

УДК 004. 27; 004.272; 004.475; 004.315.5

Ю.С. ЯКОВЛЕВ\*, А.А. ТИМАШОВ\*

**О ПРИМЕНЕНИИ КОМПЬЮТЕРНЫХ СИСТЕМ С КОЛЬЦЕВЫМИ ШИНАМИ ДЛЯ ПОЛУНАТУРНОГО МОДЕЛИРОВАНИЯ В РЕАЛЬНОМ МАСШТАБЕ ВРЕМЕНИ**

\*Институт кибернетики имени В.М. Глушкова НАН Украины, г. Киев, Украина

**Анотація.** У зв'язку з тим, що закордонними і вітчизняними вченими недостатньо уваги приділено якійсь і кількісній оцінці високопродуктивних комп'ютерних систем з кільцевими шинами, використовуваних для напівнатурного імітаційного моделювання складних динамічних процесів управління, особливо в умовах реального часу, то це завдання є актуальним. Актуальність створення таких систем також обумовлюється складністю керованих об'єктів і процесів, що безперервно росте, з одночасним скороченням часу, відведеного особою, що приймає рішення (ЛПР), на аналіз проблемної ситуації і ухвалення необхідних дій, що управляють. У роботі розглянуті основні елементи напівнатурного моделювання, а також запропоновані моделі і методи порівняльної оцінки трьох варіантів високопродуктивних комп'ютерних систем із кільцевими шинами (КШ), що відрізняються способами розпаралелювання алгоритму користувача в порівнянні з його послідовною реалізацією. При цьому для кожного типу КШ побудовані графіки залежності продуктивності від параметрів вірогідності оптимального завантаження процесорів  $P_i$  і фрагментів алгоритму, що підлягають розпаралелюванню. Приведені графіки залежності показали, що найбільш престижною для позначених вище значень параметрів виявилася інтелектуальна система пам'яті з однією кільцевою шиною, яка прийнята як базовий варіант, оскільки вона має найменшу кількість тактів процесора при реалізації алгоритму користувача в порівнянні з його послідовною реалізацією. Найбільш універсальний варіант – це інтелектуальна система пам'яті з секціонованими модулями на ПЛІС, оскільки розробник має можливість вибирати кількість секцій і секторних наборів кожної секції, виходячи з конкретного завдання розпаралелювання, використовуючи при цьому сучасну елементну базу (наприклад, ПЛІС з PCI – Express).

**Ключові слова:** комп'ютерні системи з кільцевими шинами, засоби обчислювального процесу, напівнатурне моделювання і управління, оцінка продуктивності.

**Аннотация.** В связи с тем, что заграничными и отечественными учеными недостаточно внимания уделено качественной и количественной оценке высокопроизводительных компьютерных систем с кольцевыми шинами, используемых для полунатурного имитационного моделирования сложных динамических процессов управления, особенно в условиях реального времени, то эта задача является актуальной. Актуальность создания таких систем также обуславливается непрерывно растущей сложностью управляемых объектов и процессов с одновременным сокращением времени, отведенного лицом, принимающим решения (ЛПР), на анализ проблемной ситуации и принятие необходимых управляющих действий. В работе рассмотрены основные элементы полунатурного моделирования, а также предложены модели и методы сравнительной оценки трех вариантов высокопроизводительных компьютерных систем с кольцевыми шинами (КШ), отличающихся способами распараллеливания алгоритма пользователя по сравнению с его последовательной реализацией. При этом для каждого типа КШ построены графики зависимости производительности от параметров вероятности оптимальной загрузки процессоров  $P_i$  и фрагментов алгоритма, подлежащих распараллеливанию. Приведенные графики зависимости показали, что наиболее престижной для обозначенных выше значений параметров оказалась интеллектуальная система памяти с одной кольцевой шиной, которая принята в качестве базового варианта, так как она имеет наименьшее количество тактов процессора при реализации алгоритма пользователя по сравнению с его последовательной реализацией. Наиболее универсальный вариант – это интеллектуальная система памяти с секционированными модулями на ПЛИС, так как разработчик

имеет возможность выбирать количество секций и секторных наборов каждой секции, исходя из конкретной задачи распараллеливания, используя при этом современную элементную базу (например, ПЛИС с PCI – Express).

**Ключевые слова:** компьютерные системы с кольцевыми шинами, средства вычислительного процесса, полунатурное моделирование и управление, оценка производительности.

**Abstract.** Due to the fact that foreign and domestic scientists do not pay sufficient attention to qualitative and quantitative estimation of highly-efficient computer systems with the ring buses, difficult dynamic managerial processes used for HIL imitating modelling especially in the conditions of real time is an actual problem. The urgency of creation of such systems also is caused by continuously growing complexity of operated objects and processes with simultaneous reduction of time which has been taken away by the person making decision (PMD) on the analysis of a problem situation and acceptance of necessary operating actions. The paper considers basic elements of HIL modeling. Moreover, the models and methods of a comparative estimation of three variants of highly-efficient computer systems with ring buses (RB), differing by the way of paralleling the algorithm of the user in comparison with its consecutive realization. Thus for each type of RB the dependency diagrams of productivity from parameters of probability of optimum loading of  $P_i$  processors and algorithm fragments parallelization are constructed. The given dependency diagrams showed that the most prestigious for the above values of parameters was an intelligent memory system with one ring bus, which was adopted as the base version, since it has the least number of processor cycles in the implementation of the user algorithm compared to its sequential implementation. The most universal variant is an intellectual system of memory with the partitioned modules on FPLD as the developer has the possibility to choose quantity of sections and sector sets of each section, due to a specific target of paralleling, using modern element base (for example, FPLD with PCI – Express).

**Keywords:** computer systems with ring buses, means of computing process, HIL modelling and management, a productivity estimation.

## 1. Введение

Полунатурное моделирование представляет собой подход к исследованию устройства, реализованного в виде системы из натуральных элементов, которые сообщаются с другой частью системы, реализованной в виде модели. Создание аппаратных блоков, которые максимально повторяют элементы готового изделия и управляются моделью с помощью компьютера. Это позволяет исследовать и оценивать влияние факторов, моделировать которые невозможно или нецелесообразно. Программная модель при этом позволяет исследовать множественное число разных алгоритмов работы и обработки информации. Полунатурное моделирование – это моделирование, при котором оператор использует реальные органы управления и средства отображения информации, его рабочее место сконструировано аналогично реальному. Однако уравнения, которые описывают состояние технической части (СТЧ), моделируются средствами вычислительной техники. Это позволяет оперативно и в широких границах менять параметры технической части системы, задавать разные алгоритмы работы оператора и тем самым проводить исследование в различных условиях.

## 2. Состояние разработанной проблемы

Принципы полунатурного моделирования возникли при создании современных бортовых встроенных систем реального времени (ВСРВ) для управления авиационными ракетами воздух – воздух, воздух – земля. ВСРВ являются многомашинными вычислительными комплексами. Количество каналов связи между приборами в составе ВСРВ достигает нескольких десятков. На каждой ВСРВ работает специфический набор приложений и системных задач. На ранних этапах разработки аппаратные и программные компоненты ВСРВ разрабатываются параллельно. На этих этапах для разработчиков программного обеспечения доступна лишь часть прототипов аппаратных устройств. Это создает множе-

ство проблем разработчикам ВС РВ, которые возможно решить с помощью полунатурного моделирования.

Идея перенесения отдельных фрагментов программного обеспечения на реализацию с помощью дополнительных аппаратных средств сама по себе не новая, и ею занимается большое количество ученых России, Америки и стран Европы. Среди них следует выделить академика В.М. Глушкова, который в 50-х и 60-х годах прошлого века при создании ЭВМ «Украина» и серии МИР предложил аппаратную поддержку машинного языка высокого уровня. Среди других ученых можно выделить таких: D. Newcomb, R. Roy, A. Rip, J. Conway, K. Ashley (США), L.P. Goodstein – (Денмарк), R. Asmundis (Италия), M. Gomez (Германия), Е.А. Федосов, Д.А. Поспелов, О.И. Ларичев (Россия) и др. [1, 2].

Что касается алгоритмов разделения между процессорами программы пользователя, можно отметить, что в известных решениях зарубежных авторов они, как правило, основаны на большом количестве итерационных процессов и используются на отдельных этапах процессов с вмешательством пользователя, который приводит к увеличению времени процесса разделения и отсутствия возможности автоматической оптимизации загрузки каждого процессора. Кроме того, программа разделения размещается или в рабочей ЭВМ, или в дополнительной ЭВМ такого же класса, что, в свою очередь, повышает стоимость системы в целом (Gordy, Robert Stephen, Terry, Dwight Anthony (США), Kihara, Toshiaki, Katsuo, Kenichi (Япония) и др.) [3, 4].

Что касается средств сбора, предыдущей обработки, передачи и отображения данных, необходимых для работы системы, то на основе анализа заграничных публикаций можно сделать такие выводы:

- отсутствуют комплексные методы построения типовых структур аппаратно-программных средств и способы использования типовых контроллеров, которые бы позволяли быстро строить, разворачивать, налаживать и реконфигурировать подсистемы полунатурного моделирования, особенно под краткосрочные задачи;

- отсутствует общая концепция комплексной виртуализации средств и подсистем как перспективного способа быстрого построения сложных систем полунатурного моделирования под конкретные краткосрочные задания (или задания, которые динамически изменяются), а также основные концептуальные положения создания технических средств для управления сбором, накоплением, предварительной обработкой, отображением, оптимальным хранением и передачей информации (в частности, беспроводной) в распределенных системах с виртуальными компонентами.

Исследования, проведенные в области применения адаптивных человеко-машинных интерфейсов для систем, показали, что заграничными и отечественными учеными недостаточно внимания уделяется проблеме предоставления конкретному пользователю комфортных условий работы с системой, которая создает трудности при работе и что не может не отразиться на эффективности их приложения (А.М. Довгялло, Т.А. Гаврилова, В. Денинг, В.Ф. Ходаков, А. Kobsa, В. Kules и др.) [5, 6].

Также недостаточно внимания заграничными и отечественными учеными уделено созданию концепции формирования и применения знаний при полунатурном моделировании сложных динамических процессов управления принципов построения, методов и новых алгоритмов исключения знаний за счет полунатурного имитационного моделирования сложных динамических процессов управления.

Поэтому решение задачи полунатурного моделирования, особенно в условиях реального времени, является задачей актуальной.

Повышение производительности и сокращение сроков создания сложных объектов с помощью систем полунатурного моделирования и качества принятия решений в реальном масштабе времени может быть реализовано в таких средах:

- непосредственно в среде моделирования [7, 8];

– в распределенной аппаратурной среде, которая содержит большое количество процессоров за счет максимального распараллеливания программы пользователя и оптимизации загрузки каждого процессора. При этом отдельные фрагменты алгоритма задания пользователя могут быть реализованы с помощью аппаратурных средств, работа которых может выполняться одновременно с работой программных средств, дополнительно повышая тем самым производительность системы в целом [9–12];

– в среде сбора, предварительной обработки, накопления, отображения и передачи данных, необходимых для работы системы, за счет оптимизации формирования потоков данных, исключения немотивированного дублирования и избыточности данных, обеспечения их полноты и целостности, применения средств и контроллеров для предварительной их обработки [13–16];

– в среде взаимодействия пользователя с системой через интеллектуальные человеко-машинные интерфейсы, которые обеспечивают комфортное взаимодействие и мгновенную реакцию системы в зависимости от принятия решений и действий пользователя [17–19].

### **3. Цель и основные задания, идея исследований**

Основная цель работы – предварительно оценить производительность компьютерных систем с кольцевыми шинами (КШ) и средств вычислительного процесса для систем полунатурного моделирования и управления сложными объектами.

Основными заданиями полунатурного моделирования являются проведение исследований и оценка технических решений в области структуры встроенных вычислительных систем реального времени; оценка характеристик приборов и их программного обеспечения; комплексная отработка оборудования при решении заданий информационного взаимодействия и прикладных задач встроенных вычислительных систем реального времени; проверка работоспособности аппаратуры, в том числе проверка соответствия функционирования приборов и каналов передачи данных требованиям технического задания.

*Цель исследования* – перенесение реализации одной части алгоритма испытания задачи пользователя на встроенные или присоединенные к системе аппаратно-программные средства, а второй части – на реальный сложный объект для сокращения времени испытаний и повышения производительности решения некоторых задач и улучшения качественных показателей работы системы. Это обеспечивает возможности принятия решений в реальном масштабе времени за счет оптимизации вычислительного процесса и процессов сбора, предварительной обработки, отображения и передачи данных, взаимодействия пользователя с системой, а также возможности параллельного выполнения аппаратных и программных частей алгоритма задачи пользователя.

### **4. Актуальность**

Актуальность создания таких систем обуславливается непрерывно растущей сложностью управляемых объектов и процессов с одновременным сокращением времени, отведенного лицом, принимающим решения (ЛПР), на анализ проблемной ситуации и принятие необходимых управляющих действий.

### **5. Основной материал**

При разработке больших информационно измерительных и управляющих систем целесообразно сначала наладить взаимодействие разных блоков системы друг с другом. Для этого применяется аппаратно-программное моделирование, при котором блоки максимально реализуются в виде моделей, а реализация их взаимосвязей осуществляется с помощью специального оборудования, которое имитирует их внешние сигналы и интерфейсы. В ос-

нове предложенного подхода лежит разделение устройства на аппаратные и модельные блоки. В аппаратном виде реализуются только те блоки, работу которых необходимо исследовать. Другие реализуются в виде универсальных моделей, управляемых с помощью компьютера. Это позволяет сократить расходы на модернизацию модулей при плохих результатах испытаний. Также возможно проведение исследования нескольких алгоритмов обработки информации одновременно.

Применение комплексного моделирования заключается в следующем. Сначала создается программно-аппаратный комплекс. Этот комплекс должен обеспечить с помощью методов имитационного моделирования и отработки функционирование встроенных систем реального времени (ВСРВ) и отдельных ее блоков.

Потом, по мере отработки функционирования комплексов и их реализации в виде натуральных образцов, элементы заменяют соответствующими натурными образцами, без изменения других элементов комплекса. Такой процесс разработки позволяет минимизировать количество ошибок, которые оказываются на этапах натуральных испытаний.

Предлагаются технологии полунатурного моделирования и быстрого прототипирования. Это обусловлено высокой скоростью развития электроники и жесткими требованиями к условиям разработки новых устройств. Традиционно при разработке устройств разработчик макетировал разработанные изделия для оценки достоверности принятых схемотехнических решений, исследований датчиков, алгоритмов работы устройств, обработки информации и других параметров. Однако постоянно растущая сложность устройств и используемых в них компонентов снижает эффективность макетирования. Особенно это проявляется при разработке инновационных устройств, при которых нужно проводить большое количество исследований и экспериментов. Часто даже незначительное изменение, внесенное разработчиком, вызывает необходимость изготовления нового макета, а каждый вариант алгоритма работы требует описания его на языке используемого микроконтроллера или ПЛИС. Применение технологий полунатурного моделирования и быстрого прототипирования позволяет избежать множественных трудностей и повысить эффективность процесса разработки.

Особенности проектирования средств организации вычислительного процесса определяются, в основном, особенностями современной элементной базы. Основное назначение полунатурного моделирования при применении его к сложным объектам – это помощь лицам, которые проводят испытание и принимают решение. Наряду с микропроцессорами разработчик имеет дело с функциональными расширителями и контроллерами, со схемами программируемой логики на ПЛИС.

Встроенные системы реального времени активно используют для управления большими сложными техническими объектами, такими как транспортные средства (морские, воздушные, наземные, космические), здания и сооружения и др.

Основными компонентами встроенных систем реального времени (ВСРВ) являются:

- система датчиков (далее – периферия), возможно, со средствами первичной обработки собранных ими данных;
- вычислительная система – для обработки информации от датчиков и выработки управляющих влияний на средства управления, в том числе с кольцевыми шинами (КШ) [20–25];
- средства управления – сюда могут входить, кроме собственно органов управления, и средства визуализации состояния управляемого объекта, например, если в контуре управления задействован человек;
- окружение информационного обмена между датчиками, вычислительной системой и средствами управления.

В зависимости от степени готовности приборов встроенные вычислительные системы реального времени могут полностью использовать программные модели всех приборов или комплексы из натуральных образцов приборов и программных моделей приборов, собранных в единственный стенд и соединенных через аппаратные каналы интерфейсов. Средства разработки моделей позволяют быстро создавать модели, имитирующие «окружение» прибора, который налаживается, и взаимодействуют с ним через аппаратные каналы передачи данных.

Решение этих задач позволяет с минимальными расходами для многих определенных отраслей применения строить системы, которые могут работать в реальном масштабе времени. Поэтому тема статьи является своевременной и актуальной.

Программное обеспечение полунатурного моделирования должно включать набор программных инструментов, которые обеспечивают решение задач, связанных с моделированием: создание имитационных моделей приборов, которые встроены в вычислительные системы реального времени, а также вспомогательных моделей (например, модели внешней среды); организация взаимодействия моделей, разработка набора моделей, их взаимодействие с аппаратурой в модельном и реальном времени по бортовым каналам с возможностью внесения отказов в каналы; управление процессом моделирования в диалоговом режиме или выполнение автономного эксперимента без участия оператора; оперативное отображение результатов моделирования в графическом и табличном виде.

## **6. Содержание, основные требования к выполнению работ, уровня и способов их выполнения**

При создании комплекса полунатурного моделирования одним из наиболее сложных и важных заданий есть определение физически реализованных элементов. В виде моделей реализуются элементы создаваемого устройства, кроме тех, моделирование которых нецелесообразно. После определения блоков, которые реализуются в виде моделей и устройств, нужно выбрать аппаратные и программные средства, с помощью которых данные блоки будут реализовывать, а также определять способы взаимодействия между ними:

- уровень 1 – виртуальных датчиков, приборов, средств управления, средств отображения информации, обработки и т.д., что построено в виде специализированных модулей на микроконтроллерах (МК) и/или на ПЛИС, самостоятельных или встраиваемых, которые обеспечивают работу непосредственно с объектами мониторинга (управление), а также взаимодействуют с пользователями определенных профессий, занятых обслуживанием данных объектов и оборудования;

- уровень 2 – виртуальных подсистем, в том числе распределенных подсистем, предназначенных для решения конкретных типовых заданий пользователя для заданной предметной области с комплексным общим использованием разных виртуальных средств и элементов других подсистем;

- уровень 3 – типовых виртуальных рабочих мест пользователей на базе совместно используемых виртуальных средств и подсистем разных уровней;

- уровень 4 – типовых виртуальных рабочих (информационных) пространств пользователя (индивидуально или для рабочей группы), которые используют общие данные подсистем других уровней.

Количество уровней, их характеристики, аппаратно-программные средства и подсистемы, которые применяются, должны определяться при разработке конкретной системы (или при реконфигурации типовой системы под определенные задания).

К аппаратно-программным средствам с соответствующими программными частями, которые можно использовать при полунатурном моделировании, могут быть отнесены:

- аппаратно-программные средства для создания моделей на основе ПЛИС для полунатурного моделирования процессов и объектов;

- технические и программные (аппаратно-программные) средства для сбора, предварительной обработки, отображения и передачи данных (в том числе беспроводной) с использованием микроконтроллеров, ПЛИС и другой современной элементной базы;
- технические и программные средства, которые аппаратно реализуют некоторые фрагменты алгоритма распараллеливания программы пользователя;
- интеллектуальные человеко-машинные интерфейсы для настройки, контроля и оптимизации рабочего пространства пользователя и/или контроля и оптимизации процесса полунатурного моделирования;
- архитектурно структурные, схемотехнические и программные решения для специализированных узлов и средства виртуализации компонентов и подсистем разных уровней с применением современной элементной базы.

Анализ отечественных и зарубежных публикаций и отображенных в них выполненных проектов типа систем полунатурного моделирования [26, 27] показал, что такого подхода к созданию с использованием средств аппаратно-программной поддержки одновременно на всех уровнях системы и широкого применения виртуализации компонентов с целью оптимизации их работы в реальном времени, не существует.

Принимая во внимание, что набор предложенных аппаратно-программных средств обеспечивает возможности создания с небольшими затратами типовых систем полунатурного моделирования, которые работают в реальном времени в разнообразных отраслях, можно утверждать, что материалы данной статьи являются актуальными.

## **7. Сравнительная оценка производительности компьютерных систем с кольцевыми шинами (КШ), используемых для полунатурного моделирования**

Для сравнительной оценки компьютерных систем с кольцевыми шинами (КШ) необходимо принять следующие исходные положения:

- в качестве базового варианта целесообразно принять интеллектуальную систему памяти с модифицируемой кольцевой шиной (рис. 1) [20, 23];
- все секторные наборы (СН) с целью сравнительной оценки идентичны по составу узлов и блоков их функциональному назначению и содержат активных секторных наборов  $n$  для иерархических систем памяти [21, 24], а также систем памяти с секторными наборами на ПЛИС [22, 25];
- принимаются во внимание только все активные операции над данными (формирование адреса, обращение процессора к памяти за данными, непосредственное выполнение процессором операций, запоминание результатов выполнения операций в банке памяти и выдача результатов), при этом все подготовительные операции, такие, как распределение фрагментов алгоритма по процессорам при их полной или неполной загрузке и др., не учитываются.
- все  $n$  секторных наборов одновременно (параллельно) реализуют  $m$  подпрограмм (крупных фрагментов алгоритма) задачи пользователя, выделенных блоком служебных функций (БСлФ) для каждого типа интеллектуальной системы памяти (рис. 2);
- разрядность обрабатываемых слов условно принимаем равным  $\beta$  битам, при этом не учитывается, что примерно половина  $0,5\beta$  битов передается по кольцевой шине по часовой стрелке, а вторая половина соответственно одновременно передается против часовой стрелки, что приводит к дополнительному резерву уменьшения времени передачи информации;
- так как все процессы для каждого секторного набора распараллелены, то при этом целесообразно проводить оценку времени реализации (по сравнению с последовательным её выполнением) одной подпрограммы и соответственно оценку эффективности применительно к одному секторному набору с учетом, что другие секторные наборы выполняют

другие фрагменты алгоритма пользователя, реализация которых заканчивается в одно и то же время.

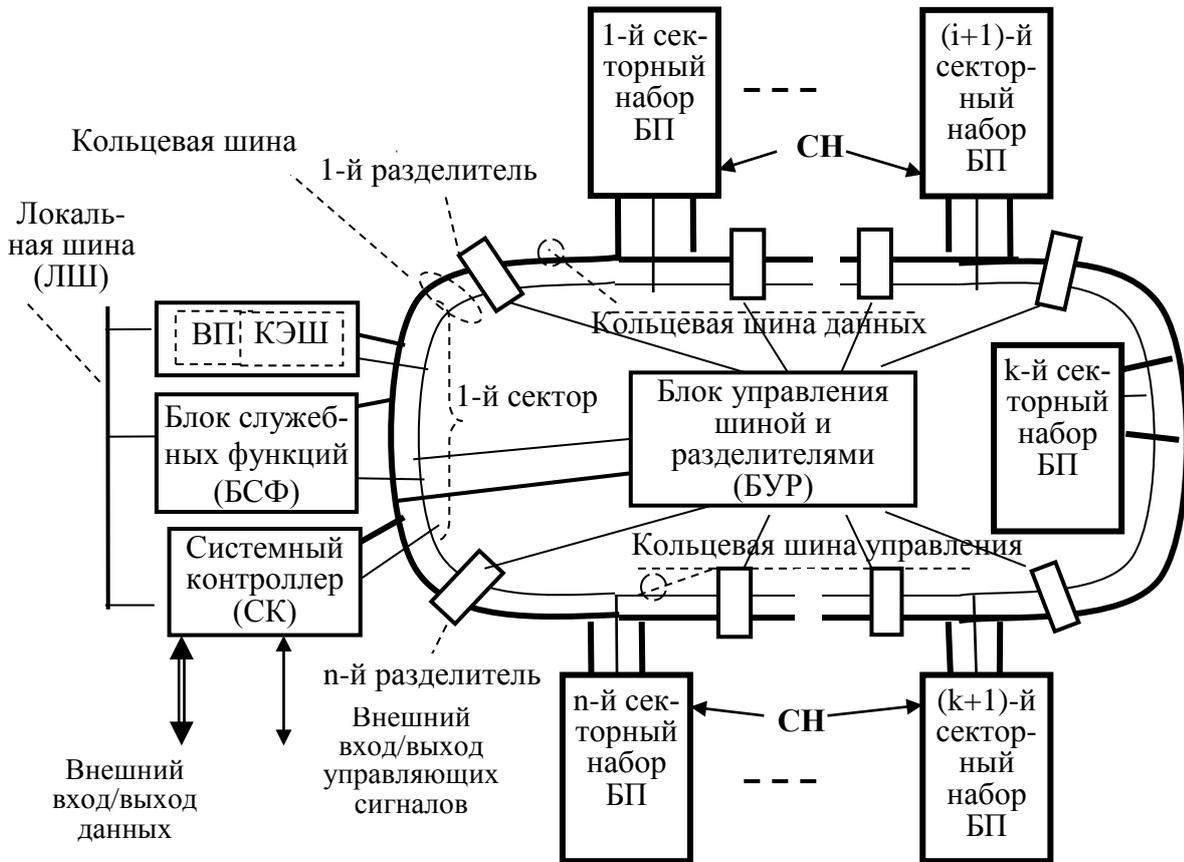


Рисунок 1 – Схема КС типа РІМ с модифицированной кольцевой шиной



Рисунок 2 – Состав секторного набора КС (СН)

С учетом приведенных выше положений разработана модель базового варианта секторного набора с отображением соответствующих процедур, которая приведена на рис. 3.

При этом приняты следующие обозначения [20, 23]: ПЯ – процессорное ядро; БП – банк памяти; КШ – кольцевая шина; ЛШ – локальная шина; РД – разделители кольцевой шины; КСН – контроллер секторного набора; ВП – ведущий процессор; БСФ – блок

служебных функций; БУР – блок управления шиной и разделителями; СН – секторный набор; СК – системный контроллер;  $t_{ki} = (t_{адр} + t_{обр})$  – время на формирование адреса и обращения к банку памяти;  $t_{ПЯi} = (t_{адр} + t_{лог} + t_{зн})$  – время на выполнение арифметических и логических операций с помощью ПЯ<sub>i</sub> и выдачу результата в БП<sub>i</sub>;  $t_{Вв/Выс}$  – время на операцию ввода/вывода после обработки информации на ПЯ;  $m \times t_{ц}$  – время, потраченное на реализацию количества циклов алгоритма пользователя;  $(n - m) \times t_{нц}$  – время, потраченное на

реализацию алгоритма пользователя при отсутствии циклов;  $k \times t_{pDi}$  – время передачи данных через  $k$  разделителей по кольцевой шине.

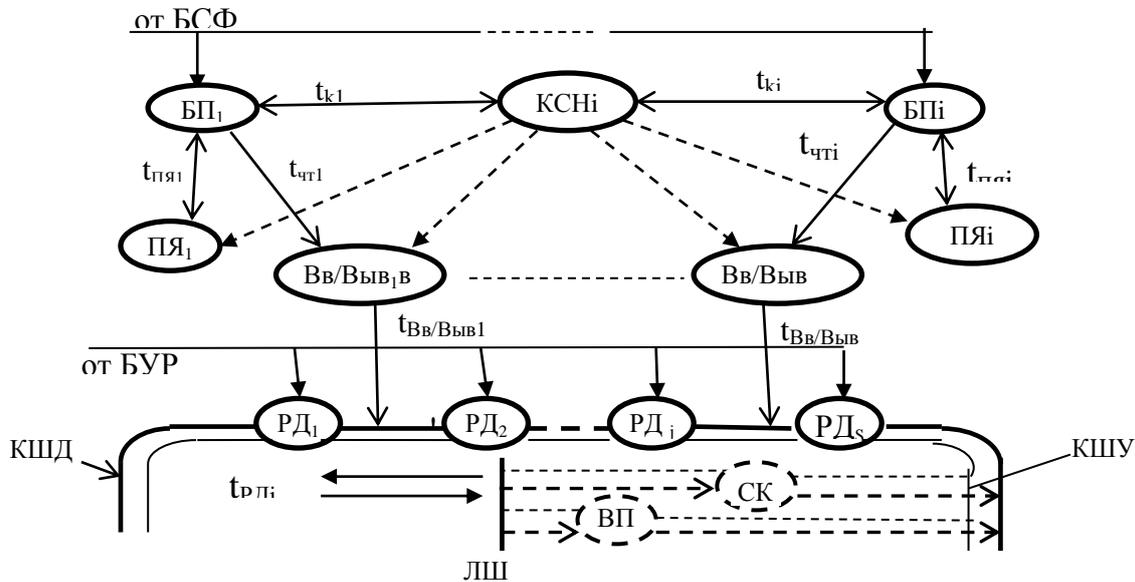


Рисунок 3 – Модель СН для оценки производительности КС с модифицированной кольцевой шиной

На основе анализа соответствующего поля входного пакета системный контроллер (СК) памяти вырабатывает управляющие сигналы инициализации процессов обработки информации, которые с выхода контроллера секторного набора (КСН<sub>i</sub>) поступают на соответствующий вход  $t_{ki}$  банка памяти (БП<sub>i</sub>), запуская в работу соответствующие процессоры (ПЯ<sub>i</sub>) банков памяти ( $t_{ПЯi}$ ). Так как в предлагаемом устройстве имеется возможность блокировки любых выбранных секторов кольцевой шины от влияния других с помощью секторных разделителей, то обработка фрагментов пользовательских задач в выбранных секторах может осуществляться одновременно, записывая результаты в БП<sub>i</sub>. Далее результат обработки фрагмента задачи в каждом БП<sub>i</sub> через его выход поступает на устройство ввода/вывода ( $t_{чTi}$ ), который затем подается через устройство ввода/вывода ( $t_{Вв/Выв}$ ) на соответствующий сектор кольцевой шины данных (рис. 3) и далее через секторные разделители РД<sub>i</sub> – на первый сектор кольцевой шины данных и на внешний ввод/вывод системы памяти к внешнему устройству или внешней системе.

После выполнения приписанной каждому сектору фрагмента задачи контроллер секторного набора выдает на соответствующий сектор кольцевой шины управления сигнал об окончании работы. При этом остальные секторные наборы банков памяти одновременно могут обрабатывать информацию, используя блокировку соответствующих секторов. Более подробная информация о работе системы памяти такого типа размещена в [20, 23].

Принимая во внимание приведенные выше исходные положения, можно предварительно оценить значение коэффициента эффективности  $Q_1$  применения модифицированной кольцевой шины в составе интеллектуальной системы памяти (по сравнению с последовательной реализацией алгоритма пользователя) по формуле

$$Q_1 = \frac{T_{\text{послед}}}{T_{\text{пар}}} = \frac{W_\alpha \sum_{i=1}^n (t_{y1} + k_{\text{нпi}} t_{pDi})}{\sum_{i=1}^m p_i (t_{y1} + k_1 t_{pDi}) + \sum_{i=n-m}^{\delta_i} (1 - p_i) (t_{y1} + k_2 t_{pDi})}, \quad (1)$$

где  $n$  – общее количество фрагментов алгоритма пользователя;

$m$  – количество фрагментов алгоритма пользователя, реализуемых в цикле с глубиной  $m$ , подлежащих распараллеливанию;

$(n - m)$  – количество фрагментов алгоритма, реализуемых в цикле с глубиной  $\delta_i$ , но не подлежащих распараллеливанию;

$p_i$  – вероятность полной (оптимальной) загрузки процессоров ПЯ на кольцевой шине;

$(1 - p_i)$  – вероятность неполной загрузки процессоров ПЯ на кольцевой шине;

$k_1 t_{PDi}$  – время задержки информации при прохождении через  $k_1$  разделителей;

$k_2 t_{PDi}$  – время задержки информации при прохождении через  $k_2$  разделителей;

$k_{\text{пя}i}$  – общее количество процессоров ПЯ на кольцевой шине, то есть

$$k_{\text{пя}i} = (k_1 + k_2);$$

$$t_{ki} = t_{\text{адп}} + t_{\text{зн}};$$

$$t_{y1} = (t_{ki} + t_{\text{ня}i} + t_{\text{чми}}(БП_i) + t_{\text{ВВ/ВЫВ}});$$

$$t_{\text{пя}i} = (t_{\text{адп}} + t_{\text{чми}} + t_{\text{логи}} + t_{\text{зн}i});$$

$W_{\alpha}$  – количество фрагментов алгоритма при последовательном выполнении алгоритма пользователя.

Что касается модели двух секторных иерархических наборов, объединенных коммутационным каналом (рис. 4), то она отличается от первой базовой модели наличием двух уровней иерархии с возможностью распараллеливания фрагментов алгоритма пользователя на каждом уровне и соответствующими задержками передачи информации между уровнями ( $t_{\text{КК1/2}}$ ), а также задержками узлов синхронизации Уз\_1(Сх) и Уз\_2(Сх) для установления согласования результатов обработки данных, поступающих от разных уровней иерархии ( $t_{\text{БСХ}}$ ). Остальные обозначения совпадают с обозначениями модели базового варианта интеллектуальной системы памяти [21, 24].

В связи с этим оценку эффективности  $Q_2$  и временных параметров для схемы такого типа по сравнению с последовательной реализацией алгоритма пользователя можно определить по формуле

$$Q_2 = \frac{T_{\text{носл}2}}{T_{\text{нар}2}} = \frac{(W_{\alpha1} + W_{\alpha2}) \sum_{i=1}^n (t_{y2} + k_{\text{пя}i} t_{PDi})}{\sum_{i=1}^m p_i (t_{y2} + k_1 t_{PDi}) + \sum_{i=n-m}^{\delta_i} (1 - p_i) (t_{y2} + k_2 t_{PDi})}, \quad (2)$$

где обозначения те же, что и для базовой модели, поскольку КШ вместе с ПЯ – одинаковые, за исключением:

$$t_{y2} = (t_{ki} + t_{\text{ня}i} + t_{\text{чми}}(БП_i) + t_{\text{ВВ/ВЫВ}} + t_{\text{КК1/2}} + t_{\text{БСХ}});$$

$t_{\text{КК1/2}}$  – время задержки при передаче информации с помощью коммутационных схем с одного уровня КШ на другой уровень КШ;

$t_{\text{БСХ}}$  – время задержки информации блоком синхронизации для устранения рассогласования результатов обработки данных между двумя иерархическим уровнями;

$W_{\alpha1}$  – выделенные фрагменты алгоритма пользователя первого иерархического уровня системы памяти;

$W_{\alpha2}$  – выделенные фрагменты алгоритма пользователя второго уровня иерархической системы памяти.

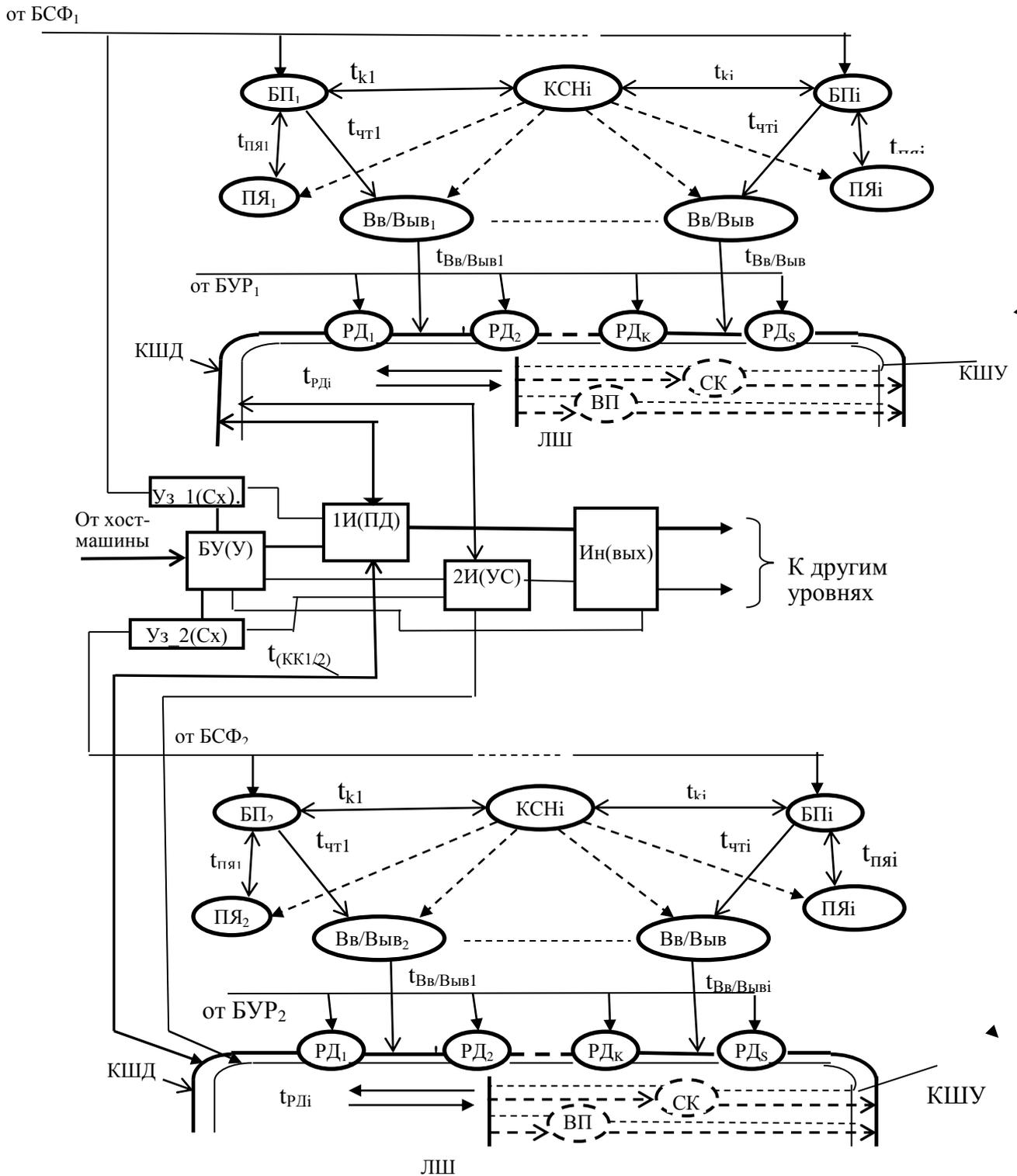


Рисунок 4 – Модели двух секторных иерархических наборов, объединенных коммутационным каналом

Модель интеллектуальной системы памяти с секционированными модулями на ПЛИС показана на рис. 5, а схема системы памяти с секционированными модулями приведена на рис. 6 [22, 25].

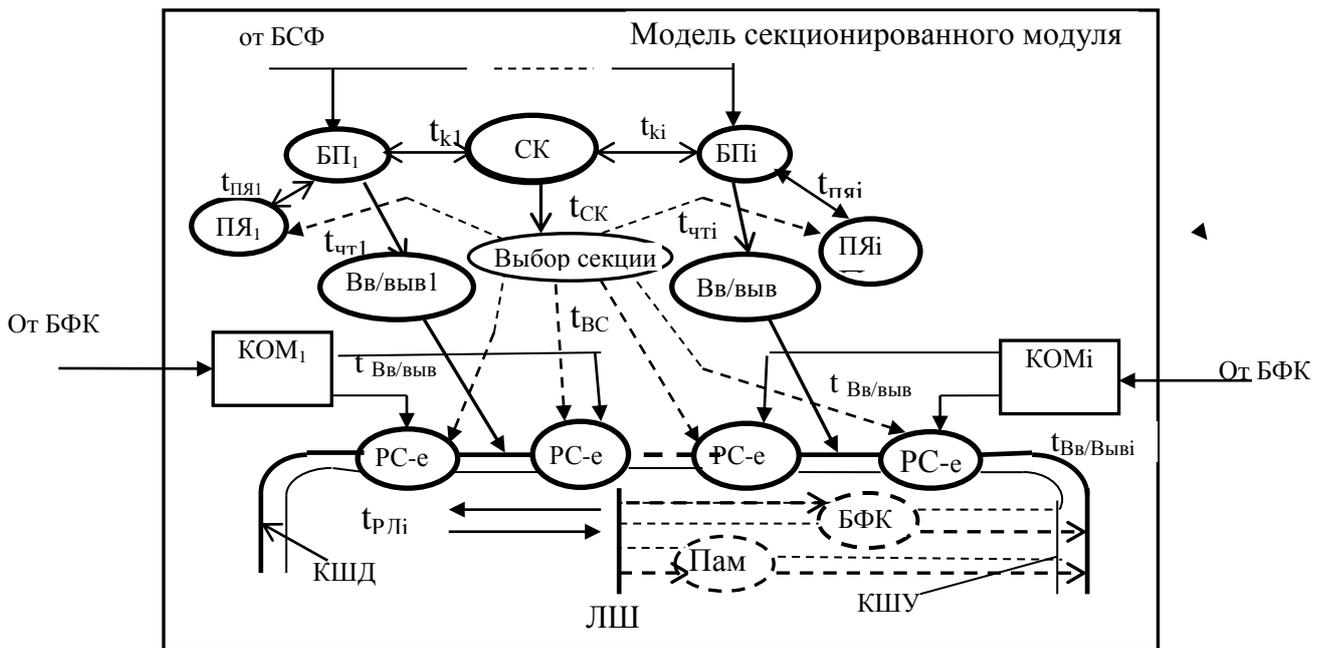


Рисунок 5 – Модель секционированного модуля на ПЛИС для расчета производительности системы с секционированными модулями

При этом параметр эффективности  $Q_3$  по сравнению с последовательной реализацией алгоритма пользователя можно определить по формуле

$$Q_3 = \frac{T_{послед3}}{T_{пар3}} = \frac{W_\alpha \sum_{i=1}^n (t_{y3} + k_{няi} t_{PДi}) + \mu_3 (t_{СВi} + t_{БСx} + t_{БУФСx})}{\sum_{i=1}^m p_i (t_{y3} + k_1 t_{PДi}) + \sum_{i=n-m}^{\delta_i} (1 - p_i) (t_{y3} + k_2 t_{PДi}) + \mu_3 (t_{СВi} + t_{БСx} + t_{БУФСx} + t_{комi})}. \quad (3)$$

Все обозначения те же, что и для базового варианта с одной кольцевой шиной, за исключением:

$\mu_3$  – количество секционных модулей на ПЛИС;

$t_{СВi}$  – время задержки селектора выбора секции;

$t_{y3} = (t_{ki} + t_{ПЯi} + t_{чТi} (БП_i) + t_{ВВ/ВЫВ})$ ;

$t_{БСx}$  – время задержки блоком синхронизации при выдаче результатов обработки информации;

$t_{БУФСx}$  – время задержки буферными схемами для синхронизации результатов данных, полученных от разных секций;

$t_{комi}$  – время задержки коммутатора для выбора  $i$ -й ПЛИС;

РС-е – тип ПЛИС с наличием PCI – Express.

Дадим качественную сравнительную оценку для рассмотренных трех вариантов систем с кольцевыми шинами. Примем следующие (произвольные) значения параметров:  $m=10$ ;  $n=12$ ;  $k_1=8$ ;  $k_2=2$ ;  $K_{ПЯi}=10$ ;  $t_{PДi}=2$ ;  $t_{адPДi}=2$ ;  $t_{зПЛi}=2$ ;  $t_{чТi} (БП_i)=3$ ;  $t_{ВВ/ВЫВ}=2$ ;  $\delta_i=5$ ;  $\mu_3=6$ ;  $t_{СВi}=3$ ;  $t_{БСxi}=2$ ;  $t_{БУФi}=2$ ;  $t_{комi}=3$ . При этом следует рассчитать параметры  $Q_j=f(P_i)$  и построить графики для основных значений:  $P_i=0,1$ ;  $P_i=0,2$ ;  $P_i=0,4$ ;  $P_i=0,6$ ;  $P_i=0,8$  и др. при следующих значениях другого параметра:  $W_\alpha=2$ ;  $W_\alpha=5$ ;  $W_\alpha=10$ ;  $W_\alpha=15$ ;  $W_\alpha=20$ . Полученные графики отражены на рис. 7 – рис. 9.

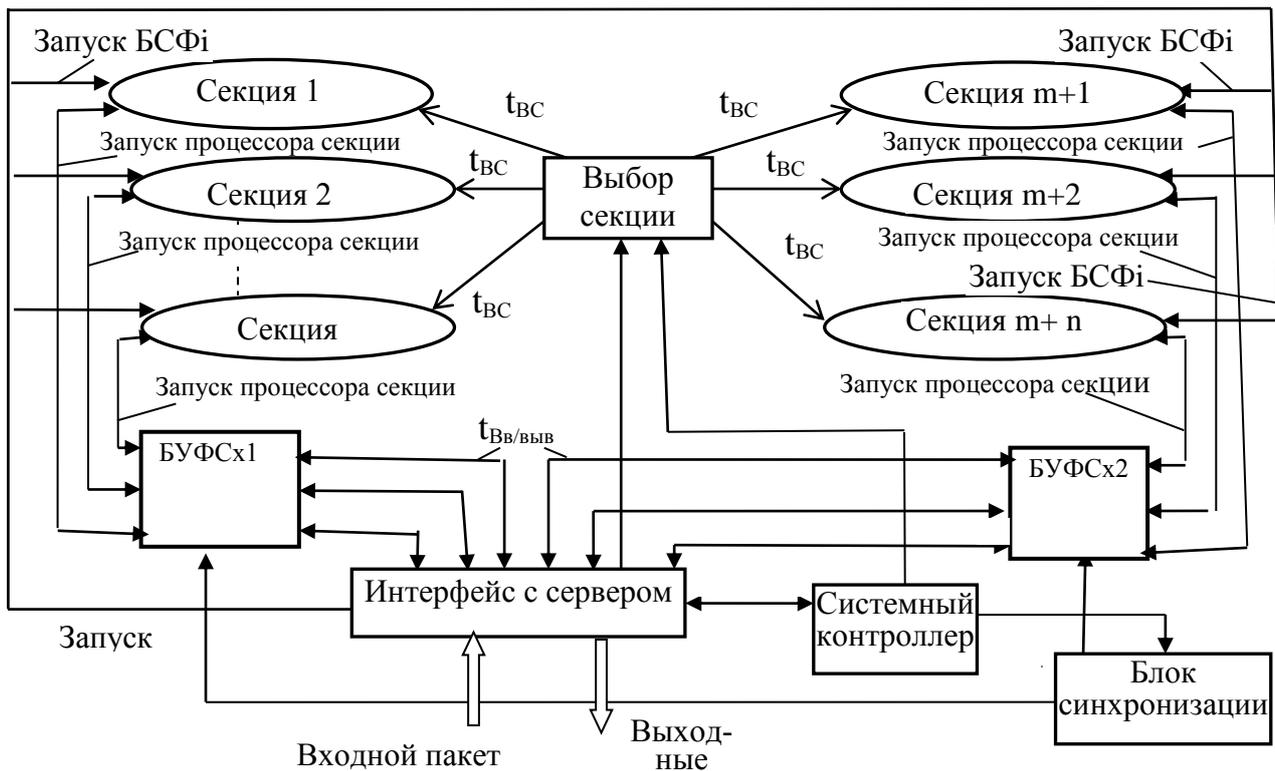


Рисунок 6 – Модель системы с секционированными модулями на ПЛИС для расчета её производительности

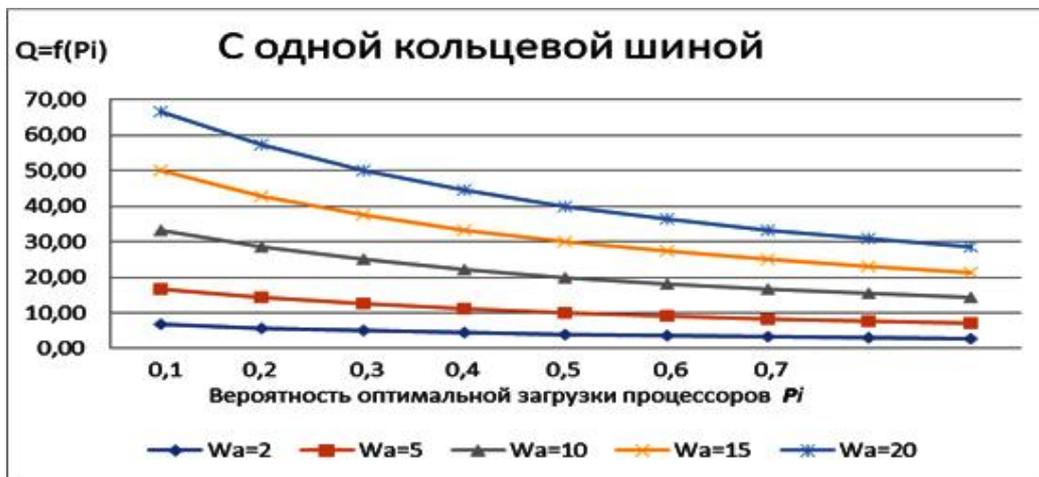


Рисунок 7 – Графики зависимости  $Q_1 = f(P_i)$  для базового варианта с модифицированной КШ

Как видно из графиков для базового варианта с одной кольцевой шиной и с секторными наборами при  $P_i=0,1$  и  $W_a=2$ , значение параметра (в тактах процессора)  $Q_1$  по сравнению с последовательной реализацией алгоритма пользователя уменьшено примерно в 7 раз, а при  $W_a=20$  – примерно в 68 раз. Это объясняется тем, что при расчетах принято количество процессоров 10, однако фрагментов для распараллеливания всего принято 2, то есть 8 процессоров при  $P_i=0,1$  останутся недогруженными либо при  $P_i=0,9$  оптимальную загрузку будут иметь только 2 процессора из 10. При этом основное время (в тактах процессора) будет потрачено на задержки, связанные с формированием адреса при обращении к банкам памяти, чтением и записью данных, а также выбором номера секторного набора,

задержкой буферных схем, вводом-выводом и др. Если говорить об этих задержках при значениях параметров  $W_a=20$ , то эти задержки будут слабо доминировать и весь процесс пойдет в направлении распараллеливания, обеспечив при  $W_a=20$  и при параметрах  $P_i=0,9$  общее уменьшение количества тактов процессора по сравнению с последовательным выполнением примерно 29 тактов. При  $P_i=0,5$  и  $W_a=20$  уменьшение количества тактов процессора достигнет величины 40.

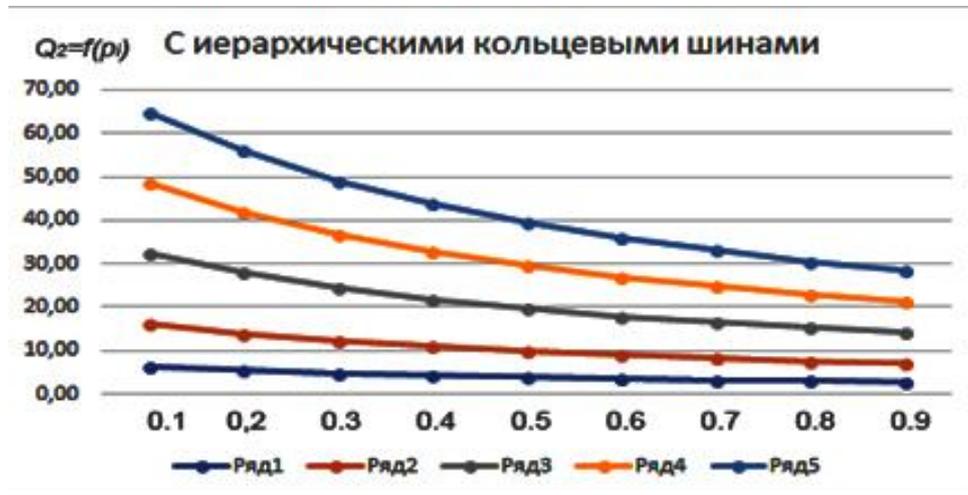


Рисунок 8 – Графики зависимости  $Q_2 = f(P_i)$  для иерархических КШ

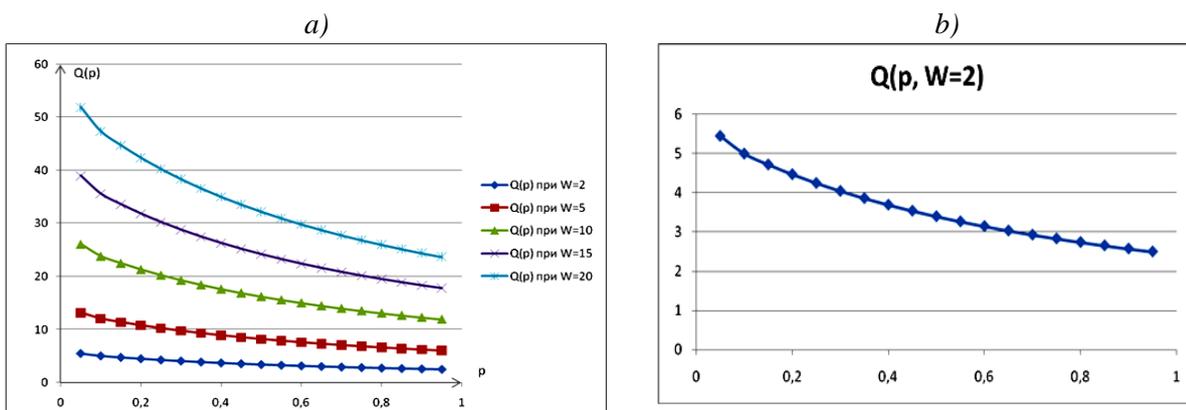


Рисунок 9 – Графики зависимости  $Q_3 = f(P_i)$ : а) для различных значений  $W_a$ ; б) для значений  $W_a=2$

## 8. Выводы

Таким образом, приведенные графики зависимости  $Q_j=f(P_i)$ , ( $j=1, 2, 3$ ;  $i=0,1-0,9$ ) показали, что наиболее престижной для обозначенных выше значений параметров оказалась интеллектуальная система памяти с одной кольцевой шиной, которая принята в качестве базового варианта, так как она имеет наименьшее количество тактов процессора при реализации алгоритма пользователя по сравнению с его последовательной реализацией.

Однако при большом количестве секторных наборов (например, 32 и больше), прикрепленных к кольцевой шине, паразитная емкость от каждого секторного набора накапливается и задержки возрастают, что затрудняет процесс синхронизации, например, результатов обработки первого и последнего секторного наборов и уменьшает производительность системы в целом. Поэтому такую систему памяти целесообразно использовать при небольшом количестве секторных наборов (например, 12–16) при оптимальной за-

грузке процессоров каждого секторного набора и при параллельной обработке конкретных фрагментов алгоритма пользователя, подлежащих распараллеливанию.

Что касается варианта системы памяти с иерархической организацией, то основные проблемы, связанные с задержками, как и для первого варианта, остаются. Однако такая система памяти обеспечивает дополнительные функции: распараллеливание на различных иерархических уровнях реализации алгоритма пользователя, например, на первом уровне – распараллеливание фрагментов алгоритма, а на втором – распараллеливание на уровне команд или подпрограмм.

Как видно из графиков, наиболее универсальный вариант – это интеллектуальная система памяти с секционированными модулями на ПЛИС, так как разработчик имеет возможность выбирать количество секций и секторных наборов каждой секции, исходя из конкретной задачи распараллеливания, используя при этом современную элементную базу (например, ПЛИС с PCI – Express). При параллельной реализации задачи пользователя для параметров  $W_a=2$  и  $P_i=0,1$  количество тактов процессоров по сравнению с последовательной реализацией уменьшено в 5 раз, а при  $P_i=0,9$  уменьшено почти в 2,5 раза. При параметрах  $W_a=20$  и при параметрах  $P_i=0,9$  общее уменьшение количества тактов процессоров по сравнению с последовательным выполнением уменьшено соответственно в 24 раза. Таковую интеллектуальную систему памяти целесообразно использовать при распараллеливании отдельных программ (подпрограмм), фрагментов программ алгоритма задачи пользователя, в том числе – на различных уровнях иерархии.

## СПИСОК ИСТОЧНИКОВ

1. Firoozshahian A. et al. A Memory System Design Framework: Creating Smart Memories. *Proc. of the 36th International Symposium on Computer Architecture (ISCA)*. Austin, TX, 2009. June. URL: <https://www.google.com/search?ei=yk6WW4XjB8z5wQKN5aHYDQ&q=Firoozshahian+A.+et+al.+A+Memory+System+Design+Framework%3A+Creating+Smart+Memories.+Proc.+of+the+36th>.
2. Kannan H. Ordering Decoupled Metadata Accesses in Multiprocessors. *Proc. of the 42nd International Conference on Microarchitecture (MICRO)*: New York City, NY, 2009. December. URL: <https://www.google.com/search?q=Firoozshahian%20A.%20et%20al.%20A%20Memory%20System%20Design%20Framework%3A%20Creating%20Smart%20Memories.%20Proc.%20of%20the%2036th%20>.
3. Arge L., Goodrich T.M., Nelson J.M., Sitchinava N. Fundamental parallel algorithms for private-cache chip multiprocessors. *ACM Symposium on Parallelism in Algorithms and Architectures (SPAA)*. URL: <https://www.google.com/search?ei=flCWW5fUMIbawQKXyry4Bg&q=Arge+L.%2C+Goodrich+T.M.%2C+Nelson+J.M.%2C+Sitchinava+N.+Fundamental+parallel+algorithms+for+private>.
4. Anderson E., Tucek J. Efficiency matters! *ACM SIGOPS Operating Systems Review*. 2010. Vol. 44, N 1. P. 40–45.
5. Dorneich M.C. et al. Supporting Real-time Cognitive State Classification on a Mobile Participant. *Journal of Cognitive Engineering and Decision Making*. 2007. Vol. 1, N 3. P. 240–270.
6. Keinrath C., Dorneich M.C., Vašek J. Designing for the future: A Cognitive-Adaptive-Man-Machine-Interface. *Annual Meeting of Human Factors and Ergonomics Society Europe Chapter: Linköping, Sweden*, 2009. 14–16 October. URL: <https://www.google.com/search?ei=flCWW5fUMIbawQKXyry4Bg&q=Arge+L.%2C+Goodrich+T.M.%2C+Nelson+J.M.%2C+Sitchinava+N.+Fundamental+parallel+algorithms+for+private>.
7. Палагин А.В., Семотюк В.М., Чичирин Е.Н., Сосненко Е.П. Моделирующая среда для создания и отладки систем цифровой обработки. *УСУМ*. 2013. № 1. С. 37–41.
8. Тимашов О.О., Зинько П.М. Про автоматизовану систему управління підприємством в реальному часі. *Вісник Київського національного університету імені Тараса Шевченка. Фізико-математичні науки*. 2012. № 2. С. 32–38.
9. Коваль В.Н., Палагин А.В., Яковлев Ю.С. и др. Интегрированная аналитическая система инфор-

- мационной поддержки (АСИП) решения задач в области СВТ: концепция, реализация, перспективы. *Проблемы программирования*. 2000. № 1–2. С. 398–408.
10. Палагин А.В., Яковлев Ю.С. Системная интеграция средств компьютерной техники: монография. Винница: «УНІВЕРСУМ-Вінниця», 2005. 680 с.
  11. Яковлев Ю.С. Однокристалльные компьютерные системы высокой производительности. Особенности архитектурно-структурной организации и внутренних процессов: монография. Винница: ВНТУ, 2009. 294 с.
  12. Яковлев Ю.С. О выборе графических ускорителей для компьютерных систем. *Наукові праці ДОННТУ. Інформатика, кібернетика та обчислювальна техніка*. 2013. № 2 (18). С. 61–71.
  13. Ерьсько В.В., Нестеренко Н.В. О построении мобильных компьютерных систем для информационной поддержки решения задач пользователя. *Управляющие системы и машины*. 2016. № 1. С. 26–33.
  14. Ерьсько В.В., Нестеренко Н.В. Типовые системы информационной поддержки для применения в различных областях. *Інформаційні технології та комп'ютерна інженерія*. 2015. Т. 2, № 33. С. 34–41.
  15. Яковлев Ю.С., Тимашов А.А. Средства сбора и предварительной обработки данных с использованием ПЛИС для технологических компьютерных систем и сетей. *Математичні машини і системи*. 2017. № 1. С. 25–38.
  16. Tymashov O., Jakovlev J. Components of an hil test system. *XXX International Conference problems of decision making under uncertainties (PDMU– 2017)*. Vilnius , Lithuania, 2017. 14–19 August. P. 123–124.
  17. Курзанцева Л.И. О построении модели пользователя интеллектуального интерфейса компьютерных систем. *Комп'ютерна математика: зб. наук. праць*. Київ : Ін-т кібернетики ім. В.М. Глушкова НАН України, 2006. № 3. С. 82–87.
  18. Яковлев Ю.С., Елисеєва О.В. Математическая модель и стратегия распределения приложений для интеллектуальной памяти распределенных компьютерных систем. *Математичні машини і системи*. 2009. № 4. С. 3–17.
  19. Яковлев Ю.С., Курзанцева Л.И. О развитии адаптивного человеко-машинного интерфейса и критериях его оценки в учебных системах. *Образовательные технологии и общество*. 2013. Т. 16, № 1. С. 547–563. URL: [http://ifets.ieee.org/russian/periodical/V\\_161\\_2013EE.html](http://ifets.ieee.org/russian/periodical/V_161_2013EE.html).
  20. Яковлев Ю.С., Тихонов Б.М., Елисеєва Е.В. Компьютерная система типа «Процессор-в-памяти» с модифицированной кольцевой шиной. *УСиМ*. 2011. № 3. С. 54–61, 88.
  21. Боюн В.П., Яковлев Ю.С. Интеллектуальная система памяти с иерархическими кольцевыми шинами. *Математичні машини і системи*. 2016. № 4. С. 18–32.
  22. Палагин А.В., Боюн В.П., Яковлев Ю.С. Интеллектуальная система памяти с секционированными модулями на ПЛИС. *УСиМ*. 2017. № 6. С. 17–25.
  23. Палагін О.В., Яковлев Ю.С., Тихонов Б.М., Єлісеєва О.В. Інтелектуальна розподілена система пам'яті з кільцевою шиною: пат. 76229, Україна, МПК (2006.01) G06F 15/16, G06F 13/42; заявл. 16.07.2010; опубл. 10.03.2011, Бюл. № 5. 41 с.
  24. Боюн В.П., Яковлев Ю.С. Інтелектуальна розподілена система пам'яті з ієрархічними кільцевими шинами: пат. 116583, Україна. МПК G06F 15/16 (2006.01), G06F 13/42 (2006.01); заявл. 24.05.2016; опубл. 12.12.2016, Бюл. № 23. С. 20.
  25. Палагін О.В., Яковлев Ю.С., Боюн В.П. Інтелектуальна розподілена система пам'яті з секціонованими модулями на ПЛИС: пат. 119772, Україна. МПК G06F 13/42 (2006/01); заявл. 03.04.2017; опубл. 10.10.2017. Бюл. № 19. С. 11.
  26. Siel W., Ebinger B., Spörl Th. A hardware-in-the-loop test environment for interconnected ECUs for passenger cars and commercial Vehicles. *Proc. of FISITA World Automotive Congress*: Seoul, Korea, 2000. 12–15 June. P. 1–6.
  27. MacDiarmid M., Basic M. Quantifying the accuracy of hardware-in-the-loop simulations. *Proc. of American Control Conference ACC '07*. 2007. P. 5147–5152. URL: <https://core.ac.uk/download/pdf/82080022.pdf>.

Стаття надійшла до редакції 31.05.2018