2. Під час настроювання практичних схем оптимальний режим роботи автодина можна підібрати без складних вимірювань на високій частоті, визначивши лише постійну складову струму стоку транзистора в режимі генерування і за його відсутності. Залежність постійної складової струму стоку від опору навантаження може бути покладена в основу вимірювання активної складової комплексного опору змінному струму.

3. За одночасного суміщення функцій генератора і перетворювача частоти в однотранзисторному каскаді найбільша крутість перетворення за кусково-параболічної апроксимації забезпечується з фактором регенерування 2–4,7, тобто з кутом відсіку близько 120°, за кусково-лінійної – відповідно 1,5–3 та 90°.

4. Розрахунки, виконані методами кусково-параболічної і кусково-лінійної апроксимації, дають близькі результати, однаково придатні для практичного використання.

УДК 681.3

Н.Д. Круцкевич, Я.М. Николайчук
Тернопільський державний економічний університет

СИНТЕЗ І ПРОЕКТУВАННЯ АСОЦІАТИВНОГО СУМАТОРА СИСТЕМИ ЗАЛИШКОВИХ КЛАСІВ В БАЗИСІ ГАЛУА

Проведено аналіз існуючих алгоритмів виконання операцій підсумовування в базисі Галуа за модулем, що використовуються в сучасній обчислювальній техніці. Запропоновано метод асоціативного підсумовування та синтезовано структуру асоціативного суматора за модулем в базисі Галуа та виконано його схемотехнічне проектування.

The analysis of existent algorithms of implementation of operations of that is sad is conducted in the base of Galois on the module, that is used in modern to the computing engineering. The method of that is associative sad is offered and synthesized the structure of the associative summarizing on the module in the base of Galois and it is executed his circuit technique design.

Вступ

Створення швидкодіючих елементів обчислювальної техніки є актуальним і важливим науково-технічним завданням. Зокрема існує гостра потреба в елементах, що реалізовують швидке виконання математичних операцій в режимі реального часу. Проте традиційні підходи до розв’язання цих задач вже досягнули теоретично можливої критичної межі (збільшення тактової частоти процесорів, розширення шини даних тощо). Виходом є дослідження нових, більш перспективних теоретико-числових базисів та реалізація швидких алгоритмів виконання математичних операцій на їх базі. До таких перспективних теоретико-числових базисів необхідно віднести теоретико-числові базиси Галуа та Крестинсона [1]

1. Аналіз методів сумування в базисі Галуа

Здійснення арифметичних операцій в полі Галуа характеризується різною формою подання двох операндів. Перший операнд подається у вигляді коду, а другий – у вигляді логічних рівнянь, які визначають операції над значенням коду першого операнда.

За умови простої технічної реалізації наведених процедур на основі регістрів зсуву швидкодія вказаного методу доволі низька та визначається розрядністю k та r операндів і відповідно кількістю тактів перемноження (максимально теоретично можлива – k+r).

Медод виконання основних математичних операцій в кодах Галуа ґрунтується на безпосередній паралельній обробці операндів на підставі синтезованих логічних функцій порозрядного сумування за mod p [2, 3].

Нехай для заданих операндів:

$$A(x) = \sum_{i=0}^{n-1} a_i x^i \bmod p\,; \tag{1}$$

та

$$D(x) = \sum_{i=0}^{n-1} d_i x^i \bmod p \tag{2}$$

результатом підсумовування визначено поліном:

$$C(x) = \sum_{i=0}^{n-1} c_i x^i \bmod p\,. \tag{3}$$

З метою отримання аналітичних закономірностей для обчислення dji необхідно здійснити кілька теоретико-числових перетворень формального переходу із послідовного рекурсивного виконання операції підсумовування в паралельне векторне.

Для прикладу поля Галуа GF(24) із подорожуючим вектором 10011 рекурсивна послідовність кодових елементів 111101011001000 подається формалізовано у такому вигляді:

$$b_1, b_2, b_3, b_4, b_1 \oplus b_2 \oplus b_4 \oplus,\ b_1 \oplus b_2 \oplus b_3 \oplus b_4,\ b_1 \oplus b_2 \oplus b_3,\ b_2 \oplus b_3 \oplus b_4,\ b_1 \oplus b_3,\ b_2 \oplus b_4,$$
$$b_1 \oplus b_3 \oplus b_4,\ b_1 \oplus b_2,\ b_2 \oplus b_3,\ b_3 \oplus b_4,\ b_1,\ b_2,\ b_3.$$

Із формалізованого рекурсивного діагонального запису кодів сум даних операція додавання двох кодів A(x) та D(x) визначається як процедура рекурсивного зсуву, починаючи з вихідної позиції заданого коду A(x) на кількість дискретних позицій, визначену десятковим еквівалентом іншого заданого коду операнда D(x). Отже, реалізація вказаної операції додавання в полі Галуа зводиться до одночасного паралельного взаємно незалежного формування кожного біта результату обчислення як суми за mod2 без необхідності виконання операцій міжрозрядних переносів, що уможливлює підвищити швидкодію обробки пропорційної розрядності слова даних порівняно із двійковою системою числення. Обчислення кожного результату операції здійснюється за один такт обчислення інваріантної розрядності слова даних.

Для систем кодування вищих порядків n аналогічно визначаються n- розрядних сум за mod2 кожного із елементів послідовності Галуа.

Для паралельного підсумовування кодів в полі GF(24) використовують операційну таблицю з заданим породжуючим вектором 1001. Для виконання операції над кодом операнда A(x), наприклад 0101(4) здійснюються дії, що визначаються логічним вектором D’(x) відповідно до значення операнда D(x), наприклад 0110(6), тобто для обчислення кожного розряду C(x) одночасно виконуються такі операції:

b1	b2	b3	b4	
A(x)=0	1	0	1	$= 4_{(10)}$
D(x)=0	1	1	0	$= 6_{(10)}$
D’(x)=>b1⊕b2⊕b3⊕b4	b1⊕b2⊕b3	b2⊕b3⊕b4	b1⊕b3	
0 ⊕ 1 ⊕ 0 ⊕ 1	0 ⊕ 1 ⊕ 0	1 ⊕ 0 ⊕ 1	0 ⊕ 0	
C(x)=0	1	0	0	$=10_{(10)}$

Код Галуа результату C(x)= 0100 відповідає десятковому числу 10(10), яке і є результатом суми чисел 4(10) та 6(10) – десяткових еквівалентів операндів Галуа A(x) та D(x).

Прикладне застосування пропонованого методу виконання арифметичних операцій полягає у розробці кодових матриць визначеного перетворення системи кодів. Для цього за операційною таблицею будується таблиця коефіцієнтів dji (Table1), що відображає розряди програмування вмісту масиву елементів програмованої логічної матриці, де кодовий рядок Галуа визначає коди d3, d2, d1, d0 адресної вибірки масиву пам’яті, а рядки dji – вихідні коди його адресованого вмісту.

Таблиця 1

Значення проміжних коефіцієнтів d_{ji}
операції додавання в полі Галуа GF(24)

№	адреса	$b_4^4\ b_3^4\ d_2^4\ d_1^4$	$b_4^3\ b_3^3\ d_2^3\ d_1^3$	$b_4^2\ b_3^2\ d_2^2\ d_1^2$	$b_4^1\ b_3^1\ d_2^1\ d_1^1$
0	1111	1000	0100	0010	0001
1	1110	0100	0010	0001	1001
2	1101	0010	0001	1001	1101
3	1010	0001	1001	1101	1111
4	0101	1001	1101	1111	1110
5	1011	1101	1111	1110	0111
6	0110	1111	1110	0111	1010
7	1100	1110	0111	1010	0101
8	1001	0111	1010	0101	1011
9	0010	1010	0101	1011	1100
10	0100	0101	1011	1100	0110
11	1000	1011	1100	0110	0011
12	0001	1100	0110	0011	1000
13	0011	0110	0011	1000	0100
14	0111	0011	1000	0100	0010

В узагальненому випадку блок перетворення кодів, що виконує функцію перетворення елементів di в dji, є масивом програмованих логічних елементів матриці з організацією $n \times n^2$.

2. Проектування асоціативного суматора в базисі Галуа

Під час реалізації блока сумування операнди подано в кодах Галуа, тобто двійкове представлення здійснюється не в двійковій системі числення, а в системі числення Галуа (рис. 1).

1	1	1	0
1	1	0	1
1	0	1	2
0	1	0	3
1	0	0	4
0	0	1	5
0	1	1	6

Рис. 1. Представлення діапазону чисел за модулем 7 в кодах Галуа

Пропонується для швидкого додавання чисел використовувати метод матричного додавання, коли числа операнда є адресами, а результат операції знаходиться в комірці пам'яті, яка адресується. Для здійснення операції додавання за модулем 7 отримаємо таку матрицю (рис. 2).

	0	1	2	3	4	5	6
0	0	1	2	3	4	5	6
1	1	2	3	4	5	6	0
2	2	3	4	5	6	0	1
3	3	4	5	6	0	1	2
4	4	5	6	0	1	2	3
5	5	6	0	1	2	3	4
6	6	0	1	2	3	4	5

	0	1	2	3	4	5	6
0	1	1	1	0	1	0	0
1	1	1	0	1	0	0	1
2	1	0	1	0	0	1	1
3	0	1	0	0	1	1	1
4	1	0	0	1	1	1	0
5	0	0	1	1	1	0	1
6	0	1	1	1	0	1	0

Рис. 2. Матриця результатів додавання за модулем 7
у десятковому та двійковому Галуа-представленнях

Операція додавання здійснюється в такий спосіб, коли на входи пам'яті подаються операнди в коді Галуа. Наприклад, під час додавання двох чисел 4 та 3 за модулем 7 оримаємо 0; подамо ці числа в базисі Галуа:

4=010;

3=100;

4+3=0;

010+100=111.

Процес додавання показано на матриці, де кольором виділено операнди та їх еквіваленти в кодах Галуа (рис. 3).

Результат додавання подано у вигляді вектора, початок якого знаходиться в комірці пам'яті. Як дешифратор використано дешифратор Галуа, що дає змогу організувати асоціативний спосіб дешифрації, де як адреси виступають результати обчислень. Використання властивостей кодів Галуа: рекурентності та циклічності дає можливість здійснювати обчислення, використовуючи при цьому 2D пам'яті, а не 3D, як за класичної реалізації таких процесорів в двійковому кодовому базисі заданою адресою (рис. 4).

	0	1	2	3	4	5	6
0	1	1	1	0	1	0	0
1	1	1	0	1	0	0	1
2	1	0	1	0	0	1	1
3	0	1	0	0	1	1	1
4	1	0	0	1	1	1	0
5	0	0	1	1	1	0	1
6	0	1	1	1	0	1	0

Рис. 3. Матриця додавання за модулем 7 з виділеними операндами 3 та 4

	0	1	2	3	4	5	6
0	1	1	1	0	1	0	0
1	1	1	0	1	0	0	1
2	1	0	1	0	0	1	1
3	0	1	0	0	1	1	1
4	1	0	0	1	1	1	0
5	0	0	1	1	1	0	1
6	0	1	1	1	0	1	0

Рис. 4. Матриця додавання за модулем 7 з результатом додавання, поданим у вигляді вектора

3. Розробка принципової схеми асоціативного суматора

Проаналізувавши метод додавання чисел за певним модулем, поданих в базисі кодів Галуа, синтезуємо структурну схему паралельного адресного суматора. Для прикладу синтезуємо структурну схему для суматора за модулем 7 (рис. 5).

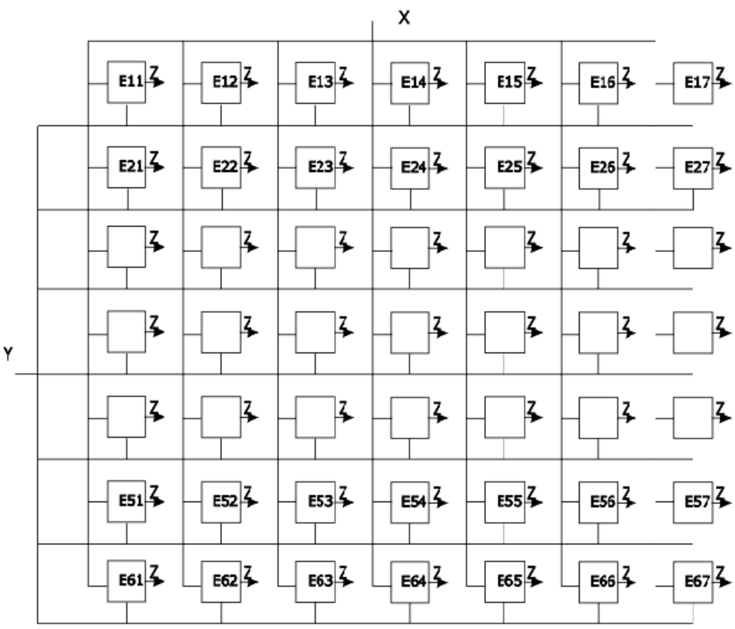

Рис. 5. Структурна схема паралельного адресного суматора за модулем 7

Проаналізувавши структурну схему суматора, бачимо, що це є регулярна структура, яка складається з однотипних блоків (рис. 6).

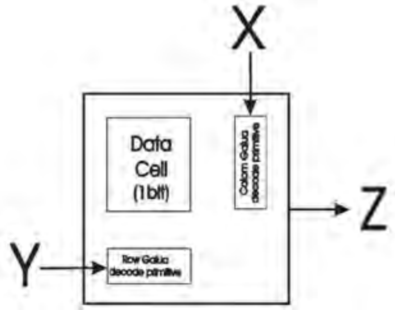

Рис. 6. Базовий елемент суматора Галуа за модулем

Цей базовий елемент складається з комірки пам'яті розміром 1 біт, дешифратора рядка та дешифратора стовпчика, вихідного елемента. За нарощення розрядності такого суматора обсяг пам'яті зростає квадратично, а не кубічно, як в традиційних пристроях підсумовування через пам'ять (рис. 7).

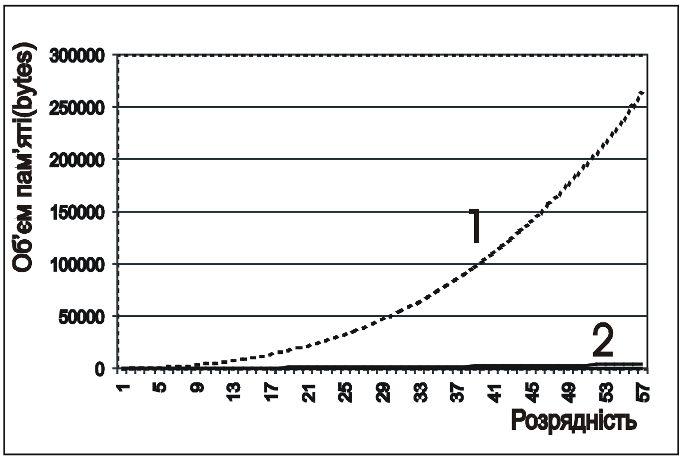

Рис. 7. Порівняльна характеристика збільшення обсягу пам'яті суматора, реалізованого на базі пам'яті: 1 – класичний метод з використанням 3D пам'яті; 2 – асоціативний суматор Галуа на базі 2D пам'яті

Проведена порівняльна характеристика двох типів суматорів дає змогу зробити висновки, що апаратні затрати на реалізації асоціативного суматора Галуа є значно менші за однакової обчислювальної потужності та швидкодії.

Висновок

Створено принципово новий процесорний елемент швидкого підсумовування, що використовує спільну обчислювальну площу.

Реалізовано асоціативний метод дешифрації, що уможливлює використовувати комірки пам'яті як дешифрувальні елементи з виключенням класичних пірамідальних дешифраторів.

Використовуючи циклічну та рекурентну властивості кодів Галуа, реалізований процесор використовує 2D пам'ять, натомість як процесори в базисі Радемахера використовують 3D пам'ять, що дає змогу значно скоротити об'єм кристала за однакової розрядності та швидкодії.

1. Николайчук Я.М., Кусик Я.Б. Коди поля Галуа та їх застосування в перетворювачах форм інформації: Доклады 7-го симпозиума: Проблемы создания преобразователей форм информации. – К., 1992. 2. Николайчук Я.М., Петришин Л.Б. Вертикальні інформаційні технології в кодових системах Галуа: Матеріали 2-ї Української конференції з автоматичного керування "Автоматика-95". – Львів, 1995. – С. 131. 3. Krutskevych N., Nykolaychuk Y. Structure and functions of memory collective access based on Galua codes // the collection ofscience jobs "Measuring and computer facilities in technological processes", Khmel'nyc'k. – 2002. – №9, V. 2. – P. 126–129.