Харківський національний університет імені В. Н. Каразіна
Міністерство освіти і науки України

Кваліфікаційна наукова праця на правах рукопису

КОВАЛЬЧУК ДМИТРО МИКОЛАЙОВИЧ

УДК 681.142

ДИСЕРТАЦІЯ

МОДЕЛІ ТА МЕТОДИ ШВИДКОЇ ОБРОБКИ ДАНИХ НА ОСНОВІ ЗАСТОСУВАННЯ СИСТЕМИ ЗАЛИШКОВИХ КЛАСІВ

Спеціальність 122 – Комп’ютерні науки
(Галузь знань 12 – Інформаційні технології).

Подається на здобуття ступеня доктора філософії.

Дисертація містить результати власних досліджень. Використання ідей, результатів і текстів інших авторів мають посилання на відповідне джерело _____Д. М. Ковальчук

Науковий керівник: Краснобаєв Віктор Анатолійович, доктор технічних наук, професор

Харків – 2024

АНОТАЦІЯ

Ковальчук Д.М. Моделі та методи швидкої обробки даних на основі застосування системи залишкових класів. – Кваліфікаційна наукова праця на правах рукопису.

Дисертація на здобуття ступеня доктора філософії за спеціальністю 122 – Комп'ютерні науки (Галузь знань 12 – Інформаційні технології). – Харківський національний університет імені В. Н. Каразіна Міністерства освіти і науки України, Харків, 2024.

Дисертація присвячена підвищенню швидкості обробки інформації програмно-апаратними системами і комплексами з елементами штучного інтелекту за рахунок використання моделей та методів швидкої обробки даних на основі застосування системи залишкових класів (СЗК).

В першому розділі аналізуються проблеми побудови програмно-апаратних систем і комплексів з елементами штучного інтелекту. Проведений аналіз можливостей програмно-апаратних систем і комплексів з елементами штучного інтелекту свідчить про те, що вони не задовольняють збільшеним вимогам до швидкості обробки інформації, що обумовлює актуальність дослідження нових моделей і методів.

Проаналізовано сучасний стан та напрями підвищення швидкодії програмно-апаратних систем і комплексів з елементами штучного інтелекту, за рахунок застосування спеціальних технологічних та архітектурних рішень, а також математичних методів для їх застосування в штучному інтелекті (ШІ). Відмічено, що застосування паралельної обробки даних на основі СЗК дозволяє значно підвищити швидкодію операцій обробки даних.

За результатами проведеного аналізу теоретичних основ СЗК визначено основні її переваги над позиційними системами числення (незалежність залишків, що дає можливість розпаралелення процесу обчислень; рівноправність залишків, що дає можливість підвищити відмовостійкість програмно-апаратних систем і комплексів з елементами штучного інтелекту та

малорозрядність залишків, що дає можливість підвищити швидкодію програмно-апаратних систем і комплексів з елементами штучного інтелекту) та її недоліки (труднощі при виконанні операції порівняння та ділення чисел, визначення переповнення допустимого діапазону), обґрунтовано необхідність використання СЗК в операційних пристроях програмно-апаратних систем і комплексів з елементами штучного інтелекту.

Формулюються задачі дисертаційного дослідження: удосконаленя методу додавання та віднімання залишків чисел по модулю СЗК; удосконалення методу табличної реалізації множення двох залишків чисел за рахунок можливості виконання операції в комплексній області; удосконалення математичної моделі процесу піднесення залишків цілих чисел до довільного ступеня натурального в СЗК; практичне підтвердження працездатності та вірогідності розроблених моделей і методів. Які будуть вирішуватись в наступних розділах дисертаційної роботи.

У другому розділі дістав подальший розвиток метод додавання і віднімання залишків чисел по модулю СЗК, який враховує конструкції суматорів по модулю з величиною корекції $\Delta Q_R > 0$.

Розроблена HDL-модель суматора по модулю $m_i = 17$ на мові Verylog. Розроблена суматора по модулю $m_i = 17$ в середовищі Quartus II.

Розглянуті приклади та результати моделювання реалізації методу модульного додавання для різних значень x_i і y_i по модулю m_i СЗК, підтверджують практичну реалізованість запропонованого методу.

Розроблена HDL-модель виконання операції віднімання на суматорі по модулю $m_i = 17$ на мові Verylog та структурна схема в середовищі Quartus II.

Розглянуті приклади та результати моделювання реалізації операції віднімання $(x_i - y_i) \bmod m_i$ по модулю m_i СЗК для різних залишків x_i і y_i, підтверджують практичну реалізованість запропонованого методу.

У третьому розділі вдосконалено метод табличної реалізації множення двох залишків чисел в системі залишкових класів за рахунок можливості

виконання операції в комплексній області, на основі використання першої фундаментальної теореми Гауса про ізоморфізм між множиною дійсних і комплексних чисел, що підвищує швидкодію реалізації операції множення в системі залишкових класів.

Вдосконалено математичну модель процесу піднесення цілих чисел до довільного степеня натурального числа в СЗК за рахунок можливості виконання операції піднесення цілих чисел до степеня, як у додатному, так і в від'ємному числових діапазонах, що підвищує швидкодію реалізації операції піднесення цілих чисел до степеня в системі залишкових класів.

Результати комп'ютерного моделювання середовищі Microsoft Visual Studio 2015 підтверджують практичну реалізованість запропонованого методу.

Четвертий розділ присвячено розробці операційного пристрою програмно-апаратних систем і комплексів з елементами штучного інтелекту, що функціонують в системі залишкових класів та прведенню аналізу швидкодії обробки даних в позиційній системі числення та системі залишкових класів.

Розроблено операційний пристрій програмно-апаратних систем і комплексів з елементами штучного інтелекту, що функціонує в системі залишкових класів.

В основу винаходу операційного пристрою програмно-апаратних систем і комплексів з елементами штучного інтелекту, що функціонують в системі залишкових класів, поставлено мету: розширити функціональні можливості наявного вже операційного пристрою. Розширення можливостей операційного пристрою досягається завдяки тому, що, крім виконання операції додавання залишків чисел x_i і y_i по модулю m_i СЗК, пристрій ще додатково виконує операцію модульного віднімання $(x_i - y_i) \bmod m_i$ в СЗК.

Пристрій функціонує у двох режимах роботи. В першому режимі знаходиться результат операції додавання залишків чисел $(x_i + y_i) \bmod m_i$ по модулю m_i СЗК. А у другому режимі знаходиться результат операції віднімання $(x_i - y_i) \bmod m_i$ по модулю m_i СЗК.

Наведено приклади виконання операцій додавання $(x_i + y_i) \bmod m_i$ і

віднімання $(x_i - y_i) \bmod m_i$ залишків чисел по модулю СЗК, що підтверджує практичну можливість використання запропонованого винаходу операційного пристрою програмно-апаратних систем і комплексів з елементами штучного інтелекту в СЗК.

Проведено розрахунок та порівняльний аналіз швидкодії обробки даних програмно-апаратних систем і комплексів з елементами штучного інтелекту у СЗК для математичної моделі штучного нейрону.

Розрахунки та порівняльна оцінка швидкодії, проведені в дисертаційній роботі, показали, що зі збільшенням розрядності сітки програмно-апаратних систем і комплексів з елементами штучного інтелекту ефективність застосування непозиційної системи числення в СЗК значно зростає.

Сукупність отриманих у дисертації нових наукових результатів, позитивна оцінка їхньої достовірності, наукової та практичної значущості дають змогу вважати сформульовану наукову задачу підвищення швидкості обробки інформації програмно-апаратними системами і комплексами з елементами штучного інтелекту за рахунок використання моделей та методів швидкої обробки даних на основі застосування системи залишкових класів, – розв'язаною, а поставлену мету – досягнутою.

Ключові слова: *комп'ютерні системи, комп'ютерні компоненти, швидкодія обробки даних, математична модель, моделі та методи швидкої обробки даних, система залишкових класів, китайська теорема про залишки, двійковий однорозрядний суматор, суматор по модулю, табличний принцип обробки даних, кодування, код табличного множення, паралельний принцип обробки даних, цифрові пристрої, мова опису апаратних засобів, моделювання, інформаційна достовірність, моделі реального часу, дискретна система, штучний інтелект, програмно-апаратні системами і комплекси з елементами штучного інтелекту.*

ABSTRACT

Kovalchuk D.M. Models and methods for fast data processing based on the use of a residual class system. – Qualification scholarly paper: a manuscript.

The dissertation submitted for obtaining the Doctor of Philosophy degree in Information Technology: Speciality 122 – Computer science. V. N Karazin Kharkiv National University, Ministry of Education and Science of Ukraine, Kharkiv, 2023.

The dissertation is devoted to increasing the speed of information processing by software and hardware systems and complexes with elements of artificial intelligence due to the use of models and methods of fast data processing based on the application of the residual class system (RCS).

The first chapter analyzes the problems of constructing hardware and software systems and complexes with elements of artificial intelligence. The analysis of the capabilities of software and hardware systems and complexes with elements of artificial intelligence indicates that they do not meet the increased requirements for information processing speed, which makes it urgent to study new models and methods.

The current state and directions for increasing the performance of software and hardware systems and complexes with elements of artificial intelligence are analyzed through the use of special technological and architectural solutions, as well as mathematical methods for their application in artificial intelligence (AI). It is noted that the use of parallel data processing based on RCS can significantly increase the speed of data processing operations.

Based on the results of the analysis of the theoretical foundations of RCS, its main advantages over positional number systems were determined (independence of remainders, which makes it possible to parallelize the calculation process; equality of remainders, which makes it possible to increase the fault tolerance of software and hardware systems and complexes with elements of artificial intelligence and low-bitness of remainders, which makes it possible to the ability to increase the performance of software and hardware systems and complexes with elements of

artificial intelligence) and its disadvantages (difficulties in performing the operation of comparing and dividing numbers, determining the overflow of the permissible range), the need to use RCS in the operating devices of software and hardware systems and complexes with elements of artificial intelligence is substantiated.

The objectives of the dissertation research are formulated: improvement of the method of adding and subtracting remainders of numbers modulo RCS; improvement of the method of tabular implementation of multiplication of two remainders of numbers due to the ability to perform the operation in the complex domain; improvement of the mathematical model of the process of raising remainders of integers to an arbitrary degree of natural in RCS; practical confirmation of the performance and likelihood of the developed models and methods. Which will be addressed in the following sections of the dissertation.

In the second chapter, the method of adding and subtracting the remainders of numbers modulo RCS was further developed, taking into account the design of modulo adders with a correction value $\Delta Q_R > 0$.

An HDL model of a modulo $m_i = 17$ adder in Verilog has been developed. An adder of a modulo $m_i = 17$ has been developed in the Quartus II environment.

The considered examples and simulation results of the implementation of the modular addition method for various values of x_i and y_i modulo m_i RCS confirm the practical implementation of the proposed method.

An HDL model for performing a subtraction operation on a modulo adder $m_i = 17$ in the Verylog language and a block diagram in the Quartus II environment have been developed.

The considered examples and simulation results of the implementation of the subtraction operation $(x_i - y_i) \bmod m_i$ modulo m_i RCS for various residues x_i and y_i confirm the practical implementation of the proposed method.

In the third chapter, the method of tabular implementation of multiplication of two remainders of numbers in a RCS is improved due to the possibility of performing the operation in the complex domain, based on the use of Gauss's first

fundamental theorem on isomorphism between the set of real and complex numbers, which increases the speed of implementation of the multiplication operation in the RCS.

The mathematical model of the process of raising integers to an arbitrary power of a natural number in the RCS has been improved due to the possibility of performing the operation of raising integers to a power in both positive and negative numerical ranges, which increases the performance of the implementation of the operation of raising integers to a power in the RCS.

The results of computer modeling in the Microsoft Visual Studio 2015 environment confirm the practical implementation of the proposed method.

The fourth chapter is devoted to the development of an operating device for software and hardware systems and complexes with elements of artificial intelligence that operate of the residual class system and analysis of the speed of data processing in the positional number system and the residual class system.

An operating device for software and hardware systems and complexes with elements of artificial intelligence operating in a RCS has been developed.

The basis for the invention of an operating device for software and hardware systems and complexes with elements of artificial intelligence operating in a RCS is the goal of expanding the functionality of an existing operating device. Expanding the capabilities of the operating device is achieved due to the fact that, in addition to performing the operation of adding the remainders of numbers x_i and y_i modulo m_i of the RCS, the device additionally performs the operation of modular subtraction $(x_i - y_i) \bmod m_i$ in the RCS.

The device operates in two operating modes. In the first mode, the result of the operation of adding the remainders of the numbers $(x_i + y_i) \bmod m_i$ modulo m_i RCS is found. And in the second mode, the result of the subtraction operation $(x_i + y_i) \bmod m_i$ modulo m_i RCS is found.

Examples are given of performing the operations of addition $(x_i + y_i) \bmod m_i$ and subtraction $(x_i - y_i) \bmod m_i$ of the remainders of numbers modulo RCS, which confirms the practical possibility of using the proposed invention of the operating

device of software and hardware systems and complexes with elements of artificial intelligence in RCS.

A calculation and comparative analysis of the data processing speed of software and hardware systems and complexes with elements of artificial intelligence in the RCS for the mathematical model of an artificial neuron was carried out.

Calculations and comparative evaluation of performance carried out in the dissertation work showed that with an increase in the grid capacity of software and hardware systems and complexes with elements of artificial intelligence, the efficiency of using a non-positional number system in RCS increases significantly.

The totality of new scientific results obtained in the dissertation, a positive assessment of their reliability, scientific and practical significance allow us to consider the formulated scientific task of increasing the speed of information processing by software and hardware systems and complexes with elements of artificial intelligence through the use of models and methods of fast data processing based on the use of a RCS - solved, and the goal set - achieved.

Keywords: *computer systems, computer components, speed of data processing, mathematical model, models and methods of fast data processing, residual class system, residue number system, Chinese Remainder Theorem, binary single-digit adder, modulo adder, tabular principle of data processing, coding, tabular multiplication code, parallel principle of data processing, digital devices, hardware description language (HDL), modeling, simulation, information reliability, real time models, discrete system, artificial intelligence, software and hardware systems and complexes with elements of artificial intelligence.*

СПИСОК ПУБЛІКАЦІЙ ЗДОБУВАЧА ЗА ТЕМОЮ ДИСЕРТАЦІЇ

Наукові праці, в яких опубліковані основні результати дисертації :

у фахових виданнях України:

1. Krasnobayev V., Koshman S., **Kovalchuk D.** The data diagnostic method of in the system of residue classes. Advanced Information Systems. 2021. Vol. 5(1). P. 123–128. DOI:https://doi.org/10.20998/2522-9052.2021.1.18.

(Особистий внесок: розробка методу діагностики даних, які представлені в СЗК, а також написання частини тексту та його переклад).

2. Krasnobayev V., Koshman, S., **Kovalchuk D.** Synthesis of structure of the adder by module. Control, Navigation and Communication Systems. Academic Journal. 2021. Vol. 1(63). P. 96-99. DOI:https://doi.org/10.26906/SUNZ.2021.1.096.

(Особистий внесок: розробка алгоритму синтезу суматора за довільним модулем СЗК, участь в обговоренні отриманих результатів а також написання частини тексту та його переклад).

3. Krasnobayev V., Koshman S., **Kovalchuk D.** The concept of performing the addition operation in the system of residual classes. Advanced Information Systems. 2022. Vol. 6(1). P. 43–47. DOI:https://doi.org/10.20998/2522-9052.2022.1.07.

(Особистий внесок: розробка методу обчислення суми залишків чисел за довільним модулем, підготовка прикладів, що наочно демонструють ефективність запропонованого методу, участь в обговоренні отриманих результатів, а також написання частини тексту та його переклад).

4. Krasnobayev V., Koshman S., **Kovalchuk D.** The concept of using the number system in the residual classes for building artificial intelligence system. Control, Navigation and Communication Systems. Academic Journal. 2022. Vol. 1(67). P. 65-70. DOI:https://doi.org/https://doi.org/10.26906/SUNZ.2022.1.065.

(Особистий внесок: аналіз можливості застосування непозиційної системи числення у залишкових класах для підвищення швидкодії програмно-апаратних систем і комплексів з елементами штучного інтелекту, а також написання частини тексту та його переклад).

5. Krasnobayev V., Koshman S., Nikolsky S., **Kovalchuk D.** Mathematical model of computer system reliability in residual classes. Advanced Information Systems. 2022. Vol. 6(4). P. 19–24. DOI:doi: 10.20998/2522-9052.2022.4.03.

(Особистий внесок: розробка математичної моделі надійності комп'ютерної системи, що функціонує у СЗК, участь в обговоренні отриманих результатів, а також написання частини тексту та його переклад).

6. Krasnobayev V. A., Yanko A. S., **Kovalchuk D. M.** Mathematical Model of the Process of Raising Integers to an Arbitrary Power of a Natural Number in the System of Residual Classes. Theoretical and Applied Cybersecurity. 2023. Vol. 5 (2), P. 5-14. DOI: https://doi.org/10.20535/tacs.2664-29132023.2.278891.

(Особистий внесок: розробка математичної моделі процесу піднесення цілих чисел до довільного степеню натурального числа в СЗК, участь в обговоренні отриманих результатів, а також написання частини тексту та його переклад).

Наукові праці, в яких опубліковані основні результати дисертації, що входять до наукометричної бази **Web of Science і Scopus:**

7. Krasnobayev V. A., Yanko A. S., **Kovalchuk D. M.** Methods for tabular implementation of arithmetic operations of the residues of two numbers represented in the system of residual classes. Radio Electronics, Computer Science, Control. 2022. № 4, P. 18-28. DOI:https://doi.org/10.15588/1607-3274-2022-4-2. (Web of Science)

(Особистий внесок: розробка табличного методу реалізації цілочисельних арифметичних модульних операцій додавання, віднімання та множення, участь в обговоренні отриманих результатів, а також написання частини тексту та його переклад).

8. Koshman S., Krasnobayev V., Nikolsky S., **Kovalchuk D.** The structure of the computer system in the residual classes. Advanced Information Systems. 2023. Vol. 7(2). P. 41–48. DOI:https://doi: 10.20998/2522-9052.2023.2.06. (Scopus)

(Особистий внесок: постановка та вирішення зворотної задачі

оптимального резервування в СЗК на основі використання методу динамічного програмування, а також написання частини тексту та його переклад).

Наукові праці, які засвідчують апробацію матеріалів дисертації:

9. Krasnobayev V., Koshman S., **Kovalchuk D.** Diagnosing data in a non-positional number system of residual classes. Сучасні напрями розвитку інформаційно-комунікаційних технологій та засобів управління: тези доповідей десятої Міжнародної науково-технічної конференції, 8 – 9 квітня 2021 р., Харків, 2021. Т. 2. С. 25.

10. Krasnobayev V., Koshman S., **Kovalchuk D.** Development of the adder structure by modulo of the system of residual classes. Сучасні напрями розвитку інформаційно-комунікаційних технологій та засобів управління: тези доповідей десятої Міжнародної науково-технічної конференції, 8 – 9 квітня 2021 р., Харків, 2021. Т. 2. С. 26.

11. Krasnobayev V., Koshman S., **Kovalchuk D.**, Kuznesova Ye. Development of the adder structure by modulo of the system of residual classes. Проблеми інформатизації: тези доповідей дев'ятої Міжнародної науково-технічної конференції, 18 – 19 листопада 2021 р., Черкаси, 2021. Т. 1. С. 82.

12. Кошман С., Краснобаєв В., **Ковальчук Д.**, Кузнецова Є. Дослідження способів реалізації арифметичних операцій у системі залишкових класів. Проблеми інформатизації: тези доповідей дев'ятої Мжнародної науково-технічної конференції, 18 – 19 листопада 2021 р., Черкаси, 2021. Т. 1. С. 83.

13. Krasnobayev V., Yanko A., **Kovalchuk D.** Method of Tabular Implementation of the Arithmetic Operation of Multiplying Two Numbers Represented in the System of Residual Classes. 2022 IEEE 9th International Conference on Problems of Infocommunications, Science and Technology (PIC S&T), 10-12 October 2022, Kharkiv, Ukraine, 2022. P. 63-68. DOI:https://doi: 10.1109/PICST57299.2022.10238624. (Міжнародна конференція Scopus)

14. Krasnobayev V. A., Yanko A. S., **Kovalchuk D. M.** Mathematical Model and Method of Raising Integers to an Arbitrary Power of a Natural Number in

the System of Residual Classes. Математичне та імітаційне моделювання систем (МОДС 2022): тези доповідей сімнадцятої Міжнародної конференції, 14 – 16 листопада 2022 р., Чернігів, Україна, 2023. С. 15.

15. Янко А. С., **Ковальчук Д. М.** Дослідження можливості відмовостійкого функціонування комп'ютерної системи в непозиційній системі числення в залишкових класах. Сучасні напрями розвитку інформаційно-комунікаційних технологій та засобів управління: тези доповідей тринадцятої міжнародної науково-технічної конференції, 26 – 27 квітня 2023 р., Харків, 2023. Т.–2. С. 50. DOI:https://doi.org/10.32620/ICT.23.t2.

16. Krasnobayev V. A., Yanko A. S., **Kovalchuk D. M.** Control, Diagnostics and Error Correction in the Modular Number System. Proceedings of The Sixth International Workshop on Computer Modeling and Intelligent Systems (CMIS 2023), 3 May 2023, Zaporizhzhia, Ukraine, 2023. P. 199-213. (Міжнародна конференція Scopus)

17. Krasnobayev V., Yanko A., Martynenko A., **Kovalchuk D.** Method for computing exponentiation modulo the positive and negative integers. Materials of the XI International Scientific Conference «Information-Management Systems and Technologies», 21-23 September 2023, Odessa, Ukraine, 2023. P. 150-153.

18. Krasnobayev V., Yanko A ., Martynenko A., **Kovalchuk D.** Method for Computing Exponentiation Modulo the Positive and Negative Integers. Proceedings of the 11-th International Conference «Information Control Systems & Technologies», Odessa, Ukraine, 21-23 September, 2023. P. 374-383. (Міжнародна конференція Scopus)

19. Krasnobayev V., Yanko A. **Kovalchuk D.** An Improved Method for Performing the Arithmetic Operations of Modulo Addition of the Remainders of Numbers. 2023 13th International Conference on Dependable Systems, Services and Technologies (DESSERT), Athens, Greece, 13-15 October 2023, pp. 1-6, DOI: https://doi.org/10.1109/DESSERT61349.2023.10416508.

Наукові праці, які додатково відображають наукові результати дисертації:

Монографії

20. Krasnobayev V., Koshman S., **Kovalchuk D.** Method of Tabular Implementation of Modular Arithmetic Operations in the System of Residual Classes. ISCI'2020: Information Security in Critical Infrastructures. Collective monograph. Edited by Gorbenko I., Krasnobayev V. Kuznetsov A. ASC Academic Publishing, USA, 2020. P. 109-118. ISBN: 978-1-7362833-0-1 (Hardback), ISBN: 978-1-7362833-1-8 (Ebook).

(Особистий внесок: розробка табличного методу реалізації цілочисельних арифметичних модульних операцій додавання, віднімання та множення, участь в обговоренні отриманих результатів, а також написання частини тексту та його переклад).

21. Krasnobayev V., Yanko A., Martynenko A., **Kovalchuk D.** Method for computing exponentiation modulo the positive and negative integers. Information processing in control and decision-making systems. Problems and solutions. Monograph. Odessa, 2023. P. 233-257.

(Особистий внесок: розробка математичної моделі процесу піднесення цілих чисел до довільного степеню натурального числа в СЗК, участь в обговоренні отриманих результатів, а також написання частини тексту та його переклад).

Патенти

22. Краснобаєв В. А., Кузнецов О. О., Кузнецова К. О., **Ковальчук Д. М.** Пристрій для додавання лишків чисел за модулем m_i системи залишкових класів: патент України 126181 Україна: МПК: G06F 7/50 (2006.01), G06F 11/10 (2006.01), G06F 7/72 (2006.01). № a202100522; заявл. 09.02.2021; опубл. 25.08.2022, бюл. № 34/2022. 9 с.

23. Краснобаєв В. А., Кузнецов О. О., Кузнецова К. О., **Ковальчук Д. М.** Операційний пристрій у системі залишкових класів: патент

України 126182 Україна: МПК: G06F 7/50 (2006.01), G06F 7/503 (2006.01), G06F 7/72 (2006.01). № a202100523; заявл. 09.02.2021; опубл. 25.08.2022, бюл. № 34/2022. 9 с.

24. Кошман С. О., Краснобаєв В. А., Кузнецов О. О., **Ковальчук Д. М.** Суматор за довільним модулем m системи залишкових класів: патент на корисну модель 148170 Україна: МПК: G06F 7/50 (2006.01). № u202100701; заявл. 17.02.2021; опубл. 14.07.2021, бюл. № 28/2021. 9 с.

25. Краснобаєв В. А., Кошман С. О., Кузнецов О. О., Мавріна М. О., **Ковальчук Д. М.** Операційний пристрій у системі залишкових класів: патент на корисну модель 149074 Україна: МПК (2006): G06F 7/00, G06F 7/72 (2006.01). № u202102897; заявл. 31.05.2021; опубл. 13.10.2021, бюл. № 41/2021. 9 с.

26. Кошман С. О., Краснобаєв В. А., Кузнецов О. О., Мавріна М. О., **Ковальчук Д. М.** Пристрій для контролю та виправлення однократних помилок у даних, які представлені системою залишкових класів: патент на корисну модель 149060 Україна: МПК: G06F 7/50 (2006.01). № u202102707; заявл. 24.05.2021; опубл. 13.10.2021, бюл. № 41/2021. 6 с.

27. Краснобаєв В. А., Кошман С. О., Кузнецов О. О., Мавріна М. О., **Ковальчук Д. М.** Пристрій для визначення лишків числа за довільним модулем системи залишкових класів: патент на корисну модель149421 Україна: МПК (2006): G06F 5/00. № u202102898; заявл. 31.05.2021; опубл. 17.11.2021, бюл. № 46/2021. 4 с.

ЗМІСТ

ПЕРЕЛІК УМОВНИХ СКОРОЧЕНЬ…….. 18
ВСТУП…… 19
РОЗДІЛ 1. АНАЛІЗ ШЛЯХІВ ПІДВИЩЕННЯ ШВИДКОДІЇ ПРОГРАМНО-АПАРАТНИХ СИСТЕМ І КОМПЛЕКСІВ З ЕЛЕМЕНТАМИ ШТУЧНОГО ІНТЕЛЕКТУ…….. 27
1.1 Аналіз проблем побудови програмно-апаратних систем і комплексів з елементами штучного інтелекту…… 27
1.2. Дослідження методів підвищення швидкодії програмно-апаратних систем і комплексів з елементами штучного інтелекту.. 31
1.3. Аналіз можливості застосування непозиційної системи числення у залишкових класах для підвищення швидкодії програмно-апаратних систем і комплексів з елементами штучного інтелекту…… 34
1.4. Вибір та обґрунтування показника для оцінки швидкодії програмно-апаратних систем і комплексів з елементами штучного інтелекту…… 40
1.5. Постановка задачі дослідження дисертації……. 45
Висновки до розділу 1…… 48
РОЗДІЛ 2. ВДОСКОНАЛЕННЯ МЕТОДІВ ВИКОНАННЯ АРИФМЕТИЧНИХ ОПЕРАЦІЙ ДОДАВАННЯ ТА ВІДНІМАННЯ ЗАЛИШКІВ ЧИСЕЛ ПО МОДУЛЮ В СЗК…….. 50
2.1. Дослідження методів додавання та віднімання залишків чисел по модулю в СЗК……. 50
2.2 Аналіз впливу додаткових зв’язків суматора по модулю на величину вмісту суматора……. 53
2.3. Метод реалізації операції додавання залишків чисел по модулю m_i СЗК…….. 70

2.4 Метод реалізації операції віднімання залишків чисел по модулю m_i СЗК …….. 88
Висновки до розділу 2……. 104
РОЗДІЛ 3. РОЗРОБКА МОДЕЛЕЙ ТА МЕТОДІВ ШВИДКОЇ ОБРОБКИ ДАНИХ НА ОСНОВІ ЗАСТОСУВАННЯ СИСТЕМИ ЗАЛИШКОВИХ КЛАСІВ…….. 105
3.1 Математична модель і метод множення двох залишків комплексних чисел в СЗК…….. 105
3.2 Математична модель процесу піднесення цілих чисел до довільного степеня натурального числа в СЗК…… 118
Висновки до розділу 3…… 134
РОЗДІЛ 4. РОЗРОБКА ПРИСТРОЇВ ВИКОНАННЯ АРИФМЕТИЧНИХ ОПЕРАЦІЙ, ПРОГРАМНО-АПАРАТНИХ СИСТЕМ І КОМПЛЕКСІВ З ЕЛЕМЕНТАМИ ШТУЧНОГО ІНТЕЛЕКТУ, ЩО ФУНКЦІОНУЮТЬ В СИСТЕМІ ЗАЛИШКОВИХ КЛАСІВ…….. 135
4.1. Розробка операційного пристрою в системі залишкових класів 135
4.2. Розрахунок та порівняльний аналіз швидкодії програмно-апаратних систем і комплексів з елементами штучного інтелекту в СЗК…… 145
Висновки до розділу 4…… 150
ВИСНОВКИ…… 152
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ…….. 156
ДОДАТОК А. Список публікацій здобувача за темою дисертації…… 165

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ

ЕОМ - електронна обчислювальна машина

ІС – інтегральна схема

КТМ – код табличного множення

ММ – математична модель

НСД – найбільший спільний дільник

ПЛІС – програмована логічна інтегральна схема

ПСЧ – позиційна система числення

СЗК – система залишкових класів

ШІ – штучний інтелект

ШФ – штучна форма

ВСТУП

Обґрунтування вибору теми дослідження

На даний час інтенсивна комп’ютеризація та штучний інтелект (ШІ) є взаємопов'язаними та взаємозалежними концепціями, які впливають на сучасну суспільну та економічну діяльність. Вони відкривають нові можливості, сприяють інноваціям, розвитку та застосуванню передових технологій у багатьох сферах і допомагають вирішувати складні завдання, які раніше було б важко або навіть неможливо вирішити. ШІ може:

- аналізувати та використовувати великі обсяги даних у реальному часі, що важливо в багатьох сферах, включаючи аналітику даних, розпізнавання образів та обробку природних мов;

- автоматизувати багато рутинних і повторюваних завдань, що раніше виконувалися вручну. Це стосується виробництва, логістики, бухгалтерії, адміністративної, що дозволить значно зменшити людську працездатність та знизити витрати;

- навчатися і адаптуватися до нових ситуацій і завдань без необхідності перепрограмовування;

- підвищити точність та продуктивність процесів в багатьох сферах, включаючи медицину, фінанси, виробництво та багато інших. Це може призвести до покращення результатів і зниження втрат.

- використовуватися для моделювання складних процесів, вирішення наукових головоломок і виявлення нових залежностей у великих наборах даних.

- створює нові галузі та інноваційні продукти, такі як роботи, автономні системи, системи розпізнавання голосу та багато інших.

Важливим фактором роботи багатьох систем ШІ, особливо в контексті реального часу або завдань, де час реакції має значення є швидкодія обробки інформації та прийняття рішень. Швидкодія обробки є критичним фактором для багатьох аспектів ШІ. ШІ вимагає великої кількості обчислень, і швидкодія

обробки може значно вплинути на продуктивність і ефективність алгоритмів штучного інтелекту. У багатьох задачах, пов'язаних з ШІ, важлива здатність до обробки даних в реальному часі. У голосових асистентах, чат-ботах та інших системах інтерактивного ШІ швидкодія обробки є важливою для надання користувачам миттєвих відповідей та взаємодії у реальному часі. Швидка обробка даних дозволяє оптимізувати використання обчислювальних ресурсів, зменшуючи час обчислень та витрати енергії.

Один з найбільш перспективних методів підвищення швидкодії сучасних обчислювальних систем - це використання паралельної обробки інформації. Ця можливість є характерною для системи залишкових класів (СЗК), яка забезпечує значне покращення характеристик обчислювальних систем порівняно з пристроями, що базуються на тій же апаратно-технологічній основі і працюють у позиційних системах числення. Першим поштовхом до дослідження СЗК стали опубліковані в 1955 – 1960 рр. праці чеських вчених М. Валаха, Н. Сабо і А. Свободи. Вагомий теоретичний та практичний внесок у розвиток СЗК та її застосування в обчислювальній техніці внесли такі науковці: І. Акушський, Д. Юдицький, В. Торгашев, В. Амербаєв, В. Краснобаєв, М. Червяков, О. Фінько, A. Shimbo, P. Paulier, M. Thornton, R. Dreschler, D. Miller, A. Mohan та інші.

Хоча в СЗК є недоліки, такі як складнощі з виконанням операції ділення, порівнянням операндів та виходом результатів за межі робочого діапазон відсутність ознак виходу результатів операції за межі робочого діапазону, проте вона успішно застосовується для виконання операцій додавання, віднімання, та множення цілих багаторозрядних чисел, що є важливим для швидкої обробки цілих числових даних. Однією з переваг СЗК є можливість виконання операцій над числами, які менші за обрані модулі, а також можливість розпаралелення процесу обчислень та відсутність міжрозрядних переносів.

Необхідно зазначити, що існує чисельний клас задач і алгоритмів, де окрім виконання цілочисельної арифметичної операції множення та операції піднесення залишків цілих чисел до довільного степеня натурального числа за

модулем у додатному числовому діапазоні, однак існує необхідність реалізації перелічених вище операцій не тільки в додатному а і у від’ємному числових діапазонах.

Викладене вище обумовлює актуальність рішення ***науково-прикладної задачі*** розробки моделей та методів швидкої обробки даних на основі застосування системи залишкових класів з метою підвищення швидкості обробки інформації програмно-апаратними системами і комплексами з елементами штучного інтелекту.

Зв’язок роботи з науковими програмами, планами, темами

Тематика дисертаційної роботи пов'язана з дослідженнями:

− Участь у НДР «Формулювання та розробка принципів, методів і засобів швидкої та достовірної обробки цілочисельних даних, що представлені у непозиційній системи числення залишкових класів в комп’ютерних системах та мережах подвійного призначення» за 2019-2021 рр. (НДР № 0119U002546), у якості виконавця.

Мета і задачі дослідження. Основною ***метою*** дисертаційної роботи є підвищення швидкості обробки інформації програмно-апаратними системами і комплексами з елементами штучного інтелекту за рахунок використання моделей та методів швидкої обробки даних на основі застосування системи залишкових класів.

Для досягнення даної мети як рішення поставленого наукового завдання в цілому, були сформульовані ряд ***задач***. До їх числа належать.

1. Аналіз проблем обробки інформації програмно-апаратними системами і комплексами з елементами штучного інтелекту в реальному часі.
2. Удосконалення методу додавання та віднімання залишків чисел по модулю СЗК.
3. Удосконалення методу табличної реалізації множення двох залишків чисел за рахунок можливості виконання операції в комплексній області.
4. Удосконалення математичної моделі процесу піднесення залишків

цілих чисел до довільного степеня натурального числа в СЗК.

5. Практичне підтвердження працездатності та вірогідності розроблених моделей і методів.

Об'єкт дослідження – процеси обробки даних у системі залишкових класів, поданих у цілочисельному вигляді.

Предмет дослідження – методи та цифрові компоненти програмно-апаратних систем і комплексів з елементами штучного інтелекту, представлених у системі залишкових класів.

Методи дослідження. Теоретичні основи роботи базуються на принципах і методах системного аналізу, математичного та імітаційного моделювання. Як математична основа для розробки моделей та методів швидкої обробки даних на основі застосування системи залишкових класів, використовуються принципи системного аналізу, теорія чисел (розділи: теорія подільності та теорія порівнянь) і теорія обчислень (під час розроблення моделей та методів швидкої обробки даних на основі застосування системи залишкових класів), теорія обчислювальних процесів і систем, методи імітаційного моделювання (під час оцінювання коректності та достовірності моделей і методів).

Наукова новизна отриманих результатів полягає в наступному

1. ***Вдосконалено*** метод табличної реалізації множення двох залишків чисел в системі залишкових класів за рахунок можливості виконання операції в комплексній області, на основі використання першої фундаментальної теореми Гауса про ізоморфізм між множиною дійсних і комплексних чисел, що підвищує швидкодію реалізації операції множення в системі залишкових класів.

2. ***Вдосконалено*** математичну модель процесу піднесення цілих чисел до довільного степеня натурального числа в СЗК за рахунок можливості виконання операції піднесення цілих чисел до степеня, як у додатному, так і в від'ємному числових діапазонах, що підвищує швидкодію реалізації операції піднесення цілих чисел до степеня в системі залишкових класів.

3. ***Дістав подальший розвиток*** метод додавання і віднімання залишків чисел по модулю СЗК, який враховує конструкції суматорів по модулю з величиною корекції $\Delta Q_R > 0$.

Особистий внесок здобувача. Дисертаційне дослідження виконано здобувачем самостійно, усі сформульовані в ньому положення та висновки з рекомендаціями обґрунтовані на основі особистих досліджень автора. Для аргументації окремих положень використані праці інших науковців, на які зроблені посилання. В індивідуальних наукових працях застосовано лише авторські ідеї та розробки.

Дисертант брав активну участь у наукових дискусіях, семінарах, підготовці наукових статей, опублікованих за темою дисертації, успішно доповідав результати досліджень на міжнародних конференціях.

У праці [1] здобувач брав участь у розробці методу діагностики даних, які представлені в СЗК. У праці [2] здобувачу належить алгоритм синтезу суматора за довільним модулем СЗК. У праці [3] здобувачем запропоновано новий метод обчислення суми залишків чисел за довільним модулем, а також здобувач навів приклади, що наочно демонструють ефективність запропонованого методу. У праці [4] здобувач провів аналіз можливості застосування непозиційної системи числення у залишкових класах для підвищення швидкодії програмно-апаратних систем і комплексів з елементами штучного інтелекту. У праці [5] здобувач брав участь у побудові математичної моделі надійності комп'ютерної системи, що функціонує у СЗК. У праці [6] здобувачем запропоновано табличні методи реалізації цілочисельних арифметичних модульних операцій додавання, віднімання та множення. У праці [7] здобувачу належить постановка та вирішення зворотної задачі оптимального резервування в СЗК на основі використання методу динамічного програмування. У праці [8] здобувачу належить розробка математичної моделі процесу піднесення цілих чисел до довільного

степеню натурального числа в СЗК. У публікаціях [9–19] здобувач брав участь у розробці моделей та методів швидкої обробки даних в СЗК, написанні тексту матеріалів тез доповідей та доповіді на конференціяхї. В розділі монографії [20] здобувачем запропоновано табличні методи реалізації цілочисельних арифметичних модульних операцій додавання, віднімання та множення. В розділі монографії [21] здобувачу належить розробка математичної моделі процесу піднесення цілих чисел до довільного степеню натурального числа в СЗК. У публікаціях [22–27] дисертант брав участь у розробці цифрових компонент комп'ютерної системи швидкої обробки даних на основі застосування непозиційної системи числення в СЗК, обговоренні результатів, написанні тексту.

Практичне значення отриманих результатів

Розроблені в дисертаційній роботі моделі і методи швидкої обробки даних є науково-методологічною основою для практичного створення програмно-апаратних системам і комплексів з елементами штучного інтелекту.

Використання методу табличної реалізації множення двох комплексних залишків чисел та моделі піднесення цілих чисел до довільного степеня натурального числа в СЗК та підвищують швидкодію реалізації модульних операцій в СЗК.

Метод додавання і віднімання залишків чисел по модулю СЗК, можна використовувати у програмно-апаратних системах і комплексах з елементами штучного інтелекту, зокрема для підвищення швидкодії обчислень, забезпечення.

Розрахунки та порівняльна оцінка швидкодії, проведені в дисертаційній роботі, показали, що зі збільшенням розрядності сітки програмно-апаратних систем і комплексів з елементами штучного інтелекту ефективність застосування непозиційної системи числення в СЗК значно зростає.

На основі розроблених методів швидкої обробки даних в СЗК, в дисертації запропоновано клас технічних засобів, на які отримано 6 патентів

України.

Отримані в процесі дослідження теоретичні положення, моделі й методи можуть бути використані при поглибленому вивченні окремих розділів обчислювальної техніки. Отримані результати використовують у процесі викладання навчальних курсів «Основи комп'ютерної схемотехніки», «Комп'ютерна схемотехніка та архітектура комп'ютерів» у Харківському національному університеті імені В. Н. Каразіна.

Апробація результатів дисертації. Основні теоретичні положення, висновки і пропозиції, які містяться в дисертації, обговорювалися та були схвалені на засіданнях кафедри електроніки та управляючих систем Харківського національного університету імені В.Н. Каразіна. Ключові положення дослідження оприлюднені у доповідях на науково-технічних конференціях всеукраїнського та міжнародного рівнів (2021–2023 роки).

- Десятій Міжнародній науково-технічній конференції «Сучасні напрями розвитку інформаційно-комунікаційних технологій та засобів управління» (Баку – Харків – Київ - Жиліна, 8 – 9 квітня 2021 р.).

- Дев'ятій Міжнародній науково-технічній конференції «Проблеми інформатизації», (Черкаси – Баку – Бельсько-Бяла, 18 – 19 листопада 2021 р.).

- 2022 IEEE 9th International Conference on Problems of Infocommunications, Science and Technology (PIC S&T, Ukraine, Kharkiv, 10-12 October 2022).

- Сімнадцятій Міжнародній конференції «Математичне та імітаційне моделювання систем» (МОДС 2022, Україна, м. Чернігів, 14 – 16 листопада 2022 р.).

- Тринадцятій Міжнародній науково-технічній конференції «Сучасні напрями розвитку інформаційно-комунікаційних технологій та засобів управління» (Баку – Харків – Київ - Жиліна, 26 – 27 квітня 2023 р.).

- The Sixth International Workshop on Computer Modeling and Intelligent Systems (CMIS 2023, Ukraine, Zaporizhzhia, 3 May 2023).

- XI International Scientific Conference «Information-Management Systems

and Technologies» (Ukraine, Odessa, 21-23 September 2023).

- 11-th International Conference «Information Control Systems & Technologies», (Ukraine, Odessa, 21-23 September).

- 13th International Conference on Dependable Systems, Services and Technologies (DESSERT), (Athens, Greece, 13-15 October 2023)

Публікації. Основні теоретичні положення і висновки дисертації викладені у 27 наукових працях, з яких 8 статті у наукових фахових виданнях України та ті, що входять до міжнародних наукометричних баз [1–8], 10 тез наукових доповідей [9–18], 2 розділи у монографіях [19-20] та 6 патентів [21–26].

Структура та обсяг дисертації. Дисертаційна робота складається зі вступу, чотирьох розділів, висновків, списку використаних джерел і 1 додатку. Загальний обсяг дисертації становить 170 сторінок: у тому числі анотації на 8 сторінках, зміст на 2 сторінках, основний текст на 137 сторінках, список використаних джерел із 73 найменуваня на 10 сторінках. Робота містить 25 таблиць та 62 рисунків, з яких 5 на окремих 9 сторінках.

РОЗДІЛ 1. АНАЛІЗ ШЛЯХІВ ПІДВИЩЕННЯ ШВИДКОДІЇ ПРОГРАМНО-АПАРАТНИХ СИСТЕМ І КОМПЛЕКСІВ З ЕЛЕМЕНТАМИ ШТУЧНОГО ІНТЕЛЕКТУ

1.1 Аналіз проблем побудови програмно-апаратних систем і комплексів з елементами штучного інтелекту

У двадцять першому столітті інформаційні технології стрімко розвиваються і набувають все більшої популярності в усьому світі, завдяки новим теоретичним і практичним досягненням у галузі обчислювальної техніки. Сучасне суспільство відводить важливе місце для новітніх технологій та робототехніки, зокрема для штучного інтелекту (ШІ), який є однією з найбільш популярних та важливих інноваційних технологій. ШІ проникає у всі сфери життя та стає необхідною складовою сучасного суспільства.

Згідно з Концепцією розвитку штучного інтелекту в Україні, ухваленою 2 грудня 2020 року, ШІ визначається як організована система інформаційних технологій, яка дозволяє вирішувати складні задачі за допомогою наукових методів досліджень і алгоритмів обробки інформації. Це включає використання отриманої або створеної інформації, створення власних баз даних та моделей прийняття рішень, а також визначення шляхів досягнення поставлених цілей [1, 2].

В розвитку ШІ фундаментальну роль відіграє математика. Різні галузі математики, такі як логіка, теорія ймовірностей, математична статистика, алгоритми та теорія обчислень, є ключовими складовими для реалізації алгоритмів та методів, які використовуються в ШІ. Комбінація цих математичних концепцій дозволяє розробляти складні моделі та алгоритми, які допомагають системам ШІ аналізувати дані, приймати рішення та виконувати різноманітні завдання.

В Україні розрізняють такі галузі використання ШІ: сфера освіти та науки, оборона, кібербезпека та захист інформації, економіка та управління суспільством, законодавче регулювання та юридична система.

Впровадження Стратегії розвитку ШІ в Україні сприятиме широкому застосуванню штучного інтелекту в галузі оборони і безпеки, що стане важливим стимулом для ефективного розвитку оборонно-промислового комплексу країни. ШІ технології будуть використані для підтримки процесу прийняття рішень у готуванні тактичних та стратегічних операцій і військових дій; у керуванні космічною та високоточною зброєю, наземними, підводними та повітряними безпілотними системами, ударними та розвідувальними комплексами; для аналізу супутникових зображень, для автоматизації трудомісткого будівництва військових та цивільних інженерних споруд, а також кіберзахисту [2].

Також важливим є впровадження систем штучного інтелекту в мобільні платформи, зокрема в підводні та наземні роботи, безпілотні літальні апарати, для боротьби з надводними та підводними човнами та безпілотними літальними апаратами противника.

Актуальним є також упровадження ШІ в мобільні системи, зокрема в безпілотні літальні апарати, наземні та підводні роботи для боротьби з літальними апаратами, підводними й надводними човнами противника. Основними шляхами впровадження штучного інтелекту в мобільні системи, зокрема БПЛА, є використання машинного навчання (включаючи глибоке навчання), комп'ютерного зору і обробки зображень, аналізу великих обсягів даних, розпізнавання мовлення, створення стійких систем зв'язку, використання мультиагентних технологій управління та організації роїв автономних роботів [3].

Стратегія розвитку штучного інтелекту в Україні передбачає створення програмно-апаратних систем і комплексів з елементами штучного інтелекту. Під програмно-апаратними системи та комплекси з елементами штучного інтелекту будемо розуміти інтегровані системи, що поєднують у собі як апаратну, так і програмну складові, які використовують методи та технології штучного інтелекту для вирішення певних завдань або досягнення певних цілей. Ці системи включають в себе різноманітні алгоритми та моделі

машинного навчання, нейронні мережі, логічні системи, системи обробки природної мови та інші методи та техніки, спрямовані на аналіз даних, роботу з зображеннями, розпізнавання образів, прийняття рішень тощо.

Функціонування програмно-апаратних систем і комплексів з елементами штучного інтелекту можна описати як неперервний процес прийняття рішень, що базується на аналізі поточних обставин з метою досягнення конкретної цілі.

Розглянемо один із напрямів програмно-апаратних систем і комплексів з елементами штучного інтелекту є створення інтелектуальних роботів.

Створення інтелектуальних роботів вимагає розробки не лише спеціалізованих комп'ютерів, але й повного набору енергетичних, електромеханічних та інформаційних систем, таких як двигуни, сенсори та джерела енергії, для локальних мехатронних компонентів. Роботи, які вимагають інтелектуального підходу, спрямовані на аналіз знань, що надходять через потоки інформації з навколишнього середовища до вбудованих комп'ютерів (наприклад, у рухомих роботах). Оскільки дані, які подаються в комп'ютер для обробки, є ситуативними, необхідно обробляти їх у реальному часі. У зв’язку з цим необхідно, щоб комп'ютерна система працювала зі швидкістю, що становить 10 мільярдів операцій на секунду. Тому актуальною є задача підвищення швидкодії програмно-апаратних систем і комплексів з елементами штучного інтелекту [4, 5].

Також одним із напрямків розвитку програмно-апаратних систем і комплексів з елементами штучного інтелекту є функціонування таких систем і комплексів у режимі реального часу. Ця ситуація передбачає здатність програмно-апаратних систем та комплексів із елементами штучного інтелекту аналізувати динамічні зміни у зовнішньому середовищі та приймати рішення вчасно, щоб адекватно реагувати на ці зміни. Програмно-апаратні системи і комплекси з елементами штучного інтелекту, що функціонують у режимі реального часу не лише піддаються інтенсивним науковим дослідженням і розробкам, а й все більше застосовуються у виробничих умовах, і стають цінним комерційним продуктом. Ця умова підтверджує актуальність задачі

підвищення швидкодії програмно-апаратних систем і комплексів з елементами штучного інтелекту.

Ще проблема підвищення швидкодії програмно-апаратних систем і комплексів з елементами штучного інтелекту характерна для таких напрямків ШІ, як розпізнавання мовлення, розпізнавання образів і комп’ютерний зір, оскільки такі системи функціонують в режимі реального часу [6-8]. Ця задача підтверджує актуальність задачі підвищення швидкодії програмно-апаратних систем і комплексів з елементами штучного інтелекту.

Також розвиток програмно-апаратних систем і комплексів з елементами штучного інтелекту тісно пов'язаний з аналізом великих даних. Оскільки для аналізу великих даних висуваються збільшені вимоги до швидкості обробки інформації, то актуальною є задача підвищення швидкодії програмно-апаратних систем і комплексів з елементами штучного інтелекту.

У результаті досліджень, проведених різними науковими та інженерними групами протягом останніх десятиліть у сфері інформаційних технологій, було виявлено, що використання ПСЧ у обчисленнях значною мірою не призводить до практичного підвищення швидкодії програмно-апаратних систем та комплексів з елементами штучного інтелекту. Це обумовлюється головним недоліком ПСЧ, який полягає в наявності міжрозрядних зв'язків між операндами, які підлягають обробці. Вплив міжрозрядних зв'язків суттєво відображається на структурі процесора, який виконує обчислення, та способах виконання арифметичних операцій, призводить до складнішої апаратури та значно обмежує швидкість виконання базових арифметичних операцій, таких як додавання, віднімання та множення. У зв'язку з цим, для поліпшення описаних характеристик програмно-апаратних систем та комплексів з використанням штучного інтелекту, в першу чергу використовується підвищення тактової частоти і розвиток методів та засобів паралельної обробки даних [9-10].

Таким чином, проведений аналіз можливостей програмно-апаратних систем і комплексів з елементами штучного інтелекту свідчить про те, що вони

не задовольняють збільшеним вимогам до швидкості обробки інформації, що обумовлює актуальність дослідження нових моделей і методів.

1.2. Дослідження методів підвищення швидкодії програмно-апаратних систем і комплексів з елементами штучного інтелекту

У більшості випадків підвищення швидкодії програмно-апаратних систем і комплексів з елементами штучного інтелекту, що полягає в здатності виконувати більшу кількість операцій протягом меншого часу, досягається за допомогою спеціальних технологічних та архітектурних рішень, а також застосуванням математичних методів. Розвиток сучасних програмно-апаратних систем і комплексів з елементами штучного інтелекту відбувається відповідно до зазначених напрямів, при цьому вони взаємно впливають один на одного. Вдосконалення технологій виробництва інтегральних схем (ІС) дозволяє збільшувати тактову частоту, що призводить до підвищення швидкості обробки даних і забезпечує зростання рівня інтеграції ІС. Збільшенню швидкості обробки даних сприяє також скорочення ланцюжків логічних елементів, що здійснюють базові операції над даними за кожен такт [11, 12].

Підвищення швидкодії архітектурними методами включає в себе різноманітні стратегії, такі як створення систем багатопроцесорних обчислювальних систем (SMP), оптимізація форматів та системи команд з огляду на прогрес у програмуванні, застосування методів паралельної обробки даних під час виконання команд і різні інші підходи [13-14].

Математичні підходи спрямовані на розробку нових методів обчислення для розв'язання різних класів задач, які можна розпаралелити для оптимізації обчислювальних процесів [13-14].

У всіх наявних способах підвищення швидкодії (ефективності) програмно-апаратних систем і комплексів з елементами штучного інтелекту є спільний недолік: неможливість паралельного виконання алгоритмів лише на рівні простих операцій.

Розглянемо наявні та потенційні шляхи підвищення швидкодії (ефективності). Під швидкодією програмно-апаратних систем і комплексів з елементами штучного інтелекту мається на увазі користувальницька швидкодія вирішення конкретної задачі. На даний момент впроваджуються або лише розробляються наступні способи підвищення користувальницької швидкодії [15]:

- розробка програмно-апаратних систем і комплексів з елементами штучного інтелекту які складаються із групи процесорів або низки окремих засобів для обробки інформації, таких як багатомашинні системи, багатопроцесорні системи, конвеєрні обчислювальні системи, тощо.

- використання різних характеристик класу задач для розв'язуваних задач, наприклад, природний паралелізм операцій, паралелізм незалежних гілок операцій та суміжних обчислень, створення штучної паралельності, тощо;

- покращення швидкодії за допомогою впровадження абсолютно нової елементної бази на основі єдиного процесора.

- розробка нової архітектури надпродуктивного процесора з урахуванням використання голографічного принципу обробки інформації.

Проаналізуємо, як деякі зазначені вище методи можуть вплинути на підвищення швидкодії програмно-апаратних систем та комплексів з елементами штучного інтелекту [16, 17].

Використання багатьох окремих процесорів може збільшити системну швидкодію системи, при цьому зберігаючи рівень користувальницької швидкодії. Однак для реалізації цього методу потрібно значну кількість обладнання, що обмежує його практичність для використання на бортових програмно-апаратних системах і комплексах з елементами штучного інтелекту.

Прогрес у розвитку сучасної мікроелектроніки, яка переважно ґрунтується на використанні ПЛІС, спонукає до вивчення можливостей використання табличних методів обробки даних. Використання методів табличної обробки даних може забезпечити високу швидкодію через можливість паралельного виконання базових операцій. Крім того, це сприяє

високому рівню структурної регулярності та однорідності пристроїв, що використовуються для цих цілей. Але, значна кількість обладнання лишається істотним недоліком табличних методів обробки даних в ПСЧ, що призводить до ускладнень і навіть неможливості їх практичної реалізації в деяких випадках. Так, нехай точність обчислень операційного пристрою програмно-апаратних систем і комплексів з елементами штучного інтелекту ПСЧ становитиме величину Z, а основа системи числення дорівнює q. У цьому випадку кількість адрес, наприклад, для матричного пристрою, дорівнює q^Z. Для значень Z=128, q=2 число адрес дорівнює 2^{128}, а число схем збігу І у вузлах матричного ПЗП дорівнює 2^{256}, що навряд чи доцільно і може бути реалізовано для програмно-апаратних систем і комплексів з елементами штучного інтелекту в ПСЧ.

Оскільки наявні системи програмно-апаратного забезпечення з елементами штучного інтелекту працюють з даними, які представляються в ПСЧ, виникають проблеми з організацією процесу обчислень та передачі інформації між бітами операндів. Алгоритмічний зв'язок в ПСЧ усіх двійкових розрядів операнда між собою обумовлює той факт, що поодинока відмова або збій схеми обробки одного двійкового розряду операційного пристрою здатний викликати не одноразову помилку, а багаторазові помилки в машинному слові. До того ж саме наявність міжрозрядних зв'язків не дозволяє розпаралелити розв’язувані алгоритми, на рівні елементарних операцій.

Отже, серед різноманітних способів підвищення швидкодії, є очевидна можливість створення перспективних програмно-апаратні системи та комплексів з елементами штучного інтелекту, використовуючи новаторську архітектуру [18, 19]. Ця архітектура спрямована на втілення принципу паралелізму у обробці даних. Однак у всіх існуючих та перспективних методах покращення швидкодії в ПСЧ виникає загальна проблема: не можливо розпаралелити алгоритми так, щоб вони виконувались паралельно на рівні елементарних операцій.

1.3. Аналіз можливості застосування непозиційної системи числення у залишкових класах для підвищення швидкодії програмно-апаратних систем і комплексів з елементами штучного інтелекту

Представлення числа в системі залишкових класів заснована на широко відомій китайській теоремі про залишки (Chinese Remainder Theorem). [20 - 23]. Вона стверджує, що знаючи найменші невід'ємні залишки від ділення цілого числа P на цілі модулі m_1, m_2, . . . , m_n, можна однозначно визначити залишок від ділення P на добуток цих модулів, за умови, що модулі попарно взаємно прості. СЗК визначається набором попарно взаємно простих модулів $(m_1, m_2, ..., m_n)$, тобто таких, що

$$НСД\ (m_i,\ m_j) = 1, \text{ для всих } i, j = 0, 1, \ldots, n;\ i \neq j,$$

де НСД – найбільший спільний дільник.

Добуток цих модулів $M = m_1 \cdot m_2 \cdot ... \cdot m_n$ визначає діапазон СЗК. Ціле число $P \in [0, M-1]$ представляється набором найменших залишків чисел $(p_1, p_2, ..., p_n)$:

$$p_1 \equiv P(\mathrm{mod}\, m_1);$$
$$p_2 \equiv P(\mathrm{mod}\, m_2);$$
$$...$$
$$p_n \equiv P(\mathrm{mod}\, m_n).$$

Крім того, китайська теорема про залишки гарантує однозначність (унікальність) в представлення цілих невід’ємних чисел із діапазону $[0, M-1]$.

Як вже зазначалось, велике ціле число в СЗК представляється набором чисел, що є залишками від ділення вихідної позиційної величини на модулі. Особливість у цьому, що залишки є взаємно незалежними, тобто, між ними не виникає переносів, що дозволяє виконувати операції додавання, віднімання та множення з кожним залишком окремо. З огляду на це, забезпечуючи можливість реалізації високошвидкісної паралельної комп'ютерної арифметики довільної розрядності, СЗК є перспективним інструментом для високопродуктивних обчислень.

В останні роки інтерес до СЗК помітно зріс у зв'язку з активним розвитком паралельних архітектур. В даний час ця система числення знаходить застосування у багатьох ресурсоємних додатках, наприклад,

- блокчейн,
- гомоморфне шифрування,
- стохастичні обчислення,
- високонадійні хмарні середовища,
- глибокі нейронні мережі,
- аналіз даних комп'ютерної томографії,
- задачі швидкого перетворення Фур'є та їх застосування,
- цифрова обробка сигналів,
- обробка зображень.

Таким чином, СЗК сьогодні широко використовується в різних сферах [24, 25]. Наприклад, в мікроелектроніці, в спеціалізованих пристроях цифрової обробки сигналів, де потрібно:

- контроль помилок шляхом застосування додаткових надлишкових модулів;
- висока швидкість роботи, яка забезпечується паралельним виконанням основних арифметичних дій;
- інформаційна безпека.

Основою створення спецпроцесорів, що працюють в СЗК, є можливість одночасного використання всіх теоретико-числових властивостей СЗК [26 - 28].

Виділимо три основні властивості СЗК [30 - 33].

1. Незалежність залишків. Ця властивість СЗК дає можливість побудувати обчислювальний пристрій у вигляді набору з n незалежних обчислювальних трактів, що працюють паралельно в часі, та функціонують незалежно один від одного за певним модулем m_i СЗК.

Слід зазначити [29]:

- Час реалізації арифметичних операцій у програмно-апаратних системах і комплексах з елементами штучного інтелекту визначається часом реалізації

операції в обчислювальному тракті з найбільшим модулем m_i СЗК.

- Можливість розпаралеління обчислюваного алгоритму на рівні мікрооперацій дає можливість реалізувати більшість арифметичних операцій за один цикл роботи обчислювального пристрою.

- Модульність конструкції програмно-апаратних систем і комплексів з елементами штучного інтелекту у СЗК, що дозволяє здійснювати технічне обслуговування та усунення відмов і несправностей обчислювальних трактів простою заміною не перериваючи процес обчислювальної задачі.

- Помилки, що виникають у довільному обчислювальному тракті програмно-апаратних систем і комплексів з елементами штучного інтелекту, за рахунок відмов схем двійкових розрядів не поширюються на інші тракти, а залишаються в межах одного залишку, що підвищує достовірність обчислень у СЗК. При цьому не важливо, це була одноразова або багаторазова помилки або сукупність помилок довжиною не більше $[\log_2(m_i - 1)]+1$ двійкових розрядів. Помилка в обчислювальному тракті програмно-апаратних систем і комплексів з елементами штучного інтелекту по модулю m_i зберігається в цьому тракті до кінця обчислень або самоусуваються в процесі подальших обчислень [30-34].

2. Рівноправність залишків. Ця властивість означає, що будь-який залишок p_i числа $P=(p_1,p_2,\ldots, p_n)$ в СЗК містить інформацію щодо усього вихідного число P. Це дає можливість програмними методами замінити обчислювальний тракт по модулю m_i, що відмовив, на працездатний тракт по модулю m_j (за умови, що $m_i < m_j$) не перериваючи рішення задачі. При цьому, спеціальний процесор в СЗК зберігає свою працездатність у разі збоїв кількох обчислювальних трактів одночасно і здатний виконувати програму з незначним зниженням точності обчислень. Тобто такий спецпроцесор має властивість функціональної живучості. У цьому аспекті спецпроцесор в СЗК можна віднести до природної відмовостійкої обчислювальної структури.

3. Низька розрядність залишків. Ця властивість дозволяє суттєво підвищити надійність програмно-апаратних систем і комплексів з елементами штучного інтелекту та швидкодію виконання арифметичних операцій як за

рахунок малорозрядності обчислювальних трактів, а також і за рахунок можливості застосування (на відміну від ПСЧ) табличної арифметики. Табличний метод дозволяє виконуються практично в один такт арифметичні операції, такі як додавання, віднімання та множення. Зокрема, малорозрядність залишків у представлені чисел в СЗК на відміну від ПСЧ дає широкий вибір варіантів системотехнічних рішень реалізації модульних арифметичних операцій, заснованих на таких принципах [24, 35-38]:

- суматорний принцип (на основі малорозрядних двійкових суматорів);

- табличний принцип (на основі використання ПЗП);

- принцип кільцевого зсуву (на основі використанні кільцевих регістрів зсуву).

До основних недоліків системи числення в залишкових класах необхідно віднести:

- по виду числа в СЗК не можна визначити його кількісного значення;

- неможливість візуального зіставлення (порівняння) чисел, так як зовнішній вигляд запису числа не дає представлення о його величині;

- відсутність простих ознак виходу результатів операції за межі діапазону $[0, M)$;

- складність виконання операції ділення ; відношення P_1 / P_2 може не бути цілим числом, а якщо і є таким, то в загальному випадку не можна знайти його точне модульне представлення, обчислюючи p_{1_i} / p_{2_i} по модулю m_i для кожного значення i;

- отримання в усіх випадках точного результату операції, що виключає можливість безпосереднього наближеного виконання операції, округлення результату і тому подібне.

Відомо, що числа P_l в СЗК представляються набором залишків $p_i = P_l - [P_l / m_i]\, m_i (i = \overline{1,n})$, отриманих від послідовного поділу їх на обрану систему залишків $\{m_i\}$.

Найчастіше розглядається СЗК, для якої залишок m_i обираються цілими

додатними числами, причому система залишків вибирається так щоб залишки були попарно взаємно простими числами, тобто:

$$НСД(m_i, m_j) = 1, \text{ для } i \neq j.$$

Таким чином, з принципу побудови коду видно, що в СЗК кожен залишок p_i несе інформацію про весь вихідний об'єкт Q, що описується інформаційним кодом $P_l\left(l = \overline{1, \prod_{i=1}^{n} m_i}\right)$, а обсяг представлення кодових слів визначається наступним числовим діапазоном $\left[0, \prod_{i=1}^{n} m_i - 1\right)$.

Для цього діапазону існує однозначна відповідність між числами в СЗК в. Чим більше число n залишків (основ) СЗК і чим вони більші за величиною, тим точніше описується інформаційний об'єкт Q.

Зазначимо, що при обробці інформації в СЗК існує можливість здійснення операцій обміну між точністю обчислення алгоритму, надійністю та швидкістю в динаміці обчислювального процесу, тобто в реальному часі.

Нехай об'єкт Q виражений числовим кодом, визначається набором основ $\{m_i\}$ СЗК $(i = \overline{1, n+k})$. Час виконання арифметичних операцій та точність рішення залежить від кількості інформаційних основ n, а надійність (достовірність) обчислень залежить від кількості контрольних основ k. Нехай у процесі обчислень виникла потреба підвищити надійність обчислень. У цьому випадку, в реальному часі, відбувається перерозподіл основ СЗК $(i = \overline{1, n' + k'})$, при цьому $n' < n$, а $k' > k$ та $n + k = n' + k' = const.$ У цьому випадку зменшується точність обчислень та підвищується швидкодія виконання арифметичних операцій, що визначаються кількістю інформаційних основ n'.

Якщо виникла необхідність на окремій ділянці програми, що обчислюється, збільшити точність рішення, то перерозподіл програми відбувається наступним чином: $i = \overline{1, n''+ k''}$ $(n + k = n' + k' = const)$. У цьому випадку при підвищенні точності обчислень $(n'' > n)$ зменшується їх надійність

(достовірність обчислень) та швидкодія (користувацька продуктивність) $(k'' > k)$ рішення даної задачі.

Зазначимо, що методи організації обмінних операцій у ПСЧ (наприклад, змінне масштабування і т.д.) не мають тієї гнучкості та універсальності, як методи, що забезпечують обмінні операції в СЗК. Крім цього, при побудові та дослідженні наведених надійнісних моделей пристроїв обробки інформації в СЗК встановлено, що при застосуванні кодів СЗК одночасно притаманні різні види резервування: структурне, інформаційне, функціональне, навантажувальне та тимчасове.

Дійсно, структурне резервування проявляється при побудові обчислювальної системи на основі набору незалежних і працюючих паралельно в часі обчислювальних трактів по відповідним основам m_i. У цьому випадку дані обчислювальні тракти $(i = \overline{1, n})$ відіграють роль основних елементів резерву, а тракти по основам $m_j = (j = \overline{n+1, n+k})$ – роль резервних елементів. Крім того, в упорядкованій $(m_i < m_i + 1)$ СЗК основи $m_j (j = \overline{n+1, k})$ відіграють роль контрольних трактів, інформація яких дає можливість організувати процес виявлення та виправлення помилок. У цьому аспекті проявляється також інформаційне резервування. Крім цього було показано, що при дотриманні умови $m_j \geq \prod_{i=1}^{r} m_{k_i}$ проявляється роль функціонального резервування, тобто. можливість одного контрольного тракту взяти на себе функції до r інформаційних обчислювальних трактів, що відмовили. Очевидно, що при побудові інформаційно-управляючих систем на принципах переробки інформації в СЗК досягається висока надійність і живучість цих систем.

1.4. Вибір та обґрунтування показника для оцінки швидкодії програмно-апаратних систем і комплексів з елементами штучного інтелекту

Відомо, що найважливішим параметром продуктивності програмно-апаратних систем і комплексів з елементами штучного інтелекту є її швидкодія, яка визначає швидкість, з якою може відбуватися обробка інформації [11, 12, 15]. Швидкодія і час виконання операцій зв'язані між собою співвідношенням (1.1):

$$V = \frac{1}{\tau}, \tag{1.1}$$

де V – швидкість виконання операцій, яка вимірюється кількістю операцій в секунду;

τ – час виконання однієї операції.

Швидкодія характеризується або числом операцій V, які виконуються за секунду, або часом виконання однієї операції $\tau = 1/V$.

Говорячи про швидкодію програмно-апаратних систем і комплексів з елементами штучного інтелекту, введемо дві різноманітні характеристики – номінальну (пікову) швидкодію V_n, та середню швидкодію $V_{ср}$.

Швидкодія програмно-апаратних систем і комплексів з елементами штучного інтелекту складається з швидкодії процесора і часу звертання до ОЗП (оперативний запам'ятовуючий пристрій). В загальному випадку швидкодія програмно-апаратних систем і комплексів з елементами штучного інтелекту суттєво відрізняється для різних процесорних операцій, які відрізняються між собою числом звернень до ОЗП або регістрам загального призначення, алгоритмами обробки, вхідними даними. Тому для характеристики швидкодії програмно-апаратних систем і комплексів з елементами штучного інтелекту можна ввести поняття номінальної швидкодії програмно-апаратних систем і

комплексів з елементами штучного інтелекту.

Номінальною (піковою) швидкодію будемо називати сумарну швидкодію всіх складових частин програмно-апаратних систем і комплексів з елементами штучного інтелекту. Номінальна швидкодія програмно-апаратних систем і комплексів з елементами штучного інтелекту визначається в векторній формі значеннями $(V_1, V_2, ..., V_M)$ або $(\tau_1, \tau_2, ..., \tau_M)$, де $V_1, V_2, ..., V_M$ середня кількість операцій типа $1, 2, .., M$, які виконуються за секунду; $\tau_1, \tau_2, ..., \tau_M$ – середній час виконання відповідних операцій. Набір $(V_1, V_2, ..., V_M)$ або $(\tau_1, \tau_2, ..., \tau_M)$ – характеризує номінальну швидкодію програмно-апаратних систем і комплексів з елементами штучного інтелекту, тобто потенційні можливості без урахування частоти використання різноманітних операцій і простоїв через відмову обладнання.

Номінальна швидкодія $(V_1, V_2, ..., V_M)$ визначає швидкість виконання усіх без виключення операцій – процесорних операцій та операцій вводу–виводу. Маючи на увазі, що програмно-апаратні системи і комплекси з елементами штучного інтелекту можуть комплектуватися різноманітними наборами зовнішніх пристроїв, номінальна швидкодія частіше усього характеризується швидкістю $(V_1, V_2, ..., V_M)$ або часом $(\tau_1, \tau_2, ..., \tau_M)$ виконання тільки процесорних операцій.

Розмірність M наборів $(V_1, V_2, ..., V_M)$ та $(\tau_1, \tau_2, ..., \tau_M)$ дуже велика – порядку 10^2, що ускладнює порівняння швидкодії різноманітних програмно-апаратних систем і комплексів з елементами штучного інтелекту. Процедура порівняння стає простою, якщо швидкодію оцінювати скалярною величиною – середньою швидкодією. Середня швидкодія (операція/с) визначається як величина, зворотна математичному очікуванню тривалості операції (1.2),

$$V_{ср} = \frac{1}{\sum_{i=1}^{M} p_i \tau_i}, \qquad (1.2)$$

де p_i – вірогідність виконання операції $i = 1, 2, ..., M$ і відповідно $\sum_{i=1}^{M} p_i = 1$; τ_i – середній час виконання операції. Вірогідності $p_1, ..., p_i, ..., p_M$ характеризують долю операцій кожного найменування, які виконуються при рішенні однієї задачі або класу задач. Таким чином, середня швидкодія залежить від номінальної швидкодії, тобто від технічних характеристик програмно-апаратних систем і комплексів з елементами штучного інтелекту, а також від класу задач, по відношення до якого визначається середня швидкодія. Якщо два класу задач відрізняються складом виконуваних операцій, то середня швидкодія, визначена для одного класу задач, відрізняється від середньої швидкодії для іншого класу задач.

Для того, щоб використовувати середню швидкодію для порівняння характеристик різноманітних програмно-апаратних систем і комплексів з елементами штучного інтелекту, необхідно прийняти угоду о класі задач, по відношенню до якого оцінюється середня швидкодія, тобто встановити значення ймовірностей $p_1, ..., p_M$.

Як вище зазначено, в складі програмно-апаратних систем і комплексів з елементами штучного інтелекту використовуються операції, що складаються з команд, різних за тривалістю виконання. У зв'язку з цим швидкодія програмно-апаратних систем і комплексів з елементами штучного інтелекту характеризується або номінальною (піковою) швидкодією, коли оцінюється виконання операцій типу «реєстр-регістр», що складаються з найкоротших команд процесора, або середньою швидкодією при виконанні деякої еталонної суміші різних операцій.

Наприклад, для 64-розрядного процесора Intel Itanium 1,0 ГГц найменший щонайменше час виконання команди в цілочисловому арифметико-логічному пристрої становить 1 нс (1 такт роботи процесора), що визначає теоретично граничну швидкодію роботи всієї ЕОМ, що дорівнює 1 мільярд команд за 1 секунду. При цьому на наборах тестів середня швидкодія значно гірша – близько 3 мільйонів операцій на секунду, де у їх складі використовуються

команди різної тривалості, що вимагають обробки у різних блоках та вузлах процесора та ЕОМ в цілому.

Тому, як можемо бачити, реальна швидкодія, яка входить в оцінку користувальницької продуктивності, буде відмінною для різних задач або різних класів задач.

Відношення реальної швидкодії V_p до номінальної швидкодії V_n будемо називати ефективність програмно-апаратних систем і комплексів з елементами штучного інтелекту $E = V_p / V_n$.

В даний час прийнятими одиницями виміру швидкодії є [12-14]:

– Mega Instruction Per Second (MIPS) – мільйон операцій на секунду над числами з фіксованою точкою (комою);

– Mega FLoating Operations Per Second (MFLOPS) – мільйон операцій над числами з плаваючою точкою (комою).

– Giga FLoating Operations Per Second (GFLOPS) – мільярд операцій над числами з плаваючою точкою (комою).

– Tera FLoating Operations Per Second (TFLOPS) – трильйон операцій над числами з плаваючою точкою (комою).

Номінальна швидкодія процесора (точніше системи процесор – оперативна пам'ять) залежить від наступних факторів:

1. швидкодії елементної бази процесора та оперативної пам'яті, тобто часу перемикання сигналу в напівпровідникових інтегральних схемах, а швидкодія схем найбільшою мірою залежить від використовуваної технології. Рівень технологій визначає, по-перше, мінімальний розмір напівпровідникових елементів, зменшення якого підвищує швидкодію елементів, по-друге, граничну кількість елементів в одній інтегральній схемі, збільшення числа яких дозволяє створювати більш високопродуктивні електронні структури;

2. структурної організації процесора, спрямованої на підвищення швидкодії за рахунок суміщення у часі обробки потоку команд та виконання операцій;

3. архітектури комп'ютера і у першу чергу системи команд процесора, тобто складу операцій, що реалізуються процесором, способів адресації та форм подання даних.

Номінальна швидкодія $V_н$ характеризує потенційні можливості пристроїв, що становлять комп'ютер: процесора, зовнішніх пристроїв, пристроїв введення-виводу і т.д.

Як зазначалося вище, найважливішими чинниками, що впливають на швидкодію є насамперед такі чинники, як: тактова частота роботи процесора, число команд програми (задачі, алгоритму) і число тактів для виконання однієї команди (середній час виконання однієї команди). У свою чергу, команда складається з послідовності арифметичних та інших операцій. У загальному випадку можна сказати, що кількісно швидкодія залежить від тактової частоти роботи процесора і від часу реалізації арифметичних та інших операцій, що входять до складу команди програми.

Надалі у роботі для порівняльного аналізу швидкодії процесорів (для кількісного зіставлення швидкодії різних типів програмно-апаратних систем і комплексів з елементами штучного інтелекту використовується аналітичне співвідношення (1.3)

$$K_{eff}^{r} = \frac{T^{(ПСЧ)}}{T^{(СЗК)}}, \qquad (1.3)$$

де:

$T^{(ПСЧ)}$ час рішення програмно-апаратних систем і комплексів з елементами штучного інтелекту в ПСЧ однієї конкретної задачі;

$T^{(СЗК)}$ час вирішення програмно-апаратних систем і комплексів з елементами штучного інтелекту в СЗК тієї ж однакової задачі.

У цьому випадку кількісне значення безрозмірного коефіцієнта ефективності K_{eff}^{r} залежить тільки від часу реалізації в ПСЧ і СЗК

арифметичних та інших операцій, що входять до складу команд розв'язуваної задачі.

Використання СЗК в програмно-апаратних системах і комплексах з елементами штучного інтелекту дозволяє підвищити швидкодію реалізації цілих арифметичних операцій [39-40]. У той же час потрібне практичне підтвердження ефективності застосування СЗК для підвищення швидкодії програмно-апаратних систем і комплексів з елементами штучного інтелекту при обробці цілочислових даних. Тому задача розрахунку та порівняльного аналізу швидкодії програмно-апаратних систем і комплексів з елементами штучного інтелекту, що функціонують в СЗК та у звичайній двійковій ПСЧ, для конкретних обчислювальних алгоритмів є актуальною та практично важливою.

У четвертому розділі дисертації будуть розглянуті приклади розрахунку та порівняльного аналізу швидкодії програмно-апаратних систем і комплексів з елементами штучного інтелекту у ПСЧ та СЗК.

1.5. Постановка задачі дослідження дисертації

Пошук альтернативних способів підвищення швидкодії обробки даних при вирішення обчислювальних задач призводить до зростання інтересу до застосування СЗК у суміжних галузях науки і техніки. Цей інтерес викликаний насамперед такими обставинами:

- у нашій країні і за кордоном активно публікується велика кількість науково-теоретичних праць, присвячених теорії та практиці розробки комп'ютерних систем і компонентів, які мають високу швидкодію, надійність, довговічність та стійкість до відмов, що функціонують у СЗК. Особлива увага приділяється розробці теорії забезпечення відмовостійкості та класифікації програмно-апаратних систем і комплексів з елементами штучного інтелекту, які функціонують на основі використання непозиційної системи числення в залишкових класах.

- банківські установи виявляють значний інтерес до напрямку, де

потрібно здійснювати обробку великих обсягів даних в реальному часі з високою надійністю та достовірністю. Це передбачає використання високопродуктивних обчислювальних ресурсів для здійснення надійних обчислень з можливістю автоматичної корекції помилок, що є характерною для корекційних кодів у СЗК;

- масовим поширенням процесорів мобільних пристроїв, які потребують швидкої обробки даних при мінімальному споживанні енергії; використання СЗК при виконанні арифметичних операцій додавання та множення чисел забезпечує високу швидкодію завдяки уникненню міжрозрядних переносів під час виконання арифметичних операцій. Такий підхід дозволяє значно знизити витрати енергії при експлуатації мобільних пристроїв.

- підвищення щільності розташування елементів на кристалі не завжди дозволяє здійснити якісне та повне тестування комп'ютерних компонентів. У таких випадках стає важливим забезпечення надійності функціонування програмно-апаратних систем та комплексів з елементами штучного інтелекту. Попередні дослідження підтверджують, що за допомогою СЗК можна забезпечити надійне функціонування програмно-апаратних систем та комплексів з штучним інтелектом реального часу.

- для ефективного виконання значної кількості операцій над багатовимірними числовими структурами в програмно-апаратних системах і комплексах з елементами штучного інтелекту у реальному часі, потрібно використовувати спеціалізовані операційні пристрої, які мають високу швидкість виконання цілих операцій додавання та множення. Це особливо актуально для таких задач, як множення матриць, скалярне множення векторів, а також для перетворення Фур'є та інших подібних операцій;

- широке застосування мікроелектроніки у всіх аспектах людського життя значно підвищує актуальність і важливість таких науково-практичних здач, як обробка цифрових сигналів і зображень, розпізнавання образів, криптографічні перетворення, а також обробка та зберігання багатовимірної інформації. Ця обставина вимагає величезних обчислювальних потужностей, які виходять за

межі можливостей програмно-апаратних систем та комплексів з елементами штучним інтелектом, що працюють у двійковій позиційній системі числення.

- у літературі зазначають, що існуючі та перспективні комп'ютерні системи та компоненти, що функціонують у ПСЧ, які використовуються в обчислювальних системах у реальному часі, не здатні забезпечити необхідну швидкодію, надійність і стійкість до відмов при обробці великих обсягів даних;

- фахівцям у галузі обчислювальної техніки очевидно, що мікроелектроніка, на якій ґрунтується сучасний рівень технологій, досягла свого піку розвитку; перспективні шляхи подальшого розвитку мікроелектроніки, що йдуть на зміну наноелектроніки, такі, як, наприклад, молекулярна і біологічна електроніка, мікромеханіка, оптичні, оптоелектронні та фотонні КС та інші екзотичні напрямки вдосконалення існуючих КС, поки що далекі від широкого промислового виробництва та практичного використання.

Кодування в СЗК дозволяє синтезувати програмно-апаратні системи і комплекси з елементами штучного інтелекту, в якій обробка всіх залишків $\{p_i\}$ числа проводиться паралельно в часі. Структурна схема програмно-апаратних систем і комплексів з елементами штучного інтелекту в СЗК представляє собою набір процесорів, що функціонують незалежно один від одного і паралельно в часі, причому кожна за своїм певним модулем m_i. У цьому випадку пристрої введення та виведення вирішують також відповідно задачі перетворення вхідної інформації програмно-апаратних систем і комплексів з елементами штучного інтелекту з позиційного коду в код СЗК і навпаки.

На підставі вимог, що пред'являються до програмно-апаратних систем і комплексів з елементами штучного інтелекту, ґрунтуючись на перспективних концепціях розвитку комп'ютерних засобів швидкої обробки даних, а також у зв'язку з невирішеністю задачі створення швидкодіючих комп'ютерних засобів обробки інформації на основі використання ПСЧ, з одного боку, та з розвиваючимися тенденціями, використання непозиційних кодових структур у СЗК, з іншого боку, тема дисертаційної роботи, присвячена розробці методів та

цифрових компонентів програмно-апаратних систем і комплексів з елементами штучного інтелекту, представлених у СЗК, є дуже важливою, актуальною як на даному етапі, так і для подальшої перспективи розвитку обчислювальної техніки та штучного інтелекту.

Для досягнення даної мети як рішення поставленого наукового завдання в цілому, були сформульовані ряд ***задач***. До їхнього числа належать.

1. Аналіз проблем обробки інформації програмно-апаратними системами і комплексами з елементами штучного інтелекту в реальному часі.
2. Удосконалення методу додавання та віднімання залишків чисел по модулю СЗК.
3. Удосконалення методу табличної реалізації множення двох залишків чисел за рахунок можливості виконання операції в комплексній області.
4. Удосконалення математичної моделі методу піднесення залишків цілих чисел до довільного ступеня натурального числа в СЗК.
5. Практичне підтвердження працездатності та вірогідності розроблених моделі і методів.

Висновки по розділу 1

В першому розділі аналізуються проблеми побудови програмно-апаратних систем і комплексів з елементами штучного інтелекту. Проведений аналіз можливостей програмно-апаратних систем і комплексів з елементами штучного інтелекту свідчить про те, що вони не задовольняють збільшеним вимогам до швидкості обробки інформації, що обумовлює актуальність дослідження нових моделей і методів.

Проаналізовано сучасний стан та напрями підвищення швидкодії програмно-апаратних систем і комплексів з елементами штучного інтелекту, за рахунок застосування спеціальних технологічних та архітектурних рішень, а також математичних методів для їх застосування в ШІ. Відмічено, що

застосування паралельної обробки даних на основі СЗК дозволяє значно підвищити швидкодію операцій обробки даних.

За результатами проведеного аналізу теоретичних основ СЗК визначено основні її переваги над позиційними системами числення (незалежність залишків, що дає можливість розпаралелення процесу обчислень; рівноправність залишків, що дає можливість підвищити відмовостійкість програмно-апаратних систем і комплексів з елементами штучного інтелекту та малорозрядність залишків, що дає можливість підвищити швидкодію програмно-апаратних систем і комплексів з елементами штучного інтелекту) та її недоліки (труднощі при виконанні операції порівняння та ділення чисел, визначення переповнення допустимого діапазону), обґрунтовано необхідність використання СЗК в операційних пристроях програмно-апаратних систем і комплексів з елементами штучного інтелекту.

Формулюються задачі дисертаційного дослідження: вдосконалення методу додавання та віднімання залишків чисел по модулю СЗК; вдосконалення методу табличної реалізації множення двох залишків чисел за рахунок можливості виконання операції в комплексній області; розробка математичної моделі методу піднесення залишків цілих чисел до довільного ступеня натурального числа за модулем СЗК; практичне підтвердження працездатності та вірогідності розроблених моделі і методів. Які будуть вирішуватись в наступних розділах дисертаційної роботи.

РОЗДІЛ 2. ВДОСКОНАЛЕННЯ МЕТОДІВ ВИКОНАННЯ АРИФМЕТИЧНИХ ОПЕРАЦІЙ ДОДАВАННЯ ТА ВІДНІМАННЯ ЗАЛИШКІВ ЧИСЕЛ ПО МОДУЛЮ В СЗК

2.1. Дослідження методів додавання та віднімання залишків чисел по модулю в СЗК

Виконання цілочисельних арифметичних операцій додавання, віднімання та множення чисел $X=(x_1,x_2...x_n)$ та $Y=(y_1,y_2...y_n)$ в системі залишкових класів здійснюється шляхом виконання відповідних арифметичних операцій над залишками x_i і y_i чисел по відповідним модулям (основами) $m_i \quad (i=\overline{1,n})$ СЗК незалежно і паралельно у часі по кожному з *n* основ СЗК.

Розглянемо правила виконання арифметичної операції додавання в СЗК [41-42]. Нехай операнди *X* і *Y* представлені відповідно залишками x_i і y_i по модулям (основами) $m_i \quad (i=\overline{1,n})$ СЗК. В діапазоні $[0,M)$, де $M=\prod_{i=1}^{n} m_i$, набір залишків $x_i(y_i)$ однозначно визначають число $X(Y)$.

Результат операції додавання чисел $X+Y$ в СЗК представляється залишками $z_i \quad (i=\overline{1,n})$ по тим же модулям (основам) m_i СЗК, тобто

$$X=(x_1,x_2...x_n),$$

$$Y=(y_1,y_2...y_n),$$

$$X+Y=(z_1,z_2...z_n),$$

при цьому має місце співвідношення:

$$X<M, \quad Y<M, \quad X+Y<M.$$

Також стверджується, що $z_i=(x_i+y_i)\bmod m_i$.

Наведемо приклади виконання арифметичних операцій для СЗК, яка задана основами $m_1=2_{10}=10_2, \quad m_2=3_{10}=11_2, \quad m_2=5_{10}=101_2$. Діапазон системи визначиться як $[0;M)$, де $M=2\cdot3\cdot5=30$. Так як в КС кожна цифра

кодується двійковим кодом, набір кодових слів для даної СЗК має наступний вигляд представлений в табл. 2.1.

Таблиця 2.1

Таблиця кодових слів в СЗК

X (Y)	$X(Y)$ в СЗК			X (Y)	$X(Y)$ в СЗК		
	$m_1 = 2_{10} = =10_2$	$m_2 = 3_{10} = =11_2$	$m_3 = 5_{10} = =101_2$		$m_1 = 2_{10} = =10_2$	$m_2 = 3_{10} = =11_2$	$m_3 = 5_{10} = =101_2$
0	$0_{10}=0_2$	$0_{10}=00_2$	$0_{10}=000_2$	15	$1_{10}=1_2$	$0_{10}=00_2$	$0_{10}=000_2$
1	$1_{10}=1_2$	$1_{10}=01_2$	$1_{10}=001_2$	16	$0_{10}=0_2$	$1_{10}=01_2$	$1_{10}=001_2$
2	$0_{10}=0_2$	$2_{10}=10_2$	$2_{10}=010_2$	17	$1_{10}=1_2$	$2_{10}=10_2$	$2_{10}=010_2$
3	$1_{10}=1_2$	$0_{10}=00_2$	$3_{10}=011_2$	18	$0_{10}=0_2$	$0_{10}=00_2$	$3_{10}=011_2$
4	$0_{10}=0_2$	$1_{10}=01_2$	$4_{10}=100_2$	19	$1_{10}=1_2$	$1_{10}=01_2$	$4_{10}=100_2$
5	$1_{10}=1_2$	$2_{10}=10_2$	$0_{10}=000_2$	20	$0_{10}=0_2$	$2_{10}=10_2$	$0_{10}=000_2$
6	$0_{10}=0_2$	$0_{10}=00_2$	$1_{10}=001_2$	21	$1_{10}=1_2$	$0_{10}=00_2$	$1_{10}=001_2$
7	$1_{10}=1_2$	$1_{10}=01_2$	$2_{10}=010_2$	22	$0_{10}=0_2$	$1_{10}=01_2$	$2_{10}=010_2$
8	$0_{10}=0_2$	$2_{10}=10_2$	$3_{10}=011_2$	23	$1_{10}=1_2$	$2_{10}=10_2$	$3_{10}=011_2$
9	$1_{10}=1_2$	$0_{10}=00_2$	$4_{10}=100_2$	24	$0_{10}=0_2$	$0_{10}=00_2$	$4_{10}=100_2$
10	$0_{10}=0_2$	$1_{10}=01_2$	$0_{10}=000_2$	25	$1_{10}=1_2$	$1_{10}=01_2$	$0_{10}=000_2$
11	$1_{10}=1_2$	$2_{10}=10_2$	$1_{10}=001_2$	26	$0_{10}=0_2$	$2_{10}=10_2$	$1_{10}=001_2$
12	$0_{10}=0_2$	$0_{10}=00_2$	$2_{10}=010_2$	27	$1_{10}=1_2$	$0_{10}=00_2$	$2_{10}=010_2$
13	$1_{10}=1_2$	$1_{10}=01_2$	$3_{10}=011_2$	28	$0_{10}=0_2$	$1_{10}=01_2$	$3_{10}=011_2$
14	$0_{10}=0_2$	$2_{10}=10_2$	$4_{10}=100_2$	29	$1_{10}=1_2$	$2_{10}=10_2$	$4_{10}=100_2$

Приклад 2.1. Додати два числа $X = 8$ і $Y = 13$.

В СЗК числа X і Y будуть представлені як (див. таблицю 2.1)

$$X = 8 = (0, 10, 011),$$

$$Y = 13 = (1, 01, 011).$$

Результат операції $X + Y$ визначається як

	$m_1 = 2_{10} = 10_2$	$m_2 = 3_{10} = 11_2$	$m_3 = 5_{10} = 101_2$
$X =$	0	10	011
$+$	$+$	$+$	$+$
$Y =$	1	01	011
$Z =$	1	00	001

де $Z = (z_1, z_2, z_3) = (1, 00, 001)$ при цьому $z_1 = (0+1) \bmod 2 = 1$, $z_2 = (10+01) \bmod 3 = 00$, $z_3 = (011+011) \bmod 5 = 001$.

Перевірка: $(8+13) \bmod 30 = 21$, що в СЗК представляється як (1, 10, 011).

Приклад 2.2. Знайти різницю чисел $X = 8$ і $Y = 13$.

Результат операції $X - Y$ визначається як

	$m_1 = 2_{10} = 10_2$	$m_2 = 3_{10} = 11_2$	$m_3 = 5_{10} = 101_2$
$X =$	0	10	011
-	-	-	-
$Y =$	1	01	011
$Z =$	1	01	000

де $Z = (z_1, z_2, z_3) = (1, 01, 000)$ при цьому $z_1 = (0-1) \bmod 2 = 1$, $z_2 = (10-01) \bmod 3 = 01$, $z_3 = (011+011) \bmod 5 = 000$.

Перевірка: $(8-13) \bmod 30 = 25$, що в СЗК представляється як (1, 01, 000).

Малозрядність залишків x_i та y_i в представлені чисел, що додаються в СЗК дає можливість здійснити модульну операцію додавання $(x_i + y_i) \bmod m_i$ на основі застосування двійкових модульних суматорів. Для упорядкованої $(m_i < m_{i+1})$ СЗК виконання операції додавання чисел визначається часом, необхідним для отримання результату операції $(x_n + y_n) \bmod m_n$ за найбільшою m_n основою СЗК. Один з методів реалізації модульної операції додавання заснований на використанні двійкових суматорів. Даний підхід пропонує широкий вибір варіантів можливої реалізації внутрішньої структури такого суматора. Це дозволяє в повній мірі використовувати наявний досвід

проектування двійкових суматорів.

Якщо залишки x_i та y_i чисел $X=(x_1,x_2...x_n)$ та $Y=(y_1,y_2...y_n)$ в СЗК представлені у двійкової ПСЧ, то суматор залишків x_i та y_i за модулем m_i. представляє послідовну сукупність із $k=[\log_2(m_i-1)]+1$ двійкових однозарядних суматорів (ДОС), об'єднаних між собою зв'язками подібно до зв'язків між позиційними двійковими суматорами.

Відомо [42-43], що структура k-розрядного двійкового суматора по модулю $M=2^k-1$ має вигляд зображений на рис. 2.1. Для того, щоб створити структуру суматора для реалізації операції додавання залишків чисел за довільним модулем m_i СЗК на основі суматора по модулю $M=2^k-1$, необхідно в структурі суматора по модулю $M=2^k-1$ додати додаткові зв'язки $X_{\downarrow i\uparrow j}$, де $X_{\downarrow i\uparrow j}$ позначає зв'язок між входом i-го та виходом j-го ДОС, при цьому виконується нерівність $(j>i)$ (рис. 2.2).

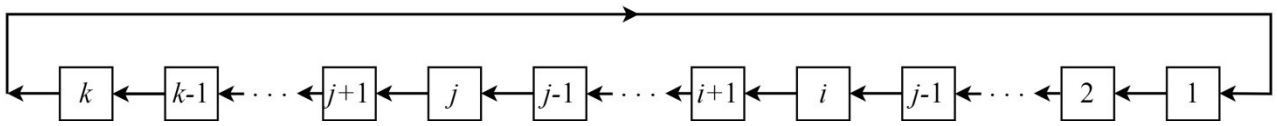

Рис. 2.1 Схема розташування та нумерація двійкових однорозрядних суматорів у суматорі за модулем $M=2^k-1$

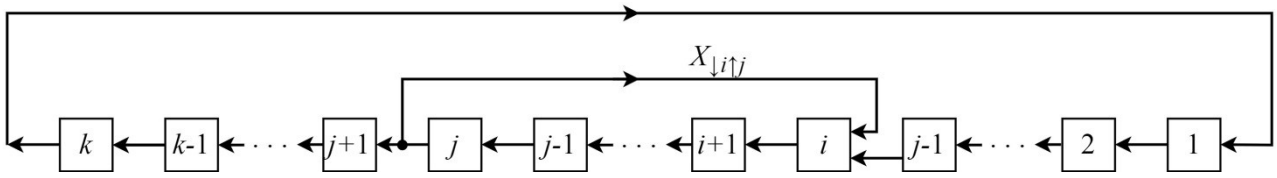

Рис. 2.2 Схема двійкового суматора з додатковим зв'язком $X_{\downarrow i\uparrow j}$

2.2 Аналіз впливу додаткових зв'язків суматора по модулю M на величину вмісту суматора

Розглянемо вплив одного додаткового зв'язку $X_{\downarrow i\uparrow j}$, установленого між

входом i-го та виходом j-го ДОС модульного суматора $M = 2^k - 1$ (див. рис. 2.2) на величину Q_R вихідного вмісту суматора [44]. Покажемо, що число $R = \{r_i\}$, $i = \overline{1,k}$, яке є вмістом величини Q_R суматора, у разі введення одного додаткового зв'язку $X_{\downarrow i \uparrow j}$ зменшується на величину $\Delta Q_R = 2^{i-j-2} \cdot \sum_{l=j+1}^{k} 2^l \cdot r_l$. Зазначимо, що введення додаткового зв'язку $X_{\downarrow_i \uparrow_j}$ переводить числення чисел із двійкової системи числення, у якій працює модульний суматор M, у поліадичну систему числення з основами $\rho_1, \rho_2 ... \rho_q$ та модулем $M = \rho_1 \cdot \rho_2 \cdot ... \cdot \rho_q - 1$. У цьому випадку вміст числа $R = \{r_i\}$, $i = \overline{1,q}$, визначається таким чином (2.1):

$$Q_R = \sum_{l=1}^{q} r_l \cdot \prod_{i=1}^{l-1} \rho_i = r_1 \cdot \rho_2 \cdot \rho_3 \cdot ... \cdot \rho_q + r_2 \cdot \rho_3 \cdot \rho_4 \cdot ... \cdot \rho_q + ... \\ ... + r_{q-2} \cdot \rho_{q-1} \rho_q + r_{q-1} \cdot \rho_q + r_q \quad (2.1)$$

Співвідношення (2.1) можна представити у такому вигляді (2.2):

$$Q_R = r_1 \cdot \prod_{i=2}^{q} \rho_i + r_2 \cdot \prod_{i=3}^{q} \rho_i + ... + r_{q-2} \cdot \prod_{i=q-1}^{q} \rho_i + r_{q-1} \prod_{i=q}^{q} \rho_i + r_q . \quad (2.2)$$

У двійковій системі числення $(\rho_1 = \rho_2 = ... = \rho_q = 2)$ вираз (2.2) набуде такого вигляду:

$$Q_R = \sum_{l=1}^{q} r_l \cdot 2^{q-l} = r_1 \cdot 2^{q-1} + r_2 \cdot 2^{q-2} + ... + r_{q-1} \cdot 2 + r_q . \quad (2.3)$$

У разі відсутності в суматорі додаткових зв'язків $X_{\downarrow i \uparrow j}$ величина Q_R вмісту суматора визначається як

$$Q_R = \sum_{l=1}^{k} r_l \cdot p_l \,, \tag{2.4}$$

де величина r_l в l-м розряді вмісту суматора може набувати значеня: $r_l = 0$ або $r_l = 1$, значення p_l є вагою l-го розряду суматора (див. рис. 2.1).

Якщо є додатковий зв'язок $X_{\downarrow i \uparrow j}$, що об'єднує ДОС із номерами від i до j в єдиний (узагальнений) розряд суматора по модулю

$$\rho_{ij} = 2^{j-(i-1)} - 1 = 2^{j-i+1} - 1, \tag{2.5}$$

то, вага p_l кожного розряду суматора з номерами від $(i-1)$-го до першого (молодшого розряду суматора) визначається як $p_l = 2^{l-1}$ $(l = \overline{1, i-1})$ (рис. 2.3).

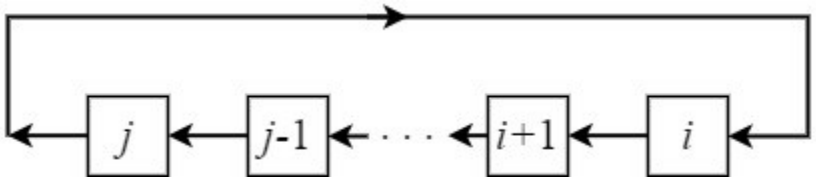

Рис. 2.3 Схема узагальненого $(j-i+1)$-го розряду суматора по модулю ρ_{ij}

Виходячи зі структури суматора по модулю з додатковим зв'язком (рис. 2.4) вага розрядів суматора з номерами від $(j+1)$-го до k-го (старшого розряду суматора) визначається виразом

$$p_l = 2^{i-1} \cdot \rho_{ij} \cdot 2^{l-j-1} \tag{2.6}$$

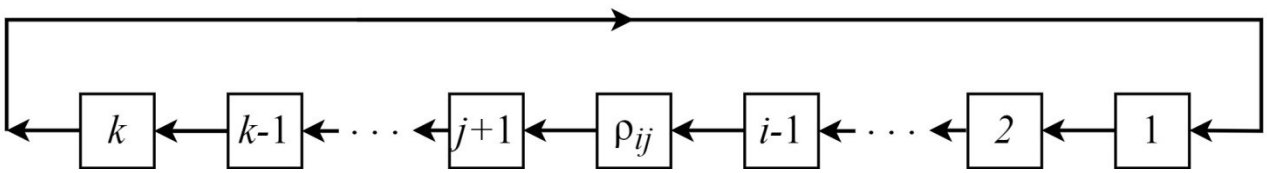

Рис. 2.4 Еквівалентна схема суматора за модулем із додатковим зв'язком $X_{\downarrow i \uparrow j}$

З урахуванням співвідношення (2.5) вираз (2.6) можна представити у вигляді

$$p_l = 2^{l+i-j-2} \cdot \rho_{ij} = 2^{l+i-j-2} \cdot (2^{j-i+1} - 1) = 2^{l-1} - 2^{l+i-j-2} \tag{2.7}$$

Вираз (2.7) для будь-якого розряду суматора можна представить таким чином:

$$p_l = 2^{l-1} - 2^{l+i-j-2} \cdot \Omega_{lj}, \tag{2.8}$$

де число Ω_{lj} може набувати двох значень:

$$\Omega_{lj} = \begin{cases} 1, & \text{якщо } l > j, \\ 0, & \text{якщо } l \le j. \end{cases}$$

Відповідно до (2.8) величина Q_R числа *R* дорівнює такій різниці:

$$Q_R = \sum_{l=1}^{k} r_l \cdot 2^{l-1} - \sum_{l=j+1}^{k} r_l \cdot 2^{l+i-j-2}. \tag{2.9}$$

Зі співвідношення (2.9) очевидно, що ведення в суматор одного додаткового зв'язку виду $X_{\downarrow i \uparrow j}$ зменшує його вміст Q_R на величину, що дорівнює значенню

$$\Delta Q_R = \sum_{l=j+1}^{k} r_l \cdot 2^{l+i-j-2} = 2^{i-j-2} \cdot \sum_{l=j+1}^{k} r_l \cdot 2^{l}.$$

Таким чином, при введенні одного додаткового зв'язку $X_{\downarrow i \uparrow j}$ вихідний вміст суматора по модулю M зменшується на величину ΔQ_R:

$$\Delta Q_R = 2^{i-j-2} \cdot \sum_{l=j+1}^{k} r_l \cdot 2^l \ . \qquad (2.10)$$

Слід особливо зазначити, що при введеному додатковому зв'язку вигляду $X_{\downarrow i \uparrow j}$ значення величини модуля $M = 2^k - 1$ вихідного суматора зменшується на величину

$$\Delta M = 2^{i-j-2} \cdot \sum_{l=j+1}^{k} S_l \cdot 2^l \ , \qquad (2.11)$$

де S_l - значення *l*-го розряду числа, що міститься в суматорі. При цьому, вочевидь, що, діапазон представлених чисел по модулю M зменшується на величину ΔM . У результаті виникає можливість за рахунок введення в суматор по модулю M певних додаткових зв'язків (або одного додаткового зв'язку) зменшить величину M модуля до необхідного значення модуля m_i СЗК. У цьому випадку має виконуватися наступна умова:

$$m_i = M - \Delta M \ \text{ або } m_i = 2^k - 1 - 2^{i-j-2} \cdot \sum_{l=j+1}^{k} r_l \cdot 2^l . \qquad (2.12)$$

Виходячи з виразу (2.12), чисельне значення заданого модуля СЗК буде визначено набором значень *k, j* та *i*. Таким чином, необхідно визначити значення *k, j* та *i,* як такі, що задовольняють виконання рівності.

Проілюструємо вплив одного додаткового зв'язку прикладами.

Розглянемо чотирьох розрядний суматор по модулю *M* = 11. Відповідно до величини $m_i = 11$ модуля СЗК, кількість *k* ДОС дорівнює чотирьом. При цьому, вихідна структура суматора по модулю $m_i = 11$ без додаткових зв'язків $X_{\downarrow i \uparrow j}$ матиме вигляд представлений на рис. 2.5.

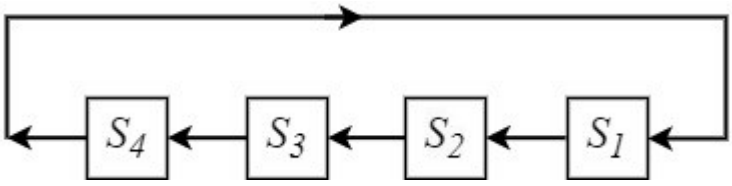

Рис. 2.5 Вихідна структура суматора по модулю $m_i = 11$ без додаткових зв’язків

Варіант 1. Введемо додатковий зв'язок $X_{\downarrow i3 \uparrow j4}$ між виходом четвертого і входом третього розряду. Тоді структура суматора по модулю $m_i = 11$ з одним додатковим зв’язком $X_{\downarrow i3 \uparrow j4}$ матиме вигляд представлений на рис. 2.6.

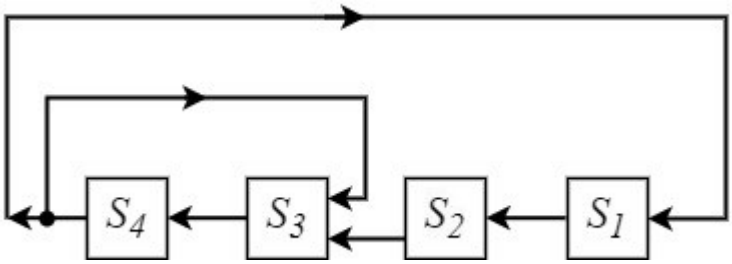

Рис. 2.6 Структура суматора по модулю $m_i = 11$ з одним додатковим зв’язком $X_{\downarrow i3 \uparrow j4}$ з $\Delta Q_r = 0$

Для структури суматора, що представлена на рис. 2.6, визначимо значення модуля $M = m_i$ СЗК. Попередньо розглянемо частину структури (див. рис. 2.7) цього суматора.

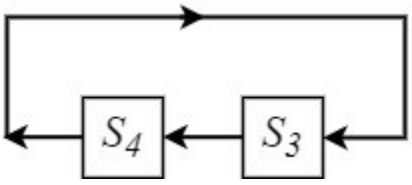

Рис. 2.7 Частина структури суматора по модулю $m_i = 11$

Для цієї частини структури суматора значення модуля M_1 визначиться, як $M_1 = \rho_4 \cdot \rho_3 - 1$. Значення модуля $M = m_i$ СЗК суматора (рис. 2.6) визначиться наступним чином (див. рис. 2.8) $m_i = M_1 \cdot \rho_2 \cdot \rho_1 = (\rho_4 \cdot \rho_3 - 1) \cdot \rho_2 \cdot \rho_1 - 1 = = (2 \cdot 2 - 1) \cdot 2 \cdot 2 - 1 = 11$.

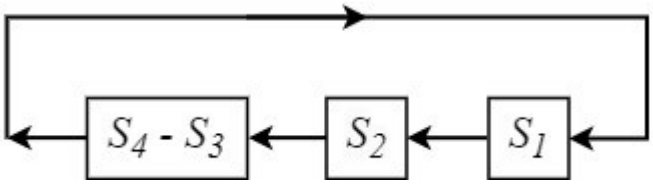

Рис. 2.8 Частина структури суматора по модулю $m_i = 11$

Нехай на суматорі записано число $R = (1, 0, 1, 0)$, яке для цієї конструкції суматора (рис. 2.6) має величину $Q_R = 10$, і співпадає з його величиною в двійковій системі числення, оскільки виконана умова $j = k$, тому величина $\Delta Q_R = 0$.

Розглянемо інші можливі конструкції суматора, який працює по модулю $m_i = 11$.

Варіант 2. Введемо додатковий зв'язок $X_{\downarrow i2 \uparrow j3}$ між виходом третього і входом другого розряду. Тоді структура суматора по модулю $m_i = 11$ з одним додатковим зв'язком $X_{\downarrow i3 \uparrow j4}$ матиме вигляд представлений на рис. 2.9.

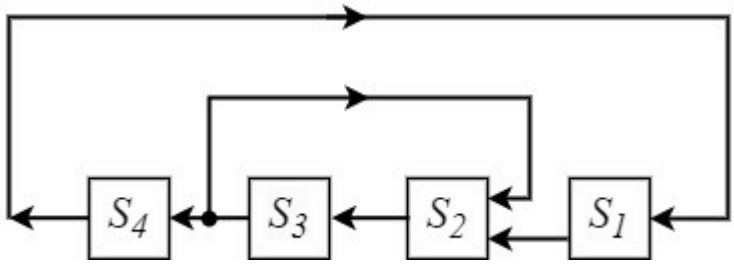

Рис. 2.9 Структура суматора по модулю $m_i = 11$ з одним додатковим зв'язком $X_{\downarrow i2 \uparrow j3}$ з $\Delta Q_R = 2$

Для структури суматора, що представлена на рис. 2.6, визначимо значення модуля $M = m_i$ СЗК. Попередньо розглянемо частину структури (див. рис. 2.10) цього суматора.

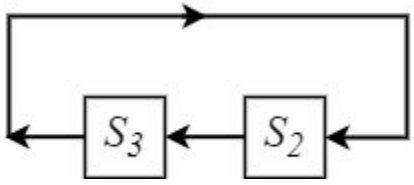

Рис. 2.10 Частина структури суматора за модулем $m_i = 11$

Для цієї частини структури суматора значення модуля M_1 визначиться, як $M_1 = \rho_3 \cdot \rho_2 - 1$. Значення модуля $M = m_i$ СЗК суматора (рис. 2.9) визначиться наступним чином (див. рис. 2.11) $m_i = \rho_4 \cdot M_1 \cdot \rho_1 = \rho_4(\rho_2 \cdot \rho_3 - 1) \cdot \rho_1 - 1 = = 2(2 \cdot 2 - 1) \cdot 2 - 1 = 11$.

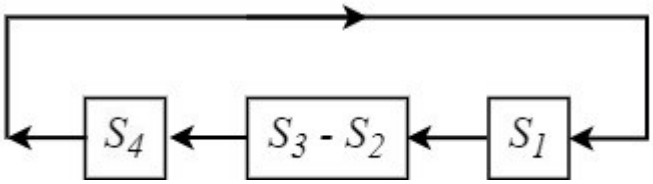

Рис. 2.11 Частина структури суматора за модулем $m_i = 11$

В цій конструкції суматора з одним додатковим зв'язком $X_{\downarrow i2 \uparrow j3}$ (рис. 2.9) величина ΔQ_R визначається як $\Delta Q_R = 2^{2-3-2} \cdot \sum_{l=4}^{k} r_l \cdot 2^l = 2 \cdot r_4$.

Варіант 3. Введемо додатковий зв'язок $X_{\downarrow i1 \uparrow j2}$ між виходом другого і входом першого розряду. Тоді структура суматора по модулю $m_i = 11$ з одним додатковим зв'язком $X_{\downarrow i1 \uparrow j2}$ матиме вигляд представлений на рис. 2.12.

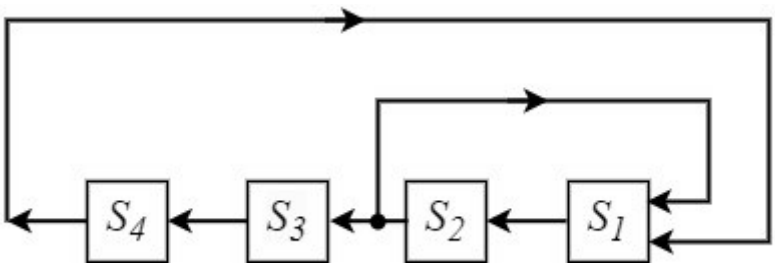

Рис. 2.12 Структура суматора по модулю $m_i = 11$ з одним додатковим зв'язком $X_{\downarrow i1 \uparrow j2}$ з $\Delta Q_R = 2$

Для структури суматора, що представлена на рис. 2.12, визначимо значення модуля $M = m_i$ СЗК. Попередньо розглянемо частину структури (див. рис. 2.13) цього суматора.

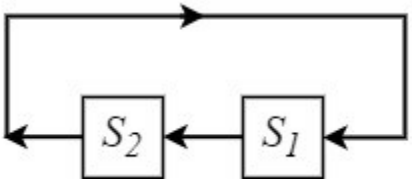

Рис. 2.13 Частина структури суматора за модулем $m_i = 11$

Для цієї частини структури суматора значення модуля M_1 визначиться, як $M_1 = \rho_2 \cdot \rho_1 - 1$. Значення модуля $M = m_i$ СЗК суматора (рис. 2.12) визначиться наступним чином (див. рис. 2.14) $m_i = \rho_4 \cdot \rho_3 \cdot M_1 = \rho_4 \cdot \rho_3(\rho_2 \cdot \rho_1 - 1) - 1 = = 2 \cdot 2 \cdot (2 \cdot 2 - 1) - 1 = 11$.

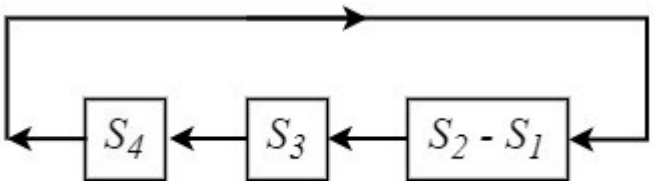

Рис. 2.14 Частина структури суматора за модулем $m_i = 11$

В цій конструкції суматора з одним додатковим зв'язком $X_{\downarrow i1 \uparrow j2}$ (рис. 2.11) величина ΔQ_R визначається як $\Delta Q_R = 2^{1-2-2} \cdot \sum_{l=3}^{k} l_l \cdot 2^l = r_3 + 2 \cdot r_4$.

У конструкціях суматорів з варіанту 2 та 3 отримуємо що $\Delta Q_R = 2$; рівність величини ΔQ_R у другому та третьому варіантах суматорів пояснюється тим, що в заданому числі R прийняли $r_3 = 0$, а тоді отримуємо, що величина ΔQ_R в обох варіантах є однаковою.

Проаналізуємо випадок, коли на суматорі введено s незалежних додаткових зв’язків. Під незалежними додатковими зв’язками будемо розуміти такі додаткові зв’язки, кожен з яких охоплює групу розрядів суматорів, не

охоплених ніяким іншим зв'язком, тобто додаткові зв'язки $X_{\downarrow i\uparrow j}$ і $X_{\downarrow m\uparrow k}$ є незалежними, якщо виконуються наступні умови: $i > k$ або $l > j$. Покажемо, що число $R = \{r_i\}$, $i = \overline{1,k}$, яке є вмістом величини Q_R суматора, у разі введення s незалежних додаткових зв'язків між виходами розрядів з індексом j_t і входами розрядів з індексами i_t, де $t = 1, 2, \ldots, s$, зменшується на величину:

$$\Delta Q_R = \sum_{t=1}^{s} \sum_{l=j_{t+1}}^{j_{t+1}} r_l \sum_{u=1}^{t} (-1)^{u+1} \times \sum_{\substack{g_1, g_2, \ldots, g_u = 1 \\ g_1 \neq g_2 \neq \ldots \neq g_u}}^{t} 2^{\theta_{g_1} + \theta_{g_2} + \ldots + \theta_{g_u} - (u-1)(l-1)}, \qquad (2.13)$$

де $\theta_g = l + i_g - j_g - 2, \quad j_{s+1} = k$.

Тобто, для розрядів з номерами l, які задовільняють умові $1 \le l \le j_1$, вага визначається як $p_l = 2^{l-1}$. Для розрядів у яких $j_1 + 1 \le l \le j_2$, вага дорівнює $p_l = 2^{l-1} - 2^{\theta_1}$, $\theta_1 = l + i_1 - j_1 - 2$. Для розрядів, номера яких $j_2 + 1 \le l \le j_3$, вага визначається як $p_l = 2^{l-1} - 2^{\theta_1} - 2^{\theta_2} + 2^{\theta_1 + \theta_2 - (l-1)}$. Для розрядів, у яких $j_3 + 1 \le l \le j_4$, тоді, як можна побачити, вага дорівнює $p_l = 2^{l-1} - 2^{\theta_1} - 2^{\theta_2} - 2^{\theta_3} + 2^{\theta_1 + \theta_2 - (l-1)} + 2^{\theta_1 + \theta_3 - (l-1)} + 2^{\theta_2 + \theta_3 - (l-1)} - 2^{\theta_1 + \theta_2 + \theta_3 - 2(l-1)}$.

Продовжуючи цей процес для розрядів номера яких дорівнюють умові $j_t + 1 \le l \le j_{t+1}$, отримаємо значення ваги

$$p_l = 2^{l-1} - (2^{\theta_1} + 2^{\theta_2} + \ldots + 2^{\theta_t}) + 2^{\theta_1 + \theta_2 - (l-1)}$$
$$+ 2^{\theta_1 + \theta_3 - (l-1)} + \ldots + 2^{\theta_{t-1} + \theta_t - (l-1)} -$$
$$- 2^{\theta_1 + \theta_2 + \theta_3 - 2(l-1)} + \ldots + 2^{\theta_{t-2} + \theta_{t-1} + \theta_t - 2(l-1)}) + \ldots$$
$$\ldots + (-1)^u (2^{\theta_1 + \theta_2 + \ldots + \theta_u - (u-1)(l-1)} + \ldots$$
$$\ldots + 2^{\theta_{t-u+1} + \ldots + \theta_t - (u-1)(l-1)}) + \ldots$$
$$\ldots + 2^{\theta_1 + \theta_2 + \ldots + \theta_t - (t-1)(l-1)},$$
$$\theta_t = l + i_t - j_t - 2,$$

і так далі. Тобто можна записати у вигляді системи:

$$p_l = \begin{cases} 2^{l-1}, \quad якщо \quad j_1+1 \le l \le j_2; \\ 2^{l-1} - 2^{\theta_1}, \quad \theta_1 = l + i_1 - j_1 - 2 \quad якщо \quad j_1+1 \le l \le j_2; \\ 2^{l-1} - 2^{\theta_1} - 2^{\theta_2} + 2^{\theta_1+\theta_2-(l-1)}, \quad якщо \quad j_3+1 \le l \le j_4; \\ 2^{l-1} - 2^{\theta_1} - 2^{\theta_2} - 2^{\theta_3} + 2^{\theta_1+\theta_2-(l-1)} + 2^{\theta_1+\theta_3-(l-1)} + \\ \quad +2^{\theta_2+\theta_3-(l-1)} - 2^{\theta_1+\theta_2+\theta_3-2(l-1)}, \quad якщо \quad j_3+1 \le l \le j_4, \\ \quad \dots \\ \quad 2^{l-1} - (2^{\theta_1} + 2^{\theta_2} + \dots + 2^{\theta_t}) + 2^{\theta_1+\theta_2-(l-1)} \\ \quad\quad +2^{\theta_1+\theta_3-(l-1)} + \dots + 2^{\theta_{t-1}+\theta_t-(l-1)} - \\ -2^{\theta_1+\theta_2+\theta_3-2(l-1)} + \dots + 2^{\theta_{t-2}+\theta_{t-1}+\theta_t-2(l-1)}) + \dots \\ \quad\quad \dots + (-1)^u (2^{\theta_1+\theta_2+\dots+\theta_\rho-(u-1)(l-1)} + \dots \quad якщо \quad j_t+1 \le l \le j_{t+1}. \\ \quad\quad\quad \dots + 2^{\theta_{t-u+1}+\dots+\theta_t-(u-1)(l-1)}) + \dots \\ \quad\quad\quad\quad \dots + 2^{\theta_1+\theta_2+\dots+\theta_t-(t-1)(l-1)}, \\ \quad\quad\quad\quad\quad \theta_t = l + i_t - j_t - 2, \end{cases}$$

Враховуючи, що вага числа на суматорі без додаткових зв'язків визначається як

$$Q_R = \sum_{l=1}^{k} r_l \cdot 2^{l-1}, \tag{2.14}$$

отримуємо зменшення величини числа ΔQ_R, яке співпадає з (2.13).

Проілюструємо вищенаведене прикладом п'ятирозрядного суматора по модулю M = 17. Відповідно до величини $m_i = 17$ модуля СЗК, кількість k ДОС дорівнює п'яти. При цьому, вихідна структура суматора по модулю $m_i = 17$ без додаткових зв'язків $X_{\downarrow i \uparrow j}$ матиме вигляд представлений на рис. 2.15.

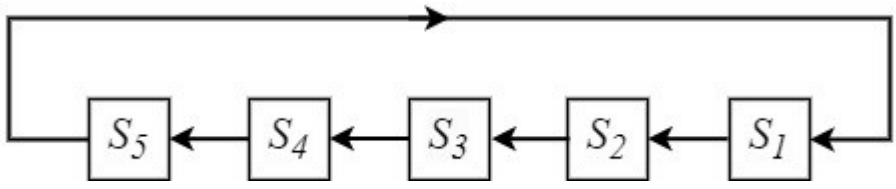

Рис. 2.15 Вихідна структура суматора по модулю $m_i = 17$ без додаткових зв'язків

Введемо два додаткових зв'язка: $X_{\downarrow i4\uparrow j5}$ – між виходом п'ятого і входом четвертого розряду та $X_{\downarrow i2\uparrow j3}$ – між виходом третього і входом другого розряду. Тоді структура суматора по модулю $m_i = 17$ з двома додатковими зв'язками $X_{\downarrow i4\uparrow j5}$ і $X_{\downarrow i2\uparrow j3}$ матиме вигляд представлений на рис. 2.16.

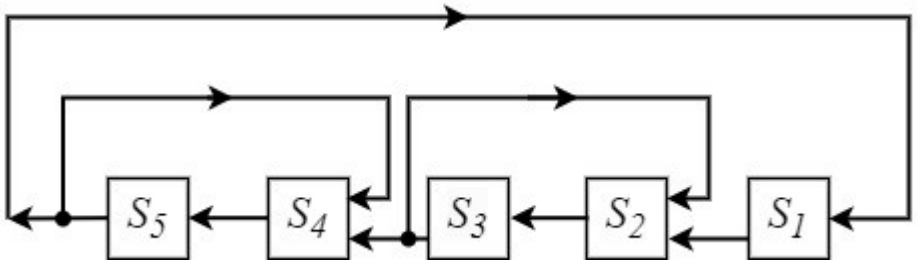

Рис. 2.16 Структура суматора по модулю $m_i = 17$ з двома додатковими зв'язками $X_{\downarrow i4\uparrow j5}$ і $X_{\downarrow i2\uparrow j3}$ з $\Delta Q_R = 2r_4 + 4r_5$

Для структури суматора, що представлена на рис. 2.16, визначимо значення модуля $M = m_i$ СЗК. Для цього, попередньо, складемо ряд структур окремих частин суматора, що представлений на рис. 2.16.

Перша частина структури суматора, представлена на рис. 2.17.

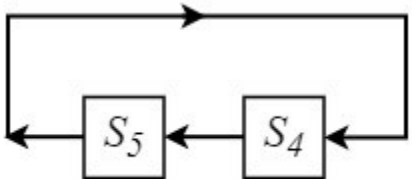

Рис. 2.17 Перша частина структури суматора по модулю $m_i = 17$

Для першої частини структури суматора значення модуля M_1 визначиться, як $M_1 = \rho_5 \cdot \rho_4 - 1$.

Друга частина структури суматора, представлена на рис. 2.18. Для цієї частини структури суматора значення модуля M_2 визначиться, як $M_2 = \rho_3 \cdot \rho_2 - 1$.

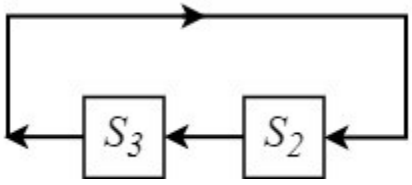

Рис. 2.18 Друга частина структури суматора за модулем $m_i = 17$

Третя частина структури суматора представлена на рис. 2.19.

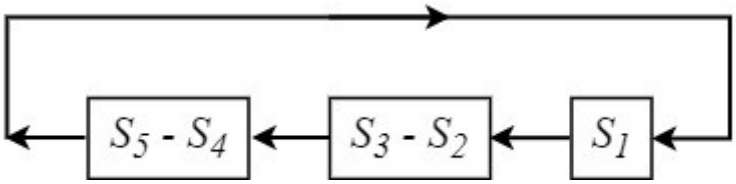

Рис. 2.19 Третя частина структури суматора за модулем $m_i = 17$

Значення модуля $M = m_i$ СЗК суматора (рис. 2.16) визначиться наступним чином (див. рис. 2.19) $m_i = M_1 \cdot M_2 \cdot \rho_1 =$
$= (\rho_5 \cdot \rho_4 - 1) \cdot (\rho_3 \cdot \rho_2 - 1) \cdot \rho_1 - 1 = (2 \cdot 2 - 1) \cdot (2 \cdot 2 - 1) \cdot 2 - 1 = 17.$

Знайдемо значення величини ΔQ_R. Значення величини ΔQ_R для конструкції суматора по модулю $m_i = 17$ з двома незалежними додатковими зв'язками $X_{\downarrow i4 \uparrow j5}$ і $X_{\downarrow i2 \uparrow j3}$ визначається як

$$\Delta Q_R = \sum_{l=j_1+1}^{k} r_l \cdot 2^{\theta_1},$$

де $\theta_1 = l + i_1 - j_1 - 2.$

Враховуючи, що $i_1 = 2, \quad j_1 = 3, \quad i_2 = 4, \quad j_2 = k = 3,$ отримуємо:

$$\Delta Q_R = \sum_{l=4}^{5} r_l \cdot 2^{l-3} = 2r_4 + 4r_5.$$

Не позбавлена інтересу конструкція суматора з залежними додатковими зв'язками, один з яких охоплює інший. У випадку введення двох додаткових зв'язків $X_{\downarrow i_1 \uparrow j_1}$ і $X_{\downarrow i_2 \uparrow j_2}$, які задовольняють умовам $i_1 \le i_2, \quad j_1 \ge j_2$ число $R = \{r_i\}$, $i = \overline{1, k}$, яке є вмістом величини Q_R суматора, зменшується на величину

$$\Delta Q_R = \sum_{l=j_1+1}^{k} 2^{\theta_1} \cdot r_l + \sum_{l=j_2+1}^{k} 2^{\theta_2} \cdot r_l , \qquad (2.15)$$

де $\theta_t = l + i_t - j_t - 2, \quad t = 1, 2.$

Дійсно, для розрядів з номерами l, які знаходяться в проміжку $j_2 < l \le j_1$, отримуємо $p_l = 2^{l-1} - 2^{\theta_2}$. Для розрядів з номерами $j_1 < l \le k$, ваги мають значення $p_l = 2^{l-1} - 2^{\theta_2} - 2^{\theta_1}$. Звідси, враховуючи (2.14) отримуємо

$$\Delta Q_R = \sum_{l=j_1}^{k} (2^{\theta_1} + 2^{\theta_2}) \cdot r_l + \sum_{l=j_2+1}^{j_1} 2^{\theta_2} \cdot r_l = \sum_{l=j_1+1}^{k} 2^{\theta_1} \cdot r_l + \sum_{l=j_2+1}^{k} 2^{\theta_2} \cdot r_l ,$$

що співпадає з (2.15).

Звідси випливає, якщо додатковий зв'язок $X_{\downarrow i_1 \uparrow j_1}$ підключено до старшого розряду суматора, тобто $j_1 = k$, то враховуючи (2.15) $\Delta Q_R = \sum_{l=j_2+1}^{k} 2^{\theta_2} \cdot r_l$.

Розглянемо випадок трьох додаткових зв'язків, кожна з яких включена в наступну. В цьому випадку число $R = \{r_i\}$, $i = \overline{1, k}$, яке є вмістом величини Q_R двійкового суматора, при введення трьох додаткових зв'язків $X_{\downarrow i_1 \uparrow j_1}$, $X_{\downarrow i_2 \uparrow j_2}$ і $X_{\downarrow i_3 \uparrow j_3}$, які задовольняють умовам $i_1 \le i_2 \le i_3$, $j_1 \ge j_2 \ge j_3$, зменшується на величину

$$\Delta Q_R = \sum_{l=j_1+1}^{k} 2^{\theta_1} \cdot r_l + \sum_{l=j_2+1}^{k} 2^{\theta_2} \cdot r_l + \sum_{l=j_3+1}^{k} 2^{\theta_3} \cdot r_l , \qquad (2.16)$$

де $\theta_t = l + i_t - j_t - 2, \quad t = 1, 2, 3.$

В цьому випадку для розрядів з номерами l, які задовольняють умові $1 \le l \le j_3$, вага визначається як $p_l = 2^{l-1}$. Для розрядів у яких $j_3 < l \le j_2$, вага дорівнює $p_l = 2^{l-1} - 2^{\theta_3}$, де $\theta_3 = l + i_3 - j_3 - 2$. Для розрядів, номера яких $j_2 < l \le j_1$, вага визначається як $p_l = 2^{\theta_2}(2^{\theta_3 - \theta_2}(2^{l-1-\theta_3} - 1) - 1) = 2^{l-1} - 2^{\theta_2} - 2^{\theta_3}$. І

нарешті для розрядів, у яких $j_1 < l \le k$, вага дорівнює $p_l = 2^{l-1} - 2^{\theta_1} - 2^{\theta_2} - 2^{\theta_3}$, звідки для величини ΔQ_R отримуємо

$$\Delta Q_R = \sum_{l=j_3+1}^{j_2} 2^{\theta_3} \cdot r_l + \sum_{l=j_2+1}^{j_1} (2^{\theta_2} + 2^{\theta_3}) \cdot r_l + \sum_{l=j_1+1}^{k} (2^{\theta_1} + 2^{\theta_2} + 2^{\theta_3}) \cdot r_l$$

що співпадає з (2.16).

В загальному випадку, введенні *s* додаткових зв'язків $X_{\downarrow i1\uparrow j1}$, $X_{\downarrow i2\uparrow j2}$ . . . $X_{\downarrow i_S\uparrow j_S}$, що задовольняють умові

$$i_1 \le i_2 \le \ldots \le i_s, \quad j_1 \ge j_2 \ge \ldots \ge j_s,$$

зменшує число $R = \{r_i\}$, $i = \overline{1,k}$, яке є вмістом величини Q_R двійкового суматора, на величину

$$\Delta Q_R = \sum_{l=j_1+1}^{k} 2^{\theta_1} \cdot r_l + \sum_{l=j_2+1}^{k} 2^{\theta_2} \cdot r_l + \ldots + \sum_{l=j_s+1}^{k} 2^{\theta_s} \cdot r_l \,. \qquad (2.17)$$

Проілюструємо вищенаведене прикладом семирозрядного суматора по модулю *M* = 67. Відповідно до величини $m_i = 67$ модуля СЗК, кількість *k* ДОС дорівнює семи. При цьому, вихідна структура суматора по модулю $m_i = 67$ без додаткових зв'язків $X_{\downarrow i\uparrow j}$ матиме вигляд представлений на рис. 2.20.

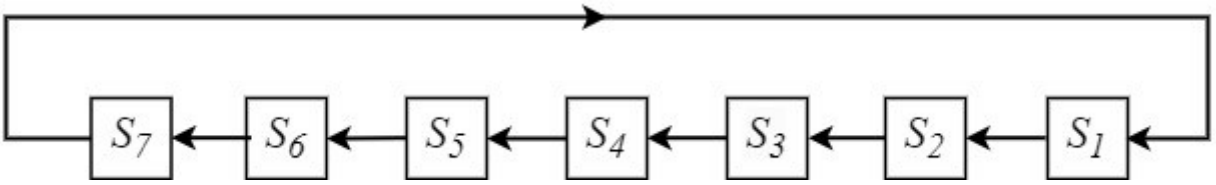

Рис. 2.20 Вихідна структура суматора по модулю $m_i = 67$ без додаткових зв'язків

Введемо три додаткових зв'язка: $X_{\downarrow i6\uparrow j7}$ – між виходом сьомого і входом шостого розряду, $X_{\downarrow i4\uparrow j5}$ – між виходом п'ятого і входом четвертого розряду та $X_{\downarrow i3\uparrow j7}$ – між виходом сьомого і входом третього розряду. Додатковий зв'язок

$X_{\downarrow i3\uparrow j7}$ охоплює незалежні між собою зв'язки $X_{\downarrow i6\uparrow j7}$ та $X_{\downarrow i4\uparrow j5}$ (див. рис. 2.21).

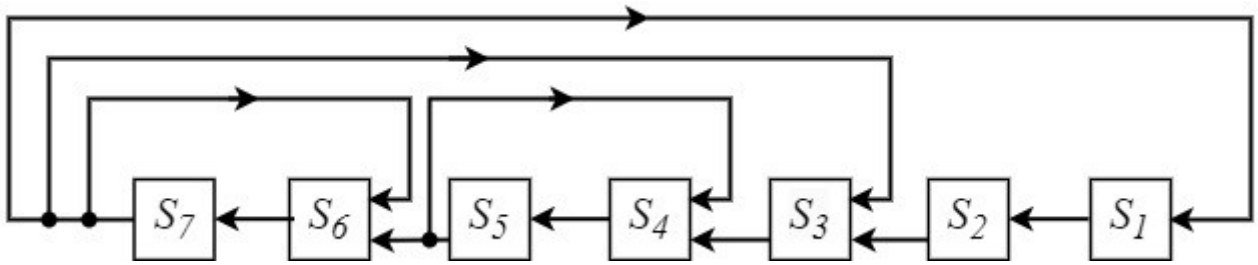

Рис. 2.21 Структура суматора по модулю $m_i = 67$ з трьома додатковими зв'язками $X_{\downarrow i6\uparrow j7}$, $X_{\downarrow i4\uparrow j5}$, $X_{\downarrow i3\uparrow j7}$, де додатковий зв'язок $X_{\downarrow i3\uparrow j7}$ охоплює незалежні між собою зв'язки $X_{\downarrow i6\uparrow j7}$ та $X_{\downarrow i4\uparrow j5}$

Для структури суматора, що представлена на рис. 2.21, визначимо значення модуля $M = m_i$ СЗК. Для цього, попередньо, складемо ряд структур окремих частин суматора, що представлений на рис. 2.21.

Перша частина структури суматора, представлена на рис. 2.22.

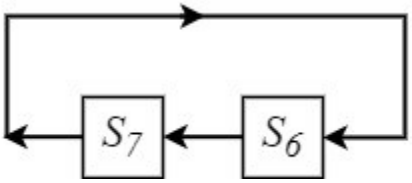

Рис. 2.22 Перша частина структури суматора по модулю $m_i = 67$

Для першої частини структури суматора значення модуля M_1 визначиться, як $M_1 = \rho_7 \cdot \rho_6 - 1$.

Друга частина структури суматора, представлена на рис. 2.23. Для цієї частини структури суматора значення модуля M_2 визначиться, як $M_2 = \rho_5 \cdot \rho_4 - 1$.

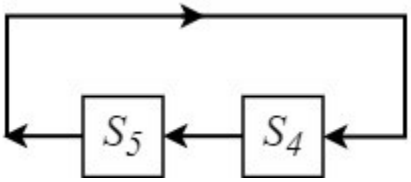

Рис. 2.23 Друга частина структури суматора по модулю $m_i = 67$

Третя частина структури суматора, представлена на рис. 2.24. Для цієї частини структури суматора значення модуля M_3 визначиться, як $M_3 = M_1 \cdot M_2 \cdot \rho_3 - 1$.

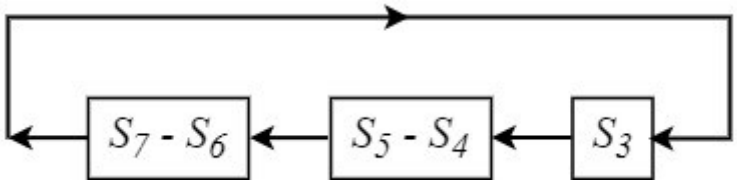

Рис. 2.24 Третя частина структури суматора по модулю $m_i = 67$

Четверта частина структури суматора представлена на рис. 2.25.

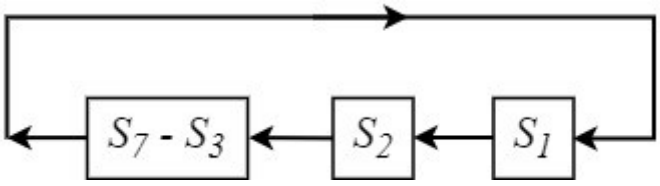

Рис. 2.25 – Четверта частина структури суматора по модулю $m_i = 67$

Значення модуля $M = m_i$ СЗК суматора (рис. 2.21) визначиться наступним виразом $m_i = M_3 \cdot \rho_2 \cdot \rho_1 - 1 = \left[M_1 \cdot M_2 \cdot \rho_3 - 1\right] \cdot \rho_2 \cdot \rho_1 - 1 =$

$= \left[(\rho_7 \cdot \rho_6 - 1) \cdot (\rho_5 \cdot \rho_4 - 1) \cdot \rho_3 - 1\right] \cdot \rho_2 \cdot \rho_1 - 1 = \left[(2 \cdot 2 - 1) \cdot (2 \cdot 2 - 1) \cdot 2 - 1\right] \cdot 2 \cdot 2 - 1 = 67.$

Визначимо значення величини ΔQ_R для конструкції суматора по модулю $m_i = 67$ з трьома додатковими зв'язками $X_{\downarrow i6 \uparrow j7}$, $X_{\downarrow i4 \uparrow j5}$, $X_{\downarrow i3 \uparrow j7}$, де додатковий зв'язок $X_{\downarrow i3 \uparrow j7}$ охоплює незалежні між собою зв'язки $X_{\downarrow i6 \uparrow j7}$ та $X_{\downarrow i4 \uparrow j5}$. Знайдемо розподіл ваг розрядів, прийнявши до уваги, що $i_1 = 3, \quad i_2 = 4, \quad i_3 = 6, \quad j_1 = j_3 = k = 7, \quad j_2 = 5$:

для $1 \le l \le j_2$ $\qquad p_l = 2^{l-1}$;

для $j_2 \le l \le j_3$ $\qquad p_l = 2^{l-1} - 2^{\theta_2}$;

звідки величина ΔQ_R визначається як

$$\Delta Q_R = \sum_{l=j_2+1}^{k} r_l \cdot 2^{\theta_2},$$

де $\theta_2 = l + i_2 - j_2 - 2.$

Звідси $\Delta Q_R = \sum_{l=6}^{7} r_l \cdot 2^{l-3} = 8r_6 + 16r_7.$

2.3. Метод реалізації операції додавання залишків чисел по модулю m_i СЗК

Метод додавання залишків чисел по довільному модулем m_i СЗК складається з двох етапів. Перший етап складається з рішення задачі побудови структури суматора модульного додавання $(x_i + y_i) \bmod m_i$. Другий етап. На основі отриманої структури суматора і чисельного значення модуля m_i, що представлений двійковим кодом, визначається схема модульного додавання залишків чисел x_i та y_i, обумовлена наявністю і використанням додаткових зв'язків $X_{\downarrow i \uparrow j}$ між j -м та i -м ДОС. Детально розглянемо кожен з етапів методу побудови суматорів за довільним модулем m_i СЗК.

Перший етап: - створення структури суматора модульного додавання $(x_i + y_i) \bmod m_i$.

Нехай задана довільна структура n-розрядного двійкового суматора в ПСЧ по модулю $M = 2^k - 1$. Потрібно створити структуру суматора модульного додавання $(x_i + y_i) \bmod m_i$, інакше кажучи, необхідно створити структуру суматора для реалізації операції додавання залишків чисел за довільним модулем m_i СЗК.

Технічно завдання побудови структури суматора за модулем формулюється в такий спосіб. Необхідно забезпечити умови, щоб вихідний суматор в ПСЧ по модулю M виконував би операцію додавання по модулю m_i. Ця процедура виконується шляхом застосування додаткових зв'язків $X_{\downarrow i \uparrow j}$ в позиційному суматорі по модулю $M = 2^k - 1$, де вираз $X_{\downarrow i \uparrow j}$ позначає

односторонній зв'язок між виходом j -го ДОС та входом i -го ДОС.

Для побудови непозиційного суматора за довільним модулем, в структурі позиційного суматора по модулю M необхідно між певною парою ДОС вихідного суматора по модулю M сформувати додаткові зв'язки виду $X_{\downarrow i \uparrow j}$. Додаткові зв'язки $X_{\downarrow i \uparrow j}$ формуються таким чином, щоб створений суматор здійснював операцію $(x_i + y_i) \bmod m_i$. В розділі 2.2 детально проведено дослідження впливу додаткових зв'язків суматора по модулю m_i на величину вмісту суматора.

Існуючий [44-45] метод реалізації операції додавання залишків чисел $(x_i + y_i) \bmod m_i$ по модулю m_i СЗК базується на використанні двійкових суматорів, складається з сукупності наступних операцій:

1. Синтез суматора по заданому модулю m_i СЗК;

2. Визначення результату $S_n\, S_{n-1} \ldots S_2\, S_1$ порозрядного додавання по модулю залишків чисел x_i та y_i по модулю два що представлені двійковим кодом;

3. Вміст двійкових розрядів, отриманої модульної суми $S_n\, S_{n-1} \ldots S_2\, S_1$, заноситься до відповідних ДОС структури суматора по модулю m_i СЗК;

4. На підставі синтезованої структури суматора по модулю m_i СЗК, реалізується алгоритм додавання залишків чисел x_i та y_i.

Однак в деяких випадках існуючий метод має обмежене застосування при додаванні залишків чисел по модулю m_i СЗК. Зокрема, в разі рівності залишків, тобто коли $x_i = y_i$, результат додавання $(x_i + y_i) \bmod m_i$ не в усіх випадках буде правильним. Це обумовлено тим, що отримана модульна сума $S_n\, S_{n-1} \ldots S_2\, S_1$, яка використовується при визначенні результату модульного додавання $(x_i + y_i) \bmod m_i$ залишків чисел x_i та y_i, не враховує співвідношення між величинами значень модулів m_i, M і значення $(x_i + y_i)$ позиційного додавання залишків чисел. В цьому випадку для отримання правильного результату

операції $(x_i + y_i) \bmod m_i$ необхідно враховувати величини значень m_i, M і значення $(x_i + y_i)$. Очевидно, що при розробці методу додавання залишків чисел $(x_i + y_i) \bmod m_i$ необхідно враховувати варіанти співвідношення між величинами значень модулів m_i, M та значенням $(x_i + y_i)$ результату позиційного додавання залишків чисел. У таблиці 2.2 представлені можливі співвідношення між величинами модулів m_i, M та значеннями $(x_i + y_i)$ результату позиційного додавання залишків чисел.

Таблиця 2.2

Співвідношення значень величин m_i, M та $(x_i + y_i)$

№ з/с	Співвідношення значень величин m_i, $(x_i + y_i)$ та $M = 2^k - 1$ $(k = 5, \quad m_i = 17, \quad M = 31)$		Режим виконання операції додавання
1	$x_i + y_i < m_i$	$x_i + y_i < 17$	Перший режим
2	$x_i + y_i = m_i$	$x_i + y_i = 17$	Другий режим
2	$m_i < x_i + y_i < 2^k - 1$	$17 < x_i + y_i < 31$	
4	$x_i + y_i = 2^k - 1$	$x_i + y_i = 31$	
5	$2^k - 1 < x_i + y_i$	$31 < x_i + y_i$	

Також вищенаведений метод реалізації операції додавання залишків чисел $(x_i + y_i) \bmod m_i$ в СЗК не враховує варіанти конструкцій суматорів з $\Delta Q_R > 0$. В цьому випадку необхідна корекція остаточного результату на величину ΔQ_R, шляхом додавання до отриманої модульної суми $S_n \, S_{n-1} \ldots S_2 \, S_1$, величини ΔQ_R у двійковому коді.

Узагальнюючи все вищевикладене, для заданого значення модуля m_i, вдосконалений метод реалізації операції додавання залишків чисел $(x_i + y_i) \bmod m_i$ буде складатися з сукупності таких операцій (рис. 2.26) [46-50]:

1. Спочатку, для заданого значення модуля m_i, здійснюється побудова

суматора за модулем m_i;

2. Формується результат позиційного підсумовування залишків чисел $(x_i + y_i)$;

3. Проводиться порівняння значень $(x_i + y_i)$ та m_i:

- якщо $(x_i + y_i) < m_i$, то значення $(x_i + y_i)$ – результат операції $(x_i + y_i) \bmod m_i$,

- якщо $(x_i + y_i) \geq m_i$, тоді для всіх можливих режимів виконання операції додавання (див. табл. 1), отриманий результат позиційного підсумовування залишків $(x_i + y_i)$, порозрядно заноситься до відповідних *k* ДОС, послідовна сукупність яких, становить структуру суматора по модулю m_i СЗК;

3.1. На підставі синтезованої, для заданого значення модуля m_i СЗК, структури суматора залишків чисел (див. п. 1), реалізується алгоритм додавання залишків чисел x_i і y_i по модулю.

3.2. Якщо $\Delta Q_R = 0$, то отриманий результат є результатом додавання залишків чисел $(x_i + y_i) \bmod m_i$ по модулю m_i СЗК, в іншому випадку, коли $\Delta Q_R > 0$ – необхідна корекція остаточного результату на величину ΔQ_R, шляхом додавання до отриманої модульної суми $S_n\, S_{n-1} \ldots S_2\, S_1$, величини ΔQ_R у двійковому коді – отриманий результат і буде результатом додавання залишків чисел $(x_i + y_i) \bmod m_i$ по модулю m_i СЗК.

Розглянемо конкретні приклади виконання операції додавання залишків чисел $(x_i + y_i) \bmod m_i$, по модулю m_i СЗК запропонованим вище методом.

Приклади виконання операції додавання розглянемо в двох режимах (див. табл. 2.2).

Перший режим виконання операції додавання: - виконується умова $(x_i + y_i) < m_i$ (див. табл. 2.2).

	Перший етап
1	Спочатку, для заданого значення модуля m_i, здійснюється побудова суматора за модулем m_i;
	Другий етап
2	Формується результат позиційного підсумовування залишків чисел $(x_i + y_i)$;
3	Проводиться порівняння значень $(x_i + y_i)$ та m_i: - якщо $(x_i + y_i) < m_i$, то значення $(x_i + y_i)$ – результат операції $(x_i + y_i) \bmod m_i$, - якщо $(x_i + y_i) \geq m_i$, тоді для всіх можливих режимів виконання операції додавання (див. табл. 2.2), отриманий результат позиційного підсумовування залишків $(x_i + y_i)$, порозрядно заноситься до відповідних *k* ДОС, послідовна сукупність яких, становить структуру суматора по модулю m_i СЗК;
	3.1. На підставі синтезованої, для заданого значення модуля m_i СЗК, структури суматора залишків чисел (див. п. 1), реалізується алгоритм додавання залишків чисел x_i і y_i по модулю.
	3.2. Якщо $\Delta Q_R = 0$, то отриманий результат є результатом додавання залишків чисел $(x_i + y_i) \bmod m_i$ по модулю m_i СЗК, в іншому випадку, коли $\Delta Q_R > 0$ – необхідна корекція остаточного результату на величину ΔQ_R, шляхом додавання до отриманої модульної суми $S_n\, S_{n-1} \ldots S_2\, S_1$, величини ΔQ_R у двійковому коді – отриманий результат і буде результатом додавання залишків чисел $(x_i + y_i) \bmod m_i$ по модулю m_i СЗК.

Рис. 2.26 Вдосконалений метод реалізації операції додавання залишків чисел по модулю СЗК

Приклад 2.3. Нехай залишки дорівнюють $x_i = 5$ і $y_i = 10$, значення модуля суматора $m_i = 17$. Суматор реалізує операцію позиційного додавання залишків

$x_i = 00101$ та $y_i = 01010$ у вигляді:

$$\begin{array}{rl} & x_i = \quad 00101 \\ + & \\ & y_i = \quad 01010 \\ \hline & (x_i + y_i) = \quad 01111 \end{array}$$

Значення суми $(x_i + y_i) = 01111$ – визначає результат операції додавання.

Перевірка: $(5+10)\bmod 17 = 15$, що у двійковому коді представляється, як 01111.

Другий режим виконання операції додавання: - виконується умова $(x_i + y_i) \geq m_i$ (див. табл. 3.2). Спочатку продемонструємо роботу метода на суматорі по модулю $m_i = 17$ з $\Delta Q_R = 0$.

Виконаємо синтезу суматора по модулю $m_i = 17$.

Відповідно до величини $m_i = 17$ модуля СЗК, визначимо кількість *k* ДОС. Для модуля $m_i = 17$ маємо, що $k = \left[\log_2(17-1)\right] + 1 = 5$. При цьому, структура суматора по модулю $M = 2^k - 1$ буде мати вигляд, який наведено на рис. 2.27.

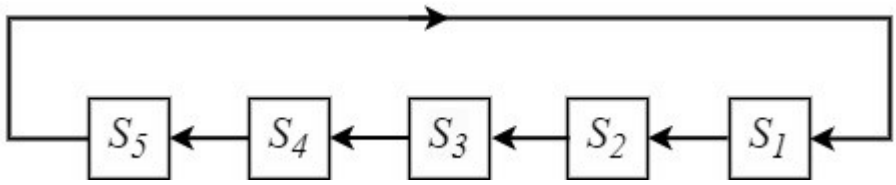

Рис. 2.27 – Вихідна структура суматора по модулю $M = 2^k - 1$

Для синтезу суматора по модулю $m_i = 17$ СЗК, попередньо, визначимо значення двійкових розрядів S_i суматора, в запису модуля $m_i = 17$ яких, містяться нулі, тобто для випадку, коли $S_i = 0$. Такими розрядами буде другий, третій та четвертий розряд, тобто $S_2 = 0$, $S_3 = 0$ та $S_4 = 0$, так як в двійковому коді модуль $m_i = 17$ має наступний вигляд 10001.

Виходячи з того, що $S_2 = 0$, $S_3 = 0$ та $S_4 = 0$, введемо в суматор по модулю $M = 2^k - 1$ три додаткові зв'язки: $X_{\downarrow 4 \uparrow 5}$, $X_{\downarrow 3 \uparrow 5}$, $X_{\downarrow 2 \uparrow 5}$. У цьому випадку

структура суматора по модулю $m_i = 17$ має вид, що представлений на рис. 2.28.

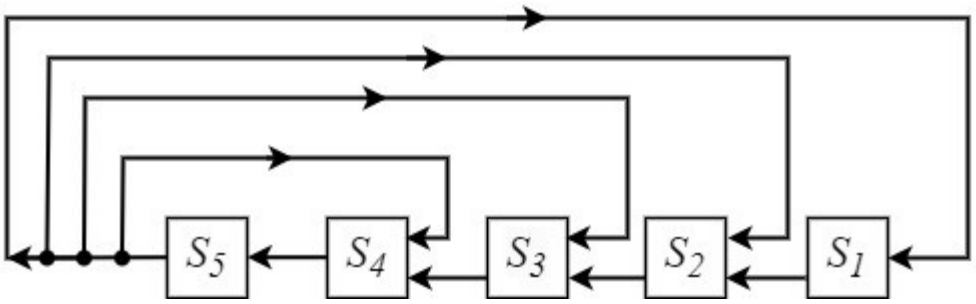

Рис. 2.28 – Структура суматора по модулю $m_i = 17$

Визначимо для даної структури суматора, що представлена на рис. 2.27, значення модуля $M = m_i$ СЗК. Для цього, попередньо, складемо ряд структур окремих частин суматора, що представлений на рис. 2.28.

Перша частина структури суматора, представлена на рис. 2.29.

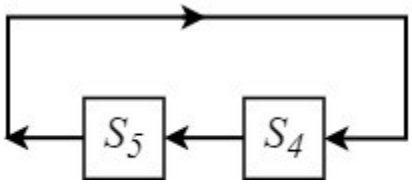

Рис. 2.29 Перша частина структури суматора по модулю $m_i = 17$

Для першої частини структури суматора значення модуля M_1 визначиться, як $M_1 = \rho_5 \cdot \rho_4 - 1$.

Друга частина структури суматора, представлена на рис. 2.30. Для цієї частини структури суматора значення модуля M_2 визначиться, як $M_2 = M_1 \cdot \rho_3 - 1$.

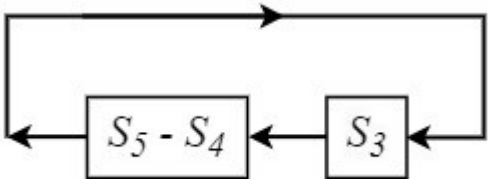

Рис. 2.30 Друга частина структури суматора по модулю $m_i = 17$

Третя частина структури суматора представлена на рис. 2.31.

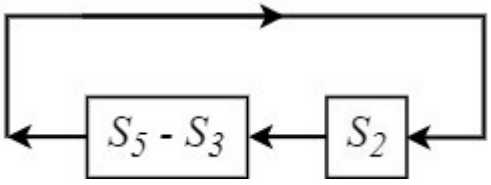

Рис. 2.31 Третя частина структури суматора по модулю $m_i = 17$

Для третьої частини структури суматора значення модуля M_1 визначиться, як $M_3 = M_2 \cdot \rho_2 - 1$.

Четверта частина структури суматора представлена на рис. 2.32

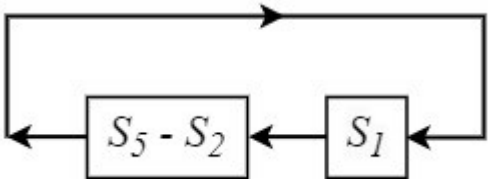

Рис. 2.32 Четверта частина структури суматора по модулю $m_i = 17$

Значення модуля $M = m_i$ СЗК суматора (рис. 2.28) визначиться наступним чином (див. рис. 2.32) $m_i = M_3 \cdot \rho_1 - 1 =$
$= [((\rho_5 \cdot \rho_4 - 1) \cdot \rho_3 - 1) \cdot \rho_2 - 1] \cdot \rho_1 - 1 = [(((2 \cdot 2 - 1) \cdot 2 - 1) \cdot 2 - 1)] \cdot 2 - 1 = 17$.

Отже, синтез суматора по модулю $m_i = 17$ проведений вірно.

Приклад 2.4. Нехай $x_i = 11$ і $y_i = 6$, значення модуля суматора $m_i = 17$. Суматор реалізує операцію порозрядного додавання залишків $x_i = 01011$ і $y_i = 00110$ по модулю два у вигляді:

$$\begin{array}{rl} & x_i = \quad 01011 \\ + & \\ & y_i = \quad 00110 \\ \hline & x_i + y_i = \quad 10001 \end{array}$$

Значення порозрядної суми $x_i + y_i = 10001$ залишків $x_i = 01011$ та $y_i = 00110$ порозрядно надходить на відповідні входи ДОС $S_5 - S_1$ (рис. 2.28). Таким чином, ДОС $S_5 - S_1$ суматора містить значення 10001. Алгоритм реалізації модульної операції представлений в таблиці 2.3 та на рис. 2.33.

Одиниця двійкового розряду надходить на вхід ДОС S_4, S_3, S_2 та S_1.

Таблиця 2.3

Алгоритм реалізації операції модульного додавання залишків $x_i = 01011$ та $y_i = 00110$ на суматорі по модулю $m_i = 17$ з $\Delta Q_R = 0$

Номер ДОС S_i	Вміст ДОС S_i	Наявність одиниці на входах ДОС S_i	Результат операції модульного додавання
S_1	1	+1	0
S_2	0	+1	0
S_3	0	+1	0
S_4	0	+1	0
S_5	1	–	0

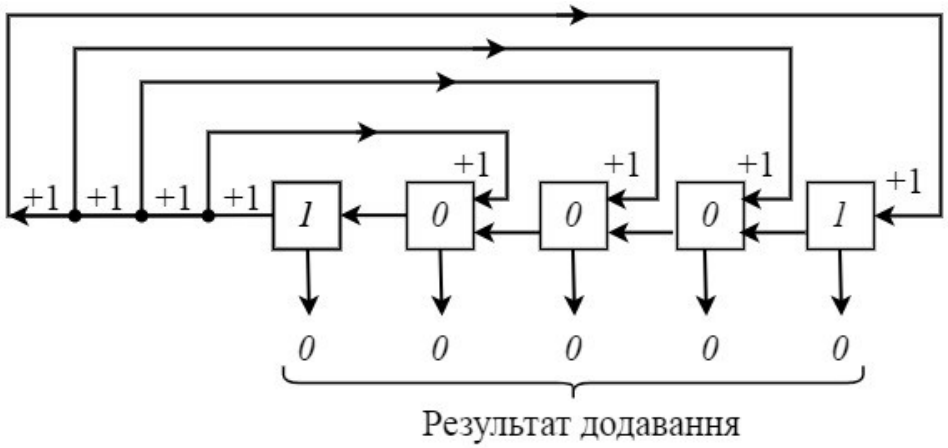

Рис. 2.33 Схема додавання залишків $x_i = 01011$ та $y_i = 00110$ на суматорі по модулю $m_i = 17$ з $\Delta Q_R = 0$

У зв'язку з тим, що усі додаткові зв'язки: $X_{\downarrow 4 \uparrow 5}$, $X_{\downarrow 3 \uparrow 5}$, $X_{\downarrow 2 \uparrow 5}$ виходять з старшого ДОС (тобто $j = k$), що забезпечує мінімальне значення величини $\Delta Q_R = 0$, корекція отриманого результату не потрібна.

Перевірка: $(11+6) \bmod 17 = 0$.

Приклад 2.5. Нехай $x_i = 11$ і $y_i = 15$, значення модуля суматора $m_i = 17$. Суматор реалізує операцію порозрядного додавання залишків $x_i = 01011$ і $y_i = 01111$ по модулю два у вигляді:

$$\begin{array}{rl} & x_i = \quad 01011 \\ + & \\ & y_i = \quad 01111 \\ \hline & x_i + y_i = \quad 11010 \end{array}$$

Значення порозрядної суми $x_i + y_i = 11010$ залишків $x_i = 01011$ та $y_i = 01111$ порозрядно надходить на відповідні входи ДОС $S_5 - S_1$. Таким чином, ДОС $S_5 - S_1$ суматора містить значення 11010. Алгоритм реалізації модульної операції представлений в таблиці 2.4 та на рис. 2.34. Одиниця двійкового розряду надходить на вхід ДОС S_4, S_3, S_2 та S_1.

Таблиця 2.4

Алгоритм реалізації операції модульного додавання залишків $x_i = 01011$ та $y_i = 01111$ на суматорі по модулю $m_i = 17$ з $\Delta Q_R = 0$

Номер ДОС S_i	Вміст ДОС S_i	Наявність одиниці на входах ДОС S_i	Результат операції модульного додавання
S_1	0	+1	1
S_2	1	+1	0
S_3	0	+1	0
S_4	1	+1	1
S_5	1	–	0

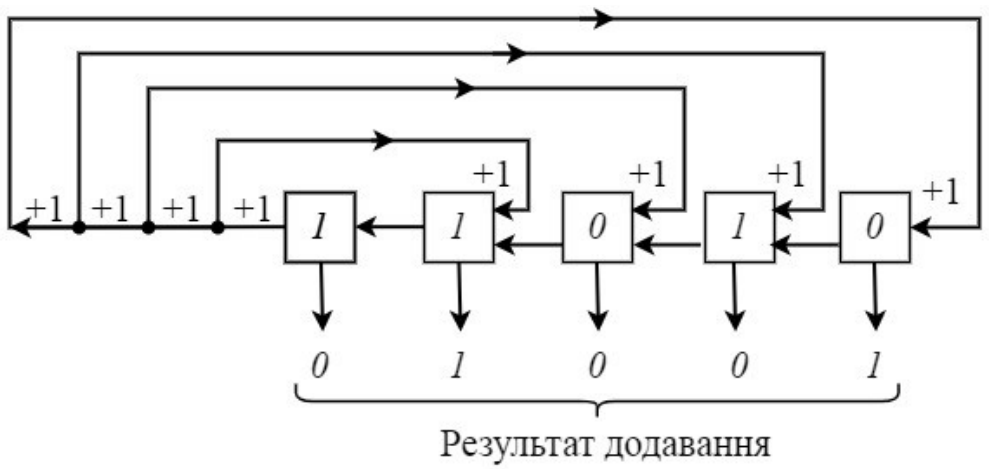

Рис. 2.34 Схема додавання залишків $x_i = 01011$ та $y_i = 01111$ на суматорі по модулю $m_i = 17$ з $\Delta Q_R = 0$

У зв'язку з тим, що усі додаткові зв'язки: $X_{\downarrow 4 \uparrow 5}$, $X_{\downarrow 3 \uparrow 5}$, $X_{\downarrow 2 \uparrow 5}$ виходять з старшого ДОС (тобто $j = k$), що забезпечує мінімальне значення величини $\Delta Q_R = 0$, корекція отриманого результату не потрібна. У підсумку проведення операції модульного додавання отримуємо результат $S_5\, S_4\, S_3\, S_2\, S_1 = 01001$.

Перевірка: $(11+15) \bmod 17 = 9$, що у двійковому коді дорівнює 1001.

Приклад2.6. Нехай $x_i = y_i = 15$, значення модуля суматора $m_i = 17$. Суматор реалізує операцію порозрядного додавання залишків $x_i = 01111$ і $y_i = 01111$ по модулю два у вигляді:

$$\begin{array}{rr} & x_i = \quad 01111 \\ + & \\ & y_i = \quad 01111 \\ \hline & x_i + y_i = \quad 11110 \end{array}$$

Значення порозрядної суми $x_i + y_i = 11110$ залишків $x_i = 01111$ та $y_i = 1111$ порозрядно надходить на відповідні входи ДОС $S_5 - S_1$. Таким чином, ДОС $S_5 - S_1$ суматора містить значення 11110. Алгоритм реалізації модульної операції представлений в таблиці 2.5 та на рис. 2.35. Одиниця двійкового розряду надходить на вхід ДОС S_4, S_3, S_2 та S_1.

Таблиця 2.5

Алгоритм реалізації операції модульного додавання залишків $x_i = 01111$ та $y_i = 10000$ на суматорі по модулю $m_i = 17$ з $\Delta Q_R = 0$

Номер ДОС S_i	Вміст ДОС S_i	Наявність одиниці на входах ДОС S_i	Результат операції модульного додавання
S_1	0	+1	1
S_2	1	+1	0
S_3	1	+1	1
S_4	1	+1	1
S_5	1	–	0

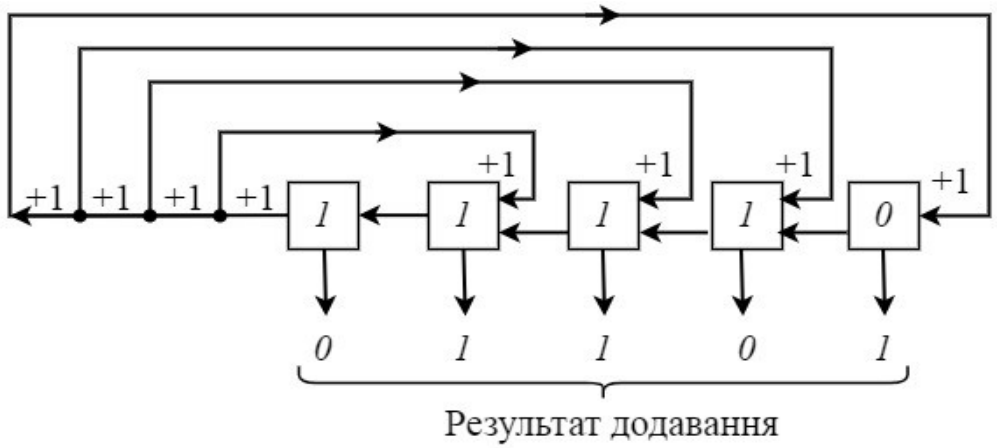

Рис. 2.35 Схема додавання залишків $x_i = 01111$ та $y_i = 10000$ на суматорі по модулю $m_i = 17$ з $\Delta Q_R = 0$

У зв'язку з тим, що усі додаткові зв'язки: $X_{\downarrow 4 \uparrow 5}$, $X_{\downarrow 3 \uparrow 5}$, $X_{\downarrow 2 \uparrow 5}$ виходять з старшого ДОС (тобто $j = k$), що забезпечує мінімальне значення величини $\Delta Q_R = 0$, корекція отриманого результату не потрібна. У підсумку проведення операції модульного додавання отримуємо результат $S_5\, S_4\, S_3\, S_2\, S_1 = 01101$.

Перевірка: $(15+15) \bmod 17 = 13$, що у двійковому коді дорівнює 01101.

Приклад 2.7. Нехай $x_i = 15$ і $y_i = 16$, значення модуля суматора $m_i = 17$. Суматор реалізує операцію порозрядного додавання залишків $x_i = 01111$ і $y_i = 10000$ по модулю два у вигляді:

$$\begin{array}{rl} & x_i = \quad 01111 \\ + & \\ & y_i = \quad 10000 \\ \hline & x_i + y_i = \quad 11111 \end{array}$$

Значення порозрядної суми $x_i + y_i = 11111$ залишків $x_i = 01111$ та $y_i = 10000$ порозрядно надходить на відповідні входи ДОС $S_5 - S_1$. Таким чином, ДОС $S_5 - S_1$ суматора містить значення 11111. Алгоритм реалізації модульної операції представлений в таблиці 2.6 та на рис. 2.36. Одиниця двійкового розряду надходить на вхід ДОС S_4, S_3, S_2 та S_1.

Таблиця 2.6

Алгоритм реалізації операції модульного додавання залишків $x_i = 01111$ та $y_i = 10000$ на суматорі по модулю $m_i = 17$ з $\Delta Q_R = 0$

Номер ДОС S_i	Вміст ДОС S_i	Наявність одиниці на входах ДОС S_i	Результат операції модульного додавання
S_1	1	+1	0
S_2	1	+1	1
S_3	1	+1	1
S_4	1	+1	1
S_5	1	–	0

У зв'язку з тим, що усі додаткові зв'язки: $X_{\downarrow 4 \uparrow 5}$, $X_{\downarrow 3 \uparrow 5}$, $X_{\downarrow 2 \uparrow 5}$ виходять з старшого ДОС (тобто $j = k$), що забезпечує мінімальне значення величини $\Delta Q_R = 0$, корекція отриманого результату не потрібна. У підсумку проведення

операції модульного додавання отримуємо результат $S_5\,S_4\,S_3\,S_2\,S_1 = 01110$.

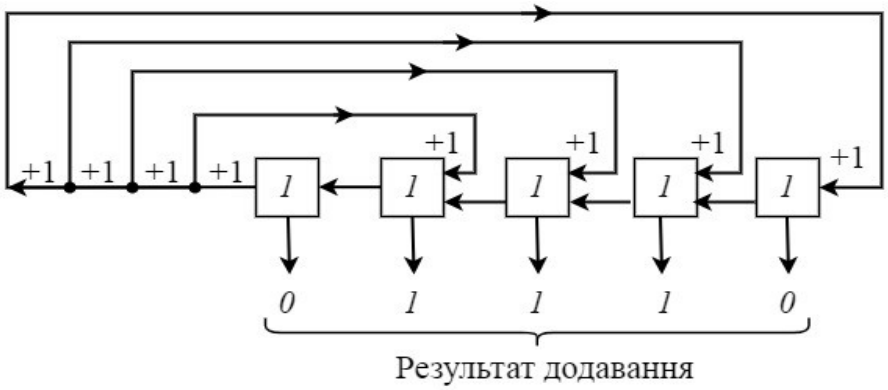

Рис. 2.36 Схема додавання залишків $x_i = 01111$ та $y_i = 10000$ на суматорі по модулю $m_i = 17$ з $\Delta Q_R = 0$

Перевірка: $(15+16)\bmod 17 = 14$, що у двійковому коді дорівнює 1110.

Розроблена HDL-модель суматора по модулю $m_i = 17$ на мові Verylog (див. рис. 2.37).

```
module add_mod_17 (X, Y, SUM, SUM_MOD);
input [4:0] X;
input [4:0] Y;
output [5:0] SUM;
output [4:0] SUM_MOD;
wire MOD = 6'b010001;
wire Cout;
wire [5:0] SUM_MOD_TEMP;
assign SUM = X + Y;
always @*
begin
if (SUM < 6'b010001)
     begin
          SUM_MOD = SUM[4:0];
     end else begin
          SUM_MOD_TEMP = SUM[4:0] + 5'b00001;
          SUM_MOD_TEMP = SUM_MOD_TEMP + 5'b00010;
          SUM_MOD_TEMP = SUM_MOD_TEMP + 5'b00100;
          SUM_MOD_TEMP = SUM_MOD_TEMP + 5'b01000;
          SUM_MOD = SUM_MOD_TEMP[4:0];
     end
end
endmodule
```

Рис. 2.37 Лістинг коду HDL-моделі суматора по модулю $m_i = 17$

Структурна схема суматора по модулю $m_i = 17$ в середовищі Quartus II наведено на рис. 2.38.

Результат моделювання суматора по модулю $m_i = 17$ в середовищі Quartus II наведено на рис. 2.39.

Якщо основним критерієм вибору конструкції суматора є прагнення до зменшення числа зв'язків, то доцільно застосувати схему, зображену на рис. 2.16. В цій схемі введено лише два додаткових зв'язка $X_{\downarrow 4 \uparrow 5}$ і $X_{\downarrow 2 \uparrow 3}$. При цьому величина $\Delta Q_R = 4$.

Продемонструємо використання запропонованого метода на суматорі по модулю $m_i = 17$ з $\Delta Q_R = 4$ (див. рис. 2.16).

Приклад 2.8. Нехай $x_i = 8$ і $y_i = 9$. Суматор реалізує операцію порозрядного додавання залишків $x_i = 01000$ і $y_i = 01001$ по модулю два у вигляді:

$$\begin{array}{rl} & x_i = \quad 01000 \\ + & \\ & y_i = \quad 01001 \\ \hline & x_i + y_i = \quad 10001 \end{array}$$

Значення порозрядної суми $x_i + y_i = 10001$ залишків $x_i = 01000$ та $y_i = 01001$ порозрядно надходить на відповідні входи ДОС $S_5 - S_1$. Таким чином, ДОС $S_5 - S_1$ суматора містить значення 10001. Алгоритм реалізації модульної операції представлений в таблиці 2.7 та на рис. 2.40. Одиниця двійкового розряду надходить на вхід ДОС S_4, S_2 та S_1. Корекція вмісту суматора на величину $\Delta Q_R = 4$ відбувається шляхом додавання 4 у двійковому коді на вхід ДОС $S_5 - S_1$ ($S_5\, S_4\, S_3\, S_2\, S_1 = 00100$).

Перевірка: $(8+9) \bmod 17 = 0$.

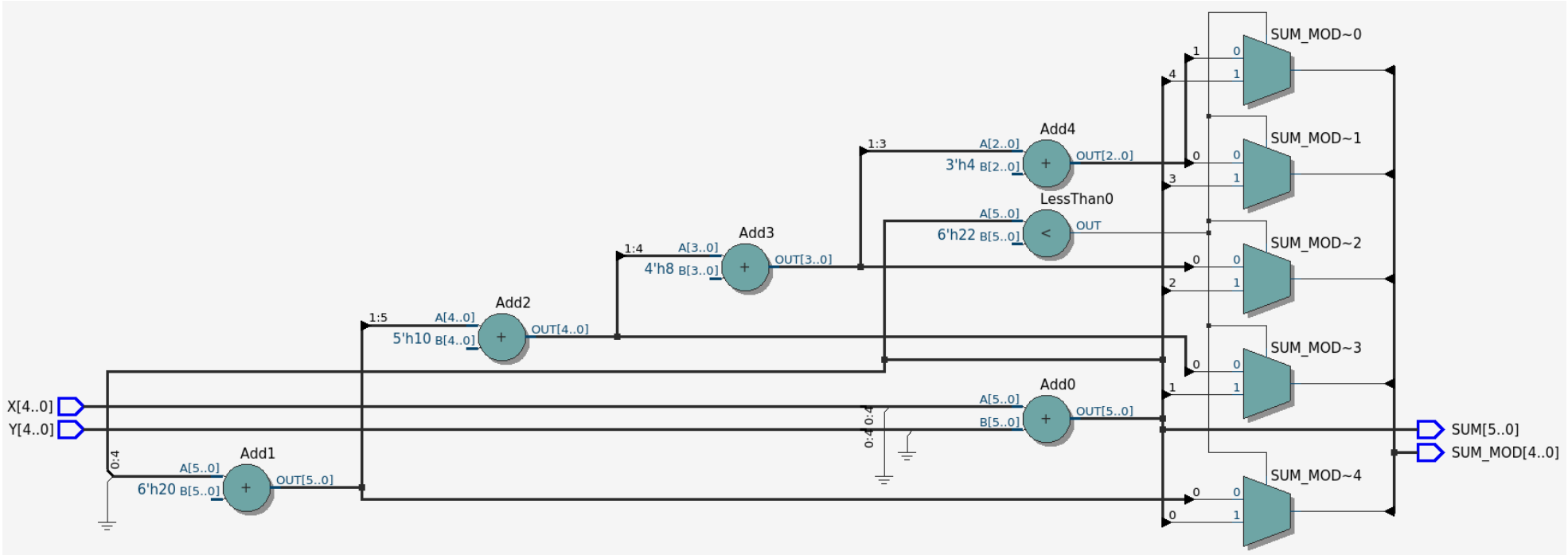

Рис. 2.38 Структурна схема суматора по модулю $m_i = 17$ в середовищі Quartus II

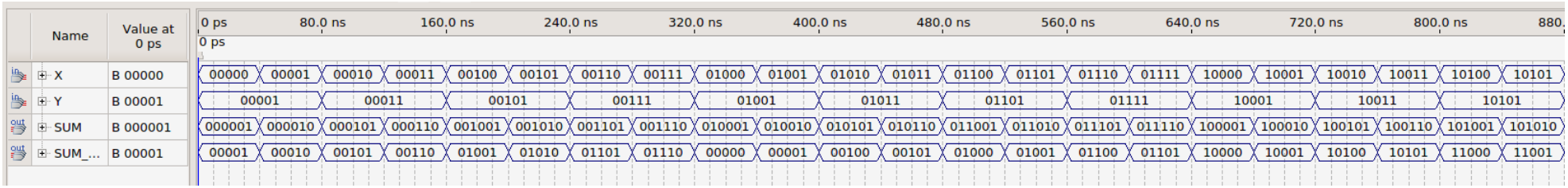

Рис. 2.39 Результат моделювання суматора по модулю $m_i = 17$ в середовищі Quartus II

Таблиця 2.7

Алгоритм реалізації операції модульного додавання залишків $x_i = 01000$ та $y_i = 01001$ на суматорі по модулю $m_i = 17$ з $\Delta Q_R = 4$

Номер ДОС S_i	Вміст ДОС S_i	Наявність одиниці на входах ДОС S_i	Корекція вмісту суматора на величину $\Delta Q_R = 4$	Результат операції модульного додавання
S_1	1	+1	0	0
S_2	0	+1	0	0
S_3	0	–	+1	0
S_4	0	+1	0	0
S_5	1	–	0	0

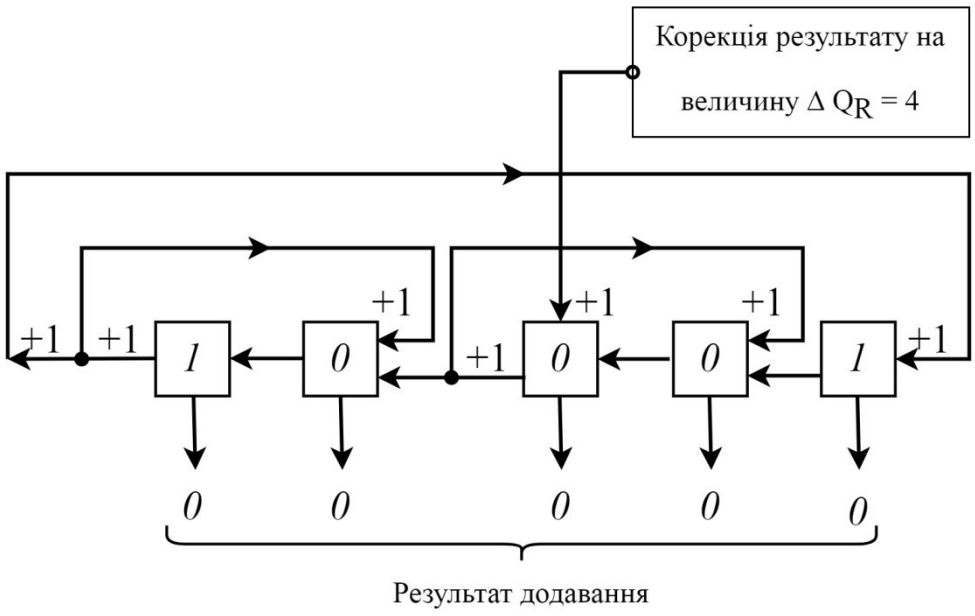

Рис. 2.40 Схема додавання залишків залишків $x_i = 01000$ та $y_i = 01001$ на суматорі по модулю $m_i = 17$ з $\Delta Q_R = 4$

Приклад 2.9. Нехай $x_i = 12$ і $y_i = 14$. Суматор реалізує операцію

порозрядного додавання залишків $x_i = 01100$ і $y_i = 01110$ по модулю два у вигляді:

$$\begin{array}{rl} & x_i = \quad 01100 \\ + & \\ & y_i = \quad 01110 \\ \hline & x_i + y_i = \quad 11010 \end{array}$$

Значення порозрядної суми $x_i + y_i = 11010$ залишків $x_i = 01100$ та $y_i = 01110$ порозрядно надходить на відповідні входи ДОС $S_5 - S_1$. Таким чином, ДОС $S_5 - S_1$ суматора містить значення 11010. Алгоритм реалізації модульної операції представлений в таблиці 2.8 та на рис. 2.41. Одиниця двійкового розряду надходить на вхід ДОС S_4, S_2 та S_1. Корекція вмісту суматора на величину $\Delta Q_R = 4$ відбувається шляхом додавання 4 у двійковому коді на вхід ДОС $S_5 - S_1$ ($S_5\ S_4\ S_3\ S_2\ S_1 = 00100$).

Таблиця 2.8

Алгоритм реалізації операції модульного додавання залишків $x_i = 01100$ та $y_i = 01110$ на суматорі по модулю $m_i = 17$ з $\Delta Q_R = 4$

Номер ДОС S_i	Вміст ДОС S_i	Наявність одиниці на входах ДОС S_i	Корекція вмісту суматора на величину $\Delta Q_R = 4$	Результат операції модульного додавання
S_1	0	+1	0	1
S_2	1	+1	0	0
S_3	0	–	+1	0
S_4	1	+1	0	1
S_5	1	–	0	0

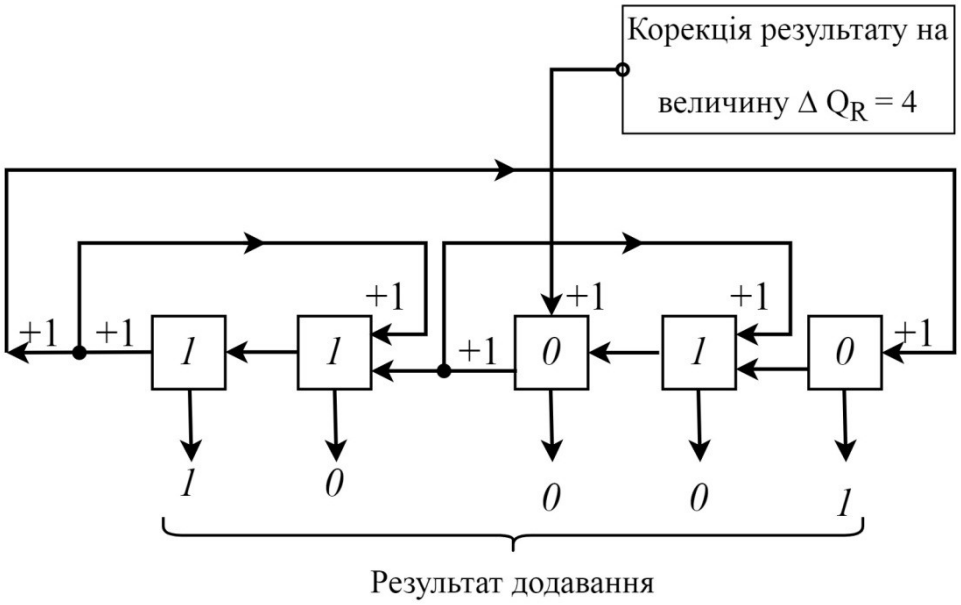

Рис. 2.41 Схема додавання залишків $x_i = 01100$ та $y_i = 01110$ на суматорі по модулю $m_i = 17$ з $\Delta Q_R = 4$

У підсумку проведення операції модульного додавання отримуємо результат $S_5\, S_4\, S_3\, S_2\, S_1 = 01001$.

Перевірка: $(12+14) \bmod 17 = 9$, що у двійковому коді представляється як 1001.

Розглянуті приклади реалізації методу модульного додавання для різних значень x_i і y_i по модулю m_i СЗК, підтверджують практичну реалізованість запропонованого методу [46-50].

2.4 Метод реалізації операції віднімання залишків чисел по модулю m_i СЗК

Розглянемо виконання операції віднімання на суматорі по модулю m_i СЗК.

Зазвичай операція віднімання числа β від числа α на суматорах виконується додаванням числа α з доповненням числа β до модуля суматора [51], тобто

$$\alpha - \beta = \alpha + \overline{\beta},$$

де

$$\overline{\beta} = M - \beta.$$

На двійкових суматорах без додаткових зв'язків доповнення числа β до величини модуля суматора M реалізується просто інверсією числа β.

Сформулюємо метод реалізації операції віднімання $(x_i - y_i) \bmod m_i$ по довільному модулю m_i (рис. 2.42). Метод також заснований на використанні суматорів по модулю $M = 2^k - 1$, які складаються з сукупності послідовних ДОС, шляхом застосування додаткових зв'язків $X_{\downarrow i \uparrow j}$.

Розглянемо конкретні приклади виконання операції $(x_i - y_i) \bmod m_i$ для довільних залишків чисел x_i і y_i по модулю $m_i = 17$ СЗК запропонованим вище методом на суматорі по модулю $m_i = 17$ з трьома додатковими зв'язками три додаткові зв'язки: $X_{\downarrow 4 \uparrow 5}$, $X_{\downarrow 3 \uparrow 5}$ і $X_{\downarrow 2 \uparrow 5}$. з $\Delta Q_R = 0$ (див. рис. 2.28) та на суматорі по модулю $m_i = 17$ двома додатковими зв'язками $X_{\downarrow 4 \uparrow 5}$ і $X_{\downarrow 2 \uparrow 3}$ з $\Delta Q_R = 4$ (див. рис. 2.16).

Приклад 2.10. Нехай $x_i = 7$ і $y_i = 9$. Знайти $(x_i - y_i) \bmod m_i$.

Знайдемо величину $\overline{y_i}$, для цього на суматорі реалізуємо операцію порозрядного додавання чисел $y_i + \overline{m_i}$ по модулю два, (де: $y_i = 01001$, величину $\overline{m_i}$ отримуємо порозрядною інверсією модуля $m_i = 10001$, тобто $\overline{m_i} = 01110$):

$$\begin{array}{rl} & y_i = \quad 01001 \\ + & \\ & \overline{m_i} = \quad 01110 \\ \hline y_i + \overline{m_i} = & \quad 10111 \end{array}$$

Далі знаходимо величину $\overline{y_i}$, шляхом порозрядної інверсії суми $y_i + \overline{m_i}$, $\overline{y_i} = 01000$.

1	Спочатку, для заданого значення модуля m_i, здійснюється синтез суматора по модулю m_i СЗК
2	Далі знаходимо доповнення $\overline{y_i}$ від'ємника y_i. Для цього:
	2.1 Формується результат позиційного підсумовування $y_i + \overline{m_i}$ від'ємника y_i з порозрядною інверсією модуля m_i;
	2.2 Отримана сума $y_i + \overline{m_i}$ порозрядно інвертується $\overline{y_i} = y_i + \overline{m_i}$;
3	Формується результат позиційного підсумовування $x_i + \overline{y_i}$ зменшуваного x_i і інверсії доповнення $\overline{y_i}$ від'ємника y_i;
4	Проводиться порівняння значень величин зменшуваного x_i і від'ємника y_i: – Якщо зменшуване x_i менше від'ємника y_i тобто $x_i < y_i$, тоді значення $x_i + \overline{y_i}$ буде результатом операції $(x_i - y_i) \bmod m_i$; – Якщо $x_i \geq y_i$, то отриманий результат $x_i + \overline{y_i}$ позиційного підсумовування зменшуваного x_i і інверсії доповнення $\overline{y_i}$ від'ємника y_i, порозрядно заноситься до відповідних k ДОС, послідовна сукупність яких, становить структуру суматора по модулю m_i СЗК;
	4.1. На підставі синтезованої, для заданого значення модуля m_i СЗК, структури суматора залишків чисел, реалізується алгоритм додавання залишків чисел x_i та $\overline{y_i}$ чисел по модулю.
	4.2. Якщо $\Delta Q_R = 0$, то отриманий результат є результатом операції віднімання $(x_i - y_i) \bmod m_i$ по модулю m_i СЗК, в іншому випадку, коли $\Delta Q_R > 0$ – необхідна корекція остаточного результату на величину ΔQ_R, шляхом додавання до отриманої модульної суми $S_n\, S_{n-1} \ldots S_2\, S_1$, величини ΔQ_R у двійковому коді – отриманий результат і буде результатом операції віднімання $(x_i - y_i) \bmod m_i$ по модулю m_i СЗК.

Рис. 2.42 Вдосконалений метод реалізації операції віднімання залишків чисел по модулю СЗК

Знаходимо значення позиційного підсумовування $x_i + \overline{y_i}$, де $x_i = 00111$, а $\overline{y_i} = 01000$:

$$\begin{array}{rl} & x_i = \quad 00111 \\ + & \\ & \overline{y_i} = \quad 01000 \\ \hline x_i + \overline{y_i} & \quad 01111 \end{array}$$

Так як $x_i < y_i$, значення порозрядної суми $x_i + \overline{y_i} = 01111$ і буде результатом операції $(x_i - y_i) \bmod m_i$.

Перевірка: $(7-9) \bmod 17 = 17 - 2 = 15$, що у двійковому коді дорівнює 01111.

Приклад 2.11. Нехай $x_i = 9$ і $y_i = 7$. Знайти $(x_i - y_i) \bmod m_i$.

Знайдемо величину $\overline{y_i}$, для цього на суматорі реалізуємо операцію порозрядного додавання чисел $y_i + \overline{m_i}$ по модулю два, (де: $y_i = 00111$, величину $\overline{m_i}$ отримуємо порозрядною інверсією модуля $m_i = 10001$, тобто $\overline{m_i} = 01110$):

$$\begin{array}{rl} & y_i = \quad 00111 \\ + & \\ & \overline{m_i} = \quad 01110 \\ \hline y_i + \overline{m_i} = & \quad 10101 \end{array}$$

Далі знаходимо величину $\overline{y_i}$, шляхом порозрядної інверсії суми $y_i + \overline{m_i}$, $\overline{y_i} = 01010$.

Знаходимо значення позиційного підсумовування $x_i + \overline{y_i}$, де $x_i = 01001$, а $\overline{y_i} = 01010$:

$$\begin{array}{rl} & x_i = \quad 01001 \\ + & \\ & \overline{y_i} = \quad 01010 \\ \hline y_i + \overline{m_i} & \quad 10011 \end{array}$$

Так як $x_i \geq y_i$, значення порозрядної суми $x_i + \overline{y_i} = 10011$ порозрядно надходить на відповідні входи ДОС $S_5 - S_1$. Таким чином, ДОС $S_5 - S_1$

суматора містить значення 10011.

Якщо використовується суматор по модулю $m_i = 17$ з $\Delta Q_R = 0$ (див. рис. 2.28), то алгоритм реалізації модульної операції представлений в таблиці 2.9 та на рис. 2.43. Одиниця двійкового розряду надходить на вхід ДОС S_4, S_3, S_2 та S_1.

Таблиця 2.9

Алгоритм реалізації операції $(x_i - y_i) \bmod m_i$ (де $x_i = 01001$, $y_i = 00111$)на суматорі по модулю $m_i = 17$ з $\Delta Q_R = 0$

Номер ДОС S_i	Вміст ДОС S_i	Наявність одиниці на входах ДОС S_i	Результат операції $(x_i - y_i) \bmod m_i$
S_1	1	+1	0
S_2	1	+1	1
S_3	0	+1	0
S_4	0	+1	0
S_5	1	–	0

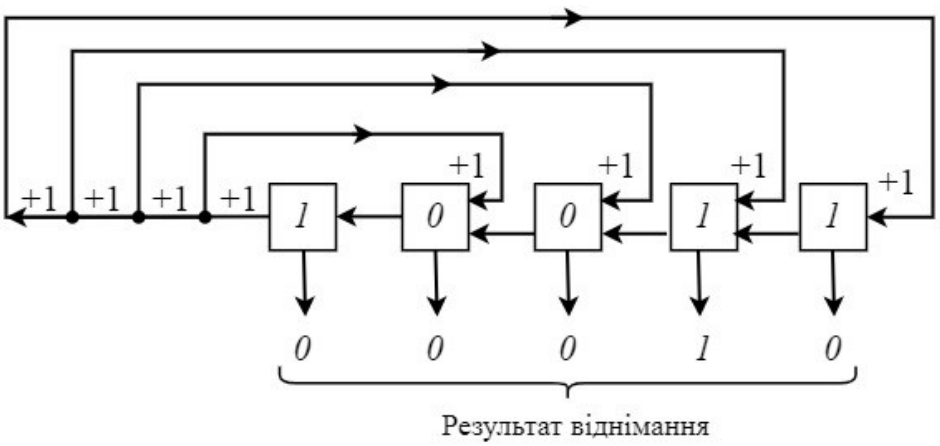

Рис. 2.43 Схема реалізації операції віднімання $(x_i - y_i) \bmod m_i$ (де $x_i = 01001$, $y_i = 00111$)на суматорі модулю $m_i = 17$ з $\Delta Q_R = 0$

У зв'язку з тим, що усі додаткові зв'язки: $X_{\downarrow 4 \uparrow 5}$, $X_{\downarrow 3 \uparrow 5}$, $X_{\downarrow 2 \uparrow 5}$ виходять з

старшого ДОС (тобто $j = k$), що забезпечує мінімальне значення величини $\Delta Q_R = 0$, корекція отриманого результату не потрібна. У підсумку проведення операції віднімання $(x_i - y_i) \bmod m_i$ отримуємо результат $S_5\, S_4\, S_3\, S_2\, S_1 = 00010$.

Якщо використовується суматор по модулю $m_i = 17$ з $\Delta Q_R = 4$ по модулю (див. рис. 2.16), то алгоритм реалізації модульної операції представлений в таблиці 2.10 та на рис. 2.44. Одиниця двійкового розряду надходить на вхід ДОС S_4, S_2 та S_1. Корекція вмісту суматора на величину $\Delta Q_R = 4$ відбувається шляхом додавання 4 у двійковому коді на вхід ДОС $S_5 - S_1$ ($S_5\, S_4\, S_3\, S_2\, S_1 =$ =00100).

Таблиця 2.10

Алгоритм реалізації операції віднімання $(x_i - y_i) \bmod m_i$ (де $x_i = 01001$, $y_i = 00111$) на суматорі по модулю $m_i = 17$ з $\Delta Q_R = 4$

Номер ДОС S_i	Вміст ДОС S_i	Наявність одиниці на входах ДОС S_i	Корекція вмісту суматора на величину ΔQ_R	Результат операції $(x_i - y_i) \bmod m_i$
S_1	1	+1	0	0
S_2	1	+1	0	1
S_3	0	–	+1	0
S_4	0	+1	0	0
S_5	1	–	0	0

У підсумку проведення операції віднімання $(x_i - y_i) \bmod m_i$ отримуємо результат $S_5\, S_4\, S_3\, S_2\, S_1 = 00010$.

Перевірка: $(9 - 7) \bmod 17 = 2$, що у двійковому коді дорівнює 00010.

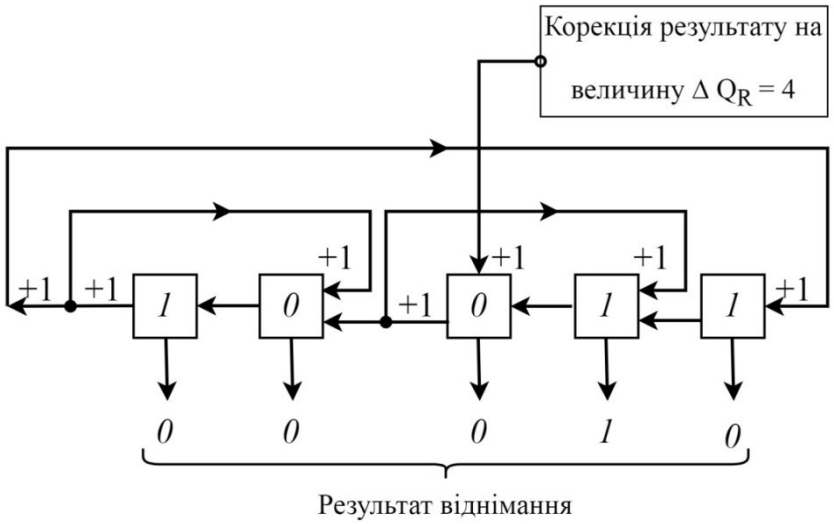

Рис. 2.44 Схема реалізації операції віднімання $(x_i - y_i) \bmod m_i$ (де $x_i = 01001$, $y_i = 00111$) на суматорі по модулю $m_i = 17$ з $\Delta Q_R = 4$

Приклад 2.12. Нехай $x_i = 11$ і $y_i = 16$. Знайти $(x_i - y_i) \bmod m_i$.

Знайдемо величину $\overline{y_i}$, для цього на суматорі реалізуємо операцію порозрядного додавання чисел $y_i + \overline{m_i}$ по модулю два, (де: $y_i = 10000$, величину $\overline{m_i}$ отримуємо порозрядною інверсією модуля $m_i = 10001$, тобто $\overline{m_i} = 01110$):

$$\begin{array}{rr} & y_i = \quad 10000 \\ + & \\ & \overline{m_i} = \quad 01110 \\ \hline y_i + \overline{m_i} = & \quad 11110 \end{array}$$

Далі знаходимо величину $\overline{y_i}$, шляхом порозрядної інверсії суми $y_i + \overline{m_i}$, $\overline{y_i} = 00001$.

Знаходимо значення позиційного підсумовування $x_i + \overline{y_i}$, де $x_i = 01011$, а $\overline{y_i} = 00001$:

$$\begin{array}{rr} & x_i = \quad 01011 \\ + & \\ & \overline{y_i} = \quad 00001 \\ \hline y_i + \overline{m_i} & \quad 01100 \end{array}$$

Так як $x_i < y_i$, значення порозрядної суми $x_i + \overline{y_i} = 01100$ і буде результатом операції $(x_i - y_i) \bmod m_i$.

Перевірка: $(11-16) \bmod 17 = 17-5 = 12$, що у двійковому коді дорівнює 01100.

Приклад 2.13. Нехай $x_i = 16$ і $y_i = 11$. Знайти $(x_i - y_i) \bmod m_i$.

Знайдемо величину $\overline{y_i}$, для цього на суматорі реалізуємо операцію порозрядного додавання чисел $y_i + \overline{m_i}$ по модулю два, (де: $y_i = 01011$, величину $\overline{m_i}$ отримуємо порозрядною інверсією модуля $m_i = 10001$, тобто $\overline{m_i} = 01110$):

$$\begin{array}{rr} & y_i = \quad 01011 \\ + & \\ & \overline{m_i} = \quad 01110 \\ \hline y_i + \overline{m_i} = & \quad 11001 \end{array}$$

Далі знаходимо величину $\overline{y_i}$, шляхом порозрядної інверсії суми $y_i + \overline{m_i}$, $\overline{y_i} = 00110$.

Знаходимо значення позиційного підсумовування $x_i + \overline{y_i}$, де $x_i = 10000$, а $\overline{y_i} = 00110$:

$$\begin{array}{rr} & x_i = \quad 10000 \\ + & \\ & \overline{y_i} = \quad 00110 \\ \hline y_i + \overline{m_i} & \quad 10110 \end{array}$$

Так як $x_i \geq y_i$, значення порозрядної суми $x_i + \overline{y_i} = 10110$ порозрядно надходить на відповідні входи ДОС $S_5 - S_1$. Таким чином, ДОС $S_5 - S_1$ суматора містить значення 10110.

Якщо використовується суматор по модулю $m_i = 17$ з $\Delta Q_R = 0$ (див. рис. 2.28), то алгоритм реалізації модульної операції представлений в таблиці 2.11 та на рис. 2.45. Одиниця двійкового розряду надходить на вхід ДОС S_4, S_3, S_2 та S_1.

Таблиця 2.11

Алгоритм реалізації операції $(x_i - y_i) \bmod m_i$ (де $x_i = 10000$, $y_i = 01011$)на суматорі по модулю $m_i = 17$ з $\Delta Q_R = 0$

Номер ДОС S_i	Вміст ДОС S_i	Наявність одиниці на входах ДОС S_i	Результат операції $(x_i - y_i) \bmod m_i$
S_1	0	+1	1
S_2	1	+1	0
S_3	1	+1	1
S_4	0	+1	0
S_5	1	–	0

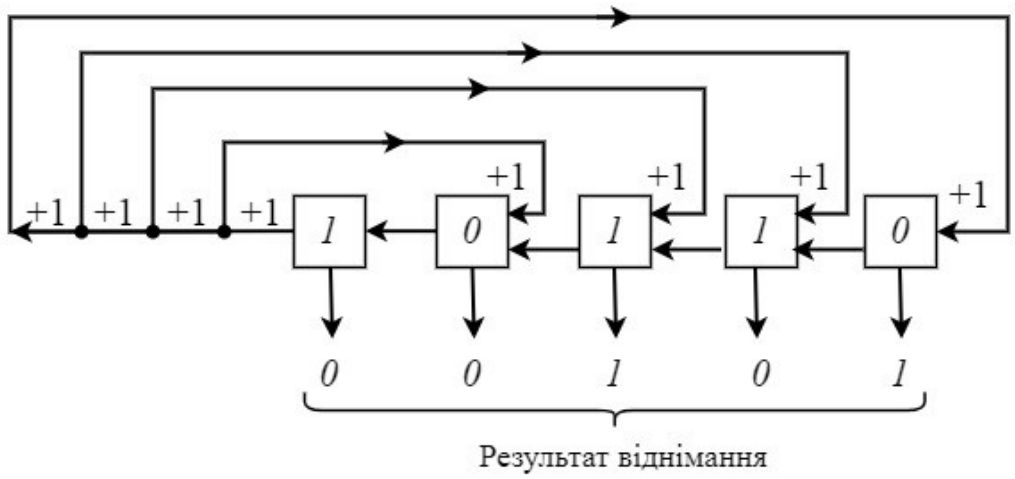

Рис. 2.45 Схема реалізації операції віднімання $(x_i - y_i) \bmod m_i$ (де $x_i = 10000$, $y_i = 01011$)на суматорі модулю $m_i = 17$ з $\Delta Q_R = 0$

У зв'язку з тим, що усі додаткові зв'язки: $X_{\downarrow 4 \uparrow 5}$, $X_{\downarrow 3 \uparrow 5}$, $X_{\downarrow 2 \uparrow 5}$ виходять з старшого ДОС (тобто $j = k$), що забезпечує мінімальне значення величини $\Delta Q_R = 0$, корекція отриманого результату не потрібна. У підсумку проведення операції віднімання $(x_i - y_i) \bmod m_i$ отримуємо результат $S_5\, S_4\, S_3\, S_2\, S_1 = 00101$.

Якщо використовується суматор по модулю $m_i = 17$ з $\Delta Q_R = 4$ (див. рис. 2.16), то алгоритм реалізації модульної операції представлений в таблиці 2.12 та

на рис. 2.46. Одиниця двійкового розряду надходить на вхід ДОС S_4, S_2 та S_1. Корекція вмісту суматора на величину $\Delta Q_R = 4$ відбувається шляхом додавання 4 у двійковому коді на вхід ДОС $S_5 - S_1$ ($S_5\ S_4\ S_3\ S_2\ S_1 = 00100$).

Таблиця 2.12

Алгоритм реалізації операції віднімання $(x_i - y_i) \bmod m_i$ (де $x_i = 10000$, $y_i = 01011$) на суматорі по модулю $m_i = 17$ з $\Delta Q_R = 4$

Номер ДОС S_i	Вміст ДОС S_i	Наявність одиниці на входах ДОС S_i	Корекція вмісту суматора на величину ΔQ_R	Результат операції $(x_i - y_i) \bmod m_i$
S_1	0	+1	0	1
S_2	1	+1	0	0
S_3	1	–	+1	1
S_4	0	+1	0	0
S_5	1	–	0	0

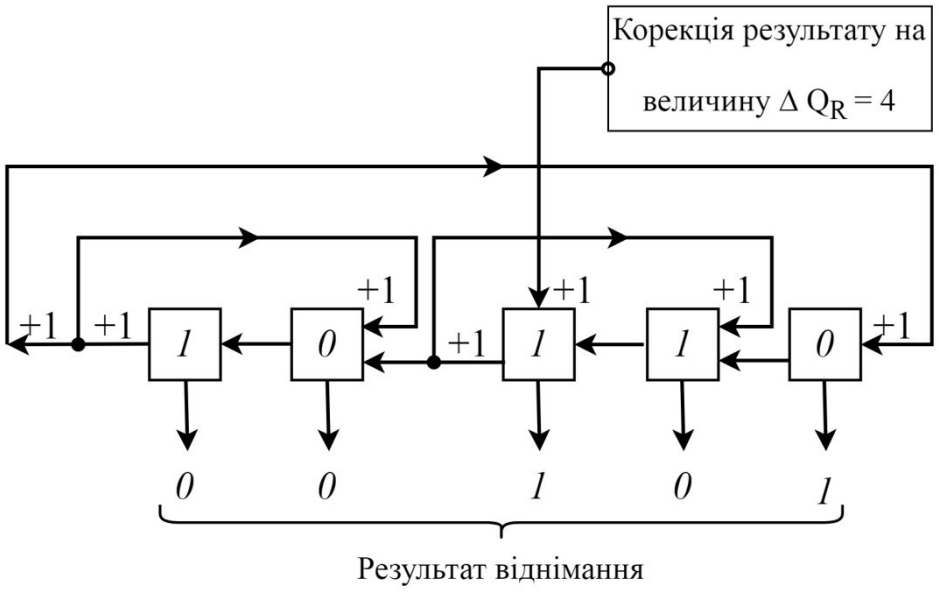

Рис. 2.46 Схема реалізації операції віднімання $(x_i - y_i) \bmod m_i$ (де $x_i = 10000$, $y_i = 10000$) на суматорі по модулю $m_i = 17$ з $\Delta Q_R = 4$

У підсумку проведення операції віднімання $(x_i - y_i) \bmod m_i$ отримуємо результат $S_5\, S_4\, S_3\, S_2\, S_1 = 00101$.

Перевірка: $(16-11) \bmod 17 = 5$, що у двійковому коді дорівнює 00101.

Приклад 2.14. Нехай $x_i = y_i = 16$. Знайти $(x_i - y_i) \bmod m_i$.

Знайдемо величину $\overline{y_i}$, для цього на суматорі реалізуємо операцію порозрядного додавання чисел $y_i + \overline{m_i}$ по модулю два, (де: $y_i = 10000$, величину $\overline{m_i}$ отримуємо порозрядною інверсією модуля $m_i = 10001$, тобто $\overline{m_i} = 01110$):

$$\begin{array}{rr} & y_i = \quad 10000 \\ + & \\ & \overline{m_i} = \quad 01110 \\ \hline y_i + \overline{m_i} = & \quad 11110 \end{array}$$

Далі знаходимо величину $\overline{y_i}$, шляхом порозрядної інверсії суми $y_i + \overline{m_i}$, $\overline{y_i} = 00001$.

Знаходимо значення позиційного підсумовування $x_i + \overline{y_i}$, де $x_i = 10000$, а $\overline{y_i} = 00001$:

$$\begin{array}{rr} & x_i = \quad 10000 \\ + & \\ & \overline{y_i} = \quad 00001 \\ \hline y_i + \overline{m_i} & \quad 10001 \end{array}$$

Так як $x_i = y_i$, значення порозрядної суми $x_i + \overline{y_i} = 10001$ порозрядно надходить на відповідні входи ДОС $S_5 - S_1$. Таким чином, ДОС $S_5 - S_1$ суматора містить значення 10011.

Якщо використовується суматор по модулю m_i з $\Delta Q_R = 0$ (див. рис. 2.28), то алгоритм реалізації модульної операції представлений в таблиці 2.13 та на рис. 2.47. Одиниця двійкового розряду надходить на вхід ДОС S_4, S_3, S_2 та S_1.

У зв'язку з тим, що усі додаткові зв'язки: $X_{\downarrow 4 \uparrow 5}$, $X_{\downarrow 3 \uparrow 5}$, $X_{\downarrow 2 \uparrow 5}$ виходять з старшого ДОС (тобто $j = k$), що забезпечує мінімальне значення величини

$\Delta Q_R = 0$, корекція отриманого результату не потрібна. У підсумку проведення операції віднімання $(x_i - y_i) \bmod m_i$ отримуємо результат $S_5\ S_4\ S_3\ S_2\ S_1 = 00000$.

Таблиця 2.13

Алгоритм реалізації операції $(x_i - y_i) \bmod m_i$ (де $x_i = y_i = 10000$)на суматорі по модулю $m_i = 17$ з $\Delta Q_R = 0$

Номер ДОС S_i	Вміст ДОС S_i	Наявність одиниці на входах ДОС S_i	Результат операції $(x_i - y_i) \bmod m_i$
S_1	1	+1	0
S_2	0	+1	0
S_3	0	+1	0
S_4	0	+1	0
S_5	1	–	0

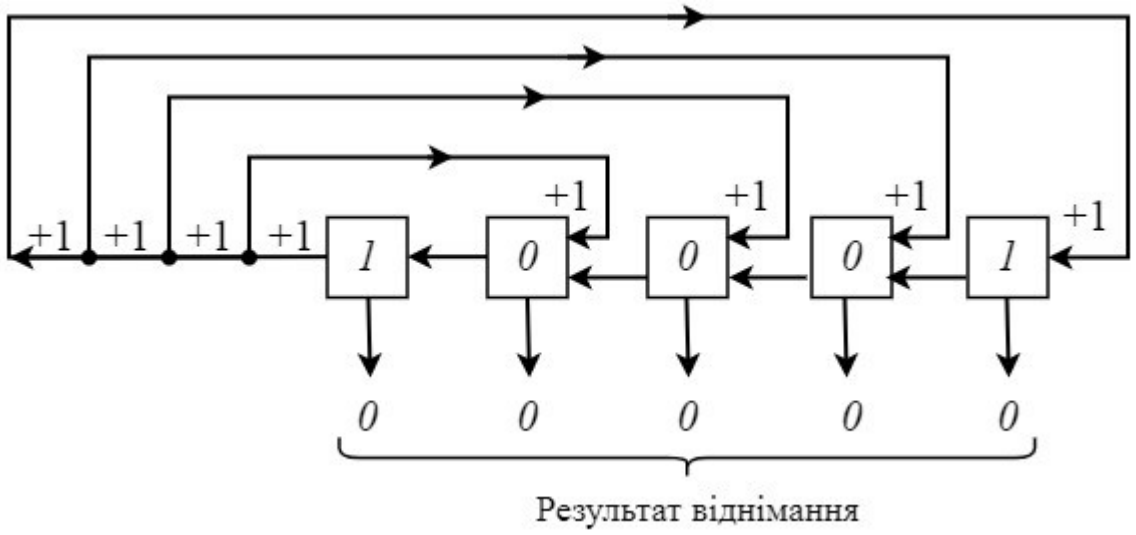

Рис. 2.47 Схема реалізації операції $(x_i - y_i) \bmod m_i$ (де $x_i = y_i = 10000$)на суматорі по модулю $m_i = 17$ з $\Delta Q_R = 0$

Якщо використовується суматор по модулю $m_i = 17$ з $\Delta Q_R = 4$по модулю (див. рис. 2.16), то алгоритм реалізації модульної операції представлений в

таблиці 2.14 та на рис. 2.48. Одиниця двійкового розряду надходить на вхід ДОС S_4, S_2 та S_1. Корекція вмісту суматора на величину $\Delta Q_R = 4$ відбувається шляхом додавання 4 у двійковому коді на вхід ДОС $S_5 - S_1$ ($S_5\,S_4\,S_3\,S_2\,S_1$ = =00100).

Таблиця 2.14

Алгоритм реалізації операції $(x_i - y_i) \bmod m_i$ (де $x_i = y_i = 10000$)на суматорі по модулю $m_i = 17$ з $\Delta Q_R = 4$

Номер ДОС S_i	Вміст ДОС S_i	Наявність одиниці на входах ДОС S_i	Корекція вмісту суматора на величину ΔQ_R	Результат операції $(x_i - y_i) \bmod m_i$
S_1	1	+1	0	0
S_2	0	+1	0	0
S_3	0	–	+1	0
S_4	0	+1	0	0
S_5	1	–	0	0

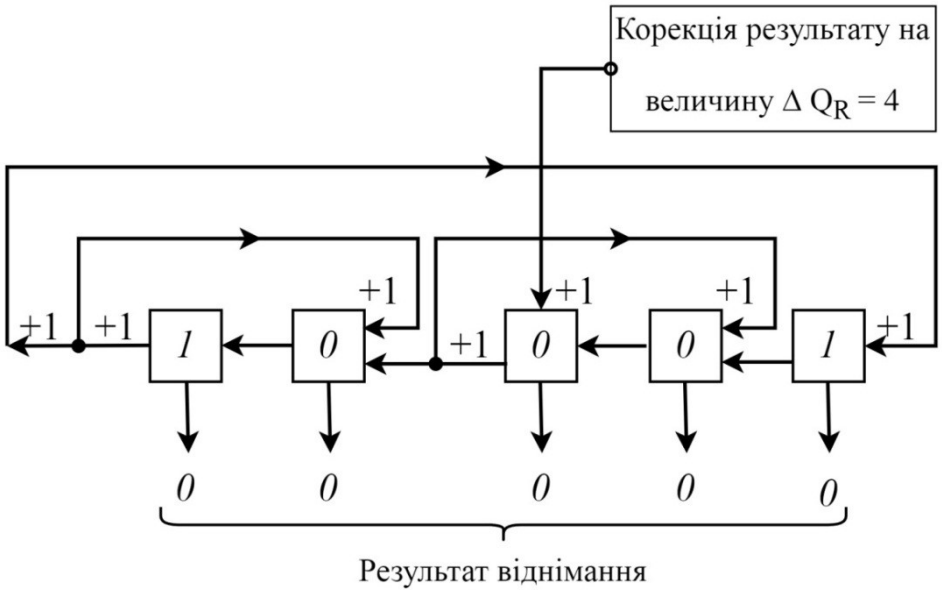

Рис. 2.48 Схема реалізації операції $(x_i - y_i) \bmod m_i$ (де $x_i = y_i = 10000$)на суматорі по модулю $m_i = 17$ з $\Delta Q_R = 4$

У підсумку проведення операції віднімання $(x_i - y_i) \bmod m_i$ отримуємо результат $S_5S_4S_3S_2S_1$=00000. Перевірка: (16-16)mod17=0.

Розроблена HDL-модель виконання операції віднімання по модулю $m_i = 17$ на суматорі по модулю на мові Verylog (див. рис. 2.49).

```
module prim1(X, Y, DIF_MOD);
input [4:0] X;
input [4:0] Y;
output [4:0] DIF_MOD;
wire [4:0] MOD = 5'b10001;
wire [4:0] NOT_MOD;
assign NOT_MOD = ~ MOD;
wire [4:0] DOP_Y;
assign DOP_Y = Y + NOT_MOD;
wire [4:0] NOT_Y;
assign NOT_Y = ~ DOP_Y;
wire [5:0] SUM;
assign SUM = X + NOT_Y;
wire [5:0] SUM_MOD;
always @*
begin
if (X < Y)
     begin
          DIF_MOD = SUM [4:0];
     end else begin
          SUM_MOD = SUM [4:0] + 5'b00001;
          SUM_MOD = SUM_MOD + 5'b00010;
          SUM_MOD = SUM_MOD + 5'b00100;
          SUM_MOD = SUM_MOD + 5'b01000;
          DIF_MOD = SUM_MOD[4:0];
     end
end
endmodule
```

Рис. 2.49 Лістинг коду HDL-моделі виконання операції віднімання по модулю $m_i = 17$ на суматорі по модулю на мові Verylog

Структурна схема в середовищі Quartus II наведено на рис. 2.50. Результат моделювання в середовищі Quartus II наведено на рис. 2.51.

Розглянуті приклади та результати моделювання реалізації операції віднімання $(x_i - y_i) \bmod m_i$ по модулю m_i СЗК для різних залишків x_i і y_i, підтверджують практичну реалізованість запропонованого методу.

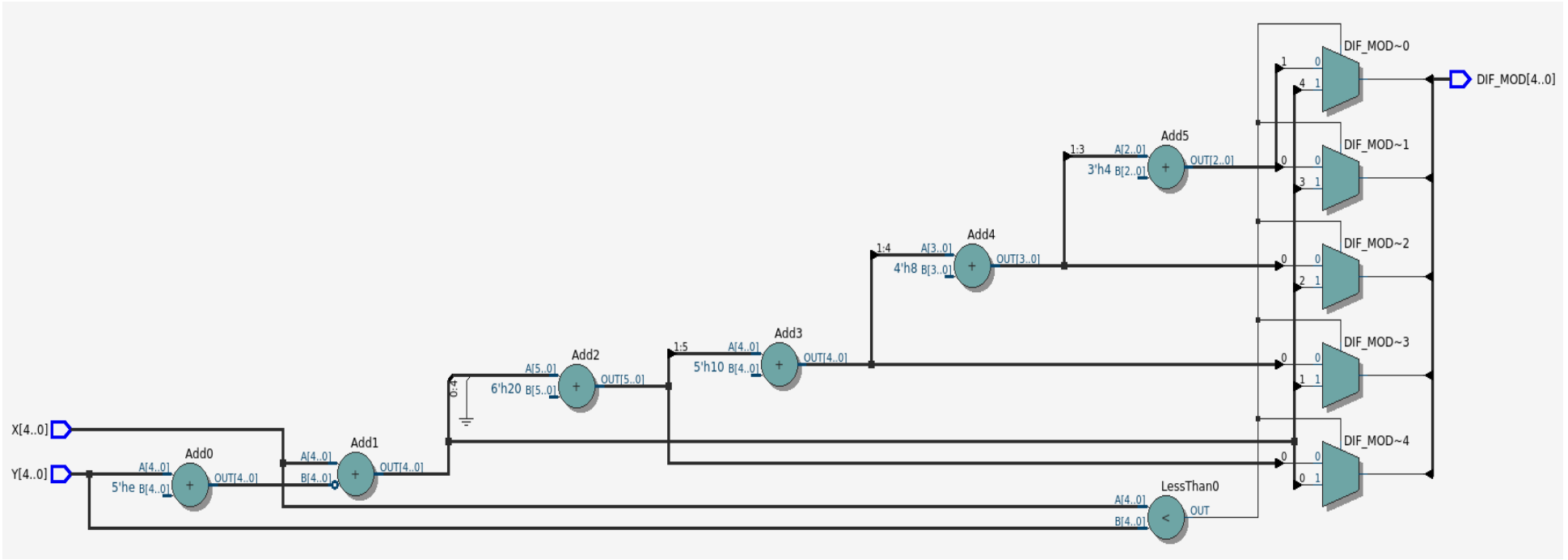

Рис. 2.50 Структурна схема HDL-моделі виконання операції віднімання по модулю $m_i = 17$ на суматорі по модулю в середовищі Quartus II

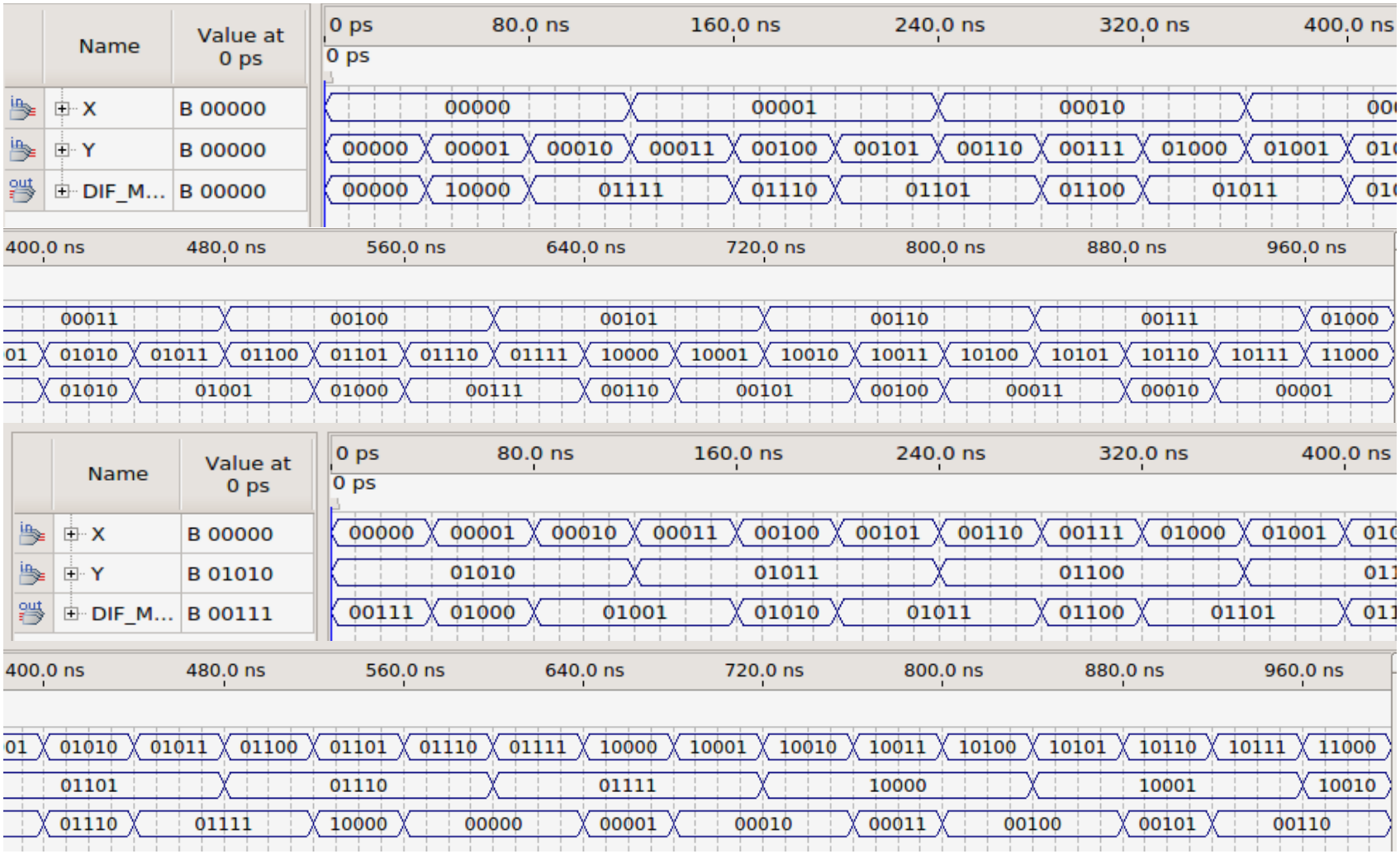

Рис. 2.51 Результат моделювання HDL-моделі виконання операції віднімання по модулю $m_i = 17$ на суматорі по модулю в середовищі Quartus II

Висновки до розділу 2

У другому розділі дістав подальший розвиток метод додавання і віднімання залишків чисел по модулю СЗК, який враховує конструкції суматорів по модулю з величиною корекції $\Delta Q_R > 0$.

Розроблена HDL-модель суматора по модулю $m_i = 17$ на мові Verylog. Розроблена суматора по модулю $m_i = 17$ в середовищі Quartus II.

Розглянуті приклади та результати моделювання реалізації методу модульного додавання для різних значень x_i і y_i по модулю m_i СЗК, підтверджують практичну реалізованість запропонованого методу.

Розроблена HDL-модель виконання операції віднімання на суматорі по модулю $m_i = 17$ на мові Verylog та структурна схема в середовищі Quartus II.

Розглянуті приклади та результати моделювання реалізації операції віднімання $(x_i - y_i) \bmod m_i$ по модулю m_i СЗК для різних залишків x_i і y_i, підтверджують практичну реалізованість запропонованого методу.

РОЗДІЛ 3. РОЗРОБКА МОДЕЛЕЙ ТА МЕТОДІВ ШВИДКОЇ ОБРОБКИ ДАНИХ НА ОСНОВІ ЗАСТОСУВАННЯ СИСТЕМИ ЗАЛИШКОВИХ КЛАСІВ

3.1 Математична модель і метод множення двох залишків комплексних чисел в СЗК

У СЗК для комплексної числової області значення залишків комплексних чисел представляються комплексними та дійсними лишками по комплексним основам. Обробка даних, представлених у комплексній області, ґрунтується на результатах теореми 1 [52].

Теорема 1. У комплексній числовій області із взаємно простими комплексними основами $\dot{m}_1,\dot{m}_2,\ldots,\dot{m}_n$ будь-яке представлене комплексне число $\dot{W}=c+di$ єдиним чином представляється в СЗК сукупністю своїх найменших комплексних лишків $\dot{w}_1,\dot{w}_2\ldots\dot{w}_n$ по основам системи.

Розглянемо операцію множення двох залишків $\dot{W}_1=c_1+d_1i$ та $\dot{W}_2=c_2+d_2i$ представлених в комплексній формі по комплексному модулю $\dot{m}=x+yi$. При цьому найбільший спільний дільник компонент *x, y* модуля $\dot{m}$ дорівнює одиниці, тобто НСД*(x, y) = 1.* У відповідності до першої фундаментальної теореми Гауса по заданому комплексному модулю $\dot{m}=x+yi$, норма якого дорівнює $E=x^2+y^2$ і для якого *x* та *y* є взаємно простими числами, комплексне число $\dot{W}=c+di$ порівняно з одним і лише одним лишком із ряду *0, 1, 2, 3, ... E-1*, тобто $\dot{W}=q\ (mod\dot{m})$, де *q* – дійсне ціле число. Визначення результату операції множення комплексних чисел $\dot{W}_1=c_1+d_1i$, $\dot{W}_2=c_2+d_2i$ по модулю $\dot{m}=x+yi$ можна замінити виконанням цієї операції над відповідними їм дійсними лишками q_1, q_2 по модулю *E*, тобто, $(q_1\cdot q_2)\bmod E$. Таким чином

$$\dot{W}_1 \cdot \dot{W}_2 = q\left(\operatorname{mod} \dot{m}\right),$$

де

$$q = q_1 \cdot q_2 (\operatorname{mod} E).$$

Значення дійсних лишків q_1 та q_2 визначаються як

$$(c_1 + \delta d_1) = q_1 \left(\operatorname{mod} E\right),$$

$$(c_2 + \delta d_2) = q_2 \left(\operatorname{mod} E\right),$$

де

$$\delta = uy - vx.$$

Вираз $uy - vx.$, за допомогою якого встановлюється відповідність між комплексним та дійсним лишком по модулю $x + yi$, називається коефіцієнтом ізоморфізма.

Значення цілих чисел u і v визначаються з відомого в теорії чисел співвідношення $ux + vy = 1$.

Ізоморфізм між комплексними числами та їх дійсними лишками дає можливість реалізувати операцію модульного множення в комплексній області за допомогою алгоритмів, які реалізують операцію модульного множення в дійсній області. Реалізацію операції множення двох дійсних лишків q_1 та q_2 по модулю E доцільно реалізувати табличним методом в якому використовуються властивості симетрії арифметичної таблиці відносно діагоналей, вертикалі та горизонталі. Це і визначає можливість реалізації в схемі модульного множення лише 0,25 повної таблиці. При цьому використовується код табличного множення [53-57].

Розглянемо загальну математичну модель реалізації операції множення двох дійсних лишків r_i та s_i табличним методом по модулю m_i (m_i - просте число). Побудуємо таблицю з числових значень добутку $(r_i \cdot s_i) \operatorname{mod} m_i$, де чисельні значення r_i будуть відкладені по горизонталі, а чисельні значення другого множника s_i - по вертикалі (таблиця 3.1). В місцях перетину будемо вказувати значення $t = (r_i \cdot s_i) \operatorname{mod} m_i$.

Таблиця 3.1

Таблиця реалізації арифметичної операції модульного множення

$$(r_i \cdot s_i) \bmod m_i$$

s \ r	r_0	…	$r_{\frac{m-1}{2}}$	$r_{\frac{m+1}{2}}$	…	r_{m-1}
s_0	t_{00}	…	$t_{\left(\frac{m-1}{2}\right)0}$	$t_{\left(\frac{m+1}{2}\right)0}$	…	$t_{(m-1)0}$
…	…	…	…	…	…	…
$s_{\frac{m-1}{2}}$	$t_{0\left(\frac{m-1}{2}\right)}$	…	$t_{\left(\frac{m-1}{2}\right)\left(\frac{m-1}{2}\right)}$	$t_{\left(\frac{m+1}{2}\right)\left(\frac{m-1}{2}\right)}$	…	$t_{(m-1)\left(\frac{m-1}{2}\right)}$
$s_{\frac{m+1}{2}}$	$t_{0\left(\frac{m+1}{2}\right)}$	…	$t_{\left(\frac{m-1}{2}\right)\left(\frac{m+1}{2}\right)}$	$t_{\left(\frac{m+1}{2}\right)\left(\frac{m+1}{2}\right)}$	…	$t_{(m-1)\left(\frac{m+1}{2}\right)}$
…	…	…	…	…	…	…
s_{m-1}	$t_{0(m-1)}$	…	$t_{\left(\frac{m-1}{2}\right)(m-1)}$	$t_{\left(\frac{m+1}{2}\right)(m-1)}$	…	$t_{(m-1)(m-1)}$

Ця таблиця симетрична відносно діагоналей, вертикалі і горизонталі, які проходять між числами $\frac{m_i-1}{2}$ і $\frac{m_i+1}{2}$. Симетричність відносно лівої діагоналі визначається комутативністю операції, симетричність відносно правої діагоналі визначається тим, що

$$\left(m_i - r_i\right)\left(m_i - s_i\right) = (r_i \cdot s_i) \bmod m_i.$$

Симетричність відносно вертикалі та горизонталі визначається тим, що сума кратних чисел кратна m_i, тобто

$$r_i \cdot s_i + r_i\left(m_i - s_i\right) = 0 \bmod m_i,$$

$$r_i \cdot s_i + \left(m_i - r_i\right) s_i = 0 \bmod m_i.$$

Таким чином, щоб відновити таблицю, достатньо мати інформацію тільки про її восьму часину. Звідси виникає можливість скоротити таблицю, що реалізує операцію множення.

Для вирішення поставленої задачі доцільно застосувати спеціальне

кодування чисел r_i та s_i - так званий «код табличного множення» (КТМ). Значення $r_i(s_i)$, які лежать в діапазоні $\left[0, \frac{m_i-1}{2}\right)$, можуть бути закодовані довільним способом. Значення $r_i(s_i)$, які лежать в діапазоні $\left[\frac{m_i+1}{2}, m_i-1\right)$, кодуються як $m_i - r_i(m_i - s_i)$.

Для відмінності діапазонів вводиться індекс $\lambda_r(\lambda_s)$, який визначається наступним чином:

$$\lambda_r(\lambda_s) = \begin{cases} 0, \textit{якщо } 0 \le r_i(s_i) \le \frac{m_i-1}{2} \\ 1, \textit{якщо } \frac{m_i+1}{2} \le r_i(s_i) < m_i. \end{cases},$$

Таким чином, якщо задано число $r_i(s_i)$, то для того, щоб отримати значення $m_i - r_i(m_i - s_i)$ достатньо проінвертувати індекс $\lambda_r(\lambda_s)$.

Метод визначення результату модульного множення, за допомогою КТМ, наступний: якщо два числа r_i та s_i задані в коді табличного множення $r_i = (\lambda_r, r_i')$, $s_i = (\lambda_s, s_i')$, то для того щоб отримати добуток чисел по модулю m_i, достатньо отримати добуток $(r_i' \cdot s_i') \bmod m_i$ в коді табличного множення та інвертрувати його індекс λ у випадку, якщо λ_r відрізняється від λ_s, тобто

$$(r_i \cdot s_i) \bmod m_i = (\lambda_i, (r_i' \cdot s_i') \bmod m_i),$$

де

$$\lambda_i = \begin{cases} \bar{\lambda}, \textit{якщо } \lambda_r \ne \lambda_s, \\ \lambda, \textit{якщо } \lambda_r = \lambda_s. \end{cases}$$

На основі загальної математичної моделі реалізації операції множення двох дійсних лишків r_i та s_i табличним методом по модулю m_i синтезуємо математичну модель множення двох дійсних лишків q_1 та q_2 по модулю E.

Операнди q_1 та q_2, представляються у вигляді КТМ $q_1 = (\lambda_1, q_1')$, $q_2 = (\lambda_2, q_2')$. Відзначимо що, $1 \le q_1, q_2 \le E-1$. Тоді, якщо E - норма комплексного модуля $\dot{m} = x + yi$ – парне число, то

$$q_1^{'}\left(q_2^{'}\right) = \begin{cases} q_1\left(q_2\right), \text{ якщо } 1 \le q_1\left(q_2\right) \le \dfrac{E}{2} \ , \\ E - q_1\left(E - q_2\right), \text{ якщо } \dfrac{E}{2} < q_1\left(q_2\right) \le E - 1, \end{cases}$$

$$\lambda_{q_1}\left(\lambda_{q_2}\right) = \begin{cases} 0, \text{ якщо } 1 \le q_1\left(q_2\right) \le \dfrac{E}{2} \ , \\ 1, \text{ якщо } \dfrac{E}{2} < q_1\left(q_2\right) \le E - 1, \end{cases} \tag{3.1}$$

при цьому $1 \le q_1\left(q_2\right) \le \dfrac{E}{2}$.

Якщо E - норма комплексного модуля $\dot{m} = x + yi$ – непарне число, то

$$q_1^{'}\left(q_2^{'}\right) = \begin{cases} q_1\left(q_2\right), \text{ якщо } 1 \le q_1\left(q_2\right) \le \dfrac{E-1}{2} \ , \\ E - q_1\left(E - q_2\right), \text{ якщо } \dfrac{E+1}{2} \le q_1\left(q_2\right) \le E - 1, \end{cases}$$

$$\lambda_{q_1}\left(\lambda_{q_2}\right) = \begin{cases} 0, \text{ якщо } 1 \le q_1\left(q_2\right) \le \dfrac{E-1}{2} \ , \\ 1, \text{ якщо } \dfrac{E+1}{2} \le q_1\left(q_2\right) \le E - 1, \end{cases} \tag{3.2}$$

при цьому $1 \le q_1\left(q_2\right) \le \dfrac{E-1}{2}$.

Тепер визначимо результат множення q двох дійсних лишків q_1 та q_2 по модулю E

$$q = q_1 \cdot q_2 \left(\operatorname{mod} E\right)$$

в коді табличного множення:

$$q_1 q_2 \left(\operatorname{mod} E\right) = \begin{cases} q_1^{'} \cdot q_2^{'} \left(\operatorname{mod} E\right), \text{ якщо } \left(\lambda_{q_1} + \lambda_{q_2}\right) \operatorname{mod} 2 = 0, \\ E - \left(q_1^{'} \cdot q_2^{'} \left(\operatorname{mod} E\right)\right), \text{ якщо } \left(\lambda_{q_1} + \lambda_{q_2}\right) \operatorname{mod} 2 = 1. \end{cases} \tag{3.3}$$

З урахуванням формул (3.1)-(3.3) результатом множення двох

комплексних $\dot{W}_1 = c_1 + d_1 i$ та $\dot{W}_2 = c_2 + d_2 i$ по комплексному модулю $\dot{m} = x + yi$ буде комплексне число ізоморфне дійсному лишку $q = q_1 q_2 \left(\mathrm{mod}\, E\right)$.

Сукупність формул (3.1)-(3.3) є математичною моделлю процесу табличної реалізації операції множення двох залишків комплексних чисел по комплексному модулю. На основі цієї ММ у роботі вдосконалено метод табличної реалізації операції множення за рахунок можливості виконання множення комплексних чисел по модулю $\dot{m} = x + yi$ (рис. 3.1) [58-61].

1	Визначити норму комплексного модуля $\dot{m} = x + yi$ $E = x^2 + y^2$
2	Визначити значення коефіцієнту ізоморфізму $\delta = uy - vx$. Значення цілих чисел *u* і *v* визначаються з рівності $ux + vy = 1$.
3	Визначити значення найменших дійсних додатних лишків q_1 та q_2 ізоморфних комплексним числам $\dot{W}_1 = c_1 + d_1 i$ та $\dot{W}_2 = c_2 + d_2 i$ розв'язавши порівняння: $\left(x_1 + \delta y_1\right) = q_1 \left(\mathrm{mod}\, E\right), \left(x_2 + \delta y_2\right) = q_2 \left(\mathrm{mod}\, E\right),$ де $\delta = uy - vx$ - коефіцієнт ізоморфізму, $E = x^2 + y^2$ - норма комплексного модуля $\dot{m} = x + yi$ $\dot{W}_1 \sim q_1, \dot{W}_2 \sim q_2.$
4	Представити найменші дійсні додатні лишки q_1 та q_2 ізоморфних комплексним числам $\dot{W}_1 = c_1 + d_1 i$ та $\dot{W}_2 = c_2 + d_2 i$ ($\dot{W}_1 \sim q_1, \dot{W}_2 \sim q_2$), в коді табличного множення: $q_1 = \left(\lambda_1, q_1'\right), q_2 = \left(\lambda_2, q_2'\right).$ $1 \le q_1, q_2 \le E - 1.$

Рис. 3.1 Метод множення двох залишків комплексних чисел по модулю $\dot{m} = x + yi$ (початок)

	Якщо E - норма комплексного модуля $\dot{m} = x + yi$ – парне число, то $q_1^{'}\left(q_2^{'}\right) = \begin{cases} q_1(q_2), \text{ якщо } 1 \le q_1(q_2) \le \frac{E}{2}, \\ E - q_1(E - q_2), \text{ якщо } \frac{E}{2} < q_1(q_2) \le E - 1, \end{cases}$ $\lambda_{q_1}\left(\lambda_{q_2}\right) = \begin{cases} 0, \text{ якщо } 1 \le q_1(q_2) \le \frac{E}{2}, \\ 1, \text{ якщо } \frac{E}{2} < q_1(q_2) \le E - 1, \end{cases}$ при цьому $1 \le q_1(q_2) \le \frac{E}{2}$.
	Якщо E - норма комплексного модуля $\dot{m} = p + qi$ – непарне число, то $q_1^{'}\left(q_2^{'}\right) = \begin{cases} q_1(q_2), \text{ якщо } 1 \le q_1(q_2) \le \frac{E-1}{2}, \\ E - q_1(E - q_2), \text{ якщо } \frac{E+1}{2} \le q_1(q_2) \le E - 1, \end{cases}$ $\lambda_{q_1}\left(\lambda_{q_2}\right) = \begin{cases} 0, \text{ якщо } 1 \le q_1(q_2) \le \frac{E-1}{2}, \\ 1, \text{ якщо } \frac{E+1}{2} \le q_1(q_2) \le E - 1, \end{cases}$ при цьому $1 \le q_1(q_2) \le \frac{E-1}{2}$.
5	Визначити результат множення q двох дійсних лишків q_1 та q_2 по модулю E $q = q_1 \cdot q_2 \left(\bmod E\right)$ в коді табличного множення: $q_1 q_2 \left(\bmod E\right) = \begin{cases} q_1^{'} \cdot q_2^{'} \left(\bmod E\right), \text{ якщо } (\lambda_{q_1} + \lambda_{q_2}) \bmod 2 = 0, \\ E - \left(q_1^{'} \cdot q_2^{'} \left(\bmod E\right)\right), \text{ якщо } (\lambda_{q_1} + \lambda_{q_2}) \bmod 2 = 1. \end{cases}$
6	За допомогою ПЗП вибрати комплексне число ізоморфне дійсному лишку $q = q_1 q_2 \left(\bmod E\right)$, яке і буде результатом операції $\dot{W}_1 \cdot \dot{W}_2 \left(\bmod \dot{m}\right)$.

Рис. 3.1 Метод множення двох залишків комплексних чисел по модулю $\dot{m} = x + yi$ (завершення)

Розглянемо в загальному вигляді приклади конкретного використання розробленого методу множення двох залишків комплексних чисел по модулю $\dot{m} = 3 + 2i$.

Спочатку визначимо значення норми комплексного модуля $\dot{m} = 3 + 2i$ дорівнює $E = x^2 + y^2 = 3^2 + 2^2 = 13$.

Тепер визначимо значення коефіцієнту ізоморфізму $\delta = uy - vx = u \cdot 2 - v \cdot 3$. Значення u і v визначаються з відомого в теорії чисел співвідношення $ux + vy = 1$, тобто $u \cdot 3 + v \cdot 3 = 1$. Шляхом підбору (перебору) визначаємо, що $u = 1$, а $v = -1$. Таким чином, $\delta = 1 \cdot 2 - (-1) \cdot 3 = 2 + 3 = 5$.

Визначимо вхідні значення найменших дійсних лишків q_i, ізоморфних найменшим комплексним лишкам $\dot{W} = c + di$.

Для $\dot{W} = 0 + 0i, \quad 0 + 5 \cdot 0 = q(\mathrm{mod}13), \quad q = 0;$

для $\dot{W} = -1 + 3i, \quad -1 + 5 \cdot 3 = q(\mathrm{mod}13), \quad q = 1;$

для $\dot{W} = 0 + 3i, \quad 0 + 5 \cdot 0 = q(\mathrm{mod}13), \quad q = 2;$

для $\dot{W} = 1 + 3i, \quad 1 + 5 \cdot 3 = q(\mathrm{mod}13), \quad q = 3;$

для $\dot{W} = 2 + 3i, \quad 2 + 3 \cdot 0 = q(\mathrm{mod}13), \quad q = 4;$

для $\dot{W} = 0 + 1i, \quad 0 + 5 \cdot 1 = q(\mathrm{mod}13), \quad q = 5;$

для $\dot{W} = 1 + 1i, \quad 1 + 5 \cdot 1 = q(\mathrm{mod}13), \quad q = 6;$

для $\dot{W} = 0 + 4i, \quad 0 + 5 \cdot 4 = q(\mathrm{mod}13), \quad q = 7;$

для $\dot{W} = 1 + 4i, \quad 1 + 5 \cdot 4 = q(\mathrm{mod}13), \quad q = 8;$

для $\dot{W} = -1 + 2i, \quad -1 + 5 \cdot 2 = q(\mathrm{mod}13), \quad q = 9;$

для $\dot{W} = 0 + 2i, \quad 0 + 5 \cdot 2 = q(\mathrm{mod}13), \quad q = 10;$

для $\dot{W} = 1 + 2i, \quad 1 + 5 \cdot 2 = q(\mathrm{mod}13), \quad q = 11;$

для $\dot{W} = 2 + 2i, \quad 2 + 5 \cdot 2 = q(\mathrm{mod}13), \quad q = 12.$

Результати обчислень найменших дійсних значень залишків q зведені в таблиці 3.2.

Таблиця 3.2

Результати обчислень найменших дійсних значень залишків q

Найменші комплексні лишки $\dot{W} = c + di$	Коефіцієнту ізоморфізму δ	Дійсні лишки
$\dot{W} = 0 + 0i$	5	$q = 0$
$\dot{W} = -1 + 3i$	5	$q = 1$
$\dot{W} = 0 + 3i$	5	$q = 2$
$\dot{W} = 1 + 3i$	5	$q = 3$
$\dot{W} = 2 + 3i$	5	$q = 4$
$\dot{W} = 0 + 1i$	5	$q = 5$
$\dot{W} = 1 + 1i$	5	$q = 6$
$\dot{W} = 0 + 4i$	5	$q = 7$
$\dot{W} = 1 + 4i$	5	$q = 8$
$\dot{W} = -1 + 2i$	5	$q = 9$
$\dot{W} = 0 + 2i$	5	$q = 10$
$\dot{W} = 1 + 2i$	5	$q = 11$
$\dot{W} = 2 + 2i$	5	$q = 12$

Запишемо усі значення найменших дійсних лишків q в коді табличного множення (табл. 3.3).

Таблиця результатів операції множення по модулю *E=13* має вигляд таблиці 3.4.

Таблиця 3.3

Значення найменших дійсних лишків *h* в коді табличного множення

Дійсні лишки q		$q=(\delta, q')$		
		δ	q'	
1	0001	0	1	0001
2	0010	0	2	0010
3	0011	0	3	0011
4	0100	0	4	0100
5	0101	0	5	0101
6	0110	0	6	0110
7	0111	1	6	0110
8	1000	1	5	0101
9	1001	1	4	0100
10	1010	1	3	0011
11	1011	1	2	0010
12	1100	1	1	0001

Таблиця 3.4

Таблиця результатів операції множення по модулю *E=13*

	Перший операнд						
Другий операнд		1	2	3	4	5	6
	1	1	2	3	4	5	6
	2	2	4	6	8	10	12
	3	3	6	9	12	2	5
	4	4	8	12	3	7	11
	5	5	10	2	7	12	4
	6	6	12	5	11	4	10

Приклад 3.1. Знайдемо добуток $\dot{W}=1+3i$ та $\dot{W}=2+2i$.

$$1+3i \sim 3$$

$$2+2i \sim 12$$

Представити найменші дійсні додатні лишки q_1 та q_2 в коді табличного множення $q_1 = (\delta_1,\ q_1'),\ q_2 = (\delta_2,\ q_2')$.

$$q_1 = (0,\ 3)$$

$$q_2 = (1,\ 1)$$

Визначити результат множення q двох дійсних лишків q_1 та q_2 по модулю E в коді табличного множення:

$$q_1 \cdot q_2 \left(\mathrm{mod}\, E\right) = 13 - 3 = 10,$$

тому що $(\lambda_{q_1} + \lambda_{q_2})\mathrm{mod}\, 2 = 1.$

У таблиці знаходимо комплексне число, яке ізоморфне дійсному лишку $q = 10$, тобто результат операції $\dot{W}_1 \cdot \dot{W}_2 \left(\mathrm{mod}\dot{m}\right) = 2i.$

Перевірка: $\dot{W}_1 \cdot \dot{W}_2 = (1+3i)\cdot(2+2i) = -4+8i.$ Визначемо найменший комплексний лишок $a+bi$ числа $-4+8i$ по комплексному модулю $\dot{m} = 3+2i.$ Напишемо систему рівнянь

$$cx + dy \equiv ax + by (\mathrm{mod}\, x^2 + y^2),$$

$$dx - cy \equiv bx - ay\ (\mathrm{mod}\, x^2 + y^2).$$

в умовах прикладу (c=-4; d=8; x=3; y=2; x^2+y^2=13):

$$-4\cdot 3 + 8\cdot 2 \equiv a\cdot 3 + b\cdot 2 \left(\mathrm{mod}13\right),$$

$$8\cdot 3 + 4\cdot 2 \equiv b\cdot 3 - a\cdot 2\ (\mathrm{mod}13).$$

Звідси отримуємо рівняння

$$\begin{cases} 3a + 2b = 4, \\ 3b - 2a = 6. \end{cases}$$

Розв'язок цієї системи: a=0, b=3, тобто $2i$ - найменший комплексний лишок комплексного числа $-4+8i$ по модулю $\dot{m} = 3+2i.$ Тобто множення виконано вірно.

Приклад 3.2. Знайдемо добуток $\dot{W}_1 = -1+3i$ та $\dot{W}_1 = i$.

$$-1+3i \sim 1$$

$$i \sim 5$$

Представити найменші дійсні додатні лишки q_1 та q_2 в коді табличного множення $q_1 = (\lambda_1, q_1'),\ q_2 = (\lambda_2, q_2')$.

$$q_1 = (0, 1)$$
$$q_2 = (1, 5)$$

Визначити результат множення q двох дійсних лишків q_1 та q_2 по модулю E в коді табличного множення:

$$q_1 \cdot q_2 \left(\mathrm{mod}\, E\right) = 5,$$

тому що $(\lambda_{q_1} + \lambda_{q_2})\mathrm{mod}2 = 0.$

У таблиці знаходимо комплексне число, яке ізоморфне дійсному лишку $q = 2$, тобто результат операції $\dot{W}_1 \cdot \dot{W}_2 \left(\mathrm{mod}\dot{m}\right) = i.$

Перевірка: $\dot{W}_1 \cdot \dot{W}_2 = \left(-1 + 3i\right) \cdot i = -3 - i.$ Визначемо найменший комплексний лишок $a + bi$ числа $-3 - i$ по комплексному модулю $\dot{m} = 3 + 2i.$ Напишемо систему рівнянь

$$cx + dy \equiv ax + by (\mathrm{mod}\, x^2 + y^2),$$
$$dx - cy \equiv bx - ay\ (\mathrm{mod}\, x^2 + y^2).$$

в умовах прикладу (c=-3; d=-1; x=3; y=2; x^2+y^2=13):

$$-3 \cdot 3 - 1 \cdot 2 \equiv a \cdot 3 + b \cdot 2 \left(\mathrm{mod}13\right),$$
$$-1 \cdot 3 + 3 \cdot 2 \equiv b \cdot 3 - a \cdot 2\ (\mathrm{mod}13).$$

Звідси отримуємо рівняння

$$\begin{cases} 3a + 2b = 2, \\ 3b - 2a = 3. \end{cases}$$

Розв'язок цієї системи: a=0, b=1, тобто i - найменший комплексний лишок комплексного числа $-3 - i$ по модулю $\dot{m} = 3 + 2i.$ Тобто множення виконано вірно.

Приклад 3.3. Знайдемо добуток $\dot{W}_1 = 2 + 3i$ та $\dot{W}_2 = 4i$.

$$2 + 3i \sim 4$$
$$4i \sim 7$$

Представити найменші дійсні додатні лишки q_1 та q_2 в коді табличного множення $q_1 = (\lambda_1, q_1'),\ q_2 = (\lambda_2, q_2')$.

$$q_1 = (0, 4)$$

$$q_2 = (1, 6)$$

Визначити результат множення q двох дійсних лишків q_1 та q_2 по модулю E в коді табличного множення:

$$q_1 \cdot q_2 \left(\mathrm{mod}\, E\right) = 13 - 11 = 2,$$

тому що $(\lambda_{q_1} + \lambda_{q_2})\mathrm{mod}2 = 1.$

У таблиці знаходимо комплексне число, яке ізоморфне дійсному лишку $q = 2$, тобто результат операції $\dot{W}_1 \cdot \dot{W}_2 \left(\mathrm{mod}\dot{m}\right) = 3i.$

Перевірка: $\dot{W}_1 \cdot \dot{W}_2 = \left(2 + 3i\right) \cdot \left(4i\right) = -12 + 8i.$ Визначемо найменший комплексний лишок $a + bi$ числа $-12 + 8i$ по комплексному модулю $\dot{m} = 3 + 2i.$ Напишемо систему рівнянь

$$cx + dy \equiv ax + by (\mathrm{mod}\, x^2 + y^2),$$

$$dx - cy \equiv bx - ay \ (\mathrm{mod}\, x^2 + y^2).$$

в умовах прикладу (c=-12; d=8; x=3; y=2; x^2+y^2=13):

$$-12 \cdot 3 + 8 \cdot 2 \equiv a \cdot 3 + b \cdot 2 \left(\mathrm{mod}13\right),$$

$$8 \cdot 3 + 12 \cdot 2 \equiv b \cdot 3 - a \cdot 2 \ (\mathrm{mod}13).$$

Звідси отримуємо рівняння

$$\begin{cases} 3a + 2b = 6, \\ 3b - 2a = 9. \end{cases}$$

Розв’язок цієї системи: a=0, b=3, тобто $3i$ - найменший комплексний лишок комплексного числа $-12 + 8i$ по модулю $\dot{m} = 3 + 2i.$ Тобто множення виконано вірно.

Перевірка показала, що використання даного метода дозволяє отримати достовірний результат операції множення двох комплексних лишків по комплексному модулю.

3.2 Математична модель процесу піднесення цілих чисел до довільного степеня натурального числа в СЗК

Існує численний клас алгоритмів і задач (задачі реалізації криптоалгоритмів, оптимізаційні задачі, обчислювальні задачі великої розмірності, тощо), де крім виконання цілочисельних арифметичних операцій складання, віднімання, множення, піднесення цілих чисел до довільного степеня натурального числа по модулю, тощо, в додатному числовому діапазоні, існує необхідність реалізації перерахованих вище арифметичних операцій у від'ємному числовому діапазоні. Необхідність виконання цих операцій у від'ємному числовому діапазоні суттєво знижує загальну ефективність використання СЗК як системи числення програмно-апаратних систем і комплексів з елементами штучного інтелекту [62-63].

Таким чином, актуальні та важливі дослідження, присвячені розробці методу піднесення цілих чисел за довільним модулем СЗК до степіня натурального числа. Однак існуючі методи реалізації модульної операції піднесення цілих чисел до степіня не завжди застосовні для їх реалізації у від'ємному числовому діапазоні. Це зумовлено в основному тим, що відсутня проста математична модель процесу піднесення цілих чисел до довільного степіня натурального числа в СЗК, як у додатному, так і в від'ємному числових діапазонах. Дослідження, проведені в цій роботі, присвячені насамперед синтезу математичної моделі процесу піднесення цілих чисел до довільного степеня натурального числа в СЗК, як у додатному, так і в від'ємному числових діапазонах [63-67].

Синтез математичної моделі. Відомо, що по вигляду числа $U_{СЗК} = (u_1, u_2 ... u_n)$, представленого СЗК, не можна визначити його приналежність до додатного чи від'ємного числового діапазону. Розглянемо варіант представлення чисел у СЗК, як в додатному, так і у від'ємному числових діапазонах.

Для реалізації процесу виконання операції піднесення залишків цілих

чисел за довільним модулем СЗК до степеня натурального числа, як в додатному, так і у від'ємному числових діапазонах, пропонується представити число $U_{СЗК} = (u_1, u_2 ... u_n)$ у штучній формі (ШФ) $U'_{СЗК}$ [1]:

$$\begin{cases} U'_{СЗК} = \dfrac{M}{2} + \left|U'_{СЗК}\right|, \text{ якщо } U \geq 0, \\ U'_{СЗК} = \dfrac{M}{2} - \left|U'_{СЗК}\right|, \text{ якщо } U < 0, \end{cases}$$

де $M = \prod_{i=1}^{n} m_i$ - обсяг діапазону представлення чисел у використовуваній СЗК. Крім цього, оброблювані числа в степені $U^{\mu}_{СЗК}$ та $(U'_{СЗК})^{\mu}$ в СЗК знаходяться у відповідних числових інтервалах

$$\begin{cases} -\dfrac{M}{2} \leq U^{\mu}_{СЗК} \leq \dfrac{(M-1)}{2}, \\ 0 \leq (U'_{СЗК})^{\mu} \leq M - 1. \end{cases}$$

Зазначимо, що у СЗК виконуються такі рівності

$$M = \prod_{i=1}^{n} m_i = (0, 0, ... 0) \tag{3.4}$$

та

$$\frac{M}{2} = \prod_{i=2}^{n} m_i = (1, 0 ... 0) \tag{3.5}$$

(n - кількість основ СЗК).

В даний час відсутній ефективний метод піднесення залишків цілих чисел, представлених в СЗК, за довільним модулем до степеня натурального

числа, одночасно, як в додатному, так і у від'ємному числових діапазонах на основі їх представлення в ШФ. Ця обставина суттєво звужує область ефективного застосування СЗК. Таким чином, актуальні дослідження в галузі створення методів та алгоритмів піднесення залишків цілих чисел, представлених у СЗК, за довільним модулем до степеня натурального числа, як в додатному, так і у від'ємному числових діапазонах на основі їх представлення в ШФ.

Щоб розробити метод процесу піднесення цілих чисел до довільного степеня натурального числа в СЗК необхідно синтезувати математичну модель $(U_{СЗК}^{\mu})' = f(U'_{СЗК})$ процесу піднесення залишків цілих чисел $U_{СЗК}$, представлених в СЗК, за довільним модулем m_i у степінь μ натурального числа. У цьому випадку треба отримати аналітичний вираз $(U_{СЗК}^{\mu})' = f(U'_{СЗК})$, який визначає залежність результату $U_{СЗК}^{\mu}$ операції піднесення числа $U_{СЗК}$ в СЗК до степені μ, представленого в ШФ, від значення числа $U'_{СЗК}$, безпосередньо представленого в ШФ.

Доказ математичної моделі. Покажемо, що в якості ММ процесу піднесення цілих чисел $U_{СЗК}$ до довільного степеня μ натурального числа в СЗК доцільно вважати математичний вираз

$$(U_{СЗК}^{\mu})' = (U'_{СЗК})^{\mu} \tag{3.6}$$

Доведемо вираз (3) методом математичної індукції по μ.

Перший етап. Перевіримо правильність виразу (3.6) для мінімального значення $\mu = 2 = \min$, тобто під час піднесення чисел до квадрату.

Відповідно до визначення ШФ чисел в СЗК маємо, що

$$U'_{СЗК} = \frac{M}{2} + U_{СЗК} \text{ та } (U_{СЗК}^{\mu})' = \frac{M}{2} + U_{СЗК}^{\mu}. \tag{3.7}$$

З урахуванням числових діапазонів зміни величин $U_{СЗК}$ і $U'_{СЗК}$ співвідношення (3.7) можна представити у вигляді

$$(U^{\mu}_{СЗК})' = \left(\frac{M}{2} + U^{\mu}_{СЗК} \right) \bmod M \,. \tag{3.8}$$

Проведемо такі числові перетворення

$$\begin{gathered}(U_{СЗК})^2 = U'_{СЗК} \cdot U'_{СЗК} = (\frac{M}{2} + U_{СЗК}) \cdot (\frac{M}{2} + U_{СЗК}) = \\ = \frac{M}{2} \cdot \frac{M}{2} + U_{СЗК} \cdot \frac{M}{2} + U_{СЗК} \cdot \frac{M}{2} + U^2_{СЗК} = U^2_{СЗК} + U_{СЗК} \cdot M + \frac{M}{2} \cdot \frac{M}{2} .\end{gathered} \tag{3.9}$$

Враховуючи вирази (3.4) і (3.5) отримаємо, що

$$U_{СЗК} \cdot M = (u_1, u_2 ... u_n) \times (0, 0 \ldots 0) = (0, 0 \ldots 0) = 0 \text{ та}$$

$$\frac{M}{2} \cdot \frac{M}{2} = (1, 0 \ldots 0) \times (1, 0 ... 0) = (1, 0 \ldots 0) = \frac{M}{2} .$$

У цьому випадку вираз (3.9) представляється у вигляді

$$(U'_{СЗК})^2 = U^2_{СЗК} + \frac{M}{2} . \tag{3.10}$$

З іншого боку, маємо, що

$$(U'_{СЗК})^2 = \frac{M}{2} + U^2_{СЗК} \text{ або } U^2_{СЗК} = (U'_{СЗК})^2 - \frac{M}{2} . \tag{3.11}$$

Підставляючи значення $U^2_{СЗК}$ (3.11) у співвідношення (3.10) отримаємо, що

$$(U'_{(СЗК)})^2 = (U^2_{(СЗК)})' - \frac{M}{2} + \frac{M}{2} \text{ або } (U'_{(СЗК)})^2 = (U^2_{(СЗК)})'. \quad (3.12)$$

Аналітичне співвідношення (3.12) є ММ процесу піднесення залишків цілих чисел до довільного степеня натурального числа СЗК у різних числових областях.

Другий етап. Припустимо, що ММ справедлива довільного припустимого значення μ, тобто $(U^{\mu}_{СЗК})' = (U'_{СЗК})^{\mu}$.

Третій етап. Покажемо, що вираз (3) справедлив і для довільного допустимого значення μ+1, тобто. виконується умова

$$(U^{\mu+1}_{СЗК})' = (U'_{СЗК})^{\mu+1}. \quad (3.13)$$

З виразу (3.13) маємо, що $(U'_{СЗК})^{\mu+1} = (\frac{M}{2} + U_{СЗК})^{\mu+1}$. Розкладемо вираз $(\frac{M}{2} + U_{СЗК})^{\mu+1}$ у вигляді бінома Ньютона. Отримаємо такий аналітичний вираз

$$(\frac{M}{2} + U_{СЗК})^{\mu+1} = U^{\mu+1}_{СЗК} + (\mu+1) \cdot U^{\mu}_{СЗК} \frac{M}{2} + \ldots (\frac{M}{2})^{\mu+1}. \quad (3.14)$$

З урахуванням співвідношень (3.4) і (3.5), аналіз виразу (3.14) показав, що при приведенні подібних членів залишаються два члени $U^{\mu+1}_{СЗК}$ та $\frac{M}{2}$. Інші члени виразу (3.14) будуть нульовими. В цьому випадку $U^{\mu+1}_{СЗК} + \frac{M}{2} = (U^{\mu+1}_{СЗК})'$. Таким чином, виконується умова (3.13), тобто отримана ММ $(U^{\mu}_{СЗК})' = (U'_{СЗК})^{\mu}$ процесу піднесення цілих чисел $U_{СЗК}$ до довільного степеня натурального числа в СЗК.

Сукупність співвідношень (3.4)-(3.14) є математичною моделлю процесу піднесення цілих чисел до довільного степеня натурального числа у системі залишкових класів. На основі цієї ММ у роботі вдосконалено метод піднесення

цілих чисел до довільного степеня натурального числа в СЗК за рахунок можливості виконання операції піднесення цілих чисел до степеня, як у додатному, так і в від'ємному числових діапазонах (рис. 3.2) [63-67].

<table>
<tr><td>1</td><td>Представлення вихідних даних для реалізації методу піднесення залишків цілого числа $U_{СЗК}=(u_1,u_2...u_n)$ за довільним модулем $m_i\,(i=\overline{1,n})$ СЗК до степені натурального числа μ.</td></tr>
<tr><td>2</td><td>Кодування вихідних чисел $U_{СЗК}$ у кодові слова, представлені у штучній формі $U'_{СЗК}$ у вигляді
$$\begin{cases} U'_{СЗК}=\dfrac{M}{2}+\left|U_{СЗК}\right|, \text{ якщо } U_{СЗК}\ge 0, \\ U'_{СЗК}=\dfrac{M}{2}-\left|U_{СЗК}\right|, \text{ якщо } U_{СЗК}<0. \end{cases} \quad \begin{cases} -\dfrac{M}{2}\le U_{СЗК}\le\dfrac{(M-1)}{2}, \\ 0\le U'_{СЗК}\le M-1. \end{cases}$$
$$\begin{cases} (U^{\mu}_{СЗК})'=\dfrac{M}{2}+\left|U^{\mu}_{СЗК}\right|, \text{ якщо } U^{\mu}_{СЗК}\ge 0, \\ (U^{\mu}_{СЗК})'=\dfrac{M}{2}-\left|U^{\mu}_{СЗК}\right|, \text{ якщо } U^{\mu}_{СЗК}<0. \end{cases} \quad \begin{cases} -\dfrac{M}{2}\le U^{\mu}_{СЗК}\le\dfrac{(M-1)}{2}, \\ 0\le (U'_{СЗК})^{\mu}\le M-1. \end{cases}$$</td></tr>
<tr><td>3</td><td>Представлення залишків u'_i числа $U'_{СЗК}=(u'_1,u'_2...u'_n)$ в штучній формі по модулям $m_i\,(i=\overline{1,n})$ на основі використання КТМ $u'_i=[\lambda'_{u_i},(u'_i)^*]$,
<table>
<tr><td>Для m_i – парного числа:
$$\lambda'_{u_i}=\begin{cases} 0, \text{ якщо } 0\le u'_i\le\dfrac{m_i}{2}, \\ 1, \text{ якщо } \dfrac{m_i}{2}<u'_i\le m_i-1. \end{cases}$$
$$(u'_i)^*=\begin{cases} u'_i, \text{ якщо } 0\le u'_i\le\dfrac{m_i}{2}; \\ \overline{u'_i}=m_i-u'_i, \text{ якщо } \dfrac{m_i}{2}<u'_i\le m_i-1, \end{cases}$$
при цьому $0\le(u'_i)^*\le\dfrac{m_i}{2}$.</td><td>Для m_i – непарного числа:
$$\lambda'_{u_i}=\begin{cases} 0, \text{ якщо } 0\le u'_i\le\dfrac{(m_i-1)}{2}, \\ 1, \text{ якщо } \dfrac{(m_i-1)}{2}<u'_i\le m_i-1. \end{cases}$$
$$(u'_i)^*=\begin{cases} u'_i, \text{ якщо } 0\le u'_i\le\dfrac{(m_i-1)}{2}; \\ \overline{u'_i}=m_i-u'_i, \text{ якщо } \dfrac{(m_i-1)}{2}<u'_i\le m_i-1, \end{cases}$$
при цьому $0\le(u'_i)^*\le\dfrac{(m_i-1)}{2}$.</td></tr>
</table></td></tr>
</table>

Рис. 3.2 Метод піднесення цілих чисел до довільного степеня натурального числа у системі залишкових класів (початок)

4	Визначення результата $(u_i')^2 = (u_i' \cdot u_i') \bmod m_i \ \ (i = \overline{1, n})$ операції модульного множення у вигляді $\lambda_i', [(u_i') \cdot (u_i')] \bmod m_i$, при цьому $(u_i' \cdot u_i') \bmod m_i = \begin{cases} [(u_i')^* \cdot (u_i')^*] \bmod m_i, \text{ якщо } (\lambda'_{u_i'} + \lambda'_{u_i'}) = 0 \ (\bmod 2); \\ \overline{(u_i')^*} = m_i - [(u_i')^* \cdot (u_i')^*] \bmod m_i, \text{ якщо } (\lambda'_{u_i'} + \lambda'_{u_i'}) = 1 \ (\bmod 2). \end{cases}$
6	Визначеня результата операції $\left[(U'_{СЗК})^{\mu-1} \cdot U'_{СЗК} \right] \bmod M =$ $= \left\{ \left[(u_1')^{\mu-1} \right] \bmod m_1, \left[(u_2')^{\mu-1} \right] \bmod m_2 ... \left[(u_n')^{\mu-1} \right] \bmod m_n \right\} \times (u_1', u_2' ... u_n')$ піднесення залишків u_i цілого числа $U_{СЗК} = (u_1, u_2 ... u_n)$ по довільному модулю $m_i \ (i = \overline{1, n})$ СЗК в степінь μ натурального числа.
7	У відповідності до математичної моделі $\left[u_1^{\mu} (\bmod m_1), u_2^{\mu} (\bmod m_2) ... u_n^{\mu} (\bmod m_1) \right]' = (u_1', u_2' ... u_n')^{\mu}$ процесса піднесення залишків ццілих чисел по довільному модулю, реалізується операція $\left[(U'_{СЗК})^{\mu-1} \cdot U'_{СЗК} \right] \bmod M = \left\{ \left[(u_1')^{\mu-1} \right] \bmod m_1, \left[(u_2')^{\mu-1} \right] \bmod m_2 ... \right.$ $\left. ... \left[(u_n')^{\mu-1} \right] \bmod m_n \right\} \times (u_1', u_2' ... u_n')$ піднесення залишків цілих чисел по довільному модулю m_i СЗК до степеня μ натурального числа, як в додатному так і в від'ємному числовых діапазонах.

Рис. 3.2 Метод піднесення цілих чисел до довільного степеня натурального числа у системі залишкових класів (завершення)

Розглянемо приклади реалізації процесу піднесення цілих чисел до довільного степеня натурального числа для конкретної СЗК, заданої основами $m_1 = 3$, $m_2 = 4$, $m_3 = 5$, $M = 3 \cdot 4 \cdot 5 = 60$. Загальний обсяг додатних кодових слів у СЗК представлений у таблиці 3.5. У таблиці 3.6 представлена відповідність числових даних U їх ШФ U' в ПСЧ.

Наведемо приклад визначення величини значення $U_{СЗК}^{\mu}$ для конкретної СЗК, заданої основами $m_1 = 3$, $m_2 = 4$, $m_3 = 5$, де $M = 60$.

Таблиця 3.5

Таблиця кодових слів в СЗК

U в ПСЧ	U в СЗК			U в ПСЧ	U в СЗК		
	$m_1 = 3$	$m_2 = 4$	$m_3 = 5$		$m_1 = 3$	$m_1 = 4$	$m_3 = 5$
0	0	0	0	30	0	2	0
1	1	1	1	31	1	3	1
2	2	2	2	32	2	0	2
3	0	3	3	33	0	1	3
4	1	0	4	34	1	2	4
5	2	1	0	35	2	3	0
6	0	2	1	36	0	0	1
7	1	3	2	37	1	1	2
8	2	0	3	38	2	2	3
9	0	1	4	39	0	3	4
10	1	2	0	40	1	0	0
11	2	3	1	41	2	1	1
12	0	0	2	42	0	2	2
13	1	1	3	43	1	3	3
14	2	2	4	44	2	0	4
15	0	3	0	45	0	1	0
16	1	0	1	46	1	2	1
17	2	1	2	47	2	3	2
18	0	2	3	48	0	0	3
19	1	3	4	49	1	1	4
20	2	0	0	50	2	2	0
21	0	1	1	51	0	3	1
22	1	2	2	52	1	0	2
23	2	3	3	53	2	1	3
24	0	0	4	54	0	2	4
25	1	1	0	55	1	3	0
26	2	2	1	56	2	0	1
27	0	3	2	57	0	1	2
28	1	0	3	58	1	2	3
29	2	1	4	59	2	3	4

Таблиця 3.6

Відповідність числових даних U їх штучним формам U'

U	U'	U	U'	U	U'	U	U'	U	U'
-30	0	-18	12	-6	24	6	36	18	48
-29	1	-17	13	-5	25	7	37	19	49
-28	2	-16	14	-4	26	8	38	20	50
-27	3	-15	15	-3	27	9	39	21	51
-26	4	-14	16	-2	28	10	40	22	52
-25	5	-13	17	-1	29	11	41	23	53
-24	6	-12	18	0	30	12	42	24	54
-23	7	-11	19	1	31	13	43	25	55
-22	8	-10	20	2	32	14	44	26	56
-21	9	-9	21	3	33	15	45	27	57
-20	10	-8	22	4	34	16	46	28	58
-19	11	-7	23	5	35	17	47	29	59

Приклад 3.4. Нехай $U=-3$, $\mu=3$. Якщо $U=-3<0$, тоді в ШФ отримаємо, що $U'=\frac{M}{2}-|U|=\frac{60}{2}-3=30-3=27$. У СЗК (виходячи з даних таблиці 3.5) отримаємо, що $U'_{СЗК_{27}}=(0,3,2)$.

У результаті множення значення $U'_{СЗК}=(0,3,2)$ саме на себе $U'_{СЗК}\times U'_{СЗК}=(U'_{СЗК})^2=(0,3,2)\times(0,3,2)$ тобто, $0\cdot 0=0(\mathrm{mod}3)$, $3\cdot 3=1(\mathrm{mod}4)$, і $2\cdot 2=4(\mathrm{mod}5)$ в результаті отримаємо, що $(U'_{СЗК})^2=(0,1,4)$.

Так, як $k=3$, то проводимо другу ітерацію операції множення $(U'_{СЗК})^2\times U'_{СЗК}=(U'_{СЗК})^3=(0,1,4)\times(0,3,2)$ тобто, $0\cdot 0=0(\mathrm{mod}3)$, $1\cdot 3=3(\mathrm{mod}4)$ і $4\cdot 2=3(\mathrm{mod}5)$ в результаті отримаємо, що $(U'_{СЗК})^3=(0,3,3)$, відповідно до даних таблиці 3.5 визначаємо, що в СЗК $(0,3,3)$ відповідає значенню 3, тобто $(0,3,3)=3$. Відповідно до таблиці 3.6 визначаємо, що $U'=3$ відповідає значенню $U=-27$.

Перевірка: $(U'_{СЗК})^3=27^3=27\times 27\times 27=19683=3(\mathrm{mod}\,60)=$

$=(0,3,2)\times(0,3,2)\times(0,3,2)=(0,3,3)=3$.

$$(U^3_{СЗК})' = \frac{M}{2} + U^3_{СЗК}, \qquad U^3_{СЗК} = (U^3_{СЗК})' - \frac{M}{2}, \qquad (-3)^3 = 3 - \frac{60}{2} = 3 - 30 = -27.$$

Таким чином, маємо, що $(-3)^3 = -27$. Результат операції є достовірним.

Приклад 3.5. Нехай $U = 3$, $\mu = 3$. Якщо $U = 3 > 0$, тоді в ШФ отримаємо, що $U' = \frac{M}{2} + U = \frac{60}{2} + 3 = 30 + 3 = 33$. У СЗК (виходячи з даних таблиці 3.5) отримаємо, що $U'_{СЗК_{33}} = (0, 1, 3)$.

Так, як $\mu = 3$, то $(U'_{СЗК})^3 = U'_{СЗК} \times U'_{СЗК} \times U'_{СЗК} = (0, 1, 3) \times (0, 1, 3) \times (0, 1, 3)$ тобто, $0 \cdot 0 \cdot 0 = 0(\mathrm{mod}3)$, $1 \cdot 1 \cdot 1 = 1(\mathrm{mod}4)$ і $3 \cdot 3 \cdot 3 = 2(\mathrm{mod}5)$ в результаті отримаємо, що $(U'_{СЗК})^3 = (0, 1, 2)$, відповідно до даних таблиці 3.5 визначаємо, що в СЗК $(0, 1, 2)$ відповідає значенню 57, тобто $(0, 1, 2) = 57$. Відповідно до таблиці 3.6 визначаємо, що $U' = 57$ відповідає значенню $U = 27$.

Перевірка: $(U'_{СЗК})^3 = 33^3 = 33 \times 33 \times 33 = 35937 = 57(\mathrm{mod}60) =$

$= (0, 1, 3) \times (0, 1, 3) \times (0, 1, 3) = (0, 1, 2) = 57$.

$$(U^3_{СЗК})' = \frac{M}{2} + U^3_{СЗК}, \qquad U^3_{СЗК} = (U^3_{СЗК})' - \frac{M}{2}, \qquad (3)^3 = 57 - \frac{60}{2} = 57 - 30 = 27.$$

Таким чином, маємо, що $(3)^3 = 27$. Результат операції є достовірним.

Приклад 3.6. Нехай $U = -2$, $\mu = 3$. Якщо $U = -2 < 0$, тоді в ШФ отримаємо, що $U' = \frac{M}{2} - |U| = \frac{60}{2} - 2 = 30 - 2 = 28$. У СЗК (виходячи з даних таблиці 3.5) отримаємо, що $U'_{СЗК_{28}} = (1, 0, 3)$.

Так, як $\mu = 3$, то $(U'_{СЗК})^3 = U'_{СЗК} \times U'_{СЗК} \times U'_{СЗК} = (1, 0, 3) \times (1, 0, 3) \times (1, 0, 3)$ тобто, $1 \cdot 1 \cdot 1 = 1(\mathrm{mod}3)$, $0 \cdot 0 \cdot 0 = 0(\mathrm{mod}4)$ і $3 \cdot 3 \cdot 3 = 2(\mathrm{mod}5)$ в результаті отримаємо, що $(U'_{СЗК})^3 = (1, 0, 2)$, відповідно до даних таблиці 3.5 визначаємо, що в СЗК $(1, 0, 2)$ відповідає значенню 52, тобто $(1, 0, 2) = 52$.

Перевірка: $(U'_{СЗК})^3 = 28^3 = 28 \times 28 \times 28 = 21952 = 52(\mathrm{mod}60) =$

$= (1, 0, 3) \times (1, 0, 3) \times (1, 0, 3) = (1, 0, 2) = 52$.

$$(U^3_{СЗК})' = \frac{M}{2} + U^3_{СЗК}, \qquad U^3_{СЗК} = (U^3_{СЗК})' - \frac{M}{2}, \qquad (3)^3 = 52 - \frac{60}{2} = 52 - 30 = 22.$$

Відповідно до таблиці 3.6 визначаємо, що $U' = 22$ відповідає значенню $U = -8$.

Таким чином, маємо, що $(-2)^3 = -8$. Результат операції є достовірним.

Комп'ютерна модель процесу піднесення цілих чисел до довільного степеня натурального числа у системі залишкових класів написана на мові програмування C# в середовищі Microsoft Visual Studio 2015 (рис. 3.3). Діаграма діяльності цієї комп'ютерної моделі наведено на рис. 3.4.

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Diagnostics;
using System.Threading;
using System.Threading.Tasks;

namespace ConsoleApplication1
{
    class Program
    {
        static void Main(string[] args)
        {
            int[] a = { 3, 4, 5 };
            int L = a.Length;
            int M = 1;

            for (int i = 0; i < L; i++)
            {
                M = M * a[i];
            }

            int[] PSS_Vektor = new int[M];

            int[,] SOK_Vektor = new int[M, 3];
            Console.WriteLine("Представлення чисел в системі
залишкових класів");
            for (int i = 0; i < M; i++)
            {
                PSS_Vektor[i] = (M / 2) - M + i;
                Console.Write(PSS_Vektor[i]);
                Console.Write(" ");
```

Рис 3.3 Лістинг коду комп'ютерної моделі процесу піднесення цілих чисел до довільного степеня натурального числа в СЗК (початок)

```
                Console.Write("[");
                for (int j = 0; j < L; j++)
                {
                    SOK_Vektor[i, j] = i % a[j];
                    Console.Write(SOK_Vektor[i,j]);
                    Console.Write(" ");
                }
                Console.Write("]");
                Console.WriteLine();
            }

            int D = -3;
            int k = 3;
            int[] D_SOK = new int[L];
            int[] D_SOK_InK = new int[L];
            Console.WriteLine("Приклад № 1");
            Console.Write("Число ");
            Console.Write(D);
            Console.Write(" в штучній формі дорівнює ");
            Console.Write((M / 2) + D);
            Console.Write(" що в СЗК дорівнює ");
            Console.Write("[");
            for (int i = 0; i < L; i++)
            {
                D_SOK[i] = ((M / 2) + D) % a[i];
                D_SOK_InK[i] = D_SOK[i];
                Console.Write(D_SOK[i]);
                Console.Write(" ");
            }
            Console.Write("]");
            Console.WriteLine();

            Parallel.For(0, L, j =>
            {
                for (int i = 1; i < k; i++)
                {
                    D_SOK_InK[j] = (D_SOK_InK[j] * D_SOK[j]) %
a[j];
                }

            });

            Console.Write("Результат в СЗК [");
            for (int j = 0; j < L; j++)
            {
                Console.Write(D_SOK_InK[j]);
                Console.Write(" ");
            }
```

Рис 3.3 Лістинг коду комп’ютерної моделі процесу піднесення цілих чисел до довільного степеня натурального числа в СЗК (продовження)

```
            Console.WriteLine("]");

            D = 3;
            k = 3;

            Console.WriteLine("Приклад № 2");
            Console.Write("Число ");
            Console.Write(D);
            Console.Write(" в штучній формі дорівнює ");
            Console.Write((M / 2) + D);
            Console.Write(" що в СЗК дорівнює ");
            Console.Write("[");
            for (int i = 0; i < L; i++)
            {
                D_SOK[i] = ((M / 2) + D) % a[i];
                D_SOK_InK[i] = D_SOK[i];
                Console.Write(D_SOK[i]);
                Console.Write(" ");
            }
            Console.Write("]");
            Console.WriteLine();

            Parallel.For(0, L, j =>
            {
                for (int i = 1; i < k; i++)
                {
                    D_SOK_InK[j] = (D_SOK_InK[j] * D_SOK[j]) %
a[j];
                }

            });

            Console.Write("Результат в СЗК [");
            for (int j = 0; j < L; j++)
            {
                Console.Write(D_SOK_InK[j]);
                Console.Write(" ");
            }
            Console.WriteLine("]");
            D = -2;
            k = 3;

            Console.WriteLine("Приклад № 3");
            Console.Write("Число ");
            Console.Write(D);
            Console.Write(" в штучній формі дорівнює ");
            Console.Write((M / 2) + D);
            Console.Write(" що в СЗК дорівнює ");
            Console.Write("[");
```

Рис 3.3 Лістинг коду комп’ютерної моделі процесу піднесення цілих чисел до довільного степеня натурального числа в СЗК (продовження)

```
            for (int i = 0; i < L; i++)
            {
                D_SOK[i] = ((M / 2) + D) % a[i];
               D_SOK_InK[i] = D_SOK[i];
               Console.Write(D_SOK[i]);
                Console.Write(" ");
            }

            Console.Write("]");
            Console.WriteLine();

            Parallel.For(0, L, j =>
            {
                for (int i = 1; i < k; i++)
                {
                    D_SOK_InK[j] = (D_SOK_InK[j] * D_SOK[j]) %
a[j];
                }

            });

            Console.Write("Результат в СЗК [");
            for (int j = 0; j < L; j++)
            {
                Console.Write(D_SOK_InK[j]);
                Console.Write(" ");
            }
            Console.Write("]");

            Console.ReadKey();

        }
    }
}
```

Рис 3.3 Лістинг коду комп’ютерної моделі процесу піднесення цілих чисел до довільного степеня натурального числа в СЗК (закінчення)

Результати комп’ютерного моделювання в середовищі Microsoft Visual Studio 2015 підтверджують практичну реалізованість запропонованого методу (рис. 3.4).

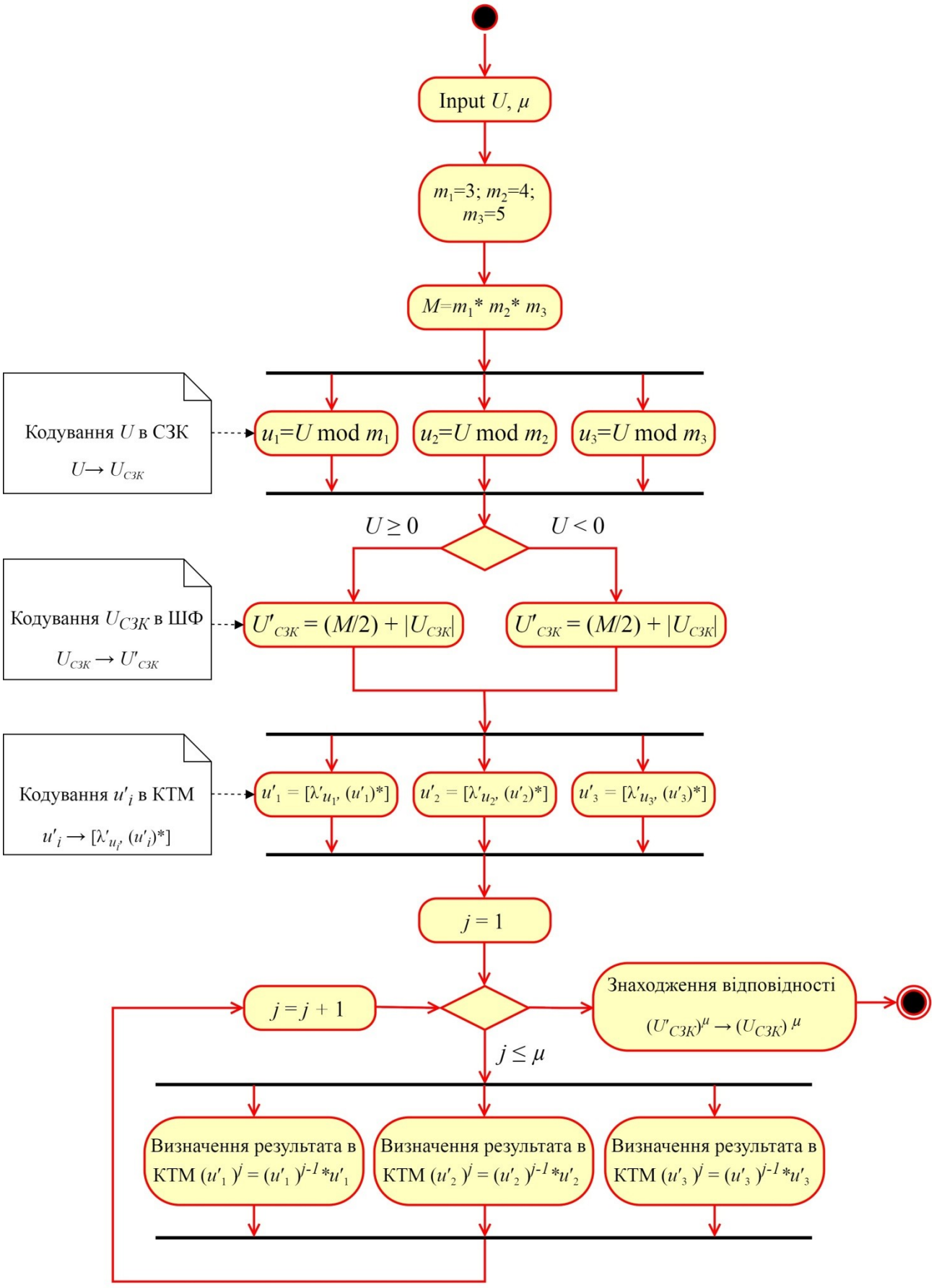

Рис. 3.5 Діаграма діяльності комп’ютерної модель процесу піднесення цілих чисел до довільного степеня натурального числа у СЗК

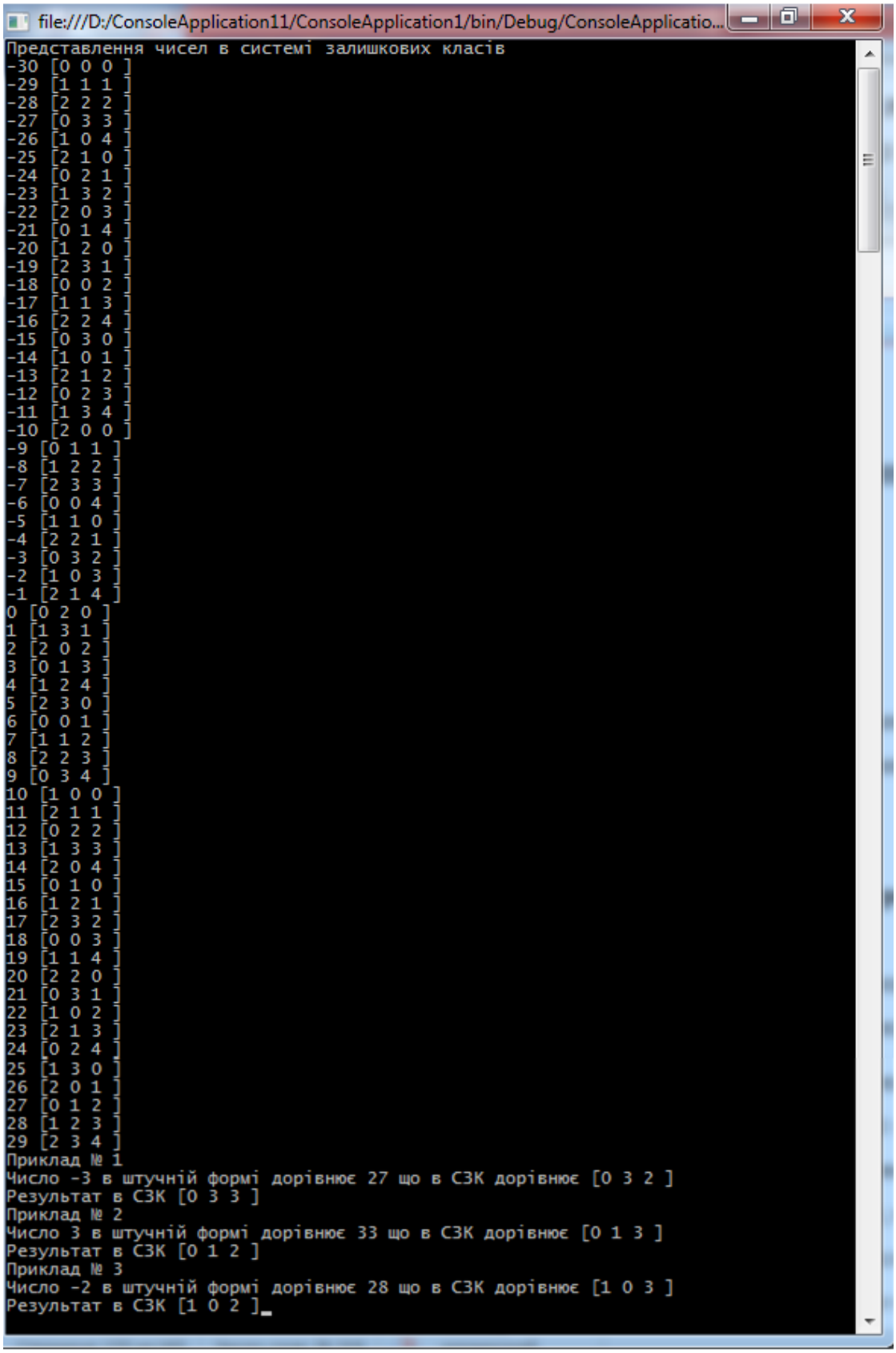

Рис. 3.5 Результати моделювання комп’ютерної моделі процесу піднесення цілих чисел до довільного степеня натурального числа в СЗК в середовищі Microsoft Visual Studio 2015

Висновки до розділу 3

У третьому розділі вдосконалено метод табличної реалізації множення двох залишків чисел в системі залишкових класів за рахунок можливості виконання операції в комплексній області, на основі використання першої фундаментальної теореми Гауса про ізоморфізм між множиною дійсних і комплексних чисел, що підвищує швидкодію реалізації операції множення в системі залишкових класів.

Вдосконалено математичну модель методу піднесення цілих чисел до довільного степеня натурального числа в СЗК за рахунок можливості виконання операції піднесення цілих чисел до степеня, як у додатному, так і в від'ємному числових діапазонах, що підвищує швидкодію реалізації операції піднесення цілих чисел до степеня в системі залишкових класів.

Результати комп'ютерного моделювання середовищі Microsoft Visual Studio 2015 підтверджують практичну реалізованість запропонованого методу.

РОЗДІЛ 4. РОЗРОБКА ПРИСТРОЇВ ВИКОНАННЯ АРИФМЕТИЧНИХ ОПЕРАЦІЙ, ПРОГРАМНО-АПАРАТНИХ СИСТЕМ І КОМПЛЕКСІВ З ЕЛЕМЕНТАМИ ШТУЧНОГО ІНТЕЛЕКТУ, ЩО ФУНКЦІОНУЮТЬ В СИСТЕМІ ЗАЛИШКОВИХ КЛАСІВ

4.1. Розробка операційного пристрою в системі залишкових класів

В основу винаходу операційного пристрою програмно-апаратних систем і комплексів з елементами штучного інтелекту, що функціонують в системі залишкових класів, поставлено мету: розширити функціональні можливості наявного вже операційного пристрою. Розширення можливостей операційного пристрою досягається завдяки тому, що, крім виконання операції додавання залишків чисел x_i і y_i по модулю m_i СЗК, пристрій ще додатково виконує операцію модульного віднімання $(x_i - y_i) \bmod m_i$ в СЗК.

На рис.4.1 представлена блок-схема цього операційного пристрою в СЗК для довільного модуля m_i [69], де:

1, 2 - входи пристрою;

3, 4 - вхідні регістри;

5 - k- розрядний двійковий суматор (k=[$\log_2(m_i-1)$]+1);

6_1-6_k – група k ДОС;

7 – вихідний регістр;

8 - вихід пристрою;

9, 12, 13 – перша, друга та третя шина керування пристроєм;

10, 11 - перша та друга групи елементів І;

14 – інвертор значення y_i по модулю m_i, для визначення значення $\overline{y}_i = (m_i - y_i)$;

15 - група елементів АБО.

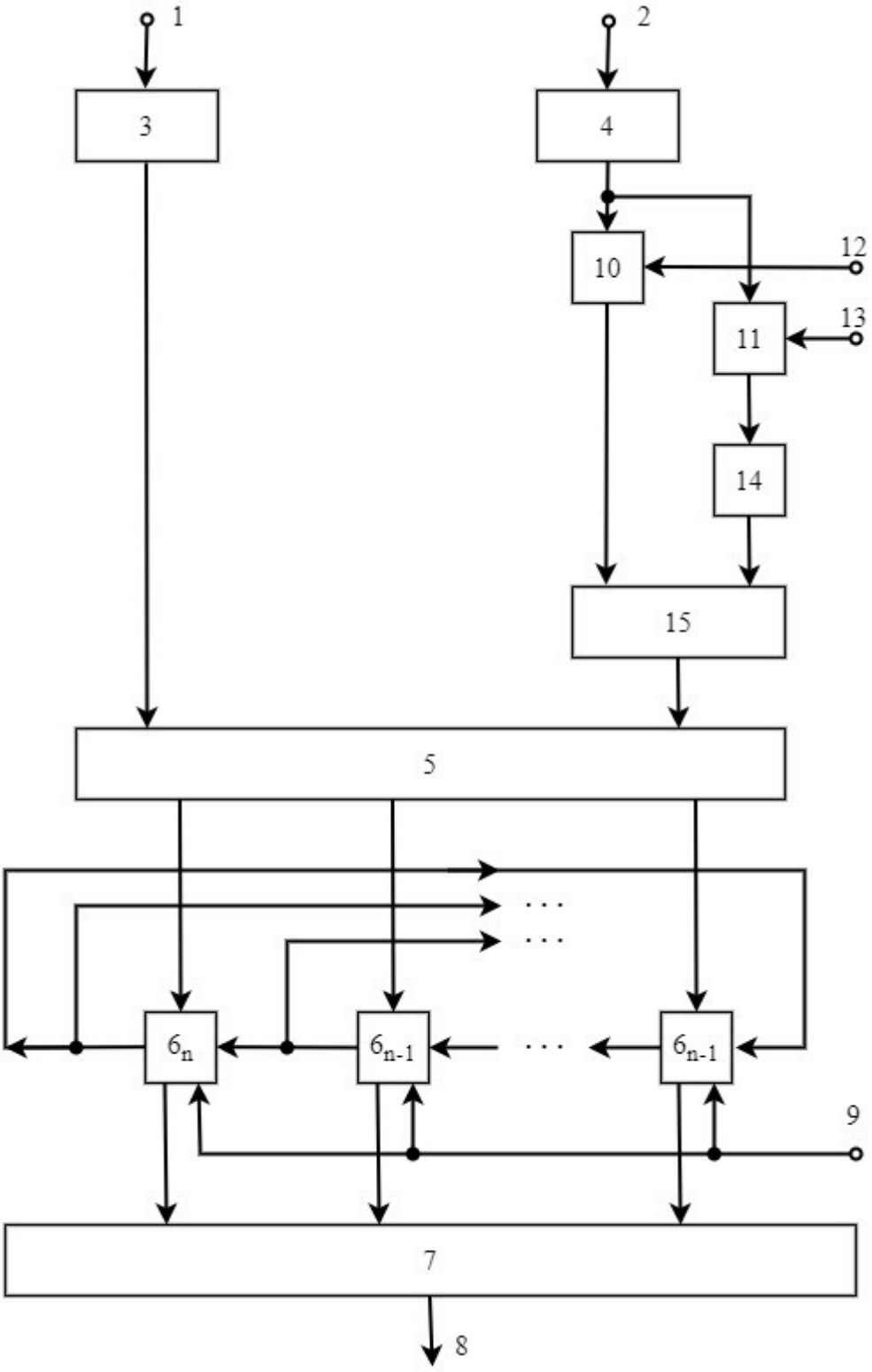

Рис. 4.1 Блок-схема винаходу для довільного модуля m_i СЗК

Пристрій функціонує у двох режимах роботи. В першому режимі знаходиться результат операції додавання залишків чисел $(x_i + y_i)\text{mod } m_i$ по

модулю m_i СЗК. А у другому режимі знаходиться результат операції віднімання $(x_i - y_i) \bmod m_i$ по модулю m_i СЗК.

Розглянемо спочатку роботу операційного пристрою у першому режимі. В цьому режимі присутній сигнал на шині 12. По першому та другому входу операційного пристрою надходять залишки чисел у двійковому коді x_i і y_i до першого (3) та другого (4) вхідних регістрів відповідно. З виходу другого регістра (4) значення залишку y_i надходить до другого входу k- розрядного двійкового суматору (5), через відкриті елементи I першої групи (10), та через елементи групи АБО (15). Далі значення порозрядної суми x_i+y_i надходить до входів групи k ДОС (6_1-6_k). Після чого, визначається результат операції модульного додавання залишків чисел $(x_i + y_i) \bmod m_i$, у відповідності до конкретної схеми для модуля m_i СЗК. Та результат $(x_i + y_i) \bmod m_i$, у двійковому коді надходить до вихідного регістра 7.

Далі розглянемо роботу операційного пристрою у другому режимі. В цьому режимі присутній сигнал на шині 13. По першому та другому входу операційного пристрою надходять залишки чисел у двійковому коді x_i і y_i до першого (3) та другого (4) вхідних регістрів відповідно. З виходу другого регістра (4) значення залишку y_i надходить до інвертор по модулю m_i (14). Після цього отримане значення з інвертор по модулю m_i (14) надходить на k-розрядний двійковий суматор (k=[$\log_2(m_i-1)$]+1) (5). Звідки значення $x_i+m_i -y_i$ надходить до входів групи k ДОС (6_1-6_k). Після чого, визначається результат $(x_i+m_i -y_i) \bmod m_i$, що і буде результатом операції модульного віднімання $(x_i - y_i) \bmod m_i$, який у двійковому коді надходить до вихідного регістра 7.

Розглянемо приклади визначення результату операції модульного додавання (у першому режимі) та віднімання (у другому режимі) для пристрою, що функціонує по модулю m_i =11 СЗК. На рис. 4.2 представлена блок-схема цього операційного пристрою [69].

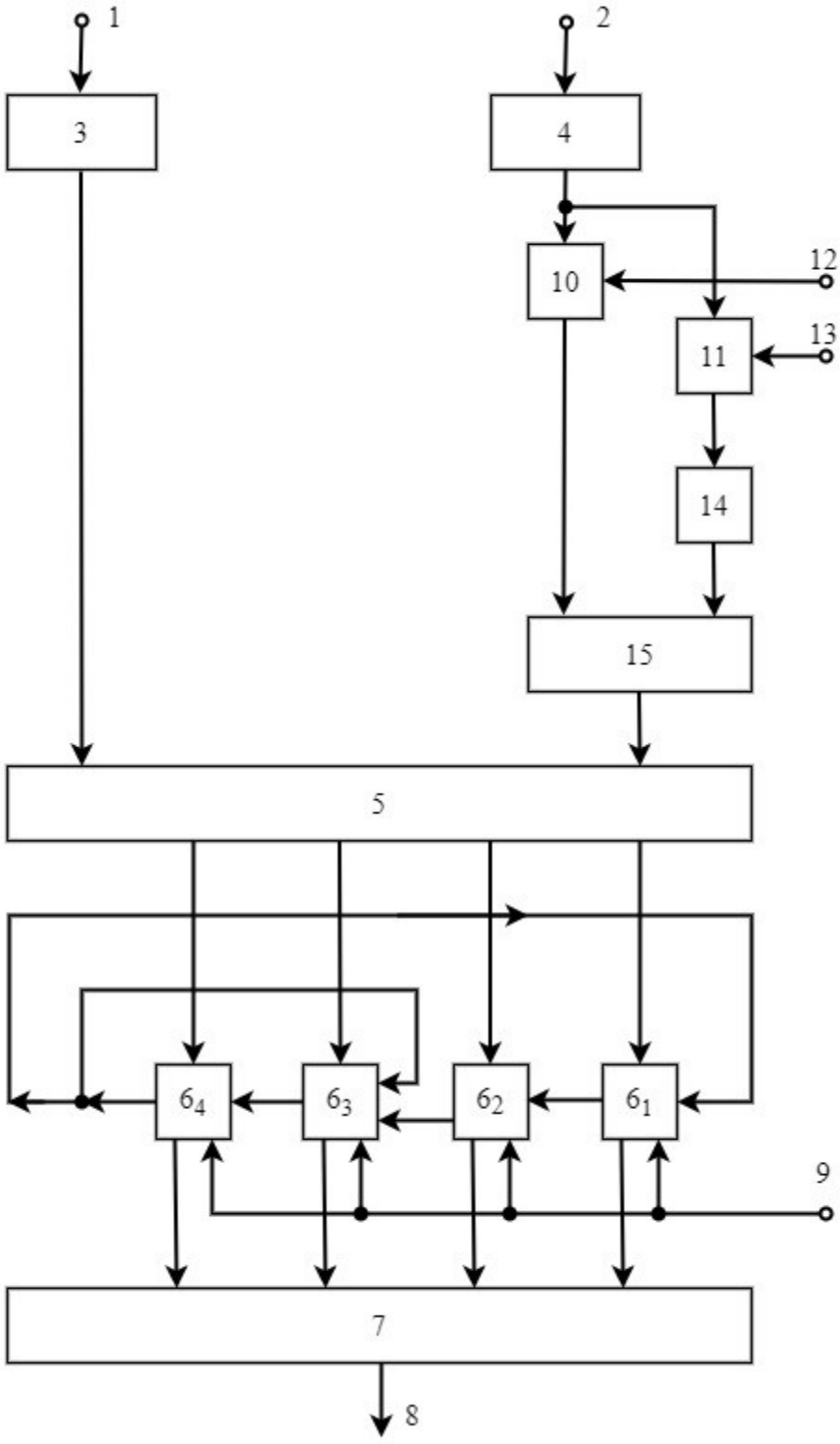

Рис. 4.2 Блок-схема винаходу для довільного модуля m_i=11 СЗК

Розглянемо приклади функціонування пристрою у перший режим роботи пристрою, тобто для визначення результату операції $(x_i + y_i)\text{mod } m_i$.

Приклад 1. Нехай x_i =1001 і y_i =0111.

Перший та другий залишки чисел x_i=1001 і y_i=0111, по входах 1 та 2 у двійковому коді надходять до першого (3) і другого (4) вхідних регістрів. З виходу другого регістра (4) значення залишку y_i надходить до другого входу *k*-розрядного двійкового суматору (5), через відкриті елементи І першої групи (10), та через елементи групи АБО (15). Далі значення порозрядної суми x_i+y_i надходить до входів групи *k* ДОС (6_1-6_k).

$$\begin{array}{rl} & x_i = 1110 \\ \oplus & y_i = 0111 \\ \hline & 1110 \end{array}$$

Після чого, визначається результат операції модульного додавання залишків чисел $(x_i + y_i)$mod m_i, у відповідності до схеми для модуля m_i=11 СЗК рис. 4.3.

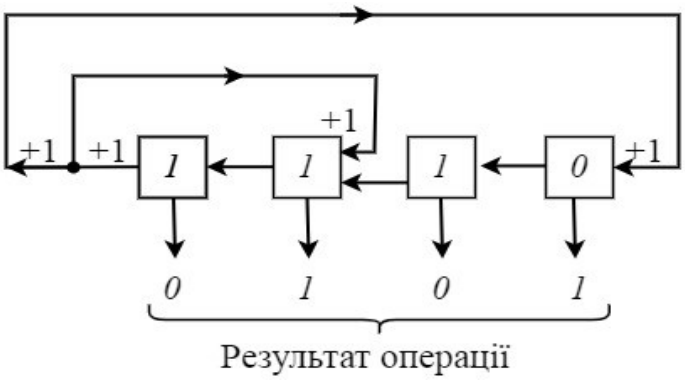

Рис. 4.3 Схемою додавання лишків x_i =1001 і y_i =0111 за модулем m_i =11

У таблиці 4.1 представлено алгоритм, за яким визначається результат операції модульного додавання для залишків x_i =1001 і y_i =0111 по модулю m_i =11. Та результат $(x_i + y_i)$mod m_i, у двійковому коді надходить до вихідного регістра 7.

Перевірка: (9+7) mod 11= 5.

Таблиця 4.1

Алгоритм визначення результату операції додавання

ДОС	Входи ДОС 6	Виходи ДОС 6
6_1	0+1	1
6_2	1+1	0
6_3	1+1+1	1
6_4	1+1	0

Приклад 2. Нехай x_i =1010 і y_i =0101.

Перший та другий залишки чисел x_i=1010 і y_i=0101, по входах 1 та 2 у двійковому коді надходять до першого (3) і другого (4) вхідних регістрів. З виходу другого регістра (4) значення залишку y_i надходить до другого входу *k*-розрядного двійкового суматору (5), через відкриті елементи I першої групи (10), та через елементи групи АБО (15). Далі значення порозрядної суми x_i+y_i надходить до входів групи *k* ДОС (6_1-6_k).

$$\begin{array}{rl} & x_i = 1010 \\ \oplus & y_i = 0101 \\ \hline & 1111 \end{array}$$

Після чого, визначається результат операції модульного додавання залишків чисел $(x_i + y_i)\bmod m_i$, у відповідності до схеми для модуля m_i=11 СЗК рис. 4.4.

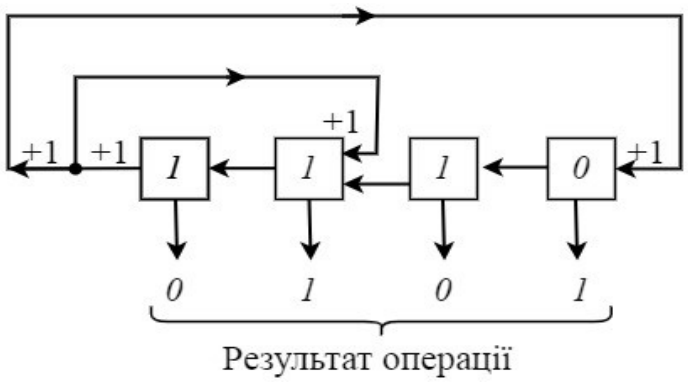

Рис. 4.4 Схема додавання лишків x_i=1010 і y_i=0101 за модулем m_i=11

У таблиці 4.2 представлено алгоритм, за яким визначається результат операції модульного додавання для залишків x_i =1010 і y_i =0101 по модулю m_i =11. Та результат $(x_i + y_i)$mod m_i, у двійковому коді надходить до вихідного регістра 7.

Таблиця 4.2

Алгоритм визначення результату операції додавання

ДОС	Входи ДОС 6	Виходи ДОС 6
6_1	1+1	0
6_2	1+1	0
6_3	1+1+1	1
6_4	1+1	0

Перевірка: (10+5) mod 11=4.

Приклад 3. Нехай x_i=0010 і y_i=0101.

Перший та другий залишки чисел x_i=0010 і y_i=0101, по входах 1 та 2 у двійковому коді надходять до першого (3) і другого (4) вхідних регістрів. З виходу другого регістра (4) значення залишку y_i надходить до другого входу *k*-розрядного двійкового суматору (5), через відкриті елементи I першої групи (10), та через елементи групи АБО (15). Далі значення порозрядної суми x_i+y_i надходить до входів групи *k* ДОС (6_1-6_k).

$$\begin{array}{rl} & x_i = 0010 \\ \oplus & \\ & y_i = 0101 \\ \hline & 0111 \end{array}$$

Після чого, визначається результат операції модульного додавання залишків чисел $(x_i + y_i)$mod m_i, у відповідності до схеми для модуля m_i=11 СЗК рис. 4.5.

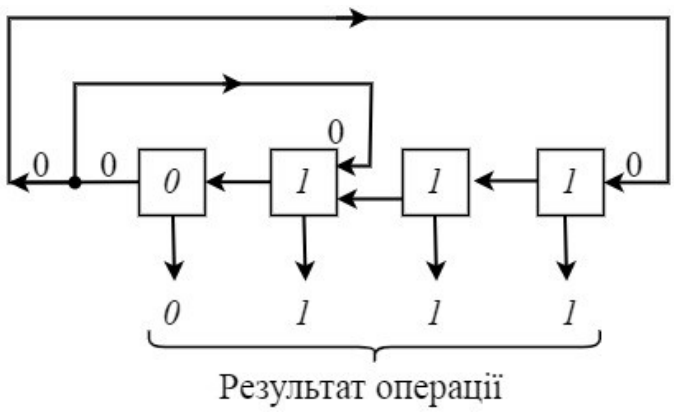

Рис. 4.5 Схема додавання лишків x_i=0010 і y_i=0101 за модулем m_i=11

У таблиці 4.3 представлено алгоритм, за яким визначається результат операції модульного додавання для залишків x_i =0010 і y_i =0101 по модулю m_i =11. Та результат $(x_i + y_i)$mod m_i, у двійковому коді надходить до вихідного регістра 7.

Таблиця 4.3

Алгоритм визначення результату операції додавання

ДОС	Входи ДОС 6	Виходи ДОС 6
6_1	0+1	1
6_2	0+1	1
6_3	0+1	1
6_4	0+0	0

Перевірка: (2+5) mod 11=7.

Розглянемо приклади роботи пристрою у другому режимі, для обчислення результату операції модульного віднімання. В цьому режимі присутній сигнал на шині 13, тобто відкриті елементи I другої групи (11).

Приклад 4. Нехай x_i=0010 і y_i=1010.

Визначемо значення $(x_i - y_i)$ mod m_i.

По першому та другому входу операційного пристрою надходять залишки чисел у двійковому коді x_i =0010 і y_i=1010 до першого (3) та другого (4) вхідних регістрів відповідно. З виходу другого регістра (4) значення

залишку y_i надходить до інвертор по модулю m_i (14). Після цього отримане значення (1011-1010=0001) з інвертора по модулю m_i (14) надходить на k-розрядний двійковий суматор (k=[$\log_2$(m_i-1)]+1) (5).

$$\begin{array}{r} x_i = 0010 \\ \oplus \\ y_i = 0001 \\ \hline 0011 \end{array}$$

Звідки значення $x_i+m_i-y_i$ надходить до входів групи k ДОС (6_1-6_k). Після чого, у відповідності до схеми модульного додавання за модулем $m_i=17$ (рис 4.6) визначається результат $(x_i+m_i-y_i)\bmod m_i$ що представлено у таблиці 4.4

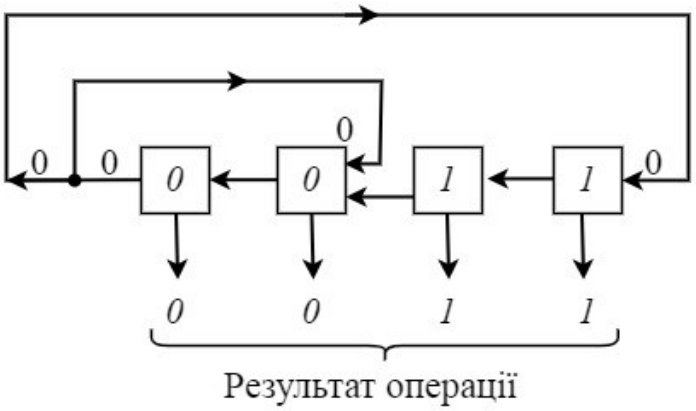

Рис. 4.6 Схема додавання лишків x_i=0010 і y_i=0101 за модулем m_i=11

Таблиця 4.4

Алгоритм визначення результату операції додавання

ДОС	Входи ДОС 6	Виходи ДОС 6
6_1	0+1	1
6_2	0+1	1
6_3	0+0	0
6_4	0+0	0

Отримане значення 0011_2 і буде результатом операції модульного віднімання $(x_i - y_i)\bmod m_i$. Далі воно у двійковому коді надходить до вихідного регістра 7.

Перевірка: (2-10) mod 11=3.

Приклад 5. Нехай x_i=0010 і y_i=0110.

Визначемо значення $(x_i - y_i) \bmod m_i$.

По першому та другому входу операційного пристрою надходять залишки чисел у двійковому коді x_i =0010 і y_i=0110 до першого (3) та другого (4) вхідних регістрів відповідно. З виходу другого регістра (4) значення залишку y_i надходить до інвертора по модулю m_i (14). Після цього отримане значення (1011-0110=0101) з інвертора по модулю m_i (14) надходить на k-розрядний двійковий суматор (k=[$\log_2$(m_i-1)]+1) (5).

$$\begin{array}{rl} & x_i = 0010 \\ \oplus & \\ & y_i = 0101 \\ \hline & 0111 \end{array}$$

Звідки значення $x_i+m_i-y_i$ надходить до входів групи k ДОС (6_1-6_k). Після чого, у відповідності до схеми модульного додавання за модулем $m_i=17$ визначається результат $(x_i+m_i-y_i)\bmod m_i$ що співпадає з алгоритмом наведеним у прикладі 3 (таблиця 4.3).

Перевірка: (2-6) mod 11=7.

Таким чином, застосування запропонованого винаходу (операційного пристрою у СЗК) суттєво дозволяє розширити функціональні можливості вже наявного пристрою-прототипу. Розширення можливостей операційного пристрою досягається завдяки тому, що, крім виконання операції додавання залишків чисел x_i і y_i по модулю m_i СЗК, пристрій ще додатково виконує операцію модульного віднімання $(x_i - y_i) \bmod m_i$ в СЗК.

Наведено приклади виконання операцій додавання $(x_i + y_i) \bmod m_i$ і віднімання $(x_i - y_i) \bmod m_i$ залишків чисел по модулю СЗК, що підтверджує практичну можливість використання запропонованого винаходу операційного пристрою програмно-апаратних систем і комплексів з елементами штучного інтелекту в СЗК [68-72].

4.2. Розрахунок та порівняльний аналіз швидкодії програмно-апаратних систем і комплексів з елементами штучного інтелекту в СЗК

Для порівняльного аналізу швидкодії програмно-апаратних систем і комплексів з елементами штучного інтелекту в ПСС та СЗК розглянемо математичну модель штучного нейрону.

Штучний нейрон складається з набору односпрямованих вхідних зв'язків - синапсів, які з'єднані з виходами інших нейронів, та має аксон - це вихідний зв'язок даного нейрона, який передає сигнал (збудження або гальмування) на синапси наступних нейронів. Кожен вхід має свою вагу, яка відображає силу синаптичного зв'язку, і сума всіх вхідних сигналів визначає рівень активації нейрона.

Математична модель нейрону має вигляд:

$$S = \sum_{i=1}^{N} x_i \cdot w_i \,. \tag{4.1}$$

Через входи, кількість яких дорівнює N , математичний нейрон приймає вхідні сигнали x_i, які він підсумовує , помножуючи кожен вхідний сигнал на деякий ваговий коефіцієнт w_i.

Передумовою створення програмно-апаратних систем і комплексів з елементами штучного інтелекту на основі апарату СЗК є схожість математичної моделі штучного нейрону і СЗК.

Математична модель представлення числа в СЗК і штучним нейроном

$$S = \sum_{i=1}^{N} \alpha_i \cdot \beta_i (\mathrm{mod}\, p_i) \tag{4.2}$$

Для проведення порівняльного аналізу програмно-апаратних систем і комплексів з елементами штучного інтелекту використовують співвідношення:

$$K_{eff}^{(r)} = \frac{T^{(ПСЧ)}}{T^{(СЗК)}} \tag{4.3}$$

де:

$T^{(ПСЧ)}$ час обчислення виразу (4.1) в програмно-апаратних системах і комплексах з елементами штучного інтелекту в ПСЧ;

$T^{(СЗК)}$ час обчислення виразу (4.2) в програмно-апаратних системах і комплексах з елементами штучного інтелекту в СЗК.

Основною операцією в (4.1) є знаходження суми:

$$S = x_1 \cdot w_1 + x_2 \cdot w_2 + ... + x_i \cdot w_i + ... + x_n \cdot w_n . \tag{4.4}$$

В ПСЧ час виконання операції (4.4) складає:

$$T^{(ПСЧ)} = N \cdot t_* + (N-1) \cdot t_+ . \tag{4.5}$$

Відомо, що час реалізації операції додавання t_+ і операції множення t_* в ПСЧ визначається наступними виразами:

$$\begin{aligned} t_+ &= \tau(2\rho - 1) \\ t_* &= 2\tau\rho^2 \end{aligned} \tag{4.6}$$

де:

ρ - кількість двійкових розрядів в представлені операндів;

τ - час спрацювання логічних елементів І (АБО).

З урахуванням вищенаведених позначень перепишемо формулу (4.5) у вигляді:

$$T^{(ПСЧ)} = 2N\tau\rho^2 + (N-1)\cdot\tau\cdot(2\rho-1). \quad (4.7)$$

В СЗК час реалізації арифметичних операцій, суматор ним принципом, визначається часом реалізації даної модульної операції для максимального за величиною модуля m_n СЗК:

$$\begin{aligned} t_+ &= \tau(2k_n - 1), \\ t_* &= 2\tau k_n^2. \end{aligned} \quad (4.8)$$

де $k_n = \left[\log_2(m_i - 1) + 1\right]$.

Тоді час виконання операції (4.4) в СЗК складає:

$$T^{(СЗК)} = 2N\tau\left[\log_2(m_i - 1) + 1\right]^2 + (N-1)\cdot(2\left[\log_2(m_i - 1) + 1\right] - 1)\cdot\tau. \quad (4.9)$$

З урахуванням виразів (4.7) та (4.9) $K_{eff}^{(r)}$ дорівнює:

$$K_{eff}^{(r)} = \frac{2N\rho^2 + (N-1)\cdot(2\rho-1)}{2N\left[\log_2(m_i - 1) + 1\right]^2 + (N-1)\cdot(2\left[\log_2(m_i - 1) + 1\right] - 1)}. \quad (4.10)$$

Порівняльний аналіз ефективності для одно- (r=1), двох- (r=2), трьох- (r=3), чотирьох (r=4) та восьмибайтової (r=8) розрядної сітки програмно-апаратних систем і комплексів з елементами штучного інтелекту при N=10, 50, 100 наведено в таблиці 4.5.

Порівняльний аналіз результатів розрахунків показав ефективність використання СЗК для математичної моделі штучного нейрону програмно-апаратних систем і комплексів з елементами штучного інтелекту.

Таблиця 4.5

Розрахункові дані ефективності для математичної моделі нейрону програмно-апаратних систем і комплексів з елементами штучного інтелекту в ПСЧ та СЗК.

Величина *r* розрядної сітки програмно-апаратних систем і комплексів з елементами штучного інтелекту	Совокупність модулів СЗК m_i	Максимальний модуль СЗК m_n	Розрядність максимального модуля СЗК k_n	*N*	*t*, [τ]		$K_{eff}^{(r)}$
					ПСЧ	СЗК	
Однобайтова (*r*=1) розрядна сітка (ρ=8)	m_i: (2; 4; 5; 7)	m_4=7	k_4=3	10	1415	225	6,29
				50	7135	1145	6,23
				100	14285	2295	6,22
Двубайтова (*r*=2) розрядна сітка (ρ=16)	m_i: (2; 5; 7; 9; 11; 13)	m_6=13	k_6=13	10	5399	383	14,10
				50	27119	1943	13,96
				100	54269	3893	13,94
Трьохбайтова (*r*=3) розрядна сітка (ρ=24)	m_i: (3; 5; 7; 9; 11; 13; 17; 19)	m_6=19	k_8=19	10	11943	581	20,56
				50	59903	2941	20,37
				100	119853	5891	20,35

Продовження таблиці 4.5

Величина r розрядної сітки програмно-апаратних систем і комплексів з елементами штучного інтелекту	Совокупність модулів СЗК m_i	Максимальний модуль СЗК m_n	Розрядність максимального модуля СЗК k_n	N	t, [τ]		$K_{eff}^{(r)}$
					ПСЧ	СЗК	
Чотирьохбайтова (r=4) розрядна сітка (ρ=32)	m_i: (2; 3; 5; 7; 11; 13; 17; 19; 23; 29)	m_{10}=29	k_{10}=29	10	20867	581	35,92
				50	104507	2941	35,53
				100	209057	5891	35,49
Восьмибайтова (r=8) розрядна сітка (ρ=64)	m_i: (2; 3; 5; 7; 11; 13; 17; 19; 23; 29; 31; 37; 41; 43; 47; 53)	m_{16}=53	k_{16}=53	10	83063	819	101,42
				50	415823	4139	100,46
				100	831773	8289	100,35

Висновки до розділу 4

Четвертий розділ присвячено розробці операційного пристрою програмно-апаратних систем і комплексів з елементами штучного інтелекту, що функціонують в системі залишкових класів та прведенню аналізу швидкодії обробки даних в позиційній системі числення та системі залишкових класів.

Розроблено операційний пристрій програмно-апаратних систем і комплексів з елементами штучного інтелекту, що функціонує в системі залишкових класів.

В основу винаходу операційного пристрою програмно-апаратних систем і комплексів з елементами штучного інтелекту, що функціонують в системі залишкових класів, поставлено мету: розширити функціональні можливості наявного вже операційного пристрою. Розширення можливостей операційного пристрою досягається завдяки тому, що, крім виконання операції додавання залишків чисел x_i і y_i по модулю m_i СЗК, пристрій ще додатково виконує операцію модульного віднімання $(x_i - y_i) \bmod m_i$ в СЗК.

Пристрій функціонує у двох режимах роботи. В першому режимі знаходиться результат операції додавання залишків чисел $(x_i + y_i) \bmod m_i$ по модулю m_i СЗК. А у другому режимі знаходиться результат операції віднімання $(x_i - y_i) \bmod m_i$ по модулю m_i СЗК.

Наведено приклади виконання операцій додавання $(x_i + y_i) \bmod m_i$ і віднімання $(x_i - y_i) \bmod m_i$ залишків чисел по модулю СЗК, що підтверджує практичну можливість використання запропонованого винаходу операційного пристрою програмно-апаратних систем і комплексів з елементами штучного інтелекту в СЗК [68-72].

Проведено розрахунок та порівняльний аналіз швидкодії обробки даних програмно-апаратних систем і комплексів з елементами штучного інтелекту у СЗК для математичної моделі штучного нейрону.

Розрахунки та порівняльна оцінка швидкодії, проведені в дисертаційній роботі, показали, що зі збільшенням розрядності сітки програмно-апаратних

систем і комплексів з елементами штучного інтелекту ефективність застосування непозиційної системи числення в СЗК значно зростає.

ВИСНОВКИ

1. На даний час інтенсивна комп’ютеризація та штучний інтелект є взаємопов'язаними та взаємозалежними концепціями, які впливають на сучасну суспільну та економічну діяльність. Вони відкривають нові можливості, сприяють інноваціям, розвитку та застосуванню передових технологій у багатьох сферах і допомагають вирішувати складні завдання, які раніше було б важко або навіть неможливо вирішити.

Стратегія розвитку штучного інтелекту в Україні передбачає створення програмно-апаратних систем і комплексів з елементами штучного інтелекту. Ці системи включають в себе різноманітні алгоритми та моделі машинного навчання, нейронні мережі, логічні системи, системи обробки природної мови та інші методи та техніки, спрямовані на аналіз даних, роботу з зображеннями, розпізнавання образів, прийняття рішень тощо. Аналіз задач програмно-апаратних систем і комплексів з елементами штучного інтелекту показав, що ці системи оброблюють великі масиви даних в режимі реального часу, тож актуальною є задача підвищення швидкодії програмно-апаратних систем і комплексів з елементами штучного інтелекту.

Проаналізовано сучасний стан та напрями підвищення швидкодії програмно-апаратних систем і комплексів з елементами штучного інтелекту, за рахунок застосування спеціальних технологічних та архітектурних рішень, а також математичних методів для їх застосування в ШІ. Відмічено, що застосування паралельної обробки даних на основі СЗК дозволяє значно підвищити швидкодію операцій обробки даних.

За результатами проведеного аналізу теоретичних основ СЗК визначено основні її переваги над позиційними системами числення (незалежність залишків, що дає можливість розпаралелення процесу обчислень; рівноправність залишків, що дає можливість підвищити відмовостійкість програмно-апаратних систем і комплексів з елементами штучного інтелекту та малорозрядність залишків, що дає можливість підвищити швидкодію

програмно-апаратних систем і комплексів з елементами штучного інтелекту) та її недоліки (труднощі при виконанні операції порівняння та ділення чисел, визначення переповнення допустимого діапазону), обґрунтовано необхідність використання СЗК в операційних пристроях програмно-апаратних систем і комплексів з елементами штучного інтелекту.

2. Найбільш важливими науковими і прикладними результатами, отриманими в роботі, є такі:

- метод табличної реалізації множення двох залишків чисел в системі залишкових класів за рахунок можливості виконання операції в комплексній області, на основі використання першої фундаментальної теореми Гауса про ізоморфізм між множиною дійсних і комплексних чисел, що підвищує швидкодію реалізації операції множення в системі залишкових класів.

- математична модель процесу піднесення цілих чисел до довільного степеня натурального числа в СЗК за рахунок можливості виконання операції піднесення цілих чисел до степеня, як у додатному, так і в від'ємному числових діапазонах, що підвищує швидкодію реалізації операції піднесення цілих чисел до степеня в системі залишкових класів.

- метод додавання і віднімання залишків чисел по модулю СЗК, який враховує конструкції суматорів по модулю з величиною корекції $\Delta Q_R > 0$.

3. Значення отриманих результатів для практики полягає в такому. Розроблено операційний пристрій програмно-апаратних систем і комплексів з елементами штучного інтелекту, що функціонує в системі залишкових класів

В основу винаходу поставлено задачу (мету) розширення функціональних можливостей пристрою. Розширення функціональних можливостей прототипу досягається за рахунок того, що крім виконання арифметичної модульної операції додавання лишків x_i і y_i чисел за модулем m_i СЗК пристрій додатково виконує ще операцію $(x_i - y_i) \bmod m_i$ визначення результату арифметичної модульної операції віднімання лишків x_i і y_i чисел за модулем m_i СЗК.

Наведено приклади виконання операцій модульного додавання і віднімання лишків, що підтверджує практичну можливість використання

запропонованого винаходу операційного пристрою у СЗК. На пристрій отримано патенти України.

4. Під час розв'язання зазначеної наукової задачі використовували такі методи досліджень: системний аналіз, математичне та імітаційне моделювання. Як математична основа для розробки моделей та методів швидкої обробки даних на основі застосування системи залишкових класів, використовуються принципи системного аналізу, теорія чисел (розділи: теорія подільності та теорія порівнянь) і теорія обчислень (під час розроблення моделей та методів швидкої обробки даних на основі застосування системи залишкових класів), теорія обчислювальних процесів і систем, методи імітаційного моделювання (під час оцінювання коректності та достовірності моделей і методів).

5. Обґрунтованість і достовірність наукових положень, висновків і рекомендацій забезпечуються коректним застосуванням загальновизнаного математичного апарату, несуперечливістю з відомими положеннями теорії обчислень, теорії обчислювальних процесів і систем,, основних припущень та обмежень, прийнятих під час моделювання, а також збіжністю результатів, отриманих під час практичного застосування розробленої системи.

Основні результати роботи реалізовано в Харківському національному університеті імені В.Н. Каразіна у рамках НДР «Формулювання та розробка принципів, методів і засобів швидкої та достовірної обробки цілочисельних даних, що представлені у непозиційній системи числення залишкових класів в комп’ютерних системах та мережах подвійного призначення» за 2019-2021 рр. (НДР № 0119U002546).

6. Показано, що для оцінки показників швидкодії розроблених моделей і методів проведено розрахунок та порівняльний аналіз швидкодії обробки даних програмно-апаратних систем і комплексів з елементами штучного інтелекту у СЗК для математичної моделі штучного нейрону.

Розрахунки та порівняльна оцінка швидкодії, проведені в дисертаційній роботі, показали, що зі збільшенням розрядності сітки програмно-апаратних систем і комплексів з елементами штучного інтелекту ефективність

застосування непозиційної системи числення в СЗК значно зростає.

7. Отримані в роботі результати можуть бути рекомендовані до використання в науково-дослідних, проєктних організаціях під час розробки операційних пристроїв програмно-апаратних систем і комплексів з елементами штучного інтелекту, що працюють в СЗК.

8. Сукупність отриманих у дисертації нових наукових результатів, позитивна оцінка їхньої достовірності, наукової та практичної значущості дають змогу вважати сформульовану наукову задачу підвищення швидкості обробки інформації програмно-апаратними системами і комплексами з елементами штучного інтелекту за рахунок використання моделей та методів швидкої обробки даних на основі застосування системи залишкових класів, – розв'язаною, а поставлену мету – досягнутою.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Про схвалення Концепції розвитку штучного інтелекту в Україні на 2021-2024 роки. Кабінет Міністрів України. Розпорядження від 2 грудня 2020 р. № 1556-р Київ. URL: https://zakon.rada.gov.ua/laws/show/1556-2020-%D1%80#Text. Дата звернення: 07.08.2023.

2. Стратегія розвитку штучного інтелекту в Україні: монографія / Шевченко А.І. та ін. Київ: ІПШІ, 2023. 305 с.

3. Ковальчук М. Л., Ушенко Ю. О., Угрин Д. І. Методи та системи штучного інтелекту. Навчальний посібник. Чернівці: Чернівецький національний університет ім. Ю. Федьковича, 2022. 318 с.

4. Удовик І.М., Коротенко Г.М., Коротенко Л.М., Трусов В.О., Харь А.Т. Навчальний посібник «Методи та системи штучного інтелекту» для студентів спеціальності 122 «Комп'ютерні науки». Дніпро: Державний ВНЗ «Національний гірничий університет», 2017. 105 с.

5. Ямпольський Л. С., Ткач Б. П., Лісовиченко О. І. Системи штучного інтелекту в плануванні, моделюванні та управлінні : підручник для студентів вищіх навчальних закладів. Київ. : ДП «Вид. дім «Персонал», 2011. 544 с.

6. Аврунін О. Г., Владов С. І., Петченко М. В., Семенець В. В., Татарінов В. В., Тельнова Г. В., Філатов В. О., Шмельов Ю. М., Шушляпіна Н. О. Інтелектуальні системи автоматизації : монографія. Кременчук : Видавництво «НОВАБУК», 2021. 322 с.

7. Hodson, R.F. Real-Time Expert Systems Computer Architecture. CRC Press, 2017. DOI:https://doi.org/10.1201/9781351076203.

8. Alford, R.S. Computer Systems Engineering Management. CRC Press, 2018. DOI:https://doi.org/10.1201/9781351070829

9. Yadin, A. Computer Systems Architecture (1st ed.). Chapman and Hall/CRC. 2016. DOI:https://doi.org/10.1201/9781315373287.

10. Николайчук Я.М., Возна Н.Я., Пітух І.Р. Проектування спеціалізованих комп'ютерних систем. Тернопіль : ТзОВ «Терно-граф», 2010.

392 с.

11. Harris S. L., Harris D. Digital Design and RISC-V Computer Architecture Textbook. 2021 ACM/IEEE Workshop on Computer Architecture Education (WCAE). Raleigh, NC, USA, 2021. P. 1-5. DOI:doi: 10.1109/WCAE53984.2021.9707615.

12. Тарарака В.Д. Архітектура комп'ютерних систем: навчальний посібник. Житомир : ЖДТУ, 2018. 383 с.

13. Тарарака В.Д. Обчислювальна техніка. Ч.І. Основи побудови ЕОМ: навчальний посібник. Житомир: ЖВІРЕ, 2003. 348 с.

14. Тарарака В.Д. Обчислювальна техніка. Ч. ІІ. Апаратні засоби персональних комп'ютерів: навчальний посібник. Житомир ЖВІРЕ, 2004. 308 с. побудови ЕОМ: навчальний посібник. Житомир: ЖВІРЕ, 2003. 348 с.

15. Демченко К. В., Радченко С. С. Підвищення продуктивності обчислювальних засобів. Електроенергетика, електромеханіка та технології в АПК: матеріали Міжнародна науково-практична конференція, 22 грудня 2022 р., Державний біотехнологічний університет. Харків, 2022. С. 196-197

16. Yanko A., Koshman S., Krasnobayev V. Algorithms of data processing in the residual classes system. 2017 4th International Scientific-Practical Conference Problems of Infocommunications. Science and Technology (PIC S&T). Kharkov, 2017. P. 117-121. DOI:https://doi.org/10.1109/INFOCOMMST.2017.8246363.

17. Кошман С. О. Концепція підвищення продуктивності обробки інформації у реальному часі. Вісник Харківського національного технічного університету сільського господарства імені Петра Василенка. Проблеми енергозабезпечення та енергозбереження в АПК України. 2011. № 117. С. 63-65.

18. Krasnobayev V., Koshman S., Kovalchuk D. The concept of using the number system in the residual classes for building artificial intelligence system. Control, Navigation and Communication Systems. Academic Journal. 2022. Vol. 1(67). P. 65-70. DOI:https://doi.org/10.26906/SUNZ.2022.1.065.

19. Krasnobayev V., Kuznetsov A., Bagmut M. Kuznetsova T. Artificial Intelligence and Number System in Residual Classes. 2021 IEEE 4th International

Conference on Advanced Information and Communication Technologies (AICT). Lviv, Ukraine, 2021. P. 171-176. DOI:https://doi.org/10.1109/AICT52120.2021.9628970.

20. Mohan P. V. A. Residue Number Systems: Theory and Applications. Birkhäuser Basel: Springer International Publishing Switzerland, 2016. 351p.

21. Patterson D. A., Hennessy J. L. Computer Organization and Design: The Hardware Software Interface: ARM Edition. Morgan Kaufmann, 2016. 720p.

22. Patterson D. A., Hennessy J. L. Computer organization and design, fifth edition: the hardware/software interface. Morgan Kaufmann, 2013. 800p.

23. Omondi A., Premkumar B. Residue Number Systems: Theory and Implementation (Advances in Computer Science and Engineering Texts). London : Imperial College Press, 2007. 312p.

24. Krasnobayev V. A., Yanko A. S., Kurchanov V. N., Koshman S. A. The analysis of the tasks and algorithms of data integer processing in the residual classes system. Радіоелектронні і комп'ютерні системи. 2016. № 1 (75). С. 19-28.

25. Загуменна К. В., Радченко С. С., Кучерявий В. М. Особливості системи залишкових класів. Вісник Харківського національного технологічного університету сільського господарства ім. П. Василенка. 2019. С. 89-90.

26. Загуменна К. В., Радченко С. С. Аналіз систем паралельної обробки алгоритмів. Вісник Харківського національного технологічного університету сільського господарства ім. П. Василенка. 2018. С. 76-78.

27. Загуменна К. В., Радченко С. С. Методи реалізації арифметичних операцій у системі залишкових класів. The 8th International scientific and practical conference “Trends, theories and ways of improving science” (February 28 – March 03, 2023) Madrid, Spain. International Science Group. 2023. P. 527-528.

28. Краснобаєв В. А., Кошман С. О. Застосування системи залишкових класів у машинній арифметиці. Вісник Харківського державного технічного університету сільського господарства. Проблеми енергозабезпечення та енергозбереження в АПК України, 19, 2003. Вип. 19. С. 134-136. (Здобувачем досліджено табличний метод реалізації арифметичних операцій для

практичного створення СКЗОІ у СЗК.)

29. Koshman S., Krasnobayev V., Nikolsky S., Kovalchuk D. The structure of the computer system in the residual classes. Advanced Information Systems. 2023. Vol. 7(2). P. 41–48. DOI:https://doi: 10.20998/2522-9052.2023.2.06. (Scopus)

30. Krasnobayev V., Koshman S., Nikolsky S., Kovalchuk D. Mathematical model of computer system reliability in residual classes. Advanced Information Systems. 2022. Vol. 6(4). P. 19–24. DOI:doi: 10.20998/2522-9052.2022.4.03.

31. Krasnobayev V., Koshman S., Kovalchuk D. The data diagnostic method of in the system of residue classes. Advanced Information Systems. 2021. Vol. 5(1). P. 123–128. DOI:https://doi.org/10.20998/2522-9052.2021.1.18.

32. Krasnobayev V., Koshman S., Kovalchuk D. Diagnosing data in a non-positional number system of residual classes. Сучасні напрями розвитку інформаційно-комунікаційних технологій та засобів управління: тези доповідей десятої Міжнародної науково-технічної конференції, 8 – 9 квітня 2021 р., Харків, 2021. Т. 2. С. 25.

33. Янко А. С., Ковальчук Д. М. Дослідження можливості відмовостійкого функціонування комп'ютерної системи в непозиційній системі числення в залишкових класах. Сучасні напрями розвитку інформаційно-комунікаційних технологій та засобів управління: тези доповідей тринадцятої міжнародної науково-технічної конференції, 26 – 27 квітня 2023 р., Харків, 2023. Т.–2. С. 50. DOI:https://doi.org/10.32620/ICT.23.t2.

34. Krasnobayev V. A., Yanko A. S., Kovalchuk D. M. Control, Diagnostics and Error Correction in the Modular Number System. Proceedings of The Sixth International Workshop on Computer Modeling and Intelligent Systems (CMIS 2023), 3 May 2023, Zaporizhzhia, Ukraine, 2023. P. 199-213.

35. Yanko A., Koshman S., Krasnobayev V. Algorithms of data processing in the residual classes system. 2017 4th International Scientific-Practical Conference Problems of Infocommunications. Science and Technology (PIC S&T), Kharkov, Ukraine, 2017. P. 117-121. DOI: https://doi.org/ 10.1109/INFOCOMMST.2017.8246363.

36. Radchenko S., Demchenko K., Piskarov O., Nechitailo J., Panov A. Methods of implementation of arithmetic operations in the residual number system. Proceedings off the X International Scientific and Practical Conference. Lisbon, 2023. P. 426-428

37. Yaskova E., Barsov V., Krasnobaev V., Koshman S., Khery Ali Abdullah. Method of realization of arithmetic operations on the basis of the use of modulyarnoy number system. Radioelectronic and Computer Systems. 2009. Vol. 7 (41). P. 70–73.

38. Яськова К. В., Хері Алі Абдуллах. Принципи реалізації модулярних операцій в модулярній арифметиці. Вісник Харківського національного технічного університету сільського господарства ім. Петра Василенка. Проблеми енергозабезпечення та енергозбереження в АПК України. 2007. Вип. 57, т. 2. С. 100-104.

39. Gorbenko I., Kuznetsov A. ISCI'2017: Information Security in Critical Infrastructures: monograph. USA, 2017. 207 p.

40. Koshman S., Krasnobayev V., Kuznetsov A., Rassomakhin S., Zamula A., Kavun S. Effective Data Processing in Coding, Digital Signals and Cryptography: monograph. ASC Academic Publishing, 2018, 352 p.

41. Николайчук, Я., Грига, В., Возна, Н., Пітух, І., Грига, Л. Високопродуктивні компоненти апаратних багаторозрядних спецпроцесорів сумування та перемноження двійкових чисел. Computer Systems and Information Technologies. 2023. № 2. P. 25–32. DOI:https://doi.org/10.31891/csit-2023-2-3.

42. Krasnobayev V., Kuznetsov A., Bagmut M., Kuznetsova Y. Design of the Residual Adder of Two Numbers. 2021 IEEE 8th International Conference on Problems of Infocommunications, Science and Technology (PIC S&T), Kharkiv, Ukraine, 2021. P. 541-545, DOI:https://doi.org/10.1109/PICST54195.2021.9772132.

43. Краснобаев В., Кузнецова К., Багмут М. Метод виконання операції додавання залишків чисел за модулем. Комп'ютерні науки та кібербезпека. 2021. № 1. С. 4-15. DOI:https://doi.org/10.26565/2519-2310-2021-1-01.

44. Багмут М., Кузнецова К., Горбачова Л. Алгоритм побудови

структури суматору двох залишків чисел по модулю. Комп'ютерні науки та кібербезпека. 2021. № 1. DOI:https://doi.org/10.26565/2519-2310-2021-1-05.

45. Krasnobayev V. A., Kuznetsov A. A., Koshman S. A., Kuznetsova K. O. A method for implementing the operation of modulo addition of the residues of two numbers in the residue number system. Cybernetics and Systems Analysis. 2020. Vol. 56, No. 6. P. 1029-1038. DOI:https://doi.org/10.1007/s10559-020-00323-9.

46. Krasnobayev V., Koshman, S., Kovalchuk D. Synthesis of structure of the adder by module. Control, Navigation and Communication Systems. Academic Journal. 2021. Vol. 1(63). P. 96-99. DOI:https://doi.org/10.26906/SUNZ.2021.1.096.

47. Krasnobayev V., Koshman S., Kovalchuk D. The concept of performing the addition operation in the system of residual classes. Advanced Information Systems. 2022. Vol. 6(1). P. 43–47. DOI:https://doi.org/10.20998/2522-9052.2022.1.07.

48. Krasnobayev V., Yanko A. Kovalchuk D. An Improved Method for Performing the Arithmetic Operations of Modulo Addition of the Remainders of Numbers. 2023 13th International Conference on Dependable Systems, Services and Technologies (DESSERT), Athens, Greece, 13-15 October 2023, pp. 1-6, DOI: https://doi.org/10.1109/DESSERT61349.2023.10416508.

49. Krasnobayev V., Koshman S., Kovalchuk D. Development of the adder structure by modulo of the system of residual classes. Сучасні напрями розвитку інформаційно-комунікаційних технологій та засобів управління: тези доповідей десятої Міжнародної науково-технічної конференції, 8 – 9 квітня 2021 р., Харків, 2021. Т. 2. С. 26.

50. Krasnobayev V., Koshman S., Kovalchuk D., Kuznesova Ye. Development of the adder structure by modulo of the system of residual classes. Проблеми інформатизації: тези доповідей дев'ятої Міжнародної науково-технічної конференції, 18 – 19 листопада 2021 р., Черкаси, 2021. Т. 1. С. 82.

51. Яськова К. В., Хері Алі Абдуллах, Деренько М. С., Краснобаєв В. А. Технічна реалізація операцій модульного додавання і віднімання в модулярній арифметиці. Вісник Харківського національного технічного університету

сільського господарства ім. Петра Василенка. Вип. 73, т. 2., 2007. С. 49-51.

52. Krasnobayev V., Kuznetsov A., Yanko A., Kuznetsova T. The Procedure for Implementing the Operation of Multiplying Two Matrices Using the Residual Number System. 2020 IEEE International Conference on Problems of Infocommunications. Science and Technology (PIC S&T). Kharkiv, Ukraine, 2020. P. 353-357. DOI: https://doi.org/10.1109/PICST51311.2020.9468076.

53. Жураковський Ю. П., Полторак В. П. Теорія інформації і кодування. Київ: Вища школа. 2001. 255 с.

54. Курко А.М., Решетник В.Я. Введення в теорія інформації. Тернопіль: Вид-во ТНТУ ім. Івана Пулюя, 2017. 108 с.

55. Фурман І. О., Краснобаєв В. А., Кошман С. О. Аналіз табличних алгоритмів реалізації модульних операцій у автоматизованих системах обробки цифрової інформації. Вісник Харківського державного технічного університету сільського господарства. Проблеми енергозабезпечення та енергозбереження в АПК України. 2004. Вип. 27. С. 174-178.

56. Koshman S., Barsov V., Krasnobaev V., Yaskova E., Derenko N. Method of bit-by-bit tabular realization of arithmetic operations in the system of residual classes. Radioelectronic and Computer Systems. 2009. № 5 (39). С. 44–48.

57. Krasnobayev V., Koshman S., Yanko A. Method of tabular realization of arithmetic operations in the system of residual classes. Computer science and cybersecurity. 2016. Issue 3(3). P. 28–35. URL: http://periodicals.karazin.ua/cscs/issue/view/533.

58. Krasnobayev V. A., Yanko A. S., Kovalchuk D. M. Methods for tabular implementation of arithmetic operations of the residues of two numbers represented in the system of residual classes. Radio Electronics, Computer Science, Control. 2022. № 4, P. 18-28. DOI:https://doi.org/10.15588/1607-3274-2022-4-2.

59. Krasnobayev V., Koshman S., Kovalchuk D. Method of Tabular Implementation of Modular Arithmetic Operations in the System of Residual Classes. ISCI’2020: Information Security in Critical Infrastructures. Collective monograph. Edited by Gorbenko I., Krasnobayev V. Kuznetsov A. ASC Academic Publishing,

USA, 2020. P. 109-118.

60. Кошман С., Краснобаєв В., Ковальчук Д., Кузнецова Є. Дослідження способів реалізації арифметичних операцій у системі залишкових класів. Проблеми інформатизації: тези доповідей дев'ятої Мжнародної науково-технічної конференції, 18 – 19 листопада 2021 р., Черкаси, 2021. Т. 1. С. 83.

61. Krasnobayev V., Yanko A., Kovalchuk D. Method of Tabular Implementation of the Arithmetic Operation of Multiplying Two Numbers Represented in the System of Residual Classes. 2022 IEEE 9th International Conference on Problems of Infocommunications, Science and Technology (PIC S&T), 10-12 October 2022, Kharkiv, Ukraine, 2022. P. 63-68. DOI:https://doi: 10.1109/PICST57299.2022.10238624.

62. Krasnobayev V., Kuznetsov A., Lokotkova I., Kiian A., Kuznetsova T. Techniques for Raising the Remainder to a Power in the System of Residual Classes. 2020 IEEE 11th International Conference on Dependable Systems, Services and Technologies (DESSERT). Kyiv, Ukraine, 2020. P. 145-150. DOI: https://doi.org/10.1109/DESSERT50317.2020.9125049.

63. Krasnobayev V. A., Yanko A. S., Kovalchuk D. M. Mathematical Model of the Process of Raising Integers to an Arbitrary Power of a Natural Number in the System of Residual Classes. Theoretical and Applied Cybersecurity. 2023. Vol. 5 (2), P. 5-14. DOI: https://doi.org/10.20535/tacs.2664-29132023.2.278891.

64. Krasnobayev V., Yanko A., Martynenko A., Kovalchuk D. Method for computing exponentiation modulo the positive and negative integers. Information processing in control and decision-making systems. Problems and solutions. Monograph. Odessa, 2023. P. 233-257.

65. Krasnobayev V. A., Yanko A. S., Kovalchuk D. M. Mathematical Model and Method of Raising Integers to an Arbitrary Power of a Natural Number in the System of Residual Classes. Математичне та імітаційне моделювання систем (МОДС 2022): тези доповідей сімнадцятої Міжнародної конференції, 14 – 16 листопада 2022 р., Чернігів, Україна, 2023. С. 15.

66. Krasnobayev V., Yanko A., Martynenko A., Kovalchuk D. Method for

computing exponentiation modulo the positive and negative integers. Materials of the XI International Scientific Conference «Information-Management Systems and Technologies», 21-23 September 2023, Odessa, Ukraine, 2023. P. 150-153.

67. Krasnobayev V., Yanko A ., Martynenko A., Kovalchuk D. Method for Computing Exponentiation Modulo the Positive and Negative Integers. Proceedings of the 11-th International Conference «Information Control Systems & Technologies», Odessa, Ukraine, 21-23 September, 2023. P. 374-383.

68. Краснобаєв В. А., Кузнецов О. О., Кузнецова К. О., Ковальчук Д. М. Пристрій для додавання лишків чисел за модулем m_i системи залишкових класів: патент України 126181 Україна: МПК: G06F 7/50 (2006.01), G06F 11/10 (2006.01), G06F 7/72 (2006.01). № a202100522; заявл. 09.02.2021; опубл. 25.08.2022, бюл. № 34/2022. 9 с.

69. Краснобаєв В. А., Кузнецов О. О., Кузнецова К. О., Ковальчук Д. М. Операційний пристрій у системі залишкових класів: патент України 126182 Україна: МПК: G06F 7/50 (2006.01), G06F 7/503 (2006.01), G06F 7/72 (2006.01). № a202100523; заявл. 09.02.2021; опубл. 25.08.2022, бюл. № 34/2022. 9 с.

70. Кошман С. О., Краснобаєв В. А., Кузнецов О. О., Ковальчук Д. М. Суматор за довільним модулем m системи залишкових класів: патент на корисну модель 148170 Україна: МПК: G06F 7/50 (2006.01). № u202100701; заявл. 17.02.2021; опубл. 14.07.2021, бюл. № 28/2021. 9 с.

71. Краснобаєв В. А., Кошман С. О., Кузнецов О. О., Мавріна М. О., Ковальчук Д. М. Операційний пристрій у системі залишкових класів: патент на корисну модель 149074 Україна: МПК (2006): G06F 7/00, G06F 7/72 (2006.01). № u202102897; заявл. 31.05.2021; опубл. 13.10.2021, бюл. № 41/2021. 9 с.

72. Краснобаєв В. А., Кошман С. О., Кузнецов О. О., Мавріна М. О., Ковальчук Д. М. Пристрій для визначення лишків числа за довільним модулем системи залишкових класів: патент на корисну модель149421 Україна: МПК (2006): G06F 5/00. № u202102898; заявл. 31.05.2021; опубл. 17.11.2021, бюл. № 46/2021. 4 с.

Додаток А.

СПИСОК ПУБЛІКАЦІЙ ЗДОБУВАЧА ЗА ТЕМОЮ ДИСЕРТАЦІЇ

Наукові праці, в яких опубліковані основні результати дисертації :

у фахових виданнях України:

1. Krasnobayev V., Koshman S., **Kovalchuk D.** The data diagnostic method of in the system of residue classes. Advanced Information Systems. 2021. Vol. 5(1). P. 123–128. DOI:https://doi.org/10.20998/2522-9052.2021.1.18.

(Особистий внесок: розробка методу діагностики даних, які представлені в СЗК, а також написання частини тексту та його переклад).

2. Krasnobayev V., Koshman, S., **Kovalchuk D.** Synthesis of structure of the adder by module. Control, Navigation and Communication Systems. Academic Journal. 2021. Vol. 1(63). P. 96-99. DOI:https://doi.org/10.26906/SUNZ.2021.1.096.

(Особистий внесок: розробка алгоритму синтезу суматора за довільним модулем СЗК, участь в обговоренні отриманих результатів а також написання частини тексту та його переклад).

3. Krasnobayev V., Koshman S., **Kovalchuk D.** The concept of performing the addition operation in the system of residual classes. Advanced Information Systems. 2022. Vol. 6(1). P. 43–47. DOI:https://doi.org/10.20998/2522-9052.2022.1.07.

(Особистий внесок: розробка методу обчислення суми залишків чисел за довільним модулем, підготовка прикладів, що наочно демонструють ефективність запропонованого методу, участь в обговоренні отриманих результатів, а також написання частини тексту та його переклад).

4. Krasnobayev V., Koshman S., **Kovalchuk D.** The concept of using the number system in the residual classes for building artificial intelligence system. Control, Navigation and Communication Systems. Academic Journal. 2022. Vol. 1(67). P. 65-70. DOI:https://doi.org/https://doi.org/10.26906/SUNZ.2022.1.065.

(Особистий внесок: аналіз можливості застосування непозиційної системи числення у залишкових класах для підвищення швидкодії програмно-апаратних систем і комплексів з елементами штучного інтелекту, а також

написання частини тексту та його переклад).

5. Krasnobayev V., Koshman S., Nikolsky S., **Kovalchuk D.** Mathematical model of computer system reliability in residual classes. Advanced Information Systems. 2022. Vol. 6(4). P. 19–24. DOI:doi: 10.20998/2522-9052.2022.4.03.

(Особистий внесок: розробка математичної моделі надійності комп'ютерної системи, що функціонує у СЗК, участь в обговоренні отриманих результатів, а також написання частини тексту та його переклад).

6. Krasnobayev V. A., Yanko A. S., **Kovalchuk D. M.** Mathematical Model of the Process of Raising Integers to an Arbitrary Power of a Natural Number in the System of Residual Classes. Theoretical and Applied Cybersecurity. 2023. Vol. 5 (2), P. 5-14. DOI: https://doi.org/10.20535/tacs.2664-29132023.2.278891.

(Особистий внесок: розробка математичної моделі процесу піднесення цілих чисел до довільного степеню натурального числа в СЗК, участь в обговоренні отриманих результатів, а також написання частини тексту та його переклад).

Наукові праці, в яких опубліковані основні результати дисертації, що входять до наукометричної бази **Web of Science і Scopus:**

7. Krasnobayev V. A., Yanko A. S., **Kovalchuk D. M.** Methods for tabular implementation of arithmetic operations of the residues of two numbers represented in the system of residual classes. Radio Electronics, Computer Science, Control. 2022. № 4, P. 18-28. DOI:https://doi.org/10.15588/1607-3274-2022-4-2. (Web of Science)

(Особистий внесок: розробка табличного методу реалізації цілочисельних арифметичних модульних операцій додавання, віднімання та множення, участь в обговоренні отриманих результатів, а також написання частини тексту та його переклад).

8. Koshman S., Krasnobayev V., Nikolsky S., **Kovalchuk D.** The structure of the computer system in the residual classes. Advanced Information Systems. 2023. Vol. 7(2). P. 41–48. DOI:https://doi: 10.20998/2522-9052.2023.2.06. (Scopus)

(Особистий внесок: постановка та вирішення зворотної задачі оптимального резервування в СЗК на основі використання методу динамічного програмування, а також написання частини тексту та його переклад).

Наукові праці, які засвідчують апробацію матеріалів дисертації:

9. Krasnobayev V., Koshman S., **Kovalchuk D.** Diagnosing data in a non-positional number system of residual classes. Сучасні напрями розвитку інформаційно-комунікаційних технологій та засобів управління: тези доповідей десятої Міжнародної науково-технічної конференції, 8 – 9 квітня 2021 р., Харків, 2021. Т. 2. С. 25.

10. Krasnobayev V., Koshman S., **Kovalchuk D.** Development of the adder structure by modulo of the system of residual classes. Сучасні напрями розвитку інформаційно-комунікаційних технологій та засобів управління: тези доповідей десятої Міжнародної науково-технічної конференції, 8 – 9 квітня 2021 р., Харків, 2021. Т. 2. С. 26.

11. Krasnobayev V., Koshman S., **Kovalchuk D.**, Kuznesova Ye. Development of the adder structure by modulo of the system of residual classes. Проблеми інформатизації: тези доповідей дев'ятої Міжнародної науково-технічної конференції, 18 – 19 листопада 2021 р., Черкаси, 2021. Т. 1. С. 82.

12. Кошман С., Краснобаєв В., **Ковальчук Д.**, Кузнецова Є. Дослідження способів реалізації арифметичних операцій у системі залишкових класів. Проблеми інформатизації: тези доповідей дев'ятої Мжнародної науково-технічної конференції, 18 – 19 листопада 2021 р., Черкаси, 2021. Т. 1. С. 83.

13. Krasnobayev V., Yanko A., **Kovalchuk D.** Method of Tabular Implementation of the Arithmetic Operation of Multiplying Two Numbers Represented in the System of Residual Classes. 2022 IEEE 9th International Conference on Problems of Infocommunications, Science and Technology (PIC S&T), 10-12 October 2022, Kharkiv, Ukraine, 2022. P. 63-68. DOI:https://doi: 10.1109/PICST57299.2022.10238624. (Міжнародна конференція Scopus)

14. Krasnobayev V. A., Yanko A. S., **Kovalchuk D. M.** Mathematical

Model and Method of Raising Integers to an Arbitrary Power of a Natural Number in the System of Residual Classes. Математичне та імітаційне моделювання систем (МОДС 2022): тези доповідей сімнадцятої Міжнародної конференції, 14 – 16 листопада 2022 р., Чернігів, Україна, 2023. С. 15.

15. Янко А. С., **Ковальчук Д. М.** Дослідження можливості відмовостійкого функціонування комп'ютерної системи в непозиційній системі числення в залишкових класах. Сучасні напрями розвитку інформаційно-комунікаційних технологій та засобів управління: тези доповідей тринадцятої міжнародної науково-технічної конференції, 26 – 27 квітня 2023 р., Харків, 2023. Т.–2. С. 50. DOI:https://doi.org/10.32620/ICT.23.t2.

16. Krasnobayev V. A., Yanko A. S., **Kovalchuk D. M.** Control, Diagnostics and Error Correction in the Modular Number System. Proceedings of The Sixth International Workshop on Computer Modeling and Intelligent Systems (CMIS 2023), 3 May 2023, Zaporizhzhia, Ukraine, 2023. P. 199-213. (Міжнародна конференція Scopus)

17. Krasnobayev V., Yanko A., Martynenko A., **Kovalchuk D.** Method for computing exponentiation modulo the positive and negative integers. Materials of the XI International Scientific Conference «Information-Management Systems and Technologies», 21-23 September 2023, Odessa, Ukraine, 2023. P. 150-153.

18. Krasnobayev V., Yanko A ., Martynenko A., **Kovalchuk D.** Method for Computing Exponentiation Modulo the Positive and Negative Integers. Proceedings of the 11-th International Conference «Information Control Systems & Technologies», Odessa, Ukraine, 21-23 September, 2023. P. 374-383. (Міжнародна конференція Scopus)

19. Krasnobayev V., Yanko A. **Kovalchuk D.** An Improved Method for Performing the Arithmetic Operations of Modulo Addition of the Remainders of Numbers. 2023 13th International Conference on Dependable Systems, Services and Technologies (DESSERT), Athens, Greece, 13-15 October 2023, pp. 1-6, DOI: https://doi.org/10.1109/DESSERT61349.2023.10416508.

Наукові праці, які додатково відображають наукові результати дисертації:

Монографії

20. Krasnobayev V., Koshman S., **Kovalchuk D.** Method of Tabular Implementation of Modular Arithmetic Operations in the System of Residual Classes. ISCI'2020: Information Security in Critical Infrastructures. Collective monograph. Edited by Gorbenko I., Krasnobayev V. Kuznetsov A. ASC Academic Publishing, USA, 2020. P. 109-118. ISBN: 978-1-7362833-0-1 (Hardback), ISBN: 978-1-7362833-1-8 (Ebook).

(Особистий внесок: розробка табличного методу реалізації цілочисельних арифметичних модульних операцій додавання, віднімання та множення, участь в обговоренні отриманих результатів, а також написання частини тексту та його переклад).

21. Krasnobayev V., Yanko A., Martynenko A., **Kovalchuk D.** Method for computing exponentiation modulo the positive and negative integers. Information processing in control and decision-making systems. Problems and solutions. Monograph. Odessa, 2023. P. 233-257.

(Особистий внесок: розробка математичної моделі процесу піднесення цілих чисел до довільного степеню натурального числа в СЗК, участь в обговоренні отриманих результатів, а також написання частини тексту та його переклад).

Патенти

22. Краснобаєв В. А., Кузнецов О. О., Кузнецова К. О., **Ковальчук Д. М.** Пристрій для додавання лишків чисел за модулем m_i системи залишкових класів: патент України 126181 Україна: МПК: G06F 7/50 (2006.01), G06F 11/10 (2006.01), G06F 7/72 (2006.01). № a202100522; заявл. 09.02.2021; опубл. 25.08.2022, бюл. № 34/2022. 9 с.

23. Краснобаєв В. А., Кузнецов О. О., Кузнецова К. О.,

Ковальчук Д. М. Операційний пристрій у системі залишкових класів: патент України 126182 Україна: МПК: G06F 7/50 (2006.01), G06F 7/503 (2006.01), G06F 7/72 (2006.01). № a202100523; заявл. 09.02.2021; опубл. 25.08.2022, бюл. № 34/2022. 9 с.

24. Кошман С. О., Краснобаєв В. А., Кузнецов О. О., **Ковальчук Д. М.** Суматор за довільним модулем m системи залишкових класів: патент на корисну модель 148170 Україна: МПК: G06F 7/50 (2006.01). № u202100701; заявл. 17.02.2021; опубл. 14.07.2021, бюл. № 28/2021. 9 с.

25. Краснобаєв В. А., Кошман С. О., Кузнецов О. О., Мавріна М. О., **Ковальчук Д. М.** Операційний пристрій у системі залишкових класів: патент на корисну модель 149074 Україна: МПК (2006): G06F 7/00, G06F 7/72 (2006.01). № u202102897; заявл. 31.05.2021; опубл. 13.10.2021, бюл. № 41/2021. 9 с.

26. Кошман С. О., Краснобаєв В. А., Кузнецов О. О., Мавріна М. О., **Ковальчук Д. М.** Пристрій для контролю та виправлення однократних помилок у даних, які представлені системою залишкових класів: патент на корисну модель 149060 Україна: МПК: G06F 7/50 (2006.01). № u202102707; заявл. 24.05.2021; опубл. 13.10.2021, бюл. № 41/2021. 6 с.

27. Краснобаєв В. А., Кошман С. О., Кузнецов О. О., Мавріна М. О., **Ковальчук Д. М.** Пристрій для визначення лишків числа за довільним модулем системи залишкових класів: патент на корисну модель149421 Україна: МПК (2006): G06F 5/00. № u202102898; заявл. 31.05.2021; опубл. 17.11.2021, бюл. № 46/2021. 4 с.

Онлайн сервіс створення та перевірки кваліфікованого та удосконаленого електронного підпису

ПРОТОКОЛ
створення та перевірки кваліфікованого та удосконаленого електронного підпису

Дата та час: 17:41:02 21.04.2024

Назва файлу з підписом: Kovalchuk_diss.pdf.asice
Розмір файлу з підписом: 2.4 МБ

Перевірені файли:
Назва файлу без підпису: Kovalchuk_diss.pdf
Розмір файлу без підпису: 4.3 МБ

Результат перевірки підпису: Підпис створено та перевірено успішно. Цілісність даних підтверджено

Підписувач: Ковальчук Дмитро Миколайович
П.І.Б.: Ковальчук Дмитро Миколайович
Країна: Україна
РНОКПП: 3558807672
Час підпису (підтверджено кваліфікованою позначкою часу для підпису від Надавача): 17:40:53 21.04.2024
Сертифікат виданий: "Дія". Кваліфікований надавач електронних довірчих послуг
Серійний номер: 382367105294AF9704000000E0572600CB755F01
Тип носія особистого ключа: ЗНКІ криптомодуль ІІТ Гряда-301
Алгоритм підпису: ДСТУ 4145
Тип підпису: Кваліфікований
Тип контейнера: Підпис та дані в архіві (розширений) (ASiC-E)
Формат підпису: З повними даними ЦСК для перевірки (CAdES-X Long)
Сертифікат: Кваліфікований

Версія від: 2024.04.15 13:00