Научная статья
УДК 004.056
https://doi.org/10.31854/1813-324X-2025-11-2-101-108
EDN:DWRJHM

Реализация электронной подписи ECC в ограниченных средах

Наурас Хуссейн Сабри, nawrashussein@mail.ru

Санкт-Петербургский государственный электротехнический университет «ЛЭТИ» им. В.И. Ульянова (Ленина), Санкт-Петербург, 197022, Российская Федерация

Аннотация

Обеспечение безопасных операций криптографии в средах с ограниченными ресурсами представляет собой сложную задачу из-за ограниченной вычислительной мощности и памяти. В условиях стремительного роста беспилотных транспортных систем возрастает потребность в эффективных и безопасных криптографических решениях. Оптимизация криптографических алгоритмов для таких систем становится особенно актуальной с учетом их ограниченных вычислительных ресурсов и высоких требований к безопасности.

***Целью** данного исследования является оптимизация операций электронной подписи на основе эллиптической кривой (ECC) для систем с ограниченными ресурсами, в частности для беспилотных транспортных систем. Исследование направлено на повышение вычислительной эффективности и снижение использования памяти, делая механизмы безопасности на основе ECC более подходящими для встроенных приложений.*

***Новизна** данного исследования заключается в интеграции множества методов оптимизации. Улучшается скалярное умножение точки, используя свойства циклической группы, противоположного числа, а также усовершенствованный оконный метод умножения. Кроме того, вводится детерминированный метод генерации одноразового используемого числа (nonc), вдохновленный EdDSA, для дальнейшего повышения эффективности цифровой подписи. Эти оптимизации в совокупности способствуют более эффективному криптографическому процессу, подходящему для сред с ограниченными ресурсами.*

***Теоретическая значимость** заключается в разработке нового математического аппарата, позволяющего оптимизировать операции электронной подписи.*

***Практическая значимость** данного исследования заключается в его применимости в маломощных встраиваемых системах, где вычислительные ресурсы и память крайне ограничены. Оптимизируя операции ECC, это исследование повышает безопасность и производительность криптографических реализаций в беспилотных транспортных системах и аналогичных встраиваемых приложениях, обеспечивая безопасную связь без превышения аппаратных ограничений.*

***Реализация предложенного метода** была осуществлена на микроконтроллере ATmega 2560, **полученные результаты показывают** сокращения количества циклов на 54,1 % и уменьшения использования SRAM на 72,6 % при генерации ключей, а также значительного повышения производительности в процессах подписи и проверки. Экспериментальные результаты подтверждают его эффективность в оптимизации операций ECC для ограниченных устройств беспилотных транспортных систем.*

Ключевые слова: *электронная подпись, умножение точек эллиптической кривой, устройства с ограниченными ресурсами, беспилотные транспортные системы, микроконтроллер*

Финансирование. *Работа выполнена в рамках государственного задания Министерства науки и высшего образования Российской Федерации № 075-00003-24-01 от 08.02.2024 (проект FSEE-2024-0003).*

Ссылка для цитирования: Сабри Н.Х. Реализация электронной подписи ECC в ограниченных средах // Труды учебных заведений связи. 2025. Т. 11. № 2. С. 101–108. DOI:10.31854/1813-324X-2025-11-2-101-108. EDN:DWRJHM

Original research
https://doi.org/10.31854/1813-324X-2025-11-2-101-108
EDN:DWRJHM

Implementation of an ECC Digital Signature Technique in Constrained Environments

Nawras H. Sabbry, nawrashussein@mail.ru

Saint Petersburg Electrotechnical University,
St. Petersburg, 197022, Russian Federation

Annotation

Relevance. *Ensuring secure cryptographic operations in resource-constrained environments presents challenges due to limited computational power and memory. With the rapid growth of Unmanned Vehicle Systems, the need for efficient and secure cryptographic solutions is increasing. Optimizing cryptographic algorithms for such systems becomes especially relevant given their limited computational resources and high security demands.*
The ***purpose*** *of this study is to optimize Elliptic Curve Cryptography (ECC) digital signature operations for resource-limited systems, particularly in unmanned vehicle systems. The research aims to enhance computational efficiency and reduce memory usage, making ECC-based security mechanisms more feasible for embedded applications.*
The ***novelty*** *of this study lies in its integration of multiple optimization techniques. It improves scalar point multiplication by leveraging cyclic group properties, additive inverses, and an enhanced windowed multiplication method. Additionally, it introduces a deterministic nonce generation approach inspired by EdDSA to further refine digital signature efficiency. These innovations collectively contribute to a more efficient cryptographic process suitable for constrained environments.*
The ***theoretical significance*** *lies in the development of a new mathematical apparatus that makes it possible to optimize electronic signature operations.*
The ***practical significance*** *of this study is its applicability in low-power embedded systems, where computational and memory resources are highly limited. By optimizing ECC operations, this research enhances the security and performance of cryptographic implementations in unmanned vehicle systems and similar embedded applications, ensuring secure communications without exceeding hardware constraints.*
The proposed method *was implemented on the Arduino Atmega 2560 R3, achieves up to a 54,1 %* ***results are showing*** *reduction in cycle count and a 72,6 % decrease in SRAM usage for key generation, alongside significant performance improvements in signing and verification processes. Experimental results confirm its effectiveness in optimizing ECC operations for constrained devices of unmanned vehicle systems.*

Keywords: *digital signature, elliptic curve point multiplication, resource-constrained devices, unmanned vehicle systems (UVS), microcontroller*

Funding. *This research was funded by the Ministry of Science and Higher Education of the Russian Science Foundation (No. 075-00003-24-02, FSEE-2024-0003).*

For citation: Sabbry N.H. Implementation of an ECC Digital Signature Technique in Constrained Environments. *Proceedings of Telecommunication Universities.* 2025;11(2):101–108. (in Russ.) DOI:10.31854/1813-324X-2025-11-2-101-108. EDN:DWRJHM

Введение

В современном мире обеспечение безопасности в средах с ограниченными ресурсами (ограниченных средах) является критически важной задачей. Криптография на эллиптических кривых (ECC, *аббр. от англ.* Elliptic Curve Cryptography) обеспечивает высокий уровень безопасности при небольших размерах ключей, что делает ее идеальной для таких устройств, как микроконтроллеры в беспилотных транспортных системах. Среди применений ECC – схемы электронной подписи, такие как

ECDSA [1] и EdDSA [2], они гарантируют аутентификацию и целостность данных. Однако вычислительные требования для операции скалярного умножения точки, используемых в ECC, создают сложности для устройств с ограниченной памятью и вычислительной мощностью.

В данном исследовании предлагается оптимизированный метод электронной подписи ECC для сред с ограниченными ресурсами. Оптимизация включает два ключевых этапа: (1) улучшение базовой операции ECC скалярного умножения точки и (2) повышение эффективности электронной подписи за счет детерминированной генерации закрытых однократно используемых чисел (nonc).

Первый этап расширяет предшествующее исследование [3], используя свойства циклической группы и противоположных чисел для оптимизации умножения точек. В частности, он снижает сложность удвоения точки в расширенных скрученных кривых Эдвардса (ε^e) [4] и интегрирует эти свойства в оконный метод умножения точек на Edwards25519.

Второй этап основывается на [5], в данной работе представлена идея оптимизации алгоритма электронной подписи за счет детерминированной генерации закрытых однократно используемых чисел, путем комбинирования сообщения с хешированным закрытым ключом (x), аналогично алгоритму, используемому в EdDSA.

Эти оптимизации были реализованы на микроконтроллере ATmega 2560 R3 для оценки их эффективности в средах с ограниченными ресурсами. Результаты демонстрируют значительное сокращение времени выполнения, использование памяти и числа тактов при реализации на низкоуровневом C и ассемблера, подчеркивая возможность применения предложенных разработок в реальных криптографических приложениях.

Структура данной работы следующая: раздел 2 описывает предлагаемую методологию, раздел 3 представляет оценку производительности, раздел 4 содержит ключевые выводы и направления будущих исследований.

Предложенная методология

Предлагаемая методология основана на реализации оптимизированных техник в рамках кривой Edwards25519, используя ε^e и оконный метод скалярного умножения точки.

Оптимизированная стратегия скалярного умножения точки для устройств с ограниченными ресурсами

Свойства циклической группы и противоположного числа могут быть интегрированы с проверенными методами умножения точек [3]. В данном исследовании эти свойства интегрируются с оконным методом скалярного умножения точки при размере окна $\omega = 4$ на кривой Edwards25519, используя ε^e.

В деталях методологию данного подхода можно представить следующим образом.

1) Свойство циклической группы эллиптической кривой.

Точки на эллиптической кривой образуют циклическую группу по сложению, т. е. существует базовая точка P такая, что любая точка может быть выражена как $Q = scalar \cdot P$. Порядок группы $\#E$ определяет момент повторения последовательности, удовлетворяя $\#E \cdot P = O$ (точка на бесконечности), $(\#E + 1) \cdot P = P$, $(\#E + 2) \cdot P = 2P$ и так далее циклически [6]. Например, вместо вычисления $(\#E + 2)\ P$ через несколько умножений оно эффективно выводится из $2P$, минимизируя общее количество вычислений.

2) Свойство противоположного числа в группах эллиптической кривой.

В теории групп свойство обратного элемента означает, что для каждой точки P существует обратная точка $-P$, такая, что $P + (-P) = O$, где O – единичный элемент. Для уменьшенных скаляров $\tilde{s}$, превышающих половину порядка эллиптической кривой $\#E$, свойство противоположного числа в теории групп обеспечивает дополнительную оптимизацию, используя обратные точки на эллиптической кривой.

3) Исключение параметров в ε^e.

Как обсуждается в [4], ε^e упрощает сложение, исключая параметры d и a, если $a = -1$. Однако при удвоении точки a остается как:

$$D \leftarrow a \cdot A,$$

где $a = -1 (\mathrm{mod}\ p)$; p – простой модуль, определяющий конечное поле, $p = 2^{255} - 19$.

В этом контексте a хранится как 32-байтовый массив в формате little-endian в 8-разрядных микроконтроллерах: $a = \{0xEC, 0xFF, \ldots, 0xFF, 0x7F\}$.

Вместо вычисления $D = -A$ через умножение $D = a \cdot A$ предложенный метод использует противоположные числа в конечных полях, заменяя умножение вычитанием: $D = p - A$.

Эта оптимизированная техника инверсии упрощает $D = -A$, снижая вычислительную сложность. Поскольку вычитание более эффективно, чем умножение, этот подход повышает производительность в устройствах с ограниченными ресурсами, особенно в оконном методе скалярного умножения точки, где частые операции удвоения выигрывают за счет уменьшенного числа вычислительных циклов.

4) Интеграция с существующими методами.

Интеграция предложенного подхода с оконным методом умножения точек эллиптической кривой осуществляется в два этапа.

Этап 1. Необходимо проверить, что скаляр s находится в пределах порядка эллиптической кривой $[0, \#E - 1]$ в зависимости от свойства циклической группы эллиптической кривой:

Пока $s \geq \#E$: Сокращенный скаляр $\tilde{s} = s - \#E$.

Этап 2. Построение оптимизированной функции оконного умножения точек.

Входные данные:
– $\tilde{s}$ или оптимизированный скаляр $\hat{\tilde{s}}$;
– базовая точка G;
– $\#E$;
– $\mathcal{p}$;
– размер окна ω (определяет длину окна в оконном методе);
– предварительно вычисленные точки (P, $2P$, $3P$, …, $P_{2^{\omega-1}}$).

Выходные данные: аффинные координаты (x, y) результирующей точки ($Q = s \cdot G$).

Описание алгоритма на псевдокоде:

1) Если ($\tilde{s} > \frac{\#E}{2}$), использовать свойство противоположного числа в группах точек эллиптической кривой:

– оптимизированный скаляр $\tilde{s} = \hat{\tilde{s}} - \frac{\#E}{2}$;

– вызвать *«Оптимизированная функция оконного умножения точек»*;

– Q_x (x-аффинная координата) = $\mathcal{p}$ (инверсная точка (x-координата));

– Q_y (y-аффинная координата) = $\mathcal{p}$ (инверсная точка (y-координата));

– осуществить возврат.

2) Q = *«Предвычисленная точка»* (наиболее значимое окно).

3) Главный цикл (итерация по окнам).

Внутренний цикл (итерация ω раз):

$$Q = 2Q \,//\, \text{Удвоение},$$

$Q = Q$ + Предвычисленные точки [текущее значение окна].

Классический оконный метод включает удвоение точки ω раз, а затем операцию сложения с предварительно вычисленной точкой, если значение окна не равно нулю. В противном случае сложение пропускается [7], что вызывает несоответствия в выполнении и делает реализацию уязвимой к атакам по времени [8, 9]. Чтобы обеспечить последовательное выполнение, в метод добавляется дополнительная предварительно вычисленная единичная точка $0G$, представленная как $(0 : 1 : 0 : 1)$ в расширенной проективной системе координат кривой ε^e. Так как сложение любой точки с $0G$ сохраняет исходную точку, это изменение поддерживает целостность операций. Размер массива предварительно вычисленных точек увеличивается следующим образом: ω = 4: 17 точек (16 стандартных + $0G$).

С добавлением $0G$ сложение выполняется равномерно, обеспечивая реализацию с постоянным временем выполнения. Поскольку безопасность ECC зависит от половины битовой длины порядка группы эллиптической кривой (из-за того, что Ро-алгоритм Полларда решает ECDLP за $O(\sqrt{\#E})$ [10], сокращение количества операций умножения на основе порядка кривой не ослабляет безопасность скалярного умножения точки.

Оптимизированный алгоритм электронной подписи на основе эллиптических кривых для устройств с ограниченными ресурсами

Оптимизация электронной подписи включает:
– использование аналогичного подхода к генерации закрытых ключей, как в алгоритме электронной подписи (EdDSA), для генерации детерминированных nonc; в данном исследовании предлагается генерировать одноразовое число путем комбинирования сообщения с хешированным значением закрытого ключа (x);
– исключение открытого одноразового числа (R) из расчета вызова $c = H_{sig}(X, R, m)$.

Алгоритм генерации и проверки подписи представляет собой последовательность действий.

Во-первых, чтобы подписать сообщение (m), подписант выполняет следующие действия.

1) Выбирает закрытый ключ (x) такой, что:

$$0 < x < \mathcal{p}.$$

2) Открытый ключ (X): $X = G^x$.

3) Вычисляет $h = H(prefix)$.

4) Вычисляет закрытое одноразовое число:

$$r = H(h \,||\, m).$$

5) Вычисляет открытое одноразовое число:

$$R = G^r.$$

6) Вычисляет $c = H(X, m)$.

7) Вычисляет $s \equiv (r + cx)$.

Подпись представляет собой пару (R, s). Таким образом, подписант отправляет сообщение (m), открытый ключ (X), (R) и (s).

В-вторых, проверяющий считает подпись действительной, только если:

$$G^s = RX^c.$$

Доказательство корректности алгоритма:

$$G^s = RX^c,$$

т. к. $R = G^r$ и $X = G^x$, то $G^s = G^r(G^x)^c = G^{r+xc} = G^s$, где $s = r + xc$.

Реализация предложенного метода на микроконтроллере ATmega2560

Алгоритмы электронной подписи обычно включают в себя генерацию ключей, подписание и проверку [2].

Процесс генерации ключей электронной подписи состоит из трех этапов.

Этап 1. Генерация случайных чисел (RNG, *аббр. от англ.* Random Number Generator)

Аппаратный источник энтропии (аналоговый вход на контакте A0) фиксирует шум окружающей среды в качестве исходного значения. Потоковый шифр ChaCha20 обрабатывает эту энтропию в криптографически безопасный 32-байтовый случайный выходной сигнал [11]. Криптографические шифры и хеш-функции обеспечивают надежную случайность для получения префикса и закрытого ключа, требуя 117,618 тактов.

Этап 2. Хеширование

Выходные данные RNG хешируются с использованием 512-битной хеш-функции, формируя 64-байтовое значение, разделенное на две 32-байтовые части [2]:

– нижние 32 байта формируют закрытый ключ;

– верхние 32 байта используются как префикс для генерации закрытого nonc, комбинируемого с сообщением при подписании.

Этот этап требует 277 575 тактов.

Этап 3. Вычисление открытого ключа

Закрытый ключ умножается на базовую точку для получения открытого ключа с использованием метода, представленного в [3], при ω = 4, требующего 15 105 854 тактов, используя 30 518 байтов (12 %) из доступных 253 952 байтов памяти программ (флеш-памяти) и потребляя 429 байтов (5 %) статической памяти с произвольным доступом (SRAM, *аббр. от англ.* Static Random Access Memory), оставляя 7 763 байта из максимальных 8 192 байтов.

Для оценки эффективности предлагаемого метода генерации ключей в электронной подписи сравнения должны проводиться в идентичных средах и с использованием одной и той же кривой, как в [12]. Сравнения с исследованиями, использующими значительно отличающиеся аппаратные условия, могут исказить точность усовершенствований предложенного метода. В таблице 1 и на рисунке 1 представлено это сравнение, демонстрирующее результаты проведенного авторами исследования и [12], оба из которых используют микроконтроллер ATmega 2560 и кривую Edwards25519. Для реализации предлагаемого метода был использован язык *C* и ассемблер.

При сравнении с высокоскоростной реализацией (High-Speed) [12] предлагаемый метод сокращает количество необходимых тактов на 31,1 % и уменьшает использование SRAM на 72,6 %. Аналогично, по сравнению с реализацией, ориентированной на минимальное использование ресурсов (Low-Area), предлагаемый метод обеспечивает сокращение тактов на 54,1 % и уменьшение использования SRAM на 66,5 %.

ТАБЛИЦА 1. Сравнение генерации ключей: предложенный метод vs исследование в сопоставимых условиях [12]

TABLE 1. Key Generation Comparison: Proposed Method vs Prior Research under Comparable Conditions [12]

Реализация	Требуемые такты	SRAM, байт
crypto_sign_keypair (High-Speed) [16] (w = 4)	21924771	1566
crypto_sign_keypair (Low-Area) [16] (w = 2)	32937,940	1282
Предложенный метод (w = 4)	15105854	429

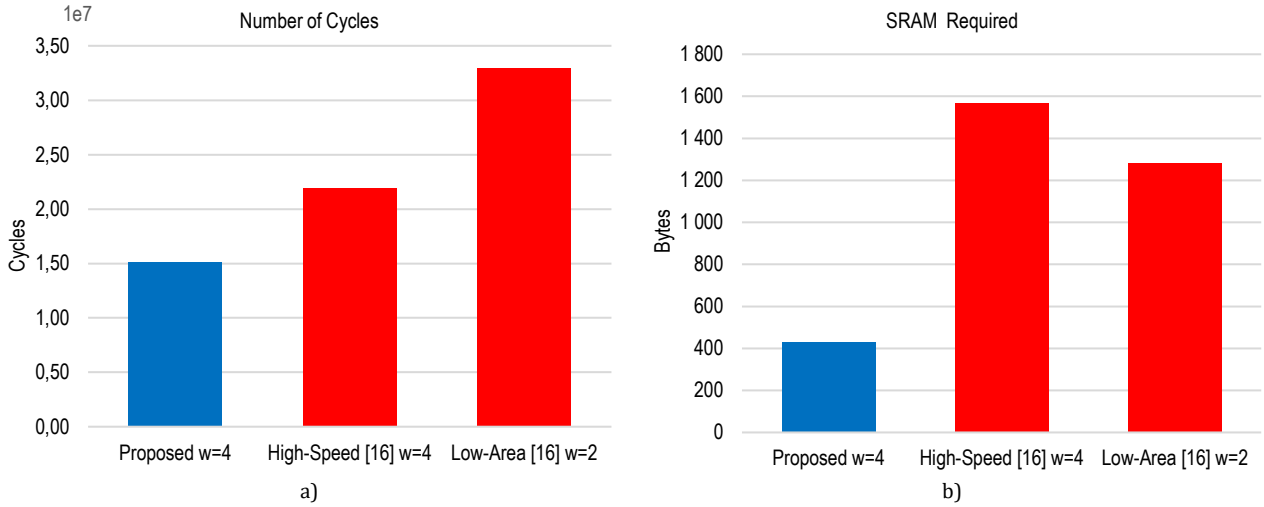

Рис. 1. Сравнение генерации ключей: предложенный метод (a) с предыдущими исследованиями (b) в сопоставимых условиях [12]

Fig. 1. Key Generation Comparison: Proposed Method (a) vs Prior Research (b) under Comparable Conditions [12]

В процессе подписания производительность предлагаемого метода требует 15 435 539 тактов, используя 25 652 байта (10 %) памяти программ (флеш-памяти) и 639 байтов (7 %) SRAM. Для оценки эффективности предлагаемого метода подписания проводятся сравнения с предыдущими исследованиями в аналогичных средах с ограниченными ресурсами. В таблице 2 и на рисунке 2 представлено сравнение количества тактов из данного исследования с результатами из crypto_sign (High-Speed) и crypto_sign (Low-Area) [12, 13], выполненных в сопоставимых условиях. Эти исследования были выбраны для обеспечения справедливой и точной оценки, так как сравнение с исследованиями, использующими существенно отличающиеся аппаратные условия, могло бы снизить четкость и надежность оценки производительности предлагаемого метода.

ТАБЛИЦА 2. Сравнение количества циклов: предложенный метод с предыдущими исследования в сопоставимых условиях [12, 13]

TABLE 2. Number of Cycles Comparison: Proposed Method vs Prior Researches under Comparable Conditions [12, 13]

Метод	Реализация	Количество циклов
Предложенный метод	ω = 4	15435539
Ed25519 [13]	–	22688583
crypto_sign [12]	High-Speed, ω = 4 Low-Area, ω = 2	23211611 34342230

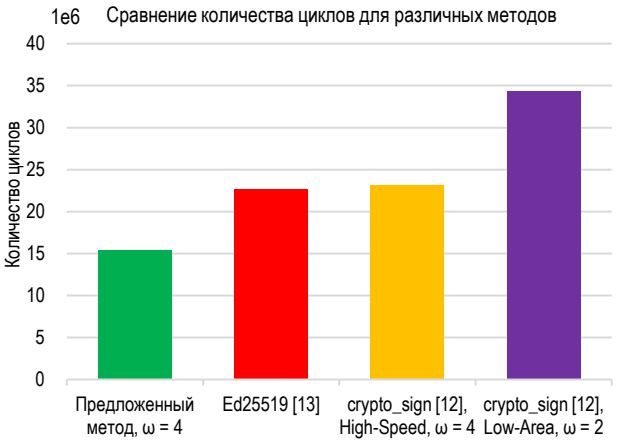

Рис. 2. Сравнение количества циклов: предложенный метод с исследования в сопоставимых условиях [12, 13]

Fig. 2. Number of Cycles Comparison: Proposed Method vs Prior Researches under Comparable Conditions [12, 13]

Предлагаемый метод требует меньше тактов: на 31,97 %, чем у Ed25519 [13], и на 33,50 %, чем у высокоскоростной реализации crypto_sign [12]. При сравнении с реализацией crypto_sign, ориентированной на минимальное использование ресурсов [12], предлагаемый метод достигает сокращения числа тактов на 55,05 %.

В процессе проверки производительность предлагаемого метода составляет 30 521 674 такта, используя 28 624 байта (11 %) памяти программ (флеш-памяти) и 1 187 байтов (14 %) SRAM. В таблице 3 и на рисунке 3 представлено сравнение между предлагаемым методом и подходами High-Speed и Low-Are), описанными в [12], выполненным в аналогичных средах с ограниченными ресурсами.

ТАБЛИЦА 3. Сравнение результатов этапа проверки: предложенный метод с предыдущими исследования в сопоставимых условиях

TABLE 3. Comparison of Verification Step Results: Proposed Method vs Prior Research under Comparable Conditions

Реализация	ω	Количество циклов	Требуемая SRAM, байт
Предложенный метод	4	30521674	1187
High-Speed [12]	4	32619197	1317
Low-Area [12]	2	40093186	1349

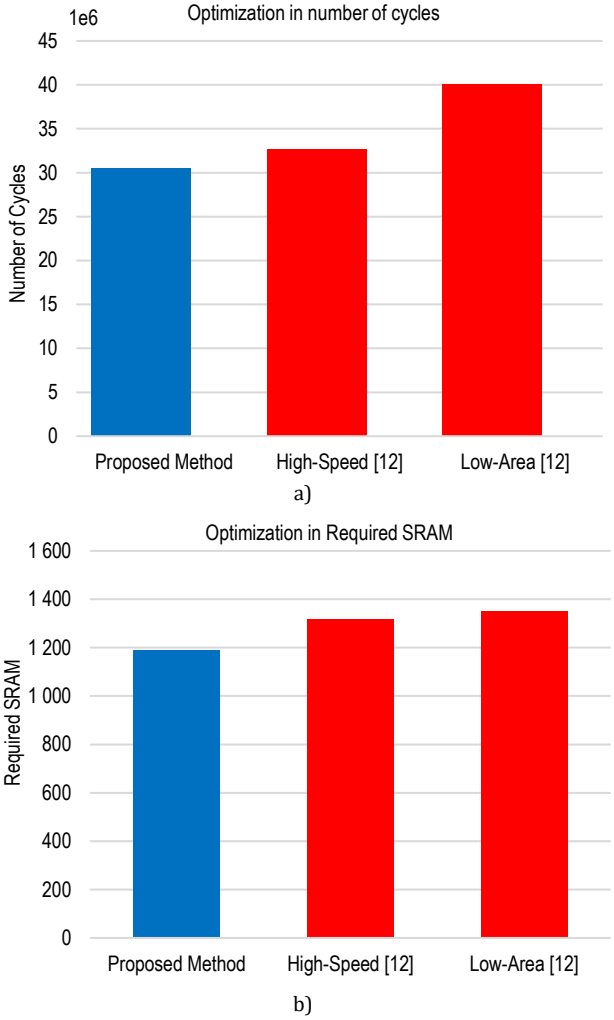

Рис. 3. Сравнение результатов этапа проверки: предложенный метод (a) с предыдущими исследования (b) в сопоставимых условиях

Fig. 3. Comparison of Verification Step Results: Proposed Method (a) vs Prior Research (b) Under Comparable Conditions

При сравнении с High-Speed [12] реализацией (размер окна – 4), предложенный метод снижает количество циклов на 6,43 % и использование SRAM – на 9,87 %. По сравнению с Low-Area реализацией (размер окна 2), предложенный метод достигает сокращения числа циклов на 23,87 % и использования SRAM – на 12,01 %. Эти результаты подчеркивают превосходную производительность и эффективность предложенного метода на этапе проверки электронной подписи при сопоставимых условиях.

Заключение

Данное исследование представляет оптимизированную технику построения электронной подписи ECC, разработанную для ограниченных сред, в частности для беспилотных транспортных систем. Вследствие повышения эффективности скалярного умножения точки за счет свойств циклической группы, противоположных чисел и модифицированного метода оконного умножения, предложенный подход значительно снижает вычислительные затраты. Кроме того, интеграция метода детерминированной генерации однократно используемого числа повышает безопасность при электронной подписи, сохраняя ее эффективность. Экспериментальные результаты на микроконтроллере ATmega2560 R3 демонстрируют значительное сокращение времени выполнения, потребления памяти и количества циклов по сравнению с существующими методами. Эти результаты подтверждают эффективность предложенных оптимизаций и повышение криптографической безопасности для встраиваемых систем с ограниченными ресурсами.

Список источников

1. Johnson D., Menezes A., Vanstone S. The Elliptic Curve Digital Signature Algorithm (ECDSA) // International Journal of Information Security. 2001. Vol. 1. PP. 36–63. DOI:10.1007/s102070100002

2. Josefsson S., Liusvaara I. Edwards-Curve Digital Signature Algorithm (EdDSA). 2017. DOI:10.17487/RFC8032

3. Sabbry N.H., Levina A.B. An Optimized Point Multiplication Strategy in Elliptic Curve Cryptography for Resource-Constrained Devices // Mathematics. 2024. Vol. 12. Iss. 6. P. 881. DOI:10.3390/math12060881. EDN:JZKADF

4. Hisil H., Wong K.K., Carter G., Dawson E. Twisted Edwards Curves Revisited // Proceedings of the 14th International Conference on the Theory and Application of Cryptology and Information Security (ASIACRYPT 2008, Melbourne, Australia, 7–11 December 2008). Lecture Notes in Computer Science. Berlin, Heidelberg: Springer, 2008. Vol. 5350. PP. 326–343. DOI:10.1007/978-3-540-89255-7_20

5. Sabbry N.H., Levina A. Nonce generation techniques in Schnorr multi-signatures: Exploring EdDSA-inspired approaches // AIMS Mathematics. 2024. Vol. 9. Iss. 8. PP. 20304–20325. DOI:10.3934/math.2024988. EDN:YSJYIC

6. Paar C., Pelzl J. Understanding Cryptography: A Textbook for Students and Practitioners. Berlin, Heidelberg: Springer, 2010. DOI:10.1007/978-3-642-04101-3

7. Hankerson D., Menezes A. Elliptic Curve Cryptography // In: Jajodia S., Samarati P., Yung M. (eds.) Encyclopedia of Cryptography, Security and Privacy. Berlin, Heidelberg: Springer, 2021. PP. 1–2. DOI:10.1007/978-3-642-27739-9_245-2

8. Izu T., Möller B., Takagi T. Improved Elliptic Curve Multiplication Methods Resistant Against Side Channel Attacks // Proceedings of the Third International Conference on Cryptology in India (INDOCRYPT 2002, Hyderabad, India, 16–18 December 2002). Lecture Notes in Computer Science. Berlin Heidelberg: Springer, 2002. Vol. 2551. PP. 296–313. DOI:10.1007/3-540-36231-2_24

9. Shenets N.N., Petushkov A.S. New Regular Sliding Window Algorithms for Elliptic Curve Scalar Point Multiplication // Automatic Control and Computer Sciences. 2021. Vol. 55. PP. 1029–1038. DOI:10.3103/S0146411621080289. EDN:FWONTD

10. Cheon J.H., Hong J., Kim M. Speeding Up the Pollard Rho Method on Prime Fields // Proceedings of the 14th International Conference on the Theory and Application of Cryptology and Information Security (ASIACRYPT 2008, Melbourne, Australia, 7–11 December 2008). Berlin, Heidelberg: Springer, 2008. PP. 471–488. DOI:10.1007/978-3-540-89255-7_29

11. Alyas H.H., Abdullah A.A. Enhancement the ChaCha20 Encryption Algorithm Based on Chaotic Maps // In: Kumar R., Mishra B.K., Pattnaik P.K. Next Generation of Internet of Things: Proceedings of ICNGIoT 2021. Lecture Notes in Networks and Systems. Singapore: Springer, 2021. Vol. 201. PP. 91–107. DOI:10.1007/978-981-16-0666-3_10. EDN:ZPPOQO

12. Hutter M., Schwabe P. NaCl on 8-Bit AVR Microcontrollers // Proceedings of the 6th International Conference on Cryptology in Africa «Progress in Cryptology» (AFRICACRYPT 2013, Cairo, Egypt, 22–24 June 2013). Lecture Notes in Computer Science. Berlin, Heidelberg: Springer, 2013. Vol. 7918. PP. 156–172. DOI:10.1007/978-3-642-38553-7_9

13. Nouma S.E., Yavuz A.A. Lightweight and Resilient Signatures for Cloud-Assisted Embedded IoT Systems // arXiv preprint arXiv:2409.13937. 2024. DOI:10.48550/arXiv.2409.13937

References

1. Johnson D., Menezes A., Vanstone S. The Elliptic Curve Digital Signature Algorithm (ECDSA). *International Journal of Information Security*. 2001;1:36–63. DOI:10.1007/s102070100002

2. Josefsson S., Liusvaara I. *Edwards-Curve Digital Signature Algorithm (EdDSA)*. 2017. DOI:10.17487/RFC8032

3. Sabbry N.H., Levina A.B. An Optimized Point Multiplication Strategy in Elliptic Curve Cryptography for Resource-Constrained Devices. *Mathematics*. 2024;12(6):881. DOI:10.3390/math12060881. EDN:JZKADF

4. Hisil H., Wong K.K., Carter G., Dawson E. Twisted Edwards Curves Revisited. *Proceedings of the 14th International Conference on the Theory and Application of Cryptology and Information Security, ASIACRYPT 2008, 7–11 December 2008, Melbourne, Australia. Lecture Notes in Computer Science, vol.5350.* Berlin, Heidelberg: Springer; 2008. p.326–343. DOI:10.1007/978-3-540-89255-7_20

5. Sabbry N.H., Levina A. Nonce generation techniques in Schnorr multi-signatures: Exploring EdDSA-inspired approaches. *AIMS Mathematics*. 2024;9(8):20304–20325. DOI:10.3934/math.2024988. EDN:YSJYIC

6. Paar C., Pelzl J. *Understanding Cryptography: A Textbook for Students and Practitioners*. Berlin, Heidelberg: Springer; 2010. DOI:10.1007/978-3-642-04101-3

7. Hankerson D., Menezes A. Elliptic Curve Cryptography. In: Jajodia S., Samarati P., Yung M. (eds.) *Encyclopedia of Cryptography, Security and Privacy*. Berlin, Heidelberg: Springer; 2021. p.1–2. DOI:10.1007/978-3-642-27739-9_245-2

8. Izu T., Möller B., Takagi T. Improved Elliptic Curve Multiplication Methods Resistant Against Side Channel Attacks. *Proceedings of the Third International Conference on Cryptology in India, INDOCRYPT 2002, 16–18 December 2002, Hyderabad, India. Lecture Notes in Computer Science, vol.2551*. Berlin Heidelberg: Springer; 2002. p.296–313. DOI:10.1007/3-540-36231-2_24

9. Shenets N.N., Petushkov A.S. New Regular Sliding Window Algorithms for Elliptic Curve Scalar Point Multiplication. *Automatic Control and Computer Sciences*. 2021;55:1029–1038. DOI:10.3103/S0146411621080289. EDN:FWONTD

10. Cheon J.H., Hong J., Kim M. Speeding Up the Pollard Rho Method on Prime Fields. *Proceedings of the 14th International Conference on the Theory and Application of Cryptology and Information Security, ASIACRYPT 2008, 7–11 December 2008, Melbourne, Australia.* Berlin, Heidelberg: Springer; 2008. p.471–488. DOI:10.1007/978-3-540-89255-7_29

11. Alyas H.H., Abdullah A.A. Enhancement the ChaCha20 Encryption Algorithm Based on Chaotic Maps. In: Kumar R., Mishra B.K., Pattnaik P.K. *Next Generation of Internet of Things: Proceedings of ICNGIoT 2021. Lecture Notes in Networks and Systems, vol.201*. Singapore: Springer; 2021. p.91–107. DOI:10.1007/978-981-16-0666-3_10. EDN:ZPPOQO

12. Hutter M., Schwabe P. NaCl on 8-Bit AVR Microcontrollers. *Proceedings of the 6th International Conference on Cryptology in Africa «Progress in Cryptology», AFRICACRYPT 2013, 22–24 June 2013, Cairo, Egypt. Lecture Notes in Computer Science, vol.7918*. Berlin, Heidelberg: Springer; 2013. p.156–172. DOI:10.1007/978-3-642-38553-7_9

13. Nouma S.E., Yavuz A.A. Lightweight and Resilient Signatures for Cloud-Assisted Embedded IoT Systems. *arXiv preprint arXiv:2409.13937*. 2024. DOI:10.48550/arXiv.2409.13937

Статья поступила в редакцию 05.03.2025; одобрена после рецензирования 25.03.2025; принята к публикации 07.04.2025.
The article was submitted 05.03.2025; approved after reviewing 25.03.2025; accepted for publication 07.04.2025.

Информация об авторе:

САБРИ НАУРАС ХУССЕЙН
аспирант кафедры факультет информационно-измерительных и биотехнических систем Санкт-Петербургского государственного электротехнического университет «ЛЭТИ» имени В.И. Ульянова (Ленина)
https://orcid.org/0009-0003-2429-5122

Автор сообщает об отсутствии конфликтов интересов.
The author declares no conflicts of interests.