

ПРОЦЕСОР ІЗ ДИНАМІЧНИМ ПЛАНУВАННЯМ КОМАНД

Проведено дослідження архітектур суперскалярних процесорів, шляхи підвищення їх продуктивності та розглянуто питання розв'язання проблеми виникнення конфліктів за даними за допомогою станцій резервування

Найбільш суттєвий метод збільшення продуктивності однопроцесорної системи – організація ідей паралелізму на рівні команд. Розвиток цих ідей привів до суперконвейерних та суперскалярних архітектур. Більшість сучасних мікропроцесорів високої продуктивності побудовано з використанням суперскалярної структурної організації.

Часова діаграма (рис. 1) порівнює роботу процесорів. Верхня частина діаграми ілюструє роботу “звичайного” конвеєра, по якому проаналізовано швидкодію суперскалярної і суперконвейерної схем, що обробляють однакову кількість команд за той же час. При цьому суперконвейерна схема програє суперскалярній при обробці початкового фрагмента програми.

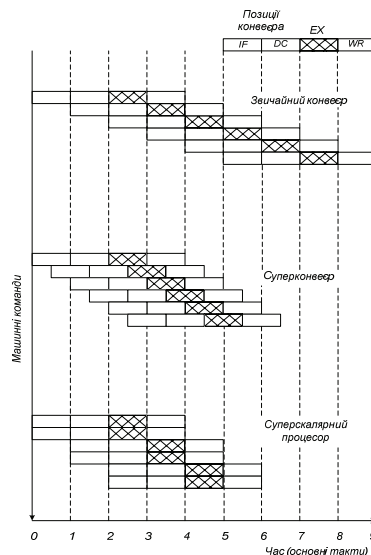


Рис. 1. Часові діаграми роботи процесорів

До класу суперскалярних відносяться процесори, в яких використовується кілька функціональних вузлів у вигляді незалежних конвеєрів операцій. Кожен з них включає декілька робочих позицій, і одночасно на ньому може опрацьовуватися кілька машинних команд. Типова архітектура суперскалярного мікропроцесора наведена на рис. 2.

Його основними компонентами є: функціональні модулі – виконання операцій із плаваючою (*FPU*) і фіксованою (*ALU*) комами, пристрій завантаження/збереження, файли регістрів, роздільна кеш-пам'ять команд і даних, а також допоміжні модулі, які забезпечують динамічне планування обчислювального процесу – пристрій зв'язку з кеш-пам'яттю 2-го рівня, блок переупорядкування команд і блок попередньої дешифрації.

В результаті аналізу факторів, від яких залежить продуктивність комп'ютерної системи, визначено, що найбільш вагомими з них є архітектура процесора та організація пам'яті. Результати цього аналізу подані на рис. 3. Права гілка відображає методи організації пам'яті, а ліва – тенденції в розвитку архітектур процесорів.

Досліджено два сімейства комп'ютерів із суперскалярною організацією: *Intel Pentium Dual-Core* та *AMD ATOM*, які втілюють більшість із існуючих тенденцій у проектуванні комп'ютерних систем. Сімейства *Intel Pentium Dual-Core* та *AMD ATOM* є представниками

комп'ютерів із *CISC*-архітектурою, у той час, як *AMD ATOM* являє собою комп'ютер із *RISC*-архітектурою.

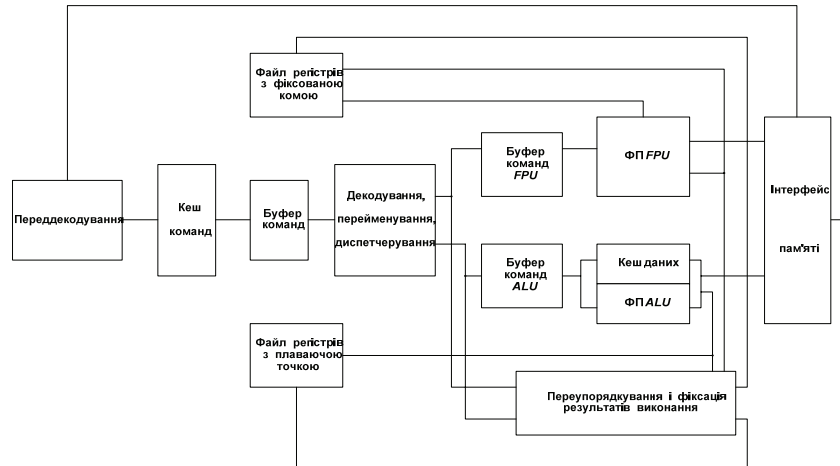


Рис. 2. Архітектура суперскалярного процесора



Рис. 3. Шляхи підвищення продуктивності комп'ютерних систем

При дослідженні архітектур суперскалярних процесорів було використано модель вікна виконання команд (рис. 4). При виконанні програми мікропроцесор просуває за статичною структурою програми вікно виконання, де команди можуть виконуватися паралельно при умові відсутності залежності. Залежність за управлінням майже в будь-якій статистичній програмі займає 1/5 від обсягу програмного коду (програми).

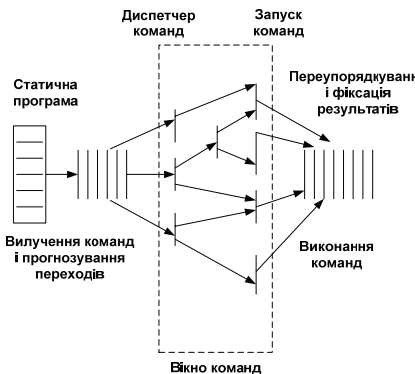


Рис. 4. Обробка команд із суперскалярною організацією

В суперскалярних архітектурах аналізується проблема мінімізації збитків по управлінню. Для усунення залежностей, викликаних командами переходів, використовується

метод передбачення, який дозволяє вилучати й умовно виконувати команди передбачуваного переходу (рис. 4).

Для прогнозування переходу використовується автоматний підхід (рис. 5). Використовується 2 стани: результати минулого прогнозу і того, який на даний час виконується. Врахування передісторії дозволяє суттєво підвищити вірогідність прогнозу. Аналіз різних джерел прогнозу для різних програмних кодів подано на рис. 6 у вигляді інтегрованого графіка прогнозу виконання різних програмних кодів.

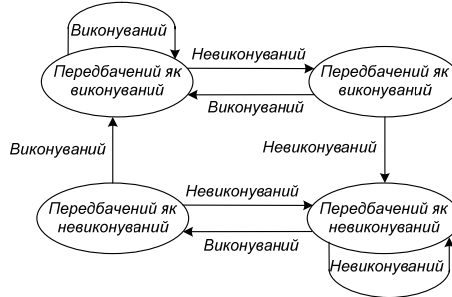


Рис. 5. Діаграма стану двобітової схеми прогнозування

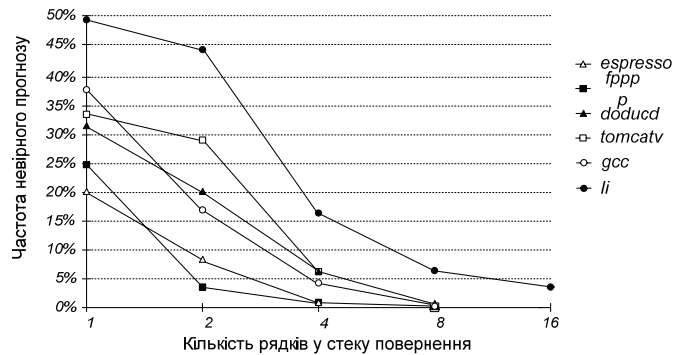


Рис. 6. Точність прогнозу для адрес повернення

Не менш важливим є те, що команди, поміщені у вікно виконання, можуть бути залежні за даними (деякі з яких можуть бути усунені шляхом перейменування регістрів. Дійсною залежністю є тільки “читання після запису”. Після видалення зайвих залежностей за управлінням і даними команди можуть виконуватися паралельно). Розв’язання цієї проблеми можливе 2-ма способами: централізований спосіб динамічного планування та децентралізований спосіб. Дослідимо децентралізований спосіб, який є більш ефективним, але вимагає значних апаратних затрат. При сучасному стані мікроелектронних технологій ця проблема практично знята. Децентралізація полягає у використанні у кожному операційному блоці спеціальних пристроїв, які отримали назву станції резервування. Централізована схема подана на рис. 7.

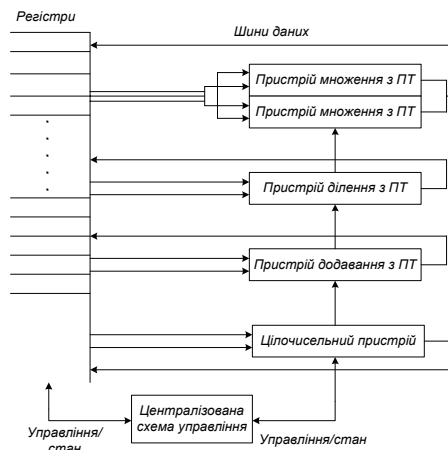


Рис. 7. Централізована схема управління

Структура блоку усунення конфліктів подана на рис. 8.

Всі буфери і станції резервування мають поля тегів, що використовуються для управління конфліктами.

Призначення пристроїв:

1. *R FPU* (блок реєстрів з плаваючою точкою) призначений для зберігання операндів та результатів. Є програмно доступним. Кількість реєстрів – 8 розрядністю 64.

2. *BUF* завантаження та *BUF* запису зберігають дані, які надходять з пам'яті і записуються у пам'ять.

3. *FIFO* команд *FPU* (черга команд із плаваючою точкою) – буфер з одним входженням для кожної ініційованої на виконання команди, який називається переупорядковувачем. Використовується для встановлення порядку команд при перериваннях (призупиненнях).

4. *CP* (станція резервування) є індивідуальною для кожного виконавчого блоку і характеризує зайнятість відповідного операційному блоку, код операції, значення операндів та станції резервування, які є джерелами операндів.

5. *ADD* та *MULT* – пристрій додавання та пристрій множення, кількістю 3 і 2 відповідно. Призначені для одночасного виконання операцій.

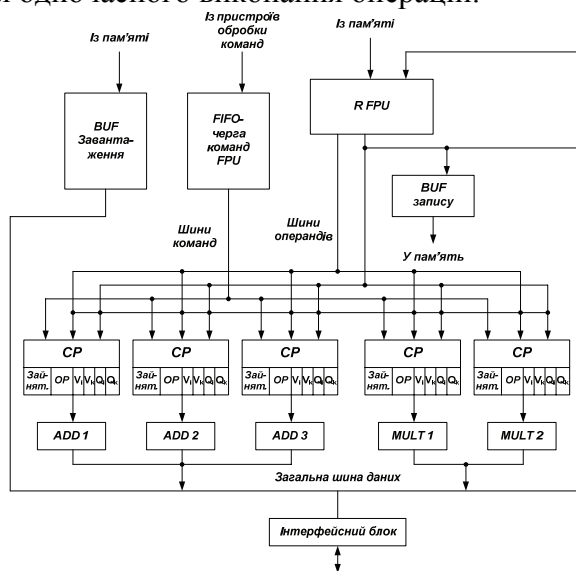


Рис. 8. Структура блоку усунення конфліктів

Висновки

В роботі було досліджено:

1. Структура суперскалярного процесора.
2. Розроблено блок динамічного планування виконання команд в процесорі з суперскалярною архітектурою.
3. Розроблена структура блоку динамічного планування на основі станцій резервування.

Суперскалярні процесори є новим напрямом у розвитку обчислювальної техніки. Такий напрям може значно підвищити її продуктивність, а також і ефективність її використання в світі сучасних інформаційних технологій.

Список літератури

1. Шпаковский Г.И. Организация параллельных ЭВМ и суперскалярных процессоров: Учеб. пособие. – Мн.: Белгосуниверситет, 1996. – 296 с.
2. В.В.Воеводин, Вл.В.Воеводин. Параллельные вычисления. Издательство БХВ-Петербург, 2002. – 609 с.