



УДК 004.274

ІНФОРМАЦІЙНА ТЕХНОЛОГІЯ РЕКОНФІГУРОВНИХ СИСТЕМ НА БАЗІ МІКРОСХЕМ ПРОГРАМОВАНОЇ ЛОГІКИ ДЛЯ КЕРУЮЧОГО МОДУЛЮ БЕЗДРОТОВОЇ МЕРЕЖІ ДАТЧИКІВ

Крайник Я.М.

Чорноморський національний університет імені Петра Могили, Миколаїв, Україна

ORCID: <https://orcid.org/0000-0002-7924-3878>

Email: yaroslav.krainyk@chmnu.edu.ua

Copyright © 2021 by author and the journal “Automation of technological and business – processes”.

This work is licensed under the Creative Commons Attribution International License (CC BY).

<http://creativecommons.org/licenses/by/4.0>

DOI: 10.15673/atbp.v14i3.2349

Анотація. У представленій статті запропонована інформаційна технологія реконфігурації систем на базі мікросхем програмованої логіки для керуючого модулю бездротової мережі сенсорів. Розроблена інформаційна технологія дозволяє змінювати налаштування мережі сенсорів (додавати, видаляти, змінювати поточні налаштування) без необхідності внесення змін у систему на програмному рівні. Така технологія робить систему більш універсальною та дозволяє проводити обробку даних для різних бездротових технологій без необхідності перезавантаження системи. Запропонований підхід базується на тому, що модулі можуть бути замінені безпосередньо під час роботи у режимі реального часу. При цьому, накладаються обмеження щодо реалізації таких модулів з точки зору внутрішньої інфраструктури мікросхеми. Передбачається, що наявний контролер керує станами модулів обробки та інтерфейсного модулю та здатен завантажувати дані прошивки у відповідні виділені блоки схеми. Завантаження відбувається засобами стандартних шин взаємодії, які можуть бути реалізовані у програмованій частині. Контроль за доступними варіантами прошивки працюючого модулю також покладено на контролер. Таким чином забезпечується додатковий рівень абстракції щодо взаємодії з зовнішніми модулями та зменшується залежність від інтерфейсу підключення таких модулів. У результаті цього підвищується рівень гнучкості системи, а також забезпечується безперервна робота основної частини схеми. Це є основною відмінністю запропонованого рішення від відомих технологій. Таке рішення є особливо актуальним для мережі бездротових датчиків, оскільки системи цього типу можуть змінювати конфігурацію кінцевих а модулів, а це вимагає змін з точки зору усіх попередніх елементів обробки.

Abstract. The presented paper introduces an information technology for reconfiguration of the system based on programmable logic integrated circuits for wireless sensor network control module. The developed information technology allows changing configuration of sensor network (add, delete, change current settings, etc.) without additional activities on the software description level. This technology makes a system more universal and makes possible data processing for different wireless technologies that can be performed without reset of the whole system. The proposed approach is based on the fact that modules can be substituted immediately during their work in real time mode. Additionally, implementation of such modules has to follow constraints from the point of view of the internal infrastructure of the circuit. It is presumed that the controller module manages states of processing modules and interface module and ensure capability of loading firmware into specific regions of the programmable logic device. Control over the available firmware versions is also a responsibility of the controller. This way, additional level of abstraction is provided to communicate with external modules and to reduce dependency from the connection interface. As a result, flexibility of the system will be increased alongside with provision of the uninterrupted work of the whole system. This is the main difference and benefit of the devised solution in comparison with available technologies. The solution is specifically important for wireless sensor network as they can change configuration of the end-node modules and that process entails changes in all processing elements.

Ключові слова: реконфігурація, бездротова мережа сенсорів, ПЛІС, контролер реконфігурації.**Keywords:** reconfiguration, wireless sensor network, FPGA, reconfiguration controller.

Вступ. Бездротові мережі сенсорів (БМС) (англ. Wireless Sensor Network - WSN) наразі отримують надзвичайно широку поширеність, у тому числі в рамках розвитку концепції “Інтернету речей” (Internet-of-Things) [1]. Такі мережі



об'єднують у своєму складі велику кількість датчиків, які розподілені по контрольованій території, зв'язаних з бездротовими модулями. Дані модулі виконують функції зчитування та передачі даних з датчиків, відстеженням отримуваних показників, контроль споживання енергії з джерела живлення, а також відстеження стану даного джерела та ін. Пріоритет щодо виконання цих функцій залежить від цільового призначення такої мережі.

У таких мережах дані від вузлів передаються до пристроїв, які виконують функції, пов'язані з контролем усієї системи загалом. Функціональність та забезпеченість обчислювальними ресурсами для таких модулів повинна бути вища, ніж у вузлових модулів. Платформою для реалізації такого модулю може бути мікроконтролер або сучасний мікропроцесор, оптимізований для роботи від автономного джерела живлення.

Особливістю бездротових мереж сенсорів є те, що вони орієнтовані на те, що їх структура може постійно змінюватись (додавання/вилучення модулів, зміна технологій або протоколів передачі даних та ін.). У випадку використання готових рішень можливості зміни конфігурації мережі передбачені, проте вони є обмеженими, наприклад, тією самою технологією передачі даних. Універсальні рішення є більш складними, проте вони дозволяють забезпечити більшу гнучкість роботи мережі, а, відповідно, більшу кількість варіантів її використання.

Програмовані логічні інтегральні схеми (ПЛІС) мають можливість змінювати свою конфігурацію [2] (і, відповідно, функції, які вони виконують). Процес зміни конфігурації для ПЛІС має назву реконфігурації. Реконфігурація може бути статичною (повна зміна налаштувань) або динамічною (часткова) [3, 4]. При цьому, другий варіант передбачає заміну лише частину апаратного блоку на інший без необхідності перепрограмування мікросхеми. З точки зору реалізації керуючого модуля для бездротової мережі датчиків використання апаратної бази ПЛІС дозволяє отримати перевагу, по-перше, у продуктивності у випадку, коли необхідно забезпечувати обробку великої кількості даних з мінімальною затримкою; по-друге, у можливості динамічної зміни налаштувань керуючого модуля в залежності від мережі, яка розгортається. Саме тому, проблема реалізації реконфігурованого керуючого модулю на базі ПЛІС для БМС є актуальною.

Окрім реалізації самого реконфігурованого модулю, невід'ємним є питання створення системи, яка відповідальна за проведення реконфігурації у необхідний момент часу. Це може бути як зовнішня система (наприклад, окрема робоча станція), так і складовий компонент системи на кристалі (найчастіше це буде процесор), на якому розміщені і ресурси програмованої логіки. У другому випадку для зв'язку з зовнішніми системами необхідний також зв'язок з іншим обчислювальним центром, наявним у мережі для того, щоб проводити зовнішній контроль за станом системи.

Відносно питання живлення такого модуля на базі ПЛІС, то, звісно, таке рішення не є орієнтованим на те, щоб його можна використовувати у системі, яка може працювати довгий час від автономного джерела живлення. У випадку, якщо даний аспект є критичним для функціонування системи, то доцільно розглянути іншу апаратну базу для реалізації контролюючого модуля.

Зважаючи на вищезазначені факти, актуальним на даний момент часу є питання розробки інформаційної технології для керуючого модуля реконфігурованої системи на базі ПЛІС, яка забезпечує роботу у складі БМС.

Огляд наукових джерел

Функції керування у БМС покладені на один або декілька пристроїв, як у мережі ZigBee. Мережі на основі технології ZigBee здатні забезпечувати надзвичайно довгу роботу від джерел автономного живлення, проте їх недоліками є низька пропускна здатність каналу передачі даних та обмеженість обчислювальних ресурсів модулів. Саме тому вони не є придатними для реалізації БМС, які передають та оброблюють великі обсяги інформації. Модулі-контролери ZigBee є відносно простими пристроями з точки зору периферійних інтерфейсів, тому до них часто підключають додаткові модулі для обробки даних.

У той же час, технологія Wi-Fi надає можливість забезпечити значно більшу пропускну здатність у порівнянні з ZigBee. Це відображається на споживанні таких модулів, проте основною перевагою даної технології є її широка підтримка великою кількістю різних типів пристроїв (смартфони, планшети та ін.), що дозволяє забезпечити значно вищий рівень інтеграції. Wi-Fi-модулі ESP8266 [5] отримали надзвичайно широке поширення через поєднання низької ціни з високою функціональністю, яку вони здатні запропонувати. Окрім того, саме технологія Wi-Fi дозволяє отримати пристроям доступ до мережі Internet.

У свою чергу, технологія Bluetooth характеризується тим, що поєднує у собі деякі характеристики Wi-Fi та ZigBee. Зокрема, вона забезпечує менше споживання і його підтримку мають різні типи пристроїв. Структура складної мережі Bluetooth близька до такої у мережі ZigBee. Тим не менш, повністю можливості Bluetooth для побудови мереж використовуються рідко і обмежуються простими варіантами (точка-точка або зірка).

Усі наведені технології є важливими та можуть використовуватись для побудови БМС. Саме тому у даній роботі розглядається система, яка здатна поєднувати у своєму складі дані технології. При цьому, така система повинна забезпечувати гнучкість щодо підключення/видалення модулів з мережі.

Можливості реконфігурації мікросхем ПЛІС знайшли застосування у різних областях. Реконфігурована система на базі ПЛІС для БМС повинна забезпечувати швидкість зміни структури мережі та здатність обробки даних у такій мережі без необхідності внесення змін у керуючий модуль (перепрограмування, редагування великої кількості файлів конфігурації та ін.). Фактично, такий модуль забезпечує підлаштування під задану структуру мережі, яка може включати модулі, що підтримують вищезазначені технології. Проблеми побудови реконфігурованих систем на базі ПЛІС



присвячена велика кількість наукових досліджень, серед яких [6-10]. Вони акцентують увагу на перевагах динамічної реконфігурації для побудови систем, які можуть визначати необхідність переходу з одного стану в інший (заміна реконфігураційних блоків). Ця задача, зазвичай, покладається на окремий компонент - контролер конфігурації.

У роботі [9] проводиться дослідження розподілу ресурсів пам'яті в реконфігурованій системі. Загалом, варто відзначити, що пам'ять є центральним елементом у роботі реконфігурованих систем, оскільки від наявності та кількісних показників даного ресурсу залежить чи є можливою реалізація системи загалом. У вищезазначеній роботі розроблена універсальна система, яка може бути відображена на довільний алгоритм роботи. Особливістю систем БМС є те, що, зазвичай, обсяг повідомлень є обмеженим (це кілобайти та десятки кілобайт), тому така система висуває менші вимоги щодо забезпеченості мікросхеми ПЛІС ресурсом пам'яті (блокової або регістрової). Окрім того, контролер конфігурації повинен мати доступ до зовнішнього накопичувача для того, щоб мати можливість зберігати дані для їх подальшої обробки [11].

Незважаючи на описані переваги використання ПЛІС в якості базового модулю для реалізації контролера, необхідно враховувати, що така організація має свої обмеження. Зокрема, сам процес реконфігурації передбачає, що інформаційні блоки всередині системи мають бути виділені заздалегідь на етапі проектування. Після цього у дані блоки може завантажуватись конкретна конфігурація. Відповідно, така система передбачає наявність резервних блоків у випадку підключення нових модулів у систему.

Основна частина

Структура БМС, для якої пропонується розроблений керуючий модуль на базі ПЛІС підтримує наступні технології:

1. Wi-Fi.
2. ZigBee.
3. Bluetooth.

Використання даних технологій забезпечується підключенням до керуючого модуля окремих модулів, що підтримують взаємодію з використанням даних технологій. Загальна структура підключення модулів представлена на рис. 1.

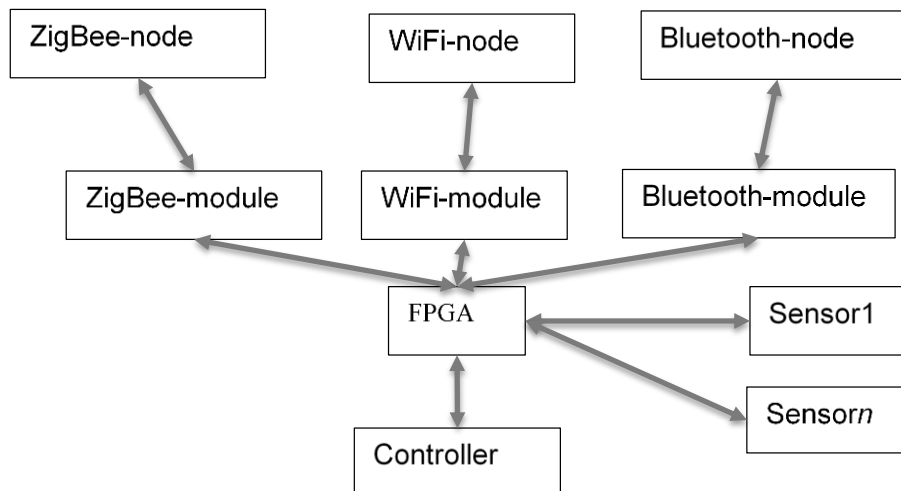


Рис. 1. – Структура підключення модулів у системі

Взаємодія між даними модулями та керуючим модулем відбувається шляхом цифрових інтерфейсів (UART, SPI, I2C та ін.). Для взаємодії необхідно, щоб інтерфейс підтримував як передачу, так прийом даних від модулів. Також на рисунку виділено той факт, що модуль ПЛІС може сам проводити обробку даних з датчиків

Процес реконфігурації характеризується тим, що відбувається завантаження/заміна одного функціонального блоку іншим усередині ПЛІС у розташуванні, яке виділено спеціально для розміщення модулів. При цьому, важливою умовою для забезпечення цього є те, що кількість задіяних виводів блоків (як усередині, так і зовні, тобто, фізичні піни) має бути однаковою для всіх таких модулів. Тому при реалізації модулів обирається модуль з найбільшою кількістю пінів і у опис інших модулів необхідно додати необхідну кількість виводів, які фактично не будуть задіяні, проте дозволять забезпечити сумісність між ними. При цьому, варто пам'ятати, що на більш низькому рівні умова сумісності є ще більш жорсткою: типи виводів, що використовуються у модулях мають бути однакові (їх кількість і розташування у кінцевій схемі). У іншому випадку, засоби генерації кінцевих файлів вкажуть на наявність помилки і проект не можна буде синтезувати.

При цьому, варто відзначити, що необхідно розрізняти блок, що забезпечує реалізацію інтерфейсу, і блок, який відповідає за обробку інформації відповідно до заданого протоколу. Відповідно, для взаємодії з окремим модулем



будуть використовуватись два реконфігуровані блоки. З приводу першого типу блоків, то вони мають обов'язково містити виводи, які будуть відображатись безпосередньо на фізичні виводи ПЛІС.

Відносно першого типу блоків, то проблему забезпечення сумісності між блоками, які реалізують різні інтерфейси можна розглянути на прикладі UART (Universal Asynchronous Receiver-Transmitter) та SPI (Serial Peripheral Interface). Перший використовує для обміну даними лише 2 лінії, в той час як другий - 4 (традиційна реалізація SPI). Відповідно, при реалізації першого блоку мають додаватись ще два додаткові виводи, які не будуть використовуватись безпосередньо у роботі модуля. Даний приклад показує порівняння послідовних інтерфейсів. Ще більшою буде різниця у кількості доданих виводів для випадку, коли використовується паралельний інтерфейс. Використання кількох паралельних інтерфейсів може суттєво обмежити кількість блоків, які можна підключити з використанням заданої мікросхеми ПЛІС.

Щодо пристрою верхнього рівня - контролеру реконфігурації, - то він є відповідальним за те, щоб ініціювати зміну реконфігурованих блоків під час роботи системи. Даний процес починається як результат команди від користувача або як самостійний процес, коли система працює в автономному режимі. Окрім цього, на даний модуль також може бути покладена відповідальність за обробку даних на найвищому рівні у системі (він є кінцевим споживачем даних і вирішує, які операції необхідно проводити далі).

Розглянемо загальну структуру реконфігурованого модулю, представлену на рис. 2.

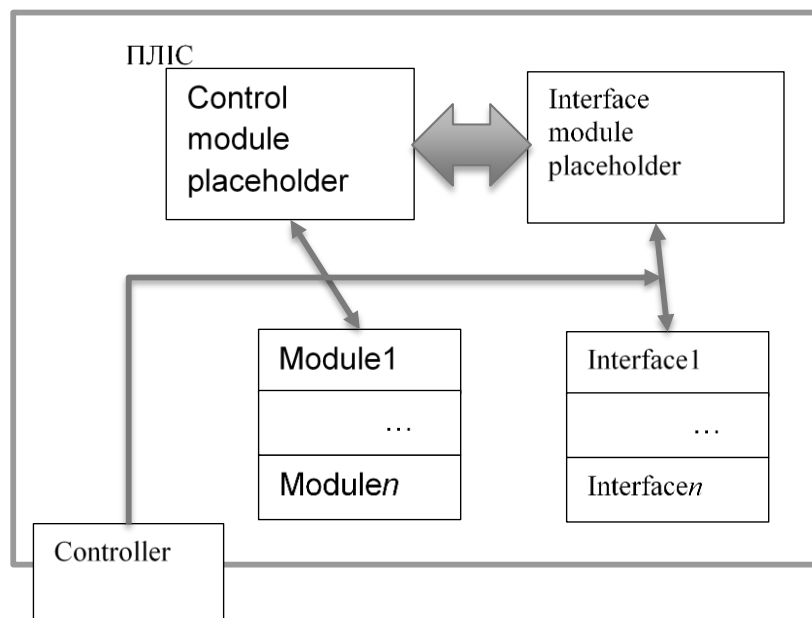


Рис. 2. – Загальна схема реконфігурованого блоку на базі ПЛІС

Як видно з рисунку, контролер реконфігурації позначений на межі ПЛІС, оскільки він може бути реалізований як засобами самої мікросхеми, так і зовнішнім пристроєм. Виділені окремо регіони для модулів, що реалізують взаємодію з інтерфейсами, і для модулів, які реалізують контроль і обробку даних, отриманих від інтерфейсу. В залежності від сигналів, які подаються контролером, у відповідний регіон завантажується необхідний модуль. На рисунку продемонстрований приклад для однієї пари модулів інтерфейс-контролер, в той час, як таких пар може бути розміщено одразу декілька. Усі пари повинні мати з'єднання з регіоном варіантів модулів. Загалом, саме це з'єднання є ключовим у ході процесу реконфігурації, оскільки саме на його основі відбувається завантаження бітових даних опису модулю у необхідний для цього регіон. Зв'язок між парою модулів забезпечується з використанням окремою шини. Причому, структура такої шини є фіксованою і змінюватись не може, що необхідно враховувати при проектуванні. Найчастіше для таких комунікацій будуть використовуватись поширені інтерфейси такі, як AXI або Avalon. Звичайно, такий підхід робить систему менш гнучкою, проте він обумовлений особливостями мікросхем ПЛІС і даний момент розвитку техніки альтернативи такому підходу будуть мати вищу складність та меншу модульність у кінцевій структурі, що зробить рішення менш гнучким.

Варто зауважити, що можливою є також і інша схема, яка передбачає те, що реконфігураційні модулі (файли для часткової реконфігурації) розміщуються не всередині ПЛІС, а у зовнішньому сховищі, до якого має доступ контролер конфігурації. Такий варіант схеми дозволяє значно розширити можливості щодо кількості конфігурацій, які можна завантажити у реконфігураційний регіон програмованої логіки.



Використання ресурсів виводів мікросхеми ПЛІС є важливим, тому відповідно, до запропонованої схеми розміщення модулів використання входів/виходів парами модулів буде визначатись як

$$total_pins = \sum_{i=1}^Z pins_inout_i, \quad (1)$$

де Z - загальна кількість пар, що розміщують у мікросхемі ПЛІС. Слід зауважити, що відображення пінів у даному випадку є логічним, а фізичні виводи мають бути сконфігуровані перед початком роботи без можливості зміни їх налаштувань. Дане значення повинне задовольняти умові і становити меншу кількість, ніж загальна кількість пінів у мікросхемі:

$$total_pins < pins_{FPGA}. \quad (2)$$

Кількість внутрішніх з'єднань, задіяних у з'єднанні між блоками всередині ПЛІС, буде дорівнювати

$$inside_inout = \sum_{i=1}^Z inside_inout_i. \quad (3)$$

Тоді, загальна кількість з'єднань визначиться як

$$total_inout = total_pins + inside_inout. \quad (4)$$

Основну роль у процесі реконфігурації відіграє контролер реконфігурації, який відповідальний за виконання даного процесу.

Модулі обробки даних можуть перебувати у чотирьох станах:

1. Стан очікування надходження даних від інтерфейсу.
2. Стан очікування надходження даних від контролеру.
3. Стан обробки даних (активний стан).
4. Пасивний стан (не виконується жодних операцій і відсутні сигнали про те, що найближчим часом розпочнеться нова операція або зі сторони інтерфейсу, або зі сторони контролеру).

Особливістю БМС є те, що інформація від вузлових модулів може надходити у будь-який момент часу. Це означає, що без використання технік по забезпеченню коректного функціонування взаємодії модулів можливими є різного роду колізії. Наприклад, після отримання сигналу від вузлового модуля модуль реконфігурації розпочав процес реконфігурації модулю, відповідального за обробку отриманих даних (або даних, які плануються до отримання). У результаті, дані не були оброблені і, відповідно, є втраченими. У найгіршому випадку, це також блокує роботу вузлового модуля, якщо він повинен очікувати на відповідь від контролеру.

У свою чергу, у БМС наявна велика кількість модулів, відповідно, можливою є ситуація, при якій повідомлення будуть надходити одразу з кількох блоків. Саме тому контролер конфігурації повинен забезпечувати паралельну обробку для кількох модулів за необхідності та, при цьому, гарантувати уникнення конфліктів при використанні спільних ресурсів. Це забезпечується на програмному рівні з використанням основних примітивів синхронізації.

Зважаючи на описані вище особливості БМС, пропонується наступна інформаційна технологія проведення реконфігурації на базі ПЛІС. Проведення реконфігурації може бути ініційоване

1. Діями адміністратора системи шляхом прямої подачі команди.
2. Процес реконфігурації може бути визначений на програмному рівні як реакція на певну подію, що відбулась у системі, відповідно, ці дії відбуваються без безпосередньої участі користувача.

Команди, які надійшли до контролеру реконфігурації, ставляться у чергу. Кожній команді відповідає лише один цільовий модуль, налаштування якого змінюються. Команди оброблюються у тому порядку, у якому вони надійшли до контролеру. У заданий момент часу у стані обробки може перебувати не більше однієї команди, відповідно, паралельна конфігурація кількох модулів неможлива. Це надає можливість більшого контролю відносно загального стану системи і забезпечує детермінованість її поведінки і спрощує узгодження взаємодії між модулями. Як тільки контролер починає обробку команди, він перевіряє, чи не знаходиться модуль у одному з трьох робочих станів, які унеможливають початок конфігурації. За необхідності відбувається перевірка до моменту переходу модуля в режим очікування. Після того, як модуль перейшов у режим очікування, він стає доступним для завантаження нової конфігурації зі сховища конфігурацій. На рис. 3. представлена схема описаної інформаційної технології для проведення реконфігурації.

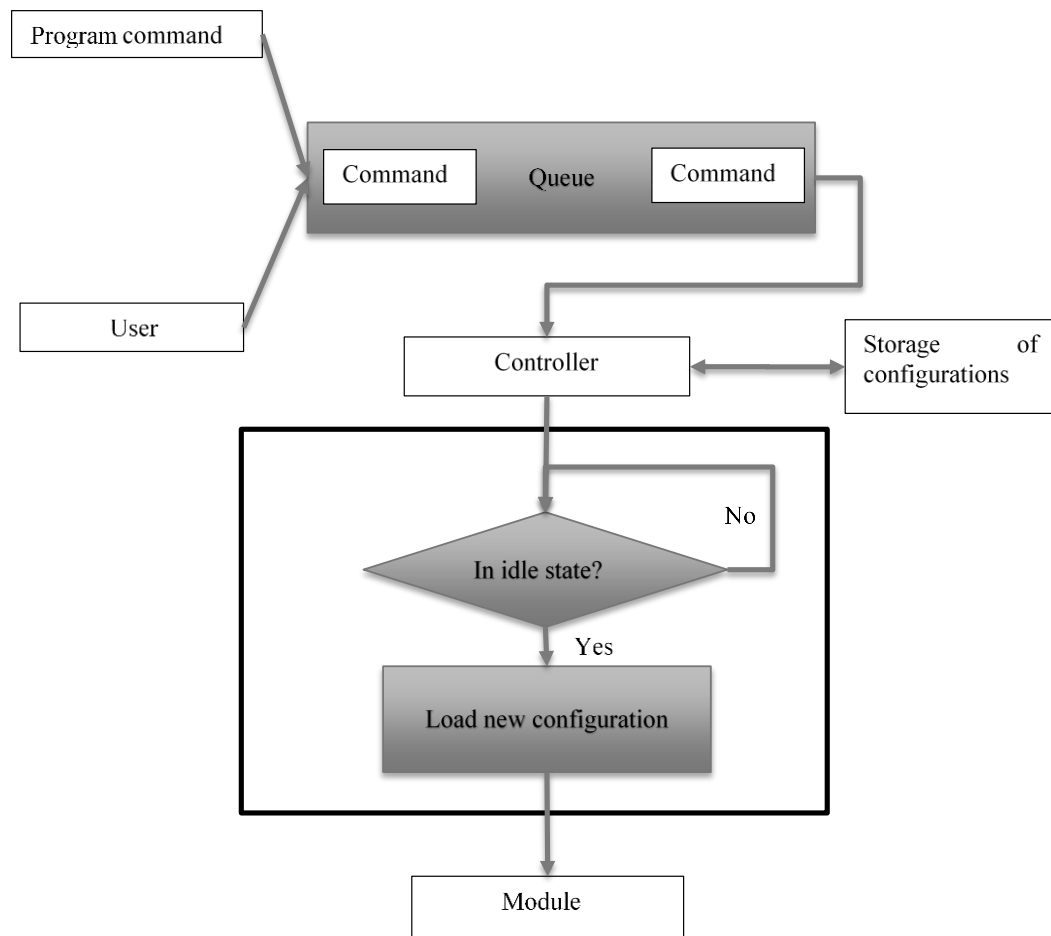


Рис. 3. – Представлення інформаційної технології проведення реконфігурації

Запропонована схема дозволяє спростити програмний рівень реалізації процесу реконфігурації, оскільки в один момент часу можлива конфігурація лише одного модулю.

З точки зору подальшого розвитку напрямку реконфігурованих систем для БМС, то важливою для цього має стати подія придбання виробником процесорів Intel фірми-виробника програмовної логіки. Це дає можливість казати про те, що контролери реконфігурації можуть бути реалізовані з використанням даних процесорів, які будуть містити ресурси програмованої логіки на кристалі. Крім того, оскільки продуктова лінійка Intel постійно розширюється у сторону рішень для вбудованих систем, то не виключним є те, що програмовані логічні ресурси будуть міститись і у мікросхемах з меншим споживання енергії, що важливо при реалізації БМС.

Висновки

У даній роботі запропоновано інформаційну технологію роботи керуючого модулю на базі ПЛІС реконфігурованої системи для БМС. Визначена сфера використання такої системи, вказані переваги та недоліки для такої мережевої архітектури. Основною перевагою запропонованої інформаційної технології є те, що вона враховує особливості роботи мікросхем ПЛІС, а також самої мережі сенсорів.

Список використаних джерел

- [1] Majid M., Habib S., Javed A. R., Rizwan M., Srivastava, G., Gadekallu, T. R., Lin, J. C. W. Applications of wireless sensor networks and internet of things frameworks in the industry revolution 4.0: A systematic literature review. *Sensors*. Vol. 2 (6). <https://doi.org/10.3390/s22062087>
- [2] Albouchi A., Ouni B., Abdellatif M. Operating Systems for Reconfigurable Systems-on-Chip. *International conference on Sciences and Techniques of Automatic control & computer engineering (STA'2010): Tunis, Tunisia, 2010* P. 1-12.
- [3] Carvalho E., Calazans N., Moraes F.G., Mesquita D.G. Reconfiguration Control for Dynamically Reconfigurable Systems. *DCIS: Vol. 4, 2004*. P. 405-410.
- [4] Dimassi S., Abdelali A.B., Mrabet A., Krifa M. N., Abdellatif M. A Modeling Tool for Dynamically Reconfigurable Systems. *International Review on Computers and Software (IRECOS)*. 2014. Vol. 9(4). P. 600-608.



- [5] Krainyk Y. M., Razzhyvin A., Bondarenko O., Simakova I. Internet-of-Things Device Set Configuration for Connection to Wireless Local Area Network. *Computer Modeling and Intelligent Systems-2019: Zaporizhzhia*, 2019. P. 885-896. <https://doi.org/10.32782/cmis/2353-70>
- [6] Najjar W.A., Ienne P. Reconfigurable Computing. *IEEE Micro*. 2014. Vol. 34(1). P. 4-6. <https://doi.org/10.1109/MM.2014.25>
- [7] Eto E. Difference-Based Partial Reconfiguration. *Application Note: Virtex Architectures*: веб-сайт. URL: <https://docs.xilinx.com/v/u/en-US/xapp290> (дата звернення: 12.09.2022).
- [8] Partial Reconfiguration. Vivado Design Suite User Guide: веб-сайт. URL: <https://docs.xilinx.com/v/u/2015.4-English/ug909-vivado-partial-reconfiguration> (дата звернення: 12.09.2022).
- [9] Zhang H., Kochte M., Imhof M.E., Bauer L., Wunderlich H.-J., Henkel J. GUARD: GUAranteed Reliability in Dynamically Reconfigurable Systems. *ACM/IEEE Design Automation Conference (DAC)*: San Francisco, CA, USA, 2014. P. 1-6. <https://doi.org/10.1145/2593069.2593146>
- [10] Tessier R., Pocek K., Dehon A. Reconfigurable Computing Architectures. *Proceedings of the IEEE*. Vol. 103 (3). P. 332-354. <https://doi.org/10.1109/JPROC.2014.2386883>
- [11] Krainyk Y., Darnapuk Y. Configurable Description of FPGA-based Control System for Sensor Processing. XIth International Scientific and Practical Conference on Electronics and Information Technologies (ELIT): Lviv, 2019. <https://doi.org/10.1109/ELIT.2019.8892313>

References

- [1] M. Majid et al., "Applications of Wireless Sensor Networks and Internet of Things Frameworks in the Industry Revolution 4.0: A Systematic Literature Review", *Sensors*, vol. 22, no. 6, p. 2087, 2022. Available: [10.3390/s22062087](https://doi.org/10.3390/s22062087). <https://doi.org/10.3390/s22062087>
- [2] A. Albouchi et al., "Operating Systems for Reconfigurable Systems-on-Chip". International conference on Sciences and Techniques of Automatic control & computer engineering (STA'2010): Tunis, Tunisia, 2010 P. 1-12.
- [3] E. Carvalho et al. "Reconfiguration control for dynamically reconfigurable systems." *DCIS*. Vol. 4. 2004.
- [4] S. Dimassi et al. "A Modeling Tool for Dynamically Reconfigurable Systems." *International Review on Computers and Software* 9.4 (2014): 600-608.
- [5] Y. Krainyk, A. Razzhyvin, O. Bondarenko, and I. Simakova, "Internet-of-things device set configuration for connection to wireless local area network," in *CEUR Workshop Proceedings*, 2019, vol. 2353, pp. 885–896. <https://doi.org/10.32782/cmis/2353-70>
- [6] Najjar, Walid A., and Paolo Ienne. "Reconfigurable computing." *Ieee Micro* 34.1 (2014): 4-6. <https://doi.org/10.1109/MM.2014.25>
- [7] Eto E. Difference-Based Partial Reconfiguration. *Application Note: Virtex Architectures*: веб-сайт. URL: <https://docs.xilinx.com/v/u/en-US/xapp290> (access date: 12.09.2022).
- [8] Partial Reconfiguration. Vivado Design Suite User Guide: веб-сайт. URL: <https://docs.xilinx.com/v/u/2015.4-English/ug909-vivado-partial-reconfiguration> (access date: 12.09.2022).
- [9] H. Zhanget al. "GUARD: Guaranteed reliability in dynamically reconfigurable systems." 2014 51st ACM/EDAC/IEEE Design Automation Conference (DAC). IEEE, 2014. <https://doi.org/10.1145/2593069.2593146>
- [10] R. Tessier et al., "Reconfigurable computing architectures." *Proceedings of the IEEE* 103.3 (2015): 332-354. <https://doi.org/10.1109/JPROC.2014.2386883>
- [11] Y. Krainyk and Y. Darnapuk, "Configurable Description of FPGA-based Control System for Sensor Processing," in 2019 11th International Scientific and Practical Conference on Electronics and Information Technologies, ELIT 2019 - Proceedings, 2019, pp. 210–213. <https://doi.org/10.1109/ELIT.2019.8892313>

Отримана в редакції 24.08.2022. Прийнята до друку 09.09.2022. Received 24 August 2022. Approved 09 September 2022. Available in Internet 30 September 2022.