1**) Яким чином можна прискорення роботу сторінкової організації пам’яті.**

Принципи сторінкової організації. При сторінкової організації *віртуальне* адресне простір розбивається на ряд сторінок рівного розміру. *Фізичне* адресний простір також розбивається на окремі частини розміром в сторінку. Кожна така частина називається *сторінковим ка*

*дром.*Наведена на рис. 7.10 основна пам'ять містить тільки один сторінковий кадр.

Особливості сторінкової організації та адресації пам'яті розглянемо на конкретному прикладі. Будемо вважати (рис. 7.10), що:

* • віртуальний адресний простір має розмір 64 К (IK = 210) і розділене на 16 сторінок по 4К кожна;
* • фізичне адресний простір основної пам'яті має розмір 32К і містить 8 сторінкових кадрів по 4К кожен.

Таким чином, необхідно 16-розрядний віртуальний адреса відобразити на 15-розрядний фізичну адресу основної пам'яті. Як засоби для відображення віртуальних адрес на фізичні адреси слід використовувати (рис. 7.11):

• *таблицю сторінок,* кожен рядок якої містить:

*■ біт присутності* , що відображає присутність / відсутність (1/0) сторінки в основний пам'яті;

■ 3-розрядний код, що відповідає одному з восьми номерів (0, 1, ..., 7) сторінкового кадру пам'яті;

• *пристрій управління пам'яттю* (MMU - Memory Management Unit), до якого входять:

■ *регістр* для зберігання *таблиці сторінок* ;

■ *вхідний* регістр *R* n, для зберігання 16-розрядної віртуальної адреси, який поділяється на дві частини.

Якщо програма звертається до сторінки, яка відсутня в основний пам'яті, необхідно не тільки викликати необхідну сторінку з диска, але і звільнити для неї місце, відправивши на диск якусь іншу сторінку. Вибір такої сторінки навмання може привести до низки помилок. Наприклад, якщо видалити сторінку з командою, виконання якої викликало помилку, то при спробі викликати наступну команду відбудеться ще одна помилка (через вилучення сторінки з невиконаною командою). Отже, *потрібен алгоритм* для визначення, яку саме сторінку необхідно видалити з пам'яті.

2) **Чи є необхідність реалізувати в системі віртуальну пам’ять, якщо відомо, що загальний обсяг Пам’яті, необхідної для всіх активних процесів, ніколи не перевищить обсяг доступної фізичної Пам’яті? Якщо така необхідність є, то які функції системи віртуальної пам’;яті варто реалізуватиобов’язково, а які - ні?**

**3) Чому розмір сторінки повинен бути степенем числа 2?**

До номера фізичної сторінки приєднується зсув (молодші розряди віртуальної адреси).

Використання в пункті 3) того факту, що розмір сторінки дорівнює ступеню 2, дозволяє застосувати операцію конкатенації (приєднання) замість більш тривалої операції додавання. Це зменшує час отримання фізичної адреси, тим самим підвищуючи продуктивність комп'ютера.

На продуктивність системи з сторінкової організацією пам'яті впливають тимчасові витрати, пов'язані з обробкою сторінкових переривань і перетворенням ВА у фізичну. При часто виникають сторінкових перериваннях система може витрачати більшу частину часу даремно, на свопинг сторінок. Щоб зменшити частоту сторінкових переривань, слід було б збільшувати розмір сторінки. Крім того, збільшення розміру сторінки зменшує розмір таблиці сторінок, а значить, зменшує витрати пам'яті. З іншого боку, якщо сторінка велика, значить, велика і фіктивна область в останній віртуальній сторінці кожної програми. В середньому на кожній програмі втрачається половина обсягу сторінки, що в сумі при великій сторінці може скласти істотну величину. Час перетворення віртуального адреси в фізичний в значній мірі визначається часом доступу до таблиці сторінок. У зв'язку з цим таблицю сторінок прагнуть розміщати в "швидких" запам'ятовуючих пристроях. Це може бути, наприклад, набір спеціальних регістрів чи пам'ять, що використовує для зменшення часу доступу асоціативний пошук і кешування даних.

4) **Перелічіть можливі переваги сторінково-сегментної організації пам`яті порівняно з чистою сегментацією і чисто сторінковою організацією.**

Розглянемо основні переваги сторінкової організації пам'яті порівняно із сег­ментацією. Вони визначаються насамперед тим, що всі сторінки мають одну й ту саму довжину.

* Реалізація розподілу і звільнення пам'яті спрощується. Усі сторінки з погля­ду процесу рівноправні, тому можна підтримувати список вільних сторінок і в разі необхідності виділяти першу сторінку із цього списку, а після звільнен­ня повертати сторінку в список. Із сегментами так чинити не можна, оскільки кожен сегмент можна використати лише за його призначенням (спроба вико­ристати сегмент для іншої мети призведе швидше за все до того, що виникне потреба у сегменті іншої довжини).
* Реалізація обміну даними з диском також спрощується. Для організації тако­го обміну ділянка на диску, яку використовують для зберігання інформації про сторінки, вивантажені з пам'яті {простір підтримки, backing store) може бути теж розбита на блоки фіксованого розміру, рівного розмірові фрейму.

Сторінкова організація пам'яті не позбавлена й недоліків.

* Насамперед цей підхід спричиняє внутрішню фрагментацію, пов'язану з тим, що розмір сторінки завжди фіксований, і в разі необхідності виділення блоку пам'яті конкретної довжини його розмір буде кратним розміру сторінки. У се­редньому розмір невикористовуваної пам'яті становить приблизно половину сторінки для кожного виділеного блоку пам'яті (аналогічного до сегмента). Така фрагментація може бути знижена зменшенням кількості та збільшенням розміру блоків, що виділяються.
* Таблиці сторінок мають бути більші за розміром, ніж таблиці сегментів. Так, для виділення неперервного діапазону пам'яті розміром 100 Кбайт знадобить­ся один елемент таблиці сегментів, що описує сегмент, виділений для цього діапазону. З іншого боку, у разі використання сторінок розміром 4 Кбайт для опису такого діапазону нам знадобиться 25 елементів таблиці сторінок — по одному елементу для кожної сторінки.