

ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ОДЕСЬКА ПОЛІТЕХНІКА»
МІНІСТЕРСТВА ОСВІТИ І НАУКИ УКРАЇНИ
Кафедра комп'ютерних інтелектуальних систем та мереж

ГОРЯЧКО Микита Сергійович

ДИПЛОМНА РОБОТА МАГІСТРА

**ДОСЛІДЖЕННЯ ДОСТОВІРНОСТІ КОНТРОЛЮ ПОМНОЖУВАЧА
ЗА ПРОЯВОМ КРАТНИХ ВІДМОВ**

Спеціальність 123 – Комп'ютерна інженерія
Спеціалізація – Комп'ютерні системи та мережі

Керівник: Дрозд Олександр Валентинович,
доктор технічних наук, професор

Одеса – 2021

ОДЕСЬКИЙ НАЦІОНАЛЬНИЙ ПОЛІТЕХНІЧНИЙ УНІВЕРСИТЕТ

Інститут комп'ютерних систем Кафедра КІСМ
Спеціальність Комп'ютерні інтелектуальні системи та мережі

ЗАТВЕРДЖУЮ:

Зав. кафедри КІСМ

“ ” 2021 р.

ЗАВДАННЯ

НА МАГІСТЕРСЬКУ РОБОТУ СТУДЕНТОВІ

Горячку Микиті Сергійовичу

1. Тема роботи “Дослідження методу контролю помножувача мантис при кратних несправностях”

затверджена наказом по університету від “ ” 2021 р. № -

2. Термін здачі студентом закінченої роботи 05.12.2021

3. Вихідні дані до роботи у розділі “Завдання на проведення досліджень”

4. Зміст розрахунково-пояснювальної записки (перелік питань, що належить розробити) _____

1. Робоче діагностування арифметичних компонентів в системах критичного застосування

2. Завдання на проведення досліджень

3. Програмна модель для дослідження достовірності контролю помножувача мантис

4. Результати проведених досліджень

5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень)

1) Виникнення кратних відмов в системах критичного застосування

2) Виконання операції множення мантис у доповняльному коді

3) Контроль за модулем повного та скороченого множення мантис

4) Введення кратних відмов в модель помножувача мантис

5) Моделювання контролю помножувача мантис при кратних відмовах

6) Результати моделювання контролю помножувача мантис

7) Виявлення суттєвих помилок добутку при кратних відмовах

8) Виявлення несуттєвих помилок добутку при кратних відмовах

9) Достовірність контролю добутку мантис при кратних відмовах.

10) Ефективність методу контролю при скороченні обчислень та кратних відмовах.

6. КОНСУЛЬТАНТИ ПО РОБОТІ, ІЗ ЗАЗНАЧЕННЯМ РОЗДІЛІВ РОБОТИ, ЩО СТОСУЮТЬСЯ ЇХ

		Підпис, дата	
		Завдання видав	Завдання прийняв

7. Дата видачі завдання _____ 15.03.2021 _____

КАЛЕНДАРНИЙ ПЛАН

№ п/п	Назва етапів магістерської роботи	Термін виконання етапів роботи	Примітка
1	<i>Вивчення науково-технічної та патентної літератури з питань робочого діагностування арифметичних пристроїв</i>	<i>15.03 - 23.04</i>	
2	<i>Визначення вимог до розробки щодо виконання магістерської роботи та постановка задачі на проведення досліджень</i>	<i>24.04 - 10.05</i>	
3	<i>Розробка алгоритмів та моделі системи контролю матричного помножувача мантис із введенням кратних відмов</i>	<i>11.05 - 30.06</i>	
4	<i>Верифікація моделі системи контролю матричного помножувача мантис, що діє в умовах прояву кратних несправностей</i>	<i>01.09 - 17.10</i>	
5	<i>Дослідження на моделі достовірності та ефективності контролю результатів, обчислюваних під дією кратних несправностей</i>	<i>18.10 - 07.11</i>	
6	<i>Оцінка результатів проведеного дослідження</i>	<i>08.11 - 15.12</i>	

Студент _____ Горячко М. В.
 Керівник роботи _____ Дрозд О.В.

№ рядка	Формат	Позначення	Найменування	Кільк.	№ екз.	Примітка
1			Документація загальна			
2			<u>Знов розроблена</u>			
3	A4	АМДР.АМ161.2020.01ПЗ	Пояснювальна записка	105		
4	A1	АМДР.АМ161.2020.01Д1	Виникнення кратних відмов	1		
5			в системах критичного			
6			застосування			
7	A1	АМДР.АМ161.2020.01Д2	Виконання операції множення	1		
8			мантис у доповняльному коді			
9	A1	АМДРАМ161.2020.01Д3	Контроль за модулем повного та	1		
10			скороченого множення мантис			
11	A1	АМДР.АМ161.2020.01Д4	Введення кратних відмов	1		
12			у модель помножувача мантис			
13	A1	АМДР.АМ161.2020.01Д5	Моделювання контролю	1		
14			помножувача мантис при кратних			
15			відмовах			
16	A1	АМДР.АМ161.2020.01Д6	Результати моделювання	1		
17			контролю помножувача мантис			
18	A1	АМДР.АМ161.2020.01Д7	Виявлення суттєвих помилок	1		
19			добутку при кратних відмовах			
20	A1	АМДР.АМ161.2020.01Д8	Виявлення несуттєвих помилок	1		
21			добутку при кратних відмовах			
22	A1	АМДР.АМ161.2020.01Д9	Достовірність контролю добутку	1		
23			мантис при кратних відмовах			
24	A1	АМДР.АМ161.2020.01Д10	Ефективність методу контролю	1		
25			при скороченні обчислень та			
26			кратних відмовах			
27						
28						

АММР.АМ161.2020

Ізм.	Лист	№ документа	Підпис	Дата			
Розробив	Горячко				Дослідження достовірності контролю помножувача за проявом кратних відмов		
Перевірів	Дрозд						
Н. контр.					Відомість дипломної роботи		
Затверд.							
					Лист.	Лист	Листів
						1	1
					ОНПУ ІКС КІСМ АМ161		

АНОТАЦІЯ

Горячко М. С. Дослідження достовірності контролю помножувача за проявом кратних відмов. – Магістерська кваліфікаційна робота. Одеса, 2021: 105 с., 23 рис., 9 табл., 1 додаток, 18 джерел.

Об'єкт дослідження – процес контролю наближених результатів на виходах матричного помножувача мантис у додатковому коді при кратних відмовах.

Предмет дослідження – достовірність контролю наближених добутків в умовах дії кратних відмов під час аварійного режиму системи критичного застосування.

Мета роботи – оцінка достовірності контролю за модулем три наближених добутків на виходах матричного помножувача мантис у додатковому коді під дією кратних відмов, що проявляються в аварійному режимі системи критичного застосування.

Робота присвячена питанням робочого діагностування матричних обчислювальних пристроїв при виконанні наближених обчислень. Досліджується метод контролю за модулем помножувача мантис у додатковому коді в умовах дії кратних відмов в аварійному режимі системи критичного застосування. На програмній моделі контролю визначені ймовірності виявлення та пропуску суттєвих та несуттєвих помилок при внесенні кратних відмов. Оцінюється достовірність контролю в умовах дії кратних відмов.

СИСТЕМА КРИТИЧНОГО ЗАСТОСУВАННЯ, АВАРІЙНИЙ РЕЖИМ, МАТРИЧНИЙ ПОМНОЖУВАЧ МАНТИС, РОБОЧЕ ДІАГНОСТУВАННЯ, КРАТНІ ВІДМОВИ, КОНТРОЛЬ ЗА МОДУЛЕМ, ДОДАТКОВИЙ КОД, ДОСТОВІРНІСТЬ КОНТРОЛЮ.

АННОТАЦИЯ

Горячко Н. С. Исследование достоверности контроля умножителя при проявлении кратных отказов. – Магистерская квалификационная работа. Одесса, 2021: 105 с., 23 рис., 9 табл., 1 приложение, 18 источников.

Объект исследования – процесс контроля приближенных результатов на выходах матричного умножителя мантисс в дополнительном коде при кратных отказах.

Предмет исследования – достоверность контроля по модулю три приближенных произведений на выходах матричного умножителя мантисс в дополнительном коде под действием кратных отказов во время аварийного режима системы критического применения.

Цель работы – оценка достоверности контроля по модулю три приближенных произведений на выходах матричного умножителя мантисс в дополнительном коде под действием кратных отказов, проявляющихся в аварийном режиме систем критического применения.

Работа посвящена вопросам рабочего диагностирования матричных вычислительных устройств при выполнении приближенных вычислений. Исследуется метод контроля по модулю умножителя мантисс в дополнительном коде в условиях действия кратных отказов в аварийном режиме системы критического применения. На программной модели контроля определены вероятности обнаружения и пропуска, существенных и несущественных ошибок при внесении кратных отказов. Оценивается достоверность контроля в условиях действия кратных отказов.

СИСТЕМА КРИТИЧЕСКОГО ПРИМЕНЕНИЯ, АВАРИЙНЫЙ РЕЖИМ, МАТРИЧНЫЙ УМНОЖИТЕЛЬ МАНТИСС, РАБОЧЕЕ ДИАГНОСТИРОВАНИЕ, КРАТНЫЕ ОТКАЗЫ, КОНТРОЛЬ ПО МОДУЛЮ, ДОПОЛНИТЕЛЬНЫЙ КОД, ДОСТОВЕРНОСТЬ КОНТРОЛЯ.

ABSTRACT

Horiachko M. S. Investigation of the trustworthiness of the multiplier control in case of multiple failures. – Master's qualification work. Odessa, 2021: 105 p., 23 fig., 9 tab., 1 appendix, 18 sources.

Object of study – process of checking the approximate results at outputs of the iterative array multiplier of mantissas in two's complement code in case of the multiply faults.

Subject of study – trustworthiness of residue checking by modulo three of the approximate products at outputs of the iterative array multiplier of mantissas in two's complement code under the influence of the multiple faults which is shown in emergency mode of safety-related systems.

Purpose – estimation in reliability of residue checking by modulo three of the approximate calculations at outputs of the iterative array multiplier of mantissas in two's complement code under the influence of the multiple faults during emergency mode of safety-related systems.

Work is dedicated to questions of on-line testing the array computing units in case of executing the approximated calculations. A method of residue checking the mantissas multiplier in two's complement code in conditions of multiply faults action in emergency mode of safety-related system is investigated. On the program model the probabilities of detection and the skipping of the essential and inessential errors in case of injecting the multiply faults are determined. Trustworthiness of the checking in condition of multiply faults action is estimated.

SAFETY-RELATED SYSTEM, EMERGENCY MODE, ITERATIVE ARRAY MULTIPLIER OF MANTISSAS, ON-LINE TESTING, MULTIPLE FAILURES, RESIDUE CHECKING, TWO'S COMPLEMENT CODE, TRUSTVORTHINESS OF THE CHECKIN

ЗМІСТ

Перелік умовних скорочень	5
Вступ.....	6
1 Робоче діагностування арифметичних компонентів в системах критичного застосування.....	10
1.1 Особливості виконання робочого діагностування в системах критичного застосування.....	10
1.2 Проблема кратних відмов.....	13
1.3 Контроль цифрових компонентів за числовим модулем	17
1.4 Скорочене множення та його контроль за модулем.....	21
1.5 Контроль за модулем множення в доповняльному коді	26
1.6 Достовірність контролю за модулем добутків мантис	27
1.7 Висновки	31
2 Завдання на проведення досліджень	33
2.1 Найменування та область застосування.....	33
2.2 Призначення досліджень	34
2.3 Підготовка та проведення досліджень	34
2.4 Постановка задачі.....	35
3 Програмна модель для дослідження достовірності контролю помножувача мантис	38
3.1 Структурне побудування програмної моделі.....	38
3.2 Режими функціонування програмної моделі.....	39
3.3 Алгоритми програмної моделі.....	40
3.4 Висновки	49
4 Результати проведених досліджень.....	50
4.1 Установки програмної моделі.....	50
4.2 Моделювання та його результати	56

4.3 Висновки	76
Висновок.....	78
Перелік посилань.....	81
Додаток А. Програмна модель для дослідження методу контролю	83

ПЕРЕЛІК УМОВНИХ СКОРОЧЕНЬ

ВМ – вузол множення

ВВ – вузол вирахування

КБ – контрольний блок

КП – контролепридатність

МКД – матриця кон'юнкцій добутку

НЕ – номер експерименту

ОД – обсяг діапазону

ОП – обчислювальний пристрій

ФО – формувач остачі

ВСТУП

Постійно збагачення навколишнього середовища об'єктами підвищеного ризику: атомними, тепловими та гідроелектростанціями, потягами підвищеної швидкості й літаками, енергетичними мережами, космічними апаратами та різними видами сучасного озброєння у різних галузях народного господарства; на транспорті та в енергетиці, в космічній та оборонній техніці. Таких об'єктів не тільки стає все більше, а й самі об'єкти розбудовуються та зростають у складності та потужності. Це збільшує ризики, що пов'язані з їхнім життєвим циклом, включаючи будівництво, функціонування та виведення з експлуатації.

Функціональна безпека об'єктів підвищеного ризику підтримується розвитком інформаційно-управляючих систем критичного застосування, які постійно й оперативно відслідковують стан об'єкту та самої системи, широко використовуючи для цього методи та засоби робочого діагностування.

Разом з тим, можливості робочого діагностування суттєво обмежуються контролепридатністю цифрових схем, яка у інформаційно-управляючих системах критичного застосування погіршується внаслідок значної надмірності схем, що будуються відмово стійкими, та двох режимів роботи: нормального та аварійного.

За цими обставинами виникає проблема прихованих несправностей, що можуть накопичуватися у нормальному режимі та проявлятися в аварійному режимі значною кількістю, тобто кратними відмовами, зі зменшенням відмовостійкості, на якій базується функціональна безпека інформаційно-управляючих систем критичного застосування.

Тому важливою задачею становиться проведення досліджень ефективності функціонування робочого діагностування в умовах виникнення кратних відмов.

Об'єктом діагностування вибрано матричний помножувач мантис в доповняльному коді. Такий вибір пояснюється низкою причин.

Перша причина полягає в тому, що операція множення є ключовою для наближених обчислень, оскільки присутня у самому представленні наближених даних у форматах з плаваючою точкою. За цими обставинами усі дії, що виконуються над мантисами, у тому чи іншому вигляді містять операцію множення, а їхні результати наслідують властивості добутку. При тому, значущість наближених обчислень постійно зростає, і відповідно зростає апаратна підтримка їх виконання у комп'ютерних системах.

Друга причина пов'язана з тим, що обчислення, як за правило, виконуються у доповняльному коді. Для операцій додавання та ділення це природно, а операція множення ускладнюється у порівнянні з її виконанням у прямому коді, однак не більше ніж складність схем переведення прямого коду до доповняльного та доповняльного до прямого, що необхідно робити при послідовності декілька різних операцій. Разом з тим, на переведенні кодів втрачається багато часу, що знижує швидкодію та продуктивність систем. Тому помножувачі також використовуються реалізованими у доповняльному коді.

Третя причина витікає із сучасного рівня елементної бази – програмованих логічних інтегральних схем, що містять масиви однорідних операційних елементів, які легко об'єднуються у матриці для виконання складних операцій за один такт. Тому помножувачі мантис зараз будуються однотоктними, матричними, що забезпечує високу продуктивність комп'ютерних систем.

Робоче діагностування матричного помножувача мантис виконується за найбільш поширеним для арифметичних операцій контролем за числовим модулем. Цей метод демонструє високу ефективність при його застосуванні для виконання точних обчислень, оскільки сам є цілочисловим.

Однак виконання наближених обчислень та їх поширення на цифрові компоненти систем критичного застосування суттєво змінюють умови імплементації методів та засобів робочого діагностування в сучасні програмовані інтегральні схеми. Ефективність методів робочого діагностування цифрових компонентів оцінюється достовірністю контролю обчислюваних результатів.

Достовірність методу контролю за числовим модулем наближених результатів необхідно досліджувати в умовах критичного застосування з урахування можливої дії кратних відмов у аварійному режимі інформаційно-управляючої системи.

Таким чином, тема роботи, що присвячена дослідженню контролю за модулем матричного помножувача мантис у доповняльному в умовах дії кратних відмов аварійного режиму систем критичного застосування відноситься до своєчасних та актуальних.

Метою роботи є оцінка достовірності контролю за модулем три наближених добутків на виходах матричного помножувача мантис у доповняльному коді під дією кратних відмов, що проявляються в аварійному режимі системи критичного застосування.

Для досягнення поставленої мети вирішуються наступні задачі:

- вивчається метод контролю за модулем операції множення мантис при виконанні в цифрових компонентах комп'ютерних систем повної та скороченої операції у доповняльному коді;
- розробляється програмна модель матричного помножувача мантис у доповняльному коді із засобами контролю за модулем три для випадків повного та скороченого виконання операції та генерації кратних відмов в об'єкті діагностування;
- перевіряється правильність розробленої моделі матричного помножувача мантис у доповняльному коді із засобами контролю;
- виконується моделювання роботи матричного помножувача мантис та його контролю за модулем три, за яким досліджуються ймовірності виявлення суттєвих і несуттєвих помилок, що викликаються в схемі кратними відмовами;
- порівнюються результати моделювання, що отримуються для випадків повного та скороченого виконання операції множення мантис під дією відмов різної кратності.

Об'єктом дослідження є процес контролю наближених результатів на виходах матричного помножувача мантис у доповняльному коді при кратних відмовах.

Предметом досліджень є достовірність контролю наближених добутків в умовах дії кратних відмов під час аварійного режиму системи критичного застосування.

До елементів наукової новизни варто віднести:

- розробку програмної моделі однократного матричного помножувача мантис у доповняльному коді з числовим контролем за модулем три та можливостями внесення кратних відмов, характерних для аварійного режиму роботи інформаційно-управляючих систем критичного застосування;
- дослідження ймовірності виявлення суттєвих і несуттєвих помилок, що викликаються в матричному помножувачі мантис кратними відмовами;
- виконання порівняльної оцінки достовірності методу контролю за числовим модулем три для випадків повного та скороченого множення мантис під дією кратних відмов.

Практичне значення проведених досліджень полягає в оцінці реальних можливостей методів та засобів робочого діагностування у виявленні суттєвих помилок, що викликаються в цифрових схемах матричних пристроїв кратними несправностями в аварійному режимі роботи інформаційно-управляючих систем критичного забезпечення. Такі дослідження сприяють підвищенню функціональної безпеки об'єктів підвищеного ризику.

1 РОБОЧЕ ДІАГНОСТУВАННЯ АРИФМЕТИЧНИХ КОМПОНЕНТІВ В СИСТЕМАХ КРИТИЧНОГО ЗАСТОСУВАННЯ

1.1 Особливості виконання робочого діагностування в системах критичного застосування

1.1.1 Спостерігається безперервний кількісний та якісний розвиток об'єктів підвищеного ризику, таких як хімічні підприємства, теплові, атомні та гідроелектростанції, швидкісний транспорт, повітряний флот, космічні апарати, що відбувається за їх зростання у чисельності, складності та потужності, призводить до постійного підвищення рівня загроз техногенних аварій та самих аварій.

Компенсуючим підходом до такого розвитку подій, що врівноважує зростаючі ризики аварій, є комп'ютерні інформаційні технології, що постійно вдосконалюються й приймають на себе відповідальність за підтримку безпеки на об'єктах підвищеного ризику.

Такі обставини вимагають разом з об'єктами підвищеного ризику будувати інформаційно-управляючі системи критичного застосування, що є комп'ютерними системами з імплементованими до них інформаційними технологіями. Вони безпосередньо спрямовані на забезпечення функціональної безпеки об'єкту управління та самої системи для упередження аварій та у випадку, якщо аварія стає неминучою – зменшення її наслідків [1].

За вимогами, що ставляться до інформаційно-управляючих систем критичного застосування, вони повинні постійно оцінювати стан об'єкту підвищеного ризику та власний стан, що виконується з використанням технічної діагностики, а, враховуючи надзвичайні вимоги до оперативності, насамперед, методами та засобами робочого діагностування [2].

1.1.2 Системи критичного застосування проектуються для роботи в двох режимах: нормальному та аварійному. Тому ці системи є парадоксальними, оскільки проектуються заради аварійного режиму, а основний час працюють в нормальному режимі.

Основні вимоги до функціональної безпеки інформаційно-управляючих систем критичного застосування регламентуються вітчизняними й міжнародними – європейськими та загальносвітовими стандартами, що передбачають побудову схем до відмовобезпечних цифрових компонентів на основі забезпечення відмовостійкості їх схем.

В основі відмовостійких рішень лежать методи та засоби робочого діагностування цифрових схем з використанням коригувальних кодів та різних видів резервування і реконфігурування блоків, побудовою мажоритарних і багатOVERСІЙНИХ структур [3].

Разом з тим, відмовостійкість цифрових компонентів не вирішує проблеми їх низької контролепридатності в нормальному режимі, коли на входи схем надходить обмежена множина слів, на яких багато точок схем не беруть участь у виконанні обчислень і немає гарантії, що правильне значення не зафіксоване назавжди константною відмовою.

Забезпечення відмовостійкості схемотехнічних рішень суттєво додає апаратного ускладнення та структурної надмірності в схеми цифрових компонентів, а структурна надмірність завжди є запорукою ще більшого зниження контролепридатності схем. Ця надмірність посилюється обмеженістю множини вхідних слів, що приходять на входи схем цифрових компонентів у нормальному режимі [4].

Наприклад, повний двійковий додавач, що обчислює функції суми S та перенесення C для трьох вхідних змінних a , b та c , може працювати в нормальному режимі інформаційно-управляючої системи критичного застосування тільки на трьох з восьми вхідних слів, що складає 37,5%, тобто значну частину їх повної множини.

В табл. 1.1 наведено функції, що реалізуються в повному двійковому додавачі на двох його виходах: перенесення C та суми S , спочатку при функціонуванні на всіх восьми наборах вхідних змінних a , b та c , а потім тільки на трьох перших наборах з наступним можливим довизначенням значень функцій на невикористаних наборах.

Таблиця 1.1 – Функції повного додавача на обмеженій множині вхідних слів

c	b	a	C	S	C	S	C	S
0	0	0	0	0	0	0	0	0
0	0	1	0	1	0	1	0	1
0	1	0	0	1	0	1	0	1
0	1	1	1	0	–	–	0	1
1	0	0	0	1	–	–	0	0
1	0	1	1	0	–	–	0	1
1	1	0	1	0	–	–	0	1
1	1	1	1	1	–	–	0	1

Як це видно з таблиці, функції перенесення C та суми S на обмеженій множині вхідних слів не відрізняються від результату їх довизначення відповідно в нульову функцію та функцію АБО двох аргументів a та b .

З цього прикладу витікає, що повний двійковий додавач в нормальному режимі системи критичного застосування може приховувати константні відмови типу «0» на виході перенесення та несправності типу «монтажне АБО» замикання входів змінних a та b .

Таким чином, самі особливості систем критичного застосування, що полягають у наявності двох режимів роботи та схемної надмірності внаслідок побудування цифрових компонентів відмово стійкими, призводять до низького рівня контролепридатності, що суттєво обмежує можливості робочого діагностування.

Низьку контролепридатність схемотехнічних рішень, вбудованих в цифрові компоненти інформаційно-управляючих систем критичного застосування, підвищують тестовим діагностуванням з використанням небезпечного імітаційного режиму, що відтворює аварійні умови і неодноразово призводило до несанкціонованого включення цієї імітації людиною або відмовою та переходу в аварійний режим [5].

1.2 Проблема кратних відмов

1.2.1 Недостатня контролепридатність схемотехнічних рішень цифрових компонентів у складі систем критичного застосування в нормальному режимі та головне різниця для нормального та аварійного режимів породжує проблему кратних відмов.

Ця проблема полягає в накопиченні прихованих несправностей протягом довготривалого нормального режиму з його низькою активністю та їх прояву з підвищенням активності вхідних даних з початком аварійного режиму, що веде до зниження відмовостійкості схем цифрових компонентів та відповідно функціональної безпеки інформаційно-управляючої системи критичного застосування [6].

Точки схеми цифрового компонента, де можуть накопичуватися такі приховані відмови, що проявляються в аварійному режимі зниженням відмовостійкості та функціональної безпеки системи критичного застосування, відносяться до потенційно загрозливих.

На лабораторних заняттях по курсу дисципліни «Проектування та діагностика систем критичного застосування», що розроблено за підтримкою проекту TEMPUS SAFEGUARD «National Safeware Engineering Network of Centres of Innovative Academia-Industry Handshaking» (158886-TEMPUS-1-2009-1-UK-TEMPUS-JPCR) досліджувався схема одноконтурного матричного

помножувача мантис на наявність потенційно загрозливих точок при різних діапазонах вхідних слів [7].

1.2.2 На рис. 1 показана головна панель програми, що досліджує матричний помножувач, виявляючи потенційно загрозливі точки та визначаючи контролепридатність схеми для розрядності мантис $n = 8$.

Change of R D N	1	2	3	4	5	6	7	8
Quan-ty Danger.p.	223	164	107	97	58	54	46	43
CHECKABILITY	45,4%	59,9%	73,8%	76,3%	85,8%	86,8%	88,8%	89,5%

Рисунок 1.1 – Основна панель програми для розрядності $n = 8$

Програма дозволяє задати 4 параметри: базове значення (Base Value) та діапазон зміни співмножників (Range of Data) у нормальному (Normal) та аварійному (Emergency) режимах.

Співмножники змінюються в однаковому діапазоні з кроком 1, починаючи з базового значення.

В аварійному режимі співмножники приймають значення порогу, з якого починається аварійний стан.

Діапазон зміни співмножників у аварійному режимі дозволяє аналізувати роботу матричного помножувача на «розмитому» порозі, який приймає декілька значень підряд.

Точка цифрової схеми є потенційно загрозовою, якщо оцінена такою на всіх цих значеннях.

Один із параметрів, що задається, можна зробити змінним у певному діапазоні, вказуючи його початкове значення (From), крок зміни (Step) та верхню границю (Up to).

Якщо верхня границя не перевищена, то задається 8 різних значень параметра, і проводиться 8 експериментів.

В кожному з них обчислюється контролепридатність цифрової схеми однотактного матричного помножувача з показом потенційно загрозових точок схеми та їх кількості.

У даному експерименті задано базові значення 128 та 245 для нормального та аварійного режиму, відповідно.

Обсяг діапазону в нормальному режимі визначено змінним параметром, який змінюється від значення 10 з кроком 10 до верхньої границі 80.

Таким чином, двійковий код мантис співмножників змінюється в нормальному режимі від 128 до 137 у першому експерименті, до 147 у другому експерименті та до 207 в останньому, восьмому.

Обсяг 1 діапазону в аварійному режимі визначає поріг на рівні базового значення 245.

Внизу панелі показано кількість потенційно загрозових точок схеми матричного помножувача та значення контролепридатності, що одержані по завершенні експериментів.

Показується також матриця операційних елементів матричного помножувача з оглядом для кожного експерименту кількості потенційно загрозливих точок, що в них містяться.

Кольором виділяються операційні елементи, що мають потенційно загрозливі точки. В даному випадку виділено 29 операційних елементів з 56 за результатами проведення другого експерименту.

При необхідності структура операційного елемента може бути розкрита з оглядом для кожної точки значень керованості та спостережуваності в нормальному та аварійному режимах.

Потенційно загрозливі точки в схемі операційного елемента матриці помножувача виділяються кольором.

Проведене моделювання мало за мету оцінити рівень контролепридатності та кількості потенційно загрозливих точок матричного помножувача у складі системи критичного застосування та вплив на нього обмеженості вхідних даних нормальному та аварійному режимах.

Результати серії експериментів з оцінки кількості потенційно загрозливих точок цифрової схеми матричного помножувача, що показані на рис. 1, наведені в табл. 1.2.

Таблиця 1.2 – Результати серії експериментів для розрядності $n = 8$

HE	1	2	3	4	5	6	7	8
ОД	10	20	30	40	50	60	70	80
ЗТ	97	71	62	36	33	31	2	0
КП, %	65,3	74,6	77,8	87,1	88,2	88,9	99,3	100

В рядках таблиці показано номери експериментів (HE), обсяги діапазону зміни співмножників у нормальному режимі (ОД), кількість потенційно загрозливих точок (ЗТ) та контролепридатність матричного помножувача (КП), що наведена в процентах.

Результати моделювання показують, що 100% контролепридатності матричного помножувача досягається лише при обсязі діапазону в 80 значень співмножників. Це складає 80^2 вхідних слів з 128^2 можливих, тобто 39% від загальної кількості вхідних слів.

При обсязі діапазону в 10 значень співмножників та відповідно 100 вхідних слів, цифрова схема матричного помножувача мантис містить 97 потенційно загрозливих точок, яким відповідає контролепридатність 65,3%.

Результати лабораторних досліджень показали низьку контролепридатність матричного помножувача мантис та значну кількість потенційно загрозливих точок, в яких протягом довготривалого аварійного режиму можуть накопичуватися приховані несправності у множинній кількості та викликати кратні відмови з початком аварійного режиму системи критичного застосування.

Тому проблема прихованих несправностей перетікає в проблему кратних відмов, що може суттєво зменшити або повністю перекрити відмовостійкість цифрових компонентів.

Таким чином, стає необхідним досліджувати цифрові компоненти та їх системи робочого діагностування в умовах критичного застосування та прояву кратних відмов, що складають певну загрозу функціональній безпеці систем та об'єктів управління у найбільш відповідальному аварійному режимі.

1.3 Контроль цифрових компонентів за числовим модулем

1.3.1 Методи та засоби робочого діагностування спрямовані на виявлення помилок, що викликаються типовими відмовами обчислювальних пристроїв у процесі виконання основних операцій над фактичними даними та спрямовані на оцінювання достовірності результатів з метою зберігання достовірних і відбраковування недостовірних [8].

Традиційні методи робочого діагностування були розвинені для обробки точних даних, до яких належать числа, що є цілими за своєю природою, тобто номери елементів множин.

Інші дані, що складають основну більшість, належать до наближених чисел, які описують результати вимірів та їхньої обробки.

Наближені дані представляються мантисами чисел у форматах із плаваючою точкою. Їхньою особливістю як наближеного числа є наявність похибки, що розділяє розряди мантиси на старші вірні та молодші невірні, тобто створює певну структуру наближеного даного.

Точне дане складається тільки з вірних розрядів. Їхнє викривлення видмовами завжди приводить до помилки, що є суттєвою для достовірності результату.

Помилки в невірних розрядах мантиси є несуттєвими для достовірності наближених результатів [9].

Робоче діагностування результатів, що одержуються при виконанні наближених обчислень, повинно виявляти суттєві помилки та ігнорувати несуттєві, що викликаються в невірних розрядах та виключаються з обчислень при відкиданні молодших розрядів добутку.

У скороченому множенні відкидається в багато разів менше розрядів, чим у повній операції. Це знижує частку викликаних несправностями несуттєвих помилок, виявлення яких призводить до відбраковування помилкових, але достовірних результатів [10].

За теорією чисел, будь-яке ціле число можна представити за наступною формою:

$$A = M d + r,$$

де M, d, r , – цілі числа;

M – модуль;

$d = E(A / M)$, тобто найменше ціле дробу A / M ;

r_A – остача числа A за модулем M , $r = 0, 1, \dots, M - 1$

Відповідність між числом A та його остачею r за модулем M записується у вигляді $r = A \bmod M$ [11].

1.3.2 Вся множина цілих чисел розбивається за модулем M на M класів еквівалентності. До кожного із цих класів належать числа з однаковим значенням остачі. Такі числа називаються порівнянними за модулем M .

Наприклад, для $M = 3$ множина цілих чисел розбивається на три класи еквівалентності, що поєднують порівнянні між собою числа із залишками, відповідно $0, 1$ та 2 .

У контролі за числовим модулем остача використовується за контрольний код KA числа A .

Для коду KA , як і до обернених кодів, визначаються два подання нуля: $+0$ та -0 , відповідно 0 і M .

Помилки обчислень виявляються за умови віднесення правильного та викривленого результатів до різних класів еквівалентності.

Арифметичні операції додавання та множення чисел перевіряються за наступними контрольними співвідношеннями: $KA + KB = KS$ та $KA \cdot KB = KC$ для суми $S = A + B$ та добутку $C = A \cdot B$, де $KA = A \bmod M$, $KB = B \bmod M$ – контрольні коди за числовим модулем операндів та контрольні коди за числовим модулем результатів: $KS = S \bmod M$, $KC = C \bmod M$.

Перевірка правильності обчислення арифметичних формул, складених з операцій додавання та множення, виконується по контрольних співвідношеннях, які виходять із формул, що складаються з цих формул, заміною операндів на їхні контрольні коди.

Контроль арифметичних операцій виконується найбільше ефективно за модулем виду $M = N - 1$, де N – основа системи числення.

Число ділиться на такий модуль із тією ж остачею, що й сума цифр числа. Тому контрольний код числа обчислюється сумою цифр, що обчислюється до одержання однієї цифри. Така операція називається згорткою за модулем.

Схема контролю містить :

- формувачі контрольного коду ΦK_A , ΦK_B та ΦK_R , що обчислюють по операндах $A\{1 \div n\}$, $B\{1 \div n\}$ і результату $R\{1 \div n_R\}$ їх контрольні коди $KA\{1 \div n\}$, $KB\{1 \div n\}$ та $KR\{1 \div n_R\}$, відповідно, де n_R – розрядність результату;
- контрольні коди операндів KA і KB , що супроводжують операнди $A\{1 \div n\}$ та $B\{1 \div n\}$ з попереднього каскаду системи контролю;
- вузли порівняння $ВП_A$ та $ВП_B$, що порівнюють вхідні та обчислені контрольні коди операндів та визначають їх дворозрядні коди контролю K_A і K_B ;
- контрольний код результату для супроводження результату $R\{1 \div n_R\}$ у наступний каскад системи контролю;
- контрольний блок $КБ$, що виконує операцію над контрольними кодами операндів $KA\{1 \div n\}$, $KB\{1 \div n\}$ та визначає контрольний код KR результату $R\{1 \div n_R\}$;
- вузол порівняння $ВП_R$, який порівнює контрольні коди $KR\{1 \div n_R\}$ та KR результату $R\{1 \div n_R\}$ та визначає код контролю K_R результату.

Схема відносить формувачі контрольного коду ΦK_A , ΦK_B та ΦK_R і зв'язані з ними вузли порівняння $ВП_A$, $ВП_B$ та $ВП_R$ до блоків контролю $БК_A$, $БК_B$ та $БК_R$ операндів і результату.

Найбільше поширення одержав контроль за модулем три, що працює в системі числення з основою 4.

1.4 Скорочене множення та його контроль за модулем

1.4.1 За методом скороченого множення матриця кон'юнкцій добутку (МКД) розбивається на дві частини: молодшу й старшу.

Молодша частина, що складається з k молодших стовпців, виключається з обчислень.

По старшій частині обчислюється код $C\{1 \div 2n - k\}$ усічені добутки [12].

Молодші розряди $C_{AT} = C\{n + 1 \div 2n - k\} \cdot 2^{-(2n-k)}$ усіченого добутку відкидаються: $C_R = C_T - C_{AT}$.

Величина $k = n - \lceil \log_2 n \rceil$ визначається за умови збереження одинарної точності округленого добутку.

1.4.2 На рис. 1.3 показана розбивка МКД на молодшу та старшу частини для $n = 8$ і $k = 5$, а також двійкові коди $C\{1 \div 2n\}$, $C\{1 \div 2n-k\}$ і $C\{1 \div n\}$ повного C_F усіченого C_T і округленого C_R добутків.

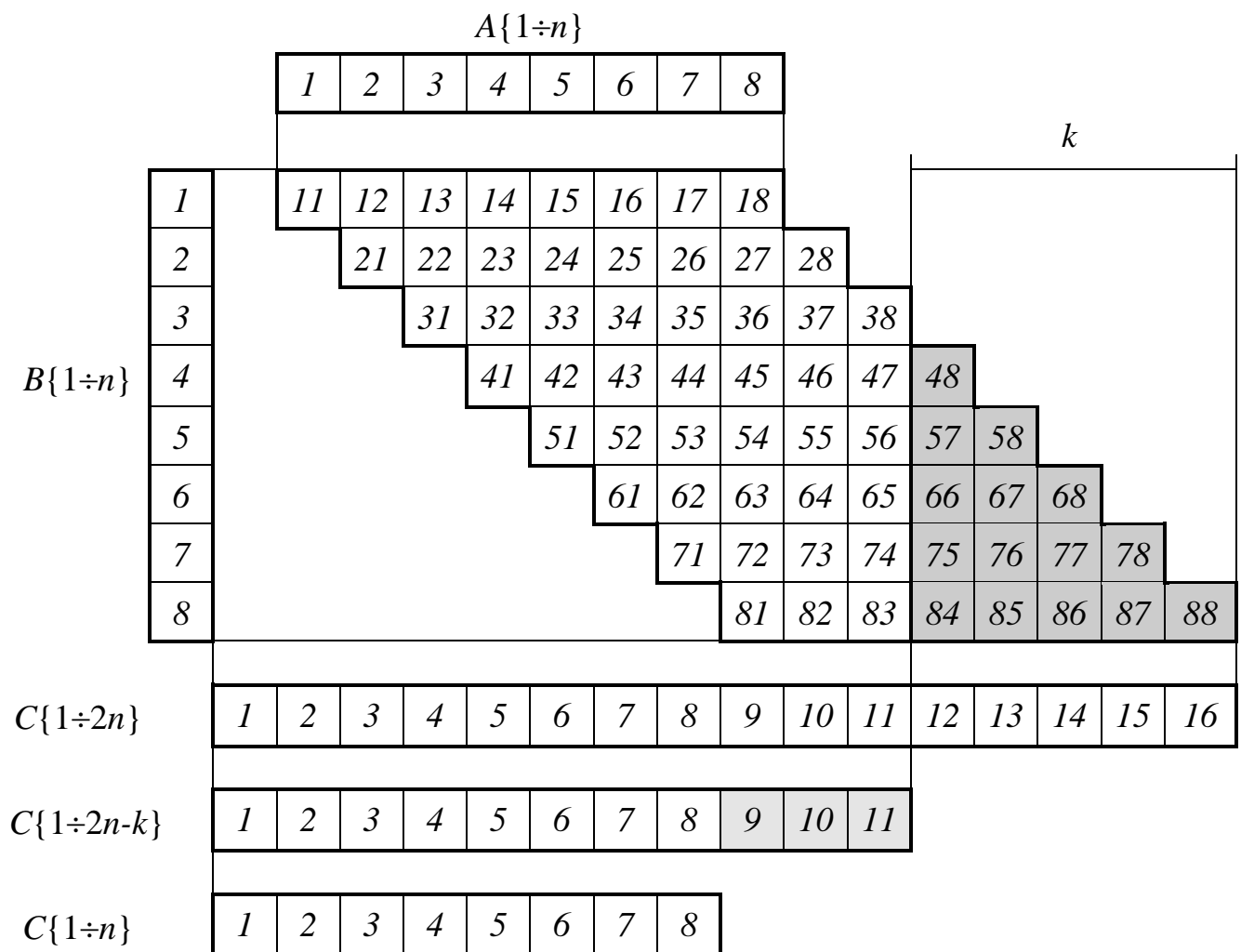


Рисунок 1.3 – Розбивка МКД на молодшу й старшу частини

Кон'юнкції МКД позначені дворозрядними кодами, складеними з номерів утворюючих їхніх розрядів множеного й множника.

Метод скороченого множення знижує кількість устаткування матричного помножувача мантис та підвищує його швидкодію.

Кількість устаткування в основному визначається складністю реалізації схеми додавання кон'юнкцій.

Вона являє собою схему стиску інформації та зручно оцінюється щодо складності реалізації q повного двійкового суматора по формулі

$$O = (n_{BX} - n_{ВИХ}) q_C,$$

де n_{BX} і $n_{ВИХ}$ – розрядності вхідного та вихідного кодів схеми, тобто кількість оброблюваних кон'юнкцій МКД і розрядність добутку, що обчислюється, відповідно;

q_C – ціна стиску даних на один біт, для схеми додавання $q_C = q$;

q – кількість устаткування повного двійкового суматора.

1.4.3 У повнорозрядному помножувачі розрядності вхідного та вихідного кодів схеми $n_{BX} = n^2$ та $n_{ВИХ} = 2n$, що визначає кількість його устаткування як

$$O_{2n} = (n^2 - 2n) q.$$

У помножувачі зі скороченим множенням розрядності вхідного та вихідного кодів схеми становлять

$$n_{BX} = n^2 - k(k + 1) / 2;$$

$$n_{ВИХ} = 2n - k,$$

а кількість його устаткування визначається за формулою:

$$O_{2n-k} = (n^2 - 2n - k(k - 1) / 2) q \approx (n^2 - k^2/2) q.$$

Таким чином, скорочення обчислень забезпечує зниження кількості устаткування в

$$\sigma O_{MV} \approx 1 / (1 - 0,5 k^2 / n^2)$$

разів, що із зростанням n наближається до двох.

Витрати часу на виконання операції в матричному помножувачі оцінюються, приймаючи, що тривалість операції множення пропорційна розрядності добутку, що обчислюється.

Скорочення обчислень зменшує розрядність добутку з $2n$ до $2n - k$ розрядів, тобто витрати часу знижуються в

$$T_{MV} = 2n / (2n - k)$$

разів, що із зростанням n також наближається до двох.

1.4.4 Контроль за модулем скороченого множення ґрунтується на розбивці старшої частини МКД на фрагменти.

Фрагмент C_i – частина МКД, що визначається добутком [13]

$$C_i = \text{sign}C_i A_i B_i,$$

де $\text{sign}C_i$ – знак фрагмента C_i , $\text{sign}C_i = +1$, якщо більшість кон'юнкцій розташована в старшій частині МКД, і $\text{sign}C_i = -1$ у протилежному випадку; A_i та B_i – множене та множник або їхні частини.

Старша частина МКД складається з $k + 1$ фрагментів, а контрольний код усіченого добутку із суми контрольних кодів фрагментів.

Контрольний код округленого добутку визначається різницею контрольного коду усіченого добутку та контрольного коду $C\{n + 1 \div 2n - k\}$ розрядів, що відкидаються.

Структурна схема контролю за числовим модулем, що виконує описані обчислення, показана на рис. 1.4

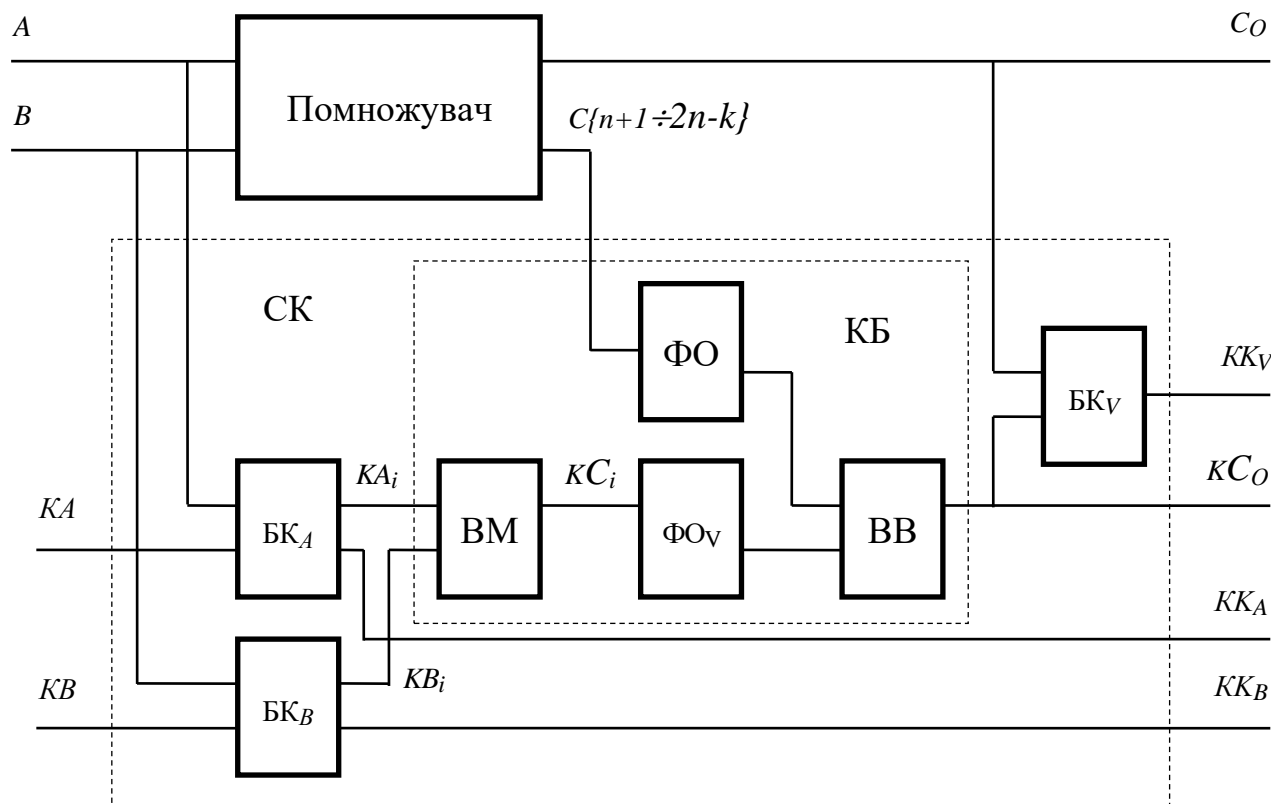


Рисунок 1.4 – Схема контролю матричного помножувача за числовим модулем

Схема контролю помножувача за модулем працює наступним чином.

Операнди A , B та їхні вхідні контрольні коди KA , KB надходять на відповідні блоки контролю BK_A , BK_B , які перевіряють операнди та їхні контрольні коди на порівнянність за модулем й обчислюють коди контролю операндів KK_A та KK_B .

Схема виконує згортку за модулем операндів та обчислює контрольні коди KA_i і KB_i частин A_i і B_i , що мають довжину більше L біт. Інші контрольні коди KA_i і KB_i складаються з розрядів операндів A і B .

Контрольні коди KA_i та KB_i частин A_i та B_i , подаються на входи контрольного блоку КБ, що обчислює контрольний код KC_O округленого результату C_O .

Контрольний блок КБ містить вузол множення ВМ, формувачі остачі ФО, FO_V і вузол вирахування ВВ.

Вузол ВМ обчислює контрольні коди фрагментів

$$KC_i = \text{sign}C_i \cdot KA_i \cdot KB_i.$$

Формувач ФО приймає з виходів схеми матричного помножувача розряди $C\{n + 1 \div 2n - k\}$ усіченого добутку C_Y , що відкидаються, і визначає їх контрольний код $KC\{n + 1 \div 2n - k\}$.

Формувач ФО складає контрольні коди фрагментів і обчислює контрольний код KC_Y усіченого добутку C_Y .

Вузол ВВ визначає контрольний код KC_O (за різницею контрольних кодів KC_Y і $KC\{n + 1 \div 2n - k\}$) для супроводу результату C_O в схеми контролю наступних арифметичних пристроїв.

Результат C_O та обчислений для нього контрольний код KC_O надходять на вихід схеми контролю матричного помножувача за числовим модулем, а також подаються на входи блоку контролю БК, що перевіряє їх на порівнянність за числовим модулем та формує код контролю KK результату, що далі його супроводжує.

1.5 Контролю за модулем множення в доповняльному коді

1.5.1 Обчислення контрольного коду мантиси, що подана у доповняльному коді, враховує те, що остача за модулем має природу обернених кодів, включаючи два представлення нуля.

Для обернених кодів остача за модулем виду $2^L - 1$ формується додаванням цифр системи числення з основою 2^L , включаючи знакові розряди, якими доповнюються неповні двійкові коди цифр.

Доповняльний код обчислюється з оберненого коду додаванням знакового розряду з одиничною вагою. Це не змінює обернений код для додатних чисел та збільшує його на одиницю у разі від'ємних чисел.

Тому контрольний код мантиси, що подана у доповняльному коді, обчислюється за правилом формування контрольного коду з оберненого коду мантиси, але за винятком знакового розряду [14].

1.5.2 Доповняльний код добутку мантис обчислюється з введенням корекції по формулі:

$$C_D = A_D \cdot B_D + \Delta C,$$

де A_D, B_D та C_D – доповняльні коди без знаків мантис співмножників A, B та добутку C ;

ΔC – величина корекції, що визначається з урахуванням знаків S_A та S_B співмножників як

$$K = ((-A_D)_D \cdot S_B + (-B_D)_D \cdot S_A)_D \cdot 2^n.$$

Контрольний код добутку обчислюється як добуток контрольних кодів співмножників без корекцій.

1.6 Достовірність контролю за модулем добутків мантис

1.6.1 При оцінці достовірності контролю необхідно приймати до уваги, що добутки мантис також представлені мантисами, як і операнди належать до наближених даних, в яких відмови цифрових схем викликають суттєві та

несуттєві помилки для достовірності результату відповідно у вірних та невідних розрядах [15, 16].

Разом з тим, методи робочого діагностування є достовірними настільки, наскільки вони визначають достовірний результат достовірним, а недостовірний результат – недостовірним.

Показник достовірності D контролю результатів визначається по наступній формулі:

$$D = P_{BC} + P_{PH},$$

де P_{BC} – ймовірність виявлення суттєвої помилки,

P_{PH} – ймовірність пропуску несуттєвої помилки результату

У випадку виявлення суттєвої та пропуску несуттєвої помилки метод контролю правильно визначає достовірний результат.

Ймовірності ймовірність виявлення суттєвої помилки P_{BC} та пропуску несуттєвої помилки P_{PH} визначаються по наступних формулах:

$$P_{BC} = P_B P_C;$$

$$P_{PH} = (1 - P_B) (1 - P_C),$$

де P_B – ймовірність виявлення помилки;

P_C – ймовірність суттєвої помилки.

1.6.2 Ймовірність P_B оцінюється відношенням кількості вхідних слів, що виявляють помилку, до загальної кількості вхідних слів.

Ймовірність P_C визначається за її доповненням до одиниці з урахуванням ряду факторів, які скорочують частку суттєвих помилок, а саме [17]:

– виключення несуттєвих помилок при відкиданні молодших невірних розрядів обчислюваного результату;

– збільшення кількості несуттєвих помилок, що відбувається при подовженні мантиси в розширених форматах даних за рахунок додавання молодших невірних розрядів;

– виключення несуттєвих помилок при втраті вірних розрядів у результатах операцій, які передують зсуву мантиси при вирівнюванні порядків із плаваючою точкою, та наступних операцій після нормалізації результату.

Врахування цих факторів, відповідно P_1 , P_2 та P_3 , дозволяє оцінити ймовірність суттєвої помилки за формулою:

$$P_C = P_1 P_2 P_3 \quad (1.1)$$

Ймовірність P_1 визначається при аналізі двох протилежних подій: поява помилки в залишених або відкинутих розрядах результату.

Ймовірності цих подій P_1 і $\bar{P}_1 = 1 - P_1$ визначаються часткою помилок у розрядах, що залишаються і відкидаються.

Для матричного помножувача ймовірності P_1 , P_2 та P_3 оцінюються, виходячи із припущення про рівномірний розподіл відмов по апаратній реалізації цифрової схеми.

Ймовірність P_1 обчислюється по формулі

$$P_1 = C_3 / C_{ОБЧ}, \quad P_1 \leq 1, \quad (1.2)$$

де C_3 – складність обладнання, відмови якого викликають арифметичні помилки в залишених розрядах результату;

C_4 – складність обладнання, відмови якого викликають арифметичні помилки у всіх обчислюваних розрядах результату.

Ймовірність P_2 визначається, приймаючи рівномірний розподіл відмов по наступній формулі

$$P_2 = C_{ТВ} / C_{ОСТ}, \quad P_2 \leq 1,$$

де C_B – складність обладнання, відмови якого викликають арифметичні помилки у вірних розрядах результату.

Можливості персональних ЕОМ по збільшенню розрядності мантис в 2,8 разів (з 24 в одинарному форматі до 64 у подвійному розширеному форматі), що закладені у форматах чисел із плаваючою точкою, визначають ймовірність $P_2 \leq 0,36$.

Ймовірність P_3 оцінюється за формулою

$$P_3 = 1 - 0,5 \delta C_C d / n,$$

де $\delta C_C = C_C / C_O$;

d – величина зсуву мантиси;

C_C – складність обладнання вузлів ОП, що передують даному зсувачу мантис; C_O – загальна складність обладнання ОП.

Середнє значення величини зсуву становить половину розрядності мантиси, тобто $d = n / 2$, а формула для ймовірності приймає вигляд $P_3 = 1 - 0,25 \delta C_C$:

Для декількох операцій вирівнювання порядків, виконуваних незалежно в даному ОП, ймовірність P_3 визначається як добуток ймовірностей, знайдених для кожної із цих операцій.

Наведені оцінки ймовірностей пропуску та виявлення суттєвих і несуттєвих помилок не враховують характер несправностей цифрових схем, що занижує їхню достовірність.

Для уточнення цих оцінок, а також показників достовірності контролю необхідно виконати моделювання обчислювальних процесів, що протікають під впливом типових відмов.

Особливості використання цифрових компонентів у складі систем критичного застосування потребують враховувати в моделюванні можливу появу кратних відмов в аварійному режимі.

1.7 Висновки

Арифметичні цифрові компоненти, що працюють у складі інформаційно-управляючих систем критичного застосування в двох режимах: нормальному та аварійному, як за правило, підтримують в цих режимах функціонування за різними вхідними даними.

Це разом з урахуванням також значної структурної надмірності внаслідок побудування цифрових схем відмовостійкими, призводить до низької контролепридатності. За тим суттєво обмежуються можливості робочого діагностування.

Внаслідок системи критичного застосування одержують проблему кратних прихованих відмов.

За такими обставинами в схемі цифрового компонента виникають потенційно загрозливі точки, в яких протягом довготривалого аварійного режиму накопичуються приховані несправності, що в аварійному режимі можуть проявлятися разом, викликаючи кратні відмови та суттєво знижуючи відмовостійкість і безпеку системи критичного застосування та об'єкту підвищеного ризику.

Проблема прихованих кратних відмов потребує проведення досліджень робочого діагностування цифрових компонентів в умовах дії кратних відмов аварійного режиму.

Цифровий компонент як об'єкт діагностування варто розглянути, приймаючи до уваги його належність до оброблювачів даних, найбільш характерних для сучасних технологій.

При цьому слід зауважити, що більшість чисел є наближеними даними і їхня частка та значущість в комп'ютерній обробці постійно росте.

Ключовою операцією в обробці наближених даних є множення, оскільки вона присутня у всіх операціях із плаваючою точкою в частині мантис та, як правило, виконується разом з іншими арифметичними операціями у доповняльному коді.

При підвищених вимогах, що пред'являються до продуктивності сучасних обчислювальних систем дана операція імплементується в структури однокітних матричних пристроїв.

Обробка мантис в однокітному матричному помножувачі найбільш ефективно здійснюється за методом скороченого множення, що виконається над двійковими кодами мантис.

Однак виконання скороченого множення під дією кратних відмов недостатньо вивчено, що обмежує використання методу в областях критичного застосування.

Для результатів, що обчислюються в інформаційно-управляючих системах критичного застосування, основною вимогою до цифрових компонентів є достовірність одержуваних результатів та достовірність робочого діагностування, що потребує досліджувати ймовірності виявлення та пропуску суттєвих і несуттєвих помилок.

Сучасні системи моделювання не забезпечують оцінки достовірності робочого діагностування при несправній роботі цифрового компонента і не розглядають проблему прихованих відмов та їх кратного прояву в критичних застосуваннях.

Вирішення цієї задачі вимагає розробки спеціальної програми моделювання обчислювального процесу, що протікає в однокітних матричних помножувачах мантис з повним та скороченим виконанням операції під дією проявлених прихованих кратних відмов.

2 ЗАВДАННЯ НА ПРОВЕДЕННЯ ДОСЛІДЖЕНЬ

2.1 Найменування та область застосування

2.1.1 Тема магістерської роботи “Дослідження достовірності контролю помножувача за проявом кратних відмов” спрямована на проведення досліджень з робочого діагностування матричного помножувача мантис у доповняльному коді, що виконується під дією кратних несправностей в аварійному режимі роботи системи критичного застосування.

2.1.2 Областю застосування розробки є системи критичного застосування, до яких пред'являються високі вимоги з функціональної безпеки та достовірності одержуваних результатів.

Об'єктом робочого діагностування служить матричний помножувач мантис для обробки мантис чисел із плаваючою точкою у доповняльному коді у складі систем критичного застосування, що спрямовані на забезпечення функціональної безпеки об'єктів підвищеного ризику: космічної та оборонної техніки, енергоблоків електростанцій та енергомереж, швидкісного транспорту та літальних апаратів, тощо. Системи критичного застосування повинні запобігати техногенним аваріям та зменшувати їхні наслідки у випадку здійснення аварії.

Предметом досліджень є достовірність контролю за модулем результатів обробки наближених даних в матричному помножувачі мантис у доповняльному коді під дією кратних відмов, що можуть статися в аварійному режимі роботи системи критичного застосування внаслідок проблеми прихованих відмов. Ця проблема виникає, коли протягом довготривалого нормального режиму роботи в цифрових компонентах системи накопичуються приховані несправності, які можуть разом проявитися на початку аварійного режиму.

2.2 Призначення досліджень

2.2.1 Дослідження магістерської роботи призначені для оцінки достовірності контролю результатів за методами та засобами робочого діагностування в умовах дії кратних несправностей на початку аварійного режиму роботи системи критичного забезпечення.

Контроль результатів виконується за числовим модулем три в матричному помножувачі, що обробляє мантиси у доповняльному коді на матриці операційних елементів протягом одного такту.

2.2.2 Для матричного помножувача мантис аналізуються несправності типу «замикання» з кратністю до трьох, що відбуваються в різних операційних елементах між точками схеми кожного з них.

Аналізується ймовірність виявлення та пропуску суттєвих та несуттєвих помилок, що породжуються характерними несправностями матричного помножувача мантис у додавальному коді при повному та скороченому виконанні операції.

Досліджується достовірність контролю за числовим модулем три результатів, що обчислюються у доповняльному коді на несправному матричному помножувачі двійкових кодів нормалізованих мантис з різною кратністю несправності при повному та скороченому виконанні операції.

2.3 Підготовка та проведення досліджень

2.3.1 Організаційний план досліджень включає їхню підготовку та проведення. На етапі підготовки вивчається науково-технічна література за темою магістерської роботи, уточняється об'єкт, предмет та мета досліджень, визначаються методи та засоби досліджень, а також очікувані результати досліджень – їхня форма представлення та зміст. Крім того, прогнозуються елементи наукової новизни та практична цінність результатів досліджень.

Проведення досліджень потребує розробки оригінального програмного продукту, що забезпечує моделювання роботи матричного помножувача мантис у доповняльному коді для варіантів повного та скороченого виконання операції, а також його контролю за числовим модулем три в умовах дії кратних несправностей, характерних для матричних пристроїв у складі систем критичного застосування.

Необхідно розробити низку режимів початкових установок та моделювання, що забезпечать верифікацію програми та одержання достовірних результатів моделювання, а саме: режимів завдання значень співмножників, місця та виду несправності вірних та невірних розрядів результату, тощо.

Результати моделювання необхідно одержувати на випадкових послідовностях значень операндів – двійкових нормалізованих мантис, що подані у доповняльному коді на входи матричного помножувача. Експеримент продовжується до одержання сталих значень результатів моделювання.

Вид несправності типу «замикання» задається довільно в межах окремого операційного елемента матриці помножувача. Несправні операційні елементи під час моделювання визначаються випадковим чином.

Несправність типу «замикання» слід задавати в межах одного операційного елемента між точками його схеми, включаючи входи та виходи елементів I та повних двійкових суматорів.

Суттєвість помилок результатів для їх достовірності слід задавати розподілом розрядів результатів на старші вірні та молодші невірні, вага яких є меншою за погрішність представлення або обчислення мантис.

2.4 Постановка задачі

Ставиться задача на проведення досліджень достовірності системи контролю матричного помножувача мантис у доповняльному коді, що виконує

повну або скорочену операцію в умовах дії кратних відмов у складі системи критичного застосування.

Контроль результатів виконується за числовим модулем з урахуванням наближеного характеру даних, що обробляються та обчислюються.

За об'єкт досліджень слід вибрати процес контролю наближених результатів на виходах матричного помножувача мантис у доповняльному коді при кратних відмовах, а предметом дослідження має бути достовірність контролю наближених добутоків в умовах дії кратних відмов під час аварійного режиму системи критичного застосування.

Дослідження необхідно спрямувати на оцінку достовірності контролю за модулем три наближених добутоків на виходах матричного помножувача мантис у доповняльному коді під дією кратних відмов, що проявляються в аварійному режимі системи критичного застосування.

Дослідження доцільно виконати на програмній моделі матричного помножувача мантис з контролем за числовим модулем три.

Програмна модель повинна відповідати вимогам функціональності та достовірності, необхідних для проведення достовірних досліджень.

Функціональність необхідно забезпечити відповідно до задач дослідження стосовно реалізації програмної моделі у відтворенні режимів роботи та функцій матричного помножувача та його контролю.

До режимів роботи та функцій матричного помножувача мантис необхідно залучити наступне:

- вибір розрядності матричного помножувача у діапазоні від восьми до п'ятнадцяти розрядів;
- завдання операндів: конкретних та випадкових значень для налагодження програмного продукту моделі, а також повного перебору значень та утворення їх випадкової послідовності безпосередньо для одержання результатів досліджень;
- завдання несправностей типу замикання двох точок в схемі операційного елемента водночас для кратності несправності 1, 2 та 3: з

визначенням конкретного місця та виду відмови, тобто конкретних операційних елементів та точок, між якими відбувається замикання, для налагодження програмного продукту моделі, а також випадкового визначення місць та виду відмови безпосередньо для одержання результатів досліджень;

- завдання розподілу розрядів результату на вірні та невірні для визначення помилок як суттєвих та несуттєвих для достовірності обчислюваних результатів множення мантис;

- відтворення функцій множення мантис із повним та скороченим виконанням операції з обчисленням повного та зсіченого добутків та їх одержання за результатами моделювання при відсутності та наявності кратних відмов.

Функціональність контролю необхідно забезпечити у наступному:

- формування контрольних кодів операндів та результатів і перевірка контрольних співвідношень між ними, що визначають правильність обчислення повного та зсіченого добутків;

- виявлення та підрахунок кількості та процентного складу виявлених та пропущених суттєвих та несуттєвих помилок, а також їх загальної кількості у повному обсязі виконаних операцій.

Достовірність програмної моделі необхідно довести верифікацією її функціонування у всіх режимах, включаючи виконання множення на конкретних та випадкових значеннях і їх послідовностях, у відсутності відмов та при їх наявності, забезпеченої за конкретним та довільним місцем та видом, для різного розподілу розрядів на вірні та невірні і повного та скороченого виконання операцій.

Результати досліджень слід відобразити залежностями виявлення та пропуску суттєвих та несуттєвих помилок, а також достовірності контролю для кратних несправностей у реалізаціях повної та скороченої операцій від розрядності матричного помножувача мантис.

3 ПРОГРАМНА МОДЕЛЬ ДЛЯ ДОСЛІДЖЕННЯ СИСТЕМИ КОНТРОЛЮ ПОМНОЖУВАЧА МАНТИС

3.1 Структурне побудування програмної моделі

3.1.1 Програмну модель матричного помножувача мантис та його системи контролю за модулем доцільно створювати у середовищі Delphi.

Програмна модель повинна забезпечувати процес підготовки, проведення та одержання і висвітлення результатів моделювання матричного помножувача мантис та його контролю за модулем три у доповняльному кодї за допомогою елементів керування та демонстраційних елементів [18].

Склад елементів керування визначається функціональністю моделі та включає:

- меню, що визначає режими завдання операндів та справний або несправний стан матричного помножувача мантис;
- елементи, що задають вид та місце конкретної видмови типу замикання двох точок операційного елемента матричного помножувача мантис;
- покажчик розподілу та перерозподілу розрядів результатів на вірні та невірні розряди;
- елементи, що показують структурні схеми матричного помножувача мантис та його контролю за модулем з можливістю їх деталізації та виведенням значень сигналів.

Демонстраційні елементи показують конкретні значення вихідних та поточних даних, а також поточні та остаточні результати моделювання, до яких належать такі:

- вибрані режими завдання операндів та справного або несправного стану матричного помножувача мантис, що відмічаються у пунктах меню;

- вихідні або поточні значення операндів та обчислених і одержаних при моделюванні повних і усічених добутків, представлених у десяткових та двійкових кодах;
- вихідний або поточний номер несправного елемента та конкретний вид несправності замикання двох вказаних точок операційного елемента;
- розподіл розрядів повних та усічених добутків, що обчислені та одержані в процесі моделювання, на вірні та невірні;
- структурна схема контролю з показом поточних значень контрольних кодів та кодів контролю операндів і результатів;
- таблиця результатів моделювання, що містить ймовірності появи, виявлення та пропуску суттєвих, несуттєвих та всіх помилок для повного та скороченого виконання операції множення мантис у доповняльному коді та трьох значень 1, 2 та 3 кратності несправності.

3.2 Режими функціонування програмної моделі

3,2,1 Програмна модель матричного помножувача мантис та його системи контролю за модулем три має забезпечувати:

- встановлення та підтримку вибраних режимів завдання операндів, стану матричного помножувача, місця та виду відмови, розподілу розрядів результатів обчислень на вірні та невірні;
- показ процесу підготовки та проведення моделювання матричного помножувача та його системи контролю за модулем три;
- показ поточних та остаточних результатів моделювання для декількох значень кратності відмови;
- введення конкретної або довільної несправності замикання у конкретний або довільний операційний елемент матричного помножувача мантис та кратної відмови довільного виду замикання у довільні операційні елементи;

- обчислення функцій матричного помножувача мантис у доповняльному коді для випадків повного та скороченого виконання операції за умови справного стану пристрою та при введенні кратної відмови;
- формування контрольних кодів у доповняльному коді для операндів та результатів та перевірку контрольних співвідношень, що їх зв'язують.
- визначення появи, виявлення та пропуску суттєвих, несуттєвих та всіх помилок для повного та скороченого виконання множення мантис і декількох значень кратності відмови, а також підрахунок ймовірності цих подій протягом процесу моделювання;
- формування таблиці результатів моделювання матричного помножувача мантис та його системи контролю.

3.3 Алгоритми програмної моделі

3.3.1 Для побудови програмної моделі матричного помножувача мантис у доповняльному коді та його системи контролю за числовим модулем три необхідно розробити низку алгоритмів підготовки та проведення моделювання, до основних з яких належать наступні:

- алгоритми введення операндів у двійковій або десятковій системах числення у вигляді конкретних або випадкових значень чи послідовностей відповідно до заданих режимів
- алгоритми обчислення основних функцій матричного помножувача мантис при виконанні повної та скороченої операції одноразово або послідовністю;
- алгоритми введення несправностей замикання двох точок операційного елемента матричного помножувача мантис, як окремих з конкретним або довільним видом та місцем виникнення, так і кратних відмов, випадково розподілених між операційними елементами пристрою з виконанням повної та скороченої операції.

– алгоритми виконання функцій системи контролю для повної та скороченої операції з обчисленням ймовірностей появи, виявлення та пропуску суттєвих, несуттєвих та всіх помилок з урахуванням розподілу розрядів результату на вірні та невірні;

– алгоритми формування та виведення результатів моделювання матричного помножувача мантис у доповняльному коді та системи контролю за числовим модулем три.

3.3.2 Алгоритм введення кратних відмов в модель матричного помножувача представлено нижче.

На рис. 3.1 показана перша частина схеми алгоритму, що містить блоки 1 – 15 та коментарі до них.

Блок 2 визначає вихідні дані для алгоритму: розрядність операнда n , кількість не обчислюваних розрядів усіченого добутку при виконанні скороченого множення мантис, масиви A і B розрядів множеного та множника і масиви M_s і M_r значень суми і перенесення на виходах повних суматорів операційних елементів матричного помножувача.

Блок 3 організовує цикл переліку значень кратності відмови замикання двох точок схем операційних елементів, що вводиться у модель матричного помножувача мантис

Блок 4 організовує цикл з повторенням ітерацій вибору точок схеми операційного елемента матричного помножувача мантис для формування моделі відмови замикання.

Блок 5 за випадковою функцією визначає першу точку схеми операційного елемента у складі матричного помножувача мантис для формування моделі відмови замикання двох точок.

Блок 6 за випадковою функцією визначає другу точку схеми операційного елемента у складі матричного помножувача мантис для формування моделі відмови замикання двох точок. Блок 7 завершує цикл з повторенням ітерацій за умовою неспівпадіння номерів першої та другої точок схеми операційного елемента, що формують модель відмови замикання.

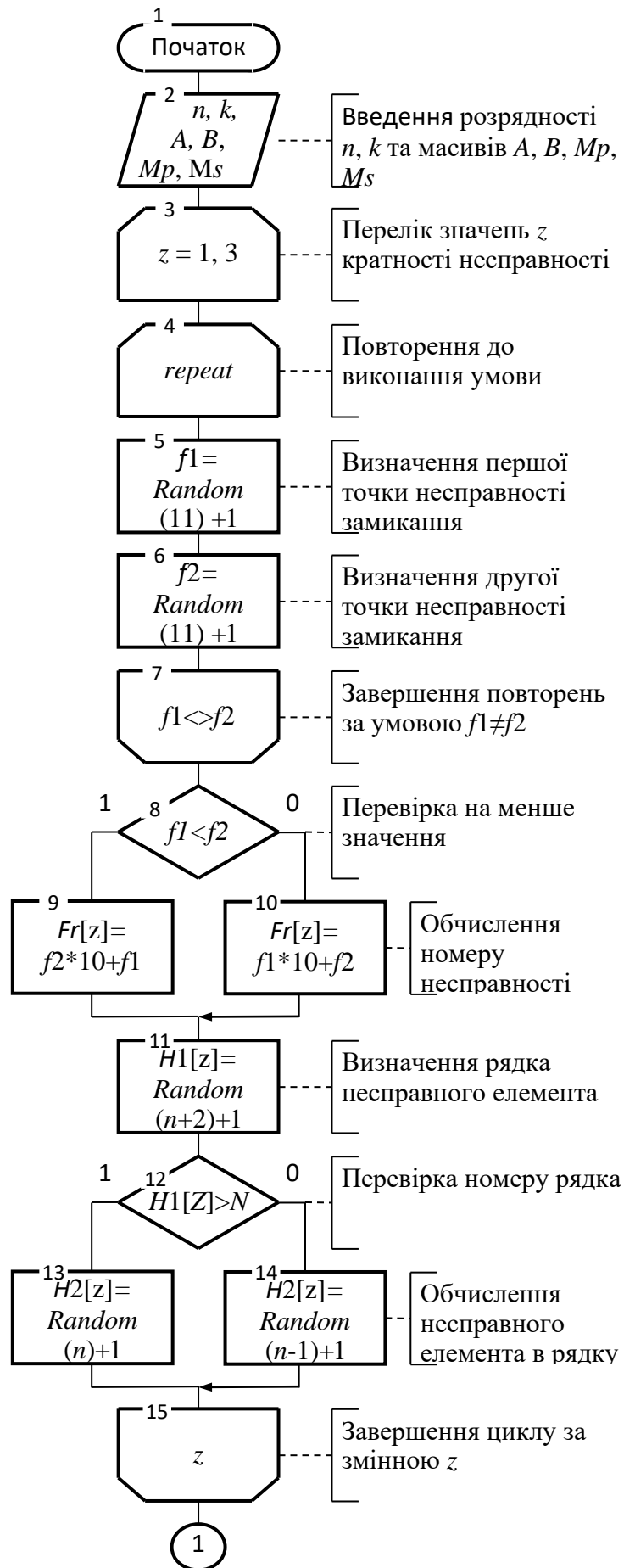


Рисунок 3.1 – Блоки 1 – 15 схеми алгоритму

Блок 8 порівнює номери точок схеми операційного елемента в несправності замикання та переходить до виконання блоку 8 або блоку 9 відповідно при меншому та більшому номері першої точки.

Блок 9 обчислює номер несправності, що складається зліва направо з більшого та меншого номерів точок схеми операційного елемента, вибираючи номери другої та першої точок.

Блок 10 обчислює номер несправності, що складається зліва направо з більшого та меншого номерів точок схеми операційного елемента, вибираючи номери першої та другої точок.

Блок 11 визначає рядок, в якому знаходиться операційний елемент з несправністю замикання двох точок його схеми.

Блок 12 перевіряє номер рядка несправного операційного елемента, чи він належить до перших рядків, що помножують доповняльні коди мантис, чи до двох останніх рядків, в яких вводиться корекція з урахуванням знаків мантис для остаточного формування доповняльного коду добутку, та переходить до виконання блоку 13 або блоку 14 відповідно у першому та другому випадках.

Блоки 13 визначає номер несправного операційного елемента, якщо він знаходиться в перших рядках матричного помножувача мантис.

Блок 14 визначає номер несправного операційного елемента, якщо він знаходиться в двох останніх рядках матричного помножувача мантис.

Блок 15 завершує цикл переліку значень кратності несправності замикання двох точок схем операційних елементів.

3.3.3 На рис. 3.2 показана друга частина схеми алгоритму, що містить блоки 16 – 27 та коментарі до них.

Блок 16 задає режим виконання повної операції в матричному помножувачі мантис.

Блок 17 організує цикл з повторенням ітерацій виконання в матричному помножувачі мантис спочатку повної, а потім скороченої операції.

Блок 18 організує цикл для перебору значень кратності несправності, що вводиться до моделі матричного помножувача мантис.

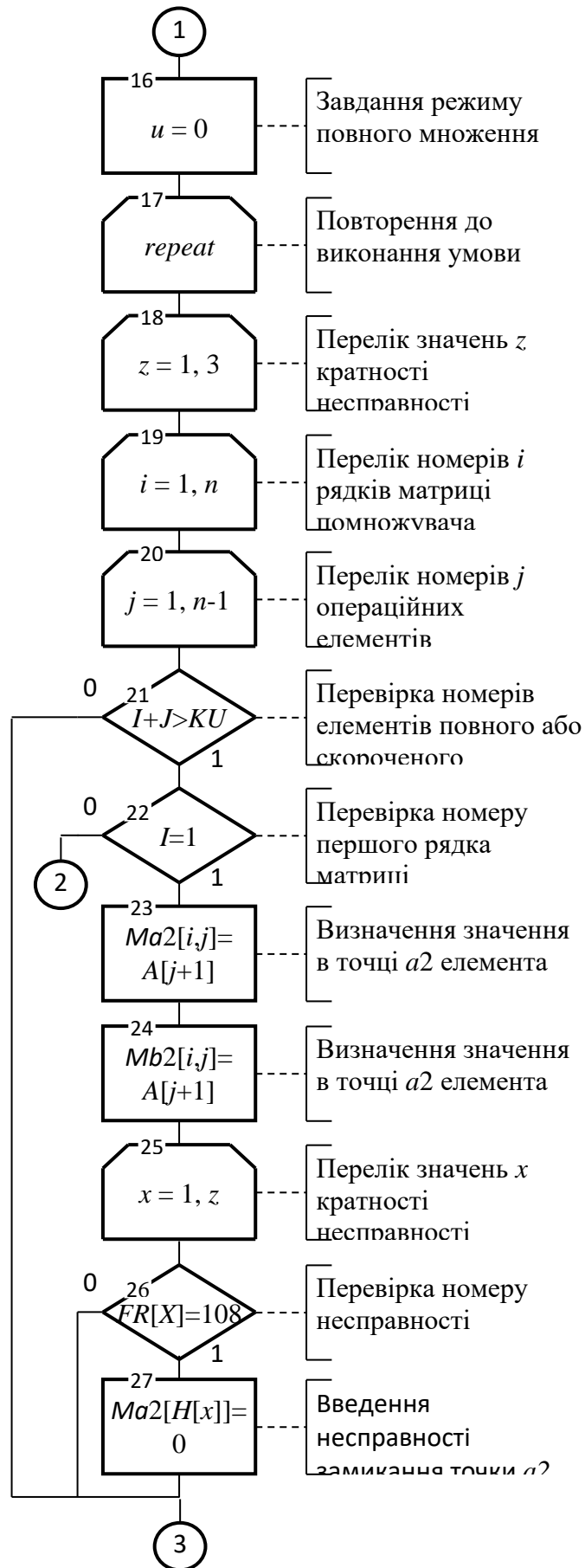


Рисунок 3.2 – Блоки 16 – 27 схеми алгоритму

Блок 19 організує цикл для перебору номерів рядків операційних елементів у складі матричного помножувача.

Блок 20 організує цикл для перебору номерів операційних елементів у складі рядків матричного помножувача.

Блок 21 перевіряє номери операційних елементів для обробки всіх або їх обмеженої множини відповідно при виконанні повної або скороченої операції множення. При невиконанні умови виконується перехід на блок 44.

Блок 22 перевіряє номери операційних елементів на їхню приналежність першому рядку матричного помножувача та переходить до блоку 39, якщо не так.

Блок 23 присвоює сигналу в точці a_2 операційного елемента матричного помножувача мантис значення розряду множеного.

Блок 24 присвоює сигналу в точці b_2 операційного елемента матричного помножувача мантис значення розряду множника.

Блок 25 організує цикл з переліком значень кратності несправності.

Блок 26 перевіряє номер несправності 108 замикання точки a_2 на нуль та переходить на блок 28, якщо це не так.

Блок 27 вводить несправність замикання точки a_2 на нуль для вибраного випадково операційного елемента першого рядка матричного помножувача.

3.3.4 На рис. 3.3 показана друга частина схеми алгоритму, що містить блоки 28 – 38 та коментарі до них.

Блок 28 перевіряє номер несправності 118 замикання точки a_2 на одиничне значення та переходить на блок 30, якщо це не так.

Блок 29 вводить несправність замикання точки a_2 на одиницю для вибраного випадково елемента першого рядка матричного помножувача.

Блок 30 перевіряє номер несправності 109 замикання точки b_2 на нуль та переходить на блок 32, якщо це не так.

Блок 31 вводить несправність замикання точки b_2 на нуль для вибраного випадково операційного елемента першого рядка матричного помножувача.

Блок 32 перевіряє номер несправності 119 замикання точки b_2 на нуль та переходить на блок 34, якщо це не так.

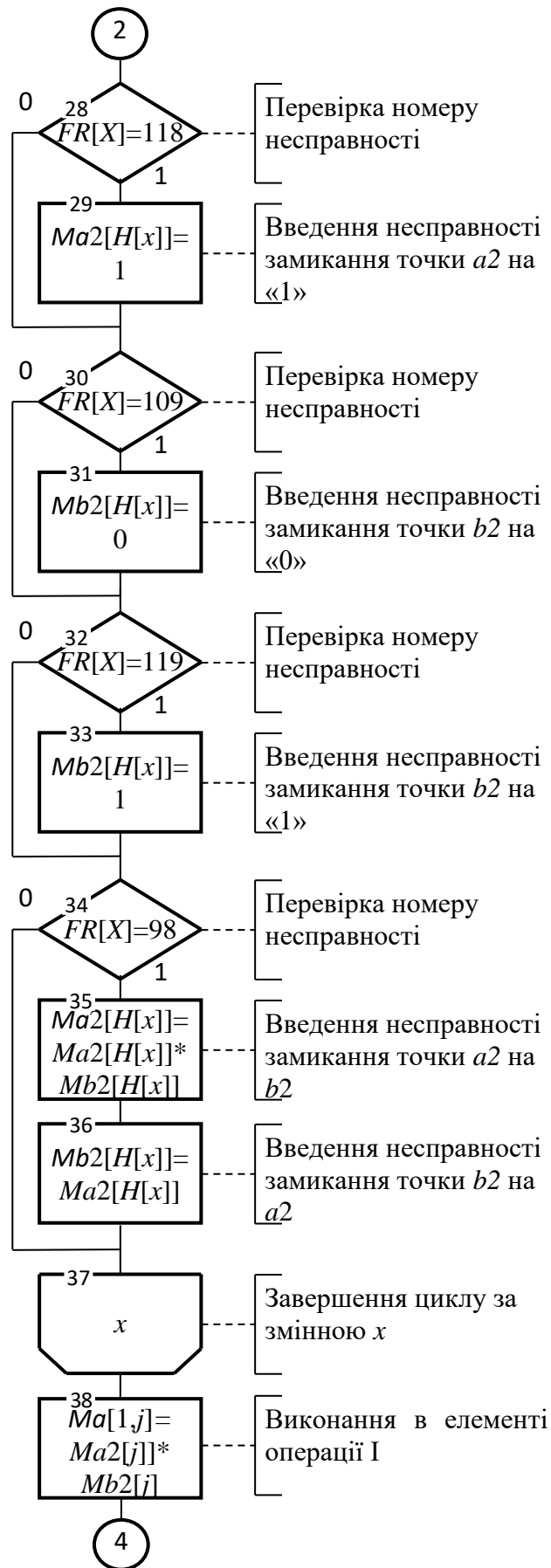


Рисунок 3.3 – Блоки 28 – 38 схеми алгоритму

Блок 33 вводить відмову замикання точки b_2 на одиницю для вибраного випадково операційного елемента першого рядка матричного помножувача.

Блок 34 перевіряє номер несправності 98 замикання точок a_2 та b_2 між собою та переходить на блок 37, якщо це не так.

Блок 35 вводить відмову замикання точки a_2 на b_2 для вибраного випадково операційного елемента першого рядка матричного помножувача.

Блок 36 вводить відмову замикання точки b_2 на a_2 для вибраного випадково операційного елемента першого рядка матричного помножувача.

Блок 37 завершує цикл з переліком значень кратності несправності для їх введення в матричний помножувач мантис.

Блок 38 виконує операцію I над значеннями точок a_2 та b_2 та присвоює одержаний результат значенню точки a операційного елемента першого рядка та переходить до блоку 44.

3.3.5 На рис. 3.4 показана друга частина схеми алгоритму, що містить блоки 39 – 50 та коментарі до них.

Блок 39 перевіряє приналежність операційних елементів до останніх в рядках матричного помножувача мантис.

Блок 40 обчислює значення точки a в останніх операційних елементів в рядках матричного помножувача мантис.

Блок 41 обчислює значення точки a у передостанніх операційних елементів в рядках матричного помножувача мантис.

Блок 42 присвоює сигналу в точці a_1 операційного елемента матричного помножувача мантис значення розряду множеного.

Блок 43 присвоює сигналу в точці b_1 операційного елемента матричного помножувача мантис значення розряду множника.

Блок 44 завершує цикл з переліком номерів операційних елементів в рядках матричного помножувача мантис.

Блок 45 завершує цикл з переліком номерів рядків операційних елементів в матричному помножувачі мантис.

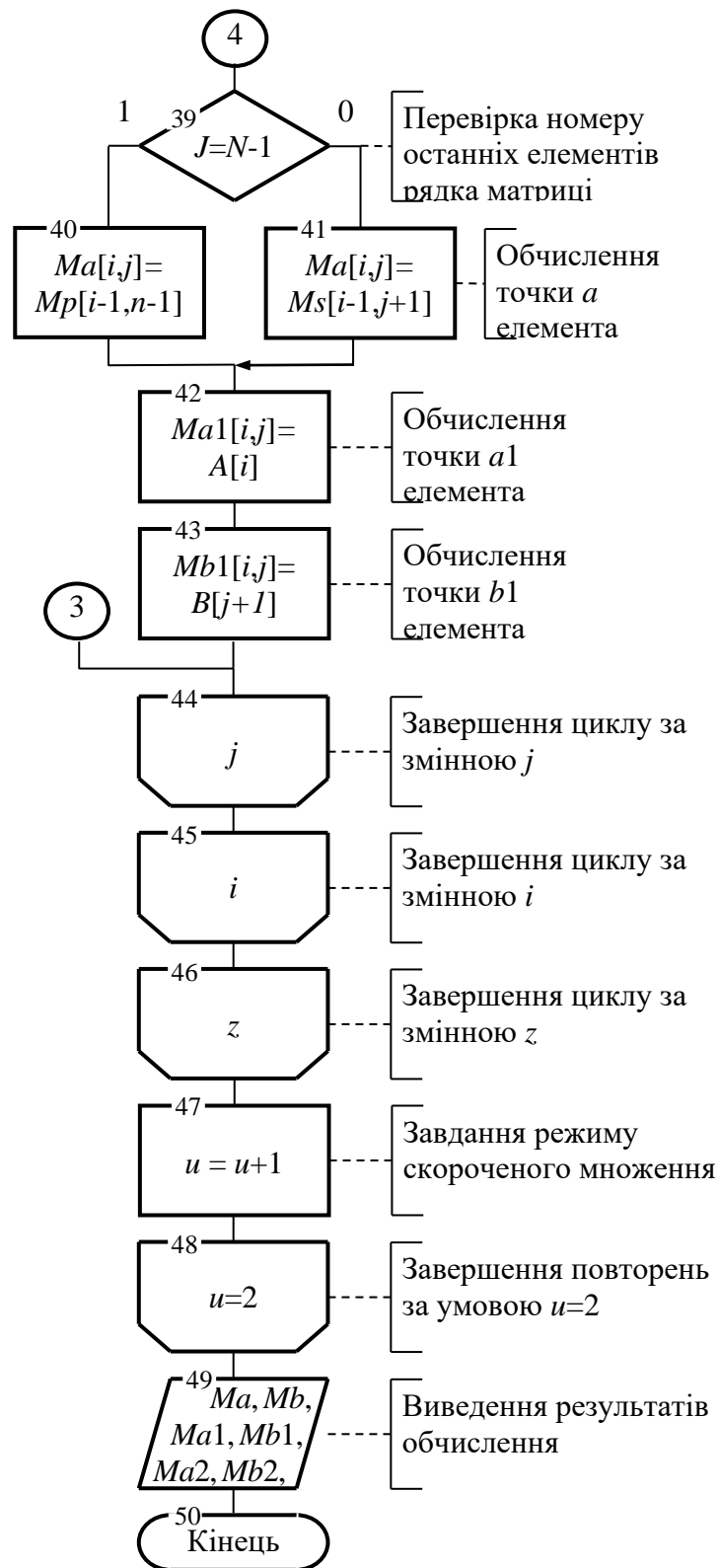


Рисунок 3.4 – Блоки 39 – 50 схеми алгоритму

Блок 46 завершує цикл з переліком значень кратності відмов для їх введення в матричний помножувач мантис.

Блок 47 задає режим виконання в матричному помножувачі мантис операції скороченого множення.

Блок 48 завершує цикл з повторенням ітерацій виконання в матричному помножувачі мантис спочатку повної, а потім скороченої операції.

Блок 49 виводить результати обчислень а розглянутих точках операційних елементів матричного помножувача мантис.

Блок 50 завершує схему алгоритму.

3.4 Висновки

Програмна модель матричного помножувача мантис та його системи контролю за модулем складається з керуючих та демонстраційних елементів та забезпечує режими функціонування з підготовки та проведення дослідження робочого діагностування цифрового компонента системи критичного застосування під дією кратних несправностей, які можуть викликатися в аварійному режимі внаслідок накопичення прихованих несправностей.

Програмна модель будується за низкою розроблених алгоритмів, включаючи алгоритми визначення вихідних даних і виконання функцій множення та контролю в умовах дії кратних несправностей.

Алгоритм введення кратних відмов визначає довільний вид замикання двох точок операційних елементів у множині номерів несправностей та довільні номери несправних операційних елементів у множині рядків та елементів рядків матричного помножувача мантис.

При збіганні визначених номерів з поточними кожна функція множення після виконання викривляється заміною значень несправних точок на їх кон'юнкцію, яка відображає відмову замикання.

4 РЕЗУЛЬТАТИ МОДЕЛЮВАННЯ

4.1 Установки програмної моделі

4.1.1 Керівництво програміста

Програмний продукт призначений для моделювання роботи матричного помножувача мантис та його контролю за модулем три. Обчислення виконуються у доповняльному коді під дією кратних відмов типу «замикання» в схемах операційних елементів матричного помножувача мантис.

Програма написана у середовищі Delphi з використанням однієї форми та 15 процедур. Вихідні дані задаються інтерактивному режимі в процесі роботи програмного продукту. Результати моделювання формуються та зберігаються на панелі контролю. Програмний продукт поставляється у вигляді завантажувального модуля і запускається на виконання по команді ENTER.

4.1.3 Користувальницький інтерфейс. При натисканні кнопки «вхід» на екрані з'являється панель, що пропонує задати розрядність матричного помножувача мантис у діапазоні 8 – 15 або прийняти розрядність 8. Надається також можливість вийти з процесу виконання програми, натискаючи кнопку “Вихід”.

При натисканні кнопки “Далі” на екрані з'являється панель опису об'єкта контролю – матричного помножувача мантис у доповняльному коді, що включає до свого складу наступні компоненти:

- меню режимів завдання операндів і стану пристрою;
- поля та кнопки для завдання та прийому операндів десятковим числом і двійковим кодом в режимах визначення конкретних даних;
- структуру матричного помножувача заданої розрядності;
- поля та кнопки завдання місця та виду відмови, що забезпечує режими їх конкретного та довільного визначення;
- показчик розподілу розрядів результату на вірні та невірні.

На рис. 4.1 показана панель матричного помножувача мантис у доповняльному коді для розрядності операндів 8.

Form1

Операнд A: -317 Операнд B: 464 Стан: **Фініш** (2)

C: 115056 T: 115008
Ce: 115056 Te: 115008

Розрядність помножувача 9

Вид несправності: закоротка

! > s p > !

Довільна

Неспр. елемент: 8349

A	0	0	1	1	0	0	0	0	0	1	1	B							
			1.9	1.8	1.7	1.6	1.5	1.4	1.3	1.2		0							
			2.9	2.8	2.7	2.6	2.5	2.4	2.3	2.2		0							
			3.9	3.8	3.7	3.6	3.5	3.4	3.3	3.2		0							
			4.9	4.8	4.7	4.6	4.5	4.4	4.3	4.2		0							
			5.9	5.8	5.7	5.6	5.5	5.4	5.3	5.2		1							
			6.9	6.8	6.7	6.6	6.5	6.4	6.3	6.2		0							
			7.9	7.8	7.7	7.6	7.5	7.4	7.3	7.2		1							
			8.9	8.8	8.7	8.6	8.5	8.4	8.3	8.2		1							
			9.9	9.8	9.7	9.6	9.5	9.4	9.3	9.2		1							
109	108	107	106	105	104	103	102	101				0							
119	118	117	116	115	114	113	112	111											
	0	1	1	1	0	0	0	0	0	1	0	1	1	1	0	0	0	0	C
	0	1	1	1	0	0	0	0	0	1	0	1	1	1	0	0	0	0	Ce
	0	1	1	1	0	0	0	0	0	1	0	1							T
	0	1	1	1	0	0	0	0	0	1	0	1							Te

Рисунок 4.1 – Панель моделювання 9-розрядного матричного помножувача мантис у доповняльному коді

При натисканні на панель «Контроль» графічне зображення матричного помножувача замінюється на панель контролю, що показана рис. 4.2.

Form1

Операнд A: 166, Операнд B: -164, Стан: Старт, Контроль

Розрядність помножувача 8

ПОВН. СКОР.	ПОМИЛКИ		
	суттєві	несуттєві	всього
Виявлені	3,2%	4,0%	7,2%
	3,3%	2,4%	5,7%
Невиявл.	0,0%	0,0%	0,0%
	0,0%	0,0%	0,0%
Всього	3,2%	4,0%	7,2%
	3,3%	2,4%	5,7%

ДОСТОВІРНІСТЬ

RC	96,8%	RT	96,7%	CC	44,1%	CT	57,4%
	92,5%		92,6%		55,2%		66,3%

1 2 3
1 2 3

Рисунок 4.2 – Панель контролю для 8-розрядного матричного помножувача мантис під дією 1- та 2-кратних відмов

Панель містить:

- поля множеного та множника у доповняльному коді;
- поля правильних результатів і результатів моделювання, обчислених під дією несправності для випадків повного та скороченого виконання операції множення;
 - схему контролю за модулем, що показує значення операндів, результатів і їхніх контрольних кодів, а також коду контролю результату;
 - покажчик розподілу результату на вірні та невірні розряди обчисленого добутку;
 - таблицю результатів моделювання, що показує ймовірності появи, виявлення та пропуску суттєвих і несуттєвих помилок для випадків повного та скороченого виконання операції у доповняльному коді під дією кратних відмов (позначено різними кольорами зверху до низу частини таблиці для декількох кратностей).

Панель моделювання контролю за модулем для випадку 10-розрядного матричного помножувача показана на рис. 4.3. На даній моделі зафіксоване виявлення помилки контролем за модулем три.

Form1

Операнд А Операнд В Стан

А -777 **Фініш** С 418026 Т 876416

В -538 **Контроль** Се 415978 Те 872320

Розрядність помножувача 10

ПОВН. СКОР.	ПОМИЛКИ		
	суттєві	несуттєві	всього
Виявлені	5,1%	6,2%	11,3%
	5,2%	3,8%	9,0%
Невиявл.	0,4%	0,2%	0,6%
	0,3%	0,1%	0,4%
Всього	5,5%	6,4%	11,9%
	5,6%	3,9%	9,5%

ПОВН. СКОР.	ПОМИЛКИ		
	суттєві	несуттєві	всього
Виявлені	6,4%	8,1%	14,5%
	6,6%	5,0%	11,6%
Невиявл.	0,9%	0,5%	1,5%
	0,8%	0,3%	1,1%
Всього	7,3%	8,7%	16,0%
	7,5%	5,3%	12,8%

ДОВОІРНІСТЬ

RC 94,4% RT 94,4% CC 44,6% CT 56,3%

92,6% 92,5% 43,3% 54,2%

1 2 3

1 2 3

Рисунок 4.3 – Панель контролю для 10-розрядного матричного помножувача мантис під дією 2- та 3-кратних відмов

Моделювання процесу контролю починається з підготовки проведення основних обчислень на панелі матричного помножувача мантис.

Установлюються режими завдання операндів.

Для цього передбачені меню, з яких для кожного з операндів вибираються позиції, що визначають його задані або довільні значення, а також повний перебір або послідовність довільних значень операнда.

За допомогою меню встановлюється також стан пристрою, що може бути в одному із двох положень: «справно» або «несправне».

У режимі заданого значення операнд може бути визначений десятковим числом або у вигляді двійкового коду, що може бути змінений шляхом інвертування окремих розрядів, обраних щигликом миші.

Для пристрою, що перебуває в стані «несправне», визначаються вид і місце конкретної несправності типу «закоротка» або встановлюються режими їхнього довільного вибору.

На початку поле вірних розрядів результату визначається розрядністю двійкового коду мантиси операнда.

Переміщенням відповідного покажчика це поле може бути збільшене на значення, що перебуває в межах розрядності операнда.

Початок обчислень визначається натисканням кнопки «Старт», після чого відбувається її перейменування на назву «Фініш».

У режимах установки одного набору операндів моделювання виконується спочатку для варіанта обчислення повного добутку, після чого з'являється запрошення перейти до скороченого множення.

При щиглику по панелі запрошення виконується моделювання варіанта обчислення скороченого множення.

Програмна модель дозволяє простежити обчислення повного або усіченого добутку мантис у доповняльному коді, розкриваючи по черзі операційні елементи матричного помножувача та переглядаючи значення їхніх внутрішніх точок.

За результатами проведеного моделювання (у випадку вибору режиму несправності) заповнюються поля доповняльних кодів правильних добутків і добутків, обчислених під впливом кратної несправності для випадків повного й скороченого множення.

Крім того, представляються десяткові значення одержуваних результатів та величини помилок.

У режимах перебору значень операндів виводяться також кількість помилок і їхня частка в загальній кількості виконуваних операцій, а у вигляді таблиці – частка суттєвих і несуттєвих помилок.

Щиглик миші, виконуваний по панелі «Множення», міняє її назву на «Контроль», а також установлює графічне зображення, що відповідає виконуваному контролю за модулем. При цьому фіксується поточний стан

операндів і результатів, що обчислюються, а також даних, які виводяться на схемі контролю та у таблиці результатів моделювання.

Щиглик миші, виконуваний по перейменованій панелі «Контроль», повертає вихідне графічне зображення матричного помножувача та назву панелі «Множення».

Переходи на панелі демонстрації основних обчислень та контролю за числовим модулем можуть здійснюватися багаторазово. Це дозволяє стежити за ходом виконання моделювання й оцінювати його проміжні та остаточні результати.

Процес генерації послідовності значень операндів може бути зупинений натисканням кнопки «Фініш», після чого відбувається її перейменування на назву «Старт».

4.1.4 Обробка екстрених ситуацій. Передбачено виведення повідомлень про помилки у наступних випадках:

- при порушенні діапазону операндів, що задані цілим десятковим числом;
- при введенні ненормалізованого доповняльного коду мантиси множеного або множника;
- при введенні неправильного номера несправного операційного елемента матричного помножувача мантис.

4.2 Моделювання та його результати

4.2.1 За результатами моделювання помножувача мантис та його системи контролю для кожної кратності несправності визначаються наступні дані:

P_{VSP} та P_{VCU} – ймовірності виявлення суттєвої помилки в повному та усіченому добутку, відповідно;

$P_{ВНП}$ та $P_{ВНУ}$ – ймовірності виявлення несуттєвої помилки в повному та усіченому добутку, відповідно;

$P_{ПСП}$ та $P_{ПСУ}$ – ймовірності пропуску суттєвої помилки в повному та усіченому добутку, відповідно;

$P_{ПНП}$ та $P_{ПНУ}$ – ймовірності пропуску несуттєвої помилки в повному та усіченому добутку, відповідно;

$P_{ВП}$ та $P_{ВУ}$ – ймовірності виявлення помилки в повному та усіченому добутку, відповідно;

$P_{ПП}$ та $P_{ПУ}$ – ймовірності пропуску помилки в повному та усіченому добутку, відповідно;

$P_{СП}$ та $P_{СУ}$ – ймовірності появи суттєвої помилки в повному та усіченому добутку, відповідно;

$P_{НП}$ та $P_{НУ}$ – ймовірності появи несуттєвої помилки в повному та усіченому добутку, відповідно.

$P_{П}$ та $P_{У}$ – ймовірності появи помилки в повному та усіченому добутку, відповідно.

За одержаними результатами моделювання оцінюється достовірність D контролю результатів за модулем три для різної кратності несправності та випадків повного та скороченого множення наступним чином:

$$D_{П} = (P_{ВСП} + P_{ПНП}) / P_{П};$$

$$D_{У} = (P_{ВСУ} + P_{ПНУ}) / P_{У};,$$

де $D_{П}$ – значення достовірності D при обчисленні повного добутку,

$D_{У}$ – значення достовірності D при обчисленні усіченого добутку.

Ефективність скорочення обчислень у частині підвищення достовірності контролю наближених результатів у доповняльному коді може бути оцінена за формулою:

$$R = D_{У} / D_{П}.$$

4.2.2 Моделювання контролю за числовим модулем три виконується при випадковому виборі несправного операційного елемента та довільному завданні виду замикання з повним перебором всіх вхідних слів або в основному режимі на випадкових вхідних послідовностях водночас для всіх значень кратності несправності: 1, 2 та 3.

У табл. 4.1 показане розташування результатів моделювання, що одержуються в ході проведення діагностичного експерименту за програмною моделлю контролю за модулем три матричного помножувача мантис у доповняльному коді для кожної кратності несправності.

Таблиця 4.1 – Розташування результатів моделювання
для кожної кратності несправності

	ПОМИЛКИ		
	суттєві	несуттєві	усього
Виявлення помилок	$R_{VСП}$	$R_{VНП}$	$R_{ВП}$
	$R_{VСУ}$	$R_{VНУ}$	$R_{ВУ}$
Пропуск помилок	$R_{ПСП}$	$R_{ПНП}$	$R_{ПП}$
	$R_{ПСУ}$	$R_{ПНУ}$	$R_{ПУ}$
Усього	$R_{СП}$	$R_{НП}$	$R_{П}$
	$R_{СУ}$	$R_{НУ}$	$R_{У}$

Результати моделювання системи контролю для випадку $n_B = n$ вірних розрядів результату та дії 1-кратної несправності наведено в табл. 4.2.

Таблиця 4.2 – Ймовірності $P_{ВСП}$, $P_{ВСУ}$, $P_{ВНП}$, $P_{ВНУ}$, $P_{ПСП}$, $P_{ПСУ}$, $P_{ПНП}$, $P_{ПНУ}$, $P_{ВП}$, $P_{ВУ}$, $P_{ПП}$, $P_{ПУ}$, $P_{СП}$, $P_{СУ}$, $P_{НП}$, $P_{НУ}$, $P_{П}$, $P_{У}$, $D_{П}$, $D_{У}$ та R для $n_B = n$ та 1-кратної несправності

№ п/п	Результати моделювання	n							
		8	9	10	11	12	13	14	15
1	$P_{ВСП}$, %	3.6	3.6	3.5	3.5	3.4	3.4	3.3	3.3
2	$P_{ВСУ}$, %	3.6	3.6	3.5	3.5	3.4	3.4	3.3	3.3
3	$P_{ВНП}$, %	2.9	2.9	3.0	3.0	3.1	3.1	3.2	3.2
4	$P_{ВНУ}$, %	1.7	1.7	1.6	1.6	1.6	1.5	1.5	1.4
5	$P_{ВП}$, %	6.5	6.5	6.5	6.5	6.5	6.5	6.5	6.5
6	$P_{ВУ}$, %	5.3	5.3	5.1	5.1	5.0	4.9	4.8	4.7
7	$P_{ПСП}$, %	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
8	$P_{ПСУ}$, %	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
9	$P_{ПНП}$, %	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
10	$P_{ПНУ}$, %	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
11	$P_{ПП}$, %	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
12	$P_{ПУ}$, %	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
13	$P_{СП}$, %	3.6	3.6	3.5	3.5	3.4	3.4	3.3	3.3
14	$P_{СУ}$, %	3.6	3.6	3.5	3.5	3.4	3.4	3.3	3.3
15	$P_{НП}$, %	2.9	2.9	3.0	3.0	3.1	3.1	3.2	3.2
16	$P_{НУ}$, %	1.7	1.7	1.6	1.6	1.6	1.5	1.5	1.4
17	$P_{П}$, %	6.5	6.5	6.5	6.5	6.5	6.5	6.5	6.5
18	$P_{У}$, %	5.3	5.3	5.1	5.1	5.0	4.9	4.8	4.7
19	$D_{П}$, %	55,4	55,4	53,8	53,8	52,3	52,3	50,8	50,8
20	$D_{У}$, %	54,7	54,7	58,8	58,8	62,0	63,3	68,8	70,2
21	R	0.99	0.99	1.09	1.09	1.19	1.21	1.35	1.38

Результати моделювання системи контролю для випадку $n_B = n - 6$ вірних розрядів результату та дії 1-кратної несправності наведено в табл. 4.3.

Таблиця 4.3 – Ймовірності $P_{ВСП}$, $P_{ВСУ}$, $P_{ВНП}$, $P_{ВНУ}$, $P_{ПСП}$, $P_{ПСУ}$, $P_{ПНП}$, $P_{ПНУ}$, $P_{ВП}$, $P_{ВУ}$, $P_{ПП}$, $P_{ПУ}$, $P_{СП}$, $P_{СУ}$, $P_{НП}$, $P_{НУ}$, $P_{П}$, $P_{У}$, $D_{П}$, $D_{У}$ та R для $n_B = n - 6$ та 1-кратної несправності

№ п/п	Результати моделювання	n							
		8	9	10	11	12	13	14	15
1	$P_{ВСП}$, %	0.4	0.6	0.7	0.9	1.0	1.1	1.2	1.3
2	$P_{ВСУ}$, %	0.4	0.6	0.7	0.9	1.0	1.1	1.2	1.3
3	$P_{ВНП}$, %	6.0	5.9	5.8	5.7	5.6	5.5	5.4	5.3
4	$P_{ВНУ}$, %	4.9	4.6	4.4	4.2	4.1	3.9	3.7	3.5
5	$P_{ВП}$, %	6.4	6.5	6.5	6.6	6.6	6.6	6.6	6.6
6	$P_{ВУ}$, %	5.3	5.2	5.1	5.1	5.1	5.0	4.9	4.8
7	$P_{ПСП}$, %	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
8	$P_{ПСУ}$, %	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
9	$P_{ПНП}$, %	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
10	$P_{ПНУ}$, %	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
11	$P_{ПП}$, %	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
12	$P_{ПУ}$, %	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
13	$P_{СП}$, %	0.4	0.6	0.7	0.9	1.0	1.1	1.2	1.3
14	$P_{СУ}$, %	0.4	0.6	0.7	0.9	1.0	1.1	1.2	1.3
15	$P_{НП}$, %	6.0	5.9	5.8	5.7	5.6	5.5	5.4	5.3
16	$P_{НУ}$, %	4.9	4.6	4.4	4.2	4.1	3.9	3.7	3.5
17	$P_{П}$, %	6.4	6.5	6.5	6.6	6.6	6.6	6.6	6.6
18	$P_{У}$, %	5.3	5.2	5.1	5.1	5.1	5.0	4.9	4.8
19	$D_{П}$, %	6.3	9.2	10.8	13.6	15.2	16.7	18.2	19.7
20	$D_{У}$, %	7.5	11.5	13.7	17.6	19.6	22.0	24.5	27.1
21	R	1.19	1.25	1.27	1.29	1.29	1.32	1.35	1.38

Результати моделювання системи контролю для випадку $n_B = n$ вірних розрядів результату та дії 2-кратної несправності наведено в табл. 4.4.

Таблиця 4.4 – Ймовірності $P_{ВСП}$, $P_{ВСУ}$, $P_{ВНП}$, $P_{ВНУ}$, $P_{ПСП}$, $P_{ПСУ}$, $P_{ПНП}$, $P_{ПНУ}$, $P_{ВП}$, $P_{ВУ}$, $P_{ПП}$, $P_{ПУ}$, $P_{СП}$, $P_{СУ}$, $P_{НП}$, $P_{НУ}$, $P_{П}$, $P_{У}$, $D_{П}$, $D_{У}$ та R для $n_B = n$ та 2-кратної несправності

№ п/п	Результати моделювання	n							
		8	9	10	11	12	13	14	15
1	$P_{ВСП}$, %	6.3	6.3	6.2	6.2	6.1	6.1	6.0	6.0
2	$P_{ВСУ}$, %	6.3	6.3	6.2	6.2	6.1	6.1	6.0	6.0
3	$P_{ВНП}$, %	4.8	4.9	5.0	5.1	5.2	5.3	5.4	5.5
4	$P_{ВНУ}$, %	2.9	2.8	2.7	2.6	2.5	2.4	2.3	2.2
5	$P_{ВП}$, %	11.1	11.2	11.2	11.3	11.3	11.4	11.3	11.5
6	$P_{ВУ}$, %	9.2	9.1	8.9	8.8	8.6	8.5	8.3	8.2
7	$P_{ПСП}$, %	0.4	0.4	0.4	0.4	0.4	0.4	0.4	0.4
8	$P_{ПСУ}$, %	0.3	0.3	0.3	0.3	0.3	0.3	0.3	0.3
9	$P_{ПНП}$, %	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1
10	$P_{ПНУ}$, %	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1
11	$P_{ПП}$, %	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5
12	$P_{ПУ}$, %	0.4	0.4	0.4	0.4	0.4	0.4	0.4	0.4
13	$P_{СП}$, %	6.7	6.7	6.6	6.6	6.5	6.5	6.4	6.4
14	$P_{СУ}$, %	6.6	6.6	6.5	6.5	6.4	6.4	6.3	6.3
15	$P_{НП}$, %	4.9	5.0	5.1	5.2	5.3	5.4	5.5	5.6
16	$P_{НУ}$, %	3.0	2.9	2.8	2.7	2.6	2.5	2.4	2.3
17	$P_{П}$, %	11.6	11.7	11.7	11.8	11.8	11.9	11.9	12.0
18	$P_{У}$, %	9.6	9.5	9.3	9.2	9.0	8.9	8.7	8.6
19	$D_{П}$, %	55,2	54,7	53,8	53,4	52,5	52,1	51,3	50,8
20	$D_{У}$, %	66,7	67,4	67,7	68,5	68,9	69,7	70,1	70,9
21	R	1.21	1.23	1.26	1.28	1.31	1.34	1.37	1.40

Результати моделювання системи контролю для випадку $n_B = n - 6$ вірних розрядів результату та дії 2-кратної несправності наведено в табл. 4.5.

Таблиця 4.5 – Ймовірності $P_{ВСП}$, $P_{ВСУ}$, $P_{ВНП}$, $P_{ВНУ}$, $P_{ПСП}$, $P_{ПСУ}$, $P_{ПНП}$, $P_{ПНУ}$, $P_{ВП}$, $P_{ВУ}$, $P_{ПП}$, $P_{ПУ}$, $P_{СП}$, $P_{СУ}$, $P_{НП}$, $P_{НУ}$, $P_{П}$, $P_{У}$, $D_{П}$, $D_{У}$ та R для $n_B = n - 6$ та 2-кратної несправності

№ п/п	Результати моделювання	n							
		8	9	10	11	12	13	14	15
1	$P_{ВСП}$, %	0.7	1.0	1.3	1.6	1.8	2.0	2.2	2.4
2	$P_{ВСУ}$, %	0.7	1.0	1.3	1.6	1.8	2.0	2.2	2.4
3	$P_{ВНП}$, %	10.4	10.1	9.8	9.6	9.5	9.3	9.1	9.0
4	$P_{ВНУ}$, %	8.5	8.1	7.6	7.2	6.9	6.6	6.2	5.9
5	$P_{ВП}$, %	11.1	11.1	11.1	11.2	11.3	11.3	11.3	11.4
6	$P_{ВУ}$, %	9.2	9.1	8.9	8.8	8.7	8.6	8.4	8.3
7	$P_{ПСП}$, %	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1
8	$P_{ПСУ}$, %	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1
9	$P_{ПНП}$, %	0.4	0.4	0.4	0.4	0.4	0.4	0.4	0.4
10	$P_{ПНУ}$, %	0.2	0.2	0.2	0.2	0.2	0.2	0.2	0.2
11	$P_{ПП}$, %	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5
12	$P_{ПУ}$, %	0.3	0.3	0.3	0.3	0.3	0.3	0.3	0.3
13	$P_{СП}$, %	0.8	1.1	1.4	1.7	1.9	2.1	2.3	2.5
14	$P_{СУ}$, %	0.8	1.1	1.4	1.7	1.9	2.1	2.3	2.5
15	$P_{НП}$, %	10.8	10.5	10.2	10.0	9.9	9.7	9.5	9.4
16	$P_{НУ}$, %	8.7	8.3	7.8	7.4	7.1	6.8	6.4	6.1
17	$P_{П}$, %	11.6	11.6	11.6	11.7	11.8	11.8	11.8	11.9
18	$P_{У}$, %	9.5	9.4	9.2	9.1	9.0	8.9	8.7	8.6
19	$D_{П}$, %	9.5	12.1	14.7	17.1	18.6	20.3	22.0	23.6
20	$D_{У}$, %	9.5	12.8	16.3	19.8	22.2	24.7	27.6	30.2
21	R	1.00	1.06	1.11	1.16	1.19	1.22	1.25	1.28

Результати моделювання системи контролю для випадку $n_B = n$ вірних розрядів результату та дії 3-кратної несправності наведено в табл. 4.6.

Таблиця 4.6 – Ймовірності $P_{ВСП}$, $P_{ВСУ}$, $P_{ВНП}$, $P_{ВНУ}$, $P_{ПСП}$, $P_{ПСУ}$, $P_{ПНП}$, $P_{ПНУ}$, $P_{ВП}$, $P_{ВУ}$, $P_{ПП}$, $P_{ПУ}$, $P_{СП}$, $P_{СУ}$, $P_{НП}$, $P_{НУ}$, $P_{П}$, $P_{У}$, $D_{П}$, $D_{У}$ та R для $n_B = n$ та 3-кратної несправності

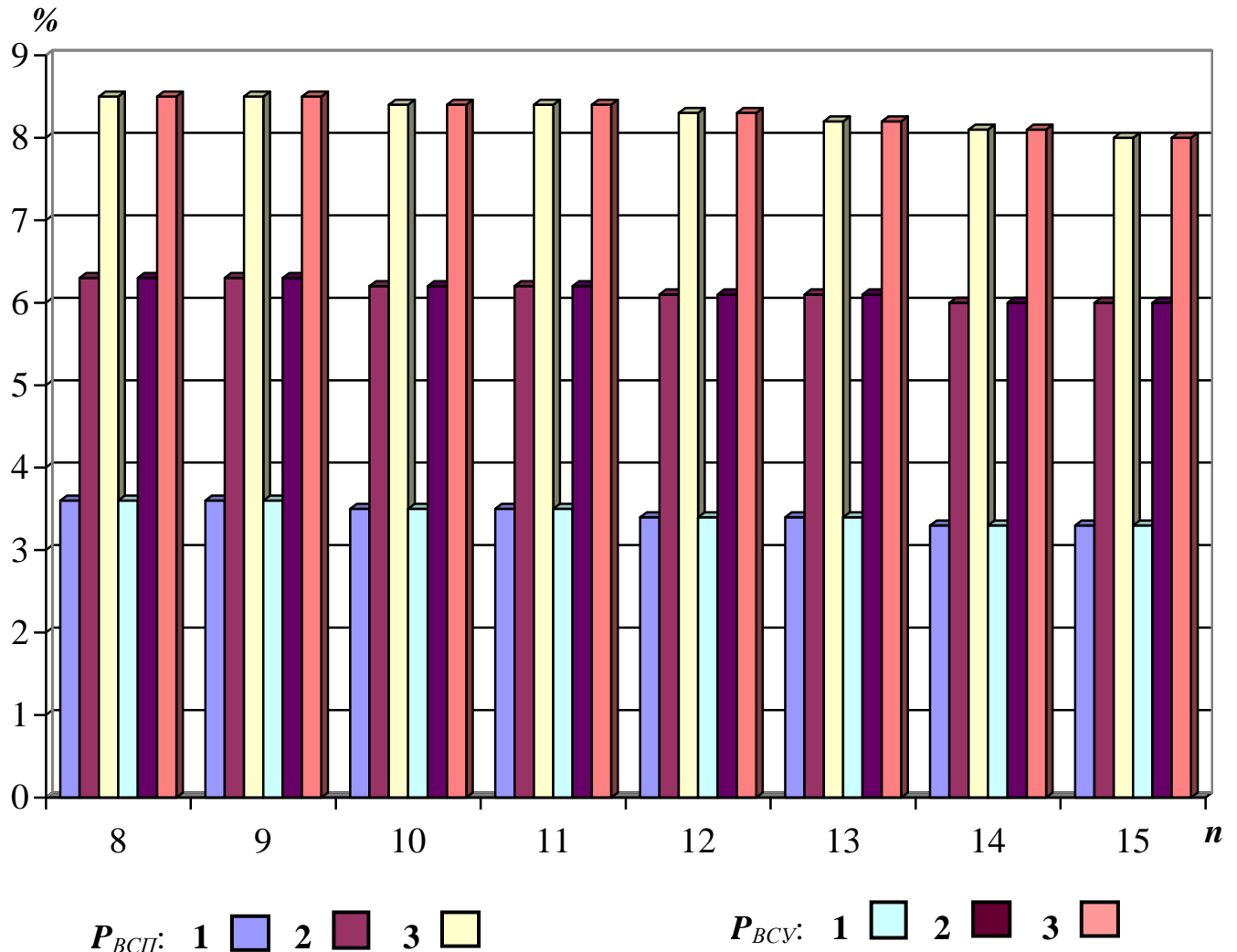
№ п/п	Результати моделювання	n							
		8	9	10	11	12	13	14	15
1	$P_{ВСП}$, %	8.5	8.5	8.4	8.4	8.3	8.2	8.1	8.0
2	$P_{ВСУ}$, %	8.5	8.5	8.4	8.4	8.3	8.2	8.1	8.0
3	$P_{ВНП}$, %	6.0	6.1	6.2	6.3	6.4	6.5	6.6	6.7
4	$P_{ВНУ}$, %	3.8	3.7	3.6	3.5	3.4	3.3	3.2	3.1
5	$P_{ВП}$, %	14.5	14.5	14.6	14.7	14.7	14.7	14.7	14.7
6	$P_{ВУ}$, %	12.3	12.2	12.0	11.9	11.7	11.5	11.3	11.1
7	$P_{ПСП}$, %	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
8	$P_{ПСУ}$, %	0.9	0.8	0.8	0.8	0.8	0.8	0.8	0.8
9	$P_{ПНП}$, %	0.3	0.3	0.3	0.3	0.3	0.3	0.3	0.3
10	$P_{ПНУ}$, %	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1
11	$P_{ПП}$, %	1.3	1.3	1.3	1.3	1.3	1.3	1.3	1.3
12	$P_{ПУ}$, %	1.0	0.9	0.9	0.9	0.9	0.9	0.9	0.9
13	$P_{СП}$, %	9.5	9.5	9.4	9.4	9.3	9.2	9.1	9.0
14	$P_{СУ}$, %	9.4	9.3	9.2	9.2	9.1	9.0	8.9	8.8
15	$P_{НП}$, %	6.3	6.4	6.5	6.6	6.7	6.8	6.9	7.0
16	$P_{НУ}$, %	3.9	3.8	3.7	3.6	3.5	3.4	3.3	3.2
17	$P_{П}$, %	15.8	15.9	15.9	16.0	16.0	16.0	16.0	16.0
18	$P_{У}$, %	13.3	13.1	12.9	12.8	12.6	12.4	12.2	12.0
19	$D_{П}$, %	55,7	55,3	54,7	54,4	53,8	53,1	52,5	51,9
20	$D_{У}$, %	64.7	65.6	65.9	66.4	66.7	66.9	67.2	67.5
21	R	1.16	1.19	1.20	1.22	1.24	1.26	1.28	1.30

Результати моделювання системи контролю для випадку $n_B = n - 6$ вірних розрядів результату та дії 3-кратної несправності наведено в табл. 4.7.

Таблиця 4.7 – Ймовірності $P_{ВСП}$, $P_{ВСУ}$, $P_{ВНП}$, $P_{ВНУ}$, $P_{ПСП}$, $P_{ПСУ}$, $P_{ПНП}$, $P_{ПНУ}$, $P_{ВП}$, $P_{ВУ}$, $P_{ПП}$, $P_{ПУ}$, $P_{СП}$, $P_{СУ}$, $P_{НП}$, $P_{НУ}$, $P_{П}$, $P_{У}$, $D_{П}$, $D_{У}$ та R для $n_B = n - 6$ та 3-кратної несправності

№ п/п	Результати моделювання	n							
		8	9	10	11	12	13	14	15
1	$P_{ВСП}$, %	1.0	1.4	1.8	2.2	2.5	2.8	3.0	3.2
2	$P_{ВСУ}$, %	1.0	1.4	1.8	2.2	2.5	2.8	3.0	3.2
3	$P_{ВНП}$, %	13.5	13.1	12.7	12.4	12.1	11.8	11.6	11.4
4	$P_{ВНУ}$, %	11.3	10.8	10.3	9.7	9.2	8.7	8.3	8.0
5	$P_{ВП}$, %	14.5	14.5	14.5	14.6	14.6	14.6	14.6	14.6
6	$P_{ВУ}$, %	12.3	12.2	12.1	11.9	11.7	11.5	11.3	11.2
7	$P_{ПСП}$, %	0.1	0.1	0.2	0.2	0.3	0.3	0.4	0.4
8	$P_{ПСУ}$, %	0.1	0.1	0.2	0.2	0.3	0.3	0.4	0.4
9	$P_{ПНП}$, %	1.2	1.2	1.1	1.1	1.0	1.0	1.0	1.0
10	$P_{ПНУ}$, %	0.9	0.8	0.7	0.7	0.6	0.6	0.5	0.5
11	$P_{ПП}$, %	1.3	1.3	1.3	1.3	1.3	1.3	1.4	1.4
12	$P_{ПУ}$, %	1.0	0.9	0.9	0.9	0.9	0.9	0.9	0.9
13	$P_{СП}$, %	1.1	1.5	2.0	2.4	2.8	3.1	3.4	3.6
14	$P_{СУ}$, %	1.1	1.5	2.0	2.4	2.8	3.1	3.4	3.6
15	$P_{НП}$, %	14.7	14.3	13.8	13.5	13.1	12.8	12.6	12.4
16	$P_{НУ}$, %	12.2	11.6	11.0	10.4	9.8	9.3	8.8	8.5
17	$P_{П}$, %	15.8	15.8	15.8	15.9	15.9	15.9	16.0	16.0
18	$P_{У}$, %	13.3	13.1	13.0	12.8	12.6	12.4	12.2	12.1
19	$D_{П}$, %	13.9	16.5	18.4	20.8	22.0	23.9	25.0	26.3
20	$D_{У}$, %	14.3	16.8	19.2	22.7	24.6	27.4	28.7	30.6
21	R	1.03	1.03	1.04	1.09	1.12	1.15	1.15	1.16

4.2.3 На рис. 4.4 показані діаграми зміни ймовірностей $p_{всп}$ та $p_{всу}$ виявлення суттєвої помилки в доповняльному коді повного та усіченого добутків для кратності несправності 1, 2, 3 та ряду значень розрядності n матричних помножувачів мантис і кількості вірних розрядів $n_e = n$.

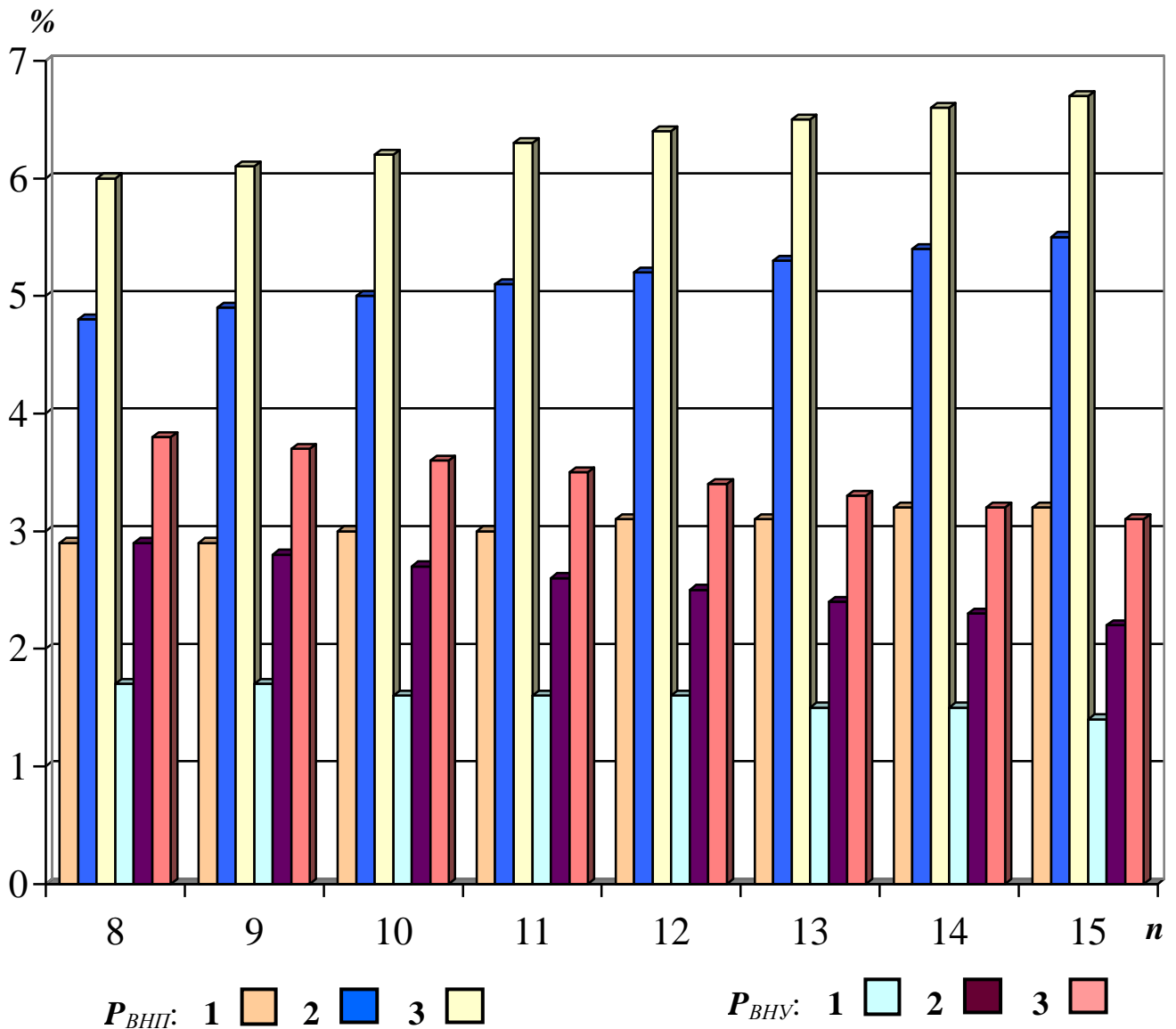


1, 2, 3 – кратність несправності

Рисунок 4.4 – Діаграми ймовірностей P_{VSP} і P_{VSU} виявлення суттєвої помилки в повному та усіченому добутку при різних крайностях несправності

Діаграми показують, що ймовірності виявлення суттєвої помилки в повному й усіченому добутках практично збігаються та істотно підвищуються із зростанням кратності несправності й незначно зменшуються при збільшенні розрядності n .

На рис. 4.5 показані діаграми зміни ймовірностей $p_{внп}$ та $p_{вну}$ виявлення несуттєвої помилки в доповняльному коді повного та зсіченого добутоків для кратності несправності 1, 2, 3 та ряду значень розрядності n матричних помножувачів мантис і кількості вірних розрядів $n_6 = n$.



1, 2, 3 – кратність несправності

Рисунок 4.5 – Діаграми ймовірностей $P_{ВНП}$ і $P_{ВНУ}$ виявлення несуттєвої помилки в повному та усіченому добутках при кратних несправностях

Ймовірності виявлення несуттєвої помилки в повному добутку більші ніж в усіченому, перші зростають, а другі зменшуються при збільшенні розрядності n та разом підвищуються із зростанням кратності несправності.

На рис. 4.6 показані діаграми зміни ймовірностей p_{en} та p_{ey} виявлення помилки в доповняльному коді повного та зсіченого добутків для кратності несправності 1, 2, 3 та ряду значень розрядності n матричних помножувачів мантис і кількості вірних розрядів $n_e = n$.

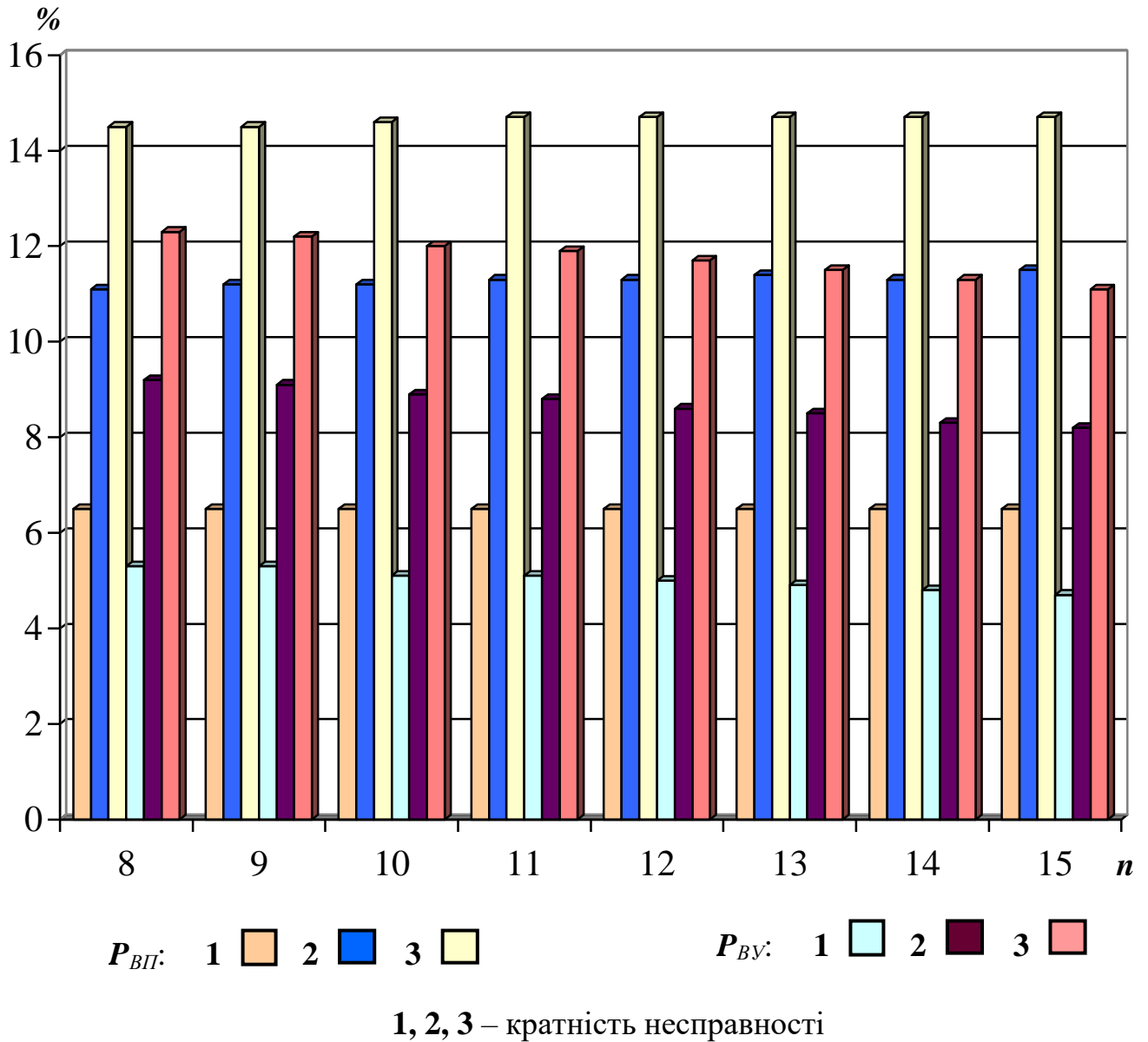


Рисунок 4.6 – Діаграми ймовірностей $P_{ВП}$ і $P_{ВУ}$ виявлення помилки в повному та усіченому добутках при кратних несправностях

Ймовірності виявлення помилки в повному добутку більші ніж в усіченому, перші майже не змінюються, а другі зменшуються при збільшенні розрядності n та разом підвищуються із зростанням кратності несправності.

на рис. 4.7 показані діаграми зміни достовірності d_n контролю повних добутоків за модулем три для кількості вірних розрядів $n_6 = n$ результату матричного помножувача мантис та кратності несправності 1, 2, 3 від ряду значень розрядності n .

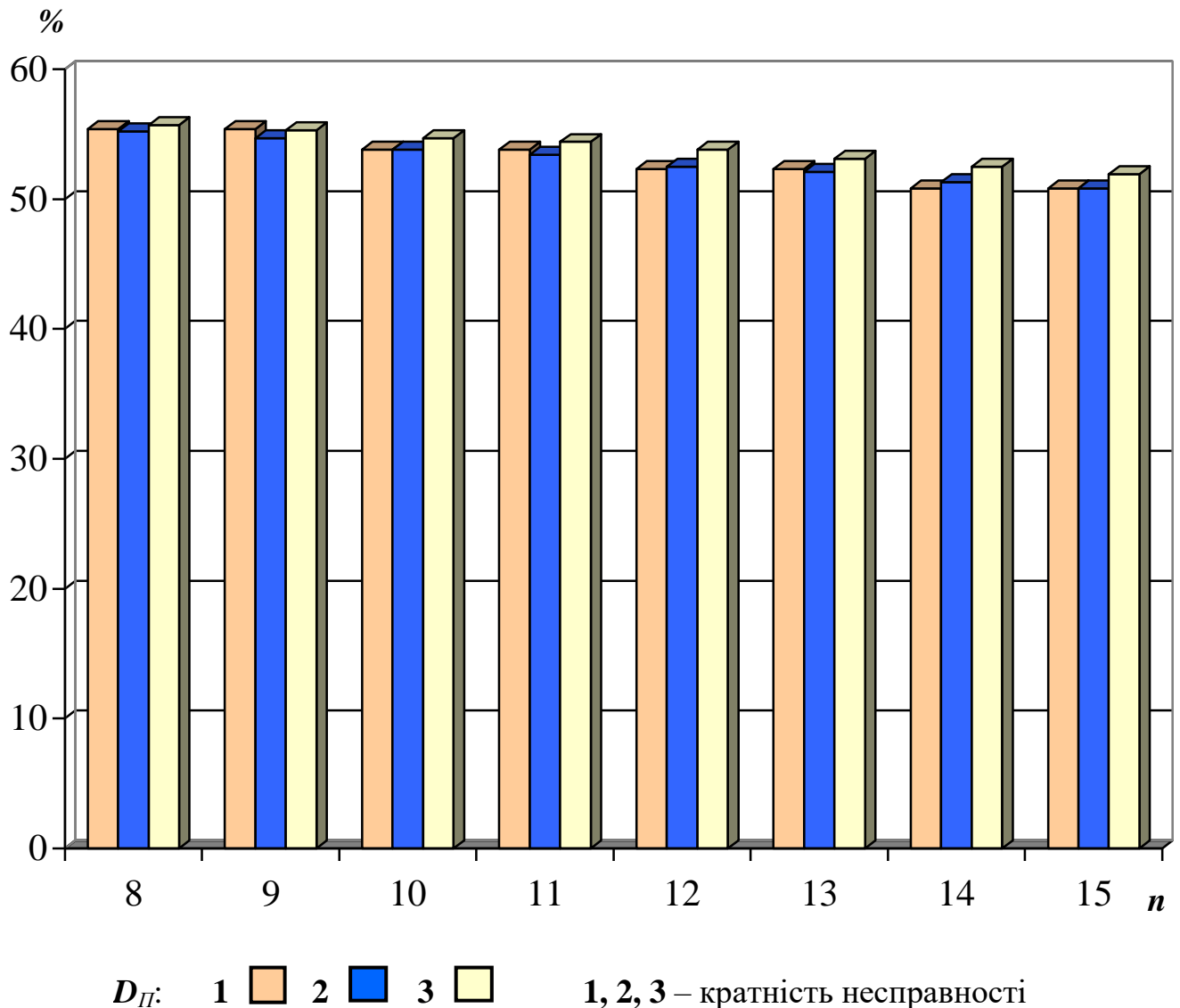


Рисунок 4.7 – Діаграми достовірності D_{Π} контролю повних добутоків при кратних несправностях

Достовірність контролю повного добутку майже не змінюється із збільшенням кратності несправності та зменшується при зростанні розрядності n операндів.

На рис. 4.8 показані діаграми зміни достовірності d_y контролю усічених добутоків за модулем три для кількості вірних розрядів $n_6 = n$ результату матричного помножувача мантис та кратності несправності 1, 2, 3 від ряду значень розрядності n .

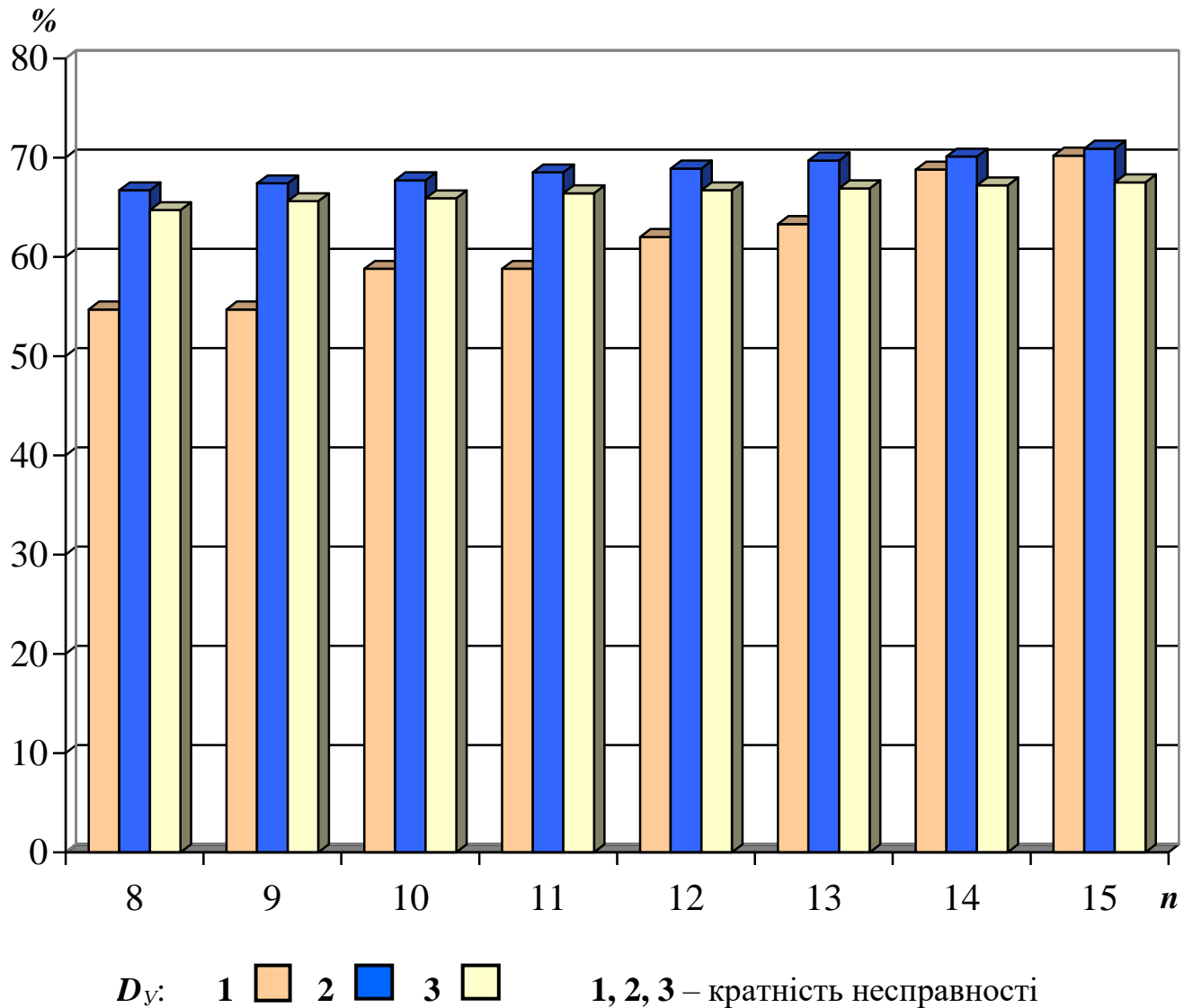


Рисунок 4.8 – Діаграми достовірності D_y контролю усічених добутоків при кратних несправностях

Достовірність контролю за модулем усічених добутоків підвищується при кратності несправності 2, у порівнянні з якою незначно зменшується при кратності 3. Із зростанням розрядності n достовірність контролю усічених добутоків зростає.

На рис. 4.9 показані діаграми зміни ефективності r контролю добутоків за модулем три при скороченні множення для кількості вірних розрядів $n_e = n$ результату матричного помножувача мантис та кратності несправності 1, 2, 3 від ряду значень розрядності n .

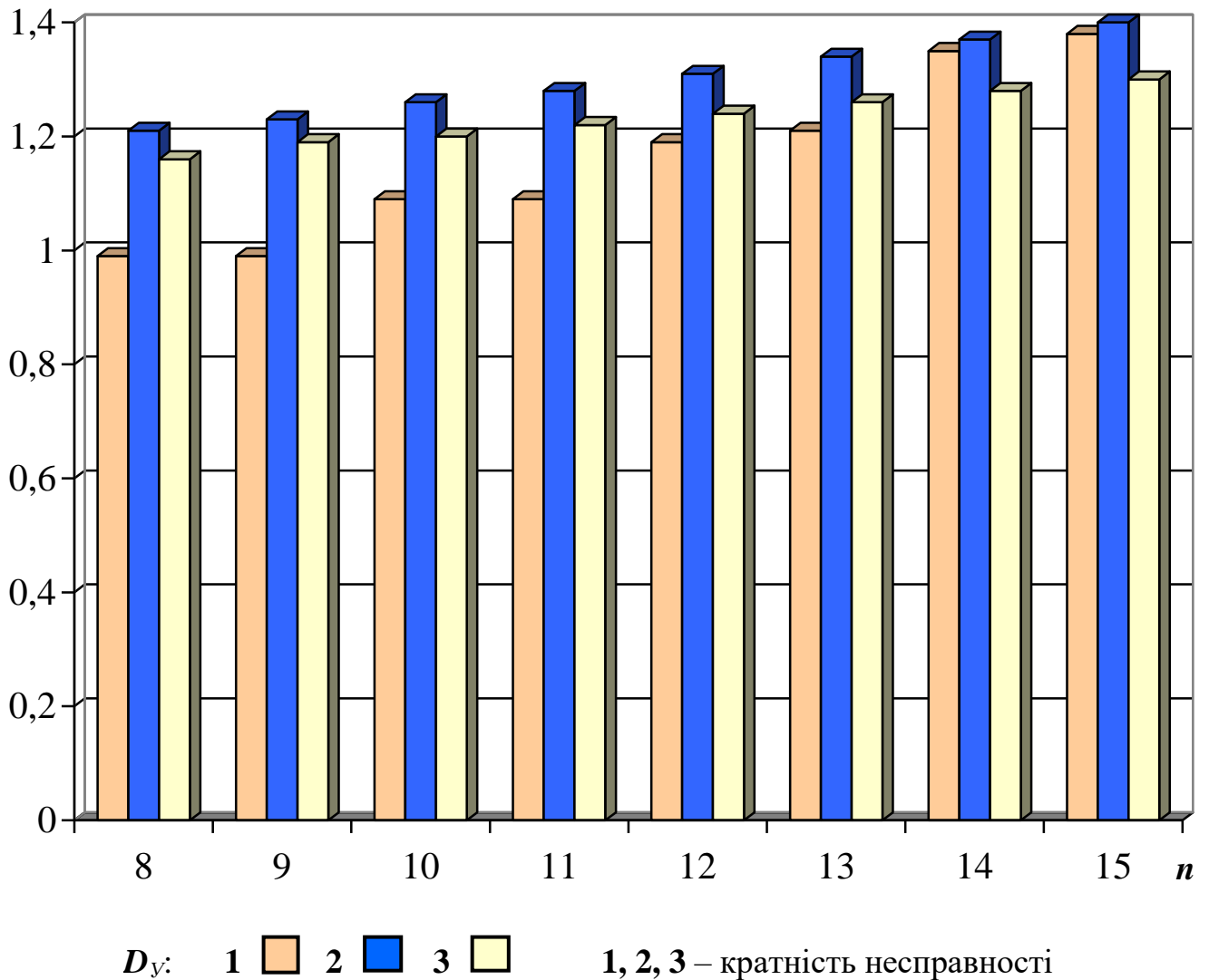


Рисунок 4.9 – Діаграми ефективності R контролю усічених добутоків при кратних несправностях

Ефективність контролю r добутоків при скороченні обчислень досягає найбільших значень при кратності несправності 2. Із зростанням розрядності n ефективність скорочення обчислень в достовірності контролю зсічених добутоків зростає.

4.2.4 на рис. 4.10 показані діаграми зміни ймовірностей $p_{всп}$ та $p_{всу}$ виявлення суттєвої помилки в доповняльному коді повного та зсіченого добутків для кратності несправності 1, 2, 3 та ряду значень розрядності n матричних помножувачів мантис і кількості вірних розрядів $n_в = n - 6$.

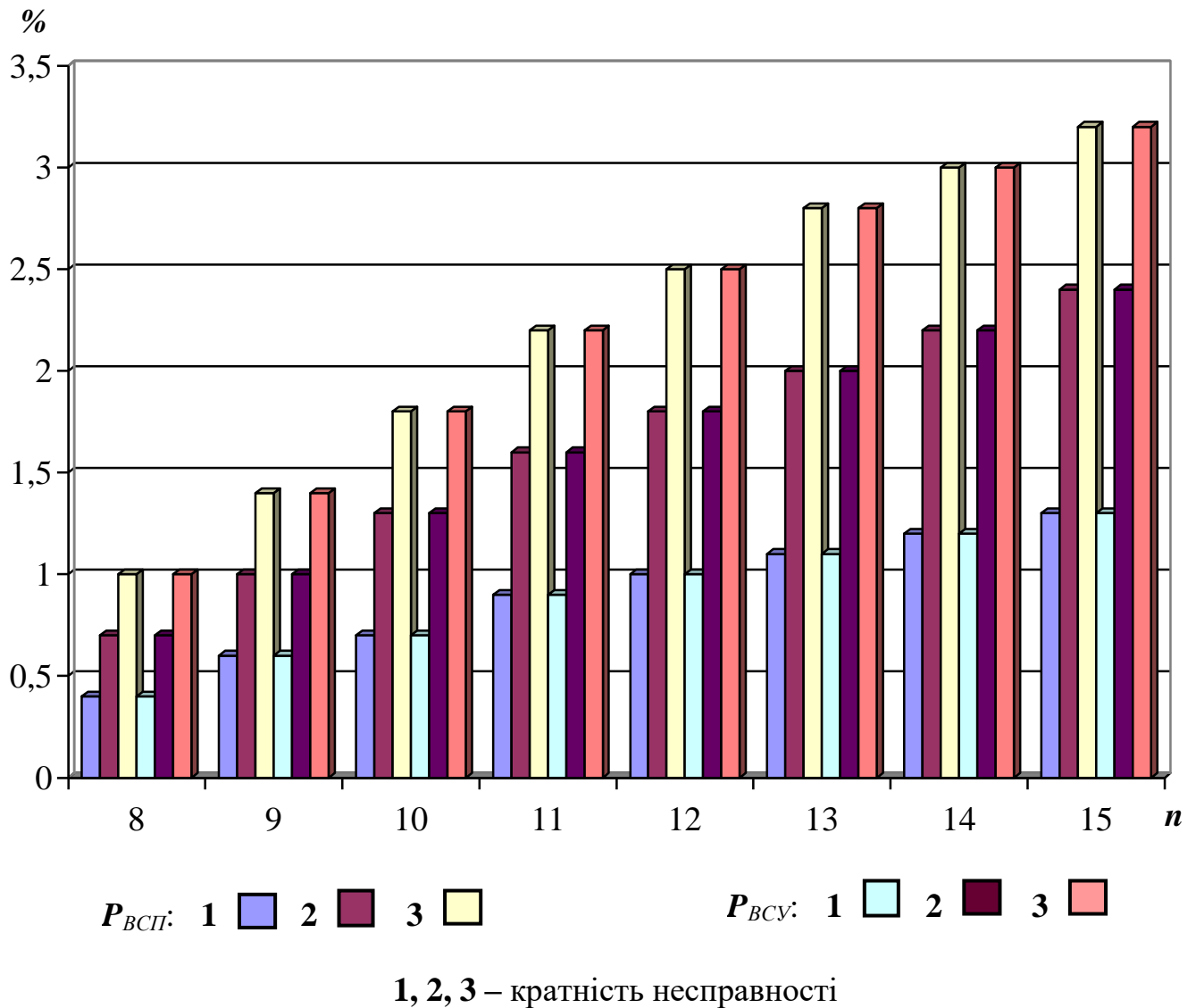


Рисунок 4.10 – Діаграми ймовірностей P_{VSP} і P_{VSU} виявлення суттєвої помилки в повному та усіченому добутку при різних кратностях несправності

Діаграми показують, що ймовірності виявлення суттєвої помилки в повному й усіченому добутках, представлених в доповняльних кодах, практично збігаються та істотно підвищуються із зростанням кратності несправності та при збільшенні розрядності n .

На рис. 4.11 показані діаграми зміни ймовірностей $p_{внп}$ та $p_{вну}$ виявлення несуттєвої помилки в доповняльному коді повного та зсіченого добутоків для кратності несправності 1, 2, 3 та ряду значень розрядності n матричних помножувачів мантис і кількості вірних розрядів $n_6 = n - 6$.

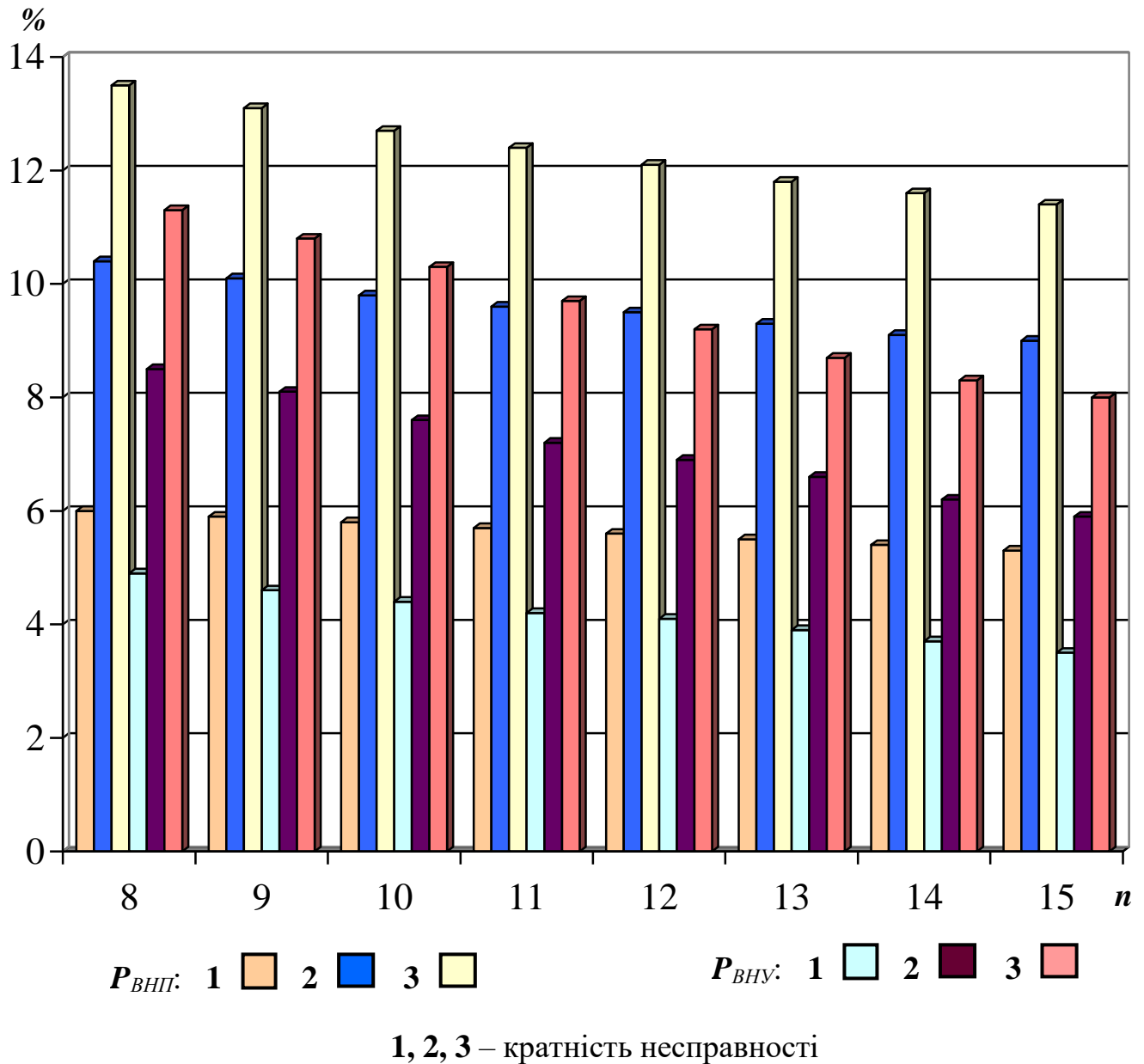


Рисунок 4.11 – Діаграми ймовірностей $P_{ВНП}$ і $P_{ВНУ}$ виявлення несуттєвої помилки в повному та усіченому добутках при кратних несправностях

Ймовірності виявлення несуттєвої помилки в повному добутку більші ніж в усіченому, разом зменшуються при збільшенні розрядності n та разом підвищуються із зростанням кратності несправності.

На рис. 4.12 показані діаграми зміни ймовірностей $p_{вп}$ та $p_{ву}$ виявлення помилки в доповняльному коді повного та зсіченого добутків для кратності несправності 1, 2, 3 та ряду значень розрядності n матричних помножувачів мантис і кількості вірних розрядів $n_6 = n - 6$.

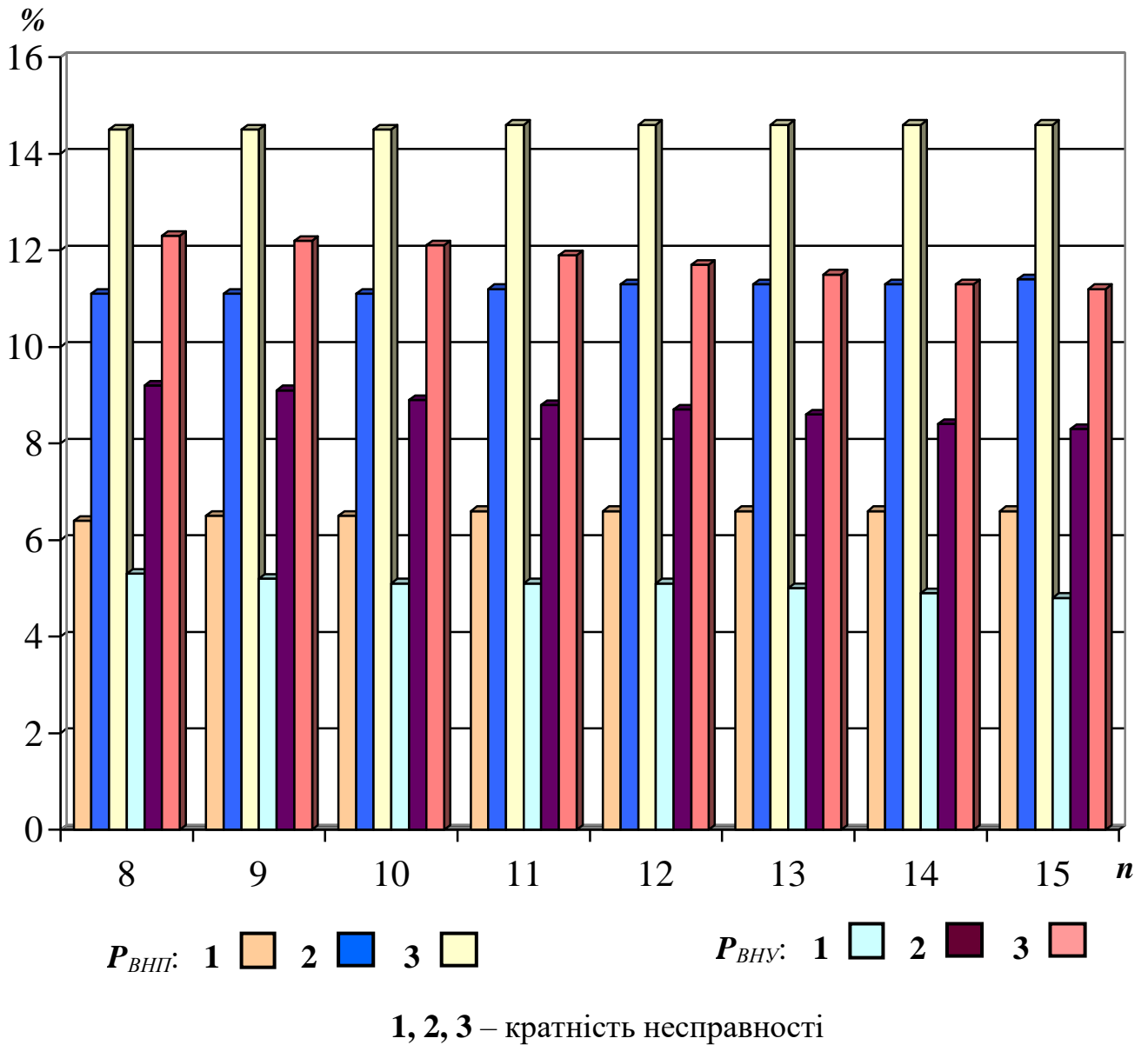


Рисунок 4.12 – Діаграми ймовірностей $P_{ВП}$ і $P_{ВУ}$ виявлення помилки в повному та усіченому добутках при кратних несправностях

Ймовірності виявлення помилки в повному добутку більші ніж в усіченому, разом майже не змінюються при збільшенні розрядності n та разом підвищуються із зростанням кратності несправності.

На рис. 4.13 показані діаграми зміни достовірності d_n контролю повних добутоків за модулем три для кількості вірних розрядів $n_e = n - 6$ результату матричного помножувача мантис та кратності несправності 1, 2, 3 від ряду значень розрядності n .

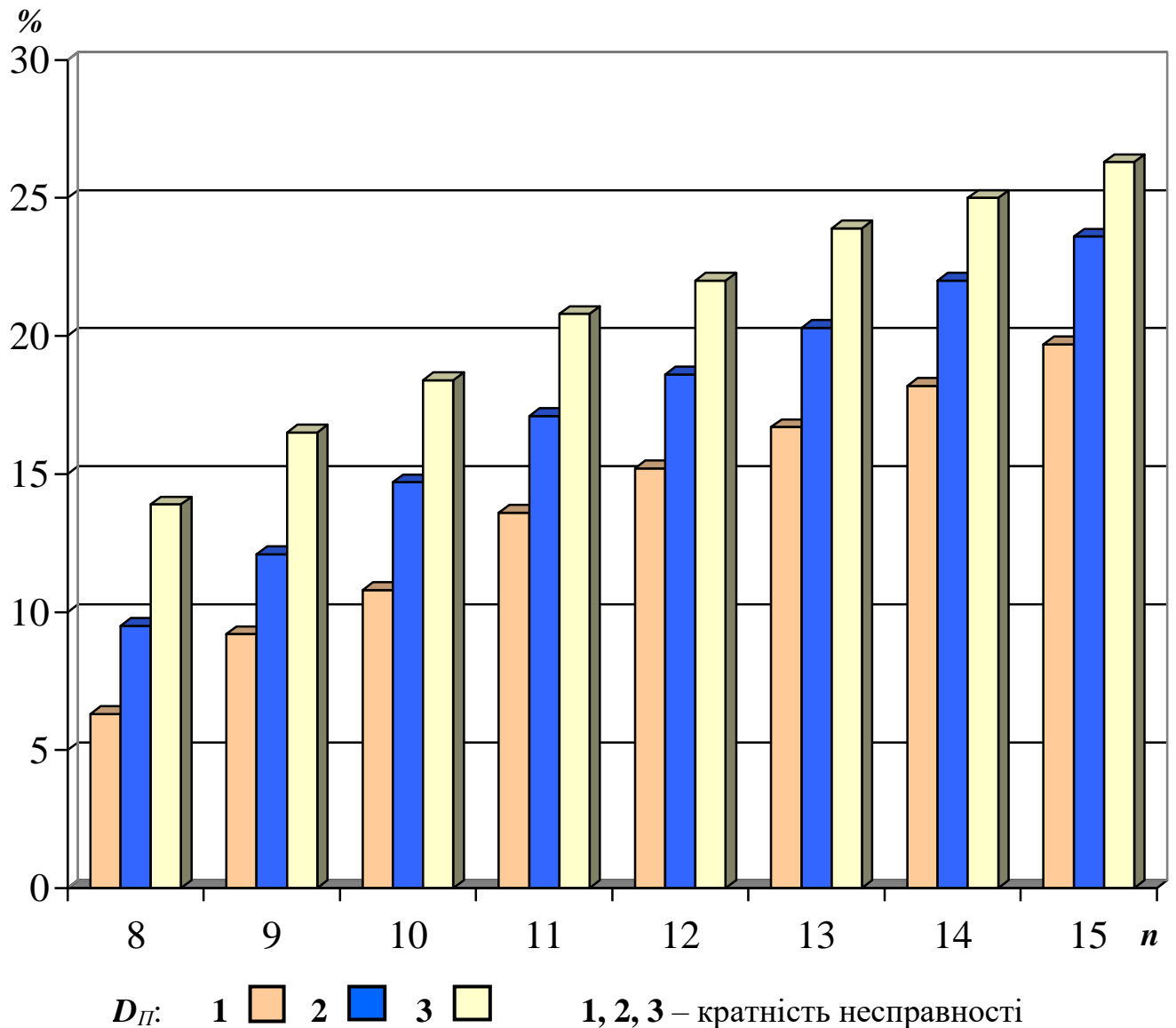


Рисунок 4.13 – Діаграми достовірності D_n контролю повних добутоків при кратних несправностях

Достовірність контролю повного добутку підвищується із збільшенням кратності несправності та при зростанні розрядності n операндів, однак значно менша за достовірність контролю для кількості вірних розрядів $n_e = n$.

На рис. 4.14 показані діаграми зміни достовірності d_y контролю усічених добутоків за модулем три для кількості вірних розрядів $n_e = n - 6$ результату матричного помножувача мантис та кратності несправності 1, 2, 3 від ряду значень розрядності n .

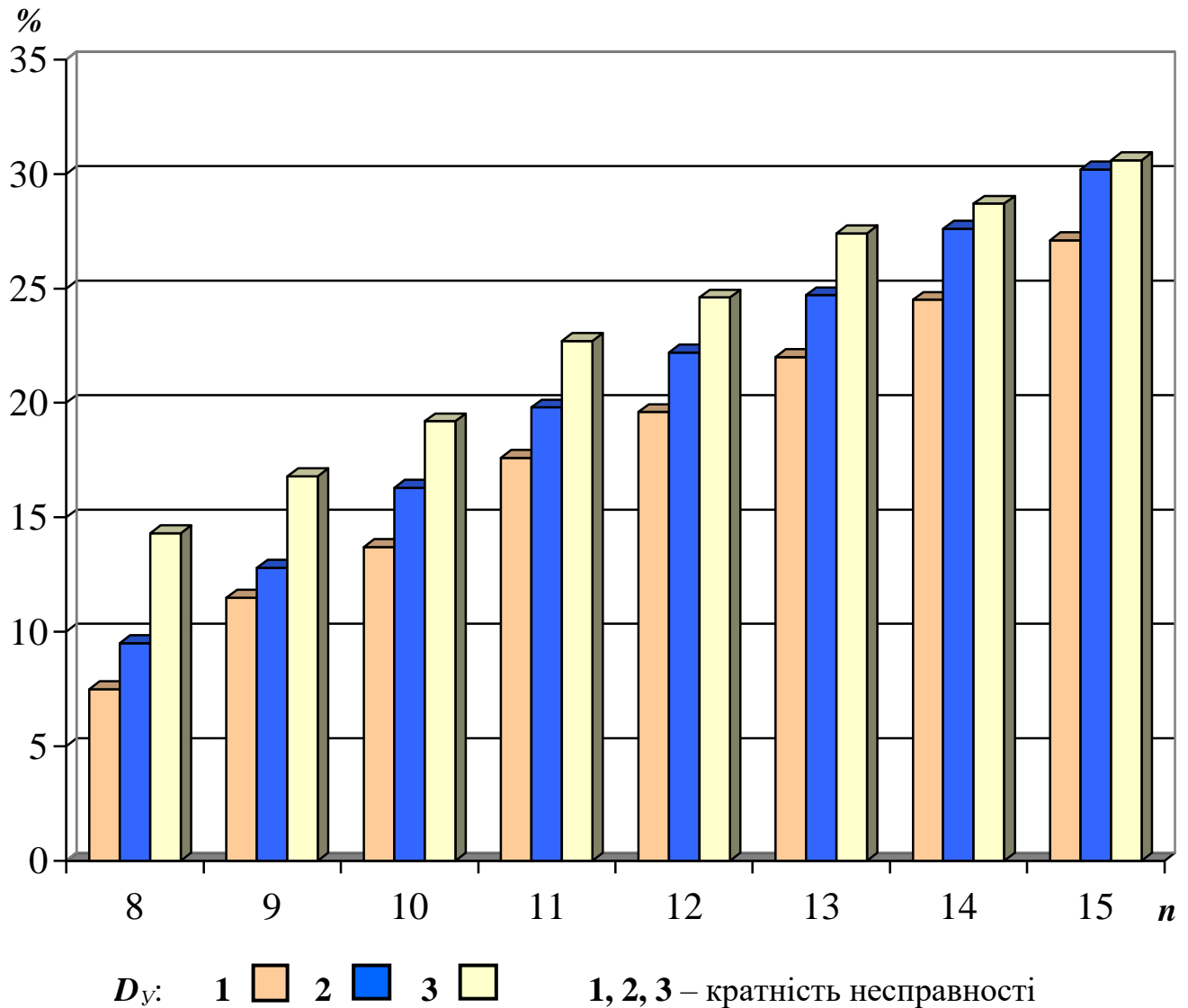


Рисунок 4.14 – Діаграми достовірності D_y контролю усічених добутоків при кратних несправностях

Достовірність контролю за модулем усічених добутоків підвищується із зростанням розрядності n та при збільшенні кратності несправності, однак досягає значно менших значень у порівнянні з достовірністю контролю для кількості вірних розрядів $n_e = n$.

На рис. 4.15 показані діаграми зміни ефективності r контролю добутоків за модулем три при скороченні множення для кількості вірних розрядів $n_6 = n - 6$ результату матричного помножувача мантис та кратності несправності 1, 2, 3 від ряду значень розрядності n .

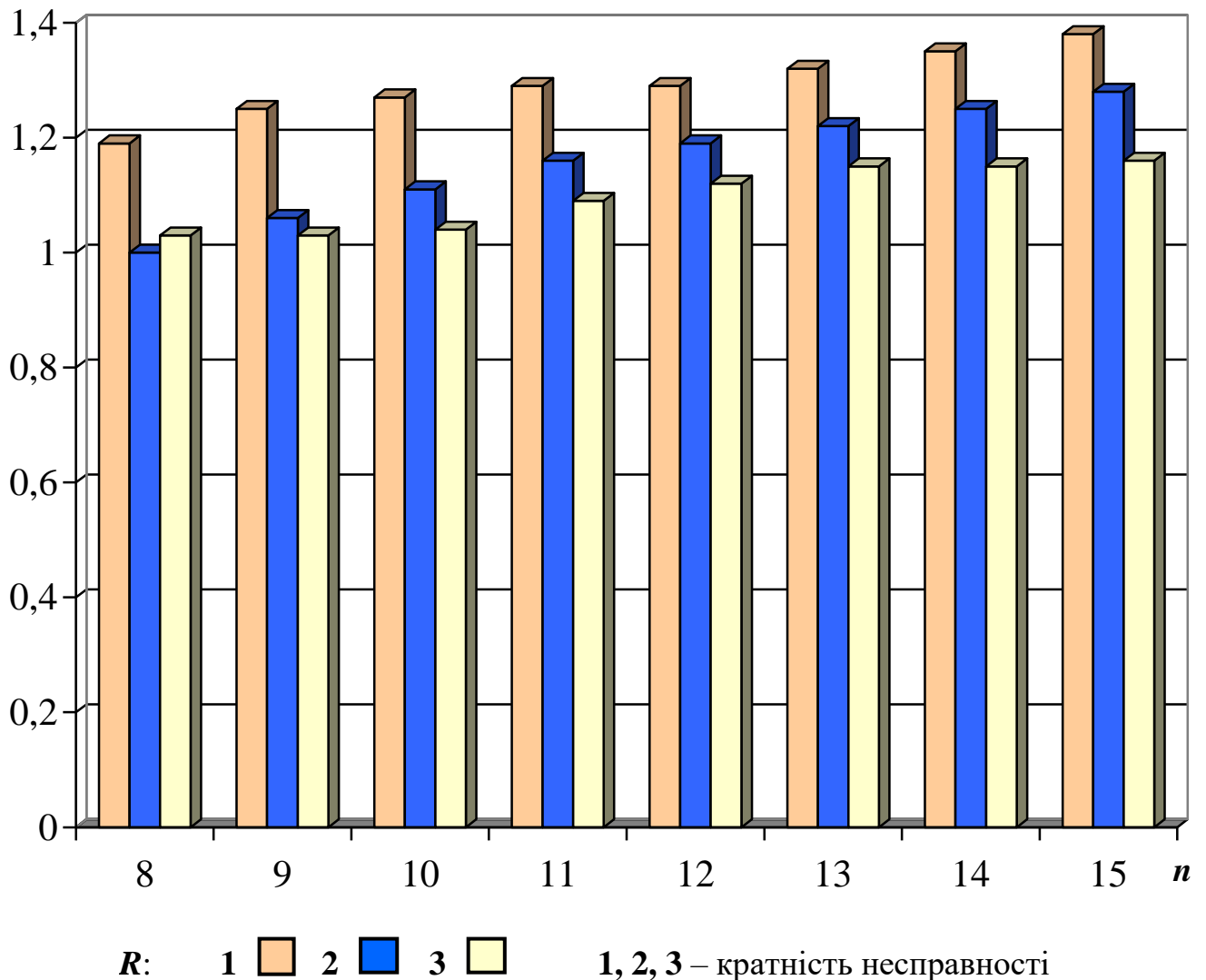


Рисунок 4.15 – Діаграми ефективності R контролю усічених добутоків при кратних несправностях

Ефективність r контролю добутоків при скороченні обчислень в достовірності підвищується із зростанням розрядності n та зменшується при збільшенні кратності несправності (за виключенням ефективності контролю двократних несправностей при розрядності n).

4.3 Висновки

Дослідження, виконані на програмній моделі, показали залежності ймовірностей виявлення та пропуску суттєвих та несуттєвих помилок, а також помилок загалом для повного та скороченого виконання операції у доповняльному коді в матричному помножувачі, які вони демонструють при зростанні розрядності операндів і кратності несправності типу замикання двох точок схеми операційного елемента та зменшенні кількості вірних розрядів результату обчислень.

Результати моделювання обчислень та контролю в матричному помножувачі мантис показали, що метод контролю за числовим модулем три виявляє всі помилки, що викликаються типовими однократними несправностями замикання двох точок в операційному елементі матричного помножувача мантис у доповняльному коді. Із зростанням кратності несправності з'являються невиявлені помилки, які впливають на достовірність контролю у залежності від їх суттєвості. Виявлення суттєвих помилок підвищує достовірність контролю, а виявлення несуттєвих – знижує, відбраковуючи помилкові, але достовірні результати обчислень.

Діаграми залежностей, що одержані за результатами моделювання матричного помножувача мантис та його контролю показують, що ймовірності виявлення суттєвої помилки в повному й усіченому добутках практично збігаються та істотно підвищуються із зростанням кратності несправності. Ці ймовірності багаторазово знижуються при зменшенні кількості вірних розрядів, незначно знижуються та істотно підвищуються відповідно при $n_B = n$ та $n_B = n - 6$ в умовах збільшення розрядності n .

Ймовірності виявлення несуттєвої помилки в повному добутку більші ніж в усіченому, перші зростають при $n_B = n$ та всі разом у інших випадках зменшуються при збільшенні розрядності n і підвищуються із зростанням кратності несправності. Ймовірності виявлення несуттєвої помилки підвищуються при зменшенні кількості вірних розрядів.

Ймовірності виявлення помилки в повному добутку більші ніж в усіченому, майже не змінюються при збільшенні розрядності n (за виключенням P_{BY} , що зменшується) та разом підвищуються із зростанням кратності несправності. Ймовірності виявлення помилки не залежать від кількості вірних та невірних розрядів результату обчислень.

Достовірність контролю повного добутку мантис майже не змінюється із збільшенням кратності несправності та зменшується при зростанні розрядності n операндів у випадку найбільшої кількості вірних розрядів $n_B = n$. Для $n_B = n - 6$ достовірність підвищується із збільшенням кратності несправності та при зростанні розрядності n операндів, однак зменшення кількості вірних розрядів істотно знижує її значення.

Достовірність контролю усічених добутків мантис перевищує її оцінку для повних добутків та підвищується при кратності несправності 2, у порівнянні з якою незначно зменшується при кратності 3 в умовах найбільшої кількості вірних розрядів $n_B = n$. Із зростанням розрядності n достовірність контролю усічених добутків підвищується. Для $n_B = n - 6$ достовірність підвищується при збільшенні кратності несправності, однак досягає значно менших значень у порівнянні з достовірністю для кількості вірних розрядів $n_B = n$.

Ефективність R контролю в скороченні обчислення добутків мантис є найбільшою при кратності несправності 2 при $n_B = n$ та зменшується із збільшенням кратності несправності у випадку $n_B = n - 6$. Із зростанням розрядності n ефективність скорочення обчислень в достовірності контролю усічених добутків підвищується.

Висновок

Розповсюдження об'єктів підвищеного ризику, таких як енергоблоки теплових, атомних та гідроелектростанцій, швидкісний наземний та повітряний транспорт, літальні та космічні апарати, а також обслуговуючі їх інфраструктури, постійне підвищення їх складності та потужності підвищує вимоги до функціональної безпеки інформаційно-управляючих систем критичного застосування, що спрямовані на запобігання техногенних аварій та зниження їх наслідків, у разі якщо аварії трапляються.

Одна з проблем, що безпосередньо важлива для функціональної безпеки інформаційно-управляючих систем критичного застосування та їх компонентів пов'язана з двома режимами роботи та можливістю накопичення множини прихованих несправностей протягом довготривалого нормального режиму в умовах відсутності робочих послідовностей вхідних даних, які здатні проявити ці несправності.

З початком аварійного режиму множина накопичених несправностей може водночас проявлятися, створюючи додатково проблему кратних несправностей в найбільш відповідальному аварійному режимі роботи систем критичного забезпечення.

Необхідність враховувати можливість такого перебігу подій визначає актуальність досліджень щодо методу робочого діагностування з контролем схем цифрових компонентів, працюючих у складі інформаційно-управляючих систем критичного застосування в умовах дії кратних несправностей.

За цифровий компонент вибраний одноклітинний матричний помножувач мантис, що виконує ключову операцію наближених обчислень з використанням найбільш продуктивних та розповсюджених матричних структур операційних елементів.

Робоче діагностування використовує найбільш ефективний та визнаний для арифметичних операцій контроль за числовим модулем три.

Обробка мантис в однотоактному матричному помножувачі виконується в доповняльному коді у двох варіантах: з повним і скороченим множенням.

Розглянута типова для сучасних інтегральних схем несправність, що проявляється у замиканні двох точок схеми операційного елемента. Множина таких несправностей включає також константні несправності як замикання точок схеми на рівень логічного нуля або одиниці.

Для проведення досліджень розроблена програмна модель однотоактного матричного помножувача мантис та його системи робочого діагностування з використанням методу контролю за числовим модулем три.

Розроблена програмна модель забезпечує підготовку та проведення моделювання роботи однотоактного матричного помножувача мантис у доповняльному коді та його системи робочого діагностування в умовах дії кратних несправностей.

Моделювання виконується для різної розрядності операндів та кількості вірних розрядів результатів на послідовностях випадкових даних із введенням кратних несправностей замикання довільного виду у випадково вибраних операційних елементах матричного помножувача.

За результатами моделювання досліджено ймовірності появи, виявлення та пропуску суттєвих, несуттєвих для достовірності результатів та всіх помилок, що викликаються однократними, двократними та трикратними несправностями матричного помножувача мантис у доповняльному коді при виконанні ним повної та скороченої операції.

Визначено залежності названих ймовірностей від розрядності однотоактного матричного помножувача, та такі залежності для достовірності контролю обчислюваних результатів за числовим модулем три та ефективності контролю в скороченні операцій за відношенням значень, одержаних для достовірності контролю повних та зсічених добутоків мантис.

Ймовірності виявлення суттєвої помилки в повному й усіченому добутках мантис практично збігаються між собою та істотно підвищуються із зростанням

значень кратності несправності й багаторазово знижуються при зменшенні кількості вірних розрядів.

Збіг ймовірностей виявлення суттєвої помилки в повному й усіченому добутках можна пояснити тим, що суттєва помилка визначається у вірних розрядах, кількість яких є однаковою для повного та скороченого виконання операції множення мантис. Підвищення ймовірностей виявлення суттєвої помилки із зростанням значень кратності несправності є новим неочікуваним результатом моделювання. Зниження ймовірностей виявлення суттєвої помилки при зменшенні кількості вірних розрядів пояснюється скороченням частини матриці помножувача мантис, за якими обчислюється множина вірних розрядів.

Ймовірності виявлення несуттєвої помилки в повному добутку більші ніж в усіченому та разом підвищуються при зменшенні кількості вірних розрядів і в умовах зростання кратності несправності, що пояснюється меншою кількістю невірних розрядів в усіченому добутку.

Достовірність контролю повного добутку мантис майже не змінюється із збільшенням кратності несправності та зменшується при зростанні розрядності операндів у випадку найбільшої кількості вірних розрядів. Зменшення кількості вірних розрядів істотно знижує її значення.

Достовірність контролю усічених добутків мантис перевищує її оцінку для повних добутків та підвищується із зростанням розрядності матричного помножувача.

Ефективність контролю в скороченні обчислення добутків підвищується в умовах зростання розрядності матричного помножувача мантис.

ПЕРЕЛІК ПОСИЛАНЬ

1. Безопасность критических инфраструктур: математические инженерные методы анализа и обеспечения / Под ред. Харченко В. С. – Министерство образования и науки Украины, Нац. аэрокосм. ун-т им. Н.Е. Жуковского «ХАИ», 2011. – 641 с.
2. Харченко, В. С. Новые информационные технологии и безопасность информационно–управляющих систем АЭС / В. С. Харченко, М. А. Ястребенецкий, В. В. Скляр // Ядерная и радиационная безопасность. – 2003. – Т. 6, № 2. – С. 19–28.
3. Харченко, В. С. Компьютерные сети информационных и управляющих систем АЭС, построенные на основе открытых стандартов / В. С. Харченко, А. В. Горбенко, В. В. Скляр // Ядерная и радиационная безопасность. – 2004. – Т. 7, № 4. – С. 80 – 87.
4. Харченко, В. С. Надежность и отказоустойчивость компьютерных средств и систем: учебное пособие / В. С. Харченко, И. В. Лысенко, О. М. Тарасюк. – Х.: ХАИ, 2007. – 44 с.
5. Gillis, D. The apocalypses that might have been / D. Gillis [Электронный ресурс] // DAMN Interesting. – 2007. – N 298. – Режим доступа: <http://www.popmech.ru/go.php?url=http%3A%2F%2Fwww.damninteresting.com%2F%3Fr%3D913>
6. Дрозд, М. О. Проблема прихованих несправностей цифрових компонентів систем критичного застосування / М. О. Дрозд // Науковий вісник Чернівецького університету. – 2014. – Т. 5, Вип. 1. – С. 28 – 33.
7. Проектирование и диагностирование важных для безопасности встроенных систем. Практикум / А.В. Дрозд, В.С. Харченко, С. А. Нестеренко, С.Г. Антощук, М.А. Дрозд / Под ред. Дрозда А.В., Харченко В.С. – Х.: Нац. аэрокосмический ун-т им. Н.Е. Жуковского «ХАИ», 2013. – 111 с.
8. ДСТУ 2868-94. Надійність техніки. Основні терміни та визначення. – К.: Держстандарт України, 1994. – 19 с.

9. IEEE Std 754™-2008 (Revision of IEEE Std 754-1985) IEEE Standard for Floating-Point Arithmetic. IEEE 3 Park Avenue New York, NY 10016-5997, USA (2008)

10. Дрозд, А. Особенности рабочего диагностирования вычислительных устройств на современном этапе развития / А. Дрозд, М. Саид, М. Дрозд, Г. Дмитриев // Годишник на Технически университет – Варна, 2010 г.–С.113 –118.

11. Дрозд О. В. Контроль за модулем обчислювальних пристроїв. Навч. посібн. для студ. спеціальності 7.091501 – «Комп’ютерні та інтелектуальні системи та мережі» / Одеськ. нац. політехн. ун-т.–Одеса: АО Бахва, 2002.–144 с.

12. Рабинович З. Л., Раманаускас В. А. Типовые операции в вычислительных машинах. – Киев: Техника, 1980. – 264 с.

13. Рабочее диагностирование безопасных информационно-управляющих систем / А.В. Дрозд, В.С. Харченко, С.Г. Антощук, Ю.В. Дрозд, М.А. Дрозд, Ю.Ю. Сулима / Под ред. Дрозда А.В., Харченко В.С. – Х.: Нац. аэрокосмический ун-т им. Н.Е. Жуковского «ХАИ», 2012. – 614 с.

14. Савельев А. Я. Прикладная теория цифровых автоматов. – М.: Высш. шк., 1987. – 272 с.

15. Щербаков Н. С. Достоверность работы цифровых устройств. – М.: Машиностроение, 1989. – 224 с.

16. Дрозд А. В. Достоверность рабочего диагностирования вычислительных устройств для обработки приближенных данных // Зб. наукових праць Харківського військового ун-ту НАНУ. Системи обробки інформації. – Харків, 2002. – Вип. 4 (20). – С. 8 – 13.

17. Drozd A. On-line testing of computing circuits at approximate data processing // Радіоелектроніка та інформатика. 2003. № 3. – С. 113 – 116.

18. Дрозд О.В., Боговик О.С, Горячко М.С., Довгань Я.О., Мовсесян А.М. Проблема достовірності результатів в арифметичних компонентах систем критичного застосування // Матеріали ІХ Міжнародної науково-практичної конференції «Інформаційні управляючі системи і технології» (ІУСТОДЕСА-2021). – Одеса, 2021. – С. 109 – 112.

Програмна модель для дослідження методу контролю

```
unit Unit1;

interface

uses
  Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,
  Dialogs, StdCtrls, ExtCtrls, Menus;

type
  TForm1 = class(TForm)
    Button1: TButton;
    Button2: TButton;
    Image1: TImage;
    Button3: TButton;
    Button4: TButton;
    Panel1: TPanel;
    Button5: TButton;
    Button6: TButton;
    Edit1: TEdit;
    Edit2: TEdit;
    Panel2: TPanel;
    Panel3: TPanel;
    Edit3: TEdit;
    Edit4: TEdit;
    Panel4: TPanel;
    Button7: TButton;
    MainMenu1: TMainMenu;
    A1: TMenuItem;
    A2: TMenuItem;
    N1: TMenuItem;
    N2: TMenuItem;
    N3: TMenuItem;
    B1: TMenuItem;
    N4: TMenuItem;
    N5: TMenuItem;
    N6: TMenuItem;
    N7: TMenuItem;
    Panel5: TPanel;
    Panel6: TPanel;
    Panel7: TPanel;
    Button8: TButton;
    Button9: TButton;
    Button10: TButton;
    Button11: TButton;
    Panel8: TPanel;
    N8: TMenuItem;
    N9: TMenuItem;
    N10: TMenuItem;
```



```
Edit9: TEdit;
Button12: TButton;
Button13: TButton;
Panel12: TPanel;
Button14: TButton;
Panel13: TPanel;
Panel14: TPanel;
Edit8: TEdit;
Edit10: TEdit;
Panel18: TPanel;
Panel19: TPanel;
Panel20: TPanel;
Panel21: TPanel;
Image2: TImage;
Panel22: TPanel;
Panel29: TPanel;
Panel30: TPanel;
Panel31: TPanel;
Panel24: TPanel;
Panel25: TPanel;
Panel26: TPanel;
Panel27: TPanel;
Panel28: TPanel;
Panel23: TPanel;
Panel32: TPanel;
Panel33: TPanel;
Button15: TButton;
Panel34: TPanel;
Panel35: TPanel;
procedure Button1Click(Sender: TObject);
procedure Button2Click(Sender: TObject);
procedure Button5Click(Sender: TObject);
procedure Button6Click(Sender: TObject);
procedure OnMouseDown(Sender: TObject; Button: TMouseButton;
  Shift: TShiftState; X, Y: Integer);
procedure Button3Click(Sender: TObject);
procedure Button4Click(Sender: TObject);
procedure Button7Click(Sender: TObject);
procedure Button8Click(Sender: TObject);
procedure Button9Click(Sender: TObject);
procedure Button10Click(Sender: TObject);
procedure Button11Click(Sender: TObject);
procedure A2Click(Sender: TObject);
procedure N1Click(Sender: TObject);
procedure N2Click(Sender: TObject);
procedure N3Click(Sender: TObject);
procedure N4Click(Sender: TObject);
procedure N5Click(Sender: TObject);
procedure N6Click(Sender: TObject);
procedure N7Click(Sender: TObject);
procedure N9Click(Sender: TObject);
procedure N10Click(Sender: TObject);
```

```

procedure Button13Click(Sender: TObject);
procedure Button12Click(Sender: TObject);
procedure Button14Click(Sender: TObject);
procedure FormDragOver(Sender, Source: TObject; X, Y: Integer;
  State: TDragState; var Accept: Boolean);
procedure FormDragDrop(Sender, Source: TObject; X, Y: Integer);
procedure Panel22MouseDown(Sender: TObject; Button: TMouseButton;
  Shift: TShiftState; X, Y: Integer);
procedure Panel23Click(Sender: TObject);
procedure Panel21Click(Sender: TObject);
procedure Panel12Click(Sender: TObject);
procedure Button15Click(Sender: TObject);

private
  { Private declarations }
public
  { Public declarations }
end;

const
  nh=15; nf=11;
var
  Form1: TForm1;
  Ma:array[1..nh+2,1..nh]of integer;
  Mb:array[1..nh+2,1..nh]of integer;
  Mc:array[1..nh+2,1..nh]of integer;
  Ma1:array[1..nh+2,1..nh]of integer;
  Mb1:array[1..nh+2,1..nh]of integer;
  Ma2:array[1..nh-1]of integer;
  Mb2:array[1..nh-1]of integer;
  Ms:array[1..nh+2,1..nh]of integer;
  Mp:array[1..nh+2,1..nh]of integer;
  AAb:array[1..nh]of integer;
  BBb:array[1..nh]of integer;
  PPb:array[1..2*nh]of integer;
  PPe:array[1..2*nh]of integer;
  TTb:array[1..2*nh]of integer;
  TTe:array[1..2*nh]of integer;
  MF:array[1..nf]of string;
  Fr:array[1..3]of integer;
  H4:array[1..3]of integer;
  H5:array[1..3]of integer;
  Ce:array[1..3]of integer;
  Ee:array[1..3]of integer;
  Ep:array[1..3]of integer;
  Et:array[1..3]of integer;
  Dep:array[1..3]of integer;
  Det:array[1..3]of integer;
  Dnp:array[1..3]of integer;
  Dnt:array[1..3]of integer;

n,aa,bb,pp,pe,tt,te,sa,sb,sp:integer;

```

```

za,zb,f1,f2,ff,df1,df2,nfe,nfr:integer;
mo,ka,kb,kp,e1,e2,ca,cb,k,pu:integer;
cc,es:integer;
implementation

```

```
{ $R *.dfm }
```

```

procedure TForm1.Button2Click(Sender: TObject);
begin
  if CloseQuery then Close;
end;

```

```

procedure TForm1.Button1Click(Sender: TObject);
  var i,j,h1,h2,h3:integer;
begin
  Button2.Top:=0; Button2.Left:=0;
  repeat
    Application.ProcessMessages;
    n:=StrToInt(Edit9.Text);
    if (n<2)or(n>15) then
      Panel34.Caption:=
        'Неправильна розрядність '+IntToStr(n);
    until (n>1)and(n<16);
    Panel34.Left:=12; Panel34.Top:=60;
    Panel34.Width:=185;
    Panel34.Caption:=
      'Розрядність помножувача '+IntToStr(n);
    Button1.Visible:=False;
    Edit9.Visible:=False;
    if n<7 then
      begin
        Panel34.Left:=Panel34.Left+420;
        Panel5.Left:=Panel5.Left+450; Panel6.Left:=Panel6.Left+450;
        Panel7.Left:=Panel7.Left+450; Panel8.Left:=Panel8.Left+450;
        Button8.Left:=Button8.Left+450; Button9.Left:=Button9.Left+450;
        Button10.Left:=Button10.Left+450; Button11.Left:=Button11.Left+450;
        Button12.Left:=Button12.Left+450; Button13.Left:=Button13.Left+450;
        Button14.Left:=Button14.Left+450;
      end;
    MF[1]:='a'; MF[2]:='b'; MF[3]:='c';
    MF[4]:='s'; MF[5]:='p'; MF[6]:='a1';
    MF[7]:='b1'; MF[8]:='a2'; MF[9]:='b2';
    MF[10]:='0'; MF[11]:='1';
    f1:=1; f2:=2; df1:=0; df2:=0;
    k:=n; pu:=0; nfr:=1; nfe:=0; es:=0;
    while (1 shl k)*(k-1)+1>(1 shl n) do
      k:=k-1;
    A1.Visible:=True;
    B1.Visible:=True;
    N8.Visible:=True;
    Panel12.Visible:=True;
    Image1.Left:=8;

```

```
Image1.Top:=56;
```

```

Image1.Width:=32*(2*n+1);      Image1.Height:=32*(n+7);
Image2.Left:=8;                Image2.Top:=56;
Image2.Width:=32*n+288;       Image2.Height:=384;
if n<7 then
begin
  Image1.Width:=620;
  Image2.Width:=620; Image2.Height:=400;
end;
Image1.Canvas.Rectangle(32*n+4,4,64*n-4,28);
Image1.Canvas.Rectangle(64*n+4,36,64*n+28,32*n+28);
Image1.Canvas.Rectangle(4,32*n+36+64,64*n-4,32*n+64+60);
Image1.Canvas.Rectangle(4,32*n+68+64,64*n-4,32*n+64+92);
Image1.Canvas.Rectangle(4,32*n+100+64,32*(2*n-k)-4,32*n+64+124);
Image1.Canvas.Rectangle(4,32*n+132+64,32*(2*n-k)-4,32*n+64+156);
Image1.Canvas.Rectangle(32*n-28,4,32*n-4,28);
Image1.Canvas.Rectangle(64*n+4,32*n+36,64*n+28,32*n+60);
Button3.Top:=60;      Button3.Left:=32*n-52;
Button4.Top:=60;      Button4.Left:=64*n+12;
Panel1.Top:=32*n+156; Panel1.Left:=64*n+12;
Panel2.Top:=32*n+188; Panel2.Left:=64*n+12;
Panel19.Top:=32*n+220; Panel19.Left:=32*(2*n-k)+12;
Panel20.Top:=32*n+252; Panel20.Left:=32*(2*n-k)+12;
Button3.Visible:=True; Button4.Visible:=True;
Button5.Visible:=True; Button6.Visible:=True;
Button7.Visible:=True; Button8.Visible:=True;
Button9.Visible:=True; Button10.Visible:=True;
Button11.Visible:=True; Button12.Visible:=True;
Button13.Visible:=True; Button14.Visible:=True;
Panel1.Visible:=True; Panel2.Visible:=True;
Panel3.Visible:=True; Panel4.Visible:=True;
Panel5.Visible:=True; Panel6.Visible:=True;
Panel7.Visible:=True; Panel8.Visible:=True;
Panel12.Visible:=True; Panel13.Visible:=True;
Panel14.Visible:=True; Panel19.Visible:=True;
Panel20.Visible:=True; Panel22.Visible:=True;
Edit1.Visible:=True; Edit2.Visible:=True;
Edit3.Visible:=True; Edit4.Visible:=True;
Edit8.Visible:=True; Edit10.Visible:=True;
Panel22.Top:=32*(n+8)+16; Panel22.Left:=32*n+4;
for i:=1 to n do
  for j:=1 to n-1 do
    begin
      h1:=64*n-32*(i+j)+8;
      with Image1.Canvas do
        begin
          if i+j>k then
            Brush.Color:=clMoneyGreen
          else
            Brush.Color:=clSilver;
          Rectangle(h1,32*i+8,h1+16,32*i+24);
          TextOut(h1+1,32*i+9,IntToStr(i)+'.'+IntToStr(j+1));
        end;
    end;
  end;
end;

```

```

end;
for i:=1 to n do
begin
h1:=64*n-32*i+8; h2:=32*(n+2); h3:=32*n;
AAb[i]:=0;
BBb[i]:=0;
with Image1.Canvas do
begin
Brush.Color:=clMoneyGreen;
Rectangle(h1-h3,h3+40,h1+16-h3,h3+56);
TextOut(h1+1-h3,h3+41,IntToStr(n+1)+IntToStr(i));
Rectangle(h1-h3,h3+72,h1+16-h3,h3+88);
TextOut(h1+1-h3,h3+73,IntToStr(n+2)+IntToStr(i));
Pen.Color:=clRed;
Brush.Color:=clLime;
Rectangle(h1-h3,h2+40,h1+16-h3,h2+56);
TextOut(h1+4-h3,h2+41,'0');
Rectangle(h1-h3,h2+72,h1+16-h3,h2+88);
TextOut(h1+4-h3,h2+73,'0');
Rectangle(h1-h3,h2+104,h1+16-h3,h2+120);
TextOut(h1+4-h3,h2+105,'0');
Rectangle(h1-h3,h2+136,h1+16-h3,h2+152);
TextOut(h1+4-h3,h2+137,'0');
Pen.Color:=clBlue;
Brush.Color:=clYellow;
Rectangle(h3-24,8,h3-8,24);
TextOut(h3-20,9,'0');
Rectangle(64*n+8,32*n+40,64*n+24,32*n+56);
TextOut(64*n+12,32*n+41,'0');
Pen.Color:=clBlue;
Brush.Color:=clMoneyGreen;
Rectangle(h1,8,h1+16,24);
TextOut(h1+4,9,'0');
Rectangle(64*n+8,32*i+8,64*n+24,32*i+24);
TextOut(64*n+12,32*i+9,'0');
Rectangle(h1,h2+40,h1+16,h2+56);
TextOut(h1+4,h2+41,'0');
Rectangle(h1,h2+72,h1+16,h2+88);
TextOut(h1+4,h2+73,'0');
if i>k then
begin
Rectangle(h1,h2+104,h1+16,h2+120);
TextOut(h1+4,h2+105,'0');
Rectangle(h1,h2+136,h1+16,h2+152);
TextOut(h1+4,h2+137,'0');
end;
end;
end;
end;
end;

procedure TForm1.Button7Click(Sender: TObject);
//Старт

```

```

var h1,h2,h3,ih,i,j,zz,pps:integer;
begin
if Button7.Caption='Старт' then
begin
Button2.Visible:=False;
if ((za>1) or (zb>1)) then Button7.Caption:='Фініш';
Randomize;
h2:=1 shl(n-1);
cc:=0;
for zz:=1 to 3 do
begin
Ce[zz]:=0; Ee[zz]:=0; Ep[zz]:=0; Et[zz]:=0;
Dep[zz]:=0; Dnp[zz]:=0; Det[zz]:=0; Dnt[zz]:=0;
end;
Edit3.Text:='0'; Edit4.Text:='0';
Edit8.Text:=' ';
Edit10.Text:=' ';
Panel21.Tag:=0;
ca:=1-(za shr 1); //признак окончання перебора A
if n<7 then h3:=120 else h3:=0;
if za=2 then aa:=h2;
repeat //перебор по A
cb:=1-(zb shr 1);
if za mod 2=1 then
begin
aa:=h2+Random(h2);
sa:=Random(2);
if sa=1 then aa:=-aa;
Edit1.Text:=IntToStr(aa);
if sa=1 then aa:=(1 shl n)+aa;
h1:=1;
for ih:=1 to n do
begin
AAb[ih]:=(aa and h1)shr(ih-1);
h1:=h1 shl 1;
Image1.Canvas.TextOut(64*n-32*ih+12,9,
IntToStr(AAb[ih]));
end;
end;
if zb=2 then bb:=h2;
repeat //перебор по B
if zb mod 2=1 then
begin
bb:=h2+Random(h2);
sb:=Random(2);
if sb=1 then bb:=-bb;
Edit2.Text:=IntToStr(bb);
if sb=1 then bb:=(1 shl n)+bb;
h1:=1;
for ih:=1 to n do
begin
BBb[ih]:=(bb and h1)shr(ih-1);

```

```

h1:=h1 shl 1;
Image1.Canvas.TextOut(64*n+12,32*ih+9, IntToStr(BBb[ih]));
end;
end;
pps:=((1 shl n)-aa)*sb+((1 shl n)-bb)*sa;
if (pps shr n)>0 then pps:=pps-(1 shl n);
pps:=pps shl n;
pp:=aa*bb;
if (pps shr (2*n-1)>0)and(pp shr (2*n-1)>0)then
begin
pps:=pps-(1 shl(2*n));
pp:=pp-(1 shl(2*n));
end;
pp:=pp+pps; //правильное полное произведение
tt:=pp; //правильное усеченное произведение
for i:=1 to k do
for j:=1 to k-i+1 do
tt:=tt-(AAb[i]*BBb[j])shl(i+j-2);
h1:=1;
for ih:=1 to 2*n do
begin
PPb[ih]:=(pp and h1)shr(ih-1);
h1:=h1 shl 1;
Image1.Canvas.TextOut(64*n-32*ih+12,32*n+105, IntToStr(PPb[ih]));
end;
h1:=1 shl k;
for ih:=k+1 to 2*n do
begin
TTb[ih]:=(tt and h1)shr(ih-1);
h1:=h1 shl 1;
Image1.Canvas.TextOut(64*n-32*ih+12,32*n+169, IntToStr(TTb[ih]));
end;
if nfe=0 then
for ih:=1 to 3 do
begin //задание произвольного неисправного элемента
H4[ih]:=Random(n+2)+1;
if H4[ih] > n then H5[ih]:=Random(n)+1
else H5[ih]:=Random(n-1)+1;
end;
Button13.Caption:=IntToStr(H4[ih])+IntToStr(H5[ih]);
if nfr=1 then
for ih:=1 to 3 do
begin //задание произвольной неисправности
repeat
f1:=Random(11)+1;
f2:=Random(11)+1;
until not((f1=f2)or
((MF[f1]='0')and(MF[f2]='1'))or
((MF[f1]='1')and(MF[f2]='0'))or
((MF[f1]='0')and(MF[f2]='0'))or
((MF[f1]='1')and(MF[f2]='1')));
Panel6.Caption:=MF[f1];

```

```

Panel7.Caption:=MF[f2];
if f1<f2 then Fr[ih]:=f2*10+f1 else Fr[ih]:=f1*10+f2;
end;
repeat //выполнение полного и сокращенного умножения
for zz:=1 to 3 do
begin
for i:=1 to n do
for j:=1 to n-1 do
if i+j>k*pu then
begin
if i=1 then
begin
Ma2[j]:=AAb[j+1];
Mb2[j]:=BBb[1];
if ff=1 then
for ih:=1 to zz do
begin
if Fr[ih]=108 then Ma2[H5[ih]]:=0;
if Fr[ih]=118 then Ma2[H5[ih]]:=1;
if Fr[ih]=109 then Mb2[H5[ih]]:=0;
if Fr[ih]=119 then Mb2[H5[ih]]:=1;
if Fr[ih]=98 then
begin
Ma2[H5[ih]]:=Ma2[H5[ih]]*Mb2[H5[ih]];
Mb2[H5[ih]]:=Ma2[H5[ih]];
end;
end;
Ma[1,j]:=Ma2[j]*Mb2[j];
end
else
if j=n-1 then Ma[i,j]:=Mp[i-1,n-1]
else Ma[i,j]:=Ms[i-1,j+1];
Ma1[i,j]:=AAb[i];
Mb1[i,j]:=BBb[j+1];
if ff=1 then
for ih:=1 to zz do
begin
if Fr[ih]=106 then Ma1[H4[ih],H5[ih]]:=0;
if Fr[ih]=116 then Ma1[H4[ih],H5[ih]]:=1;
if Fr[ih]=107 then Mb1[H4[ih],H5[ih]]:=0;
if Fr[ih]=117 then Mb1[H4[ih],H5[ih]]:=1;
if Fr[ih]=76 then
begin
Ma1[H4[ih],H5[ih]]:=Ma1[H4[ih],H5[ih]]*Mb1[H4[ih],H5[ih]];
Mb1[H4[ih],H5[ih]]:=Ma1[H4[ih],H5[ih]];
end;
end;
Mc[i,j]:=Ma1[i,j]*Mb1[i,j];
if j>1 then Mb[i,j]:=Mp[i,j-1]
else Mb[i,j]:=0;
if i+j=(k+1)*pu then Mb[i,j]:=0;
if ff=1 then

```



```

for ih:=1 to zz do
begin
if Fr[ih]=101 then Ma[H4[ih],H5[ih]]:=0;
if Fr[ih]=111 then Ma[H4[ih],H5[ih]]:=1;
if Fr[ih]=102 then Mb[H4[ih],H5[ih]]:=0;
if Fr[ih]=112 then Mb[H4[ih],H5[ih]]:=1;
if Fr[ih]=103 then Mc[H4[ih],H5[ih]]:=0;
if Fr[ih]=113 then Mc[H4[ih],H5[ih]]:=1;
if Fr[ih]=21 then
begin
Ma[H4[ih],H5[ih]]:=Ma[H4[ih],H5[ih]]*Mb[H4[ih],H5[ih]];
Mb[H4[ih],H5[ih]]:=Ma[H4[ih],H5[ih]];
end;
if Fr[ih]=31 then
begin
Ma[H4[ih],H5[ih]]:=Ma[H4[ih],H5[ih]]*Mc[H4[ih],H5[ih]];
Mc[H4[ih],H5[ih]]:=Ma[H4[ih],H5[ih]];
end;
if Fr[ih]=32 then
begin
Mb[H4[ih],H5[ih]]:=Mb[H4[ih],H5[ih]]*Mc[H4[ih],H5[ih]];
Mb[H4[ih],H5[ih]]:=Mc[H4[ih],H5[ih]];
end;
if Fr[ih]=61 then
begin
Ma[H4[ih],H5[ih]]:=Ma[H4[ih],H5[ih]]*Ma1[H4[ih],H5[ih]];
Ma1[H4[ih],H5[ih]]:=Ma[H4[ih],H5[ih]];
end;
if Fr[ih]=62 then
begin
Mb[H4[ih],H5[ih]]:=Mb[H4[ih],H5[ih]]*Ma1[H4[ih],H5[ih]];
Ma1[H4[ih],H5[ih]]:=Mb[H4[ih],H5[ih]];
end;
if Fr[ih]=63 then
begin
Mc[H4[ih],H5[ih]]:=Mc[H4[ih],H5[ih]]*Ma1[H4[ih],H5[ih]];
Ma1[H4[ih],H5[ih]]:=Mc[H4[ih],H5[ih]];
end;
if Fr[ih]=71 then
begin
Ma[H4[ih],H5[ih]]:=Ma[H4[ih],H5[ih]]*Mb1[H4[ih],H5[ih]];
Mb1[H4[ih],H5[ih]]:=Ma[H4[ih],H5[ih]];
end;
if Fr[ih]=72 then
begin
Mb[H4[ih],H5[ih]]:=Mb[H4[ih],H5[ih]]*Mb1[H4[ih],H5[ih]];
Mb1[H4[ih],H5[ih]]:=Mb[H4[ih],H5[ih]];
end;
if Fr[ih]=73 then
begin
Mc[H4[ih],H5[ih]]:=Mc[H4[ih],H5[ih]]*Mb1[H4[ih],H5[ih]];
Mb1[H4[ih],H5[ih]]:=Mc[H4[ih],H5[ih]];
end;

```

```

end;
if (Fr[ih]=81)and(i=1) then
begin
  Ma[H4[ih],H5[ih]]:=Ma[H4[ih],H5[ih]]*Ma2[H5[ih]];
  Ma2[H5[ih]]:=Ma[H4[ih],H5[ih]];
end;
if (Fr[ih]=82)and(i=1) then
begin
  Mb[H4[ih],H5[ih]]:=Mb[H4[ih],H5[ih]]*Ma2[H5[ih]];
  Ma2[H5[ih]]:=Mb[H4[ih],H5[ih]];
end;
if (Fr[ih]=83)and(i=1) then
begin
  Mc[H4[ih],H5[ih]]:=Mc[H4[ih],H5[ih]]*Ma2[H5[ih]];
  Ma2[H5[ih]]:=Mc[H4[ih],H5[ih]];
end;
if (Fr[ih]=91)and(i=1) then
begin
  Ma[H4[ih],H5[ih]]:=Ma[H4[ih],H5[ih]]*Mb2[H5[ih]];
  Mb2[H5[ih]]:=Ma[H4[ih],H5[ih]];
end;
if (Fr[ih]=92)and(i=1) then
begin
  Mb[H4[ih],H5[ih]]:=Mb[H4[ih],H5[ih]]*Mb2[H5[ih]];
  Mb2[H5[ih]]:=Mb[H4[ih],H5[ih]];
end;
if (Fr[ih]=93)and(i=1) then
begin
  Mc[H4[ih],H5[ih]]:=Mc[H4[ih],H5[ih]]*Mb2[H5[ih]];
  Mb2[H5[ih]]:=Mc[H4[ih],H5[ih]];
end;
end;
Ms[i,j]:=Ma[i,j] xor Mb[i,j] xor Mc[i,j];
Mp[i,j]:=Ma[i,j]*Mb[i,j] or Ma[i,j]*Mc[i,j] or Mb[i,j]*Mc[i,j];
if ff=1 then
for ih:=1 to zz do
begin
if Fr[ih]=104 then Ms[H4[ih],H5[ih]]:=0;
if Fr[ih]=114 then Ms[H4[ih],H5[ih]]:=1;
if Fr[ih]=105 then Mp[H4[ih],H5[ih]]:=0;
if Fr[ih]=115 then Mp[H4[ih],H5[ih]]:=1;
if Fr[ih]=41 then
begin
  Ms[H4[ih],H5[ih]]:=Ms[H4[ih],H5[ih]]*Ma[H4[ih],H5[ih]];
  Ma[H4[ih],H5[ih]]:=Ms[H4[ih],H5[ih]];
end;
if Fr[ih]=42 then
begin
  Ms[H4[ih],H5[ih]]:=Ms[H4[ih],H5[ih]]*Mb[H4[ih],H5[ih]];
  Mb[H4[ih],H5[ih]]:=Ms[H4[ih],H5[ih]];
end;
if Fr[ih]=43 then

```

```

begin
  Ms[H4[ih],H5[ih]]:=Ms[H4[ih],H5[ih]]*Mc[H4[ih],H5[ih]];
  Mc[H4[ih],H5[ih]]:=Ms[H4[ih],H5[ih]];
end;
if Fr[ih]=51 then
begin
  Mp[H4[ih],H5[ih]]:=Mp[H4[ih],H5[ih]]*Ma[H4[ih],H5[ih]];
  Ma[H4[ih],H5[ih]]:=Mp[H4[ih],H5[ih]];
end;
if Fr[ih]=52 then
begin
  Mp[H4[ih],H5[ih]]:=Mp[H4[ih],H5[ih]]*Mb[H4[ih],H5[ih]];
  Mb[H4[ih],H5[ih]]:=Mp[H4[ih],H5[ih]];
end;
if Fr[ih]=53 then
begin
  Mp[H4[ih],H5[ih]]:=Mp[H4[ih],H5[ih]]*Mc[H4[ih],H5[ih]];
  Mc[H4[ih],H5[ih]]:=Mp[H4[ih],H5[ih]];
end;
if Fr[ih]=54 then
begin
  Mp[H4[ih],H5[ih]]:=Mp[H4[ih],H5[ih]]*Ms[H4[ih],H5[ih]];
  Ms[H4[ih],H5[ih]]:=Mp[H4[ih],H5[ih]];
end;
if Fr[ih]=64 then
begin
  Ms[H4[ih],H5[ih]]:=Ms[H4[ih],H5[ih]]*Ma1[H4[ih],H5[ih]];
  Ma1[H4[ih],H5[ih]]:=Ms[H4[ih],H5[ih]];
end;
if Fr[ih]=74 then
begin
  Ms[H4[ih],H5[ih]]:=Ms[H4[ih],H5[ih]]*Mb1[H4[ih],H5[ih]];
  Mb1[H4[ih],H5[ih]]:=Ms[H4[ih],H5[ih]];
end;
if (Fr[ih]=84)and(i=1) then
begin
  Ms[H4[ih],H5[ih]]:=Ms[H4[ih],H5[ih]]*Ma2[H5[ih]];
  Ma2[H5[ih]]:=Ms[H4[ih],H5[ih]];
end;
if (Fr[ih]=94)and(i=1) then
begin
  Ms[H4[ih],H5[ih]]:=Ms[H4[ih],H5[ih]]*Mb2[H5[ih]];
  Mb2[H5[ih]]:=Ms[H4[ih],H5[ih]];
end;
if Fr[ih]=65 then
begin
  Mp[H4[ih],H5[ih]]:=Mp[H4[ih],H5[ih]]*Ma1[H4[ih],H5[ih]];
  Ma1[H4[ih],H5[ih]]:=Mp[H4[ih],H5[ih]];
end;
if Fr[ih]=75 then
begin
  Mp[H4[ih],H5[ih]]:=Mp[H4[ih],H5[ih]]*Mb1[H4[ih],H5[ih]];

```

```

    Mb1[H4[ih],H5[ih]]:=Mp[H4[ih],H5[ih]];
end;
if (Fr[ih]=85)and(i=1) then
begin
    Mp[H4[ih],H5[ih]]:=Mp[H4[ih],H5[ih]]*Ma2[H5[ih]];
    Ma2[H5[ih]]:=Mp[H4[ih],H5[ih]];
end;
if (Fr[ih]=95)and(i=1) then
begin
    Mp[H4[ih],H5[ih]]:=Mp[H4[ih],H5[ih]]*Mb2[H5[ih]];
    Mb2[H5[ih]]:=Mp[H4[ih],H5[ih]];
end;
end;
end;
i:=n+1;
for j:=1 to n do
begin
    Ma1[i,j]:=1-AAb[j];
    Mb1[i,j]:=sb;
    if ff=1 then
    for ih:=1 to zz do
    begin
        if Fr[ih]=106 then Ma1[H4[ih],H5[ih]]:=0;
        if Fr[ih]=116 then Ma1[H4[ih],H5[ih]]:=1;
        if Fr[ih]=107 then Mb1[H4[ih],H5[ih]]:=0;
        if Fr[ih]=117 then Mb1[H4[ih],H5[ih]]:=1;
        if Fr[ih]=67 then
        begin
            Ma1[H4[ih],H5[ih]]:=Ma1[H4[ih],H5[ih]]*Mb1[H4[ih],H5[ih]];
            Mb1[H4[ih],H5[ih]]:=Ma1[H4[ih],H5[ih]];
        end;
    end;
    if j<n then Ma[i,j]:=Ms[n,j] else Ma[i,j]:=Mp[n,n-1];
    Mc[i,j]:=Ma1[i,j]*Mb1[i,j];
    if j>1 then Mb[i,j]:=Mp[i,j-1] else Mb[i,j]:=sb;
    if ff=1 then
    for ih:=1 to zz do
    begin
        if Fr[ih]=101 then Ma[H4[ih],H5[ih]]:=0;
        if Fr[ih]=111 then Ma[H4[ih],H5[ih]]:=1;
        if Fr[ih]=102 then Mb[H4[ih],H5[ih]]:=0;
        if Fr[ih]=112 then Mb[H4[ih],H5[ih]]:=1;
        if Fr[ih]=103 then Mc[H4[ih],H5[ih]]:=0;
        if Fr[ih]=113 then Mc[H4[ih],H5[ih]]:=1;
        if Fr[ih]=21 then
        begin
            Ma[H4[ih],H5[ih]]:=Ma[H4[ih],H5[ih]]*Mb[H4[ih],H5[ih]];
            Mb[H4[ih],H5[ih]]:=Ma[H4[ih],H5[ih]];
        end;
        if Fr[ih]=31 then
        begin
            Ma[H4[ih],H5[ih]]:=Ma[H4[ih],H5[ih]]*Mc[H4[ih],H5[ih]];

```

```

Mc[H4[ih],H5[ih]]:=Ma[H4[ih],H5[ih]];
end;
if Fr[ih]=32 then
begin
Mb[H4[ih],H5[ih]]:=Mb[H4[ih],H5[ih]]*Mc[H4[ih],H5[ih]];
Mb[H4[ih],H5[ih]]:=Mc[H4[ih],H5[ih]];
end;
if Fr[ih]=61 then
begin
Ma[H4[ih],H5[ih]]:=Ma[H4[ih],H5[ih]]*Ma1[H4[ih],H5[ih]];
Ma1[H4[ih],H5[ih]]:=Ma[H4[ih],H5[ih]];
end;
if Fr[ih]=62 then
begin
Mb[H4[ih],H5[ih]]:=Mb[H4[ih],H5[ih]]*Ma1[H4[ih],H5[ih]];
Ma1[H4[ih],H5[ih]]:=Mb[H4[ih],H5[ih]];
end;
if Fr[ih]=63 then
begin
Mc[H4[ih],H5[ih]]:=Mc[H4[ih],H5[ih]]*Ma1[H4[ih],H5[ih]];
Ma1[H4[ih],H5[ih]]:=Mc[H4[ih],H5[ih]];
end;
if Fr[ih]=71 then
begin
Ma[H4[ih],H5[ih]]:=Ma[H4[ih],H5[ih]]*Mb1[H4[ih],H5[ih]];
Mb1[H4[ih],H5[ih]]:=Ma[H4[ih],H5[ih]];
end;
if Fr[ih]=72 then
begin
Mb[H4[ih],H5[ih]]:=Mb[H4[ih],H5[ih]]*Mb1[H4[ih],H5[ih]];
Mb1[H4[ih],H5[ih]]:=Mb[H4[ih],H5[ih]];
end;
if Fr[ih]=73 then
begin
Mc[H4[ih],H5[ih]]:=Mc[H4[ih],H5[ih]]*Mb1[H4[ih],H5[ih]];
Mb1[H4[ih],H5[ih]]:=Mc[H4[ih],H5[ih]];
end;
if (Fr[ih]=81)and(i=1) then
begin
Ma[H4[ih],H5[ih]]:=Ma[H4[ih],H5[ih]]*Ma2[H5[ih]];
Ma2[H5[ih]]:=Ma[H4[ih],H5[ih]];
end;
if (Fr[ih]=82)and(i=1) then
begin
Mb[H4[ih],H5[ih]]:=Mb[H4[ih],H5[ih]]*Ma2[H5[ih]];
Ma2[H5[ih]]:=Mb[H4[ih],H5[ih]];
end;
if (Fr[ih]=83)and(i=1) then
begin
Mc[H4[ih],H5[ih]]:=Mc[H4[ih],H5[ih]]*Ma2[H5[ih]];
Ma2[H5[ih]]:=Mc[H4[ih],H5[ih]];
end;

```

```

if (Fr[ih]=91)and(i=1) then
begin
Ma[H4[ih],H5[ih]]:=Ma[H4[ih],H5[ih]]*Mb2[H5[ih]];
Mb2[H5[ih]]:=Ma[H4[ih],H5[ih]];
end;
if (Fr[ih]=92)and(i=1) then
begin
Mb[H4[ih],H5[ih]]:=Mb[H4[ih],H5[ih]]*Mb2[H5[ih]];
Mb2[H5[ih]]:=Mb[H4[ih],H5[ih]];
end;
if (Fr[ih]=93)and(i=1) then
begin
Mc[H4[ih],H5[ih]]:=Mc[H4[ih],H5[ih]]*Mb2[H5[ih]];
Mb2[H5[ih]]:=Mc[H4[ih],H5[ih]];
end;
end;
Ms[i,j]:=Ma[i,j] xor Mb[i,j] xor Mc[i,j];
Mp[i,j]:=Ma[i,j]*Mb[i,j] or Ma[i,j]*Mc[i,j] or Mb[i,j]*Mc[i,j];
if ff=1 then
for ih:=1 to zz do
begin
if Fr[ih]=104 then Ms[H4[ih],H5[ih]]:=0;
if Fr[ih]=114 then Ms[H4[ih],H5[ih]]:=1;
if Fr[ih]=105 then Mp[H4[ih],H5[ih]]:=0;
if Fr[ih]=115 then Mp[H4[ih],H5[ih]]:=1;
if Fr[ih]=41 then
begin
Ms[H4[ih],H5[ih]]:=Ms[H4[ih],H5[ih]]*Ma[H4[ih],H5[ih]];
Ma[H4[ih],H5[ih]]:=Ms[H4[ih],H5[ih]];
end;
if Fr[ih]=42 then
begin
Ms[H4[ih],H5[ih]]:=Ms[H4[ih],H5[ih]]*Mb[H4[ih],H5[ih]];
Mb[H4[ih],H5[ih]]:=Ms[H4[ih],H5[ih]];
end;
if Fr[ih]=43 then
begin
Ms[H4[ih],H5[ih]]:=Ms[H4[ih],H5[ih]]*Mc[H4[ih],H5[ih]];
Mc[H4[ih],H5[ih]]:=Ms[H4[ih],H5[ih]];
end;
if Fr[ih]=51 then
begin
Mp[H4[ih],H5[ih]]:=Mp[H4[ih],H5[ih]]*Ma[H4[ih],H5[ih]];
Ma[H4[ih],H5[ih]]:=Mp[H4[ih],H5[ih]];
end;
if Fr[ih]=52 then
begin
Mp[H4[ih],H5[ih]]:=Mp[H4[ih],H5[ih]]*Mb[H4[ih],H5[ih]];
Mb[H4[ih],H5[ih]]:=Mp[H4[ih],H5[ih]];
end;
if Fr[ih]=53 then
begin

```

```

Mp[H4[ih],H5[ih]]:=Mp[H4[ih],H5[ih]]*Mc[H4[ih],H5[ih]];
Mc[H4[ih],H5[ih]]:=Mp[H4[ih],H5[ih]];
end;
if Fr[ih]=54 then
begin
Mp[H4[ih],H5[ih]]:=Mp[H4[ih],H5[ih]]*Ms[H4[ih],H5[ih]];
Ms[H4[ih],H5[ih]]:=Mp[H4[ih],H5[ih]];
end;
if Fr[ih]=64 then
begin
Ms[H4[ih],H5[ih]]:=Ms[H4[ih],H5[ih]]*Ma1[H4[ih],H5[ih]];
Ma1[H4[ih],H5[ih]]:=Ms[H4[ih],H5[ih]];
end;
if Fr[ih]=74 then
begin
Ms[H4[ih],H5[ih]]:=Ms[H4[ih],H5[ih]]*Mb1[H4[ih],H5[ih]];
Mb1[H4[ih],H5[ih]]:=Ms[H4[ih],H5[ih]];
end;
if Fr[ih]=65 then
begin
Mp[H4[ih],H5[ih]]:=Mp[H4[ih],H5[ih]]*Ma1[H4[ih],H5[ih]];
Ma1[H4[ih],H5[ih]]:=Mp[H4[ih],H5[ih]];
end;
if Fr[ih]=75 then
begin
Mp[H4[ih],H5[ih]]:=Mp[H4[ih],H5[ih]]*Mb1[H4[ih],H5[ih]];
Mb1[H4[ih],H5[ih]]:=Mp[H4[ih],H5[ih]];
end;
end;
end;
i:=n+2;
for j:=1 to n do
begin
Ma1[i,j]:=1-BBb[j];
Mb1[i,j]:=sa;
if ff=1 then
for ih:=1 to zz do
begin
if Fr[ih]=106 then Ma1[H4[ih],H5[ih]]:=0;
if Fr[ih]=116 then Ma1[H4[ih],H5[ih]]:=1;
if Fr[ih]=107 then Mb1[H4[ih],H5[ih]]:=0;
if Fr[ih]=117 then Mb1[H4[ih],H5[ih]]:=1;
if Fr[ih]=67 then
begin
Ma1[H4[ih],H5[ih]]:=Ma1[H4[ih],H5[ih]]*Mb1[H4[ih],H5[ih]];
Mb1[H4[ih],H5[ih]]:=Ma1[H4[ih],H5[ih]];
end;
end;
Ma[i,j]:=Ms[n+1,j];
Mc[i,j]:=Ma1[i,j]*Mb1[i,j];
if j>1 then Mb[i,j]:=Mp[i,j-1] else Mb[i,j]:=sa;
if ff=1 then

```

```

for ih:=1 to zz do
begin
if Fr[ih]=101 then Ma[H4[ih],H5[ih]]:=0;
if Fr[ih]=111 then Ma[H4[ih],H5[ih]]:=1;
if Fr[ih]=102 then Mb[H4[ih],H5[ih]]:=0;
if Fr[ih]=112 then Mb[H4[ih],H5[ih]]:=1;
if Fr[ih]=103 then Mc[H4[ih],H5[ih]]:=0;
if Fr[ih]=113 then Mc[H4[ih],H5[ih]]:=1;
if Fr[ih]=21 then
begin
Ma[H4[ih],H5[ih]]:=Ma[H4[ih],H5[ih]]*Mb[H4[ih],H5[ih]];
Mb[H4[ih],H5[ih]]:=Ma[H4[ih],H5[ih]];
end;
if Fr[ih]=31 then
begin
Ma[H4[ih],H5[ih]]:=Ma[H4[ih],H5[ih]]*Mc[H4[ih],H5[ih]];
Mc[H4[ih],H5[ih]]:=Ma[H4[ih],H5[ih]];
end;
if Fr[ih]=32 then
begin
Mb[H4[ih],H5[ih]]:=Mb[H4[ih],H5[ih]]*Mc[H4[ih],H5[ih]];
Mb[H4[ih],H5[ih]]:=Mc[H4[ih],H5[ih]];
end;
if Fr[ih]=61 then
begin
Ma[H4[ih],H5[ih]]:=Ma[H4[ih],H5[ih]]*Ma1[H4[ih],H5[ih]];
Ma1[H4[ih],H5[ih]]:=Ma[H4[ih],H5[ih]];
end;
if Fr[ih]=62 then
begin
Mb[H4[ih],H5[ih]]:=Mb[H4[ih],H5[ih]]*Ma1[H4[ih],H5[ih]];
Ma1[H4[ih],H5[ih]]:=Mb[H4[ih],H5[ih]];
end;
if Fr[ih]=63 then
begin
Mc[H4[ih],H5[ih]]:=Mc[H4[ih],H5[ih]]*Ma1[H4[ih],H5[ih]];
Ma1[H4[ih],H5[ih]]:=Mc[H4[ih],H5[ih]];
end;
if Fr[ih]=71 then
begin
Ma[H4[ih],H5[ih]]:=Ma[H4[ih],H5[ih]]*Mb1[H4[ih],H5[ih]];
Mb1[H4[ih],H5[ih]]:=Ma[H4[ih],H5[ih]];
end;
if Fr[ih]=72 then
begin
Mb[H4[ih],H5[ih]]:=Mb[H4[ih],H5[ih]]*Mb1[H4[ih],H5[ih]];
Mb1[H4[ih],H5[ih]]:=Mb[H4[ih],H5[ih]];
end;
if Fr[ih]=73 then
begin
Mc[H4[ih],H5[ih]]:=Mc[H4[ih],H5[ih]]*Mb1[H4[ih],H5[ih]];
Mb1[H4[ih],H5[ih]]:=Mc[H4[ih],H5[ih]];
end;
end;

```



```

end;
end;
Ms[i,j]:=Ma[i,j] xor Mb[i,j] xor Mc[i,j];
Mp[i,j]:=Ma[i,j]*Mb[i,j] or Ma[i,j]*Mc[i,j] or Mb[i,j]*Mc[i,j];
if ff=1 then
for ih:=1 to zz do
begin
if Fr[ih]=104 then Ms[H4[ih],H5[ih]]:=0;
if Fr[ih]=114 then Ms[H4[ih],H5[ih]]:=1;
if Fr[ih]=105 then Mp[H4[ih],H5[ih]]:=0;
if Fr[ih]=115 then Mp[H4[ih],H5[ih]]:=1;
if Fr[ih]=41 then
begin
Ms[H4[ih],H5[ih]]:=Ms[H4[ih],H5[ih]]*Ma[H4[ih],H5[ih]];
Ma[H4[ih],H5[ih]]:=Ms[H4[ih],H5[ih]];
end;
if Fr[ih]=42 then
begin
Ms[H4[ih],H5[ih]]:=Ms[H4[ih],H5[ih]]*Mb[H4[ih],H5[ih]];
Mb[H4[ih],H5[ih]]:=Ms[H4[ih],H5[ih]];
end;
if Fr[ih]=43 then
begin
Ms[H4[ih],H5[ih]]:=Ms[H4[ih],H5[ih]]*Mc[H4[ih],H5[ih]];
Mc[H4[ih],H5[ih]]:=Ms[H4[ih],H5[ih]];
end;
if Fr[ih]=51 then
begin
Mp[H4[ih],H5[ih]]:=Mp[H4[ih],H5[ih]]*Ma[H4[ih],H5[ih]];
Ma[H4[ih],H5[ih]]:=Mp[H4[ih],H5[ih]];
end;
if Fr[ih]=52 then
begin
Mp[H4[ih],H5[ih]]:=Mp[H4[ih],H5[ih]]*Mb[H4[ih],H5[ih]];
Mb[H4[ih],H5[ih]]:=Mp[H4[ih],H5[ih]];
end;
if Fr[ih]=53 then
begin
Mp[H4[ih],H5[ih]]:=Mp[H4[ih],H5[ih]]*Mc[H4[ih],H5[ih]];
Mc[H4[ih],H5[ih]]:=Mp[H4[ih],H5[ih]];
end;
if Fr[ih]=54 then
begin
Mp[H4[ih],H5[ih]]:=Mp[H4[ih],H5[ih]]*Ms[H4[ih],H5[ih]];
Ms[H4[ih],H5[ih]]:=Mp[H4[ih],H5[ih]];
end;
if Fr[ih]=64 then
begin
Ms[H4[ih],H5[ih]]:=Ms[H4[ih],H5[ih]]*Ma1[H4[ih],H5[ih]];
Ma1[H4[ih],H5[ih]]:=Ms[H4[ih],H5[ih]];
end;
if Fr[ih]=74 then

```

```

begin
  Ms[H4[ih],H5[ih]]:=Ms[H4[ih],H5[ih]]*Mb1[H4[ih],H5[ih]];
  Mb1[H4[ih],H5[ih]]:=Ms[H4[ih],H5[ih]];
end;
if Fr[ih]=65 then
begin
  Mp[H4[ih],H5[ih]]:=Mp[H4[ih],H5[ih]]*Ma1[H4[ih],H5[ih]];
  Ma1[H4[ih],H5[ih]]:=Mp[H4[ih],H5[ih]];
end;
if Fr[ih]=75 then
begin
  Mp[H4[ih],H5[ih]]:=Mp[H4[ih],H5[ih]]*Mb1[H4[ih],H5[ih]];
  Mb1[H4[ih],H5[ih]]:=Mp[H4[ih],H5[ih]];
end;
end;
end;
if pu=0 then
begin
  cc:=cc+1;
  PPe[1]:=AAb[1]*BBb[1];
  for ih:=1 to n-1 do
    PPe[ih+1]:=Ms[ih,1];
  for ih:=1 to n do
    PPe[n+ih]:=Ms[n+2,ih];
  pe:=0;
  for ih:=1 to 2*n do
  begin
    pe:=pe+PPe[ih]shl(ih-1);
    Image1.Canvas.TextOut(64*n-32*ih+12,32*n+137, IntToStr(PPe[ih]));
  end;
  Edit3.Text:=IntToStr(pp);
  Edit4.Text:=IntToStr(pe);
  sp:=sa xor sb;
  ka:=(aa-sa) mod 3;
  kb:=(bb-sb) mod 3;
  kp:=(pe-sp) mod 3;
  if ka*kb=kp then e1:=0 else e1:=1;
  if pp<>pe then
  begin
    Ce[zz]:=Ce[zz]+1;
    if (abs(pp-pe)>=(1 shl(n+es)))then
    begin
      Ep[zz]:=Ep[zz]+1; //кол-во сущ ошибок в полн произв
      Dep[zz]:=Dep[zz]+e1; //кол-во обн сущ ошибок в полн произв
    end
    else Dnp[zz]:=Dnp[zz]+e1; //кол-во обн несущ ошибок в полн произв
  end;
end
else
begin
  for ih:=k to n-1 do
    TTe[ih+1]:=Ms[ih,1];

```

```

for ih:=1 to n do
  TTe[n+ih]:=Ms[n+2,ih];
te:=0;
for ih:=k+1 to 2*n do
begin
  te:=te+TTe[ih]shl(ih-1);
  Image1.Canvas.TextOut(64*n-32*ih+12,32*n+201, IntToStr(TTe[ih]));
end;
Edit8.Text:=IntToStr(tt);
Edit10.Text:=IntToStr(te);
if tt mod 3=te mod 3 then e2:=0 else e2:=1;
if tt<>te then
begin
  Ee[zz]:=Ee[zz]+1;
  if (abs(tt-te)>=(1shl(n+es)))then
  begin
    Et[zz]:=Et[zz]+1; //кол-во суц ошибок в усеч произв
    Det[zz]:=Det[zz]+e2; //кол-во обн суц ошибок в усеч произв
  end
  else Dnt[zz]:=Dnt[zz]+e2; //кол-во обн несущ ошибок в усеч произв
end;
end;
end; // zz multiply of faults
if (za<2)and(zb<2) then
begin
  Panel21.Visible:=True;
  if pu=0 then
  begin
    Button7.Visible:=False;
    Panel21.Caption:='Повне множення';
    repeat
      Application.ProcessMessages;
    until Panel21.Tag=1;
  end
  else
    Panel21.Caption:='Скорочене множення';
    Button7.Visible:=True;
  end;
pu:=1-pu;
Panel18.Visible:=False;
Panel23.Visible:=False;
until pu=0;
if (za=3)and(zb>1) then cb:=1;
Application.ProcessMessages;
if zb=2 then
begin
  bb:=bb+1;
  if bb=2*h2 then cb:=1;
end;
until cb=1;
if za=2 then
begin

```

```
aa:=aa+1;
if aa=2*h2 then ca:=1;
end;
until ca=1;
end;
Button2.Visible:=True;
Button7.Caption:='Старт';
ca:=1; cb:=1;
end;
```