УДК 004.383

В. Глухов
Національний університет “Львівська політехніка”,
кафедра електронних обчислювальних машин

БАГАТОРІВНЕВІ СТРУКТУРИ І ПРОБЛЕМА ОДНОЧАСНОГО ПРОЕКТУВАННЯ АПАРАТНИХ І ПРОГРАМНИХ ЗАСОБІВ

Запропоновано і обґрунтовано метод оцінювання ефективності вирішення проблеми одночасного розроблення апаратного і програмного забезпечення комп’ютерних систем шляхом оцінювання апаратної складності протокольних процесорів і спецпроцесорів, що входять до складу системи. Для багаторівневих комп’ютерних систем ефективність розв’язання проблеми експоненціально збільшується із збільшенням кількості рівнів. Найкращою за цим критерієм є система з найбільшою кількістю рівнів. Рекомендується під час проектування багаторівневих систем обирати системи з максимально допустимою за іншими критеріями кількістю рівнів.

Ключові слова: одночасне розроблення апаратного і програмного забезпечення, багаторівнева комп’ютерна система, ієрархічна система, апаратна складність, еталонна модель взаємозв’язку відкритих систем.

The method of computer systems Hardware-Software Co-design problem decision efficiency evaluation is described. The method is grounded in evaluation of hardware complexity of protocol and dedicated processors which are used in the system. For multilevel computer systems efficiency of problem decision is exponentially increased with the increase of levels amount. The best system according to this criterion is one with most of levels. It is recommended during planning of the multilevel systems to choose one with the maximally possible amount of levels.

Key words: Hardware-Software Co-design, multilevel computer systems, hardware complexity, open system interconnection model.

Вступ

Розроблення комп’ютерної системи полягає в проектуванні її апаратної і програмної частин. При цьому програмування може починатися тільки після закінчення проектування апаратної частини та її випробовування. Для зменшення часу проектування намагаються почати розробляти програмне забезпечення до закінчення проектування апаратного. Це призводить до постійних вимушених корекцій і переробок вже розробленого програмного забезпечення внаслідок змін, що вносяться до апаратури під час проектування. У цьому і полягає проблема одночасного розроблення апаратного і програмного забезпечення – *Hardware-Software Codesign* (*HSC)*.

Ця проблема є наслідком необхідності розроблення структурних автоматів і переходу від програмної до апаратної реалізації алгоритмів. При цьому у процесі розроблення структури автомата для відомого алгоритму (у процесі переходу від абстрактного автомата до структурного) може виникати і виникає необхідність зміни структурного алгоритму, тобто програмного забезпечення. Відомі рішення цієї проблеми для криптографічного захисту інформації на основі однокристальних мікроЕОМ. Іншими апаратними платформами для вирішення проблеми є мікропроцесори, процесори цифрової обробки сигналів, ПЛІС, інтегральні мікросхеми спеціального призначення (*ASIC*). Але універсальні рішення проблеми на цих платформах невідомі. Також невідомі кількісні показники, які давали б змогу оцінювати проектні рішення за ступенем розв’язання цієї проблеми.

Багаторівневі структури широко використовуються в КС для збільшення продуктивності та кількості задач, що одночасно розв'язуються. Відомі дворівневі та багаторівневі системи захисту інформації, багатопроцесорні системи, багаторівнева організація комп'ютера, багаторівнева організація пам'яті комп'ютера, ієрархічні рівні багаторівневої структури, неоднорідні системи на кристалі (*SoC)*, багаторівневі структури систем автоматизації. Для побудови відкритих мереж використовується багаторівневу модель взаємозв'язку відкритих систем (*OSI)*.

Відомі методи оцінювання апаратних витрат на реалізацію протокольних процесорів і спецпроцесорів кожного рівня. Кожний протокольний процесор є універсальним, тому основна частина роботи з проектування програмного забезпечення припадає на розроблення програмного забезпечення протокольних процесорів. Основна складність розробки апаратного забезпечення припадає на проектування спецпроцесорів. За оцінку ефективності рішення проблеми одночасного проектування апаратного і програмного забезпечення пропонується взяти співвідношення апаратних витрат на реалізацію універсальних протокольних процесорів і спецпроцесорів. Що більша величина цього співвідношення, то краще розв'язана проблема.

1. Аналіз основних досліджень та публікацій

Проблема одночасного розроблення апаратного і програмного забезпечення є наслідком необхідності розроблення структурних автоматів і переходу від програмної до апаратної реалізації алгоритмів [1]. При цьому у процесі розроблення структури автомата для відомого алгоритму (у процесі переходу від абстрактного автомата до структурного) може виникати і виникає необхідність зміни структурного алгоритму, тобто програмного забезпечення. Відомі рішення цієї проблеми для криптографічного захисту інформації на основі однокристальних мікроЕОМ [2]. Іншими апаратними платформами для вирішення проблеми є мікропроцесори, процесори цифрової обробки сигналів, ПЛІС, інтегральні мікросхеми спеціального призначення (*ASIC*) [3].

Багаторівневі структури широко використовуються в КС для збільшення продуктивності та кількості задач, що одночасно розв'язуються. Відомі дворівневі [4] та багаторівневі системи захисту інформації [5], багатопроцесорні системи, багаторівнева організація комп'ютера [6], багаторівнева організація пам'яті комп'ютера [7], ієрархічні рівні багаторівневої структури [8], неоднорідні SoC [9], багаторівневі структури систем автоматизації [10]. Для побудови відкритих мереж використовують багаторівневу модель взаємозв'язку відкритих систем *OSI*. Найбільш структуровано принцип побудови багаторівневих систем викладено у [11–13], де наведено базову семирівневу еталонну модель *OSI*. Основи обміну даними між сумісними рівнями ілюструє рис. 2, де фігурують протокольний блок даних та сервісний блок даних (звідси випливає необхідність в універсальних протокольних процесорах і сервісних спецпроцесорах). Рис. 1 ілюструє більш універсальний характер моделі – вказує умовні номери рівнів N+1, N, N-1, … .

Багаторівнева модель використовується не тільки для організації каналів передавання інформації, але й для організації зберігання інформації (у флеш-пам'яті [14]).

Протокольні процесори можуть бути реалізовані в складі ПЛІС [15, 16] або ззовні ПЛІС. Вони можуть бути RISC-процесорами та CISC-процесорами, 8-, 16-, 32-розрядними або можуть мати іншу розрядність [16]. Сучасні ПЛІС дають змогу реалізовувати універсальні процесори різного типу і продуктивності: *hard*-процесори (апаратно-реалізовані і розміщені на кристалі ПЛІС у процесі її виробництва нереконфігуровані процесори) і *soft*-процесори [16] (реконфігуровані, розроблені користувачем ядра [4]).

Відомі методи оцінювання апаратних витрат на реалізацію протокольних процесорів і спецпроцесорів кожного рівня [17–19].

2. Постановка проблеми

Розроблення комп'ютерної системи полягає в проектуванні її апаратної і програмної частин. При цьому програмування може починатися тільки після закінчення проектування апаратної частини і її випробовування. Для зменшення часу проектування намагаються почати розроблення програмного забезпечення до закінчення проектування апаратного. Це призводить до постійних вимушених корекцій і переробок вже розробленого програмного забезпечення внаслідок змін, що вносяться до апаратури під

час проектування. У цьому і полягає проблема одночасного розроблення апаратного і програмного забезпечення – *Hardware-Software Codesign* (*HSC)*. Відомі рішення цієї проблеми для КЗІ на основі однокристальних мікроЕОМ. Іншими апаратними платформами для рішення проблеми є мікропроцесори, процесори цифрової обробки сигналів, ПЛІС, інтегральні мікросхеми спеціального призначення (*ASIC*). Але універсальні рішення проблеми на цих платформах невідомі. Також невідомі кількісні показники, які дають змогу оцінювати проектні рішення за ступенем розв'язання цієї проблеми і обирати серед них найкращий. Визначення показників ефективності рішення проблеми та визначення найкращої системи за обраними критеріями є важливою і актуальною задачею.

3. Цілі статті

Метою роботи є розроблення методів кількісного оцінювання варіантів вирішення проблеми одночасного розроблення апаратного і програмного забезпечення комп'ютерних систем, визначення критеріїв порівняння комп'ютерних систем за ефективністю вирішення проблеми і визначення найкращої системи за обраним критерієм.

4. Еталонна модель взаємозв'язку відкритих систем

Найбільш структуровано принцип побудови багаторівневих систем викладено у [11], де наведена базова семирівнева еталонна модель взаємозв'язку відкритих систем (рис. 1, 2). Основи обміну даними між сумісними рівнями ілюструє рис. 2, де фігурують протокольний та сервісний блоки даних (звідси випливає необхідність мати у складі *N*-рівня протокольний та спеціалізований процесор). Рис. 1 ілюструє універсальний характер моделі, оскільки не фіксує назви рівнів, а вказує їхні умовні номери (…, *N*+1, *N*, *N*-1, …) .

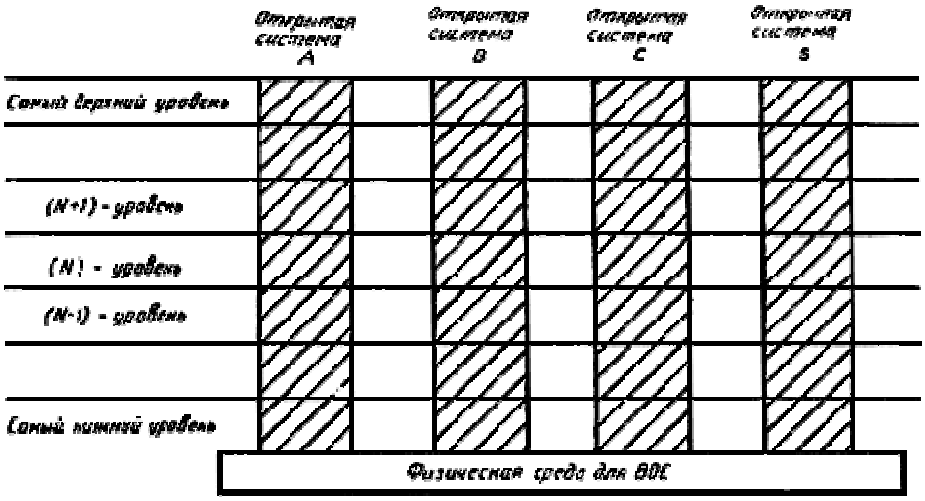

Рис. 1 Еталонна модель взаємозв'язку відкритих систем

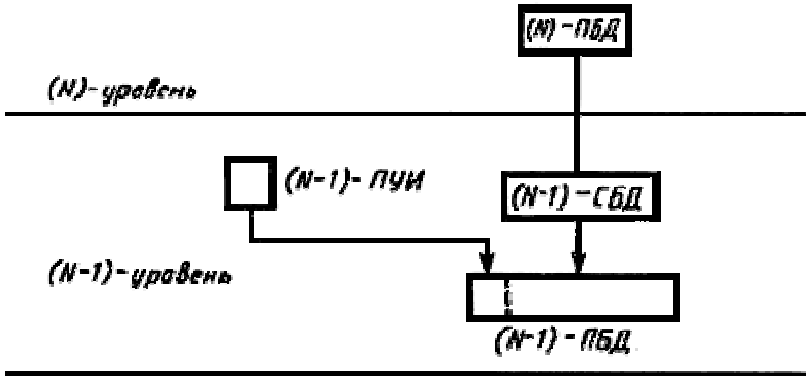

Рис. 2 Приклад перетворення блоків даних у суміжних рівнях

ПУИ – протокольна керуюча інформація, ПБД – протокольний блок даних, СБД – сервісний блок даних.

5. Структура багаторівневої системи

Комп'ютерну систему відповідно до базової семирівневої еталонної моделі взаємозв'язку відкритих систем можна подати у вигляді функціонального каналу дворівневої (протокольний процесор і спецпроцесор) системи [20].

Для побудови кожного рівня використовується центральний протокольний процесор і спецпроцесор (або декілька спецпроцесорів), при цьому кожний із спецпроцесорів, своєю чергою, складається з протокольного процесора і спецроцесора (спецпроцесорів) нижчого рівня. Спецпроцесори можуть бути виконані у вигляді ядер HBIC [4].

Представлення комп'ютерної системи як каналу підказує використання еталонної моделі взаємозв'язку відкритих систем [11] для розподілення функцій між елементами системи. Потік інформації під час її оброблення у системі послідовно йде з найвищого рівня на найнижчий, а потім знову підіймається на найвищий (наприклад, під час роботи системи криптографічного захисту інформації (зашифрування) відкрита інформація опускається з верхнього рівня на найнижчій, а потім зашифрована інформація підіймається з найнижчого рівня на найвищий (рис. 3)).

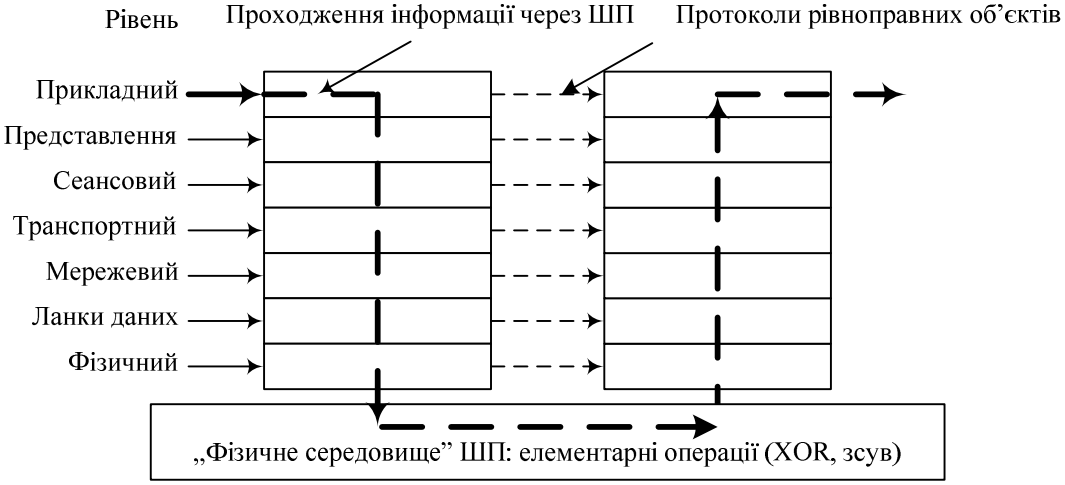

Рис. 3. Структура шифропроцесора

Кожний *N*-рівень є *N*-спецпроцесором, який складається з протокольного *N*-процесора і (*N*-1)-спецпроцесора. Усі спецпроцесори мають аналогічну структуру (рис. 4).

6. Оцінювання ефективності рішення проблеми *HSC*

Відомі методи оцінювання апаратних витрат на реалізацію протокольних процесорів і спецпроцесорів кожного рівня. Кожний протокольний процесор є універсальним, тому основна частина роботи з проектування програмного забезпечення припадає на розроблення програмного забезпечення протокольних процесорів. Основна складність розроблення апаратного забезпечення припадає на проектування спецпроцесорів. За оцінку ефективності рішення проблеми одночасного проектування апаратного і програмного забезпечення пропонується взяти співвідношення апаратних витрат на реалізацію універсальних протокольних процесорів і спецпроцесорів. Що більша величина цього співвідношення, то краще розв'язана проблема.

У роботах [17–19] показано, що в багаторівневій структурі можна прийняти апаратні витрати на реалізацію протокольного процесора в k разів більшими за апаратні витрати на реалізацію спецпроцесора цього самого рівня. Там же показано, що значення k наближається до 1.

Якщо вважати, що значення k є приблизно однаковим для усіх рівнів, то можна оцінити апаратні витрати на реалізацію усієї багаторівневої системи. Якщо позначити апаратні витрати на реалізацію спецпроцесора найнижчого (першого) рівня як v, то визначимо апаратні витрати на реалізацію кожного з рівнів за наведених припущень (див. таблицю).

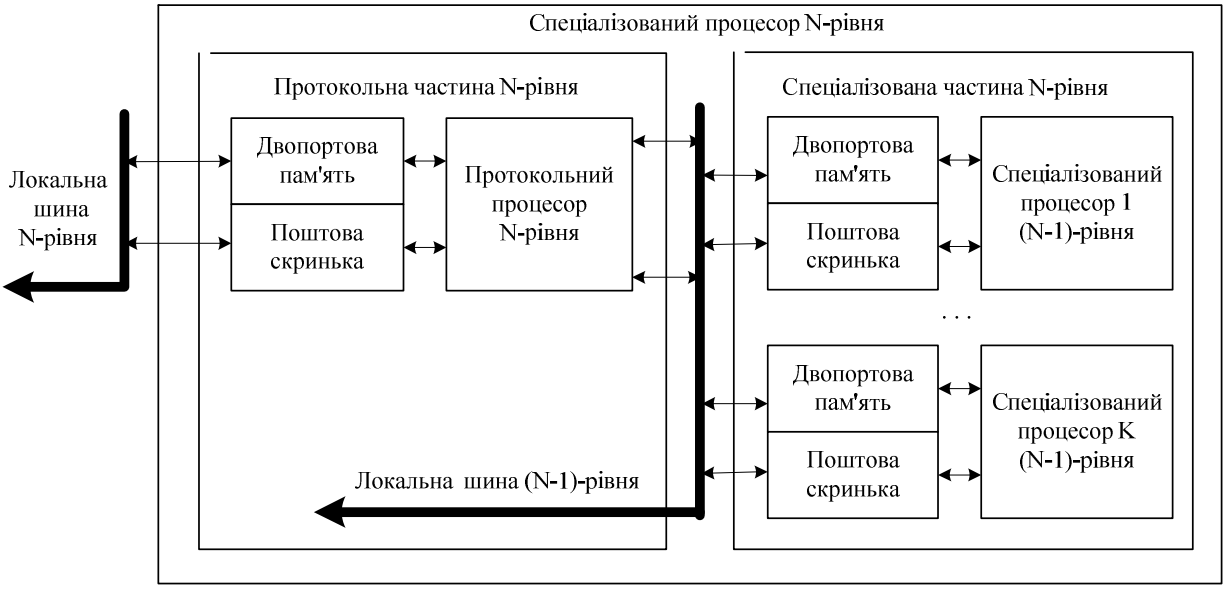

Рис. 4. Структура N-спецпроцесора (N-рівня)

Апаратні витрати

Рівень (i)	**Апаратні витрати на реалізацію спецпроцесора (A_{ci})**	Апаратні витрати на реалізацію протокольного процесора ($A_п$)	Апаратні витрати на реалізацію рівня ($A_i=A_c+A_п$)	Ефективність рішення проблеми HSC (A_i/A_{c1})
1	**v**	vk	v(1+k)	$(1+k)^1$
2	v(1+k)	v(1+k)k	$v(1+k)^2$	$(1+k)^2$
3	$v(1+k)^2$	$v(1+k)^2k$	$v(1+k)^3$	$(1+k)^3$
4	$v(1+k)^3$	$v(1+k)^3k$	$v(1+k)^4$	$(1+k)^4$
5	$v(1+k)^4$	$v(1+k)^4k$	$v(1+k)^5$	$(1+k)^5$
6	$v(1+k)^5$	$v(1+k)^5k$	$v(1+k)^6$	$(1+k)^6$
7	$v(1+k)^6$	$v(1+k)^6k$	$v(1+k)^7$	$(1+k)^7$

За такого підходу практично необхідно розробляти апаратуру тільки спецпроцесора найнижчого 1-го рівня, його апаратна складність A_{c1} дорівнює v. Ефективність вирішення проблеми одночасного розроблення апаратного і програмного забезпечення комп'ютерних систем оцінюється співвідношенням цієї величини A_{c1} до величини апаратної складності відповідного i-го рівня системи A_i: $e = A_{c1}/A_i$. Як видно (див. таблицю), ця величина зростає експоненціально із збільшенням кількості рівнів у комп'ютерній системі.

Найскладнішим випадком є вирішення проблеми на найнижчому рівні. Із зростанням номеру рівня складність вирішення проблеми зменшується. Для ієрархічної багаторівневої системи складність вирішення проблеми експоненціально залежить від кількості рівнів.

Висновки

У роботі запропоновано і обґрунтовано метод оцінювання ефективності розв'язання проблеми одночасного розроблення апаратного і програмного забезпечення комп'ютерних систем шляхом оцінювання апаратної складності протокольних процесорів і спецпроцесорів, які входять до складу системи. Для багаторівневих комп'ютерних систем ефективність розв'язання проблеми експоненціально збільшується із збільшенням кількості рівнів. Найкращою за цим критерієм є система з найбільшою кількістю рівнів. Рекомендується під час проектування багаторівневих систем обирати системи з максимально допустимою за іншими критеріями кількістю рівнів.

1. David Stewart. Migrating software into hardware. EDN Europe, Issue 12/2008, p. 48. 2. Hardware/Software Co-design for Hyperelliptic Curve Cryptography (HECC) on the 8051 μP. Lejla Batina, David Hwang1, Alireza Hodjat, Bart Preneel, and Ingrid Verbauwhede. CHES 2005, LNCS 3659, pp. 106–118, © International Association for Cryptologic Research 2005. 3. Patrick R. Schaumont. A Practical Introduction to Hardware/Software Codesign. © Springer Science+Business Media, Springer New York Dordrecht Heidelberg London, LLC 2010. 4. Мельник А.О., Коркішко Т.А. Система підтримки виконання алгоритмів криптографічного захисту інформації на основі програмованого процесора та криптографічних акселераторів // Вісник Державного університету "Львівська політехніка" № 385 «Комп'ютерні системи та мережі». Львів. Видавництво Державного університету «Львівська політехніка». 2000. С. 77 – 80. 5. Грушо А.А. Тимонина Е.Е. Теоретические основы защиты информации. Издательство Агентства "Яхтсмен". 1996 г. http://kiev-security.org.ua. 6. Таненбаум А. Многоуровневая организация ЭВМ. – М.: Изд-во «Мир», 1979. 7. Мельник А.О. Архітектура комп'ютера. Підручник. – Луцьк: Волинська обласна друкарня, 2008. – 470 с. 8. Рабинович З.Л., Раманаускас В.А. Типовые операции в вычислительных машинах. – К.: Техніка, 1980. – 264 с., ил. 9. David Steward, Richard Taylor and Skip Hovsmith. Programmable coprocessor generation from executable code: Part 1. Source versus binary hardware/software partitioning. (02/09/09, 06:49:00 PM EST) http://www.embedded.com/design/multicore/213401916.10. Applying Multicore and Virtualization to Industrial and SafetyRelated Applications. http://www.embedded-know-how.com. Printed in USA 0209/SI/S2/PDF. Copyright © 2009 Intel Corporation. 11. ДСТУ ISO/IEC 7498-1:2004. Інформаційні технології. Взаємозв'язок відкритих систем. Базова еталонна модель. Частина 1. Базова модель (ISO/IEC 7498-1:1994, IDT). 12. Протоколы информационно-вычислительных сетей: Справочник / С.А. Аничкик, С.А. Белов, А.В. Бернштейн и др.; Под ред. И.А. Мизина, А.П.Кулешова. – М.: Радио и связь, 1990. – 504 ст.: ил. 13. ГОСТ 28906-91 (ИСО 7498-84, ИСО 7498-84 Доп.1-84). Системы обработки информации. Взаимосвязь открытых систем. Базовая эталонная модель. 14. Dave Hughes.Designing NAND Flash into embedded systems. August 2010 issue of Boards & Solutions Magazine, ECE, pp. 6-7.15. Грушвицкий Р.И., Мураев А.Х., Угрюмов Е.П. Проектирование систем на микросхемах программируемой логики. – СПб.: БХВ-Петербург, 2002. – 608 с. 16. Березко Л.О., Троценко В.В. Мультипроцесор на ПЛІС. Вісник Національного університету "Львівська політехніка" "Комп`ютерні системи та мережі". № 573. – Львів, 2006. – С.10–14. 17. Глухов В.С. Оцінювання апаратних витрат на реалізацію багаторівневої комп'ютерної системи з врахуванням закону Амдаля // Вісник Нац. ун-ту «Львівська політехніка» «Комп'ютерні науки та інформаційні технології» № 663. – Львів, 2010. – С.17–23. 18. Глухов В.С. Оцінка апаратних витрат на реалізацію багаторівневої комп'ютерної системи // Вісник Нац. ун-ту «Львівська політехніка» «Комп'ютерні науки та інформаційні технології» № 629. – Львів, 2008. – С.13–20. 19. Глухов В.С. Вибір багатоядерних структур для пристроїв обробки ЕЦП // Вісник Нац. ун-ту "Львівська політехніка" "Комп`ютерні системи та мережі". № 658. – Львів, 2009. – С.35–39. 20. Глухов В.С., Євтушенко К.С., Заіченко Н.В., Оліярник Б.О. "Криптографічні засоби спеціалізованої бортової ЕОМ для бронетехніки" // Вісник Хмельницького нац. ун-ту. – 2007. – № 2. – С.29–33.