

МЕТОДИ ТА АПАРАТНІ ЗАСОБИ СОРТУВАННЯ МАСИВІВ ДВІЙКОВИХ ЧИСЕЛ

Грига В.М.¹⁾, Николайчук Я.М.²⁾

¹⁾Прикарпатський національний університет ім. В. Стефаника, к.т.н

²⁾Тернопільський національний економічний університет, д.т.н., професор

І. Постановка проблеми

Сортування є одною з типових проблем обробки даних і зазвичай розуміється, як задача розміщення елементів неупорядкованого набору значень масивів даних, в порядку монотонного зростання або спадання [1]. Забезпечити виконання такої операції у реальному часі є можливим на спеціалізованих засобах, архітектура яких повністю відображає структуру алгоритму сортування і орієнтована на реалізацію у вигляді програмованих інтегральних схем (ПЛІС) чи надвеликих інтегральних схем (НВІС). Використання спеціалізованих пристроїв дає можливість з високою ефективністю виконати операцію сортування заданого алгоритму по відношенню до універсальної комп'ютерної системи. Відомою є значна кількість алгоритмів послідовного та паралельного сортування масивів даних, які можна апаратно реалізувати [1-6].

ІІ. Мета роботи

Метою наукового дослідження є вдосконалення алгоритмів та апаратних засобів сортування масивів двійкових чисел.

ІІІ. Аналіз алгоритмів сортування масивів двійкових чисел

Нижче проаналізовано алгоритми сортування масивів двійкових чисел поданих у вигляді потокових графів та оцінено їхні обчислювальні характеристики.

На рис. 1 зображено граф алгоритму сортування чисел методом “бульбашки”.

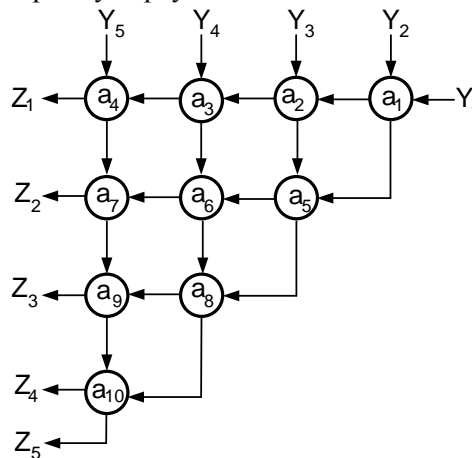


Рис. 1. Граф алгоритму сортування 5-ти значень методом “бульбашки”.

Структура даного алгоритму складається з однотипних операцій “порівняти і переставити”. Для N вхідних значень кількість операцій алгоритму сортування методом “бульбашки” складає $N(N-1)/2$. Ширина потокового графу алгоритму (найбільша кількість вершин у ярусі) складає $N/2$, а висота ПГА (кількість усіх ярусів) складає $(2N-3)$ операцій.

На рис. 2 зображено граф алгоритму сортування “парно-непарної” перестановки для 6-ти вхідних значень.

Для N вхідних значень кількість операцій алгоритму сортування методом “парно-непарної” перестановки складає $N(N-1)/2$. Ширина ПГА складає $N/2$, а висота ПГА N операцій “порівняти і переставити”.

На рис. 3 зображено граф алгоритму сортування методом Бетчера.

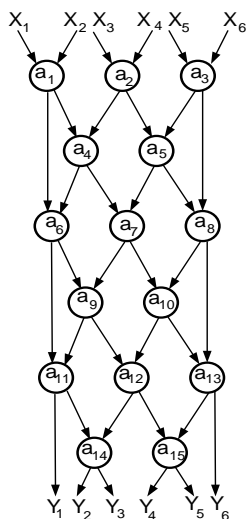


Рис. 2. Граф алгоритму сортування 6-ти значень методом “парно-непарної” перестановки.

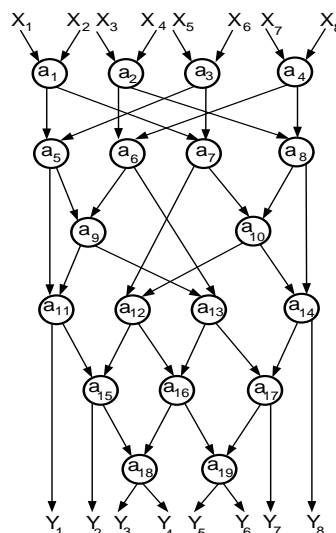


Рис. 3. Граф алгоритму сортування 8-ми значень методом Бетчера.

Для N вхідних значень кількість операцій порівняння і переставлення алгоритму сортування методом Бетчера складає $0,48N \ln^2 N$. Ширина ПГА сортування методом Бетчера складає $-N/2$, а висота ПГА - $(\frac{1}{2}[\log_2 N](\log_2 N + 1))$ операцій “порівняти і переставити”.

Апаратна реалізація описаних вище алгоритмів сортування передбачає повне відображення їхніх ПГ у структуру операційного пристрою, в якому вершинам графу (функціональним операторам) буде відповідати апаратний блок (операція) а дугам – лінії для передачі вхідних даних та проміжних і кінцевих результатів. Розглянуті алгоритми сортування масивів чисел складаються з базових елементів, які виконують однотипну операцію “порівняти і переставити”. Внутрішня структура базового елемента, який виконує операцію “порівняти і переставити” представлена на рис. 4.

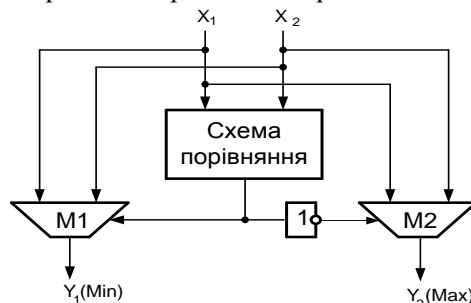


Рис. 4. Внутрішня структура базового елемента операції “порівняти і переставити”.

Щоб виконати порівняння двох чисел (X_1, X_2) , їх потрібно подати на схему порівняння, яка повинна визначити більше число $(X_1 > X_2)$, в результаті на виході схеми сформується ознака порівняння, пряме значення якої буде керувати мультиплексором M1, для видачі меншого числа (Y_1) а інверсне значення буде керувати мультиплексором M2 для видачі більшого числа (Y_2) .

На рис. 5 наведено комбінаційну схему порівняння двох чотирьохрозрядних двійкових чисел “на більше” [5], яка містить логічні елементи “НЕ”, “І”, “АБО” та “ВИКЛЮЧАЮЧЕ АБО”.

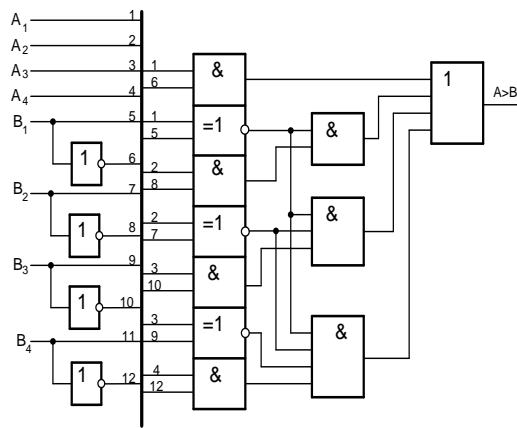


Рис. 5. Комбінаційна схема порівняння двох чотирьох- розрядних двійкових чисел “на більше”.

Враховуючи, що елемент “Виключаюче АБО” складається з 4-ох вентилів і має часову затримку 3 мікротакти, то апаратна складність представленої комбінаційної схеми буде рівна - $A_{СП} = 24$ (вентилі) а часова складність - $\tau_{СП} = 6\nu$ (мікротактів).

На рис. 6 подано спецпроцесор сортування (СПС) 4-ох розрядних двійкових чисел ($m=4$) для 8-ми вхідних значень ($n=8$) методом “бульбашки”.

Вхідні дані ($X_1, Y_2 - Y_8$) поступають на входи однотипних базових елементів, які виконують операцію “порівняти і переставити”, сумарна кількість яких при $n=8$ складає 28 елементів.

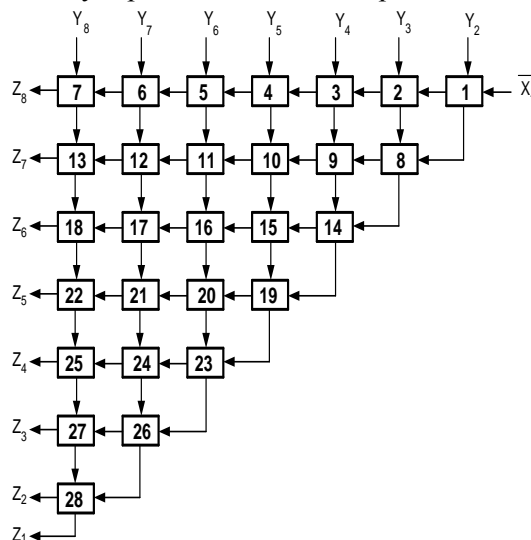


Рис. 6. Структура спецпроцесора сортування 8-ми чисел методом “бульбашки”.

На виходах ($Z_1 - Z_8$) СПС отримується відсортована послідовність масиву вхідних даних у порядку спадання.

Враховуючи, що однобітовий 2-входовий мультиплексор складається з 4-ох вентилів, то при $m=4$ його апаратна складність буде рівною - $A_{МП} = 28$ (вентилів) а часова складність - $\tau_{МП} = 3\nu$ (мікротакти).

Отже, сумарна апаратна складність даного СПС для $n=8$ і $m=4$ буде рівною:
 $A_{СПС} = n \times (n-1)/2 \times (A_{СП} + 2 \times A_{МП}) = 28 \times (24 + 56) =$
 $= 2240$ (вентилів).

Сумарна часова складність такого СПС буде рівна:
 $\tau_{СПС} = k \times (\tau_{СП} + \tau_{МП}) = 13 \times (6 + 3) = 117\nu$ (мікротактів), де k - кількість ярусів ПГА.

На рис.7 подано графік залежності кількості логічних елементів (вентилів) від кількості вхідних даних для структур відомого (класичного) та удосконаленого СПС виконання алгоритму сортування методом “бульбашки”.

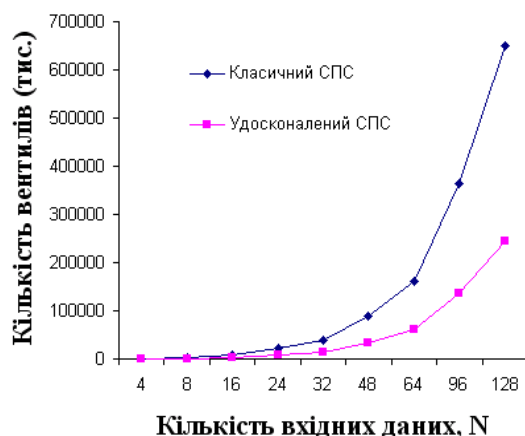


Рис.7. Графік залежності кількості вентилів від величини вхідних даних для структур СПС.

На графіку можна бачити, що кількість логічних елементів для удосконаленого СПС є в 2,5 рази меншої ніж для класичного СПС.

На рис.8 подано графік залежності часової складності вираженої в мікротактах від кількості вхідних даних для структур відомого (класичного) та удосконаленого СПС виконання алгоритму сортування методом “бульбашки”.

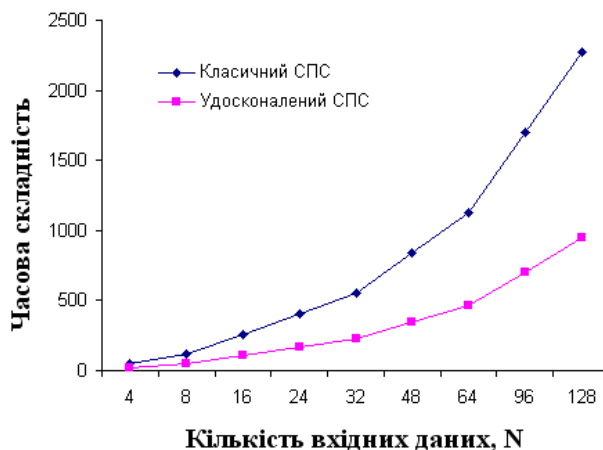


Рис.8. Графік залежності часової складності від величини вхідних даних для структур СПС.

На графіку можна бачити, що часова складність удосконаленого СПС потребує приблизно в 2,6 разів менше мікротактів, ніж класичного СПС.

Висновок

У результаті проведеного аналізу методів та алгоритмів сортування масивів двійкових чисел запропонована удосконалена структура спецпроцесора з підвищеною швидкістю та меншою апаратною складністю.

Список використаних джерел

1. D. Knuth The art of computer programming. V.3: Transl. from eng. – М.: “Peace”, 1978. -841 p.
2. Thomas Braunl. Parallel Programming: Transl. from german – К.: Vyscha Shkola., 1997.– 358 p
3. S.Y. Kung. VLSI array processors. Trans. from English - М.: “Peace”, 1991. – 672 pp.
4. Valeriy Zadiraka, Yaroslav Nykolaichuk. Methods of effective protection of information flows.- Ternopil: Terno-graf, 2014. - 308 p.
5. A. Melnyk. Memory with ordered access: monograph – Lviv: Press Lviv Polytechnic, 2014. – 296 p.
6. R. Dunets, V. Gryga Spatio-temporal synthesis of transformation matrix of reverse fast cosine transformation // Modern problems of radio engineering, telecommunications and computer science . Proceedings of XIIIth International Conference. CADSM’2015. – Lviv-Poljana, Ukraine, 2015. – P.45-49.