

УДК 004.94

## ОБҐРУНТУВАННЯ ВИБОРУ МЕТОДУ ТА ЗАСОБІВ МОДЕЛЮВАННЯ МЕРЕЖ НА КРИСТАЛІ

*О. Ю. Романов, О. М. Лисенко*, д-р техн. наук, проф.

Національний технічний університет України «Київський політехнічний інститут»

kaf201@fel.kpi.ua

*Розглянуто актуальні напрями дослідження мереж на кристалі та відомі підходи до їх моделювання. Охарактеризовано аналітичне та високорівневе / низькорівневе імітаційне моделювання. Запропоновано застосування мови програмування SystemC як альтернативне рішення з побудови моделей мереж на кристалі.*

**Ключові слова:** мережа на кристалі; імітаційне моделювання; HDL-моделювання; SystemC.

*The actual trends of networks-on-chip research and known approaches to their modeling are considered. The characteristics of analytic and high- / low- level simulation are given. The programming language SystemC as an alternative solution to create models of networks-on-chip is proposed.*

**Keywords:** network on chip; simulation; HDL-modeling; SystemC.

### Вступ

Постійний розвиток сучасних систем на кристалі (СнК) привів до появи багатопроцесорних систем. Так, наприклад, компанія Intel розробила два дослідні процесори з 48 та 80 ядрами [1; 2], а також веде розробку експериментального процесора на 167 ядер [3]. Процесор Tile64 має 64 ядра і випускається серійно [4].

У багатопроцесорних СнК об'єднання вузлів за допомогою шин не є ефективним [5], а отже застосовується мережевий підхід, який викликав появу нового класу систем — мереж на кристалі (МнК). МнК — це множина обчислювальних вузлів, об'єднаних загальною підсистемою зв'язку, яка складається з роутерів і коротких з'єднань між ними.

Підсистема зв'язку МнК займає значні ресурси кристала і є енергоємною, що зумовлює необхідність пошуку оптимальних рішень її побудови [6].

### Постановка проблеми та цілей дослідження

У праці [7] визначено основні актуальні напрями пошукових досліджень за тематикою МнК:

- 1) моделювання мережевого трафіку та створення відповідних тестових задач;
- 2) відображення задач на МнК та їх планування;
- 3) маршрутизація та керування потоками даних у МнК;
- 4) забезпечення необхідної якості обслуговування;
- 5) керування енергоспоживанням, температурним режимом та тактуванням;
- 6) забезпечення надійності та відмовостійкості МнК;
- 7) створення оптимальної топології з'єднань у МнК;

8) розробка ефективної структури роутерів та мережевих каналів;

9) планування розміщення МнК на кристалі;

10) прототипування, тестування та верифікація МнК;

11) моделювання, аналіз та симуляція МнК.

Велика кількість напрямів досліджень відображає складність МнК як об'єкта досліджень. Слід також наголосити, що на основі моделювання, аналізу і симуляції МнК ґрунтуються інші напрями пошуку. Тому вибір адекватних методів та засобів моделювання МнК є актуальним.

### Аналітичне моделювання МнК

Моделювання МнК має на меті отримання та аналіз критичних характеристик мереж, таких як пропускна спроможність, енергоспоживання та витрати ресурсів, стійкість до помилок та ін. Залежно від мети дослідження, моделі існують різного рівня абстракції і, відповідно, мають різну точність та витрати ресурсів на проведення моделювання. Типовим підходом є виведення, аналіз та апроксимація аналітичних формульних залежностей, що описують процеси, які відбуваються в МнК, або їх характеристики.

У загальному вигляді процес синтезу МнК відбувається шляхом відображення характеристичного графу задачі (APCG) на архітектуру МнК. APCG являє собою  $G = G(C, A)$  напрямлений граф, де  $C$  — множина вершин, що характеризують обчислювальні вузли;  $A$  — множина комунікаційних процесів, які відбуваються між вузлами. У свою чергу, архітектура МнК характеризується: топологією  $T(R, Ch)$ , де  $R$  і  $Ch$  — множини маршрутизаторів та фізичних каналів між ними; механізмом маршрутизації ( $P_R$ ); функцією відображення вершин APCG на маршрутизатори МнК ( $\Omega(C)$ ).

Згідно з наведеними вище визначеннями можна, наприклад, вивести залежність витрат енергії на комунікацію:

$$E = \sum_{\forall a_{i,j}} v(a_{i,j}) E_{bit}(\Omega(c_i), \Omega(c_j)), \quad (1)$$

де  $v(a_{i,j})$  — ємність комунікаційного процесу між вузлами  $i, j$ ;  $E_{bit}(\Omega(c_i), \Omega(c_j))$  — енергія, що витрачається на передачу 1 біта даних між вузлами  $c_i$  та  $c_j$ .

Задачею мінімізації енергії комунікації є пошук такого  $\Omega(C)$ , щоб з'єднання комунікаційного процесу з великою ємністю мали низькі витрати енергії на передачу 1 біта. Для регулярних топологій МнК дана задача розв'язується в роботі [8].

Подібним чином виглядає і формула знаходження загального об'єму даних, що передається в МнК між вузлами:

$$V = \sum_{\forall a_{i,j}} v(a_{i,j}) L_{i,j}(P_R(r, i, j)), \quad (2)$$

де  $L_{i,j}(P_R(r, i, j))$  — відстань між вузлами  $i$  та  $j$  згідно з алгоритмом маршрутизації.

Застосування алгоритму маршрутизації, який зменшує середню відстань між вузлами, дає можливість зменшити навантаження на мережу.

Формули (1), (2) є типовою квадратичною задачею призначень, що являє собою мінімізацію суми добутків функцій вартості на їх ваги. Аналогічним чином можна знайти формули визначення інших характеристик МнК шляхом підстановки відповідних метрик продуктивності або ресурсовитрат, а об'єднавши декілька функцій у систему, отримаємо аналітичну модель МнК. Такий підхід застосовний для 2, 3, 4, 5, 6, 7 та 9 напрямів пошукових досліджень з тематики МнК, визначених раніше.

Деяко окремо стоять аналітичні моделі мережевого трафіку. Для МнК із однорідним трафіком застосовні стохастичні моделі та моделі для самоподібного трафіку, у випадку мультимедійних додатків [9; 10].

Аналітичне моделювання МнК має низку переваг: це очевидний підхід, який не потребує застосування спеціальних систем автоматизованого проектування (САПР), хоча використання, наприклад, Mathcad або MatLab значно полегшує обчислення, а Simulink навіть дозволяє описати модель у графічному вигляді. Однак аналіз та оптимізація таких моделей ускладнені через їхню комплексність і нелінійність поведінки МнК.

Прикладом аналітичної моделі є робота [11], в якій у вигляді виразу подана залежність швидкості паралельної обробки даних від параметрів МнК і

аналізується вплив на неї затримок при передаванні даних із збільшенням розмірності мережі.

### Імітаційні моделі МнК високого рівня

Інша група моделей МнК — імітаційні. Залежно від рівня деталізації розрізняють моделі низького та високого рівнів.

Низькорівневе моделювання — це емуляція роботи мережі на рівні логічних вентилів; воно буде розглянуто нижче.

Високорівневе моделювання — це симуляція роботи мережі на рівні розповсюдження потоків даних. Цей підхід характеризується швидкістю розробки, гнучкістю налаштувань та відносно невеликим часом моделювання. При цьому підході моделювання являє собою тестування моделі розповсюдження даних у МнК, описаної мовою високого рівня.

Прикладом високорівневої моделі є виконана в роботі [12] розробка, де передача даних у МнК подана у вигляді паралельно розв'язаних задач, описаних на мові Сі.

У роботі [13] запропоновано універсальний програмний симулятор мовою Java та наведено результати моделювання різних регулярних топологій. Модель являє собою опис маршрутизаторів та обчислювальних вузлів МнК як окремих об'єктів, що функціонують незалежно один від одного. Обчислювальні вузли виступають генераторами / споживачами мережевого трафіку, а маршрутизатори виконують прийом та передачу даних згідно з алгоритмом маршрутизації.

Такий підхід до опису моделі МнК є дуже поширеним і має багато переваг: виконується імітація роботи мережі, наближена до натурального експерименту, існує можливість індивідуальної конфігурації кожного маршрутизатора, налаштування алгоритму маршрутизації, підключення різноманітних тестових послідовностей мережевого трафіку і т. ін.

У праці [14] запропоновано швидко високорівневу модель МнК на основі мережевої моделі OSI також із застосуванням мови Java та фреймворку Qt Jambi.

Порівняно з попереднім прикладом, реалізовано моделювання нерегулярних топологій: кожний маршрутизатор містить таблицю маршрутизації, а топологія МнК задається за допомогою матриці зв'язків між роутерами. Параметри моделі задаються за допомогою конфігураційного xml-файлу.

Результати моделювання виводяться у діалогове вікно, а вибрані параметри зберігаються у зведену таблицю. Симулятор дає можливість виконувати декілька ітерацій моделювання поспіль із різною конфігурацією.

Застосування мови програмування Java з використанням фреймворку Qt Jambi надає всі переваги об'єктно-орієнтованого програмування, кросплатформності програмних рішень і швидкості їх розробки, а повна незалежність компонентів системи дає можливість виконувати розробку, модифікацію та апробацію різних моделей МнК.

У попередній моделі програмними засобами реалізовано графічний інтерфейс, однак існують уже готові програмні продукти, які полегшують проведення моделювання. Так, наприклад, у праці [15] використовується моделювання у мережах Petri у середовищі симулятора Visual Object Net для аналізу конкуренції, взаємодії та конфліктів даних у комунікаційному просторі МнК.

Увагу слід приділити агентному підходу, описаному в роботі [16]. Суть його полягає в тому, що предметна галузь подається у вигляді безлічі агентів, що взаємодіють між собою. Розробником моделі описуються правила створення, знищення та зміни агентів. Агент — це деякий об'єкт, що володіє пам'яттю і здатністю приймати рішення, і, таким чином, власною поведінкою різного рівня складності. Внутрішня структура агента може бути описана різними способами — від формальної логіки до нейронних мереж. У момент запуску процесу моделювання кожен агент починає функціонувати згідно з індивідуальним алгоритмом роботи, і глобальна поведінка системи виникає як результат взаємодії всієї множини агентів. Завдяки цьому існує можливість поступового внесення коректив у алгоритм роботи агента, тим самим, деталізуючи модель.

Отже, наявність декількох сценаріїв функціонування агента різного рівня складності дає можливість моделювати роботу системи на різних рівнях абстракції. Опис моделей такого типу виконується на спеціалізованих мовах опису моделей (наприклад, UML), а розробка — у САПР типу AnyLogic.

Високорівневе імітаційне моделювання застосовне для більшості напрямів дослідження МнК, де немає прив'язки до апаратної реалізації та необхідно швидко отримати результати моделювання з достатньою точністю.

### **Імітаційні моделі МнК нижнього рівня**

Недоліком високорівневих моделей МнК є неможливість їх синтезу та відносно низька точність. Альтернативою є низькорівневе моделювання — емуляція роботи мережі на рівні логічних вентилів. Компоненти моделі формуються із застосуванням мов опису апаратури (наприклад, Verilog або VHDL). При цьому їх функціонування аналізується за допомогою спеціалізованих

програм моделювання апаратних засобів (наприклад, пакет ModelSim), і така модель може бути синтезованою за допомогою спеціалізованих САПР (наприклад, Quartus II або Synplify Pro). HDL-мови підтримують інтерфейс для високорівневих мов програмування, що полегшує сумісну симуляцію та верифікації (наприклад, VPI/PLI [17], DPI [18]).

Таким чином, даний підхід використовується за необхідності моделювання, максимально наближеного до реалізації МнК на фізичному рівні (моделювання на рівні циклів, cycle-accurate), синтезу і прототипування.

Цей підхід став дуже поширеним. Так, у роботі [19] за допомогою HDL опису синтезовано різні варіанти реалізації маршрутизаторів *Ethereal* МнК для оцінювання займаної площі на кристалі та максимальної тактової частоти. У роботі [20] застосовуються маршрутизатори, описані на Verilog для побудови, моделювання та прототипування mesh 4x4 МнК MPEG-2 декодера. У роботі [21] VHDL модель МнК використовується для оцінювання енергетичних витрат, як і в роботі [22], де за допомогою VHDL моделі генеруються тестові послідовності та HDL список з'єднань для подальшого SPICE моделювання. Класичний шлях від опису на HDL, а потім і до моделювання в ModelSim та сумісної емуляції на FPGA, застосовується в роботі [23] для швидкого апаратно-програмного моделювання МнК. У дисертаційній роботі [24] запропоновано цільну VHDL модель маршрутизатора і МнК MoCREs, побудовану на його основі, для проведення моделювання та синтезу в FPGA. Слід окремо виділити Verilog бібліотеку з відкритим вихідним кодом Netmaker [25; 26], у якій реалізовано опис класичного маршрутизатора з віртуальними каналами та засоби генерації регулярних топологій. У праці [27] можливості бібліотеки поширені на моделювання нерегулярних МнК шляхом модифікації модуля побудови зв'язків між вузлами і модуля маршрутизації. Ще один приклад — Verilog бібліотека NoCSimp [28], в основі якої лежить wormhole маршрутизатор зі спрощеною структурою та FCFS (First Come First Serve) арбітражем [29]. Реалізована можливість проведення моделювання нерегулярних топологій МнК шляхом настроювання таблиць маршрутизації, а також полегшений синтез МнК завдяки простоті реалізації.

Низькорівневий підхід застосовний практично для всіх напрямів дослідження МнК, і головною його перевагою є висока точність та настроюваність моделей та можливість синтезу МнК. Однак створення таких моделей потребує значних витрат часу, а моделювання відбувається в спеці-

алізованих програмах моделювання апаратних засобів (наприклад, пакет ModelSim). Згідно з роботою [30], максимальна швидкість моделювання за допомогою ModelSim становить приблизно  $3,2 \cdot 10^3$  циклів/с, чого не достатньо для аналізу МНК великої розмірності. Існуючі ж підходи сумісної апаратно-програмної симуляції та прототипування МНК вимагають спеціалізованого обладнання та специфічних підходів, що ускладнює застосування таких рішень.

### **SystemC як компромісне рішення між вискорівневим та низькорівневим моделюванням**

SystemC — мова проектування та верифікації моделей системного рівня, реалізована у вигляді C++ бібліотеки з відкритим кодом. Бібліотека містить ядро подієвого моделювання, що дає можливість отримати виконувану модель пристрою. Мова застосовується для побудови транзакційних і поведінкових моделей, а також для високорівневого синтезу пристроїв. У SystemC використовується низка понять, схожих на ті, які мають мови опису апаратури VHDL і Verilog — інтерфейси, процеси, сигнали, подієвість, ієрархія модулів. Завдяки цьому SystemC придатна для поведінкового моделювання та RTL-синтезу (Register Transfer Level — на рівні регістрів).

SystemC широко застосовується серед розробників МНК. На основі SystemC створено бібліотеку Xpipes [31], що дає можливість здійснювати повний цикл моделювання та синтезу МНК [32]. У роботі [33] запропоновано високорівневу SystemC-модель ARTS для порівняльного моделювання мережевого та шинного методів побудови систем на кристалі. На основі SystemC достатньо відомими є симулятори Noxim [34], NIRGAM [35] та ін.

Популярність SystemC зумовлена тим, що вона ґрунтується на мові C/C++, на якій розроблено багато стандартних бібліотек, завдяки чому полегшується сумісна симуляція та верифікація моделей МНК. Однак за своєю природою C/C++ є послідовною мовою (інструкції виконуються одна за одною), в той час як процеси в апаратурі відбуваються одночасно і паралельно. Це вимагає від програміста освоєння нової парадигми програмування, а також специфічних інструментів, таких як процеси, події, сигнали та ін.

Хоча SystemC є синтезованою мовою, стосовно МНК вона, в першу чергу, використовується як високорівнева мова для поведінкового абстрактного моделювання, завдяки чому моделювання є більш швидким, ніж за допомогою HDL-мов (до  $20 \cdot 10^3$  циклів/с) [36; 37], але синтез МНК є більш складним. У роботі [38] наведено резуль-

тати порівняння моделей на System Verilog та SystemC, що реалізують МНК типу mesh 8x8. Застосування SystemC дало можливість зменшити тривалість моделювання у 3.9-10.2 разів та обсяг займаної пам'яті у 29.4-121 разів при максимальній похибці передбачення 6.1%. У деяких роботах застосовується гібридний підхід, де за допомогою SystemC генеруються тестові послідовності, а сама модель реалізована на HDL [39; 40].

### **Висновки**

Таким чином, при розробці та дослідженні мереж на кристалі актуальним є вибір універсального підходу до їх моделювання. Серед типових підходів виділяють: аналітичний (аналіз таких моделей ускладнено через їх комплексність та нелінійність поведінки МНК); високорівневе імітаційне моделювання (застосовне для більшості напрямів дослідження МНК, де немає прив'язки до апаратурної реалізації та необхідно швидко отримати результати моделювання з достатньою точністю); низькорівневе HDL-моделювання (висока точність, настроюваність моделей та можливість синтезу МНК, але високі витрати часу на розробку моделей та проведення моделювання).

Запропоновано використання мови SystemC для побудови моделей МНК, що дає можливість зменшити недоліки та об'єднати переваги низькорівневого та високорівневого підходів. Моделювання за допомогою SystemC є застосовним для усіх напрямів пошукових досліджень за тематикою МНК.

### **ЛІТЕРАТУРА**

1. Howard J. A 48-core IA-32 processor in 45 nm CMOS using on-die message passing and DVFS for performance and power scaling / J. Howard, S. Dighe, S.R. Vangal, et al. // IEEE Journal of Solid-State Circuits. — 2011. — Vol. 46. — No. 1. — P. 173–183.
2. Hoskote Y. A 5-GHz mesh interconnect for a teraflops processor / Y. Hoskote, S. Vangal, A. Singh // IEEE Micro. — 2007. — Vol. 27. — No. 5. — Pp. 51–61.
3. Truong D. N. A 167-processor computational platform in 65 nm CMOS / D. N. Truong, W. H. Cheng, T. Mohsenin, et al. // IEEE Journal of Solid-State Circuits. — 2009. — Vol. 44. — No. 4. — P. 1130–1144.
4. Wentzloff D. On-chip interconnection architecture of the tile processor / D. Wentzloff, P. Griffin, H. Hoffmann, et al. // IEEE Micro. — 2007. — Vol. 27. — No. 5. — P. 15–31.
5. Bjerregaard T. A survey of research and practices of Network-on-chip / T. Bjerregaard, S. Mahadevan // ACM Computing Surveys. — 2006. — Vol. 38. — No. 1. — 51 p.
6. Taylor M. The Raw Processor — A Scalable 32-bit Fabric for Embedded and General Purpose

Computing / M. Taylor, J. Kim, J. Miller // *Proceedings of Hotchips XIII*. — MIT, 2001. — P. 3–6.

7. *Marculescu R.* Outstanding Research Problems in NoC Design: System, Microarchitecture, and Circuit Perspectives / R. Marculescu, U. Ogras // *IEEE Transactions on Computer Aided Design of Integrated Circuits and Systems*. — 2009. — Vol. 28. — No. 1. — P. 3–21.

8. *Hu J.* Energy-aware mapping for tile-based NOC architectures under performance constraints / J. Hu, R. Marculescu // *Proceedings of Asia South Pacific Design Automatic Conference*. — Kitakyushu, Japan. — 2003. — P. 233–239.

9. *Soteriou V.* A statistical traffic model for on-chip interconnection networks / V. Soteriou, H.-S. Wang, L. Peh // *Proceedings International Symposium Modeling, Analysis, Simulation Computer Telecommunication Systems*. — 2006. — P. 104–116.

10. *Varatkar G.* On-chip traffic modeling and synthesis for MPEG-2 video applications / G. Varatkar, R. Marculescu // *IEEE Transactions Very Large Scale Integrated (VLSI) Systems*. — 2004. — Vol. 12. — No. 1. — P. 108–119.

11. *Xiaowen Ch.* Speedup Analysis of Data-parallel Applications on Multi-core NoCs / Ch. Xiaowen, et al. // *8<sup>th</sup> IEEE Conference on ASIC (ASICON)*. — Changsha, China, 2009. — P. 105–108.

12. *Mingsong Lv.* RTNoC: A Simulation Tool for Real-Time Communication Scheduling on Networks-on-Chips / Lv. Mingsong, et al. // *International Conference on Computer Science and Software Engineering*. — 2008. — Vol. 4. — P. 102–105.

13. *Al-Nayeem A.* GpNoCsim 1.0 User's Guide / A. Al-Nayeem, T. Z. Islam. — 2006. — 13 p.

14. *Романов А. Ю.* Разработка программного симулятора сетей на кристалле / А. Ю. Романов, Д. А. Феськов // *Электроника и связь: Электроника и нанотехнологии*. — К. : НТУУ «КПИ», 2011. — Т. 4(63). — С. 48–52.

15. *Freitas H. C.* Evaluating On-Chip Interconnection Architectures for Parallel Processing / H. C. Freitas, P.O.A. Navaux // *11th IEEE International Conference on Computational Science and Engineering Workshops, 2008 (CSE WORKSHOPS'08)*. — 2008. — P. 188–193.

16. *Короткий Е. В.* Метод моделирования реконфигурируемых сетей на кристалле / Е. В. Короткий, А. Н. Лысенко // *Вісник НТУУ «КПІ». Інформатика, управління та обчислювальна техніка: зб. наук. пр.* — К. : Век+, 2009. — №51. — С. 218–224.

17. *Saifhashemi A.* Verilog HDL, Powered by PLI: a Suitable Framework for Describing and Modeling Asynchronous Circuits at All Levels of Abstraction / A. Saifhashemi, H. Pedram // *Design Automation Conference, 2003. Proceedings*. — 2003. — P. 330–333.

18. *Sutherland S.* Integrating SystemC Models with Verilog and SystemVerilog Models Using the Sys-

temVerilog Direct Programming Interface / S. Sutherland // *SNUG Europe*. — 2004. — 17 p.

19. *Goossens K.* Æthereal network on chip: concepts, architectures, and implementations / K. Goossens, J. Diellisen, A. Radulescu // *IEEE Design & Test of Computers*. — 2005. — Vol. 22. — No. 5. — P. 414–421.

20. *Ogras U. Y.* Challenges and Promising Results in NoC Prototyping Using FPGAs / U. Y. Ogras, R. Marculescu, H.G. Lee, et al. // *IEEE Micro*. — 2007. — Vol. 27. — No. 5. — P. 86–95.

21. *Palma J. C. S.* Mapping embedded systems onto NoCs: the traffic effect on dynamic energy estimation / J.C.S. Palma, C.A.M. Marcon, F.G. Moraes, et al. // *Proceedings of the 18th annual symposium on Integrated circuits and system design (SBCCI'05)*. — ACM, 2005. — P. 196–201.

22. *Marcon C. A. M.* Modeling the Traffic Effect for the Application Cores Mapping Problem onto NoCs / C.A.M. Marcon, J.C.S. Palma, N.L.V. Calazans, et al. // *VLSI-SoC: From Systems To Silicon*. — Springer, 2007. — Vol. 240. — P. 179–194.

23. *Genko N.* A Complete Network-On-Chip Emulation Framework / N. Genko, D. Atienza, G. De Micheli, et al. // *Design, Automation and Test in Europe, 2005. Proceedings*. — 2005. — Vol. 1. — P. 246–251.

24. *Janarthanan A.* Networks-on-chip based high performance communication architectures for FPGAs: Ph.D. dissertation: Computer engineering / A. Janarthanan. — Cincinnati: Univ. of Cincinnati, 2008. — 143 p.

25. *Mullins R.* The design and implementation of low-latency on-chip network / R. Mullins, A. West, S. Moore // *Proc. of 11 ASPDAC*. — Japan, 2006. — P. 164–169.

26. Fully-synthesizable parameterized NoC implementations library: Netmaker, <http://www-dyn.cl.cam.ac.uk/~rdm34/wiki>.

27. *Romanov O.* The Comparative Analysis of the Efficiency of Regular and Pseudo-optimal Topologies of Networks-on-Chip Based on Netmaker / O. Romanov, O. Lysenko // *Advances and Challenges in Embedded Computing. Proceedings*. — Montenegro, Bar: 2012. — P. 13–16.

28. *Романов О. Ю.* Порівняльний аналіз результатів HDL-модельовання квазіоптимальних і регулярних топологій мереж на кристалі / О. Ю. Романов // *Проблеми інформатизації та управління. Збірник наукових праць*. — К. : НАУ, 2012. — Вип. 3 (39). — С. 124–129.

29. *Романов О. Ю.* Ресурсоефективний роутер для багатопроцесорної мережі на чипі / О. Ю. Романов, О.М. Лисенко // *Вісник НТУ «ХПІ». Тематичний випуск: Інформатика і моделювання*. — Харків: НТУ «ХПІ». — 2011. — № 17. — С. 86–92.

30. *Genko N.* A Complete Network-On-Chip Emulation Framework / N. Genko, D. Atienza, G. De Mich-

eli, et al. // Design, Automation and Test in Europe, 2005. Proceedings. — 2005. — Vol. 1. — P. 246–251.

31. Bertozzi D. Xpipes: A network-on-chip architecture for gigascale systems-on-chip / D. Bertozzi, L. Benini // IEEE Circuits and Systems Magazine. — 2004. — Vol. 4. — No. 2. — P. 18–31.

32. Bertozzi D. NoC synthesis flow for customized domain specific multiprocessor systems-on-chip / D. Bertozzi, S. Murali, A. Jalabert // IEEE Transactions on Parallel and Distributed Systems. — 2005. — Vol. 16. — No. 2. — P. 113–129.

33. Mahadevan S. ARTS: A SystemC-based framework for multiprocessor Systems-on-Chip modelling / S. Mahadevan, K. Virk, J. Madsen // Design Automation for Embedded Systems. — Springer, 2007. — Vol. 11. — No. 4. — P. 285–311.

34. Fazzino F. Noxim: Network-on-chip simulator. — [Електронний ресурс] / F. Fazzino, M. Palesi, D. Patti. — Режим доступу: <http://noxim.sourceforge.net/>.

35. Jain L. NIRGAM: A Simulator for NoC Interconnect Routing and Application Modeling. — [Електронний ресурс] / L. Jain. — [Version 1.1]. — 2007. — 27 p. — Режим доступу: <http://nirgam.ecs.soton.ac.uk/Documentation.php>.

36. Goossens K. Æthereal network on chip: concepts, architectures, and implementations / K. Goossens, J. Dielissen, A. Radulescu // IEEE Design & Test of Computers. — 2005. — Vol. 22. — No. 5. — P. 414–421.

37. Genko N. Feature-NoC emulation: a tool and design flow for MPSoC / N. Genko, D. Atienza, G. De Micheli, L. Benini // IEEE Circuits and Systems Magazine. — 2007. — Vol. 7. — No. 4. — P. 42–51.

38. Короткий Є.В. Високопродуктивна модель маршрутизатора для мережі-на-кристалі з агрегацією каналів / Є. В. Короткий, О. М. Лисенко // Збірник наукових праць ВІТІ НТУУ «КПІ». — 2012. — № 1. — С. 56–67.

39. Chan J. NoCGEN: A Template Based Reuse Methodology for Networks on Chip Architecture / J. Chan, S. Parameswaran // 17<sup>th</sup> International Conference on VLSI Design, 2004. Proceedings. — 2004. — P. 717–720.

40. Goossens K. A Design Flow for Application-Specific Networks on Chip with Guaranteed Performance to Accelerate SOC Design and Verification / K. Goossens, J. Dielissen, O. Gangwal // Proceedings of the conference on Design, Automation and Test in Europe (DATE'05). — 2005. — Vol. 2. — P. 1182–1187.

Стаття надійшла до редакції 25.11.2013