

УДК 004.31

БЛОК АССОЦИАТИВНОГО СОПРОЦЕССОРА НА БАЗЕ ПЛИС ДЛЯ СПЕЦИАЛИЗИРОВАННЫХ ВЫЧИСЛИТЕЛЬНЫХ СИСТЕМ

Мартышкин А.И.

*ФГБОУ ВО «Пензенский государственный технологический университет», Пенза,
e-mail: rector@penzgtu.ru*

В работе рассматривается возможность реализации блока ассоциативного сопроцессора на современной элементной базе для специализированных вычислительных систем. Целью статьи является разработка и исследование блока ассоциативного сопроцессора на базе ПЛИС, для специализированных вычислительных, например, многопроцессорных, систем, выполняющего ассоциативные функции и функции хранения данных. Объектом разработки и исследования данной статьи является ассоциативный сопроцессор на базе ПЛИС. Операция поиска широко используется в программах разного назначения, как пользовательских, так и системных. Однако эта операция является одной из наиболее трудоемких и требует больших затрат времени при реализации традиционным способом, когда искомые данные последовательно читаются из оперативной памяти машины в процессор, где над ними выполняют соответствующую операцию. В статье предлагается ассоциативный сопроцессор, подключаемый к шине PCI вычислительной системы, обеспечивающий операции поиска и сравнения на «больше – меньше» одновременно по 32-м словам, загруженным предварительно в ассоциативную память. В заключении сделаны соответствующие выводы по работе. Применение языка VHDL как универсального средства описания аппаратуры интегральных схем обеспечивает гибкость проекта и легкость при отладке функционирования устройства. Моделирование работы ассоциативного сопроцессора проводилось в САПР Web pack ISE фирмы Xilinx. Это позволяет производить оценку правильности работы аппаратного сопроцессора в составе вычислительных систем без построения реальных макетов. Эффективность внедрения разработанного блока обеспечивается за счет того, что сопроцессор выполняет трудоемкие операции поиска и сравнения данных, тем самым разгружает центральный процессор и увеличивает производительность вычислительной системы в целом.

Ключевые слова: блок, ассоциативная память, адресация, сопроцессор, вычислительная система, ячейка памяти, фиксация реакций, анализатор многократного совпадения, адресация памяти, шинный интерфейс, аппаратная реализация, приоритетный анализатор, цикл записи, цикл чтения

ASSOCIATIVE COPROCESSOR MODULE BASED ON FPGA FOR SPECIALIZED COMPUTER SYSTEMS

Martyshkin A.I.

Penza State Technological University, Penza, e-mail: rector@penzgtu.ru

The paper considers the possibility of implementing a module of the associative coprocessor by modern element base for specialized computer systems. The goal of the article is the development and study of the associative co-processor module on FPGA, for specialized computing, for example, multi-processor, systems, performing associative function and data storage functions. The object of research and development of this article is the associative coprocessor based on FPGA. The search operation is widely used in different purposes as users and systems. However, this operation is one of the most labor-intensive and requires time-consuming when implemented in the traditional way, when the search data is sequentially read from the machine RAM to the CPU, where they perform the appropriate operation. The paper proposes associative co-processor connected to the PCI bus of a computer system, providing search and comparison of «more – less» simultaneously in 32 of said pre-loaded in the associative memory. Finally, conclusions have been made. The use of the VHDL language as a universal means of hardware description of integrated circuits provides the flexibility of the project and the ease of debugging the operation of the device. Modeling the associative co-processor was carried out in CAD Web pack ISE of Xilinx company. This allows to evaluate the correct operation of the hardware coprocessor in the composition of computing systems without building the actual layout. Introduction efficiency of the module is assured by the fact that the coprocessor performs a time-consuming operation to search for and compare data and thus relieves the CPU and increases the performance of the computing system as a whole.

Keywords: module, associative memory, addressing, coprocessor, computer system, memory cell, fixation reactions, analyzer of multiple coincidence, memory addressing, bus interface, hardware implementation, priority analyzer, write cycle, read cycle

Основная область применения компьютеров сегодня – работа с большими объемами данных. Самыми трудоемкими операциями при такой работе являются всевозможные поиски и сортировки данных. Существующие вычислительные системы (ВС) используют адресную архитектуру памяти, т.е. для того, чтобы осуществить поиск данных

в памяти, необходимо произвести чтение по каждому адресу модуля памяти и сравнить его с аргументом поиска. В итоге на поиск информации уходит много машинного времени. Это обстоятельство негативно сказывается на быстродействии ВС в целом. Намного быстрее получать доступ к данным по ассоциации (содержанию) [1].

Постановка задачи

Настоящая статья в целом носит исследовательский характер. В ходе изучения предметной области был проанализирован ряд литературных источников [1, 2, 8] с целью поиска незатронутых и нерешенных проблем. Ряд вопросов, связанных с возможностью аппаратной реализации ассоциативного сопроцессора для быстрого поиска данных, не нашел должного отражения в публикациях, однако частично проблемные вопросы были рассмотрены в работах [3–7].

Целью настоящей статьи является разработка и исследование блока ассоциативного сопроцессора на базе ПЛИС для специализированных ВС, например, многосопроцессорных систем. Данный вопрос сегодня является актуальным ввиду глобальной информатизации и практически повсеместного оперирования колоссальными объемами данных. Для достижения поставленной цели в статье решаются задачи по определению структуры устройства и принципов его функционирования. Разрабатываемый аппаратный сопроцессор обладает возможностью адресного и ассоциативного доступа к хранящимся в памяти данным. Адресный доступ необходим для работы с конкретной записью и для использования библиотек тестов, разработанных для адресной памяти.

Устройство состоит из двух частей: основной, реализующей функции ассоциативного сопроцессора, а именно: адресная запись в ассоциативное запоминающее устройство (АЗУ); ассоциативная запись в АЗУ; адресное чтение из АЗУ; ассоциативное чтение из АЗУ; поиск совпадений и части сопряжения с ВС, на которую возлагается функция преобразования сигналов, идущих от центрального процессора (ЦП), в сигналы, с которыми будет работать сопроцессор, т.е. эта часть устройства организует интерфейс с ЦП.

Сегодня ассоциативный доступ реализуется двумя способами: программным, основанным на распределении памяти в зависимости от содержания данных и реализуемым с помощью программных средств и аппаратным, основанным на применении специальных аппаратных средств, предназначенных для хранения и ассоциативного поиска данных. Он может быть реализован в виде параллельного АЗУ, где аргумент поиска поступает во все ячейки памяти (ЯП) параллельно и поиск выполняется за один такт. Другим вариантом реализации является последовательно-поразрядное АЗУ, где поиск происходит поразрядно и время поиска зависит от количества разрядов шины данных.

Подключение устройства к ВС возможно несколькими способами [9]: прямое подключение к шине процессора; подключение к последовательному интерфейсу (USB); подключение к шине расширения компьютера (PCI). На основе упомянутых способов подключения блока к ВС определим архитектурно-структурные особенности сопроцессора, предварительно сравнив их с существующими аналогами. При прямом подключении к шине процессора блок придется включать в состав системной платы, что приведет к увеличению стоимости сопроцессора и его универсальности. Быстродействие такой ВС оказывается высоким. При подключении с помощью интерфейса USB сопроцессор будет представлен отдельным внешним устройством, но при этом работа с ним будет осуществляться последовательно, что снижает быстродействие. Зато такой блок сравнимо дешевле предыдущего. Подключение сопроцессора к шине PCI позволит реализовать его в сравнительно дешевой в реализации плате расширения. При этом работа с блоком будет происходить по параллельному интерфейсу, что позволяет достичь максимальной производительности по сравнению с аналогами. Проанализировав достоинства и недостатки способов организации основной части устройства и части сопряжения с ВС, было решено выполнить ее в виде параллельного АЗУ, т.к. этот способ обладает максимальной производительностью. Подключение к специализированной ВС реализуется по шине PCI, т.к. она имеет достаточно высокую пропускную способность.

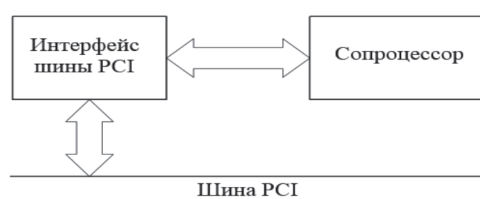
Определение структуры ассоциативного сопроцессора

Упрощенно структурную схему сопроцессора можно представить в виде двух блоков (рис. 1, а): сопроцессор (основная часть) и интерфейс шины PCI (часть сопряжения с системой). Проанализировав способ реализации основной части в составе сопроцессора, решено, что главным блоком является модуль памяти АЗУ, представляющий собой массив ЯП и выполняющий хранение данных и поиск ассоциаций с аргументом. ЯП состоит из запоминающего элемента, выполняющего хранение данных, и схемы сравнения, осуществляющей поиск, т.е. вырабатывающей сигналы, указывающие на равенство или неравенство содержимого ячейки аргументу (рис. 1, б).

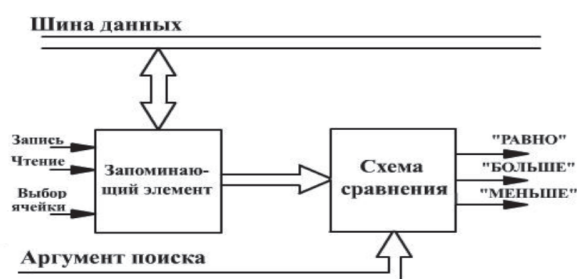
Для реализации функций записи и чтения данных из ЯП блока в состав сопроцессора нужно ввести мультиплексор и дешифратор адреса, вырабатывающий на основании адреса сигнал «Выбор ячейки» для конкретной ЯП. Также нужно вклю-

чить в состав сопроцессора модуль памяти фиксации реакций, для сохранения значений откликнувшихся ячеек. Для подсчета их количества в состав сопроцессора введен блок подсчета совпадений. Блок анализатора многократных совпадений (АМС) необходим для приоритетного выбора, по результатам работы которого вырабатываются сигналы для шифратора, необходимого для преобразования двоичной последовательности в адрес, при ассоциативном

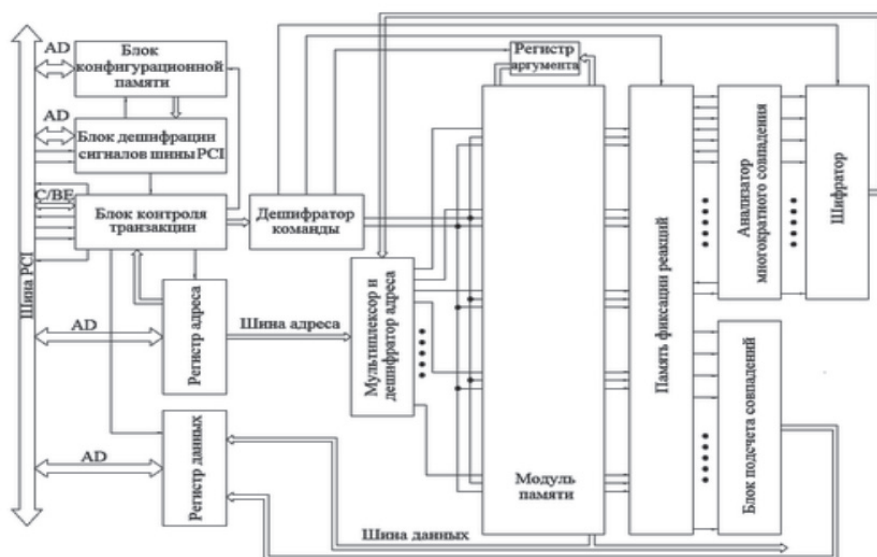
чтении или записи поступающий на входы селектора адреса. Для хранения аргумента поиска имеется регистр аргумента. Управляет работой сопроцессора блок дешифрации команд, вырабатывающий управляющие сигналы. Из состава сопроцессора исключен регистр маски, т.к. схема сравнения вырабатывает три сигнала («РАВНО», «БОЛЬШЕ» и «МЕНЬШЕ») и надобность в маскировании бит аргумента поиска отпадает (рис. 1, в).



а)



б)



в)

Рис. 1. Упрощенная структурная схема ассоциативного сопроцессора (а); структурная схема ячейки памяти (б); подробная структурная схема ассоциативного сопроцессора (в)

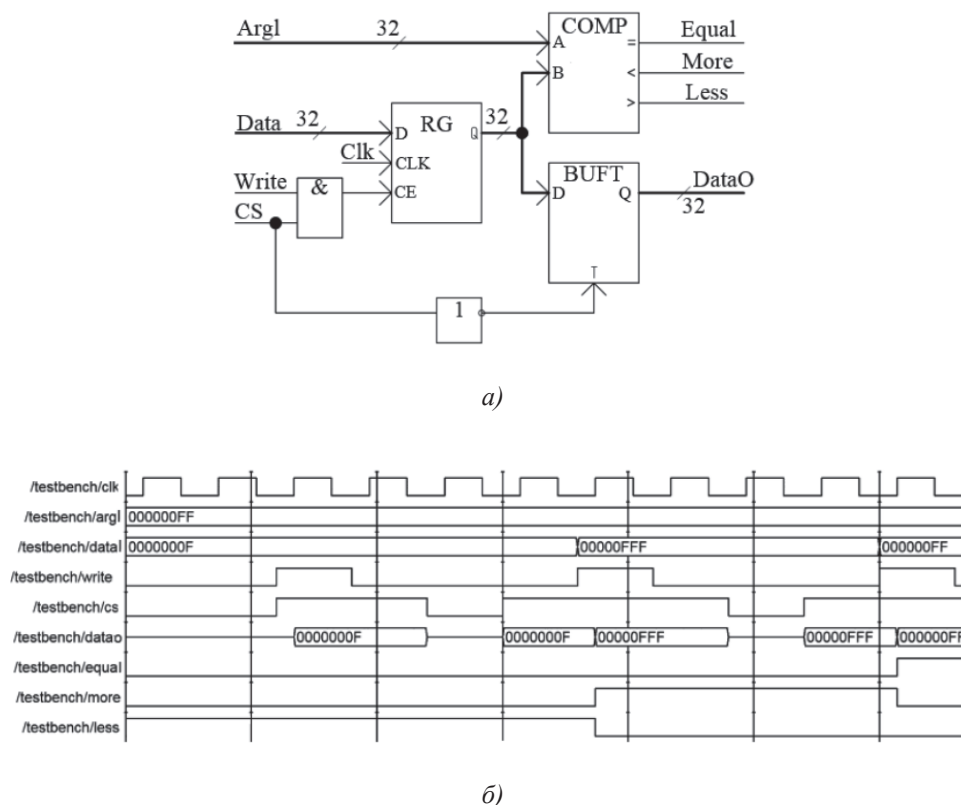
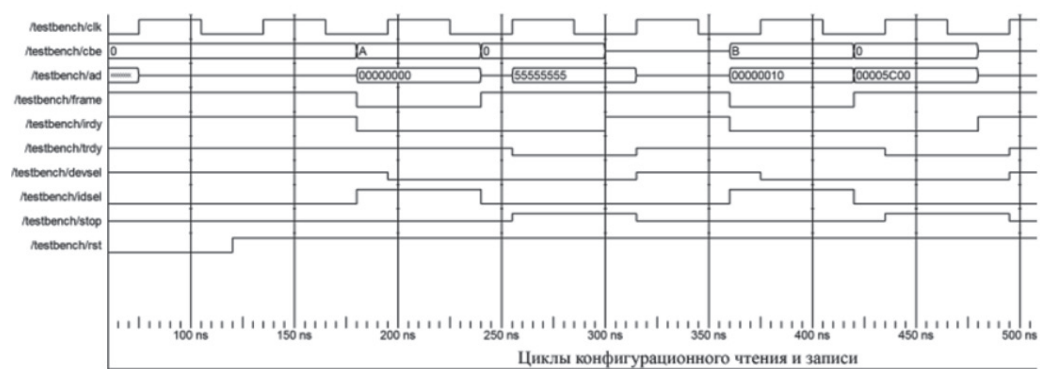


Рис. 2. Функциональная схема ЯП (а); временные диаграммы работы ЯП (б)

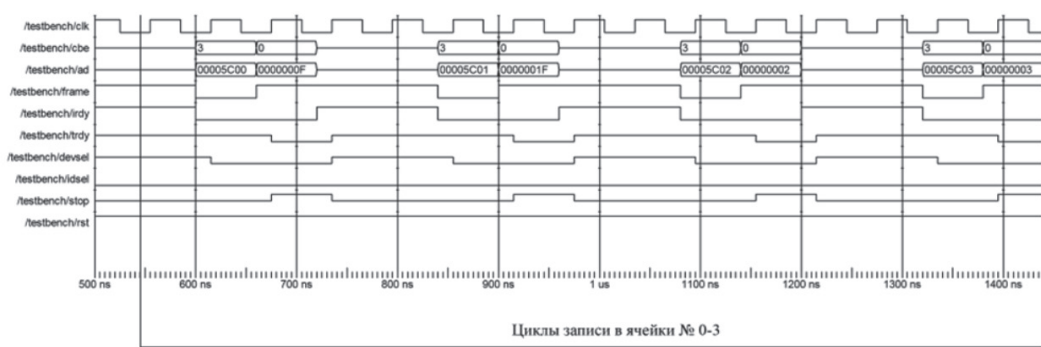
На основании описания шины PCI, в состав шинного интерфейса добавлен блок конфигурационной памяти, для реализации функций автоконфигурирования и Plug and Play. Блок контроля транзакции нужен для анализа прохождения транзакции и вырабатывания соответствующих сигналов шины PCI и шины команд сопроцессора. Блок дешифрации сигналов шины PCI отвечает за выработку сигнала ответа устройства при обращении к нему. Так как шина PCI мультиплексированная, то адрес ячейки, к которой осуществляется обращение, должен фиксироваться на регистре адреса. Для фиксации данных читаемых или записываемых в устройство необходим регистр данных (рис. 1, в).

Основой ассоциативного сопроцессора является память, представляющая собой массив ЯП. Запоминающий элемент ЯП реализован на параллельном регистре, являющемся массивом D-триггеров, обеспечивающих максимальную производительность и минимум логики, необходимой для обеспечения запоминания информации. Основными сигналами для регистра являются: 32-разрядный сигнал *D*, по которому поступают данные, сигнал *CE*, и 32-разрядный

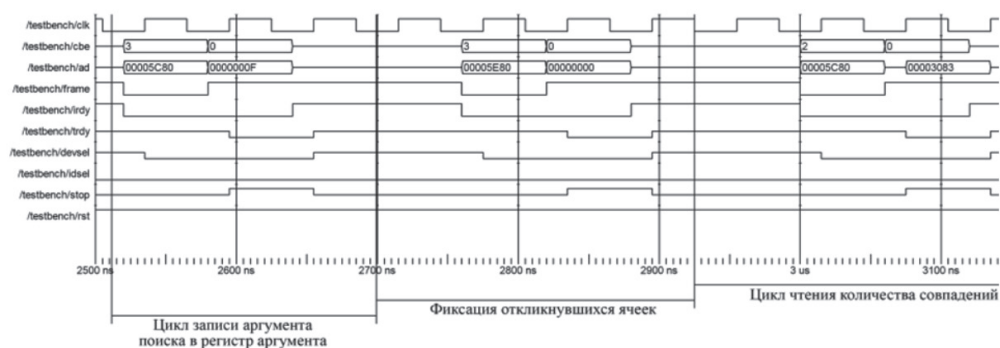
сигнал *Q*, с которого данные, хранящиеся в регистре, считываются. Схема сравнения реализована на базе компаратора. Основными сигналами для него являются: 32-разрядные сигналы *A* и *B*, на которые подаются аргументы для сравнения, сигналы *=*, *<* и *>*, с которых считываются результаты сравнения. В состав ячейки памяти включен буферный элемент для отключения выходной шины данных ЯП от общей выходной шины. Основными сигналами буферного элемента являются: 32-разрядный сигнал *D*, 32-разрядный сигнал *Q* и сигнал *T*. Функциональная схема ЯП приведена на рис. 2, а. Принцип работы ЯП следующий. На вход *ArgI* подается аргумент поиска. Через вход *DataI* данные попадают в ЯП. С выхода *DataO* данные считываются из ЯП. Входы *Write* и *CS* служат для управления работой ЯП. Сигналом *CS* ЯП выбирается, т.е. буферный элемент BUFT пропускает сигналы с выхода регистра RG на выходную шину *DataO*. По единичному сигналу *Write* данные с входа *DataI* записываются в RG. На выходах *Equal*, *More* и *Less* формируются сигналы «Равно», «Больше» и «Меньше» соответственно.



а)



б)



в)



г)

Рис. 3. Временные диаграммы работы ассоциативного сопроцессора: конфигурирование устройства (а); запись в ячейки памяти (б); запись в аргумент поиска, фиксация совпадений и чтение количества совпадений (в); чтение из ячеек памяти, содержимое которых больше и меньше аргумента (г)

Временные диаграммы работы ЯП, подтверждающие ее работоспособность, приведены на рис. 2, б. Здесь показана запись ряда значений: «F», «FFF» и «FF». Аргументом поиска является «F». Из полученных временных диаграмм видно: пока сигнал CS равен логическому нулю, выходная шина (DataO) находится в третьем состоянии, т.е. отключена. При подаче на вход CS логической единицы с выходной шины данных (DataO) можно считывать значения, хранящиеся в RG. Запись в ячейку осуществляется подачей логической единицы на вход Write. Из рис. 2, б видно, что как только в ЯП записывается новое значение на выходах Equal, More и Less устанавливаются результаты поиска.

Вычислительный эксперимент

Работа сопроцессора начинается с включения питания или с аппаратного сброса.

После сброса устройство не отвечает на обращение к пространству ввода/вывода. Начинается его конфигурирование с помощью циклов конфигурационной записи/чтения. По завершению конфигурирования устройство готово к работе.

Временные диаграммы работы ассоциативного сопроцессора приведены на рис. 3, а–г).

На рис. 3, а показаны временные диаграммы циклов конфигурирования. В первом цикле, который занимает два такта, происходит конфигурационное чтение из ячейки конфигурационной памяти с адресом 0h. По шине CBE передается команда конфигурационного чтения «1010» в шестнадцатеричной системе счисления это А.

На рис. 3, б представлены временные диаграммы циклов записи в ячейки АП. Цикл записи состоит из фазы адреса и фазы данных.

На рис. 3, в показаны циклы записи аргумента, фиксации совпадений и чтение количества совпадений. Цикл записи аргумента поиска ничем не отличается от записи в ячейку памяти. На него также необходимо два такта. Цикл фиксации совпадений начинается с адреса 5E80h. Цикл чтения количества совпадений начинается с адреса идентичного цикла записи аргумента. Разница состоит в том, что цикл чтения количества начинается с команды чтения из порта ввода/вывода, а цикл записи аргумента – с команды записи в порт ввода/вывода.

На рис. 3, г представлены циклы чтения ячеек памяти, содержимое которых больше и меньше аргумента. На один цикл чтения необходимо три такта. Это связано с тем, что необходим такт на чтение адреса откликнувшейся ячейки.

Рассматриваемый аппаратный ассоциативный сопроцессор реализован на ПЛИС фирмы Xilinx. Разработаны VHDL-коды устройства. Опытным путем получено, что применение аппаратного сопроцессора (при выполнении операций поиска и сравнения) позволяет повысить производительность ВС в среднем примерно на 25% по сравнению с системами, включающими только лишь традиционные процессоры. Иными словами, ВС с ассоциативным сопроцессором показывает в 1,25 раза большую производительность, чем аналоги. Отсюда следует целесообразность применения данной разработки на практике. Плата расширения с блоком ассоциативного сопроцессора может быть использована для повышения производительности серверов баз данных типа Oracle и SQL Server, расположенных на машинах x86, а также для обработки графики, при этом применение платы расширения должно быть экономически оправдано по критерию «стоимость – эффективность».

Выводы

В статье разработана структура блока ассоциативного сопроцессора на базе ПЛИС. Рассматриваемое устройство физически можно реализовать в виде платы расширения ВС, подключаемой по интерфейсу PCI. Разрядность шины данных описанного в статье сопроцессора составляет 32 бита (двойное машинное слово). Емкость массива АП составляет 32 двойных машинных слова. Ассоциативный сопроцессор реализован аппаратно, что позволяет выполнять трудоемкие операции поиска и сравнения, тем самым разгружая центральный процессор и увеличивая производительность ВС в целом. Работоспособность устройства и отдельных его частей проверена тестированием и отладкой синтезированных VHDL-кодов.

Работа выполнена при финансовой поддержке РФФИ (грант № 16-07-00012).

Список литературы

1. Бикташев Р.А., Князьков В.С. Архитектура параллельных вычислительных систем. Пенза: РИО Упринформпечати, – 1993. – 166 с.
2. Кохонен Т. Ассоциативные запоминающие устройства: Пер. с англ. – М.: Мир, 1982. – 384 с.
3. Мартенс-Атюшев Д.С., Мартышкин А.И. Разработка реконфигурируемой вычислительной системы для цифровой обработки сигнала // Технические науки – от теории к практике. – 2015. – № 52. – С. 50–58.
4. Мартышкин А.И. Разработка аппаратного буферного устройства памяти многопроцессорной системы // Фундаментальные исследования. – 2015. – № 12–3. – С. 485–489.
5. Мартышкин А.И. Реализация аппаратного буфера памяти многопроцессорной системы // Труды XII Международной научно-технической конференции «Новые информационные технологии и системы» (г. Пенза, 17–19 ноября 2015 г.). – Пенза: ПГУ, 2015. – С. 96–99.

6. Мартышкин А.И. Математическое моделирование аппаратного буфера памяти многопроцессорной системы // Сборник материалов XII Международной научно-технической конференции «Опτικο-электронные приборы и устройства в системах распознавания образов, обработки изображений и символической информации. Распознавание-2015» (г. Курск, 12–16 мая 2015 г.). – Курск: Юго-Зап. гос. у-нт, 2015. – С. 247–249.

7. Мартышкин А.И. Разработка и исследование математических моделей подсистемы «процессор-память» многопроцессорных вычислительных систем архитектур NUMA и SUMA на разомкнутых сетях массового обслуживания // Вестник Рязанского государственного радиотехнического университета. – 2016. – № 55. – С. 55–63.

8. Таненбаум Э., Бос Х. Современные операционные системы. – СПб.: Питер, – 2015. – 1120 с.

9. Цилькер Б.Я., Орлов С.А. Организация ЭВМ и систем (2-е изд.). – СПб.: Питер, 2011. – 688 с.

References

1. Biktashev R.A., Knjazkov V.S. Arhitektura paralelnyh vychislitelnyh sistem. Penza: RIO Uprinformpechat, 1993. 166 p.

2. Kohonen T. Associativnye zapominajushhie ustrojstva: Per. s angl. M.: Mir, 1982. 384 p.

3. Martens-Atjushev D.S., Martyshekin A.I. Razrabotka rekonfiguriruemoj vychislitelnoj sistemy dlja cifrovoy obrabotki

signala // Tehnicheskie nauki ot teorii k praktike. 2015. no. 52. pp. 50–58.

4. Martyshekin A.I. Razrabotka apparatnogo buferного ustrojstva pamjati mnogoprocessornoj sistemy // Fundamentalnye issledovaniya. 2015. no. 12–3. pp. 485–489.

5. Martyshekin A.I. Realizacija apparatnogo bufera pamjati mnogoprocessornoj sistemy // Trudy XII Mezhdunarodnoj nauchno-tehnicheskoy konferencii «Novye informacionnye tehnologii i sistemy» (g. Penza, 17–19 nojabrja 2015 g.). Penza: PGU, 2015. pp. 96–99.

6. Martyshekin A.I. Matematicheskoe modelirovanie apparatnogo bufera pamjati mnogoprocessornoj sistemy // Sbornik materialov XII Mezhdunarodnoj nauchno-tehnicheskoy konferencii «Optiko-jelektronnye pribory i ustrojstva v sistemah raspoznavaniya obrazov, obrabotki izobrazhenij i simvolnoj informacii. Raspoznavanie-2015» (g. Kursk, 12–16 maja 2015 g.). Kursk: Jugo-Zap. gos. u-nt, 2015. pp. 247–249.

7. Martyshekin A.I. Razrabotka i issledovanie matematicheskikh modelej podsistemy «processor-pamjat» mnogoprocessornyh vychislitelnyh sistem arhitektur NUMA i SUMA na razomknutyh setjah massovogo obsluzhivaniya // Vestnik Rjazanskogo gosudarstvennogo radiotekhnicheskogo universiteta. 2016. no. 55. pp. 55–63.

8. Tanenbaum Je., Bos H. Sovremennye operacionnye sistemy. SPb.: Piter, 2015. 1120 p.

9. Cilker B.Ja., Orlov S.A. Organizacija JeVM i sistem (2-e izd.). SPb.: Piter, 2011. 688 p.