4.5 КЕШУВАННЯ В ПРОЦЕСОРІ PENTIUM

У процесорі Pentium кешування використовується у випадках, описаних нижче.

Кешування дескрипторів сегментів в прихованих регістрах. Для кожного сегментного регістра в процесорі є так званий прихований регістр дескриптора. У прихований регістр при завантаженні сегментного регістра поміщається інформація з дескриптора, на який вказує цей сегментний регістр. Інформація з дескриптора сегменту використовується для перетворення віртуальної адреси у фізичну при чисто сегментній організації пам'яті або для отримання лінійної віртуальної адреси при сторінковому механізмі. Доступ до прихованого регістра виконується швидше, ніж пошук і витягання інформації з таблиці сторінок, що знаходиться в оперативній пам'яті. Тому, якщо чергове звернення належатиме до одного з сегментів, дескриптор якого ще зберігається в прихованому регістрі (а ймовірність цього велика), то перетворення адрес буде виконане швидше. Тим самим приховані регістри грають роль кеша таблиці дескрипторів і прискорюють роботу процесора.

Кешування пар номерів віртуальних і фізичних сторінок у буфері асоціативної трансляції TLB (Translation Lookaside Buffer) дозволяє прискорювати перетворення віртуальних адрес у фізичні при сегментно- сторінковій організації пам'яті. TLB є асоціативною пам'яттю невеликого об'єму, призначеною для зберігання інтенсивно використовуваних дескрипторів сторінок. У процесорі Pentium є окремі TLB для інструкцій і даних.

Кешування даних і інструкцій в кеш-пам'яті першого рівня. Ця пам'ять ще називається також внутрішньою кеш-пам'яттю, оскільки вона розміщена безпосередньо на кристалі мікропроцесора і має об'єм 16/32 Кб. У процесорі Pentium кеш першого рівня розділений на пам'ять для зберігання даних і пам'ять для зберігання інструкцій.

Кешування даних і інструкцій в кеш-пам'яті другого рівня. Ця пам'ять називається також зовнішньою кеш-пам'яттю, оскільки вона встановлюється у вигляді окремої мікросхеми на системній платі. Кеш-пам'ять другого рівня є загальною для даних і інструкцій і має об'єм 256/512 Кб. Пошук в кеші другого рівня виконується у разі, коли констатується промах в кеші першого рівня. Для узгодження

даних в кеші другого рівня може використовуватися як наскрізний, так і зворотний запис.

Розглянемо детальніше принципи роботи буфера асоціативної трансляції і кеша першого рівня.

4.5.1 Буфер асоціативної трансляції

У принципі кожна віртуальна адреса викликає звернення до двох фізичних адрес: одне для вибірки відповідного запису з таблиці сторінок, і ще одне – для звернення до адресних даних. А в разі використання дворівневих таблиць сторінок потрібні три операції доступу: до каталогу (розділу) сторінок, до таблиці сторінок і безпосередньо за фізичною адресою. Отже, проста схема віртуальної пам'яті, по суті, подвоює звернення до пам'яті.

Проблема прискорення пошуку вирішується на рівні архітектури комп'ютера. Відповідно до властивості локальності більшість програм впродовж деякого проміжку часу звертаються до невеликої кількості сторінок, тому активно використовується тільки невелика частина таблиці сторінок. Комп'ютер забезпечується апаратним пристроєм для відображення віртуальних сторінок у фізичні без звернення до таблиці сторінок, який має невелику, швидку кеш- пам'ять, що зберігає необхідну на даний момент частину таблиці сторінок.

Для вирішення цієї проблеми більшість схем віртуальної пам'яті використовують спеціальний високошвидкісний кеш для записів таблиць сторінок, який називають буфером швидкого перетворення адреси, або буфером пошуку трансляції (translation lookaside buffer – TLB, або буфер асоціативної трансляції, іноді і асоціативною пам'яттю (рис. 4.16)).

Грунтуючись на правилі «дев'яносто до десяти» (правилі локалізації), яке стверджує, що більшість програм схильна робити величезну кількість звернень до невеликої кількості сторінок, комп'ютер забезпечується невеликим апаратним пристроєм, що служить для відображення віртуальних адрес у фізичні без проходу таблицею сторінок. Цей пристрій знаходиться усередині диспетчера пам'яті і складається з декількох записів, від 8 до 4096. Так, в архітектурі Intel-32 таких елементів до Pentium-4 було 32 (що забезпечується 98% попадань в кеш), починаючи з Pentium-4 – 128 елементів.

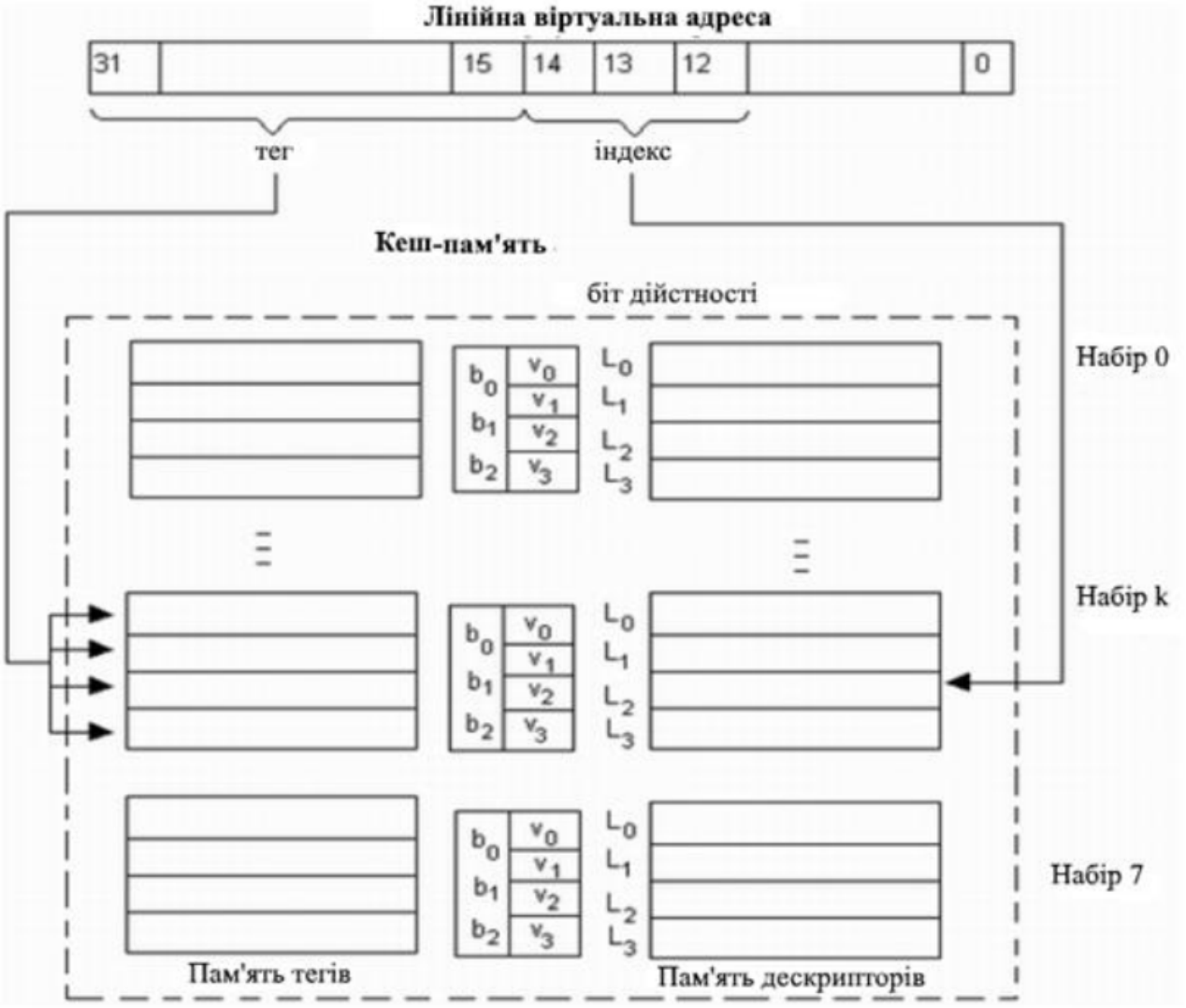

Рисунок 4.16 – Буфер асоціативної трансляції

У буфері TLB кешуються дескриптори сторінок з таблиці сторінок. Для зберігання дескриптора в кеші відводиться один рядок. Кожен рядок доповнений тегом, в якому міститься номер відповідної віртуальної сторінки. Рядки об'єднані по чотири в групи, що називаються наборами.

Таблиця TLB, яка використовується для перетворення адрес інструкцій, має 32 рядки і відповідно до 8 наборів. Номер набору називають індексом (index). Таким чином, шляхом кешування може бути отримана фізична адреса для доступу до 32 сторінок пам'яті, що містять інструкції.

Після того як механізмом сегментації отримана лінійна адреса, він має бути перетворений у фізичну адресу. Для цього передусім необхідно знайти дескриптор сторінки, до якої належить ця адреса, і витягнути з нього номер фізичної сторінки. Звичайна процедура передбачає звернення до таблиці розділів, а потім до таблиці

сторінок. Проте фізична адреса може бути отримана набагато швидше завдяки тому, що в буфері TLB зберігаються копії дескрипторів найбільш інтенсивно використовуваних сторінок. Тому перед тим, як почати порівняно тривалу процедуру перетворення адрес, робиться спроба виявити потрібний дескриптор сторінки в швидкій асоціативній пам'яті TLB. Потім на підставі номера фізичної сторінки, отриманого з TLB, обчислюється фізична адреса.

При пошуку даних в TLB використовується лінійна віртуальна адреса. Розряди 12-14 використовуються як індекс набору. Далі перевіряються біти дійсності (V) усіх рядків вибраного набору. На початку роботи кеш-пам'яті біти дійсності усіх рядків скидаються в нуль. Біт дійсності набуває значення 1, коли у відповідному рядку міститься достовірна інформація, і скидається в нуль, коли рядок оголошується вільним, в результаті роботи алгоритму заміщення. Для усіх дійсних рядків виконується асоціативна процедура порівняння тегів із старшими розрядами (15-31 розряд) лінійної віртуальної адреси. Якщо сталося кеш- попадання, то номер фізичної сторінки швидко поступає в схему формування фізичної адреси.

Якщо стався промах і потрібного дескриптора в TLB немає, то запускається багатоетапна процедура перетворення адреси, що включає звернення до таблиць розділів і сторінок. Коли потрібний дескриптор відшукується в таблиці сторінок, він копіюється в TLB. Номер набору, в який записується кешований дескриптор, визначається трьома молодшими розрядами номера віртуальної сторінки (розряди 12-14 лінійної віртуальної адреси).

Проте оскільки в наборі є чотири рядки, необхідно визначити, в яку саме потрібно помістити кешовані дані. Дескриптор записується або в перший вільний рядок, що попався, або, якщо усі рядки зайняті, в рядок, до якого найдовше не зверталися. Ознакою зайнятості рядка служить біт дійсності V, наявний у кожного рядка. Якщо V=0, значить, рядок вільний для запису в нього нового вмісту. Для визначення рядка, який не використовувався довше за всіх інших в цьому наборі, застосовується спрощений варіант алгоритму PseudoLRU (Pseudo Least Recently Used). Цей алгоритм грунтується на аналізі трьох біт: b0, b1, b2, що називаються бітами звернення. Бити звернення приписуються набору і встановлюються відповідно

до алгоритму, наведеному на рис. 4.17. Тут L0, L1, L2, L3 означають послідовні рядки набору. На заміну вибирається один з таких рядків:

- L0, якщо b0=0 і b1=0;
- L1, якщо b0=0 і b1=1;
- L2, якщо b0=1 і b2=0;
- L3, якщо b0=1 і b2=1.

Можна легко показати, що ця процедура не завжди призводить до вибору дійсно довшого за всіх рядка, що не викликався. Нехай, наприклад, звернення до рядків виконувалися в наступній хронологічній послідовності: L0, L2, L3, L1, тобто найближче за часом звернення було до рядка L1, найдовше ж не було звернень до рядка L0. Бити звернення в даному випадку набудуть наступних значень. Оскільки останнє за часом звернення було до рядка з пари (L0, L1), значить, b0=1. А в парі (L2, L3) останнє звернення було до L3, отже, b2=0. Звідси, за правилом, наведеним вище, на заміну вибирається рядок L2, замість рядка L0, до якого насправді найдовше не було звернень.

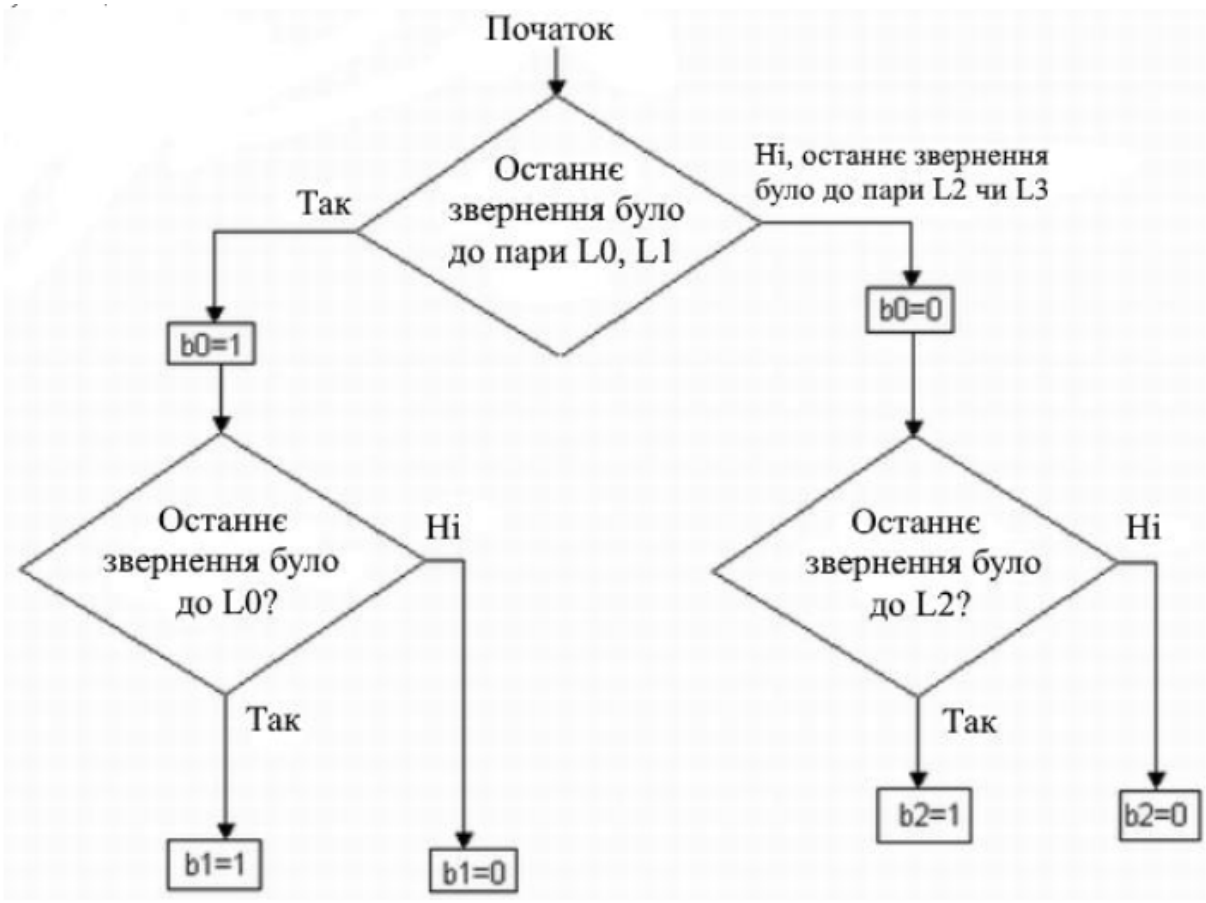

Рисунок 4.17 – Алгоритм установки бітів звернення

Проте в більшості випадків цей алгоритм дає результат, співпадаючий з оптимальним. Наприклад, для послідовності L0, L3, LI, L2 біти звернення мають значення b0=0, b1=0, звідси точне рішення – L0. Навіть у разі помилки (ймовірність якої складає 0,3) рішення, знайдені за алгоритмом PseudoLRU, близькі до оптимальних. Так, у першому прикладі замість рядка L0, що є правильним рішенням, алгоритм дав найближчий до нього за часом звернення рядок L2.

Незважаючи на те що алгоритм PseudoLRU дає в загальному випадку наближені рішення, він широко застосовується при кешуванні, оскільки є швидким і економічним, що надзвичайно важливо для кеш-пам'яті.

Таким чином, у буфері TLB процесора Pentium використовується комбінований спосіб відображення кешованих даних на кеш-пам'ять: пряме відображення дескрипторів на набори і випадкове відображення на рядки в межах набору.

Наявність TLB дозволяє в переважному числі випадків замінити порівняно довгу процедуру перетворення адрес, пов'язану з декількома зверненнями до оперативної пам'яті, швидким пошуком в асоціативній пам'яті.

4.5.2 Кеш першого рівня

Кеш першого рівня використовується на етапі обробки запиту до основної пам'яті за фізичною адресою. Робота кеш-пам'яті першого рівня має багато спільного з роботою буфера TLB. У TLB одиницею зберігання є дескриптор, а в кеші першого рівня – байт даних. Оновлення даних в кеші відбувається блоками по 16 байт. Таким чином, молодші 4 біти фізичної адреси байта можуть інтерпретуватися як зміщення в блоці, а старші розряди – як номер блоку (рис. 4.18).

Для зберігання блоків даних в кеші відводяться рядки, що також мають об'єм 16 байт. Рядки об'єднані в набори по чотири. При об'ємі кеша 16 Кб в нього входять 256 (28) наборів (16*210 = 24*210/24*22 = 28).

При копіюванні даних в кеш номери блоків основної пам'яті прямо відображаються на номери наборів. Для цього в адресі основної пам'яті, що належить до одного з байтів, що входять у блок, значення 8 бітів, що знаходяться перед бітами зміщення, інтерпретується як номер набору в кеш-пам'яті. Інші старші біти адреси надалі використовуються як тег.

Так само як в TLB, вибір рядка в наборі здійснюється на основі аналізу бітів дійсності і бітів звернення за алгоритмом PseudoLRU. Блок даних заноситься в рядок кеш-пам'яті разом зі своїм тегом. Біт дійсності рядка встановлюється в 1.

При виникненні запиту на читання з основної пам'яті спочатку робиться спроба знайти дані в кеші. За індексом, витягнутим з адреси запиту, визначається набір, в якому можуть знаходитися шукані дані. Потім для рядків цього набору виконується асоціативний пошук. Старші розряди адреси із запиту порівнюються з тегами усіх рядків набору. Якщо для якого-небудь рядка фіксується співпадання, це означає, що сталося кеш-попадання, і з відповідного рядка витягається байт, зміщення якого відносно початку рядка визначається чотирма молодшими розрядами з адреси запиту.

Для узгодження даних в кеші 1-го рівня використовується метод наскрізного запису, тобто при виникненні запиту на запис оновлюється не лише вміст відповідного елементу основної пам'яті, але і його копія в кеш-пам'яті.

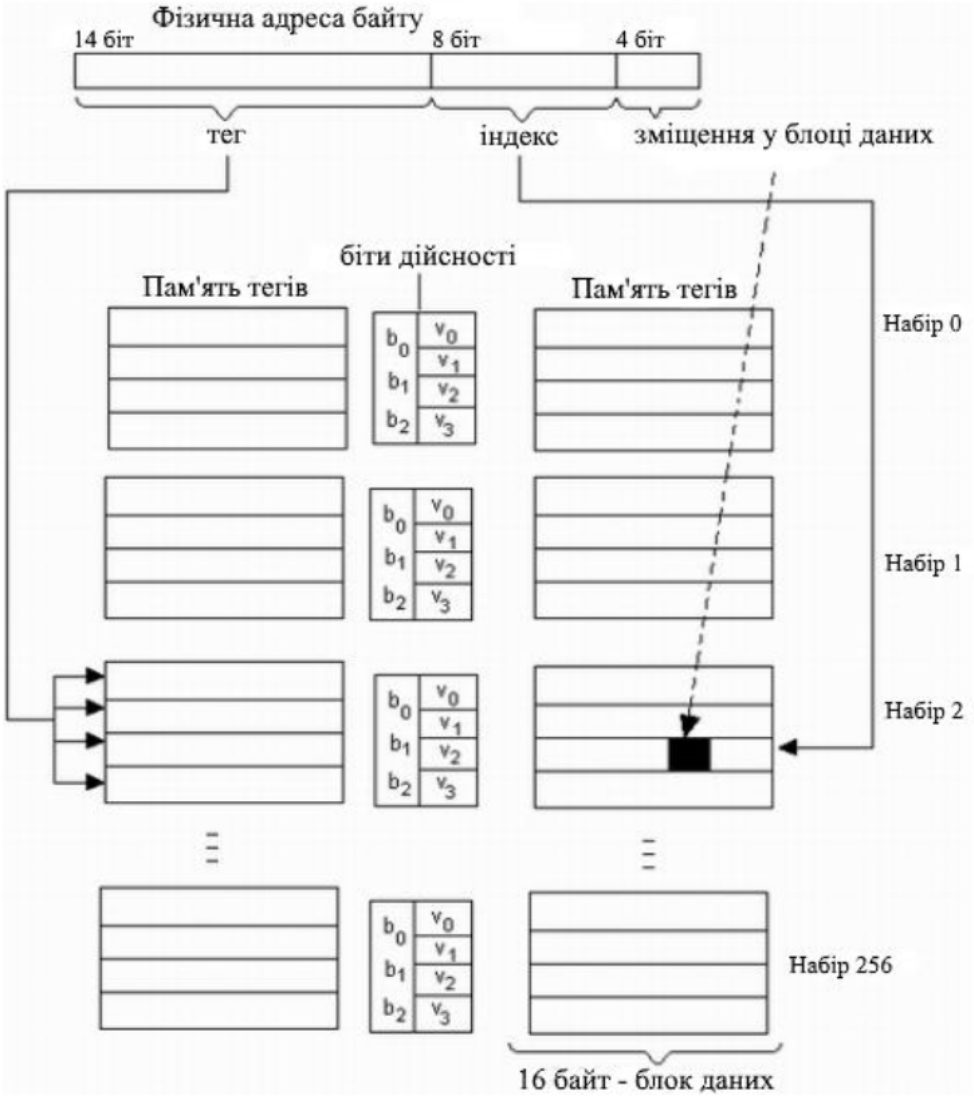

Рисунок 4.18 – Кеш першого рівня процесора Pentium