УДК 621.376.57

Л.В. Пирская

О ВОЗМОЖНОСТИ ИСПОЛЬЗОВАНИЯ ДЕЛЬТА-ПРЕОБРАЗОВАНИЙ ПЕРВОГО ПОРЯДКА ДЛЯ ПОСТРОЕНИЯ СПЕЦИАЛИЗИРОВАННОГО ВЫЧИСЛИТЕЛЯ*

Целью работы является исследование возможности использования алгоритма параллельного решения системы линейных алгебраических уравнений на основе дельта-преобразования первого порядка с переменным квантом, адаптированного для эффективной реализации специализированного вычислителя и представляющего предпосылки для достижения высоких (по быстродействию и аппаратным ресурсам) показателей. Рассматривается структурная схема устройства, реализующая разработанный имеющий специфические особенности алгоритм. С ориентацией на ПЛИС получены оценки по аппаратным затратам и быстродействию. Данные оценки показывают, что при использовании алгоритма на основе дельта-преобразования первого порядка и переменного кванта для решения СЛАУ порядка $n=3$ *возможно сокращение аппаратных затрат в* ~ 2 *раз, а увеличение быстродействия при оперировании с 32-разрядными данными в* $\sim 2{,}5$ *раза в сравнении с методом простой итерации. При увеличении порядка СЛАУ* n *увеличиваются сравнительная эффективность использования аппаратных ресурсов и быстродействия предложенного алгоритма. В работе получена комплексная сравнительная оценка в виде произведения относительных аппаратных и по быстродействию оценок, демонстрирующая возможность повышения эффективности использования разработанного алгоритма при оперировании с 32-разрядными данными в* ~ 5 *раз по отношению к методу простой итерации, причем эта оценка с увеличением порядка системы* n *резко увеличивается.*

Итерационные методы; решение систем линейных алгебраических уравнений; дельта-преобразования первого порядка; специализированные вычислители; ПЛИС.

L.V. Pirskaya

ABOUT THE POSSIBILITY OF USING THE FIRST ORDER DELTA-TRANSFORMATIONS FOR CONSTRUCTION SPECIAL-PURPOSE COMPUTER

The aim of this paper is explored the possibility of using the algorithm of parallel solving a linear algebraic equations system based on the first order delta-transformations with variable quantum adapted for the effective implementation of special-purpose computer and represent the prerequisites for achieving high-level (in performance and hardware resources) characteristic. In this paper it is considered a block diagram of the device that implements engineered algorithm. It is obtained FPGA-oriented estimates from the quantity of hardware and performance. These estimates show that the quantity of hardware may be reduced in ~ 2 *times using the algorithm based on the first order delta-transformations with variable quantum, and increase the performance, when operating with a 32-bit data, in* $\sim 2{,}5$ *times versus using a simple iteration method. With an increase in the order of* n *linear algebraic equation increases the comparative effectiveness of the use of hardware resources and the performance of the proposed algorithm. In this paper it is obtained the integrated comparative estimate, including the hardware and performance estimates. It shows the possibility of increasing efficiency using the engineered algorithm, when operating with a 32-bit data, in* ~ 5 *times as compared to use of the simple iteration method and this estimate with the order of* n *increases rapidly.*

Iterative methods; systems of linear algebraic equations; delta transformation of the first order; special-purpose computer; FPGA.

* Работа выполнена в рамках выполнения базовой части государственного задания проект № 3442.

Введение. При проектировании систем, функционирующих в реальном времени, накладываются жесткие требования на скорость обработки информации; при этом требуется одновременное выполнение большого количества однотипных операций. Для ускорения выполнения этих операций разрабатываются специализированные вычислительные устройства, имеющие высокую производительность при параллельной реализации. Для решения задач такого рода чаще всего используют ПЛИС типа FPGA, характерные доступной ценой, возможностями программирования и перепрограммирования «на лету», реализации сложного функционала [1].

Архитектура FPGA обладает большой гибкостью и на ее базе можно осуществить распараллеливание операций, что позволяет существенно повысить производительность всей системы [2–4]. FPGA Xilinx последних поколений позволяют реализовывать более эффективные по сравнению с сигнальными процессорами алгоритмы ЦОС: существенно увеличить производительность позволяют параллельно работающие аппаратные узлы ПЛИС [5].

Задачи вычислительной математики, в частности, представляемые в виде систем линейных алгебраических уравнений (СЛАУ), часто являются отдельными компонентами задач, решаемых в рамках специализированных вычислительных средств. В связи с этим представляет интерес создание таких алгоритмических преобразований итерационных процессов решения СЛАУ, которые ориентированы на повышение качественных характеристик соответствующих вычислительных структур: сокращение необходимого оборудования и обеспечение количества итераций, приближенного к методу простой итерации.

Известны работы, в которых представлены методы решения СЛАУ, базирующиеся на дельта-преобразованиях первого порядка и ориентированные для построения специализированных вычислителей [6–11]. Однако данные методы обладают рядом недостатков. В работах отсутствует теоретическое обоснование организации итерационного процесса, что приводит к сужению возможностей реализации данного метода. Момент завершения итераций фиксируется с использованием некоторой константы (величины максимальной ошибки) или по количеству итераций, априорные определения значений которых проблематичны, а длительности итерационных процессов оказываются мало предсказуемыми и в значительной мере протяженными. В работах [6–11] возможности обеспечения высокой производительности ограничиваются принятой в технической литературе традиционной архитектурой вычислителя.

В работах [12–14] освещен метод организации итерационного процесса решения СЛАУ, в котором отсутствуют отмеченные выше недостатки. В [13–14] впервые излагаются основные теоретические положения решения вопроса минимизации количества итераций при использовании переменного кванта, обосновывается возможное расширение рекомендуемых соотношений между квантами соседних циклов. Предлагаются также позволяющие минимизировать количество итераций способы завершения итераций в каждом цикле, простые в реализации и не требующие априорной численной оценки. Теоретически обоснована и экспериментально показана возможность сокращения количества итераций по сравнению с использованием дельта-преобразований с постоянным квантом в сотни – тысячи раз при обеспечении одинаковой точности, а также обеспечения длительности итерационного процесса при переменном кванте, в значительной мере приближенной по количеству итераций к методу простой итерации. Учитывается также возможность введения в структуру вычислителя блоков памяти для хранения и однотактной выборки сумм произведений коэффициентов СЛАУ и дельта-признаков, что позволяет дополнительно существенно повысить производительность.

На конечную алгоритмическую эффективность по производительности и аппаратным затратам, возможностям реализации параллельных процессов значительное влияние должна оказывать архитектура специализированного вычислителя.

Целью данной работы является исследование возможности использования алгоритма итерационного решения СЛАУ для построения специализированного вычислителя.

Требуется представить структурную схему устройства, реализующую разработанный имеющий специфические особенности алгоритм, и получить относительные оценки по затратам оборудования и по быстродействию.

1. Алгоритм итерационного решения СЛАУ, адаптированный для специализированного вычислителя. Пусть исходная СЛАУ, содержащая матрицу постоянных коэффициентов с постоянными свободными членами, и удовлетворяющая условиям сходимости при реализации метода простой итерации имеет вид:

$$BY^{*}(t)=G. \tag{1}$$

Преобразуем запись системы (1) к форме с использованием итерационного метода и введением невязки $z(t)$:

$$z(t)=Y(t)-AY(t)-D. \tag{2}$$

В приведенных системах $B=[b_{rj}]$, $A=[\frac{b_{rj}}{b_{rr}}]$ - матрицы коэффициентов размерности $n\times n$; $G=[g_r]$, $D=[\frac{g_r}{b_{rr}}]$ – вектор-столбцы свободных членов системы; $Y^{*}(t)$ – вектор-столбец неизвестных системы; $z(t)$, $Y(t)$ – вектор-столбцы невязок и приближенных значений неизвестных; t – независимая переменная; $\det A\neq 0$.

Сущность методов решения СЛАУ на основе дельта-преобразований первого порядка и переменного кванта заключается в представлении итерационного процесса в виде l ($l=1,2,...,P_{\text{int}}$) итерационных циклов, в каждом из которых осуществляется формирование переменных при постоянных по модулю квантах преобразования c_l ($c_l>0$). В рамках данного метода P_{int} – количество итерационных циклов, выполняемых при постоянном по модулю значении кванта, R_{int} – постоянная величина, характеризующая изменение кванта преобразования и перемасштабирование всех переменных СЛАУ при переходе от цикла к циклу ($R_{\text{int}}=2^{k}, k\in[1;2]$; принимаются числа $R_{\text{int}}=2$ и $R_{\text{int}}=4$ (наиболее близкие к значению $R=e$) [13–14]).

Алгоритм приближенного решения СЛАУ с использованием дельта-преобразования первого порядка и переменного кванта [13-15] представим в следующей разностной форме для i-го шага при начальных условиях $Y_{r01}=0$, $z_{r01}=-D_r$, $\left|z_{01}\right|_{\max}$, $r=\overline{1,n}$, $c_p=2^{-s}$, $s\in N$:

- формирование значений невязок и неизвестных перед каждым итерационным циклом:

$$z_{r0l}=z_{rR^{*}_{\text{int}}(l-1)}\cdot R_{\text{int}}; \tag{3.1}$$

$$Y_{r0l}=Y_{rR^{*}_{\text{int}}(l-1)}\cdot R_{\text{int}}; \tag{3.2}$$

$$r=\overline{1,n},\ l=1,2,...,P_{\text{int}};$$

- формирование знаков квантов первых разностей переменных на каждой итерации в циклах:

$$\Delta_{ril} = -sign(z_{r(i-1)l}); \tag{3.3}$$

$$\Delta_{ril} \in \{+1,-1\};\ r = \overline{1,n},\ i = 1,2,...,R^*_{\text{int}};\ l = 1,2,...,P_{\text{int}};$$

демодуляция:

$$Y_{ril} = Y_{r(i-1)l} + \Delta_{ril}; \tag{3.4}$$

- ♦ формирование значений невязок на каждой итерации в циклах:

$$\nabla z_{ril} = \Phi_r(\Delta_{jil},\ j = \overline{1,n}) \tag{3.5}$$

$$z_{ril} = z_{r(i-1)l} + \nabla z_{ril}; \tag{3.6}$$

- ♦ условия формирования момента завершения итерационных процессов в итерационном цикле:

$$\left.\begin{array}{l} 1.\ sign(z_{rR^*_{\text{int}}l}) = -sign(z_{r(R^*_{\text{int}}-1)l}) \\ \text{или} \\ z_{rR^*_{\text{int}}l} = 0; \\ l = 1,2,\dots,P_{\text{int}},\ r = \overline{1,n}. \end{array}\right\} \tag{3.7}$$

$$\left.\begin{array}{l} 2.\ sign(z_{rR^*_{\text{int}}l} - sign(z_{rR^*_{\text{int}}l})) = -sign(z_{rR^*_{\text{int}}l}) \\ \text{или} \\ sign(z_{rR^*_{\text{int}}l} - sign(z_{rR^*_{\text{int}}l})) = 0; \\ l = 1,2,\dots,P_{\text{int}},\ r = \overline{1,n}. \end{array}\right\} \tag{3.8}$$

Значение постоянной величины P_{int} задается в соответствии с полученным в работе [13–14] соотношением:

$$P_{\text{int}} = \left\lceil \frac{\ln \frac{|z_{01}|_{\max}}{c_P}}{\ln R_{\text{int}}} \right\rceil, \tag{4}$$

где начальные значения невязки $|z_{01}|_{\max} = \max |z_{r01}|$, $r = \overline{1,n}$. и вес минимального кванта преобразования $c_p = 2^{-s}$, $s \in N$, связанный с необходимой точностью вычислений, задаются в качестве начальных условий алгоритма (3).

Количество итераций реального вычислительного процесса в цикле R^*_{int}, в соответствии с введенными в работах [13–14] заключениями, может быть большим или меньшим относительно значения R_{int}.

В алгоритме (3) соотношения (3.1) и (3.2) отражают введение в начале каждого цикла l перед выполнением соотношений (3.3)-(3.8) нового значения переменного кванта и иллюстрируют процедуру нормированного на текущем цикле формирования значений z_{r0l} и Y_{r0l} путем их сдвига на 1 разряд при $R_{\text{int}} = 2$ или на 2 разряда при $R_{\text{int}} = 4$.

Знаки квантов первых разностей переменных (3.3) на каждой итерации могут принимать значения ±1 и используются в дальнейшем по ходу выполнения алгоритма в формировании неизвестной переменной Y_{ril} (3.4) путем сложения или вычитания единицы от данного текущего значения. Указанная особенность нормирования переменных представляет возможности организации вычислительного процесса на основе целочисленного представления данных. На конечном этапе работы алгоритма возможно приведение значений переменных к исходному вещественному виду с учетом веса минимального кванта преобразования c_p:

$$Y^{*}_{rR^{*}_{\text{int}}P_{\text{int}}} = Y_{rR^{*}_{\text{int}}P_{\text{int}}} \cdot c_p. \tag{5}$$

Здесь сочетание $R^{*}_{\text{int}}P_{\text{int}}$ в индексе обозначает формирование самого последнего значения неизвестного, то есть конечного результата.

В представленном алгоритме (3) используются для формирования момента завершения итерационных процессов в циклах условия, требующие выполнение итераций в каждом цикле до тех пор, пока по всем уравнениям СЛАУ в цикле одновременно или распределено во времени по итерациям не выполнится хотя бы одно из соотношений (3.7) или (3.8) [13–14].

Разработанный на основе дельта-преобразований первого порядка и использовании переменного кванта алгоритм (3) решения СЛАУ, характеризующихся выполнением условий сходимости, позволяет существенно сократить количество итераций по сравнению с известными аналогичными методами, существенно приблизить итерационные процессы по длительности к простой итерации [13–14].

2. Особенности построения архитектуры специализированного вычислителя. На рис. 1 представлена структурная схема устройства, реализующего работу адаптированного для построения специализированного вычислителя алгоритма (3) итерационного решения системы линейных алгебраических уравнений с использованием дельта-преобразования первого порядка и переменного кванта.

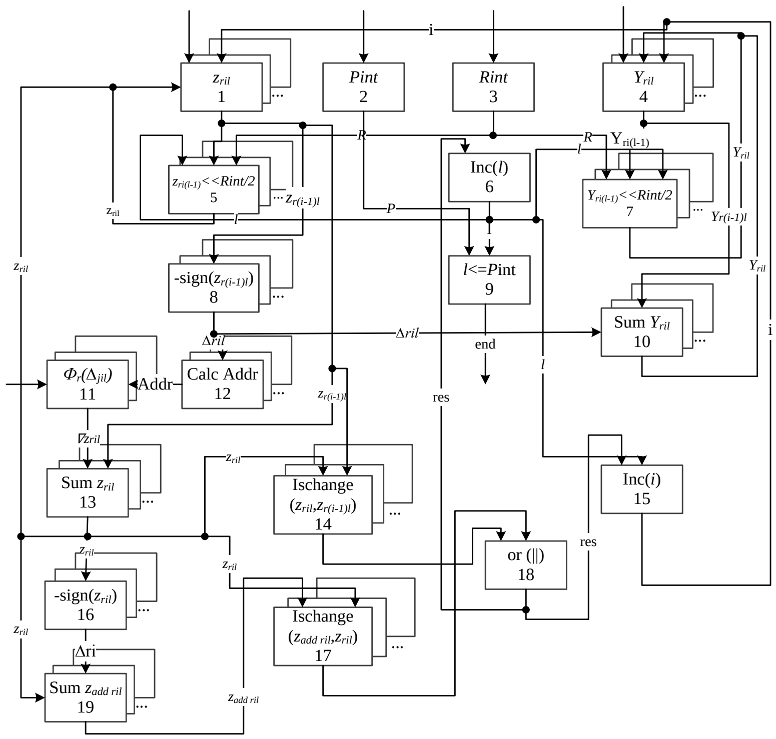

Рис. 1. Структурная схема устройства

Блоки 1, 4 включают *r* регистров ($r = \overline{1,n}$) для хранения невязок z_{ril}, неизвестных Y_{ril}, $r = \overline{1,n}$, которые на начальном этапе инициализируются значениями $z_{r0l} = -D_r$ и $Y_{r0l} = 0$ соответственно.

Блоки 2, 3 – регистры, содержащие значения количества итерационных циклов, выполняемых при постоянных по модулю значениях квантов, и одно из двух значений R_{int} соответственно. Данные значения определяются до начала работы алгоритма (3).

Блоки 5, 7 – r регистров сдвига ($r = \overline{1,n}$), реализующие сдвиг значений z_{r0l} (3.1) и Y_{r0l} (3.2) на 1 разряд при $R_{\text{int}} = 2$ или на 2 разряда при $R_{\text{int}} = 4$ в начале каждого цикла l. Полученные значения z_{r0l} и Y_{r0l} после сдвига поступают в регистры блоков 1 и 4, соответственно.

Блок 6 реализует работу циклов алгоритма (3), условия окончания работы алгоритма (3) в целом проверяется в блоке 9.

В блоках 8, 16 происходит формирование знаков квантов первых разностей $z_{r(i-1)l}$ предыдущей итерации и z_{ril} текущей итерации для всех уравнений системы, рассчитываемых в соответствии с соотношением (3.3).

Блок 10 представляет группу сумматоров ($r = \overline{1,n}$) для вычисления значений неизвестных Y_{ril} в соответствии с соотношением (3.4). На входы сумматоров поступают знаки квантов первых разностей переменных Δ_{ril} и значения неизвестных Y_{ril}. Результаты работы данных сумматоров поступают далее в блок 4.

В структурной схеме на рис.1 вычисление текущего значения невязки ∇z_{ril} реализуется на основе табличного метода в виде (3.5), где

$$\Phi_r(\Delta_{jil},\ j = \overline{1,n}) = \Delta_{ril} - \sum_{j=1}^{n} a_{rj}\Delta_{ji},$$

и запоминающих устройств ($r = \overline{1,n}$) блока памяти 11 для хранения таблиц. Таблица представляет совокупность сумм произведений коэффициентов СЛАУ на кванты преобразования для каждого уравнения. Выборка значений заранее сформированных сумм произведений выполняется по адресу, формируемому на основе совокупности текущих значений Δ_{jil}, $j = \overline{1,n}$ в блоке 12. Такая реализация соотношения (3.5) позволяет исключить операцию умножения и формировать результат за один такт.

При большой размерности матрицы *A* целесообразно разбиение суммы $\Phi_r(\Delta_{jil},\ j = \overline{1,n})$ на m блоков:

$$\Phi_r(\Delta_{jil},\ j = \overline{1,n}) = \sum_{g=1}^{m} \Phi_r^*(\Delta_{jil},\ j = \overline{1,n}),$$

и хранить табличные значения для каждого блока.

Блок 13 – r сумматоров ($r = \overline{1,n}$), предназначен для обеспечения выполнения соотношения (3.6). На вход каждого сумматора поступают соответствующие уравнениям системы знаки невязки z_{ri} и значения, полученные в блоке 11.

Блоки 14 и 17 предназначены для реализации проверки условий (3.7), (3.8), фиксирующих моменты завершения итерационных процессов в l -м итерационном цикле для каждого уравнения СЛАУ. В блоке 18 проверяется сформированность данных моментов для всех уравнений системы либо по (3.7), либо по (3.8). При успешном выполнении одного из условий, в блоке 6 увеличивается счетчик на единицу, и работа алгоритма организуется на новом итерационном цикле l .

Блок 15 – счетчик, с помощью которого выполняется подсчет количества итераций в рамках одного итерационного цикла *l*. При переходе на каждый следующий цикл l осуществляется инициализация счетчика i путем установки в нуль.

Блоки 16, 20 – это группы сумматоров ($r = \overline{1, n}$), обеспечивающих вычисление значений невязок текущих z_{ril} (3.6) и дополнительных $z_{add\ ril}$, где $z_{add\ ril} = z_{rR^*_{\text{int}}l} - sign(z_{rR^*_{\text{int}}l})$.

3. Исследование эффективности использования алгоритма для специализированного вычислителя. Комплексную эффективность реализации алгоритма дельта-преобразования первого порядка с переменным квантом по сравнению использованием метода простой итерации [16–18] на специализированном вычислителе можно рассматривать в виде взаимосвязанной совокупности сравнительных оценок по быстродействию и аппаратным затратам. Учитывая, как показано в работах [13–14], что количество итераций при использовании дельта-преобразований и переменного кванта, может быть большим или меньшим по сравнению с простой итерацией, при исследовании это количество принято одинаковым для обоих методов.

Можно рассматривать реализацию структурной схемы устройства, представленного на рис. 1, с использованием ПЛИС, в частности, типа FPGA. Ресурсные характеристики реализации основных арифметических операций при их аппаратном исполнении существенно неравнозначны. Особенно высокие аппаратные затраты, выражаемые в логических ячейках, требуются для множительных устройств. Операцию умножения будем рассматривать как выполнение умножения с использованием однотактных аппаратных схем параллельной реализации, а также при расширении разрядной сетки сомножителей – экономичные параллельно-последовательные реализации путем выполнения умножения за несколько тактов по алгоритму «сдвиг с накоплением».

Наиболее просто реализуется структура простого матричного суммирования, которая формирует параллельный умножитель как массив одноразрядных соединенных локальными меж соединениями сумматоров [19]. Данная схема наиболее эффективна на операндах небольшой разрядности (4 и менее), где параллельный умножитель 4×4 требует для своей реализации 12 сумматоров. При дальнейшем наращивании разрядности матрица одноразрядных сумматоров значительно разрастается, одновременно увеличивается критический путь распространения сигнала переноса, соответственно ограничивается быстродействие, и реализация умножителя становится малорациональной. Таким образом, в данной работе умножители большей разрядности рассматривались как комбинация умножителей 4×4 [19].

Формирование относительных оценок по аппаратным затратам базировалось на учете количества задействованных логических вентилей [20–21]. В соответствии с алгоритмом (3) и структурной схемой на рис. 1, решение уравнений выполняется параллельно. Также параллельно реализуются решения уравнений и по методу простой итерации.

Полученные оценки показали, что использование алгоритма, основанного на дельта-преобразованиях первого порядка и переменного кванта при решении СЛАУ порядка n=3, имеет преимущество по аппаратным затратам в ~ 2 раза по сравнению с методом простой итерации, а с увеличением порядка системы n эффективность увеличивается.

Оценка по быстродействию производилась в тактах. Временные затраты были рассчитаны в рамках одного прохода по циклу алгоритма, реализующего алгоритмическую последовать действий для всех уравнений системы. Полученная относительная оценка по быстродействию показала, что с увеличением разрядности данных увеличивается количество тактов на выполнение простой итерации. Количество тактов на выполнение алгоритма на основе дельта-преобразования первого порядка и переменного кванта остается постоянным при различной разрядности данных. Относительная оценка по быстродействию данного алгоритма при решении СЛАУ порядка n=3 по сравнению с методом простой итерации имеет преимущество по быстродействию в $\sim 2{,}5$ раза при оперировании с 32-разрядными данными, причем с увеличением порядка системы n имеет место увеличение данной эффективности.

Сравнительная комплексная оценка эффективности реализации алгоритма на основе дельта-преобразования первого порядка и переменного кванта была сформирована как произведение полученных выше относительных оценок по аппаратным затратам и быстродействию системы. В соответствии с данной оценкой разработанный алгоритм при решении СЛАУ порядка n=3 имеет преимущество при оперировании с 32-разрядными данными в ~ 5 раз по сравнению с использованием метода простой итерации, причем эта оценка с увеличением порядка системы n резко увеличивается.

Заключение. Исследована возможность использования алгоритма параллельного решения СЛАУ с переменным квантом на основе дельта-преобразования первого порядка [13–14], адаптированного для эффективной реализации специализированного вычислители и представляющего возможным для достижения высоких (по быстродействию и аппаратным ресурсам) показателей в сравнении с методом простой итерации [16–18]. Также в работе представлены особенности построения архитектуры специализированного вычислителя. Получены оценки по аппаратным затратам, быстродействию и комплексная сравнительная оценка, показывающие возможность эффективного использования алгоритма для специализированного вычислителя. Данные обстоятельства создают предпосылки, связанные, в частности, с расширением ресурсных возможностей ПЛИС для одновременной реализации в реальном времени сложных задач, в качестве отдельных компонент которых используются СЛАУ.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. *Максфилд К.* Проектирование на ПЛИС. – М.: Додека-XXI, 2007. – 408 с.
2. *Корчинский А.П., Бурцева Н.В.* Применение программируемых логических интегральных схем в электронной аппаратуре // Електроніка та системи управління. – 2009. – № 4 (22). – С. 5-13.
3. *Yang H., Ziavras S.G.* FPGA-based vector processor for algebraic equation solvers // IEEE International SOC Conference. – 2005. – P. 115-116.
4. *Zhang W., Betz V., Rose J.* Portable and Scalable FPGA-Based Acceleration of a Direct Linear System Solver // ACM Transactions on Reconfigurable Technology and Systems (TRETS). – 2012. – Vol. 5, № 1. – Article 6.
5. *Тарасов И.* ПЛИС Xilinx и цифровая обработка сигналов. Особенности, преимущества, перспективы // Электроника НТБ. – 2011. – № 3. – С. 70-74.
6. *Малиновский Б.Н., Боюн В.П., Козлов Л.Г.* Алгоритмы решения систем линейных алгебраических уравнений, ориентированные на структурную реализацию // Управляющие системы и машины. – 1977. – № 5. – С. 79-84.

7. *Третьяков С.И.* Алгоритмы работы специализированных процессоров для решения систем уравнений // Кибернетика. – 1978. – № 5. – С. 34-36.
8. *Байков В.Д., Сергеев М.Б.* Структурно-ориентированный алгоритм решения систем линейных алгебраических уравнений // Управляющие системы и машины. – 1986. – № 2. – С. 81-84.
9. *Гомозов О.В. Ладыженский Ю.В.* Инкрементные алгоритмы решения систем линейных алгебраических уравнений и архитектура мультипроцессоров на программируемой логике // Научные труды ДонНТУ. Серия «Информатика, кибернетика и вычислительная техника». – 2010. – № 12 (165). – С. 34-40.
10. *Кравченко П.П.* Инкрементные методы решения систем линейных алгебраических уравнений // Многопроцессорные вычислительные структуры. –1983. – № 5 (XIV). – С. 30-32.
11. *Kravchenko P.P.* Solving systems of algebraic and differential equations by second-order difference modulation // Cybernetics. – 1989. – Vol. 25(2). – P. 218-229.
12. *Pirskaya L.V.* The iterative algorithm for solving systems of linear algebraic equations without the multibit multiplication operation // International Conference «Engineering & Telecommunication En&T 2014». Book of Abstracts. – Dolgoprudny: MIPT, 2014. – P. 210-212.
13. *Кравченко П.П., Пирская Л.В.* Итерационный метод решения систем линейных алгебраических уравнений, исключающий операцию многоразрядного умножения // Известия ЮФУ. Технические науки. – 2014. – № 7 (156). – С. 214-224.
14. *Kravchenko P.P., Pirskaya L.V.* The method of organizing the iterative process of the system of the linear algebraic equations solution excluding the multidigit multiplication operation // Biosciences Biotechnology Research Asia. – 2014. – Vol. 11, № 3. – P. 1831-1839.
15. *Кравченко П.П.* Основы теории оптимизированных дельта-преобразований второго порядка. Цифровое управление, сжатие, параллельная обработка информации: Монография. – Таганрог: Изд-во ТТИ ЮФУ, 2008. – 192 с.
16. *Бахвалов Н.С., Жидков Н.П., Кобельков Г.М.* Численные методы. – М.: БИНОМ. Лаборатория знаний, 2006. – 632 с.
17. *Greenbaum A.* Iterative Methods for Solving Linear Systems. – SIAM, 1997. – 220 p.
18. *Vuik C.* Iterative solution methods. – The Netherlands: Delft Institute of Applied Mathematics, Delft, 2012. – 118 p.
19. *Стешенко В.* Занятие 6. Реализация вычислительных устройств на ПЛИС // Компоненты и технологии. – 2000. – № 8. – С. 88-91.
20. *Максимов Н.В., Партыка Т.Л., Попов И.И.* Архитектура ЭВМ и вычислительных систем: Учебник. – 3-е изд. – М.: Форум, 2010. – 512 с.
21. *Таненбаум Э., Остин Т.* Архитектура компьютера. – 6-е изд. – СПб.: Питер, 2013. – 816 с.

REFERENCES

1. *Maksfild K.* Proektirovanie na PLIS [FPGA design]. Moscow: Dodeka-XXI, 2007, 408 p.
2. *Korchinskiy A.P., Burtseva N.V.* Primenenie programmiruemykh logicheskikh integral'nykh skhem v elektronnoy apparature [The use of a programmable logic integrated circuits in electronic equipment], *Elektronika ta sistemi upravlinnya* [Electronics and control systems], 2009, No. 4 (22), pp. 5-13.
3. *Yang H., Ziavras S.G.* FPGA-based vector processor for algebraic equation solvers, *IEEE International SOC Conference*, 2005, pp. 115-116.
4. *Zhang W., Betz V., Rose J.* Portable and Scalable FPGA-Based Acceleration of a Direct Linear System Solver, *ACM Transactions on Reconfigurable Technology and Systems (TRETS)*, 2012, Vol. 5, No. 1, Article 6.
5. *Tarasov I.* PLIS Xilinx i tsifrovaya obrabotka signalov. Osobennosti, preimushchestva, perspektivy [The Xilinx FPGA and digital signal processing. Features, benefits, prospects], *Elektronika NTB* [Electronika: Science, Technology, Business], 2011, No. 3, pp. 70-74.
6. *Malinovskiy B.N., Boyun V.P., Kozlov L.G.* Algoritmy resheniya sistem lineynykh algebraicheskikh uravneniy, orientirovannye na strukturnuyu realizatsiyu [Algorithms for solving systems of linear algebraic equations-oriented structural implementation], *Upravlyayushchie sistemy i mashiny* [Control systems and machines], 1977, No. 5, pp. 79-84.

7. *Tret'yakov S.I.* Algoritmy raboty spetsializirovannykh protsessorov dlya resheniya sistem uravneniy [Algorithms specialized processors for solving systems of equations], *Kibernetika* [Cybernetics], 1978, No. 5, pp. 34-36.
8. *Baykov V.D., Sergeev M.B.* Strukturno-orientirovannyy algoritm resheniya sistem lineynykh algebraicheskikh uravneniy [Structurally-oriented algorithm for solving systems of linear algebraic equations], *Upravlyayushchie sistemy i mashiny* [Control systems and machines], 1986, No. 2, pp. 81-84.
9. *Gomozov O.V. Ladyzhenskiy Yu.V.* Inkrementnye algoritmy resheniya sistem lineynykh algebraicheskikh uravneniy i arkhitektura mul'tiprotsessorov na programmiruemoy logike [Incremental algorithms for solving systems of linear algebraic equations and the architecture of multiprocessors on programmable logic], *Nauchnye trudy DonNTU. Seriya «Informatika, kibernetika i vychislitel'naya tekhnika»* [Scientific works of Donetsk national technical University. Series "Informatics, Cybernetics and computer engineering"], 2010, No. 12 (165), pp. 34-40.
10. *Kravchenko P.P.* Inkrementnye metody resheniya sistem lineynykh algebraicheskikh uravneniy [Incremental methods for solving systems of linear algebraic equations] *Mnogoprotsessornye vychislitel'nye struktury* [Multiprocessor computing patterns], 1983, No. 5 (XIV), pp. 30-32.
11. *Kravchenko P.P.* Solving systems of algebraic and differential equations by second-order difference modulation, *Cybernetics*, 1989, Vol. 25 (2), pp. 218-229.
12. *Pirskaya L.V.* The iterative algorithm for solving systems of linear algebraic equations without the multibit multiplication operation, *International Conference «Engineering & Telecommunication En&T 2014». Book of Abstracts*. Dolgoprudny: MIPT, 2014, pp. 210-212.
13. *Kravchenko P.P., Pirskaya L.V.* Iteratsionnyy metod resheniya sistem lineynykh algebraicheskikh uravneniy, isklyuchayushchiy operatsiyu mnogorazryadnogo umnozheniya [The iterative method for solving systems of linear algebraic equations, exclusive the multi-bit multiplication operation], *Izvestiya YuFU. Tekhnicheskie nauki* [Izvestiya SFedU. Engineering Sciences], 2014, No. 7 (156), pp. 214-224.
14. *Kravchenko P.P., Pirskaya L.V.* The method of organizing the iterative process of the system of the linear algebraic equations solution excluding the multidigit multiplication operation, *Biosciences Biotechnology Research Asia*, 2014, Vol. 11, No. 3, pp. 1831-1839.
15. *Kravchenko P.P.* Osnovy teorii optimizirovannykh del'ta-preobrazovaniy vtorogo poryadka. Tsifrovoe upravlenie, szhatie, parallel'naya obrabotka informatsii: Monografiya [Fundamentals of the theory of optimized Delta-transformations of the second order. Digital control, compression, parallel information processing: Monograph]. Taganrog: Izd-vo TTI YuFU, 2008, 192 p.
16. *Bakhvalov N.S., Zhidkov N.P., Kobel'kov G.M.* Chislennye metody [Numerical methods], Moscow: BINOM. Laboratoriya znaniy, 2006, 632 p.
17. *Greenbaum A.* Iterative Methods for Solving Linear Systems. SIAM, 1997, 220 p.
18. *Vuik C.* Iterative solution methods. The Netherlands: Delft Institute of Applied Mathematics, Delft, 2012, 118 p.
19. *Steshenko V.* Zanyatie 6. Realizatsiya vychislitel'nykh ustroystv na PLIS [Implementation of computing devices on the FPGA], *Komponenty i tekhnologii* [Components & Technologies], 2000, No. 8, pp. 88-91.
20. *Maksimov N.V., Partyka T.L., Popov I.I.* Arkhitektura EVM i vychislitel'nykh sistem: uchebnik [Computer architecture and computer systems: a Tutorial], 3 rd ed. Moscow: Forum, 2010, 512 p.
21. *Tanenbaum E., Ostin T.* Arkhitektura komp'yutera [The architecture of the computer], 6th ed. St. Petersburg: Piter, 2013, 816 p.

Статью рекомендовал к опубликованию д.т.н., профессор Я.Е. Ромм.

Пирская Любовь Владимировна – Южный федеральный университет; e-mail: lyubov.pirskaya@gmail.com; 347928, г. Таганрог, пер. Некрасовский, 44; тел.: +79515367641; кафедра математического обеспечения и применения ЭВМ; аспирант.

Pirskaya Lyubobov' Bladimirovna – Southern Federal University; e-mail: lyubov.pirskaya@gmail.com; 44, Nekrasovskiy, Taganrog, 347928, Russia; phone: +79515367641; the department of software engineering; postgraduate student.