

РЕКОНФИГУРИРУЕМАЯ ВЫЧИСЛИТЕЛЬНАЯ МОДУЛЬНАЯ СИСТЕМА

Шевелев С. С. – канд. техн. наук, доцент кафедры информационной безопасности, Юго-Западный государственный университет, Курск, Россия.

АННОТАЦИЯ

Актуальность. Современные универсальные вычислительные машины способны реализовать любой алгоритм, но при решении определенных задач по скорости обработки не могут конкурировать со специализированными вычислительными модулями. Специализированные устройства имеют высокое быстродействие, эффективно решают задачи по обработке массивов, задач искусственного интеллекта, используются в качестве управляющих устройств. Применение специализированных микропроцессорных модулей, которые реализуют обработку строк символов, логических и числовых значений, представленных в виде целых и действительных чисел, позволяет повысить быстродействие выполнения арифметических операций за счет использования параллелизма при обработке данных.

Цель. Разработать принципы построения микропроцессорных модулей для вычислительной модульной системы с перестраиваемой структурой, арифметико-символьного процессора, специализированных вычислительных устройств, коммутационных систем способных конфигурировать микропроцессоры и специализированные вычислительные модули в мультиконвейерную структуру для повышения быстродействия выполнения арифметических и логических операций, высокоскоростных алгоритмов проектирования специализированных процессоров-акселераторов символьной обработки. Разработать алгоритмы, структурные и функциональные схемы специализированных математических модулей, выполняющих арифметические операции в прямых кодах на нейроподобных элементах и систем децентрализованного управления работой блоков.

Метод. Построен информационный граф вычислительного процесса модульной системы с перестраиваемой структурой. Разработаны структурные и функциональные схемы, алгоритмы, реализующие построение специализированных модулей для выполнения арифметических и логических операций, поисковых операций и функций замены вхождений в обрабатываемых словах. Разработано программное обеспечение моделирования работы арифметико-символьного процессора, специализированных вычислительных модулей, систем коммутаций.

Результаты. Разработана структурная схема реконфигурируемой вычислительной модульной системы, которая состоит из совместимых функциональных модулей, она способна к статической и динамической реконфигурации, имеет параллельную структуру соединения процессора и вычислительных модулей за счет использования интерфейсных каналов. Система состоит из арифметико-символьного процессора, специализированных вычислительных модулей и систем коммутаций, выполняет специфические задачи символьной обработки информации, арифметические и логические операции.

Выводы. Архитектура реконфигурируемых вычислительных систем может динамически изменяться в процессе их функционирования. Появляется возможность адаптации архитектуры вычислительной системы под структуру решаемой задачи, создавать проблемно-ориентированные вычислители, структура которых соответствует структуре решаемой задачи. В качестве основного вычислительного элемента в реконфигурируемых вычислительных системах используются не универсальные микропроцессоры, а программируемые логические интегральные схемы, которые объединены с помощью высокоскоростных интерфейсов в единое вычислительное поле. Реконфигурируемые мультиконвейерные вычислительные системы на основе полей являются эффективным средством для решения потоковых задач обработки информации и управления.

КЛЮЧЕВЫЕ СЛОВА: информационный граф, параллельно-конвейерная структура, модульная система, специализированные вычислительные блоки.

АББРЕВИАТУРЫ

БВАО – блок выполнения арифметических операций;

БППЗВОС – блок параллельного поиска и замены в обрабатываемых словах;

БВЛО – блок выполнения логических операций;

БХР – блок хранения результатов;

КСБУ – коммутационные схемы блока управления;

СКПМ система коммутаций процессор-модуль;

СКММ система коммутаций модуль-модуль;

СПЕЦ МОД1 – первый специализированный модуль;

СПЕЦ МОД2 – второй специализированный модуль;

СПЕЦ МОД3 – третий специализированный модуль;

СПЕЦ МОД4 – четвертый специализированный модуль;

СПЕЦ МОД5 – пятый специализированный модуль;

СКММ – система коммутаций модуль-модуль;

БВД – блок ввода данных;

СМК – система электронных ключей;

БХРП – блок хранения результатов подстановок;

ЭЛВ – элементарные вычислители логических операций;

БПВС – блок памяти вхождений и обрабатываемых слов;

БАВС – блок сравнения и анализа вхождений в обрабатываемых словах;

БЗАМ – блок замены вхождений в обрабатываемых словах;

БРГОС – блок регистра обрабатываемого слова;

БРГВХ – блок регистра вхождений;
БРГЗМ – блок регистра замены;
КОМ – компаратор;
БРГРЗ – блок регистр результата замены;
ЭлКл – схема электронных ключей;
БПВ – блок поиска вхождений;
БАЗУ – блок ассоциативных запоминающих устройств;
БОРС – блок обработки результатов сравнения;
БХУС – блок хранения упорядоченных слов;
MR – младшие разряды двоичных чисел;
CR – старшие двоичных разряды чисел;
ДС – схема дешифратора;
ЛЦл – локализованные центры системы распределения ресурсов;
БВД – блок ввода данных;
БСПЧ – блок сортировки положительных чисел;
БСОЧ – блок сортировки отрицательных чисел;
БВАО – блок выполнения арифметических операций;
БППЗВОС – блока параллельного поиска и замены вхождений в обрабатываемых словах;
БВЛО – блок выполнения логических операций;
АСП – арифметико-символьный процессор;
БС – блок сортировки;
БДШ – блок дешифрации;
ЭКл – электронные ключи;
ШФ – шинный формирователь.

НОМЕНКЛАТУРА

S – множество вершин информационного графа;
 S_i – вершина информационного графа;
 O_i – допустимые операции, приписанные к вершинам графа;
 F – множество дуг графа;
 L – множество логических микроопераций;
 Z – количество входных векторов;
 N – количество векторов входных данных;
 $t_{БВД}$ – время обработки блоком ввода данных;
 $t_{Сик}$ – время обработки данных системой электронных ключей;
 $t_{ЭлВ}$ – время обработки данных элементарными вычислителями логических операций;
 $t_{БХР}$ – время обработки блоком хранения результатов;
 M – число вершин на графе $G(S,F)$;
 $S(O_i)$ – число тактов работы блока устройства;
 τ – продолжительность такта;
 $T_{Пос}$ – время результата последовательной обработки;
 $T_{Пар}$ – время результата параллельной обработки;
 $T_{Кон}$ – время результата конвейерной обработки;
 $V_{Пос}$ – время результата последовательной обработки;
 $V_{Кон}$ – скорость конвейерной обработки;

$V_{Пар}$ – скорость параллельной обработки;
 K – количество процессоров;
 P_i – число элементарных процессоров в устройстве;
 $T_{ТАВС}$ – самый максимальный по времени этап обработки данных;
 R – множество арифметических микроопераций;
 C – множество микроопераций символьной обработки информации;
 β – символы вхождения;
 α – символы обрабатываемого слова;
 s – позиция вхождения в обрабатываемом слове;
 m – число сдвигов вправо в слове;
 x – множество символов вхождения для поиска в обрабатываемом слове;
 w – множество символов обрабатываемого слова при параллельном поиске;
 a_i, b_i, g_i – двоичные переменные;
 T_{max} – длительность такта самого медленного этапа обработки;
 $T_{Кон}$ – время при конвейерной обработке;
 $t_{БОФЗ}$ – время работы блока операций в формате с фиксированной запятой;
 $t_{БОПРПЗ}$ – время работы блока операций порядков в формате с плавающей запятой;
 $t_{БОМПЗ}$ – время работы блока операций мантисс в формате с плавающей запятой;
 H – число процессоров в конвейерной организации вычислительного процесса;
 y – основание системы счисления;
 q – количество чисел в методе сортировки выбором;
 u – количество букв в регистре вхождения;
 A, B, P, Σ – двоичные разряды чисел;
 $\Sigma_0, \Sigma_1, \Sigma_{10}, \Sigma_{11}$ – блоки суммирования.

ВВЕДЕНИЕ

Идея концепции построения реконфигурируемых вычислительных систем заключается в аппаратной реализации всех операций, предписанных вершинами информационного графа задачи, всех каналов передачи данных между вершинами, соответствующими дугам графа, и всех информационных каналов, соответствующих входным и выходным вершинам. Задача, определенная информационным графом, будет выполнена максимально быстро, поскольку обеспечивается максимально возможное распараллеливание вычислений. Информационный граф большой задачи сегментируется на фрагменты – непересекающиеся базовые подграфы, физически реализуемые в аппаратуре реконфигурируемых вычислительных систем.

Основными вычислительными блоками в реконфигурируемой вычислительной системе являются макропроцессоры, которые позволяют реализовывать операции, предписанные вершинами информационно-

го графа. Макропроцессор представляет собой некоторый набор элементарных процессоров, объединяемых в единый программно-неделимый вычислительный ресурс с помощью локального пространственно-коммутирующего коммутатора. Для реализации информационного графа решаемой задачи макропроцессоры должны иметь возможность соединения в вычислительные параллельно-конвейерные структуры с помощью системного коммутатора, который обеспечивает различные варианты соединения макропроцессоров друг с другом. В состав системы входит распределенная память, обеспечивающая возможность параллельной выдачи массивов входных данных на входы макропроцессоров и записи результатов вычислений с их выходов [1].

Объектом исследования является процесс разработки алгоритмов работы устройств, структурных и принципиальных схем арифметико-символьного процессора, специализированных микропроцессорных модулей, систем коммутаций, реализующих вычислительную модульную систему с перестраиваемой структурой.

Применение специализированных микропроцессорных модулей позволяет повысить быстродействие выполнения арифметических операций за счет реализации конвейеризации. Надежность работы системы повышается за счет дублирования и резервирования каналов передачи информации.

Предметом исследования являются методы, модели, алгоритмы функционирования и структурно-функциональная организация устройств выполнения арифметических и логических операций, осуществление поисковых операций и функций замены в словах текста, сортировка информации и распределение ресурсов.

Универсальные микропроцессоры предназначены для решения широкого круга задач, они способны реализовать любой алгоритм. Специализированные микропроцессоры предназначены для решения определенного класса задач. Эти устройства имеют высокое быстродействие, производительно решают задачи по обработке массивов, интерпретации языков высокого уровня задач искусственного интеллекта, проектирования и создания управляющих устройств. Их существенными особенностями являются простота управления, компактность аппаратных средств, низкая стоимость и малая мощность потребления [1–2].

Целью работы является разработка теоретических основ и принципов построения микропроцессорных модулей для вычислительной открытой развиваемой асинхронной модульной системы с перестраиваемой структурой, арифметико-символьного процессора, специализированных вычислительных модулей и систем коммутаций.

Для достижения поставленной цели необходимо решить следующие задачи:

– проанализировать современные методы, алгоритмы и принципы построения микропроцессорных

модулей вычислительной системы с перестраиваемой структурой;

– разработка коммутационной системы способной конфигурировать микропроцессоры и специализированные вычислительные модули в мультikonвейерную структуру;

– разработка аппаратно-ориентированных процедур и методов структурно-функциональной организации специализированных микропроцессорных модулей;

– разработка структурных и функциональных схем специализированных вычислительных модулей, систем коммутаций;

– экспериментальное подтверждение достоверности результатов и теоретических исследований функционирования алгоритмов.

1 ПОСТАНОВКА ЗАДАЧИ

Вычислительную систему можно представить как иерархию универсальной ЭВМ, систему специализированных вычислительных модулей и гибкую подсистему интерфейсных каналов параллельной передачи информации между устройствами. Задача заключается в создании и исследовании микропроцессорных специализированных модулей для потоковой обработки данных. Потоковая обработка данных заключается в преобразовании потоков векторов входных данных $A_i (i=1, 2, \dots, n)$ и $B_i (i=1, 2, \dots, n)$ в поток векторов $R_i \langle r^i_1, r^i_2, \dots, r^i_k \rangle (i=1, 2, \dots, n)$ выходных данных в соответствии с графом алгоритма $G(S, F)$. Блок арифметических операций выполняет операции: сум-

мирования $\sum_{i=1}^n (a_i + b_i)$, вычитания $V_{i=1}^n (a_i - b_i)$, про-

изведения $P_{i=1}^n (a_i * b_i)$, деления $D_{i=1}^n (a_i / b_i)$ в формате с фиксированной запятой и в формате с плавающей

$\sum_{i=1}^n (a_i 2^x + b_i 2^m)$, $V_{i=1}^n (a_i 2^x - b_i 2^m)$, $P_{i=1}^n (a_i 2^x * b_i 2^m)$,

$D_{i=1}^n (a_i 2^x / b_i 2^m)$ запятой. Блок параллельного поиска

и замены в обрабатываемых словах выполняет левую L_k и правую P_k конкатенацию цепочек

$\alpha = \alpha_1 \alpha_2 \dots \alpha_k$ и $\beta = \beta_1 \beta_2 \dots \beta_k$. Левая конкатенация цепочек равна $L_k = \beta_1 \beta_2 \dots \beta_k \alpha_1 \alpha_2 \dots \alpha_k$, правая кон-

катенация цепочек α и β равна $P_k = \alpha_1 \alpha_2 \dots \alpha_k \beta_1 \beta_2 \dots \beta_k$. Блок выполняет операцию

параллельного поиска вхождений в обрабатываемых словах. Требуется найти все вхождения $\beta = \beta_1 \beta_2 \dots \beta_n$

в слове $\alpha = \alpha_1 \alpha_2 \dots \alpha_m$ в качестве подстроки. Определить все смещения S , для которых подстрока

$\alpha_s = \alpha_{s+1} \dots \alpha_{s+n}$ совпадает с $\beta_1 \dots \beta_n$. Для поиска вычисляется функция $F = (\alpha_1 = \beta_1) \& \dots \& (\alpha_n = \beta_n)$,

если F равно единице, то вхождение найдено в слове. Применяя конечный упорядоченный набор формул

подстановки к обрабатываемому слову α : получается

новое слово $\alpha' = \alpha_1 \dots \alpha_s \beta_1 \beta_2 \dots \beta_n \alpha_{s+n+1} \alpha_m$. Блок логических операций выполняет: инверсию $\overline{a_i}$, конъюнкцию $a_i \& b_i$, дизъюнкцию $a_i \vee b_i$, исключающее ИЛИ $a_i \oplus b_i$, эквивалентность $a_i = b_i$, $a_i \in \{0,1\}$; $b_i \in \{0,1\}$.

2 ОБЗОР ЛИТЕРАТУРЫ

В работе [1] рассмотрены основные архитектурные концепции построения средств обработки информации, модели функциональной организации вычислительных машин и параллельных вычислительных систем. Описаны канонические функциональные структуры конвейерных, матричных, мультипроцессорных распределенных систем.

Дальнейшие пути решения проблемы по перспективному развитию суперкомпьютеров это создание реконфигурированных вычислительных систем, в которых используются программируемые логические интегральные схемы. Данный подход обеспечивает высокую реальную производительность вычислительной системы [2].

В работе [3] представлена вычислительная система, блоки которой выполняют основные арифметические операции двоичных чисел в прямых кодах с фиксированной и плавающей запятой, осуществляют поисковые функции и операции замены, выполняют логические операции. В модульной системе устройства имеются собственные средства обработки, оперативные запоминающие устройства, локальная коммутация и автономный блок управления.

Недостатком представленной организацией возникает проблема организации коммутации и обменов между процессором и устройствами, а также функциональное соединение между вычислительными модулями.

В работе [4] рассмотрены основные вопросы организации параллельной обработки данных, проанализированы возникающие при этом проблемы, а также пути их решения. Показаны особенности организации вычислительного процесса в параллельных системах, рассмотрены языки параллельного программирования и характерные параллельные алгоритмы.

В работе [5] модульная система имеет параллельную структуру за счет использования интерфейсных каналов соединения процессора и вычислительных модулей, характеризуется также асинхронностью работы составляющих ее устройств. В структуру вычислительной системе входят специализированные устройства, которые позволяют сформировать шифротекст информационных данных, используя алгоритмы преобразования на основе операций поиска и замены, которые преобразовывают открытый текст в последовательность некоторых знаков в шифротекст.

В работе [6] предложено устройство, которое выполняет операции поиска вхождений и подстановку символов в словах текста. Новые слова можно формировать © Шевелев С. С., 2021
DOI 10.15588/1607-3274-2021-1-19

мировать с помощью операций поиска и замены вхождений, а также при выполнении операций левой и правой конкатенаций. Необходимо достигнуть высокой скорости поиска и замены вхождений в обрабатываемом слове. Поисковая операция выполняется в определении вхождений, имеющих общие части, так и без общих частей.

В работе [7] предложено устройство сортировки слов. При решении информационно-логических задач требуется выбирать слова по их признакам. Применение ассоциативного запоминающего устройства по упорядочению символьной информации повышает надежность и производительность, позволит упростить алгоритм работы сортировочного устройства.

Устройство сортировки слов содержит блоки поиска вхождений, схемы обработки результатов сравнения, хранения упорядоченных слов.

Недостатком представленной структуры является применение ассоциативных запоминающих устройств, слова в которых отыскиваются по их признакам, по их содержанию, а не по адресам ячеек, в которых записаны эти слова. В АЗУ, параллельных по словам и последовательных по разрядам, для обработки данных используются последовательные алгоритмы.

В работе [8] рассматривается вопрос по взаимораспределению ресурсов между процессорами.

Назначение системы взаимораспределения ресурсов должно обеспечивать режим управления информацией, который обеспечивает оптимальное сбалансированное функционирование системы взаимодействующих локализованных узлов.

В работе [9] в представленном сумматоре происходит сложение и вычитание двоичных чисел по правилам выполнения арифметических операций в троичной системе счисления, которая обладает наибольшей плотностью записи информации. Троичные компьютеры имеют большую удельную емкость памяти и большую удельную производительность процессора, чем двоичные ЭВМ.

В состав устройства входят: блоки выполнения операций, комбинационные схемы по формированию сигнала и перевода числа в код, дешифраторы, двоичные триггеры.

Недостатком представленной структуры является разработка комбинационных схем дешифраторов по определению комбинации суммы чисел, при которой необходимо ввести коррекцию для получения окончательного результата.

В работе [10] упорядочение данных включает анализ возможностей аппаратных средств вычислительных систем, расположения их каналов, объема оперативной памяти, частоты обращений, быстродействие, диапазона обработки входной числовой и символьной информации.

В представленном устройстве сортировки информации выполняется задача сортировки входной числовой и символьной информации по возрастанию и убыванию. В устройстве осуществляются поисковые

операции по указанному интервалу или по конкретному заданному символу или числу.

Недостатком представленной структуры является то, что устройство сортировки информации может использовать методы внешней сортировки данных. Внешние методы приемлемы для файлов данных, которые слишком велики, чтобы поместиться в оперативной памяти процессора.

3 МАТЕРИАЛЫ И МЕТОДЫ

Граф $G(S, F)$ рис. 1 называется информационным графом вычислительной открытой развиваемой асинхронной модульной системы, или графом алгоритма решения задачи. Граф $G(S, F)$ содержит множество вершин $s_i \in S$, каждой из которых приписана некоторая операция O_i , принадлежащая множеству допустимых операций. Дуги $f(s_i, s_{i+1}) \in F$ определяют последовательность выполнения операций, графа, причем если две вершины s_i и s_{i+1} соединены дугой $f(s_i, s_{i+1})$, то это означает, что результат операции O_i является входным данным для операции O_{i+1} . Граф $G(S, F)$ имеет также множество входных дуг $f(s_{0i}, s_i)$, которые определяются блоком входных данных, а также выходных дуг $f(s_i, s_k)$, определяющих хранение результатов ее решения.

Множество вершин S определяется множеством процессоров вычислительной открытой асинхронной системы. Множество дуг представляет собой множество каналов коммуникаций между процессорами. Входные и выходные дуги графа $G(S, F)$ определяются каналами связи с источником входных и приемником выходных данных, в частности, с памятью системы. Операции O_i из множества допустимых операций O реализуются в процессорах вычислительной системы в форме последовательных процедур обработки данных.

Вычислительную открытую развиваемую асинхронную модульную систему можно представить как иерархию универсальной ЭВМ, систему специализированных вычислительных модулей и гибкую подсистему интерфейсных каналов параллельной передачи информации между устройствами [2].

В представленной системе отдельное устройство имеет собственные средства обработки, оперативное запоминающее устройство для хранения данных, локальную коммутацию и автономный блок управления. Он становится самостоятельным, функционально полным вычислительным модулем.

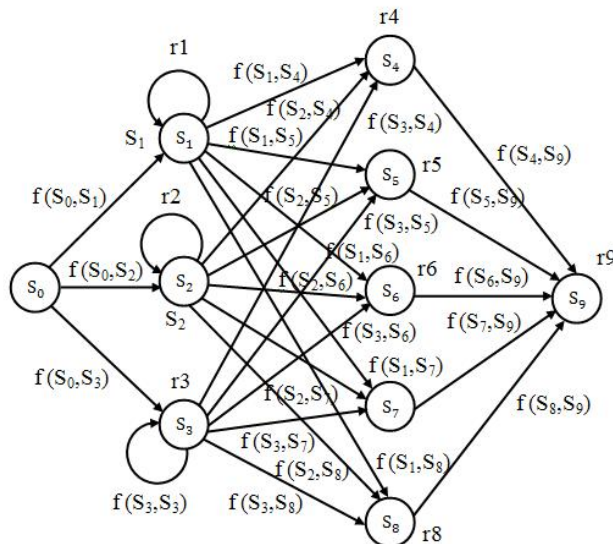


Рисунок 1 – Информационный граф вычислительной открытой развиваемой асинхронной модульной системы

Функциональное назначение устройств вычислительной системы (рис. 2): БВАО – блок выполнения арифметических операций, БПЗВОС – блок параллельного поиска и замены в обрабатываемых словах, БВЛО – блок выполнения логических операций, БХР – блок хранения результатов, используемый для записи и чтения данных; коммутационные системы: КСБУ обеспечивает соединение между процессором и другими устройствами, СКПМ процессор–модуль обеспечивает двустороннюю передачу информации между процессором и специализированными вычислительными модулями, СКММ модуль–модуль обеспечивает соединения между специализированными модулями; специализированные модули: СПЕЦ МОД1 выполняет операции по символьной обработке, СПЕЦ МОД2 осуществляет сортировку слов входного текста, СПЕЦ МОД3 решает задачу распределения ресурсов со многими параметрами между локализованными центрами, СПЕЦ МОД4 выполняет арифметическую операцию сложения чисел. СПЕЦ МОД5 выполняет сортировку двоичной информации.

Семейство высокопроизводительных многопроцессорных вычислительных систем с динамически перестраиваемой, программируемой архитектурой на основе реконфигурируемой элементной базы является конкурентоспособной альтернативой традиционной организации многопроцессорных систем, основанных на использовании микропроцессоров в качестве основного вычислительного элемента.

Вычислительная открытая развиваемая асинхронная модульная система сочетает в себя как последовательный, так и конвейерный способ обработки информации, поскольку входные данные обрабатываются одновременно по разным конвейерным цепочкам процессоров.

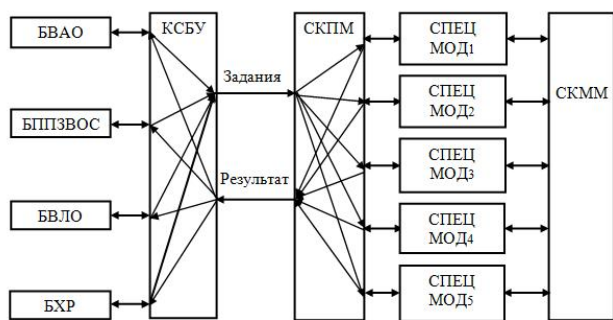


Рисунок 2 – Граф вычислительного процесса вычислительной открытой развиваемой асинхронной модульной системы

Основными вычислительными блоками в реконфигурируемой вычислительной системе являются макропроцессоры, которые позволяют реализовывать операции, предписанные вершинами информационного графа. Макропроцессоры соединяются в вычислительные параллельно-конвейерные структуры с помощью системного коммутатора, который обеспечивает различные варианты соединения вычислительных модулей друг с другом. В состав системы входит распределенная память, обеспечивающая параллельную выдачу массивов входных данных на входы макропроцессоров и записи результатов вычислений с их выходов.

Для решения этой задачи можно использовать блок выполнения логических операций с последовательной архитектурой, основными элементами которого являются элементарные вычислители, выполняющие логические операции: $\{\&, V, \bar{\quad}\}$, $\{\&, \bar{\quad}\}$, $\{V, \bar{\quad}\}$, $\{\&, \oplus, 1\}$, которые относятся к функционально полной системе булевых функций.

К множеству микроопераций $L = \{L_{12}, L_{13}, \dots, L_{1Z}\}$ выполняемых блоком логических операций S_3 относятся: инверсия $L_{12} \equiv g = \bar{a}$, конъюнкция $L_{12} \equiv g = a \& (a_i b_i)$, дизъюнкция $L_{12} \equiv g = a V (a_i b_i)$, исключающее ИЛИ $L_{12} \equiv g = a \oplus (a_i b_i)$, $g_i \in \{0, 1\}$; $a_i \in \{0, 1\}$; $b_i \in \{0, 1\}$, эквивалентность $L_{12} \equiv g = a \approx (a_i b_i)$, $g_i \in \{0, 1\}$; $a_i \in \{0, 1\}$; $b_i \in \{0, 1\}$.

На рис. 3 представлены: блок ввода данных БВД, система электронных ключей СиК, элементарные вычислители логических операций ЭлВ, блок хранения результатов БХР.

Время результата $T_{\text{пос}}$ обработки N векторов входных данных определяется по формуле

$$T_{\text{пос}} = N \cdot (t_{\text{БВД}} + t_{\text{СиК}} + t_{\text{ЭлВ}} + t_{\text{БХР}}) = N \sum_{i=1}^M s(O_i) \cdot \tau. \quad (1)$$

Сокращения времени $T_{\text{пар}}$ обработки потока входных векторов является распараллеливания про- © Шевелев С. С., 2021
 DOI 10.15588/1607-3274-2021-1-19

цесса обработки. В этом случае параллельный способ подразумевает наличие K процессоров $P_i (i=1, 2, \dots, K)$, каждый из которых может работать независимо от других процессоров. Каждый из процессоров $P_i (i=1, 2, \dots, K)$ запрограммирован на реализацию графа $G(S, F)$. Множество входных векторов $Z_i (i=1, 2, \dots, N)$ разбивается на (N/K) непересекающихся подмножеств, каждое из этих подмножеств входных векторов может быть обработано на процессоре P_i независимо, то есть параллельно с другими подмножествами.

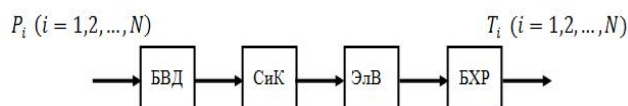


Рисунок 3 – Блоки выполнения логических операций с последовательной архитектурой

Для решения этой задачи можно использовать блок выполнения математических операций в форматах с фиксированной и плавающей запятой с параллельной архитектурой, который приведен на рис. 4. Основными элементами, которого являются математические вычислители, выполняющие арифметические и сдвиговые операции.

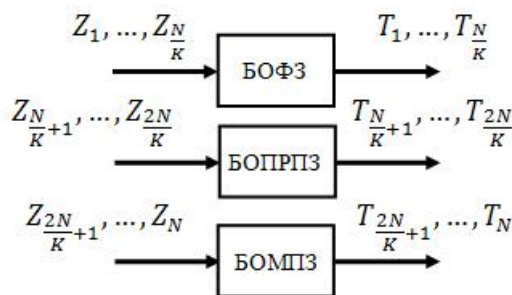


Рисунок 4 – Параллельный способ обработки векторов входных данных

К множеству микроопераций $R = \{R_{12}, R_{13}, \dots, R_{1k}\}$ выполняемых блоком арифметических операций S_1 относятся: суммирование $R_{12} = \sum_{i=1}^n (a_i + b_i)$, вычитание

$R_{13} = V_{i=1}^n (a_i - b_i)$, произведение $R_{14} = P_{i=1}^n (a_i * b_i)$, деление $R_{15} = D_{i=1}^n (a_i / b_i)$ в формате с фиксированной запятой и в формате с плавающей запятой $R_{16} = \sum_{i=1}^n (a_i 2^x + b_i 2^m)$, $R_{17} = V_{i=1}^n (a_i 2^x - b_i 2^m)$, $R_{18} = P_{i=1}^n (a_i 2^x * b_i 2^m)$, $R_{19} = D_{i=1}^n (a_i 2^x / b_i 2^m)$.

В этом случае время получения результата $T_{\text{пар}}$, при обработке всего множества векторов входных данных, будет вычисляться по формуле

$$T_{\text{пар}} = \left(\frac{N}{K}\right) \cdot (t_{\text{БОВФЗ}} + t_{\text{БОВПРПЗ}} + t_{\text{БОВМПЗ}}) = \left(\frac{N}{K}\right) \sum_{i=1}^M s(O_i) \cdot \tau. \quad (2)$$

Время обработки потока сокращается в K раз по сравнению с последовательной обработкой всех входных векторов.

К множеству микроопераций $C = \{C_{12}, C_{13}, \dots, C_{1t}\}$ выполняемых блоком параллельного поиска и замены в обрабатываемых словах S_2 относятся: левая конкатенация β и α цепочек $\alpha = \alpha_1 \alpha_2 \dots \alpha_k$ и $\beta = \beta_1 \beta_2 \dots \beta_p$ символы одного этого же алфавита V , $\alpha_i \in V$, $\forall i=1, \dots, k$ и $\beta_i \in V$, $\forall i=1, \dots, p$, равна $\gamma = \beta_1 \beta_2 \dots \beta_p \alpha_1 \alpha_2 \dots \alpha_k$, правая конкатенация цепочек α и β принимает вид $\gamma = \alpha_1 \alpha_2 \dots \alpha_k \beta_1 \beta_2 \dots \beta_p$ над тем же алфавитом V .

Блок выполняет операцию параллельного поиска вхождений в обрабатываемых словах. Требуется найти все вхождения $x = \beta_1 \beta_2 \dots \beta_m$ в слове $w = \alpha_1 \alpha_2 \dots \alpha_n$ в качестве подстроки. Определить все смещения s , для которых подстрока $w_s = \alpha_{s+1} \dots \alpha_{s+m}$ совпадает с вхождением $x = \beta_1 \dots \beta_m$. Для поиска вычисляется функция $F = (\alpha_{s+1} = \beta_1) \& (\alpha_{s+2} = \beta_2) \& \dots \& (\alpha_{s+m} = \beta_m)$, если F равно единице, то вхождение найдено в слове. Смещение s определяет позицию вхождения в слове, сдвиг символов слова осуществляется на m позиций вправо [3].

В блоке выполняется операция замены вхождений в обрабатываемом слове, которая задается формулой подстановки. Применяя конечный упорядоченный набор формул подстановки к обрабатываемому слову w :

$$\begin{cases} \alpha_{s+1} \rightarrow \beta_1 \\ \cdot \quad \cdot \quad \cdot \\ \alpha_{s+m} \rightarrow \beta_m \end{cases} \quad (k \geq 1), \quad (3)$$

получается новое слово $w' = \alpha_1 \dots \alpha_s \beta_1 \beta_2 \dots \beta_m \alpha_{s+m+1} \alpha_n$. Если это слово взять за исходное и применить те же процедуры, то получается новое слово w'' и так далее $w \rightarrow w' \rightarrow w'' \dots$.

Быстродействие конвейера при длительной работе будет значительно превышать быстродействие, которое достигается при последовательной обработке команд. Это увеличение будет тем больше, чем меньше длительность такта конвейера и чем больше количество выполненных за исследуемый период команд.

В блоке параллельного поиска и замены в обрабатываемых словах вычислительной системы можно использовать конвейерную обработку. Блок имеет 4 элементарных процессора $P_i (i=1, 2, 3, 4)$, соединенных в цепочку, как показано на рис. 5, каждый из элементарных процессоров P_i запрограммирован на реализацию соответствующего подграфа

$G_i(S_i, F_i) (i=1, 2, 3, 4)$ общего графа алгоритма $G(S, F)$. На рис. 5 представлены процессоры: памяти вхождений и обрабатываемых слов БПВС, сравнения и анализа вхождений в обрабатываемых словах БАВС, замены вхождений в обрабатываемых словах БЗАМ, хранения результатов подстановок БХРП.

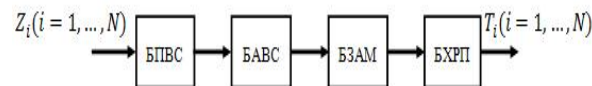


Рисунок 5 – Конвейерная организация работы блока параллельного поиска и замены в обрабатываемых словах

Длительность такта T_{max} будет равна длительности самого медленного этапа обработки – блока сравнения и анализа вхождений в обрабатываемых словах БАВС.

$$T = \max \{T_{\text{БПВС}}, T_{\text{БАВС}}, T_{\text{БЗАМ}}, T_{\text{БХРП}}\} = T_{\text{БАВС}}. \quad (4)$$

Время $T_{\text{кон}}$ при конвейерной обработке N команд будет вычисляться следующей формулой

$$T_{\text{кон}} = P_i \cdot T_{\text{БАВС}} + (N-1) \cdot T_{\text{БАВС}} = T_{\text{БАВС}} \cdot (P_i + (N-1)). \quad (5)$$

Реконфигурируемые вычислительные системы с динамически перестраиваемой архитектурой превосходят по технико-экономическим характеристикам многопроцессорные вычислительные системы с традиционной архитектурой в сегментах рынка супер-ЭВМ и рабочих станций, а также ускорителей к персональному компьютеру.

Сравнительный анализ времени и скорости конвейерной и последовательной архитектур на примере работы блока параллельного поиска и замены в обрабатываемых словах вычислительной системы. Время и скорость при последовательной организации работы блока вычислительной системы $T_{\text{пос}}$ и $V_{\text{пос}}$ определяются по формулам

$$\begin{aligned} T_{\text{пос}} &= N \cdot (\tau_1 + \tau_{\text{max}} + \tau_3 + \tau_4), \\ V_{\text{пос}} &= \frac{1}{4} = \frac{1}{\tau_1 + \tau_{\text{max}} + \tau_3 + \tau_4}, \end{aligned} \quad (6)$$

Время $T_{\text{кон}}$ и скорость $V_{\text{кон}}$ конвейерной организации работы блока параллельного поиска и замены в обрабатываемых словах определяются из отношений

$$\begin{aligned} T_{\text{кон}} &= (N+4-1) \cdot \tau_{\text{max}} = (N+3) \cdot \tau_{\text{max}}, \\ V_{\text{кон}} &= (\tau_{\text{max}})^{-1} = \frac{1}{\tau_{\text{max}}}. \end{aligned} \quad (7)$$

Отношение времени последовательной организации вычислительного процесса $T_{\text{пос}}$ к конвейерной $T_{\text{кон}}$ будет определяться как

$$\frac{T_{\text{пос}}}{T_{\text{кон}}} = \frac{\tau_1 + \tau_{\text{max}} + \tau_3 + \tau_4}{\tau_{\text{max}}} \approx \frac{4 \cdot \tau_{\text{max}}}{\tau_{\text{max}}} = 4. \quad (8)$$

Отношение скорости конвейерной $V_{\text{кон}}$ к последовательной $V_{\text{пос}}$ определится как

$$\frac{V_{\text{кон}}}{V_{\text{пос}}} = \frac{\tau_1 + \tau_{\text{max}} + \tau_3 + \tau_4}{\tau_{\text{max}}} \approx \frac{4 \cdot \tau_{\text{max}}}{\tau_{\text{max}}} = 4. \quad (9)$$

При большом значении числа входных векторов N конвейерная организация вычислительного процесса требует меньше времени, а скорость будет больше, чем последовательная более чем в 4 раза, где 4 – число элементарных процессоров в конвейерной цепочке блока. Эффективность архитектуры конвейерной организации вычислительного процесса по сравнению с последовательной приблизительно в N раз выше [4].

Время выполнения операции блоком параллельного поиска и замены в обрабатываемых словах конвейерной и последовательной архитектур приведены в таблице 1.

Таблица 1 – Время определения поиска и замены в словах

Число входных векторов N	1	5	10	15	20
Время $T_{\text{пос}}$ нс	500	2500	5000	7500	10000
Время $T_{\text{кон}}$ нс	800	1600	2600	3600	4600

Сравнительный анализ времени конвейерной и последовательной архитектур. Изображенный график на рис. 6 иллюстрирует разницу в эффективности архитектуры конвейерной организации вычислительного процесса по сравнению с последовательной.

Важным компонентом массово-параллельных компьютеров является коммуникационная среда. Это набор аппаратных и программных средств, обеспечивающих обмен сообщениями между процессорами. Она предназначена для достижения высоких скоростей передачи с малым временем задержки и при этом обеспечивает масштабируемую архитектуру, позволяющую строить системы, состоящие из множества блоков.

Система коммутации процессор-модуль обеспечивает передачу информации от арифметико-символьного процессора на входы специализированных вычислительных модулей системы в виде заданий, а также формирует интерфейсный канал процессор-модуль передачи информации с выходов специализированных вычислительных модулей системы в виде результата для хранения его в памяти арифметико-символьного процессора.

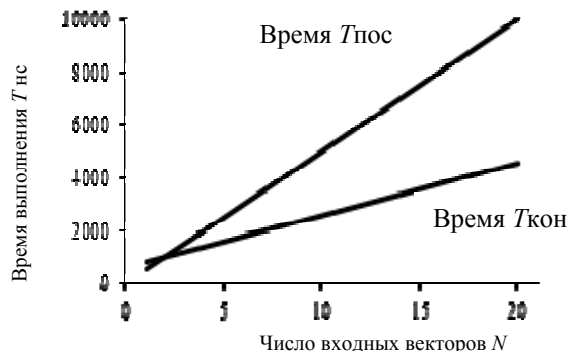


Рисунок 6 – График сравнение времени конвейерной и последовательной архитектур

Система коммутации модуль-модуль обеспечивает передачу информации с выходов специализированных вычислительных модулей на входы других специализированных вычислительных модулей системы в виде заданий и результатов, а также формирует интерфейсный канал передачи информации с выходов специализированных вычислительных модулей системы в виде результатов для хранения их в памяти арифметико-символьного процессора [11].

Специализированный вычислительный модуль СпецВычМод1 выполняет операции поиска вхождения в обрабатываемых словах приведен на рис. 7. Первый формат работы устройства определяет вхождение, которые имеют одинаковые части. Это означает, что предыдущие вхождение и последующие имеют общую часть, состоящую из одной буквы или цепочки символов. Вторым форматом работы системы характеризуется определением вхождений, не имеющих общих частей. В этом случае определяется адрес только через и сдвиг. Если необходимо произвести замену найденного вхождения на подстановку буквы или слово, заранее определенную и записанную в регистр подстановок.

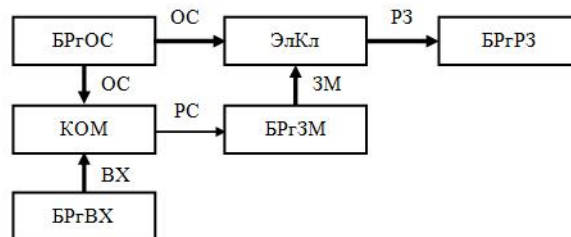


Рисунок 7 – Специализированный вычислительный модуль СпецВычМод1

БРгОС – блок регистра обрабатываемого слова; БРгВХ – блок регистра вхождений; БРгЗМ – блок регистра замены; КОМ – компаратор; БРгРЗ – блок регистра результата замены; ЭлКл – схема электронных ключей.

В этом случае, вначале осуществляется операция поиска вхождений без общих частей, затем производится операция замены, найденного вхождения на подстановку. Процедуры сдвига возможны при помощи реверсивных регистров, которые осуществляют

сдвиг информации как влево, так и вправо. В параллельной системе поиска и замены осуществляется процесс обработки информации сразу с несколькими словами и несколькими входениями в параллельном формате. Для выполнения параллельной обработки в системе имеются n -блоков поиска и замены [6].

Специализированный вычислительный модуль СпецВычМод2 выполняет операцию сортировки слов с помощью ассоциативного запоминающего устройства приведен на рис. 8.

Поиск слов в памяти происходит по содержанию. В ассоциативном запоминающем устройстве для обработки данных используются последовательные алгоритмы, поиск осуществляется параллельно по словам и последовательно по разрядам. Это позволяет проводить операции на равенство искомых слов равных заданному признаку. В режиме поиска АЗУ считываются из памяти все слова, подчиняющиеся определенным условиям: равные, больше или меньше, поиск минимального, максимального значения, поиск величин, заключенных в заданном интервале, упорядоченная сортировка [7].

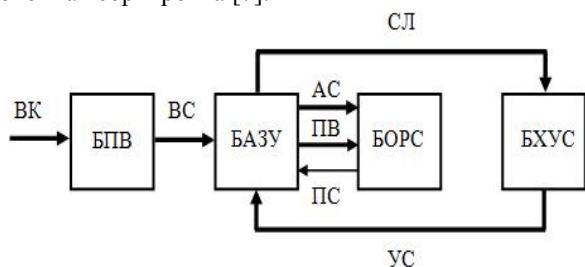


Рисунок 8 – Специализированный вычислительный модуль СпецВычМод2

БПВ – блок поиска входений; БАЗУ – блок ассоциативных запоминающих устройств; БОРС – блок обработки результатов сравнения; БХУС блок хранения упорядоченных слов.

Специализированный вычислительный модуль СпецВычМод3 формирует режимы управления материальными потоками или информацией, которые обеспечивают оптимальное сбалансированное функционирование системы взаимодействующих локализованных узлов по многим ресурсам и параметрам приведен на рис. 9.

Локализованные центры системы служат для формирования избыточного ресурса, который необходимо распределить по ассоциативным запоминающим устройствам других блоков, а также для получения адреса хранения и анализа определенного количества ресурса, полученного от других блоков. Система распределения ресурсов определяет «свободные» места в ассоциативных запоминающих устройствах избыточных ресурсов и их параметров. Процесс передачи избытка на свободные места произойдет тогда, когда объемы избыточных ресурсов и параметров равны или меньше предоставленных объемов «свободных» мест [8].

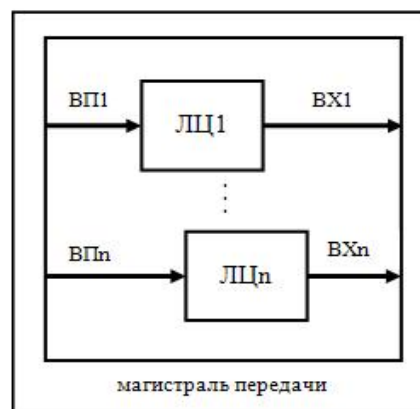


Рисунок 9 – Специализированный вычислительный модуль СпецВычМод3

ЛЦ n – локализованные центры системы распределения; магистраль передачи.

Специализированный вычислительный модуль СпецВычМод4 выполняет микрооперацию сложения чисел в троичной системе счисления приведен на рис. 10.

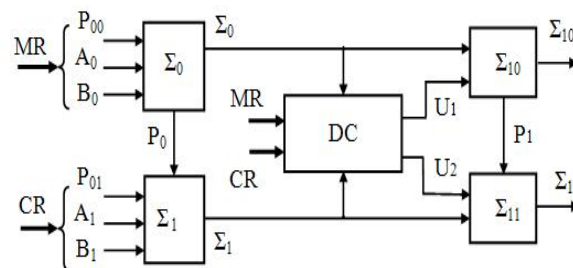


Рисунок 10 – Специализированный вычислительный модуль СпецВычМод4

Использование троичной системы счисления обосновывается следующим результатом. При сложении младших разрядов A_0 и B_0 , старших A_1 и B_1 разрядов и переносов из младших разрядов в старшие P_{00}, P_{01} формируются суммы Σ_0, Σ_1 и переносы $P_0, P_1, A_i \in \{0,1\}; B_i \in \{0,1\}$. Сигналы Σ_{10} и Σ_{11} представляют младший и старший разряды окончательной суммы [9].

$$\Sigma_0 = \sum_{i=1}^3 \oplus (A_0 B_0 P_{00}), \Sigma_0 \in \{0,1\};$$

$$A_0 \in \{0,1\}; B_0 \in \{0,1\}; P_{00} \in \{0,1\}. \quad (10)$$

$$\Sigma_1 = (\sum_{i=1}^2 \oplus (P_0 P_{01})) \oplus (\sum_{i=1}^2 \oplus (A_1 B_1)),$$

$$\Sigma_1 \in \{0,1\}; P_0 \in \{0,1\}; P_{01} \in \{0,1\};$$

$$A_1 \in \{0,1\}; B_1 \in \{0,1\}. \quad (11)$$

$$\Sigma_{10} = \sum_{i=1}^2 \oplus (\Sigma_0 U_1), \Sigma_{10} \in \{0,1\};$$

$$\Sigma_0 \in \{0,1\}; U_1 \in \{0,1\}; \quad (12)$$

$$\Sigma_{11} = \sum_{i=1}^3 \oplus (P_1 \Sigma_1 U_2), \Sigma_{11} \in \{0,1\};$$

$$P_1 \in \{0,1\}; \Sigma_1 \in \{0,1\}; U_2 \in \{0,1\}. \quad (13)$$

Специализированный вычислительный модуль СпецВычМод5 выполняет сортировку информации методом преобразования данных в адрес приведен на рис. 11. Входные данные поступают одновременно на информационные и адресные входы оперативных запоминающих устройств. По сформированным адресам данные и числа одинаковых символов и чисел записываются в оперативную память устройства.

Преобразования данных в адрес происходит с помощью хеш-функции, которая отображает каждый ключ из набора S во множество целых чисел без коллизий.

Инъективное отображение для целого $k \geq 1$ описывается функцией $h(k): U \rightarrow [m]$, она является k -идеальной хеш-функцией для $S \subseteq U$, если для каждого $j \in [m]$ имеем $|\{x \in S | h(x) = j\}| \leq k$.

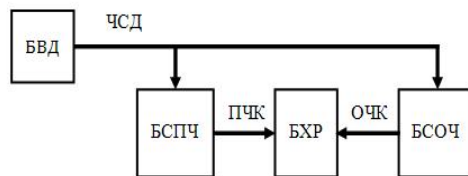


Рисунок 11 – Специализированный вычислительный модуль СпецВычМод5

БВД – блок ввода данных; БСПЧ – блок сортировки положительных чисел; БСОЧ – блок сортировки отрицательных чисел; БХР – блок хранения результата.

Блок-схема алгоритма работы устройства упорядочения информации методом преобразования данных в адрес приведена на рис. 12.

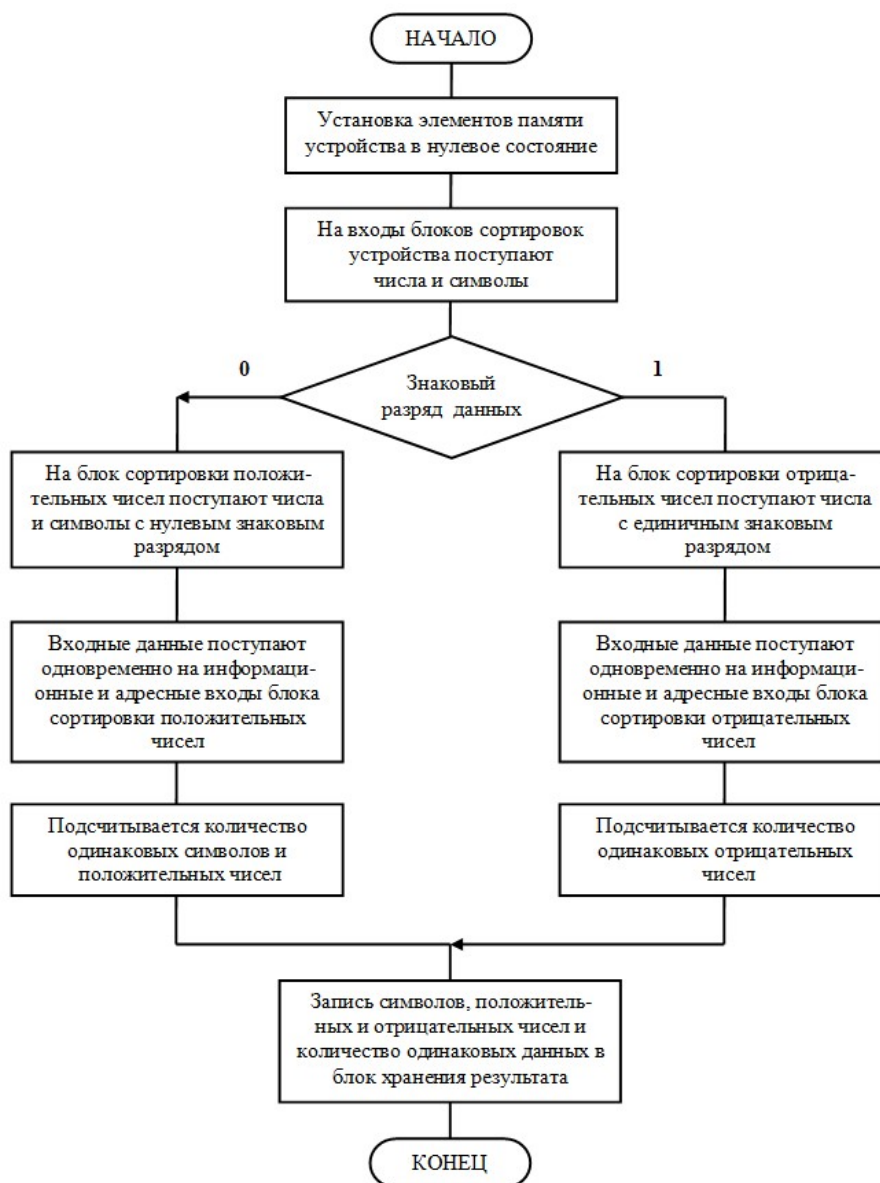


Рисунок 12 – Блок-схема алгоритма работы устройства упорядочения информации методом преобразования данных в адрес

4 ЭКСПЕРИМЕНТЫ

Целью эксперимента было моделирование работы арифметического вычислителя при выполнении операций суммирования, вычитания, умножения и деления в формате с фиксированной точкой в прямых кодах (табл. 2).

Если входные числа имеют разные знаки, то сравнения чисел в вычислителе не происходит, выполняется операция вычитания из первого числа второе. Если формируется заем из знакового разряда, то числа с помощью коммутатора меняются местами на входе нейронов сумматора-вычитателя. На втором такте от второго большего числа вычитается первое меньшее. Если заема не образуется, то от большего первого числа вычитается меньшее второе. Операция умножения выполняется в прямом коде умножением младшими разрядами множителя со сдвигом суммы частичных произведений влево. Операция деления выполняется по алгоритму деления чисел без восстановления остатков. Для определения следующей цифры частного, необходимо сдвинуть текущий остаток влево на один разряд, а затем алгебраически прибавить к нему модуль делителя, которому приписывается знак, противоположный знаку текущего остатка. Операция сдвигов и алгебраических сложений повторяется до тех пор, пока в частном не получится требуемое количество цифр.

5 РЕЗУЛЬТАТЫ

Результаты проектирования вычислительной системы, специализированных модулей, систем коммутаций приведены на рис. 13–16, содержат:

1. Разработаны варианты структурно-функциональной организации вычислительной системы, специализированных вычислительных модулей и гибкой подсистемы интерфейсных каналов. Модули имеют свою систему команд и работают под управлением микропроцессора.

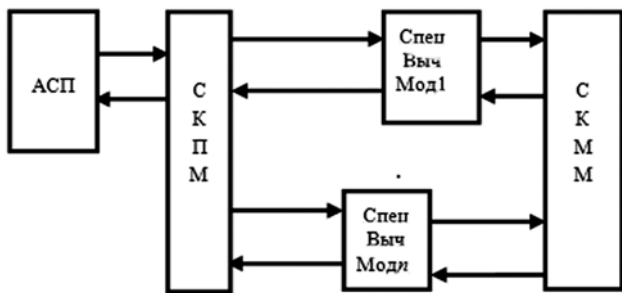


Рисунок 13 – Структурно-функциональная организация вычислительной открытой развиваемой асинхронной модульной системы

Таблица 2 – результаты выполнения арифметических операций в прямых кодах

1 число в $10^{\text{й}}$ с/с	2 число в $10^{\text{й}}$ с/с	1 число в $2^{\text{й}}$ с/с	2 число в $2^{\text{й}}$ с/с	Арифмет. операция	Результат в 2 й с/с	Результат в 10 й с/с
74	32	0.01001010	0.00100000	сложение	0.01101010	106
25	68	0.00011001	0.01000100	вычитание	1.00101011	-43
-9	14	1.1001	0.1110	умножение	1.01111110	-126
83	-24	0.1010011	1.11000	деление	1.11,0111	-3,4583

2. Разработаны варианты структурно-функциональной организации арифметико-символьного процессора, выполняющего основные арифметические операции, поисковые функции и операции замены, логические операции.

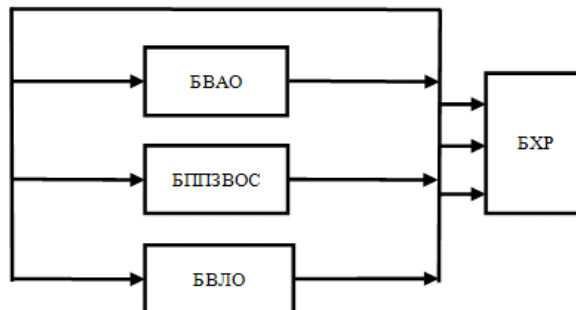


Рисунок 14 – Структурно-функциональная организация арифметико-символьного процессора

3. Разработаны варианты структурно-функциональной организации устройства сортировки информации методом дешифрации данных.

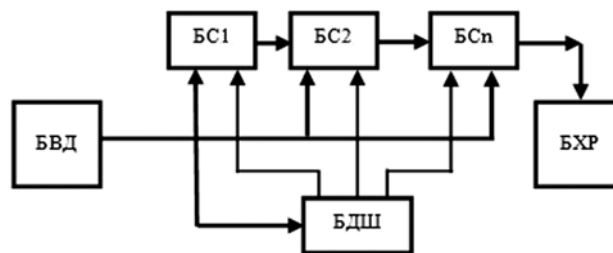


Рисунок 15 – Структурная схема устройства сортировки информации методом дешифрации данных

4. Разработана и исследована архитектура системы коммутации модуль-модуль, осуществляющая передачу информации с выходов специализированных вычислительных модулей на входы других специализированных вычислительных модулей системы в виде заданий и результатов.

ИНТЕРФЕЙСНЫЙ КАНАЛ

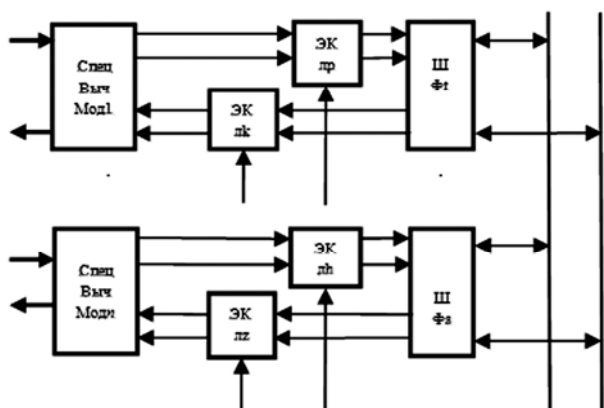


Рисунок 16 – Структурно-функциональная организация системы коммутации модуль-модуль

6 ОБСУЖДЕНИЕ

Разработан алгоритм работы и структурно-функциональная организация устройства упорядочения информации методом преобразования данных в адрес [10], обеспечивающая повышение быстродействия по сравнению с методом сортировки выбором не менее чем в 1,5 раза. Сортировка выбором основана на формировании отсортированной последовательности путем присоединения к ней одного элемента за другим в правильном порядке [12]. Общее количество операций при этом методе равно $T_{ОП} = 1/2 * (2 * q + q) = 3/2 * q$. Упорядочение информации методом преобразования данных в адрес происходит без операции сравнения чисел.

Специализированное устройство системы реализует алгоритм ускоренного умножения младшими разрядами множителя вперед со сдвигом множителя влево [12]. Анализируются два разряда множителя, сдвиг осуществляется на два разряда вправо. Все цифры множителя такие, то суммирование выполняется при анализе каждой пары разрядов множителя. Максимальное время определяется как: $T_{умн.мах} = n/2 * (t_{\Sigma} + 2t_{сдв})$. Матричный множитель вычисляет сумму поразрядных произведений двоичных чисел, которая является результатом умножения. Поразрядное произведение определяется, как логическая функция И. Разрядность произведения определяется, как сумма разрядностей сомножителей время умножения определяется как: $T_{умн.мах} = 2 * (n-1) * t_{\Sigma}$. Сравнивая максимальное время выполнения операции умножения приведенными алгоритмами можно сделать вывод, что операция умножения ускоренного множителя двоичных чисел на нейронах не менее чем в 2 раза вычисляет результат быстрее, чем известное устройство.

Троичная позиционная несимметричная система счисления по затратам числа знаков наиболее экономична из позиционных показательных несимметричных систем счисления. Она обладает наибольшей плотностью записи информации, описывается уравнением $y = (\ln(y)) / y$. Троичные ЭВМ имеют большую

удельную емкость памяти и большую удельную производительность процессора, чем двоичные. При сложении тритов в троичных полусумматорах и в троичных сумматорах количество сложений в $\log_2 3 = 1,58$ раза меньше, чем при сложении битов в двоичных полусумматорах и в двоичных сумматорах, и, следовательно, быстродействие при сложении в 1,58.. раза (на 58 %) больше. Сумматор-вычитатель выполняет сложение и вычитание двоичных чисел по правилу выполнения арифметических операций в троичной системе счисления [9]. При выполнении арифметической операции анализируются знаковые разряды чисел и код операции. Если выполняется операция суммирования, то числа представлены в прямом коде, при выполнении вычитания в обратном коде.

ВЫВОДЫ

Научная новизна полученных результатов состоит в том, что предложены микропроцессорные модули, которые могут использоваться для создания высокопроизводительных, быстродействующих вычислительных систем. Специализированные модули выполняют основные арифметические, символьные и логические операции.

Научная новизна полученных результатов заключается в том, что арифметические операции выполняются в прямых кодах с фиксированной и плавающей запятой, символьные операции выполняют поисковые функции и операции замены вхождений в обрабатываемых словах, выполняются логические операции на пороговых и нейроподобных элементах.

Микропроцессорные модули представляют собой цифровые специализированные устройства, которые выполняют определенную задачу в вычислительной системе. Модули представлены в виде структурной схемы устройства, структурно-функциональных схем блоков и алгоритма работы устройства.

Практическая значимость полученных результатов заключается в использовании микропроцессорных модулей в перестраиваемой вычислительной системе, что позволит повысить быстродействие выполнения арифметических операций за счет реализации конвейеризации и использование параллелизма при обработке данных, системы коммутации обеспечивают двухстороннюю информационную связь между блоками, позволяющая обеспечить распределенное децентрализованное управление ресурсами системы.

Перспективы дальнейших исследований состоят в создании вычислительной модульной системы с перестраиваемой структурой за счет системы коммутации процессор-модуль, обеспечивающая передачу информации от арифметико-символьного процессора на входы специализированных вычислительных модулей системы в виде заданий, а также система коммутаций модуль-модуль, осуществляющая передачу информации с выходов специализированных вычислительных модулей на входы других специализированных вычислительных модулей системы в виде заданий и результатов.

БЛАГОДАРНОСТИ

Работа выполнена в рамках госбюджетной научно-исследовательской темы в ООО «НПЦ «ИнноТех» (г. Курск) для построения вычислительного кластера, где использовались созданные структурные и функциональные схемы, а также разработанные алгоритмы управления работой специализированных микропроцессоров.

ЛІТЕРАТУРА / LITERATURA

1. Хорошевский В. Г. Архитектура вычислительных систем: учеб. пособие / В. Г. Хорошевский. – М.: Изд-во МГТУ им. Н. Э. Баумана, 2008. – 520 с.
2. Гузик В. Ф. Реконфигурируемые вычислительные системы / В. Ф. Гузик, И. А. Каляев, И. И. Левин. – Таганрог: Изд-во Южный федеральный университет, 2016. – 472 с.
3. Пат. Российская Федерация, МПК G 06 F 15/76. Вычислительная открытая развиваемая асинхронная модульная система ВОРАМС / С. С. Шевелев; заявитель и патентообладатель Юго-Западный государственный университет. – № 2453910; заявл. 08.04.2009; опубл. 20.06.2012, Бюл. № 17. – 81 с.
4. Хокни Р. Параллельные ЭВМ. Архитектура, программирование и алгоритмы / Р. Хокни, К. Джессхоуп. – М.: Радио и связь, 1986. – 392 с.
5. Шевелев С. С. Вычислительная открытая развиваемая асинхронная модульная система / С. С. Шевелев // Проблемы информационной безопасности. Компьютерные системы Санкт-Петербургского государственного политехнического университета. – 2013. – No. 4. – С. 86–92.
6. Пат. Российская Федерация, МПК G 11C 15/00. Способ параллельного поиска и замены строки и однородная запинающая матрица для его реализации / Е. А. Титенко, И. С. Зерин, В. С. Евсюков, К. С. Скорняков, Е. Б. Тутов; заявитель и патентообладатель Юго-Западный государственный университет. – № 2012113755; заявл. 06.04.2012; опубл. 20.10.2013, Бюл. № 29

7. Пат. Российская Федерация, МПК G06F 17/30. Устройство сортировки слов / С. С. Шевелев; заявитель и патентообладатель Курский государственный технический университет. – № 2223538; заявл. 08.04.2002; опубл. 10.02.2004, Бюл. № 4.
8. Пат. Российская Федерация, МПК G 06 F 17/50, G 06 N 7/08. Устройство для моделирования систем массового обслуживания / Е. А. Тимофеева, Н. Б. Парамонов, Ю. Н. Парамонов; заявитель и патентообладатель Е. А. Тимофеева, Н. Б. Парамонов, Ю. Н. Парамонов. – № 2447496; заявл. 24.12.2010; опубл. 10.04.2012, Бюл. № 10
9. Пат. Российская Федерация, МПК G06F 7/50. Функциональная структура параллельного сумматора троичной системы счисления $f(+1,0,-1)$ в позиционно-знаковой ее реализации $f(+/-)$ / Л. П. Петренко; заявитель и патентообладатель Л. П. Петренко. – № 2008116450; заявл. 29.04.2008; опубл. 10.11.2009.
10. Пат. Российская Федерация, МПК G06F 7/08. Устройство сортировки информации / И. Н. Емельянова, В. В. Ефремов; заявитель и патентообладатель ГОУ высшего профессионального образования Курский государственный технический университет. – № 2004117660; заявл. 09.06.2004; опубл. 10.01.2006.
11. Каляев И. А. Реконфигурируемые мультимониторные вычислительные структуры / И. А. Каляев, И. И. Левин, Е. А. Семерников. – Ростов н/Д: Изд-во ЮНЦ РАН, 2008. – 320 с.
12. Желтов С. А. Адаптация метода Шермана-Лемана решения задачи факторизации к вычислительной архитектуре CUDA / С. А. Желтов // Вестник РГГУ. – 2012. – № 14. – С. 84–91.
13. Кнут Д. Э. Искусство программирования, том 3. Сортировка и поиск, 2-е изд.: Пер с англ. – М.: Издательский дом «Вильямс», 2001. – 832 с.

Статья поступила в редакцию 24.11.2020.
После доработки 01.02.2021.

УДК 681.3

РЕКОНФІГУРОВАЛЬНА ОБЧИСЛЮВАЛЬНА МОДУЛЬНА СИСТЕМА

Шевельов С. С. – канд. техн. наук, доцент кафедри інформаційної безпеки, Південно-Західний державний університет, Курськ, Росія.

АНОТАЦІЯ

Актуальність. Сучасні універсальні обчислювальні машини здатні реалізувати будь-який алгоритм, але при вирішенні певних завдань по швидкості обробки не можуть конкурувати зі спеціалізованими обчислювальними модулями. Спеціалізовані пристрої мають високу швидкість, ефективно вирішують завдання з обробки масивів, задач штучного інтелекту, використовуються у якості керуючих пристроїв. Застосування спеціалізованих мікропроцесорних модулів, які реалізують обробку рядків символів, логічних і числових значень, представлених у вигляді цілих і дійсних чисел, дозволяє підвищити швидкість виконання арифметичних операцій за рахунок використання паралелізму при обробці даних.

Мета. Розробити принципи побудови мікропроцесорних модулів для обчислювальної модульної системи з перебудовуваною структурою, арифметико-символьного процесора, спеціалізованих обчислювальних пристроїв, комутаційних систем, здатних конфігурувати мікропроцесори і спеціалізовані обчислювальні модулі у мультимониторну структуру для підвищення швидкості виконання арифметичних і логічних операцій, високошвидкісних алгоритмів проектування спеціалізованих процесорів-акселераторів символічної обробки. Розробити алгоритми, структурні і функціональні схеми спеціалізованих математичних модулів, що виконують арифметичні операції в прямих кодах на нейрорподібних елементах і систем децентралізованого управління роботи блоків.

Метод. Побудований інформаційний граф обчислювального процесу модульної системи з перебудовуваною структурою. Розроблено структурні та функціональні схеми, алгоритми, що реалізують побудову спеціалізованих модулів для виконання арифметичних і логічних операцій, пошукових операцій і функцій заміни входжень в оброблюваних словах. Розроблено програмне забезпечення моделювання роботи арифметико-символьного процесора, спеціалізованих обчислювальних модулів, систем комутацій.

Результати. Розроблено структурну схему реконфігурованою обчислювальної модульної системи, яка складається з сумісних функціональних модулів, вона здатна до статичної та динамічної реконфігурації, має паралельну структуру сполуки процесора і обчислювальних модулів за рахунок використання інтерфейсних каналів. Система складається з арифметико-символьного процесора, спеціалізованих обчислювальних модулів і систем комутацій, виконує специфічні завдання символічної обробки інформації, арифметичні і логічні операції.

Висновки. Архітектура реконфігурованих обчислювальних систем може динамічно змінюватися в процесі їх функціонування. З'являється можливість адаптації архітектури обчислювальної системи під структуру розв'язуваної задачі, створювати проблемно-орієнтовані обчислювачі, структура яких відповідає структурі розв'язуваної задачі. В якості основного обчислювального елемента в реконфігурованих обчислювальних системах використовуються не універсальні мікропроцесори, а програмовані логічні інтегральні схеми, які об'єднані за допомогою високошвидкісних інтерфейсів в єдине обчислювальне поле. Реконфігуровані мультимониторні пристрої.

вейерніе обчислювальні системи на основі полів є ефективним засобом для вирішення потокових задач обробки інформації та управління.

КЛЮЧОВІ СЛОВА: інформаційний граф, паралельно-конвеєрна структура, модульна система, спеціалізовані обчислювальні блоки.

UDC 681.3

RECONFIGURABLE COMPUTING MODULAR SYSTEM

Shevelev S. S. – PhD, Associate Professor of the Department of Information Security, Southwestern State University, Kursk, Russia.

ABSTRACT

Context. Modern general purpose computers are capable of implementing any algorithm, but when solving certain problems in terms of processing speed they cannot compete with specialized computing modules. Specialized devices have high performance, effectively solve the problems of processing arrays, artificial intelligence tasks, and are used as control devices. The use of specialized microprocessor modules that implement the processing of character strings, logical and numerical values, represented as integers and real numbers, makes it possible to increase the speed of performing arithmetic operations by using parallelism in data processing.

Objective. To develop principles for constructing microprocessor modules for a modular computing system with a reconfigurable structure, an arithmetic-symbolic processor, specialized computing devices, switching systems capable of configuring microprocessors and specialized computing modules into a multi-pipeline structure to increase the speed of performing arithmetic and logical operations, high-speed design algorithms specialized processors-accelerators of symbol processing. To develop algorithms, structural and functional diagrams of specialized mathematical modules that perform arithmetic operations in direct codes on neural-like elements and systems for decentralized control of the operation of blocks.

Method. An information graph of the computational process of a modular system with a reconstructed structure has been built. Structural and functional diagrams, algorithms that implement the construction of specialized modules for performing arithmetic and logical operations, search operations and functions for replacing occurrences in processed words have been developed. Software has been developed for simulating the operation of an arithmetic-symbolic processor, specialized computing modules, and switching systems.

Results. A block diagram of a reconfigurable computing modular system has been developed, which consists of compatible functional modules, it is capable of static and dynamic reconfiguration, has a parallel structure for connecting the processor and computing modules through the use of interface channels. The system consists of an arithmetic-symbolic processor, specialized computing modules and switching systems, performs specific tasks of symbolic information processing, arithmetic and logical operations.

Conclusions. The architecture of reconfigurable computing systems can change dynamically during their operation. It becomes possible to adapt the architecture of a computing system to the structure of the problem being solved, to create problem-oriented computers, the structure of which corresponds to the structure of the problem being solved. As the main computing element in reconfigurable computing systems, not universal microprocessors are used, but programmable logic integrated circuits, which are combined using high-speed interfaces into a single computing field. Reconfigurable multipipeline computing systems based on fields are an effective tool for solving streaming information processing and control problems.

KEYWORDS: information graph, parallel pipeline structure, modular system, specialized computing units.

REFERENCES

1. Khoroshevskiy V. G. Arkhitektura vychislitel'nykh sistem: ucheb. Posobiye. Moscow, Izd-voMGU im. N. E. Baubana, 2008, 520 p.
2. Guzik V. F., Kalyayev I. A., Levin I. I. Rekonfiguriruyemye vychislitel'nyye sistemy. Taganrog, Izd-vo Yuzhnyy federal'nyy universitet, 2016, 472 p.
3. Shevelev S. S. Pat. Rossiyskaya Federatsiya, MPK G 06 F 15/76. Vychislitel'naya otkrytaya razvivayemaya asinkhronnaya modul'naya sistema VORAMS ; zayavitel' i patentoobladatel' Yugo-Zapadnyy gosudarstvennyy universitet. № 2453910; zayavl. 08.04.2009; opubl. 20.06.2012, Byul. № 17, 81 p.
4. Hockney R., Jesshope, K. Parallel computers. Architecture, programming and algorithms. Moscow, Radio and communication, 1986, 392 p.
5. Shevelev S. S. Computational open developing asynchronous modular system, *Problems of information security. Computer systems of the St. Petersburg State Polytechnic University*, 2013, No. 4, pp. 86–92.
6. Titenko E. A., Zerina I. S., Evsyukov V. S., Skorniyakov K. S., Tutov E. B. Pat. Russian Federation, IPC G 11C 15/00. A method of parallel search and replacement of a string and a homogeneous memory matrix for its implementation /; applicant and patent holder of the South-Western State University. No. 2012113755; application 06.04.2012; publ. 20.10.2013, Byul. no. 29
7. Shevelev S. S. Pat. Russian Federation, IPC G06F 17/30. Word sorting device ; applicant and patentee Kursk State Technical University. No. 2223538; declared 04/08/2002; publ. 10.02.2004, Bul. No. 4.
8. Timofeeva E. A., Paramonov N. B., Paramonov Yu. N.; applicant and patent holder E. A. Timofeeva, N. B. Paramonov, Yu. N. Paramonov Pat. Russian Federation, IPC G 06 F 17/50, G 06 N 7/08. A device for modeling queuing systems. No. 2447496; declared on 24.12.2010; published on 10.04.2012, Byul. No. 10
9. Petrenko L. P.; applicant and patent holder Petrenko L. P. Pat. Russian Federation, IPC G06F 7/50. Functional structure of the parallel sum of the ternary number system $f(+1,0,-1)$ in its position-sign implementation $f(+/-) / -$ No. 2008116450; declared on 29.04.2008; published on 10.11.2009.
10. Emelianova I. N., Efremov V. V. Pat. Russian Federation, IPC G06F 7/08. Information sorting system /; applicant and patent holder of the Kursk State Technical University of Higher Professional Education. No. 2004117660; declared on 09.06.2004; published on 10.01.2006.
11. Kalyayev I. A., Levin I. I., Semernikov E. A. Reconfigurable multiconveyor computational structures. Rostov n / a, Publishing house of the SSC RAS, 2008, 320 p.
12. Zheltov S. A. Adaptation of the Sherman-Lehman method for solving the factorization problem to the CUDA computing architecture, *Vestnik RSUH*, 2012, No. 14, pp. 84–91.
13. Knut D. E. The Art of Programming, Volume 3. Sorting and Searching, 2nd ed. : Per from English. Moscow, Publishing house "Williams", 2001, 832 p.