УДК 004.056.53
DOI: 10.26102/2310-6018/2026.55.4.022

Алгоритм функционирования программно-аппаратной подсистемы биометрической идентификации на основе анализа клавиатурного почерка

Е.В. Шкляр✉

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

Резюме. В настоящей работе представлен алгоритм функционирования программно-аппаратной подсистемы биометрической идентификации на основе анализа клавиатурного почерка. Система поддерживает режим идентификации (1:*N*) и режим верификации (1:1) в соответствии с ГОСТ Р 54412–2019. Обзор современной научной литературы по теме исследования показал, что биометрические системы могут использовать различные характеристики, такие как скорость печати или время набора пар клавиш. Выявлено, что аппаратные средства позволяют повысить точность захвата временных интервалов между нажатиями последовательных пар клавиш (биграмм), однако отсутствуют решения, соответствующие ГОСТ Р 54412–2019. Алгоритм обеспечивает полный цикл обработки в распределенной архитектуре, включающей клиент, сервер и аппаратный модуль на базе Arduino. Проведена оценка модели на соответствие требованиям стандарта, доказана устойчивость работы на платформе ATmega32U4. Показана эффективность интеграции в биометрические системы за счет поддержки онлайн и офлайн режимов работы. Время сравнения составляет ≤ 190 мс, а потребление памяти ~1,9 Кб. Описана возможность использования модели в подсистеме обработки сигнала и принятия решений с применением метрик сходства распределений. Результаты исследования могут быть использованы при разработке систем биометрической идентификации, соответствующих ГОСТ, обеспечивающих защиту доступа без модификации клиентских операционных систем.

Ключевые слова: клавиатурный почерк, идентификация, биометрия, математическая модель, биометрический контрольный шаблон.

Для цитирования: Шкляр Е.В. Алгоритм функционирования программно-аппаратной подсистемы биометрической идентификации на основе анализа клавиатурного почерка. *Моделирование, оптимизация и информационные технологии.* 2026;14(4). URL: https://moitvivt.ru/ru/journal/article?id=2255 DOI: 10.26102/2310-6018/2026.55.4.022

Algorithm for the operation of a software-hardware subsystem for biometric identification based on keystroke dynamics analysis

E.V. Shklyar✉

Saint Petersburg Electrotechnical University "LETI" named after V.I. Ulyanov (Lenin), Saint Petersburg, the Russian Federation

Abstract. This paper presents an algorithm for the operation of a software-hardware subsystem for biometric identification based on keystroke dynamics analysis. The system supports identification mode (1:*N*) and verification mode (1:1) in accordance with GOST R 54412–2019. A review of current scientific literature indicates that biometric systems can utilize various features, such as typing speed or digraph entry times. It was found that hardware solutions improve the accuracy of capturing time

intervals between consecutive key presses (digraphs); however, no existing solutions comply with GOST R 54412–2019. The proposed algorithm ensures a complete processing cycle within a distributed architecture comprising a client, a server, and an Arduino-based hardware module. The model was evaluated for compliance with standard requirements, demonstrating robust performance on the ATmega32U4 platform. Integration efficiency into biometric systems is shown through support for both online and offline modes. Comparison time is ≤ 190 ms, with memory consumption of approximately 1.9 KB. The applicability of the model in signal processing and decision-making subsystems using distribution similarity metrics is described. These results can be employed in developing GOST-compliant biometric identification systems that secure access without modifying client operating systems.

Keywords: keystroke dynamics, identification, biometrics, mathematical model, biometric reference.

For citation: Shklyar E.V. Algorithm for the operation of a software-hardware subsystem for biometric identification based on keystroke dynamics analysis. *Modeling, Optimization and Information Technology*. 2026;14(4). (In Russ.). URL: https://moitvivt.ru/ru/journal/article?id=2255 DOI: 10.26102/2310-6018/2026.55.4.022

Введение

Задача идентификации пользователей на основе анализа клавиатурного почерка относится к классу задач биометрической идентификации и аутентификации субъектов доступа и объектов доступа (далее – ИАФ) в соответствии с ГОСТ Р 54412–2019[1]. В исследовании [1] клавиатурный почерк (далее – КП) определен как «индивидуальный способ печати на клавиатуре, уникальный у разных людей». Представление и описание КП зависит от набора биометрических признаков (например, скорости печати или времени набора пар клавиш) и привычки к клавиатуре.

Стационарные персональные компьютеры, как правило, не укомплектованы ни объемными камерами, ни датчиками отпечатков пальца, ни какими-то другими биометрическими датчиками. Для таких систем предлагается использование биометрической идентификации на основе анализа КП. Клавиатура – основной и привычный способ взаимодействия с компьютером для пользователей. На практике программно-аппаратная подсистема анализа КП может применяться для идентификации пользователей и реализации разграничения доступа.

Для использования в перечисленных выше ситуациях, устройство должно подключаться между клавиатурой и компьютером и проводить анализ КП при вводе пароля. При успешной идентификации устройство вводит ожидаемый пароль в систему путем эмуляции клавиатуры. Схема подключения устройства показана на Рисунке 1.

В исследовании [2] рассмотрены устройства для сбора биометрических шаблонов и их влияние на точность захвата временных характеристик. Показано, что точность измерений зависит от типа клавиатуры, частоты опроса клавиш и точности системных таймеров. Вместе с тем, в работе [3] предложены аппаратные методы противодействия анализу КП, направленные на анонимизацию ввода текста. Авторы разработали специальный модуль на базе микроконтроллера Atmel AT89C2051, подключаемый между клавиатурой и компьютером. Устройство выравнивает временные интервалы между событиями нажатия и отпускания клавиш, нивелируя индивидуальные особенности КП. Показано, что использование подобного модуля исключает возможность идентификации пользователя по КП.

[1] ГОСТ Р 54412-2019. Информационные технологии. Биометрия. Общие положения и примеры применения: национальный стандарт Российской Федерации: издание официальное: утвержден и введен в действие Приказом Федерального агентства по техническому регулированию и метрологии от 19 ноября 2019 г. № 1184-ст: взамен ГОСТ Р 54412-2011/ISO/IEC/TR 24741:2007: дата введения 2020-06-01. Москва: Стандартинформ; 2019. 38 с.

Аналогичные выводы сделаны в исследовании [4], где анализ временных параметров проводится на основе сравнения с эталонными гауссовскими сигналами. Подчеркивается, что успешное применение метода требует высокой точности захвата данных, что потенциально предполагает использование внешних аппаратных синхронизаторов или более точных таймеров, чем доступные в стандартной конфигурации операционных систем.

Рисунок 1 – Схема подключения устройства
Figure 1 – Device connection diagram

В работах [5] и [6] зафиксирована зависимость точности анализа КП от используемой платформы. Так, в [5] отмечается различие в эффективности в зависимости от типа клавиатуры, а в [6] наилучшие показатели EER = 0,6% достигнуты при использовании мобильных устройств с сенсорным вводом. Таким образом, аппаратные средства, применяемые для повышения точности идентификации и для защиты от анализа поведенческих характеристик становятся важной составляющей исследований в области КП. В работах [7] и [8] авторами проведена оценка результативности систем идентификации на мобильны устройствах с использованием дополнительных датчиков, включая акселерометр и гироскоп.

Несмотря на наличие ряда исследований возможностей аппаратного анализа КП, в настоящее время в открытых источниках нет информации о существовании отдельного аппаратного решения для биометрической идентификации на основе анализа КП, соответствующего ГОСТ 54412–2019. Для подтверждения возможности практической реализации результатов, изложенных в предшествующих исследованиях [1] и [9], был разработан алгоритм программно-аппаратной подсистемы биометрической идентификации пользователей информационной системы и его выполнена его реализация на основе Arduino.

Существующие аппаратные решения для анализа КП можно разделить на три класса: программные реализации на одноплатных компьютерах (Raspberry Pi, ESP32), промышленные биометрические терминалы и встраиваемые микроконтроллерные модули. Решения на базе Raspberry Pi обладают высокой вычислительной мощностью, однако требуют установки операционной системы и не обеспечивают одновременную поддержку USB-Host и HID-эмуляции без дополнительных компонентов. Промышленные биометрические терминалы ориентированы на облачную инфраструктуру и не предусматривают автономного офлайн-режима. Ни одно из рассмотренных решений не удовлетворяет требованиям ГОСТ Р 54412–2019 в части автономной работы и архитектуры БСОВ. Сравнение предлагаемого решения с аналогами приведено в Таблице 1.

Таблица 1 – Сравнение предлагаемого решения с аналогичными аппаратными реализациями
Table 1 – Comparison of the proposed solution with similar hardware implementations

Характеристика	Raspberry Pi	Промышленные терминалы	Предлагаемое решение (ATmega32U4)
Требует ОС	Да	Да	Нет
USB-Host + HID одновременно	Нет (без доп. модулей)	Нет	Да
Автономный офлайн-режим	Частично	Нет	Да
Соответствие ГОСТ Р 54412–2019	Нет данных	Нет данных	Да
Стоимость компонентов	~ $35	> $200	~$15
Объём памяти под БКШ	Не ограничен	Не ограничен	1,9 КБ (EEPROM)

Научная новизна заключается в том, что в результате выполнения данного исследования построен новый алгоритм функционирования программно-аппаратной биометрической системы идентификации на основе анализа КП в локальном и онлайн-режимах.

Практическая значимость заключается в возможности создания недорогого программируемого устройства на базе Arduino (или схожих платформ), объединяющего методы захвата, предобработки и идентификации на основе анализа КП вне зависимости от используемой ОС и типа клавиатуры, соответствующего требованиям ГОСТ.

Материалы и методы

Непосредственно КП представлен в работе [1] в виде совокупности функций плотности распределения времени набора всех уникальных биграмм исходного текста:

$$\mathcal{F} = \left\{ \widehat{f_{ij}}(t) \middle| (c_i, c_j) \in P \right\}. \quad (1)$$

Структурно-функциональная схема программно-аппаратной подсистемы биометрической идентификации пользователей представлена в распределенной архитектуре, состоящей из клиента для ввода текста, сервера для обработки данных биометрического контрольного шаблона клавиатурного почерка (далее – БКШ КП) и аппаратного модуля, взаимодействующего с клавиатурой (Рисунок 2).

Такое разделение подсистем соответствует структуре биометрической системы общего вида (БСОВ) по ГОСТ Р 54412–2019 и позволяет обеспечить баланс между производительностью, безопасностью и масштабируемостью.

Каждая подсистема реализует одну из подсистем БСОВ, модели и методы функционирования которых описаны в предшествующих работах [1, 9]:

– подсистема сбора данных реализует формирование биометрического предъявления на основе 30 слов, сгенерированных по алгоритму из [9] с контролируемым распределением биграмм,

– подсистема обработки сигнала извлекает временные характеристики и формирует БКШ КП в виде совокупности KDE-оценок плотности распределения времени набора биграмм, согласно математической модели из [1],

– подсистема сравнения и принятия решений вычисляет сходство между текущим и эталонным БКШ КП на основе комбинации четырех метрик сходств.

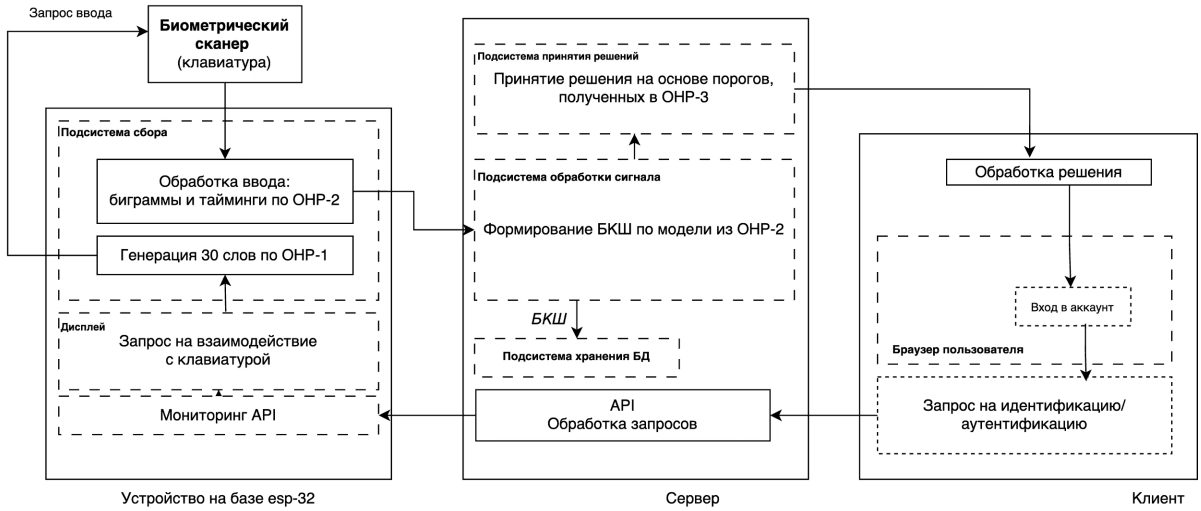

Рисунок 2 – Структурно-функциональная схема программно-аппаратного комплекса
Figure 2 – Structural and functional diagram of the software-hardware complex

Разработанный на основе представленной структурно-функциональной схемы алгоритм (Рисунок 3) обеспечивает полный цикл обработки КП: от сбора «сырых» данных событий ввода на клавиатуре до принятия решения о допуске или недопуске пользователя в информационную систему. Такой подход соответствует как принятому в существующих открытых исследованиях КП [10], так и требованиям ГОСТ.

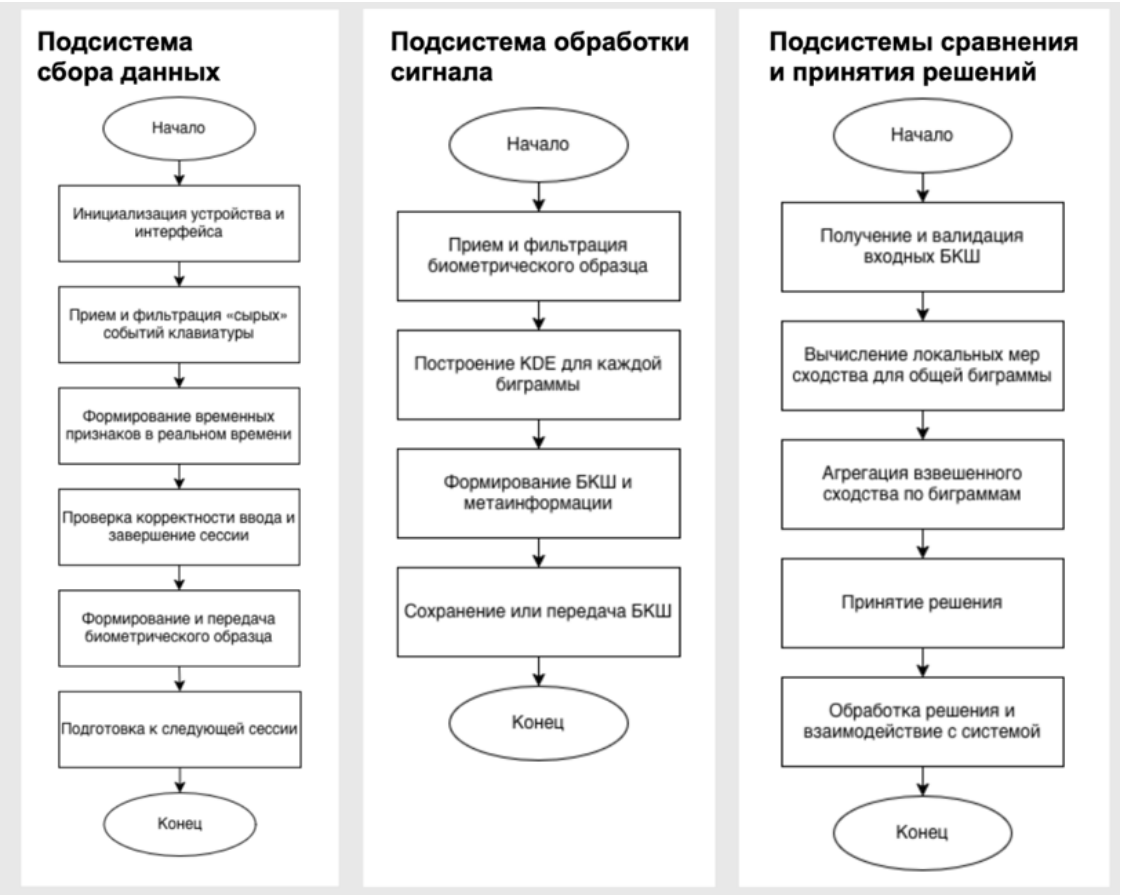

Рисунок 3 – Алгоритмы подсистем программно-аппаратного комплекса
Figure 3 – Algorithms for the subsystems of a hardware-software complex

Реализация алгоритма возможна на платформах Arduino или ESP32, однако для работы в режиме USB Host (прием сигналов с клавиатуры) и одновременной эмуляции HID-устройства для передачи пароля требуется встроенная поддержка обоих протоколов. Из совместимых с USB Host Shield плат этим критериям удовлетворяет только Arduino Leonardo на базе ATmega32U4. Для хранения биометрических шаблонов в офлайн-режиме память микроконтроллера расширена внешним модулем EEPROM AT24C256 (32 Кб). Таким образом, аппаратная часть системы состоит из трех компонентов: плата Arduino Leonardo (ATmega32U4), плата расширения USB Host Shield и модуль энергонезависимой памяти EEPROM AT24C256.

Результаты и обсуждение

В результате исследования получены данные о функционировании подсистемы принятия решений, включающие информацию о точности вычислений, времени сравнения и требуемом объеме памяти. Для обеспечения воспроизводимости эксперимента в разделе приведено детальное описание разработанных алгоритмов (Рисунок 3) функционирования каждой подсистемы БСОВ, включающее формализованные шаги, ограничения реализации и технические особенности аппаратной части.

Алгоритм функционирования подсистемы сбора биометрических данных. Подсистема сбора биометрических данных предназначена для захвата событий взаимодействия пользователя с клавиатурой, предварительной фильтрации и преобразования «сырых» сигналов в структурированный биометрический образец, пригодный для дальнейшей обработки в подсистеме формирования БКШ.

Изначально устройство подключается одновременно и как USB-HID-устройство, и как USB-Host. После подачи питания инициализируется внутренний 16-битный таймер ATmega32U4 с разрешением ≥ 1 мс, настраивается UART/I²C для дисплея (LCD1602C с I²C-конвертером), на дисплей выводится приглашение к вводу. После этого устройство переходит в режим ожидания данных от клавиатуры.

При каждом событии клавиатуры (через USB Host Shield) устройство получает 8-битный скан-код клавиши, тип события (нажатие или отпускание клавиши) и $timestamp$ – абсолютное время события в мс. Дополнительно предусмотрена двухуровневая фильтрация дребезга. Так, если интервал между двумя нажатиями одной и той же клавиши < 10 мс, то второе событие игнорируется.

Из-за ограничения объема энергозависимой памяти устройство не хранит все события, а вычисляет признаки «на лету» – согласно модели, разработанной в [1].

Это означает, что для каждой пары последовательных нажатий:

– фиксируются текущее событие: (k_i, t_i) и предыдущее событие: $(k_{i-1}), (t_{i-1})$,

– формируется биграмма $b = (char_{i-1}, char_i)$,

– рассчитывается интервал $\Delta t = t_i - t_{i-1}$ (в мс) – время для биграммы b,

– сохраняется пара (b, Δt) в буфер событий (RAM, не EEPROM).

Ввод считается завершенным, когда пользователь ввел ровно 30 слов (разделенных пробелами или Enter). При корректном завершении буфер событий очищается от биграмм с $\Delta t > 1000$ мс (межсловные паузы); биграмм, не входящих в кириллический алфавит. В результате в буфере остаются только валидные пары (b, Δt) для биграмм русского текста.

Итоговый БКШ КП представляет собой структуру данных, соответствующую модели из [1]. Из-за ограничения памяти образец не сохраняется на устройстве, а передается или на внешний хост (ПК/сервер) через эмуляцию HID-клавиатуры, или временно буферизуется в RAM и используется для сравнения с расшифрованным БКШ.

Выполнение перечисленных шагов обеспечивает достаточную производительность и совместимость с описанной математической моделью КП. Обработка одного события занимает ≤ 0,2 мс (на 16 МГц ATmega32U4), что позволяет обрабатывать ввод со скоростью до 20 нажатий/сек без потерь.

Алгоритм функционирования подсистемы обработки сигнала. Подсистема обработки сигнала предназначена для преобразования биометрического образца в БКШ КП. Функционирование подсистемы обработки сигнала осуществляется на удаленном сервере или локально. Далее в разделе описаны оба пути, но акцент сделан на онлайн-реализацию.

На вход подсистеме обработки сигнала поступает JSON-структура или бинарный пакет, содержащий информацию о пользователе, идентификатор сессии и данные о событиях клавиатуры в виде массива массивов времени набора биграмм.

Далее для каждой биграммы строится ядерная оценка плотности распределения времени ее набора. Для этого определяется ширина окна сглаживания h по правилу Сильвермана для нормального распределения:

$$h = 1{,}06 \cdot \hat{\sigma} \cdot n^{-\frac{1}{5}}, \quad (2)$$

где $\hat{\sigma}$ – выборочное стандартное отклонение.

Далее вычисляется сетка значений плотности распределения вероятности:

– интервал $[min(\Delta t_b) - 2h, max(\Delta t_b) + 2h]$,

– число точек $N = 32$ (для обеспечения объема порядке 2,1 Кб на 15 биграмм),

– шаг сетки $\Delta x = \frac{x_{max} - x_{min}}{N-1}$.

После чего для каждой точки сетки x_j ($j = 0..1$) вычисляется $p_b(x_j)$:

$$p_b(x_j) = \frac{1}{nh} \sum_{i=1}^{n} \mathcal{K}\left(\frac{x_j - \Delta t_i}{h}\right), \quad (3)$$

где $\mathcal{K}$ – ядро (например, Гауссово). В завершение шага формируется нормированный вектор плотности p_b.

Итоговый БКШ сохраняется в виде JSON-структуры, включающей идентификатор пользователя, метку времени сессии, KDE-векторы по каждой биграмме и нормированные веса биграмм. Формат обеспечивает прямую совместимость с подсистемой сравнения.

В онлайн-режиме БКШ сохраняется в базе данных на сервер. Хеш БКШ (SHA-256) добавляется в учетную запись пользователя. В офлайн-режиме БКШ не сохраняется и используется только для однократного сравнения с расшифрованным эталоном.

Реализация подсистемы обработки сигнала в предложенном виде позволяет выполнить требования по производительности, аппаратной совместимости и безопасности. Так, построение KDE для 15 биграмм по 20 точкам ($n \approx 5 \ldots 30$) занимает ≤ 28 мс на CPU 2.4 ГГц (при использовании Python и NumPy); на Arduino Leonardo – ≤ 180 мс (при использовании C++). Для обеспечения аппаратной совместимости с Arduino Leonardo ядро $\mathcal{K}(u)$ аппроксимируется кусочно-линейной функцией для ускорения вычислений без потери точности. Для обеспечения безопасной передачи и хранения в БКШ не включаются исходные Δt – только $p_b(x_j)$. Это минимизирует возможность восстановления биометрического образца по утечке БКШ.

Алгоритм функционирования подсистем сравнения и принятия решений. В подсистемах сравнения и принятия решений выполняется сравнение текущего БКШ КП с несколькими эталонами при идентификации. Принятие решения о допуске или недопуске осуществляется на основе порога, оптимизированного по критерию EER.

Принятие решения производится или на сервере с использованием полного набора метрик и оптимизированных весов, или в офлайн-режиме с ограниченным набором операций.

Сравнение двух БКШ КП происходит на основе метрик сходства распределений: расстояния Хеллингера, критерия согласия Колмогорова-Смирнова, дивергенции Кульбака-Лейблера и расстояния Вассерштейна. Для каждой функции сходства задаются веса $\tau_к$. На первом этапе выполняется фильтрация биграмм по числу вхождений:

$$P^* = (c_i, c_j) | (c_i, c_j) \in P^{(1)} \cap P^{(2)}, \left|{T_{ij}}^{(1)}\right| \geq m, \left|{T_{ij}}^{(2)}\right| \geq m. \quad (4)$$

На втором этапе вычисляется сходство каждой пары биграмм $(c_I, c_j) \in P^*$ на основе каждой d_k:

$$S_{ij}^{(k)} = d_k\left(\widehat{f_{iJ}}^{(1)}, \widehat{f_{iJ}}^{(2)}\right),\ S_{ij}^{(k)} \in [0,1]. \quad (5)$$

Нормализованные веса определяются как доля веса каждой биграммы (c_i, c_j) относительно суммарного веса всех биграмм, входящих в множество P^* (то есть в набор биграмм, используемый в сравнении).

$$\omega_{ij}^{norm} = \frac{\omega_{ij}}{\sum_{(c_i, c_j) \in P^*} \omega_{ij}}. \quad (6)$$

Определяется итоговое сходство с учетом нормализованных весов и функций, где α_k – коэффициент влияния каждой функции сходства.

$$S_{total} = \sum_{(c_i, c_j) \in P^*} \omega_{ij}^{norm} \sum_k \alpha_k S_{ij}^{(k)}. \quad (7)$$

Таким образом, два биометрических контрольных шаблона считаются принадлежащими одному человеку, если $S_{total} \geq \tau$.

В офлайн-режиме на вход подсистемы принятия решений поступают B_{test} – текущий БКШ, и B_{ref} – эталонный БКШ. Для каждой общей биграммы b из обоих БКШ вычисляются значения сходства на основе перечисленных выше метрик расстояния, затем вычисляется взвешенная локальная оценка на основе вычисленных оптимальных весов [3]:

$$s_b = 0{,}40 \cdot H_b + 0{,}05 \cdot W_b + 0{,}25 \cdot KS_b + 0{,}30 \cdot KL_b. \quad (8)$$

Затем вычисляется общая оценка сходства:

$$S = \sum_{b \in B} w_b \cdot s_b, \quad (9)$$

где веса w_b рассчитываются с помощью логарифмической нормализации частоты биграммы в корпусе. Успешная верификация (режим 1:1) происходит при достижении порогового значения τ:

$$accept \iff S \geq \tau. \quad (10)$$

Согласно ГОСТ Р 54412–2019, верификация (режим 1:1) и идентификация (режим 1:*N*) являются различными процессами БСОВ. В режиме верификации система проверяет, принадлежит ли предъявленный БКШ КП заявленному пользователю (сравнение с одним эталоном). В режиме идентификации система самостоятельно определяет пользователя путём сравнения с N эталонами, выбирая наиболее схожий при условии $S_{i_{max}} \geq \tau$. В режиме идентификации тестируемый БКШ КП сравнивается с N эталонов, хранимых в базе данных:

– вычисляется S_i для всех $i = 1 \ldots N$,

– выбирается $i_{max} = argmax(S_i)$, решение принимается, если $S_{i_{max}} \geq \tau$.

При функционировании в онлайн-режиме сервер отправляет ответ о статусе идентификации через API. При успешной проверке пользователя генерируется JWT-токен или активируется веб-сессия для текущего пользователя.

Результаты экспериментальной отработки приведенных алгоритмов функционирования подсистем приведены в Таблице 2.

Экспериментальная отработка алгоритмов проводилась с участием 256 пользователей без разделения по возрастным или иным группам [1, 9]. Каждый участник однократно вводил набор из 30 слов (~ 213 символов), сгенерированных с заданным распределением биграмм. Сбор данных проводился в неконтролируемых условиях: участники использовали собственные клавиатуры в привычной обстановке; данные с мобильных устройств были исключены из анализа. Оценка эффективности проводилась методом попарных сравнений: «свой–свой» (образцы одного пользователя) и «свой–чужой» (образцы легитимного пользователя против образцов остальных 255 участников). Оптимизация параметров (пороговое значение τ, минимальное число вхождений биграммы m, весовые коэффициенты метрик α) выполнялась методом Grid Search на кросс-валидационной выборке.

Таблица 2 – Данные о функционировании подсистемы принятия решений
Table 2 – Data on the operation of the decision-making subsystem

Параметр	Онлайн (Python)	Офлайн (C++, Arduino)
Точность вычислений	float64 (NumPy)	float32 (ограничение ATmega32U4)
Время сравнения, мс	≤ 28 (на 15 биграмм)	≤ 190 (реализация без log, sqrt в цикле)
Память под БКШ, Кб	2,1 (RAM)	1,9 (EEPROM: 15×32×4 = 1920 байт)
Метрики расстояний	H, KS, W, KL	Только H и KS (более стабильны при float32) с весами 0,6:0,4
Защита от атак	Аудит логов, rate-limiting	Ограничение количества попыток

Заключение

В представленной работе разработана и реализована программно-аппаратная подсистема биометрической идентификации по КП, соответствующая структуре БСОВ ГОСТ Р 54412–2019. Предложенная архитектура поддерживает два режима работы:

1. Онлайн: серверная обработка с точностью EER = 1,49 % (4 метрики);
2. Офлайн: идентификация на микроконтроллере ATmega32U4 за ≤ 190 мс.

Выявлено, что ограниченные ресурсы не препятствуют анализу свободного текста: обработка событий происходит «на лету» (≤ 0,2 мс), а сравнение на основе расстояний Хеллингера и критерия согласия Колмогорова–Смирнова требует всего 1,9 КБ памяти. Схема обеспечивает защиту от имитации КП и обхода устройства благодаря разделению паролей и шифрованию данных. Прототип подтверждает возможность внедрения технологии в реальные информационные системы без модификации клиентских ОС при соблюдении требований к биометрическим характеристикам.

СПИСОК ИСТОЧНИКОВ / REFERENCES

1. Шкляр Е.В., Шульженко А.Д. Математическая модель биометрического контрольного шаблона клавиатурного почерка. *Моделирование, оптимизация и*

информационные технологии. 2026;14(1). https://doi.org/10.26102/2310-6018/2026.52.1.001
Shklyar E.V., Shulzhenko A.D. The mathematical model of keystroke dynamics biometric reference. *Modeling, Optimization and Information Technology.* 2026;14(1). (In Russ.). https://doi.org/10.26102/2310-6018/2026.52.1.001

2. Довгаль В.А. Особенности захвата параметров клавиатурного почерка. *Вестник Адыгейского государственного университета. Серия: Естественно-математические и технические науки.* 2017;(2):102–108.
Dovgal V.A. Features of obtaining information on parameters of keystroke dynamics. *Bulletin of the Adygea State University. Series: Natural-Mathematical and Technical Sciences.* 2017;(2):102–108. (In Russ.).
3. Иванов Д.А., Никитин А.П. Противодействие анализу клавиатурного почерка. *Вестник РГГУ. Серия: Документоведение и архивоведение. Информатика. Защита информации и информационная безопасность.* 2014;(11):178–183.
Ivanov D., Nikitin A. Counteraction against keyboard handwriting analysis. *Bulletin of the RSUH. Series: Documentation and Archival Science. Computer Science. Information Protection and Information Security.* 2014;(11):178–183. (In Russ.).
4. Шарипов Р.Р., Катасев А.С., Кирпичников А.П. Методы анализа клавиатурного почерка пользователей с использованием эталонных гауссовских сигналов. *Вестник Технологического университета.* 2016;19(13):157–160.
5. Ulinskas M., Woźniak M., Damaševičius R. Analysis of Keystroke Dynamics for Fatigue Recognition. In: *Computational Science and Its Applications – ICCSA 2017: Proceedings: Part V: 17th International Conference, 03–06 July 2017, Trieste, Italy*. Cham: Springer; 2017. P. 235–247. https://doi.org/10.1007/978-3-319-62404-4_18
6. Monaco J.V., Tappert Ch.C. The partially observable hidden Markov model and its application to keystroke dynamics. *Pattern Recognition.* 2018;76:449–462. https://doi.org/10.1016/j.patcog.2017.11.021
7. Shadman R., Wahab A.A., Manno M., et al. Keystroke Dynamics: Concepts, Techniques, and Applications. *ACM Computing Surveys.* 2025;57(11). https://doi.org/10.1145/3733103
8. Senerath D., Tharinda S., Vishvajith M., et al. BehaveFormer: A Framework with Spatio-Temporal Dual Attention Transformers for IMU-Enhanced Keystroke Dynamics. In: *2023 IEEE International Joint Conference on Biometrics (IJCB), 25–28 September 2023, Ljubljana, Slovenia.* IEEE; 2023. P. 1–9. https://doi.org/10.1109/IJCB57857.2023.10448997
9. Шкляр Е.В. Алгоритм формирования списка слов с заданным распределением биграмм для регистрации биометрических контрольных шаблонов клавиатурного почерка. *Безопасность информационных технологий.* 2025;32(3):74–89.
Shklyar E.V. An Algorithm for Generating Word Lists with a Specified Bigram Distribution for Keystroke Dynamics Biometric Template Registration. *IT Security (Russia).* 2025;32(3):74–89. (In Russ.).
10. Roy S., Pradhan J., Kumar A., et al. A Systematic Literature Review on Latest Keystroke Dynamics Based Models. *IEEE Access*. 2022;10:92192–92236. https://doi.org/10.1109/ACCESS.2022.3197756

ИНФОРМАЦИЯ ОБ АВТОРЕ / INFORMATION ABOUT THE AUTHOR

Шкляр Евгений Вадимович, старший преподаватель кафедры информационной безопасности, Санкт-Петербургский государственный электротехнический университет «ЛЭТИ» им В.И. Ульянова-Ленина, Санкт-Петербург, Российская Федерация.
e-mail: evgeniy.shklyar@yandex.ru
ORCID: 0000-0002-1894-8065

Evgeniy V. Shklyar, Senior Lecturer, Information Security Department, Saint Petersburg Electrotechnical University "LETI" named after V.I. Ulyanov (Lenin), Saint Petersburg, the Russian Federation.

Статья поступила в редакцию 27.02.2026; одобрена после рецензирования 23.03.2026; принята к публикации 24.04.2026.

The article was submitted 27.02.2026; approved after reviewing 23.03.2026; accepted for publication 24.04.2026.