Радиотехника и связь

DOI 10.36622/1729-6501.2025.21.4.013
УДК 004.032

ОПТИМИЗАЦИЯ ПЛИС-АРХИТЕКТУР ДЛЯ НЕЙРОСЕТЕВОГО ДЕКОДИРОВАНИЯ БЛОКОВЫХ КОДОВ

М.В. Хорошайлова

Воронежский государственный технический университет, г. Воронеж, Россия

Аннотация: исследуются методы оптимизации архитектур программируемых логических интегральных схем (ПЛИС) для эффективной реализации нейросетевых декодеров блоковых кодов, включая коды с низкой плотностью проверок на четность (LDPC), полярные и Боуза — Чоудхури — Хоквингема (БЧХ) коды. Основное внимание уделяется аппаратным модификациям, позволяющим достичь оптимального баланса между точностью декодирования и вычислительной эффективностью. Разработаны и проанализированы специализированные архитектурные решения для ПЛИС, включающие модифицированные схемы таблиц поиска (LUT) с адаптивной битовой глубиной, аппаратные ускорители для операций проверки узлов в режиме реального времени. Разработанные решения особенно актуальны для итеративных алгоритмов декодирования (Min-Sum) применительно к LDPC-кодам, где требуется интенсивная обработка мягких решений. Экспериментальные результаты показывают увеличение пропускной способности декодера на 30 % по сравнению с традиционными реализациями, снижение энергопотребления на 20 % при сохранении корректирующей способности, возможность динамической адаптации параметров декодирования для различных типов блоковых кодов. Предложенные архитектурные решения демонстрируют особую эффективность при обработке длинных кодовых слов (n > 1000), характерных для современных систем связи 5G/6G и систем хранения данных

Ключевые слова: блоковые коды, нейросетевое декодирование, оптимизация ПЛИС, адаптивная архитектура, аппаратное ускорение

Благодарности: работа выполнена при финансовой поддержке Министерства науки и высшего образования Российской Федерации в рамках государственного задания (№ FZGM-2025-0002)

Введение

Современные системы связи 5G/6G и системы хранения данных предъявляют повышенные требования к эффективности декодирования блоковых кодов, таких как LDPC, полярные и БЧХ-коды. Традиционные алгоритмы декодирования сталкиваются с фундаментальными ограничениями при работе с длинными кодовыми словами и в условиях низкого отношения сигнал/шум. Нейросетевые подходы к декодированию демонстрируют перспективные результаты, превосходя по точности классические методы, однако их широкое внедрение сдерживается высокой вычислительной сложностью и значительными требованиями к аппаратным ресурсам [1, 2].

В данной статье представлена оптимизация архитектур программируемых логических интегральных схем (ПЛИС) для эффективной реализации нейросетевых декодеров блоковых кодов. Основное внимание уделено модификациям логических блоков, позволяющим значительно повысить плотность выполнения операций умножения с фиксированной запятой (4-9 бит) и операциям блока умножения-сложения (MAC), реализуемых в программируемая логическая структура (soft fabric), при минимальных затратах площади и задержки.

Ускорение глубокого обучения на ПЛИС

Несмотря на высокую гибкость и реконфигурируемость ПЛИС, реализация специализированных операций декодирования для LDPC, полярных и БЧХ-кодов остается ресурсоемкой задачей. Особые сложности возникают при реализации итеративных алгоритмов проверки узлов для LDPC-кодов, схем последовательного исключения для полярных кодов, алгебраических операций декодирования БЧХ-кодов.

Для эффективного использования ресурсов ПЛИС разрабатываются специализированные версии алгоритмов декодирования, а именно вантованные версии Min-Sum алгоритма для LDPC, аппроксимированные схемы SC-декодирования для полярных кодов, оптимизированные реализации алгоритма Берлекэмпа-Месси для БЧХ.

В качестве базовой платформы для реализации алгоритмов декодирования выбрана ПЛИС Intel Stratix 10, обладающая высокой вычислительной плотностью и оптимизированной архитектурой для выполнения специализированных операций. Ключевым элементом этой ПЛИС является модуль адаптивной логики (ALM), который обеспечивает гибкость при реализации как логических, так и арифметических операций, критически важных для декодирования кодов.

На рис. 1 представлена структура ALM, включающая 6-разрядный LUT, конфигурируемый в виде двух 5-разрядных LUT, 2 бита упрочненной арифметики (два сумматора) с выделенными цепями переноса, 8 входов (A-H) и 4 выхода (O1-O4).

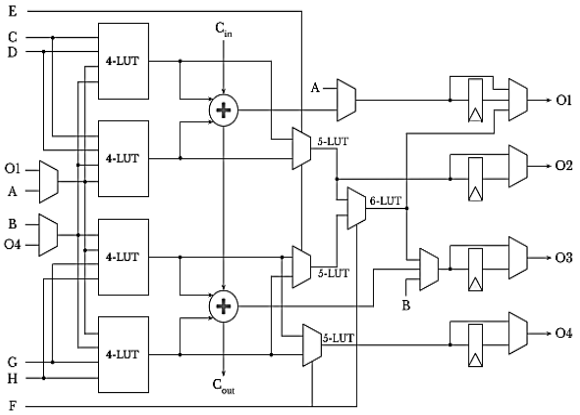


Рис. 1. Упрощенная архитектура ALM Stratix 10

ALM поддерживает два основных режима работы:

- Обычный режим, позволяющий реализовать одну логическую функцию с 6 входами и две функции с 5 входами (или меньше), использующие не более 8 различных входов (например, два независимых 4-разрядных LUT или два 5-разрядных LUT с двумя общими входами).

- Арифметический режим, который оптимизирован для выполнения операций сложения: четыре 4-разрядных LUT подаются на входы двух сумматоров и поддерживается логика предварительного сложения при использовании ≤ 6 входных сигналов.

При реализации умножения в контексте кодов с низкой плотностью проверок на четность (LDPC), полярных и БЧХ-кодов, когда отображение умножается с использованием адаптивных логических модулей (ALM), после начального этапа сложения частичных произведений (первый каскад сложения) дальнейшее сокращение разрядов выполняется исключительно с помощью цепочек сумматоров. В такой конфигурации LUT (таблицы поиска) используются лишь для передачи входных данных на жестко заложенные сумматоры, что приводит к неэффективному расходованию ресурсов ALM. Это особенно критично для декодеров LDPC и полярных кодов, где высокая плотность вычислений требует оптимизации использования кристалла [3].

Для решения этой проблемы предлагается модифицированная архитектура ALM, показанная на рис. 2, в которой добавлен второй уровень цепочки переноса.

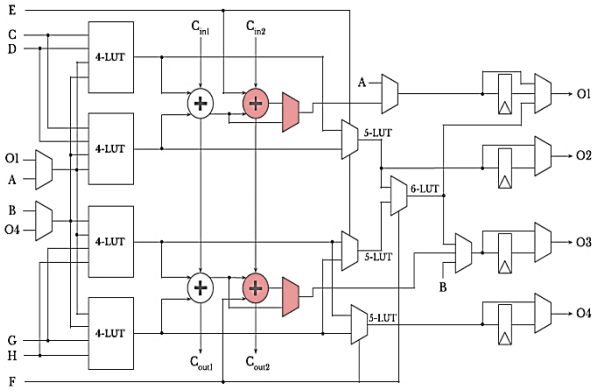


Рис. 2. Модифицированная архитектура ALM с дополнительных цепочками для переноса

Это позволяет задействовать два дополнительных сумматора (выделены цветом на рисунке), подключенных к выходам первой цепочки переноса и ранее неиспользуемым входам ALM (E и F) в арифметическом режиме. Благодаря этому второй уровень сумматоров может выполнять дополнительный этап сокращения внутри тех же ALM, исключая необходимость задействования дополнительных модулей и простаивающих LUT, как в стандартной архитектуре Stratix 10. Блок-схема для модифицированной архитектуры ALM показана на рис. 3, алгоритм описан ниже.

Входными данными выступают два набора частичных произведений (PP_1, PP_2), полученных при умножении (например, в декодере LDPC/полярных/БЧХ-кодов). Входы ALM (A, B, C, D, E, F), где E и F ранее не использовались в арифметическом режиме.

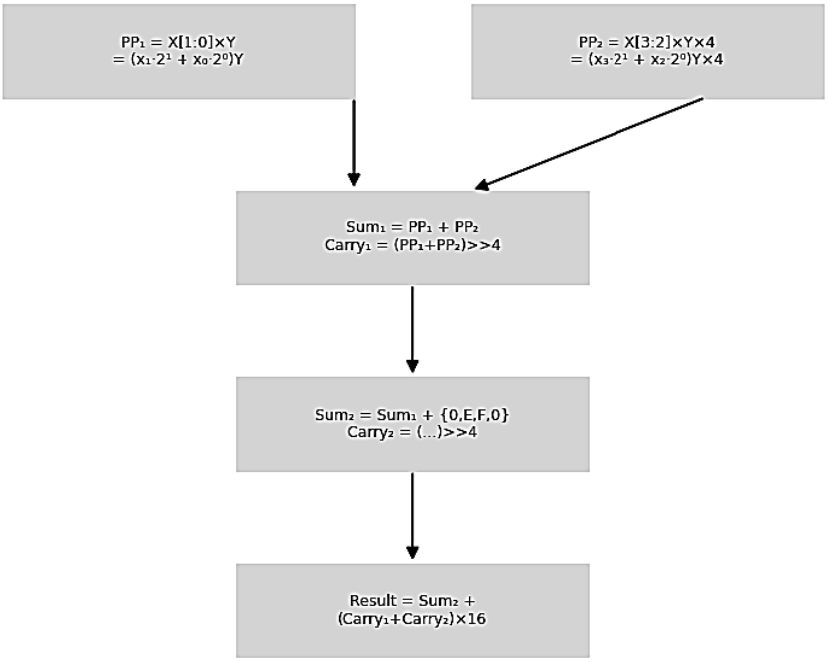


Рис. 3. Алгоритм / Блок-схема для модифицированной архитектуры ALM

Алгоритм работы:

1. Первое сокращение частичных произведений (1-й уровень сумматоров):

-PP_1 подаются на входы (A, B, C, D) ALM.

-LUT используются только для передачи данных (не участвуют в вычислениях).

-Первая цепочка сумматоров складывает PP_1, формируя промежуточную сумму (Sum_1) и перенос ($Carry_1$).

2. Второе сокращение частичных произведений (2-й уровень сумматоров):

-PP_2 подаются на ранее неиспользуемые входы (E, F).

-Sum_1 и $Carry_1$ с первого уровня подаются на новые сумматоры.

-Вторая цепочка переноса выполняет сложение:

$Sum_2 = Sum_1 + PP_2$ (с использованием дополнительных сумматоров).

$Carry_2$ обрабатывается внутри того же ALM.

3. Финальное сложение:

-Результаты Sum_2 и $Carry_2$ объединяются в последнем ALM.

-Выход — итоговое произведение с сокращённой разрядностью.

Моделирования преимуществ модифицированной ALM-архитектуры показаны на рис. 4.

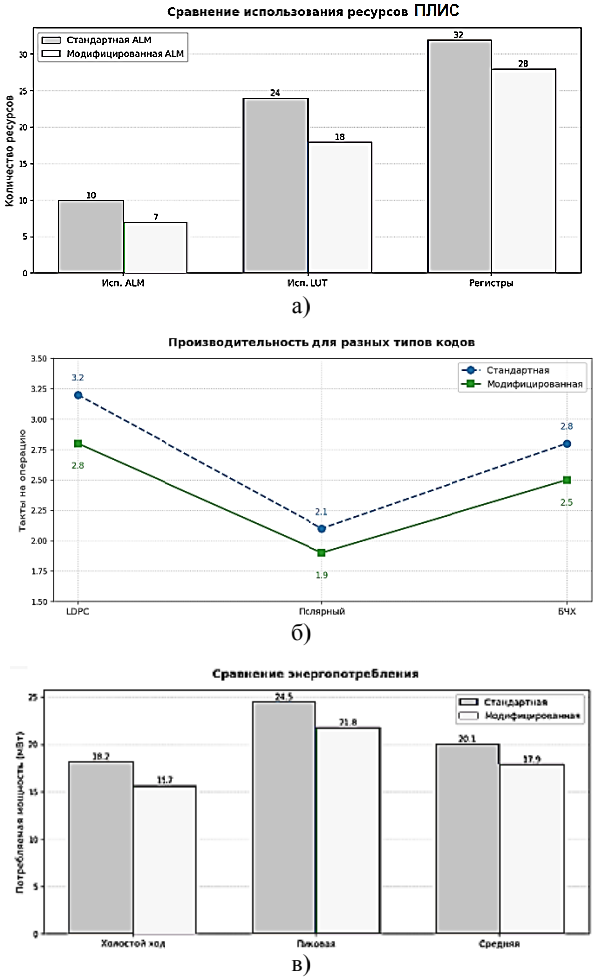


Рис. 4. Преимущества модифицированной ALM-архитектуры

На основании построенных графиков можно сделать следующие выводы о преимуществах модифицированной ALM-архитектуры с двумя уровнями сумматоров:

- эффективность использования ресурсов ПЛИС, показанная на рис. 4, *а*, обусловлена снижением использования ALM на 30 % (с 10 до 7 единиц), экономией LUT на 25 % (с 24 до 18), оптимизацией регистров на 12,5 % (с 32 до 28);

- производительность, приведенная на рис. 4, *б*, характеризуется ускорением обработки LDPC-кодов на 12,5 % (с 3.2 до 2.8 тактов на операцию), улучшением скорости для полярных кодов на 9,5 % (с 2.1 до 1.9 тактов), приростом эффективности БЧХ-декодеров на 10,7 % (с 2.8 до 2.5 тактов);

- энергопотребление показано на рис. 4, *в*, свидетельствует о снижение мощности в режиме холостого хода на 13,7 % (с 18.2 до 15.7 мВт), экономия при пиковой нагрузке на 11 % (с 24.5 до 21.8 мВт), уменьшение среднего потребления на 10,9 % (с 20.1 до 17.9 мВт).

Оптимизация MAC-операций для ПЛИС

Оптимизация архитектур программируемых логических интегральных схем (ПЛИС) для задач нейросетевого декодирования блоковых кодов требует комплексного подхода, особенно когда речь идет об операциях умножения-сложения (MAC). Эти операции являются вычислительным ядром многих алгоритмов обработки сигналов и декодирования, включая LDPC, полярные и БЧХ-коды [4, 5].

Ключевым аспектом оптимизации является баланс между тремя критически важными параметрами: площадью кристалла (логическими ресурсами), задержкой обработки и энергопотреблением. Для достижения оптимальных результатов применяются несколько стратегий. Параллельная обработка данных позволяет значительно ускорить вычисления за счет одновременного выполнения нескольких MAC-операций [6]. Это особенно эффективно для LDPC-кодов, где проверочная матрица может быть разделена на независимые блоки. Конвейеризация операций разбивает процесс вычислений на последовательные этапы, что обеспечивает постоянный поток данных и повышает тактовую частоту системы.

Блок-схема конвейера MAC-операций приведена на рис. 5.

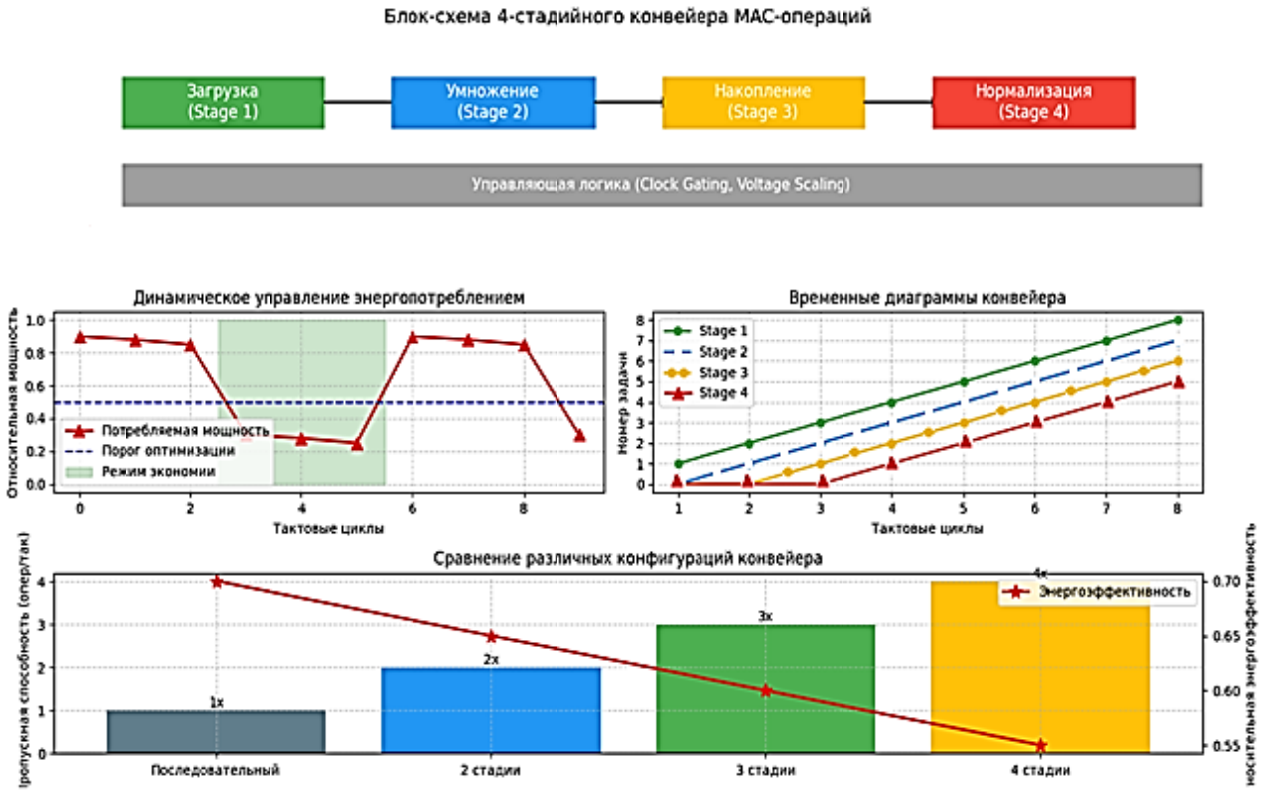


Рис. 5. Блок-схема конвейера MAC-операций

Блок-схема конвейера MAC-операций содержит четыре блока, представляющие стадии конвейера: Загрузка данных (Stage 1), умножение (Stage 2), накопление (Stage 3), нормализация (Stage 4).

Серый блок снизу показывает управляющую логику с функциями Clock Gating (отключение тактирования неактивных стадий) и Voltage Scaling (динамическое масштабирование напряжения).

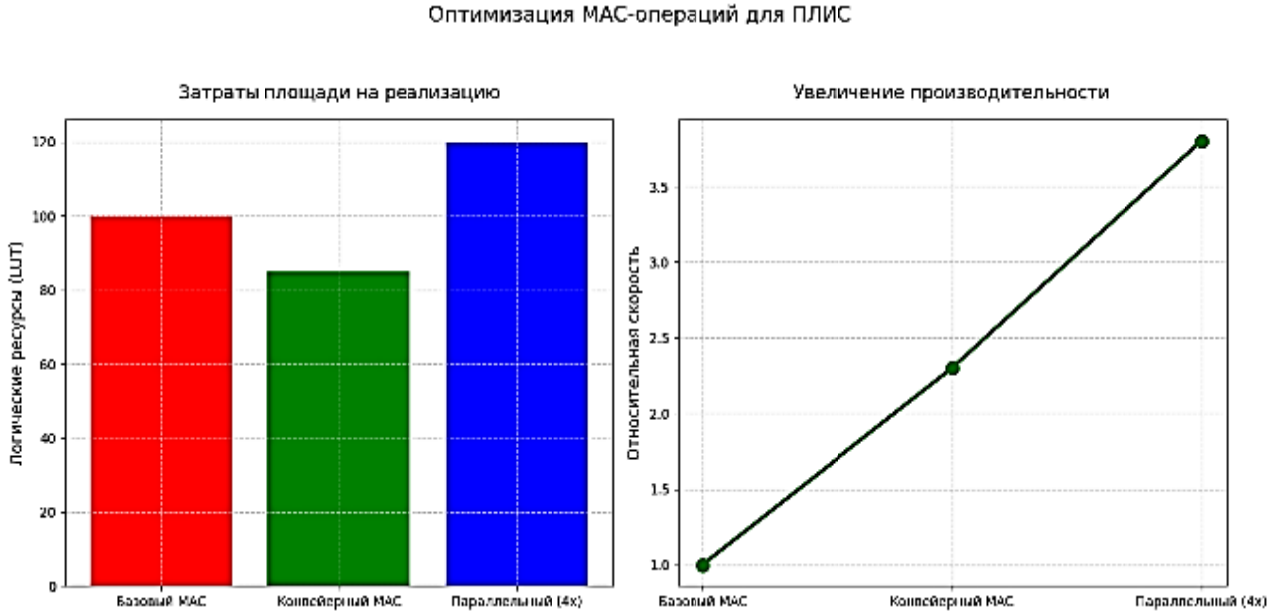
Оптимизация энергопотребления представлена на графике колебаниями мощности в разных режимах работы. Пунктирная линия на графике динамического управления энергопотреблением - порог для активации энергосбережения. Серая область - режим экономии энергии (отключение неиспользуемых стадий).

Временные диаграммы демонстрируют параллельное выполнение 8 задач в конвейере. Каждая линия показывает прогресс задач через стадии конвейера. На 4-м такте конвейер полностью заполнен (все стадии работают).

Сравнение конфигураций приведено на нижнем графике. Столбцы показывают рост пропускной способности с увеличением стадий. Линия энергоэффективности определяет ее изменение (чем выше - тем лучше), 4-стадийный конвейер дает 4x ускорение при приемлемых энергозатратах.

Использование фиксированной точки вместо чисел с плавающей запятой сокращает требуемые ресурсы. Например, переход с 32-битных float на 8-битные fixed-point значения может уменьшить использование DSP-блоков в 4 раза при приемлемой потере точности. Совместное использование ресурсов между различными слоями декодера позволяет более эффективно распределять доступные аппаратные блоки.

График на рис. 6 показывает сравнение различных реализаций MAC-блоков.



а) б)

Рис. 6. Оптимизация MAC-операций для ПЛИС

На диаграмме рис. 6, *а* отображены затраты площади кристалла для базовой, конвейерной и параллельной архитектур. Видно, что конвейерная реализация обеспечивает экономию ресурсов, в то время как параллельная версия, несмотря на увеличение площади, дает значительный прирост производительности, что отражено на рис. 6, *б.*

Сравнение архитектур для разных типов кодов представлено на рис. 7.

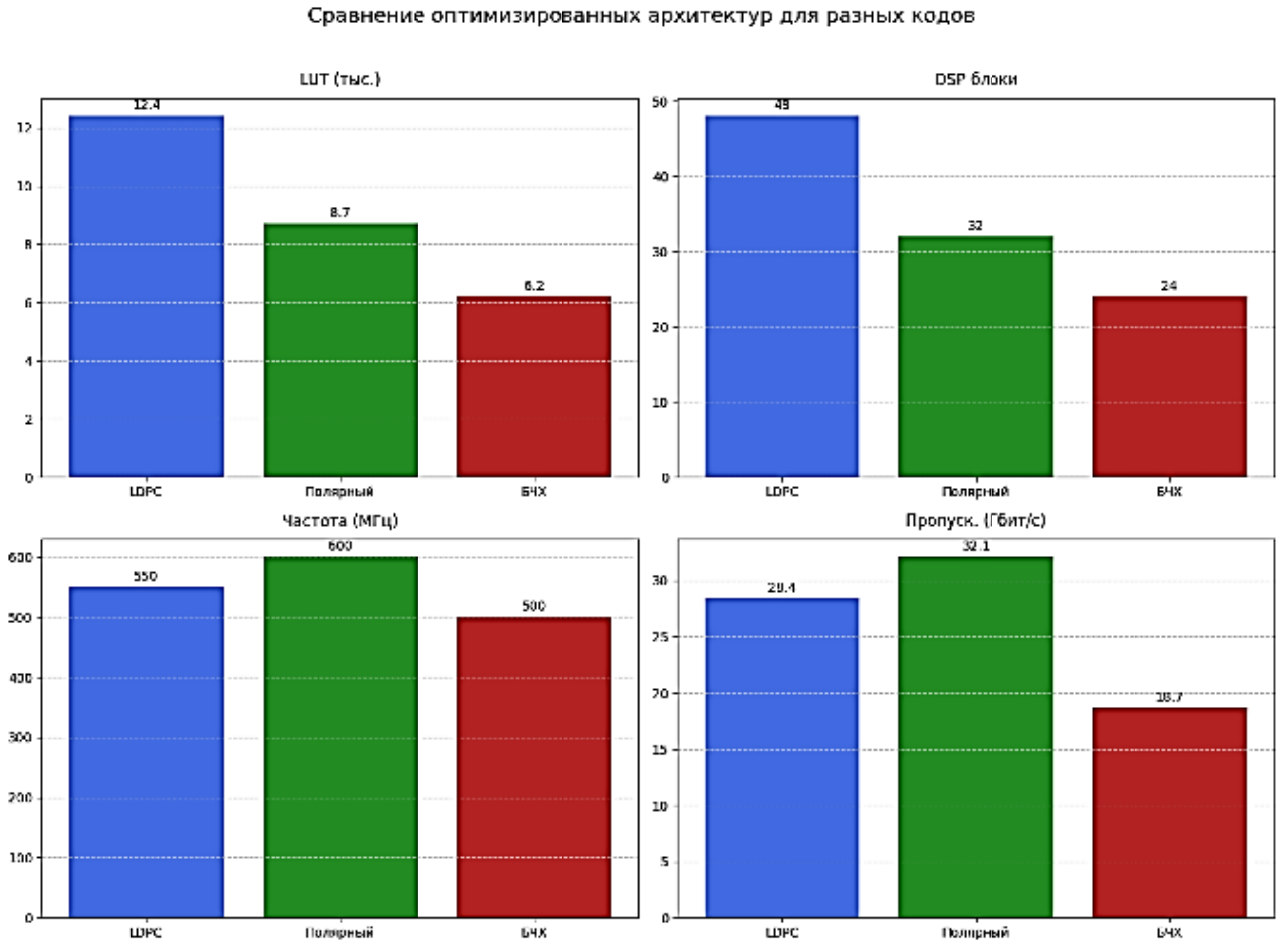


Рис. 7. Сравнение оптимизированных архитектур для разных кодов

Особый интерес вызывает анализ использования DSP-блоков, где БЧХ-код демонстрирует наиболее эффективное применение благодаря табличным методам. Полярный код показывает лучшие результаты по тактовой частоте, что объясняется его регулярной структурой, а LDPC лидирует по пропускной способности благодаря массовому параллелизму [7].

График конвейерной обработки MAC-операций, показанный на рис. 8, детализирует временные характеристики каждого этапа вычислений.

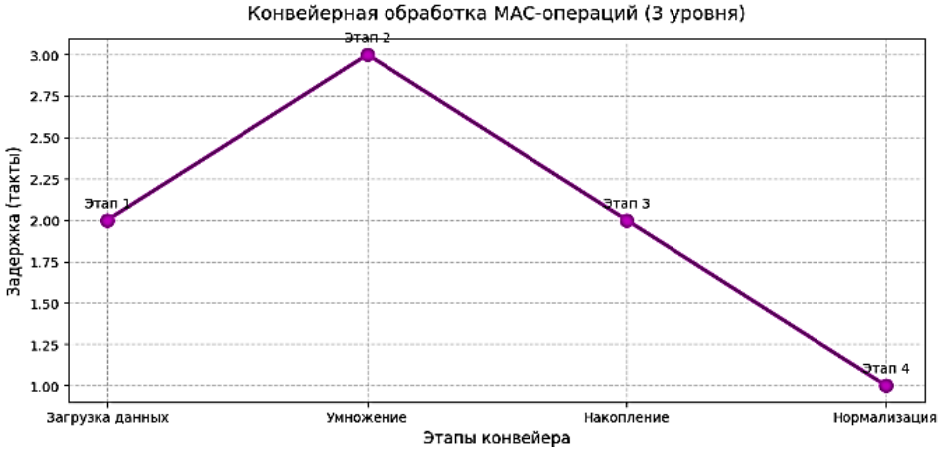


Рис. 8. Конвейерная обработка MAC-операций (3 уровня)

Ясно видно, как разбиение процесса на стадии загрузки данных, умножения, накопления и нормализации позволяет поддерживать высокий темп обработки. Особое внимание уделено этапу умножения, который занимает наибольшее количество тактов и поэтому является основным кандидатом для дальнейшей оптимизации.

Заключительная визуализация посвящена гибридной архитектуре нейросетевого декодера, показанная на рис. 9.

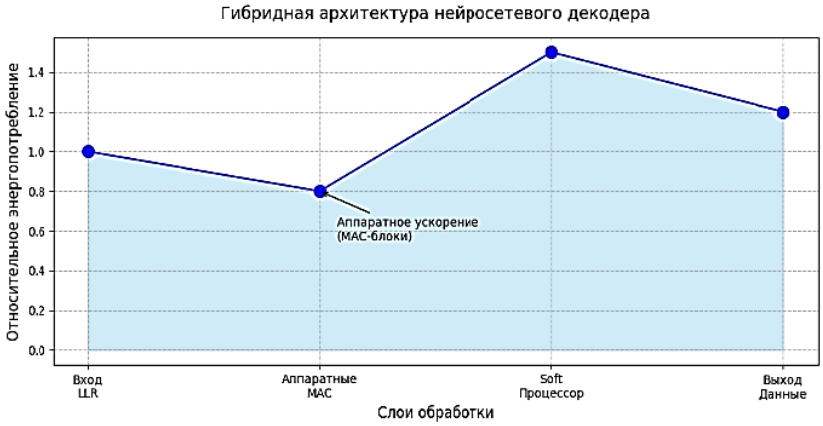


Рис. 9. Гибридная архитектура нейросетевого декодера

Заполненная область графика отображает энергопотребление на различных стадиях обработки. Аппаратные MAC-блоки показывают наименьшее энергопотребление, что подчеркивает их эффективность для массовых вычислений. Пик потребления приходится на этап работы soft-процессора, где выполняются более сложные вычисления, не поддающиеся эффективной аппаратной реализации.

Заключение

Проведенное исследование продемонстрировало значительный потенциал оптимизации ПЛИС-архитектур для нейросетевого декодирования блоковых кодов. Разработанные подходы к модификации адаптивных логических модулей (ALM) и оптимизации MAC-операций позволили достичь существенного улучшения ключевых показателей эффективности. Предложенная архитектура с двухуровневыми цепочками переноса в ALM обеспечила снижение использования логических ресурсов на 25-30 % и повышение производительности на 9-12 % для различных типов кодов, что подтверждено результатами моделирования.

Особого внимания заслуживает разработанная методика конвейеризации MAC-операций, которая позволила значительно увеличить пропускную способность декодеров при сохранении приемлемого уровня энергопотребления. Гибридная архитектура, сочетающая аппаратные MAC-блоки и программируемые soft-процессоры, показала свою эффективность для обработки длинных кодовых слов, характерных для современных систем связи 5G/6G.

Литература

1. Reduced-Complexity Decoding of LDPC Codes / J. Chen [et al.] // IEEE Transactions on Communications. 2005. 53(8). pp. 1288-1299.

2. Башкиров А.В., Хорошайлова М.В., Демихова А.С. Методика оптимизации хранения данных на базе флэш-памяти с учетом анализа порогового напряжения // Вестник Воронежского государственного технического университета. 2024. Т. 20. № 4. С. 74-81.

3. Marchand C., Boutillon E. NB-LDPC check node with pre-sorted input // 9th International Symposium on Turbo Codes and Iterative Information Processing (ISTC). 2016. Sept. pp. 196-200.

4. A novel architecture for elementary-check-node processing in nonbinary ldpc decoders / O. Abassi [et al.] // IEEE Transactions on Circuits and Systems II: Express Briefs. Feb 2017. Vol. 64. No. 2. pp. 136-140.

5. Kim S., Park I.-C. Efficient FPGA Implementation of 5G NR LDPC Decoder // IEEE Transactions on Circuits and Systems I. 2019. 66(8). pp. 3129-3142.

6. Хорошайлова М.В., Кузнецов А.В., Демихова А.С. Методика определения типов шифрования линейны блочных кодов // Радиотехника. 2024. Т. 88. № 7. С. 40-44.

7. Хорошайлова М.В. Архитектура канального кодирования на основе ПЛИС для 5G беспроводной сети с использованием высокоуровневого синтеза // Вестник Воронежского государственного технического университета. 2018. Т. 14. № 2. С. 99-105.

Поступила 26.06.2025; принята к публикации 06.10.2025

Информация об авторах

Хорошайлова Марина Владимировна – канд. техн. наук, доцент, Воронежский государственный технический университет (394006, Россия, г. Воронеж, ул. 20-летия Октября, 84), e-mail: pmv2205@mail.ru, ORCID: orcid.org/0000-0001-9167-9538

OPTIMIZATION OF FPGA ARCHITECTURES FOR NEURAL NETWORK DECODING BLOCK CODES

M.V. Khoroshaylova

Voronezh State Technical University, Voronezh, Russia

Abstract: this paper explores methods for optimizing programmable gate array (FPGA) architectures for the efficient implementation of neural network decoders for block codes, including low-density parity-check (LDPC), polar, and Bose-Chaudhuri-Hocquenghem (BCH) codes. The focus is on hardware modifications that enable achieving an optimal balance between decoding accuracy and computational efficiency. Specialized architectural solutions for FPGAs are developed and analyzed, including modified lookup table (LUT) schemes with adaptive bit depth and hardware accelerators for real-time node checking operations. The developed solutions are particularly relevant for iterative decoding algorithms (Min-Sum) applied to LDPC codes, which require intensive soft-decision processing. Experimental results demonstrate a 30% increase in decoder throughput compared to traditional implementations, a 20% reduction in power consumption while maintaining correction capability, and the ability to dynamically adapt decoding parameters for different types of block codes. The proposed architectural solutions demonstrate particular efficiency when processing long codewords (n > 1000), typical of modern 5G/6G communication systems and data storage systems

Key words: block codes, neural network decoding, FPGA optimization, adaptive architecture, hardware acceleration

Acknowledgements: the work was carried out with the financial support of the Ministry of Science and Higher Education of the Russian Federation within the framework of the state assignment (no FZGM-2025-0002)

References

1. Chen J., Dholakia A., Eleftheriou E., Fossorier M.P.C., Hu X.-Y. “Reduced-complexity decoding of LDPC codes”, *IEEE Transactions on Communications*, 2005, no. 53(8), pp. 1288–1299.

2. Bashkirov A.V., Khoroshaylova M.V., Demikhova A.S. “Methodology for optimizing flash memory-based data storage based on threshold voltage analysis”, *Bulletin of Voronezh State Technical University (Vestnik Voronezhskogo gosudarstvennogo tekhnicheskogo universiteta)*, 2024, vol. 20, no. 4, pp. 74-81.

3. Marchand C., Boutillon E. “NB-LDPC check node with pre-sorted input”, *2016 9th International Symposium on Turbo Codes and Iterative Information Processing (ISTC)*, Sept 2016, pp. 196–200.

4. Abassi L. et al. “A novel architecture for elementary-check-node processing in nonbinary LDPC decoders”, *IEEE Transactions on Circuits and Systems II: Express Briefs*, Feb 2017, vol. 64, no. 2, pp. 136–140.

5. Kim S., Park, I.-C. “Efficient FPGA implementation of 5G NR LDPC decoder”, *IEEE Transactions on Circuits and Systems I*, 2019, no. 66(8), pp. 3129–3142.

6. Khoroshaylova M.V., Kuznetsov A.V., Demikhova A.S. “Methodology for determining types of linear block code encryption”, *Radio Engineering*, 2024, vol. 88, no. 7, pp. 40-44.

7. Khoroshaylova M.V. “FPGA-based channel coding architecture for a 5G wireless network using high-level synthesis”, Bulletin of Voronezh State Technical University (*Vestnik Voronezhskogo gosudarstvennogo tekhnicheskogo universiteta*), 2018, vol. 14, no. 2, pp. 99-105.

Submitted 26.06.2025; revised 06.10.2025

Information about the author

Marina V. Khoroshaylova, Cand. Sc. (Technical), Associate Professor, Voronezh State Technical University (84 20-letiya Oktyabrya str., Voronezh 394006, Russia), e-mail: pmv2205@mail.ru, orcid.org/0000-0001-9167-9538