

РЕАЛИЗАЦИЯ СУПЕРКОМПЬЮТЕРОВ С ДИНАМИЧЕСКОЙ АРХИТЕКТУРОЙ НА СОВРЕМЕННОЙ ЭЛЕМЕНТНОЙ БАЗЕ

В. А. Торгашев^а, доктор техн. наук, профессор

И. В. Царев^а, ведущий программист

^аСанкт-Петербургский институт информатики и автоматизации РАН, Санкт-Петербург, РФ

Постановка проблемы: современные суперкомпьютеры обладают рядом недостатков по технико-экономическим характеристикам, кроме того, производительность суперкомпьютеров при решении реальных задач оказывается значительно ниже, чем их теоретическая производительность. **Цель:** разработка принципов создания суперкомпьютеров на основе динамических автоматных сетей, включая различные варианты схмотехнических решений, позволяющих существенно улучшить удельные характеристики суперкомпьютеров. **Результаты:** сформулированы базовые принципы создания суперкомпьютеров с динамической архитектурой на основе динамических автоматных сетей, включая реализацию динамических автоматов с использованием программируемых логических интегральных схем и специально разрабатываемых отечественных больших интегральных схем, которые обеспечивают крайне высокую регулярность структуры микросхем и модулей вычислительной системы, что существенно упрощает создание суперкомпьютеров с динамической архитектурой по сравнению с традиционными суперкомпьютерами и позволяет значительно улучшить такие их характеристики, как габариты, энергопотребление и стоимость. **Практическая значимость:** полученные результаты дают возможность намного сократить сроки и стоимость разработки вычислительных систем со сколь угодно высокой производительностью, обеспечивая при этом неограниченную масштабируемость, крайне высокую надежность и информационную безопасность таких систем.

Ключевые слова — динамические автоматные сети, суперкомпьютеры с динамической архитектурой, элементная база суперкомпьютеров.

Введение

Все современные суперкомпьютеры, или, как их иногда называют, «вычислительные суперсистемы», построены на основе существующих микропроцессоров, большей частью Intel, Sun Microsystems (Sparc), NEC и некоторых других, а также акселераторов на основе графических процессоров, преимущественно фирмы NVIDIA. Общим для всех этих процессоров является то, что они построены на традиционной, или, как ее еще называют, фон-неймановской архитектуре, основанной на алгоритмической, т. е. последовательной модели вычислений. Поскольку основной проблемой при реализации суперкомпьютеров является достижение максимально возможной вычислительной мощности, которая определяется как временем выполнения операций, так и доступными объемами и скоростью работы памяти, то основные достижения в этой сфере до недавнего времени основывались на увеличении тактовой частоты микропроцессоров (уменьшении времени выполнения операций). Это порождало также ряд проблем, приводивших к все большему увеличению сложности самих процессоров и сопутствующих им устройств (памяти, шин, конвейеров и т. п.). В настоящее время увеличение тактовой частоты почти достигло барьера, вызванного чисто физическими причинами, частота работы процессоров практически не увеличивается уже более десяти лет. Дальнейшее увеличение производительности процессоров достигалось в основном за счет увеличения слож-

ности их архитектуры, создания многоядерных процессоров и других решений, которые только увеличивали сложность программного обеспечения (для эффективного использования многоядерных процессоров требуются специальные, достаточно сложные приемы программирования, которые в большинстве программ не используются). Кроме того, увеличение сложности архитектуры процессоров приводит к тому, что все меньшая часть аппаратуры выполняет собственно вычисления, а остальная часть обслуживает дополнительные архитектурные усложнения.

В связи с этим становится ясно, что дальнейшее увеличение производительности суперкомпьютеров возможно только на основе кардинальных изменений в их архитектуре либо в отказе от традиционных принципов построения компьютеров, т. е. в изменении самой модели вычислений. В статье Ю. Н. Митропольского [1], посвященной новым концепциям построения вычислительных суперсистем, с одной стороны, подтверждается то, что было сказано выше, с другой стороны, рассматривается ряд возможных концепций архитектуры, таких как «мультиархитектура», которая предусматривает построение всей системы на основе неоднородной структуры, содержащей специализированные процессоры, ориентированные на выполнение отдельных функций в системе. Рассматриваются также некоторые решения, основанные на изменениях топологии микросхем, уменьшающих время распространения сигналов и обращений к памяти, а также решения, основанные на взаимной адаптации аппаратуры

и программного обеспечения. Рассмотренные концепции за счет специализации позволяют существенно упростить архитектуру отдельных процессоров, но при этом усложняется структура суперсистемы в целом, а также программное обеспечение. Все эти решения выполняются в рамках все той же алгоритмической, последовательной модели вычислений.

Альтернативой традиционной модели являются автоматные сети, предложенные тем же фон-Нейманом в его известной работе о самовоспроизводящихся автоматах [2]. Автоматные сети бывают стационарными, перестраиваемыми и динамическими. На базе программно-перестраиваемых автоматных сетей основаны разработки Научно-исследовательского института мультипроцессорных вычислительных систем Южного федерального университета (НИИ МВС ЮФУ). Данные идеи, впервые изложенные в работах А. В. Каляева [3], развиваются до настоящего времени И. А. Каляевым и его коллегами [4]. В принципе эти разработки позволяют существенно улучшить характеристики мультипроцессорных систем, повышают гибкость архитектуры, поскольку так же, как и рассматриваемые в данной статье, реализуются с использованием программируемых логических интегральных схем (ПЛИС). Однако в архитектуре отдельных процессоров все так же доминирует традиционная алгоритмическая архитектура, но при этом реализация перестраиваемых автоматных сетей требует также достаточно сложного программного обеспечения, причем перестройка автоматной сети осуществляется однократно, перед решением задачи.

Динамические автоматные сети (ДАС) были предложены нами в 80-х гг. прошлого века как теоретическая модель, альтернативная традиционным моделям фон-Неймана и Тьюринга, для представления разнообразных вычислений, в том числе и параллельных, выполняемых в различных распределенных вычислительных средах. Основное отличие ДАС от стационарных и программно-перестраиваемых автоматных сетей заключается в том, что в процессе вычислений ДАС постоянно изменяет свою конфигурацию за счет порождения новых динамических автоматов (ДА), уничтожения ДА, выполнивших свою функцию, а также за счет изменения связей между автоматами. Этот процесс называется «автотрансформацией ДАС». В то же время стационарные автоматные сети принципиально не могут изменять свою конфигурацию, а конфигурация программно-перестраиваемых сетей может быть изменена (перепрограммирована) только один раз, перед решением конкретной задачи. При практической реализации некоторая начальная ДАС по сути является представлением программы для решения какой-либо задачи,

а традиционные последовательные программы, основанные на алгоритмической модели, не применяются.

Динамическая автоматная сеть изменяется до тех пор, пока не потеряет способности к автотрансформации, что происходит при исчезновении активных элементов (автоматов), которые способны порождать новые ДА или подсети. При этом такое неизменяемое состояние ДАС представляет собой некоторую структуру из пассивных элементов, т. е. данных, которая и является решением задачи. Эта структура данных может быть некоторым образом выведена во внешнюю среду, т. е. записана в файл на жестком диске, выведена в удобном для восприятия результатов виде на экран дисплея, на печать и т. д.

Каждый ДА представляет собой конечный автомат, дополненный множеством связей с другими автоматами, которые называются «смежными», а также так называемой «функцией связей» (или множеством таких функций). Последняя как раз и предназначена для порождения или уничтожения ДА, а также для изменения связей между автоматами в ДАС. В общем случае любой ДА может быть отнесен к одному из двух классов: «операционный автомат», выполняющий некоторую вычислительную функцию, и «ресурсный автомат», представляющий данные.

Теоретическая модель ДАС может быть использована на практике для реализации различных высокопроизводительных вычислительных систем, включая распределенные вычислительные системы. Наиболее перспективным направлением является реализация суперкомпьютеров с динамической архитектурой (СКДА), удельные характеристики которых, такие как энергопотребление, габариты и стоимость, оказываются в десятки, а то и в сотни раз меньшими, чем соответствующие характеристики существующих суперкомпьютеров при сопоставимой производительности. Основная причина этого заключается в том, что в СКДА не используются сложные последовательные программируемые процессоры, не используются высокие тактовые частоты, при которых аппаратура требует мощной системы охлаждения, а используются весьма простые по структуре и функциям автоматы. Число ДА, реализуемых в одной микросхеме, в зависимости от применяемой элементной базы и конкретных архитектурных и схемотехнических решений может составлять сотни тысяч или миллионы, при этом все автоматы работают параллельно, полностью асинхронно и не требуют никакого централизованного управления. Другим важным свойством СКДА, отличающим их от суперкомпьютеров с традиционной архитектурой, является неограниченная масштабируемость, дающая возможность наращивать вычислительную

мощность без каких бы то ни было изменений в структуре как аппаратуры, так и программного обеспечения.

Более подробные варианты рассмотрения модели ДАС, а также структуры и функций отдельных ДА можно найти в работах [5–9].

Возможность реализации компьютеров на основе модели ДАС и их высокая эффективность впервые были показаны на примере спроектированной, изготовленной и прошедшей государственные испытания «машины с динамической архитектурой» (МДА) ЕС-2704 в конце 80-х гг. Во время испытаний, проходивших в Научно-исследовательском центре электронной вычислительной техники (НИЦЭВТ, Москва), где МДА и была изготовлена, ряд тестовых задач выполнялся как на МДА, так и на типовой для того времени вычислительной машине ЕС-1055. При этом МДА показала в десятки и сотни раз более высокую производительность в зависимости от решаемой задачи, чем ЕС-1055, притом что она была выполнена на той же элементной базе и в том же конструктивном оформлении, что и ЕС-1055, занимая при этом всего одну стандартную стойку ЕС ЭВМ. Производительность сравнивалась не по скорости выполнения отдельных операций, а по времени решения задачи в целом. Было выпущено несколько экземпляров ЕС-2704, которые успешно работали в ряде организаций СССР.

Однако с тех пор элементная база и некоторые архитектурные и технические решения устарели. В 90-х гг. и позднее вся вычислительная техника в мире была ориентирована, как было указано выше, на типовые микропроцессоры, которые не позволяют реализовать ДАС иначе, чем посредством программно-аппаратной эмуляции, что не обеспечивает требуемую эффективность. Появление в середине 90-х гг. ПЛИС, а также их развитие в настоящее время способствуют возвращению к идее МДА с использованием всех преимуществ ДАС и МДА, дающих возможность получить в десятки и сотни раз большую производительность, чем при использовании традиционной архитектуры, при том же объеме аппаратных средств.

Данная статья посвящена некоторым аспектам современной реализации и применения вычислительных систем на основе ДАС.

Варианты реализации СКДА на современной элементной базе

Поскольку ДАС могут включать в свой состав любое, сколь угодно большое число автоматов, то и СКДА должен иметь возможность включать в свой состав произвольное число устройств, обеспечивающих реализацию автоматов. Во избежание терминологической путаницы элементы

ДАС будем называть динамическими автоматами, а элементы СКДА — физическими или функциональными автоматами (ФА). Каждый ФА может находиться в занятом или свободном состоянии. ФА является занятым, если в нем размещается ДА, определяющий функцию, которую выполняет ФА. Любой ДА представлен в СКДА в виде информационного пакета определенной структуры, одинаковой для любых типов ДА. Информация, включая фрагменты программы, в СКДА представлена исключительно в форме ДА, что обеспечивает высокий уровень информационной безопасности. Последнее обеспечивается тем, что любой ДА, включенный в ДАС, доступен для использования или изменения только смежным элементам ДАС. Это определяется структурой сетевой программы и ничем другим. В принципе возможно одновременное выполнение нескольких программ, представленных в форме ДАС, в одном СКДА, но элементы этих различных программ никак между собой не связаны и никаким образом не могут получить доступ к элементам другой программы. Это обеспечивается посредством особой организации адресного пространства памяти, позволяющей адресовать в данном автомате только элементы (дескриптор и тело) данного автомата и автоматов, смежных с ним.

На абстрактном уровне СКДА можно представить как множество одинаковых ФА, образующих регулярную двумерную структуру, каждый элемент которой соединен с четырьмя соседями. Каждое соединение является многоканальным, причем каналы распределяются как в пространстве, так и во времени. Каждый ФА, независимо от его состояния, обеспечивает автоматическое динамическое распределение и коммутацию проходящих через него каналов и передачу по этим каналам информационных пакетов, соответствующих ДА.

Такие фундаментальные свойства ДАС, как порождение и уничтожение ДА, в СКДА реализуются достаточно просто. Если некоторый ДА решит создать потомков, то ФА, в котором он находится, устанавливает связь с ближайшим свободным ФА и передает туда модифицированную копию родителя, после чего потомок начинает самостоятельную деятельность. Для ликвидации ДА после завершения выполняемой функции или истечения времени жизни, определенного при рождении, ДА разрывает все связи с другими ДА, включая родителя. При этом ФА переходит в свободное состояние, стирая информацию о погибшем ДА. ДА может быть уничтожен принудительно, если все ДА, связанные с ним, разорвут свои связи. В каждом ФА имеются механизмы, позволяющие выявлять и обрывать случайно сохранившиеся связи.

Далее рассмотрим варианты реализации МДА на базе современных технологий. Следует отметить, что, наряду с суперкомпьютерами, на основе ДАС возможно создание и небольших компьютеров с динамической архитектурой для конкретных применений, для обозначения которых будем употреблять исторически устоявшийся термин «машина с динамической архитектурой». Далее в тексте статьи можно считать термины «СКДА» и «МДА» эквивалентными, разумеется, имея в виду, что «суперкомпьютерами» все же называются высокопроизводительные системы.

Реализация МДА на базе ПЛИС

Для создания МДА любой производительности и различного назначения, начиная от небольших, но мощных бортовых систем и кончая стационарными суперкомпьютерами, достаточно использовать лишь два типа модулей: вычислительный модуль с динамической архитектурой (ВМДА) и интерфейсный модуль с динамической архитектурой (ИМДА). Далее в данном разделе имеется в виду реализация на основе вполне определенных микросхем — ПЛИС фирмы Altera, о выборе которых речь пойдет ниже, но вышеприведенные соображения о структуре реализации справедливы и для любых других вариантов реализации, включая рассмотренные в следующем разделе.

Любой из этих модулей, реализуемый на современной элементной базе, размещается на печатной плате размером 110×110 мм. На каждой стороне печатной платы размещаются два одноэлементных (one-piece) разъема размерами 45×10 мм и с 60 линиями подключения. Разъемы используются для соединения между собой печатных плат. На одну печатную плату устанавливается разъем, а вторая печатная плата привинчивается к разъему так, чтобы имеющиеся на ее поверхности контактные площадки попали на пружинные контакты разъема.

На плате ВМДА устанавливаются четыре ПЛИС, каждая из которых соединена с двумя другими ПЛИС и с двумя разъемами. К каждой ПЛИС подключается динамическая память и энергонезависимая память, обеспечивающая настройку ПЛИС и хранение данных.

На плате ИМДА устанавливаются две ПЛИС, энергонезависимая память, обеспечивающая настройку ПЛИС, и два дополнительных одноэлементных разъема, на которые в мезонинном стиле устанавливается плата интерфейсного блока ДА. Одна из ПЛИС ИМДА соединена со всеми восьмью основными разъемами модуля, а также со второй ПЛИС и выполняет исключительно коммутационные функции. В этом плане она полностью эквивалентна четырем ПЛИС

ВМДА. Вторая ПЛИС является контроллером интерфейсного блока. Интерфейсный блок может иметь множество разновидностей в зависимости от вида входной и выходной информации (цифровая или аналоговая), от способов физического преобразования сигналов (гальваническая развязка, преобразование уровней напряжения), от типа внешних устройств (цифровые каналы, видеокамеры, радиолокационные или гидроакустические станции, датчики и устройства управления для автоматических систем и т. д.). Как один из вариантов интерфейсного блока можно рассматривать обычный микропроцессор, выполняющий последовательные программы, не требующие высокой производительности. ВМДА и ИМДА могут входить в состав МДА в любом соотношении. Как вырожденный случай можно представить МДА, содержащий только интерфейсные модули с микропроцессорными блоками. В такой МДА может одновременно выполняться большое число обычных последовательных программ. При этом сохраняются такие свойства МДА, как сверхвысокая надежность и информационная безопасность.

Соединение модулей в горизонтальной плоскости выполняется с помощью печатных плат размерами 45×20 мм, привинчивающихся к одноэлементным разъемам смежных модулей. Для вертикальных соединений используются гибкожесткие платы двух типов с длиной гибкой части 40 и 90 мм соответственно при условии, что межмодульное расстояние по вертикали составляет 25 мм. Жесткие части этих плат привинчиваются к разъемам модулей, расположенных друг над другом либо непосредственно, либо минуя два модуля. Для вертикального крепления используются разборные стойки, расположенные по четырем углам каждого модуля.

Структура МДА, как и любой ее части (модуля, платы), в высшей степени регулярна, независимо от конкретной реализации. С одной стороны, это сильно упрощает процесс проектирования конкретной аппаратной реализации, с другой стороны, не имеет смысла иллюстрировать эти структуры рисунками, поскольку любой из них будет содержать некоторую матрицу из прямоугольных элементов (представляющих собой различные автоматы), соединенных горизонтальными и вертикальными стрелками, так что не будет особенно информативным.

Определимся теперь с выбором ПЛИС для реализации МДА. У фирмы Altera имеется два больших семейства ПЛИС: высокопроизводительные микросхемы серии Stratix IV, выполненные по технологии 40 нм, и дешевые (так их называют в самой фирме Altera) микросхемы серии Cyclone IV, выполненные по технологии 60 нм. В каждое семейство входит полтора десятка микросхем, отли-

чающихся по числу транзисторов и по архитектуре. При сравнении наиболее мощных представителей каждого семейства EP4S100G5 (Stratix IV) и EP4CGX150 (Cyclone IV) можно заметить следующее: первая микросхема имеет производительность в 6 раз выше и стоимость в 50 раз больше. При сравнении других представителей семейств это соотношение сохраняется. Таким образом, экономическая эффективность семейства Stratix на порядок ниже, чем у Cyclone. Поскольку производительность МДА обеспечивается в первую очередь за счет количества автоматов (и, соответственно, ПЛИС), а не скорости их работы, то выбор в пользу семейства Cyclone становится очевидным. Хотя во всем мире при создании высокопроизводительных вычислительных устройств на ПЛИС однозначный выбор делается в пользу семейства Stratix или аналогичных микросхем семейства Virtex фирмы Xilinx.

Наибольшей экономической эффективностью среди ПЛИС семейства Cyclone обладает микросхема EP4CE40. При рабочей частоте 250 МГц обеспечивается производительность 58 Гфлопс при операциях с плавающей запятой. При обработке цифровых сигналов, например с помощью алгоритмов быстрого преобразования Фурье, производительность ПЛИС возрастает в 3 раза и составляет 174 млрд арифметических операций в секунду. Относительно небольшое энергопотребление микросхемы (менее 2 Вт при максимальной загрузке всех автоматов) позволяет обойтись без принудительного охлаждения. Следует заметить, что энергопотребление ПЛИС пропорционально числу одновременно работающих автоматов. Если загрузка автоматов и, соответственно, производительность уменьшаются в несколько раз, то во столько же раз падает и энергопотребление.

В целом модуль ВМДА обеспечивает производительность 232 Гфлопс при энергопотреблении не более 10 Вт. Объем оперативной памяти модуля составляет 2 ГБ. Объем энергонезависимой памяти — 16–128 ГБ в зависимости от типа установленных микросхем. Пропускная способность коммуникационных каналов ВМДА, а также ИМДА — 12 ГБ/с. Пропускная способность оперативной памяти ВМДА — 16 ГБ/с. Минимальная МДА состоит из двух модулей, один из которых должен быть интерфейсным.

Итак, впервые в истории вычислительной техники у пользователя появляется возможность самому создавать вычислительное устройство с любой сколь угодно большой производительностью, с различными типами интерфейсов в сочетании со сверхвысокой надежностью и информационной безопасностью. Устройство создается на базе небольшого набора комплектующих, включающего ВМДА, ИМДА, один или несколько типов

интерфейсных блоков, три типа соединительных печатных плат, разборные стойки, элементы корпуса. Для сборки МДА любых размеров требуется лишь отвертка для привинчивания соединительных плат к разъемам и элементов корпуса к стойкам и гаечный ключ для завинчивания стоек. Полный комплект программного обеспечения МДА находится в энергонезависимой памяти каждого из поставляемых ВМДА или ИМДА. В комплект поставки включается также пакет программ для терминального компьютера, обеспечивающий взаимодействие пользователя с МДА.

Машина с динамической архитектурой может применяться для обработки информации непосредственно на борту космического аппарата или беспилотного летательного аппарата. Учитывая суперкомпьютерную производительность даже небольших МДА, состоящих всего из нескольких модулей, можно существенно сократить объем передаваемой на Землю информации, выполняя частичную или полную обработку этой информации непосредственно на борту.

Необходимо отметить, что любая МДА является не только высокопроизводительным вычислительным устройством, но и мощным коммуникационным узлом, способным коммутировать большое число различных цифровых каналов со скоростями до 3,125 ГБ/с. Поэтому МДА может эффективно использоваться в узлах любых информационных сетей, обеспечивая не только коммутацию каналов, но и обработку информации.

Возможности создания МДА на отечественной элементной базе

Описанная выше реализация МДА на базе ПЛИС имеет как некоторые достоинства, так и недостатки. К первым можно отнести относительную легкость проектирования (в том числе программирования ПЛИС), возможность приобретения и использования уже существующих микросхем, не слишком дорогих, учитывая приведенный выше выбор ПЛИС. К недостаткам можно отнести, во-первых, необходимость использования импортных микросхем, что не всегда желательно, например, для систем оборонного назначения. Во-вторых, существенным недостатком является некоторая избыточность ПЛИС, связанная с гибкостью их структуры и возможностью программирования этой структуры. Значительная часть элементов ПЛИС предназначена как раз для этого.

Далее рассмотрим возможность реализации МДА на основе отечественной элементной базы. При этом имеется в виду не использование существующих отечественных микросхем, а проектирование и изготовление БИС, специально предназначенных для реализации МДА. Такой под-

ход позволяет избавиться от обоих недостатков использования ПЛИС, указанных выше, существенно увеличивает эффективность реализации МДА и позволяет еще более повысить удельные характеристики реализации. Анализируется реализация с использованием технологий, доступных в настоящее время в нашей стране.

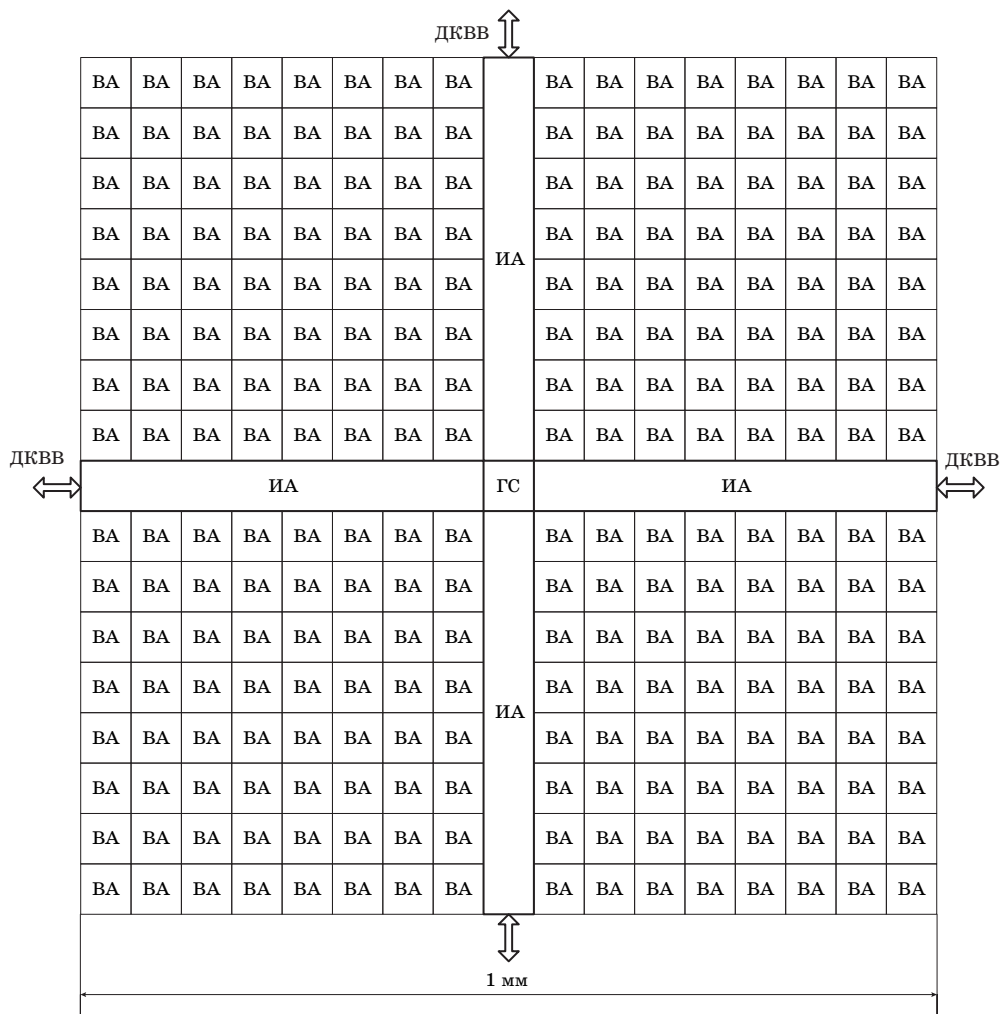
Исследуем непосредственную реализацию МДА в микроэлектронной структуре. Назовем двумерную структуру, содержащую $N \times N$ автоматов ФА и выполненную в виде микросхемы, динамической электронной структурой с размерностью N (ДЭС- N). Например, микросхема ДЭС-32 содержит 1024 автомата ФА, образующих структуру 32×32 .

Поскольку внешние выводы размещаются по краям структуры, естественным является применение планарного корпуса с расстоянием между выводами 0,5 мм. Для соединения между ФА используются дуплексные последовательные каналы с двумя физическими линиями. Крайним ФА соответствуют два внешних вывода корпуса (для

угловых ФА — четыре вывода). Соответственно ФА на кристалле занимает площадь ровно 1 мм^2 , а площадь микросхемы ДЭС-32 составляет 1024 мм^2 , независимо от используемой технологии.

Каждый автомат ФА содержит 256 вычислительных автоматов ВА, четыре интерфейсных автомата ИА и генератор синхроимпульсов ГС; по границам ФА проходят двухпроводные дуплексные каналы ввода-вывода ДКВВ. Примерное размещение элементов ФА показано на рисунке.

В состав каждого ВА входят следующие устройства: перепрограммируемое запоминающее устройство типа флеш-памяти, оперативное запоминающее устройство динамического типа и процессор, состоящий из примитивного операционного устройства и устройства управления. Операционное устройство осуществляет последовательную побитовую обработку информации. Простые операции типа сложения выполняются за n тактов, где n — число двоичных разрядов. Для выполнения операций умножения и деления требуется n^2 тактов.



■ Структура функционального автомата

Работа всех устройств, входящих в состав ФА, осуществляется синхронно с помощью ГС. С максимальной частотой f_{\max} работает операционное устройство каждого ВА. На этой же частоте осуществляется передача информации по последовательным каналам между ИА. Передача информации между ВА, между ВА и ИА, а также между процессором и устройствами памяти в каждом ВА осуществляется по последовательным каналам с частотой в 16 раз более низкой, чем f_{\max} .

Запоминающие устройства занимают 98 % от объема (числа транзисторов) ВА. Менее 1 % объема процессора (1/5000 от объема ВА) составляет операционное устройство. Однако на это устройство, несмотря на его пренебрежимо малые размеры, приходится более 98 % всего энергопотребления ВА. Поэтому величина энергии, потребляемой ФА, определяется числом активных ВА, занятых вычислениями. Свободный автомат или занятый, но ожидающий получения данных, практически не потребляет энергии.

Каждый ВА связан с четырьмя соседями. Однако у крайних автоматов в матрице имеются лишь три соседа, а у угловых автоматов — лишь два. Обычно каналы, соответствующие отсутствующим связям, остаются неиспользованными. Но в микросхеме ДЭС-1 все неиспользованные связи подаются на внешние выводы микросхемы. Четыре основных канала ФА используются для соединения с другими элементами МДА, а 64 дополнительных канала — для подключения к внешним устройствам либо напрямую, либо с помощью интерфейсного блока. Эти каналы разбиты на четыре банка, по 16 каналов в банке. Для каждого банка может быть определено свое напряжение (1; 1,2; 1,5; 1,8; 2,5; 3,3 В), подаваемое извне, а также тип связей (однопроводные или двухпроводные), подобно тому, как это делается у ПЛИС фирмы Altera. Таким образом, ДЭС-1 является универсальным интерфейсным контроллером, обеспечивающим связь МДА с любыми внешними устройствами.

Рассмотрим теперь возможные параметры микросхемы ДЭС-32, реализованной по технологии 90 нм, которая к настоящему времени освоена на предприятиях электронной промышленности России. При использовании данной технологии на площади 1 мм^2 можно разместить примерно 120 млн транзисторов. Соответственно, объем ВА составит 480 тыс. транзисторов. Этого количества транзисторов достаточно для размещения 36 КБ перепрограммируемой памяти и 9 КБ динамической памяти. Энергопотребление ВА пропорционально частоте работы операционного устройства f_{\max} . Если эта частота равна 1 ГГц, то энергопотребление ВА составит 40 мкВт. При этом производительность ВА при вычислениях с плавающей запятой двойной точности составит

4 Мфлопс. При обработке изображений или других цифровых сигналов с помощью алгоритмов быстрого преобразования Фурье производительность ВА составит 24 млн оп./с.

Интерфейсный автомат по структуре аналогичен ВА, но без энергонезависимой памяти. В то же время оперативная память увеличена до 72 КБ.

Для ДЭС-32 объем энергонезависимой/оперативной памяти составит 9/2,25 ГБ. При тактовой частоте 1 ГГц производительность ДЭС-32 составит 1–6 трлн оп./с в зависимости от классов решаемых задач. В случае полной загрузки всех ВА энергопотребление такой ДЭС-32 составит 10 Вт.

Следует сказать, что тактовая частота каждого ФА микросхемы ДЭС-32 может изменяться в широком диапазоне от 1 ГГц до десятков кГц даже непосредственно в процессе работы. Соответственно, будет изменяться производительность и энергопотребление. Микросхема ДЭС-32 может автоматически поддерживать энергопотребление не выше заданного уровня.

Микросхема ДЭС-32 может быть размещена в корпусе размером $40 \times 40 \text{ мм}$ с 320 планарными выводами, 64 из которых используются для подачи питания и земли. Из одной и той же полупроводниковой пластины можно получать также микросхемы ДЭС-16, ДЭС-8 и ДЭС-4. Последняя микросхема при размерах корпуса $5 \times 5 \text{ мм}$ может обеспечить реальную производительность 100 млрд оп./с при энергопотреблении всего 160 мВт.

Динамическую электронную структуру можно рассматривать как пример «вечной» микросхемы, которую можно вывести из строя только с помощью механического воздействия. Действительно, большая часть неисправностей будет приходиться на ячейки памяти, которые занимают 98 % объема ДЭС. Однако каждый ВА непрерывно контролирует состояние своей памяти и в случае возникновения неисправности исключает соответствующий блок памяти из списка доступных. Тем самым немного уменьшаются объем памяти ВА, что практически никак не сказывается на его функционировании. Выход из строя ВА уменьшает производительность ФА, в состав которого он входит, всего на 0,25 %. Если, однако, число отказавших ВА окажется достаточно большим, так что снижение производительности ФА станет заметным, автоматически увеличится тактовая частота, и производительность ФА восстановится до исходного значения. При этом энергопотребление ФА сохранится на прежнем уровне, так как отказавшие ВА энергии не потребляют. Аналогичная ситуация возникает и на уровне микросхемы. Если число отказавших ФА составляет заметную величину, каждый из исправных ФА повышает свою тактовую частоту, чтобы восстановить производительность

микросхемы до исходного значения. Итак, несмотря на полное отсутствие резервирования микросхемы ДЭС при наличии множественных отказов элементов, число которых может составлять десятки тысяч, не только сохраняет работоспособность, но и поддерживает номинальное значение производительности. Это свойство позволяет использовать ДЭС в условиях сильного радиационного или космического излучения, где обычная электроника мгновенно выходит из строя. МДА, выполненные на базе ДЭС, независимо от размеров могут работать сколь угодно долгое время с номинальными параметрами без какого-либо ремонта. Конструкцию такой МДА можно делать неразборной.

Возможность регулировки теплового режима микросхемы в сочетании с рассмотренным в предыдущем разделе методом соединения элементов МДА позволяет получать многослойные трехмерные БИС в целях создания компактных сверхнадежных максимально защищенных баз данных в сочетании с мощной поисковой системой. Например, уложив друг на друга 64 кристалла ДЭС-64 и соединив по вертикали их выводы, получим структуру с размерами 80×80×40 мм (256 тыс. ФА или 65 млн ВА) и емкостью энергонезависимой памяти более 2 ТБ (10^{12} Б). При тактовой частоте 2 МГц и энергопотреблении не более 5 Вт данная структура может выполнять до 16 трлн операций сравнения символов в секунду. Пропускная способность каналов ввода-вывода такого устройства составит 32 МБ/с в каждом направлении.

Итак, реализация МДА исключительно на отечественной элементной базе вполне возможна в относительно короткие сроки и при относительно небольших финансовых затратах. Однако до начала разработки микросхем типа ДЭС необходимо проверить все аппаратные и программные решения на образцах МДА, выполненных на ПЛИС фирмы Altera, и экспериментально доказать все рассмотренные выше преимущества МДА. На этих же образцах можно осуществить функциональное моделирование основных узлов ДЭС.

Машины с динамической архитектурой в автоматических и автоматизированных системах управления

Применение МДА в автоматических и автоматизированных системах управления позволяет получить такие характеристики, которые принципиально недостижимы при использовании микропроцессоров с традиционной архитектурой.

Пожалуй, главным достоинством систем, использующих МДА, являются исключительно высокая надежность и живучесть. Даже простейшие автоматические системы, например управ-

ляющие самонаведением ракет или торпед, при использовании МДА будут состоять из большого числа одинаковых простых автоматов, каждый из которых выполняет лишь малую часть общей задачи. При выходе какого-либо автомата из строя та часть задачи, которую он выполнял, автоматически перейдет в другой исправный автомат. Даже одновременный выход из строя нескольких автоматов не приводит к отказу системы. Это свойство особенно ценно для систем, работающих в небезопасных системах в условиях космических или радиационных излучений. Для обычного микропроцессора любое повреждение фатально. Обычное резервирование ситуации качественно не изменяет. Поэтому вблизи радиационных источников электроника быстро выходит из строя. При использовании же МДА время жизни систем в условиях радиационных излучений возрастает многократно. Необходимо отметить, что таким свойством обладают лишь ДАС, в отличие от стационарных и перестраиваемых автоматных сетей, в которых выполняемые задачи жестко привязаны к конкретным автоматам.

Не менее важным свойством МДА является исключительно высокая информационная безопасность, в принципе недостижимая для традиционных архитектур. Для любых вычислительных систем существуют две основные угрозы безопасности: неумышленные ошибки системных и прикладных программистов и умышленное создание программ с вредоносными свойствами (программист-диверсант). Реальные методы борьбы с этими угрозами в системах традиционного типа отсутствуют. Автоматические методы тестирования программ позволяют выявить лишь типовые грубые ошибки, а обнаружить вредоносную программу, созданную квалифицированным программистом, крайне затруднительно. Низкая информационная безопасность современных компьютеров обусловлена тем, что программы, представленные на машинном языке, имеют непосредственный, практически неконтролируемый доступ к физическим ресурсам (оперативной или дисковой памяти). В МДА и программы, и данные независимо от их размещения представлены в форме информационных пакетов — программных элементов (ПЭ). Любой ПЭ включает в свой состав служебную информацию, определяющую тип ПЭ (данные, операторы, ссылки и т. д.) и указатели на те ПЭ, с которыми данный ПЭ связан и которые (и только они) могут иметь доступ к ПЭ по определенному протоколу, включающему в ряде случаев динамическую систему паролей. Поэтому никакая программа в МДА не может иметь несанкционированный доступ к любой другой программе, поскольку контроль доступа осуществляется на аппаратном уровне, что и обеспечивает практически абсолютную

информационную безопасность даже в случае системного программиста-диверсанта. Необходимо отметить, что если архитектура современных микропроцессоров и операционных систем общеизвестна, то архитектура МДА известна лишь узкому кругу разработчиков. Поэтому никакие хакеры или компьютерные вирусы не смогут нанести ущерб МДА хотя бы в силу отсутствия информации об архитектуре.

Применение МДА позволяет вообще отказаться от использования импортных микросхем. На уже существующих в России технологиях (например, 90 нм или даже 130 нм) можно создать микросхему ДЭС, которая по таким характеристикам, как производительность и энергоэффективность, намного превзойдет не только современные зарубежные микропроцессоры, выполненные по технологии 22–28 нм, но и те, технологии для которых еще только разрабатываются, а их производство планируется не ранее 2020 г. Поскольку ДЭС будет включать в свой состав оперативную и энергонезависимую память, распределенную по автоматам, номенклатура комплектующих изделий для МДА любой производительности может состоять лишь из одной микросхемы отечественного производства.

Основным фактором, ограничивающим возможности современных автоматических и автоматизированных систем управления, является производительность существующих микропроцессоров. Наиболее мощным российским микропроцессором является четырехъядерный МЦСТ-Р1000, выполненный по технологии 90 нм, с частотой 1 ГГц, пиковой производительностью 8 Гфлопс и энергопотреблением 20 Вт. Однако реальная производительность микропроцессора может приближаться к указанному значению лишь при вычислении скалярного произведения векторов, когда в одном такте могут выполняться две операции — умножения и сложения. В остальных случаях производительность снижается по меньшей мере в 2 раза даже в случае оптимального программирования при распараллеливании задачи на четыре ядра. При невысокой квалификации программистов реальная производительность может снизиться еще в десятки раз. Разделить задачу на несколько микропроцессоров далеко не всегда представляется возможным, тем более что каналы сетевого обмена между микропроцессорами Р1000 достаточно медленные (не более 10 МБ/с). В то же время имеется ряд задач (обработка радиолокационных или гидроакустических сигналов, обработка изображений, ситуационное моделирование), для эффективного решения которых требуется производительность, измеряемая сотнями и тысячами миллиардов операций в секунду. В итоге разработчикам автоматизированных си-

стем управления приходится либо существенно ограничивать характеристики решаемых задач, либо пытаться решить задачу на большом числе процессоров, что далеко не всегда представляется возможным, даже не принимая во внимание трудности программирования. При использовании МДА проблема производительности даже не встает. Любая требуемая производительность легко достигается включением в состав системы достаточного числа автоматов. При этом ни рабочие частоты, ни уровень интеграции особой роли не играют. Вопросы распараллеливания в этом случае решаются автоматически, без какого-либо участия программистов. В МДА отсутствуют такие операции, как условные и безусловные переходы, обращение к памяти, обработка прерываний, вызов процедур, занимающие в традиционных микропроцессорах значительную часть времени. Эффективность программ с точки зрения производительности практически не зависит от программиста. Итак, применение МДА, в отличие от традиционных микропроцессоров, позволяет решать в полном объеме практически любые задачи автоматизированной системе управления.

Машина с динамической архитектурой позволяет создавать вычислительные устройства, отличающиеся существенно большей производительностью и меньшим энергопотреблением по сравнению с традиционными микропроцессорами. В случае непосредственной реализации МДА в виде специально разработанной микросхемы ДЭС-32 можно получить производительность порядка 6 трлн оп./с при энергопотреблении не более 10 Вт. При этом энергопотребление микросхемы пропорционально реальной производительности. Если для конкретной задачи требуется производительность 6 млрд оп./с, то энергопотребление составит лишь 10 мВт. В результате можно создавать бортовые системы с производительностью, не достижимой для современных наземных систем при энергопотреблении, измеряемом единицами ватт.

Характерной особенностью МДА, выполненных на базе современных ПЛИС фирмы Altera, является возможность непосредственно подключать их к любым цифровым внешним устройствам и цифровым каналам со скоростями передачи до 28 Гбит/с. В частности, можно без промежуточных интерфейсных схем подключаться к современным микропроцессорам по шинам PCI или PCIe, к дисковой памяти по шине SATA и к другим современным интерфейсам, включая LVDS-каналы. Лишь для подключения к каналу USB требуется соответствующий микроконтроллер. Поэтому вычислительные устройства на базе МДА отличаются минимальной номенклатурой применяемых микросхем. Кроме ПЛИС, в со-

став МДА обязательно входят микросхемы оперативной и энергонезависимой памяти, а также конвертеры напряжения, обеспечивающие питание ПЛИС. Основная часть выводов ПЛИС используется для соединения между собой ПЛИС, которые образуют автоматную сеть, а также для подключения к микросхемам памяти. Часть выводов у некоторых ПЛИС может использоваться для подключения к различным интерфейсам. Указанная особенность МДА позволяет полностью унифицировать любые автоматические и автоматизированные системы управления на структурном, конструктивном и программном уровнях, а также по номенклатуре комплектации.

В целом применение МДА позволяет создавать автоматические и автоматизированные системы управления, качественно превосходящие мировой уровень.

Помимо вышеописанных задач создания автоматизированных систем управления, а также некоторого множества характерных для суперкомпьютеров задач, перечисленных в статье [10], МДА могут быть использованы и для решения более сложных в структурном смысле задач. Наиболее интересной является возможность использования МДА в робототехнических системах, поскольку в них сочетается множество задач искусственного интеллекта (распознавание образов, анализ сцен, принятие решений, обучение и др.) с задачами автоматического управления механическими и прочими системами робота. В то же время, поскольку МДА является весьма компактным устройством при высокой производительности, вполне возможно размещение такой

системы в корпусе любого робота. Следует также отметить, что для решения задач искусственного интеллекта возможно создание на основе ДАС нейронных сетей.

Заключение

Предложенные в статье методы реализации СКДА, основанных на теории ДАС, с использованием как существующих микросхем ПЛИС, так и специально разрабатываемых отечественных БИС, являются полностью оригинальными и не имеют аналогов в мире. Других работ в области теории и применения ДАС, а также разработки МДА/СКДА в последние четверть века никем не предлагалось.

Главным практическим результатом работы является возможность создания отечественной элементной базы для реализации СКДА, характеристики которых существенно превосходят характеристики существующих суперкомпьютеров по габаритам, стоимости и энергопотреблению при сопоставимой производительности. При этом обеспечивается неограниченная масштабируемость вычислительной системы, а также крайне высокая надежность и информационная безопасность.

Работа поддержана Программой фундаментальных научных исследований отделения нанотехнологий и информационных технологий АН «Архитектурно-программные решения и обеспечение безопасности суперкомпьютерных информационно-вычислительных комплексов новых поколений».

Литература

1. Митропольский Ю. И. Новые концепции построения вычислительных суперсистем // Тр. Физико-технологического института РАН. 2016. Т. 25. С. 22–37.
2. J. von Neuman. Theory of Self-reproducing Automata. — Urbana and London: University of Illinois Press, 1966. — 403 p.
3. Каляев А. В. Многопроцессорные системы с программируемой архитектурой. — М.: Радио и связь, 1984. — 240 с.
4. Каляев И. А., Левин И. И. Многопроцессорные вычислительные структуры с динамически реконфигурируемой архитектурой на основе ПЛИС // Сб. науч. тр. ИТМиВТ. 2008. № 1. С. 44–45.
5. Торгашев В. А. Автоматные сети и компьютеры: история развития и современное состояние // История информатики и кибернетики в Санкт-Петербурге (Ленинграде): сб. — СПб.: Наука, 2012. — С. 46–66.
6. Торгашев В. А. Динамические автоматные сети // Тр. СПИИРАН. 2013. Вып. 4(27). С. 23–34.
7. Торгашев В. А., Царев И. В. Средства организации параллельных вычислений и программирования в мультипроцессорах с динамической архитектурой // Программирование. 2001. № 4. С. 53–68.
8. Торгашев В. А., Царев И. В. Семейство суперкомпьютеров с динамической архитектурой — концептуальные основы // Искусственный интеллект. 2009. № 3. С. 251–257.
9. Торгашев В. А., Царев И. В. Динамические автоматные сети как модель параллельных вычислений в мультипроцессорах с динамической архитектурой // Вестник компьютерных и информационных технологий. 2009. № 3. С. 11–20.
10. Торгашев В. А., Царев И. В. Технологии решения сложных задач на основе динамических автоматных сетей // Информационно-управляющие системы. 2015. № 6. С. 57–65. doi:10.15217/issn1684-8853.2015.6.57

UDC 681.3.01:681.3.02:681.325.6

doi:10.15217/issn1684-8853.2016.6.74

Modern Circuitry Implementation of Dynamic Architecture SupercomputersTorgashev V. A.^a, Dr. Sc., Tech, Professor, tor@spiiras.nw.ruTsarev I. V.^a, Leading Programmer, civ@iiias.spb.su^aSaint-Petersburg Institute for Informatics and Automation of RAS, 39, 14 Line, V. O., 199178, Saint-Petersburg, Russian Federation

Introduction: Modern supercomputers have a number of shortcomings concerning their technical and economic characteristics. Besides, the performance of supercomputers when solving real problems is significantly lower than their theoretical performance. **Purpose:** We develop ways to build supercomputers based on dynamic automata networks, including special circuitry solutions which allow you to considerably improve the specific characteristics of the supercomputers. **Results:** We have formulated the fundamental principles of creating dynamic architecture supercomputers based on dynamic automata networks, including the implementation of dynamic automata with programmable logic chips and custom domestic LSI chips, providing very high structure regularity for the chips and modules of the system. This considerably simplifies creating supercomputers with dynamic architecture as compared to the conventional supercomputers, and can significantly improve them in terms of their size, energy consumption and cost. **Practical relevance:** The obtained results allow you to considerably reduce the time and cost of developing computational systems with unlimitedly high performance, providing unlimited scalability, extremely high reliability and informational security of such systems.

Keywords — Dynamic Automata Networks, Dynamic Architecture Supercomputers, Supercomputer Circuitry.

References

1. Mitropolsky Y. I. New Concepts for Construction of Computing Supersystems. *Trudy Fiziko-tekhnologicheskogo instituta RAN*, 2016, vol. 25, pp. 22–37 (In Russian).
2. J. von Neuman. *Theory of Self-reproducing Automata*. Urbana and London, University of Illinois Press, 1966. 403 p.
3. Kalyayev A. V. *Mnogoprotsessornye sistemy s programmamiyemoy arkhitekturoi* [Multiprocessor Systems with Programmable Architecture]. Moscow, Radio i sviaz' Publ., 1984. 240 p. (In Russian).
4. Kalyayev I. A., Levin I. I. Multiprocessor Computing Structures with Dynamically Reconfigurable Architecture on the Base of FPGA. *Sbornik nauchnykh trudov ITMiVT*, 2008, no. 1, pp. 44–45 (In Russian).
5. Torgashev V. A. Automata Networks: History of Development and Contemporary State. In: *Istoriia informatiki i kibernetiki v Sankt-Peterburge (Leningrade)* [History of Informatics and Cybernetics in Saint-Petersburg (Leningrad)]. Saint-Petersburg, Nauka Publ., 2012, vol. 3, pp. 46–66 (In Russian).
6. Torgashev V. A. Dynamic Automata Networks. *Trudy SPIIRAN*, 2013, vol. 4(27), pp. 23–34 (In Russian).
7. Torgashev V. A., Tsarev I. V. Means for Organization of Parallel Computations and Programming in Multiprocessors with Dynamic Architecture. *Programmirovaniye*, 2001, no. 4, pp. 53–68 (In Russian).
8. Torgashev V. A., Tsarev I. V. A Family of Supercomputers with Dynamic Architecture — Conceptual Basis. *Iskusstvennyi intellekt*, 2009, no. 3, pp. 251–257 (In Russian).
9. Torgashev V. A., Tsarev I. V. Dynamic Automata Networks as a Model of Parallel Computations in the Multiprocessors with Dynamic Architecture. *Vestnik komp'yuternykh i informatsionnykh technologii* [Herald of Computer and Information Technologies], 2009, no. 3, pp. 11–20 (In Russian).
10. Torgashev V. A., Tsarev I. V. Technologies of Solving Complicated Problems on the Base of Dynamic Automata Networks. *Informatsionnoupavliayushchie sistemy* [Information and Control Systems], 2015, no. 6, pp. 57–65 (In Russian). doi:10.15217/issn1684-8853.2015.6.57