**Оригинальная статья / Original article**

УДК 004.383.3

https://doi.org/10.21869/2223-1560-2025-29-4-111-124



# Уменьшение аппаратных затрат цифровой фильтрации в системе остаточных классов на основе усеченных блоков умножения с накоплением

**П. А. Ляхов** [1] ✉

[1] Северо-Кавказский федеральный университет
ул. Пушкина, д. 1, г. Ставрополь 355017, Российская Федерация

✉ e-mail: ljahov@mail.ru

## Резюме

***Цель исследования.*** *Параллельная обработка данных на основе системы остаточных классов позволяет уменьшить аппаратные затраты устройств цифровой фильтрации сигналов, что является одной из ключевых проблем цифровой обработки сигналов. Распараллеливание вычислений позволило разработать метод цифровой фильтрации сигналов на основе использования усеченных блоков умножения с накоплением в системе остаточных классов. В данной статье представлены преимущества применения разработанного подхода и его ограничения.*

***Методы.*** *В исследовании применялись методы организации вычислений в системе остаточных классов с диапазонами в 32 и 48 бит и с использованием сбалансированных наборов модулей вида* $\{2^n-1,\ 2^n,\ 2^n+1\}$, *аналитической оценки сложности вычислительного устройства и аппаратное моделирование в среде Synopsys Design Compiler с использованием стандартной библиотеки.*

***Результаты.*** *Снижение аппаратных затрат зафиксировано при использовании модулей специального вида* $\{2^n-1,\ 2^n,\ 2^n+1\}$, *позволяющих уменьшить их до 16 139,30* $\text{мкм}^2$ *для фильтров 3-го порядка, 31 152,99* $\text{мкм}^2$ *для фильтров 7-го порядка, 62 507,06* $\text{мкм}^2$ *для фильтров 15-го порядка и 126 564,46* $\text{мкм}^2$ *для фильтров 31-го порядка с организацией арифметической обработки 32-разрядных данных в системе остаточных классов. Таким образом, аппаратные затраты были снижены на 21,5%-23,0% относительно фильтров на основе параллельно-префиксных сумматоров по методу Когге-Стоуна и на 20,6%-22,2% на основе сумматоров параллельного переноса с распространением. Для 48-битных цифровых фильтров с арифметической обработкой данных в системе остаточных классов результаты моделирования показали уменьшение аппаратных затрат от 9,45% до 14%, в зависимости от их порядка.*

***Заключение.*** *Проведение вычислений в системе остаточных классов позволяет улучшить эксплуатационные характеристики устройств цифровой обработки сигналов, для которых первостепенной задачей является минимизация аппаратных затрат.*

***Ключевые слова:*** *система остаточных классов; цифровая обработка сигналов; цифровая фильтрация; площадь аппаратной реализации;, блок умножения с накоплением.*

***Конфликт интересов:*** *Автор декларирует отсутствие явных и потенциальных конфликтов интересов, связанных с публикацией настоящей статьи.*

**Для цитирования:** Ляхов П. А. Уменьшение аппаратных затрат цифровой фильтрации в системе остаточных классов на основе усеченных блоков умножения с накоплением // Известия Юго-Западного государственного университета. 2025; 29(4): 111-124. https://doi.org/ 10.21869/2223-1560-2025-29-4-111-124.

*Поступила в редакцию 03.07.2025* *Подписана в печать 28.08.2025* *Опубликована 22.12.2025*

# Reduction of digital filtering area in the residue number system based on truncated multiplication with accumulation blocks

**Pavel A. Lyakhov** [1] ✉

[1] North-Caucasus Federal University
1, Pushkin str., Stavropol 355017, Russian Federation

✉ e-mail: ljahov@mail.ru

## Abstract

***Purpose of research.*** *Parallel data processing based on the residue number system allows to reduce the hardware costs of digital signal filtering devices, which is one of the key problems of digital signal processing. Parallelization of calculations allowed to develop a method of digital signal filtering based on the use of truncated blocks of multiplication with accumulation in the residue number system. This article presents the advantages of using the developed approach and its limitations.*

***Methods.*** *The study used a method for organizing calculations in a system of residual classes with ranges of 32 and 48 bits and using balanced sets of modules of the form* $\{2^n - 1,\ 2^n,\ 2^n + 1\}$*, an analytical assessment of the complexity of the device calculation, and hardware modeling in the Synopsys Design Compiler environment using the standard library.*

***Results.*** *The hardware cost reduction was recorded when using special modules* $\{2^n - 1,\ 2^n,\ 2^n + 1\}$*, which allowed them to be reduced to 16,139.30* $\mu m^2$ *for 3-rd order filters, 31,152.99* $\mu m^2$ *for 7-th order filters, 62,507.06* $\mu m^2$ *for 15-th order filters, and 126,564.46* $\mu m^2$ *for 31-st order filters with the organization of 32-bits arithmetic data processing in the residue number system. Thus, the hardware costs were reduced by 21.5%-23% relative to filters based on parallel-prefix adders using the Kogge-Stone method and by 20.6%-22.2% based on parallel carry adders with propagation. For 48-bit digital filters with arithmetic processing of data in the residue number system, the simulation results showed a reduction in hardware costs from 9.45% to 14% depending on their order.*

***Conclusion.*** *Carrying out calculations in the system of residual classes allows improving the operational characteristics of digital signal processing devices, for which the primary task is to minimize hardware costs.*

***Keywords:*** *residue number system; digital signal processing; digital filtering; hardware implementation area; multiply-accumulate block.*

***Conflict of interest.*** *The Author declare the absence of obvious and potential conflicts of interest related to the publication of this article.*

**For citation:** Lyakhov P. A. Reduction of digital filtering area in the residue number system based on truncated multiplication with accumulation blocks // *Izvestiya Yugo-Zapadnogo gosudarstvennogo universiteta = Proceedings of the Southwest State University*. 2025; 29(4): 111-124 (In Russ.). https://doi.org/10.21869/2223-1560-2025-29-4-111-124.

*Received 03.07.2025* *Accepted 28.08.2025* *Published 22.12.2025*

***

## Введение

Обработка цифровых сигналов лежит в основе многих отраслей современной промышленности [1]. Одним из методов их обработки на практике является цифровая фильтрация (ЦФ) с конечной импульсной характеристикой, использование которой позволяет избежать ошибок в вычислениях за счет их устойчивости и отсутствия обратной связи [2]. Реа-

лизация ЦФ требует определенных, зачастую весьма существенных, аппаратных, временных и энергетических затрат [3].

Проблемой уменьшения потребления ресурсов устройствами ЦФ занимаются научные коллективы со всего мира. Так, исследователи из Японии для уменьшения аппаратных затрат устройства ЦФ используют конструкцию симметричной свертки, которая позволяет уменьшить число множителей, благодаря чему достигается желаемый результат [4].

Еще одним способом уменьшения ресурсных затрат является использование параллельной обработки данных, например, с использованием естественного арифметического параллелизма системы остаточных классов (СОК) [5]. Сложность использования СОК для ЦФ заключается в выборе эффективного набора модулей, учитывая существующее ограничение к ним, заключающееся в попарной взаимной простоте друг к другу. Подобрать такие модули пытались многие зарубежные исследователи. Например, Хиасат А. предложил метод определения знака числа в СОК на основе пяти модулей $\{2^{2n+1},2^n-1,2^n+1, 2^n-2^{n+\frac{1}{2}}+1,2^n+2^{n+\frac{1}{2}}+1\}$, где $n=2k+1$, $k>0$ и $p\in N, p<n-\frac{5}{2}$ [6]. Тораби и Джаберипур предложили метод сравнения чисел в СОК на основе модулей $\{2^n-1, 2^n, 2^n+1\}$ [7]. Коллеги из Индии составили краткий обзор методов преобразований в СОК чисел с набором модулей $\{2^n-1, 2^n, 2^n+1\}$ [8].

Цель настоящего исследования заключается в снижении аппаратных затрат на проведение ЦФ сигналов в СОК. В работе представлен модифицированный метод ЦФ в СОК на основе усеченных блоков умножения с накоплением. Представлены преимущества применения разработанного подхода и его ограничения.

Оставшаяся часть статьи организована следующим образом. В разделе «Материалы и методы» излагается предложенный метод реализации усеченных блоков умножения с накоплением для модулей СОК специального вида. В разделе «Результаты и их обсуждение» представлены теоретическая оценка производительности разрабатываемых устройств и их аппаратное моделирование на современных специализированных вычислительных устройствах. В заключение подводятся итоги работы.

## Материалы и методы

Методы реализации ЦФ на основе модулей СОК вида $2^n+1$ имеют ряд недостатков, связанных с затратами на арифметические операции, например, для необходимого специализированного преобразования Diminished-one (*D*-1) [9]. Такое преобразование позволяет повысить эффективность вычислений по модулю $2^n+1$. Принцип преобразования *D*-1 заключается в следующем. Если остаток по модулю $2^n+1$ отличен от 0, то из него вычитается единица. Если остаток по модулю $2^n+1$ равен 0, то значение старшего значащего бита указанного остатка заменяется на 1. Сложение и умножение чисел по модулю

$2^n+1$ с использованием D-1 преобразования выполняются согласно выражениям:

$$x'+y'+\overline{C_{вых}}=s', \quad (1)$$

$$x'y'+x'+y'=p', \quad (2)$$

где $x'$ и $y'$ являются операндами, представленными в кодах $D$-1; $s'$ и $p'$ являются результатами сложения и умножения, соответственно, которые представлены в коде $D$-1, а $\overline{C_{вых}}$ представляет собой противоположный элемент старшего бита переноса [10]. Обратное преобразование из $D$-1 в традиционное представление по модулю $2^n+1$ СОК происходит путем получения суммы младших бит значения $x'$ до $n$-1 бита и инверсии $\overline{x'_n}$:

$$x_{n:0}=x'_{n-1:0}+\overline{x'_n}, \quad (3)$$

где $x_{n:0}$ представляет собой исходное значение остатка по модулю $2^n+1$. Для уменьшения площади цифровой фильтрации в СОК использована методика распараллеливания вычислений со сбалансированным набором модулей, описанная далее.

В этой работе для наборов модулей СОК специального вида $2^n-1$, $2^n$ предлагается использовать модифицированные усеченные блоки умножения с накоплением (УБУН) и усеченный блок умножения с цикличным переносом старшего бита (УБУН-ЦПСБ) [11, 12]. Модификация заключается в использовании сумматора с последовательным переносом (СПП), представленного на рис. 1, и СПП с цикличным переносом старшего бита (СПП-ЦПСБ), представленного на рис. 2 заместо параллельно-префиксного сумматора Когге-Стоуна (СКС). На рис. 1 и 2 A,B – слагаемые, $S$ – бит суммы, блок ПС – блок полного сумматора, блок ПП – блок полусумматора.

Для модуля вида $2^n+1$ применен модифицированный УБУН-ЦПСБ, основанный на наличии блоков перевода из СОК в $D-1$ и обратно (рис. 3 и 4), для которых предложены модифицированные сумматоры ПП и СПП.

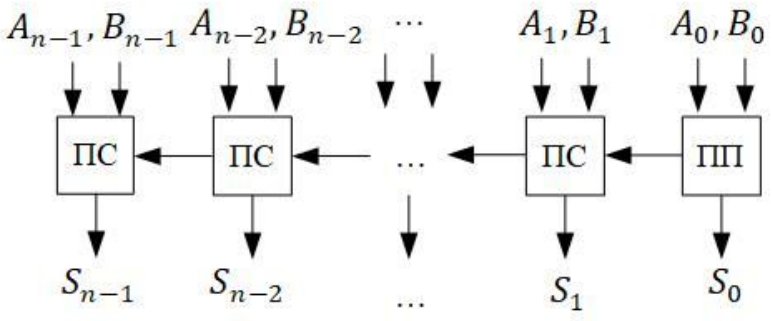


**Рис. 1.** Архитектура сумматора СПП по модулю $2^n$

**Fig. 1.** Architecture of CPA modulo $2^n$

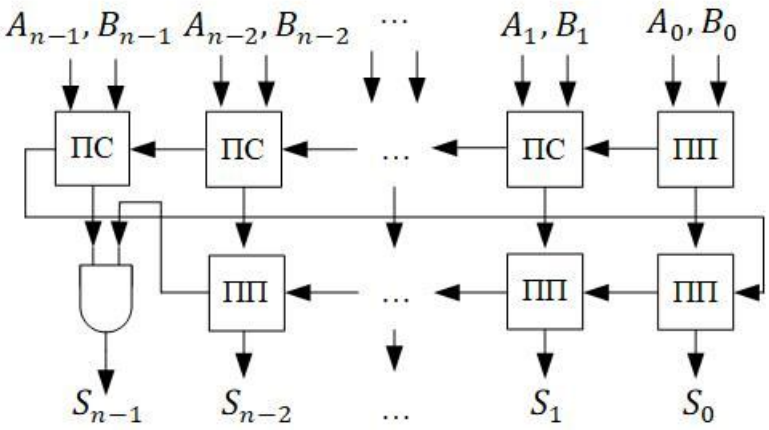


**Рис. 2.** Архитектура сумматора СПП-ЦПСБ по модулю $2^n - 1$

**Fig. 2.** Architecture of EAC-CPA modulo $2^n - 1$

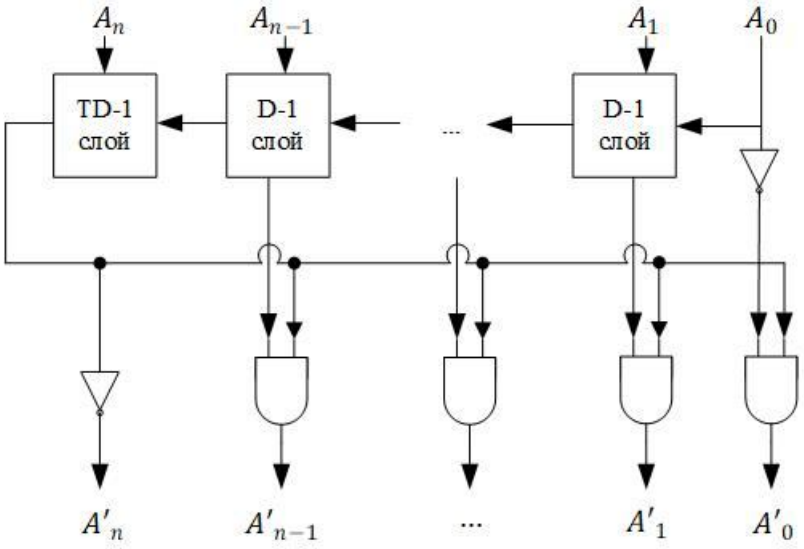


**Рис. 3.** Схема реализации метода преобразования числа по модулю $2^n + 1$ в $D - 1$, основанного на СПП

**Fig. 3.** Scheme of implementation of the method of converting a number by modulo $2^n + 1$ into $D - 1$ based on CPA

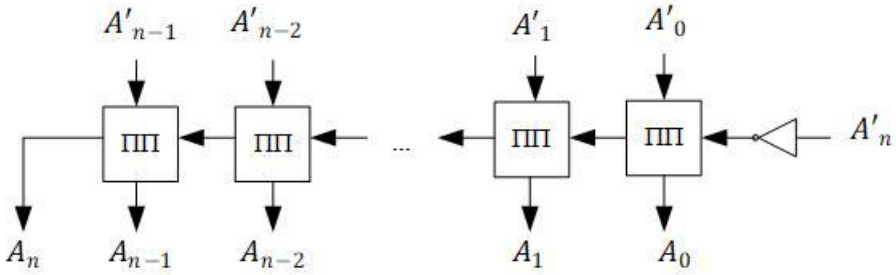


**Рис. 4.** Схема реализации метода преобразования числа по модулю $2^n + 1$ в $D - 1$, основанного на ПП

**Fig. 4.** Scheme of implementation of the method of converting a number by modulo $2^n + 1$ into $D - 1$ based on HA

Схема, представленная на рис. 3, использует сумматор СПП, упрощенный за счет следующих свойств логических выражений:

$$S=A\oplus 1\oplus D=\overline{A}\oplus D=A\odot D, \quad (4)$$

$$C=\big((A\oplus 1)\wedge D\big)\vee(A\wedge 1)=$$

$$=(\overline{A}\wedge D)\vee A=A\vee D, \quad (5)$$

$C$ – бит переноса, $\oplus,\odot,\wedge,\vee$ и $\overline{A}$ – логические элементы, обозначающие “Исключающее ИЛИ”, “Исключающее НЕ-ИЛИ”, “И”, “ИЛИ”, “НЕ”, соответственно. Схемы блоков для преобразования чисел по модулю $2^n+1$ в $D$-1, используемые в рис. 3, представлены на рис. 5. Для обратного перевода числа из $D$-1 достаточно сложить старший бит с его инверсией, что продемонстрировано на рис. 3.

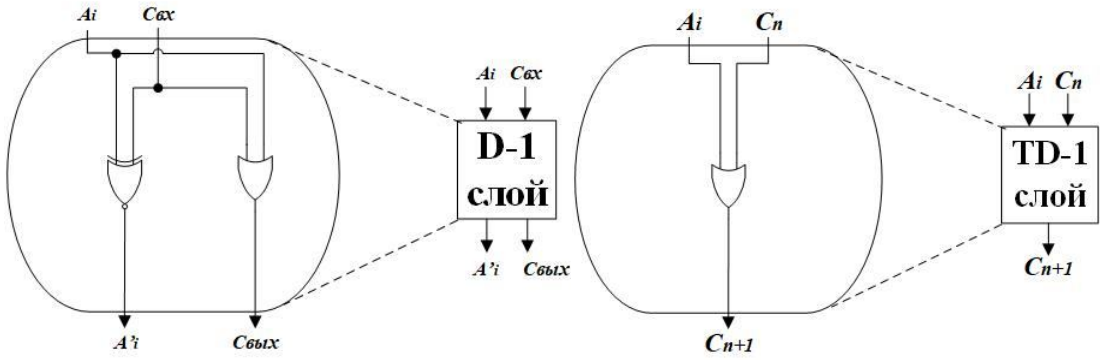


**Рис. 5.** Схемы метода вычисления для перевода числа в $D-1$

**Fig. 5.** Schemes of the calculation method for converting a number to $D-1$

Блоки УБУН, УБУН-ЦПСБ и модифицированный УБУН-ЦПСБ используют генератор частичных произведений (ГЧП) и дерево сумматоров с сохранением переноса, подробно описанных в [13]. Отличительной чертой модифицированный УБУН-ЦПСБ от двух остальных блоков является отличное от них число операций сложений, наличие мультиплексоров и определитель старшего бита множителей. Если старший бит одного из множителей равен «1» (обозначает значение “0” в кодах $D-1$), то результатом вычисления будут два слагаемых с предыдущего модифицированного УБУН-ЦПСБ (рис. 6). Если это первый блок УБУН, то результатом вычисления будут два слагаемых, сумма которых обращается в нуль. В промежуточных вычислениях старшие биты множителей не отслеживаются. Завершающее суммирование двух чисел выполняется на основе предложенного модифицированного сумматора СПП-ЦПСБ, представленного на рис. 7. В отличие от СКС, старший бит переноса (БС) инвертируется и подается на последовательно соединенные блоки ПП. В первом ПП и в каждом ПС необходимо отслеживать результат операции сложения по модулю 2 двух чисел $A\oplus B$.

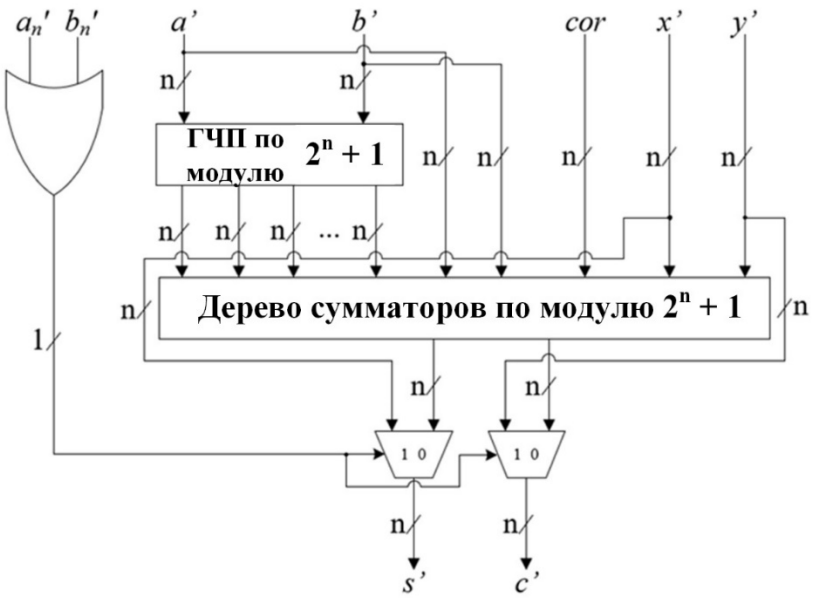


**Рис. 6.** Архитектура модифицированного УБУН-ЦПСБ

**Fig. 6.** Architecture of the modified IEAC-TMAC

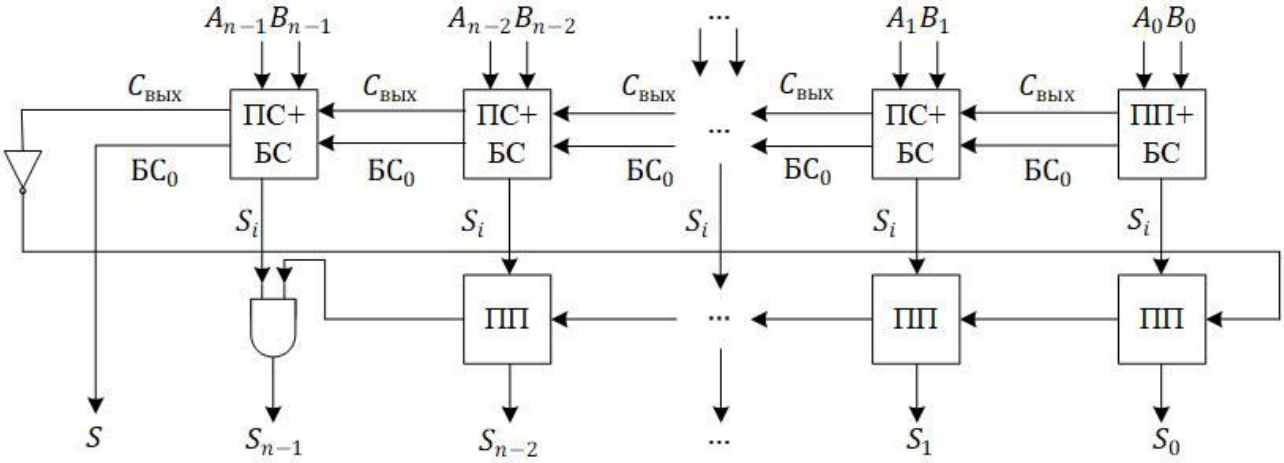


**Рис. 7.** Архитектура модифицированного сумматора СПП-ЦПСБ

**Fig. 7.** Architecture of the modified EAC-CPA adder

В следующем разделе автором представлены модель оценки сложности вычислительного устройства (МОСВУ), описанных в данном разделе и результаты аппаратной реализации предложенного метода.

## Результаты и их обсуждение

Автором рассмотрены фильтры с организацией вычислений в СОК с диапазонами в 32 и 48 бит и с использованием сбалансированных наборов модулей из [13]. Моделирование проведено для фильтров 3, 7, 15 и 31 порядка. Оценка предложенной архитектуры фильтра по аппаратным затратам произведена на основе МОВСУ [14]. В табл. 1 представлены результаты расчетов МОСВУ для каждого из блоков УБУН, УБУН - ЦПСБ и модифицированного УБУН -

ЦПСБ по различным модулям. По результатам в табл. можно заметить, что модифицированный УБУН - ЦПСБ занимает больше аппаратных ресурсов по сравнению с УБАН и УБУН - ЦПСБ.

**Таблица 1.** Расчет МОСВУ площади устройства для различных блоков УБУН

**Table 1.** Calculation of the computational device complexity estimation model (CDCEM) area of the device for various TMAC units

| | УБУН / UBUN | УБУН - ЦПСБ / UBUN - TSPSB | Модифицированный УБУН - ЦПСБ / Modified UBUN - TSPSB |
|---|---|---|---|
| Суммирование и умножение за 1 этап | $8n^2$ | $8n^2$ | $n^2$+27$n$+18n |
| Суммирование последних чисел | 7$n$ | 10$n$-5 | 11$n$-6 |
| $D-1$ преобразование | – | – | 7$n$-6 |
| Итого | $8kn^2$+$8n^2$+7$n$ | $8kn^2$+$8n^2$+10$n$-5 | $8kn^2$+27$kn$+$k$+$8n^2$+$45n$-11 |

Табл. 2 и 3 демонстрируют результаты оценки затрачиваемой площади устройства на основе МОСВУ для фильтров с 32- и 48-разрядными диапазонами СОК. Анализ полученных теоретических результатов показал, что предложенный фильтр имеет низкие аппаратные затраты по сравнению с известными конструкциями: на основе сумматоров СКС от 9,7% до 10,3% и на основе сумматоров СПП от 8,4% до 9,5% для 32-разрядных диапазонов СОК и от 9,7% до 10,3% на основе сумматоров СКС и от 8,4% до 9,5% на основе СПП, в зависимости от порядка фильтра.

**Таблица 2.** Оценка МОСВУ для фильтров с 32-разрядным диапазоном СОК на основе сумматоров СПП и СКС

**Table 2.** CDCEM evaluation for filters with a 32-bit range of RNS based on CPA and KSA adders

| Набор модулей / Set of modules | 3-й порядок / 3rd order | | 7-й порядок / 7th order | | 15-й порядок / 15th order | | 31-й порядок / 31st order | |
|---|---|---|---|---|---|---|---|---|
| | СПП | СКС | СПП | СКС | СПП | СКС | СПП | СКС |
| 2047, 2048, 4095 | 12649 | 12881,38 | 25001 | 25233,38 | 49705 | 49937,38 | 99113 | 99345,38 |
| 127, 255, 511, 512 | 9088 | 9274,13 | 17888 | 18074,13 | 35488 | 35674,13 | 70688 | 70874,13 |
| 31, 32, 127, 255, 511 | 8113 | 8284,20 | 15921 | 16092,20 | 31537 | 31708,20 | 62769 | 62940,20 |
| 7, 15, 16, 31, 127, 2047 | 7855 | 8015,21 | 15407 | 15567,21 | 30511 | 30671,21 | 60719 | 60879,21 |

**Таблица 3.** Оценка МОСВУ для фильтров с 48-разрядным диапазоном СОК на основе сумматоров СПП и СКС

**Table 3.** CDCEM evaluation for filters with a 48-bit range of RNS based on CPA and KSA adders

| Набор модулей / Set of modules | 3-й порядок / 3rd order | | 7-й порядок / 7th order | | 15-й порядок / 15th order | | 31-й порядок / 31st order | |
|---|---|---|---|---|---|---|---|---|
| | СПП | СКС | СПП | СКС | СПП | СКС | СПП | СКС |
| 127, 511, 1023, 2047, 4096 | 16274 | 16586,42 | 32114 | 32426,42 | 63794 | 64106,42 | 127154 | 127466,42 |
| 31, 127, 255, 511, 512, 2047 | 13910 | 14191,12 | 27382 | 27663,12 | 54326 | 54607,12 | 104876 | 105148,70 |

Моделирование разрабатываемых цифровых фильтров с вычислениями в СОК на аппаратном уровне проводилось в среде Synopsys Design Compiler с использованием стандартных параметров и библиотек. Вычисления проводились на рабочей станции с процессором Intel Core i5-12450H, с 16 ГБ оперативной памяти и 64-битной операционной системой Windows 10. Для сравнения выбирались сбалансированные наборы модулей СОК, продемонстрировавшие лучшие показатели по аппаратным затратам. Предложенный фильтр сравнивался с известными архитектурами с 32-битными и 48-битным диапазонами фильтра СОК [13, 15]. Результаты моделирования представлены в табл. 4 и 5.

Результаты аппаратного моделирования 32-разрядных фильтров в СОК продемонстрировали, что предложенный фильтр требует меньше аппаратных затрат, чем фильтры на основе СКС [13] на 21,5%-23% и 20,6%-22,2% на основе СПП [13], в зависимости от порядка фильтра. Для 48-разрядного фильтра СОК результаты моделирования показали уменьшение аппаратных затрат от 9,45% до 14% в зависимости от порядка фильтра [15-20].

**Таблица 4**. Результаты аппаратного моделирования 32-разрядных фильтров с вычислениями в СОК: площадь устройства, мкм$^2$

**Table 4.** Hardware modeling results for 32-bit filters with calculations in RNS: circuit area, $\mu m^2$

| Набор модулей / Set of modules | 3-й порядок / 3rd order | 7-й порядок / 7th order | 15-й порядок / 15th order | 31-й порядок / 31st order |
|---|---|---|---|---|
| {7, 15, 16, 31, 127, 2047} на основе сумматоров СКС | 20724,69 | 40456,07 | 79856,12 | 161175,75 |
| {7, 15, 16, 31, 127, 2047} на основе сумматоров СПП | 20346,38 | 40074,16 | 79465,79 | 160775,37 |
| {7, 9, 17, 31, 32, 65, 127} (предложенный) | 16139,30 | 31152,99 | 62507,06 | 126564,46 |

**Таблица 5.** Результаты аппаратного моделирования 48-разрядных фильтров с вычислениями в СОК: площадь устройства, мкм$^2$

**Table 5.** Hardware modeling results for 48-bit filters with calculations in RNS: circuit area, $\mu m^2$

| Набор модулей / Set of modules | 3-й порядок / 3rd order | 7-й порядок / 7th order | 15-й порядок / 15th order | 31-й порядок / 31st order |
|---|---|---|---|---|
| {64, 59, 53, 47, 43, 41, 37, 35, 33} | 36751,78 | 68714,95 | 135060,56 | 273103,34 |
| {17, 31, 32, 33, 65, 127, 257, 511} (предложенный) | 31490,98 | 61075,75 | 122237,34 | 247278,22 |

Приведенные данные позволяют сделать вывод о том, что использование реализации ЦФ, построенного на основе метода организации арифметической обработки со сбалансированной СОК вида $\{2^n\text{-}1,2^n,2^n\text{+}1\}$, уменьшает аппаратные расходы устройства. Полученный результат может быть использован для разработки перспективных систем и устройств цифровой обработки сигналов, изображений и видео.

## Выводы

В данной работе предлагается метод реализации цифровой фильтрации сигналов в СОК с использованием сбалансированных наборов модулей, включающих модули вида $2^n$+1. Результаты теоретического и практического моделирования показали уменьшение аппаратных затрат по сравнению с известными аналогами до 23%. Разработанные фильтры позволяют улучшить эксплуатационные характеристики устройств цифровой обработки сигналов, где первостепенной задачей является минимизация аппаратных затрат. Наиболее перспективными областями для практического внедрения разработанного подхода являются специализированные ускорители нейросетевых вычислений, медицинская визуализация и беспилотные транспортные системы.

## Список литературы

1. Sundararajan D. Digital Signal Processing: An Introduction // *Springer Nature*. 2024. 483 с. https://doi.org/10.1007/978-3-030-62368-5.

2. Kaur R., Singh Patterh M., Dhillon J.S. A new greedy search method for the design of digital IIR filter // *Journal of King Saud University - Computer and Information Sciences*. 2015; №27(3): 278–287. https://doi.org/10.1016/j.jksuci.2014.03.021.

3. Tomczak T. Fast Sign Detection for RNS $\{2^n\text{-}1,2^n,2^n\text{+}1\}$ // *IEEE Transactions on Circuits and Systems I: Regular Papers*. 2008. Vol. 55, № 6. P. 1502–1511. https://doi.org/10.1109/TCSI.2008.917994.

4. Ye J., Yanagisawa M., Shi Y. Scalable hardware efficient architecture for parallel FIR filters with symmetric coefficients // *Electronics*. 2022. №11(20). P. 3272. https://doi.org/10.3390/electronics11203272.

5. Omondi A. R., Premkumar A. B. Residue number systems: theory and implementation // *World Scientific*. 2007. Vol. 2.

6. Cheon J. E., Kang M., Kim T., Jung J., Yeo Y. Batch Inference on Deep Convolutional Neural Networks With Fully Homomorphic Encryption Using Channel-By-Channel Convolutions // *IEEE Transactions on Dependable and Secure Computing*. 2025. Vol. 22, no. 2. P. 1674-1685. https://doi.org/10.1109/tdsc.2024.3448406.

7. Mu L. Enhanced Redundant Residue Number System Codes for Reliable Diffusive Molecular Communication // *IEEE Transactions on Nanobioscience*. 2025. Vol. 24, no. 3. P. 366-373. https://doi.org/10.1109/tnb.2025.3553183.

8. Federated learning using a memristor compute-in-memory chip with in situ physical unclonable function and true random number generator / X. Li, B. Gao, Q. Qin, P. Yao, J. Li, H. Zhao, C. Liu, Q. Zhang, Z. Hao, Y. Li, D. Kong, J. Xu, J. Yang, J. Tang, Y. Niu, X. Yan, Q. He, H. Wu // *Nature Electronics*. 2025. Vol. 8, no. 6. P. 518-528. https://doi.org/10.1038/s41928-025-01390-6.

9. Isupov K. An overview of high-performance computing using the residue number system // *Program systems theory and applications.* 2021. Vol. 12, no. 2. P. 137-192. https://doi.org/10.25209/2079-3316-2021-12-2-137-192.

10. Selianinau M., Woźna-Szcześniak B. An Efficient Implementation of Montgomery Modular Multiplication Using a Minimally Redundant Residue Number System // *Applied Sciences (Switzerland)*. 2025. Vol. 15, no. 10. P. 5332. https://doi.org/10.3390/app15105332.

11. Hiasat A. A reverse converter and sign detectors for an extended RNS fivemoduli set // *IEEE Trans. Circuits Syst. I Regul. Pap*. 2017. № 64 (1). P. 111–121. https://doi.org/10.1109/TCSI.2016.2612723.

12. Torabi Z., Jaberipur G. Low-Power/Cost RNS comparison via partitioning the dynamic range // *IEEE Trans. Very Large Scale Integr. VLSI Syst*. 2016. № 24(5). P.1849–1857. https://doi.org/10.1109/TVLSI.2015.2484618.

13. Mohan P.A., Phalguna P.S. Evaluation of mixed-radix digit computation techniques for the three moduli RNS $\{2^n-1, 2^n, 2^n+1\}$ // *IEEE Trans. Circuits Syst. Express Briefs*. 2021. № 68(4). P.1418–1422. https://doi.org/10.1109/.

14. Bergerman M., Lyakhov P., Abdulsalyamova A. Modulo $2^k+1$ Truncated Multiply-Accumulate Unit // *International Conference on Actual Problems of Applied Mathematics and Computer Science. Cham: Springer Nature Switzerland*. 2022. P.343-352. https://doi.org/10.1007/978-3-031-34127-4_33

15. Gupta T., Akhter S. Design and implementation of area-power efficient generic modular adder using flagged prefix addition approach // *2021 7th International Conference*

*on Signal Processing and Communication (ICSC)*. 2021. P.302-307. https://doi.org/10.1109/icsc53193.2021.9673363.

16. Efstathiou C., Vergos H.T., Nikolos D. Fast parallel-prefix modulo $2^n$+1 adders // *IEEE Trans. Comput.* 2004. № 53(9). P. 1211–1216. https://doi.org/10.1109/TC.2004.60.

17. Бергерман М.В. Использование системы остаточных классов с модулями вида $\{2^n-1,2^n,2^n+1\}$ для снижения аппаратных затрат цифрового фильтра // Известия высших учебных заведений. Поволжский регион. Технические науки. 2023. №1. С.32-43. https://doi.org/10.21685/2072-3059-2023-1-3

18. High-performance digital filtering on truncated multiply-accumulate units in the residue number system / P. Lyakhov, M. Valueva, G. Valuev, N. Nagornov // *IEEE Access*. 2020. Vol. 8. P. 209181-209190. https://doi.org/10.1109/ACCESS.2020.3038496.

19. Rajanala A., Tyagi A. An area estimation technique for module generation // Proceedings., 1990 IEEE International Conference on Computer Design: VLSI in Computers and Processors, 1990. P. 459-462.

20. Belghadr A., Jaberipur G. Efficient variable-coefficient RNS-FIR filters with no restriction on the moduli set // *Signal, Image and Video Processing*. 2022. Vol. 16, № 6. P. 1443-1454. https://doi.org/10.1007/s11760-021-02097-9.

## References

1. Sundararajan D. Digital Signal Processing: An Introduction. *Springer Nature*. 2024; 483 p. https://doi.org/10.1007/978-3-030-62368-5.

2. Kaur R., Singh Patterh M., Dhillon J.S. A new greedy search method for the design of digital IIR filter. *Journal of King Saud University – Computer and Information Sciences*. 2015; (27): 278–287. https://doi.org/10.1016/j.jksuci.2014.03.021.

3. Tomczak T. Fast Sign Detection for RNS $\{2^n-1,2^n,2^n+1\}$. *IEEE Transactions on Circuits and Systems I: Regular Papers*. 2008; 55 (6):1502–1511. https://doi.org/10.1109/TCSI.2008.917994.

4. Ye J., Yanagisawa M., Shi Y. Scalable hardware efficient architecture for parallel FIR filters with symmetric coefficients. *Electronics*. 2022; (11): 3272. https://doi.org/10.3390/electronics11203272.

5. Omondi A. R., Premkumar A. B. Residue number systems: theory and implementation. *World Scientific*. 2007; 2.

6. Cheon J. E., Kang M., Kim T., Jung J., Yeo Y. Batch Inference on Deep Convolutional Neural Networks With Fully Homomorphic Encryption Using Channel-By-Channel Convolutions. *IEEE Transactions on Dependable and Secure Computing*. 2025, 22(2): 1674-1685. https://doi.org/10.1109/tdsc.2024.3448406.

7. Mu L. Enhanced Redundant Residue Number System Codes for Reliable Diffusive Molecular Communication. *IEEE Transactions on Nanobioscience*. 2025; 24(3): 366-373. https://doi.org/10.1109/tnb.2025.3553183.

8. Li X., Gao B., Qin Q., Yao P., Li J., Zhao H., Liu C., Zhang Q., Hao Z., Li Y., Kong D., Xu J., Yang J., Tang J., Niu Y., Yan X., He Q., Wu H. Federated learning using a memristor compute-in-memory chip with in situ physical unclonable function and true random number generator. *Nature Electronics*. 2025; 8(6): 518-528. https://doi.org/10.1038/s41928-025-01390-6.

9. Isupov K. An overview of high-performance computing using the residue number system. *Program systems theory and applications.* 2021; 12(2): 137-192. https://doi.org/10.25209/2079-3316-2021-12-2-137-192.

10. Selianinau M., Woźna-Szcześniak B. An Efficient Implementation of Montgomery Modular Multiplication Using a Minimally Redundant Residue Number System. *Applied Sciences (Switzerland)*. 2025; 15(10): 5332. https://doi.org/10.3390/app15105332.

11. Hiasat A. A reverse converter and sign detectors for an extended RNS fivemoduli set. *IEEE Trans. Circuits Syst. I Regul. Pap*. 2017; (64): 111–121. https://doi. org/10.1109/TCSI.2016.2612723.

12. Torabi Z., Jaberipur G. Low-Power/Cost RNS comparison via partitioning the dynamic range. *IEEE Trans. Very Large Scale Integr. VLSI Syst*. 2016; (24): 1849–1857. https://doi.org/10.1109/TVLSI.2015.2484618.

13. Mohan P.A., Phalguna P.S. Evaluation of mixed-radix digit computation techniques for the three moduli RNS $\{2^n-1, 2^n, 2^n+1\}$. *IEEE Trans. Circuits Syst. Express Briefs*. 2021; (68): 1418–1422. https://doi.org/10.1109/.

14. Bergerman M., Lyakhov P., Abdulsalyamova A. Modulo $2^k+1$ Truncated Multiply-Accumulate Unit. In: *International Conference on Actual Problems of Applied Mathematics and Computer Science. Cham: Springer Nature Switzerland*. 2022. P. 343-352. https://doi.org/10.1007/978-3-031-34127-4_33

15. Gupta T., Akhter S. Design and implementation of area-power efficient generic modular adder using flagged prefix addition approach. In: *2021 7th International Conference on Signal Processing and Communication (ICSC)*. 2021. P. 302-307. https://doi.org/10.1109/icsc53193.2021.9673363.

16. Efstathiou C., Vergos H.T., Nikolos D. Fast parallel-prefix modulo 2n+1 adders. *IEEE Trans. Comput.* 2004; (53): 1211–1216. https://doi.org/10.1109/TC.2004.60.

17. Bergerman M.V. Using a residual class system with modules of the form $\{2^n-1, 2^n, 2^n+1\}$ to reduce the hardware costs of a digital filter. *Izvestiya vysshikh uchebnykh zavedenii. Povolzhskii region. Tekhnicheskie nauki = News of higher educational institutions. Volga region. Technical sciences.* 2023; (1): 32-43. (In Russ.). https://doi.org/10.21685/2072-3059-2023-1-3.

18. Lyakhov P., Valueva M., Valuev G., Nagornov N. High-performance digital filtering on truncated multiply-accumulate units in the residue number system. *IEEE Access*. 2020; 8: 209181-209190. https://doi.org/10.1109/ACCESS.2020.3038496.

19. Rajanala A., Tyagi A. An area estimation technique for module generation. *Proceedings., 1990 IEEE International Conference on Computer Design: VLSI in Computers and Processors.* 1990. P. 459-462.

20. Belghadr A., Jaberipur G. Efficient variable-coefficient RNS-FIR filters with no restriction on the moduli set. *Signal, Image and Video Processing*. 2022; 16(6): 1443-1454. https://doi.org/10.1007/s11760-021-02097-9.

## Информация об авторе / Information about the Author

**Ляхов Павел Алексеевич**, кандидат физико-математических наук, заведующий кафедрой математического моделирования, Северо-Кавказский федеральный университет, г. Ставрополь, Российская Федерация, e-mail: ljahov@mail.ru, ORCID https://orcid.org/0000-0003-0487-4779

**Pavel A. Lyakhov**, Cand of Sci. (Physico-Mathematical), Head of the Mathematical Modeling Department, North Caucasus Federal University, Stavropol, Russian Federation, e-mail: ljahov@mail.ru, ORCID https://orcid.org/0000-0003-0487-4779