

МОДЕЛЬ ПАРАЛЕЛЬНОГО СОРТУВАЛЬНИКА ДЛЯ АСОЦІАТИВНОГО ПРОЦЕСОРА

¹Вінницький національний технічний університет

Процес сортування та вибірки за ключем є основною процедурою у багатьох пошукових системах таких, як бази даних та пошукові системи в Інтернеті. Водночас сучасні обчислювальні засоби вимагають ефективних методів і засобів, пов'язаних з асоціативною обробкою інформації під час розроблення програмного та апаратного забезпечення. Тому виникає потреба у високошвидкісному необчислювальному (асоціативному) обробленні значних обсягів інформації, що вимагає відповідної організації та вдосконалення технічних засобів сортування. Відомі алгоритми та засоби сортування чисел дозволяють регулювати інтенсивність виконання цього процесу та підвищувати його ефективність, використовуючи паралельні пристрої, але вони вимагають значних апаратних витрат. Тому метою подальших досліджень є розробка нових та вдосконалення відомих методів сортування з орієнтацією на зменшення апаратних витрат та збільшення швидкості цього процесу. В роботі запропоновано структурну схему сортувальника як обчислювальної частини асоціативного процесора, яка має регулярну логічну структуру і паралельно-последовні зв'язки між блоками обробки даних. Це значно спрощує «розміщення» сортувальника в мікросхемі програмованої логічної ІС (ПЛІС). Крім того, функціонально у сортувальнику реалізовано багатофункціональність обробки числових масивів даних завдяки формуванню рангів елементів вхідного масиву. Це дозволяє визначити не тільки екстремальні елементи числового масиву, але й елемент, що займає середнє значення у відсортованому масиві, що є необхідною умовою для швидкісної медіанної фільтрації зображень. В запропонованому сортувальнику в процесі сортування використовуються швидкісні операції інкременту/декременту на масивах лічильників замість витратної за часом операції попарного порівняння паралельно для всіх елементів масиву з подальшою їх перекомутацією.

Ключові слова: сортувальник, медіанна фільтрація, асоціативний процесор, ранжування.

Вступ

Процедури сортування та вибірки за ключем [1], [2] є базовими у багатьох пошукових програмах, наприклад, у базах даних та пошукових системах INTERNET [3]. Разом з цим, сучасні обчислювальні засоби мають потребу в ефективних методах та засобах асоціативної обробки інформації для розробки програмного й апаратного забезпечення [4]—[7]. Це пов'язано з тим, що є необхідність у швидкісній необчислювальній обробці великих масивів інформації, що у свою чергу, потребує відповідної організації перспективних технічних засобів, зокрема сортувальників для паралельної обробки масивів числових даних [5]—[8].

Актуальність тематики пов'язана з розв'язанням таких задач, як прискорення спрацювання систем управління базами даних (СУБД) [9], прискорення пошуку інформації за ключем у пам'яті даних [5], [6], планування паралельної обробки в багато процесорних системах [8]—[10] та удосконалення роботи цифрових медіанних фільтрів для передобробки зображень [11]. Так у роботах [12], [13] подано алгоритми та засоби, які дають можливість регулювати інтенсивність сортування чисел, а також підвищують ефективність через паралельне використання обладнання, але є складними і потребують значних апаратних витрат. Тому необхідно вдосконалювати інші методи для зменшення апаратних витрат та збільшення швидкодії сортування. Наприклад, у статі [14] запропоновано новий підхід до асоціативної паралельної обробки без операції порівняння, а через застосування паралельно операції декременту до всіх елементів масиву чисел. В результаті час ви-

конання базових операцій пошуку пропорційний максимальному значенню серед чисел у заданому масиві даних і не залежить від їх кількості, що значно прискорює процес обробки значних масивів даних. Такий спосіб асоціативної обробки є послідовно-порозрядною (вертикальною) обробкою [6]—[8].

Метою роботи є розширення функціональних можливостей сортувальника у складі асоціативного процесора за рахунок формування рангів елементів масиву чисел в процесі їх паралельного оброблення.

Постановка задачі

Найефективнішим та найнаочнішим у попередній обробці зображень є застосування результатів

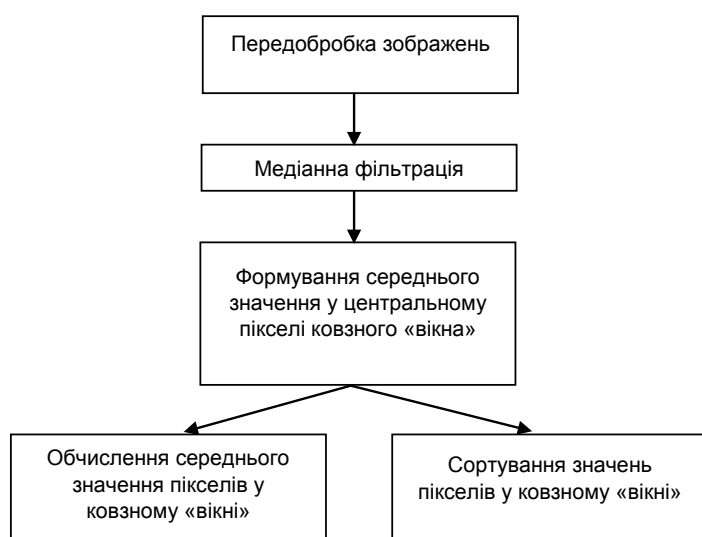


Рис. 1. Особливості застосування медіанної фільтрації

сортування масиву чисел (пікселів) в процесі медіанної фільтрації [15], де середнє значення відліків ковзного «вікна» фільтра можна отримати в результаті їх сортування, а не обчислення середнього арифметичного значення всіх елементів цього «вікна» (рис. 1). Такий підхід значно прискорює процес медіанної фільтрації напівтонових зображень для визначення значення центрального елемента «вікна» апертури [11].

На рис. 2 подано приклад застосування сортування для визначення значення центрального елемента у «вікні» розміром 3×3 пікселя при медіанній фільтрації зображення [11]. Цей приклад наочно ілюструє ефективність застосування сортування замість підрахунку середнього арифметичного елементів «вікна» у вигляді

$$\alpha_{\text{сеп}} = \frac{1}{n} \sum_{i=1}^n a_i, \quad (1)$$

де a_i — числове значення i -го пікселя у «вікні».

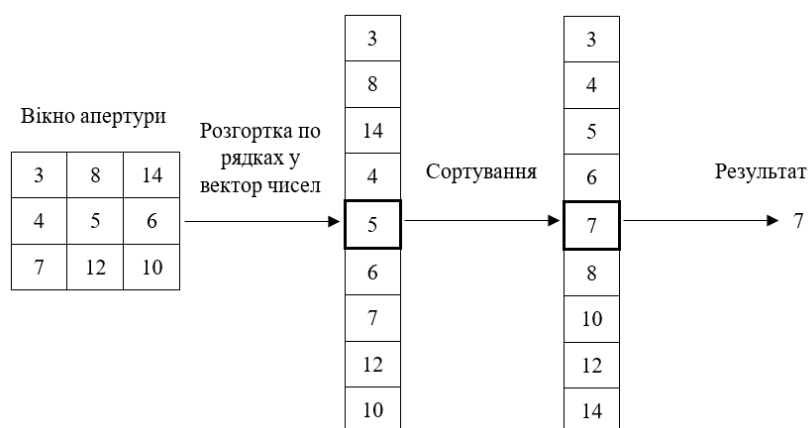


Рис. 2. Процес сортування при медіанній фільтрації

При цьому, на практиці апертура фільтра для спрощення алгоритмів обробки даних, як правило, встановлюється з непарною кількістю відліків [11]. Отже, для прискорення процесу медіанної фільтрації у першу чергу необхідно використати швидкісний метод сортування незначного масиву чисел. При цьому, як видно з рис. 2, практичний інтерес становить центральний елемент відсортованого масиву. Якщо використовувати сортування з ранжуванням елементів масиву, то шуканому елементу буде присвоєно середній ранг, за яким його буде легко зчитати з виходу сортувальника.

У статті [14] запропоновано пошуковий метод, в якому відбувається визначення екстремальних (мінімального/максимального) чисел у масиві через швидкісну операцію декремента, яка застосовується одночасно до всіх елементів масиву. Але в процесі цієї обробки неможливо визначити елемент вхідного масиву чисел, який займає середнє значення у відсортованому масиві. Для такої вертикальної асоціативної обробки можливим є формування рангів для чисел вхідного масиву, що дозволить в подальшому визначити не тільки екстремальні числа, але й середнє число у відсортованому масиві.

Особливості асоціативного процесора для сортування числових даних

На рис. 3 показано структурну схему обчислювальної частини асоціативного процесора, в якій реалізовано сортування масиву чисел з формуванням їх рангів. Структурна схема містить такі базові вузли, як регістрова пам'ять, сортувальник та пам'ять рангів. Регістрова пам'ять містить у собі масив з n регістрів, сортувальник, у свою чергу, містить масив n вхідних лічильників та аналізатор реакцій, а пам'ять рангів — масив n вихідних лічильників, де n — розмірність вхідного масиву чисел. Сам процес сортування виконується у сортувальнику, оскільки регістрова пам'ять використовується для збереження вхідних даних, а пам'ять рангів — для формування та збереження рангів елементів вхідного масиву чисел [16].

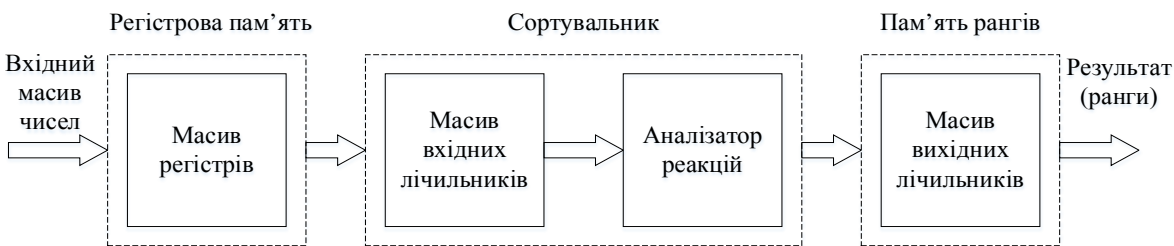


Рис. 3. Структурна схема обчислювальної частини асоціативного процесора

На рис. 4 показано функціональну схему обчислювальної частини асоціативного процесора, що містить масив регістрів RG , масив вхідних лічильників CT , а також елементи маски ME , багатовхідні логічні елементи I та АБО у складі аналізатора реакцій (показаний пунктиром) і масив вихідних лічильників CT , що входять до пам'яті рангів. Також на рис. 4 показано інформаційні входи a_1, \dots, a_n , вхід початкового встановлення $Reset$, вхід початкового стану Set , вхід тактових імпульсів Tl , вихід рангів r_1, \dots, r_n , вихід завершення сортування — Кінець.

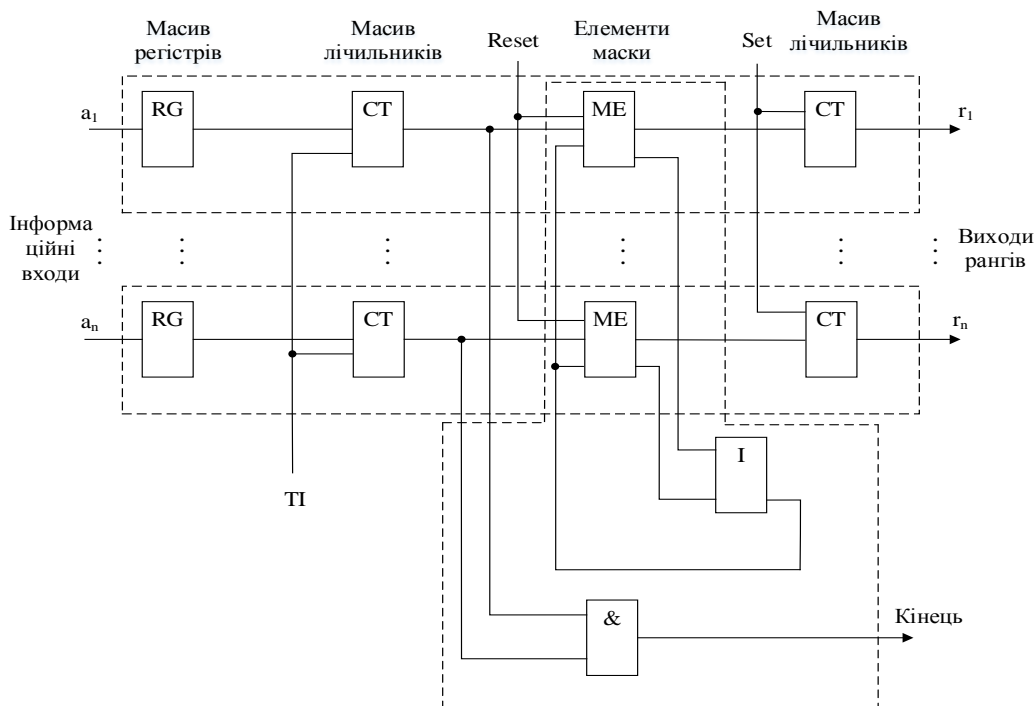


Рис. 4. Функціональна схема обчислювальної частини асоціативного процесора

Отже, обчислювальна частина асоціативного процесора (рис. 4) організована як регулярна структура з однонаправленим паралельно-послідовним зв'язком між блоками обробки даних (рис. 3). Зворотний зв'язок використовується в аналізаторі реакцій через багатовхідний елемент АБО для корегування стану елементів маски МЕ.

На рис. 4 пунктиром також показано n горизонтальних гілок для кожного з n елементів вхідного масиву чисел. Кожна гілка складається з регістра RG, вхідного лічильника СТ, елемента маски МЕ та вихідного лічильника СТ.

Асоціативний процесор працює в такий спосіб (рис. 4) [16]:

Крок 1. Встановлення в початковий одиничний стан елементів маски МЕ за сигналом Reset, оскільки кожний з них містить тригер, тобто

$$m_i = 1. \quad (2)$$

Крок 2. Запис в масив регістрів RG векторного масиву чисел вигляду

$$A = a_1, \dots, a_2, \dots, a_n \quad (3)$$

та у вихідні лічильники СТ пам'яті рангів початкового одиничного значення за сигналом Set

$$r_i = 1, \quad (4)$$

причому масив рангів має вигляд

$$R = r_1, \dots, r_i, \dots, r_n. \quad (5)$$

Крок 3. Перезапис масиву чисел A (3) у вхідні лічильники СТ сортувальника зі збереженням даних у регістровій пам'яті.

Крок 4. Зворотна лічба (операція декремента) у масиві вхідних лічильників СТ сортувальника з надходженням кожного тактового імпульсу ТІ.

Крок 5. Перевірка умови обнулення l -го вхідного лічильника СТ сортувальника

$$\exists a_l = 0 \quad (6)$$

з обнуленням (маскуванням) l -го елемента маски МЕ, тобто

$$m_l = 0, \quad (7)$$

а також збільшення на одиницю (операція інкремента) стану всіх вихідних лічильників СТ пам'яті рангів, крім замаскованого l -го, стан якого зберігається.

Крок 6. Перевірка умови обнулення всіх вхідних лічильників СТ сортувальника

$$\forall a_i = 0, \quad i = \overline{1, n}. \quad (8)$$

Якщо умова (8) виконується, то відбувається перехід до кроку 7. У протилежному випадку перехід до кроку 4.

Крок 7. Маскування всіх елементів маски МЕ, збереження станів вихідних лічильників СТ пам'яті рангів і формування сигналу «Кінець» аналізатором реакцій.

Оскільки ємність вихідних лічильників СТ пам'яті рангів дорівнює $k = \log_2 n$, то у кожному з них може бути сформовано максимальний ранг будь-якого з чисел масиву A (3). Крім того, можлива візуалізація результатів ранжування через під'єднання до масиву вихідних лічильників СТ пам'яті рангів блока індикації на семисегментних індикаторах [17].

Базові характеристики сортувальника

В роботі [14] для асоціативного процесора (АП), який вибрано як прототип, що виконує процедуру пошуку екстремальних чисел, визначено загальний час цієї асоціативної вибірки таким чином:

$$T_a = (2^q - 1)\tau_{ct} + t_{lg} + t_{wr} = (2^q - 1)\tau_{ct} + t_{wr} + 4\tau_{lg}, \quad (9)$$

де τ_{ct} — час затримки при лічбі (операції декремента) двійкового лічильника; t_{wr} — час, що витрачається відповідно на запис результату у тригер; t_{lg} — час, що витрачається на логічні операції у логічному вузлі на елементах АБО/І; τ_{lg} — час спрацювання логічних елементів; q — розрядність даних.

Для запропонованого сортувальника на лічильниках час асоціативної обробки становить

$$T_s = (2^q - 1)\tau_{ct} + t_{lg} + t_{ct} = 2^q \tau_{ct} + t_{wr} + \tau_{lg}, \quad (10)$$

оскільки час спрацювання елементів аналізатора реакцій розраховується таким чином [16]:

$$t_{lg} = t_{wr} + \tau_{lg}. \quad (11)$$

В свою чергу апаратна складність АП [14] становить

$$K_a = (K_{ct} + K_t + 5K_{lg})n + 2K_{lg} = (K_{ct} + K_t)n + (5n + 2)K_{lg}, \quad (12)$$

де K_{ct} — складність лічильника; K_t — складність тригера; K_{lg} — складність логічного елемента; n — розмірність вхідного масиву чисел.

Відповідно апаратна складність сортувальника матиме вигляд

$$K_s = (2K_{ct} + K_t + 3K_{lg})n = (2K_{ct} + K_t)n + 3nK_{lg}. \quad (13)$$

Оскільки для будь-якої асоціативної операції бажано визначити її максимальну часову характеристику, то вважатимемо, що кількість слів $n = 32$, а їх розрядність дорівнює $q = 8$. Формули для розрахунку часових і апаратних витрат для АП [14] та сортувальника наведено у таблиці, де враховано, що максимальне 8-розрядне двійкове число потребує 255 тактів операції декременту.

Порівняльна характеристика асоціативних процесорів

Назва	Апаратна складність	Часові витрати	Функціональні можливості
АП з паралельно-последовною обробкою з пам'яттю на лічильниках	$K_a = (K_{ct} + K_t) \cdot 32 + 162K_{lg}$	$t_a = 255\tau_{ct} + t_{wr} + 4\tau_{lg}$	Пошук за ключем, визначення екстремальних чисел
Сортувальник на лічильниках	$K_s = (2K_{ct} + K_t) \cdot 32 + 96K_{lg}$	$t_s = 256\tau_{ct} + t_{wr} + \tau_{lg}$	Сортування, ранжування, визначення екстремальних чисел

Аналіз таблиці показав, що максимальний час асоціативної обробки у запропонованому сортувальнику фактично дорівнює часу обробки у розглянутому прототипі АП. Враховуючи реалізацію запропонованих асоціативних процесорів на ПЛІС з орієнтовним часом спрацювання базових вузлів в таких мікросхемах у наносекундному діапазоні [18], [19], можна отримати кінцевий результат у мілісекундному діапазоні. У запропонованому сортувальнику для операцій декременту/інкременту використовуються два масиви лічильників, що значно збільшує його складність. Але логічна частина сортувальника відрізняється простотою та регулярністю зв'язків у порівнянні з відповідним логічним вузлом АП. Разом з тим, простота та регулярність структури і зв'язків є важливим фактором для реалізації сортувальника на перспективній елементній базі — ПЛІС [18], [19]. Крім того, сортувальник реалізує багатофункціональну обробку масиву чисел за рахунок формування їх рангів, що дозволяє визначити будь-який елемент вхідного масиву за його рангом (місцем розташування) у відсортованому масиві.

Висновки

1. Аналіз методів та засобів сортування масивів чисел показав актуальність застосування апаратних методів сортування елементів масивів чисел, в чому є потреба, зокрема, при медіанній фільтрації цифрових зображень.

2. Запропонована структурна схема сортувальника як обчислювальної частини асоціативного процесора має регулярну логічну структуру і паралельно-последовні зв'язки між блоками обробки даних, що в подальшому значно спростить її «розміщення» у мікросхемі ПЛІС через відсутність $n/2$ схем порівняння та складну мережу перекомутувань.

3. У функціональному плані сортувальнику властива багатофункціональність обробки масиву чисел за рахунок формування їхніх рангів, що дозволяє визначити не тільки екстремальні елементи масиву чисел, але й елемент, що займає середнє значення у відсортованому масиві. Крім того, в процесі сортування використовуються операції інкремента/декременту на лічильниках, що дозво-

ляє застосувати їх паралельно до всіх елементів масиву, тобто реалізувати вертикальну асоціативну обробку.

4. Прискорення процесу запропонованого сортування з ранжуванням чисел масиву досягається за рахунок максимального паралелізму обробки $O(n)$, а також відсутності часових витрат на виконання операції попарного порівняння та перекомутації елементів масиву чисел.

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

- [1] Г. Лорин, *Сортировка и системы сортировки*. Москва: Мир, 1983.
- [2] Д. Э. Кнут, *Искусство программирования. Т.3. Сортировка и поиск*. Москва, Россия: Изд. дом «Вильямс», 2003.
- [3] *Ассоциативная обработка*. [Электронный ресурс]. Режим доступа: <http://www.ngpedia.ru/id97447pl.html>. Дата обращения: февр. 20, 2020.
- [4] Т. Б. Мартинюк, О. І. Черняк, Б. І. Круківський, і Мохамед Салем Нассер Мохамед, «Обчислювальна складність мережевої моделі сортування лінійного масиву чисел», *Інформаційні технології та комп'ютерна інженерія*, № 2, с. 64-71. 2019.
- [5] І. Г. Цмоць, *Інформаційні технології та спеціалізовані засоби обробки сигналів і зображень у реальному часі*. Львів, Україна: Видавництво УАД, 2005.
- [6] Т. Кохонен, *Ассоциативные запоминающие устройства*. Москва: Мир, 1982.
- [7] К. Дж. Тербер, *Архитектура высокопроизводительных вычислительных систем*. Москва: Гл. ред. физ.-мат. лит., 1985.
- [8] Я. И. Фет, *Параллельные процессоры для управляющих систем*. Москва: Энергоиздат, 1981.
- [9] В. В. Коваленко, і В. О. Гуменюк, «Аналіз найактуальніших серверних систем управління базами даних», *Вісник Національного університету "Львів. політехніка"*, № 603, с. 65-72, 2007.
- [10] А. В. Богданов, В. В. Корхов, В. В. Мареев, і Е. Н. Станкова, *Архитектуры и топологии многопроцессорных вычислительных систем*. Москва, Россия: ИНТУИТ.РУ, 2009.
- [11] У. Прэтт, *Цифровая обработка изображений*. Москва: Мир, 1982.
- [12] І. Г. Цмоць, і В. Я. Антонів, «Апаратні засоби сортування даних методом злиття в реальному часі», *Інформаційні системи та мережі*, № 814, с. 171-185, 2015.
- [13] І. Г. Цмоць, і В. Я. Антонів, «Алгоритми та паралельні структури сортування даних методом вставки», *Науковий вісник НЛТУ України*, вип. 26.1, с. 340-350, 2016.
- [14] Т. Б. Мартинюк, Н. О. Денисюк, і Б. І. Круківський, «Асоціативні процесори з паралельно-послідовною обробкою даних», *Інформаційні технології та комп'ютерна інженерія*, № 1, с. 27-36, 2019.
- [15] *Медианная фильтрация*. [Электронный ресурс]. Режим доступа: https://ru.bmstu.wiki/Медианная_фильтрация. Дата обращения: март, 20, 2020.
- [16] Т. Б. Мартинюк, і Б. І. Круківський, «Пристрій для ранжування чисел», *Патент України G06F 7/06. № 139604 МПК (2008)*, 12.08.2019.
- [17] Т. Б. Мартинюк, Б. І. Круківський, і А. І. Друзюк, «Сортувальник з індикацією рангів елементів масиву», на *Шостій міжнарод. наук.-техн. конф. «Фотоніка ОДС – 2018»*, ВНТУ, 2-4 жовтня 2018р., с. 30.
- [18] В. Б. Стещенко, *ПЛИС фирмы «ALTERA»: элементная база, система проектирования и языки описания*. Москва, Россия: Додека-XXI, 2002.
- [19] Микросхеми «Altera» [Электронный ресурс]. Режим доступа: <http://altera.ru/Mikroshemi.html>. Дата обращения: окт. 5, 2020.

Рекомендована кафедрою обчислювальної техніки ВНТУ

Стаття надійшла до редакції 5.11.2020

Мартинюк Тетяна Борисівна — д-р техн. наук, професор, професор кафедри обчислювальної техніки, e-mail: martyniuk.t.b@gmail.com ;

Круківський Богдан Ігорович — аспірант кафедри обчислювальної техніки, e-mail: smiletex11@gmail.com .
Вінницький національний технічний університет, Вінниця

T. B. Martyniuk¹
B. I. Krukivskiy¹

Model of Parallel Sorter for Associative Processor

¹Vinnitsia National Technical University

The process of sorting and selecting by key is a basic procedure in many search systems such as databases and Internet search systems. At the same time, modern computing tools require efficient methods and tools which are connected with associative information processing in the development of software and hardware. Therefore, there is a need for high-speed non-computational (associative) processing of large amounts of information, which requires appropriate organization and improvement of technical means of sorting. The well-known algorithms and means for number sorting make it possible to regulate the intensity of this process and increase its efficiency using parallel devices, but they require significant hardware costs. Therefore, the purpose of further research is to develop new and improve known methods of sorting with an orientation on reducing hardware costs and increasing the speed of this process. In this paper, there has been proposed a block diagram of a sorter as a computational part of an associative processor, which has a regular logical structure and parallel-serial connections between data processing units. This greatly simplifies the "placement" of the sorter in a programmable logic IS (FPGA) chip. In addition, the sorter functionally implements the multifunctionality of processing numerical data arrays due to the formation of the ranks of the input array of elements. This allows determining not only the extreme elements of the numeric array but also the element occupying the average value in the sorted array, which is a necessary condition for high-speed median filtering of images. In the proposed sorter, the sorting process uses fast increment/decrement operations on the counter arrays instead of the time-consuming operation of pairwise comparison in parallel for all arrays of elements with their subsequent re-commutation.

Keywords: sorter, median filtration, associative processor, ranking.

Martyniuk Tatiana B. — Dr. Sc. (Eng.), Professor, Professor of the Chair of Computer Engineering, e-mail: martyniuk.t.b@gmail.com ;

Krukivskiy Bohdan I. — Post-Graduate Student of the Chair of Computer Engineering, e-mail: smiletex11@gmail.com

Т. Б. Мартынюк¹
Б. И. Круковский¹

Модель параллельного сортировщика для ассоциативного процессора

¹Вінницький національний технічний університет

Процесс сортировки и выборки по ключу является основной процедурой во многих поисковых системах, таких как базы данных и поисковые системы в Интернете. В то же время современные вычислительные средства требуют эффективных методов и средств, связанных с ассоциативной обработкой информации при разработке программного и аппаратного обеспечения. Поэтому возникает необходимость в высокоскоростной невычислительной (ассоциативной) обработке больших объемов информации, что требует соответствующей организации и совершенствования технических средств сортировки. Известные алгоритмы и средства сортировки чисел позволяют регулировать интенсивность выполнения этого процесса и повысить его эффективность, используя параллельные устройства, однако они требуют значительных аппаратных затрат. Поэтому целью дальнейших исследований является разработка новых и усовершенствование известных методов сортировки с ориентацией на уменьшение аппаратных затрат и увеличение скорости этого процесса. В работе предложена структурная схема сортировщика как вычислительной части ассоциативного процессора, которая имеет регулярную логическую структуру и параллельно-последовательные связи между блоками обработки данных. Это значительно упрощает «размещение» сортировщика в микросхеме программируемой логической ИС (ПЛИС). Кроме того, функционально в сортировщике реализована многофункциональность обработки числовых массивов данных благодаря формированию рангов элементов входного массива. Это позволяет определить не только экстремальные элементы числового массива, но и элемент, занимающий среднее значение в отсортированном массиве, что является необходимым условием при скоростной медианной фильтрации изображений. В предлагаемом сортировщике в процессе сортировки используются быстрые операции инкремента/декремента на массивах счетчиков вместо затратной по времени операции попарного сравнения параллельно для всех элементов массива с последующей их перекмутацией.

Ключевые слова: сортировщик, медианная фильтрация, ассоциативный процессор, ранжирование.

Мартынюк Татьяна Борисовна — д-р техн. наук, профессор, профессор кафедры вычислительной техники, e-mail: martyniuk.t.b@gmail.com ;

Круковский Богдан Игоревич — аспирант кафедры вычислительной техники, e-mail: smiletex11@gmail.com