

Paginação: Suporte de Hardware

Marcelo Johann

Lembrando: paginação

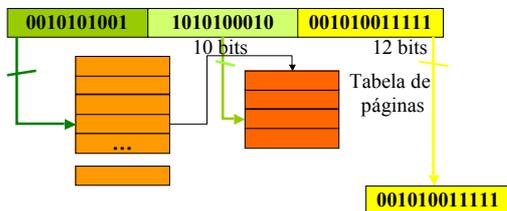
- O Sis. Op. gerencia a memória através de paginação
 - Aloca páginas (lógicas) em quadros (físicos).
 - Tamanho típico: 4 Kbytes / página.
- Para tal, usa-se uma tabela de páginas:
 - Local a cada processo
 - Usa um mecanismo de número de páginas+deslocamento para converter os endereços lógicos/físicos.

INF01142 - Sistemas Operacionais I N - Marcelo Johann - 2010/2

Aula 15 : Slide 2

O caso da INTEL 8086

- Mecanismo com dois níveis.
 - Endereço lógico: 4 GB (32 bits)
 - Páginas de 4 KB
 - Número total de 4 GB/4 KB = 1 Mega-páginas.
- O endereço lógico é dividido em 10+10+12 bits.



INF01142 - Sistemas Operacionais I N - Marcelo Johann - 2010/2

Aula 15 : Slide 3

Tamanho da tabela de páginas

- 10 bits são usados para endereçar a tabela de 2º nível.
 - Logo, são $2^{10} = 1K$ entradas na tabela.
- Quantos bits por linha dessa tabela?
 - É preciso de q bits, sendo que $q+12 =$ largura da palavra da RAM.
 - Supondo que se trate de uma arquitetura 32 bits, temos que $q+12 = 32$, e $q=20$ bits.
- Essa tabela ocupa então 20×2^{10} bits
 - Isso é equivalente a **20 KBytes**.

INF01142 - Sistemas Operacionais I N - Marcelo Johann - 2010/2

Aula 15 : Slide 4

Como implementar/armazenar as tabelas?

- Precisa-se armazenar em algum lugar essas tabelas (uma por processo):
 - Ou em registradores;
 - Ou na RAM;
 - Ou em uma memória especial
- Além disso, o Sis. Op. deve manter uma lista de quadros disponíveis
 - Qual implementação?
 - Bitmap, lista...
 - Onde?
 - Registradores, RAM ?

INF01142 - Sistemas Operacionais I N - Marcelo Johann - 2010/2

Aula 15 : Slide 5

Controle de uso dos quadros

- Em geral, é feito através de um bitmap
 - Velocidade de acesso.
- O tamanho do bitmap é proporcional ao tamanho da memória RAM
 - 1 bit por quadro na RAM.
- O bitmap é relativamente pequeno, e precisa ser de rápido acesso:
 - RAM de 1 GB, quadros de 4KB = 250 K quadros
 - Armazenado na RAM, em geral.

INF01142 - Sistemas Operacionais I N - Marcelo Johann - 2010/2

Aula 15 : Slide 6

Implementação das tabelas de páginas

- As tabelas são muito mais complexas.
- Tem uma por processo.
- Estrutura lógica: cada tabela é um vetor indexado pelo número da página.
- Cada entrada (linha) na tabela ocupa:
 - q bits (número do quadro);
 - Mais alguns bits de controle (validade, segurança, modo de acesso...).
- É preciso ter um **compromisso** entre o tamanho da estrutura e seu lugar de armazenamento.

INF01142 - Sistemas Operacionais I N - Marcelo Johann - 2010/2

Aula 15 : Slide 7

Tabela de páginas em HW (registradores)

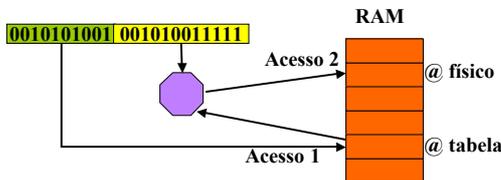
- Cada entrada na tabela de páginas é mapeada em um registrador de HW (da MMU).
- É preciso que o contexto do processo salve o conteúdo desses registradores a cada troca de contexto!
 - Os registradores serão re-aproveitados por um outro processo.
- Acesso extremamente rápido!
- Grande problema: o número limitado de registradores que pode disponibilizar a MMU.
 - **Problema de custo!**

INF01142 - Sistemas Operacionais I N - Marcelo Johann - 2010/2

Aula 15 : Slide 8

Tabela de páginas em RAM

- A tabela é mantida em memória
 - Cada processo dispõe de 2 registradores
 - Um para o endereço base da tabela na RAM (PTBR)
 - Um para o número de entradas na tabela (PTLR).
 - Assim, o tamanho da tabela pode evoluir no tempo.
- Problema: desempenho
 - Cada acesso à memória lógica necessita no mínimo 2 acessos à RAM física!



INF01142 - Sistemas Operacionais I N - Marcelo Johann - 2010/2

Aula 15 : Slide 9

O melhor de dois mundos: a TLB

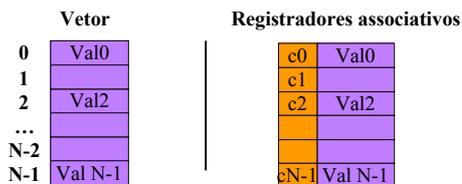
- *Translation Look-aside Buffer*, cache de Tabela de Páginas
- Meio-termo entre a solução por registradores e a solução em memória:
 - O HW espelha parte da tabela que fica na RAM.
 - Já que é uma parte da tabela, não se pode usar um vetor!
 - Usa poucos registradores para auxiliar a estrutura em SW do Sis. Op.
- Quando se acessa uma página que está nos registradores, é igual à solução por HW (*TLB-hit*).
- Quando a página não está no TLB, passa-se a acessar a tabela na RAM (*TLB-miss*).

INF01142 - Sistemas Operacionais I N - Marcelo Johann - 2010/2

Aula 15 : Slide 10

“Look-aside buffer”

- A TLB usa registradores associativos:
 - Em vez de um vetor acessado através de um endereço, armazena pares (chave, valor).

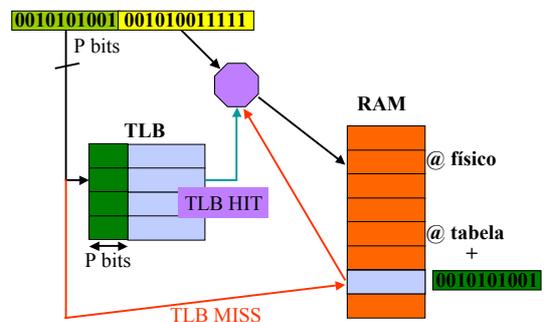


- Estrutura maior (deve armazenar as chaves)
- Possibilita menos entradas
- Possibilita a procura em paralelo!

INF01142 - Sistemas Operacionais I N - Marcelo Johann - 2010/2

Aula 15 : Slide 11

Interação Sis. Op. / TLB



INF01142 - Sistemas Operacionais I N - Marcelo Johann - 2010/2

Aula 15 : Slide 12

Efeito do uso da TLB

- Melhora, e muito, o desempenho
 - TLB hits muito mais frequentes que TLB miss, e metade do tempo de acesso
 - Usa **localidade**
- Mas usa recursos caros!
 - Limitação no número de registradores associativos empregados.
 - Tipicamente, 8-2048 entradas da tabela são espelhadas.
 - A TLB é compartilhada pelos processos!
- Quando há TLB-miss, a página vai ser procurada na memória.
 - Atualiza-se a TLB com essa nova entrada.
 - Em geral, a entrada mais antiga na TLB é descartada.
- TLB e paginação multi-nível...
 - Só o 1o nível é contemplado pela TLB.
 - TLB quase acabou com multi-nível em HW!

INF01142 - Sistemas Operacionais I N - Marcelo Johann - 2010/2

Aula 15 : Slide 13

Taxa de acerto & tempo de acesso à RAM

- Seja a a taxa de acerto na TLB
 - TLB hit em a % dos casos.
- Seja t_{tlb} o tempo de acesso à TLB e t_{ram} o tempo de acesso à RAM.
- O tempo médio de obtenção de um dado na RAM será:
$$a(t_{tlb} + t_{ram}) + (1-a) \times (t_{tlb} + 2t_{ram})$$

INF01142 - Sistemas Operacionais I N - Marcelo Johann - 2010/2

Aula 15 : Slide 14

Em um computador moderno...

- Arquitetura 64 bits ($y=64$)
- Espaço de endereçamento lógico de 64 bits / processo
 - $2^{64} = 16 \times 1G \times 1G \times 1 \text{ Byte}$
- Página de 4 Kbytes
 - Tem-se $4 \times 1G \times 1G / 1K$ páginas = $4M \times 1G$.
- Nunca vai funcionar com uma tabela de páginas
 - Mecanismos multi-nível com 3 níveis

INF01142 - Sistemas Operacionais I N - Marcelo Johann - 2010/2

Aula 15 : Slide 15

Alternativa: tabela de páginas invertida

- As tabelas de páginas vistas até agora são locais a cada processo.
- Pode-se também usar uma tabela única para todo o sistema que mapeia os endereços FÍSICOS (quadros).
 - Cada entrada será um conjunto (pid, endereço lógico);
 - A tabela associará o endereço de um **quadro** à entrada (pid, @lógico);
 - O mesmo, junto com o deslocamento, fornecerá o endereço físico.
- **Vantagem:**
 - Tem tantas entradas como endereços **físicos**.
- **Problema:** cada vez que um processo quer acessar a memória, precisa fazer uma procura em toda a tabela.

INF01142 - Sistemas Operacionais I N - Marcelo Johann - 2010/2

Aula 15 : Slide 16

Tabela invertida & TLB

- A TLB ajuda a manter tabelas invertidas!
 - Ela contém “cópias” (parciais) de associações página/quadro!!!!
 - Em caso de TLB hit, não tem problema de procura.
- Mas piora (muito) quando tem TLB miss...
- Para compensar: tabelas de Hash.
 - Endereços virtuais são “hashadas”
 - Mantém-se uma tabela Hash-endereço físico.

INF01142 - Sistemas Operacionais I N - Marcelo Johann - 2010/2

Aula 15 : Slide 17