

Memória Virtual

Prof. Edson Pedro Ferlin

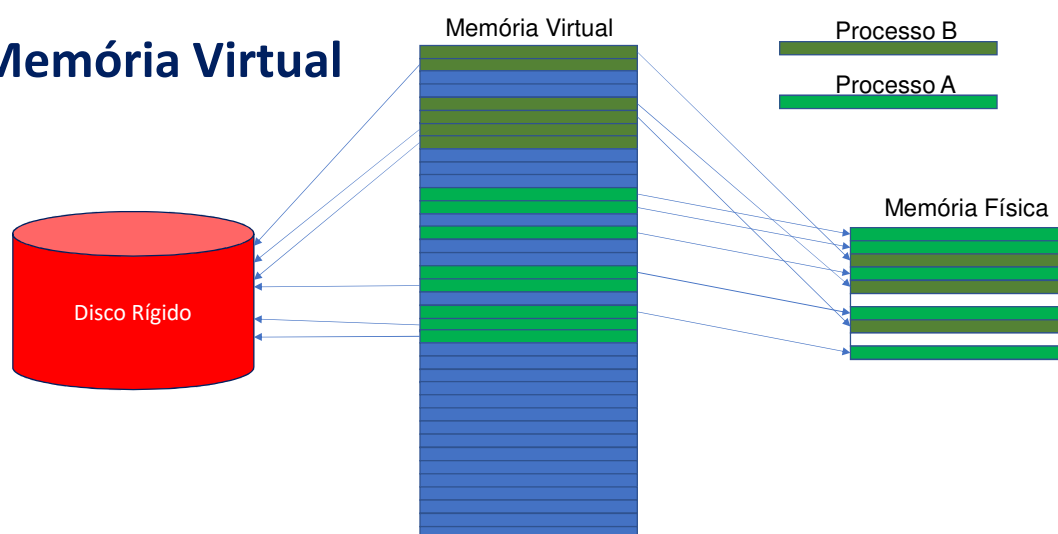
Agradecimento ao Prof. Osmar Betazzi Dordal

- **Objetivos**
 - Estudar a memória virtual, seu funcionamento e estruturas
- **Conteúdos**
 - Paginação
 - Segmentação

Memória Virtual

- É uma técnica utilizada para trabalhar com a memória secundária como um “cache” para partes do espaço dos processos
- Por que é utilizada a memória virtual?
 - O tamanho são cada vez maiores;
 - Eleva o grau de multiprogramação;
 - Execução de programas maiores que a RAM.
- Um processo utiliza endereços virtuais e não físicos
 - Utiliza o MMU para a conversão.

Memória Virtual



Técnicas de Memória Virtual

- **Paginação**

- Blocos de tamanhos fixos
 - Exemplo: 4KB
- Os espaços de endereçamento virtual é dividido em páginas virtuais

- **Segmentação**

- Blocos de tamanho arbitrário chamados de segmentos
- Contém o mesmo tipo de informação
 - Exemplo: texto, códigos, dados, pilha, entre outros
- Mapeamento entre endereços reais e virtuais são realizados pela MMU
- **Muitos Sistemas Operacionais utilizam uma mescla das duas técnicas**

Técnicas de Memória Virtual

- **Paginação**

- Blocos de tamanhos fixos
 - Exemplo: 4KB
- Os espaços de endereçamento virtual é dividido em páginas virtuais

- **Segmentação**



Melhor Proteção devido a divisão dos tipos de informação

- Blocos de tamanho arbitrário chamados de segmentos
- Contém o mesmo tipo de informação
 - Exemplo: texto, códigos, dados, pilha, entre outros
- Mapeamento entre endereços reais e virtuais são realizados pela MMU
- **Muitos Sistemas Operacionais usam uma mescla das duas técnicas**

Memória Virtual – Paginação

- **Páginas**
 - Unidades de tamanho fixo **localizado** no dispositivo secundário
- **Frames**
 - Unidades correspondentes **localizado** na memória física – RAM
- **Page Fault (Page-In e Page-Out)**
 - É o evento quando uma página que não está na RAM é referenciada
 - Usa uma *trap* para carregar ou substituir uma página
- **Tabela de Páginas**
 - Estrutura para mapear uma página ao frame correspondente, sendo que cada processo tem um

Memória Virtual – Paginação

- No caso de um sistema que gera 64KB de endereços virtuais (16 páginas e 8 frames)
- MMU faz o mapeamento
- MOV REG, 5
- Ela está mapeada ao terceiro frame, que começa em 8k = 8192
- O endereço enviado ao barramento é $5 + 8192 = 8197$

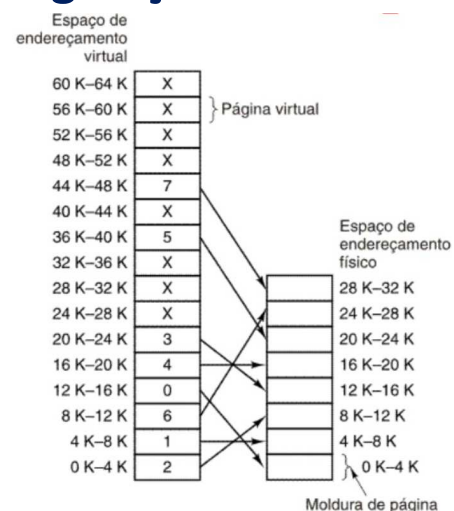
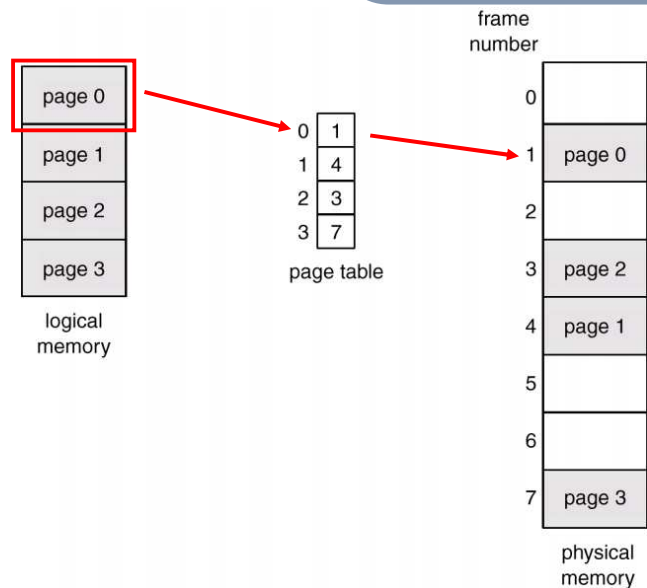


Tabela de Páginas



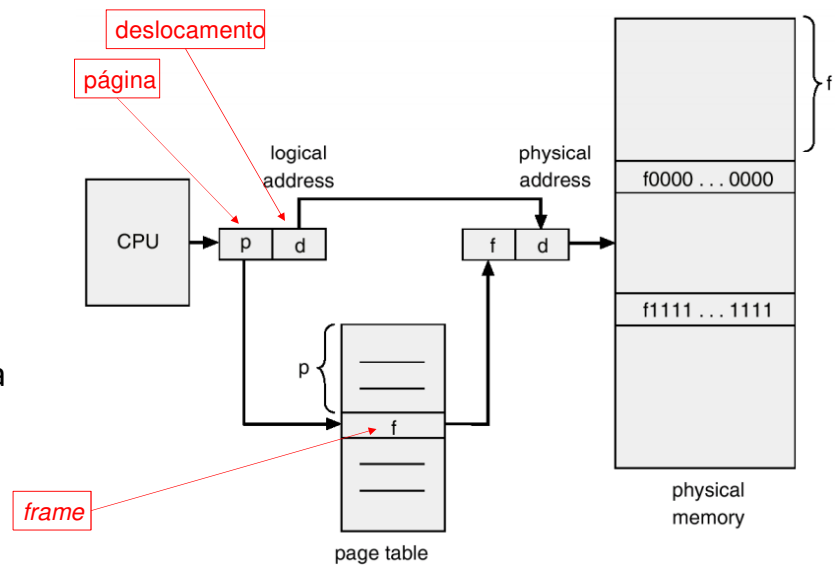
9

Memória Virtual

Prof. Edson Pedro Ferlin

Tradução de Endereços

- Busca sequencial na tabela de páginas
- Qualquer que seja a alternativa, é lenta
- Ideal – a própria página servir como índice na tabela



10

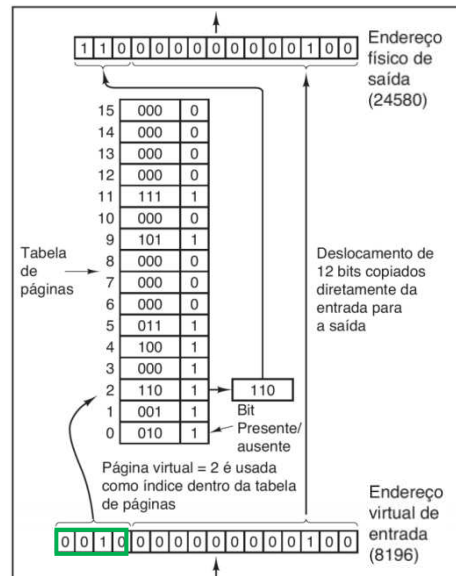
Memória Virtual

Prof. Edson Pedro Ferlin

Exemplo de Busca

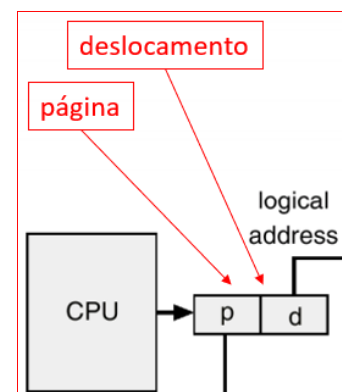
- Endereço 8196
 - 0010 - 000000000100
 - MMU com 16 páginas de 4Kb
 - Endereço virtual de 16 bits
 - A tabela tem 16 entradas – 0000 a 1111
 - Hardware com 8 frames
 - Endereço virtual com 15 bits
 - 110-000000000100

0	-	0000
1	-	0001
2	-	0010
3	-	0011
4	-	0100
5	-	0101
6	-	0110
7	-	0111
8	-	1000
9	-	1001
10	-	1010
11	-	1011
12	-	1100
13	-	1101
14	-	1110
15	-	1111



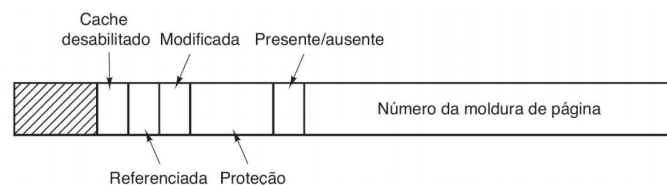
Componentes do Endereço

- Número de páginas (p)** – usado como um índice para uma tabela de páginas
- Deslocamento de página (d)** – combinado com endereço de base para definir o endereço de memória física que é enviado à unidade de memória
- Páginas maiores:** leitura mais eficiente, tabela menor, mas possui fragmentação interna
- Páginas menores:** leitura menos eficiente, tabela maior, mas possui menos fragmentação



Componentes da Tabela

- **Número de moldura de página** – identifica o número da página real (*page frame number*)
- **Bit de residência** (presente/ausente) – se for 1, então a página correspondente é válida e está na RAM (*page fault*)
- **Bit de proteção** – [0 = leitura/escrita], [1 = leitura] e [2 = execução]
- **Bit de codificação** – [1 = página alterada] e [0 = não alterada]
- **Bit de referencia** – [1 = foi referenciado ou não "recentemente"]
- **Bit de cache** – permite desabilitar o *caching* da página

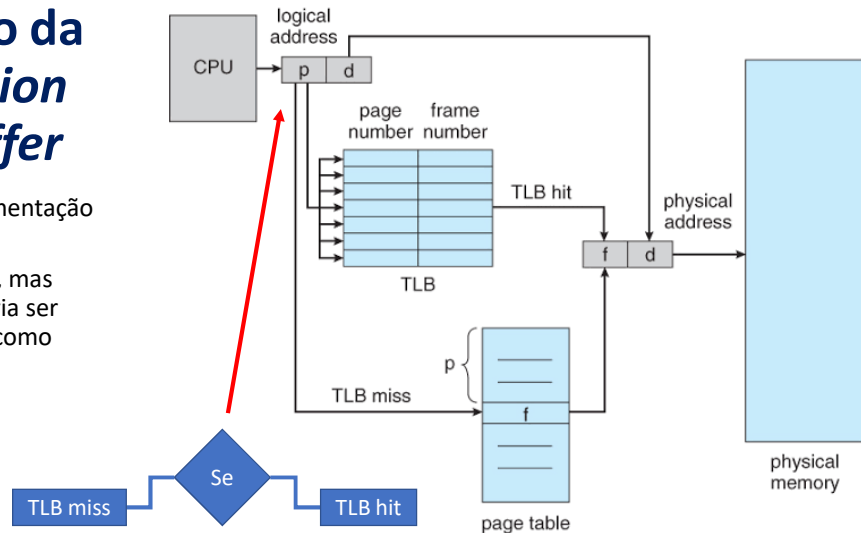


Local de Armazenamento das Tabelas

- **Vetor de Registradores**, se a memória for pequena
 - Mantidos no hardware
- **Na própria memória RAM**
 - A MMU gerencia utilizando um ou dois registradores
- **Em uma memória cache na MMU chamada Memória Associativa**
 - Usa para melhorar o desempenho da tabela na RAM
 - Na tabela RAM são usados dois registradores
 - Registrador de base de tabela de página (PTBR)
 - Registrador de tamanho da tabela de página (PTLR)
 - **Problema:** Dois acessos para instrução/dado na RAM
 - Solução: cache chamado TLB

Funcionamento da TLB – Translation Lookaside Buffer

- Tem possibilidade de implementação em **hardware** ou **software**
- **Em hardware é mais rápido**, mas ocupa um espaço que poderia ser usado para outras funções, como cache



15

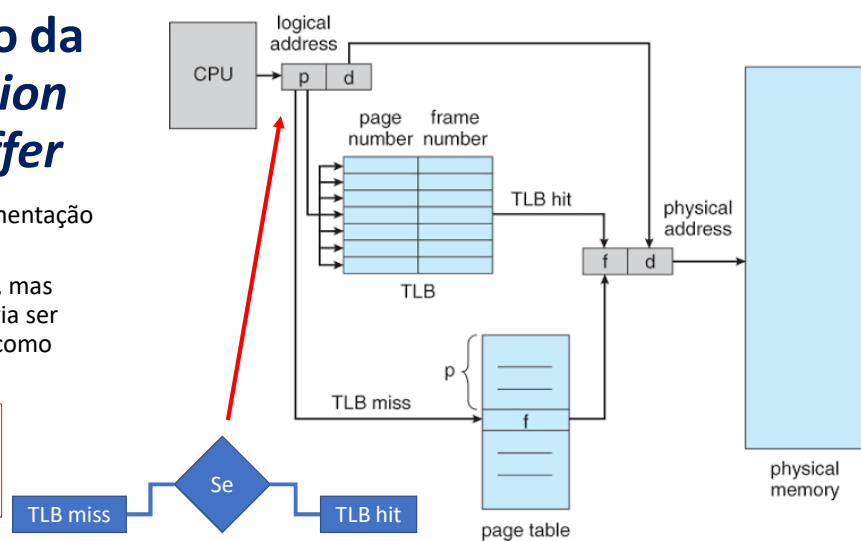
Memória Virtual

Prof. Edson Pedro Ferlin

Funcionamento da TLB – Translation Lookaside Buffer

- Tem possibilidade de implementação em **hardware** ou **software**
- **Em hardware é mais rápido**, mas ocupa um espaço que poderia ser usado para outras funções, como cache

Soft miss: quando a página referenciada não está na TLB, mas está na RAM (Basta utilizar a TLB)



16

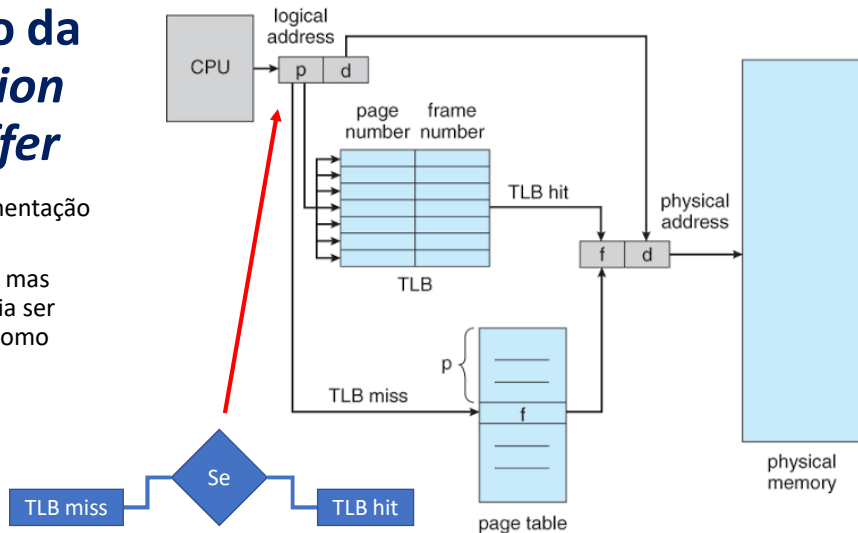
Memória Virtual

Prof. Edson Pedro Ferlin

Funcionamento da TLB – Translation Lookaside Buffer

- Tem possibilidade de implementação em **hardware** ou **software**
- Em **hardware** é mais rápido, mas ocupa um espaço que poderia ser usado para outras funções, como cache

Hard miss: quando a página referenciada não está na memória física e nem na TLB (Trazer do Disco para a RAM e depois para a TLB (lento))



Contato



eferlin@live.com



(BLOG) professorferlin.blogspot.com

(SITE) professorferlin.webnode.com.br

(YOUTUBE) ProfEdsonPedroFerlin