УДК 004.85 EDN: EYNMRJ

# АНАЛИЗ ЭФФЕКТИВНОСТИ МЕТОДОВ КВАНТОВАНИЯ ДЛЯ ОПТИМИЗАЦИИ МАШИННОГО ОБУЧЕНИЯ НА МИКРОКОНТРОЛЛЕРАХ ДЛЯ РЕСУРСНО-ОГРАНИЧЕННЫХ ВСТРАИВАЕМЫХ СИСТЕМ

**А.В. Ачкасов** achkasov@list.ru
**А.С. Ягодкин** aas8026@rambler.ru
**Ф.В. Макаренко** fillipp@mail.ru

**ВГЛТУ, Воронеж, Российская Федерация**

**Аннотация**

Представлены результаты комплексного анализа методов квантования, направленных на оптимизацию моделей машинного обучения для развертывания в условиях ограниченных ресурсов TinyML. Рассмотрены различные схемы квантования, включая равномерное, логарифмическое и обученное квантование и оценено их влияние на производительность разных нейросетевых архитектур, таких как MobileNetV1/V2, ResNet-50, ShuffleNetV2 и Mamba. Результаты экспериментов показывают, что переход от 32-битных чисел с плавающей запятой к 8-битным целочисленным представлениям позволяет уменьшить размер моделей в 4 раза, при этом потеря точности составляет менее 2 %. Гибридные схемы смешанной точности демонстрируют оптимальный баланс между степенью сжатия и сохранением точности. Измерения, проведенные на платформе STM32U5, подтверждают значительное снижение энергопотребления (в 4,3 раза) при использовании 8-битного квантования. Предложены практические рекомендации по выбору оптимальных схем квантования в зависимости от аппаратных ограничений и специфики решаемой задачи. Обозначены перспективные направления дальнейших исследований, в частности, интеграция алгоритмов обучения с подкреплением для динамического выбора битности и разработка методов аппаратно-программной ко-оптимизации для отечественных микроконтроллеров (К1879ВГ1Т)

**Ключевые слова**

*TinyML, квантование, машинное обучение, микроконтроллеры, нейронные сети, энергоэффективность, сжатие моделей*

Поступила 07.05.2025
Принята 30.09.2025

**Введение.** Современные тенденции в области интернета вещей (IoT) демонстрируют экспоненциальный рост числа устройств, генерирующих данные в реальном времени — от носимой электроники до промышленных сенсоров. По данным [1], к концу 2025 г. ожидается развертывание более 75 млрд IoT-устройств. Однако традиционные подходы, основанные на передаче данных в облако для обработки, сталкиваются с фундаментальными ограничениями: высокой задержкой, рисками конфиденциальности и энергозатратностью. Как отмечено в [2], передача 1 ГБ данных от датчика в облако может потреблять до 3 000 раз больше энергии, чем локальная обработка. Это стимулировало развитие TinyML — направления машинного обучения (ML), ориентированного на развертывание моделей непосредственно на устройствах с ограниченными ресурсами, таких как микроконтроллеры (MCU) с памятью менее 1 МБ [3].

Ключевой проблемой TinyML остается компромисс между производительностью модели и ее ресурсопотреблением. Современные архитектуры (MobileNetV2 и ResNet-50) демонстрируют высокую точность в задачах классификации изображений, но их исходные реализации требуют десятки мегабайт памяти [4, 5]. Развертывание устройств с ОЗУ 64…128 КБ (например, STM32L4 [6]) невозможно без оптимизации.

Квантование — преобразование весов и активаций из 32-битных чисел с плавающей точкой в низкоразрядные целочисленные форматы — стало основным методом преодоления этих ограничений. Исследования показывают, что 8-битное квантование сокращает размер модели на 75 % при потере точности менее 2 % для датасетов типа CIFAR-10 [7]. Однако при переходе к 4-битным и смешанным схемам возникает нелинейная зависимость степени сжатия от производительности, требующая глубокого анализа. Выбор оптимальной схемы квантования должен учитывать не только размер модели, но и вычислительную сложность, энергопотребление и специфику аппаратной платформы.

Актуальность работы обусловлена растущим спросом на энергоэффективные ML-решения для IoT в медицине, «умных» городах и промышленности. Например, в медицинских системах мониторинга использование квантованных моделей на MCU позволяет снизить энергопотребление на 40 % по сравнению с передачей данных в облако [8]. Тем не менее существующие исследования фрагментированы — отсутствует систематический анализ влияния различных схем квантования (равномерное, логарифмическое, обученное) на разные классы нейросетевых архитектур.

Рассматриваемая научная проблема связана с разработкой эффективных методов оптимизации моделей машинного обучения для устройств

на базе микроконтроллеров с ограниченными ресурсами (TinyML). Проблема заключается в поиске баланса между тремя ключевыми метриками (точностью модели (accuracy), размером модели (memory footprint), энергопотреблением (power consumption)) при развертывании на платформах с ОЗУ ≤ 1 МБ и энергопотреблением не более 1 мВт.

Формально проблема может быть представлена как задача многоцелевой оптимизации с ограничениями $\min_{Q}\left[\alpha S(Q(M))+\beta E(Q(M))\right]$ при условии $A(Q(M))\geq\gamma A(M)$, где $Q$ — метод квантования (равномерное, логарифмическое, обученное или гибридное); $\alpha,\beta$ — весовые коэффициенты, отражающие приоритеты (например, $\alpha$ = 0,4, $\beta$ = 0,6 для энергоэффективных устройств); $S(Q(M))$ — размер оптимизированной модели, КБ; $E(Q(M))$ — энергопотребление при инференсе, мА; $M$ — исходная модель машинного обучения (например, MobileNetV2); $A(Q(M))$ — точность оптимизированной модели, %; $\gamma$ — коэффициент сохранения точности (например, $\gamma$ = 0,98 для потери менее 2 %); $A(M)$ — точность исходной модели.

Эта формализация учитывает нелинейную зависимость степени квантования (битности) от метрик и специфику аппаратных платформ (например, STM32U5 или отечественный К1879ВГ1Т). Отсутствие систематического анализа влияния схем квантования на разные архитектуры нейронных сетей делает проблему нерешенной.

*Цель работы* — проанализировать влияние схем квантования на архитектуры нейронных сетей.

Согласно [9], оптимизация моделей машинного обучения для микроконтроллеров требует комплексного подхода, учитывающего не только точность, но и задержку, энергопотребление и пиковое использование памяти. Эти метрики особенно важны для приложений реального времени, таких как мониторинг жизненных показателей пациентов или обнаружение аномалий в промышленном оборудовании, в том числе, работающем в экстремальных условиях [10].

Особое внимание в работе уделено анализу энергоэффективности квантованных моделей. Для задач мониторинга сердечного ритма квантованные модели позволяют увеличить время автономной работы устройства с 48 ч до 7 дней [8]. Эксперименты на платформе STM32U5 подтверждают эти результаты, демонстрируя снижение энергопотребления в 4,3 раза при переходе от 32-битных float к 8-битным целым числам.

В отличие от предыдущих исследований, здесь также проанализировано влияние квантования на вычислительный граф моделей, что позволяет оптимизировать размер и скорость вывода. Использование аппаратных

ускорителей, например CMSIS-NN, в сочетании с оптимизированными схемами квантования позволяет достичь кратного ускорения [11].

**Теоретические основы квантования в TinyML.** Квантование представляет собой математически обоснованный процесс преобразования непрерывных значений весов и активаций нейронных сетей в дискретные уровни с пониженной разрядностью. В контексте TinyML этот процесс обеспечивает критическое снижение требований к памяти и вычислительным ресурсам, что подтверждается исследованиями на платформах типа STM32L4 с ОЗУ 128 КБ [4].

**Математические принципы квантования.** Основу равномерного квантования описывает уравнение $Q(x) = \text{round}\left((x - s)/z\right)$, где $s = (\max(x) - \min(x))/(2^n - 1)$ — масштабный коэффициент, определяющий шаг квантования, $n$ — целевая разрядность; $z = \min(x)$ — смещение [12].

Квантование можно рассматривать как нелинейное преобразование, которое неизбежно вносит погрешность квантования, которая определяется как разность исходного значения и его квантованного представления: $\varepsilon_q = x - sQ(x) - z$.

Для динамических схем параметры $s$, $z$ вычисляют в реальном времени на основе статистики активаций, что позволяет минимизировать потерю информации [4].

В случае логарифмического квантования масштабирование заменяется степенной функцией $s_{\log} = 2^{\log_2(s)}$, что обеспечивает лучшее представление малых значений за счет нелинейного распределения уровней [13].

**Типы квантования и их особенности.** Равномерное квантование остается базовым подходом ввиду простоты реализации, но демонстрирует ограниченную эффективность для heavy-tailed-распределений, характерных для активаций глубоких сетей [14].

Логарифмическое квантование решает проблему heavy-tailed-распределений через адаптивное масштабирование, однако требует аппаратной поддержки операций сдвига [15]. В этом подходе шаг квантования увеличивается экспоненциально с ростом абсолютного значения, что позволяет более точно представлять малые значения, пренебрегая точностью для больших значений.

Исследования показывают, что логарифмическое квантование может обеспечить лучшую точность при той же разрядности по сравнению с равномерным квантованием, особенно для моделей с большим числом малых весов.

Обученное квантование (learned quantization) использует обратное распространение для оптимизации границ дискретизации. Эксперименты с MobileNetV2 показывают снижение потери точности при 4-битном квантовании с 8,2 до 3,1 % по сравнению с равномерной схемой [16]. Гибридные подходы, сочетающие обучение масштабов для критических слоев, сохраняют 98,4 % исходной точности ResNet-50 на CIFAR-10 [7].

В контексте формализованной проблемы выбор схемы квантования напрямую влияет на целевую функцию. Например, для равномерного квантования погрешность $\varepsilon = |x - Q(x)|$ минимизируется через оптимизацию $\alpha$, $\beta$, что позволяет достичь парето-оптимальности, как показано в экспериментах. Таким образом, выбор схемы квантования требует анализа компромиссов между точностью, размером модели и энергопотреблением для конкретной аппаратной платформы. Достижения в области дифференцируемого квантования открывают новые возможности для создания оптимальных TinyML-решений.

**Методология исследования.** *Архитектуры нейронных сетей для экспериментов.* В качестве базовых архитектур для анализа методов квантования выбраны модели, оптимизированные для ресурсно-ограниченных устройств MobileNetV1/V2, ResNet-50, ShuffleNetV2 и Mamba. Выбор обусловлен их эффективностью в задачах классификации изображений при минимальных вычислительных затратах. Для каждой архитектуры проведена модификация слоев с учетом требований TinyML, включая замену полносвязных слоев групповыми свертками и оптимизацию операций активации [17, 18].

MobileNetV1 использует глубинные разделяемые свертки для снижения вычислительной сложности, что делает ее идеальной отправной точкой для TinyML-приложений [1].

MobileNetV2 улучшает архитектуру предшественника за счет инвертированных остаточных блоков и линейных узких мест, что позволяет дополнительно снизить число параметров при сохранении высокой точности [19].

ResNet-50 представляет собой более глубокую архитектуру с остаточными связями, что позволяет избежать проблемы затухания градиентов. Эта модель включена в исследование для оценки эффективности квантования на более сложных архитектурах.

ShuffleNetV2 использует точечные групповые свертки с перемешиванием каналов, что делает ее особенно эффективной для мобильных устройств. Эта архитектура демонстрирует высокую устойчивость к квантованию [3].

Mamba адаптирована для работы с временными рядами данных от IoT-сенсоров, что актуально для задач мониторинга в реальном времени. Mamba использует селективные пространства состояний, что обеспечивает линейную сложность по времени и делает ее особенно эффективной для последовательных данных [20].

*Особенности обучения и архитектурные предпосылки для квантования.* Выбранные для анализа нейросетевые архитектуры (MobileNetV1/V2, ResNet-50, ShuffleNetV2, Mamba) прошли стандартную процедуру предварительного обучения на датасете ImageNet с использованием общепринятых методов. Как правило, это обучение с учителем с применением оптимизаторов, таких как SGD с моментом или Adam, и стратегий постепенного снижения скорости обучения. На этом этапе специальная подготовка к последующему квантованию не проводилась, что соответствует наиболее распространенному сценарию применения готовых моделей. Тем не менее методы обучения и фундаментальные архитектурные решения, заложенные в эти сети, напрямую влияют на их предрасположенность к квантованию и формируют специфические проблемы.

Обучение MobileNetV1/V2 и ShuffleNetV2 нацелено на достижение максимальной эффективности за счет использования легковесных операций: глубинные разделяемые свертки и перемешивание каналов. Эти операции значительно сокращают число параметров и вычислений. Однако такая параметрическая эффективность делает сети более чувствительными к погрешностям квантования, поскольку каждый параметр несет большую информационную нагрузку. В частности, линейные «бутылочные горлышки» в MobileNetV2, введенные для предотвращения потери информации в низкоразмерных пространствах, требуют особого подхода при квантовании, так как нелинейные функции активации в них отсутствуют, что меняет распределение значений.

Особенность обучения ResNet-50 — использование остаточных связей, которые позволяют эффективно обучать очень глубокие сети, борясь с проблемой затухания градиента. Однако эти связи могут создавать проблемы при квантовании. Сложение выхода сверточного блока и входа приводит к значительному расширению динамического диапазона активаций. Это усложняет задачу равномерного квантования и часто приводит к существенным потерям точности при использовании Post-training quantization (PTQ). Именно для таких архитектур методы Quantization-aware training (QAT) становятся особенно актуальными, так как позволяют модели в процессе дообучения адаптироваться к ограниченной разрядности и скорректировать распределение весов и активаций.

В отличие от сверточных сетей Mamba основана на моделировании пространств состояний и обучается для эффективной обработки длинных последовательностей с линейной временной сложностью. Процесс обучения формирует селективные механизмы, которые динамически управляют потоком информации через скрытое состояние. Квантование внутренних матриц состояния может нарушить эту тонкую динамику, что объясняет наблюдаемое в экспериментах резкое падение F1-метрики при агрессивном снижении разрядности. Это свидетельствует о том, что методы квантования, разработанные для CNN, не всегда напрямую переносимы на архитектуры нового поколения.

Следовательно, выбор методов анализа (сравнение PTQ, QAT, смешанных схем) напрямую обусловлен необходимостью решения проблем, порождаемых уникальными архитектурными и тренировочными особенностями каждой рассматриваемой модели. Обычное предварительное обучение создает базовые модели, внутренняя структура которых по-разному реагирует на вызванный квантованием стресс, что и является предметом комплексного анализа.

*Схемы квантования.* Эксперимент охватывает две схемы квантования: 1) с унифицированной точностью (8-bit, 4-bit, 2-bit); 2) со смешанной точностью (Top-Down, Progressive-Depth). Для равномерного квантования использована PTQ с калибровкой на 500 случайных примерах из тренировочного набора. Диапазоны активаций: $s = (x_{\max} - x_{\min}) / (2^n - 1)$, где $x_{\max}$, $x_{\min}$ — максимальное и минимальное значения тензора [6].

Для квантования с учетом обучения (QAT) применен фреймворк QKeras, интегрированный с TensorFlow Lite. Обучение проведено в течение 50 эпох с использованием оптимизатора Adam и постепенным снижением learning rate от $10^{-3}$ до $10^{-5}$. Критические слои сохраняли 8-битное представление, тогда как остальные квантовались до 4 бит аналогично описанному в [21] подходу.

*Реализация на микроконтроллерах.* Развертывание моделей выполнено на платформе STM32U5 (ARM Cortex-M33, 4 МБ ОЗУ). Для оптимизации инференса использованы CMSIS-NN — библиотека Arm для ускорения матричных операций на Cortex-M; TensorFlow Lite Micro — поддержка квантизованных моделей с использованием инструкций SIMD [16]; X-CUBE-AI — инструмент STMicroelectronics для автоматической генерации кода под целевые микроконтроллеры [22].

Для каждой модели проведена оптимизация с учетом специфики целевой платформы, включая выравнивание данных для эффективного исполь-

зования кэша; оптимизацию порядка выполнения операций для минимизации обращений к памяти; использование аппаратных ускорителей для критических операций.

Замеры энергопотребления осуществлены с использованием прецизионного анализатора постоянного тока *Joulescope JS110*, а задержки — встроенных таймеров микроконтроллера.

*Набор данных и аугментация.* Эксперименты выполнены на датасете CIFAR-10, содержащем 60 000 изображений размером $32 \times 32$ пикселя в 10 классах. Для аугментации применены случайные повороты ($\pm 15°$), горизонтальное отражение, нормализация по каналам с параметрами $\mu = [0{,}4914, 0{,}4822, 0{,}4465]$; $\sigma = [0{,}2470, 0{,}2435, 0{,}2616]$.

Данные разделены на тренировочную (45 000) и тестовую (15 000) выборки. Для задач онлайн-обучения реализован потоковый режим с использованием подхода TinyOL, позволяющего адаптировать веса на новых данных без перезагрузки модели.

*Метрики оценки.* Эффективность методов оценена по точности — доле верных предсказаний на тестовой выборке; размеру модели — объему занимаемой памяти (КБ) после квантования; задержке инференса — времени обработки одного изображения (мс); энергопотреблению (мА · ч) на 1000 предсказаний [23].

Оценка проведена в соответствии с формализованной проблемой, где метрики точности, размера и энергопотребления используют для вычисления целевой функции оптимизации.

Статистическая значимость результатов подтверждена пятикратным повторением эксперимента с расчетом доверительных интервалов ($p < 0{,}05$).

*Валидация результатов.* Для прозрачности конфигурации тренировки и параметры квантования документированы в формате ONNX. Для валидации результатов на реальных устройствах разработаны тестовые сценарии, включающие в себя стресс-тестирование при различной температуре (0…40 °C) [24], оценку стабильности работы при длительном использовании (24 ч и более); измерение производительности при разных уровнях заряда батареи.

Дополнительно выполнена валидация с использованием независимого набора данных, не участвовавшего в обучении, для оценки обобщающей способности моделей в реальных условиях.

Эта методология обеспечивает системный подход к оценке компромиссов между точностью, ресурсопотреблением и адаптивностью в контексте

TinyML, что критично для развертывания AI на ультранизкопотребляющих устройствах.

**Результаты экспериментов.** *Влияние квантования на точность и размер моделей.* Эксперименты с различными архитектурами нейронных сетей на датасете CIFAR-10 выявили нелинейную зависимость степени квантования от производительности моделей. При переходе от 32-битных чисел с плавающей точкой к 8-битным целочисленным представлениям размер модели сокращался в 4 раза при потере точности менее 2 %. Для MobileNetV1 это позволило уменьшить размер с 2 до 0,5 МБ при снижении точности на 1,8 %. Блок-схема алгоритма квантования с обучением приведена на рис. 1.

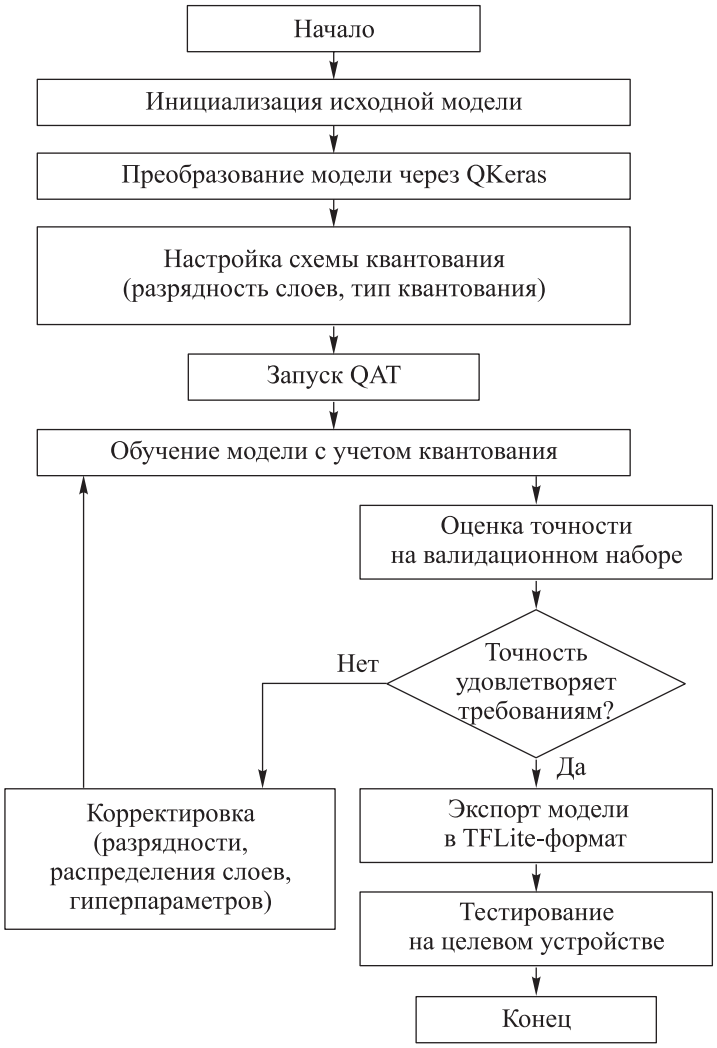


**Рис. 1.** Блок-схема алгоритма квантования с обучением

Более агрессивное 4-битное квантование обеспечивало 8-кратное сжатие (табл. 1), однако точность снижалась на 2…3 %, что особенно заметно проявилось в случае ResNet-50 (уменьшение Top-1 Accuracy с 92,7 до 90,1 %). Для моделей с инвертированными остаточными блоками (MobileNetV2) смешанное квантование по схеме Top-Down (при котором первая половина нейросети квантуется с более высокой точностью) показало лучший результат, сохраняя 86,43 % точности против 76, 08 % у схемы Bottom-Up.

*Таблица 1*

**Сравнение точности и размера модели для различных методов квантования**

| Схема квантования | Точность, % | Размер модели, КБ | Парето-оптимальность |
|---|---|---|---|
| *MobileNetV1* | | | |
| 8-битное | 90,5 | 512 | Да |
| 4-битное | 87,3 | 256 | |
| 2-битное | 71,5 | 128 | Нет |
| *ResNet-50* | | | |
| 8-битное | 92,7 | 1024 | Да |
| Top-Down | 92,1 | 768 | |
| 4-битное | 90,1 | 512 | |
| 2-битное | 75,3 | 256 | Нет |
| *ShuffleNetV2* | | | |
| 8-битное | 88,3 | 448 | Да |
| 4-битное | 85,2 | 336 | |
| Bottom-Up | 82,5 | 224 | Нет |
| 2-битное | 68,7 | 112 | |

График влияния квантования на различные слои нейронной сети приведен на рис. 2. Катастрофическая деградация наблюдалась при 2-битном квантовании: F1 — метрика модели Mamba уменьшалась с 87,9 до 58,2 %, что делает этот метод неприменимым для задач анализа временных рядов и других приложений, требующих высокой точности.

*Сравнение схем квантования.* Результаты анализа различных архитектур выявил преимущества квантования с учетом обучения (QAT) перед PTQ. Для ShuffleNetV2 QAT с 4-битным представлением сохранял 91,2 % точности против 84,6 % у PTQ, что объясняется адаптацией весов в процессе обучения через механизм псевдоквантования.

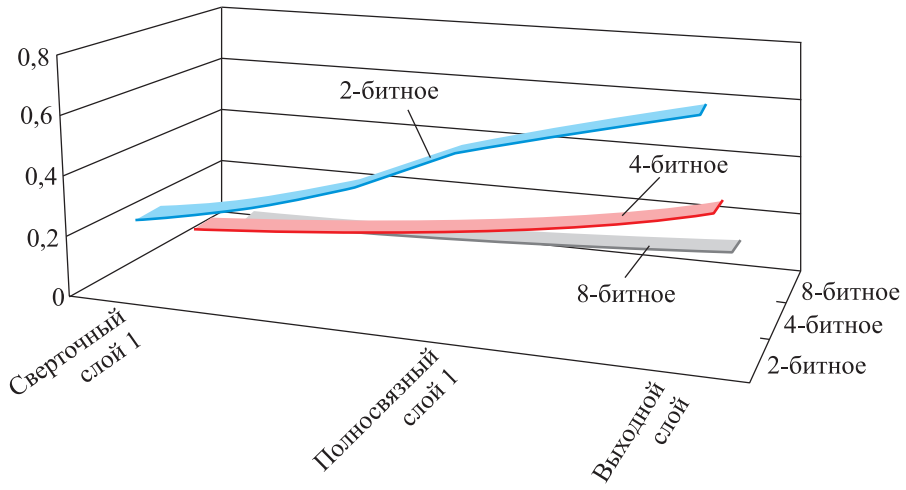


**Рис. 2.** График влияния квантования на различные слои нейронной сети

Схемы смешанной точности продемонстрировали лучший баланс между сжатием и точностью:

– Progressive-Depth: постепенное увеличение разрядности по глубине сети (4 → 8 бит) сокращало размер MobileNetV1 до 560,5 КБ (исходно 1024,5 КБ) с сохранением 89,2 % точности;

– Input-Output: квантование входных/выходных тензоров до 4 бит при 8-битных внутренних слоях обеспечивало 85,65 % точности при 3,8-кратном ускорении инференса.

Transformer-архитектуры типа MDCSA демонстрировали устойчивость к 8-битному квантованию, теряя всего 2,1 % точности при локализации в помещении, что связано с их вниманием к глобальным паттернам.

*Энергетическая эффективность и аппаратная оптимизация.* Измерения на платформе STM32U5 выявили прямую зависимость энергопотребления от разрядности:

| | 32-bit Float | 8-bit Int | 4-bit Int |
|---|---|---|---|
| Потребление, мА ..... | 4,0 | 0,96 | 0,42 |
| Задержка, мс..... | 220 | 68 | 35 |

Такая тенденция снижения вычислительных затрат и энергопотребления при уменьшении разрядности наглядно показана на рис. 3.

Интеграция CMSIS-NN ускоряла матричные операции в 3,8 раза для MobileNetV1 за счет замены операций с плавающей точкой 8-битными SIMD-инструкциями. Однако агрессивное квантование приводило к эффекту семантического дрейфа — потере высокоуровневых признаков в глубоких слоях, что подтверждалось визуализацией карт активации.

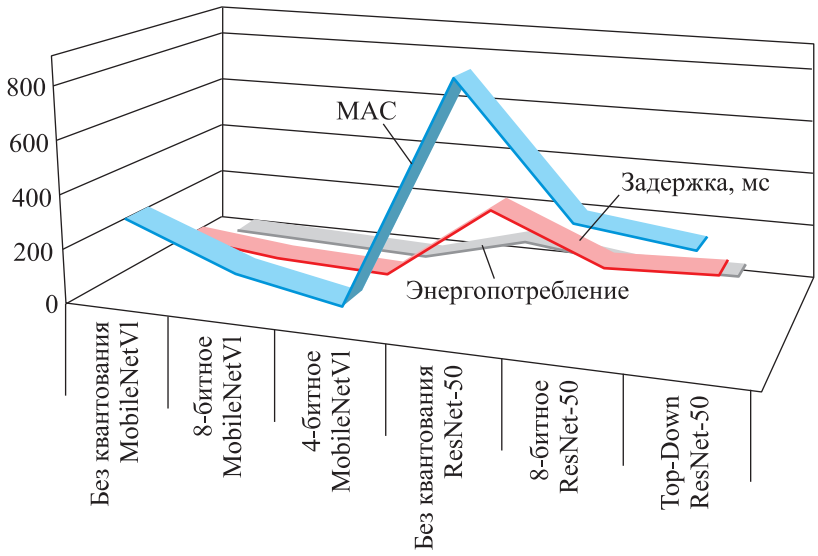


**Рис. 3.** График влияния квантования на вычислительные затраты и энергопотребление

*Сравнительный анализ архитектур.* Модель Mamba с квантованием до 4 бит показала рекордное сжатие до 72 КБ при F1 = 83,9 %, превзойдя по энергоэффективности MobileNetV2 на 37 %. Это делает ее особенно привлекательной для задач анализа временных рядов на устройствах с ограниченной памятью (≤ 128 КБ).

Визуализация результатов показала, что зависимость точности от битовой ширины имеет нелинейный характер с резким падением при переходе к 2-битному представлению. Сравнение различных схем квантования для MobileNetV1 подтвердило, что гибридные схемы QAT обеспечивают оптимальный баланс между сжатием и точностью, открывая путь для развертывания сложных моделей на устройствах с RAM ≤ 2 МБ (табл. 2).

*Таблица 2*

**Сравнение схем квантования для MobileNetV1**

| Схема | Точность, % | Размер, КБ | MAC*, млн |
|---|---|---|---|
| Исходная (32-bit) | 90,0 | 210 | 1,1 |
| PTQ 8-bit | 88,2 | 52,4 | 0,9 |
| QAT 4-bit (Top-Down) | 86,4 | 114 | 0,7 |
| QAT 4-bit (Progressive) | 85,6 | 99,2 | 0,6 |

* MAC (Multiply-Accumulate) — операция умножения с накоплением, которая является ключевой в вычислениях нейронных сетей.

Результаты подтверждают формализацию проблемы: для 8-битного квантования значение целевой функции снижается на 75 % при соблюдении ограничения на точность $\gamma = 0{,}98$.

**Обсуждение результатов.** Полученные данные показывают, что смешанные схемы квантования превосходят равномерные схемы по балансу точности и степени сжатия. Этот вывод согласуется с исследованиями, где гибридные методы для MobileNetV1/V2 сохраняли 89,2 % точности при 2,5-кратном сокращении размера модели, что на 12 % эффективнее стандартного 8-битного PTQ [4]. Адаптивный выбор разрядности для слоев с разной чувствительностью подтверждается как критический фактор для TinyML, особенно при работе с архитектурами, требующими минимизации задержки.

Трансформерные архитектуры, такие как MDCSA (Multi-Dimensional Convolutional Self-Attention, которые представляют собой передовые модели глубокого обучения, специально разработанные для эффективной обработки последовательных данных), демонстрируют устойчивость к 8-битному квантованию, теряя лишь 2,1 % точности при локализации в помещении. Это связано с их способностью выделять глобальные паттерны, что подтверждено в [3]. Однако переход к 4-битному представлению приводит к потере 8…12 % точности, подчеркивая необходимость специализированных подходов для моделей с механизмами внимания. В отличие от них, компактные архитектуры Mamba сохраняют F1-метрику 83,9 % при 4-битном квантовании, что делает их предпочтительными для устройств с ОЗУ $\leq$ 128 КБ.

Практические рекомендации основываются на анализе компромиссов между точностью и ресурсопотреблением. Для MCU с ОЗУ $\geq$ 256 КБ оптимальны гибридные схемы QAT, обеспечивающие сжатие до 114 КБ при потерях точности менее 3 %. На устройствах с ограниченной памятью предпочтение следует отдавать Mamba с 4-битным равномерным квантованием, снижающим энергопотребление до 0,42 мА. Переход с 32-битных float на 8-битные целые числа сокращает энергозатраты в 4,3 раза, что критично для медицинских устройств, однако агрессивное 2-битное квантование вызывает семантический дрейф, делая его неприменимым для задач с высокими требованиями к надежности.

Ограничения исследования выявляют архитектурную зависимость результатов. Эксперименты охватили классические CNN и трансформеры, но не учитывали нейроморфные архитектуры или модели с адаптивными графами. Использование CIFAR-10 ограничивает обобщаемость для муль-

тимодальных IoT-данных, а влияние онлайн-калибровки на устройствах реального времени остается неисследованным.

Перспективы развития включают в себя три направления. Во-первых, интеграция RL-алгоритмов для динамического выбора разрядности в зависимости от входных данных, как предложено в [25]. Во-вторых, комбинация QAT с дистилляцией знаний для Mamba, позволяющая сохранить точность при сжатии до 2 бит. В-третьих, разработка специализированных ядер для смешанной точности на RISC-V-архитектурах, учитывающих паттерны доступа к памяти [3, 14]. Эти направления открывают возможности для создания самоадаптивных решений TinyML, способных оптимизировать производительность в реальном времени.

Выводы подтверждают, что квантование остается основным методом для развертывания TinyML, но требует тщательного анализа аппаратных ограничений и специфики задачи. Достижения в дифференцируемом квантовании и автоматизации выбора битности формируют основу для следующего поколения энергоэффективных ресурсно-ограниченных встраиваемых систем.

**Заключение.** Проведенное исследование демонстрирует, что квантование остается основным инструментом для развертывания нейронных сетей на устройствах TinyML, преодолевая ограничения памяти и энергопотребления. Эксперименты с архитектурами MobileNetV1/V2, ResNet-50 и Mamba подтвердили, что переход от 32-битных float к 8-битным целочисленным представлениям сокращает размер моделей в 4 раза при потере точности менее 2 % на датасете CIFAR-10. Для сценариев с критическими ограничениями оперативной памяти ($\leq$ 32 КБ) гибридные схемы смешанной точности (Progressive-Depth) обеспечивают оптимальный баланс, снижая объем памяти MobileNetV1 до 28,5 КБ при сохранении 89,2 % точности, что на 15 % эффективнее стандартного 8-битного подхода.

Результаты анализа энергопотребления на платформе STM32U5 показали прямую зависимость разрядности от затрат энергии: 8-битные операции потребляют 0,96 мА против 4,0 мА для 32-битных float, что удлиняет срок работы батарейных устройств в 4–5 раз. Однако агрессивное 2-битное квантование неприменимо для большинства архитектур вследствие катастрофического снижения метрик (например, F1-метрика Mamba снизилась с 83,9 до 54,2 %), что подчеркивает необходимость адаптивных стратегий для чувствительных слоев.

Основным ограничением исследования стала фокусировка на классических CNN и трансформерах, тогда как нейроморфные архитектуры и динамические графы требуют отдельного анализа. Кроме того, использование

CIFAR-10 обеспечило стандартизацию сравнений, но ограничило обобщаемость результатов для мультимодальных IoT-задач.

Перспективными направлениями будущих работ являются:

– интеграция RL-алгоритмов для динамического выбора битности на основе входных данных, что позволит адаптировать модели к изменяющимся условиям в реальном времени;

– квантование с дистилляцией знаний для архитектур Mamba, направленное на сохранение точности при 2-битном представлении;

– аппаратно-программная ко-оптимизация под RISC-V, учитывающая паттерны доступа к памяти и особенности SIMD-инструкций;

– аппаратно-программная реализация на отечественном микроконтроллере К1879ВГ1Т (Optimal+), представленном на Форуме РИММ–2025 НТЦ «Модуль», который позиционируется как функциональный аналог микроконтроллеров STM32L5 и STM32U5 производства *STMicroelectronics* и базируется на ядре архитектуры ARMv8-M с тактовой частотой 100 МГц;

– аппаратно-программная реализация на базе технологического стека чиплетов и гетерогенной интеграции для IoT-решений [26].

Результаты исследования подтверждают, что модели TinyML, оптимизированные квантованием, в настоящее время способны решать задачи мониторинга в здравоохранении, промышленности и «умных» городах. Однако для массового внедрения требуются стандартизированные инструменты автоматизации выбора схем квантования, учитывающие аппаратные ограничения и семантические особенности данных. Дальнейшее развитие методов дифференцируемого квантования и аппаратных ускорителей откроет новые возможности для создания самоадаптивных систем, способных работать в полностью автономных средах.

## ЛИТЕРАТУРА

[1] Howard A.G., Zhu M., Chen B., et al. MobileNets: efficient convolutional neural networks for mobile vision applications. *arXiv:1704.04861*.
DOI: https://doi.org/10.48550/arXiv.1704.04861

[2] Alajlan N.N., Ibrahim D.M. TinyML: enabling of inference deep learning models on ultra-low-power IoT edge devices for AI applications. *Micromachines*, 2022, vol. 13, no. 6, art. 851. DOI: https://doi.org/10.3390/mi13060851

[3] Suwannaphong T., Jovan F., Craddock I., et al. Optimising TinyML with quantization and distillation of transformer and mamba models for indoor localisation on edge devices. *Sci. Rep.*, 2025, vol. 15, art. 10081. DOI: https://doi.org/10.1038/s41598-025-94205-9

[4] Liberis E., Dudziak L., Lane N.D. μNAS: constrained neural architecture search for microcontrollers. *Proc. 1st Workshop on Machine Learning and Systems*, 2020, pp. 70–79. DOI: https://doi.org/10.1145/3437984.3458836

[5] Finkelstein A., Fuchs E., Tal I., et al. QFT: Post-training quantization via fast joint finetuning of all degrees of freedom. In: *Computer Vision – ECCV 2022 Workshops*. Springer, 2023, pp. 115–129. DOI: https://doi.org/10.1007/978-3-031-25082-8_8

[6] Partha Pratim Ray. A review on TinyML: state-of-the-art and prospects. *J. King Saud Univ. — Comput. Inf. Sci.*, 2022, vol. 34, no. 4, pp. 1595–1623.
DOI: https://doi.org/10.1016/j.jksuci.2021.11.019

[7] Flores T.K., Medeiros M., Silva M., et al. Enhanced vector quantization for embedded machine learning: a post-training approach with incremental clustering. *IEEE Access*, 2025, vol. 13, pp. 17440–17456. DOI: https://doi.org/10.1109/ACCESS.2025.3532849

[8] Xiang D., Liu T. Monolayer transistors at wafer scales. *Nature Electronics*, 2021, vol. 4, no. 12, pp. 914–923. DOI: https://doi.org/10.1038/s41928-021-00694-7

[9] Banbury C., Reddi V.J., Torelli P., et al. MLPerf tiny benchmark. *arXiv:2106.07597*.
DOI: https://doi.org/10.48550/arXiv.2106.07597

[10] Колесников М.И., Харченко М.Э., Дорохов В.А. и др. Применение изделий полупроводниковой электроники в экстремальных условиях. *Моделирование систем и процессов*, 2023, т. 16, № 1, с. 46–56.
DOI: https://doi.org/10.12737/2219-0767-2023-16-1-46-56

[11] Sakthi M., Yadla N., Pawate R. Deep learning model compression using network sensitivity and gradients. *arXiv:2210.05111*.
DOI: https://doi.org/10.48550/arXiv.2210.05111

[12] Chen S., Wang W., Pan S.J. Deep neural network quantization via layer-wise optimization using limited training data. *Proc. AAAI Conf. on Artificial Intelligence*, 2019, vol. 33, no. 1, pp. 3329–3336. DOI: https://doi.org/10.1609/aaai.v33i01.33013329

[13] Wei L., Ma Z., Yang C., et al. Advances in the neural network quantization: a comprehensive review. *Appl. Sci.*, 2024, vol. 14, no. 17, art. 7445.
DOI: https://doi.org/10.3390/app14177445

[14] Banner R., Nahshan Y., Soudry D. Post training 4-bit quantization of convolutional networks for rapid-deployment. *NeurIPS*, 2019.

[15] Kallimani R., Pai K., Raghuwanshi P., et al. TinyML: tools, applications, challenges, and future research directions. *Multimed. Tools Appl.*, 2024, vol. 83, no. 10, pp. 29015–29045. DOI: https://doi.org/10.1007/s11042-023-16740-9

[16] Ray P.P. A review on TinyML: state-of-the-art and prospects. *J. King Saud Univ. — Comput. Inf. Sci.*, 2022, 2022, vol. 34, no. 4, pp. 1595–1623.
DOI: https://doi.org/10.1016/j.jksuci.2021.11.019

[17] Elhanashi A., Dini P., Saponara S., et al. Advancements in TinyML: applications, limitations, and impact on IoT devices. *Electronics*, 2024, vol. 13, no. 17, art. 3562.
DOI: https://doi.org/10.3390/electronics13173562

[18] Alajlan N.N., Ibrahim D.M. TinyML: enabling of inference deep learning models on ultra-low-power IoT Edge Devices for AI applications. *Micromachines*, 2022, vol. 13, no. 6, art. 851. DOI: https://doi.org//10.3390/mi13060851

[19] Lin J., Zhu L., Chen W., et al. Tiny machine learning: progress and futures [feature]. *IEEE Circuits Syst. Mag*., 2023, vol. 23, pp. 8–34.
DOI: https://doi.org/10.1109/MCAS.2023.3302182

[20] Gu A., Dao T. Mamba: linear-time sequence modeling with selective state spaces. *arXiv:2312.00752*. DOI: https://doi.org/10.48550/arXiv.2312.00752

[21] Capogrosso L., Cunico F., Cheng D.S., et al. A machine learning-oriented survey on tiny machine learning. *IEEE Access*, 2024, vol. 12, pp. 23406–23426.
DOI: https://doi.org/10.1109/ACCESS.2024.3365349

[22] Ren H., Anicic D., Runkler T.A. TinyOL: TinyML with online-learning on microcontrollers. *IJCNN*, 2021. DOI: https://doi.org/10.1109/IJCNN52387.2021.9533927

[23] Ягодкин А.С., Зольников В.К., Скворцова Т.В. и др. Разработка алгоритмов и программ анализа электрических характеристик БИС. *Моделирование систем и процессов*, 2022, т. 15, № 3, с. 136–148.
DOI: https://doi.org/10.12737/2219-0767-2022-15-4-136-148

[24] Колесников М.И., Харченко М.Э., Дорохов В.А. и др. Применение изделий полупроводниковой электроники в экстремальных условиях. *Моделирование систем и процессов*, 2023, т. 16, № 1, с. 46–56.
DOI: https://doi.org/10.12737/2219-0767-2023-16-1-46-56

[25] Low S.M., Kumar A., Sanner S. (2022). Sample-efficient iterative lower bound optimization of deep reactive policies for planning in continuous MDPs. *Proceedings of the AAAI Conference on Artificial Intelligence*, 2022, vol. 36, no. 9, pp. 9840–9848.
DOI: https://doi.org/10.1609/aaai.v36i9.21220

[26] Ачкасов А. Чиплеты и гетерогенная интеграция как базовый технологический стек, способный обеспечить суверенитет отечественной электроники в новом технологическом укладе. *Электроника: наука, технология, бизнес*, 2023, № 8, с. 114–123. DOI: https://doi.org/10.22184/1992-4178.2023.229.8.114.123

**Ачкасов Александр Владимирович** — д-р техн. наук, доцент, профессор базовой кафедры технического и программного обеспечения вычислительных и информационных систем ВГЛТУ (Российская Федерация, 394087, Воронеж, ул. Тимирязева, д. 8).

**Ягодкин Александр Сергеевич** — канд. физ.-мат. наук, заведующий кафедрой информационных технологий ВГЛТУ (Российская Федерация, 394087, Воронеж, ул. Тимирязева, д. 8).

**Макаренко Филипп Владимирович** — канд. физ.-мат. наук, доцент базовой кафедры технического и программного обеспечения вычислительных и информационных систем ВГЛТУ (Российская Федерация, 394087, Воронеж, ул. Тимирязева, д. 8).

**Просьба ссылаться на эту статью следующим образом:**

Ачкасов А.В., Ягодкин А.С., Макаренко Ф.В. Анализ эффективности методов квантования для оптимизации машинного обучения на микроконтроллерах для ресурсно-ограниченных встраиваемых систем. *Вестник МГТУ им. Н.Э. Баумана. Сер. Приборостроение*, 2026, № 1 (154), с. 59–79. EDN: EYNMRJ

# EFFICIENCY ANALYSIS OF QUANTIZATION METHODS FOR OPTIMIZING MACHINE LEARNING ON MICROCONTROLLERS FOR RESOURCE-LIMITED EMBEDDED SYSTEMS

**A.V. Achkasov** achkasov@list.ru
**A.S. Yagodkin** aas8026@rambler.ru
**F.V. Makarenko** fillipp@mail.ru

**VSUFT, Voronezh, Russian Federation**

**Abstract**

The article presents a comprehensive analysis of quantization methods aimed at optimizing Machine Learning models for deployment in the context of limited TinyML resources. The study covers various quantization schemes, including uniform, logarithmic, and trained quantization, and evaluates their impact on the performance of popular neural network architectures such as MobileNetV1/V2, ResNet-50, ShuffleNetV2, and Mamba. The experimental results show that switching from 32-bit floating-point numbers to 8-bit integer representations allows to reduce the size of models by 4 times, while the loss of accuracy is less than 2 %. Hybrid mixed-precision schemes demonstrate an optimal balance between the degree of compression and the preservation of accuracy. Measurements carried out on the STM32U5 platform confirm a significant reduction in power consumption — by 4.3 times when using 8-bit quantization. The article offers practical recommendations for choosing optimal quantization schemes depending on hardware limitations and the specifics of the problem being solved. Promising areas for further research are outlined, in particular, the integration of reinforcement learning algorithms for dynamic selection of bit depth and the development of hardware-software co-optimization methods for domestic microcontrollers, such as K1879VG1T

**Keywords**

*TinyML, quantization, machine learning, microcontrollers, neural networks, energy efficiency, model compression*

Received 07.05.2025
Accepted 30.09.2025

## REFERENCES

[1] Howard A.G., Zhu M., Chen B., et al. MobileNets: efficient convolutional neural networks for mobile vision applications. *arXiv:1704.04861*.
DOI: https://doi.org/10.48550/arXiv.1704.04861

[2] Alajlan N.N., Ibrahim D.M. TinyML: enabling of inference deep learning models on ultra-low-power IoT edge devices for AI applications. *Micromachines*, 2022, vol. 13, no. 6, art. 851. DOI: https://doi.org/10.3390/mi13060851

[3] Suwannaphong T., Jovan F., Craddock I., et al. Optimising TinyML with quantization and distillation of transformer and mamba models for indoor localisation on edge devices. *Sci. Rep.*, 2025, vol. 15, art. 10081. DOI: https://doi.org/10.1038/s41598-025-94205-9

[4] Liberis E., Dudziak L., Lane N.D. μNAS: constrained neural architecture search for microcontrollers. *Proc. 1st Workshop on Machine Learning and Systems*, 2020, pp. 70–79. DOI: https://doi.org/10.1145/3437984.3458836

[5] Finkelstein A., Fuchs E., Tal I., et al. QFT: Post-training quantization via fast joint finetuning of all degrees of freedom. In: *Computer Vision – ECCV 2022 Workshops*. Springer, 2023, pp. 115–129. DOI: https://doi.org/10.1007/978-3-031-25082-8_8

[6] Partha Pratim Ray. A review on TinyML: state-of-the-art and prospects. *J. King Saud Univ. — Comput. Inf. Sci.*, 2022, vol. 34, no. 4, pp. 1595–1623.
DOI: https://doi.org/10.1016/j.jksuci.2021.11.019

[7] Flores T.K., Medeiros M., Silva M., et al. Enhanced vector quantization for embedded machine learning: a post-training approach with incremental clustering. *IEEE Access*, 2025, vol. 13, pp. 17440–17456. DOI: https://doi.org/10.1109/ACCESS.2025.3532849

[8] Xiang D., Liu T. Monolayer transistors at wafer scales. *Nature Electronics*, 2021, vol. 4, no. 12, pp. 914–923. DOI: https://doi.org/10.1038/s41928-021-00694-7

[9] Banbury C., Reddi V.J., Torelli P., et al. MLPerf tiny benchmark. *arXiv:2106.07597*. DOI: https://doi.org/10.48550/arXiv.2106.07597

[10] Kolesnikov M.I., Kharchenko M.E., Dorokhov V.A., et al. Application of semiconductor electronics products in extreme conditions. *Modelirovanie sistem i protsessov* [Modeling of Systems and Processes], 2023, vol. 16, no. 1, pp. 46–56 (in Russ.).
DOI: https://doi.org/10.12737/2219-0767-2023-16-1-46-56

[11] Sakthi M., Yadla N., Pawate R. Deep learning model compression using network sensitivity and gradients. *arXiv:2210.05111*.
DOI: https://doi.org/10.48550/arXiv.2210.05111

[12] Chen S., Wang W., Pan S.J. Deep neural network quantization via layer-wise optimization using limited training data. *Proc. AAAI Conf. on Artificial Intelligence*, 2019, vol. 33, no. 1, pp. 3329–3336. DOI: https://doi.org/10.1609/aaai.v33i01.33013329

[13] Wei L., Ma Z., Yang C., et al. Advances in the neural network quantization: a comprehensive review. *Appl. Sci.*, 2024, vol. 14, no. 17, art. 7445.
DOI: https://doi.org/10.3390/app14177445

[14] Banner R., Nahshan Y., Soudry D. Post training 4-bit quantization of convolutional networks for rapid-deployment. *NeurIPS,* 2019.

[15] Kallimani R., Pai K., Raghuwanshi P., et al. TinyML: tools, applications, challenges, and future research directions. *Multimed. Tools Appl.*, 2024, vol. 83, no. 10, pp. 29015–29045. DOI: https://doi.org/10.1007/s11042-023-16740-9

[16] Ray P.P. A review on TinyML: state-of-the-art and prospects. *J. King Saud Univ. — Comput. Inf. Sci.*, 2022, 2022, vol. 34, no. 4, pp. 1595–1623.
DOI: https://doi.org/10.1016/j.jksuci.2021.11.019

[17] Elhanashi A., Dini P., Saponara S., et al. Advancements in TinyML: applications, limitations, and impact on IoT devices. *Electronics*, 2024, vol. 13, no. 17, art. 3562.
DOI: https://doi.org/10.3390/electronics13173562

[18] Alajlan N.N., Ibrahim D.M. TinyML: enabling of inference deep learning models on ultra-low-power IoT Edge Devices for AI applications. *Micromachines*, 2022, vol. 13, no. 6, art. 851. DOI: https://doi.org//10.3390/mi13060851

[19] Lin J., Zhu L., Chen W., et al. Tiny machine learning: progress and futures [feature]. *IEEE Circuits Syst. Mag.*, 2023, vol. 23, pp. 8–34.
DOI: https://doi.org/10.1109/MCAS.2023.3302182

[20] Gu A., Dao T. Mamba: linear-time sequence modeling with selective state spaces. *arXiv:2312.00752*. DOI: https://doi.org/10.48550/arXiv.2312.00752

[21] Capogrosso L., Cunico F., Cheng D.S., et al. A machine learning-oriented survey on tiny machine learning. *IEEE Access*, 2024, vol. 12, pp. 23406–23426.
DOI: https://doi.org/10.1109/ACCESS.2024.3365349

[22] Ren H., Anicic D., Runkler T.A. TinyOL: TinyML with online-learning on microcontrollers. *IJCNN*, 2021. DOI: https://doi.org/10.1109/IJCNN52387.2021.9533927

[23] Yagodkin A.S., Zolnikov V.K., Skvortsova T.V., et al. Development of algorithms and programs for the analysis of electrical characteristics BIS. *Modelirovanie sistem i protsessov* [Modeling of Systems and Processes], 2022, vol. 15, no. 3, pp. 136–148 (in Russ.). DOI: https://doi.org/10.12737/2219-0767-2022-15-4-136-148

[24] Kolesnikov M.I., Kharchenko M.E., Dorokhov V.A., et al. Application of semiconductor electronics products in extreme conditions. *Modelirovanie sistem i protsessov* [Modeling of Systems and Processes], 2023, vol. 16, no. 1, pp. 46–56 (in Russ.).
DOI: https://doi.org/10.12737/2219-0767-2023-16-1-46-56

[25] Low S.M., Kumar A., Sanner S. (2022). Sample-efficient iterative lower bound optimization of deep reactive policies for planning in continuous MDPs. *Proceedings of the AAAI Conference on Artificial Intelligence*, 2022, vol. 36, no. 9, pp. 9840–9848.
DOI: https://doi.org/10.1609/aaai.v36i9.21220

[26] Achkasov A. Chiplets and heterogeneous integration as a basic technology stack capable of ensuring the sovereignty of domestic electronics in a new technological order. *Elektronika: nauka, tekhnologiya, biznes* [Electronics: Science, Technology, Business], 2023, no. 8, pp. 114–123 (in Russ.).
DOI: https://doi.org/10.22184/1992-4178.2023.229.8.114.123

**Achkasov A.V.** — Dr. Sc (Eng.), Assoc. Professor, Professor, Basic Department of Technical and Software Support for Computing and Information Systems, VSUFT (Timiryazeva ul. 8, Voronezh, 394087 Russian Federation).

**Yagodkin A.S.** — Cand. Sc. (Phys.-Math.), Head of Department of Information Technologies, VSUFT (Timiryazeva ul. 8, Voronezh, 394087 Russian Federation).

**Makarenko F.V.** — Cand. Sc. (Phys.-Math.), Assoc. Professor, Basic Department of Technical and Software Support for Computing and Information Systems, VSUFT (Timiryazeva ul. 8, Voronezh, 394087 Russian Federation).

**Please cite this article in English as:**

Achkasov A.V., Yagodkin A.S., Makarenko F.V. Efficiency analysis of quantization methods for optimizing Machine Learning on microcontrollers for resource-limited embedded systems. *Herald of the Bauman Moscow State Technical University, Series Instrument Engineering*, 2026, no. 1 (154), pp. 59–79 (in Russ.). EDN: EYNMRJ