

**Коваленко Алексей Геннадьевич** – Общество с ограниченной ответственностью «НИЦ супер-ЭВМ и нейрокомпьютеров»; e-mail: k.a.g@bk.ru; 347928, г. Таганрог, пер. Итальянский, 106; тел.: 89281890413, 88634612111; к.т.н.; начальник сектора.

**Доронченко Юрий Иванович** – e-mail: doronchenko@superevm.ru; тел.: 88634648078, 88634612111; к.т.н.; технический директор.

**Kovalenko Alexey Gennadievich** – “Scientific Research Centre of Supercomputers and Neurocomputers” Co Ltd; e-mail: k.a.g@bk.ru; 106, Italyanskiy lane, Taganrog, 347928, Russia; phones: +79281890413, +78634612111; cand. of eng. sc.; head of sector.

**Doronchenko Yuri Ivanovich** – e-mail: doronchenko@superevm.ru; phone: +78634648078, +78634612111; cand. of eng. sc.; technical director.

УДК 004.272.34

DOI 10.18522/2311-3103-2016-11-1326

**И.П. Осинин**

### **МОДУЛЯРНО-ЛОГАРИФМИЧЕСКИЙ ПРОЦЕССОР\***

*Представлена организация перспективного арифметического модулярно-логарифмического процессора. Ключевое отличие от аналогов состоит в использовании непозиционной системы счисления на базе однородных вычислительных сред, что распараллелило вычисления вплоть до разрядов операнда, существенно повысив скорость вычислений, а также привнесло ряд уникальных свойств, значительно повысив надежность процесса счета. Применение системы остаточных классов совместно с логарифмической системой счисления вместо плавающей точки позволило использовать диапазон представления чисел аналогичный формату IEEE-754, отказавшись при этом от медленного округления и выравнивания порядков. Текущая реализация в виде IP-блока софт-процессора подтверждает реализуемость архитектурных особенностей, описанных в статье. В перспективе такой процессор может быть создан не только в качестве дополнения системы на кристалле, но и как самостоятельное устройство. Например, в виде арифметического ускорителя, подключаемого к компьютеру с традиционной архитектурой, либо в качестве центрального процессора на базе СБИС. В любом из этих вариантов высокая производительность и масштабируемость и реконфигурируемость достигается за счет применения однородных вычислительных сред, функционирующих в базе системы остаточных классов и логарифмической системы счисления. Сферами применения МЛП являются: высокопроизводительные вычисления, требующие работы с большой разрядностью данных – 1024 бит и более (задачи моделирования климата, исследование электромагнитного рассеяния, исследование орбитальной эволюции небесных тел и другие), где за счет распараллеливания счёта на множество независимых остатков достигается прирост производительности по сравнению с известными аналогами (например, для целых чисел минимум в два раза по сравнению с Intel AVX-512 при сокращении аппаратных затрат на 17 %); высоконадежные вычисления, требующие постоянного контроля правильности вычислений (задачи наведения ракет, управления атомной электростанцией, функционирования космических аппаратов и подобные), где за счёт применения корректирующих свойств СОК достигается уменьшение аппаратных затрат по сравнению с аналогами.*

*Процессор; система остаточных классов; логарифмическая система счисления; реконфигурируемая архитектура; высоконадежные вычисления.*

---

\* Исследование выполнено в ФГУП «Российский федеральный ядерный центр – Всероссийский научно-исследовательский институт экспериментальной физики».

I.P. Osinin

## MODULAR-LOGARITHMIC PROCESSOR

*The article describes the organization of perspective modular-logarithmic processor. The key difference from analogues is the use of non-positional number system based on homogeneous computing environments which parallelize computations until operand bits, increasing the speed of calculations, and also has introduced a number of unique properties that significantly improve the reliability of computations. Application of residual classes in conjunction with the logarithmic number system instead of the floating-point allowed us to use a range of the digit representation analogous to IEEE-754 format, while refusing the slow operation of rounding and smoothing orders. The current implementation in the form of soft-block IP processor confirms the feasibility of the architectural features which are described in the article. In the future, such a processor can be designed not only as a complement system on a chip, but also as an independent device. For example, as an arithmetic accelerator connected to the traditional architecture of the computer or as a CPU based on VLSI. In any of these embodiments, the high performance, scalability and reconfigurability are achieved by the use of homogeneous computing environments, operating in the basis of the residual class system and the logarithmic number system. The application areas for MLP are the following: high performance computing, requiring working with large digit data – 1024 bits or more (climate modeling tasks, the study of electromagnetic scattering study of the orbital evolution of celestial bodies and others), where due to paralleling the account on multiple independent residuals achieved growth performance compared to known analogs (such as integers at least twice compared with Intel AVX-512 while reducing the hardware cost by 17 %); highly reliable computing, requiring constant monitoring of correctness of calculations (missile guidance problem, the control of nuclear power plant operation of spacecraft and the like), which through the use of SOC correcting properties achieved is a decrease in hardware costs if to compare to the analogs.*

*Processor; RNS; LNS; reconfigurable architecture; highly reliable computing.*

**1. Введение.** Крупные задачи, критичные к скорости вычислений, возникают в самых разнообразных областях: в различных сферах промышленности, моделирования климата, в физике, космологии и многих других. Их решение с использованием средств вычислительной техники требует постоянного повышения скорости вычислений, что особенно актуально при создании систем рекордной производительности. На сегодняшний день большинство этих задач решается с использованием кластерных решений [1].

При этом известны такие традиционные пути повышения быстродействия, как совершенствование техпроцесса, например, создание трехмерных транзисторов, совершенствование алгоритмов выполнения программ, например, распараллеливание вычислений. Однако резерв повышения производительности, заложен в ускорении выполнения самих вычислительных операций, например, при применении модулярной арифметики на базе системы остаточных классов (СОК). Естественный параллелизм устройств, функционирующих на основе СОК, позволяет распараллелить процесс вычислений, как на программном, так и на аппаратном уровне, а модульность и однородность обеспечивает эффективное проектирование структур в сверхбольшом интегральном исполнении (СБИС). Научные работы Червякова Н.И., Акушко И.Я., Akiu Sasaki, Garner H., Omondi A. посвящены этому направлению [2–5].

При этом у СОК имеются и недостатки, среди которых медленное выполнение операций «округление» и «деление», что существенно затрудняет применение плавающей точки. Решение, описанное в данной работе, состоит в применении логарифмической системы счисления (ЛСС), в которой отсутствует операция «выравнивания порядков», а деление исходных чисел заменяется вычитанием их логарифмов. Основополагающий вклад внесли ученые Coleman J., Arnold M., Chester E., Lewis D. [6, 7].

С другой стороны, конвейерным параллелизмом обладают однородные вычислительные среды (ОВС), т.е. среды, аппаратура которых может реконфигурироваться, меняя свои функции, в зависимости от решаемых вычислительной системой задач. Это позволяет эффективно адаптировать архитектуру системы под структуру решаемой задачи, обеспечивая тем самым высокий уровень скорости вычислений. Данной тематике посвящен ряд работ отечественных и зарубежных ученых Варшавского В.И., Каляева И.А., Князькова В.С., Flynn M., Moore G., MacSorley L. [8, 9]. В общем случае ОВС представляет собой массив вычислительных ячеек структуры, которые объединены регулярными связями. Её применение в вычислительном ядре СБИС-процессора позволило практически пропорционально повышать производительность с увеличением числа ячеек в силу естественного параллелизма их работы.

Объединение в единой процессорной архитектуре рассмотренных форм естественного параллелизма приведено далее в статье.

**2. Организация архитектуры процессора.** Параллелизм на уровне разрядов чисел в случае использования позиционной системы счисления (ПСС) существенно ограничен фактом распространения межразрядных переносов, что негативно сказывается на скорости его работы. Данный недостаток отсутствует в системе остаточных классов (СОК), где вычисления по каждому модулю (основанию) из базиса  $\{p_1, p_2, \dots, p_n\}$  осуществляются независимо друг от друга.

**2.1. Арифметическое устройство.** В разработанном и запатентованном арифметическом устройстве [10] именно эта особенность стала первой плоскостью распараллеливания арифметических операций. Например, система из двух модулей ложится в два независимых вычислительных канала. В общем случае каналов может быть  $n$  – по количеству остатков СОК (рис. 1).

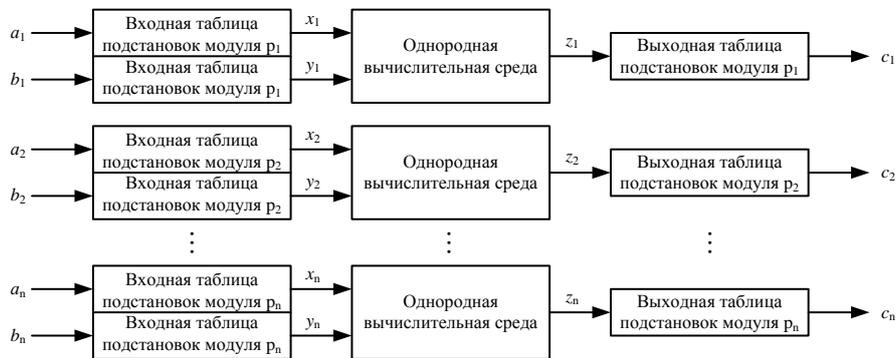


Рис. 1. Организация арифметического устройства

Важным преимуществом СОК является также то, что все основные арифметические операции могут выполняться также быстро как обычное суммирование. Для операций «умножение» и «деление нацело» это становится возможным при использовании входных и выходных таблиц подстановок (Look up table – LUT), осуществляющих однотактные преобразования операндов. Для  $i$ -ых остатков  $a_i$  и  $b_i$  первого и второго операнда соответственно подставляются дискретные логарифмы  $x_i = \left| \log_g a_i \right|_{p_i}$ ,  $y_i = \left| \log_g b_i \right|_{p_i}$ , затем вычисляется их сумма  $z_i = x_i + y_i$  и подставляется её антилогарифм  $c_i = \left| g^{z_i} \right|_{p_i}$ , где  $g$  – порождающий элемент [2] поля  $GF(p_i)$ ,  $p_i$  – модуль СОК,  $i \in [1, n]$ .

Так как сложение остатков является основополагающей операцией в разработанном арифметическом устройстве (к ней сводятся операции «вычитание», «умножение» и «деление нацело»), её выполнение распараллелено во второй плоскости с помощью ОВС – квадратной матрицы базовых элементов (БЭ) размерностью  $m+1$ , где  $m$  – разрядность модуля  $i$ -го вычислительного канала. В такой систолической структуре (рис. 2,а) базовые элементы (рис. 2,б) соединены регулярными связями, образуя вычислительный конвейер. На информационные входы  $v_j$  и  $w_j$  ОВС подаются соответствующие разряды двоичных  $m$ -разрядных остатков  $x_i$  и  $y_i$ , с информационных выходов  $u_j$  снимаются разряды искомой суммы этих остатков, где  $i \in [1, n]$ ,  $j \in [1, m]$ .

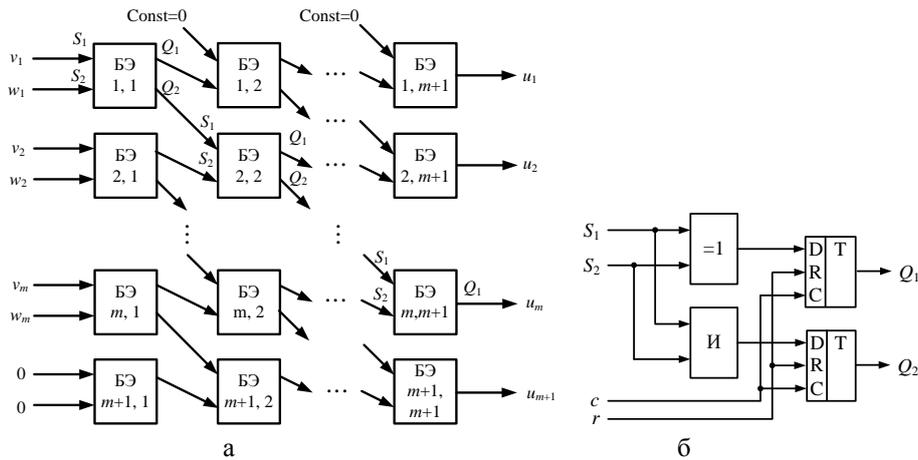


Рис. 2. Организация: (а) ОВС, (б) базового элемента

В состав БЭ входят: информационные входы  $S_1$  и  $S_2$ , вход синхронизация  $c$ , вход сброса  $r$ , информационные выходы  $Q_1$  и  $Q_2$ , логический элемент «исключающее ИЛИ» (=1), логический элемент «И» и два информационных D-триггера (на рис. 2,б обозначены «Т»).

Время заполнения вычислительного конвейера ОВС составляет  $T_{OBC} = t_{БЭ} \cdot (m + 1)$ , где  $m$  – разрядность двоичного модуля  $i$ -го вычислительного канала,  $t_{БЭ} = \max\{t_{XOR}, t_{AND}\} + t_{DFF}$  – время срабатывания БЭ, которое складывается из времени работы логического элемента и информационного DFF-триггера.

После заполнения конвейера время выполнения арифметической операции равно  $t_{БЭ}$ . Таким образом, первый уровень распараллеливания обеспечен независимым счетом по каждому модулю, второй уровень – конвейерным выполнением операций в ОВС. Ускорение в общем случае достигает  $k$  раз для операций сложение и вычитание и  $k^2$  раз для операций «умножение» и «деление нацело» по сравнению с последовательными алгоритмами ПСС, где  $k$  – разрядность позиционных операндов.

**2.2. Вычислительное ядро.** Рассмотренное арифметическое устройство является частью вычислительного ядра (рис. 3), где для снижения задержек доступа к памяти используется иерархическая структура памяти с многократным расслоением доступа для каждого модуля СОК. Помимо трехпортовой кэш-памяти (два порта – чтение, один – запись), имеются буферные очереди операндов и результата. Благодаря линейной структуре FIFO, они обладают высоким быстродействием, обеспечивая своевременную подкачку данных в арифметическое устройство.

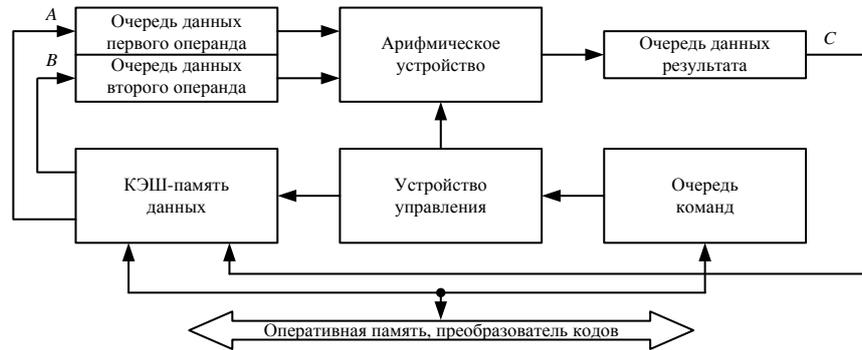


Рис. 3. Организация вычислительного ядра

В разработанном модулярно-логарифмическом MIMD-процессоре таких вычислительных ядер может быть несколько. Рассмотрим вариант, когда их четыре, как представлено на рис. 4. На исходные коды четырехядерного 32-разрядного процессора получено свидетельство о регистрации программы на языке программирования аппаратуры [11].

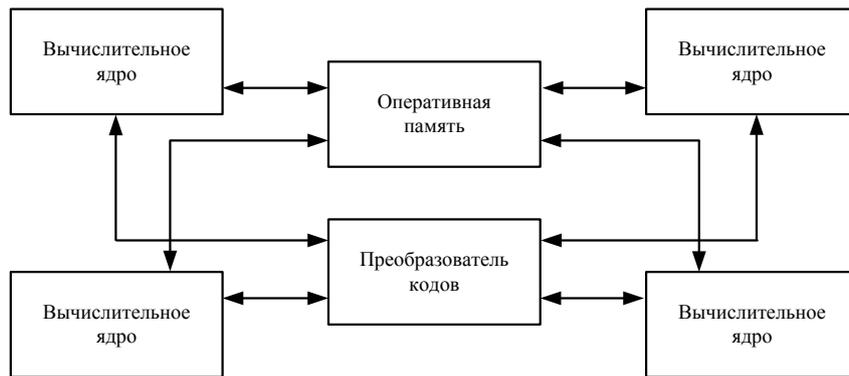


Рис. 4. Организация модулярно-логарифмического процессора

Ключевой особенностью процессора является динамическая реконфигурация числа ядер за счет уникальности набора оснований каждого вычислительного ядра. Ядра могут работать либо параллельно и независимо (32 бита), либо объединяться парами для удвоения точности (64 бита), либо объединяться в одно ядро для учетверения точности вычислений (128 бит). Это бывает необходимо для устранения ошибки переполнения разрядной сетки в процессе счета с помощью увеличения диапазона представления чисел.

Для преобразования из одной группы модулей в другую (при передаче данных для вычисления на другом ядре) применяется преобразователь кодов, рассмотренный далее.

**2.3. Преобразователь кодов.** Наиболее значительным фактором, сдерживающим широкое распространение СОК, является медленное выполнение операций преобразования из позиционной системы и обратно, вычисление позиционной характеристики числа и некоторых других. Поэтому особую важность имеет повышение скорости их выполнения. Рассмотрим, за счет чего в процессоре происходит их ускорение на примере операции преобразования кодов СОК в ПСС. Одним из методов преобразования кода системы остаточных классов в позиционный

код является способ преобразования через интервальную характеристику числа, позволяющую оценить величину числа без полного преобразования в ПСС [3]. Для этого используется выражение:

$$A = \left| \sum_{i=1}^n l_i \cdot a_i \right|_{p_i} \cdot p_{max} + a_{max},$$

где  $A$  – позиционное представление числа,  $p_{max}$  и  $a_{max}$  – наибольший модуль СОК и остаток по этому модулю соответственно,  $l_i = \frac{p_i^{\varphi(p_i)} - 1}{p_i}$ ,  $a_i$  –  $i$ -ый остаток числа в

СОК,  $n$  – количество модулей СОК,  $\varphi(p_i)$  – функция Эйлера [2] модуля  $p_i$ ,  $i \in [1;n]$ . Данный способ удобен тем, что позволяет вычислять интервальную характеристику числа. Она необходима при выполнении немодульных операций, требующих позиционного представления числа. Имея такую характеристику числа в СОК, преобразование в ПСС осуществляется за одну операцию «умножение с суммированием» (рис. 5). Как и коррекция по модулю, эта операция является частным случаем мультиоперандного суммирования, то есть суммирования нескольких слагаемых одновременно.



Рис. 5. Организация преобразователя кодов из СОК в ПСС

Разработанный мультиоперандный сумматор представляет собой систолическую структуру, функционирующую на базе арифметики разрядных срезов, которая также обладает естественным параллелизмом.

Одномерный массив слагаемых  $M$  представим приведенной ниже двумерной битовой матрицей  $V$  размерностью  $m$  строк и  $n$  столбцов, где  $m$  – число слагаемых,  $n$  – разрядность модулей чисел.

$$V = \begin{pmatrix} v_{1,n}, \dots, v_{1,2}, v_{1,1} \\ v_{2,n}, \dots, v_{2,2}, v_{2,1} \\ \dots \\ v_{m,n}, \dots, v_{m,2}, v_{m,1} \end{pmatrix}.$$

Можно выполнять обработку матрицы  $V$  не по строкам – скалярным значением слагаемых в двоичном представлении, как это реализуется в известных способах суммирования, а по столбцам – разрядным срезам  $SR_i$  исходных чисел,  $i \in [1;n]$ . Каждый разрядный срез является одномерным двоичным вектором из  $m$  элементов.

Специализированная ОВС (рис. 6,а) выполняет подсчет количества единичных бит в  $i$ -ом разрядном срезе, подаваемом на входы  $v_{ji}$  в параллельно-конвейерном режиме, после чего младший разряд подсчета становится разрядом искомой суммы (выход  $u$ ), а остальные являются разрядами переноса в следующий разрядный срез  $SR_{i+1}$ ,  $i \in [1;n]$ ,  $n$  – разрядность исходных чисел,  $j \in [1;m]$ ,  $m$  – коли-

чество слагаемых. Такая организация вычислений позволяет добиться повышения скорости вычислений в  $m$  раз (время мультиоперандного суммирования в ОВС составляет  $n$  тактов при  $n \gg m$ ) по сравнению с традиционным последовательным способом, где время мультиоперандного суммирования составляет  $n \cdot m$  тактов [12].

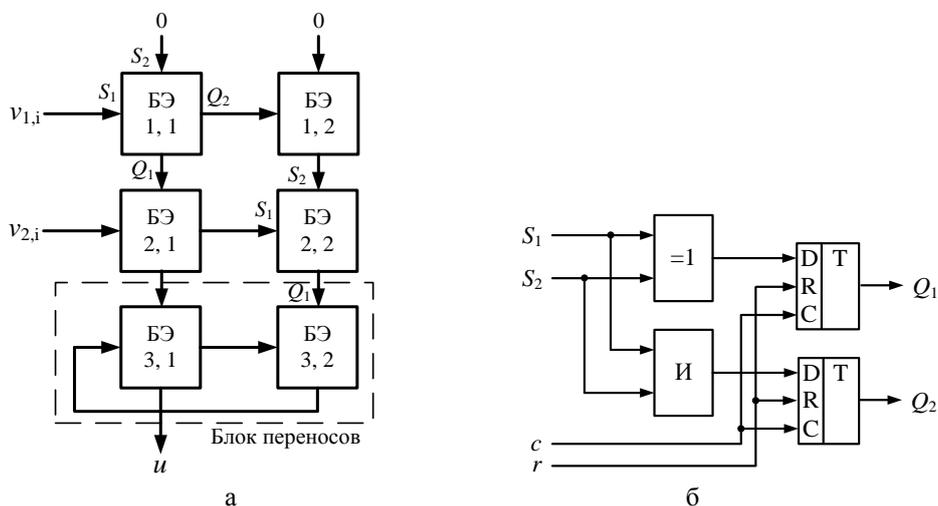


Рис. 6. Организация: (а) фрагмента ОВС для мультиоперандного суммирования, (б) базового элемента

**Пример.** Требуется вычислить сумму модулей четырех двоичных чисел:  $v_1=00111_2$ ,  $v_2=00101_2$ ,  $v_3=00001_2$ ,  $v_4=00111_2$ . Тогда:

$$V = \begin{pmatrix} 00111 \\ 00101 \\ 00001 \\ 00111 \end{pmatrix}.$$

Битовая матрица  $V$  массива слагаемых содержит пять разрядных срезов:  $SR_1=1111_2$ ,  $SR_2=1001_2$ ,  $SR_3=1101_2$ ,  $SR_4=0000_2$ ,  $SR_5=0000_2$ . На первом этапе выполняется параллельное вычисление количества единиц  $B_i$  в них:  $B_1=100_2$ ,  $B_2=010_2$ ,  $B_3=011_2$ ,  $B_4=000_2$ ,  $B_5=000_2$ . Младший разряд  $B_1$  является младшим разрядом  $s_1=0$  итоговой суммы чисел. Затем вычисляется значение  $B_2^*=(B_1 \rightarrow)+B_2=100_2$ , где  $\rightarrow$  обозначает операцию сдвига двоичного числа на один разряд вправо. Младший разряд полученной суммы  $B_2^*$  является вторым  $s_2=0$  разрядом искомой суммы  $S$ . В итоге серии аналогичных вычислений вычисляется искомая сумма  $S(s_5, s_4, s_3, s_2, s_1)=10100_2$ .

Важным моментом является использование тех же БЭ (рис. 8,б), что и в ОВС, входящей в состав арифметического устройства. Это позволяет значительно повысить регулярность элементов процессора, что, в конечном счете, обеспечивает высокую технологичность в случае массового производства на базе СБИС. Однако, помимо естественного параллелизма, СОК обладает свойствами, повышающими надежность процесса вычислений. Средства, которые их реализуют в разработанном процессоре, приведены в следующем разделе.

**3. Надежность вычислений.** Надежность является одним из основных свойств любого вычислительного устройства, во многом определяющим его пригодность для использования по назначению. На сегодняшний день в супер-ЭВМ

содержатся сотни тысяч и даже миллионы процессорных ядер. Сбои и отказы в таких системах приводят к тому, что алгоритм программы становится недетерминированным и в результате расчета можно получить разный результат при одинаковых наборах входных данных. Кроме того, существуют области вычислений, где ошибка в вычислениях может привести к фатальным последствиям. Например, к ним относятся задачи наведения ракет, управления атомной электростанцией, функционирования космических аппаратов и другие.

Существуют такие традиционные пути повышения надежности, как: резервирование на уровне компонент системы, дублирование вычислений, дополнительная обработка исключительных ситуаций и т.п.

Их недостаток состоит в том, что часть оборудования простаивает, либо дублирует работу, вплоть до отказа одного из компонентов системы. Данного недостатка можно избежать, внедрив средства повышения надежности на уровне архитектуры. Рассмотрим более подробно, как это реализовано в предлагаемом процессоре.

**3.1. Средства повышения надежности процессора.** Сочетание корректирующих свойств системы остаточных классов и динамической реконфигурации ОВС позволили реализовать следующие средства повышения надежности.

Во-первых, это выявление и исправление ошибок за счет расширения вычислительного диапазона путем назначения одного или нескольких вычислительных каналов в качестве контрольных (рис. 7). Доказано [2, 5], что при возникновении ошибки, результат операции окажется в запрещенном контрольном интервале. Причем, в процессоре применен механизм, локализирующий вычислительный канал, где произошел сбой. В случае подобного сбоя происходит исправление ошибки путем вычисления верного остатка результата и инкремент счетчика ошибок вычислительного канала.

	41	40	39	32	31	24	23	16	15	8	7	0
Группа модулей	Остаток по модулю $p_5$		Остаток по модулю $p_4$		Остаток по модулю $p_3$		Остаток по модулю $p_2$		Остаток по модулю $p_1$			
	Контрольный диапазон			Рабочий диапазон								

Рис. 7. Отображение областей чисел динамического диапазона

Во-вторых, при превышении заданного порога в счетчике ошибок возникает ситуация отказа и канал блокируется. Однако в отличие от повсеместно распространенных универсальных вычислительных устройств, модулярно-логарифмический процессор (МЛП) продолжает работу, сокращается лишь диапазон представления чисел. Если при этом нельзя снижать точность вычислений, то можно использовать вычислительные каналы другого ядра, в том числе и контрольные (рис. 8). Такая реконфигурация позволяет создать процессор, устойчивый к постепенной деградации. В пределе, вычисления будут продолжаться, пока хотя бы один вычислительный канал будет работоспособен. При этом разрядность обрабатываемых чисел будет равна разрядности канала.

В-третьих, рассмотренная в п.2.2 динамическая реконфигурация числа ядер позволяет управлять разрядностью операндов для устранения ошибок переполнения разрядной сетки. Затем, при необходимости, разрядность чисел может быть понижена до исходной.

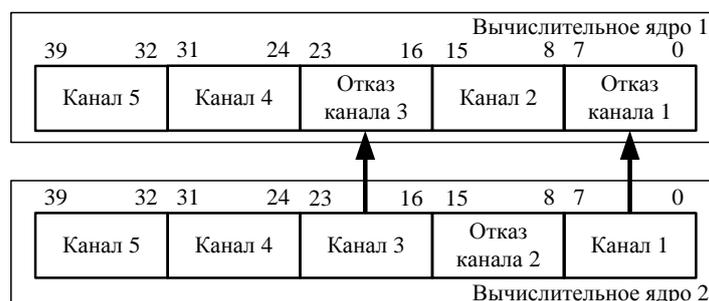


Рис. 8. Замена отказавших вычислительных каналов путем реконфигурации

В-четвертых, на базе предложенного процессора имеется возможность реализовать совершенно новые подходы в области аппаратного шифрования данных, например, можно настолько быстро менять параметры остатков, чтобы период действия ключа на базе текущих параметров был меньше минимального времени его вскрытия [4].

Кроме того, в процессе счета могут возникать невяные ошибки, связанные с округлением чисел в процессе вычислений по стандарту IEEE-754 [13]. К тому же работа с плавающей точкой связана с делением чисел, что негативно сказывается на скорости выполнения операций в СОК, так как данная операция является немодульной. Эти причины привели к необходимости применения аналога плавающей точки – логарифмической системы счисления (ЛСС), где в процессе вычислений отсутствуют фазы нормализации и денормализации, что избавляет от ошибок округления и необходимости деления на степень двойки.

**3.2. Оценка надежности модулярно-логарифмического процессора.** При построении надежных характеристик предположим, что:

- ◆ в начальный момент времени в каждом из 4 вычислительных ядер имеется  $n=k+r$  каналов, где  $k=4$  – количество информационных каналов,  $r=1$  – количество контрольных каналов;
- ◆ минимальное количество каналов, необходимых для функционирования процессора  $k$ ;
- ◆ отказ более чем  $3 \cdot k + 4 \cdot r$  каналов является отказом всего процессора;
- ◆ отказавшие каналы не восстанавливают работоспособность;
- ◆ отказы каналов являются статистически независимыми событиями.

Таким образом, надежная структура процессора соответствует модели скользящего резервирования, где резервные элементы находятся в нагруженном состоянии [2]. Тогда вероятности безотказной работы:

$$R_{МП} = \sum_0^{3k+4r} P^{4(k+r)-i} \cdot (1-P)^i,$$

где  $P(t) = e^{-\lambda t}$  – вероятность безотказной работы канала,  $\lambda$  – интенсивность его отказа.

Определение и оперативная коррекция ошибок в ПСС возможна лишь при условии одновременной работы нескольких устройств по принципу голосования, например, «2 из 3». Вероятность безотказной работы такой системы:

$$R_{ПСС} = 3 \cdot P_{ПСС}^2 - 2 \cdot P_{ПСС}^3,$$

где  $P_{ПСС}(t) = e^{-n \cdot \lambda_{ПСС} \cdot t}$ ,  $\lambda_{ПСС}$  – интенсивность отказа одного разряда,  $n$  – разрядность позиционного процессора.

Результаты расчета надежности в течении  $t=2000$  часов функционирования представлены графически на рис. 10 [14]. Анализ полученных зависимостей свидетельствует о преимуществе в надежности модулярно-логарифмического процессора. Например, при интенсивности отказов  $\lambda=2 \cdot 10^{-6}$  он обеспечивает вероятность безотказной работы  $R=0,5$ ; тогда как позиционный процессор  $R=0,1$ . Избыточность аппаратных затрат составляет 25 % и 66 % соответственно.

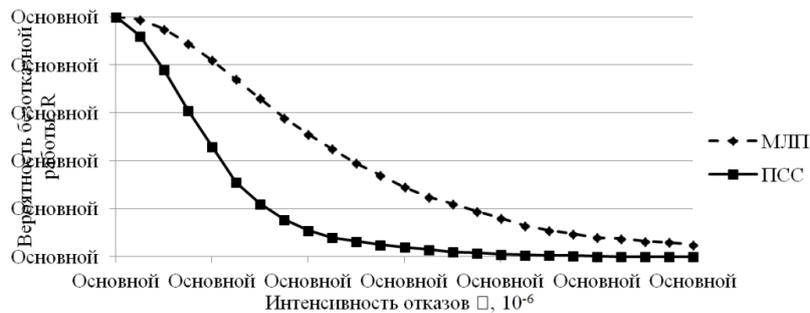


Рис. 9. Результаты расчета надежности процессора

**4. Прототип и оценка производительности процессора.** На данный момент модулярно-логарифмический процессор существует в виде прототипа – IP-блока RISC-процессора NIOS на базе программируемой логической интегральной схемы (ПЛИС). Данный выбор обусловлен простотой реализации, так как можно использовать уже готовые функциональные блоки процессора NIOS, дополняя его систему команд и используя уже готовый компилятор, для написания и отладки программ.

Проектирование функциональных схем процессора проводилось в САПР Quartus II Web Edition фирмы Altera. Отладка и тестирование выполнялось на базе платы с ПЛИС Altera Cyclone V.

Так как элементные базы ПЛИС и СБИС существенно отличаются по скоростным характеристикам, то при сравнительной оценке целесообразно использовать количество операций за такт работы устройства. Аппаратные затраты блока векторных расширений AVX процессора Xeon Phi [18] семейства Knight Landing оценены приблизительно (порядка 86 млн. транзисторов) по изображениям планировки функциональных блоков (floorplan) выпускаемого на данный момент процессора семейства процессоров Knights Corner.

Предложенный прототип выполняет до четырех операций за такт. Аппаратные затраты составляют 9,2 млн. транзисторов. При увеличении числа вычислительных ядер до 32 штук МЛП способен догнать по числу операций за такт блок векторного расширения Intel AVX-512 (16 команд за такт). Экстраполировав количество транзисторов, необходимое для создания 32-ядерного МЛП, получим 17 % выигрыш по аппаратным затратам (73,6 млн. против 86 млн. транзисторов) по сравнению с блоком AVX-512 [19].

Кроме того, сгруппировав ядра МЛП воедино, можно обеспечить выполнение одной 1024-разрядной операции за такт, тогда как в AVX-512 предусмотрены максимум 512-разрядные операции, выполняемые за такт. В этом случае на целочисленных операциях МЛП будет вдвое быстрее выполнять сложение (два такта AVX-512) и в 11 быстрее умножение (11 тактов AVX-512 с использованием алгоритма Карацубы [20]).

Помимо универсальных процессоров, у МЛП существуют специализированные аналоги. Например, модулярный нейропроцессор, разработанный в Ставропольском военном институте связи [2, 15]. Ключевое отличие МЛП от него состоит в применении ОВС вместо нейросетей, а также наличии блока обработки вещественных чисел на базе логарифмической системы счисления. Другой аналог – «European Logarithmic Microprocessor», разработанный в университете Ньюкасл [6, 16, 17] и функционирующий на базе ЛСС. Ключевое отличие МЛП состоит в применении СОК и ОВС для распараллеливания вычислений на уровне разрядов чисел.

**Заключение.** Таким образом, в представленном модулярно-логарифмическом процессоре заложен потенциал параллельного счета и обеспечения отказоустойчивости на уровне архитектуры, что делает актуальным его применение для расчета задач, критичных не только к скорости, но и к надежности вычислений.

Высокая производительность достигается за счет:

- ◆ распараллеливания арифметических операций на уровне остатков непозиционной системы счисления;
- ◆ параллельно-конвейерной обработки каждого модуля в однородной вычислительной среде;
- ◆ замены операций «умножение» и «деление» вещественных чисел операциями «сложение» и «вычитание» соответственно, благодаря свойствам логарифмов.

Надежность вычислений обеспечивается:

- ◆ обнаружением и коррекцией ошибок;
- ◆ устойчивостью к постепенной деградации оборудования;
- ◆ динамической реконфигурацией числа ядер для управления разрядностью вычислений;
- ◆ отсутствием ошибок округления за счет применения ЛСС.

Текущая реализация в виде IP-блока софт-процессора подтверждает реализуемость архитектурных особенностей, описанных в данной статье. В перспективе такой процессор может быть создан не только в качестве дополнения системы на кристалле, но и как самостоятельное устройство. Например, в виде арифметического ускорителя, подключаемого к компьютеру с традиционной архитектурой, либо в качестве центрального процессора на базе СБИС. В любом из этих вариантов высокая производительность, масштабируемость и реконфигурируемость достигается за счет применения однородных вычислительных сред, функционирующих в базе системы остаточных классов и логарифмической системы счисления.

Сферами применения МЛП являются:

- ◆ высокопроизводительные вычисления, требующие работы с большой разрядностью данных – 1024 бит и более (задачи моделирования климата, исследование электромагнитного рассеяния, исследование орбитальной эволюции небесных тел и другие), где за счет распараллеливания счёта на множество независимых остатков достигается прирост производительности по сравнению с известными аналогами (например, для целых чисел минимум в два раза по сравнению с Intel AVX-512 при сокращении аппаратных затрат на 17 %);
- ◆ высоконадёжные вычисления, требующие постоянного контроля правильности вычислений (задачи наведения ракет, управления атомной электростанцией, функционирования космических аппаратов и подобные), где за счёт применения корректирующих свойств СОК достигается уменьшение аппаратных затрат по сравнению с аналогами.

## БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. *Осинин И.П., Князьков В.С.* Направления развития архитектуры реконфигурируемых вычислительных платформ // Математическое моделирование развивающейся экономики, экологии и технологий (ЭКОМОД-2016): Труды IX Всероссийской научной конференции (Киров, 4-9 июля 2016 г.). – Киров: Изд-во ВятГУ, 2016. – С. 486-495.
2. *Червяков Н.И.* Модулярные параллельные вычислительные структуры нейропроцессорных систем. – М.: Изд-во Физматлит, 2003. – 288 с.
3. *Otondi A.* Residue Number System: Theory and Implementation. – London: Imperial College Press, 2007. – 312 p.
4. *Магомедов Ш.Г.* Передача и прием данных в вычислительных устройствах с использованием системы остаточных классов. – URL: <http://www.mathnet.ru/vagtu355> (дата обращения: 24.10.2015).
5. *Калмыков И.А.* Теоритические основы вычислений в полиномиальной системе классов вычетов, ориентированных на построение отказоустойчивых систем. – Ставрополь: Изд-во СКФУ, 2006. – 347 с.
6. *Coleman J.N., Chester E.I.* Arithmetic on the European Logarithmic Microprocessor // IEEE Transactions on Computers. – 2000. – Vol. 49, No. 7. – P. 702-715.
7. *Ismail R.C., Hussin R., Muzad S.A.* Interpolator Algorithms for approximating the LNS addition and subtraction // IEEE International Conference on Circuits and Systems (ICCS). Kaula Lumpur, 3-4 Oct. 2012. – P. 174-179.
8. *Каляев И.А.* Реконфигурируемые мультиконвейерные вычислительные структуры. – Ростов-на-Дону: Изд-во ЮНЦ РАН, 2008. – 320 с.
9. Патент 2477513 РФ. МПК G06F7/72. Ячейка однородной вычислительной среды, однородная вычислительная среда и устройство для конвейерных арифметических вычислений по заданному модулю / Осинин И.П., Князьков В.С. Оpubл. 10.03.2013г., Бюл. №7.
10. *Осинин И.П., Князьков В.С.* Организация параллельно-конвейерной СБИС-структуры с реконфигурируемой микроядерной архитектурой // Известия высших учебных заведений. Поволжский регион. Серия: Технические науки. – 2013. – № 3. – С. 74-83.
11. Патент 2491612 РФ. Способ организации вычислений суммы  $n$ -разрядных чисел / Осинин И.П., Князьков В.С. Оpubл. 27.08.2013г., Бюл. №24.
12. *Осинин И.П., Князьков В.С.* Концепция разрядно-параллельного арифметико-логического устройства на базе СБИС-структур // Супервычисления и математическое моделирование: Труды международной научной конференции (Саров, 1–5 октября 2012 г.). – Саров: РФЯЦ-ВНИИЭФ, 2013. – С. 449-458.
13. 754-2008 – IEEE Standard for floating-Point Arithmetic. Revision of ANSI/IEEE Std 754-1985. – Режим доступа: <http://www.ieeexplore.ieee.org>, 2008.
14. *Осинин И.П.* Высоконадежный модулярно-логарифмический процессор с реконфигурируемой архитектурой // Параллельные вычислительные технологии (ПаВТ'2016): Труды международной научной конференции (Архангельск, 28 марта – 1 апреля 2016 г.). – Челябинск: Изд-во ЮУрГУ, 2016. – С. 642-654.
15. *Оцокон Ш.А.* Структурная организация нейропроцессора использованием модели безошибочных вычислений // Нейрокомпьютеры: разработка и применение. – 2004. – № 12.
16. *Ismail R.C. and Coleman J.N.* ROM-less LNS // 20th IEEE Symposium on Computer Arithmetic (ARITH). – July 2011. – P. 43-51.
17. *Coleman J.N., Sofitley C.I., Kadlec J., Matousek R., Tichy M., Pohl Z., Hermanek A., and Benschop N.F.* The European Logarithmic Microprocessor // IEEE Transactions on Computers. – April 2008. – No. 57 (4). – P. 532-546.
18. Intel® Xeon Phi™ Coprocessor. – Режим доступа: [software.intel.com/sites/default/files/managed/ee/4e/intel-xeon-phi-coprocessor-quick-start-developers-guide.pdf](http://software.intel.com/sites/default/files/managed/ee/4e/intel-xeon-phi-coprocessor-quick-start-developers-guide.pdf).
19. Optimizing Performance with Intel® Advanced Vector Extensions. Режим доступа: [www.intel.com/content/dam/www/public/us/en/documents/white-papers/performance-xeon-e5-v3-advanced-vector-extensions-paper.pdf](http://www.intel.com/content/dam/www/public/us/en/documents/white-papers/performance-xeon-e5-v3-advanced-vector-extensions-paper.pdf).
20. Алгоритм Карацубы умножения чисел. – Режим доступа: [https://compscicenter.ru/media/course\\_class\\_attachments/02-01-multiplication.pdf](https://compscicenter.ru/media/course_class_attachments/02-01-multiplication.pdf).

REFERENCES

1. *Osinin I.P., Knyaz'kov V.S.* Napravleniya razvitiya arkhitektury rekonfiguriruemyykh vychislitel'nykh platform [Directions of development of architecture of reconfigurable computing platforms]. *Matematicheskoe modelirovanie razvivayushcheyssya ekonomiki, ekologii i tekhnologiy (EKOMOD-2016): Trudy IX Vserossiyskoy nauchnoy konferentsii (Kirov, 4-9 iyulya 2016 g.)* [Mathematical modeling of a developing economy, ecology and technology (EKOMOD-2016): Proceedings of the IX all-Russian scientific conference (Tyumen, July 4-9, 2016)]. Kirov: Izd-vo VyatGU, 2016, pp. 486-495.
2. *Chervyakov N.I.* Modulyarnye parallel'nye vychislitel'nye struktury neyroprotsessornykh system [Modular structure of the parallel computing systems neuroprocessor]. Moscow: Izd-vo Fizmatlit, 2003, 288 p.
3. *Omondi A.* Residue Number System: Theory and Implementation. London: Imperial College Press, 2007, 312 p.
4. *Magometov Sh.G.* Peredacha i priem dannykh v vychislitel'nykh ustroystvakh s ispol'zovaniem sistemy ostatochnykh klassov [Sending and receiving data in computing devices using residue number system]. Available at: <http://www.mathnet.ru/vagtu355> (Accessed 24 October 2015).
5. *Kalmykov I.A.* Teoriticheskie osnovy vychisleniy v polinomial'noy sisteme klassov vychetov, orientirovannykh na postroenie otkazoustoychivykh system [Theoretically bases of calculations in polynomial system of residue classes, focused on building a fault-tolerant systems]. Stavropol': Izd-vo SKFU, 2006, 347 p.
6. *Coleman J.N., Chester E.I.* Arithmetic on the European Logarithmic Microprocessor, *IEEE Transactions on Computers*, 2000, Vol. 49, No. 7, pp. 702-715.
7. *Ismail R.C., Hussin R., Muzad S.A.* Interpolator Algorithms for approximating the LNS addition and subtraction, *IEEE International Conference on Circuits and Systems (ICCAS), Kaula Lumpur, 3-4 Oct. 2012*, pp. 174-179.
8. *Kalyaev I.A.* Rekonfiguriruemye mul'tikonveyernye vychislitel'nye struktury [Reconfigurable computing multipipelined structure]. Rostov-on-Don: Izd-vo YuNTs RAN, 2008, 320 p.
9. *Osinin I.P., Knyaz'kov V.S.* Yacheyka odnorodnoy vychislitel'noy sredy, odnorodnaya vychislitel'naya sreda i ustroystvo dlya konveyernykh arifmeticheskikh vychisleniy po zadannomu modulyu [Cell homogeneous computing environment, homogeneous computing environment and a device for conveying arithmetic for a given module]. Patent 2477513 RF. MPK G06F7/72. Published 10.03.2013. Bull. No. 7.
10. *Osinin I.P., Knyaz'kov V.S.* Organizatsiya parallel'no-konveyernoy SBIS-struktury s rekonfiguriruemoy mikroyadernoy arkhitekturoy [The organization of parallel-pipelined VLSI structures with reconfigurable microkernel architecture], *Izvestiya vysshikh uchebnykh zavedeniy. Povolzhskiy region. Seriya: Tekhnicheskie nauki* [Proceedings of the higher educational institutions. Volga region. Series: Technical science], 2013, No. 3, pp. 74-83.
11. *Osinin I.P., Knyaz'kov V.S.* Sposob organizatsii vychisleniy summy n m-razryadnykh chisel [A method of computing the amount of the organization n m-bit numbers]. Patent 2491612 RF. Published 27.08.2013, Bull. No. 24.
12. *Osinin I.P., Knyaz'kov V.S.* Kontseptsiya razryadno-parallel'nogo arifmetiko-logicheskogo ustroystva na baze SBIS-struktur [Concept-bit parallel ALU based on VLSI structures], *Supervychisleniya i matematicheskoe modelirovanie: Trudy mezhdunarodnoy nauchnoy konferentsii (Sarov, 1-5 oktyabrya 2012 g.)* [Supercomputing and Mathematical Modeling: Proceedings of the International Scientific Conference (Sarov, Russia, October, 1-5, 2012)]. Sarov: RFYaTs-VNIIEF, 2013, pp. 449-458.
13. 754-2008 – IEEE Standard for floating-Point Arithmetic. Revision of ANSI/IEEE Std 754-1985. Available at: <http://www.ieeeexplore.ieee.org>, 2008.
14. *Osinin I.P.* Vysokonadezhnyy modulyarno-logarifmicheskyy protsessor s rekonfiguriruemoy arkhitekturoy [Osinin IP The highly-modular logarithmic processor with a reconfigurable architecture], *Parallel'nye vychislitel'nye tekhnologii (PaVT'2016): Trudy mezhdunarodnoy nauchnoy konferentsii (Arkhangel'sk, 28 marta – 1 aprelya 2016 g.)* [Parallel Computing Technologies (PaCT'2016): Proceedings of the International Scientific Conference (Arkhangelsk, 28 March - 1 April 2016)]. Chelyabinsk: Izd-vo YuUrGU, 2016, pp. 642-654.
15. *Otsokov Sh.A.* Strukturnaya organizatsiya neyroprotsessora ispol'zovaniem modeli bezoshibochnykh vychisleniy [The structural organization of the neural model using error-free computing], *Neyrokomp'yutery: razrabotka i primeneniye* [Neurocomputers: development and application], 2004, No. 12.

16. *Ismail R.C. and Coleman J.N.* ROM-less LNS, *20th IEEE Symposium on Computer Arithmetic (ARITH)*, July 2011, pp. 43-51.
17. *Coleman J.N., Sofiley C.I., Kadlec J., Matousek R., Tichy M., Pohl Z., Hermanek A., and Benschop N.F.* The European Logarithmic Microprocessor, *IEEE Transactions on Computers*, April 2008, No. 57 (4), pp. 532-546.
18. Intel® Xeon Phi™ Coprocessor. Available at: <http://www.software.intel.com/sites/default/files/managed/ee/4e/intel-xeon-phi-coprocessor-quick-start-developers-guide.pdf>.
19. Optimizing Performance with Intel® Advanced Vector Extensions. Available at: <http://www.intel.com/content/dam/www/public/us/en/documents/white-papers/performance-xeon-e5-v3-advanced-vector-extensions-paper.pdf>.
20. Algorithm Karatsuby umnozheniya chisel [The algorithm Karatsuba multiplication of numbers]. Available at: [https://compscicenter.ru/media/course\\_class\\_attachments/02-01-multiplication.pdf](https://compscicenter.ru/media/course_class_attachments/02-01-multiplication.pdf).

Статью рекомендовал к опубликованию д.т.н., профессор И.И. Левин.

**Осинин Илья Петрович** – ФГУП «Российский федеральный ядерный центр Всероссийский научно-исследовательский институт экспериментальной физики»; e-mail: [iposinin@mail.ru](mailto:iposinin@mail.ru); 607188, Нижегородская обл., г. Саров, пр-т Ленина, 4, тел.: +79091419811; аспирант.

**Osinin Ilya Petrovich** – Russian Federal Nuclear Center – All-Russian Scientific Research Institute of Experimental Physics; e-mail: [iposinin@mail.ru](mailto:iposinin@mail.ru); 4, Lenin street, Sarov, 607188, Russia; phone: +79091419811; postgraduate student.