

СИСТЕМНЫЕ АСПЕКТЫ ПРОЕКТИРОВАНИЯ ГАРАНТОСПОСОБНЫХ ОБЛАЧНЫХ ВЫЧИСЛЕНИЙ

Н.А. Косовец, И.Е. Щетинин, Л.Н. Товстенко

Рассмотрены системные аспекты проектирования гарантоспособных облачных вычислений. Техническая реализация такого вычислительного контента возможна при использовании пространственного отказоустойчивого мультимикропроцессора реального масштаба времени с варьируемым соотношением производительности и достоверности при дефиците ресурсов надёжности средствами операционной системы, динамически изменяя архитектурный облик системы со структурной избыточностью. Особое внимание уделено рассмотрению абстрактного ресурса – надёжности, его исключительной роли при проектировании мультимикропроцессора со свойствами отказоустойчивости. Система реконфигурируется, задание перераспределяется между процессорами. При наличии свободных процессоров последние инициализируются и включаются в вычислительный процесс. Авторами предложена и сформулирована оценка эффективности данной отказоустойчивой технологии.
Ключевые слова: гарантоспособность, облачные вычисления, мультимикропроцессор, реконфигурация, вариация производительности и надёжности, архитектура.

Розглянуті системні аспекти проектування гарантоздатних хмарних обчислень. Технічна реалізація такого обчислювального контенту можлива при використанні просторового відмовостійкого мультимікропроцесора реального масштабу часу з варіюваними співвідношенням продуктивності і достовірності, при дефіциті ресурсів надійності, засобами операційної системи, динамічно змінюючи архітектурний вид системи зі структурним надлишком. Особливу увагу приділено розгляду абстрактного ресурсу – надійності, його виключній ролі при проектуванні мультимікропроцесора з властивостями відмовостійкості. Система реконфігурується, завдання перерозподіляється між процесорами. При наявності вільних процесорів, останні ініціалізуються та включаються в обчислювальний процес. Авторами запропонована та сформульована оцінка ефективності даної відмово стійкої технології.
Ключові слова: гарантоздатність, хмарні обчислення, мультимікропроцесор, реконфігурація, варіація продуктивності і надійності, архітектура.

There is discussed the system aspects of designing secure cloud computing. Technical realization of such computational content is possible with the use of a spatial fault-tolerant multiprocessor of a real time scale with a variable ratio of performance and reliability with a lack of reliability resources by means of the operating system, dynamically changing the architectural appearance of the system with structural redundancy. Particular attention is paid to the consideration of an abstract resource – reliability, its exceptional role in the design of a multiprocessor with the properties of fault tolerance. The system is reconfigured the task is redistributed among the processors. If there are free processors, the latter are initialized and included in the computational process. The authors proposed and formulated an assessment of the effectiveness of this fault-tolerant technology.
Key words: reliability, cloud computing, multiprocessor, reconfiguration, variation of performance and reliability, architecture.

Введение

В данной статье рассмотрены системные аспекты проектирования гарантоспособных облачных вычислений. Техническая реализация такого вычислительного контента возможна при использовании пространственного отказоустойчивого мультимикропроцессора реального масштаба времени с варьируемым соотношением производительности и достоверности при дефиците ресурсов надёжности средствами операционной системы, динамически изменяя архитектурный облик системы со структурной избыточностью.

Введение в облачные вычисления понятия гарантоспособность позволит дать количественную оценку фактического предоставления услуг, которые не отличаются от спецификации. Гарантоспособность устраняет путаницу между понятиями надёжности вообще и мерой надёжности, как математической количественной величины, характеризующей степень надёжности системы. Она подчёркивает, что надёжность, ремонтпригодность, безопасность и другие показатели являются количественными характеристиками различных проявлений одного и того же свойства системы – гарантоспособности.

Нарушение гарантоспособности определяется отсутствием обоснованной уверенности в предоставлении системой надлежащих услуг в течение достаточно длительного периода времени, указанного в спецификации. Для предотвращения понижения гарантоспособности, разрабатываем способы обеспечения гарантоспособности, являющихся совокупностью методов, средств и проектных решений, которые позволяют придать системе способность к гарантоспособному выполнению требуемых функций и придать пользователю уверенность в наличии у системы такой способности. Показателями гарантоспособности могут служить коэффициент появления неисправностей в системе, ошибок, надёжность, готовность, безопасность и способы предотвращения неисправностей, ошибок (отказоустойчивость), прогнозирование неисправностей и их устранение.

Наиболее важным показателем гарантоспособности есть отказоустойчивость, которой уделим наибольшее внимания на всех этапах создания системы облачных вычислений. Отказоустойчивость на проектном уровне достигается за счёт применения двух не исключających друг друга подходов: функционального разбиения и проектного разнообразия.

Используя метод функционального разбиения, каждая функция системы разбивается на ряд подфункций, таким образом, чтобы отказ компонента, реализующего некоторую подфункцию, оказывался изолированным и

не мешал выполнять функции в целом, хотя и при ухудшении характеристик. Указанный подход требует тщательного проектирования мультипроцессора и, особенно, его межпроцессорных связей.

Другой подход – проектное разнообразие, состоящее в обеспечении выполнения требуемых функций обслуживания по нескольким независимо спроектированным и отдельно реализованным вычислительным каналам.

Ресурсное обеспечение облачных вычислений

Современные средства представления доступа к облачным ресурсам средствами Интернет вещей (IoT), скоростного транспорта посредством 5G Интернет, внедрением новых организаций построения беспроводных сетей (MIMO), новых способов передачи информации в различных частотных диапазонах (OFDM) обусловили информационный взрыв в области передачи и обработки информации. Облачные вычисления получили наибольшее развитие в области современных информационных технологий хранения и обработки данных. Это совершенно новое информационное пространство, дающее неограниченные возможности. Но пользование общими виртуальными Интернет ресурсами несёт в себе определённую опасность: уменьшается безопасность передаваемых данных, нет гарантии в своевременном получении услуги, сложность организации своего безопасного «облака» для организации бизнеса, проблемы продажи информационных услуг. Часть проблем нивелируется внедрением частного облака, предоставляющего безопасную ИТ-инфраструктуру, контролируруемую и эксплуатируемую в интересах частного предприятия.

Частная модель облачных вычислений обеспечивает высокую степень масштабируемости. Она позволяет быстро расширить набор ИТ-сервисов или получить к ним доступ без необходимости капитальной модернизации своего базового центра обработки данных. Это обеспечивает организации такое важное преимущество, как снижение совокупной стоимости владения ИТ-ресурсом, что, в свою очередь, повышает ее доходность, упрощает реинвестирование капитала

Устойчивая тенденция роста научных публикаций об организации гарантоспособных облачных вычислений обусловлена кризисом архитектурных моделей, способных адекватно отобразить прикладную задачу с требуемым значением надёжности на облачных ресурсах. Для большинства задач архитектура облака не отражает проблем организации вычислений, ранжируемых по уровням гарантоспособности, и проблем организации вычислений в реальном масштабе времени. Системное проектирование микро-ЭВМ выдвигает ряд актуальных задач, связанных с обоснованием архитектуры и выбором структуры микро-ЭВМ.

Архитектура микро-ЭВМ должна учитывать современные каналы передачи информации и средства доступа. Это позволит внедрить сервис - ориентированные архитектуры, оптимизировать обработку данных и управление информацией.

При больших объёмах вычислений, повышенных требованиях к защите информации, обеспечения гарантоспособных вычислений в рамках корпорации целесообразно рассмотреть модель облачных вычислений – частное облако (private cloud), которое обладает многими преимуществами компьютерной среды. В частном облаке управление данными и процессами осуществляется внутри корпорации. Другими словами, в этом случае отсутствуют такие проблемы, как ограничение пропускной способности сети, угрозы безопасности и необходимость нормативного соответствия, которые могли бы возникнуть при использовании публичных облаков посредством открытых сетей общего пользования. Кроме того, сервисы на базе частных облаков способны предложить поставщику и конечному пользователю более высокую степень контроля, в том числе доступа пользователей к сети, что существенно улучшает безопасность и устойчивость.

Следует отметить важность построения архитектурных моделей, что позволяет оценить информационные потоки модели до того, как начнется процесс реализации. Функциональные возможности аппаратных и программных компонент представляются архитектурными моделями с использованием высокоуровневой абстракции в виде потоков данных и абстрактно представляют технологию реализации во времени. Архитектурные модели содержат схемы арбитража, допускают параметризацию и типизацию. Так, параметры конфигурации архитектурной модели позволяют определить соотношение реализации функций аппаратными и программными средствами. Проектирование на системном уровне с применением архитектурного моделирования упрощает спецификацию проекта, делает плавным переход от функциональных к формальным требованиям, поскольку разделяет проблемы выработки функциональных требований и спецификаций проекта, так как обычно отсутствуют средства количественной оценки спецификаций.

Логические аспекты проектирования гарантоспособных вычислений

Создание гарантоспособных облачных вычислений (то есть обеспечивающих гарантоспособное выполнение облачных функций системой, для которых она предназначена) сводится к упрощению предоставления вычислительных сервисов, поскольку необходимо обеспечить надлежащее взаимодействие большого числа усложняющихся компонент системы, которые реализуются на основе гарантоспособных

вычислений и обеспечения отказоустойчивости при проектировании. Отказ в системе происходит тогда, когда предоставляемые ею услуги отличаются от услуг, указанных в системной спецификации, то есть ожидаемые виды обслуживания согласованные между пользователем и разработчиком. Методы предотвращения неисправности и методы обеспечения отказоустойчивости можно рассматривать в качестве способов реализации гарантоспособности, показывающие как придать системе способность к предоставлению требуемых услуг. Методы устранения и прогнозирования могут считаться способами предотвращения гарантоспособности, так как они указывают, каким образом можно достичь уверенности в способности системы к выполнению предопределенных услуг.

Доминирующим в последнее время направлением развития архитектур вычислителя заключалась в увеличении производительности, информационной ёмкости. Однако принципы организации высокопроизводительных вычислений кардинальных изменений не претерпели. основополагающим принципом построения высокопроизводительных вычислений до настоящего времени остается принцип макроконвейерных вычислений. Организация макроконвейерных вычислений основывается на параллелизме алгоритмов, сводимых к системе слабосвязанных и независимых операций компонент алгоритма. С целью формализации такого представления алгоритма вводим и формализуем крупные общематематические и специальные понятия, обеспечивающие эффективное описание и разрешение проблем реализации алгоритмов заданного класса. Канонический набор таких понятий называемых базой организации макроконвейерных вычислений, содержит объект типа задание, объект типа задача, объект типа семафор, объект типа почтовый ящик и объекты других типов. Операционные системы, ориентированные на реализацию алгоритмов в макроуровневом представлении, строим как интерпретаторы операторов и директив базы организации макроконвейерных вычислений. Достижимый при этом эффект заключается в макроуровневом управлении последовательностью реализации компонент алгоритма с помощью потока данных и событий. Кроме того, специализация функций операционной системы, состоящая в ориентации на организацию взаимодействия компонент алгоритма, значительно упрощает выбор стратегии распределения ресурсов микро-ЭВМ. Наконец, использование макроуровневого представления алгоритма, позволяет откладывать процедуру связывания компонент алгоритма и средств их реализации до момента начала решения задач.

Неравнозначность требований к достоверности решений задач определяет необходимость многоуровневого ранжирования задач по гарантоспособности. Отсюда следует, что база организации вычислений, ориентированная на решение задач при дефиците ресурсов надежности, должна включать, по меньшей мере, две разновидности заданий – задание, обеспечивающее более высокий уровень достоверности решения задач, и задание, обеспечивающее менее высокий уровень достоверности решения задач [1]. Формализация проблем реализации алгоритмов основана представлением алгоритма операционной моделью. Остовом этой модели является ориентированный граф, вершины которого отождествляют с операциями, а дуги – со связями между ними, причем содержательная интерпретация операций зависит от уровня детализации алгоритма.

Альтернативы отказоустойчивым технологиям при построении облачных вычислений реального времени нет. Значение надежности рассчитывается и закладывается на системном уровне и зависит не только от аппаратных и программных ресурсов, а в большей степени от их взаимодействия, управления ресурсами. В результате надежность выступает как ресурс системы и варьируется в зависимости от выполняемой задачи.

Принцип организации высокопроизводительных вычислителей, в основе которых лежит организация макроконвейерных вычислений, основан на параллелизме алгоритмов, сводимых к системе слабосвязанных и независимых операций ветвей алгоритма, не претерпел кардинальных изменений в «облаках». Определение объектов состоит в типе объекта, атрибутах объекта и интерпретации объекта набором применимых к нему операций.

Для формализации вычислительного процесса создаются логические ресурсы и механизм доступа к ним. Организацию взаимодействия в такой гипотетической машине возлагается на операционную систему. Целью ее является определение среды для создания и выполнения программ [2]. Фактическим стандартом средств интерпретации макроконвейерных вычислений является мультизадачная операционная система. Построение оптимального варианта средств реализации заданного алгоритма обеспечивает многовариантный арбитраж системной магистрали и резидентных ресурсов. Таким образом, облачные вычислители являются основой обширного класса мультипроцессорных микро-ЭВМ с распределенной пространственной магистралью, повышение производительности которых достигается за счет одновременной реализации параллельных компонент алгоритма. Существенная взаимозависимость надежности и производительности определяет необходимость создания облачных вычислителей с варьируемым соотношением производительности и достоверности. Проектирование ЭВМ такого класса выдвигает ряд актуальных задач, к наиболее сложным из которых относится формальная интерпретация проблем обработки информации при дефиците ресурсов надёжности [3].

Операционные системы, ориентированные на реализацию алгоритмов в макроуровневом представлении, строятся как интерпретаторы операторов и директив базы макроконвейерных вычислений. Достижимый эффект заключается в макроуровневом управлении последовательностью реализации компонент алгоритма с помощью потока данных и событий. Кроме того, специализация функций

операционной системы, состоящая в ориентации на организацию взаимодействия компонент алгоритма, значительно упрощает выбор стратегии распределения ресурсов встроенной мультипроцессорной ЭВМ для облачных вычислений.

В настоящее время можно выделить два направления развития проектирования микро-ЭВМ для облачных вычислений. Первая из этих двух тенденций – стремление к созданию моделей большой общности и универсальных стратегий системного проектирования микро-ЭВМ, а вторая – разработка частных моделей микро-ЭВМ и специальных стратегий системного проектирования. Необходимостью частных реализаций системного подхода обусловлено противоречием между эффективностью и общностью подхода. Наиболее эффективна проблемно-ориентированная архитектура с гибкой функционально перестраиваемой структурой.

Для управления ресурсами мультипроцессора используем совокупность процедур определенного типа, называемых системным монитором. Каждому монитору предоставлено исключительное право доступа к системным таблицам. Он отвечает за распределение ресурсов, взаимосвязанных и взаимозависимых с ресурсом надежности.

Системные аспекты проектирования гарантоспособных вычислений

Можно выделить системные и структурные свойства, рассматриваемые в системном и структурном аспектах соответственно. Первый из этих двух взаимосвязанных аспектов охватывает комплекс вопросов, связанных с функциональными возможностями микро-ЭВМ по отношению к классу решаемых задач, и формированием понятийного аппарата обеспечивающее компактное описание и эффективную реализацию алгоритмов заданного класса. Второй аспект направлен на внутреннее строение микро-ЭВМ и определяет подход к выбору структуры, обеспечивающей требуемые показатели качества.

Системные свойства характеризуют интегральные качества микро-ЭВМ, которые в зависимости от сложности микро-ЭВМ подразделяются на несколько уровней. К числу этих уровней относится способность решать поставленные задачи, отражающие потенциальную эффективность микро-ЭВМ. Это качество объединяет такие определяющие потребительскую ценность – свойства микро-ЭВМ, как производительность, надёжность обеспечения различного рода ресурсами. Свойства второй группы характеризуют внутреннее строение микро-ЭВМ. Именно эта группа свойств формируется в результате принятия решения по выбору структуры и стратегии использования микро-ЭВМ, целью которых является придание микро-ЭВМ нужных функциональных возможностей и потребительских свойств.

Формализация микро-ЭВМ, как объекта проектирования, базируется на двух концепциях моделирования: архитектурной и структурной. Согласно первой концепции модель отражает функциональные возможности микро-ЭВМ в отношении класса решаемых задач, согласно второй концепции – находится в отношении структурного подобия со средствами реализации алгоритма. При этом концептуальную целостность модели обеспечивает трактовка архитектуры как совокупность потребительских свойств и характеристик функциональных возможностей микро-ЭВМ, являющихся внешним проявлением свойств внутреннего строения. Основанное на этих концепциях моделирования описание микро-ЭВМ будем называть архитектурно-структурной моделью, которая в общем случае имеет следующий вид:

$$U = \langle S, A, F, X \rangle ,$$

где $S = \{s: \langle S_0, R \rangle\}$ – множество архитектурно-структурных вариантов микро-ЭВМ, определяемое минимальным структурным базисом $S_0 = \{S_{0j}\}$ и множеством $R = \{R_d\}$ операторов и процедуры преобразования структуры;

$\{AC, AV\}$ – множество характеристик функциональных возможностей и устойчивых (инвариантных относительно преобразования структуры) потребительских свойств микро-ЭВМ, включающее множество $FC = \{AC_k\}$ неуправляемых характеристик и множество $AV = \{AV_n\}$ управляемых характеристик. Определение множества значений управляемых характеристик сводится к отображению $\Phi = \{\Phi_n: S \rightarrow AV(S)\}$ множества структурных вариантов микро-ЭВМ на множество значений рассматриваемой характеристики. Таким образом, множество значений каждой варьируемой характеристики может быть представлено в виде:

$$AV(S) = \{av(s): \langle S = \{S = \{s: \langle S_0, R \rangle\}, \Phi \rangle\}.$$

$F: A \rightarrow S_0$ – отображение, задающее привязку компонент архитектурного облика к структурному базису микро-ЭВМ;

$X = \{X_j\}$ – множество возможных стратегий управления структурой микро-ЭВМ.

Постановка задачи проектирования облачных вычислений, ориентированных на реализацию алгоритмов, сводимых к системе слабосвязанных и независимых операций, основана на отыскании ярусно-параллельной формы алгоритма. Найденную ярусно-параллельную форму алгоритма описывают ориентированным графом,

вершины которого отождествляют с операциями, а дуги – со связями и отношениями между ними. Операционные характеристики компонент алгоритма описывают с помощью множества атрибутов и отображения, задающего привязку атрибутов к вершинам и дугам графа. Сформированная таким образом операционная модель алгоритма имеет следующий вид:

$$D = \langle G(Z, B), T, P \rangle,$$

где $G(Z, B)$ – ярусно-параллельная форма представления графа алгоритма, определяемая множеством $Z = \{Z_j\}$ операций и отношением B , задающим информационные связи на множестве Z операций;
 $T = \{T_j\}$ – множество атрибутов вершин и дуг графа $G(Z, B)$;
 $P: T \rightarrow G(Z, B)$ – отображение, задающее привязку атрибутов к вершинам и дугам графа $G(Z, B)$.

Глобальной целью, преследуемой при проектировании мультипроцессорной ЭВМ, является определение наилучшего, при заданных условиях, варианта аппаратно-программных средств реализации алгоритмов заданного класса. В большинстве случаев оптимальный вариант микро-ЭВМ должен обеспечивать экстремальное значение векторного показателя эффективности. Поэтому задачи выбора оптимального варианта средств реализации заданного алгоритма, вообще говоря, не имеют единственного решения. Если допустимо свертывание векторного показателя эффективности в скалярный, то данная задача сводится к оптимизационной задаче однокритериального типа, общая постановка которой имеет следующий вид:

$$R = \langle D, V, O, H, K, U, u^* \rangle,$$

где D – множество проблем, связанных с реализацией алгоритмов заданного класса;
 V – множество возможных архитектурно-структурных вариантов микро-ЭВМ;
 O – множество ограничений, накладываемых на средства реализации алгоритма;
 H – отображение множества управляемых характеристик средств реализации заданного алгоритма на множество значений показателя эффективности;
 K – показатель эффективности средств реализации заданного алгоритма;
 U – множество допустимых, удовлетворяющих ограничениям на средства реализации алгоритма, архитектурно-структурных вариантов мультипроцессорной ЭВМ;
 u^* – оптимальный архитектурно-структурный вариант мультипроцессорной ЭВМ.

Система разрабатывается в виде набора базисных механизмов, управляющих комплексом абстрактных ресурсов. Поэтому нет необходимости встраивать средства управления, например, базами данных, их можно создать посредством базисных механизмов. Вертикальная структура позволит оптимально реализовать режим реального времени. Учитывая ограничение, что процессы слабосвязаны, проектирование сводится к выделению набора специализированных функциональных процессоров, предназначенных для выполнения функций различных системных компонент, например, планировщика, программ управления файлами, обработчика директивы запуска, механизма передачи/приема сообщений или управления памятью. Нагрузка между процессорами должна распределяться по возможности равномерно. Избыточные процессы должны управляться отдельными процессорами, что дает преимущества при отказах процессоров, управляющих основными процессами. Минимизация количества процессоров не является основной целью, если требуется обеспечить высокую надежность системы.

При организации облачных вычислений кроме достоверности вычислений важно время выполнения задачи, то есть гарантируется время получения результата. Класс задач, время выполнения которых гарантируется, относим до гарантоспособных. При проектировании микро-ЭВМ с гарантоспособными вычислениями, в качестве оптимального параметра выбираем критерий наименьшего времени реализации алгоритма. Выбор оптимального варианта средств обработки информации осуществляем с позиции подхода, вовлекающего в совместное рассмотрение заданный алгоритм и средства его реализации.

Сущность предлагаемого подхода в решении поставленных задач заключается в конструктивном отображении проблем обработки информации на архитектуру микро-ЭВМ и проведении преобразований формы представления заданного алгоритма и данных, с одной стороны, и структуры средств его реализации, с другой стороны, направленных на достижение экстремального значения целевой функции при удовлетворении существующих ограничений. Конструктивное отображение проблем на архитектуру микро-ЭВМ состоит в гомоморфном отображении проблем обработки информации на базу организации вычислений и отображении базы на архитектуру микро-ЭВМ.

С целью конструктивного отображения проблем обработки информации на архитектуру микро-ЭВМ, максимальная ярусно-параллельная форма алгоритма описывается операционной моделью, а облик микро-ЭВМ задаётся гомоморфным отображением проблем обработки информации на архитектуру микро-ЭВМ [4]. Это отображение определяет минимальную совокупность аппаратно-программных средств, обладающих необходимыми для реализации алгоритма свойствами, а операции преобразования структуры этих средств

обеспечивают порождение микро-ЭВМ с требуемой конфигурацией. Функциональные возможности микро-ЭВМ описываются с помощью отображения, задающего привязку компонент внешнего облика к структурному базису, причём неуправляемым компонентам облика микро-ЭВМ соответствует множество неуправляемых характеристик, а варьируемым – множество значений множества управляемых характеристик. Таким образом, архитектурно-структурная модель микро-ЭВМ описывает множество вариантов микро-ЭВМ, определяемых базовой микро-ЭВМ и набором процедур преобразования архитектуры и структуры [5].

Организация вычислений при дефиците ресурсов надежности требует распределения задач между процессорами, базирующегося на априорных и прогнозных характеристиках ресурсов производительности и надежности процессоров. Распределение задач между процессорами включает исходное распределение задач между процессорами и последующие распределения, связанные с отказами процессоров. Фактическая производительность мультипроцессора облачных вычислений зависит от соотношения времени, затрачиваемого на решение задач восстанавливаемых заданий и времени, затрачиваемого на решение задач гарантоспособных заданий.

Выводы

В статье рассмотрено понятие ресурса облачного отказоустойчивого мультипроцессора реального времени и управление им посредством операционной системы. Особое внимание уделено рассмотрению абстрактного ресурса – надежности, его исключительной роли при проектировании мультипроцессора со свойствами отказоустойчивости.

Для обнаружения ошибки используется тест содержательности, выполнение процедур достоверности или теста приемлемости. После обнаружения ошибки неисправные компоненты локализуются и исключаются из вычислительного процесса. Система реконфигурируется, задание перераспределяется между процессорами. Свободные процессоры инициализируются и включаются в вычислительный процесс.

Авторами предложена и сформулирована оценка эффективности данной отказоустойчивой технологии. При уменьшении количества процессоров ниже критичного значения, система деградирует, то есть процессы перераспределяются среди наличных процессоров, в таком случае падает производительность системы без ухудшения качествам.

Статья посвящена актуальной проблеме системного проектирования облачных вычислений на базе отказоустойчивого мультипроцессора реального времени с варьируемым соотношением производительности и достоверности при дефиците ресурсов надежности и имеет практическую направленность. Решена задача отображения проблем обработки сложной информации при дефиците ресурсов надёжности на архитектуру микро-ЭВМ. Описана нетрадиционная база организации вычислений ранжируемых по уровням гарантоспособности и рассмотрены вопросы обоснования архитектуры микро-ЭВМ с варьируемым соотношением производительности и надёжности.

Литература

1. Косовец Н.А. База организации макроконвейерных вычислений, ранжируемых по уровням гарантоспособности. Специальная электроника. 1990. Серия 10, Вып. 1 (26). М. С. 34–37.
2. Палагин А.В. Об ЭВМ с виртуальной архитектурой. УСМ. 1999. № 3. С. 33–43.
3. Косовец Н.А. Об аппаратных средствах повышения надежности отказоустойчивых микропроцессорных систем. Кибернетика и вычислительная. 1993. Выпуск № 99. Сложные системы управления. С. 102–104.
4. Kosovets M., Tovstenko L. The concept of creation of the modern cloud computing on a basis the distributed multiprocessor of real time. Связь. 2017. № 4. С. 71–75.
5. Косовец Н.А. Актуальные задачи проектирования встроенных микроЭВМ или задачи вариации соотношения производительности и надежности микро-ЭВМ. Спец. электроника. М. 1990.

References

1. Kosovets M. Database organization of macro-pipelined computing, ranked by levels of reliability. Coll: Special electronics. 1990 Series 10, issue. 1 (26). М. P. 34–37.
2. Palagin A. About a computer with a virtual architecture. USM. 1999. N 3. P. 33–43.
3. Kosovets M. On hardware means to increase the reliability of fault-tolerant microprocessor systems. Cybernetics and computing. 1993. Issue N 99. Complex management systems. P. 102–104.
4. Kosovets M., Tovstenko L. The concept of creation of the modern cloud computing on a basis the distributed multiprocessor of real time. Journ.: Communications. N 4. 2017. P. 71–75.
5. Kosovets M. Actual problems of designing embedded microcomputers or the task of varying the ratio of performance and reliability of microcomputers. In: Special electronics. Moscow: 1990.

Об авторах:

Косовец Николай Андреевич,
научный сотрудник
кафедры моделирования сложных систем
факультета компьютерных наук и кибернетики КНУ имени Тараса Шевченко.
Количество научных публикаций в украинских изданиях – 40.
Количество научных публикаций в зарубежных изданиях – 5.
Индекс Хирша – 2.
<https://orcid.org/0000-0001-8443-7805>,

Щетинин Игорь Евгеньевич,
старший научный сотрудник,
кандидат технических наук,
временно исполняющий обязанности зав. отделом № 115
Института кибернетики имени В.М. Глушкова НАН Украины.
Количество научных публикаций в украинских изданиях – 30.
<https://orcid.org/0000-0001-8443-7805>,

Товстенко Лилия Николаевна,
ведущий инженер-программист
Института кибернетики имени В.М. Глушкова НАН Украины.
Количество научных публикаций в украинских изданиях – 13.
Количество научных публикаций в зарубежных изданиях – 2.
Индекс Хирша – 2.
<https://orcid.org/0000-0002-3348-6065>.

Место работы авторов:

Киевский национальный университет имени Тараса Шевченко,
02000, Киев, проспект Академика Глушкова, 4д.
Институт кибернетики имени В.М. Глушкова НАН Украины.
03187, Киев-187, проспект Академика Глушкова, 40.
Тел.: +38 (044) 526 2008.
Факс +38 (044) 526 4178.
E-mail: incyb@incyb.kiev.ua