

**В. А. Мельник<sup>1,2</sup>, І. І. Лопіт<sup>1</sup>, А. Ю. Кіт<sup>1</sup>**

<sup>1</sup> Національний університет “Львівська політехніка”,  
кафедра безпеки інформаційних технологій;

<sup>2</sup> Люблінський Католицький Університет Івана Павла II, м. Люблін, Польща,  
Інститут математики і інформатики.

## **ВИМОГИ ДО ПРОГРАМНОГО ІНТЕРФЕЙСУ РЕКОНФІГУРОВНИХ АПАРАТНИХ ПЛАТФОРМ ОБЧИСЛЮВАЛЬНИХ ВУЗЛІВ КФС**

© Мельник В. А., Лопіт І. І., Кіт А. Ю., 2016

**В доповіді запропоновано спосіб організації та передачі інформації для автоматичного створення комп’ютерних пристроїв в реконфігуровних апаратних платформах обчислювальних вузлів кіберфізичних систем. Сформульовано основні вимоги до програмного інтерфейсу, що реалізує протокол обміну інформацією.**

**Ключові слова – кіберфізична система, реконфігуровна апаратна платформа, програмовна логічна інтегральна схема, програмний інтерфейс.**

## **REQUIREMENTS FOR SOFTWARE INTERFACE OF RECONFIGURABLE HARDWARE PLATFORMS OF CYBER-PHYSICAL SYSTEMS COMPUTING NODES**

© Melnyk V. A., Lopit I. I., Kit A. Y., 2016

**This report proposes a way to organize and transfer information to automatically create computer devices in reconfigurable hardware platforms of the cyber-physical systems computing nodes. Basic requirements for the software interface, which implements the interconnection protocol, are given.**

**Keywords – cyber-physical system, reconfigurable hardware platform, field-programmable gates array, software interface.**

### **Вступ**

Для виконання високопродуктивних обчислень сьогодні широко використовують реконфігуровні апаратні платформи. Використання пристроїв із реконфігуровною логікою дозволяє уникнути таких недоліків, притаманних комп’ютерним системам на базі універсальних процесорів, як низька реальна продуктивність, зумовлена невідповідністю структури таких систем структурі виконуваних ними алгоритмів, висока споживана потужність та низька ефективність використання обладнання [1]. Водночас використання реконфігуровного середовища, побудованого на базі програмовних логічних інтегральних схем (ПЛІС), дозволяє уникнути обмежень, які накладає спеціалізація, оскільки ПЛІС може бути в разі потреби перепрограмована для виконання іншого потрібного алгоритму.

### **Постановка задачі**

Для використання реконфігуровного середовища як апаратної платформи обчислювального вузла кіберфізичної системи (КФС) постає питання інтеграції пристроїв реконфігуровної логіки та автоматичного створення в них комп’ютерних пристроїв. Регламентує сеанс з’єднання та передачі інформації в реконфігуровне середовище протокол

обміну інформацією для автоматичного створення комп'ютерних пристроїв. Для забезпечення роботи протоколу обміну інформацією необхідно розробити формат пакета даних, яким оперуватиме протокол обміну інформацією, та вимоги до програмного інтерфейсу, що він реалізує.

### 1. Формат пакета даних для обміну інформацією між керуючим вузлом і реконфігуровним середовищем обчислювального вузла КФС

Завданням реконфігуровного середовища в обчислювальному вузлі КФС є виконання обчислень. Проте перед використанням ПЛІС як обчислювального ресурсу необхідно налагодити з'єднання, отримати інформацію про тип використовуваної ПЛІС, автоматично сформувані конфігурацію та передати її в реконфігуровне середовище. Протокол повинен бути гнучким та надійним, а формат повідомлення, яким він оперує, повинен забезпечити можливість передачі всієї необхідної інформації.

Протокол обміну інформацією слугує для передачі інформації про ПЛІС і її конфігурації. Ця інформація містить дані або корисне навантаження пакета (анг. *payload*). Також для підтримки з'єднання та зберігання керуючої інформації пакет повинен містити відповідний заголовок. На Рис. 1 наведено структуру пакета, в який організовано інформацію та яким оперує протокол обміну.

Біт	0-3	4-7	8-15	16-31
0	Ідентифікатор клієнта		Ідентифікатор сервера	
32	Номер сесії			
64	Порядковий номер пакета			
96	Версія протоколу	Тип пакету	Довжина заголовку	Контрольна сума
128	Відсоток виконання		Зарезервовано	
128+	Дані			

Рис. 1. Структура пакета даних для обміну інформацією між керуючим вузлом і реконфігуровним середовищем обчислювального вузла КФС

У форматі пакета передбачено наступні поля:

1. *Ідентифікатори клієнта та сервера.* Якщо КФС орієнтована на виконання великої кількості обчислювано складних алгоритмів, то до її складу може входити велика кількість пристроїв реконфігуровної логіки. Сервер, що здійснює конфігурування, може також бути не єдиним у системі. Щоб однозначно ідентифікувати клієнта та сервера, необхідно ввести ідентифікатори та вказувати їх у заголовку пакетів, якими обмінюватимуться клієнт із сервером.
2. *Номер сесії.* Це поле дозволяє ідентифікувати конкретний потік передачі даних.
3. *Порядковий номер пакета.* За допомогою цього номера можна пронумерувати пакети в кожній конкретній сесії. При втраті одного із пакетів у полі *порядковий номер* вказується номер, який необхідно надіслати повторно.
4. *Версія протоколу.* В даному полі вказується версія поточного протоколу обміну інформацією для автоматичного створення комп'ютерних пристроїв у реконфігуровних апаратних платформах обчислювальних вузлів.
5. *Тип пакета.* Кожному типу пакета (запит на встановлення з'єднання, підтвердження з'єднання, автентифікація, передача конфігурації, запит на повторне надсилання

пакета та ін.) ставиться у відповідність певна послідовність бітів. Інформація у полях заголовка та дані пакета будуть трактуватись у різний спосіб в залежності від значення типу пакета.

6. *Довжина заголовка.* Це поле визначає розмір заголовка пакета в 4-байтних словах.
7. *Контрольна сума.* У цьому полі вказується хеш-значення, пораховане для всього пакета для здійснення контролю цілісності переданої інформації.
8. *Відсоток виконання.* В цьому полі вказується відсоток переданої інформації від сервера до клієнта чи навпаки.
9. *Дані.* У цьому полі передається конфігурація від сервера до клієнта, характеристики ПЛІС, дані автентифікації та ін., залежно від типу пакета.
10. *Зарезервовані біти.* В пакеті декілька бітів зарезервовано на майбутні потреби, для забезпечення розширюваності протоколу.

## **2. Вимоги до програмного інтерфейсу, що реалізує протокол обміну інформацією**

Реалізуючи протокол обміну інформацією, потрібно забезпечити його надійну роботу та можливість працювати із усіма необхідними йому ресурсами, надавши належної гнучкості. Формуючи вимоги до програмного інтерфейсу, потрібно врахувати особливості архітектури обчислювального вузла КФС та характеристики середовища, в якому він працюватиме.

Програмний інтерфейс повинен відповідати таким вимогам:

1. Гнучкість. Оскільки КФС орієнтована на широкий спектр застосувань, програмний інтерфейс не повинен бути залежним від конкретного застосування.
2. Розширюваність та модульність. Програмний інтерфейс повинен складатись із незалежних модулів, кожен із яких виконує окрему функцію у системі. Така організація програмного інтерфейсу дасть можливість легко пристосовувати його до змін в структурі КФС.
3. Підтримка роботи із базою даних. Існує необхідність зберігати та отримувати доступ до конфігурації, її характеристик та характеристик ПЛІС.
4. Сумісність із протоколами, що лежать на нижчих рівнях мережної моделі. При цьому програмний інтерфейс повинен надавати можливість передачі інформації через Internet чи локальну мережу.
5. Можливість працювати в різних середовищах. Програмний інтерфейс повинен працювати на різних апаратних платформах та бути сумісним із програмним забезпеченням, що може використовуватись в різних реалізаціях КФС.
6. Інкапсульованість – приховування деталей імплементації від клієнтів. Клієнти не потребуватимуть перекомпіляції при зміні імплементації типу вказівників. Якщо динамічна бібліотека використовуватиме непрозорі вказівники, то її модифікація не призведе до бінарної несумісності з прикладними програмами [2].
7. Робота у системі, де існує, як мінімум, один об'єкт, що розсилає повідомлення, та не менше одного одержувача повідомлень, причому їхня кількість і склад можуть динамічно змінюватись під час роботи КФС. При цьому при зміні стану одного об'єкта всіх інших, залежних від нього, сповіщають про цю подію.
8. Програмний інтерфейс повинен працювати під час зміни поведінки об'єкта в залежності від свого стану.
9. Надійність. Коректність роботи програмного інтерфейсу повинна бути відтестована та перевірена.

## Висновок

В доповіді запропоновано формат пакета даних та вимоги до реалізації програмного інтерфейсу, який оперуватиме пакетами для автоматичного створення комп'ютерних пристроїв у реконфігурованих апаратних платформах обчислювальних вузлів КФС. Використання розробленого формату пакета даних дозволить здійснювати передачу інформації про ПЛІС і її конфігурації за допомогою відповідного протоколу обміну інформацією, а виконання вимог до програмного інтерфейсу, що реалізує протокол обміну інформацією, забезпечить його надійну роботу та можливість працювати із усіма необхідними йому ресурсами, надавши належної гнучкості.

*1. Мельник А. О. Персональні суперкомп'ютери: архітектура, проектування, застосування: монографія / А. О. Мельник, В. А. Мельник. – Львів: Видавництво Львівської політехніки, 2013. – 516 с. 2. Gamma, Erich; Helm, Richard; Johnson, Ralph; Vlissides, John (1995). Design Patterns: Elements of Reusable Object-Oriented Software. Addison-Wesley. ISBN 0-201-63361-2.*

Наукові результати, подані у цій доповіді, було отримано в рамках дослідницького проекту ДБ/КІБЕР з реєстраційним номером 0115U000446, 01.01.2015 – 31.12.2017, фінансово підтриманим Міністерством науки та освіти України.