PPI

23.11.2011



Virtuálna pamäť



- je taká správa pamäti, pri ktorej sa každému procesu dáva k dispozícii vlastná pamäť, ktorá má inú veľkosť alebo iný spôsob adresovania ako je fyzická pamäť
- Historicky vznikla hlavne z potreby vykonávať programy, ktoré sú väčšie než fyzická pamäť počítača
- Niektoré stránky virtuálnej pamäti sú vo fyzickej pamäti, iné sú odložené na disku





- MMU memory management unit jednotka správy pamäte
- FAP fyzický adresový priestor-priradený pamäti
- LAP priradený programu
- Ideálne LAP=FAP, nie je realita ⇒ správa pamäte
 - Program potrebuje LAP > FAP ⇒ vytvorenie virtuálnej pamäte na pevnom disku
 - Rozdelenie pamäte na viacero programov dynamická relokácia premiestňovanie LA na rôzne FA
 - Ochrana FAP prístupové práva

Zavedenie *správy pamät*e má tri hlavné príčiny:



- Program môže vyžadovať pre svoje vykonanie väčší logický adresový priestor, ako je fyzický adresový priestor, ktorý má počítač k dispozícii.
- Vyžaduje sa priradenie a rozdelenie fyzického adresového priestoru viacerým používateľským programom súčasne.
- Požaduje sa ochrana fyzického adresového priestoru, priradeného jednému používateľskému programu, pred ovplyvňovaním iným používateľskými programom.

Segmentovanie

Deskriptor segmentu je záznam, ktorý tieto informácie o segmente:



- Bázová adresa segmentu (Base). Je to adresa, od ktorej je segment uložený v hlavnej pamäti
- Veľkosť segmentu (Limit). Pre každú inštrukciu alebo údaj v danom segmente musí platiť, že Offset < Limit.
- Atribúty segmentu (Attributes). Patrí sem:
 - Informácia o prítomnosti segmentu v hlavnej pamäti. Ak sa sprístupňovaný segment nenachádza v hlavnej pamäti, použije sa položka 4. a segment sa načíta z vonkajšej pamäte.
 - Informácia o type segmentu (vykonateľný segment, vykonateľný segment s možnosťou čítania, údajový segment len pre čítanie, údajový segment pre čítanie i zápis, zásobníkový segment atď.).
 - Informácia o privilegovanej úrovni segmentu (má význam v prípade, ak viacero programov môže používať ten istý segment)
 - Adresa segmentu vo vonkajšej pamäti (External).



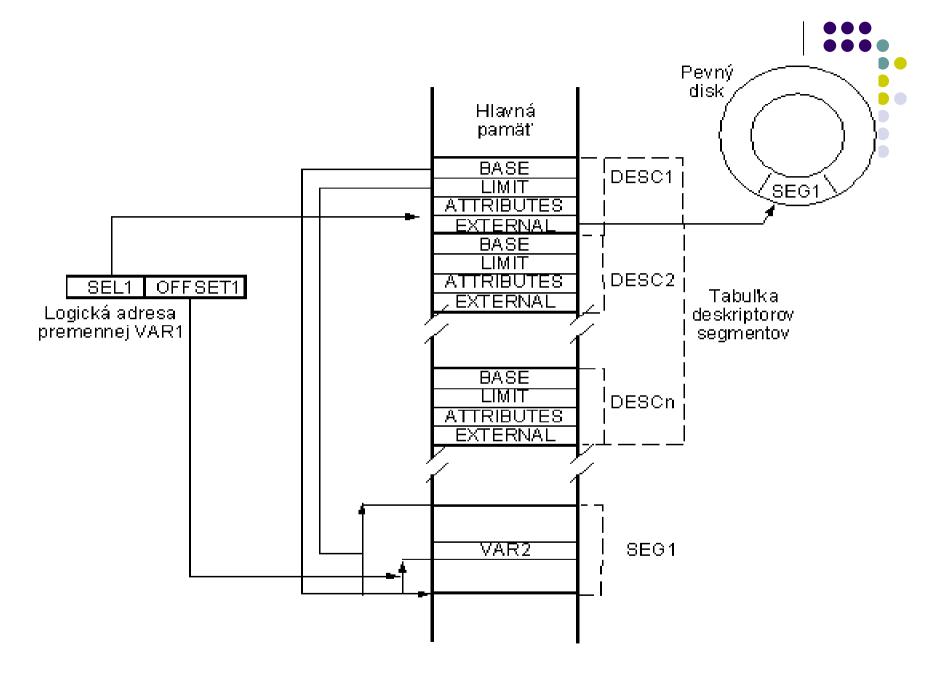
 Logická adresa sa v prípade segmentovania skladá z dvoch častí, ktoré sa spracúvajú samostatne:

Selektor: Posunutie

- Selektor je ukazovateľ do tabuľky deskriptorov segmentov. Selektor sa nachádza v špeciálnom registri a manipuluje sa s ním nezávisle od posunutia. Tabuľka deskriptorov segmentov sa nachádza v hlavnej pamäti a sú v nej deskriptory všetkých segmentov.
- Posunutie (offset) reprezentuje vzdialenosť inštrukcie alebo údaja od začiatku segmentu. Iba posunutie vystupuje ako parameter v adresovej časti inštrukcií.

Premenná VAR1 patrí do segmentu SEG1 a má logickú adresu SEL1:OFFSET1. Segment SEG1 je opísaný deskriptorom DESC1. Fyzická adresa sa vypočíta tak, že k hodnote bázy sa pripočíta hodnota posunutia.

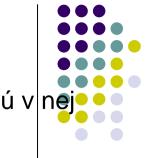
- Výhoda-segment možno presunúť na od ľub. adresy v pamäti, stačí zmeniť v deskriptore hodnotu bázy
- Externá fragmentácia



Krajčovič, T.Počítače, STU, 2000, ISBN 80-227-1399-6

Stránkovanie

- Pri stránkovaní je hlavná pamäť rozdelená na úseky rovnake dĺžky, ktoré sa nazývajú stránkové rámy (page frames).
 Stránkové rámy sú očíslované. Program i údaje sú takisto rozdelené na úseky rovnakej dĺžky stránky. Veľkosť stránky je rovnaká, ako veľkosť stránkového rámu.
- Logická adresa sa v prípade stránkovania skladá z dvoch častí, ktoré sú však na rozdiel od segmentovania spracovávané spoločne - stránka a posunutie.
- Posunutie (offset) reprezentuje vzdialenosť inštrukcie alebo údaja od začiatku stránky.
- Stránka (page) je ukazovateľ do tabuľky deskriptorov stránok.

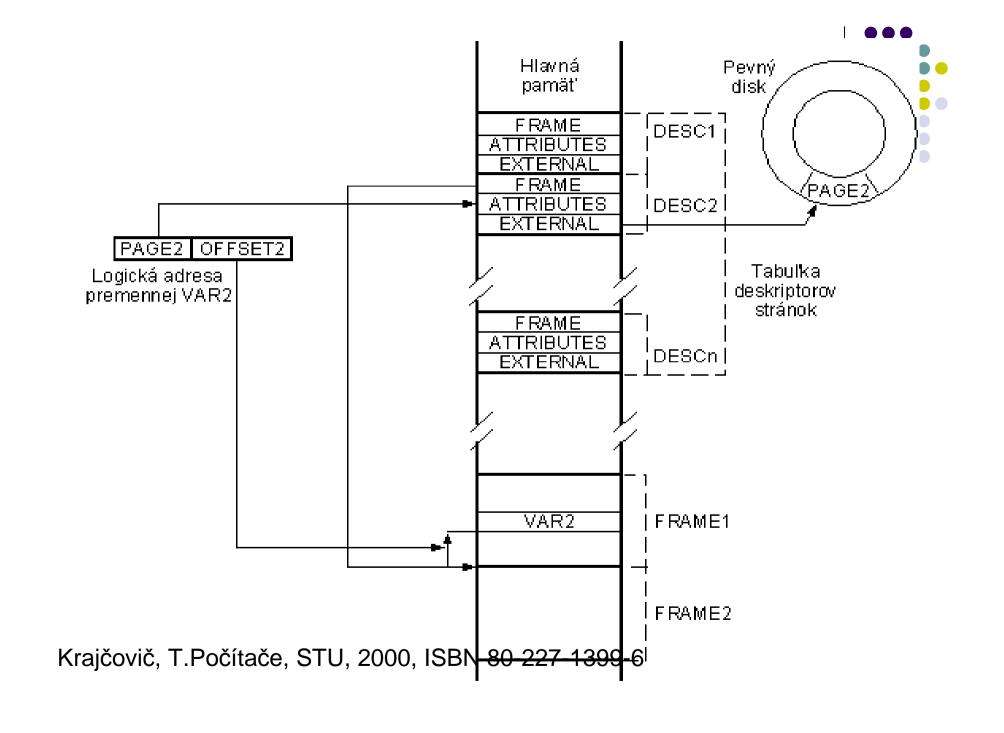


Tabuľka deskriptorov stránok sa nachádza v hlavnej pamäti a sú v nej deskriptory všetkých stránok.

Deskriptor stránky je záznam, ktorý obsahuje tieto informácie o stránke:

- Číslo stránkového rámu (Frame) všetky stránkové rámy majú rovnakú veľkosť, číslo stránkového rámu jednoznačne určuje adresu, od ktorej je stránka v hlavnej pamäti uložená (začiatok stránky).
- Atribúty stránky (Attributes). Tieto sú rovnaké, ako atribúty segmentu.
- Adresa stránky vo vonkajšej pamäti (External).

! Pretože všetky stránky majú rovnakú veľkosť, deskriptor stránky informáciu o veľkosti stránky *nemusí obsahovať*.



- Veľkosť programov alebo údajov nie je vo všeobecnosti celočíselným násol veľkosti stránky

 interná fragmentáciia
- Posledná stránka potom nie je plne využitá. Čím viac programov alebo údajových štruktúr sa v pamäti nachádza, tým viac sa prejavuje nevyužitie hlavnej pamäte.
- Čím je väčšia veľkosť stránky, tým je problém vypuklejší. Na druhej strane, ak zmenšíme veľkosť stránky, výrazne rastie tabuľka deskriptorov. ⇒kompromis.

 Okrem jednoduchého stránkovania sa používa aj viacúrovňové stránkovanie, prípadne sa kombinuje segmentovanie a stránkovanie.
 Vtedy hovoríme o stránkovaných segmentoch.

Komplexné riešenie ochrany pamäte si vyžaduje spoluprácu technických a programových prostriedkov počítača.



!!! Kontrola

- či sa sprístupňovaný segment nachádza v hlavnej pamäti,
- či adresa sprístupňovanej inštrukcie alebo údaja nepresiahla limit segmentu,
- či je použitie daného typu segmentu korektné (napr. či nedochádza k výberu inštrukcie z údajového segmentu)
- či je oprávnené použitie daného segmentu (žiadateľ má dostatočnú privilegovanú úroveň).

$$A + B = B + A$$

$$A.B = B.A$$

$$A + (B + C) = (A + B) + C$$

$$A.(B.C) = (A.B).C$$

$$A.(B + C) = A.B + A.C$$

$$A.(B+C) = A.B + A.C$$
 $A+B.C = (A+B).(A+C)$

$$A + A = A$$

$$A.A = A$$

$$A + 0 = A$$

$$A.0 = 0$$

$$A + 1 = 1$$

$$A. 1 = A$$

$$A + \bar{A} = 1$$

$$A, \bar{A} = 0$$

$$A = \bar{A}$$

$$\overline{A + B} = \overline{A} \cdot \overline{B}$$

$$\overline{A.B} = \overline{A} + \overline{B}$$

$$A.(A+B)=A$$

$$A + A \cdot B = A$$

$$A + \overline{A} \cdot B = A + B$$

$$A.(\overline{A} + B) = A.B$$

$$\bar{A} + A.B = \bar{A} + B$$

$$\bar{A}.(A+B)=\bar{A}.B$$



1.
$$x+y=y+x$$
 $x\cdot y=y\cdot x$ komutatívne zákony
2. $(x+y)+z=x+(y+z)$ $(x\cdot y)\cdot z=x\cdot (y\cdot z)$ asociatívne zákony
3. $(x+y)\cdot z=(x\cdot z)+(y\cdot z)$ $(x\cdot y)+z=(x+z)\cdot (y+z)$ distributívne zákony
4. $\overline{x+y}=\overline{x}\cdot \overline{y}$ $\overline{x\cdot y}=\overline{x}+\overline{y}$ de Morganove zákony
5. $x+x=x$ z zákony idempotentnosti
6. $x+\overline{x}=1$ $x\cdot \overline{x}=0$ zákony komplementárnosti
7. $\overline{x}=x$ z zákony involúcie
8. $x+(x\cdot y)=x$ z zákony absorpcie
9. $x+0=x$ z zákony identity
10. $x+1=1$ z zákon jednotkového sčitovania a nulového násobenia

Logické systémy, Galanová, Kaprálik, Polakovič, FEI STU