УДК 004.42:004.056.5:512.64 DOI https://doi.org/10.35546/kntu2078-4481.2025.4.2.14

О. А. МЕЛЬНИКОВА
кандидат технічних наук,
доцент кафедри безпеки інформаційних технологій
Харківський національний університет радіоелектроніки
ORCID: 0000-0001-8843-9648

С. В. ГРАСМІК
студентка бакалаврату кафедри безпеки інформаційних технологій
Харківський національний університет радіоелектроніки
ORCID: 0009-0008-5521-9831

О. Є. ПЕТРЕНКО
кандидат технічних наук, доцент,
доцент кафедри безпеки інформаційних технологій
Харківський національний університет радіоелектроніки
ORCID: 0000-0002-7862-5399

І. В. ОЛЕШКО
кандидат технічних наук, доцент,
доцент кафедри безпеки інформаційних технологій
Харківський національний університет радіоелектроніки
ORCID: 0000-0002-8021-0467

СТВОРЕННЯ АЛГОРИТМІВ ПРИВЕДЕННЯ ПО ФІКСОВАНИМ МОДУЛЯМ ДЛЯ КРИПТОСИСТЕМ, ЩО ВИКОРИСТОВУЮТЬ АРИФМЕТИКУ РОЗШИРЕНИХ ДВІЙКОВИХ ПОЛІВ

Низка діючих вітчизняних стандартів, зокрема цифрового підпису [1] та блокового симетричного шифрування [2], використовує арифметику розширених двійкових полів $GF(2^m)$ на основі поліноміального базису для криптографічних перетворень. Для високого рівня стійкості криптографічні стандарти рекомендують великі значення m для поля $GF(2^m)$. Зокрема, стандарт [1] визначає 60 варіантів незвідних поліномів f(t) з простими степенями m від 163 до 509. А стандарт [2] використовує 3 варіанти незвідних поліномів f(t) для парних значень m, визначених бітовими розмірами блоку шифру (128, 256 чи 512). При великих m операції в полі $GF(2^m)$ мають високу обчислювальну складність. Одним із ключових обмежувальних факторів швидкодії є операція приведення по модулю f(t) (редукція), і зменшення її складності є критично важливим для реалізацій у режимі реального часу.

Загальні методи модулярної редукції не завжди враховують особливості оптимізації для конкретного незвідного модуля f(t) m-го степеню (триному чи пентаному). Розробка алгоритмів редукції, оптимізованих саме з прив'язкою до рекомендованих стандартами фіксованих значень f(t), може зменшувати обчислювальну складність, в тому числі й при апаратній реалізації. Зменшення складності досягається завдяки незмінності модуля f(t) протягом сеансу, що дозволяє інтегрувати попередні розрахунки безпосередньо в структуру алгоритмів приведення по модулю.

Стаття присвячена створенню та дослідженню множин алгоритмів приведення по фіксованим модулям f(t) з метою оптимізації базових операцій у криптосистемах, що використовують $GF(2^m)$. Розглянуто методику створення подібних алгоритмів, яка дозволяє формалізувати та частково автоматизувати їх створення. Можливість програмної автоматизації цього процесу є важливою, оскільки стандарти рекомендують велику кількість варіантів полів $GF(2^m)$. Також вона необхідна для забезпечення формалізованого підходу до перевірки коректності та тестування цих реалізацій.

Результатом роботи є розробка серії оптимізованих алгоритмів приведення по модулю, прив'язаних до модулів, які рекомендовані українськими криптографічними стандартами.

Ключові слова: *алгоритми приведення за модулем, арифметика розширених двійкових полів, приведення за фіксованими модулями, оптимізація алгоритмів, криптоалгоритми.*

O. A. MELNYKOVA
Candidate of Technical Sciences,
Associate Professor at the Department of Information Technology Security
Kharkiv National University of Radio Electronics
ORCID: 0000-0001-8843-9648

S. V. HRASMIK
Bachelor's Student at the Department of Information Technology Security
Kharkiv National University of Radio Electronics
ORCID: 0009-0008-5521-9831

O. YE. PETRENKO
Candidate of Technical Sciences, Associate Professor,
Associate Professor at the Department of Information Technology Security
Kharkiv National University of Radio Electronics
ORCID: 0000-0002-7862-5399

I. V. OLESHKO
Candidate of Technical Sciences, Associate Professor,
Associate Professor at the Department of Information Technology Security
Kharkiv National University of Radio Electronics
ORCID: 0000-0002-8021-0467

DESIGN OF FIXED MODULI MODULAR REDUCTION ALGORITHMS FOR CRYPTOSYSTEMS THAT USE EXTENDED BINARY FIELD ARITHMETIC

A number of current domestic standards, including the digital signature standard [1] and the block symmetric encryption standard [2], utilize extended binary field arithmetic GF(2^m) based on a polynomial basis for cryptographic transformations. To ensure a high level of cryptographic resilience, these standards recommend large values of m for the GF(2^m) field. Specifically, standard [1] defines 60 variants of irreducible polynomials f(t) with prime degrees m ranging from 163 to 509, while standard [2] uses 3 variants of irreducible polynomials f(t) for even values of m (128, 256, or 512), corresponding to the cipher's block sizes. Operations within the GF(2^m) field incur high computational complexity for large values of m. The operation of modular reduction by f(t) is a key limiting performance factor, and reducing its complexity is critical for real-time implementations.

General methods of modular reduction do not always account for the specific optimization properties of a particular irreducible modulus f(t) of degree m (a trinomial or a pentanomial). Developing reduction algorithms optimized specifically for the fixed values of f(t) specified by standards can decrease computational complexity, particularly in hardware implementations. This complexity reduction is achieved because the modulus f(t) remains invariant throughout the session, which enables the integration of pre-calculations directly into the structure of the reduction algorithms.

This article focuses on the creation and analysis of sets of algorithms for reduction by fixed moduli f(t), aiming to optimize basic operations in cryptosystems that utilize GF(2^m). We consider a method for creating such algorithms that enables their formalization and partial automation. The possibility of software automation is crucial, since standards recommend a large number of GF(2^m) field variants. Furthermore, it is also necessary to ensure a formalized approach for verifying the correctness and testing these implementations.

The research resulted in the development of a series of optimized modular reduction algorithms tied to the specific moduli recommended by Ukrainian cryptographic standards.

Key words: *modular reduction algorithms, extended binary fields arithmetic, reduction with fixed moduli, algorithms optimization, cryptographic algorithms.*

Постановка проблеми

Сучасний розвиток інформаційних технологій та зростання обсягів даних висувають підвищені вимоги до продуктивності та криптографічної стійкості засобів захисту інформації. Забезпечення конфіденційності, цілісності та автентичності інформації є критично важливою науковою та практичною задачею. Для її вирішення застосовують надійні криптографічні стандарти, як міжнародні, так і вітчизняні. Низка цих стандартів використовує операції в розширеному двійковому полі GF(2^m) для великих значень m. Як правило, в поліноміальному базисі, якщо мова не йде про виключно апаратні реалізації деяких алгоритмів, де має сенс застосовувати оптимальний нормальний базис.

Серед прикладів таких стандартів можна назвати цифровий підпис ДСТУ 4145-2002 [1], блоковий симетричний шифр "Калина" (ДСТУ 7624:2014) [2] в режимах GCM, GMAC, XTS/XTS-p. Міжнародні стандарти щодо режимів блокових симетричних шифрів (таких, як AES), які підтримують автентифікацію та спеціалізоване

шифрування сховищ (GCM, GMAC [3,4], XTS [5,6]). До них же відносяться різні міжнародні стандарти цифрового підпису в групах точок еліптичних кривих (ЕК) [7-10], які використовують варіант ЕК над полем GF(2^m), а також деякі стандарти протоколів встановлення ключів [11-12] та навіть несиметричного шифрування [13]. І це далеко не повний перелік подібних стандартів та криптографічних алгоритмів.

Реалізація подібних криптографічних перетворень вимагає високоефективного виконання операцій множення з модульною редукцією у розширених двійкових полях GF(2^m). Саме швидкість реалізації цих операцій часто стає обмежувальним фактором продуктивності всього криптографічного алгоритму. Проблема у загальному вигляді полягає у необхідності розробки нових, або суттєвого підвищення ефективності існуючих алгоритмів арифметики GF(2^m), здатних забезпечити необхідну пропускну здатність криптографічних пристроїв на рівні сучасних вимог.

В цій роботі розглядається розробка алгоритмів поліноміальної редукції зі зменшеною обчислювальною складністю для полів GF(2^m) з великими значеннями m. Наприклад такими, що використовуються в режимах блокового симетричного шифру [2] (m = 128, 256, 512 в залежності від бітового розміру блоку шифру) та в цифровому підпису [1], де значення m є простими числами в діапазоні від 163 до 509.

Аналіз останніх досліджень і публікацій

Математичний та алгоритмічний апарат виконання базових операцій в полі GF(2^m) і, зокрема, операції приведення по модулю незвідного полінома (тринома чи пентанома), представлено як в криптографічних стандартах, наприклад [1, 14], монографіях [15-18], так і в низці наукових публікацій [19-25]. При цьому аналізуються як варіанти поліноміального, так і оптимального нормального базису представлення елементів GF(2^m). Хоча для програмної реалізації найчастіше використовуються алгоритми в поліноміальному базисі (як більш швидкі), а нормальний базис зазвичай призначений для апаратних реалізацій.

Серед базових операцій в GF(2^m) найбільш обчислювально складними є множення та приведення за модулем (редукція). Можна виділити такі основні типи алгоритмів редукції (приведення за модулем):

1) побітова редукція;
2) побітовий варіант редукції з попередніми обчисленнями; зокрема, особливі варіанти “одночасної” редукції для модулів зі стандарту [2] (блоковий симетричний шифр “Калина” в режимах GMAC, GCM, XTS/XTS-p);
3) порозрядна редукція за *довільним* модулем (триномом / пентаномом) (зокрема варіанти, згадані в додатку В.3 стандарту [1]);
4) порозрядна редукція за заданим *незмінним* модулем (триномом / пентаномом) (один з прикладів якої наведено в різних літературних джерелах, зокрема, в [15, 19]).

В цьому контексті розмір “розрядів” зазвичай прив’язується до розміру машинного слова певної обчислювальної техніки (32, 64 розряди), хоча не виключено й використання більших значень (наприклад, 128 при застосуванні SSE). Алгоритми в цьому переліку наведено, здебільшого, в порядку зменшення їх обчислювальної складності. Хоча додатково треба враховувати як максимальний бітовий розмір m у представленні елементів базового поля GF(2^m), так і особливості обраних поліномів-модулів $f(t)$. Наприклад, чи є вони триномами чи пентаномами, як розташований в пам’яті старший біт модуля (на межі розрядів чи всередині), як згруповані в пам’яті молодші коефіцієнти модуля (в межах якої кількості розрядів) і т.і. Бо ці фактори можуть змінити співвідношення обчислювальної складності для деяких алгоритмів з цього списку (зокрема, для “одночасної” редукції).

Формулювання мети дослідження

Серед алгоритмів, наведених у списку вище, саме порозрядні алгоритми приведення по модулю потенційно можуть бути краще оптимізовані, в тому числі при апаратній реалізації, за рахунок більш точного використання можливостей розпаралеленої обробки, якщо реалізуються окремо для кожного варіанту незвідного полінома-модуля $f(t)$, який визначає розширене двійкове поле GF(2^m). Тобто це так званий варіант порозрядної редукції за заданим *фіксованим* модулем (триномом / пентаномом).

Мається на увазі що параметри певного поля в криптосистемі використовуються є незмінними (фіксованими) протягом досить довгого сеансу використання. Наприклад, для криптосистем з використанням ЕК над GF(2^m) варіанти ЕК (визначені для певного примітивного поліному-модулю степеню m) можуть змінюватись не часто, що дозволяє використовувати той самий незмінний поліном $f(t)$. А при використанні згаданих раніше режимів блокових симетричних шифрів m взагалі залежить від бітового розміру блоку шифру (наприклад, 128, 256, 512 для [2]), тобто взагалі не змінюється в межах певної версії алгоритму шифрування.

Проблемним в цій ситуації є те, що криптографічні стандарти можуть рекомендувати досить великі списки варіантів базового поля GF(2^m), тобто поліномів-модулів $f(t)$ степеню m. Так, наприклад, вітчизняний стандарт цифрового підпису [1] визначає 60 варіантів примітивних поліномів $f(t)$ для простих значень m в діапазоні від 163 до 509, а стандарт [9] в попередній версії використовував навіть більше значення m = 571. Тобто таких алгоритмів, окремо для кожного фіксованого модуля, має бути досить багато. В літературі зазвичай наводиться, в якості прикладів подібних алгоритмів, лише декілька варіантів для невеликих значень m (163, 167) [15, 19]. І, як правило, саме для пентаномів, а не для триномів. Тому потрібно сформувати методику побудови подібних алгоритмів, яку

можна програмно автоматизувати та полегшити завдання як по створенню формалізованих описів цих алгоритмів, так і по перевірці коректності та тестуванню створених на їх основі програмних реалізацій.

Викладення основного матеріалу дослідження

Формування параметрів алгоритмів порозрядної редукції при фіксованому модулі розглядається для розрядів розміром $W=32$ (біти). Для прикладу в роботі наведено сформовані програмно таблиці розрахункових коефіцієнтів для модулів з нижченаведеного переліку.

1) $f(t)=t^{503}+t^3+1$ – трином для максимального простого m, з рекомендованих в [1];
2) $f(t)=t^{509}+t^{23}+t^3+t^2+1$ – пентаном для максимального простого m, з рекомендованих в [1];
3) $f(t)=t^{512}+t^8+t^5+t^2+1$ – пентаном для максимального парного m, з рекомендованих в [2].

Демонстраційні варіанти модулів обиралися з міркувань покриття основних тестових випадків. Розрахункові коефіцієнти для побудування алгоритмів редукції за цими модулями наведено в таблицях 1-3, відповідно.

Побудування таблиці розрахункових коефіцієнтів за заданим модулем $f(t)$ при розміру розрядів $W=32$

Таблиця 1

Розрахункові коефіцієнти для триному $f(t)=t^{503}+t^3+1$; $W=32$

In	S_{32} **розряд**			**In**	S_{17} **розряд**			**In**	S_{16} **розряд**		
	i	r_1	r_2		i	r_1	r_2		i	r_1	r_2
Out	17	FFF	1FF	Out	2	FFF	1FF	Out	1	FF8	1FF
	16	FFFFF000	FFFFFE00		1	FFFFF000	FFFFFE00		-	-	-

Таблиця 2

Розрахункові коефіцієнти для пентаному $f(t)=t^{509}+t^{23}+t^3+t^2+1$; $W=32$

In	S_{32} **розряд**				
	i	r_1	r_2	r_3	r_4
Out	17	3FFFFFF	3F	1F	7
	16	FC000000	FFFFFFC0	FFFFFFE0	FFFFFFF8
In		S_{17} розряд			
	i	r_1	r_2	r_3	r_4
Out	2	3FFFFFF	3F	1F	7
	1	FC000000	FFFFFFC0	FFFFFFE0	FFFFFFF8
In	S_{16} розряд				
	i	r_1	r_2	r_3	r_4
Out	1	3800000	38	1C	7
	-	-			

Порівняння таблиці 1 з таблицями 2 та 3 демонструє спрощеність процесу формування розрахункових коефіцієнтів для триномів порівняно з пентаномами.

Таблиця 3

Розрахункові коефіцієнти для пентаному $f(t)=t^{512}+t^8+t^5+t^2+1$; $W=32$

In	S_{34} **розряд**				
	i	r_1	r_2	r_3	r_4
Out	19	FF	1F	3	0
	18	FFFFFF00	FFFFFFE0	FFFFFFFC	FFFFFFFF
In		S_{18} розряд			
	i	r_1	r_2	r_3	r_4
Out	3	FF	1F	3	0
	2	FFFFFF00	FFFFFFE0	FFFFFFFC	FFFFFFFF
In	S_{17} розряд				
	i	r_1	r_2	r_3	r_4
Out	2	FF	1F	3	0
	1	FFFFFF00	FFFFFFE0	FFFFFFFC	FFFFFFFF

Побудування алгоритму редукції по таблиці розрахункових коефіцієнтів для заданого модуля $f(t)$ при розміру розрядів $W=32$

Розглянемо побудування алгоритмів порозрядної редукції по сформованим раніше таблицям розрахункових коефіцієнтів за модулями з наведеного раніше переліку. В цьому розділі наведено лише декілька зі створених

алгоритмів приведення по модулю (3 з 68 побудованих тестових варіантів). Умовні позначення: M – розмір модуля $f(t)$ у $W_$бітових розрядах; M – вхідного $s(t)$, яке треба привести по модулю $f(t)$, у $W_$бітових розрядах (найчастіше $N=2{\cdot}M$, хоча теоретично може бути будь яким, в межах розрядної сітки, яку підтримує певна версія спеціалізованої бібліотеки для зберігання великих значень). Для всіх подальших алгоритмів:

IN: $s(t)=(s_{2\cdot m-2}\dots s_1 s_0)_2=(S_N\dots S_2 S_1)_{2^w}$.

OUT: $s(t)=s(t) \bmod f(t)$.

На виході алгоритму:

$\deg(s(t))\le m-1$, тобто $s(t)=(s_{t-1}\dots s_1 s_0)_2=(S_L\dots S_2 S_1)_{2^w}$; $t\le m-1$; $L\le M$.

Наведені приклади алгоритмів порозрядної редукції за фіксованим модулем побудовані з використанням розрахункових коефіцієнтів з таблиць 1-3.

Алгоритм 1. Редукція за модулем $f(t)=t^{503}+t^3+1$

1. for ($i=$ **N-1**; $i\ge$ **M-1** ; $i\_$-) // по W – бітовим розрядам $s(t)$

1.1. $T=S_i$;

1.2. $S_{i-16} \oplus= (T << 8) \oplus (T << 5) \oplus (T << 2) \oplus T$;

1.3. $S_{i-15} \oplus= (T >> 24) \oplus (T >> 27) \oplus (T >> 30)$;

5. // корекція (нормалізація) довжини s (t);

Для швидкої зміни варіантів модулів, яка не буде впливати на обчислювальну складність критичних за часом виконання відрізків програмного коду можна використовувати масив вказівників на окремі функції редукції за різними модулями. Обрання певного елементу з цього масиву (тобто обрання функції редукції за певним модулем) виконувати при ініціалізації поточного варіанту базового поля. Так, для цифрових підписів в групах точок ЕК це буде відбуватися лише один раз при початковому встановленні параметрів ЕК та базового поля, тобто нечасто та в некритичні проміжки часу. А для шифрів обрання функції редукції за певним модулем за рахунок звернення до певного елементу масиву вказівників на функції редукції відбуватиметься взагалі тільки при переході до іншої версії шифру (з іншою бітовою довжиною блока шифру).

Висновки

В ході досліджень створені та перевірені на коректність, в тому числі за рахунок тестування відповідного програмної бібліотеки, близько 68 оптимізованих алгоритмів приведення по фіксованим модулям, прив'язаних до рекомендацій криптографічних стандартів: 60 модулів з [1], 3 модулі з [2], 4 з [9] та декілька додаткових тестових значень. Всі алгоритми були перевірені на коректність та пройшли ретельне тестування за рахунок перевірки результатів відповідних програмних реалізацій. Експерименти показали зменшення часу виконання порівняно з алгоритмами редукції інших типів (побітових варіантів, порозрядних для довільних триномів або пентаномів). Зі збільшенням значень m різниця в часі виконання стає більш помітною.

На базі цієї ж методики можна перейти до узагальнення та побудувати алгоритми редукції $W\ge 64$. Це дозволить зменшити обчислювальну складність алгоритмів редукції через зменшення кількості ітерацій основного циклу.

Список використаної літератури

1. ДСТУ 4145-2002. Інформаційні технології. Криптографічний захист інформації. Цифровий підпис, що ґрунтується на еліптичних кривих. Формування та перевіряння: [Чинний від 2003–07–01]. Київ: Держстандарт України, 2002. 35 с. (Національний стандарт України).

2. ДСТУ 7624:2014. Інформаційні технології. Криптографічний захист інформації. Алгоритм симетричного блокового перетворення: [Чинний від 2015–07–01]. Київ: Мінекономрозвитку України, 2014. 228 с. (Національний стандарт України).

3. NIST SP 800-38D. Recommendation for Block Cipher Modes of Operation: Galois/Counter Mode (GCM) and GMAC. Gaithersburg, MD: *National Institute of Standards and Technology (NIST)*, 2007. 39 p. (NIST Special Publication).

4. ISO/IEC 19772:2020. Information security – Authenticated encryption. Geneva: *ISO/IEC*, 2020. 40 p.

5. NIST SP 800-38E. Recommendation for Block Cipher Modes of Operation: The XTS-AES Mode for Confidentiality on Storage Devices. Gaithersburg, MD: *National Institute of Standards and Technology (NIST)*, 2010. 21 p. (NIST Special Publication).

6. IEEE Std 1619-2007. Standard for Cryptographic Protection of Data on Block-Accessible Storage Devices. New York: *Institute of Electrical and Electronics Engineers*, 2008. 48 p. (Reaffirmed 2018).

7. ДСТУ ISO/IEC 14888-3:2018. Інформаційні технології. Методи захисту. Цифрові підписи з додатком. Частина 3. Механізми на основі дискретного логарифму: [Чинний від 2019–01–01]. Київ: ДП «УкрНДНЦ», 2018. 114 с. (Національний стандарт України).

8. ANSI X9.62-2005. Public Key Cryptography for the Financial Services Industry: The Elliptic Curve Digital Signature Algorithm (ECDSA). Washington, D.C.: *ANSI*, 2005. 50 p. (Reaffirmed 2011).

9. FIPS 186-4. Digital Signature Standard (DSS). Gaithersburg, MD: *National Institute of Standards and Technology (NIST)*, 2013. 98 p. (U.S. Department of Commerce).

10. IEEE 1363a-2004. Standard Specifications for Public-Key Cryptography – Amendment 1: Additional Public-Key Techniques. New York: *Institute of Electrical and Electronics Engineers*, 2004. 70 p. (Amendment to IEEE 1363-2000).

11. NIST SP 800-56A Rev. 3. Recommendation for Pair-Wise Key-Establishment Schemes Using Discrete Logarithm Cryptography. Gaithersburg, MD: *National Institute of Standards and Technology (NIST)*, 2018. 143 p. (NIST Special Publication).

12. ANSI X9.63-2011. Public Key Cryptography for the Financial Services Industry: Key Establishment Techniques. Washington, D.C.: *ANSI*, 2011. 82 p. (Reaffirmed 2016).

13. ISO/IEC 18033-2:2006. Information technology – Security techniques – Encryption algorithms – Part 2: Asymmetric ciphers. Geneva: *ISO/IEC*, 2006. 138 p.

14. ДСТУ ISO/IEC 15946-1:2017. Інформаційні технології. Методи захисту. Криптографічні методи, що ґрунтуються на еліптичних кривих. Частина 1. Загальні положення: [Чинний від 2017–12–18]. Київ: УкрНДНЦ, 2017. 42 с. (Національний стандарт України).

15. Hankerson D.R., Menezes A.J., Vanstone S.A. Guide to Elliptic Curve Cryptography. New York, NY: *Springer-Verlag*, 2004. 320 p.

16. Cohen H., Frey G., Bailey D. H. та ін. Handbook of Elliptic and Hyperelliptic Curve Cryptography. Boca Raton: *Chapman and Hall/CRC*, 2006. 824 p.

17. Washington L. C. Elliptic Curves: Number Theory and Cryptography. Boca Raton, FL: *Chapman and Hall/CRC*, 2008. 536 p.

18. Menezes A. J., van Oorschot P. C., Vanstone S. A. Handbook of Applied Cryptography. Boca Raton, FL: *CRC Press*, 1996. 816 p.

19. Hankerson D., López J. H., Menezes A. J. Software implementation of the NIST elliptic curves over binary fields. В *Selected Areas in Cryptography: International Workshop (SAC 2000)*. Berlin; Heidelberg: Springer, 2001. P. 1–19. (Lecture Notes in Computer Science ; Vol. 2012).

20. Hasan S. M. A., Bandara M. L. S. D. K. S., Al-Amin S. M. K. та ін. An Efficient and Lightweight Architecture for Modular Reduction in GF(2^m) based on Hybrid Polynomial Representation. *IEEE Access*. 2024. Vol. 12. P. 49302–49313.

21. Ghouma S. A., Al-Sarayrah F. S., Al-Shurman O. A. M. High-Performance Polynomial Basis Multiplier for GF(2^m) using Karatsuba-like Algorithm. IEEE Access. 2023. Vol. 11. P. 11054–11065.

22. Chowdhury M. M. I., Hasan M. A. Efficient hardware implementation of extended Euclidean algorithm for GF(2^m) on FPGA. *Microprocessors and Microsystems*. 2023. Vol. 101. 104889.

23. Islam M. S., Hwang Y. S. High-performance and efficient finite field arithmetic for cryptographic applications. *Journal of Circuits, Systems and Computers*. 2022. Vol. 31, Issue 4. 2250064 (16 p.).

24. Hasan M. A., Lwin T. K., Tso K. S. Efficient Implementation of Binary Field Multipliers using Modified Karatsuba Algorithm. *IEEE Access*. 2021. Vol. 9. P. 110903–110915.

25. Certik J., Hankerson D., Menezes A. High-speed software implementation of the NIST recommended elliptic curves over GF(2^m). *IEEE Transactions on Computers*. 2003. Vol. 52, Issue 8. P. 1060–1070.

References

1. Derzhstandart Ukrainy. (2002). Informatsiini tekhnolohii. Kryptohrafichnyi zakhyst informatsii. Tsyfrovyi pidpys, shcho hruntuiet'sia na eliptychnykh kryvykh. Formuvannia ta pereviriannia [Information technologies. Cryptographic protection of information. Digital signature based on elliptic curves. Generation and verification] (DSTU 4145-2002). Kyiv: Derzhstandart Ukrainy. [in Ukrainian].

2. Minekonomrozvytku Ukrainy. (2014). Informatsiini tekhnolohii. Kryptohrafichnyi zakhyst informatsii. Algorytm symetrychnoho blokovoho peretvorennia [Information technologies. Cryptographic protection of information. Symmetric block cipher algorithm] (DSTU 7624:2014). Kyiv: Minekonomrozvytku Ukrainy. [in Ukrainian].

3. NIST. (2007). Recommendation for Block Cipher Modes of Operation: Galois/Counter Mode (GCM) and GMAC (SP 800-38D). *National Institute of Standards and Technology*.

4. ISO/IEC. (2020). Information security – Authenticated encryption (ISO/IEC 19772:2020). *ISO/IEC*. Retrieved from https://www.iso.org/standard/76508.html

5. NIST. (2010). Recommendation for Block Cipher Modes of Operation: The XTS-AES Mode for Confidentiality on Storage Devices (SP 800-38E). *National Institute of Standards and Technology*.

6. IEEE. (2008). Standard for Cryptographic Protection of Data on Block-Accessible Storage Devices (Std 1619-2007). *Institute of Electrical and Electronics Engineers*. https://doi.org/10.1109/IEEESTD.2008.4554890

7. DP "UkrNDNTs". (2018). Informatsiini tekhnolohii. Metody zakhystu. Tsyfrovi pidpysy z dodatkom. Chastyna 3. Mekhanizmy na osnovi dyskretnoho loharyfmu [Information technology. Security techniques. Digital signatures with appendix. Part 3. Discrete logarithm-based mechanisms] (DSTU ISO/IEC 14888-3:2018). Kyiv: DP "UkrNDNTs". [in Ukrainian].

8. ANSI. (2005). Public Key Cryptography for the Financial Services Industry: The Elliptic Curve Digital Signature Algorithm (ECDSA) (X9.62-2005). *American National Standards Institute*.

9. NIST. (2013). Digital Signature Standard (DSS) (FIPS 186-4). *National Institute of Standards and Technology*.

10. IEEE. (2004). Standard Specifications for Public-Key Cryptography – Amendment 1: Additional Public-Key Techniques (IEEE 1363a-2004). *Institute of Electrical and Electronics Engineers*. https://doi.org/10.1109/IEEESTD.2004.95727

11. NIST. (2018). Recommendation for Pair-Wise Key-Establishment Schemes Using Discrete Logarithm Cryptography (SP 800-56A Rev. 3). *National Institute of Standards and Technology*. https://doi.org/10.6028/NIST.SP.800-56Ar3

12. ANSI. (2011). Public Key Cryptography for the Financial Services Industry: Key Establishment Techniques (X9.63-2011). *American National Standards Institute*.

13. ISO/IEC. (2006). Information technology – Security techniques – Encryption algorithms – Part 2: Asymmetric ciphers (ISO/IEC 18033-2:2006). *ISO/IEC*. Retrieved from https://www.iso.org/standard/40733.html

14. UkrNDNTs. (2017). Informatsiini tekhnolohii. Metody zakhystu. Kryptohrafichni metody, shcho hruntuiut'sia na eliptychnykh kryvykh. Chastyna 1. Zahalni polozhennia [Information technology. Security techniques. Cryptographic techniques based on elliptic curves. Part 1. General] (DSTU ISO/IEC 15946-1:2017). Kyiv: UkrNDNTs. [in Ukrainian].

15. Hankerson, D. R., Menezes, A. J., & Vanstone, S. A. (2004). Guide to Elliptic Curve Cryptography. Springer-Verlag. https://doi.org/10.1007/b97644

16. Cohen, H., Frey, G., Avanzi, R. M., Doche, C., Lange, T., Nguyen, K., & Vercauteren, F. (2006). Handbook of Elliptic and Hyperelliptic Curve Cryptography. *Chapman and Hall/CRC*. https://doi.org/10.1201/9781420010903

17. Washington, L. C. (2008). Elliptic Curves: Number Theory and Cryptography. *Chapman and Hall/CRC*. https://doi.org/10.1201/9781420011801

18. Menezes, A. J., van Oorschot, P. C., & Vanstone, S. A. (1996). Handbook of Applied Cryptography. *CRC Press*. https://doi.org/10.1201/9781439821916

19. Hankerson, D., López, J. H., & Menezes, A. J. (2001). Software implementation of the NIST elliptic curves over binary fields. In *Selected Areas in Cryptography*: International Workshop (SAC 2000) (Vol. 2012, pp. 1–19). Springer. https://doi.org/10.1007/3-540-45372-5_1

20. Hasan, S. M. A., Bandara, M. L. S. D. K. S., Al-Amin, S. M. K., Al-Qutayri, M., & Al-Haj Hasan, M. I. (2024). An Efficient and Lightweight Architecture for Modular Reduction in GF(2^m) based on Hybrid Polynomial Representation. *IEEE Access*, 12, 49302–49313. https://doi.org/10.1109/ACCESS.2024.3385759

21. Ghouma, S. A., Al-Sarayrah, F. S., & Al-Shurman, O. A. M. (2023). High-Performance Polynomial Basis Multiplier for GF(2^m) using Karatsuba-like Algorithm. *IEEE Access*, 11, 11054–11065. https://doi.org/10.1109/ACCESS.2023.3238618

22. Chowdhury, M. M. I., & Hasan, M. A. (2023). Efficient hardware implementation of extended Euclidean algorithm for GF(2^m) on FPGA. *Microprocessors and Microsystems*, 101, 104889. https://doi.org/10.1016/j.micpro.2023.104889

23. Islam, M. S., & Hwang, Y. S. (2022). High-performance and efficient finite field arithmetic for cryptographic applications. *Journal of Circuits, Systems and Computers*, 31(4), 2250064. https://doi.org/10.1142/S021812662250064X

24. Hasan, M. A., Lwin, T. K., & Tso, K. S. (2021). Efficient Implementation of Binary Field Multipliers using Modified Karatsuba Algorithm. IEEE Access, 9, 110903–110915. https://doi.org/10.1109/ACCESS.2021.3101569

25. Certik, J., Hankerson, D., & Menezes, A. (2003). High-speed software implementation of the NIST recommended elliptic curves over GF(2^m). *IEEE Transactions on Computers,* 52(8), 1060–1070. https://doi.org/10.1109/TC.2003.1223933

Дата першого надходження рукопису до видання: 21.11.2025
Дата прийнятого до друку рукопису після рецензування: 18.12.2025
Дата публікації: 31.12.2025