

УДК 004.33

Н.М. Ліщина

Луцький національний технічний університет

МЕТОДИ ПОБУДОВИ ПАМ'ЯТІ З ВПОРЯДКОВАНИМ ДОСТУПОМ

Ліщина Н.М. Методи побудови пам'яті з впорядкованим доступом. Розглядаються методи побудови пам'яті з впорядкованим доступом. Описані підходи до побудови пам'яті з впорядкованим доступом, виділено її особливості.

Ключові слова: комутуюча мережа, сортувальна мережа, структура ПВД.

Лищина Н.М. Методы построения памяти с упорядоченным доступом. Рассматриваются методы построения памяти с упорядоченным доступом. Описаны подходы к построению памяти с упорядоченным доступом, выделены ее особенности.

Ключевые слова: коммутирующая сеть, сортировочная сеть, структура ПВД.

Lishchyna Natalia. Methods of memory orderly access. Methods of constructing an ordered memory access. We describe approaches to the construction of an ordered memory access, highlighted its features.

Keywords: commutating network, sorting network, structure of MOA.

Постановка проблеми. Як і пам'ять з послідовним доступом, пам'ять з впорядкованим доступом орієнтована на роботу з масивами даних. В цій пам'яті забезпечується доступ до даних у програмно встановленому порядку, тобто індекс, який поступає в пам'ять разом з даним, або під час його зчитування, вказує місце даного у вихідному масиві.

Пам'ять з впорядкованим доступом може бути реалізована з використанням трьох підходів: на базі пам'яті з довільним доступом, на базі пам'яті з асоціативним доступом і на базі пристрою сортування даних (ПСД) [1].

При використанні першого методу можливі два варіанти реалізації ПВД. Згідно з першим варіантом дані в режимі запису розміщуються в комірках пам'яті з довільним доступом, адреси яких відповідають потрібному порядку зчитування, а в режимі зчитування адреси формуються в порядку зростання на одиницю. Згідно з другим варіантом дані в режимі запису розміщуються в послідовних комірках пам'яті з довільним доступом, а при зчитуванні потрібний порядок задається адресами. До недоліків ПВД на базі пам'яті з довільним доступом потрібно віднести велику інерцію, оскільки розпочати зчитування даних можна лише після завершення прийому всього масиву даних, а також потребу використання двох паралельних блоків пам'яті з довільним доступом для побудови пам'яті з впорядкованим доступом з розділеними входом і виходом.

При використанні другого методу виходять з того, що функції ПВД і процесора сортування даних (ПСД) досить подібні: ПВД виконує впорядкування елементів вихідного масиву даних відповідно до величини індексів, які їх супроводжують, а ПСД сортує числа в порядку їх зростання чи спадання. Тому логічно закріпивши номери даних M в масиві за цими даними провести їх впорядкування, і, тим самим, виконати функцію ПВД на ПСД. Такий підхід є ближчим за функціональними ознаками до принципів роботи ПВД і дозволяє використати всі її переваги. Зокрема, відпадає необхідність накопичувати весь масив даних, які підлягають зчитуванню та подальшому опрацюванню. Достатньо накопичити стільки даних, щоб можна було почати їх опрацювання одночасно з прийомом наступних. Наприклад, в процесорах швидкого перетворення Фур'є достатньо накопичити половину вибірки, щоб розпочати опрацювання даних.

Метою дослідження є розгляд методів побудови пам'яті з впорядкованим доступом, опис і аналіз підходів до побудови пам'яті з впорядкованим доступом.

Основна частина

Метод побудови ПВД на основі ПДД

При використанні першого методу використовується структура ПВД, варіант якої для випадку $L = N = 1$ показаний на рис.1. [1].

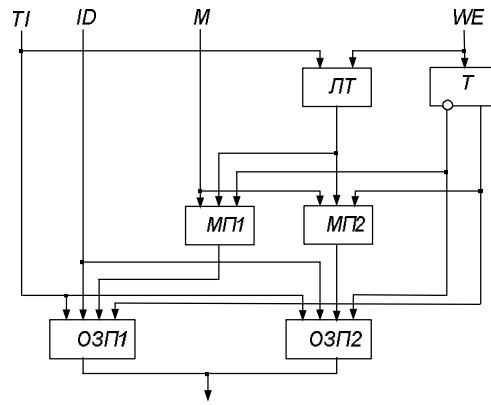


Рис. 1. ПВД на основі ОЗП.

До складу ПВД входять два ОЗП і відповідне обрамлення. Для пояснення роботи прийемо, що на даний момент часу ОЗП1 працює в режимі зчитування, а ОЗП2 - в режимі запису. З приходом нового масиву по сигналу дозволу запису QE відбудеться початкова установка лічильника тактів ЛТ, а тригер Т перемикається в протилежний попередньому стан, сигнали з прямого і інверсного виходів якого переключать ОЗП1 в режим запису, а ОЗП2 в режим зчитування. По першому тактовому сигналу з входу подання тактових імпульсів TI в пам'ять записується дане нового масиву в комірку, номер якої вказує індекс M, який поступає через мультиплексор МП1, керований інверсним виходом тригера Т. Надалі при кожному зверненні в ОЗП1 записується дане нового масиву. Одночасно з ОЗП2 зчитується раніше накопичений масив даних за послідовними адресами, які формуються в лічильнику тактів ЛТ та проходять на адресний вхід пам'яті через мультиплексор МП2, яким керує сигнал з прямого виходу тригера Т. Після запису нового масиву в ОЗП2 і зчитування раніше прийнятого масиву з ОЗП1 в ПВД поступає наступний масив даних, що супроводжується сигналом QE. Цей сигнал скидає лічильник ЛТ в нульовий стан, а також перемикає тригер Т, сигнали з виходів якого встановлюють ОЗП1 в режим зчитування, а ОЗП2 в режим запису, а також через мультиплексори МП1 і МП2 під'єднують до адресних входів ОЗП1 вихід ЛТ, а ОЗП2 - вхід поступлення індексів M. Таким чином змінюються режими роботи ОЗП1 і ОЗП2 і надалі їх робота відбувається аналогічно описаній. Як видно, за наведеною схемою ПВД здійснює одночасний прийом нового масиву і зчитування раніше записаного масиву даних з використанням одного адресного входу – входу поступлення індексів M, причому адреса вказує, яким по порядку в даному масиві зчитується це число.

У розглянутій ПВД, якщо послідовно записувані масиви мають різну величину, будуть перерви в прийомі і видачі даних. Якщо подальший масив менший попереднього, повинна бути перерва в прийомі інформації, а якщо більший – в видачі інформації. Введення в ПВД суматора для зсуву адрес дозволяє забезпечити накопичення в одній ПВД декількох масивів даних. У цьому плані цікавим виглядає варіант коли обидва ОЗП розбивають на сегменти рівної ємності. В цьому випадку в ПВД можна проводити впорядкування і зберігання M масивів, де M - кількість сегментів, причому розмір масиву не повинен перевищувати ємність сегменту пам'яті. При цьому в ПВД необхідно ввести лічильник масивів, а розрядність зовнішньої адреси зменшиться на $\log_2 M$ розрядів. Як основні недоліки ПВД на базі ОЗП варто відзначити наявність перерви в прийомі і видачі даних при відмінній величині масивів, а також велику інерцію, оскільки розпочати зчитування з ОЗП1 або ОЗП2 можна тільки після завершення прийому подальшої інформації.

Метод побудови ПВД на основі комбінаційних сортувальних мереж

Якщо подати вхідні дані у вигляді матриці

$$\begin{pmatrix} ID_{0,0} & ID_{0,1} & \dots & ID_{0,l-1} \\ ID_{1,0} & ID_{1,1} & \dots & ID_{1,l-1} \\ \dots & \dots & \dots & \dots \\ ID_{k-1,0} & ID_{k-1,1} & \dots & ID_{k-1,l-1} \end{pmatrix},$$

вихідні дані подати у вигляді матриці

$$\begin{vmatrix} OD_{0,0} & OD_{0,1} & \dots & OD_{0,n-1} \\ OD_{1,0} & OD_{1,1} & \dots & OD_{1,n-1} \\ \dots & \dots & \dots & \dots \\ OD_{m-1,0} & OD_{m-1,1} & \dots & OD_{m-1,n-1} \end{vmatrix},$$

а індекси вхідних даних подати у вигляді матриці

$$\begin{vmatrix} SID_{0,0} & SID_{0,1} & \dots & SID_{0,l-1} \\ SID_{1,0} & SID_{1,1} & \dots & SID_{1,l-1} \\ \dots & \dots & \dots & \dots \\ SID_{k-1,0} & SID_{k-1,1} & \dots & SID_{k-1,l-1} \end{vmatrix}.$$

то задача впорядкованими масиву вхідних даних за величиною їх індексів у масив вихідних даних виглядає наступним чином:

$$MOD = MID \cdot Q \cdot MID,$$

де MOD, MID та MID – матриці вихідних, вхідних даних та їх індексів, Q – оператор впорядкування.

Нехай

$$\begin{vmatrix} SV_{0,0} & SV_{0,1} & \dots & SV_{0,r-1} \\ SV_{1,0} & SV_{1,1} & \dots & SV_{1,r-1} \\ \dots & \dots & \dots & \dots \\ SV_{p-1,0} & SV_{p-1,1} & \dots & SV_{p-1,r-1} \end{vmatrix}$$

- матриця впорядкованих індексів вхідних даних така, що $SV_{0,0} < SV_{0,1} < SV_{0,2} \dots < SV_{p-1,r-1}$, де $p=k, r=1$.

Тоді елементи матриці вихідних даних визначаються з виразу [12]

$$OD_{i,j} \{i=0,1,\dots,m-1, j=0,1,\dots,n-1\} = ID_{t,f} \{t=0,1,\dots,k-1, f=0,1,\dots,l-1\} \text{ при } SV_{i,j} = ID_{t,f}$$

де i, j – номер індекси у матриці впорядкованих індексів вхідних даних

$$\begin{vmatrix} SV_{0,0} & SV_{0,1} & \dots & SV_{0,r-1} \\ SV_{1,0} & SV_{1,1} & \dots & SV_{1,r-1} \\ \dots & \dots & \dots & \dots \\ SV_{p-1,0} & SV_{p-1,1} & \dots & SV_{p-1,r-1} \end{vmatrix},$$

t, f – номер індекси у матриці індексів вхідних даних

$$\begin{vmatrix} SID_{0,0} & SID_{0,1} & \dots & SID_{0,l-1} \\ SID_{1,0} & SID_{1,1} & \dots & SID_{1,l-1} \\ \dots & \dots & \dots & \dots \\ SID_{k-1,0} & SID_{k-1,1} & \dots & SID_{k-1,l-1} \end{vmatrix}.$$

Таким чином, виходячи з формалізованого опису, метод впорядкування матриці даних за їх індексами та відповідний метод побудови ПВД полягає в наступному [2]:

1. Потрібно впорядкувати матрицю індексів вхідних даних

$$\begin{vmatrix} SID_{0,0} & SID_{0,1} & \dots & SID_{0,l-1} \\ SID_{1,0} & SID_{1,1} & \dots & SID_{1,l-1} \\ \dots & \dots & \dots & \dots \\ SID_{k-1,0} & SID_{k-1,1} & \dots & SID_{k-1,l-1} \end{vmatrix}$$

та отримати матрицю впорядкованих індексів вхідних даних

$$\begin{vmatrix} SV_{0,0} & SV_{0,1} & \dots & SV_{0,r-1} \\ SV_{1,0} & SV_{1,1} & \dots & SV_{1,r-1} \\ \dots & \dots & \dots & \dots \\ SV_{p-1,0} & SV_{p-1,1} & \dots & SV_{p-1,r-1} \end{vmatrix}$$

таку, що

$$SV_{0,0} < SV_{0,1} < SV_{0,2} \dots < SV_{p-1,r-1}, \text{ де } p=k, r=1.$$

Зауваження 1: Випадку, коли $ID_{ij} = ID_{t,s}$ де $i,t = 0,1, \dots, p-1$; $j,s = 0,1, \dots, r-1$ бути не може за визначенням задачі.

2. Для виконання впорядкування індексів використати елемент, який має назву «сортувальна мережа», (СМ).

3. Для виконання впорядкування даних відповідно до їх індексів використати елемент, який має назву «комутуюча мережа», (КМ). Задачею КМ є переміщення даних до відповідних виходів за результатами порівняння їх індексів в СМ.

4. Для запам'ятовування матриці даних

$$\begin{vmatrix} ID_{0,0} & ID_{0,1} & \dots & ID_{0,l-1} \\ ID_{1,0} & ID_{1,1} & \dots & ID_{1,l-1} \\ \dots & \dots & \dots & \dots \\ ID_{k-1,0} & ID_{k-1,1} & \dots & ID_{k-1,l-1} \end{vmatrix}$$

віділити kl комірок пам'яті, в ролі яких використати регістри.

5. Записувати в режимі запису до регістрів ПВД дані, та здійснити їх впорядкування відповідно до величини їх індексів.

6. Зчитувати в режимі зчитування з регістрів ПВД дані, які в них зберігалися, на виходи ПВД.

7. Дані на входи ПВД можуть поступати паралельно або послідовно. Необхідно забезпечити на входи ПВД відповідний інтерфейс.

2. Дані на виходи ПВД можуть поступати паралельно або послідовно. Необхідно забезпечити на виходи ПВД відповідний інтерфейс.

У загальному випадку, коли в ПВД записується матриця $k \times l$ даних і з її виходу зчитується матриця $m \times n$ даних, структура ПВД має вигляд, показаний на рис.2.

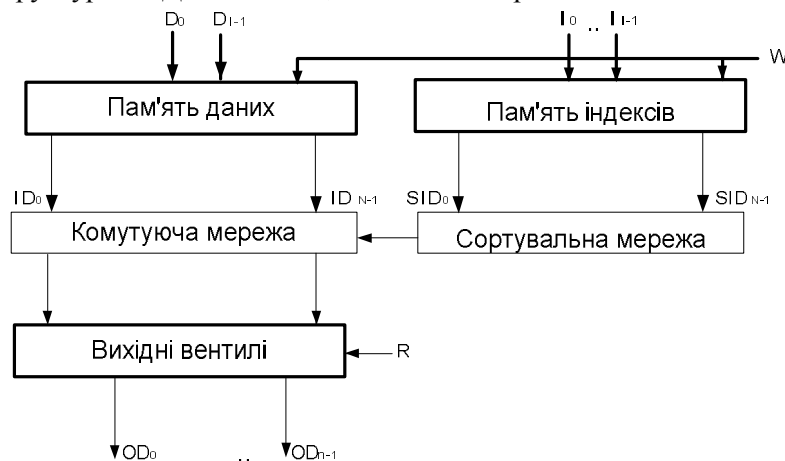


Рис.2. Структура ПВД на основі комбінаційної сортувальної мережі.

ПВД цього типу містить пам'ять для зберігання даних та індексів відповідно, до яких дані і індекси записують групами, а після того проводиться впорядкування даних за величиною їх індексів використовуючи сортувальну СМ та комутуючу КМ мережі.

Вхідні дані та їх індекси записуються до пам'яті сигналом запису W в відповідні комірки. В КМ здійснюється впорядкування даних з пам'яті даних відповідно до значень їх індексів з пам'яті індексів за правилом, заданим величинами індексів, які сортуються в СМ. З ПВД цього типу одночасно зчитують n чисел, для чого на її виході включено вихідні вентиля. З виходів цих вентилів вихідні дані при поступленні сигналу зчитування R будуть подані на відповідну вихідну шину.

Детальніше внутрішня структура ПВД цього типу показана на рис.3. [3].

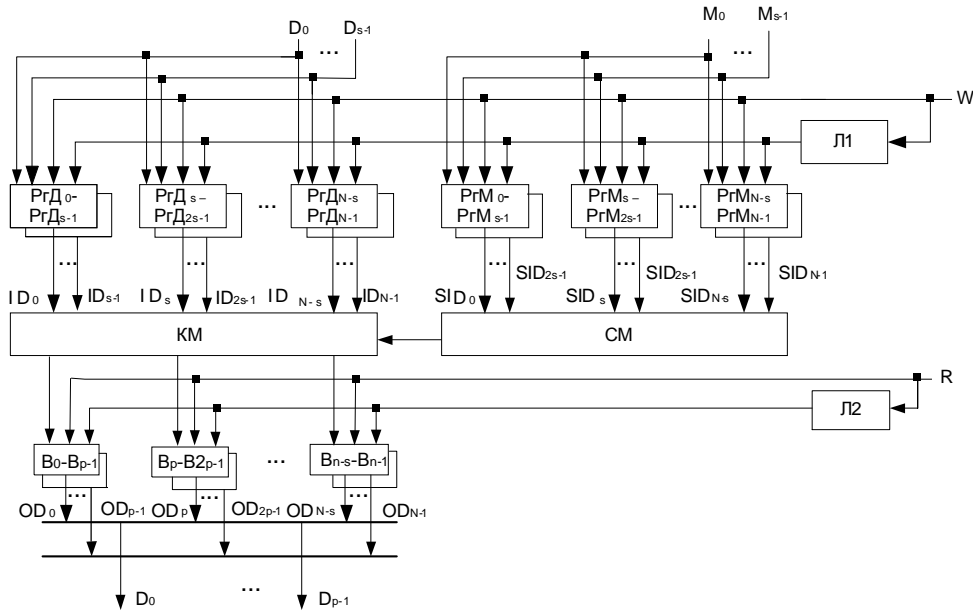


Рис.3. Внутрішня структура ПВД.

ПВД цього типу містить згруповані по k регістрів вхідні регістри даних PrD_0-PrD_{k-1} , PrD_k-PrD_{2k-1} , ... $PrD_{l-k}-PrD_{l-1}$, та вхідні регістри індексів PrM_0-PrM_{k-1} , PrM_k-PrM_{2k-1} , ... $PrM_{l-k}-PrM_{l-1}$ для зберігання даних та індексів відповідно, до яких дані і індекси записують групами, а після того проводиться впорядкування даних за величиною їх індексів використовуючи сортувальну CM та комутуючу KM мережі.

Сигналом запису W вхідні дані та їх індекси запишуться в відповідні регістри, причому місце запису даних та індексів вказується сигналом з виходу лічильника L , який формує сигнали дозволу запису. Цей лічильник спочатку перебуває в стані «0» та дозволяє запис першої групи даних і їх індексів відповідно до регістрів PrD_0-PrD_{k-1} та PrM_0-PrM_{k-1} , після поступлення першого сигналу запису W переходить в стан «1» та дозволяє запис другої групи даних і їх індексів відповідно до регістрів PrD_k-PrD_{2k-1} та PrM_k-PrM_{2k-1} , після поступлення другого сигналу запису W переходить в стан «2» та дозволяє запис третьої групи даних і їх індексів відповідно до регістрів $PrD_{2k}-PrD_{3k-1}$ та $PrM_{2k}-PrM_{3k-1}$ і так до запису l/k -ї групи даних та їх індексів до регістрів $PrD_{l-k}-PrD_{l-1}$ та $PrM_{l-k}-PrM_{l-1}$. Після подачі сигналу зчитування R на KM буде здійснено впорядкування даних з регістрів PrD_0-PrD_{k-1} , PrD_k-PrD_{2k-1} , ... $PrD_{l-k}-PrD_{l-1}$ відповідно до значень їх індексів з регістрів PrM_0-PrM_{k-1} , PrM_k-PrM_{2k-1} , ... $PrM_{l-k}-PrM_{l-1}$. З ПВД цього типу одночасно зчитують m чисел, для чого на її виході включено вентиля, які поділено на групи по m вентилів B_0-B_{m-1} , B_m-B_{2m-1} , ... $B_{n-m}-B_{n-1}$ та виходи кожного i -го вентиля, де $i=(j) \bmod m$, $j=0,1,\dots,n$ об'єднані спільною шиною. Лічильник L підраховує сигнали зчитування R та вказує з якого номера вентиля на якому номері сигналу зчитування R дозволяється зчитування. З виходів цих вентилів вихідні дані при поступленні сигналу зчитування R будуть подані на відповідну вихідну шину.

Як видно з рис.5, ця пам'ять має таку ж кількість портів поступлення індексів, як і кількість портів вхідних даних, що робить проблематичною її реалізацію в вигляді окремої мікросхеми через значну кількість виводів. Крім того, в цій пам'яті необхідно забезпечувати зберігання індексів, що суттєво збільшує її апаратну складність. Для зберігання індексів, кількість яких рівна кількості даних, потрібно виділяти таку ж кількість комірок пам'яті, як і для зберігання даних. Названі два недоліки суттєво знижують ефективність існуючої ПВД.

Метод побудови ПВД на основі ПАД

Пам'ять з асоціативним доступом має ширші функціональні можливості у порівнянні з ПВД. Щоб вона виконувала функції ПВД, потрібно до ПАД разом з даними записувати і їх індекси. Вони записуються в довільні вільні регістри пам'яті. Дані вибираються з такої пам'яті на основі збігу їх індексів з заданими. Для цього індекси даних з усіх регістрів пам'яті поступають на схему порівняння (рис.4), де порівнюються з заданим індексом із регістра ознаки, і на вихід пам'яті поступають дані, індекси яких збігаються з заданим. Для цього до регістра ознаки потрібно по

черзі записувати індекси $ИД_{i0}, ИД_{i1}, \dots, ИД_{i(n-1)}$, де i – номер стрічки вихідної матриці ($i = 0, 1, \dots, m-1$). [2]

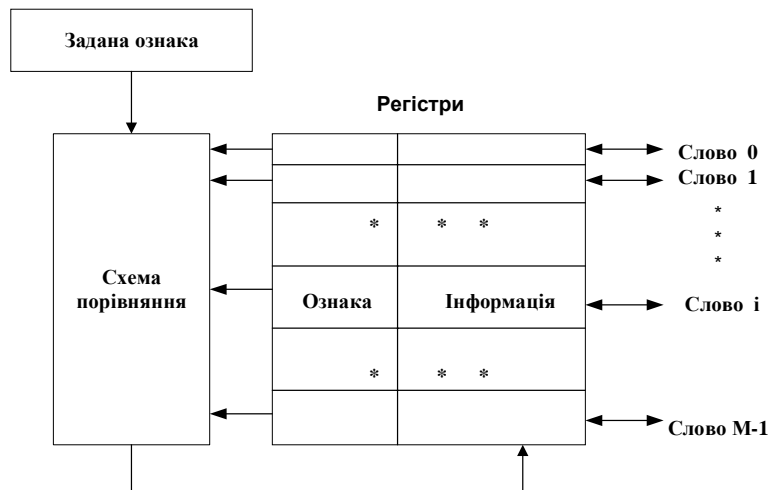


Рис.4. Структура пам'яті з асоціативним доступом.

Ця пам'ять є послідовною та відповідно повільнішою у порівнянні з вище розглянутою ПВД на основі комбінаційних сортувальних мереж.

Порівняно з пам'яттю з довільним, асоціативним та послідовним доступом, пам'ять з впорядкованим доступом має ряд суттєвих переваг, основні з яких наступні :

- В цій пам'яті наявне тільки однократне звернення за даним, оскільки код номера даного присутній лише при його запису або зчитуванні, тоді як в пам'яті з довільним та з асоціативним доступом звернення є двократним. Тим самим в два рази зменшується кількість розрядів в адресній частині команд і, відповідно, спрощується керування та зменшуються затрати обладнання.
- У варіанті пам'яті з впорядкованим доступом з розділеними входом і виходом забезпечується одночасний запис та зчитування даних, оскільки дані записуються в пам'ять без прив'язки до конкретної її комірки, а зчитуються в заданому порядку, який не пов'язаний з вхідними даними. Тим самим в два рази зростає швидкодія в порівнянні з пам'яттю з довільним та з асоціативним доступом.
- Немає прив'язки даних до комірок пам'яті, тому відпадає потреба в складних та повільних дешифраторах адрес. Їх функції можуть бути дезінтегровані, що дозволяє суттєво зменшити їх затримку та прискорити роботу пам'яті;
- Можуть ефективно вирішуватись питання розпаралелення роботи пам'яті з впорядкованим доступом, оскільки в ній немає прив'язки даних до комірок пам'яті.

Висновки. В даній статті розглянуто основи побудови пам'яті з впорядкованим доступом та оцінені вигоди від її застосування в різних архітектурах комп'ютерів. Показано, що критеріям ефективності паралельної пам'яті найкраще відповідає ПВД, оскільки вона є багатопортовою, надає можливість одночасного безконфліктного доступу до даних з багатьох портів, забезпечує одночасний запис даних, які поступають на її входи, та зчитування на виходи раніше записаних даних, забезпечує можливості виконання операцій реорганізації масивів та впорядкування даних в масивах, однократність звернення, виключення потреби звернень до комірок пам'яті, виключення потреби зберігання інформації про місце знаходження даного в пам'яті.

Список використаних джерел.

1. Мельник А.О. Архітектура комп'ютера. Наукове видання. – Луцьк: Волинська обл. друк., 2008. – 470 с.
2. Кун С. Матричные процессоры на СБИС: Пер. с англ. - М.: Мир, 1991. - 672с.
3. К. Е. Batcher, "Sorting networks and their applications," Proc. AFIPS Spring Joint Computer Conference, vol. 32, pp. 307-314, April 1968.