras de 32 bits, blocos com 8 palavras, cache com 16 conjuntos e páginas com 4 KB