Выходит с 2006 г.

# СИСТЕМЫ ВЫСОКОЙ ДОСТУПНОСТИ

**№ 3, т. 14, 2018**

**Highly available systems**

**Журнал включен в перечень ВАК**



Журнал издается под научно-методическим руководством Федерального исследовательского центра «Информатика и управление» Российской академии наук.

**Редакторы выпуска – академик Академии криптографии РФ, д.т.н., проф. В. И. Будзко; академик Академии криптографии РФ, д.ф.-м.н. проф. А.П. Баранов; д.т.н., проф. А.М. Чеповский**

## Труды VI Международной научно-практической конференции «Управление информационной безопасностью в современном обществе»

## СОДЕРЖАНИЕ

## CONTENTS

Все статьи, представленные в данном выпуске журнала, соответствуют номенклатуре специальностей научных работников (Приказ Минобрнауки РФ от 11.08.2009 № 294) по отраслям технических наук.

**Journal «Sistemy' vy'sokoj dostupnosti» («Highly available systems»).**
**The journal covers scientific and engineering problems of ensuring confidentiality, availability, and integrity for the class of information-telecommunication systems of high availability (HA ITS), which contain such critical technologies of development**

**Необходимую информацию о журнале и полный список опубликованных статей, а также аннотации к ним Вы найдете на нашем сайте http://www.radiotec.ru**


**Учредитель: ООО «Издательство «Радиотехника».**

Лицензия № 065229. Свидетельства о регистрации ПИ № ФС 77-25037 от 12 июля 2006 г.
Сдано в набор 15.08.2018 г. Подписано в печать 17.09.2018 г.
Печ. л. 11,5. Тираж 400 экз. Изд. № 116.
Адрес Издательства «Радиотехника»: 107031, Москва, К-31, Кузнецкий мост, д. 20/6. Тел./факс 621-4837.
E-mail: info@radiotec.ru
http: // www. radiotec.ru /

Дизайн и допечатная подготовка ООО «САЙНС-ПРЕСС».
Отпечатано с предоставленных готовых файлов в полиграфическом центре ФГУП Издательство «Известия».
127254, ул. Добролюбова, д. 6. Контактный телефон (495) 650-38-80. izv-udprf.ru. Заказ №.

ISSN 2072-9472

# К читателям

**DOI: j20729472-201803-01**

Настоящий выпуск журнала «Системы высокой доступности» посвящен научному форуму – VI Международной научно-практической конференции «Управление информационной безопасностью в современном обществе», которая состоялась в Москве в июне 2018 г. в Высшей школе экономики. Публикуемые материалы научных докладов в полной мере соответствуют требованиям и духу нашего журнала.

***Главный редактор журнала***
***«Системы высокой доступности»,***
***академик Академии криптографии РФ*** ***В.И. Будзко***

Состоявшаяся 5−7 июня 2018 г. конференция «Управление информационной безопасностью в современном обществе» собрала более 300 специалистов в области обеспечения безопасности информационно-аналитических систем. На конференции были рассмотрены актуальные направления развития и представлены новейшие разработки ряда ведущих фирм России. Большой интерес вызвали обсуждения проблем управления процессами создания и эксплуатации отечественных операционных систем и компонент коммуникационного оборудования, создаваемого в рамках импортозамещения. Можно констатировать, что обмен мнениями и научный подход к проблемам импортозамещения позволил в определенной степени консолидировать усилия различных разработчиков, более четко сформулировать области применения разных изделий, а в ряде случаев устранить недопонимание функциональных особенностей разработок.

На пленарном заседании выступали ведущие ученые в области создания аналитических информационных систем (АИС), выделившие основные перспективные направления развития отрасли. Обширные доклады содержали полезную информацию как для ученых и специалистов-разработчиков, так и для преподавателей и студентов различных учебных заведений.

Выражаю благодарность всем участникам конференции за активное и заинтересованное обсуждение проблем. Успешное проведение конференции было бы невозможно без поддержки руководства Высшей школы экономики и Школы бизнес-информатики профессора С.В. Мальцевой.

***Член редколлегии журнала***
***«Системы высокой доступности»,***
***председатель Программного комитета конференции,***
***академик Академии криптографии РФ*** ***А.П. Баранов***

УДК 007

# Информационная безопасность и блокчейн



**В.И. Будзко** – академик Академии криптографии РФ, д.т.н., зам. директора по научной работе,
Институт проблем информатики ФИЦ ИУ РАН (Москва);
профессор, Национальный исследовательский ядерный университет «МИФИ» (Москва)
E-mail: vbudzko@ipiran.ru
**Д.А. Мельников** – к.т.н., доцент, вед. науч. сотрудник, Институт проблем информатики ФИЦ ИУ РАН (Москва);
доцент, Национальный исследовательский ядерный университет «МИФИ» (Москва)
E-mail: DAMelnikov@mephi.ru

Представлен краткий анализ современной технологии «блокчейн» (blockchain) с точки зрения обеспечения информационной безопасности и защиты данных. Дана общая характеристика и рассмотрены особенности технологии «блокчейн». Приведены примеры отрицательных свойств данной технологии, ее несовершенства и ненадежности, что обуславливает ухудшение криминогенной обстановки в современном Интернет-сообществе. Представлены выводы о перспективах применения этой технологии.

***Ключевые слова:*** *блокчейн, информационная безопасность, ассиметричные криптографические системы, биткойн, криптовалюта, дерево Меркля.*

The article presents a brief analysis of modern «blockchain» technology in terms of information security management and data protection. The general characteristics and features of the «blockchain» technology are presented. Examples of negative properties of this technology, its imperfections and unreliability are given, which causes the deterioration of the criminogenic situation in the modern Internet community. Conclusions about the prospects of this technology are presented.

***Keywords:*** *blockchain, information security, asymmetric-key cryptography, bitcoin, cryptocurrency, Merkle tree.*

**DOI: 10.18127/j20729472-201803-02**

За последние два года в среде специалистов в области информационных технологий все возрастающее внимание уделяется технологии блокчейн (далее БЧ-технология). Есть бесчисленные статьи и видеоролики, описывающие «волшебство» блокчейна. Существует высокий уровень «шумихи» вокруг использования блокчейнов, но при этом нет согласованного единого определения этой технологии и в целом она недостаточно осознана. Из различных источников звучат уверенные заявления о том, что внедрение этой технологии позволит увеличить скорость выполнения расчетов, сократит расходы на предоставление и хранение бумажных документов, уменьшит риски фальсификации и утери документов.

Об этой технологии говорят на самом высоком уровне, например, руководство Банка России. Так, Банк России готовит законодательную базу использования этой технологии в расчетах. Он опубликовал проект изменений в положение 383-П [1], которые дают возможность участникам торговой сделки использовать цифровые аккредитивы и смарт-контракты. Как отметила управляющий директор ассоциации «ФинТех» (АФТ) Татьяна Жаркова, цифровые аккредитивы на БЧ-платформе «Мастерчейн» появятся уже в этом году [2]. В программном комплексе (ПК) «Мастерчейн» [3] реализован способ взимания технических комиссий за транзакции и выполнение смарт-контрактов, аналогичный способу, реализованному на платформе Ethereum [4]. В 2019 г. она станет доступной для массового применения. Это один из примеров ажиотажа вокруг этой технологии. Многочисленные предложения клиентам исходят от различных коммерческих организаций, которые предлагают им применить эту технологию для создания новой или «апгрейда» действующей автоматизированной информационной системы (АИС). Как и для всех новых технологий, характерна такая тенденция, как желание применять ее в каждом секторе и во всех отношениях. Обзор этой технологии представлен в серии статей журнала «Системы высокой доступности» [5–8].

Родоначальник применения БЧ-технологии – Биткоин. Протокол платформы Ethereum – это не только цифровая валюта, он также используется при создании смарт-контрактов. У Биткоина нет таких функциональных возможностей – его технология позволяет только переводить денежные средства.

Что же это за фантастический новый способ построения АИС? Попробуем разобраться в этом явлении. При этом будем исходить из следующей парадигмы: будем следовать принципу, что наша основная

задача – создать АИС, обладающую требуемыми функциональными возможностями и характеристиками производительности и обеспечивающую требуемый уровень информационной безопасности.

Цель работы – провести краткий анализ БЧ-технологии с точки зрения обеспечения информационной безопасности и защиты данных.

## Основные общие характеристики БЧ-технологии

Точного определения БЧ-технологии нет, она не стандартизирована, как, например, IP-протокол. ISO, по утверждению специалистов, продвинулся только на 20%. Но, несмотря на многочисленные варианты реализации блокчейна, в большинстве вариантов используются некоторые общие базовые концепции. Они удачно изложены в виде предварительной основы стандарта на БЧ-технологию [4].

Основная особенность БЧ-технологии состоит в том, что на ее основе строятся системы с неизменяемыми цифровыми реестрами, которые реализуются распределенным образом (то есть без центрального хранилища) и обычно без централизованного управления. Некоторому сообществу пользователей обеспечивается возможность записывать транзакции в реестре, который является доступным для этого сообщества, так что никакая транзакция не может быть изменена после опубликования. Пользователи используют открытые и закрытые ключи для цифровой подписи и обеспечения безопасности транзакции внутри систем.

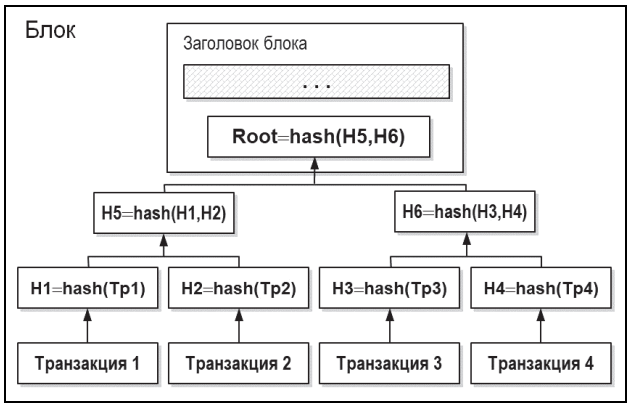


**Рис. 1.** Дерево Меркля (Merkle Tree)

БЧ-технология – основа современных АИС криптовалют, названных так вследствие широкого использования криптографических функций. Во всем мире функционируют несколько десятков систем криптовалют.

Каждая транзакция включает в себя один или несколько адресов и запись того, что произошло, а также цифровую подпись. Блокчейны состоят из блоков, каждый блок представляет собой группу транзакций. Все транзакции в блоке группируются вместе и объединяются с результатом криптографической хэш-функции предыдущего блока. Наконец, вычисляется новый результат хэш-функции для заголовка текущего блока с целью его размещения в самих данных блока, а также в следующем блоке (рис. 1).

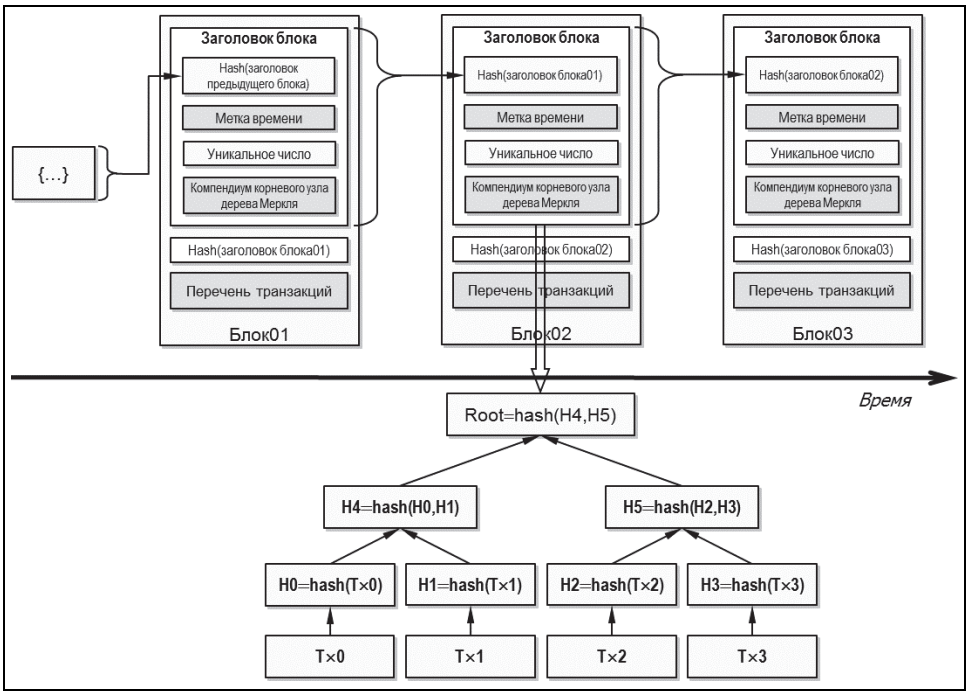


**Рис. 2.** Блокчейн-система на основе «дерева Меркля»

В дальнейшем каждый блок «привязывается» к предыдущему блоку в «цепи» (последовательности) блоков путем добавления результата хэш-функции предыдущего блока в заголовок текущего блока (рис. 2).

Основополагающий аппарат, используемый БЧ-технологиями, – ассиметричная криптографическая система (также называемая криптографией на основе откры-

тых/закрытых ключей). Ассиметричная криптография использует пару ключей: открытый ключ и закрытый ключ, которые математически «привязаны» друг к другу. Открытый ключ может быть раскрыт без снижения уровня защищенности процесса, но закрытый ключ должен храниться «в тайне» (оставаться секретным) в случае, если данные должны оставаться криптографически защищенными. Несмотря на то, что между двумя ключами существует взаимосвязь, закрытый ключ не может быть эффективно определен на основе знания открытого ключа.

Ассиметричная криптография использует различные ключи из пары ключей при реализации определенных функций, что зависит от предоставляемой услуги. Например, когда необходимо подписать данные с помощью цифровой подписи, криптоалгоритм использует закрытый ключ, предназначенный для подписи. Затем подпись может быть проверена с использованием соответствующего открытого ключа.

В итоге, ассиметричная криптография в БЧ-технологии применяется следующим образом:

при формировании цифровой подписи транзакций используются закрытые ключи;

для извлечения адресов используются открытые ключи, что позволяет реализовать принцип псевдоанонимности (*pseudonymity*, то есть использование псевдонимов) «один для многих» (*one-to-many*), одна пара открытых ключей может обеспечить формирование нескольких адресов, в некоторых случаях для создания нескольких адресов используется несколько пар открытых ключей;

для проверки подписей, сформированных на основе закрытых ключей, используются открытые ключи;

ассиметричная криптография предоставляет возможность проверить, что пользователь, выполняющий расчет (передающий стоимость) с другим пользователем, владеет закрытым ключом, на основе которого можно сформировать подпись стоимости.

Адрес пользователя представляет собой короткую буквенно-цифровую последовательность, извлекаемую из открытого ключа пользователя с использованием хэш-функции, которая дополняется некоторыми данными (используемыми для обнаружения ошибок). Адреса используются для отправки и получения цифровых активов. Большинство БЧ-технологий используют адреса в качестве обозначения конечных точек сквозного соединения в транзакции («from» и «to», «от» и «до»). Адреса короче открытых ключей и не являются секретными. Как правило, процедура формирования адреса означает получение открытого ключа, его преобразование с помощью хэш-функции и преобразование полученного компендиума в текстовый формат: *открытый ключ → хэш-функция → адрес*.

Пользователи по своему желанию могут сформировать достаточно много пар ключей и, следовательно, много адресов, что позволяет им варьировать уровни псевдоанонимности. Адреса выступают в качестве открытых параметров подлинности («identity») пользователей систем с БЧ-технологией. При этом зачастую выполняется преобразование адреса в QR-код для его более удобного использования.

## Особенности БЧ-технологии

При внедрении БЧ-технологии следует понимать ряд важных ее особенностей [5, 6]. Миллионы компьютеров выполняют следующие действия:

проверяют одни и те же транзакции по одним и тем же правилам, производят идентичную работу;

записывают в блокчейн (если повезло и дали возможность записать) одно и то же;

хранят (каждый) всю историю за все время, одинаковую, одну на всех.

Как результат, в настоящее время Биткойн потребляет электроэнергии больше, чем вся Ирландия. К 2020 г. будет потреблять столько же электроэнергии, сколько вся Дания. Цепочка Биткойн – 100 Гб данных. Чем больше совершается транзакций в сети Биткойн, тем быстрее растет объем. Большая часть появилась за последнюю пару лет. А в сети Ethereum всего за два года после запуска и полгода активного использования в блокчейне уже накопилось 200 Гб.

Если каждый узел сети делает одно и то же, то очевидно, что пропускная способность всей сети равна пропускной способности одного узла сети. Сегодня распределенные реестры не могут обрабатывать большое число транзакций (масштабируемость (scalability)). Биткойн может обработать максимум семь транзакций в секунду – на всех. Транзакции записываются лишь раз в 10 мин. А после появления записи для надежности принято подождать еще 50 мин, потому что записи регулярно самопроизвольно откатываются. В то же время Visa, которая не использует эту технологию, обрабатывает тысячи опера-

ций в секунду, а при необходимости легко увеличит мощности, ведь классические банковские технологии масштабируемы.

***Существует риск «атаки 51%».*** Суть атаки в том, что если кто-то контролирует больше половины всех майнинг-мощностей, то он может скрытно ото всех писать альтернативную финансовую историю, в которой он свои деньги никому не передавал, и иметь возможность тратить свои деньги несколько раз. Распределение вычислительных мощностей между основными странами-участниками: Китай – 81%, Исландия – 5%, у остальных – 3% и менее на каждого. Очевидно, комментарии не требуются.

В мае 2018 г. Биткойн торговал около 7500 долларов за одну монету. Через три года ожидается, что его цена возрастет до 30−40 000. Совокупный объем рыночной капитализации всех криптовалют на 06.01.2018 составил $801,7 млрд. Госдолг США сейчас – около 21 трлн долларов. ВВП США за 2017 г. – около 17 трлн долларов.

***Блокчейн открыт, все всё видят.*** Поэтому в блокчейне нет анонимности, у него есть «псевдонимность». Можно использовать в Биткойне миксер, который смешивает грязные деньги с большим количеством чистых, и тем самым «отмывает» их. Но кто его владелец? Если кому-то вы вернули долг, то после этого он знает:

сколько у вас всего денег в любой момент времени;

сколько и, главное, на что именно вы их тратили за все время;

что вы покупали, в какую рулетку играли, какого политика поддерживали «анонимно».

Если между вредоносными пользователями заключен преступный сговор, могут последовать следующие умышленные враждебные действия:

игнорирование транзакций конкретных пользователей, узлов и даже целых стран;

скрытое формирование модифицированной последовательности блоков, а затем представление ее в качестве более длинной альтернативной последовательности по сравнению с реальной (вместо реальной) – «добросовестные» узлы будут присоединяться к последовательности, на обработку которой потребовался наибольший объем «работы» (в соответствие с протоколом блокчейн), а это может нарушить концепцию «неизменяемости»;

отказ от передачи блоков на другие узлы, что существенно нарушает процесс распределение информации.

***Внесение изменений в программное обеспечение.*** Первая проблема состоит в том, что измененное программное обеспечение (ПО) должно быть размещено на всех узлах. На время запуска новой версии работа БЧ-системы будет приостановлена. Вторая проблема – наличие ошибки в новой версии охватит все пространство действия БЧ-системы. Так, в 2013 г. разработчики Биткойн выпустили новую версию самого популярного ПО для клиента, которая имела существенный недостаток (ошибку), что послужило началом развития двух конкурирующих блокчейнов. Ошибка распространилась повсеместно.

***Работа с закрытыми ключами.*** Пользователи обязаны обеспечивать себя закрытыми ключами сами. А это означает, что если один из них потерян, то все, что связано с этим закрытым ключом, также будет потеряно (цифровые активы и т.п.). В блокчейне нет функций «забыл мой пароль» или «восстановите мою учетную запись». Если закрытый ключ был украден, то злоумышленник получит полный доступ ко всем активам, находящимся под контролем этого закрытого ключа. Обеспечение безопасности закрытых ключей также является очень важной функцией, и поэтому многие пользователи используют специально защищенные программно-аппаратные комплексы для их хранения.

Хранилище закрытых ключей является чрезвычайно важным компонентом блокчейн-технологии. Когда в новостях прошло сообщение о том, что «*Биткойн был украден из…*», это почти наверняка означает, что были найдены закрытые ключи и они использовались для подписания транзакции, отправляющей деньги на новый счет, но это вовсе не означает, что была скомпрометирована система. Следует заметить: в связи с тем, что в блокчейне данные в большинстве случаев не могут быть изменены, то после того, как преступник украл закрытый ключ и публично переместил связанные с этим ключом финансовые средства на другой счет, такую незаконную транзакцию отменить уже нельзя.

Процедура проверки подписи транзакции связывает транзакции с владельцами закрытых ключей, но не реализует функцию привязки реальных параметров подлинности с их владельцами. В некоторых случаях возможна привязка реальных параметров подлинности с закрытыми ключами, но такая привязка может быть осуществлена с использованием внешних процедур и не обеспечивается блокчейном в явном виде.

В целом криптографические технологии, позволяющие реестрам сохранять в секрете структуру торговых операций, все еще разрабатываются.

***Проблемы разграничения прав доступа.*** При традиционном использовании БД существует возможность детального разграничения прав доступа. Это же может потребоваться прикладным системам, построенным на блокчейне, для включения в систему необходимого числа участников (полномочия предоставляются, как правило, хорошо известным участникам системы, таким как: администратор, пользователь, проверяющий, аудитор и т.д.). Это также относится и к администрированию полномочий/прав. В большой системе требуется гибкая и надежная технология изменения полномочий (сотрудник заболел, ушел в отпуск, уволился, и его полномочия необходимо своевременно передать другому).

Если пользователю предоставляется доступ в блокчейн с правом записи в систему, то можно ли аннулировать это разрешение? Большинство реальных блокчейнов неизменяемые, что может усложнить систему предоставления полномочий/прав.

***Внесение изменений в сохраненные данные.*** Что произойдет, когда в организации реализуется блокчейн, а затем возникает необходимость внести изменения в сохраненные данные? При традиционном использовании БД это может быть выполнено с помощью простого запроса (или могут быть сделаны значительные изменения путем обновления схемы БД или ПО). Однако на блокчейне гораздо сложнее изменить данные или обновить ПО «БД». Надо понимать чрезвычайную сложность в изменении чего-либо, что уже находится на блокчейне, и что изменения в программном и информационном обеспечении блокчейна могут вызвать распад БЧ-системы.

***Достоверность транзакции.*** Еще одним критическим аспектом технологии блокчейн является то, как участники блокчейна соглашаются с тем, что транзакция достоверна. Этот аспект называется «достижением консенсуса», и для него существует множество моделей, каждая из которых имеет свои положительные и отрицательные стороны с точки зрения конкретного бизнес-решения. Эта технология стала всемирно известной, начиная с 2008 г., когда она была использована с целью создания условий появления и применения цифровых валют в распределенных финансовых системах, в которых осуществляется доставка электронных денег.

## Внедрение БЧ-технологии

Многие считают, что внедрение блокчейна избавит от необходимости посредников в разных областях, а именно: 1) при оформлении различных видов документов; 2) в денежных переводах; 3) в страховании и др.

В Интернете можно найти множество заявлений на эту тему. Дождемся, когда появится из «песочницы» отлаженная версия «Мастерчейн» в виде законченного и полностью документированного программного продукта с юридической ответственностью производителя за его сопровождение.

Но проблемы, которые возникнут при внедрении описанной выше БЧ-технологии, можно показать на простом примере автоматизации с ее применением процесса реализации TAX FREE. При осуществлении возврата денег TAX FREE есть только два субъекта, для которых важна достоверность данных: покупатель и торговая организация. При этом конкретный магазин не интересует и не должен знать, что вы приобрели и потратили в другом магазине. Если для упрощения предположить, что возврат денег возможен только на определенный банковский счет покупателя (не кэшем), то схема может быть следующей:

1. Каждая покупка в конкретном магазине – отдельная транзакция, которая записывается в блок, привязанный к пользователю и магазину. Идентичные копии блока должны быть у магазина и покупателя.

2. Блок имеет идентификатор магазин/покупатель, и магазин «видит» только свои блоки, покупатель – свои. В блоке в качестве отдельной транзакции указываются идентифицирующие признаки счета, на который надо возвратить деньги. Получается, что есть цепочка блоков покупателя с характеристиками покупок в разных магазинах и цепочка блоков магазина из покупок покупателей в этом магазине.

3. Таможня проверяет наличие вывозимого товара по цепочке покупателя, «маркируя» товар, который ей предоставлен, что тут же находит отражение во всех реестрах для каждого магазина.

4. Измененные блоки должны «веером» разлететься по магазинам. Завершение таможенной проверки должно вызвать для магазинов сигнальное информирование о необходимости перевода денег на счет покупателя – получение ими транзакций о приобретенном вывозимом товаре.

5. Факт перевода денег также должен быть зафиксирован соответствующими транзакциями во всех реестрах (для покупателя и магазина).

И даже в такой простой задаче много важных моментов, а именно:

неизменность блоков по БЧ-технологии требует их дублирования с внесением изменений (версии блоков после покупки, после таможни и после перевода денег);

в эту систему должны быть вовлечены все магазины страны пребывания;

в эту систему должны быть вовлечены все таможенные пункты страны;

пользователи должны быть оснащены мобильными устройствами с соответствующим ПО, или в таможне должны быть установлены специальные терминалы;

должна быть обеспечена высокая доступность системы.

Наконец, в каждом магазине или при нем должен быть свой узел. А как быть с покупателями? Или один узел на всех? Но это не тот блокчейн, который рассматривался выше. Это централизация. Хотя для защиты отдельных блоков и цепочки в целом можно было бы воспользоваться деревом Меркля.

- На основе приведенных рассуждений можно сделать следующий вывод.

  1. БЧ-технология не стандартизована.

  2. БЧ-технология в разных ее вариациях имеет ряд свойств, позволяющих обеспечить высокую достоверность и сохранность данных. Однако эти свойства успешно работают при наличии надежной системы криптографической защиты.

  3. Существующие варианты БЧ-технологии не позволяют строить системы высокой доступности и высокой производительности.

  4. Еще рано говорить об изделии «Мастерчейн» как об основе построения высокопроизводительной и надежной системы расчетов.

  5. Централизованные системы имеют ряд принципиальных преимуществ перед распределенными, построенными на основе БЧ-технологии, с точки зрения обеспечения нормального жизненного цикла системы в условиях ее постоянного развития и обеспечения информационной безопасности. Но в них может быть оправдано применение отдельных элементов БЧ-технологии, что, правда, уже и так давно делается.

## Литература


1. Пояснительная записка к проекту указания Банка России «О внесении изменений в Положение Банка России от 19 июня 2012 г. № 383-П «О правилах осуществления перевода денежных средств». URL = http://www.cbr.ru/Publ/Vestnik/ves120628034.pdf.
2. Ассоциация корпоративных казначеев приступает к тестированию блокчейна в СПФС. URL = https://bits.media/news/assotsiatsiya-korporativnykh-kaznacheev-pristupaet-k-testirovaniyu-blokcheyna-v-spfs/.
3. Децентрализованная сеть обмена и хранения информации «Мастерчейн» Версия 1.1. Whitepaper. Ассоциация развития финансовых технологий. Москва. 2017. URL = http://Fintechru.Org/Documents/Masterchain_Whitepaper_11_08.Pdf.
4. *Chris Dannen* Introducing Ethereum and Solidity: Foundations of Cryptocurrency and Blockchain Programming for Beginners. Brooklyn, New York, USA. 2017.
5. *Акимова Г.П.*, *Даниленко А.Ю.*, *Пашкина Е.В.*, *Подрабинович А.А.* Применение технологии блокчейн в информационных системах. Часть 1. Защищенный электронный документооборот // Системы высокой доступности. 2018. Т. 14. № 1. С. 3−7.
6. *Даниленко А.Ю.*, *Пашкина Е.В.*, *Пашкин М.А.*, *Соловьев А.В.* Применение технологии блокчейн в информационных системах. Часть 2. Подтверждение авторства и обеспечение целостности // Системы высокой доступности. 2018. Т. 14. № 1. С. 9−11.
7. *Акимова Г.П.*, *Даниленко А.Ю.*, *Пашкина Е.В.*, *Пашкин М.А.*, *Соловьев А.В.* Применение технологии блокчейн в информационных системах. Часть 3. Цифровая экономика и сохранность электронных документов // Системы высокой доступности. 2018. Т. 14. № 1. С. 13−19.
8. *Акимова Г.П.*, *Пашкина Е.В.*, *Пашкин М.А.*, *Соловьев А.В.*, *Тарханов И.А.* Применение технологии блокчейн в информационных системах. Часть 4. Концептуальное решение задачи обеспечения сохранности электронных документов в условиях цифровой экономики // Системы высокой доступности. 2018. Т. 14. № 1. С. 20−26.
9. *Dylan Yaga, Peter Mell, Nik Roby, Karen Scarfone* Blockchain Technology Overview. Draft NISTIR 8202. National Institute of Standards and Technology Internal Report 8202. National Institute of Standards and Technology Attn: Computer Security Division. Information Technology Laboratory. 100 Bureau Drive (Mail Stop 8930) Gaithersburg. MD 20899-8930. January 2018.
10. *Алексей Маланов* Шесть мифов о блокчейне и Биткойне, или почему это не такая уж эффективная технология. URL = https://habr.com/company/kaspersky/blog/336036/.
11. *Артем Генкин* Блокчейн-скептики: реальны ли угрозы и риски? URL = https://www.if24.ru/blokchejn-skeptiki-realny-li-ugrozy-i-riski/.


**Поступила 3 августа 2018 г.**

# Information security and blockchain



**V.I. Budzko** – Member of Russian Cryptography Academy, Dr.Sc.(Eng.), Deputy Director on Research and Development, Institute of Informatics Problems of FRC CSC RAS (Moscow);
Professor, National Research Nuclear University «MEPhI» (Moscow)
E-mail: vbudzko@ipiran.ru
**D.A. Melnikov** – Ph.D.(Eng.), Associate Professor, Leading Research Scientist, Institute of Informatics Problems of FRC CSC RAS (Moscow); Associate Professor, National Research Nuclear University «MEPhI» (Moscow)
E-mail: DAMelnikov@mephi.ru

The article presents a brief analysis of modern «blockchain» technology in terms of information security management and data protection. The general characteristics and features of the «blockchain» technology are presented. Examples of negative properties of this technology, its imperfections and unreliability are given, which causes the deterioration of the criminogenic situation in the modern Internet community. Conclusions about the prospects of this technology are presented. Centralized systems have a number of principal advantages over distributed, built on the basis of «blockchain» technology, from the point of view of ensuring the normal life cycle of the system in the conditions of its constant development and ensuring information security. But they can justify the use of separate elements of «blockchain» technology, which, however, has been done for a long time already.

**References**

1. Explanatory Note to the Draft Instruction of the Bank of Russia «On Amendments to Bank of Russia Regulation № 383-P of June 19, 2012 «On the Rules for Implementing the Transfer of Funds». URL = http://www.cbr.ru/ Publ/Vestnik/ves120628034.pdf.
2. Association of Corporate Treasurers Starts Testing Blockchain in Financial Message Transfer System of Bank of Russia. URL = https://bits.media/news/assotsiatsiya-korporativnykh-kaznacheev-pristupaet-k-testirovaniyu-blokcheyna-v-spfs/.
3. Decentralized network of information exchange and storage «Masterchain» Version 1.1. Whitepaper. FinNech Associftion. Moscow. 2017. URL = http://Fintechru.Org/Documents/Masterchain_Whitepaper_11_08.Pdf.
4. *Chris Dannen* Introducing Ethereum and Solidity: Foundations of Cryptocurrency and Blockchain Programming for Beginners. Brooklyn, New York, USA. 2017.
5. *Akimova G.P., Danilenko A.Yu., Pashkina Ye.V., Podrabinovich A.A.* Primeneniye tekhnologii blokcheyn v informatsionnykh sistemakh. Chast' 1. Zashchishchennyy elektronnyy dokumentooborot // Sistemy vysokoy dostupnosti. 2018. T. 14. № 1. S. 3–7.
6. *Danilenko A.Yu., Pashkina Ye.V., Pashkin M.A., Solov'yev A.V.* Primeneniye tekhnologii blokcheyn v informatsionnykh sistemakh. Chast' 2. Podtverzhdeniye avtorstva i obespecheniye tselostnosti // Sistemy vysokoy dostupnosti. 2018. T. 14. № 1. S. 9–11.
7. *Akimova G.P., Danilenko A.Yu., Pashkina Ye.V., Pashkin M.A., Solov'yev A.V.* Primeneniye tekhnologii blokcheyn v informatsionnykh sistemakh. Chast' 3. Tsifrovaya ekonomika i sokhrannost' elektronnykh dokumentov // Sistemy vysokoy dostupnosti. 2018. T. 14. № 1. S. 13–19.
8. *Akimova G.P., Pashkina Ye.V., Pashkin M.A., Solov'yev A.V., Tarkhanov I.A.* Primeneniye tekhnologii blokcheyn v informatsionnykh sistemakh. Chast' 4. Kontseptual'noye resheniye zadachi obespecheniya sokhrannosti elektronnykh dokumentov v usloviyakh tsifrovoy ekonomiki // Sistemy vysokoy dostupnosti. 2018. T. 14. № 1. S. 20–26.
9. *Dylan Yaga, Peter Mell, Nik Roby, Karen Scarfone* Blockchain Technology Overview. Draft NISTIR 8202. National Institute of Standards and Technology Internal Report 8202, National Institute of Standards and Technology Attn: Computer Security Division. Information Technology Laboratory. 100 Bureau Drive (Mail Stop 8930) Gaithersburg. MD 20899-8930. January 2018.
10. *Alexey Malanov* Six myths about the blockade and Bitcoin, or why it is not such an effective technology. URL = https://habr.com/company/kaspersky/blog/336036/.
11. *Artiom Genkin* Blockchain skeptics: how real is the threat and risks? URL = https://www.if24.ru/blokchejn-skeptiki-realny-li-ugrozy-i-riski/.

УДК 004.056

# Актуальные вопросы выявления уязвимостей и недекларированных возможностей в программном обеспечении



**А.В. Барабанов** – к.т.н., зам. ген. директора по НИР, АО «НПО «Эшелон» (Москва)
E-mail: a.barabanov@npo-echelon.ru
**А.С. Марков** – д.т.н., президент АО «НПО «Эшелон» (Москва)
E-mail: a.markov@npo-echelon.ru
**В.Л. Цирлов** – к.т.н., ген. директор АО «НПО «Эшелон» (Москва)
E-mail: v.tsirlov@npo-echelon.ru

Проведен обзор текущего состояния тематики программной безопасности. Сделан вывод о соответствии понятийной базы безопасности программ современным парадигмам информационных технологий. Рассмотрены достоинства и ограничения современных подходов к тестированию программ по требованиям безопасности. Отмечено, что для обеспечения полноты проверок необходимо сочетание различных подходов к тестированию программ. Показан современный методический подход к проведению испытаний по выявлению уязвимостей и недекларированных возможностей (НДВ). Дан краткий обзор зарубежных исследований. Описаны компенсационные меры программной безопасности. Приведена статистика по выявлению уязвимостей в процессе сертификации. Указаны пути повышения результативности тестирования и испытаний программ.

***Ключевые слова:*** *программная безопасность, тестирование, испытания, программные средства защиты, дефекты, уязвимости, угрозы, безопасное программное обеспечение, недекларированные возможности, испытательная лаборатория.*

The paper reviews the current state of the software security topics. The conclusion that the conceptual basis of software security corresponds to modern paradigms of information technologies is made. The advantages and limitations of modern approaches to testing programs for security requirements are shown. It is shown that to ensure the completeness of the checks, a combination of different approaches to software testing is necessary. A modern methodical approach to conducting tests to identify vulnerabilities and undeclared capabilities is given. A brief overview of foreign research is given. Compensatory measures of software security are considered. The statistics on the identification of vulnerabilities in the certification process are presented. The ways of increasing the effectiveness of software security testing are indicated.

***Keywords:*** *software security, testing, conformity assessment, software security tools, weakness, vulnerabilities, threats, secure software, undeclared capabilities, testing laboratory.*

**DOI: 10.18127/j20729472-201803-03**

В области информационной безопасности (ИБ) информационных инфраструктур можно выделить два фактора: 1) рост техногенного риска, обусловленного чрезвычайной сложностью программного обеспечения (ПО) высокотехнологичных систем; 2) расширение спектра кибератак, использующих уязвимости программных систем, что обусловлено развитием информационных компьютерных технологий [1].

Несмотря на усилия сообществ по безопасности программ, число ошибок и дефектов безопасности, уязвимостей и атак не снижается. В нашей стране, как отмечено в Стратегии национальной безопасности РФ (п.68), проблемная ситуация усугубляется наличием угроз, связанных с импортными технологиями [2].

Ц е л ь  р а б о т ы – представить общий обзор состояния тематики безопасности ПО, главным образом, с точки зрения опыта испытательной лаборатории (ИЛ).

## Понятийный аппарат программной безопасности

Можно утверждать, что понятийный аппарат программной безопасности в настоящее время сформирован и находится в итерационном развитии в соответствии с современными парадигмами [3]. Под безопасным ПО понимают ПО, разработанное и применяемое с использованием совокупности мер, направленных на предотвращение появления и устранение уязвимостей. К базовым факторам программной безопасности относят понятие *дефекта* (weakness), *уязвимости* (vulnerability), *угрозы ИБ* и *риска ИБ*. На данный момент классификации дефектов и базы данных уязвимостей, их стандарты критичности, а так-

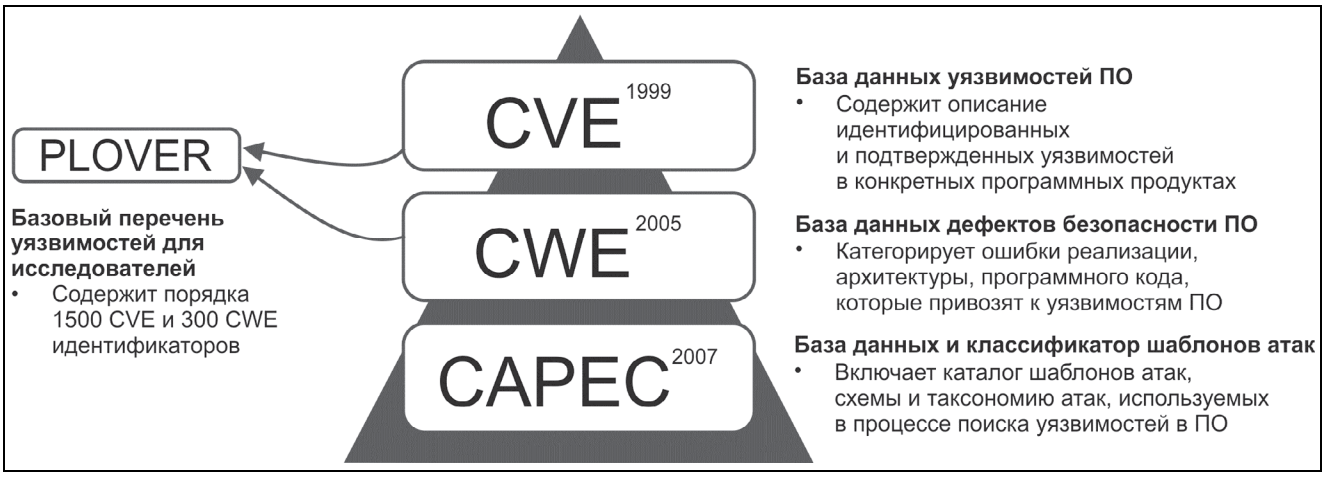


**Рис. 1.** Фрагмент онтологии MITRE с примерами для исследователей

же рейтинги хорошо известны [4]. Для наглядности на рис. 1 приведен фрагмент онтологии MITRE для исследований.

## Тестирование и испытания программ

Методический аппарат тестирования программ берет свое начало со второй половины прошлого столетия [5] и в самом общем виде классифицируется по:

наличию исходного кода (методы черного ящика и белого ящика);

прогону ПО (динамический анализ и статический анализ);

жизненному циклу (верификация по спецификациям, статический анализ и отладка, функциональное тестирование).

В табл. 1 представлены техники тестирования, направленные на выявление дефектов, уязвимостей и соответствующих угроз ИБ.

**Таблица 1.** ***Примеры техник тестирования программ***

| Техники тестирования | Типовые дефекты и уязвимости | Основные ограничения |
|---|---|---|
| *При наличии доступа к исходному коду, спецификациям и компоновочной среде* | | |
| Прикладная верификация | Некорректности программирования | Математическая ресурсоемкость |
| Инспекция кода | Дефекты, уязвимости статического характера | Требования к квалификации эксперта |
| Статический анализ (контроль НДВ) | Мертвый код | Сложность контроля динамических конструкций и покрытия кода |
| Статический анализ (сигнатурный, эвристический) | Потенциально опасные фрагменты кода | Сложность выявления динамических конструкций |
| Динамический анализ (анализ трассы) | Проявления дефектов и уязвимостей | Влияние контрольных точек трассировки, сложность покрытия кода |
| *При отсутствии доступа к исходному коду* | | |
| Функциональное тестирование | Функциональные ошибки | Сложность выявления преднамеренных уязвимостей |
| Фаззинг | Уязвимости на отказ в обслуживании | Сложность преднамеренных уязвимостей, связанных с редко используемыми входными данными |
| Тестирование на проникновения | Известные уязвимости | Ограниченность одной или несколькими уязвимостями |

Легко видеть, что не существует универсального метода тестирования, при этом все методы подразумевают привлечение квалифицированных экспертов по ИБ. Ситуация усложняется тем, что современные уязвимости имеют многофакторный вид.

Что касается инструментария тестирования, то в нашей стране можно отметить научные институты ИСП РАН и СПбПУ, ведущие изыскания по прикладной верификации, а также организации-

разработчики: СиПроВер (PVS-Studio), Positive Technologies (PT Application Inspector), InfoWatch (APPURCUT), Solar (inCode), НПО «Эшелон» (AppChecker, AK-VS) и др. [6].

### Испытания программных средств

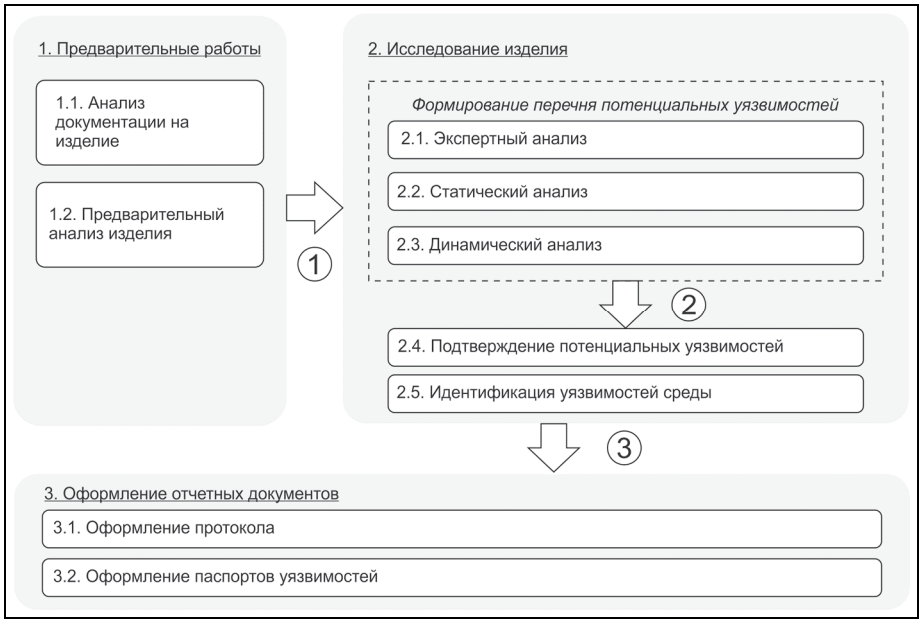


**Рис. 2.** Схема методического подхода к выявлению уязвимостей

Требования к проведению испытаний программных средств защиты информации (СЗИ) задаются нормативно-правовыми актами (НПА) федеральных систем сертификации СЗИ. В течение последних 20 лет испытания СЗИ включали в себя функциональное тестирование и контроль отсутствия недекларированных возможностей (НДВ). В настоящее время по линии ФСТЭК России проводятся исследования, направленные на повышение *результативности* оценки соответствия, как то: создание и поддержка базы данных уязвимостей; проведение анализа уязвимостей в рамках сертификации (согласно ISO/IEC TR 20004 [7]); разработка НПА нового поколения (AVA_VAN); создание национального стандарта по уязвимостям систем; разработка рекомендаций по обновлению СЗИ; разработка требований к разработке безопасного ПО. Современный методический подход к выявлению уязвимостей в рамках сертификации представлен на рис. 2.

Как видно из рис. 2, методика выявления уязвимостей включает в себя полный набор прикладных практик в данной области, а именно: анализ документации и доступных источников о ПО (как объекте оценки); предварительный анализ ПО и анализ структуры [8, 9] (в том числе выявление заимствованных компонент [10]); экспертный анализ [11]; статический анализ [12−16], который включает в себя как контроль НДВ [9], так и эвристики [16]; динамический анализ [15, 17, 18]; ручной анализ и подтверждение выявленных потенциальных уязвимостей [19].

### Зарубежные изыскания в области тестирования и испытаний программ

Проблематика безопасности ПО является весьма востребованной, об этом свидетельствуют как современные тенденции в области сертификации [20], так и изыскания DARPA [21] (табл. 2).

**Таблица 2.** ***Исследования DAPRA по тематике уязвимостей программ***

| Научные изыскания | Целевое назначение |
|---|---|
| Vetting Commodity IT Software and Firmware (VET) | Контроль НДВ и закладок в ПО устройств (телефоны, факсы, принтеры и пр.) |
| Automated Program Analysis for Cybersecurity (APAC) | Создание средств автоматизации выявления НДВ и закладок в ПО для мобильных платформ (Android) |
| Cyber Grand Challenge (CGC) | Академические соревнования по выявлению уязвимостей |
| Clean-slate design of Resilient, Adaptive, Secure Hosts | Создание методов и инструментальных средств (среда разработки, среда верификации, среда выполнения) разработки компьютерных систем, устойчивых к атакам |
| Mining and Understanding Software Enclaves | Повышение надежности ПО, часть работ посвящена созданию средств верификации, в том числе статических анализаторов |

### Компенсационный подход

Опыт работы ИЛ показал, что уровень зрелости компании-разработчика напрямую связан с числом ошибок в создаваемом ПО. Эффективным средством повышения безопасности разрабатываемого ПО является внедрение в организации системы менеджмента безопасной разработки в соответствии с ГОСТ Р 56939. Данный стандарт регламентирует связку между процессами жизненного цикла ПО (по ISO/IEC 12207), угрозами и мерами по безопасной разработке [22].


**Рис. 3.** Диаграмма наличия преднамеренных и непреднамеренных дефектов

### Некоторая статистика

В рамках работы аккредитованной ИЛ накоплена статистика, подтверждающая результативность современных эвристических подходов по выявлению уязвимостей как при наличии исходных кодов, так и без них [23]. На рис. 3 показано, что в проверяемом ПО высокую долю уязвимостей можно отнести к преднамеренным.

- Проблема наличия уязвимостей в ПО далека от окончательного решения, главным образом, из-за проклятья размерности. При этом по причине человеческого фактора трудно сформулировать показатели оценки гарантированного уровня безопасности программ.

  Несмотря на то, что имеется понятийная, методическая и инструментальная база программной безопасности, результативность и эффективность процесса выявления уязвимостей существенно зависит от квалификации экспертов ИЛ, что обусловливает высокие избирательные требования к аккредитации ИЛ.

  В инженерном плане можно сказать, что прорывным для нашей страны стало создание национальной базы угроз и уязвимостей под эгидой ФСТЭК России, совершенствование методик выявления уязвимостей и разработка серии стандартов, гармонизированных с ГОСТ 569636.

  Одним из направлений развития современных технологий выявления уязвимостей можно предположить появление киберполигонов и центров компетенции, интегрирующих ИЛ с целью предоставления соответствующих, например, облачных услуг.

  Следует ожидать новый научный этап развития тематики с развитием прикладных супервычислителей, появлением квантового компьютера и широкого внедрения методов искусственного интеллекта.

## Литература


1. *Баранов А.П.* Актуальные проблемы в сфере обеспечения информационной безопасности программного обеспечения // Вопросы кибербезопасности. 2015. № 1. С. 2−5.
2. *Муравник В.Б.*, *Захаренков А.И.*, *Добродеев А.Ю.* Некоторые предложения по подходу и порядку реализации политики и стратегии импортозамещения в интересах национальной безопасности и укрепления обороноспособности Российской Федерации // Вопросы кибербезопасности. 2016. № 1(14). С. 2−8.
3. *Марков А.С.*, *Шеремет И.А.* Теоретические аспекты сертификации средств защиты информации // Оборонный комплекс научно-техническому прогрессу России. 2015. № 4(128). С. 7−15.
4. *Марков А.С.*, *Фадин А.А.* Систематика уязвимостей и дефектов безопасности программных ресурсов // Защита информации. Инсайд. 2013. № 3(51). С. 56−61.
5. *Липаев В.В.* Надежность и функциональная безопасность комплексов программ реального времени // Программная инженерия. 2013. № 8. С. 10−18.
6. *Захаров В.Н.* По итогам 4-й Международной научно-практической конференции «Инструменты и методы анализа программ» TMPA-2017 // Системы высокой доступности. 2017. Т. 13. № 2. С. 73−74.
7. *Барабанов А.В.*, *Евсеев А.Н.* Вопросы повышения эффективности анализа уязвимостей при проведении сертификационных испытаний программного обеспечения по требованиям безопасности информации // Труды Междунар. симпозиума «Надежность и качество». 2015. Т. 1. С. 330−333.

8. *Барабанов А.В.*, *Гришин М.И.*, *Кубарев А.В.* Моделирование угроз безопасности информации, связанных с функционированием скрытых в вредоносных компьютерных программах // Вопросы кибербезопасности. 2014. № 4(7). С. 41–48.
9. *Марков А.С.*, *Цирлов В.Л.*, *Барабанов А.В.* Методы оценки несоответствия средств защиты информации. М.: Радио и связь. 2012. 192 с.
10. *Кононов Д.С.* Итерационный метод поиска клонированного кода, основанный на вычислении редакционного расстояния // Вопросы кибербезопасности. 2017. № 1. С. 16–21.
11. *Жидков И.В.*, *Кадушкин И.В.* О признаках потенциально опасных событий в информационных системах // Вопросы кибербезопасности. 2014. № 1(2). С. 40–48.
12. *Аветисян А.*, *Белеванцев А.*, *Бородин А.*, *Несов В.* Использование статического анализа для поиска уязвимостей и критических ошибок в исходном коде программ // Труды Института системного программирования РАН. 2011. Т. 21. С. 23–38.
13. *Александров Я.А.*, *Сафин Л.К.*, *Чернов А.В.*, *Трошина К.Н.* Определение границ подпрограмм при статическом анализе бинарных образов // Вопросы кибербезопасности. 2016. № 1(14). С. 53–60.
14. *Колосов А.П.*, *Рыжков Е.А.* Применение статического анализа при разработке программ // Известия Тульского государственного университета. Технические науки. 2008. № 3. С. 185–190.
15. *Рибер Г.*, *Малмквист К.*, *Щербаков А.* Многоуровневый подход к оценке безопасности программных средств // Вопросы кибербезопасности. 2014. № 1(2). С. 36–39.
16. *Марков А.С.*, *Матвеев В.А.*, *Фадин А.А.*, *Цирлов В.Л.* Эвристический анализ безопасности программного кода // Вестник МГТУ им. Н.Э. Баумана. Сер.: Приборостроение. 2016. № 1(106). С. 98–111.
17. *Поляков С.А.*, *Карасев С.В.* Особенности получения информации о ходе выполнения программы (трассы) с использованием аппаратного окружения // Вопросы кибербезопасности. 2016. № 3(16). С. 40–44.
18. *Мельников П.В.*, *Горюнов М.Н.*, *Анисимов Д.В.* Подход к проведению динамического анализа исходных текстов программ // Вопросы кибербезопасности. 2016. № 3(16). С. 33–39.
19. *Sviridov P.Y.*, *Zaytsev G.Y.*, *Ivachev A.S.* The universal vulnerability exploitation platform for CTF // Прикладная дискретная математика. Приложение. 2014. № 7. С. 106–108.
20. *Барабанов А.В.*, *Марков А.С.*, *Цирлов В.Л.* Международная сертификация в области информационной безопасности // Стандарты и качество. 2016. № 7. С. 30–33.
21. *Петренко А.А.*, *Петренко С.А.* НИОКР агентства DARPA в области кибербезопасности // Вопросы кибербезопасности. 2015. № 4(12). С. 2–22.
22. *Барабанов А.В.*, *Марков А.С.*, *Цирлов В.Л.* 28 магических мер разработки безопасного программного обеспечения // Вопросы кибербезопасности. 2015. № 5. С. 2–10.
23. *Барабанов А.В.*, *Марков А.С.*, *Фадин А.А.*, *Цирлов В.Л.* Статистика выявления уязвимостей программного обеспечения при проведении сертификационных испытаний // Вопросы кибербезопасности. 2017. № 2(20). С. 2–8.

**Поступила 3 августа 2018 г.**

# Topical issues of identifying vulnerabilities and undeclared capabilities in software



**A.V. Barabanov** – Ph.D.(Eng.), Deputy General Director of NPO «Echelon» (Moscow)
E-mail: a.barabanov@npo-echelon.ru
**A.S. Markov** – Dr.Sc.(Eng.), President of NPO «Echelon» (Moscow)
E-mail: a.markov@npo-echelon.ru
**V.L. Tsirlov** – Ph.D.(Eng.), General Director of NPO «Echelon» (Moscow)
E-mail: v.tsirlov@npo-echelon.ru

The paper reviews the current state of the software security topics. The conclusion that the conceptual basis of software security corresponds to modern paradigms of information technologies is made. The advantages and limitations of modern approaches to testing programs for security requirements are shown. It is shown that to ensure the completeness of the checks, a combination of different approaches to software testing is necessary. A modern methodical approach to conducting tests to identify vulnerabilities and undeclared capabilities is given. A brief overview of foreign research is given. Compensatory measures of software security are considered. The statistics on the identification of vulnerabilities in the certification process are presented. The ways of increasing the effectiveness of software security testing are indicated. We should expect a new scientific stage of the development of the subject with the development of applied supercomputers, the emergence of a quantum computer and the widespread introduction of methods of artificial intelligence.

**References**

1. *Baranov A.P.* Aktual'nye problemy v sfere obespecheniya informacionnoj bezopasnosti programmnogo obespecheniya // Voprosy kiberbezopasnosti. 2015. № 1. P. 2–5.

2. *Muravnik V.B., Zaharenkov A.I., Dobrodeev A.Yu.* Nekotorye predlozheniya po podhodu i poryadku realizacii politiki i strategii importozameshcheniya v interesah nacional'noj bezopasnosti i ukrepleniya oboronosposobnosti Rossijskoj Federacii // Voprosy kiberbezopasnosti. 2016. № 1(14). P. 2–8.
3. *Markov A.S., Sheremet I.A.* Teoreticheskie aspekty sertifikacii sredstv zashchity informacii // Oboronnyj kompleks nauchno-tekhnicheskomu progressu Rossii. 2015. № 4(128). P. 7–15.
4. *Markov A.S., Fadin A.A.* Sistematika uyazvimostej i defektov bezopasnosti programmnyh resursov // Zashchita informacii. Insajd. 2013. № 3(51). P. 56–61.
5. *Lipaev V.V.* Nadezhnost' i funkcional'naya bezopasnost' kompleksov programm real'nogo vremeni // Programmnaya inzheneriya. 2013. № 8. P. 10–18.
6. *Zaharov V.N.* Po itogam 4-j Mezhdunarodnoj nauchno-prakticheskoj konferencii «Instrumenty i metody analiza programm» TMPA-2017 // Sistemy vysokoj dostupnosti. 2017. T. 13. № 2. P. 73–74.
7. *Barabanov A.V., Evseev A.N.* Voprosy povysheniya ehffektivnosti analiza uyazvimostej pri provedenii sertifikacionnyh ispytanij programmnogo obespecheniya po trebovaniyam bezopasnosti informacii // Trudy Mezhdunar. simpoziuma «Nadezhnost' i kachestvo». 2015. T. 1. P. 330–333.
8. *Barabanov A.V., Grishin M.I., Kubarev A.V.* Modelirovanie ugroz bezopasnosti informacii, svyazannyh s funkcionirovaniem skrytyh v vredonosnyh komp'yuternyh programmah // Voprosy kiberbezopasnosti. 2014. № 4(7). P. 41–48.
9. *Markov A.S., Cirlov V.L., Barabanov A.V.* Metody ocenki nesootvetstviya sredstv zashchity informacii. M.: Radio i svyaz'. 2012. 192 s.
10. *Kononov D.S.* Iteracionnyj metod poiska klonirovannogo koda, osnovannyj na vychislenii redakcionnogo rasstoyaniya // Voprosy kiberbezopasnosti. 2017. № 1. P. 16–21.
11. *Zhidkov I.V., Kadushkin I.V.* O priznakah potencial'no opasnyh sobytij v informacionnyh sistemah // Voprosy kiberbezopasnosti. 2014. № 1(2). P. 40–48.
12. *Avetisyan A., Belevancev A., Borodin A., Nesov V.* Ispol'zovanie staticheskogo analiza dlya poiska uyazvimostej i kriticheskih oshibok v iskhodnom kode program // Trudy Instituta sistemnogo programmirovaniya RAN. 2011. T. 21. P. 23–38.
13. *Aleksandrov Ya.A., Safin L.K., CHernov A.V., Troshina K.N.* Opredelenie granic podprogramm pri staticheskom analize binarnyh obrazov // Voprosy kiberbezopasnosti. 2016. № 1(14). P. 53–60.
14. *Kolosov A.P., Ryzhkov E.A.* Primenenie staticheskogo analiza pri razrabotke program // Izvestiya Tul'skogo gosudarstvennogo universiteta. Tekhnicheskie nauki. 2008. № 3. P. 185–190.
15. *Riber G., Malmkvist K., Shcherbakov A.* Mnogourovnevyj podhod k ocenke bezopasnosti programmnyh sredstv // Voprosy kiberbezopasnosti. 2014. № 1(2). P. 36–39.
16. *Markov A.S., Matveev V.A., Fadin A.A., Tsirlov V.L.* EHvristicheskij analiz bezopasnosti programmnogo koda // Vestnik MGTU im. N.E. Baumana. Ser.: Priborostroenie. 2016. № 1(106). P. 98–111.
17. *Polyakov S.A., Karasev S.V.* Osobennosti polucheniya informacii o hode vypolneniya programmy (trassy) s ispol'zovaniem apparatnogo okruzheniya // Voprosy kiberbezopasnosti. 2016. № 3(16). P. 40–44.
18. *Mel'nikov P.V., Goryunov M.N., Anisimov D.V.* Podhod k provedeniyu dinamicheskogo analiza iskhodnyh tekstov program // Voprosy kiberbezopasnosti. 2016. № 3(16). P. 33–39.
19. *Sviridov P.Y., Zaytsev G.Y., Ivachev A.S.* The universal vulnerability exploitation platform for CTF // Prikladnaya diskretnaya matematika. Prilozhenie. 2014. № 7. P. 106–108.
20. *Barabanov A.V., Markov A.S., Cirlov V.L.* Mezhdunarodnaya sertifikaciya v oblasti informacionnoj bezopasnosti // Standarty i kachestvo. 2016. № 7. P. 30–33.
21. *Petrenko A.A., Petrenko S.A.* NIOKR agentstva DARPA v oblasti kiberbezopasnosti // Voprosy kiberbezopasnosti. 2015. № 4(12). P. 2–22.
22. *Barabanov A.V., Markov A.S., Cirlov V.L.* 28 magicheskih mer razrabotki bezopasnogo programmnogo obespecheniya // Voprosy kiberbezopasnosti. 2015. № 5. P. 2–10.
23. *Barabanov A.V., Markov A.S., Fadin A.A., Cirlov V.L.* Statistika vyyavleniya uyazvimostej programmnogo obespecheniya pri provedenii sertifikacionnyh ispytanij // Voprosy kiberbezopasnosti. 2017. № 2(20). P. 2–8.

УДК 004

# Настраиваемые специализированные СБИС – реальная основа создания будущих экзамасштабных суперкомпьютеров (зарубежный и отечественный опыт)



**Л.К. Эйсымонт** – к.ф.-м.н., Федеральный реестр экспертов научно-технической сферы Минобрнауки России (Москва)
E-mail: verger-lk@yandex.ru

Рассмотрено направление создания проблемно-ориентированных специализированных СБИС (ПОС-СБИС) в качестве варианта асимметричного замещения зарубежной элементно-компонентной базы (ЭКБ) в отечественных суперкомпьютерах и облачных серверах высшего диапазона производительности. Даны систематизация и анализ достижений в данной области. Приведено сопоставление с отечественными разработками.

***Ключевые слова:*** *суперкомпьютеры высшего диапазона производительности, элементная база суперкомпьютеров, блокчейн, глубокое обучение, настраиваемая проблемно-ориентированная специализированная СБИС, программно управляемое устройство, реконфигурируемое устройство, мультитредовость, энергоэффективность.*

The direction of creation of problem-oriented specialized VLSI (POS-VLSI) as a variant of asymmetric replacement of the foreign element-component base (ECB) in domestic supercomputers and cloud servers of the highest performance range is considered. The article systematizes and analyzes achievements in this field, compares it with domestic developments.

***Keywords:*** *high-end (HEC) supercomputers, supercomputer electronics components, blockchain, deep learning, configurable problem-oriented special VLSI, programmable unit, reconfigurable unit, energy efficiency.*

**DOI: 10.18127/j20729472-201803-04**

Данная работа является публикацией независимого экспертного мнения, предоставленного ФГУП «РИНКЦЭ» для формирования рекомендаций экспертного сообщества по оптимизации реализации утвержденной Указом Президента РФ от 1 декабря 2016 г. №642 «Стратегии научно-технологического развития Российской Федерации».

Цель работы – рассмотреть направление создания проблемно-ориентированных специализированных СБИС (ПОС-СБИС) в качестве варианта асимметричного замещения зарубежной элементно-компонентной базы (ЭКБ) в отечественных суперкомпьютерах и облачных серверах высшего диапазона производительности.

В работе [1] на примере американских, китайских и японских суперкомпьютеров высшего диапазона производительности был продемонстрирован современный уровень их ЭКБ в части применяемых вычислительных (процессорных) СБИС. Был сделан вывод, что архитектуры этих СБИС, их высочайшие уровень сложности и используемые микроэлектронные технологии делают создание российских аналогов с целью достижения импортозамещения практически невозможным. Следует добавить, что на современном этапе, даже в случае закупки лицензии, воспроизвести заинтересовавший образец зарубежной ЭКБ будет чрезвычайно трудно, практически невозможно в силу еще и отсутствия в нашей стране готовых к такой сложной работе коллективов, оцениваемых в несколько сотен разноплановых высококвалифицированных специалистов. Например, в [1] был выделен микропроцессор IBM Power 9 [2] и обосновывалась целесообразность его лицензирования как очень удачного скалярного процессора, который хорош и как процессор-менеджер для вычислительных узлов суперкомпьютеров с большим числом специализированных СБИС-ускорителей. Такой вариант вычислительных узлов характерен для существующих суперкомпьютеров и будет еще больше востребован в будущем. Исходя из похожих соображений, в КНР в 2016 г. была приобретена лицензия на Power 8 за 100 млн долларов. При этом дополнительно около 50 млн долларов пришлось затратить на организацию работ по этой лицензии, включая создание соответствующего коллектива в 350 специалистов.

Наличие отечественных суперкомпьютеров и обладание технологиями их разработки – один из основных факторов обеспечения информационной и даже национальной безопасности страны. В связи

с этим, существующее сильное, на первый взгляд, даже безнадежное отставание в области их ЭКБ представляется совершенно недопустимым. Один из вариантов решения этой проблемы за счет усиления работ по ПОС-СБИС также рассматривался в [1]. Прошел год, и теперь имеется возможность оценить жизненность этого варианта, который годился не только для того, чтобы достичь существующего мирового уровня ЭКБ отстающими, но и вырваться вперед тем, кто этим уровнем уже обладал.

Основная идея предлагавшегося варианта решения проблемы импортозамещения ЭКБ для российских суперкомпьютеров состояла в разработке множества разнообразных ПОС-СБИС. Предполагалось, что такие ПОС-СБИС будут иметь много каналов обработки (процессорных ядер) и разделяться на следующие три типа.

Тип 1 – ненастраиваемая СБИС, прямая аппаратная реализация в каналах того или иного алгоритма. При этом обычно достигается максимальная производительность и энергоэффективность. Недостатки – узкая специализация; предельная производительность ограничена плотностью размещения функциональных устройств, которая, в свою очередь, ограничена допустимой плотностью мощности потребления (~0,4…0,5 Вт/мм$^2$).

Тип 2 – СБИС с настройкой средней сложности, программируемые и/или реконфигурируемые каналы с небольшой универсальной частью в виде сверхлегкого RISC-процессора, но с большим числом подключенных к нему сильно специализированных функциональных устройств. По производительности приближается к СБИС типа 1, если удается минимизировать накладные расходы из-за программируемости и реконфигурируемости. Это наиболее востребованный вариант ПОС-СБИС, но задача их разработки наиболее сложна на содержательном уровне в сравнении с другими типами ПОС-СБИС.

Тип 3 – СБИС с легкой настройкой, массовое применение ядер типа простых RISC-процессоров с классической микроархитектурной схемой реализации с небольшой добавкой специальных операций в их наборы команд. Характерна простота реализации. Возможно достижение высокой производительности за счет массового параллелизма, энергоэффективность немного лучше, чем для обычной ЭКБ суперкомпьютеров за счет простоты RISC и специализации. Тем не менее, проблема больших накладных расходов по энергетике для программируемых процессоров с классической схемой реализации остается.

Такие ПОС-СБИС в той или иной степени позволяют выйти на уровни производительности и энергоэффективности лучших образцов зарубежной ЭКБ, но при этом в целом гораздо проще в разработке, поскольку реализуют гораздо меньшую функциональность. Для разработки ПОС-СБИС не требуются большие коллективы, вполне достаточно наличие порядка двух десятков сотрудников, но многое при этом зависит от правильности выбранных решений по архитектуре и микроархитектуре.

Разработка ПОС-СБИС не так уж и нова, особенно для отечественных разработчиков суперкомпьютеров для задач информационной безопасности и обработки сигналов. В современных условиях важность этому направлению придают два новых фактора: первый фактор – исторически сложившееся существенное отставание отечественной ЭКБ суперкомпьютеров от зарубежной, причем ПОС-СБИС – это вариант сокращения этого отставания; второй фактор – в силу резкого замедления развития КМОП-технологий в мире (освоен уровень 14−12 нм и возможно достижение предела в 7 нм до 2020 г.) дальнейшее развитие ЭКБ прогнозируется именно через создание множества ПОС-СБИС.

Итак, предложенный вариант создания отечественных ПОС-СБИС с целью достижения импортозамещения совпадает с мировыми трендами разработки процессорных СБИС в ближайшее десятилетие с целью дальнейшего улучшения вычислительных устройств, пока не появятся новые технологии, получившие название «пост-Муровских».

Прошедший год подтвердил идею целесообразности создания ПОС-СБИС. К сожалению, лавры первенства пока достались не российским, а китайским специалистам, осуществившим значительный рывок в области суперкомпьютерной ЭКБ именно за счет удачной разработки образцов ПОС-СБИС разных типов. Определенные успехи были достигнуты и в нашей стране. Далее рассматриваются некоторые зарубежные и отечественные достижения.

Изначально остановимся на уточнении понятия проблемной ориентации. Его можно связать с установившейся классификацией суперкомпьютеров по двум характеристикам: $BW$ – баланс пиковой пропускной способности оперативной памяти и пиковой производительности, единица измерения B/F, байт на операцию над 64-разрядными числами с точкой (флоп, далее обозначим FP64); $V$ – объем оператив-

ной памяти, единица измерения ПБ, петабайт, то есть $10^{15}$ байт. Выделенные классы суперкомпьютеров на практике определяют и их проблемную ориентацию, то есть для какого типа задач наиболее характерно их применение. Далее будут важны три из четырех выделенных классов: GP {*BW* 0,01…0,1; *V* 10…100} – научно-технические расчеты, из-за универсальности этих суперкомпьютеров возможно их применение для решения задач суперкомпьютеров других классов, если не требуется достижение предельных уровней производительности и энергоэффективности; RM {*BW* 0,1…1,0; *V* 0,01…0,1} – обработка сигналов и изображений, задачи информационной безопасности (например, блочные и потоковые шифраторы, хеш-функции), малые и средние сети с глубоким обучением; CB {*BW* 1…2; *V* 100…1000} – обработка больших данных, графовые базы данных в оперативной памяти, социально-экономические задачи, большие информационные системы, большие сети с глубоким обучением. Для каждого из этих классов проблема импортозамещения в области ЭКБ выглядит по-разному.

В отечественных суперкомпьютерах *GP-класса* используются преимущественно зарубежные многоядерные мультитредовые (несколько аппаратных тредов на ядро) суперскалярные процессоры (CPU) и массово-мультитредовые (от нескольких десятков до нескольких сотен тредов на ядро) графические процессоры (GPU). Их можно заменить на отечественные микропроцессоры семейств Эльбрус и Байкал, хотя и с потерей производительности при замене процессора на процессор на один-два порядка, что связано не только с меньшей вычислительной мощностью, но и с менее развитыми архитектурами. Зарубежные изделия позволяют за счет аппаратно поддержанной мультитредовости эффективно работать с памятью даже в режимах с плохой пространственно-временной локализацией обращений к ней, что для современных и будущих приложений крайне важно.

Потери из-за импортозамещения могут быть в определенной степени компенсированы за счет применения большего числа отечественных процессоров, а также введения ПОС-СБИС с массово-мультитредовой архитектурой типа [3], выступающих посредниками при работе с памятью, и ПОС-СБИС для ускорения собственно счета.

Хорошим примером эффективности использования сопроцессоров-ускорителей счета является суперкомпьютер Окриджской национальной лаборатории США (ORNL) с названием SUMMIT [4]. Сообщение о его запуске появилось 8 июня 2018 г., и он претендует в настоящее время на место самого мощного суперкомпьютера в мире. Его пиковая производительность составляет 200 PF (петафлопс, $10^{15}$ операций над FP64) за счет того, что в его вычислительных узлах к двум Power 9 подключены платы с шестью массово-мультитредовыми графическими процессорами NVIDIA GPU Volta [5]. GPU Volta, в свою очередь, включает в себя специализированные блоки-ускорители (TPU, устройства тензорных операций) для работы с матрицами 4×4 с элементами в виде 16-разрядных (FP16) и 32-разрядных (FP32) чисел с плавающей точкой.

Эффект от введения TPU значителен, для одного GPU Volta для FP64 в блоках стандартной арифметики пиковая производительность составляет 7,5 TF (терафлопс, $10^{12}$ операций в секунду), а для чисел FP32 пиковая производительность равна 15 TF. В блоках TPU этого GPU на FP16 и FP32 пиковая производительность суммарно равняется 120 TF. За счет этого производительность на нейровычислениях суперкомпьютера SUMMIT оценивается в 3,3 EF (экзафлопс, $10^{18}$ операций в секунду). Дополнительно отметим, что уже сообщалось, что на задаче из области геномики была достигнута за счет таких блоков TPU реальная производительность в 1,88 EF [4]. Заметим, что на задачах над FP64 без применения специальных ускорительных блоков типа TPU уровень экзафлопса планируется достичь лишь в 2021 г. в следующем суперкомпьютере ORNL с названием Frontier [4].

Вообще говоря, замечено, что GP-суперкомпьютеры дрейфуют в сторону CB-класса, что связано с увеличением сложности и детализацией по учитываемым физическим эффектам решаемых научно-технических задач, а также с расширением областей применения GP-суперкомпьютеров.

Насколько известно автору, суперкомпьютеры непосредственно *CB-класса* имеются в настоящее время лишь в виде изготовленных по специальным закрытым проектам военных образцов в трех странах – в США (развитие массово-мультитредового суперкомпьютера Cray XMT, Центр АНБ в штате Юта), а также, по сведениям из мировой экспертной среды, суперкомпьютеры стратегических военных разведок Китая и Японии, созданные на базе процессоров с массово-мультитредовой архитектурой, имеющие кодовые названия, соответственно, «Удар грома» и «Стрела времени».

В России попытка организации создания суперкомпьютера СВ-класса была осуществлена в виде проекта СКСН Ангара [6], но в итоге по достаточно странным причинам не получила должной поддержки в целом. Полученные результаты: создание сети «Ангара» [7], которая близка по характеристикам к мировому уровню 10-летней давности, но является для нашей страны уникальной; создание RTL-модели базового для СКСН Ангара 64-тредового микропроцессора J7 на ПЛИС и его базового программного обеспечения; реализация потактовой параллельной имитационной модели СКСН Ангара и получение на ней результатов исследований [8]. Есть сведения из экспертной среды, что опубликованные материалы по проекту СКСН Ангара были использованы при создании китайского суперкомпьютера «Удар грома». Итак, получается, что в настоящее время работы в классе СВ-суперкомпьютеров в нашей стране особенно запущены. Зарубежная ЭКБ для этого класса суперкомпьютеров к нам не поставляется, поскольку засекречена в странах-разработчиках.

В области отечественных суперкомпьютеров *RM-класса* ситуация гораздо лучше, поскольку накоплен отличный опыт применения сверхбольших программируемых интегральных схем (FPGA) фирм Xilinx и Altera, а в последнее время для повышения энергоэффективности стали разрабатываться ПОС-СБИС разного типа. Таким образом, невозможное повторение в отечественных условиях FPGA для импортозамещения не так уж и необходимо. В области ПОС-СБИС для обработки сигналов достигнуты заметные успехи инженеров молодого и среднего поколения отечественного оборонно-промышленного комплекса (ОПК). Это означает, что появились, хоть и небольшие, но хорошо подготовленные коллективы специалистов по микроэлектронике, их опыт может быть применен не только в создании ЭКБ для военных бортовых и встроенных систем, но и для создания ЭКБ суперкомпьютеров RM-класса, а в перспективе и СВ-класса. Это неизбежно произойдет, поскольку в бортовых и встроенных системах также уже требуются суперкомпьютеры, хотя и другого уровня производительности, но с применением тех же технологий.

Наибольшего внимания при решении проблемы импортозамещения требуют суперкомпьютеры RM- и СВ-класса, причем для СВ-класса у нас даже замещать нечего, надо разрабатывать отечественное ЭКБ практически с нуля. В процессе работ по импортозамещению большую пользу можно извлечь из внимательного изучения мирового опыта применения ПОС-СБИС в суперкомпьютерах этих классов. Причины особенной активности работ в мире именно для этих классов суперкомпьютеров состоят в следующем.

Традиционно, суперкомпьютеры RM- и СВ-класса необходимы в большей степени спецслужбам и военным, их уровень требований по производительности ничем не ограничивается, что нельзя сказать про суперкомпьютеры GP-класса. Это установившееся мнение существует как за рубежом (оно даже зафиксировано Президентом США Б. Обамой в указе №13702 от 29 июля 2015 г. о Национальной суперкомпьютерной инициативе), так и в нашей стране. Тем не менее, в прошедшие несколько лет, особенно в последний год, такое представление значительно изменилось. Оказалось, что из-за появления новых общезначимых приложений (блокчейн, криптовалюта, нейровычисления, видеообработка), требующих схожих вычислений с теми, что поддерживают суперкомпьютеры этих классов, круг пользователей и разработчиков резко расширился, поскольку такие средства стали вдруг глобально востребованными. В результате достаточно быстро нашлись эффективные решения посредством создания ПОС-СБИС разных типов, что позволило в короткие сроки создать специализированные суперкомпьютеры и вычислительные облачные серверы на их основе с чудовищно большой производительностью, на порядки превышающей суперкомпьютеры из рейтинга самых мощных суперкомпьютеров мира Top500. Появилось даже понятие специализированных вычислительных средств «планетарного масштаба» [9]. Лучшей иллюстрации рассмотренного в [1] подхода импортозамещения и прорыва за счет разработки разных ПОС-СБИС и не придумать, но это заслуга, в основном, опять же китайских специалистов.

Наиболее впечатляющий результат – создание суперкомпьютера RM-класса в виде вычислительного облачного сервера для криптовалюты Bitcoin (используется хеш-функция SHA-2) с эквивалентной производительностью $3{,}7\times10^{21}$ операций в секунду [9]. Это на три порядка больше, чем в лучших стационарных суперкомпьютерах, применяемых в мире спецслужбами и военными, например, таких, как упоминавшийся суперкомпьютер ORNL SUMMIT. Такой успех был достигнут благодаря применению в качестве вычислительных узлов такого облачного сервера специализированных устройств-майнеров, содержащих большое число узкоспециализированных ПОС-СБИС (тип 1) с прямой реализацией алго-

ритма SHA-2. Они подключены к одному процессору-менеджеру ARM-архитектуры, который обеспечивает сервисные функции и выход в межузловую сеть. Такие майнеры выпускаются появившимися в период 2008−2013 гг. китайскими фирмами Bitmain, Innosilicon, iBeLink, Pinidea, которые входят в крупнейший китайский консорциум Tianneng Bo Information Technology Co., Ltd, включающий также много филиалов и научно-исследовательских центров в Пекине, Чжен-Чжоу, Нанкине, Шанхае и Силиконовой долине в США. Такое объединение государственных организаций, частных фирм и научных институтов крайне важно для процесса создания ПОС-СБИС.

Например, майнер Antminer S9 (фирма Bitmain) имеет габаритные размеры 350×135×158 мм и воздушное охлаждение от двух вентиляторов. Кроме небольшой платы с управляющим процессором и интерфейсами он содержит три больших платы, на каждой из которых имеются по 63 ПОС-СБИС BM1387. BM1387 изготовлена на фабрике TSMC (Тайвань) по технологии 16 нм, ее производительность составляет 74 GH/c ($74\times10^9$ хеш-функций в секунду), мощность потребления – 7,8 Вт, энергоэффективность – 9,5 GH/c/Вт. По оценке автора, BM1367 имеет не менее 100 конвейерных каналов с прямой реализацией алгоритма SHA-2. Общая производительность майнера Antminer S9 составляет 14 TH/c ($14\times10^{12}$ хеш-функций в секунду), общая мощность потребления 1475 Вт. Значимость этой разработки можно оценить исходя из приведенных ниже сравнений.

BM1387 (площадь кристалла оценивается в 15 $мм^2$) сравним с GPU Volta (его площадь 815 $мм^2$, технология 12 нм), который применен в суперкомпьютере ORNL Summit. Этот GPU на алгоритме SHA-2 развивает производительность 7,6 GH/c, что в 9,7 раза меньше, чем в BM1387. При этом BM1387 потребляет лишь 7,6 Вт, а GPU Volta – 250 Вт то есть по энергоэффективности выигрыш более, чем в 300 раз.

Если сравнивать с FPGA фирмы Xilinx, то, например, в Kintex K7-410T на реализации SHA-2 достигается производительность в 1,5 GH/c при мощности потребления 30 Вт, то есть энергоэффективность 0,05 GH/c/Вт не намного лучше, чем у GPU Volta (0,0304 GH/c/Вт).

Для объективности отметим, что объединение в одном майнере 189 СБИС BM1387 снижает общую энергоэффективность по сравнению с одной такой СБИС на порядок. Виной этому – накладные расходы на организацию параллельной работы такого большого числа СБИС. Ну и, конечно, следует помнить, что BM1387 может выполнять только алгоритм SHA-2. Это принципиальный недостаток такого типа ПОС-СБИС с прямой реализацией алгоритмов, от которого, однако, пытаются избавиться в настраиваемых (или полиморфных) ПОС-СБИС типа 2 и 3 Это наиболее актуальные направления в ПОС-СБИС, особенно тип 2.

Образцы настраиваемых ПОС-СБИС типа 2 реализованы, например, для криптовалюты Dashcoin, которая использует алгоритм хеширования X11. Этот алгоритм представляет собой последовательное применение 11 алгоритмов хеширования, которые были претендентами на втором этапе выбора стандарта SHA-3. Применяемые в майнерах для алгоритма X11 китайские СБИС вызвали значительный интерес в среде отечественных специалистов. Главный вопрос состоял в выборе схем реализации в этих СБИС достаточно разных алгоритмов по ресурсам и времени обработки. Публикации по таким ПОС-СБИС отсутствуют. Тем не менее, в результате работы в экспертной среде удалось составить следующее представление о таких СБИС:

это гибридный многотайловый процессор, ПОС-СБИС типа 2;

тайлы процессора содержат одинаковую для всех тайлов часть в виде управляющего RISC-процессора и небольшой статической памяти, а также специальную часть в виде множества специальных сложно-функциональных устройств, причем эта часть в разных тайлах процессора может быть разной;

сложно-функциональные устройства реконфигурируемы, ориентированы на выполнение криптографических операций, общее число их типов около десятка, наиболее вероятно, что такой тайл похож на структуру VLIW крипто-СБИС, описанную в [10] разработчиками из Института в Чжень-Чжоу и Фуданьского университата в Шанхае;

загрузка тайлов заданиями, которые представляют собой фрагменты алгоритмов средней степени сложности, производится в динамике, задания для одного алгоритма могут выполняться на разных тайлах.

Платы майнеров представляют собой сети из таких ПОС-СБИС. Например, на одной плате майнера Innosilicon A5 с двусторонним монтажом имеются 57 ПОС-СБИС собственно для вычислений и 46 ком-

муникационных СБИС. Распределение заданий на такой сети также производится динамически, как и внутри ПОС-СБИС между тайлами.

Американский вариант организации специальной части тайла ПОС-СБИС типа 2 рассмотрен в работе [13] исследователей Техасского университета. Этот проект не закончился изготовлением реального кристалла, был только проведен синтез СБИС по RTL-описанию. Однако в методическом и системном плане это явно можно считать знаковой работой, на нее часто ссылаются в публикациях по крипто-СБИС. Данное исследование выполнялось по заказу АНБ США, но дальнейшая его судьба неизвестна, хотя по результатам сначала была идея запустить два новых проекта: первый – секретный, с подключением опытных специалистов по микроэлектронике; второй – открытый, коммерческий, с участием талантливой молодежи, отобранной по конкурсу из множества разных стран.

Тем не менее, описанный в [13] процессор Cryptoraptor нельзя назвать удачным. По мнению автора, это произошло из-за чрезмерного использования в нем метода настройки за счет реконфигурируемости, что объясняется принятием за аксиому его разработчиками существующего мифа о предпочтительности реконфигурируемости по сравнению с программируемостью для достижения большей производительности и энергоэффективности. Заметим, что этот миф был развеян исследователями из Стэнфордского университета под руководством одного из самых известных специалистов по архитектуре в мире Вильяма Далли [14]. Было доказано, что при локализации программного управления непосредственно при функциональных устройствах, включая соединяющий их коммутатор, а также выделении из регистрового файла и локализации при функциональных устройствах небольших регистровых памятей, применение программируемой в соответствии с логикой программы подкачки в быстрые памяти вместо просто кэш-памятей может значительно сократить накладные расходы по энергетике и, в целом, на порядок уменьшить энергетические затраты на выполнение одной команды по сравнению, например, с RISC-процессором со стандартной реализацией. Такие новые микроархитектурные приемы целесообразно использовать и для управления функциональными устройствами в специальных частях тайлов ПОС-СБИС типа 2.

Американские варианты ПОС-СБИС типа 2 рассмотрены в работах [9, 11], но это исследовательские проекты.

Для суперкомпьютеров RM-класса разрабатывались также и ПОС-СБИС типа 3. Здесь следует упомянуть американскую недавнюю разработку – гибридную СБИС Celerity [15] типа системы на кристалле (технология 16 нм, фабрика TSMC, кристалл 25 мм$^2$, 385 млн транзисторов), RTL-описание было разработано в открытой экосистеме RISC-V и включает кроме интерфейсных блоков:

пять мощных 64-разрядных RISC-процессоров RV64G в универсальной управляющей части, каждый RV64G может выполнять 150 разных команд, обрабатываемые данные 64-разрядные целые (Int64), FP32 и FP64, имеется набор команд с атомарными операциями с памятью, по 16 Кбайт кэш-памяти команд и данных, пятиступенчатый конвейер с дисциплиной выполнения команд в порядке следования («in-order»), тактовая частота 625 МГц, площадь 0,97 мм$^2$;

2D-сеть из 496 тайлов, в каждом тайле легкий 32-разрядный RISC-процессор RV32IM целочисленной обработки, две быстрых блокнотных памяти по 4 Кбайт соответственно для команд и данных (памяти тайлов адресуются через единое адресное пространстве всей 2D-сети, но разрешена лишь операция удаленной записи), пятиступенчатый конвейер с дисциплиной выполнения команд «in-order», маршрутизатор сети с четырьмя дуплексными линками по 80 битов (дополнительно по 10 битов линии подтверждения) в каждую сторону, тактовая частота 1,05 ГГц, площадь 0,024 мм$^2$;

ускорительный блок нейровычислений BNN в девять слоев, объем памяти 13,4 Мбайт, площадь 0,356 мм$^2$;

10 резервных ядер RM32IM.

Перечисленные блоки занимают около 70% площади кристалла, при этом универсальная часть занимает 20%, а специальная в виде сети 2D-тайлов и блока BNN – 50%.

Интересны полученные результаты по эффективности использования специализации в этой ПОС-СБИС:

блок BNN меньше ядра RV64G в 2,7 раза, однако на задаче обработки изображений BNN развивает в 200 раз большую производительность, чем RV64G, при этом он в 200 раз энергоэффективнее;

если использовать 496 ядер RV32IM для обмена данными с целью обслуживания блока BNN, то в такой конфигурации обработка изображений происходит в 1200 раз быстрее, чем на RV64G, и в 400 раз энергоэффективнее.

Рассмотренные ПОС-СБИС можно отнести к ЭКБ суперкомпьютеров RM-класса, но настоящей сенсацией было появление в начале этого года ПОС-СБИС для суперкомпьютеров СВ-класса. Это ПОС-СБИС для майнера Antminer E3 криптовалюты Etherium. В этой криптовалюте применяется хеш-функция Dagger-Hashimoto (или алгоритм Ethash В. Бутерина). Этот алгоритм строит ориентированный граф рекурсивных вызовов функций и работает с ним. В настоящее время такой граф занимает до 4 Гбайт памяти и имеет тенденцию к быстрому росту.

Ранее считалось, что с этим алгоритмом может справиться только GPU, алгоритм из-за этого и был придуман, чтобы исключить возможность использования китайских ПОС-СБИС, по поводу которых считалось, что они ориентированы на высокоскоростные вычисления, но не могут выполнять работу с большой памятью, используя ее большую пропускную способность. Исходя из такого положения, сеть Etherium стала безопасным «заповедником» для GPU, в ней сейчас насчитывается около 7,5 млн таких процессоров. За счет такого активного использования GPU фирма NVIDIA получала солидный доход. Такая сеть имеет черты СВ-суперкомпьютера. Однако представление о том, что для алгоритма Ethash нельзя разработать конкурирующую с GPU ПОС-СБИС, оказалось ошибкой.

Появление Antminer E3, в основе которого лежит архитектура 64/128-тредового микропроцессора, причем, предположительно, ранее реализованного в проекте создания китайского военного суперкомпьютера «Удар грома», вызвало панику в среде пользователей сети Etherium и даже понижение акций фирмы NVIDIA, поскольку GPU этой фирмы с появлением такого майнера оказывались экономически сильно невыгодными. В конечном итоге между фирмой Bitmain и администраторами сети Etherium было достигнуто компромиссное соглашение – стоимость китайского майнера была увеличена в два с лишним раза. Однако потенциальная опасность для GPU от существования такого майнера на мультитредовой ПОС-СБИС осталась.

СБИС для Antminer E3 можно классифицировать как ПОС-СБИС типа 2, в которой в тайлах легкий однотредовый RISC-процессор заменен на мультитредовый. Дело в том, что в алгоритме Ethash требуется не только работать с памятью с большой пропускной способностью, но и быстро выполнять алгоритм SHA-3.

В свою очередь, американцы также запустили год назад проект DARPA HIVE [19] по разработке мультитредовой специализированной ПОС-СБИС в виде мультитредового микропроцессора HIVE для работы с графическими базами данных, который бы обеспечил на три порядка большую энергоэффективность, чем GPU. Ранее проект процессора такого типа также рассматривался в работе [20], он велся в очень серьезной организации, Линкольновской лаборатории Массачусетского технологического института. Связь этого проекта с DARPA HIVE пока не установлена, но оба процессора специализированы для работы с разреженными матрицами.

В рамках проекта DARPA HIVE будут доработаны и модули памяти для обеспечения эффективного доступа к словам, а не к большим страницам памяти, как это сейчас принято в кристаллах DDR. Исходя из того, что интегратором в этом проекте будет фирма Northrop Grumman, можно предположить, что такой процессор может быть применен в военных интеллектуальных бортовых системах, предположительно в крупном ударном беспилотнике типа X-45 и X-47.

Вообще, тема ПОС-СБИС для систем искусственного интеллекта в настоящее время популярна и обещает разработчикам таких изделий большие прибыли. Специалисты фирмы Bitmain оценили темпы роста потребностей в ПОС-СБИС для нейросетей с глубоким обучением как близкие к тем, которые были на начальных этапах освоения средств для криптовалюты Bitcoin. В связи с этим ими была быстро выпущена ПОС-СБИС BM1680 с пиковой производительностью 2 TF на FP32.

Множество выпускаемых ПОС-СБИС для суперкомпьютеров RM- и СВ-класса, да и других вычислительных устройств для типов задач этих классов суперкомпьютеров, постоянно расширяется. Одновременно происходит отработка быстрых технологических методов создания таких ПОС-СБИС, где важную роль играет накопление и использование в дальнейшем готовых макроблоков для быстрой сборки на их основе той или иной ПОС-СБИС. Это, например, поставлено целью в проекте DARPA CRAFT [21].

В нашей стране работы по ПОС-СБИС также ведутся, и уже получены первые положительные результаты. Можно выделить следующие законченные выпуском кристаллов отечественные разработки: 96-ядерная ПОС-СБИС типа 3 для задач информационной безопасности с производительностью порядка $10^{12}$ операций в секунду [16]; 21-ядерная ПОС-СБИС типа 3 NM6408 для обработки сигналов, изображений и нейровычислений с пиковой производительностью 0,5 TF на FP32 [17]. Рассматриваются возможности повышения пиковой производительности в следующих СБИС после NM6408 до уровня 3 TF [18]. Разработаны проекты сверхлегких RISC-ядер LWP и динамически реконфигурируемых специальных частей DRCP для тайлов ПОС-СБИС типа 2, эти работы суммируют подходы работ [10−14], а также проекты мультитредовых ядер mt-LWP для ПОС-СБИС отечественных суперкомпьютеров СВ-класса, что приближает их к китайскому майнеру Antminer E3, имеющему корни в российском проекте [6]. На ядра mt-LWP стоит обратить особое внимание, поскольку если не будет похожей отечественной разработки, то «Цифровая экономика России», в которой ориентируются на использование технологии блокчейн Etherium, будет построена либо на американских GPU (а они уже закупаются Россией в больших количествах), либо на мультитредовых ПОС-СБИС китайских майнеров Antminer E3.

- Работы по отечественным ПОС-СБИС ведутся в ряде локальных проектов. Необходимо усиление таких работ за счет расширения и придания этому направлению комплексности с помощью мобилизации для участия в них новых организаций-разработчиков микроэлектронных изделий и прикладных систем, а также привлечения научных и учебных заведений. Очень важна постановка информационно-аналитических работ с систематизаций направлений и подходов, обеспечение распространения новых сведений в среде отечественных разработчиков ПОС-СБИС. Целесообразность всего этого показывает китайский опыт разработки ПОС-СБИС коммерческими фирмами, но с сильнейшей государственной поддержкой не только от гражданских, но и военных структур и спецслужб.

  С практической точки зрения, для накопления опыта и повышения технологичности необходима организация работ по созданию разного типа макроблоков для ПОС-СБИС, включая: сложнофункциональные устройства для базовых операций прикладных областей; процессорные ядра разной степени сложности и управления выполнением операций; векторные и мультитредовые ядра, внутрикристальные сети с разной топологией. Макроблоки должны быть собраны в общую библиотеку, и должна быть разработана методика быстрого синтеза проектов ПОС-СБИС на базе такой библиотеки по заданным исходным данным.

## Литература


1. *Эйсымонт Л.К.* Гибридная стратегия развития элементной базы // Открытые системы. 2017. № 2. URL = https://www.osp.ru/os/2017/02/13052216/.
2. *Sadsiven S.K.*, *Tompto B.W.*, *Kalla R.*, *Starke W.J.* IBM Power 9 Processor Architecture. IEEE Micro. March/April 2007. P. 40−51.
3. *Kogge P.M.*, *Brockman J.B.*, *Harper D.T.*, *Smith B.*, *Callahan C.D.* Computer systems with lightweight multi-threaded architectures. United States Patent Application Publication. Pub. No.: US 2007/0198785 A1. Pub. Date: Aug. 23. 2007. 34 p.
4. *Schor D.* ORNL's 200-petaFLOPS Summit Supercomputer Has Arrived, To Become World's Fastest. URL = https://fuse.wikichip.org/news/1351/ornls-200-petaflops-summit-supercomputer-has-arrived-to-become-worlds-fastest/.
5. *Durant L. et al*. Inside Volta: The World's Most Advanced Data Center GPU. 10 May 2017. URL = https://devblogs.nvidia.com/parallelforall/inside-volta/.
6. *Слуцкин А.И.*, *Эйсымонт Л.К.* Российский суперкомпьютер с глобально адресуемой памятью // Открытые системы. 2007. № 9. С. 42−51. URL = http://www.osp.ru/os/2007/09/4569294/.
7. *Жабин И.*, *Макагон Д.*, *Симонов А.*, *Сыромятников Е.*, *Фролов А.*, *Щербак А.* Кристалл для «Ангары» // Суперкомпьютеры. 2013. № 4(16). С. 46−49.
8. *Эйсымонт Л.К.*, *Семенов А.С.*, *Соколов А.А.*, *Фролов А.С.*, *Шворин А.Б.* Моделирование российского суперкомпьютера «Ангара» на суперкомпьютере. Суперкомпьютерные технологии в науке, образовании и промышленности / Под ред. акад. *В.А. Садовничего*, акад. *Г.И. Савина*, чл.-корр. РАН *Вл.В. Воеводина*. М.: Изд-во Московского университета. 2009. 5 с. URL = http://hpc-russia.ru/book1/18.pdf.
9. *Khazraee M.*, *Gutierrez L.V.*, *Magaki I.*, *Tailor M.B.* Specializing a Planet's Computation: ASIC Clouds. IEEE Micro. May/June 2017. P. 62−69.
10. *Wei L.*, *Xiaoyang Z.*, *Zibin D.*, *Longmei N.*, *Tao C.*, *Chao M.* A High Energy-Efficient Reconfigurable VLIW Symmetric Cryptographic Processor with Loop Buffer Structure and Chain Processing Mechanism // Chinese Journal of Electronics. Nov. 2017. V. 26. № 6. P. 1161−1167.

11. *Novatsky T. et al*. Pashing the Limits of Accelerator Efficiency While Retaining Programmability // IEEE High performance computer architecture conference. 2016. 13 p. URL = http://pages.cs.wisc.edu/~vinay/pubs/hpca16-lssd.pdf.
12. *Govindaraju V. et al*. Dynamically Specialized Datapaths for Energy Efficient Computing // 17th IEEE International Symposium on High Performance Computer Architecture. 2011. 12 p. URL = http://research.cs.wisc.edu/vertical/papers/2011/hpca11-dyser.pdf.
13. *Sayliar G. G.*, *Chiou D.* Cryptoraptor: High Throughput Reconfigurable Cryptographic Processor // ICCAD 2014. 8 p. URL = https://pdfs.semanticscholar.org/6d0e/a468a737ef666d6a9da0ee1c2f1cff3f43b3.pdf.
14. *Dally W.*, *Balford J. et al*. An Energy-Efficient Processor Architecture // IEEE Computer Architecture Letters. 2008. V. 7. № 1. 4 p. URL = http://cva.stanford.edu/projects/elm/pubs/cal_Jan_2008.pdf.
15. *Ajavi T. et al*. Experiences Using the RISC-V Ecosystems to Design an Accelerator-Centric SoC in TSMC 16 nm // First Workshop on Computer Architecture Research with RISC-V. August 2017. 6 p.
16. *Елизаров С.Г.*, *Лукьянченко Г.А.*, *Марков Д.С.*, *Монахов А.М.*, *Роганов В.А.*, *Сизов А.Д.* Программируемые на языках высокого уровня специализированные СБИС для задач информационной безопасности с предельной энергоэффективностью // Сб. докладов VI Междунар. научно-практич. конф. «Управление информационной безопасностью в современном обществе». 5–7 июня 2018. Москва.
17. *Эйсымонт А.Л.*, *Черников А.В.*, *Косоруков Д.Е.*, *Насонов И.И.*, *Комлев А.А.* Гетерогенная многопроцессорная система на кристалле с производительностью 512 Gflops // Текст доклада. Микроэлектроника-2017. Алушта. 11 с. URL = https://module.ru/upload/images/1507808903Гетерогенная многопроцессорная СнК с производительностью 512 Gflops.pdf.
18. *Черников В.М.*, *Виксне П.Е.* Перспективы повышения характеристик и расширения областей применения транстерафлопсных СБИС семейства NeroMatrix // Сб. докладов VI Междунар. научно-практич. конф. «Управление информационной безопасностью в современном обществе». 5–7 июня 2018. Москва.
19. Broad Agency Announcement Hierarchical Identify Verify Exploit (HIVE) Microsystems Technology Office DARPA-BAA-16-52. August 2. 2016. 50 p.
20. *Song W.S.*, *Kepner J.*, *Gleyzer V.*, *Nguyen H.T.*, *Kramer J.I.* Novel Graph Processor Architecture // Lincoln Laboratory Journal. 2013. V. 20. № 1. P. 92–104.
21. Broad Agency Announcement Circuit Realization At Faster Timescales (CRAFT) Microsystems Technology Office DARPA-BAA-15-55. August 17. 2015. 46 p.

**Поступила 3 августа 2018 г.**

# Configurable special VLSI – real basis for future exascale supercomputers (foreign and domestic experience)



**L.K. Eisymont** – Ph.D.(Phys.-Math.), The Federal Register of Scientific and Technical Experts of the Russia Ministry of Education and Science (Moscow)
E-mail: verger-lk@yandex.ru

The direction of creation of problem-oriented specialized VLSI (POS-VLSI) as a variant of asymmetric replacement of the foreign element-component base (ECB) in domestic supercomputers and cloud servers of the highest performance range is considered. Configurable POS-VLSI, in contrast to VLSI with direct hardware implementation of algorithms (non-configurable), have the ability to implement them by programming or reconfiguring their blocks. Some POS-VLSI have a weak or strong specialization. Strong specialization is most important, but it is difficult to do without waste of productivity and energy efficiency. This is a serious scientific and technical problem, the emerging options for its solution require careful study. In the world practice, the development of PIC-VLSI is also underway, Chinese developers have made special progress. In the world practice, the development of this type of POS-VLSI is also under way, Chinese developers have made special progress. Judging by the results achieved, this becomes a strategy for ECB development under threatening restrictions because of the approaching end of the development of CMOS technologies and the way ahead of the rival countries in the field of high technologies. The article systematizes and analyzes achievements in this field, compares it with domestic developments.

**References**

1. *Eisymont L.K.* Gibridnaya strategiya razvitiya ehlementnoj bazy // Otkrytye sistemy. 2017. № 2. URL = https://www.osp.ru/os/2017/02/13052216/.
2. *Sadsiven S.K., Tompto B.W., Kalla R., Starke W.J.* IBM Power 9 Processor Architecture. IEEE Micro. March/April 2007. P. 40–51.
3. *Kogge P.M.*, *Brockman J.B.*, *Harper D.T.*, *Smith B.*, *Callahan C.D.* Computer systems with lightweight multi-threaded architectures. United States Patent Application Publication. Pub. No.: US 2007/0198785 A1. Pub. Date: Aug. 23. 2007. 34 p.
4. *Schor D.* ORNL's 200-petaFLOPS Summit Supercomputer Has Arrived, To Become World's Fastest. URL = https://fuse.wikichip.org/news/1351/ornls-200-petaflops-summit-supercomputer-has-arrived-to-become-worlds-fastest/.
5. *Durant L. et al.* Inside Volta: The World's Most Advanced Data Center GPU. 10 May 2017. URL = https://devblogs.nvidia.com/parallelforall/inside-volta/.
6. *Sluckin A.I., Eisymont L.K.* Rossijskij superkomp'yuter s global'no adresuemoj pamyat'yu // Otkrytye sistemy. 2007. № 9. S. 42–51. URL = http://www.osp.ru/os/2007/09/4569294/.

7. *Zhabin I., Makagon D., Simonov A., Syromyatnikov E., Frolov A., Sherbak A.* Kristall dlya «Angary» // Superkomp'yutery. 2013. № 4(16). S. 46–49.
8. *Eisymont L.K., Semenov A.S., Sokolov A.A., Frolov A.S., Shvorin A.B.* Modelirovanie rossijskogo superkomp'yutera «Angara» na superkomp'yutere. Superkomp'yuternye tekhnologii v nauke, obrazovanii i promyshlennosti / Pod red. akad. *V.A. Sadovnichego*, akad. *G.I. Savina*, chl.-korr. RAN *Vl.V. Voevodina*. M.: Izd-vo Moskovskogo universiteta. 2009. 5 s. URL = http://hpc-russia.ru/book1/18.pdf.
9. *Khazraee M.*, *Gutierrez L.V.*, *Magaki I.*, *Tailor M.B.* Specializing a Planet's Computation: ASIC Clouds. IEEE Micro. May/June 2017. P. 62–69.
10. *Wei L.*, *Xiaoyang Z.*, *Zibin D.*, *Longmei N.*, *Tao C.*, *Chao M.* A High Energy-Efficient Reconfigurable VLIW Symmetric Cryptographic Processor with Loop Buffer Structure and Chain Processing Mechanism // Chinese Journal of Electronics. Nov. 2017. V. 26. № 6. P. 1161–1167.
11. *Novatsky T. et al.* Pashing the Limits of Accelerator Efficiency While Retaining Programmability // IEEE High performance computer architecture conference. 2016. 13 p. URL = http://pages.cs.wisc.edu/~vinay/pubs/hpca16-lssd.pdf.
12. *Govindaraju V. et al.* Dynamically Specialized Datapaths for Energy Efficient Computing // 17th IEEE International Symposium on High Performance Computer Architecture. 2011. 12 p. URL = http://research.cs.wisc.edu/vertical/papers/2011/hpca11-dyser.pdf.
13. *Sayliar G. G.*, *Chiou D.* Cryptoraptor: High Throughput Reconfigurable Cryptographic Processor // ICCAD 2014. 8 p. URL = https://pdfs.semanticscholar.org/6d0e/a468a737ef666d6a9da0ee1c2f1cff3f43b3.pdf.
14. *Dally W.*, *Balford J. et al.* An Energy-Efficient Processor Architecture // IEEE Computer Architecture Letters. 2008. V. 7. № 1. 4 p. URL = http://cva.stanford.edu/projects/elm/pubs/cal_Jan_2008.pdf.
15. *Ajavi T. et al.* Experiences Using the RISC-V Ecosystems to Design an Accelerator-Centric SoC in TSMC 16 nm // First Workshop on Computer Architecture Research with RISC-V. August 2017. 6 p.
16. *Elizarov S.G., Luk'yanchenko G.A., Markov D.S., Monahov A.M., Roganov V.A., Sizov A.D.* Programmiruemye na yazykah vysokogo urovnya specializirovannye SBIS dlya zadach informacionnoj bezopasnosti s predel'noj ehnergoehffektivnost'yu // Sb. dokladov VI Mezhdunar. nauchno-praktich. konf. «Upravlenie informacionnoj bezopasnost'yu v sovremennom obshchestve». 5–7 iyunya 2018. Moskva.
17. Ejsymont A.L., Chernikov A.V., Kosorukov D.E., Nasonov I.I., Komlev A.A. Geterogennaya mnogoprocessornaya sistema na kristalle s proizvoditel'nost'yu 512 Gflops // Tekst doklada. Mikroehlektronika-2017. Alushta. 11 s. URL = https://module.ru/upload/images/1507808903Гетерогенная многопроцессорная СнК с производительностью 512 Gflops.pdf
18. *Chernikov V.M., Viksne P.E.* Perspektivy povysheniya harakteristik i rasshireniya oblastej primeneniya transteraflopsnyh SBIS semejstva NeroMatrix // Sb. dokladov VI Mezhdunar. nauchno-praktich. konf. «Upravlenie informacionnoj bezopasnost'yu v sovremennom obshchestve». 5–7 iyunya 2018. Moskva.
19. Broad Agency Announcement Hierarchical Identify Verify Exploit (HIVE) Microsystems Technology Office DARPA-BAA-16-52. August 2. 2016. 50 p.
20. *Song W.S.*, *Kepner J.*, *Gleyzer V.*, *Nguyen H.T.*, *Kramer J.I.* Novel Graph Processor Architecture // Lincoln Laboratory Journal. 2013. V. 20. № 1. P. 92–104.
21. Broad Agency Announcement Circuit Realization At Faster Timescales (CRAFT) Microsystems Technology Office DARPA-BAA-15-55. August 17. 2015. 46 p.

УДК 004.383

# Перспективы повышения характеристик и расширения областей применения транстерафлопсных СБИС семейства NeuroMatrix



**В.М Черников** – к.т.н., гл. конструктор – начальник отделения, ЗАО НТЦ «Модуль» (Москва)
E-mail: tchern@module.ru
**П.Е. Виксне** – начальник сектора, ЗАО НТЦ «Модуль» (Москва)
E-mail: pvixne@module.ru

Рассмотрены вопросы возможности построения многоядерной гетерогенной СБИС транстерафлопсной производительности как дальнейшего развития отечественной системы на кристалле (СнК) 1879ВМ8Я (NM6408MP) семейства NeuroMatrix. С целью дальнейшего повышения производительности и энергоэффективности предложено: перейти на новую технологию (14 нм и ниже); увеличить число процессорных ядер на кристалле; модифицировать эти процессорные ядра для повышения их вычислительной мощности за счет использования новой аппаратуры и большей специализации на конкретных классах задач.

***Ключевые слова:*** *архитектура NeuroMatrix, мультипроцессорная система на кристалле (МПСнК), многоядерная гетерогенная СнК, DSP, векторная архитектура, VLIW, Late Write.*

This paper represents the approaches to build multicore heterogeneous SoC of over teraflop performance with use of NeroMatrix family SoC NM6408MP as prototype. To achieve more performance and power efficiency it is proposed as follows: usage of new technology (14 nm and bellow), increasing of processor cores on chip, modification of these processor cores to take more computing power with help of new hardware adding and more problem-oriented specialization.

***Keywords:*** *NeuroMatrix architecture, multiprocessor system on chip (MPSoC), multicore heterogeneous SoC, DSP, vector architecture, VLIW, Late Write.*

**DOI: 10.18127/j20729472-201803-05**

Ц е л ь  р а б о т ы – рассмотреть возможность построения многоядерной гетерогенной СБИС транстерафлопсной производительности как дальнейшего развития отечественной системы на кристалле (СнК) 1879ВМ8Я (NM6408MP) семейства NeuroMatrix.

## Увеличение числа процессорных ядер на кристалле как основное направление развития вычислительной техники

Если до 2003 г. ежегодный рост производительности процессоров составлял приблизительно 50%, то с тех пор он оценивается в 22% в год. Это объясняется тем, что из-за ограничения на потребляемую мощность нельзя больше заметно увеличивать тактовую частоту. Так, если с 1987 г. по 2003 г. тактовая частота каждый год росла на 40%, то с 2003 г. по настоящее время – рост около 1% в год. Таким образом, основное увеличение производительности процессоров ранее шло за счет увеличения тактовой частоты и дальнейшего усложнения аппаратуры при использовании супер-скалярных технологий, при этом эффективность использования как потребляемой мощности, так и транзисторов на кристалле стала снижаться. Сравним, например, два процессора одной фирмы Intel: i486 и Pentium 4. Последний в шесть раз быстрее i486 при изготовлении по той же топологии, но при этом потребляет в 23 раза больше мощности. То есть на выполнение одной команды более новый процессор тратил почти в четыре раза больше энергии, чем его предшественник.

В 2004 г. фирма Intel объявила о том, что она будет повышать производительность не за счет увеличения тактовой частоты, а за счет использования нескольких процессорных ядер на одном кристалле. Причем эти ядра становятся более простыми, но зато более эффективными с точки зрения отношения производительность/потребляемая мощность. Этот подход постепенно стал общепринятым, поскольку, как и прежде, каждый год число транзисторов на кристалле увеличивается на 40…50% (удваивается каждые 18−24 месяца), что позволяет постоянно наращивать число ядер. Если в настоящее время обычным стало 4−8 процессорных ядра на кристалле, то в течение пяти ближайших лет их число, по оценкам,

увеличится в 3,5 раза, а к 2020 г. их будет насчитываться до тысячи. Появился даже новый термин: не просто система на кристалле (СнК), а именно мультипроцессорная система на кристалле (МПСнК).

### Однородные и гетерогенные мультипроцессорные системы на кристалле

При построении МПСнК появляется выбор: реализовать ее в виде однородной вычислительной системы из универсальных процессорных ядер или в виде гетерогенной вычислительной системы, включающей в себя управляющую часть (один или несколько универсальных процессорных ядер) и исполнительную (ускорители или DSP-процессоры). Конечно, с точки зрения пользователя однородные системы проще программировать. Для таких приложений, как, например, офисные, когда надо обработать несколько мало взаимосвязанных между собой процессов, такие системы подойдут лучше всего. Однако есть широкий класс задач, относящихся к потоковой обработке данных: 2D/3D-графика, аудио- и видеообработка, радиолокация, навигация и т.д., когда их решение на универсальных процессорах на один-два порядка менее эффективно с точки зрения потребляемой мощности, чем для графических ускорителей или DSP-процессоров. Исследователи из Lawrence Berkeley National Laboratory посчитали, что при моделировании климата с сеткой в 1 км суперкомпьютер петафлопной производительности, построенный на универсальных процессорах фирмы AMD, будет потреблять 179 МВт. Если же для этой цели использовать специализированные микросхемы, потребление уменьшится до 2,5 МВт. Такова цена за универсальность.

Попробуем теперь понять, за счет чего на своих задачах DSP-процессоры более эффективны, чем универсальные процессоры. В качестве примера на рис. 1 приведены требования к радиолокационной обработке, выдвигаемые широким кругом отечественных пользователей.

Как видно из рис. 1, четко выделяются т р и  у р о в н я  обработки сигналов РЛС.

На  п е р в о м  у р о в н е  происходит предварительная обработка данных, поступающих с многоканального АЦП. Разрядность АЦП обычно находится в диапазоне от 12 до 14 разрядов, требуемый темп оцифровки – 40…120 млн выборок/с. Результатами обработки на данном уровне являются несколько потоков 16-…18-разрядных данных с частотой 12…30 МГц. Требуемый уровень производительности составляет 1,2…8,0 млрд операций с фиксированной точкой в секунду. В основном применяется однородная параллельная обработка потоков данных. Как правило, предварительная обработка реализуется в виде жесткой логики, обычно на ПЛИС.

На  в т о р о м  у р о в н е  осуществляется первичная обработка сигналов РЛС. Результатами обработки на этом уровне являются несколько потоков 16-…32-разрядных данных с частотой 12…30 МГц.

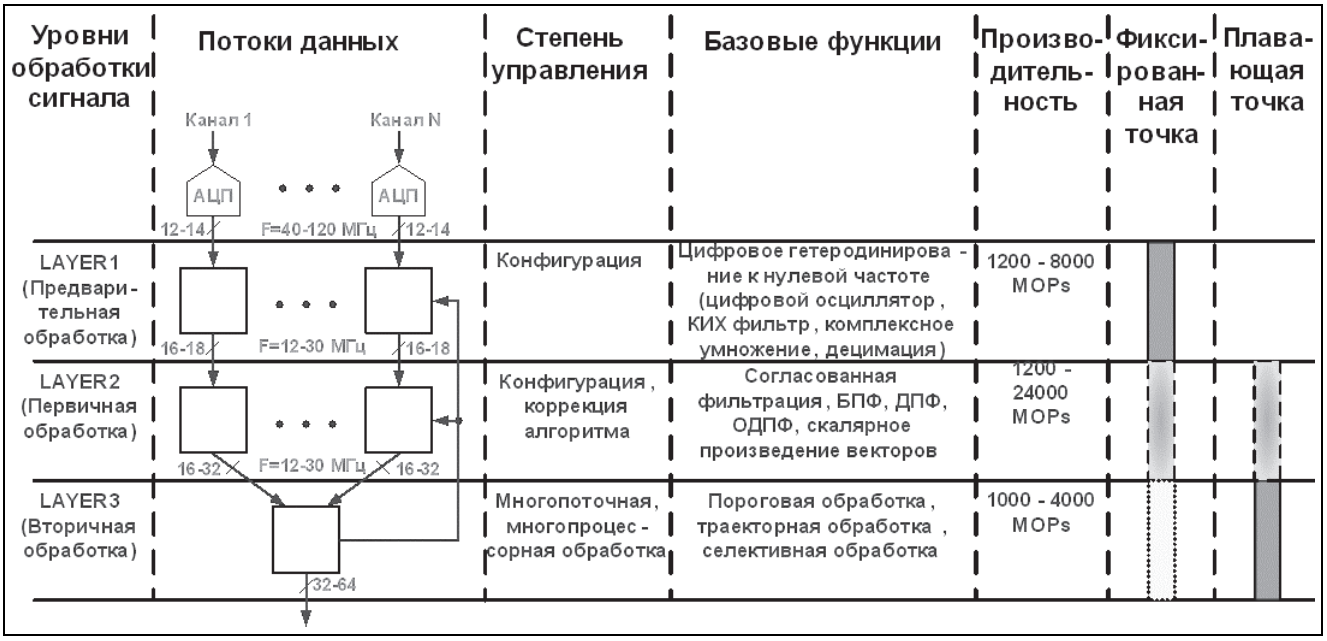


**Рис. 1.** Уровни обработки радиолокационных сигналов

Требуемый уровень производительности составляет 1,2…24,0 млрд операций с фиксированной или плавающей точкой в секунду. Применяется более сложная обработка потоков данных, решаются такие задачи, как согласованная фильтрация, БПФ, ДПФ и ОДПФ, скалярное произведение векторов. Обычно первичная обработка реализуется на многопроцессорной системе, строящейся на базе высокопроизводительных DSP.

На третьем уровне происходит вторичная обработка сигналов РЛС. Результатом обработки на данном уровне является один поток 32-…64-разрядных данных. Требуемый уровень производительности составляет 1,0…4,0 млрд операций с фиксированной или плавающей точкой в секунду. Помимо вычислительных задач, решаются сложные задачи управления. Как правило, вторичная обработка осуществляется несколькими высокопроизводительными универсальными процессорами.

Как видно, для решения задачи радиолокационной обработки в целом требуется использование именно гетерогенной многопроцессорной системы, попытка решить эту задачу на однородной вычислительной системе из универсальных процессоров будет крайне неэффективна.

Далее рассмотрим более подробно первичную обработку сигнала и попробуем понять, почему для нее должны использоваться DSP, а не универсальные процессоры. Задачи, решаемые на этом уровне, имеют некоторые общие особенности, которые сильно отличают их от приложений, характерных для универсальных процессоров, а именно:

получение решения в реальном времени;

обработка потоков данных: для хранения программ достаточно внутренней памяти, при этом поток данных поступает из внешней памяти;

достигается значительный параллелизм на уровне данных (возможно выполнение одновременно до нескольких тысяч однотипных операций над разными данными);

значительные вычислительные затраты на реорганизацию данных (упаковка, распаковка, перестановка и т.д.), что делает затруднительным использование традиционных SIMD-архитектур, где данные затраты могут быть слишком велики;

небольшие циклы с вложенностью от двух до шести, причем 95% времени вычислений тратится на двух вложенных циклах, итераций циклов обычно насчитывается 10 и больше, а некоторые циклы имеют сотни и тысячи итераций;

требуется большая пропускная способность не только для внутренней, но и для внешней памяти;

в основном обрабатываются 16- и 32-разрядные целочисленные данные, иногда используются данные с плавающей точкой одинарной точности;

выполняется гораздо больше арифметических операций на одну операцию ввода/вывода по сравнению с приложениями для универсальных процессоров;

данные, хранящиеся в памяти, обладают хорошей пространственной локальностью и очень слабой временной локальностью: обычно процессор загружает небольшой блок данных, обрабатывает его, и в дальнейшем этот блок повторно практически не используется, что приводит к тому, что кэш универсальных процессоров используется крайне неэффективно.

Поскольку при разработке архитектуры DSP-процессоров учитываются указанные выше особенности, и они, в отличие от универсальных процессоров, предназначены для решения гораздо менее широкого класса задач, их использование будет всегда более эффективным, чем использование универсальных процессоров на этих задачах.

### DSP-процессор для гетерогенных мультипроцессорных систем на кристалле

Требование обработки потоков данных в реальном времени с производительностью, сравнимой с производительностью универсальных процессоров, и потреблением, характерным для встроенных систем, заставляет использовать гетерогенные МПСнК. Общее управление вычислительным процессом с помощью операционной системы осуществляется одним или несколькими универсальными процессорными ядрами, а основные вычисления производятся на нескольких процессорных DSP-ядрах. В зависимости от класса решаемых задач и требований к производительности и потреблению число процессорных ядер как одного, так и другого типа может варьироваться в значительных пределах.

В качестве DSP-процессорного ядра для таких гетерогенных систем ЗАО НТЦ «Модуль» предлагает использовать процессорное ядро архитектуры NeuroMatrix, имеющее RISC-ядро, матричный сопро-

цессор операций в формате с плавающей точкой и целочисленный матричный сопроцессор. Оно отличается следующими основными особенностями:

векторно-конвейерным принципом выполнения операций, позволяющим одним потоком команд задавать большое число параллельно выполняемых операций («зацепление» группы команд по данным);

наличием одного или нескольких векторных сопроцессоров, работающих под управлением одного RISC-процессора и имеющих свои шины ввода/вывода данных;

использованием внешних адресных генераторов, что обеспечивает эффективное использование адресных регистров ядра при адресации векторных данных;

конвейером с очередью команд на ступени выборки операндов из памяти, обеспечивающим эффективную работу с банками внутренней и внешней памяти, имеющими различную глубину конвейера;

выдачей и приемом данных на одной и той же ступени конвейера (Late Write), что резко снижает число конфликтов по данным;

использованием команд, задающих несколько операций (статический VLIW).

Характеристики DSP-процессорного ядра архитектуры NeuroMatrix для разных технологических реализаций представлены в таблице.

**Таблица.** ***Характеристики DSP-процессорного ядра архитектуры NeuroMatrix в зависимости от используемой технологии***

| Характеристики | Технология (год) | | |
|---|---|---|---|
| | 65 нм (2015 г.) | 28 нм (2017 г.) | 14 нм (2020 г.) |
| Название СБИС | NM6407(1879ВМ6Я) | NM6408MP(1879ВМ8Я) | NM6409MP |
| Число процессорных ядер:<br>ARM<br>NeuroMatrix | <br>–<br>2 | <br>5<br>16 | <br>21<br>64 |
| Производительность СБИС/1 ядро, GFLOPS | 16/16 | 512/32 | 3072/48 |
| Производительность СБИС/1 ядро, GMACS (8-разрядные данные) | 12/12 | – | 2304/36 |
| Удельная производительность, GFLOPS/Вт | 8 | 15 | 30 |
| Тактовая частота, ГГц | 0,5 | 1 | 1,5 |

Перечислим возможные пути дальнейшего повышения производительности СБИС с процессорными ядрами семейства NeuroMatrix.

***Увеличение числа процессорных ядер на кристалле.*** Переход на технологию 14 нм позволит построить систему на кристалле, содержащую 64 процессорных ядра NeuroMatrix. Схема такой системы представлена на рис. 2. Ее основным элементом является кластер PCxx, состоящий из четырех процессорных систем NPUxxx (ядро NeuroMatrix + память) и управляющей процессорной системы CPUxx (ядро ARM + память). Обмен данными внутри кластера осуществляется по 64-разрядной шине AXI 3.0. Для связи между соседними четырьмя кластерами служит шина AXI следующего, второго уровня иерархии. Шина AXI третьего уровня позволяет осуществить обмен каждого кластера (процессорного ядра) системы с каждым. Для связи с внешним миром служат линки на базе PCI-express (HI, EL0-EL3) и шины внешней памяти DDR3 (DDR3 #0–DDR3 #4). Такая система на кристалле позволит на частоте 1,5 ГГц обеспечить пиковую производительность до 3 TFLOPS.

***Использование одним процессорным ядром двух сопроцессоров.*** Процессорное RISC-ядро NeuroMatrix позволяет работать с векторным сопроцессором как с фиксированной, так и с плавающей точкой, что можно видеть на рис. 3. Пока такая возможность не использовалась. В NM6407(1879ВМ6Я) у каждого из двух процессорных ядер был свой сопроцессор: один с фиксированной точкой, другой – с плавающей. В NM6408MP(1879ВМ8Я) все процессорные ядра использовали только сопроцессор с плавающей точкой. Возможность работы ядер с двумя типами сопроцессоров позволит значительно увеличить функциональные возможности при смешанной обработке данных с фиксированной и плавающей точкой.

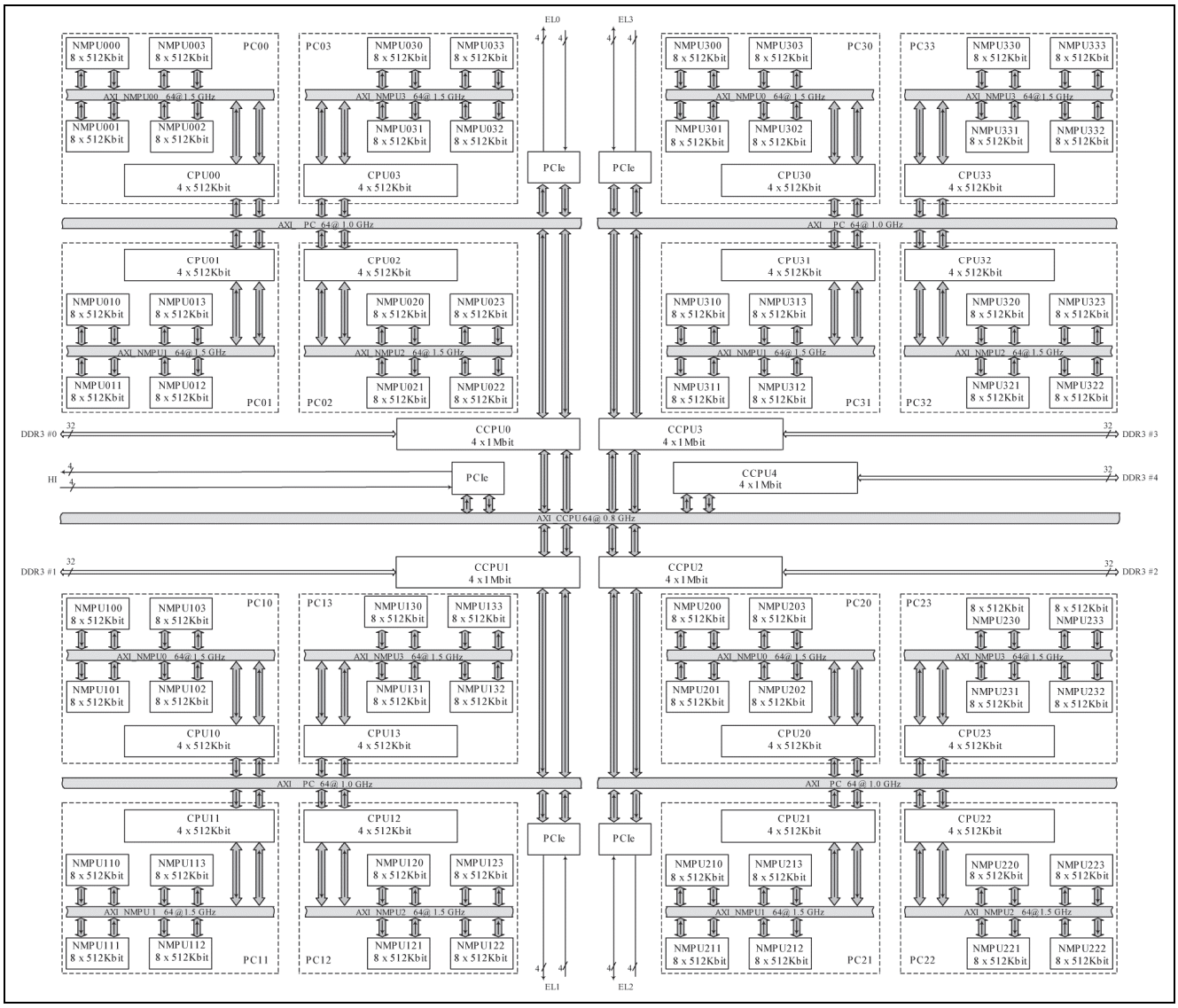


**Рис. 2.** Схема системы на кристалле с 64 процессорными ядрами NeuroMatrix

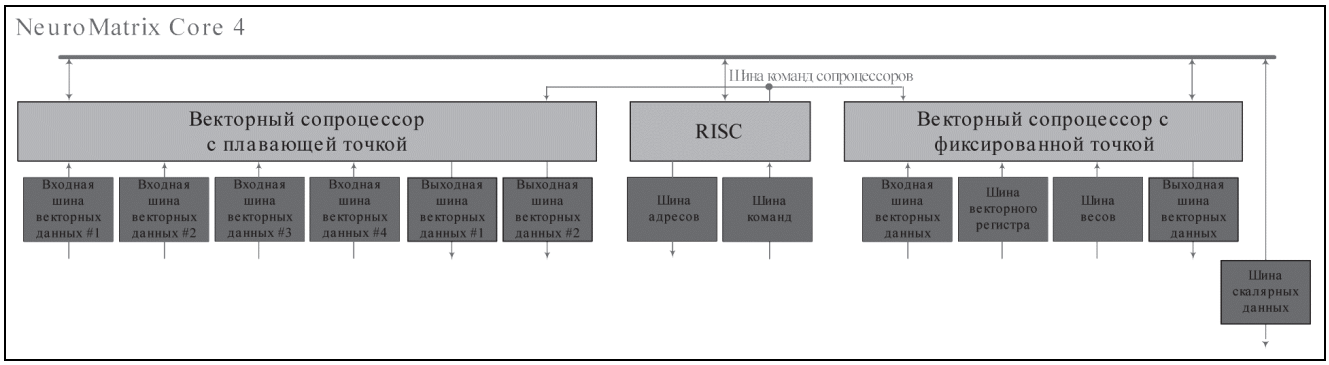


**Рис. 3.** Схема одного RISC-ядра с двумя сопроцессорами – с фиксированной и плавающей точкой

***Увеличение числа процессорных ячеек с четырех до восьми в сопроцессоре арифметики с плавающей точкой.*** Система команд процессорного ядра NeuroMatrix поддерживает работу с восемью процессорными ячейками, как показано на рис. 4. В реальности пока использовались только четыре таких ячейки. Таким образом, возможно увеличение производительности в два раза на одно процессорное ядро. Проблемой при этом становится обеспечить доступ для процессорных ячеек за данными в память.

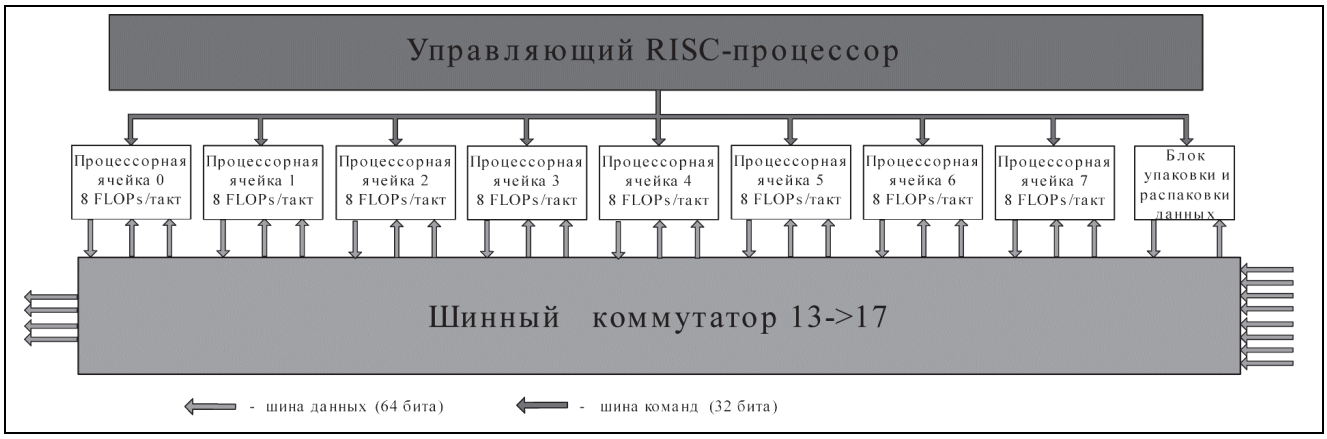


**Рис. 4.** Схема сопроцессора арифметики с плавающей точкой с восемью процессорными ячейками

В настоящее время память процессорной системы расслоена на восемь банков, что обеспечивает чтение до четырех операндов и записи до двух результатов за такт. Использование восьми процессорных ячеек может потребовать расслоения памяти до 16 банков.

***Использование процессорных ячеек с арифметикой заказчика.*** Процессорные ячейки сопроцессора арифметики с плавающей точкой поддерживают операции над 32- и 64-разрядными данными с плавающей точкой и только. Существует возможность сохранить всю внутреннюю структуру СБИС, но арифметические операции внутри процессорных ячеек заменить на те, что необходимы для аппаратной поддержки алгоритмов заказчика (рис. 5). Таким образом, за счет специализации можно заметно увеличить производительность на отдельных классах задач.

- Основным элементом любой современной аппаратуры обработки сигналов является ядро процессора цифровой обработки. Все повышающиеся требования к увеличению производительности при обработке сигналов (в настоящее время от 10 до 100 млрд операций в секунду, в перспективе на порядок больше) требуют новых подходов при проектировании таких систем. Поскольку заметное повышение тактовых частот из-за ограничений, накладываемых ростом потребляемой мощности, больше не является основным фактором увеличения производительности, для обработки больших объемов сигнальной информации в режиме реального времени требуется создание многоядерных вычислительных систем, обладающих большей вычислительной мощностью и гибкостью, чем одноядерные системы. Так, например, один из самых мощных зарубежных процессоров Tiger-SHARC TS-201 фирмы Analog Devices, работающий на частоте до 600 МГц, имеет пиковую производительность 3,6 GFLOPS. Очевидно, чтобы достичь производительности в сотни GFLOPS, необходимо использовать несколько ядер на кристалле. Таким путем, например, пошла фирма Texas Instruments, создав систему на кристалле 66AK2H12 из четырех процессорных ядер Cortex-A15 и восьми собственных DSP-ядер C66x с общей производительностью до 153,6 GFLOPS.

  Создание вычислительных систем с производительностью до нескольких TFLOPS позволит отечественной промышленности производить конкурентоспособную аппаратуру цифровой обработки сигналов для систем радиолокации и гидроакустики. Кроме того, на данной аппаратуре возможно эффективное решение и других важных задач, таких как реализация нейросетей глубокого обучения.

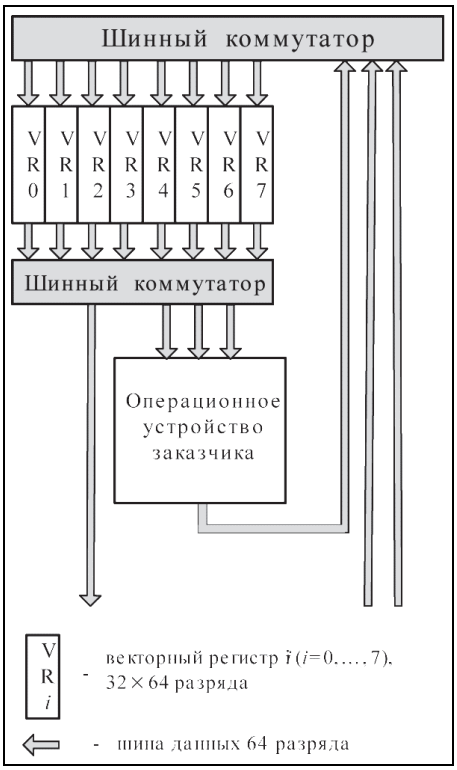


**Рис. 5.** Схема процессорной ячейки с арифметикой заказчика

## Литература

1. *Henessy J.L.*, *Patterson D.A.* Computer Architecture: A Quantitative Approach. Fifth Edition. Elvister Science. 2012. 856 p.
2. The Future of Computing Performance: Game Over or Next Level / Ed. by *S.H. Fuller*, *L.I. Millett.* Committee on Sustanding Growth in Computing Performance. National Research Council. National Academy of Science. 2010. 180 p.
3. Multiprocessor System-on-Chip / Ed. by *M. Hübner*, *J. Becker.* Hardware Design and Tool Integration. Springer Science + Business Media. 2011. 270 p.
4. *Shahbahrami A.*, *Juurlink B.*, *Vassiliadis S.* A Comparison Between Processor Architectures for Multimedia Applications // Proc. of 15th Annual Workshop on Circuits, System and Signal Processing (ProRISC 2004). The Netherlands. November 2004. P. 138−152.
5. *Talla D.*, *John L.K.*, *Burger D.* Bottlenecks in Multimedia Processing with SIMD Style Extensions and Architectural Enhancements // IEEE Transactions on Computers. August 2003. V. 52. № 8. P. 1015−1031.
6. *Эйсымонт Л.К.* Гибридная стратегия развития элементной базы // Открытые системы. 2017. № 2. URL = https://www.osp.ru/os/2017/02/13052216/.
7. *Эйсымонт А.Л.*, *Черников А.В.*, *Косоруков Д.Е.*, *Насонов И.И.*, *Комлев А.А.* Гетерогенная многопроцессорная система на кристалле с производительностью 512 Gflops // Текст доклада. Микроэлектроника-2017. Алушта. 11 с. URL = https://module.ru/upload/images/1507808903Гетерогенная многопроцессорная СнК с производительностью 512 Gflops.pdf.
8. *Черников А.В.*, *Черников В.М.*, *Виксне П.Е.*, *Шелухин А.М.* Высокопроизводительное процессорное ядро NMC4 для обработки векторных данных в форматах с плавающей и фиксированной точками // Текст доклада. Микроэлектроника-2017. Алушта. 8 с. URL = https://module.ru/upload/images/1507871419Высокопроизводительное процессорное ядро NMC4 для обработки векторных данных в форматах с плавающей и фиксированной точками.pdf.


**Поступила 3 августа 2018 г.**

# New approaches to obtain overteraflop performance and to expand the range of use for NeroMatrix family SoCs



**V.M. Chernikov** – Ph.D.(Eng.), Main Designer – Head of Department, RC «Module» (Moscow)
E-mail: tchern@module.ru
**P.E. Viksne** – Head of Sector, RC «Module» (Moscow)
E-mail: pvixne@module.ru

This paper represents the approaches to build multicore heterogeneous SoC of over teraflop performance with use of NeroMatrix family SoC NM6408MP as prototype. To achieve more performance and power efficiency it is proposed as follows: usage of new technology (14 nm and bellow), increasing of processor cores on chip, modification of these processor cores to take more computing power with help of new hardware adding and more problem-oriented specialization. The development of computing systems with a capacity of up to several TFLOPS will enable the domestic industry to produce competitive digital signal processing equipment for radar and hydroacoustic systems. In addition, this equipment can effectively solve other important tasks, such as the implementation of neural networks of in-depth training.

**References**

1. *Henessy J.L.*, *Patterson D.A.* Computer Architecture: A Quantitative Approach. Fifth Edition. Elvister Science. 2012. 856 p.
2. The Future of Computing Performance: Game Over or Next Level / Ed. by *S.H. Fuller*, *L.I. Millett.* Committee on Sustanding Growth in Computing Performance. National Research Council. National Academy of Science. 2010. 180 p.
3. Multiprocessor System-on-Chip / Ed. by *M. Hübner*, *J. Becker.* Hardware Design and Tool Integration. Springer Science + Business Media. 2011. 270 p.
4. *Shahbahrami A.*, *Juurlink B.*, *Vassiliadis S.* A Comparison Between Processor Architectures for Multimedia Applications // Proc. of 15th Annual Workshop on Circuits, System and Signal Processing (ProRISC 2004). The Netherlands. November 2004. P. 138−152.
5. *Talla D.*, *John L.K.*, *Burger D.* Bottlenecks in Multimedia Processing with SIMD Style Extensions and Architectural Enhancements // IEEE Transactions on Computers. August 2003. V. 52. № 8. P. 1015−1031.
6. *Eisymont L.K.* Gibridnaya strategiya razvitiya ehlementnoj bazy // Otkrytye sistemy. 2017. № 2. URL = https://www.osp.ru/os/2017/02/13052216/.
7. *Ejsymont A.L., Chernikov A.V., Kosorukov D.E., Nasonov I.I., Komlev A.A.* Geterogennaya mnogoprocessornaya sistema na kristalle s proizvoditel'nost'yu 512 Gflops // Tekst doklada. Mikroehlektronika-2017. Alushta. 11 s. URL = https://module.ru/upload/images/1507808903Гетерогенная многопроцессорная СнК с производительностью 512 Gflops.pdf.
8. *Chernikov A.V., Chernikov V.M., Viksne P.E., SHeluhin A.M.* Vysokoproizvoditel'noe processornoe yadro NMC4 dlya obrabotki vektornyh dannyh v formatah s plavayushchej i fiksirovannoj tochkami // Tekst doklada. Mikroehlektronika-2017. Alushta. 8 s. URL = https://module.ru/upload/images/1507871419Высокопроизводительное процессорное ядро NMC4 для обработки векторных данных в форматах с плавающей и фиксированной точками.pdf.

УДК 004

# Проблемы локализации адресов обращений к памяти при обработке данных в некоторых системах обработки сигналов и варианты их решения



**А.В. Зайцев** – начальник сектора, АО «Всероссийский НИИ Радиотехники» (Москва)
E-mail: alex.zaitsev@gmail.com
**Л.К. Эйсымонт** – к.ф.-м.н., Федеральный реестр экспертов научно-технической сферы Минобрнауки России (Москва)
E-mail: verger-lk@yandex.ru

Поставлена задача исследования процессов, происходящих в системе обработки сигналов с целью определения положения наблюдаемых тел в некотором пространстве, а далее – выбора вычислительных устройств для эффективного выполнения этих процессов. В качестве основной особенности процессов выделены режимы пространственно-временной локализации обращений к памяти в них. Определены основные процессы системы, предложены наиболее подходящие для них векторные и мультитредовые процессоры, а также программная эмуляция мультитредовой архитектуры на многоядерном процессоре. В качестве объектов изучения выбраны процессоры Эльбрус и NeuroMatrix NM6408. Рассмотрен выбор перспективных процессоров для системы.

***Ключевые слова:*** *обработка сигналов, графовые базы данных, принятие решений, пространственно-временная локализация, тест APEX-MAP, векторные процессоры, мультитредовые процессоры, программная эмуляция, комплексные методики оценочного тестирования.*

The task of studying the processes taking place in the signal processing system in order to determine the position of the observed bodies in a certain space, and then – the choice of computing devices for the effective implementation of these processes. As the main feature of the processes, the modes of spatio-temporal localization of memory accesses are distinguished. The main processes of the system are determined, the most suitable vector and multithreaded processors, software emulation of multithreaded architecture on a multi-core processor are offered. The objects of study are processors Elbrus and NeuroMatrix NM6408. As a result, it is planned to select promising processors for system.

***Keywords:*** *signal processing, graph databases, decision-making, spatio-temporal localization, APEX-MAP test, vector processors, multithreaded processors, software emulation, complex evaluation testing technique.*

**DOI: 10.18127/j20729472-201803-06**

Ц е л ь  р а б о т ы – рассмотреть постановочную стадию решения проблемы выбора вычислительных устройств для выполнения процессов, происходящих в обобщенной системе обработки сигналов с целью определения положения наблюдаемых тел в некотором пространстве.

## Модель обработки сигналов и характеристики процессов обработки

Сигналы о местоположении тел поступают от приемников сигналов в блок обработки сигналов (БОС), далее из БОС информация поступает в блок принятия решения (БПР), который по командам оператора или автоматически ее обрабатывает. Основная обработка в БОС – быстрое преобразование Фурье (FFT). В БПР используется база данных о наблюдаемых телах. Для простоты представим, что она имеет формат STINGER [1] базы неоднородных данных в виде графа. Решаемые задачи в БПР пока не конкретизируются, важен лишь режим работы с памятью при их решении, после чего возникают сообщения оператору и управляющие воздействия на приемники сигналов.

Цикл «обработка сигнала – анализ и принятие решения – выработка воздействия» происходит за очень короткое время и повторяется много раз в реальном времени отслеживая тел. Качество обработки и выработанных воздействий существенно зависит от быстродействия вычислительных устройств в БОС и БПР. Авторы полагают (и это будет проверяться экспериментально), что емкие по времени вычислительные процессы БОС и БПР объясняются большими задержками выполнения в них операций обращений к памяти, которые возникают из-за плохой *пространственной и временной локализации* этих обращений.

*Пространственная локализация* отражает «предсказуемость» возникающих адресов обращений. Высокая пространственная локализация благоприятствует опережающей преднакачке строк в кэш-память процессора. *Временная локализация* отражает «кучность» выдаваемых адресов обращений. Вы-

сокая временная локализация благоприятствует попаданию в строки кэш-памяти без промахов. Процессоры могут с разной эффективностью реагировать на пространственно-временную локализацию обращений к памяти. Наилучший вариант – когда они толерантны (нечувствительны) к ней.

Для оценки эффективности процессоров при работе с памятью с середины прошлого десятилетия используется разработанный в Лаборатории Лоуренса (LBNL, США) тест APEX-map [2]. Этот тест автоматически измеряет среднее время в тактах процессора, приходящееся на одно обращение к памяти в автоматически устанавливаемых разных режимах пространственно-временной локализации. Результаты представляются в виде APEX-поверхности в системе координат <*L*, *A*> (каждая пара <*L*, *A*> определяет режим пространственно-временной локализации): *L* – параметр, влияющий на пространственную локализацию (чем больше *L*, тем она лучше); *A* – параметр, влияющий на временную локализацию (чем меньше *A*, тем локализация лучше, 1 – случайные адреса). Обращения к памяти в тесте происходят к элементам массива объемом во всю физическую память.

Авторами APEX-map была разработана также методика, по которой для любой задачи можно найти точку APEX-поверхности (пару <*L*, *A*>), с которой эта задача коррелируется наилучшим образом по пространственно-временной локализации обращений к памяти. Например, в соответствии с этой методикой крайние точки APEX-поверхности, отмеченные как *L*, *T*, *F* и *G*, были соотнесены со следующими оценочными тестами: *L* (*пространственная и временная локализации хорошие*) – тест Linpack рейтинга Top 500, умножение плотно заполненных матриц; *T* (*пространственная хорошая, временная плохая*) – транспонирование матриц, неявные методы математической физики; *F* (*пространственная плохая, временная хорошая*) – быстрое преобразование Фурье; *G* (*пространственная и временная плохие*) – тест Random Access, хеш-функция Etherium, задачи на графовых базах данных, разреженные матрицы, рейтинговые тесты Graph 500 и HPCG.

Вычислительный процесс в БОС можно сопоставить с точкой *F*, а процесс в БПР – с точкой *G*. В исследовании это еще будет уточняться с использованием собственной обновленной методики корреляции задач с точками APEX-поверхности, которая основана на применении таблиц промахов в кэш-памяти всех уровней для точек APEX-поверхности и промахах, полученных для заданной задачи.

## Обоснование выбора вычислительных средств

В точке *G* для микропроцессора Intel Sandy Bridge (частота 2,2 ГГц) обращение к памяти происходит не менее, чем за 60 тактов. Для лучшего на момент измерений процессора Эльбрус 2С ситуация в точке *G* еще хуже – задержка обращения к памяти более 100 тактов (частота 0,5 ГГц). Таким образом, соотношение времен обращения к памяти таково: Sandy Bridge – 28,2 нс, Эльбрус 2С – 208,4 нс (отставание в 7,4 раза). В точке *F* соотношение времен обращений к памяти близкое: Sandy Bridge – 3,73 нс, Эльбрус 2С – 25 нс (отставание в 6,76 раза). При этом заметим, что отставание из-за разных частот составляет 4,4 раза, остальное – за счет архитектуры Эльбрус 2С, поскольку в нем затруднена возможность эффективно работать по хаотично выдаваемым адресам памяти.

Такие характеристики процессоров в режимах точек *G* и *F* объясняют, по мнению авторов, проблемы организации эффективных вычислений в БПР и БОС систем обработки сигналов интересующего типа. Заметим, что динамическая суперскалярная архитектура Sandy Bridge на самом деле не является наиболее подходящей для работы в режимах, характерных для точек *G* и *F*, для таких приложений применяются процессоры с другими архитектурами, то есть отставание от зарубежного уровня на самом деле значительно больше.

## Пути модернизации вычислительных средств

Улучшение эффективности обработки сигналов требует снижения реально видимого приложением времени обращения к памяти, для чего можно использовать архитектурные и программные решения, чтобы совмещать вычисления и выполнение обращений к памяти в программах. Иными словами, выполнять эти обращения на фоне вычислительных операций. Для реализации этого потребуется ряд решений, суть которых в следующем.

Первоочередным является построение памяти из большого числа способных одновременно работать модулей, что позволяет распараллелить выполнение обращений, повысив таким образом пропускную способность памяти, что непосредственно снижает видимое программой время обращения к памяти. Однако, чтобы воспользоваться такой многомодульной памятью, подключенный к ней процессор

должен обладать способностью выдавать из программы большое число обращений и принимать их результаты. Не любая архитектура процессора на такое способна.

Например, ограниченность возможностей по работе с памятью микропроцессоров с динамической суперскалярной архитектурой даже с дисциплиной выполнения команд не в порядке их следования в программе («out-of-order») демонстрируют результаты теста обхода *N* списков.

***Мультитредовая архитектура и ее программная эмуляция.*** Порождение большого потока обращений к памяти обеспечивает мультитредовая архитектура процессора. Единственное условие успеха – наличие в программе большого числа одновременно выполняемых легких в плане переключения процессов-тредов и эффективная аппаратная поддержка их выполнения в процессоре.

Реализация мультитредовых архитектур – это специальная дорогостоящая разработка, которую до недавнего времени не все могли себе позволить. В Америке, Китае и Японии такие разработки были выполнены для областей, в которых финансирования не жалеют (разведка). Для широкого круга такие аппаратно поддержанные возможности оказались слишком дороги, да и недоступны по другим причинам, но обладать ими обычным пользователям было бы заманчиво, хотя бы и не с такими рекордными показателями. Исходя из этого, в работе [3] было высказано предложение программно эмулировать мультитредовую архитектуру на серийных многоядерных высокочастотных суперскалярных микропроцессорах, а также экспериментально доказана состоятельность этого предложения.

Программная эмуляция мультитредовости основана на двух базовых идеях: 1) реализация в ядре на уровне пользовательской программы (посредством run-time системы) чрезвычайно малых по времени переключения («легких») процессов-корутин; 2) трехфазовое выполнение обращения к памяти.

Ф а з а 1 – выдача в виде сообщения запроса на чтение из выполняемой (текущей) корутины.

Ф а з а 2 – пассивация выдавшей запрос корутины для освобождения ядра процессора.

Ф а з а 3 – получение результата выполнения выданного ранее запроса и активация корутины на процессорном ядре для продолжения вычислений.

Рассмотренная идея эмуляции вызывала сомнение у специалистов из-за возможных накладных расходов, связанных с большим числом передач сообщений, реализующих запросы обращений к памяти и работу с корутинами. Однако состоятельность такого подхода была доказана в работе [3] экспериментально. Авторами данной статьи эти эксперименты были повторены, результаты подтвердились [4].

***Векторная архитектура.*** Для задач, характеризующихся средней и хорошей пространственной локализацией обращений, эффективно использование векторных архитектур, поскольку вектор является непосредственным источником появления множества обращений за его элементами. Чем больше длина вектора в приложении, тем лучше работает векторная архитектура.

Однако проблемными оказываются задачи, оперирующие с короткими векторами. Проблема здесь в том, что короткие вектора не обеспечивают большой поток обращений в память, в результате она простаивает, и высокая пропускная способность не используется.

Проблему коротких векторов иногда удается обойти за счет искусственного укрупнения решаемой задачи, а именно: за счет решения поставленной задачи сразу для множества разных наборов данных. Обычно число таких одновременно решаемых задач выбирают равным числу элементов в векторном регистре процессора.

Попытки разработки отечественного векторного процессора осуществлялись с конца 1970-х годов сразу после появления американского суперкомпьютера Cray-1. Однако все они оказывались неудачными. Лишь в конце 2017 г. был изготовлен и прошел тестирование первый отечественный гибридный 21-ядерный векторный микропроцессор NM6408 [6] (главный конструктор В.М. Черников, ЗАО «НТЦ «Модуль»). Этот микропроцессор в настоящее время является самым быстрым в России (пиковая производительность 0,5 TF) и ориентирован на решение задач обработки сигналов и нейровычисления. Уже прорабатываются варианты поднять производительность в следующих микропроцессорах этого типа до 3 TF [7]. Представляет интерес оценка возможности применения NM6408 в рассматриваемой системе обработки сигналов, а также оценка перспективности его планируемых модификаций [7].

Кроме высокой производительности на скалярно-векторных ядрах (их 16 штук), микропроцессор NM6408 имеет еще следующие особенности: пять интерфейсов для работы с DDR-памятью; пять высокоскоростных линков для связи таких микропроцессоров напрямую с целью создания на их основе сети.

Высокая производительность NM6408 достигается благодаря векторной архитектуре его ядер. Это позволяет эффективно выполнять обращения к памяти, но при условии, что обрабатываемые данные размещаются подходящим образом в расслоенной на восемь блоков сравнительно небольшой локальной памяти каждого скалярно-векторного ядра объемом 512 КБ. Очевидно, что применение такой неболь-

шой локальной памяти означает, что векторные вычисления с данными из этой памяти должны совмещаться с подкачкой в нее новых данных из области памяти вне скалярно-векторного ядра. Подкачка данных в локальную память скалярно-векторного ядра возможна посредством *асинхронных коммуникационных портов* (программируемых устройств прямого доступа к памяти (DMA), их в процессоре 84 штуки) из любой внутренней памяти NM6408 и даже из внешней DDR-памяти. Для такой подкачки можно воспользоваться программными тредами обмена данными, выполняемыми на скалярно-векторных ядрах совместно с вычислительными тредами, реализующими векторные вычисления.

Итак, для повышения эффекта от векторной архитектуры в скалярно-векторных ядрах мультитредовость может быть поддержана программно посредством эмуляции, что рассматривалось ранее. Практичность этого интересно проверить экспериментально.

Для написания параллельных программ для 21-ядерного NM6408 и сети из таких микропроцессоров разработана Библиотека управления процессами и их взаимодействиями, содержащая некоторые функции стандартов MPI и SHMEM.

Словом, есть смысл поэкспериментировать с NM6408 для реализации системы обработки сигналов исследуемого типа.

***Мультитредовая архитектура.*** Все-таки наиболее перспективным подходом, позволяющим эффективно выполнять программы с любой пространственно-временной локализацией обращений к памяти, является использование аппаратно реализованной мультитредовой архитектуры, хотя бы в упрощенном виде. Как отмечалось, этот подход оказывается успешным, если удается выделить большое число легких процессов-тредов, которые может квазипараллельно выполнять такой процессор. Ключевая особенность мультитредовых архитектур – способность выдавать от множества тредов в сумме тысячи обращений к памяти, что позволяет полностью использовать пропускную способность памяти с большим расслоением.

В России в рамках проекта «Ангара» разработана архитектура мультитредовых процессоров J7 (64-тредовый) и J10 (256-тредовый) [5], но реальный мультитредовый микропроцессор изготовить не довелось. Эти работы целесообразно восстановить, тем более, что даже в достаточно удачном микропроцессоре NM6408 есть части, где, по мнению авторов, можно было бы такую архитектуру попробовать.

## Дальнейшие этапы исследования

В рамках планируемого исследования намечаются следующие этапы:

проведение оценочного тестирования в соответствии с методикой [8] процессоров семейства Эльбрус и NM6408 с целью получения основных знаний об их особенностях;

разработка специальных тестов-моделей алгоритмов, применяемых в БОС и БПР, исследование их на зарубежных суперскалярных микропроцессорах с развитыми системами сбора статистики о выполняемых программах;

исследование тестов-моделей на новых процессорах семейства Эльбрус и NM6408;

обобщение полученных на зарубежных и отечественных процессорах результатов оценочного тестирования и тестирования на тестах-моделях алгоритмов, выработка рекомендаций по выбору и по разработке новых отечественных процессоров;

имитационное моделирование новых процессоров, отвечающих выработанным рекомендациям, и исследование поведения на них тестов-моделей алгоритмов.

- Планируемое исследование согласуется со стратегией импортозамещения в области элементно-конструкторской базы (ЭКБ) суперкомпьютеров, изложенной в работах [9, 10] и ориентированной на разработку множества проблемно-ориентированных специализированных СБИС (ПОС-СБИС), а не на создание прямых аналогов лучших образцов зарубежной ЭКБ. Подход с ПОС-СБИС является реальным асимметричным ответом, но он невозможен без глубокого проникновения в прикладные области. Примером такого проникновения и является описанное в данной работе планируемое исследование.

## Литература


1. *Ediger D.*, *McColl R.*, *Riedy J.*, *Bader D.A.* STINGER: High Performance Data Structure for Streaming Graphs // 2012 IEEE Conference «High Performance Extreme Computing (HPEC)». 10−12 Sept. 2012. 5 p.
2. *Strohmaier E.*, *Shan H.* Architecture Independent Performance Characterization and Benchmarking for Scientific Applications. Volendam. The Netherlands. Oct. 2004. URL = https://ftg.lbl.gov/ApeX/mascots.pdf.
3. *Nelson J. et al*. Crunching Large Graphs with Commodity Processors. HotPar 2011. 6 p.

4. *Зайцев А.В.*, *Биконов Д.В.*, *Андрюшин Д.В.* HPGAS: опыт эмуляции массово-мультитредовых систем на многоядерных кластерных суперкомпьютерах. URL = https://www.osp.ru/netcat_files/userfiles/MSKF_2015/MSCF_2015_Fin3.pdf.
5. *Митрофанов В.В.*, *Слуцкин А.И.*, *Эйсымонт Л.К.* Суперкомпьютерные технологии для стратегически важных задач // Журнал «Электроника: НТБ». 2008. № 7. С. 66–79.
6. *Эйсымонт А.Л.*, *Черников А.В.*, *Косоруков Д.Е.*, *Насонов И.И.*, *Комлев А.А.* Гетерогенная многопроцессорная система на кристалле с производительностью 512 Gflops // Текст доклада. Микроэлектроника-2017. Алушта. 11 с. URL = https://module.ru/upload/images/1507808903Гетерогенная многопроцессорная СнК с производительностью 512 Gflops.pdf.
7. *Черников В.М.*, *Виксне П.Е.* Перспективы повышения характеристик и расширения областей применения транстерафлопсных СБИС семейства NeroMatrix // Сб. докладов VI Междунар. научно-практич. конф. «Управление информационной безопасностью в современном обществе». 5–7 июня 2018. Москва.
8. *Горбунов В.С.*, *Эйсымонт Л.К.* Комплексная методика тестирования производительности суперкомпьютеров // Вычислительные методы и программирование. 2013. Т. 14. С. 115–121. URL = http://num-meth.srcc.msu.ru/zhurnal/tom_2013/pdf/v14r216.pdf.
9. *Эйсымонт Л.К.* Гибридная стратегия развития элементной базы // Открытые системы. 2017. № 2. URL = https://www.osp.ru/os/2017/02/13052216/.
10. *Эйсымонт Л.К.* Настраиваемые специализированные СБИС – реальная основа создания будущих экзамасштабных суперкомпьютеров, зарубежный и отечественный опыт // Сб. докладов VI Междунар. научно-практич. конф. «Управление информационной безопасностью в современном обществе». 5–7 июня 2018. Москва.

**Поступила 3 августа 2018 г.**

# Memory access address localization problems in some signal processing systems and variants of their solution



**A.V. Zaitsev** – Head of Sector, JSC «All-Russia SRI of Radio Engineering» (Moscow)
E-mail: alex.zaitsev@gmail.com
**L.K. Eisymont** – Ph.D.(Phys.-Math.), The Federal Register of Scientific and Technical Experts of the Russia Ministry of Education and Science (Moscow)
E-mail: verger-lk@yandex.ru

The task of studying the processes taking place in the signal processing system in order to determine the position of the observed bodies in a certain space, and then – the choice of computing devices for the effective implementation of these processes. As the main feature of the processes, the modes of spatio-temporal localization of memory accesses are distinguished. The main processes of the system are determined, the most suitable vector and multithreaded processors, software emulation of multithreaded architecture on a multi-core processor are offered. The objects of study are processors Elbrus and NeuroMatrix NM6408. As a result, it is planned to select promising devices of the system processes. The planned study is consistent with the strategy of import substitution in the field of elemental design base of supercomputers aimed at developing a set of problem-oriented specialized VLSI rather than creating direct analogs of the best samples of foreign elemental design base.

**References**

1. *Ediger D.*, *McColl R.*, *Riedy J.*, *Bader D.A.* STINGER: High Performance Data Structure for Streaming Graphs // 2012 IEEE Conference «High Performance Extreme Computing (HPEC)». 10–12 Sept. 2012. 5 p.
2. *Strohmaier E.*, *Shan H.* Architecture Independent Performance Characterization and Benchmarking for Scientific Applications. Volendam. The Netherlands. Oct. 2004. URL = https://ftg.lbl.gov/ApeX/mascots.pdf.
3. *Nelson J. et al.* Crunching Large Graphs with Commodity Processors. HotPar 2011. 6 p.
4. *Zajcev A.V., Bikonov D.V., Andryushin D.V.* HPGAS: opyt ehmulyacii massovo-mul'titredovyh sistem na mnogoyadernyh klasternyh superkomp'yuterah // Tezisy stendovogo doklada. URL = https://www.osp.ru/netcat_files/userfiles/MSKF_2015/MSCF_2015_Fin3.pdf.
5. *Mitrofanov V.V., Sluckin A.I., Eisymont L.K.* Superkomp'yuternye tekhnologii dlya strategicheski vazhnyh zadach // Zhurnal «Ehlektronika: NTB». 2008. № 7. S. 66–79.
6. *Ejsymont A.L., Chernikov A.V., Kosorukov D.E., Nasonov I.I., Komlev A.A.* Geterogennaya mnogoprocessornaya sistema na kristalle s proizvoditel'nost'yu 512 Gflops // Tekst doklada. Mikroehlektronika-2017. Alushta. 11 s. URL = https://module.ru/upload/images/1507808903Гетерогенная многопроцессорная СнК с производительностью 512 Gflops.pdf.
7. *Chernikov V.M., Viksne P.E.* Perspektivy povysheniya harakteristik i rasshireniya oblastej primeneniya transteraflopsnyh SBIS semejstva NeroMatrix // Sb. dokladov VI Mezhdunar. nauchno-praktich. konf. «Upravlenie informacionnoj bezopasnost'yu v sovremennom obshchestve». 5–7 iyunya 2018. Moskva.
8. *Gorbunov V.S., Eisymont L.K.* Kompleksnaya metodika testirovaniya proizvoditel'nosti superkomp'yuterov // Vychislitel'nye metody i programmirovanie. 2013. T. 14. S. 115–121. URL = http://num-meth.srcc.msu.ru/zhurnal/tom_2013/pdf/v14r216.pdf.
9. *Eisymont L.K.* Gibridnaya strategiya razvitiya ehlementnoj bazy // Otkrytye sistemy. 2017. № 2. URL = https://www.osp.ru/os/2017/02/13052216/.
10. *Eisymont L.K.* Nastraivaemye specializirovannye SBIS – real'naya osnova sozdaniya budushchih ehkzamasshtabnyh superkomp'yuterov, zarubezhnyj i otechestvennyj opyt // Sb. dokladov VI Mezhdunar. nauchno-praktich. konf. «Upravlenie informacionnoj bezopasnost'yu v sovremennom obshchestve». 5–7 iyunya 2018. Moskva.

УДК 004.272.23

# Программируемые на языках высокого уровня энергоэффективные специализированные СБИС для решения задач информационной безопасности



**С.Г. Елизаров** – к.ф.-м.н., руководитель лаборатории, Физический факультет МГУ им. М.В. Ломоносова
E-mail: elizarov@physics.msu.ru
**Г.А. Лукьянченко** – к.ф.-м.н., науч. сотрудник, Физический факультет МГУ им. М.В. Ломоносова
E-mail: lukyanchenko@physics.msu.ru
**Д.С. Марков** – ст. науч. сотрудник, Физический факультет МГУ им. М.В. Ломоносова
E-mail: markovds@maltsystem.com
**А.М. Монахов** – науч. сотрудник, Физический факультет МГУ им. М.В. Ломоносова
E-mail: monahov.aleksandr@physics.msu.ru
**А.Д. Сизов** – науч. сотрудник, Физический факультет МГУ им. М.В. Ломоносова
E-mail: anatoliy.sizov@gmail.com
**В.А. Роганов** – ст. науч. сотрудник, Институт механики МГУ им. М.В. Ломоносова
E-mail: radug-a@ya.ru

Рассмотрен и обоснован набор требований к программируемым на языках высокого уровня специализированным СБИС, предназначенным для решения задач информационной безопасности с высокой энергоэффективностью. Изложен практический опыт коллектива авторов по разработке и созданию такой СБИС с использованием современных технологий. Приведены данные по производительности и энергоэффективности разработанных и разрабатываемых СБИС, достигаемых на задачах информационной безопасности. Полученные результаты сопоставлены с характеристиками коммерчески доступных универсальных процессоров, графических ускорителей и программируемых логических интегральных схем (ПЛИС) на задачах хеширования и потокового шифрования. Сделан вывод о перспективности предложенного авторами подхода к созданию программируемых специализированных СБИС, предназначенных для решения задач информационной безопасности.

***Ключевые слова:*** *RISC, SIMD, информационная безопасность, программируемые СБИС, сопроцессор, ГОСТ Р 34.12-2015, AES128, Blowfish, many-core, ASIC.*

In this article we define and evaluate a set of requirements for custom energy-efficient programmable ASIC intended to be used in digital security applications. We discuss our practical experience of development and implementation of such ASICs using modern semiconductor technology. We present data on cryptography performance and energy efficiency of our ASICs which are already created and projected characteristics for future chips. Performance and energy efficiency values of our ASICs for hashing and stream encryption/decryption algorithms are compared against values for commercially available general purpose CPUs, GPUs and FPGAs. We conclude that our approach with development of custom programmable ASICs for cryptographic applications is viable and have several advantages over general purpose computing systems.

***Keywords:*** *RISC, SIMD, Information Security, programming ASIC, coprocessor, GOST R 34.12-2015, AES128, Blowfish, many-core, ASIC.*

**DOI: 10.18127/j20729472-201803-07**

Информационная безопасность сегодня является ключевой задачей обеспечения устойчивого функционирования и развития инфраструктур, основанных на цифровых ресурсах. Решение этой задачи требует не только анализа имеющихся и перспективных информационных угроз, развития математического и алгоритмического аппарата, создания современных программных комплексов, но и достижения на каждом временном отрезке уровня производительности на целевых задачах, достаточного для успешной борьбы с мировой киберпреступностью и для сохранения паритета со специальными службами других стран.

Указанные выше соображения объясняют необходимость разработки специализированных аппаратных решений, обеспечивающих качественный скачок в производительности при решении задач информационной безопасности относительно коммерчески доступных вычислительных систем общего назначения. Необходимость исключительно быстрой реакции на новые информационные угрозы требует сохранения в специализированных устройствах достаточной гибкости и возможности быстрого перепрограммирования.

Цель работы – рассмотреть и обосновать набор требований к программируемым на языках высокого уровня специализированным СБИС, предназначенным для решения задач информационной безопасности с высокой энергоэффективностью.

Спектр задач информационной безопасности чрезвычайно широк. В данной работе будем рассматривать задачи, связанные с отечественными и зарубежными алгоритмами симметричного шифрования, а также с решением прямых и обратных задач хеширования, применяемых в технологии блокчейн и ее производных. Все эти задачи объединяет исключительная вычислительная сложность, выражающаяся в числе элементарных операций, выполняемых с каждым блоком обрабатываемых данных, и характере этих операций, так или иначе сводящихся к сдвигам, наложениям масок, операциям сложения, умножения и табличных замен.

Отметим, что большинство рассматриваемых задач информационной безопасности относятся к группе математических задач с идеальным или близким к идеальному распараллеливанием по данным в части наиболее вычислительно сложных математических процедур.

Производительность и энергопотребление микросхемы в настоящей работе измеряется на наборе базовых преобразований. Набор таких преобразований и построенные на его основе эталонные задачи определяются представлением о современном состоянии отрасли, составленном авторами в результате изучения открытых источников.

В настоящее время потребляемая электроэнергия – это одна из основных расходных статей центров по обработке данных (ЦОД) [1]. Максимальная плотность компоновки элементов современных заказных вычислительных систем [2] также определяется тепловыделением, а не размером компонентов. Таким образом, характеристикой «качества» специализированной вычислительной системы сегодня является не производительность, а энергоэффективность, то есть производительность на ватт потребляемой мощности, измеренная на эталонной задаче. Именно эту характеристику будем использовать далее в нашей работе.

С точки зрения критерия энергоэффективности, очень узким, но весьма показательным классом задач информационной безопасности являются переборные задачи, решаемые при майнинге криптовалют. Например, это биткоин, где за последние семь лет за счет перехода от процессоров общего назначения к специализированным интегральным микросхемам (далее ASIC – Application-Specific Integrated Circuit) производительность возросла на три порядка, а энергоэффективность – на пять порядков [3]. Отметим, что в итоге индустрия информационной безопасности получила феноменально энергоэффективную микросхему типа ASIC, но при этом совершенно негибкую и непрограммируемую.

Полной противоположностью ASIC для майнинга являются процессоры общего назначения, выпускаемые, например, компанией Intel. Они способны обеспечивать высокую для универсальных вычислительных ядер производительность на поток на задачах информационной безопасности при небольшом (от единиц до десятков) числе одновременно исполняемых потоков. Универсальность и развитые средства программирования определяют исключительную гибкость и популярность этих процессоров. Обратной стороной универсальности является невысокая полная производительность процессора в целом, обусловленная низким уровнем параллелизма из-за большого размера вычислительного ядра [4].

Поиск коммерчески доступных решений, сочетающих энергоэффективность ASIC и гибкость процессоров общего назначения, осуществлялся ранее и производится сейчас. В первую очередь, разработчиков заинтересовали программируемые логические интегральные схемы (далее ПЛИС), которые сочетают возможность реализации на уровне схемотехники заданного алгоритма с возможностью быстрого перепрограммирования [5]. Энергоэффективность ПЛИС на задачах информационной безопасности за счет их производства по наиболее совершенным технологическим процессам часто оказывается сравнимой с ASIC, изготовленными по доступным и экономически оправданным проектным нормам [6]. Однако из-за аппаратной избыточности, насыщенности периферией и относительной узости рынка цена наиболее производительных ПЛИС настолько высока [7], что их массовое применение вряд ли оправдано.

Графические ускорители (далее GPU – Graphics Processing Unit) изначально проектировались для обработки и визуализации сцен в компьютерных играх и по архитектуре были ближе к ASIC с множеством однотипных конвейеров, чем к универсальным процессорам. Однако по мере развития GPU превратились в массово-параллельные программируемые вычислители с тысячами исполняемых одновременно аппаратных потоков и развитыми средствами программирования. Различные факторы, в том числе взрывной рост блокчейн-технологий [8], обусловили развитую поддержку в GPU элементарных логических и математических операций, характерных для задач информационной безопасности. Огромные объемы производства и ориентация на массовый рынок развлечений определили достаточно низкую цену GPU для конечного пользователя. Таким образом, индустрия информационной безопасности получила недорогой и очень производительный чип.

Можно предположить, что пользователь хотел бы получить специализированную микросхему вычислителя (далее СБИС), сочетающую гибкость и программируемость процессора общего назначения и энергоэффективность ASIC с жестко заданной логикой работы, но это невозможно. Попробуем снизить требования и предположить возможность создания специализированной СБИС, сочетающей гибкость GPU и энергоэффективность ПЛИС на заданном классе задач. В настоящей статье покажем, что такая задача может быть решена на рассматриваемом множестве задач информационной безопасности при сохранении требования программируемости СБИС.

Для достижения высокой энергоэффективности на целевых задачах при разработке специализированной программируемой СБИС необходимо учитывать следующее: технические особенности современных маршрутов проектирования; теплофизические и электрофизические возможности СБИС, в том числе так называемый эффект «темного» кремния; нелинейную зависимость сложности оптимизации IP-ядер от их размера, определяющую оптимальный размер вычислительного ядра; достаточную для достижения заданного уровня производительности глубину специализации устройств на кристалле; требуемые пользователями-программистами гибкость и поддержку распространенных языков программирования, например, C/C++; поддержку подходов к параллельному или многопотоковому исполнению программ на множестве однотипных ресурсов или ядер. Рассмотрим далее эти соображения более подробно.

Снижение линейного размера транзистора приводит к квадратичному росту числа элементарных вентилей на кристалле той же площади. Однако соблюдавшийся до середины 2000-х годов закон Денарда квадратичного уменьшения рабочего напряжения транзистора перестал выполняться. В связи с этим потребляемая кристаллом той же площади мощность стала расти. Этот эффект необходимо учитывать на технологических нормах 28 нм и более современных [9]. Расчетное энергопотребление современной СБИС при средней частоте переключения транзисторов 50% на один-два порядка превышает TDP (требования по теплоотводу). Это значит, что существенная часть кристалла во время работы должна бездействовать, чтобы не превысить TDP. Эта часть кристалла получила название «темный» кремний.

«Темный» кремний – это не только не используемые в настоящий момент части блоков памяти и автоматически выключенные на определенных тактах с помощью механизма clock gating функциональные элементы, но и, в первую очередь, крупные специализированные узлы процессора, ненужные «именно сейчас» и законсервированные путем снижения тактовой частоты, понижения напряжения питания или переведенные в режим «сна». Концепция «темного» кремния позволяет размещать на кристалле процессора заведомо большее число вычислительных блоков, чем может быть задействовано одновременно в рамках заданного TDP, и консервировать те из них, которые на данный момент не нужны или не могут быть задействованы по энергетическим соображениям.

В данной работе рассматриваются «manycore» архитектуры [10] (массово-многоядерные), способные предоставить задаче пользователя сотни или тысячи вычислительных ядер. Для загрузки такого объема ресурсов расчетные потоки должны быть порождены на уровне пользовательского кода. Если при этом учитывать концепцию «темного» кремния, то для рассматриваемого типа специализированных СБИС при ряде описанных ниже ограничений на одном вычислительном элементе целесообразно исполнять не более одного потока.

Первые 8-битные процессоры были предельно компактными – Intel 8080 содержал всего 4800 транзисторов и имел производительность на уровне 0,1 MIPS/МГц. Современный Intel Core i7 содержит несколько миллиардов транзисторов и имеет производительность лишь порядка 10 MIPS/МГц.

Одна из причин снижения эффективности использования транзистора в том, что собственно вычисления перестали быть основной задачей процессора общего назначения. Поддержка вычислений, в первую очередь в части работы с памятью, – вот на что сегодня тратятся миллионы транзисторов. В специализированных программируемых СБИС предлагается вернуться к компактным вычислительным ядрам, заметная часть которых по числу задействованных транзисторов выполняет целевую вычислительную задачу.

Требования к задачам меняются быстрее, чем разрабатываются СБИС, а цена СБИС определяется объемом выпуска и широтой спектра применения, поэтому программируемость остается обязательным требованиям практически для всех целевых задач. Исключения составляют СБИС для очень узких, но востребованных задач, например, майнинга криптовалют. Если же класс задач включает несколько алгоритмов и невозможно обойтись одной непрограммируемой микросхемой, а разработка целого ряда

СБИС под все возможные алгоритмы не оправдана ни экономически, ни технически, пользователи возвращаются к вычислителям с универсальной архитектурой, CPU/GPU или ПЛИС.

Сегодня альтернативой этому подходу является специализация программируемой СБИС на уровне вычислительных блоков на кристалле. Такой подход повышения энергоэффективности широко используется при разработке сетевых процессоров, графических карт, процессоров для мобильных устройств, других специализированных вычислителей, каждый из которых можно рассматривать как спецвычислитель для задачи обработки сетевых пакетов, рендеринга сцен в компьютерных играх или воспроизведения мобильного контента. Авторы считают, что именно специализация блоков внутри программируемой СБИС позволит создать чип для задач информационной безопасности с высокой энергоэффективностью.

## Результаты исследования

Базовую часть архитектуры рассматриваемой специализированной СБИС составляют десятки или сотни в зависимости от площади кристалла компактных универсальных вычислительных ядер, объединенных внутрикристальной сетью собственной разработки. Иерархия универсальных ядер включает три уровня: управляющее ядро (УЯ), коммуникационные ядра (КЯ) и доступные для задач пользователя вычислительные ядра (ВЯ). К последним подключены программируемые ускорители, исключительно эффективно выполняющие целевые задачи информационной безопасности, при чем каждый ускоритель может содержать от 8 до 128 однотипных процессорных элементов (ПЭ) (рис. 1). Все универсальные ядра и ПЭ имеют собственную локальную память данных. Все универсальные ядра непосредственно адресуют общую внешнюю память СБИС. Арбитраж доступа к внешней общей памяти обеспечивается аппаратным контроллером общей памяти (КОП). Описанная архитектура получила название MALT – Manycore Architecture with Lightweight Threads, что в переводе означает «многоядерная архитектура с поддержкой легких потоков» [11, 12].

Универсальные вычислительные ядра СБИС построены на базе процессорного ядра общего назначения [13] – целочисленного 32-разрядного RISC-процессора, совместимого на уровне машинного кода с архитектурой MicroBlaze [14]. Исполняемая программа может частично храниться в ПЗУ, либо подгружаться с промежуточным хранением инструкций в кэше команд. Данные исполняемой программы могут быть загружены прямо в локальную память ядра или запрошены из общей памяти. Возможность работы с общей памятью позволяет исполнять на ВЯ практически любой код на C/C++. Благодаря низколатентному доступу к локальной памяти данных достигается эффективная работа ядер на вычислительно сложных фрагментах задач информационной безопасности, оперирующих с небольшими объемами данных.

Рассматриваемые задачи информационной безопасности, как указано во введении к настоящей статье, требуют многократного выполнения набора вычислительно сложных операций над различными блоками данных. Такие вычисления эффективно исполняются на программируемых ускорителях, построенных из ПЭ с организацией управления по принципу SIMD (single instruction, multiple data – одиночный поток команд, множественный поток данных). Это позволяет выделить из общей схемы программируемого ускорителя контроллер управления потоком команд (КП). Программируемый ускоритель в этом случае функционально сводится к массиву арифметико-логических устройств (АЛУ) и блоков локальной памяти (ЛП), объединенных общими шинами и КП, как показано на рис. 2. Командное слово от КП попадает на первый ПЭ в цепочке, из которого оно без изменений переходит по конвейеру в следующий (сверху вниз).

Кроме вертикальных связей для передачи команд управления ПЭ объединены в горизонтальные цепочки через блок локальной памяти. Каждый ПЭ имеет два порта доступа к локальной памяти – «слева» и «справа», соседний элемент имеет точно такие же два порта, и между этими элементами один блок

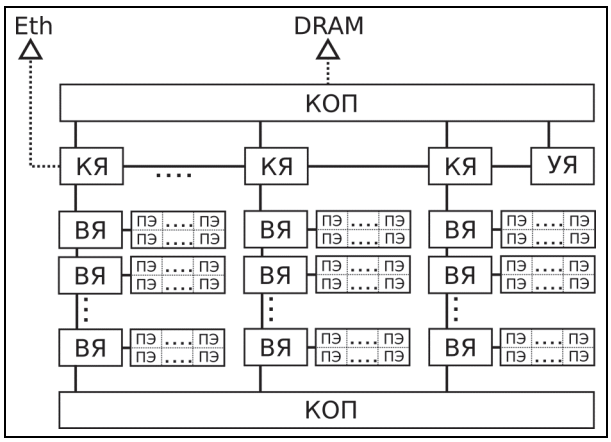


**Рис. 1.** Архитектура специализированной СБИС

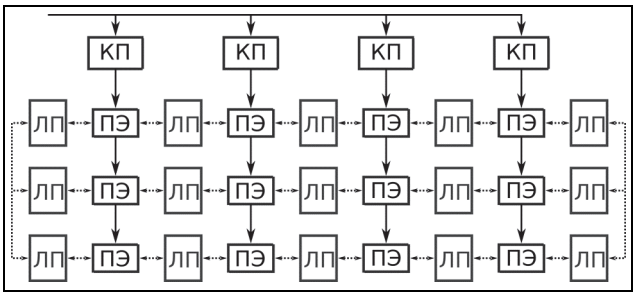


**Рис. 2.** Схема программируемого ускорителя

памяти является общим. Такая организация позволяет использовать строку из ПЭ как конвейер, каждая ступень которого выполняет тот или иной целевой алгоритм.

ПЭ имеет в своем составе регистровый файл, дерево АЛУ. Регистровый файл состоит из 32 регистров разрядностью 32 бита с возможностью чтения из четырех портов (включая два буферных регистра) и записи в два порта за такт. Дерево АЛУ представляет собой три независимых АЛУ, соединенных в древовидную структуру. Каждое АЛУ выполняет типовые операции add, sub, and, or, xor и др., а также имеет два независимых порта доступа к памяти.

Разработка программируемого ускорителя происходила поэтапно.

На первом этапе был проанализирован целевой класс задач, выделены часто встречающиеся последовательности элементарных логических и математических операций, создан ПЭ, позволяющий выполнять выделенные выше последовательности операций за возможно меньшее число тактов, и оценена энергоэффективность предложенного ПЭ.

На втором этапе был разработан эмулятор ПЭ, поддерживающего строго последовательное исполнение команд. Затем был предложен ассемблер для ПЭ, на котором были реализованы базовые алгоритмы.

На третьем этапе было разработано RTL-описание ПЭ, проведены синтез и оптимизация для технологического процесса TSMC 28 нм, оценена энергоэффективность по результатам симуляции с задержками, выявлены наиболее энергозатратные команды, внесены соответствующие изменения в RTL-описание и ассемблер ПЭ.

На четвертом этапе проведен выбор оптимальных с точки зрения скорости работы и энергетики блоков синтезированной памяти, а также разработан контроллер управления с поддержкой циклов и вызовов функций (КП на рис. 2), что позволило существенно сэкономить объем памяти команд ускорителя.

Далее следовала разработка компилятора Си для программируемого ускорителя, эмулятора и отладчика. Произведена параметризация RTL-представления программируемого ускорителя для сборки в различных базисах.

На заключительном этапе была произведена интеграция программной части и RTL-описания SIMD с универсальными компонентами специализированной СБИС.

Проверка основных решений, рассмотренных ранее и заложенных в семейство специализированных СБИС, осуществлялась на тестовой СБИС, далее 9Mb96G (индекс микросхемы определяется числом универсальных ядер, числом и типом ПЭ) (рис. 3).

В 2017 г. с фабрики TSMC (Тайвань) были получены образцы микропроцессора 9Mb96G (рис. 4) [15]. Тестовая партия изготовлена по технологическому процессу 28 нм. Образцы 9Mb96G успешно прошли входные испытания, показали на полной нагрузке соответствие проектируемым характеристикам.

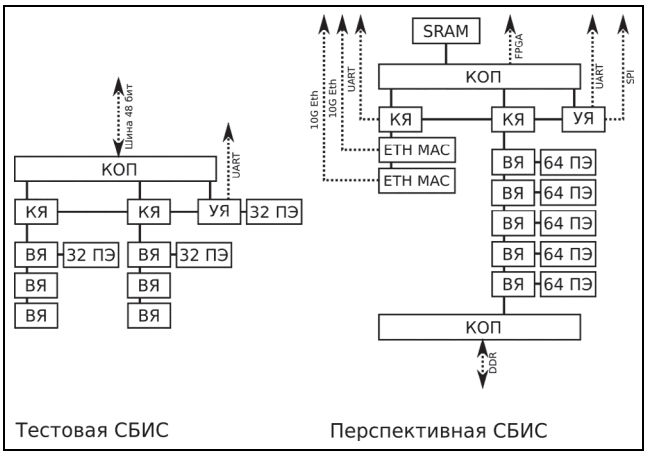


**Рис. 3.** Схема тестовой и перспективной СБИС

В табл. 1 представлены основные характеристики 9Mb96G. Отметим невысокое для СБИС такой площади энергопотребление, обусловленное основными требованиями к тестовому кристаллу: проверкой работоспособности и надежностью множества отдельных схемотехнических решений, заложенных с 9Mb96G, а не максимальной плотностью вычислений. Корпус LQFP208 – это стандартный заводской корпус под разварку проволокой (wire bond), удобный для быстрого прототипирования и тестов, но редко применяемый для серийных изделий.

Внешние интерфейсы 9Mb96G: последова-

тельный порт RS232TTL – 230 кбит/с; интерфейс к ПЛИС CMOS18 x16 – 1 Гбит/с.

Основной обмен с внешними источниками и потребителями данных, в том числе с внешней памятью, осуществляется через интерфейсную ПЛИС, подключенную через параллельную двунаправленную шину. Такой выбор интерфейса обусловлен его исключительной простотой и гибкостью, во время отладки решения на 9Mb96G «на стороне ПЛИС» можно создавать различные тестовые окружения, в том числе замещающие КОП, что практически гарантирует запуск 9Mb96G. Перенос внешнего интерфейса в ПЛИС также позволяет тестировать 9Mb96G с внешней ОЗУ типа DDR и высокоскоростными интерфейсами типа PCIe без сложной и дорогостоящей интеграции соответствующих IP-блоков в проект 9Mb96G.

В табл. 2 приведены показатели производительности и энергоэффективности 9Mb96G на паре базовых алгоритмов, входящих в рассматриваемый класс задач информационной безопасности. Отметим, что выбраны используемые и хорошо описанные в открытых источниках преобразования.

Подходы к проектированию специализированных СБИС, изложенные в настоящей статье и проверенные на тестовой СБИС, применяются авторами сегодня для создания семейства специализированных СБИС для решения задач информационной безопасности с высокой энергоэффективностью [16]. Рассмотрим далее перспективную СБИС, индекс 8Mb160P (рис. 3), разрабатываемую в настоящий момент.

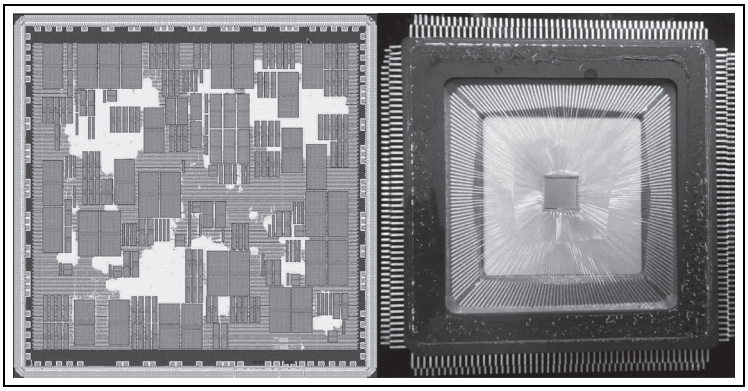

**Рис. 4.** Внешний вид тестовой СБИС без крышки

**Таблица 1.** ***Основные характеристики тестовой и перспективной СБИС***

| Характеристика | 9Mb96G | 8Mb160P |
|---|---|---|
| Технологический процесс, нм | 28 | 28 |
| Площадь кристалла, мм$^2$ | 12 | 12 |
| Тепловыделение (TDP), Вт | 1 | 5 |
| Напряжение питания ядра, В | 0,9 | 0,9 |
| Напряжение питания IO, В | 1,8 | 1,8 |
| Тип корпуса | LQFP208 | FTG256 |
| Рабочая частота, МГц | 800 | 1000 |
| Число универсальных ядер, шт. | 7 | 8 |
| Число специализированных ядер, шт. | 96 | 160 |

**Таблица 2.** ***Производительность тестовой и перспективной СБИС***

| Параметр | 9Mb96G | 8Mb160P |
|---|---|---|
| Производительность на преобразовании типа AES128, млн хеш/с | 100 | 1500 |
| Энергоэффективность на преобразовании типа AES128, млн хеш/Дж | 160 | 350 |
| Производительность на преобразовании типа Blowfish, млн хеш/с | 0,1 | 4,3 |
| Энергоэффективность на преобразовании типа Blowfish, млн хеш/Дж | 0,4 | 1,4 |

8Mb160P построена на базе подходов, описанных выше, не является образцом или демонстратором и предназначена для применения в серийных изделиях. 8Mb160P построена на базе оптимизированного ПЭ, отличающегося более высокой энергоэффективностью и плотностью вычислений на мм$^2$ относительно ПЭ, применяемого в тестовой СБИС. 8Mb160P снабжена необходимыми интерфейсами для работы с периферией, является полноценной системой на кристалле (СнК), не требует для работы внешнего ПЛИС, хотя и допускает применение ПЛИС для расширения коммуникационных возможностей.

Коммуникации с внешними пользователями осуществляются в 8Mb160P через ряд высокоскоростных интерфейсов: интерфейс к ПЛИС для поддержки нестандартной периферии и интерфейсов и два 10G Ethernet контроллера в первую очередь для задач потокового шифрования. 8Mb160P содержит контроллер внешней динамической памяти типа DDR. Дополнительно на кристалле 8Mb160P расположен 1 МБ встроенной доступной всем универсальным ядрам общей статической памяти, что делает возможным работу широкого спектра целевых задач без внешней ОЗУ.

8Mb160P содержит два интерфейса UART, один из которых является периферийным устройством для

управляющего ядра и реализует стандартный консольный доступ, второй доступен всем универсальным ядрам, что значительно расширяет возможности отладки параллельных программ. К УЯ как периферийное устройство подключен SPI-контроллер для загрузки исполняемых программ из внешней SPI Flash.

В настоящий момент перспективная СБИС 8Mb160P находится в стадии разработки топологии кристалла. Получены оценки по результатам синтеза и симуляции, ожидаемые характеристики перспективной СБИС представлены в табл. 1 и 2.

В табл. 1 представлены основные характеристики 8Mb160P. Отметим, что выбор корпуса типа BGA и монтаж кристалла flip-chip обеспечивает лучший теплоотвод и меньшие габаритные размеры чипа.

Внешние интерфейсы 8Mb160P: последовательный порт RS232TTL x2 – 11 кбит/с; интерфейс к внешней ОЗУ LPDDR1 – 40 Гбит/с; интерфейс к внешней flash SPI – 25 Мбит/с; интерфейс к ПЛИС LVDS18 x16 – 10 Гбит/с; интерфейс Ethernet 10G x2.

Перспективная СБИС снабжена гибридным CMOS18/LVDS18 интерфейсом к ПЛИС, который, с одной стороны, предлагает широкие отладочные возможности, с другой стороны, так как построен на базе низковольтных дифференциальных пар (англ. low-voltage differential signaling или LVDS), позволяет организовывать помехоустойчивые высокоскоростные (до 10 Гбит/с) двунаправленные интерфейсы различного назначения.

В табл. 2 приведены показатели производительности и энергоэффективности 8Mb160P. Отметим, что все указанные характеристики выше относительно тестовой СБИС.

## Обсуждение результатов исследования

В соответствии с соображениями, описанными во введении, проведем сравнение с наиболее вероятными коммерчески доступными конкурентами: высокопроизводительной ПЛИС Xilinx Kintex-410T и распространенным GPU NVidia GTX1080, изготовленными по аналогичному или лучшим технологическим процессам.

Проведем сравнение на паре вычислительно-сложных преобразований, результаты которых приведены в табл. 3. Отметим, что все реализации указанных алгоритмов для GPU взяты из оптимизированной библиотеки HashCat [17], а реализации для ПЛИС построены по принципу конвейера, с помощью которого достигается максимальная производительность и энергоэффективность.

Из табл. 3 видно, что полная производительность перспективной СБИС на рассмотренных задачах информационной безопасности превышает производительность выбранных GPU или ПЛИС. При переходе от производительности к энергоэффективности отличие становится еще более заметным: выигрыш составляет от одного до двух порядков.

Проведем сравнение перспективной СБИС с процессорами общего назначения и GPU на задачах потокового шифрования и убедимся, что гибкость решения на 8Mb160P достаточна для реализации всех аспектов рассматриваемых задач информационной безопасности. Так как многие популярные криптоалгоритмы реализованы в коммерчески доступных CPU аппаратно, для корректного сравнения кроме CPU и перспективной СБИС возьмем GPU, а в набор тестов включим пару отечественных шифров, заведомо не поддержанных аппаратно в зарубежных решениях.

В табл. 4 приведены значения пропускной способности 8Mb160P по сравнению с CPU и GPU на нескольких популярных алгоритмах шифрования. Блочные шифры используются в режиме потокового шифрования. Без скобок приведены теоретические значения, определяемые вычислительной производительностью процессора. В круглых скобках – практически достижимые значения, учитывающие пропускную способность внешних интерфейсов. Для CPU/GPU использовались наиболее эффективные реализации, применяющие все доступные аппаратные ресурсы [18−21].

**Таблица 3.** ***Сравнение производительности и энергоэффективности СБИС, GPU, ПЛИС***

| Параметр | 8Mb160P | GTX1080 | 410T |
|---|---|---|---|
| Производительность на AES128, млн преобр./с | 1500 | 2200 | 6500 |
| Энергоэффективность на AES128, млн преобр./Дж | 450 | 12 | 200 |
| Производительность на Blowfish, млн преобр./с | 4,3 | 0,9 | 6,2 |
| Энергоэффективность на Blowfish, млн преобр./Дж | 1,4 | 0,005 | 0,2 |

Из табл. 4 видно, что вычислительная производительность перспективной СБИС для всех алгоритмов превышает характеристики CPU/GPU.

В настоящий момент внешние интерфейсы не позволяют использовать вычислительную производительность перспек-

**Таблица 4.** ***Сравнение пропускной способности СБИС, CPU, GPU на различных алгоритмах шифрования***

| Алгоритм | 8Mb160P, Гбит/с | Процессоры общего назначения | | Видеокарты | |
|---|---|---|---|---|---|
| | | CPU | Гбит/с | GPU | Гбит/с |
| ГОСТ Р 34.12-2015 «Кузнечик» | 10(10) | Intel Core i7-3537U | 0,12 [18] | NVIDIA GT750M | 2,2 [18] |
| ГОСТ 28147-89 «Магма» | 120(10) | Intel Xeon E5 2697 v3 | 22 [19] | NVIDIA GTX750 | 21,2 [19] |
| AES 128 | 200(10) | Семейство Intel Xeon E5v2 | 50 [20] | NVIDIA GTX1080 | 280(128) [21] |

тивной СБИС полностью, но открывают возможности создания встраиваемых энергоэффективных специализированных решений с производительностью на уровне CPU/GPU и потребляемой мощностью на уровне 1 Вт, а также, используя кристаллы большей площади, сверхвысокопроизводительных потоковых шифраторов со скоростью обработки данных 100 Гбит/с и выше.

- Предложенный авторами и описанный в настоящей работе подход к созданию специализированных энергоэффективных процессоров, основанный на архитектуре MALT (Manycore Architecture with Lightweight Threads), позволяет разрабатывать решения, демонстрирующие на задачах информационной безопасности производительность на ватт на уровне, близком к заказным системам на ПЛИС, и сравнимые по гибкости и программируемости с универсальными x86/GPU/ARM вычислительными системами. На ряде целевых задач разрабатываемые решения на порядок превосходят универсальные вычислительные системы по показателю производительность на ватт.

## Литература


1. *Пирогова Л.А.*, *Грекул В.И.*, *Поклонов Б.Е.* Оценка совокупной стоимости владения центром обработки данных // Бизнес-информатика. 2016. С. 32−40. URL = https://bijournal.hse.ru/2016–2 (36)/186020074.html.
2. *Kurokawa M.* The k computer: 10 peta-flops supercomputer // In the 10th International Conference on Optical Internet (COIN2012). 1−1. May 2012.
3. *O'Dwyer K.J.*, *Malone D.* Bitcoin mining and its energy footprint // In 25th IET Irish Signals Systems Conference 2014 and 2014 China-Ireland International Conference on Information and Communications Technologies (ISSC 2014/CIICT 2014). 06 2014. P. 280−285.
4. Modern Microprocessors: A 90-Minute Guide! URL = http://www.lighterra.com/papers/modernmicroprocessors/.
5. *Тарасов И.Е.* Разработка цифровых устройств на основе ПЛИС Xilinx с применением языка VHDL. Современная электроника. М.: Горячая линия–Телеком. 2005. URL = https://books.google.ru/books?id=QU69AQAACAAJ.
6. *Francisco Rodriguez-Henriquez*, *Saqib N.A.*, *Arturo Daz Prez, Cetin Kaya Koc* Cryptographic Algorithms on Reconfigurable Hardware. Springer Publishing Company, Inc. 1st edition. 2010.
7. Xilinx Virtex UltraScale+ FPGA VCU118 Evaluation Kit. URL = https://www.xilinx.com/products/boards-and-kits/vcu118.html.
8. A Next-Generation Smart Contract and Decentralized Application Platform. URL = https://github.com/ethereum/wiki/wiki/White-Paper.
9. *Eitan N. Shauly* Cmos leakage and power reduction in transistors and circuits: process and layout considerations // Journal of Low Power Electronics and Applications. 2012. № 2(1). P. 1−29. URL = http://www.mdpi.com/2079-9268/2/1/1.
10. *B.D. de Dinechin*, *Ayrignac R.*, *Beaucamps P.E.*, *Couvert P.*, *Ganne B.*, *P.G. de Massas*, *Jacquet F.*, *Jones S.*, *Chaisemartin N.M.*, *Riss F.*, *Strudel T.* A clustered manycore processor architecture for embedded and accelerated applications // In 2013 IEEE High Performance Extreme Computing Conference (HPEC). 09 2013. P. 1−6.
11. Manycore Architecture with Lightweight Threads. URL = https://maltsystem.ru/.
12. Пат. РФ 2018.05.21. Рег. № 2018118432/08(028834). Вычислительный модуль для многопотоковой обработки цифровых данных и способ обработки с использованием данного модуля / *Елизаров С.Г.*, *Лукьянченко Г.А.*, *Монахов А.М.*, *Сизов А.Д.*, *Советов П.Н.*
13. *Kranenburg T.*, *R. Van Leuken* Mb-lite: a robust, light-weight soft-core implementation of the microblaze architecture // In 2010 Design, Automation Test in Europe Conference Exhibition (DATE 2010). 03 2010. P. 997−1000.
14. MicroBlaze Processor Reference Guide. URL = https://www.xilinx.com/support/documentation/sw_manuals/mb_ref_guide.pdf.
15. С фабрики TSMC получены первые образцы микропроцессора MALT-C 9Mb96G. URL = https://maltsystem.ru/ru/news/135-fabget.
16. Процессоры MALT-C. URL = https://maltsystem.ru/ru/product/malt-processors#malt-c.
17. Hashcat. URL = https://hashcat.net/hashcat/.
18. *Удальцов В.А.*, *Павлов В.Э.* Увеличение скорости работы алгоритма шифрования «КУЗНЕЧИК» с использованием технологии CUDA // Теория. Практика. Инновации. 2017. № 4(16). С. 5−11.
19. *Кролевецкий А.* Производительность ГОСТ-шифрования на x86- и GPU-процессорах // Storage News. 2014. 4/60. URL = http://www.storagenews.ru/60/Code_Sec_cripto_GOST_60.pdf.
20. *John M.* AES-GCM Encryption Performance on Intel® Xeon® E5 v3 Processors. URL = https://software.intel.com/en-us/articles/aes-gcm-encryption-performance-on-intel-xeon-e5-v3-processors.
21. *Abdelrahman A.A.*, *Fouad M.M.*, *Dahshan H.*, *Mousa A.M.* High performance CUDA AES implementation: A quantitative performance analysis approach // Computing Conference. London. 2017. P. 1077−1085. URL = http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=8252225&isnumber=8252073.


**Поступила 3 августа 2018 г.**

# Programmable in high-level languages energy-efficient specialized VLSI for solving information security problems



**S.G. Elizarov** – Ph.D.(Phys.-Math.), Head of Laboratory, Physics Faculty of Lomonosov Moscow State University
E-mail: elizarov@physics.msu.ru
**G.A. Lukyanchenko** – Ph.D.(Phys.-Math.), Research Scientist, Physics Faculty of Lomonosov Moscow State University
E-mail: lukyanchenko@physics.msu.ru
**D.S. Markov** – Senior Research Scientist, Physics Faculty of Lomonosov Moscow State University
E-mail: markovds@maltsystem.com
**A.M. Monakhov** – Research Scientist, Physics Faculty of Lomonosov Moscow State University
E-mail: monahov.aleksandr@physics.msu.ru
**A.D. Sizov** – Research Scientist, Physics Faculty of Lomonosov Moscow State University
E-mail: anatoliy.sizov@gmail.com
**V.A. Roganov** – Senior Research Scientist, Institute of Mechanics of Lomonosov Moscow State University
E-mail: radug-a@ya.ru

In this article we define and evaluate a set of requirements for custom energy-efficient programmable ASIC intended to be used in digital security applications. We discuss our practical experience of development and implementation of such ASICs using modern semiconductor technology. We present data on cryptography performance and energy efficiency of our ASICs which are already created and projected characteristics for future chips. Performance and energy efficiency values of our ASICs for hashing and stream encryption/decryption algorithms are compared against values for commercially available general purpose CPUs, GPUs and FPGAs. We conclude that our approach with development of custom programmable ASICs for cryptographic applications is viable and have several advantages over general purpose computing systems. On a number of targets, the solutions being developed are an order of magnitude superior to universal computing systems in terms of performance per watt.

**References**

1. *Pirogova L.A., Grekul V.I., Poklonov B.E.* Ocenka sovokupnoj stoimosti vladeniya centrom obrabotki dannyh // Biznes-informatika. 2016. S. 32–40. URL = https://bijournal.hse.ru/2016–2 (36)/186020074.html.
2. *Kurokawa M.* The k computer: 10 peta-flops supercomputer // In the 10th International Conference on Optical Internet (COIN2012). 1–1. May 2012.
3. *O'Dwyer K.J.*, *Malone D.* Bitcoin mining and its energy footprint // In 25th IET Irish Signals Systems Conference 2014 and 2014 China-Ireland International Conference on Information and Communications Technologies (ISSC 2014/CIICT 2014). 06 2014. P. 280–285.
4. Modern Microprocessors: A 90-Minute Guide! URL = http://www.lighterra.com/papers/modernmicroprocessors/.
5. *Tarasov I.E.* Razrabotka cifrovyh ustrojstv na osnove PLIS Xilinx s primeneniem yazyka VHDL. Sovremennaya ehlektronika. M.: Goryachaya liniya–Telekom. 2005. URL = https://books.google.ru/books?id=QU69AQAACAAJ.
6. *Francisco Rodriguez-Henriquez*, *Saqib N.A.*, *Arturo Daz Prez, Cetin Kaya Koc* Cryptographic Algorithms on Reconfigurable Hardware. Springer Publishing Company, Inc. 1st edition. 2010.
7. Xilinx Virtex UltraScale+ FPGA VCU118 Evaluation Kit. URL = https://www.xilinx.com/products/boards-and-kits/vcu118.html.
8. A Next-Generation Smart Contract and Decentralized Application Platform. URL = https://github.com/ethereum/wiki/wiki/White-Paper.
9. *Eitan N. Shauly* Cmos leakage and power reduction in transistors and circuits: process and layout considerations // Journal of Low Power Electronics and Applications. 2012. № 2(1). P. 1–29. URL = http://www.mdpi.com/2079-9268/2/1/1.
10. *B.D. de Dinechin*, *Ayrignac R.*, *Beaucamps P.E.*, *Couvert P.*, *Ganne B.*, *P.G. de Massas*, *Jacquet F.*, *Jones S.*, *Chaisemartin N.M.*, *Riss F.*, *Strudel T.* A clustered manycore processor architecture for embedded and accelerated applications // In 2013 IEEE High Performance Extreme Computing Conference (HPEC). 09 2013. P. 1–6.
11. Manycore Architecture with Lightweight Threads. URL = https://maltsystem.ru/.
12. Pat. RF 2018.05.21. Reg. № 2018118432/08(028834). Vychislitel'nyj modul' dlya mnogopotokovoj obrabotki cifrovyh dannyh i sposob obrabotki s ispol'zovaniem dannogo modulya / *Elizarov S.G., Lukyanchenko G.A., Monahov A.M., Sizov A.D., Sovetov P.N.*
13. *Kranenburg T.*, *R. Van Leuken* Mb-lite: a robust, light-weight soft-core implementation of the microblaze architecture // In 2010 Design, Automation Test in Europe Conference Exhibition (DATE 2010). 03 2010. P. 997–1000.
14. MicroBlaze Processor Reference Guide. URL = https://www.xilinx.com/support/documentation/sw_manuals/mb_ref_guide.pdf.
15. S fabriki TSMC polucheny pervye obrazcy mikroprocessora MALT-C 9Mb96G. URL = https://maltsystem.ru/ru/news/135-fabget.
16. Processory MALT-C. URL = https://maltsystem.ru/ru/product/malt-processors#malt-c.
17. Hashcat. URL = https://hashcat.net/hashcat/.
18. *Udal'cov V.A., Pavlov V.Eh.* Uvelichenie skorosti raboty algoritma shifrovaniya «KUZNECHIK» s ispol'zovaniem tekhnologii CUDA // Teoriya. Praktika. Innovacii. 2017. № 4(16). S. 5–11.
19. *Kroleveckij A.* Proizvoditel'nost' GOST-shifrovaniya na h86- i GPU-processorah // Storage News. 2014. 4/60. URL = http://www.storagenews.ru/60/Code_Sec_cripto_GOST_60.pdf.
20. *John M.* AES-GCM Encryption Performance on Intel® Xeon® E5 v3 Processors. URL = https://software.intel.com/en-us/articles/aes-gcm-encryption-performance-on-intel-xeon-e5-v3-processors.
21. *Abdelrahman A.A.*, *Fouad M.M.*, *Dahshan H.*, *Mousa A.M.* High performance CUDA AES implementation: A quantitative performance analysis approach // Computing Conference. London. 2017. P. 1077–1085. URL = http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=8252225&isnumber=8252073.

УДК 004.383

# Гетерогенная многопроцессорная система на кристалле с производительностью 512 Gflops



**А.Л. Эйсымонт** – начальник сектора, ЗАО НТЦ «Модуль» (Москва)
E-mail: eisymont@module.ru
**В.М. Черников** – к.т.н., гл. конструктор – начальник отделения, ЗАО НТЦ «Модуль» (Москва)
E-mail: tchern@module.ru
**Ан.В. Черников** – начальник сектора, ЗАО НТЦ «Модуль» (Москва)
E-mail: chernant@module.ru
**Ал.В. Черников** – зам. начальника отделения, ЗАО НТЦ «Модуль» (Москва)
E-mail: achernikov@module.ru
**Д.Е. Косоруков** – начальник отдела, ЗАО НТЦ «Модуль» (Москва)
E-mail: dkos@module.ru
**И.И. Насонов** – начальник сектора, ЗАО НТЦ «Модуль» (Москва)
E-mail: nasonov@module.ru
**А.А. Комлев** – вед. инженер, ЗАО НТЦ «Модуль» (Москва)
E-mail: a.komlev@module.ru

Рассмотрены вопросы реализации энергоэффективной гетерогенной и толерантной к задержкам выполнения операций с памятью системы на кристалле (СнК) с тактовой частотой 1 ГГц, пиковой производительностью 512 Gflops и иерархически организованной внутренней памятью. Отмечено, что СнК содержит 16 векторных ядер NMC4 из семейства NeuroMatrix и пять скалярных ядер Cortex-A5 фирмы ARM.

***Ключевые слова:*** *архитектура NeuroMatrix, многоядерная гетерогенная СнК, векторная архитектура, VLIW, проблема стены памяти, встроенный вычислитель, обработка сигналов, нейронные сети, плавающая точка.*

This article is devoted to questions and methods of implementing energy efficient heterogeneous and tolerant to memory latency system on chip (SoC) operating at 1 GHz frequency, with 512 Gflops peak performance and hierarchically organized internal memory. SoC contains sixteen NeuroMatrix NMC4 processor cores and five ARM Cortex-A5.

***Keywords:*** *NeuroMatrix architecture, multicore heterogeneous SoC, vector architecture, VLIW, memory wall problem, embedded computer, digital signal processing, neural networks, floating-point.*

**DOI: 10.18127/j20729472-201803-08**

Система на кристалле (СнК) NM6408 разрабатывалась для встроенных вычислителей информационно-управляющих систем (ИУС). Современные приложения для встроенных вычислителей, такие как первичная обработка сигналов (преобразования Фурье, Адамара, фильтры и т.д.) и многослойные нейронные сети разного типа, в настоящее время требуют достижения реальной производительности около десятка Tflops над вещественными числами одинарной точности. До недавнего времени встроенные вычислители достигали высокой производительности и энергоэффективности за счет использования в них специализированных процессоров. Сейчас же для экономии затрат на разработку и эксплуатацию, а также из-за быстро меняющихся прикладных задач и алгоритмов, стараются вместо жестко специализированных процессоров создавать конкурирующие с ними по производительности и энергоэффективности проблемно-ориентированные процессоры с более широкими возможностями для решения различных задач и использования широкого класса алгоритмов из выбранных предметных областей [1, 2].

При разработке СнК NM6408 первоочередными были вопросы организации работы с памятью и общей организации вычислений, соответствующих приложениям для встроенных систем. Исследования профилей работы с памятью для наиболее актуальных приложений, таких как первичная обработка сигналов и нейросети, показывают, что обращения к памяти в этих приложениях обладают либо плохой пространственной локализацией при хорошей временной (обработка сигналов), либо наоборот (нейронные сети) [6, 7]. Таким образом, обычное применение кэш-памятей с малыми задержками выполнения обращений к ним для таких задач оказывается практически бесполезным, поскольку применение кэш-памятей предполагает одновременно хорошую и пространственную, и временну́ю локализацию обращений к памяти. Известно, что в таких случаях при создании процессоров для суперкомпьютеров обычным решением является использование архитектуры, обеспечивающей высокий темп выполнения обращений к памяти.

Такой способностью обеспечивать мощный поток обращений к памяти обладают архитектуры векторных и мультитредовых процессоров. Такие архитектуры называют толерантными по производительности к большим задержкам обращений к памяти. Еще известно, что при наличии в программе высокого параллелизма по данным такие архитектуры не уступают архитектурам, использующим кэш-памяти, на задачах с одновременно хорошей пространственной и временной локализацией, а такие приложения также встречаются и для встроенных систем.

В ядрах NMC4 СнК NM6408 выбрана классическая архитектура векторного процессора с конвейерными функциональными устройствами, векторными регистрами и внутренней памятью с большим расслоением. Такая архитектура была впервые применена в американском суперкомпьютере Cray-1 и в модифицированном виде используется до сих пор, особенно успешно в векторных микропроцессорах SX-ACE японской фирмы NEC [8], а также при построении специальных мультитредово-векторных суперкомпьютеров CB-класса (Capacity Bandwidth, с большой памятью с высокой пропускной способностью) [4].

Кроме наиболее требовательных по производительности нейросетевых задач и задач обработки сигналов, перед современными встроенными системами также стоят и вычислительные задачи, лучше решаемые на обычных универсальных CPU. Для таких задач, а также для задач управления и операционной системы в СнК NM6408 были включены скалярные RISC-ядра с популярной архитектурой ARM (Cortex A5).

Ц е л ь  р а б о т ы – рассмотреть вопросы реализации энергоэффективной гетерогенной и толерантной к задержкам выполнения операций с памятью СнК с тактовой частотой 1 ГГц, пиковой производительностью 512 Gflops и иерархически организованной внутренней памятью.

## Схема системы на кристалле

На рис. 1 приведена упрощенная схема СнК NM6408, которая содержит 21 процессорный узел (ПУ), пять интерфейсов с внешней памятью типа DDR3, интерфейс с хост-процессором на базе PCIe2.0 и четыре высокоскоростных интерфейса для связи с внешними процессорными системами. В составе СнК имеются: 16 идентичных ПУ на базе NMC4 (распределены по четырем кластерам PC0…PC3), которые обеспечивают суммарную производительность системы в 512 Gflops; пять ПУ на базе Cortex-A5 (четыре из них входят в кластеры, один – внекластерный, центральный, для общего управления (CCPU)).

*CCPU (Central Cortex Processing Unit)* – центральный управляющий ПУ, содержит: ядро Cortex A5 (640 МГц, кэш-памяти данных и команд L1 по 32 КБ, общая кэш-память L2 – 512 КБ); четыре банка локальной памяти, общий объем 512 КБ; интерфейс с DDR3 памятью по 32-разрядной шине данных (DDR3#4); четыре высокоскоростных коммуникационных порта. CCPU также подключен к главному коммутатору AXI_PC.

*PCi (Processing Cluster)* – процессорный кластер ($i = 0…3$). Каждый кластер содержит: четыре ПУ на базе векторных ядер NMC4 (NMPU*ij*, $j = 0…3$); ПУ на базе ядра Cortex-A5 (CPU*i*). Пиковая производительность кластера 128 Gflops (32-разрядные числа). ПУ NMPU*ij* и ПУ CPU*i* подключены к локальной шине AXI_NMPU*i*.

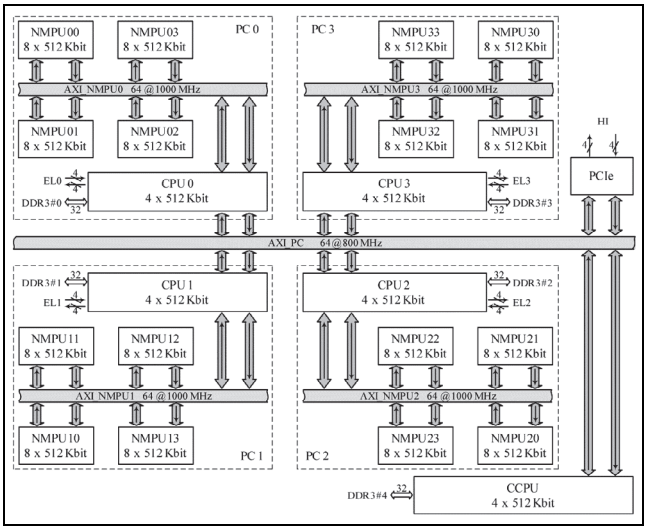


**Рис. 1.** Упрощенная схема СнК NM6408

*CPUi (Cortex Processing Unit)* – кластерный управляющий ПУ, содержит: ядро Cortex-A (800 МГц, L1 кэш-памяти данных и команд, объемом по 32 КБ); четыре банка внутренней памяти, общий объем 256 КБ; интерфейс с DDR3 памятью (DDR3#*i*); четыре коммуникационных порта; четыре коммутатора коммуникационных портов; высокоскоростной интерфейс Eli (External Link), внешний для NM6408 4-проводной дуплексный интерфейс, работающий на частоте 5 ГГц и предназначенный для обмена данными с другими СнК NM6408 в многопроцессорных системах, обеспечивает пропускную способность в 2 ГБ/с в каждую сторону.

*NMPUij (NeuroMatrix Processing Unit)* – векторный вычислительный ПУ ($j = 0…3$) на базе ядра NMC4.

*DDRk* – один из пяти ($k = 0…4$) интерфейсов с внешней памятью типа DDR3-1600 с 32-разрядной шиной данных, обеспечивающей пропускную способность 6,4 ГБ/с. Максимальный объем внешней памяти, подключаемой к одному такому интерфейсу, составляет 1 ГБ.

*HI (Host Interface)* – интерфейс с хост-процессором, реализованный на стандартном 4-канальном дуплексном PCIe2.0 интерфейсе с пропускной способностью 2 ГБ/с в каждую сторону.

Система коммутаторов, соединяющая все ПУ, построена в соответствии с AMBA AXI 3.0 протоколом.

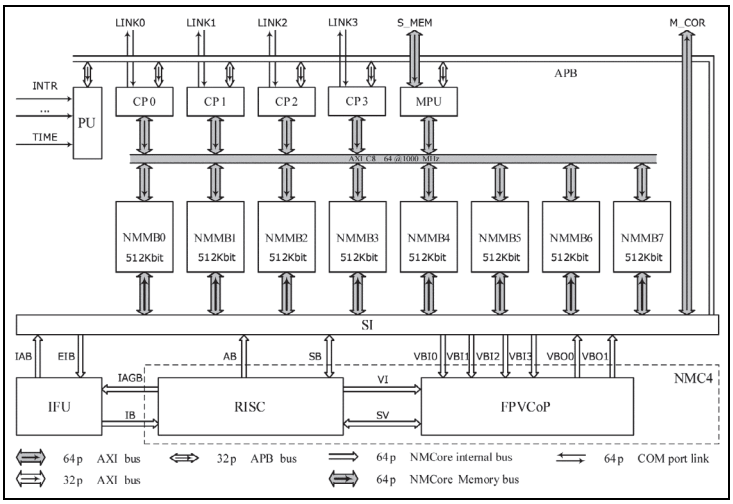


**Рис. 2.** Схема векторного вычислительного ПУ NMPU

На рис. 2 представлена структура векторного вычислительного ПУ NMPU.

*NMC4* – процессорное ядро со скалярным RISC-процессором с VLIW-архитектурой и матрично-векторным сопроцессором арифметики с плавающей точкой (FPVCoP) из семейства NeuroMatrix [9, 10]. Это ядро связано через системный интегратор SI с банками внутренней памяти NMMB0…NMMB7. RISC-процессор дешифрирует скалярные и векторные команды, отслеживает информационные зависимости скалярных команд и часть зависимостей векторных команд. Далее векторные команды передаются в векторный сопроцессор FPVCoP по шине VI, в нем происходят дальнейшие проверки зависимостей и конвейерное выполнение команд.

*IFU (Instruction Fetch Unit)* – блок предвыборки команд, содержит кэш-память команд объемом 8 КБ.

*SI (System Integrator)* – системный интегратор, обеспечивает доступ к банкам внутренней памяти NMMB0…NMMB7, а также к банкам памяти других ПУ через AXI порт M_COR и к регистрам расположенных на периферийной шине APB-блоков. SI содержит адресный генератор для выборки команд RISC-процессора, восемь адресных генераторов для доступа к элементам векторов из любой памяти СнК NM6408, коммутатор адресных шин, коммутатор векторных данных, логические блоки контроля и управления ресурсами.

*NMMB0…NMMB7 (NM Memory Bank)* – восемь банков внутренней памяти, общий объем 512 КБ. Каждый банк имеет два порта доступа, один – со стороны системного интегратора SI, другой – со стороны коммуникационных портов CP0…CP3 и блока защиты памяти MPU.

*MPU (Memory Protect Unit)* – блок защиты банков памяти NMMB, обслуживает внешние запросы на доступ во внутреннюю память ПУ по AXI порту S_MEM. Если запрос попадает в разрешенную область памяти, то он обслуживается обычным образом, если нет, то он блокируется с формированием соответствующего прерывания.

*CP0…CP3 (Communication Port)* – коммуникационные порты, обеспечивают обмен по двунаправленным 64-х разрядным каналам связи LINK0…LINK3 и доступ в режиме прямого доступа к памяти (ПДП) к банкам памяти NMMB0…NMMB7.

*PU (Peripheral Unit)* – блок периферийных устройств, содержащий контроллер прерываний и таймеры.

На рис. 3 представлен основной компонент СнК NM6408 – процессорный кластер PC.

*NMPU0…NMPU3* – четыре процессорных узла, которые соединены непосредственно «каждый с каждым» тремя 64-разрядными полнодуплексными коммуникационными портами CP1, CP2 и CP3. Обмен данными выполняется с использованием механизма ПДП. Эти ПУ дополнительно соединены через коммуникационные порты CP0 с локальным для кластера управляющим ПУ CPU через коммутаторы коммуникационных портов LC0…LC3. Дополнительно ПУ NMPU-кластера соединены между собой через коммутатор AXI_NMPU, который также обеспечивает произвольный доступ к памятям всех других ПУ системы.

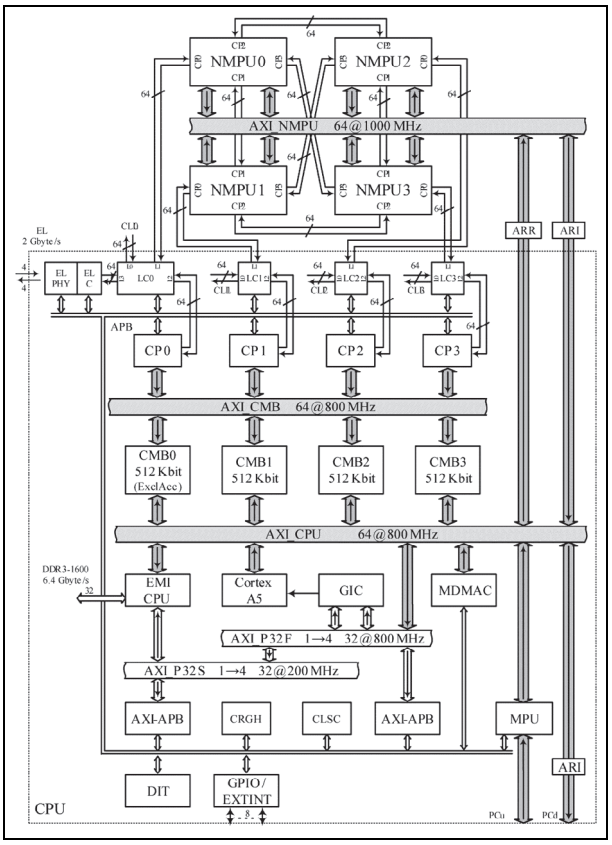


**Рис. 3.** Схема процессорного кластера PC

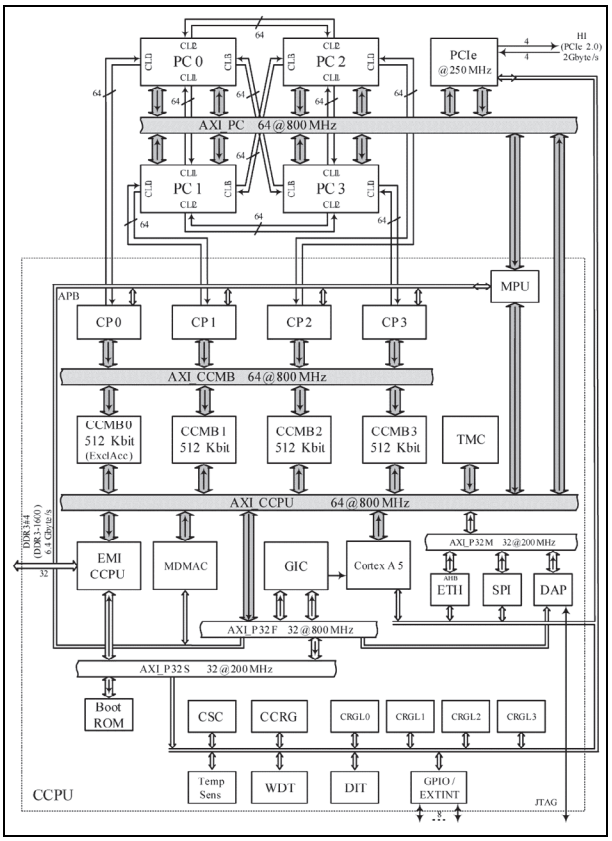


**Рис. 4.** Схема СнК NM6408

*GIC* – контроллер прерываний, который поддерживает до 64-х прерываний с программно настраиваемыми адрес-векторами и приоритетами.

*MDMAC* – высокопроизводительный контроллер ПДП для пересылок «память–память». Основной задачей этого контроллера является обмен данными между внешней памятью DDR3 и внутренними банками кластера CMB0…CMB3. Контроллер также имеет доступ к внутренним банкам памяти NMMB всех NMPU.

*MPU* – блок защиты памяти. Управляющий узел CPU может запрещать доступ внешних устройств к своим внутренним банкам CMB и банкам NMMB внутри своих процессорных узлов NMPU. Если запрос попадает в разрешенную область памяти, то он обслуживается обычным образом, если нет, то он блокируется с формированием соответствующего прерывания, а генерирующему запрос устройству выдается признак ошибки.

*GPIO/EXTINT* – универсальный блок интерфейсов общего назначения. Он управляет работой восьми внешних выводов, каждый из которых может программно настраиваться на функционирование в качестве выводов общего назначения (GPIO), внешних входов прерывания (EXTINT), а пара выводов GPIO может реализовать интерфейс «запрос–подтверждение».

*CLSC* – системный контроллер кластера. Он содержит идентификатор своего кластера, а также служит для генерации запросов на прерывание к другим кластерам.

*CRGH* – генератор тактовых сигналов и сигналов сброса. Он предназначен для управления генерацией синхросигналов и сигналов сброса для NMPU0…NMPU3.

Одним из основных принципов построения СнК NM6408 был принцип иерархии, поэтому схема верхнего уровня СнК, представленная на рис. 4, напоминает схему процессорного кластера PC (рис. 3). На рис. 4 показано, что процессорные кластеры PC0…PC3 соединены непосредственно «каждый с каждым» тремя 64-разрядными полнодуплексными коммуникационными портами CLL1, CLL2 и CLL3. Обмен данными через эти порты выполняется с использованием механизмов ПДП с пропускной способностью 6,4 ГБ/с в каждую сторону.

Каждый из процессорных кластеров PC0…PC3 также соединен с центральным управляющим ПУ CCPU своим коммуникационным портом CLL0. Со стороны CCPU эти порты CLL0 подключены к коммуникационным портам CP0…CP3 соответственно.

Центральный управляющий ПУ CCPU по своему составу похож на ПУ CPU, использованный внут-

ри PC, но имеет следующие особенности: дополнительные внешние низкоскоростные интерфейсы Ethernet 10/100 и SPI; блоки интервальных таймеров DIT и сторожевого таймера WDG; загрузочное ПЗУ BootROM; блок контроллера измерения температуры кристалла с температурным датчиком (TempSens); CCRG – генератор тактовых сигналов и сигналов сбросов, используемых в CCPU; CRGL0…CRGL3 – четыре генератора тактовых сигналов и сигналов сброса для кластеров PC0…PC3; блок DAP согласования с внешним отладочным JTAG-интерфейсом (реализована стандартная аппаратная схема в соответствии с технологией ARM CoreSight, при которой кроме стандартной пошаговой отладки поддерживается сбор трасс выполнения программ со всех пяти ядер Cortex A5, трассы собираются в реальном времени в буфере TMC, после чего их можно выгрузить через порт JTAG и анализировать на персональном компьютере, используя стандартные программно-аппаратные решения фирмы ARM).

### Механизмы межпроцессорного взаимодействия

Для мультипроцессорных систем особенно важными являются средства взаимодействия параллельных процессов и их синхронизации, поскольку возникающие накладные расходы влияют на допустимую гранулированность параллелизма, а, следовательно, и на развиваемую реальную производительность. Далее выделим такие средства, аппаратно реализованные в СнК NM6408: имеется возможность доступа любого процессора и любого контроллера ПДП ко всем словам памяти через глобальное адресное пространство, обеспечиваемое системой коммутаторов, соединяющих все процессоры и банки памяти СнК, и построенного в соответствии с AMBA AXI протоколом; банки памяти CMB0 процессорных кластеров и банк CCMB0 центрального ПУ CCPU поддерживают команды эксклюзивного доступа, которые могут использоваться для реализации семафорных переменных для синхронизации параллельных процессов при работе с критическими разделяемыми ресурсами, прежде всего, участками памяти; взаимодействие параллельных процессов облегчено возможностью асинхронной передачи пакетов данных через каналы межпроцессорного обмена (коммуникационные порты, соединенные парами, позволяют вести обмен данными в режиме ПДП между любыми банками внутренней памяти различных ПУ); для синхронизации процессов, протекающих в различных ПУ, предусмотрена многоярусная система межпроцессорных прерываний.

### Сопроцессор арифметики с плавающей точкой

Для ускорения выполнения арифметических операций над данными, представленными в формате с плавающей точкой, используется векторно-матричный сопроцессор арифметики с плавающей точкой. Сопроцессор выполняет операции в соответствии со стандартом IEEE 754-2008 [11] и работает с данными одинарной точности (32 разряда) и двойной точности (64 разряда).

Схема сопроцессора представлена на рис. 5.

Рассмотрим основные узлы матрично-векторного сопроцессора.

*CENTRAL CONTROL UNIT* – центральный блок управления сопроцессором. Он принимает команды от управляющего RISC-ядра по внутренней шине команд (INTERNAL INSTRUCTION BUS), дешифрирует и проверяет их на правильность. Если команда правильная, то она запускается на выполнение при условии, что все требуемые ей ресурсы свободны. В случае ошибочности команды она не выполняется, и при этом формируется соответствующее прерывание от сопроцессора. 32-разрядная скалярная шина данных (SCALAR DATA BUS) используется для чтения/записи программно-доступных скалярных регистров блока управления.

*REPACK UNIT* – блок упаковки и распаковки данных. Данный блок предназначен для различных преобразований данных в формате с плавающей точкой в целые числа и обратно, а также 64-разрядных данных в 32-разрядные и наоборот.

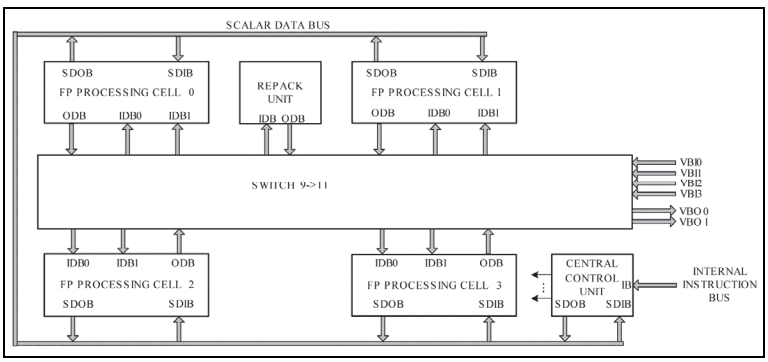


**Рис. 5.** Схема сопроцессора арифметики с плавающей точкой

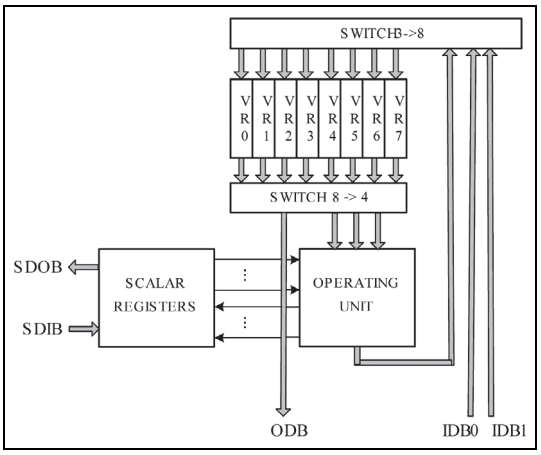


**Рис. 6.** Схема процессорной ячейки

*SWITCH 9->11* – коммутатор 9 в 11, обеспечивающий обмен данными между функциональными узлами сопроцессора с плавающей точкой и памятью процессорной системы.

*FP PROCESSING CELL 0…FP PROCESSING CELL 3* – четыре одинаковые процессорные ячейки, каждая из которых осуществляет арифметические операции над данными в формате с плавающей точкой как одинарной (32 разряда), так и двойной (64 разряда) точности. Каждая ячейка имеет следующие 64-разрядные шины: две входных – IDB0, IDB1, и одну выходную – ODB, что позволяет за один такт осуществить до двух операций чтения и одной операции записи. 32-разрядная скалярная шина данных (SCALAR DATA BUS) используется для чтения/записи программно-доступных скалярных регистров процессорных ячеек. Процессорные ячейки являются основным вычислительным элементом сопроцессора арифметики с плавающей точкой. Схема процессорной ячейки приведена на рис. 6.

Процессорная ячейка включает в себя следующие основные узлы.

*VR0…VR7* – векторные регистры общего назначения, которые используются как в операциях ввода/вывода, так и в арифметических операциях над данными с плавающей точкой. Их максимальная емкость – 32 64-разрядных слова упакованных данных. В некоторых типах операций векторные регистры образуют регистровые пары, формируя вектора максимальным размером 32 элемента по 128 разрядов.

*SWITCH 3->8* – коммутатор 3 в 8.

*SWITCH 8->4* – коммутатор 8 в 4.

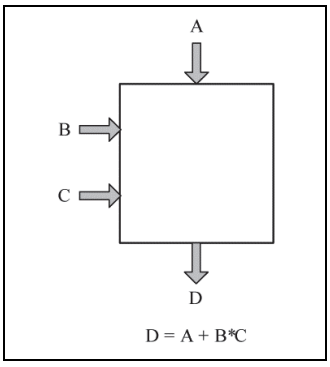


**Рис. 7.** Операции над данными в формате с плавающей точкой двойной точности

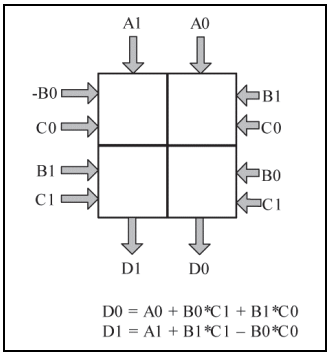


**Рис. 8.** Операции над комплексными данными в формате с плавающей точкой одинарной точности

*SCALAR REGISTERS* – программно-доступные скалярные регистры, входящие в состав каждой процессорной ячейки. Скалярные регистры хранят значение признаков, формируемых при выполнении арифметических операций, также с помощью данных регистров осуществляется управление маскированием выполнения векторных операций.

*OPERATING UNIT* – операционное устройство для выполнения векторных и матричных операций над данными в формате с плавающей точкой. Данное устройство может работать в одном из четырех режимов:

1. *Операции над данными в формате с плавающей точкой двойной точности* (рис. 7). В этом режиме все входные операнды *A*, *B* и *C* и результат *D* представляют собой 64-разрядные числа в формате с плавающей точкой двойной точности. Основная операция режима – умножение двух чисел и сложение с третьим.

2. *Операции над комплексными данными в формате с плавающей точкой одинарной точности* (рис. 8). Данный режим характеризуется тем, что все входные операнды *A*1 и *A*0, *B*1 и *B*0, *C*1 и *C*0, а также результат *D*1 и *D*0 представляют собой комплексные 64-разрядные числа, причем старшие 32 разряда содержат действительную часть (*A*1, *B*1, *C*1 и *D*1), а младшие 32 разряда – мнимую часть (*A*0, *B*0, *C*0 и *D*0). Основная операция режима – умножение двух комплексных чисел и сложение с третьим.

3. *Операции над векторными данными в формате с плавающей точкой одинарной точности* (рис. 9). В этом режиме все входные операнды *A*1 и *A*0, *B*1 и *B*0, *C*1 и *C*0, а также результат *D*1 и *D*0 образуют 64-разрядные вектора из двух 32-разрядных элементов. Основная операция режима – умножение двух векторов по два элемента и сложение с третьим двухэлементным вектором.

4. *Операции над матричными данными в формате с плавающей точкой одинарной точности* (рис. 10). Данный режим характеризуется тем, что

все входные операнды $A1$ и $A0$, $B1$ и $B0$, $C1$ и $C0$, а также результат $D1$ и $D0$ образуют 64-разрядные векторы из двух 32-разрядных элементов, как и в предыдущем случае. Основная операция режима – умножение вектора из двух элементов [$A1$ $A0$] на матрицу 2×2 элемента $\begin{bmatrix} B3 & B1 \\ B2 & B0 \end{bmatrix}$ и сложение с третьим двухэлементным вектором [$D1$ $D0$]. Особенностью данной операции является то, что матрица считывается сразу из пары векторных регистров: из того, что указан в команде (он обязан иметь четный номер) считывается столбец $\begin{bmatrix} B1 \\ B0 \end{bmatrix}$, а из регистра с номером на единицу больше – столбец $\begin{bmatrix} B3 \\ B2 \end{bmatrix}$. В остальных случаях операнд в виде одного 64-разрядного данного или вектора из двух 32-разрядных данных считывается только из одного векторного регистра. Результат также пишется только в один векторный регистр.

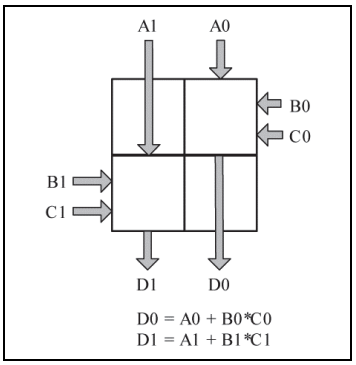


**Рис. 9.** Векторные операции над данными в формате с плавающей точкой одинарной точности

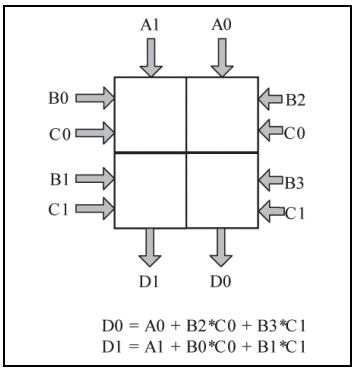


**Рис. 10.** Матричные операции над данными в формате с плавающей точкой одинарной точности

- Микросхема NM6408, разработанная специалистами ЗАО НТЦ «Модуль», – большая СнК, реализующая передовые архитектурные концепции обеспечения толерантности к задержкам обращений к памяти и гетерогенной организации с использованием разнотипных ядер. Первые инженерные образцы получены и проходят испытания.

  Ближайший этап – освоение описанных архитектурных особенностей в программном обеспечении и пользователями. Концепции проведения этих работ составлены, детализированы и находятся в стадии реализации. Программирование будет осуществляться на языке C/C++ и ассемблере. Для управления параллельными процессами и их взаимодействием разрабатывается собственная библиотека функций на основе библиотек MPI (Message Passing Interface) и SHMEM (Cray Research «shared memory» library), но сильно упрощенная и адаптированная под имеющиеся архитектурные особенности СнК NM6408 (наличие коммуникационных портов, ярко выраженной иерархии системы памяти, многоярусной системы прерываний, наличия высокопроизводительных MDMAC контроллеров ПДП). Особое внимание уделено средствам отладки, трассировки и анализа процессов выполнения параллельных программ. Разрабатываются имитационные модели разного уровня детализации для тонкой машинно-зависимой оптимизации программ.

  Уже востребована разработка новых микросхем типа СнК NM6408, но с повышенной в разы пиковой производительностью и улучшенной архитектурой. Кроме простого увеличения числа ПУ в СБИС такого типа новых поколений, в качестве рассматриваемых вариантов улучшения архитектуры можно назвать усиление скалярного RISC-процессора в ядрах NMC и введение новых типов операций в векторном сопроцессоре.

## Литература


1. *Dally W.*, *Balford J. et al*. An Energy-Efficient Processor Architecture // IEEE Computer Architecture Letters. Jan. 2008. V. 7. № 1. P. 29–31.
2. *Nowatzki T.*, *Wright G. et al*. Pushing the Limits of Accelerator Efficiency While Retaining Programmability // IEEE High performance computer architecture conference. 2016. 13 p.
3. *Durant L.*, *Harris M. et al*. Inside Volta: The World's Most Advanced Data Center GPU. 10 May 2017. URL = https://devblogs.nvidia.com/parallelforall/inside-volta (дата обращения: 01.09.2017).
4. *Эйсымонт Л.К.* Гибридная стратегия развития элементной базы // Открытые системы. СУБД. 2017. № 2. С. 8–11. URL = https://www.osp.ru/os/2017/02/13052216 (дата обращения: 01.09.2017).
5. *Mujtaba H.* NVIDIA Announces Xavier Tegra SOC – Features Volta GPU With 7 Billion Transistors, 512 CUDA Cores and 8 ARM64 Custom Cores. Sep. 28, 2016. URL = http://wccftech.com/nvidia-xavier-soc-tegra-volta-gpu-announced (дата обращения: 01.09.2017).
6. *Weinberg J.* Quantifying locality in the memory access patterns of HPC Applications. University of California, San Diego. 2005. 50 p.

7. *Murphy R.C.*, *Kogge P.M.* On the Memory Access Patterns of Supercomputer Applications: Benchmark Selection and Its Implications // IEEE Transactions on Computers. July 2007. V. 56. № 7. 9 p.
8. *Egawa R. et al*. Early evaluation of the SX-ACE Processor SC14. November 2014. 2 p.
9. *Черников В.М.*, *Виксне П.Е.*, *Шелухин А.М.*, *Шевченко П.А.*, *Панфилов А.П.*, *Косоруков Д.Е.*, *Черников А.В.* Семейство процессоров обработки сигналов с векторно-матричной архитектурой NeuroMatrix // Электронные компоненты. 2006. № 6. С. 79–84.
10. *Черников В.М.*, *Виксне П.Е.*, *Шелухин А.М.*, *Панфилов А.П.* Отечественные высокопроизводительные процессоры цифровой обработки сигналов векторно-матричной архитектуры, перспективы развития // Материалы конф. «Перспективы развития высокопроизводительных архитектур. История, современность и будущее отечественного компьютеростроения». М.: ИТМиВТ им С.А. Лебедева РАН. 2008. № 1. С. 52–59.
11. IEEE 754-2008 – IEEE Standard for Floating-Point Arithmetic. © Copyright IEEE. 2008.

**Поступила 3 августа 2018 г.**

# Heterogeneous multicore system on chip with 512 Gflops peak performance



**A.L. Eisymont** – Head of Sector, RC «Module» (Moscow)
E-mail: eisymont@module.ru
**V.M. Chernikov** – Ph.D.(Eng.), Main Designer – Head of Department, RC «Module» (Moscow)
E-mail: tchern@module.ru
**An.V. Chernikov** – Head of Sector, RC «Module» (Moscow)
E-mail: chernant@module.ru
**Al.V. Chernikov** – Deputy Head of Department, RC «Module» (Moscow)
E-mail: achernikov@module.ru
**D.E. Kosorukov** – Head of Department, RC «Module» (Moscow)
E-mail: dkos@module.ru
**I.I. Nasonov** – Head of Sector, RC «Module» (Moscow)
E-mail: nasonov@module.ru
**A.A. Komlev** – Leading Engineer, RC «Module» (Moscow)
E-mail: a.komlev@module.ru

This article is devoted to questions and methods of implementing energy efficient heterogeneous and tolerant to memory latency system on chip (SoC) operating at 1 GHz frequency, with 512 Gflops peak performance and hierarchically organized internal memory. SoC contains sixteen NeuroMatrix NMC4 processor cores and five ARM Cortex-A5. The next step is to master the described architectural features in the software and users. The concepts of carrying out these works are compiled, detailed and under implementation.

**References**

1. *Dally W.*, *Balford J. et al*. An Energy-Efficient Processor Architecture // IEEE Computer Architecture Letters. Jan. 2008. V. 7. № 1. P. 29–31.
2. *Nowatzki T.*, *Wright G. et al*. Pushing the Limits of Accelerator Efficiency While Retaining Programmability // IEEE High performance computer architecture conference. 2016. 13 p.
3. *Durant L.*, *Harris M. et al*. Inside Volta: The World's Most Advanced Data Center GPU. 10 May 2017. URL = https://devblogs. nvidia.com/parallelforall/inside-volta (data obrashcheniya: 01.09.2017).
4. *Eisymont L.K.* Gibridnaya strategiya razvitiya ehlementnoj bazy // Otkrytye sistemy. SUBD. 2017. № 2. S. 8–11. URL = https:// www.osp.ru/os/2017/02/13052216 (data obrashcheniya: 01.09.2017).
5. *Mujtaba H.* NVIDIA Announces Xavier Tegra SOC – Features Volta GPU With 7 Billion Transistors, 512 CUDA Cores and 8 ARM64 Custom Cores. Sep. 28, 2016. URL = http://wccftech.com/nvidia-xavier-soc-tegra-volta-gpu-announced (data obrashcheniya: 01.09.2017).
6. *Weinberg J.* Quantifying locality in the memory access patterns of HPC Applications. University of California, San Diego. 2005. 50 p.
7. *Murphy R.C.*, *Kogge P.M.* On the Memory Access Patterns of Supercomputer Applications: Benchmark Selection and Its Implications // IEEE Transactions on Computers. July 2007. V. 56. № 7. 9 p.
8. *Egawa R. et al*. Early evaluation of the SX-ACE Processor SC14. November 2014. 2 p.
9. *Chernikov V.M., Viksne P.E., Sheluhin A.M., Shevchenko P.A., Panfilov A.P., Kosorukov D.E., Chernikov A.V.* Semejstvo processorov obrabotki signalov s vektorno-matrichnoj arhitekturoj NeuroMatrix // Elektronnye komponenty. 2006. № 6. S. 79–84.
10. *Chernikov V.M., Viksne P.E., Sheluhin A.M., Panfilov A.P.* Otechestvennye vysokoproizvoditel'nye processory cifrovoj obrabotki signalov vektorno-matrichnoj arhitektury, perspektivy razvitiya // Materialy konf. «Perspektivy razvitiya vysokoproizvoditel'nyh arhitektur. Istoriya, sovremennost' i budushchee otechestvennogo komp'yuterostroeniya». M.: ITMiVT im S.A. Lebedeva RAN. 2008. № 1. S. 52–59.
11. IEEE 754-2008 – IEEE Standard for Floating-Point Arithmetic. © Copyright IEEE. 2008.

УДК 004.382.2

# Специализированные реконфигурируемые вычислители в сетецентрических суперкомпьютерных системах



**А.П. Антонов** – к.т.н., доцент, Санкт-Петербургский политехнический университет Петра Великого
E-mail: alexander.antonov.ru@yandex.ru
**В.С. Заборовский** – д.т.н., профессор, директор Института компьютерных наук и технологий, Санкт-Петербургский политехнический университет Петра Великого
E-mail: vlad2tu@yandex.ru
**И.О. Киселев** – аспирант, Санкт-Петербургский политехнический университет Петра Великого
E-mail: kio.93@mail.ru

Показано, как реконфигурируемые вычислительные системы способны повысить эффективность суперкомпьютерных систем в соответствии со следующими наиболее важными параметрами: отношением производительности к энергопотреблению (ТФлопс/Вт); реальной производительностью (отношением реальной производительности к пиковой производительности); требованиями к возможности обмена данными с вычислителем без участия Хост-вычислителя. Предложена и обоснована идея «искусственного интеллекта», реализующего процедуру профилирования выполняемых программ и автоматизированной реконфигурации внутренней архитектуры вычислительных модулей под алгоритм решаемой задачи и обеспечивающего кратное улучшение указанных параметров суперкомпьютерных систем. Рассмотрена реализованная плата реконфигурируемого вычислителя.

***Ключевые слова:*** *гетерогенные суперкомпьютерные системы, стандарт OpenCL, реконфигурируемые вычислители, архитектуры суперкомпьютерных систем, ПЛИС.*

The requirements to modern supercomputer systems get tougher each year. The most significant values are: performance to power consumption rate (TFlops/Wt), effective performance, i.e. actual performance to peak performance rate. In addition, data transfer within computational unit without host computer management is required. Paper shows how reconfigurable computers allow to increase the effectiveness of supercomputer systems in order to suit those requirements. Besides, we offer and justify an idea of «artificial intelligence», that implements the procedure of profiling the executing programs and automatically reconfigures the hardware architecture to fit the exact algorithm. This approach can increase supercomputer system parameters by an order of magnitude. Paper also shows a custom-designed reconfigurable platform for high performance computing.

***Keywords:*** *heterogeneous supercomputer systems, OpenCL, reconfigurable computational module, supercomputer systems architecture, FPGA.*

**DOI: 10.18127/j20729472-201803-09**

Процессы цифровой трансформации различных секторов отечественной и мировой экономики позволяют с новых позиций обосновать приоритетные направления развития компьютерных наук и технологий. В современных условиях особое значение приобретают не только чисто технические характеристики компьютерных систем, такие как уровень реальной производительности решения конкретных прикладных задач, обеспечение информационной безопасности и энергопотребление, но также эксплуатационно-экономические характеристики – стоимость владения, возможности «эволюционной» модернизации и пр. Достижение технико-экономической эффективности использования компьютерных систем требует поиска сбалансированных в аспекте «стандартизация/специализация» решений.

В последние годы магистральным направлением совершенствования компьютерных систем стал тотальный переход к применению технологий кластеризации с использованием массово-параллельных, многоядерных и мультитредовых архитектур, реализуемых на базе MIMD-микропроцессоров, гетерогенных SIMD-ускорителей и реконфигурируемых вычислителей на базе программируемых логических интегральных схем (FPGA). Суперкомпьютерные кластеры, занимающие лидирующие позиции в списке TOP 500, в настоящее время превратились в гибридные вычислительные системы, имеющие «на борту» несколько десятков тысяч вычислительных узлов, более 10 млн процессорных ядер и 1 петабайт оперативной памяти. Однако реальная производительность таких систем весьма чувствительна к возможностям «натянуть» алгоритм решения конкретной прикладной программы на аппаратно-программную архитектуру суперкомпьютерного кластера. Поэтому современная таксономия высокопроизводительных

компьютеров, которые интегрируют в единую вычислительную систему различные устройства обработки и хранения данных, существенно расширилась по сравнению с эпохой «суперпроцессоров», реализующих с помощью имеющихся ресурсов памяти и процессоров определенный алгоритм.

Так, в SMP-системах (Symmetric Multi-Processing) все данные доступны для всех процессорных узлов с одними и теми же характеристиками задержки. В NUMA-системе (Non-Uniform Memory Access) доступ к «чужим» данным более медленный, чем к данным, расположенным в так называемой локальной памяти процессора. Система с архитектурой MPP (Massively Parallel Processing) образует сеть доступа к «чужим» данным, что описывается в программе в виде запроса на обмен подобно запросу на обмен с жестким диском или другим устройством внешней памяти. Оценки эффекта ускорения вычислений за счет параллельной работы процессоров в рамках модели «одна программа – много данных» (модель SPM) феноменологически описывается законами Амдала–Уэра для программ с неизменной долей последовательных и параллельных вычислений и законом Густавсона–Борсиса для программ, которые могут усложняться за счет увеличения объема обрабатываемых данных. Очевидно, что универсального и радикального решения проблемы повышения производительности энерго-вычислительной эффективности компьютерных систем нет. Однако сблизить значения реальной и пиковой производительности компьютерных систем можно, если, опираясь на законы Амдала и Густавсона, динамически реконфигурировать архитектуру вычислителя, учитывая особенности алгоритма, структуру обрабатываемых данных и критерий эффективности работы системы в целом. В общем случае оптимизация «на лету» алгоритма вычислений, которая учитывает различные ограничения, связанные с контекстом решения задачи, и конкретные архитектурные особенности компьютера, является весьма сложной как с теоретической, так и с технологической точек зрения проблемой.

Ниже рассматривается одно из возможных решений, основанное на изменении модели вычислений за счет перехода от «вызов процедуры-при-упоминании/передача-по-ссылке» на «вызов процедуры по значению/контексту», когда аргументы процедуры вычисляются до их передачи телу процедуры. В такой модели возможен «вызов функции по необходимости», что позволяет автоматически реконфигурировать архитектуру вычислителя в соответствии с информационной структурой алгоритма, одновременно поддерживая высокий уровень реальной производительности и удельной энерго-вычислительной эффективности вычислительных процессов.

В качестве программной платформы для апробации предлагаемых решений выбран стандарт OpenCL – язык программирования, который позволяет использовать преимущества вычислительных моделей, реализованных на базе стандартных и специализированных процессоров GPGPU и FPGA. Программная платформа OpenCL обладает развитыми выразительными средствами описания вычислительных процессов, включая абстракции типа: процессорный элемент, вычислительный блок, вычислительное устройство, хост-система и др. Эти возможности позволяют эффективно отображать, в том числе с использованием методов машинного обучения, особенности прикладных алгоритмов на имеющиеся у вычислителя программно-аппаратные ресурсы, учитывая при этом динамически изменяющиеся структуры данных и структуру сетецентрической связанности узлов суперкомпьютерного кластера. В результате процесс вычислений может быть организован так, чтобы обеспечить компромисс между затратами электроэнергии на вычисления, доступностью ресурсов памяти, временем на обработку данных и производительностью сети связи.

Ц е л ь  р а б о т ы – показать, как реконфигурируемые вычислительные системы способны повысить эффективность суперкомпьютерных систем в соответствии с такими важными параметрами, как отношение производительности к энергопотреблению и реальная производительность (отношение реальной производительности к пиковой производительности).

### Реконфигурация вычислителя, управляемая алгоритмом

Широко используемые решения ресурсоемких вычислительных задач с применением гибридных кластеров на базе многоядерных микропроцессоров, ПЛИС-ускорителей и графических ускорителей известны, но за счет своей структурной избыточности характеризуются высоким потреблением электроэнергии, что существенно повышает стоимость эксплуатации таких кластеров и требует создания специализированной системы охлаждения, что ограничивает их доступность для конечных пользователей. Как

было отмечено ранее, объединение отдельных вычислителей в кластерную систему для решения одной вычислительной задачи – процесс нетривиальный, требующий сбалансированного использования аппаратно-программных ресурсов. Эффективность кластеризации зависит от того, как информационная структура конкретного алгоритма, представленная в виде программы, отображается на архитектуру вычислителя, ключевым требованием к которому является обеспечение высокого уровня параллелизма процедуры решения задачи по командам и данным (рис. 1.)

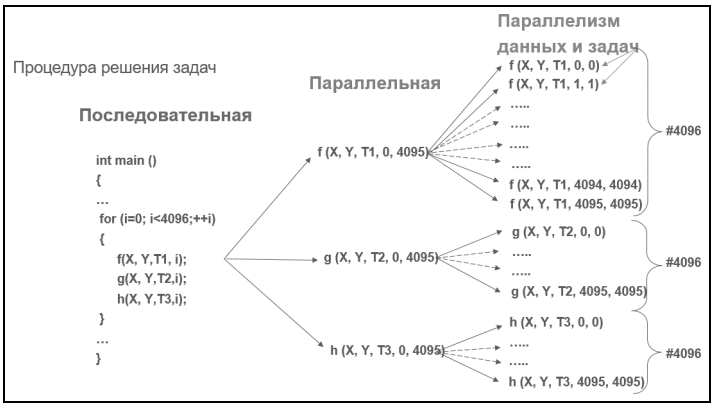


**Рис. 1.** Параллелизм процедуры решения задач по командам и данным

Другой важной характеристикой процессов вычислений является потребляемая электрическая мощность, необходимая для работы процессора, системы памяти и интерфейсных контроллеров. Диссипация энергии связана с необратимым в физическом смысле характером процессов вычислений, определяющим связь между информационной и термодинамической энтропией. Так как современные микропроцессоры на макроуровне построены на логических вентилях «И-НЕ» или «ИЛИ-НЕ», то все такие операции уменьшают информационную энтропию входного сигнала, что приводит к увеличению термодинамической энтропии, то есть к выделению тепла. Принцип Ландауэра устанавливает, что потеря одного бита информации во входном сигнале должна приводить к выделению не менее $kT\ln 2$ Дж, где $k$ – постоянная Больцмана; $T$ – температура системы. Сама по себе эта энергия невелика: для $T = 300$ K она составляет всего 0,017 эВ на бит, но в пересчете на число логических вентилей в современном микропроцессоре суммарная энергия вырастает уже до величин порядка 1 Дж за каждую секунду работы.

Хотя реконфигурируемые вычислители на базе FPGA имеют по сравнению с микропроцессорами CPU и графическими ускорителями GPGPU большую удельную емкость логических элементов, такие устройства проводят вычисления на существенно более низких частотах и с меньшими потерями информации, что позволяет либо снизить потребление энергии при сравнимой производительности, либо при тех же энергетических затратах получить большую производительность (по оценкам, энергоемкость реконфигурируемого вычислителя 10 Вт на 1 Тфлопс с ограничением 10 Тфлопс на одну интегральную схему (ИС)). Основные преимущества стандарта OpenCL заключаются в его кроссплатформенности, то есть поддержке со стороны крупнейших производителей ИС. Это позволяет запускать одну и ту же программу на гибридном кластере, состоящем из разных типов вычислителей. Современная реализация OpenCL дает возможность, не внося существенных изменений в прикладное ПО (ППО), «на лету» заменять один тип сопроцессора на другой. Модель работы в OpenCL приведена на рис. 2.

Применение OpenCL для FPGA базируется на технологии частичной реконфигурации, при которой часть аппаратных ресурсов, которая отвечает за работу шины PCI Express и периферийных устройств, остается неизменной, а часть FPGA, отвечающая за вычисления в соответствии с прикладным алгоритмом, может изменять свою структуру согласно с описанием реализуемой функции. Это позволяет в режиме реального времени изменять конфигурацию FPGA без перезагрузки системы, уменьшая расходы электроэнергии на передачу данных по коммуникационным шинам.

Функциональная схема и схемотехническое решение прототипа реконфигурируемого вычислительного устройства приведены на рис. 3.

Прототип реализован как PCIe-устройство, выполненное в стандарте графической карты, которое может быть установлено в любую вычислительную систему, имеющую разъемы PCIe16x. Прототип содержит четыре FPGA (KCU115 компании Xilinx), объединенных в вычислительный кластер (полная связность через каналы PCIe8x и высокоскоростные каналы Aurora GTH8x). Каждый реконфигурируемый вычислитель в кластере имеет собственную память 4 Гбайт (1 Гбайт 32-разрядных слов), что позволяет отнести его к классу MPP-систем.

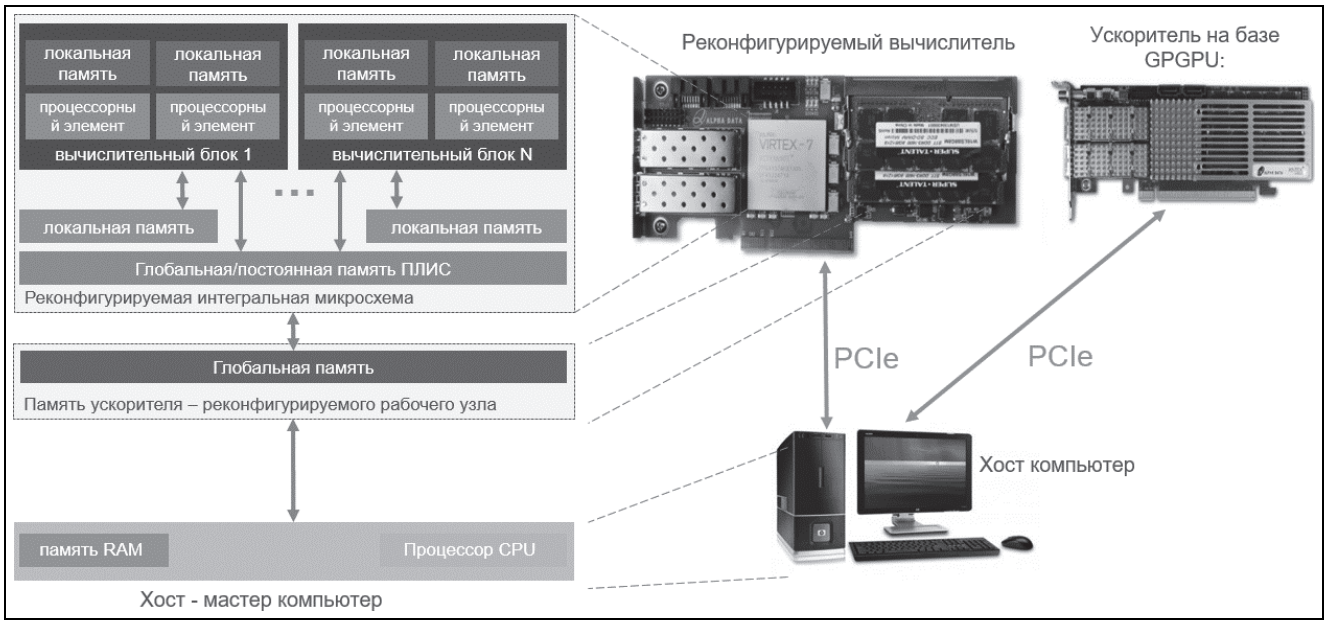


**Рис. 2.** Модель работы в OpenCL

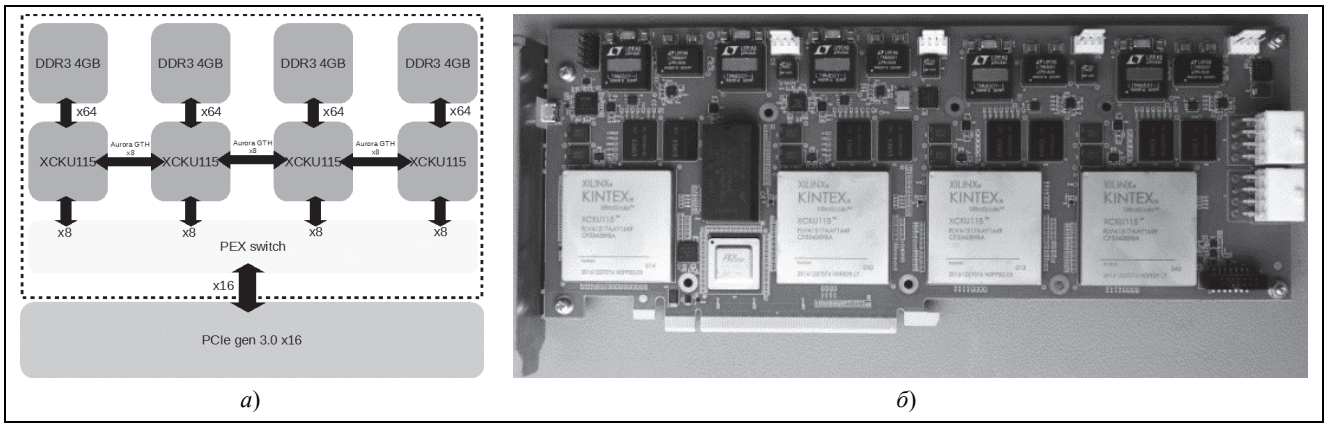


**Рис. 3.** Функциональная схема (*а*) и схемотехническое решение (*б*) прототипа реконфигурируемого вычислительного устройства

Возможности реконфигурации аппаратной архитектуры позволяют использовать методы машинного обучения для оптимизации вычислительного процесса, применяя принципы высокоуровневого проектирования. Для этого на этапе системной формализации задачи вычислений в терминах языка OpenCL формулируется алгоритм решения, а затем с помощью стандартных инструментальных средств производителя FPGA синтезируется реконфигурируемый вычислитель, архитектура которого отражает информационную структуру реализуемого алгоритма.

Технология интеграции в гетерогенную вычислительную систему реконфигурируемых вычислителей предполагает использование стандартных FPGA и специального ППО. Для разработки таких систем требуются подготовленные специалисты, имеющие соответствующие знания, умения и практические навыки работы с новыми технологиями. Поэтому одним из направлений развития исследований в области проектирования и эксплуатации гибридных суперкомпьютерных кластеров является создание в ИКНТ (СПбПУ) научно-лабораторного комплекса (НЛК) «Проектирование реконфигурируемых аппаратных ускорителей для гетерогенных вычислительных систем». На базе НЛК будет организована подготовка магистров по направлениям «Информатика и вычислительная техника» и «Математика и компьютерные науки».

Поскольку стоимость коммерчески доступных реконфигурируемых вычислителей достаточно высока, для оборудования НЛК, содержащего десятки учебных реконфигурируемых установок, предполагается:

1) создать универсальную учебно-лабораторную платформу для исследования возможностей реконфигурации вычислителей (функциональная схема платы и организация вычислительной среды приведены на рис. 4);

2) разработать прототип системы машинного обучения для системы исполнения заданий на основе интеграции и/или реконфигурации вычислительных ресурсов гибридных кластеров с целью достижения наивысшей производительности и энерго-вычислительной эффективности при решении задач инженерного моделирования.

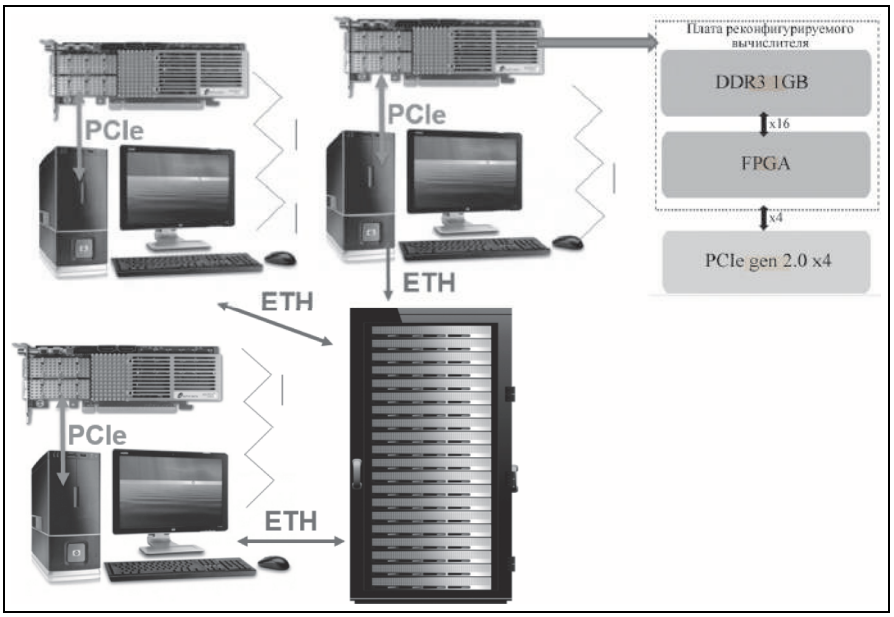


**Рис. 4.** Организация вычислительной среды и функциональная схема платы

- Решение приведенных задач позволит создать прототип самонастраивающейся гибридной кластерной системы, которая станет частью инструментальной облачной среды СКЦ «Политехнический». Применение такой системы позволит в соответствии с различными критериями оптимизации (производительность, удельная энерго-вычислительная эффективность и пр.) на основе данных профилирования прикладных алгоритмов и загрузки узлов автоматизировать процесс реконфигурации вычислительного поля суперкомпьютерного кластера. При этом, если в ходе анализа процессов обнаруживается изменение контекста задачи, структуры данных или целевой функции системы, то процедура реконфигурации структуры вычислителя запускается снова. В результате «эволюция» суперкомпьютерной платформы и обучение системы управления процессами распределения задач продолжается непрерывно как с точки зрения настройки прикладного алгоритма, так и с точки зрения добавления новых аппаратных или программных компонент.

## Литература


1. Top 500 list [Электронный ресурс]. URL = https://www.top500.org/ (дата обращения: 01.05.2018).
2. OpenCL specification 1.2. Khronos Group. 2012. 380 с.
3. *Каляев И.А.*, *Левин И.И.*, *Семерников Е.А.*,*Шмойлов В.И.* Реконфигурируемые мультиконвейерные вычислительные структуры: Учебник для вузов. Изд. 1-е. Ростов-на-Дону: ЮНЦ РАН. 2008. 393 с.
4. *Каляев И.А.*, *Левин И.И.*, *Семерников Е.А.*,*Шмойлов В.И.* Развитие отечественных многокристалльных реконфигурируемых вычислительных систем: от воздушного к жидкостному охлаждению // Труды СПИИРАН. 2017. № 1. С. 27−39.
5. *Zaborovsky V.S.*, *Mulukha V.A. et al.* Preemptive Queuing System with Randomized Push-Out Mechanism // Communications in Nonlinear Science and Numerical Simulation. April 2015. V. 21. № 1−3. P. 147−158.
6. *Румянцев Ю.А., Захаров П.Н.*, *Абрашитова Н.А.*, *Шматок А.В.*, *Рыжих В.О.*, *Гудимчук Н.Б.*, *Атауллаханов Ф.И.* Применение ПЛИС для расчета деполимеризации микротрубочки методом броуновской динамики // Труды Института системного программирования РАН. 2016. Т. 28. № 3. С. 241−266.
7. *Wang Z.*, *Zhang S.*, *He B.*, *Zhang W.* Melia: A MapReduce framework on OpenCL-based FPGA // IEEE Transactions on Parallel and Distributed Systems. 2015. С. 27−39.
8. *Kobayashi R.*, *Oobata Y.*, *Fujita N.*, *Yamaguchi Y.*, *Boku T.* OpenCL-ready High Speed FPGA Network for Reconfigurable High Performance Computing // Proc. of the International Conference on High Performance Computing in Asia-Pacific Region (HPC Asia 2018). 2018. С. 192−201.
9. *Zwagerman M.* High Level Synthesis, a Use Case Comparison with Hardware Description Language // Grand Valley State University Masters Theses. 2015. 36 с.


**Поступила 3 августа 2018 г.**

# The reconfigurable computational modules in network-centric supercomputer systems



**A.P. Antonov** – Ph.D.(Eng.), Associate Professor, Peter The Great St. Petersburg Polytechnic University
E-mail: alexander.antonov.ru@yandex.ru
**V.S. Zaborovskij** – Dr.Sc.(Eng.), Professor, Director of Institute of Computer Science and Technology, Peter The Great St. Petersburg Polytechnic University
E-mail: vlad2tu@yandex.ru
**I.O. Kiselev** – Post-graduate Student, Peter The Great St. Petersburg Polytechnic University
E-mail: kio.93@mail.ru

The requirements to modern supercomputer systems get tougher each year. The most significant values are: performance to power consumption rate (TFlops/Wt), effective performance, i.e. actual performance to peak performance rate. In addition, data transfer within computational unit without host computer management is required. Paper shows how reconfigurable computers allow to increase the effectiveness of supercomputer systems in order to suit those requirements. Besides, we offer and justify an idea of «artificial intelligence», that implements the procedure of profiling the executing programs and automatically reconfigures the hardware architecture to fit the exact algorithm. This approach can increase supercomputer system parameters by an order of magnitude. Paper also shows a custom-designed reconfigurable platform for high performance computing. The solution of these problems will allow creating a prototype of a self-adjusting hybrid cluster system that will become part of the instrumental cloud environment of the «Politehnicheskiy» SCC.

**References**

1. Top 500 list [Elektronnyj resurs] URL = https://www.top500.org/ (data obrashcheniya: 01.05.2018).
2. OpenCL specification 1.2. Khronos Group. 2012. 380 s.
3. *Kalyaev I.A., Levin I.I., Semernikov E.A.,Shmojlov V.I.* Rekonfiguriruemye mul'tikonvejernye vychislitel'nye struktury: Uchebnik dlya vuzov. Izd. 1-e. Rostov-na-Donu: YUNC RAN. 2008. 393 s.
4. *Kalyaev I.A., Levin I.I., Semernikov E.A.,Shmojlov V.I.* Razvitie otechestvennyh mnogokristall'nyh rekonfiguriruemyh vychislitel'nyh sistem: ot vozdushnogo k zhidkostnomu ohlazhdeniyu // Trudy SPIIRAN. 2017. № 1. S. 27–39.
5. *Zaborovsky V.S.*, *Mulukha V.A. et al.* Preemptive Queuing System with Randomized Push-Out Mechanism // Communications in Nonlinear Science and Numerical Simulation. April 2015. V. 21. № 1–3. P. 147–158.
6. *Rumyancev Yu.A., Zaharov P.N., Abrashitova N.A., Shmatok A.V., Ryzhih V.O., Gudimchuk N.B., Ataullahanov F.I.* Primenenie PLIS dlya rascheta depolimerizacii mikrotrubochki metodom brounovskoj dinamiki // Trudy Instituta sistemnogo programmirovaniya RAN. 2016. T. 28. № 3. S. 241–266.
7. *Wang Z.*, *Zhang S.*, *He B.*, *Zhang W.* Melia: A MapReduce framework on OpenCL-based FPGA // IEEE Transactions on Parallel and Distributed Systems. 2015. S. 27–39.
8. *Kobayashi R.*, *Oobata Y.*, *Fujita N.*, *Yamaguchi Y.*, *Boku T.* OpenCL-ready High Speed FPGA Network for Reconfigurable High Performance Computing // Proc. of the International Conference on High Performance Computing in Asia-Pacific Region (HPC Asia 2018). 2018. S. 192–201.
9. *Zwagerman M.* High Level Synthesis, a Use Case Comparison with Hardware Description Language // Grand Valley State University Masters Theses. 2015. 36 s.

УДК 004.4:519.178

# Особенности хранения графов социальной сети[1]



**И.В. Поляков** – к.ф.-м.н., доцент, НИУ ВШЭ (Москва)
E-mail: ipolyakov@hse.ru
**В.О. Полякова** – студент, НИУ ВШЭ (Москва)
E-mail: igorp86@mail.ru
**А.А. Чеповский** – к.ф.-м.н., доцент, НИУ ВШЭ (Москва)
E-mail: aachepovsky@hse.ru

Рассмотрены различные варианты хранения графов социальной сети больших размеров. Показано, что совместное использование двух типов хранилищ позволяет получить гибкую схему, ориентированную как на быстрое пополнение графа, так и на быстрые операции поиска путей и выгрузки подграфов.

***Ключевые слова:*** *хранилище графов, большие данные, поиск путей, выгрузка подграфов.*

In this paper we propose different approaches for compact storaging of big social graphs. The joint use of two types of storages makes it possible to obtain a flexible scheme that is oriented both to the fast completion of the graph and to fast searches for the paths and extracting of subgraphs.

***Keywords:*** *graph warehouse, big data, searching for paths, extracting subgraphs.*

**DOI: 10.18127/j20729472-201803-10**

Специализированные базы данных для хранения и обработки графов получили широкое распространение в настоящее время. Область применения такого рода баз включает в себя как химические и биологические задачи, так и задачи хранения и обработки графов социальных сетей. Высокая масштабируемость и меньшая структурированность моделей данных дает возможность обрабатывать объемы данных, существенно превосходящие возможности систем, построенных на реляционных базах [1–3]. Некоторое увеличение объемов обрабатываемых данных может быть достигнуто без наращивания используемых аппаратных мощностей путем использования алгоритмов компрессии данных, специфичных для каждой рассматриваемой предметной области [4–6]. Храня данные в сжатом формате, можно даже получить прирост быстродействия всей системы, поскольку для некоторых хранилищ становится возможным полное хранение данных в оперативной памяти вместо использования более медленных жестких дисков. В случае использования распределенной системы возникают задержки, связанные с выполнением сетевых операций, в результате чего становится возможной ситуация, когда сжатое хранение данных на одной машине оказывается более выгодным по сравнению с распределенным хранением несжатых данных.

Сжатие данных в базах общего назначения либо не производится вовсе, либо выполняется одним из стандартных механизмов, как это происходит в GBASE [7]. В то же время для графов с определенной структурой атрибутов связей можно предложить более эффективные методы компрессии данных, позволяющие лучше учитывать априорную структуру поступающих данных. Результатом становится более чем двукратное повышение плотности хранения данных [4].

Анализ социальных сетей является одной из самых актуальных задач в области обработки графов [8, 9]. По своей сути данные, полученные из социальной сети, являются данными о взаимосвязях большого числа объектов, каждый из которых имеет некоторое число атрибутов, содержащих данные различного типа. Связи, возникающие между пользователями, могут не иметь дополнительных атрибутов (дружба пользователей социальной сети или наличие подписки одного пользователя на новостной контент другого), при этом связи другого типа могут нести в себе достаточно большой объем текстовой информации (комментарии одного пользователя к сообщениям другого). В задачу графовой базы данных, обеспечивающей хранение графа социальной сети, входит в том числе и создание полнотекстовых индексов, позволяющих осуществлять поиск в такого рода данных.

Среди задач, ориентированных на анализ графовой структуры социальной сети, можно выделить

[1] Работа выполнена при поддержке РФФИ, гранты №16-29-09546 и №16-07-00641.

следующие [6, 12]: 1) анализ источников возникновения некоторой информации; 2) поиск узлов с наивысшими показателями цитирования по заданной рубрике; 3) определение путей и каналов распространения информации по заданной рубрике.

Среди базовых операций, наиболее востребованных в такого рода анализе, особое внимание уделяется: 1) операции выгрузки подграфа, задаваемого окрестностью некоторого множества вершин [11] (такого рода данные требуются для последующей визуализации и детального анализа [12]); 2) поиску путей между двумя заданными множествами вершин [10].

Данные, размещенные в социальной сети, обладают высокой степенью изменчивости и постоянно обновляются. Для получения доступа к актуальной информации требуется постоянно перестраивать хранилище данных, используемое для хранения такого рода графов. В то же время структура данных, оптимальная для выполнения поисковых операций, может быть слишком медленной с точки зрения обеспечения скорости пополнения данных.

Ц е л ь  р а б о т ы – предложить схему хранения графов с использованием промежуточной стадии хранения. В момент построения графа используется схема данных, оптимальная для выполнения операций добавления вершин и связей. В то же время операции выгрузки подграфов и поиска путей могут выполняться параллельно с процессом построения хранилища, однако их скорость будет существенно ниже, чем при переводе хранилища в финальную стадию хранения. Этот режим используется для архивного хранения полученного графа. Время выполнения процесса перестроения линейно зависит от размера графа. В архивном режиме операции пополнения выполняются медленнее, однако операции поиска и выгрузки существенным образом ускоряются.

## Используемые определения

Множества вершин и связей хранимого графа $G$ обозначим через $V(G)$ и $E(G)$ соответственно. Для каждой связи $e$ из множества $E(G)$ определены в качестве атрибута функции $T(e)$ (например, соответствующая времени взаимодействия объектов, которые соединены данной связью) и $C(e)$ – тип взаимодействия. Множество допустимых типов взаимодействия обозначим через $S$.

Для атрибутов вершин определен набор доменов $A_1, A_2, \ldots, A_D$. Каждая вершина $v$ из множества $V(G)$ имеет один или несколько атрибутов, принимающих значения из доменов $A_1, A_2, \ldots, A_D$. Один из атрибутов $k(v)$ считается ключевым, его значение должно уникальным образом идентифицировать вершину. Остальные атрибуты вершины обозначим через $a_1(v), \ldots, a_{N(v)}(v)$. Каждый из них принимает значение в одном из доменов $A_1, A_2, \ldots, A_D$. Допускается, чтобы два и более атрибутов принимали значения в одном и том же домене.

На всех доменах считается определенной функция хэширования *HASH* на множество натуральных чисел и ноль:

$$HASH : A_i \to \{0, 1, \ldots\}. \tag{1}$$

Используя значение, получаемое для ключевого атрибута, можно считать функцию *HASH* определенной на множестве вершин:

$$HASH(v) = HASH(k(v)). \tag{2}$$

Если в качестве $k(v)$ выступает некоторый числовой идентификатор, то в качестве хэширующей функции может быть взято тождественное отображение, поскольку данный идентификатор уже задан на домене натуральных чисел.

Считается определенной функция *COMBINE*, хэширующая несколько положительных целых чисел в одно:

$$COMBINE: (\mathbb{N} \cup \{0\}) \times (\mathbb{N} \cup \{0\}) \to (\mathbb{N} \cup \{0\}). \tag{3}$$

## Схема хранения связей в режиме 1

Множество вершин $V$ разбивается на $M$ кластеров $V_0, V_1, \ldots, V_{M-1}$ по значениям функции *HASH*:

$$V_i = \left\{ v \in V : HASH(v)\%M = i \right\}. \quad (4)$$

Хранение связей кластера $V_i$ на диске выполняется совместно в виде объединенных в двунаправленный список блоков:

$$L_i = \left\{ B_j^i \right\}, \quad j = 1, \ldots, N_i. \quad (5)$$

Каждый блок содержит набор записей, включающих следующие поля: $k\left(v_1^m\right)$ – ключевой атрибут первой вершины связи; $k\left(v_2^m\right)$ – ключевой атрибут второй вершины связи; $T^m$ – атрибут связи (например, время взаимодействия); $C^m$ – тип связи. При этом вершина $v_1^m$ обязана принадлежать кластеру $V_i$.

Хранение блоков списка $L_i$ выполняется внутри некоторого файла файловой системы сервера. Все блоки лежат в интервале адресного пространства файла заранее определенного размера.

Каждый блок $B_j^i$ состоит из следующих элементов:

$$B_j^i = \left[ R(i,j), L(i,j), Mask_j^i, t_{\min}^{ij}, t_{\max}^{ij}, \bigcup\nolimits_{l=1,\ldots,N_{ij}} C_j^{il} \right], \quad (6)$$

где $\left\{ C_j^{il} \right\}$ – набор независимых подразделов произвольного размера; $R(i,j)$ – указатель на предыдущий блок списка; $L(i,j)$ – указатель на следующий блок списка; $Mask_j^i$ – битовая маска фиксированного размера $N_M$, при добавлении очередной связи в ней устанавливается бит с номером $HASH\left(v_1^m\right)\%N_M$; $t_{\min}^{ij}, t_{\max}^{ij}$ – границы интервала значений атрибута, содержащие минимальное и максимальное значение атрибута по всем вошедшим в данный блок записям.

Хранение элементов $R(i,j), L(i,j), M_j^i, t_{\min}^{ij}, t_{\max}^{ij}$ выполняется в оперативной памяти компьютера. Битовая маска $Mask_j^i$ используется для быстрой фильтрации блоков, если нужно получить все связи заданной вершины $v$. В таком случае можно пропустить блоки, в битовой маске которых не установлен бит $HASH(v)\%N_M$.

Каждый подраздел $C_j^{il}$ формируется по мере добавления связей, относящихся к вершинам кластера $V_i$. Каждый кластер получает в оперативной памяти буфер фиксированного размера. Добавляемые связи, относящиеся к вершинам кластера $V_i$, хранятся в буфере. По мере заполнения буфера выполняется его сжатие алгоритмом, описанным в [4]. На основе сжатых данных происходит формирование очередного подраздела $C_j^{il}$. При этом к сжатым данным дописываются поля $t_{\min}^{ijl}, t_{\max}^{ijl}$, определяемые как минимальное и максимальное значение атрибута по всем связям, вошедшим в данный буфер. Сформированный подраздел должен быть добавлен в последний блок двунаправленного списка. В случае, если в последнем блоке недостаточно свободного места, происходит выделение очередного блока, который добавляется в конец списка $L_i$. При добавлении очередного подраздела в блоке производится обновление элементов $Mask_j^i, t_{\min}^{ij}, t_{\max}^{ij}$.

### Схема хранения связей в режиме 2

Связи каждой вершины $v$ хранятся на жестком диске подряд в виде массива $L^v$, содержащего те же записи, что и в режиме 1, за исключением поля $k\left(v_1^m\right)$, необходимость в котором отпадает, так как все связи списка имеют в качестве исходящей вершины одну и ту же вершину $v$. Сжатие данных в списках используется в случае необходимости, однако сжимать списки слишком маленького размера нецелесообразно, поскольку экономия дискового пространства становится слишком незначительной. В такой си-

туации возможно использование схемы с объединением нескольких небольших списков в один с последующим их сжатием. Вторым компонентом в данном режиме является структура данных *Shift*, позволяющая по каждой вершине $v$ получать адрес списка $L^v$.

Выбор используемой структуры *Shift* зависит от общего числа различных вершин и суммарного размера их ключевых атрибутов. В случае, если все атрибуты помещаются в оперативной памяти, целесообразно использовать хэш-таблицу. Если же число вершин слишком велико, используется B-дерево. В таком случае адрес каждого списка может быть получен по вершине в среднем за одно обращение к дисковой памяти.

Такого рода структура позволяет получать доступ к списку $L^v$ за одну-две дисковых операции для каждой вершины. Кроме того, в случае, если используется сжатие, нет необходимости обрабатывать данные всех вершин того кластера, куда была отнесена данная вершина. Вследствие этого операции, связанные с поиском в ширину или обходом графа, выполняются значительно быстрее. Однако пополнение такого рода структуры новыми связями, как правило, требует ее полного перестроения, что является неприемлемым при работе с активно пополняемыми данными.

### Алгоритм перестроения хранилища

Для перехода от одной структуры данных к другой можно преобразовывать списки $L_i$ независимо. Как правило, емкости оперативной памяти достаточно для обработки всего списка целиком, в противном случае преобразование можно будет выполнить в несколько проходов (обрабатывая за один раз лишь порцию вершин). Для преобразования данных требуется сформировать список связей каждой вершины кластера, разместить его на диске и сохранить адрес списка в структуре *Shift*. Без учета времени сжатия данных вся операция требует линейного времени от размера списка $L_i$, поскольку декомпрессия данных выполняется для подразделов $C_j^{il}$, размер которых ограничен сверху, а, следовательно, время декомпрессии также ограничено некоторой константой.

- Эффективность предложенной структуры подтверждена экспериментами с данными социальной сети ВКонтакте. Предложенная структура позволяет обеспечивать быстрое построение хранилища при использовании схемы хранения связей в режиме 1. В то же время при переходе к режиму 2 можно добиться существенного роста скорости выполнения базовых операций с хранилищем. Универсальным решением является использование *пары хранилищ*: 1) *основного архивного*, находящегося в режиме 1; 2) *небольшого динамического*, находящегося в режиме 2.

  При этом динамическое хранилище служит своего рода буфером для аккумуляции поступающих данных. По мере того как его размер превышает некоторое пороговое значение, требуется запускать процесс перестроения основного хранилища для переноса всех данных из динамического хранилища в архивное.

  Предложенная схема позволяет строить гибкие хранилища данных социальной сети, обеспечивающие возможность постоянного пополнения наряду с доступом к сравнительно быстрым базовым операциям выгрузки подграфов и поиска путей между группами вершин.

### Литература


1. *Angles R.* A Comparison of Current Graph Database Models // Proc. of the 2012 IEEE 28th International Conference on Data Engineering Workshops (ICDEW '12). IEEE Computer Society. Washington, DC, USA. 2012. P. 171−177.
2. *Shalini Batra, Charu Tyagi* Comparative Analysis of Relational and Graph Databases // International Journal of Soft Computing and Engineering (IJSCE). 2012. V. 2. № 2. P. 509−512.
3. *Shrinivas S.G. et al*. Applications of Graph Theory in Computer Science an Overview // International Journal of Engineering Science and Technology. 2010. V. 9. P. 4610−4621.
4. *Поляков И.В.*, *Чеповский А.А.*, *Чеповский А.М.* Сжатие данных в хранилище больших графов // Фундаментальная и прикладная математика. 2016. Т. 21. № 4. С. 125−132.
5. *Поляков И.В.*, *Чеповский А.А.* Буферизация и сжатие данных при хранении мультиграфа // Труды Междунар. научной конф. Московского физико-технического института (государственного университета) и Института физико-технической информатики (SCVRT1516). М., Протвино: Институт физико-технической информатики. 2016. С. 76−78.

6. *Поляков И.В.*, *Чеповский А.А.*, *Чеповский А.М.* Хранение и обработка графа социальных сетей // Вестник НГУ. Сер.: Информационные технологии. 2013. Т. 11. № 4. С. 77–83.
7. *Kang U.*, *Hanghang Tong, Jimeng Sun, Ching-Yung Lin, Christos Faloutsos* GBASE: a scalable and general graph management system // Proc. of the 17th ACM SIGKDD International conference on Knowledge discovery and data mining. August 21–24, 2011. San Diego, California, USA.
8. *Базенков Н.И.*, *Губанов Д.А.* Обзор информационных систем анализа социальных сетей // Сб. трудов «Управление большими системами». 2013. С. 357–394.
9. *Батура Т.В.* Методы анализа компьютерных социальных сетей // Вестник НГУ. Сер.: Информационные технологии. 2012. Т. 10. № 4. С. 13–28.
10. *Поляков И.В.*, *Чеповский А.А.*, *Чеповский А.М.* Алгоритмы поиска путей на графах большого размера // Фундаментальная и прикладная математика. 2014. Т. 19. № 1. С. 165–172.
11. *Коломейченко М.И.*, *Поляков И.В.*, *Чеповский А.А.*, *Чеповский А.М.* Выделение сообществ в графе взаимодействующих объектов // Фундаментальная и прикладная математика. 2016. Т. 21. № 3. С. 131–139.
12. *Коломейченко М.И.*, *Поляков И.В.*, *Чеповский А.А.*, *Чеповский А.М.* Методы визуального анализа графов. М.: Национальный открытый университет «ИНТУИТ». 2016.

**Поступила 3 августа 2018 г.**

# Features of social network graph storage



**I.V. Polyakov** – Ph.D.(Phys.-Math.), Associate Professor, HSE (Moscow)
E-mail: ipolyakov@hse.ru
**V.O. Polyakova** – Student, HSE (Moscow)
E-mail: igorp86@mail.ru
**A.A. Chepovskiy** – Ph.D.(Phys.-Math.), Associate Professor, HSE (Moscow)
E-mail: aachepovsky@hse.ru

In this paper we propose different approaches for compact storaging of big social graphs. The joint use of two types of storages makes it possible to obtain a flexible scheme that is oriented both to the fast completion of the graph and to fast searches for the paths and extracting of subgraphs. The effectiveness of the proposed structure is confirmed by experiments with data from the social network VKontakte.

**References**

1. *Angles R.* A Comparison of Current Graph Database Models // Proc. of the 2012 IEEE 28th International Conference on Data Engineering Workshops (ICDEW '12). IEEE Computer Society. Washington, DC, USA. 2012. P. 171–177.
2. *Shalini Batra, Charu Tyagi* Comparative Analysis of Relational and Graph Databases // International Journal of Soft Computing and Engineering (IJSCE). 2012. V. 2. № 2. P. 509–512.
3. *Shrinivas S.G. et al.* Applications of Graph Theory in Computer Science an Overview // International Journal of Engineering Science and Technology. 2010. V. 9. P. 4610–4621.
4. *Polyakov I.V., Chepovskiy A.A., Chepovskiy A.M.* Sgatie dannikh v khranilishe bolshikh graphov // Fundamentalnaya I prikladnaya matematika. 2016. V. 21. № 4. S. 125–132.
5. *Polyakov I.V., Chepovskiy A.A.* Buferizaciya i sgatie dannikh pri khranenii multigrapha // Trudi Megdunar. nauchnoy konf. Moskovskogo phiziko-tekhnicheskogo instituta (gosudarstvenniy universitet) и Instituta phiziko-tekhnicheskoy informatiki (SCVRT1516). M., Protvino: Institut phiziko-tekhnicheskoy informatiki. 2016. S. 76–78.
6. *Polyakov I.V., Chepovskiy A.A., Chepovskiy A.M.* Khranenie I obrabotka grapha socialnikh setei // Vestnik NGU. Ser.: Informacionnie tekchnologii. 2013. V. 11. № 4. S. 77–83.
7. *Kang U.*, *Hanghang Tong, Jimeng Sun, Ching-Yung Lin, Christos Faloutsos* GBASE: a scalable and general graph management system // Proc. of the 17th ACM SIGKDD International conference on Knowledge discovery and data mining. August 21–24, 2011. San Diego, California, USA.
8. *Bazenkov N.I., Gubanov D.A.* Obzor informacionnikh system analiza socialnikh setei // Sb. Trudov «Upravlenie bolshimi sistemami». 2013. S. 357–394.
9. *Batur T.V.* Metodi analyza komputernikh socialnikh setei // Vestnik NGU. Ser.: Informacionnie tekchnologii. 2012. V. 10. № 4. S. 13–28.
10. *Polyakov I.V., Chepovskiy A.A., Chepovskiy A.M.* Algoritmi poiska putey na graphakh bolshogo razmera // Fundamentalnaya I prikladnaya matematika. 2014. V. 19. № 1. S. 165–172.
11. *Kolomeychenko M.I., Polyakov I.V., Chepovskiy A.A., Chepovskiy A.M.* Vy'delenie soobshhestv v grafe vzaimodejstvuyushhix ob^ektov // Fundamentalnaya I prikladnaya matematika. 2016. V. 21. № 3. S. 131–139.
12. *Kolomeychenko M.I., Polyakov I.V., Chepovskiy A.A., Chepovskiy A.M.* Metodi visualnogo analyza graphov. M.: Nacionalniy otkritiy universitet «INTUIT». 2016.

УДК 004.932.4

# Отпечатки пальцев как биометрический признак. Алгоритмы обработки и сегментации дактилоскопических изображений



**А.Г. Грижебовская** – аспирант, механико-математический факультет, МГУ им. М.В. Ломоносова
E-mail: a_gree@mail.ru

Рассмотрены вопросы обработки изображений отпечатков пальцев, полученных с биометрического сканера. Разработан и реализован алгоритм нахождения отдельных сегментов отпечатков пальцев для изображения четырех пальцев руки или двух больших. Показано, что данный алгоритм выделяет изображения отдельных сегментов, их координаты, угол поворота. Отмечено, что алгоритм использует двухступенчатый метод анализа, что позволяет повысить точность выделения отдельных сегментов, также дополнительно используется анализ геометрических характеристик сегментов. Выявлено, что алгоритм устойчиво работает при больших углах поворота и при наличии артефактов на изображении, таких как: следы пальцев, загрязненность. Алгоритм тестировался на базе 1000 изображений четырех пальцев руки и 500 изображений, содержащих два больших пальца.

***Ключевые слова:*** *отпечатки пальцев, сегментация, когерентность, биометрическая система.*

The processing of fingerprint images that received from the biometric scanner is considered. Four and two slap segmentation algorithm is developed and implemented. This algorithm finds the images of separate segments, their coordinates and an angle of rotation. The algorithm uses approximate and more exact analysis at different stages. It allows increasing the accuracy of placement of separate segments. The analysis of geometric characteristics of segments is used. The algorithm was tested on the database consisting of 1000 images of a hand and 500 images containing two thumbs.

***Keywords:*** *fingerprint, segmentation, four slap fingerprint, coherence, biometric system.*

**DOI: 10.18127/j20729472-201803-11**

Отпечатки пальцев являются одним из биометрических признаков, уникальной физической характеристикой человека. Отпечатки пальцев, линии папиллярных узоров не меняются с течением времени и выступают как стабильный признак [1]. Распознавание отпечатков может включать в себя несколько разных стадий, при этом первой стадией является получение изображения с биометрического сканера или устройства. Полученное с биометрического сканера изображение может дополнительно обрабатываться или сохраняться в базе данных, а также может преобразовываться в шаблон специального вида, который направляется на распознавание или сохраняется.

Сканеры отпечатков пальцев могут иметь разную область захвата. Существуют сканеры, которые позволяют получить цифровое изображение не только одного, а также двух или четырех пальцев одной руки. Устройства, позволяющие получить изображение четырех пальцев левой или правой руки, обычно имеют размер 2×3 или 3×3 дюйма и разрешение 500 dpi (dots per inch – точек на дюйм). Под сегментацией отпечатков пальцев подразумевается разделение на область пальца и область фона, либо выделение областей отдельных пальцев на изображении, содержащем несколько пальцев [1, 2].

Цель работы – рассмотреть сегментацию четырех отпечатков пальцев одной руки: указательного, среднего, безымянного и мизинца, а также выделение отпечатков больших пальцев рук. Такая сегментация может производиться в автоматическом режиме, отдельные сегменты сохраняются в базе данных и могут использоваться для распознавания в дальнейшем. Сегментация используется при паспортно-визовом контроле, учете миграции населения, в системах контроля доступа.

### Алгоритм сегментации двух-четырех отпечатков пальцев

Предложенный алгоритм сегментации отпечатков пальцев состоит из двух этапов: первый этап включает в себя первичное нахождение угла поворота руки и зон отпечатков пальцев; на втором этапе вычисляются углы поворота отдельных сегментов, координаты сегментов рассчитываются с большей точностью. Таким образом, вначале определяется общий угол поворота и координаты

сегментов с меньшей точностью, далее анализируются полученные отдельные сегменты и более точно определяются параметры. Применяется обработка изображения по области 8×8 пикселей [3, 4], используются параметры яркости и контраста изображения. Пусть $I$ – яркость в каждой точке. Для блока $W$ размером 8×8, где $n = 8$, вычисляются следующие величины:

$$Mean = \frac{1}{n \cdot n} \sum I \,, \tag{1}$$

$$Var = \frac{1}{n \cdot n} \sum \left(I - Mean\right)^2 \,, \tag{2}$$

$$Coh = \frac{\sqrt{\left(G_{xx} - G_{yy}\right)^2 + 4G_{xx}^2}}{G_{xx} + G_{yy}} \,, \tag{3}$$

где $\left(G_x, G_y\right)$ – локальный градиент; $G_{xx} = \sum_w G_x^2$; $G_{yy} = \sum_w G_y^2$; $G_{xy} = \sum_w G_x G_y$; *Mean* – средняя яркость; *Var* – дисперсия; *Coh* – когерентность для сегмента 8×8 пикселей [1, 4–6].

На первом шаге применяется бинаризация по усредненному порогу яркости и контрасту, используются характеристики *Mean* и *Var* для блоков точек *W* по (1) и (2). Для бинаризованного изображения находим оптимальный угол поворота всех пальцев, используя метод проекций по разным направлениям с шагом 3°. Метод проекций описан в работе [3]. Далее анализируется форма, число, координаты и размеры объектов на бинарном изображении, объекты малого размера удаляются. Выделяется до четырех объектов на изображении. Используются морфологические операции открытия и закрытия для удаления дефектов формы на бинаризованном изображении. Полученные отдельные сегменты отпечатков пальцев обрабатываются дополнительно, для анализа используется полутоновое изображение областей.

Для нахождения углов поворота отдельных сегментов в дополнение к углу, полученному на первом этапе обработки изображения, используется нахождение центра масс и вычисление центральных моментов второго порядка. Для более точного нахождения координат сегментов используется анализ когерентности малых областей методом анализа градиентов или полей направлений, которые ортогональны градиентам [1, 5]. По формуле (3) вычисляется когерентность и более точно определяются зоны отпечатков пальцев, координаты центров и размеры сегментов. Таким образом, на первом этапе обрабатывается полутоновое изображение полностью, а на втором этапе повторно обрабатываются области полученных сегментов. Такой алгоритм позволяет более точно вычислить углы и координаты сегментов, а также сократить объем вычислений и ускорить обработку.

Результаты сегментации представлены на рис. 1–3.

**Рис. 1.** Сегментация отпечатков левой руки

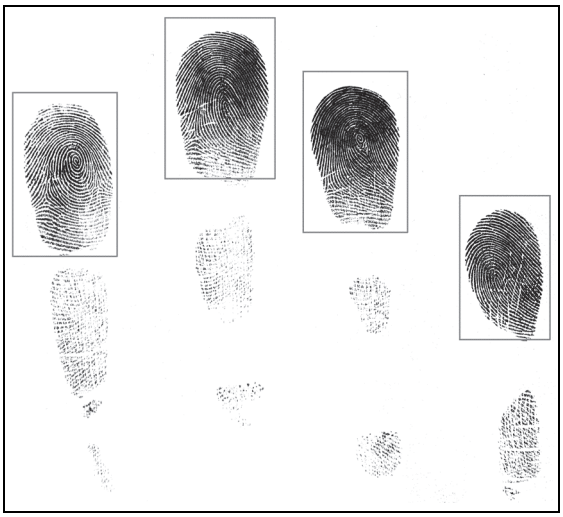

**Рис. 2.** Сегментация отпечатков правой руки

**Рис. 3.** Изображения сегментов отпечатков пальцев

- Предложенный алгоритм тестировался на базе отпечатков пальцев, полученных с биометрического сканера. Размер изображения 1600×1500 пикселей. Использовалась база отпечатков, состоящая из 1000 изображений пальцев правой и левой руки и 500 изображений больших пальцев рук. Анализировались разные случаи малоконтрастных изображений, различные повороты пальцев, случаи прикладывания пальцев под разными углами, зашумленные изображения, содержащие следы ранее приложенных пальцев, другие случаи. Алгоритм показывает устойчивую работу для углов поворота до 45°. Реализовано определение по изображению типа руки – левая или правая. Произведена оценка качества сегментов и загрязненности рабочей поверхности сканера.

### Литература


1. *Maltoni D.*, *Maio D.*, *Jain A.K.*, *Prabhakar S.* Handbook of Fingerprint Recognition. New York: Springer-Verlag. 2009. 510 p.
2. *Watson C.*, *Flanagan P.*, *Cochran B.* Slap Fingerprint Segmentation Evaluation II. Testing Procedure and Results. National Institute of Standards and Technology. NISTIR 7553. 2009. 164 p.
3. *Watson C.*, *Garris M.*, *Tabassi E.*, *Wilson C.*, *McCabe R.M.*, *Janet S.*, *Ko K.* User's Guide to NIST Biometric Image Software (NBIS). NISTIR 7392. 2009.
4. *Singh N.*, *Nigam A.*, *Gupta P.*, *Gupta P.* Four Slap Fingerprint Segmentation // ICIC 2012: Intelligent Computing Theories and Applications. 2012. P. 664−671.
5. *Гудков В.Ю.*, *Бойцов А.В.* Улучшение изображений отпечатков пальцев с помощью фильтра Габора // Вестник ЮУрГУ. Сер. «Компьютерные технологии, управление, радиоэлектроника». 2015. Т. 15. № 1. С. 128−132.
6. *Nimkar R.*, *Mishra A.* Fingerprint Segmentation Algorithms: A Literature Review // International Journal of Computer Applications (0975 – 8887). 2014. V. 95. № 5. P. 20−24.


**Поступила 3 августа 2018 г.**

# Biometric properties of fingerprints. Processing and slap fingerprint segmentation algorithms



**A.G. Grizhebovskaya** – Post-graduate Student, Faculty of Mechanics and Mathematics,
Lomonosov Moscow State University
E-mail: a_gree@mail.ru

The problems of processing images of fingerprints obtained from a biometric scanner are considered. An algorithm has been developed and implemented for finding individual segments of fingerprints for the image of four fingers of the hand or two large fingers. It is shown that this algorithm selects images of individual segments, their coordinates, and the angle of rotation. It is noted that the algorithm uses a two-step analysis method, which allows to improve the accuracy of individual segments, and also uses analysis of geometric characteristics of segments. It is revealed that the algorithm works steadily at large angles of rotation and in the presence of artifacts on the image, such as: fingerprints, contamination. The algorithm was tested on the basis of 1000 images of four fingers and 500 images containing two thumbs.

### References


1. *Maltoni D.*, *Maio D.*, *Jain A.K.*, *Prabhakar S.* Handbook of Fingerprint Recognition. New York: Springer-Verlag. 2009. 510 p.
2. *Watson C.*, *Flanagan P.*, *Cochran B.* Slap Fingerprint Segmentation Evaluation II. Testing Procedure and Results. National Institute of Standards and Technology. NISTIR 7553. 2009. 164 p.
3. *Watson C.*, *Garris M.*, *Tabassi E.*, *Wilson C.*, *McCabe R.M.*, *Janet S.*, *Ko K.* User's Guide to NIST Biometric Image Software (NBIS). NISTIR 7392. 2009.
4. *Singh N.*, *Nigam A.*, *Gupta P.*, *Gupta P.* Four Slap Fingerprint Segmentation // ICIC 2012: Intelligent Computing Theories and Applications. 2012. P. 664−671.
5. *Gudkov V.Yu.*, *Boytsov A.V.* Enhancement of fingerprint images using Gabor Filter // Bulletin of the South Ural State University. Series «Computer Technologies, Automatic Control, Radio Electronics». 2015. Т. 15. № 1. S. 128−132.
6. *Nimkar R.*, *Mishra A.* Fingerprint Segmentation Algorithms: A Literature Review // International Journal of Computer Applications (0975 – 8887). 2014. V. 95. № 5. P. 20−24.

УДК 004.89

# Мониторинг чрезвычайных происшествий с помощью анализа данных из социальных сетей[1]



**Д.А. Девяткин** – гл. специалист, ФИЦ «Информатика и управление» РАН (Москва)
E-mail: devyatkin@isa.ru
**А.О. Шелманов** – к.т.н., науч. сотрудник, ФИЦ «Информатика и управление» РАН (Москва)
E-mail: shelmanov@isa.ru
**Д.С. Ларионов** – студент, РУДН (Москва)
E-mail: dslarionov@protonmail.com

Исследовано создание прототипа системы для решения задач мониторинга чрезвычайных ситуаций (ЧС) в заданной географической области на основе анализа данных из социальных сетей. Рассмотрена архитектура системы, ее основные компоненты, а также лежащие в их основе методы сбора и анализа текстовой информации. Описаны методы сфокусированного сбора сообщений о ЧС из разнородных источников, извлечения информации из текстов, включая названия географических объектов и наименования морских и речных судов, классификации сообщений, а также методы обнаружения новых ЧС в потоке сообщений и их визуализации на географической карте.

***Ключевые слова:*** *извлечение информации из текстов, выявление событий, мониторинг, чрезвычайные ситуации, поисково-спасательные работы, выявление новых тем.*

The paper presents a prototype of a system for monitoring emergency events in a particular geographic region by analyzing social media data. We consider architecture, the main components of the system, as well as methods for crawling and processing emergency-related messages. The methods provide functionality for collecting emergency reports, information extraction, including the names of geographical locations and names of vessels, text classification, as well as new emergencies detection, and visualizing extracted events on a geographical map.

***Keywords:*** *information extraction, event detection, monitoring, emergencies, rescuer operations.*

**DOI: 10.18127/j20729472-201803-12**

Текстовые сообщения свидетелей и участников чрезвычайных ситуаций (ЧС) в социальных сетях часто содержат полезную информацию, которая, как было показано ранее [1], может использоваться при проведении поисково-спасательных работ (ПСР). Для извлечения и обработки этой информации необходимо создавать специализированные инструменты мониторинга ЧС, в которых реализованы методы сбора и анализа текстовых данных. Ранее было создано несколько систем и платформ анализа сообщений о ЧС [2], но большинство из них учитывают сообщения лишь одного источника данных (Twitter), а также работают только с текстами на английском языке.

Ц е л ь  р а б о т ы – создать прототип системы для решения задач мониторинга ЧС в заданной географической области. Данная работа является продолжением исследований, представленных в [2–4]. В отличие от предыдущих работ, в которых изучались вопросы выявления сообщений о ЧС, извлечения именованных сущностей и обнаружения новых ЧС в потоке сообщений, в настоящей работе рассматривается архитектура прототипа системы мониторинга ЧС, его структурные компоненты и их взаимодействие.

Создание подобных систем связано с необходимостью решить ряд следующих научно-технических проблем и задач:

1) небольшое число готовых лингвистических инструментов и корпусов для обучения, особенно для русского языка;

2) необходимость сфокусированного сбора информации, так как глубокая обработка всех текстовых сообщений, порождаемых в социальных сетях, труднореализуема;

3) разнородность источников данных (как короткие сообщения с преобладанием сленга и сокращений, как в Twitter, так и более длинные публикации, как в Facebook), разнородность программных интерфейсов (API) социальных сетей;

[1] Работа выполнена при поддержке Российского фонда фундаментальных исследований, проекты: 15-29-06045 «офи-м», 15-29-06031 «офи_м».

4) необходимость «выявлять» сообщения, посвященные новым ЧС, на фоне всех сообщений и обсуждений предыдущих ЧС;

5) разнородность ЧС и, как следствие, большое число тематик, связанных с ними.

## Описание прототипа системы мониторинга чрезвычайных ситуаций с помощью анализа социальных сетей

Прототип системы мониторинга чрезвычайных происшествий по данным социальных сетей состоит из трех основных подсистем: 1) *подсистемы сбора и извлечения информации из социальных сетей*; 2) *подсистемы выявления новых событий и загрузки информации из дополнительных источников*; 3) *подсистемы визуализации новых событий на географической карте*. Все перечисленные компоненты представляют собой автономные программные модули, совместно использующие хранилище извлеченной информации о ЧС. Схема подсистемы сбора и извлечения информации из социальных сетей представлена на рис. 1. В качестве основного источника данных прототипа системы выступает социальная сеть Twitter. Во многих работах [5] показана применимость этой социальной сети для решения задачи мониторинга ЧС. Сбор сообщений из других социальных сетей (Facebook) производится после обнаружения нового ЧС подсистемой выявления новых событий.

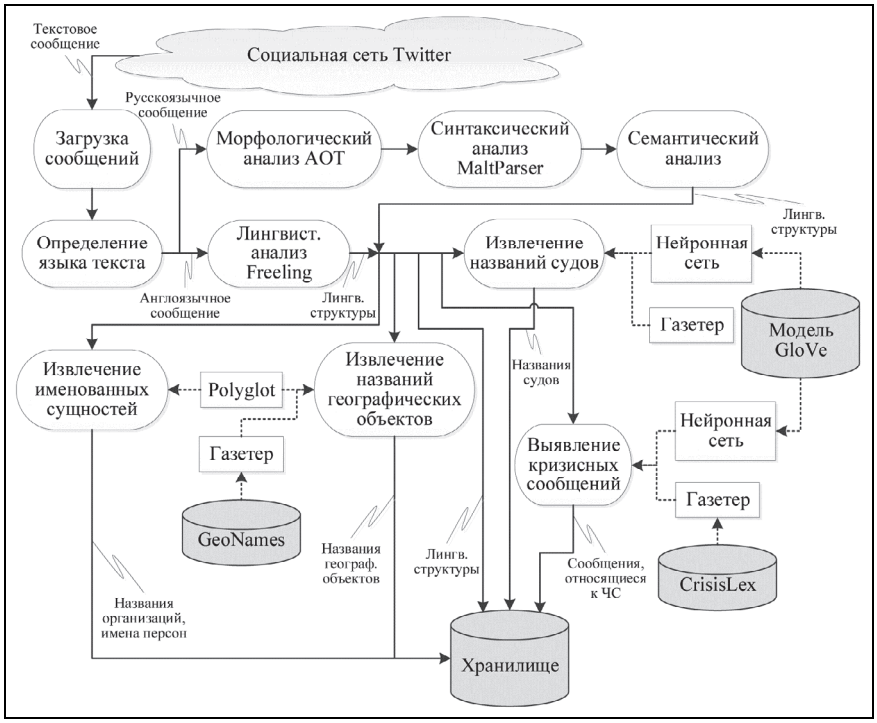


**Рис. 1.** Схема подсистемы сбора и извлечения информации из социальных сетей

После сбора сообщений с помощью API тематического поиска Twitter производится определение языка, на котором они написаны, затем выполняется полный лингвистический анализ сообщений, в зависимости от языка задействуются различные цепочки лингвистических анализаторов [6, 7]. С помощью предобученных классификаторов отфильтровываются сообщения, не посвященные ЧС [3]. Фильтрация сообщений происходит с помощью классификатора на основе сверточной нейронной сети с использованием векторных представлений слов, полученных с помощью метода FastText [8]. Комбинация классификатора и представления признаков была получена в ходе эмпирического сравнения на размеченном корпусе CrisisLexT6 [9] с другими классификаторами и представлениями признаков: случайным лесом, градиентным бустингом на деревьях решений, логистической регрессией.

Далее производится выявление именованных сущностей: наименований географических локаций, судов, организаций, персон. Для извлечения информации о локациях используется словарь локаций, сформированный на основе данных ресурсов GeoNames и CrisisLex. В дальнейшем для повышения качества извлечения локаций планируется обучение классификатора, учитывающего информацию о контексте употребления слов и словосочетаний. Информация также извлекалась с помощью инструмента Polyglot [10], обученного на базе знаний Freebase (http://www.freebase.com) и Википедии (https://wikipedia.org). Необходимо отметить, что этот метод обладает довольно низкой полнотой, однако совместное использование словарного метода и Polyglot позволяет добиться удовлетворительных результатов.

Затем тегированные сообщения о ЧС записываются в хранилище и индексируются поисковой машиной Exactus [11]. С использованием средств этой машины аналитик имеет возможность проводить

полнотекстовый поиск с учетом извлеченной информации.

Подсистема выявления новых событий и загрузки информации из дополнительных источников представлена на рис. 2.

На основе сообщений о ЧС за каждый день, извлекаемых из хранилища, обучается мультимодальная тематическая модель [12], выявляются новые тематики. В качестве дополнительной модальности в этой тематической модели используется лексика, относящаяся к географическим локациям. Для выявления новых тематик вычисляется дивергенция Дженсена–Шеннона [4] между новыми тематиками и тематиками, появлявшимися ранее. Далее с использованием ключевой лексики выявленных новых тематик производится сбор данных с Facebook. Затем эти сообщения отфильтровываются. Прямой мониторинг Facebook невозможен ввиду ограничений его API, поэтому собираются только открытые публикации по подтвержденным новым ЧС с помощью эмулятора браузера Ghost.py (http://ghost-py.readthedocs.io/en/latest/). Оставшиеся сообщения пополняют хранилище и индексируются поисковой машиной.

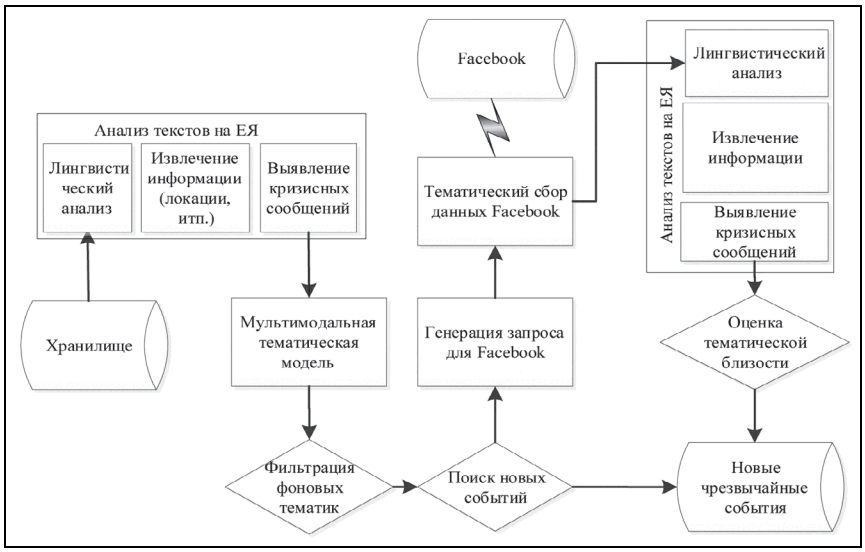


**Рис. 2.** Схема подсистемы выявления новых событий и загрузки информации из дополнительных источников

Результат работы подсистемы визуализации ЧС на географической карте представлен на рис. 3. Эта

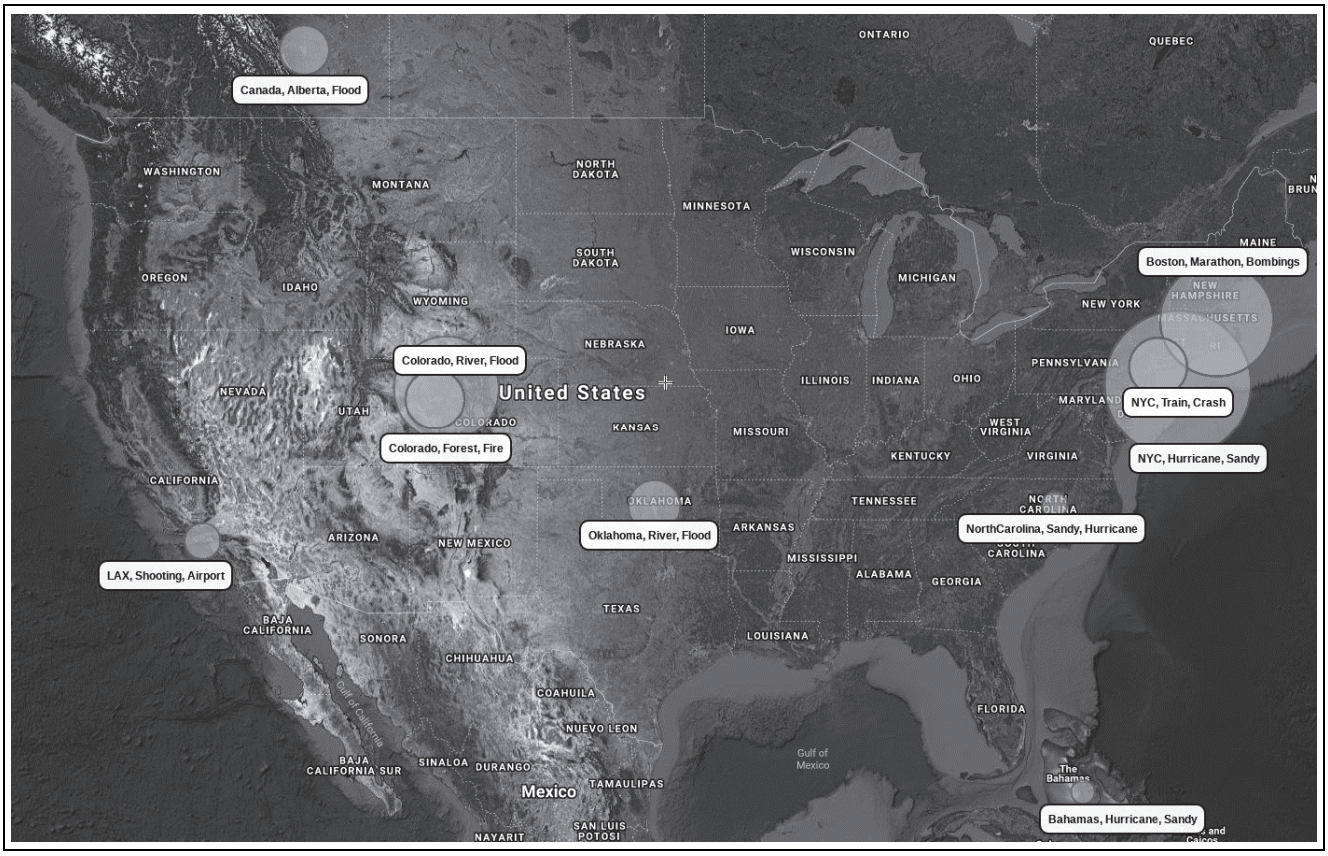


**Рис. 3.** Результаты работы подсистемы визуализации ЧС на географической карте

система основана на компоненте Scribblemaps (https://www.scribblemaps.com/api), позволяющем отображать произвольные векторные графические метки на географической карте.

Для построения метки используется несколько источников информации о географических локациях событий: географические координаты, указываемые непосредственно в сообщениях; наиболее частотная лексика, относящаяся к локациям и извлеченная из тематик, релевантных событию. Для определения географических координат по наименованию локации используется сервис Google Geocoding (https://developers.google.com/maps/documentation/geocoding/intro). Еще одним важным источников информации о локации являются данные автоматических навигационных систем (АНС) судов и систем автоматического независимого наблюдения вещания самолетов (АЗН-В), собираемые из открытых источников.

- Представлен прототип системы мониторинга ЧС по данным социальных сетей. Рассмотрена архитектура прототипа системы, ее основные компоненты, а также методы сбора и извлечения информации, лежащие в их основе.

  В качестве одной из возможных будущих функций системы предлагается рассмотреть оценку информативности сообщений, публикуемых в социальных сетях и других источниках. Оценка информативности могла бы быть полезна как при сборе данных, так и при расчете релевантности ответов при поиске информации в системе.

**Литература**


1. *Sixto J.*, *Pena O.*, *Klein B.*, *López-de Ipina D.* Enable tweet-geolocation and don't drive ERTs crazy! Improving situational awareness using Twitter // Proc. of SMERST. 2013. P. 27−31.
2. *Девяткин Д.А.*, *Шелманов А.О.* Анализ неструктурированных текстовых данных для поддержки поисково-спасательных работ // Системы высокой доступности. 2015. № 4. С. 45−60.
3. *Devyatkin D. and Shelmanov A.* Text processing framework for emergency event detection in the Arctic zone // Communications in Computer and Information Science. 2017. № 706. P. 74−88.
4. *Devyatkin D.*, *Shelmanov A. and Larionov D.* Discovering Novel Emergency Events in Text Streams // Proc. of the XX International Conference «Data Analytics and Management in Data Intensive Domains» (DAMDID/RCDL'2018). Moscow, Russia. October 9−12, 2018. [В печати].
5. *Ashktorab Z. et al*. Tweedr: Mining twitter to inform disaster response // ISCRAM. 2014.
6. *Carreras X. et al*. FreeLing: An Open-Source Suite of Language Analyzers // LREC. 2004. С. 239−242.
7. *Nivre J.*, *Hall J.*, *Nilsson J.* Maltparser: A data-driven parser-generator for dependency parsing // Proc. of LREC. 2006. Т. 6. С. 2216−2219.
8. *Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado and Jeff Dean* Distributed representations of words and phrases and their compositionality // Advances in neural information processing systems. 2013. P. 3111−3119.
9. *Alexandra Olteanu, Carlos Castillo, Fernando Diaz and Sarah Vieweg* CrisisLex: A lexicon for collecting and filtering microblogged communications in crises // Proc. of ICWSM. 2014.
10. *Al-Rfou R. et al*. Polyglot-NER: Massive multilingual named entity recognition // Proc. of the 2015 SIAM International Conference on Data Mining. Society for Industrial and Applied Mathematics. 2015. С. 586−594.
11. *Osipov G., Smirnov I., Tikhomirov I., Sochenkov I. and Shelmanov A.* Exactus expert search and analytical engine for research and development support // Novel Applications of Intelligent Systems. Springer. 2016. P. 269−285.
12. *Ianina A., Golitsyn L. and Vorontsov K.* Multi-objective topic modeling for exploratory search in tech news // Conference on Artificial Intelligence and Natural Language. Springer. 2017. P. 181−193.


**Поступила 3 августа 2018 г.**

# Monitoring of emergency events using social media



**D.A. Devyatkin** – Main Specialist, FRC «Computer Science and Control» of RAS (Moscow)
E-mail: devyatkin@isa.ru
**A.O. Shelmanov** – Ph.D.(Eng.), Research Scientist, FRC «Computer Science and Control» of RAS (Moscow)
E-mail: shelmanov@isa.ru
**D.S. Larionov** – Student, RUDN Univercity (Moscow)
E-mail: dslarionov@protonmail.com

The paper presents a prototype of a system for monitoring emergency events in a particular geographic region by analyzing social media data. We consider architecture, the main components of the system, as well as methods for crawling and processing emergency-related messages. The methods provide functionality for collecting emergency reports, information extraction, including the names of geographical locations and names of vessels, text classification, as well as new emergencies detection, and visualizing extracted events on a geographical map. As one of the possible future functions of the system, it is proposed to consider the evaluation of the informative nature of messages published in social networks and other sources. Evaluation of informativeness could be useful both in data collection and in the calculation of the relevance of answers when searching information in the system.

**References**

1. *Sixto J.*, *Pena O.*, *Klein B.*, *López-de Ipina D.* Enable tweet-geolocation and don't drive ERTs crazy! Improving situational awareness using Twitter // Proc. of SMERST. 2013. P. 27–31.
2. *Devyatkin D.A., Shelmanov A.O.* Unstructured data analysis to support rescue operations [Analiz nestructurirovannyh tekstovyh dannyh dlya podderzki poiskovo-spssatelnyh rabot] // High availability systems [Systemy vysokoi dostupnosti]. 2015. № 4. P. 45–60.
3. *Devyatkin D. and Shelmanov A.* Text processing framework for emergency event detection in the Arctic zone // Communications in Computer and Information Science. 2017. № 706. P. 74–88.
4. *Devyatkin D.*, *Shelmanov A. and Larionov D.* Discovering Novel Emergency Events in Text Streams // Proc. of the XX International Conference «Data Analytics and Management in Data Intensive Domains» (DAMDID/RCDL'2018). Moscow, Russia. October 9–12, 2018. [V pechati].
5. *Ashktorab Z. et al*. Tweedr: Mining twitter to inform disaster response // ISCRAM. 2014.
6. *Carreras X. et al*. FreeLing: An Open-Source Suite of Language Analyzers // LREC. 2004. P. 239–242.
7. *Nivre J.*, *Hall J.*, *Nilsson J.* Maltparser: A data-driven parser-generator for dependency parsing // Proc. of LREC. 2006. T. 6. P. 2216–2219.
8. *Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado and Jeff Dean* Distributed representations of words and phrases and their compositionality // Advances in neural information processing systems. 2013. P. 3111–3119.
9. *Alexandra Olteanu, Carlos Castillo, Fernando Diaz and Sarah Vieweg* CrisisLex: A lexicon for collecting and filtering microblogged communications in crises // Proc. of ICWSM. 2014.
10. *Al-Rfou R. et al*. Polyglot-NER: Massive multilingual named entity recognition // Proc. of the 2015 SIAM International Conference on Data Mining. Society for Industrial and Applied Mathematics. 2015. P. 586–594.
11. *Osipov G., Smirnov I., Tikhomirov I., Sochenkov I. and Shelmanov A.* Exactus expert search and analytical engine for research and development support // Novel Applications of Intelligent Systems. Springer. 2016. P. 269–285.
12. *Ianina A., Golitsyn L. and Vorontsov K.* Multi-objective topic modeling for exploratory search in tech news // Conference on Artificial Intelligence and Natural Language. Springer. 2017. P. 181–193.

УДК 81'33: 519.76

# Создание специальных корпусов текстов на основе расширенной платформы TXM[1]



**А.М. Лаврентьев** – к.ф.н., Институт истории представлений и идей нового времени НЦНИ и Высшей нормальной школы Лиона (Лион, Франция)
E-mail: alexei.lavrentev@ens-lyon.fr
**И.В. Смирнов** – к.ф.-м.н., зав. отделом, ФИЦ «Информатика и управление» РАН (Москва)
E-mail: ivs@isa.ru
**М.И. Суворова** – науч. сотрудник, ФИЦ «Информатика и управление» РАН (Москва)
E-mail: ananyeva@isa.ru
**Ф.Н. Соловьев** – науч. сотрудник, Институт физико-технической информатики (г. Протвино)
E-mail: the0@yandex.ru
**А.И. Фокина** – студент, НИУ ВШЭ (Москва)
E-mail: aifokina@edu.hse.ru
**А.М. Чеповский** – д.т.н., профессор, НИУ ВШЭ (Москва)
E-mail: achepovskiy@hse.ru

Рассмотрено расширение платформы TXM для корпусного анализа. Предложено использовать выделение псевдооснов в словах текста на базе метода структурных схем и выявление именных групп в структуре текста для выделения подкорпусов по параметрам. Описаны результаты анализа корпуса экстремистских текстов.

***Ключевые слова:*** *корпусная лингвистика, автоматический морфологический анализ, платформа TXM, анализ соответствий, специфичность, выявление экстремистских текстов.*

The extension of the TXM platform for case analysis is considered. It is proposed to use the allocation of pseudo-words in words of text on the basis of the method of structural schemes and the identification of nominal groups in the structure of the text for selecting subcorps in terms of parameters. The results of the analysis of the corpus of extremist texts are described.

***Keywords:*** *case linguistics, automatic morphological analysis, the TXM platform, the analysis of correspondences, specificity, the identification of extremist texts.*

**DOI: 10.18127/j20729472-201803-13**

Средства автоматизированного анализа текста находят свое применение в широком диапазоне как прикладных, так и исследовательских задач, таких как: поиск информации, коммуникация, ведение электронного документооборота, анализ потоков данных, классификация текстов, выявление языковых закономерностей, построение и применение моделей структуры языка. Особую значимость на сегодняшний день приобретает разработка методов и программных комплексов для интеллектуального анализа текстовых данных с целью выявления экстремистской и террористической направленности текстов на естественных языках.

Для облегчения и автоматизации анализа корпусов требуется программное обеспечение, позволяющее искать в корпусе различные языковые конструкции, сопоставлять их друг с другом, вычислять разнообразные статистические характеристики таких конструкций, параметрически выделять части корпуса для анализа и сравнения. Описание широкого ряда методов, применяемых для такого анализа, приведено в [5].

В настоящей работе рассматривается программный комплекс – платформа TXM [8], включающая в себя средства корпусного анализа. Предлагается ряд расширений, позволяющих дополнить и усложнить анализ корпусов, а именно: автоматический морфологический анализ словоформ и приведение их к канонической форме; выделение псевдооснов (аналога корня слова); выделение именных групп и комбинирование результатов работы предлагаемых расширений. Выделение псевдооснов в словах текста основано на использовании метода структурных схем. Такой подход позволяет анализировать текстовые конструкции, не только опираясь на точные словоформы, что в определенной степени ограничивает полноту и гибкость анализа, но и задавая более сложные запросы к корпусу, включающие в себя раз-

[1] Работа выполнена при поддержке РФФИ, гранты №16-29-09546 и №16-07-00641.

личные комбинации псевдооснов и грамматических характеристик слова. Метод структурных схем является хорошо масштабируемым подходом [5, 7] и позволит в будущем расширить поддержку средств корпусного анализа на другие языки.

Добавленные к платформе ТХМ средства анализа позволяют исследовать возможности применения различных дифференцирующих признаков для целей корпусного анализа и решения задач тематической классификации текстов, а также для создания обучающих выборок для задачи распознавания неструктурированных данных.

Ц е л ь  р а б о т ы – рассмотреть применение платформы ТХМ для создания и исследования корпусов текстов по специальным тематикам.

## Средства, расширяющие платформу ТХМ

С целью расширения возможностей корпусного анализа в процессе предобработки текстов при загрузке корпусов в платформу ТХМ были использованы разработанные авторами ранее следующие процедуры обработки текстов на естественном языке: выделение псевдооснов словоупотреблений; словарный морфологический анализ; выделение именных групп в тексте.

При анализе содержания текста полезно рассматривать различные словоформы одной лексемы как разные употребления одного и того же (в лексическом отношении) слова. Один из подходов [5] к решению этой задачи состоит в автоматическом морфологическом анализе словоформ с последующим приведением их к канонической форме. Возможность привести словоформу к канонической форме позволяет анализировать различные элементы словоизменительной парадигмы как одну и ту же структурную единицу текста. Это, в свою очередь, позволяет более корректно проводить содержательный статистический анализ текста, например, путем рассмотрения частот лексем вместо частот отдельных словоформ.

Используемая в настоящей работе реализация автоматического анализа словарной морфологии основана на словаре Зализняка [4]. В словаре содержатся словоформы вместе с их грамматическими пометами. Используемая морфологическая модель относит каждое слово к одной из 24 морфологических категорий, близких к части речи в традиционном понимании. Каждая из морфологических категорий характеризуется набором грамматических категорий: род, падеж, число, наклонение и др. Программная реализация [5] словарной морфологии русского языка опирается на представление словаря в специализированной структуре данных «бор», что позволяет осуществлять поиск словоформ за линейное по числу букв словоформы время. Каждая словоформа, хранящаяся таким образом, содержит свои грамматические характеристики, а также информацию о ее канонической (начальной) форме и об основе словоформы.

При анализе коротких текстов сети Интернет, таких как комментарии и сообщения, значительна доля неологизмов и жаргонизмов. По этой причине для анализа специальных корпусов текстов, помимо словарной морфологии, был использован аналитический метод выделения псевдооснов – аналога корня слов, что также позволило избежать анализа частот отдельных словоформ.

Для выделения псевдооснов используется метод структурных схем [3, 7], который заключается в получении псевдоосновы словоформы путем рассмотрения и отбрасывании ее аффиксов. Аффиксы подразделяются на типы в соответствии с их позицией относительно корня слова. Префиксы (или приставки) предшествуют корню, суффиксы – следуют за корнем. Пусть $P$ – множество приставок слов, включающее в себя пустую приставку, $A$ – множество суффиксов слов. Псевдоосновой называется часть слова, не содержащая суффиксов и приставок. Каждое слово описывается последовательностью типов аффиксов $s = \left( p,\, a_{i_1}, \ldots,\, a_{i_s} \right),\, p \in P,\, a_{ij} \in A$. Пусть $S_{\text{valid}}$ – множество таких последовательностей, допустимых в рассматриваемом (русском) языке. Элементы $S_{\text{valid}}$ называются структурами некорневой части слова. Отсюда и название метода. Если дана словоформа $w$ и известно $S_{\text{valid}}$, то псевдооснова слова получается отбрасыванием всех подходящих слову структур из $S_{\text{valid}}$ (то есть слово начинается с соответствующей приставки и заканчивается соответствующими суффиксами).

Дополнительную информацию о содержании текста можно почерпнуть, анализируя не только словоформы, но и целые именные группы. Именная группа – группа слов, у которой главное слово существительное, а другие слова связаны с ним подчинительными синтаксическими связями. Рассмотрение частотных именных групп и их сочетаний в совокупности с анализом отдельных словоупотреблений позволяет получить более полную картину содержания текста.

Используемый алгоритм описан в [5] и основан на анализе подчинительных синтаксических связей (в том числе и внутри конструкций с однородными членами) после того, как текст разделен на предложения. Определенную сложность при выделении именных групп представляет множественность морфологических разборов при омонимии. При анализе слов в предложении рассматривается все множество возможных морфологических разборов каждого слова. Подчинительные связи устанавливаются между парами слов в соответствии с правилами [5], где учитывается порядок слов в предложении.

Все построенные подчинительные связи образуют граф $G=(V,E)$, где множество вершин $V$ – множество слов предложения; $E=E_q \bigcup E_u$ – множество подчинительных связей; $E_q$ – множество связей между существительными; $E_u$ – множество связей, где прилагательное подчинено существительному. Именными группами являются пути $v_1 \ldots v_k$ в графе $G$, где: 1) если $k=2$, то $(v_1,v_2) \in E_q$; 2) если $k>2$, то в пути есть хотя бы одно ребро $(v_i,v_{i+1}) \in E_u$.

## Интеграция с платформой TXM

Платформа TXM [8] представляет собой программное обеспечение с открытым кодом, предназначенное для подготовки, обработки, анализа и публикации корпусов среднего размера (примерно до 10 000 000 словоупотреблений). TXM развивается в лаборатории IHRIM[2] Высшей нормальной школы Лиона и Национального центра научных исследований Франции и в лаборатории ELLIAD[3] университета Безансона. Особенностью платформы является гармоничное сочетание инструментов количественного и качественного анализа. От статистических инструментов (факторный анализ и специфичность [9]) всегда можно перейти к конкордансам значимых для анализа словоупотреблений.

Открытый код и модульная архитектура позволяют интегрировать в TXM внешние инструменты обработки и анализа текстовых данных.

В настоящей работе также используется программный пакет TreeTagger [11], предоставляющий возможность совместного морфологического анализа слов предложения на основе статистической модели путем сопоставления словоупотреблений специальных меток, кодирующих их морфологические характеристики. TreeTagger предоставляет возможность определения начальных форм слов предложения. Преимуществом пакета является однозначность морфологического анализа, поскольку при сопоставлении меток словоупотреблений учитываются все слова предложения в совокупности.

Все упомянутые средства были объединены в набор утилит, позволяющий вычислять для текстовых корпусов ряд характеристик: словоформы, встреченные в исходном корпусе; начальные формы слов по словарной морфологии; морфологические характеристики слов по словарной морфологии; начальные формы слов, полученные с помощью TreeTagger; морфологические характеристики, полученные с помощью TreeTagger; псевдоосновы слов; именные группы, составленные из словоформ, вместо отдельных словоупотреблений; именные группы, составленные из начальных форм, полученных из словарной морфологии, вместо отдельных словоупотреблений; именные группы, составленные из морфологических характеристик, полученных из словарной морфологии, вместо отдельных словоупотреблений; именные группы, составленные из начальных форм, полученных с помощью TreeTagger, вместо отельных словоупотреблений; именные группы, составленные из морфологических характеристик, вместо отдельных словоупотреблений; именные группы, составленные из псевдооснов, вместо отдельных словоупотреблений.

Корпуса с вычисленными характеристиками преобразуются в формат для импорта пакетом TXM.

## Анализ корпуса противоправных текстов

На исследуемом корпусе из 709 текстов противоправной и нейтральной тематики общим объемом 900 000 словоупотреблений, составленном в [1, 2], был проведен факторный анализ соответствий, реализуемый с использованием соответствующей функции TXM.

[2] Институт истории представлений и идей в новое время, http://ihrim.ens-lyon.fr.

[3] Издание, Литература, Языки, Информатика, Искусства, Дидактика и Дискурс, http://elliadd.univ-fcomte.fr.

Методика анализа соответствий, используемая ТХМ, была предложена Ж.-П. Бензекри [6] и реализована в пакете FactoMineR [10]. Подход анализа соответствий состоит в анализе частот совместного появления значений этих переменных на предмет равномерного распределения за счет поиска наиболее информативного представления таблицы частот совместного появления переменных $X$ и $Y$ в виде набора точек на плоскости, где точки отвечают либо значениям переменной $X$, либо значениям переменной $Y$.

Вследствие понижения размерности матрицы, проводимой в ходе анализа соответствий, изначально состоящей в данном случае из семи столбцов, соответствующих условному делению текстов экстремистской направленности на подкорпуса, есть возможность оценить их пространственное расположение относительно друг друга. Расположение вблизи пересечения осей координат интерпретируется как нейтральность подкорпуса по отношению к какому-либо из располагающихся вдали, то есть соответствующих экстремистским тематикам, в числе которых терроризм, идеология, религиозная ненависть, сепаратизм, национализм, агрессия и призывы к беспорядкам, фашизм. Отдаление от начала координат указывает на отклонение от нормальности в пользу одной из упомянутых выше противоправных тематик.

Анализ соответствий демонстрирует пространственное расположение подкорпусов на основе анализа частот совместного появления значений переменных. Результаты наглядно представлены в графической интерпретации факторного анализа соответствий.

На рис. 1 представлено пространственное расположение подкорпусов на основе именных групп, составленных из псевдооснов. Распределение точек соответствует частоте появления значений переменных в измерении, соответствующем оси координат, значения вариации которых равны 36% и 16% для осей 1 и 2 соответственно. «Нейтральный» подкорпус расположен на оси абсцисс чуть левее оси ординат. Из этого можно сделать вывод, что он действительно нейтрален по своему содержанию в сравнении с каждым из экстремистских подкорпусов. Измерения же подкорпусов противоправной тематики близки друг к другу по оси 1, отвечающей за 36% вариации в корпусе, которая является наиболее показательной по псевдоосновам для именных групп. Что касается оси 2, здесь распределение достаточно равномерно, что указывает на ряд характерных тематических отличий у подкорпусов по данной оси.

Схожую картину можно наблюдать и на рис. 2, где представлены результаты факторного анализа по псевдоосновам для отдельных слов: «нейтральный» подкорпус опять же находится в отдалении от каждого из экстремистских на оси 1 (горизонтальной). «Идеология», «религия», «терроризм», «национализм», «агрессия» и «сепаратизм» близки по своим показателям и образуют «облако» в первой четверти координатной плоскости. Подкорпус «фашизм», находящийся в четвертой четверти, близок ко всем экстремистским по оси 1 (46% вариации) и противостоит им по оси 2 (13% вариации).

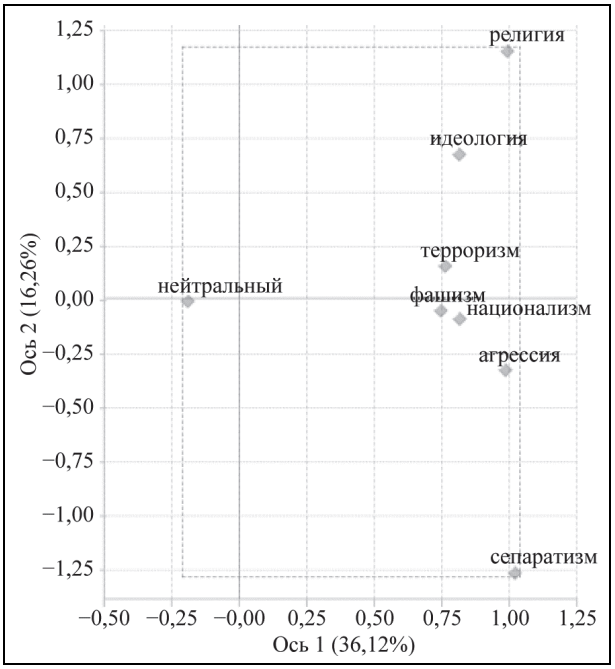


**Рис. 1.** Графическая интерпретация факторного анализа по именным группам, составленным из псевдооснов

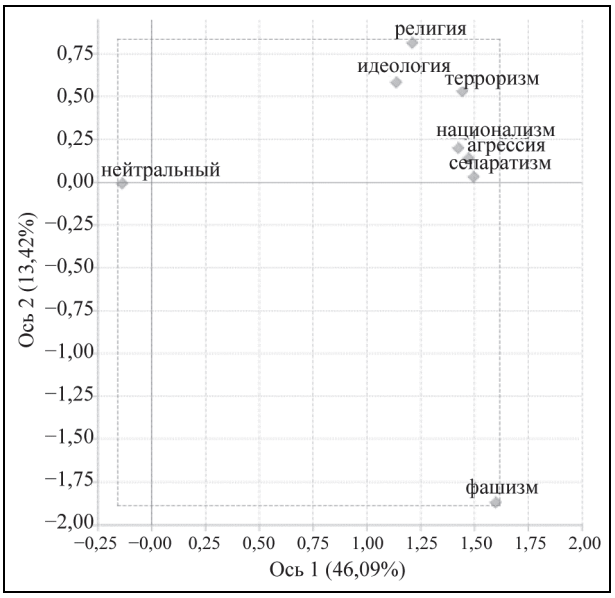


**Рис. 2.** Графическая интерпретация факторного анализа по псевдоосновам для отдельных слов

Если сравнивать результаты по факторному анализу по псевдоосновам для отдельных слов (рис. 2) и именных групп (рис. 1), то здесь наблюдается сходство, которое выражается в расположенности нейтральных текстов на оси абсцисс чуть левее оси ординат, в то время как противоправные тексты расположены правее оси 2 и рассредоточены относительно нее. Однако в анализе именных групп (рис. 1) нет обособленности подкорпуса «фашизм», напротив, он находится в центре относительно остальных противоправных подкорпусов. Из этого следует вывод, что в именных группах нет выраженной частотности во встречаемости фашистской тематики относительно всего корпуса.

- В результате интеграции инструментов автоматической обработки текста (выделение псевдооснов и именных групп, полный морфологический анализ) в платформу корпусного анализа TXM продемонстрирована возможность расширения статистического анализа текстов.

  Показано, что «нейтральный» подкорпус противостоит каждому из «экстремистских» по результатам анализа по предложенным авторами характеристикам. В силу выявленных особенностей и противопоставленности нейтрального подкорпуса остальным, сформированный корпус может быть использован для решения задач классификации текстов на предмет выявления потенциально противоправного содержания.

  Полученные результаты анализа корпуса текстов показывают возможность применять все рассмотренные факторы в качестве дифференцирующих признаков для тематической идентификации текстов, а средства платформы TXM в совокупности с предложенными расширениями могут применяться как средства выявления признаков и формирования обучающих выборок для задачи распознавания.

**Литература**


1. *Ананьева М.И.*, *Кобозева М.В.*, *Соловьев Ф.Н.*, *Поляков И.В.*, *Чеповский А.М.* О проблеме выявления экстремистской направленности в текстах // Вестник Новосибирского гос. ун-та. Сер.: Информационные технологии. Новосибирск: Новосибирский национальный исследовательский государственный университет. 2016. Т. 14. № 4. С. 5−13.
2. *Ананьева М.И.*, *Девяткин Д.А.*, *Кобозева М.В.*, *Смирнов И.В.*, *Соловьев Ф.Н.*, *Чеповский А.М.* Исследование характеристик текстов противоправного содержания // Труды Института системного анализа РАН. М.: ФИЦ ИУ РАН. 2017. Т. 67. № 3. С. 86−97.
3. *Болховитянов А.В.*, *Чеповский А.М.* Методы автоматического анализа словоформ // Информационные технологии. 2011. № 4(176). С. 24−29.
4. *Зализняк А.А.* Грамматический словарь русского языка. М.: Русский язык. 1977. 879 с.
5. *Чеповский А.М.* Информационные модели в задачах обработки текстов на естественных языках. Изд. 2-е, перераб. М.: Национальный открытый университет «ИНТУИТ». 2015.
6. *Benzécri J.-P.* L'analyse des données: l'analyse des Correspondances. V. 2. 2nd ed. Paris: Dunod. 1979.
7. *Egorova E.*, *Chepovskiy A.*, *Lavrentiev A.* A structural pattern based method for automated morphological analysis of word forms in a natural language // Journal of Mathematical Sciences. M.: Plenum Publishers. 2016. V. 214. № 6. P. 802−813.
8. *Heiden S.* The TXM Platorm: Building Open-Source Textual Analysis Sofware Compatile with the TEI Encoding Scheme // 24th Pacific Asia Conference on Language, Information and Computation – PACLIC24 / Ed. by *R. Otoguro*, *K. Ishikawa*, *H. Umemoto*, *K. Yoshimoto and Y. Harada*. Institute for Digital Enhancement of Cognitive Development. Waseda University, Sendai, Japan. 2010. P. 389−398. URL = http://halshs.archiies-ouiertes.fr/halshs-00549764.
9. *Lafon P.* Sur la variabilité de la fréquence des formes dans un corpus // Mots. 1980. № 1. P. 127−165.
10. *Lê S.*, *Josse J.*, & *Husson F.* FactoMineR: an R package for multivariate analysis // Journal of statistical software. 2008. № 25(1). P. 1−18.
11. *Schmid H.* Probabilistic Part-of-Speech Tagging Using Decision Trees // Proc. of International Conference on New Methods in Language Processing. Manchester, UK. 1994. URL = http://www.cis.uni-muenchen.de/sschmid/tools/TreeTagger/data/tree-tagger1.pdf.


**Поступила 3 августа 2018 г.**

# Creating text corpora for special purposes on the basis of extended TXM platform



**A.M. Lavrentiev** – Ph.D.(Philol.), CNRS & ENS de Lyon (France)
E-mail: alexei.lavrentev@ens-lyon.fr

**I.V. Smirnov** – Ph.D.(Phys.-Math.), Head of Department, FRC «Computer Science and Control» of RAS (Moscow)
E-mail: ivs@isa.ru
**M.I. Suvorova** – Research Scientist, FRC «Computer Science and Control» of RAS (Moscow)
E-mail: ananyeva@isa.ru
**F.N. Solov'ev** – Research Scientist, Institute of Physical and Technical Informatics (Protvino)
E-mail: the0@yandex.ru
**A.I. Fokina** – Student, HSE (Moscow)
E-mail: aifokina@edu.hse.ru
**A.M. Chepovskiy** – Dr.Sc.(Eng.), Professor, HSE (Moscow)
E-mail: achepovskiy@hse.ru

TXM platform suggests a wide range of corpus analysis capabilities including correspondence analysis, clusterization, lexical table construction, parametrized subcorpus selection. The default structural unit of analysis for the TXM platform is a token. However it is possible to supply each token with a number of features enabling more sophisticated, complex while flexible corpus analysis. The only extension available by default is the TreeTagger augmenting TXM platform with automated token morphological analysis capability. In this work we present a number of tools for even more extensive and complex corpus analysis relying both on our previously developed tools as well as on publicly available tools.

**References**

1. *Anan'eva M.I., Kobozeva M.V., Solov'ev F.N., Polyakov I.V., Chepovskiy A.M.* O probleme vyyavleniya ekstremistskoy napravlennosti v tekstakh // Vestnik Novosibirskogo gos. un-ta. Ser.: Informatsionnye tekhnologii. Novosibirsk: Novosibirskiy natsional'nyy issledovatel'skiy gosudarstvennyy universitet. 2016. T. 14. № 4. S. 5–13.
2. *Anan'eva M.I., Devyatkin D.A., Kobozeva M.V., Smirnov I.V., Solov'ev F.N., Chepovskiy A.M.* Issledovanie kharakteristik tekstov protivopravnogo soderzhaniya // Proc. of the Institute for Systemic Analysis of RAS. M.: FRC «Computer Science and Control» of RAS. 2017. T. 67. № 3. S. 86–97.
3. *Bolkhovityanov A.V., Chepovskiy A.M.* Metody avtomaticheskogo analiza slovoform // Informatsionnye tekhnologii. 2011. № 4(176). S. 24–29.
4. *Zaliznyak A.A.* Grammaticheskiy slovar' russkogo yazyka. M.: Russkiy yazyk. 1977. 879 s.
5. *Chepovskiy A.M.* Informatsionnye modeli v zadachakh obrabotki tekstov na estestvennykh yazykakh. Izd. 2-e, pererab. M.: Natsional'nyy otkrytyy universitet «INTUIT». 2015.
6. *Benzécri J.-P.* L'analyse des données: l'analyse des Correspondances. V. 2. 2nd ed. Paris: Dunod. 1979.
7. *Egorova E.*, *Chepovskiy A.*, *Lavrentiev A.* A structural pattern based method for automated morphological analysis of word forms in a natural language // Journal of Mathematical Sciences. M.: Plenum Publishers. 2016. V. 214. № 6. P. 802–813.
8. *Heiden S.* The TXM Platorm: Building Open-Source Textual Analysis Sofware Compatile with the TEI Encoding Scheme // 24th Pacific Asia Conference on Language, Information and Computation – PACLIC24 / Ed. by *R. Otoguro*, *K. Ishikawa*, *H. Umemoto*, *K. Yoshimoto and Y. Harada*. Institute for Digital Enhancement of Cognitive Development. Waseda University, Sendai, Japan. 2010. P. 389–398. URL = htp://halshs.archiies-ouiertes.fr/halshs-00549764.
9. *Lafon P.* Sur la variabilité de la fréquence des formes dans un corpus // Mots. 1980. № 1. P. 127–165.
10. *Lê S.*, *Josse J.*, & *Husson F.* FactoMineR: an R package for multivariate analysis // Journal of statistical software. 2008. № 25(1). P. 1–18.
11. *Schmid H.* Probabilistic Part-of-Speech Tagging Using Decision Trees // Proc. of International Conference on New Methods in Language Processing. Manchester, UK. 1994. URL = http://www.cis.uni-muenchen.de/sschmid/tools/TreeTagger/data/tree-tagger1.pdf.

УДК 004.8:519.178

# Анализ профилей сообществ социальных сетей[1]



**Т.В. Соколова** – аспирант, РТУ МИРЭА (Москва)
E-mail: sokolova.stv@mail.ru
**А.А. Чеповский** – к.ф.-м.н., доцент, НИУ ВШЭ (Москва)
E-mail: aachepovsky@hse.ru

Рассмотрен подход к формированию профиля пользователя на основе данных из социальных сетей, при котором используются данные как самого пользователя, так и его друзей. Показано, что при анализе друзей применяются алгоритмы выделения сообществ в графах для разбиения множества друзей пользователя на сообщества и группы общения, при этом для каждого сообщества формируется собственный профиль с использованием характеристик пользователей, которые в него входят, и групп в социальной сети, в которых эти пользователи состоят.

***Ключевые слова:*** *социальные сети, социальный граф, выделение сообществ, профиль пользователя, профиль сообщества.*

This paper presents the problem of forming user profiles based on data from social networks. For user profiling both the user and his friends data are used. Community allocation algorithms in social graphs are used to detect groups of communication. Each community has its own profile, which includes the characteristics of the users that belong to it and the groups that these users belong to.

***Keywords:*** *social networks, social graph, community detection, user profile, community profile.*

**DOI: 10.18127/j20729472-201803-14**

Ежедневно люди используют электронные устройства, выходят в Интернет, потребляют и публикуют информацию в сети, тем самым оставляя за собой цифровой след. По мере пользования устройствами и сервисами накапливаются данные, которые позволяют описать каждого из пользователей, выделить его отличительные черты, составить его портрет. Такой портрет будем называть *профилем пользователя*.

Профили пользователей формируются из фактических и поведенческих характеристик пользователей. Фактические характеристики пользователей – характеристики, которые пользователи сами о себе указывают. Поведенческих характеристики пользователей – характеристики, которые были получены в результате анализа поведения пользователей.

Профили пользователей широко применяются в различных сферах, например: в безопасности для выделения пользователей или групп пользователей, склонных к антигосударственной деятельности; в рекламе при показе персонализированных рекламных сообщений; в маркетинге при анализе поведения клиентов и их предпочтений; в банковской сфере при построении скоринговых моделей и принятии решения о выдаче кредита заемщику; в сфере управления персоналом при принятии решения о приеме кандидата на работу.

Выбор способа построения профиля сильно зависит от области их применения, исходных данных и характеристик пользователя, которые должны быть включены в итоговый профиль. Если, например, к данным мобильного оператора о звонках абонентов или историям покупок в Интернет-магазине имеет доступ ограниченный круг лиц, то открытые данные в социальных сетях доступны всем, поэтому рассмотрим тему социальных сетей более подробно.

Ц е л ь  р а б о т ы – рассмотреть подход к формированию профиля пользователя на основе данных из социальных сетей, при котором используются данные как самого пользователя, так и его друзей.

## Профили пользователей социальных сетей

Построение профиля пользователя на основе данных из социальных сетей имеет ряд особенностей. С одной стороны, широкий спектр данных может быть указан пользователем самостоятельно на странице в социальной сети. С другой стороны, эти данные заполняются неравномерно. Одни пользователи указывают всю возможную информации о себе, другие же предпочитают оставить большую часть полей незаполненными.

[1] Работа выполнена при поддержке РФФИ, гранты №16-29-09546 и № 16-07-00641.

В социальных сетях всегда присутствуют пользователи, которые не ведут активную деятельность в сети и не выступают в качестве создателей контента, предпочитая оставаться «молчаливыми» наблюдателями и оставляя минимальное количество информации в сети. Также некоторая часть пользователей скрывает информацию о себе и публикуемую информацию из общего доступа, используя настройки приватности. В этом случае профиль такого пользователя окажется достаточно скудным и будет содержать малое число характеристик.

Однако анализ друзей и последующее распространение их характеристик на пользователя позволяет дополнить портрет пользователя атрибутами, которые не были явно указаны или были скрыты в настройках от всеобщего обозрения. Например, в работах [3, 5, 6] рассматриваются различные методы распространения атрибутов на основе дружеских связей между пользователями и на основе групп, в которые входят эти пользователи.

Предлагаемая методика включает в себя: 1) скачивание социального графа; 2) выделение и визуализацию сообществ в социальном графе; 3) построение профилей сообществ и их анализ.

## Социальные графы

Связи между пользователями можно представить в виде графа, в котором в качестве узлов выступают пользователи социальной сети и их характеристики, а в качестве ребер – связи между пользователями. Такой граф еще называют социальным.

В рамках данной задачи интерес представляют ближайшее окружение пользователя, его непосредственные друзья и дружеские связи между ними. Это значит, что необходимо:

1) добавить исходного пользователя в граф;

2) получить список друзей пользователя из п.1. и добавить полученных пользователей и их связи с исходным пользователем в граф;

3) получить списки друзей пользователей из п.2. и дополнить граф дружескими связями между пользователями, которые уже были добавлены в граф на предыдущих этапах.

Для построения графа используется API социальной сети, который позволяет скачать: список друзей пользователя; открытые данные пользователя, которые указаны в его профиле в социальной сети; список групп, в которых пользователь состоит, и публичных страниц и мероприятий, на которые он подписан.

В то время как подписки на публичные страницы открыты у всех пользователей, список групп часто скрыт через настройки приватности. Его можно было бы частично восстановить, загрузив список пользователей всех групп в социальной сети. Однако это ресурсоемкий процесс, так что ограничимся проверкой только тех групп, которые известны про рассматриваемую группу пользователей. Поэтому после загрузки списка групп, в которых состоят пользователь и все его друзья, дополнительно проверяем, если ли среди их участников рассматриваемые пользователи. Тем самым частично восстанавливаем список групп для тех пользователей, у которых он скрыт в настройках.

## Сообщества пользователей

Для социальных графов характерно наличие сплоченных групп пользователей, иначе говоря, сообществ, для которых характерна высокая плотность связей внутри сообществ и низкая плотность связей между сообществами.

Часто образование таких сообществ происходит в силу естественных причин и событий в жизни человека. Самые большие и плотные сообщества часто соответствуют группе одноклассников в школе или сокурсников в университете. Сообщества могут образовываться из людей, которых объединяют общие увлечения. Например, это может быть группа любителей конного спорта, которая регулярно встречается на одной конюшне, или активные болельщики одного из футбольных клубов.

Алгоритмы выделения и визуализации сообществ, то есть групп, которые характеризуются высокой плотностью связей внутри сообществ и низкой плотностью связей между сообществами, подробно описаны в [1].

Выделение в социальном графе сообществ помогает лучше понять пользователя, обозначить круги его общения, выделить интересы. Одной из главных задач анализа сообществ является определение того, что привело к формированию того или иного сообщества, и нахождение общих характеристик пользователей этого сообщества.

## Профили сообществ пользователей

Профиль сообщества формируется из фактических и поведенческих характеристик пользователей, которые в него входят.

Из фактических характеристик для описания сообществ пользователей используются пол, возраст и город проживания, так как эксперименты в [2] показали, что эти характеристики чаще других указаны у пользователей социальной сети, и они могут быть использованы для того, чтобы охарактеризовать группу людей. Визуальное представление характеристик зависит от их типа. Для количественных характеристик используются гистограммы и коробчатые диаграммы. Для качественных признаков – столбчатые диаграммы. Представление строится для всех друзей пользователя и для всех друзей пользователя с разбиением по сообществам.

В качестве источника поведенческих характеристик особый интерес представляют группы в социальной сети, в которых состоят пользователи. Формируем для каждого сообщества списки групп, в которых состоит большинство членов сообщества, и списки групп, в которых наибольшая доля пользователей является членами рассматриваемого сообщества.

## Результаты экспериментов

В ходе работы проводились эксперименты на случайной выборке пользователей социальной сети ВКонтакте. Приведем несколько наблюдений по результатам экспериментов. В качестве примера возьмем одного случайно выбранного пользователя.

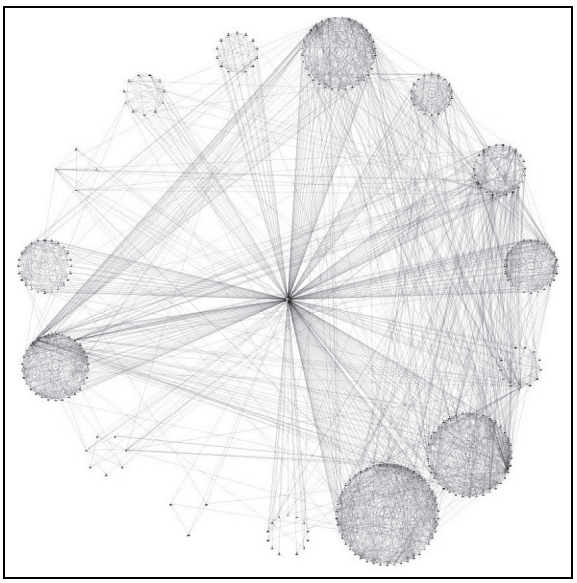

**Рис. 1.** Граф с выделением сообществ

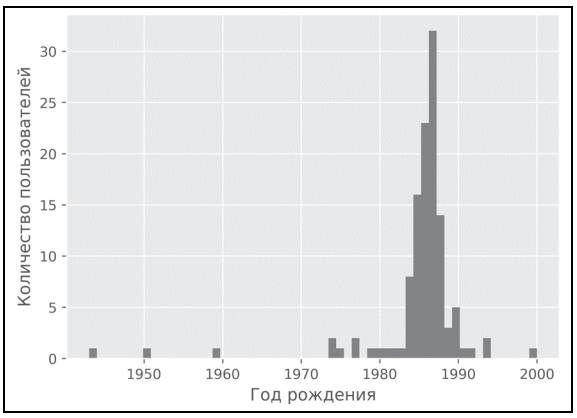


**Рис. 2.** Диаграмма распределения по годам рождения пользователей

С использованием API ВКонтакте из социальной сети была загружена информация о пользователе, его друзьях и дружеских связях между ними. По этим данным был построен граф и в программном комплексе AVS [1] с применением алгоритма [4] получено разбиение пользователей на сообщества. У пользователя в социальной сети оказалось 302 друга, которые с помощью алгоритма были поделены на 15 непересекающихся сообществ.

Граф с разбиением на сообщества представлен на рис. 1. Вершина исходного пользователя помещена в центр. Вершины пользователей каждого из сообществ размещаются на отдельной окружности. Центры полученных окружностей также располагаются на окружности большего диаметра.

После разбиения на сообщества переходим к анализу характеристик всех друзей пользователя и характеристик пользователей в каждом из сообществ.

Средний возраст друзей является хорошим индикатором реального возраста пользователя. В текущем примере день и месяц рождения были указаны у 72% друзей пользователя, а год рождения – только у 40%. Распределение годов рождения пользователей представлено на рис. 2.

На рис. 3 представлены медиана, нижняя и верхняя квартили и выбросы для годов рождения друзей того же пользователя, но уже с разбивкой по сообществам (коробчатые графики). Малый разброс годов рождения характерен, например, для сообществ, которые соответствуют сообществам друзей в школьном классе, групп в университете, а большой – сообществам родственников, коллег по работе, друзей по интересам.

Города проживания друзей указывают на города, в которых человек жил на протяжении жизни. В большинстве случаев на диаграммах можно увидеть сильное преобладание одного города, как на рис. 4. Реже встречается преобладание двух и более городов. Это может быть связано с временной (туристические, учебные, рабочие поездки) или постоянной сменой места проживания.

Университет, школа, место работы и интересы могут быть указаны в профиле пользователя ВКонтакте, но в большинстве случаев остаются незаполненными. Список групп, в которых состоят пользователи, позволяет расширить знание о пользователе и дает более глубокую и точечную оценку.

Часто пользователи создают небольшие группы для общения и обмена информацией в школьных классах, в университетских группах, перед туристическими поездками и другими мероприятиями или подписываются на страницы компаний, в которых работают, и вступают в группы по интересам. Знание о том, в каких группах состоит большинство членов сообщества, и в каких группах наибольшая концентрация членов этого же сообщества, позволяет сделать выводы о том, что привело к формированию того или иного сообщества.

Изначально список групп удалось получить для 20% друзей пользователя из примера. Но с помощью подхода, описанного ранее в параграфе про формирование социального графа, удалось частично восстановить списки групп и для тех пользователей, у которых они были скрыты. В итоге для 82% друзей пользователя была известна хотя бы одна группа.

Рассмотрим описание с использованием групп на примере двух сообществ. Часть названий групп в табл. 1 и 2 скрыта (***).

У пользователей сообщества под номером 1 из 38 пользователей, представленного в табл. 1, преобладают группы университетской направленности.

Пользователей из сообщества под номером 3 из 30 пользователей, описанного в табл. 2, объединяет интерес к песням и танцам и загородный центр творчества.

Стоит отметить, что списки групп, в которых состоит большинство членов сообщества, позволяют сделать вывод об общих интересах этой группы (например, университет). А списки групп, в которых наибольшая доля пользователей является членами рассматриваемого сообщества,

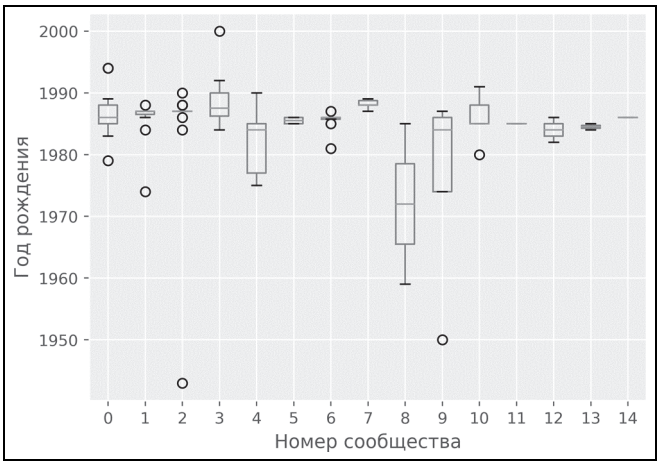


**Рис. 3.** Коробчатые графики распределения по годам рождения пользователей с разбиением по сообществам

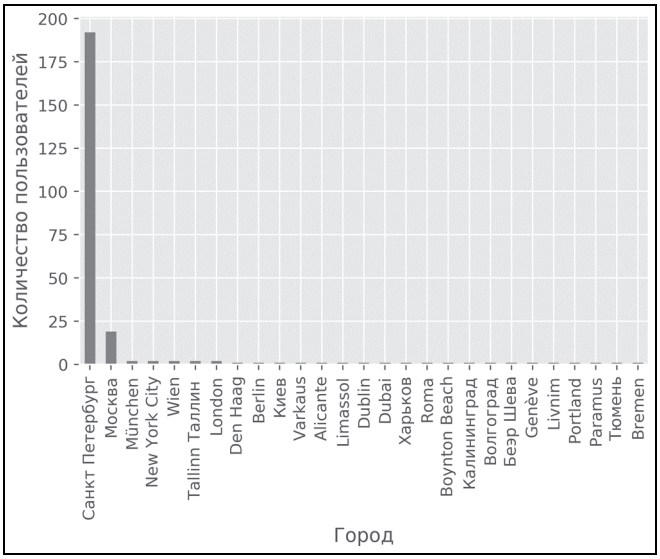


**Рис. 4.** Диаграмма распределения городов проживания пользователей

**Таблица 1.** ***Описание группами сообщества из 38 пользователей***

| Название группы | Размер группы | Пользователей в группе из сообщества |
|---|---|---|
| ЭкФак | 5763 | 9 (0,16%) |
| FAKULTET *** | 2045 | 5 (0,24%) |
| Выпускной *** | 20 | 3 (15%) |

**Таблица 2.** ***Описание группами сообщества из 30 пользователей***

| Название группы | Размер группы | Пользователей в группе из сообщества |
|---|---|---|
| Ансамбль песни и танца *** | 1636 | 18 (1,1%) |
| ЗЦДЮТ *** | 16948 | 16 (0,09%) |
| *** | 16 | 6 (37,5%) |

позволяют выделить небольшие локальные группы, которые были созданы членами этого сообщества (например, конкретная учебная группа в университете).

- Представлена и апробирована методика, которая позволяет анализировать группы общения пользователей, выделять их основные характеристики и определять, что привело к их формированию. Также предложенная методика может быть использована для восстановления и дополнения характеристик, которые не были указаны или были скрыты в настройках, для каждого отдельного пользователя.

  Сформированные таких образом профили могут использоваться в задачах, в которых осуществляется ручная обработка профилей (в безопасности, банковской сфере, в кадровых службах и пр.), а составляющие такого профиля могут выступать в качестве факторов для моделей при автоматической обработке профилей.

### Литература


1. *Коломейченко М.И.*, *Поляков И.В.*, *Чеповский А.А.*, *Чеповский А.М.* Методы визуального анализа графов. М.: Национальный открытый университет «ИНТУИТ». 2016.
2. *Соколова Т.В.*, *Чеповский А.М.* Задача анализа профилей пользователей социальных сетей // Сб. трудов Междунар. науч. конф. SCVRT2017. М., Протвино: Институт физико-технической информатики. 2017. С. 198–201.
3. *He J.*, *Chu W.W.*, *Liu Z.V.* Inferring privacy information from social networks // International Conference on Intelligence and Security Informatics. Springer, Berlin, Heidelberg. 2006. С. 154–165.
4. *Kolomeychenko M.I.*, *Chepovskiy A.A.*, *Chepovskiy A.M.* An Algorithm for Detecting Communities in Social Networks // Journal of Mathematical Sciences. 2015. V. 211. № 3. P. 310–318.
5. *Mislove A. et al*. You are who you know: inferring user profiles in online social networks // Proc. of the third ACM international conference on Web search and data mining. ACM. 2010. С. 251–260.
6. *Zhang B. et al*. Predicting demographic attributes based on online behavior: заяв. пат. 11/366,526. США. 2006.


**Поступила 3 августа 2018 г.**

# Analysis on communities profiles in social networks



**T.V. Sokolova** – Post-graduate Student, RTU MIREA (Moscow)
E-mail: sokolova.stv@mail.ru
**A.A. Chepovskiy** – Ph.D.(Phys.-Math.), Associate Professor, HSE (Moscow)
E-mail: aachepovsky@hse.ru

This paper presents the problem of forming user profiles based on data from social networks. For user profiling both the user and his friends data are used. Community allocation algorithms in social graphs are used to detect groups of communication. Each community has its own profile, which includes the characteristics of the users that belong to it and the groups that these users belong to. Profiles formed in this way can be used in tasks in which manual processing of profiles (security, banking, personnel services, etc.) is carried out, and the components of such a profile can act as factors for models in the automatic processing of profiles.

### References


1. *Kolomeychenko M.I., Polyakov I.V., Chepovskiy A.A., Chepovskiy A.M.* Metodi visualnogo analyza graphov. M.: Nacionalniy otkritiy universitet «INTUIT». 2016.
2. *Sokolova T.V., Chepovskiy A.M.* Zadacha analyza profiley polzovateley socialnikh setey // Sb. trudov Megdunar. nauch. konf. SCVRT2017. M., Protvino: Institut phiziko-tekhnicheskoy informatiki. 2017. S. 198–201.
3. *He J.*, *Chu W.W.*, *Liu Z.V.* Inferring privacy information from social networks // International Conference on Intelligence and Security Informatics. Springer, Berlin, Heidelberg. 2006. S. 154–165.
4. *Kolomeychenko M.I.*, *Chepovskiy A.A.*, *Chepovskiy A.M.* An Algorithm for Detecting Communities in Social Networks // Journal of Mathematical Sciences. 2015. V. 211. № 3. P. 310–318.
5. *Mislove A. et al*. You are who you know: inferring user profiles in online social networks // Proc. of the third ACM international conference on Web search and data mining. ACM. 2010. S. 251–260.
6. *Zhang B. et al*. Predicting demographic attributes based on online behavior: zayav. pat. 11/366,526. USA. 2006.

УДК 004.042

# Информационные волны в социальных сетях: проблематизация, определение, механизмы распространения[1]



**Г.В. Градосельская** – к.соц.н., вед. науч. сотрудник, НИУ ВШЭ (Москва)
E-mail: mss981009@mail.ru
**Т.Е. Щеглова** – стажер-исследователь, НИУ ВШЭ (Москва)
E-mail: teshcheglova@gmail.com
**И.А. Карпов** – мл. науч. сотрудник, НИУ ВШЭ (Москва)
E-mail: karpovilia@gmail.com

Рассмотрен новый аналитический концепт исследования информационного пространства социальных сетей – «информационные волны». Подчеркнута двойственность природы информационных волн, поскольку, с одной стороны, это поток информационных единиц, связывающий акторов, сообщества, паблики и т.п., а с другой стороны, в основе таких информационных потоков лежат социальные закономерности, зачастую основанные на проективных манипулятивных процессах, просчитывающих массовую реакцию, или управляющих взаимодействиями акторов. Показано, что такой подход требует специального определения исследуемого процесса (информационных волн), а также сущностного рассмотрения его механизмов. Приведены прикладные результаты исследований информационных волн в социальных сетях.

***Ключевые слова:*** *анализ социальных сетей, информационные волны.*

The article is devoted to the consideration of the new analytical research concept of information space of social networks – «information waves». The duality of the nature of information waves is emphasized. On the one hand, it is the flow of information units connecting actors, communities, publics, etc. On the other hand, in the basis of such information flows are the social regularities that are often based on the projective manipulative processes counting mass response or managing interactions of actors. Such approach requires special determination of the researched process (information waves) and also intrinsic reviewing of its mechanisms. In the article application-oriented research results of information waves in social networks are given.

***Keywords:*** *analysis of social networks, information waves.*

**DOI: 10.18127/j20729472-201803-15**

Ц е л ь  р а б о т ы – рассмотреть новый аналитический концепт исследования информационного пространства социальных сетей – «информационные волны».

## Научная проблема определения информационной волны

Информационные волны в последнее время становятся все более отточенным механизмом управляемого распространения информации, сочетающим как технологические, так и социальные ресурсы.

Сам термин «информационная волна» (в английской транскрипции «information wave») не является новым. Упоминание этого термина встречается еще в начале 1990-х годов в маркетинговых журналах [1]. В России этот термин появляется десятилетием позже, в начале 2000-х уже в политическом контексте. Сходство подходов на том этапе проявилось в рассмотрении информационной волны с позиций манипулятора – исследовались технологии распространения и достижения определенных целей [2].

Возвращение к теме «информационных волн» произошло еще через 10 лет – после 2012–2014 гг. Технические возможности распространения информации через социальные сети и социальная природа распространения этой информации поставили вопрос обнаружения этих волн уже со стороны исследователей. При этом наработки со стороны математиков (определение каскадов и т.п.) в практическом плане приносили мало пользы именно потому, что в технических математических моделях отсутствовала социальная компонента, а также манипулятивная составляющая информационных волн.

Из внимания авторов теоретических работ информационного поля упускалась одна важная деталь – работа с целевыми аудиториями. А ведь речь идет не просто о конструировании целенаправленных потоков доступа к широким слоям потребителей информационного продукта, а о доступе к узко таргети-

[1] Исследование финансировалось в рамках государственной поддержки ведущих университетов Российской Федерации «5-100».

рованной группе, точном донесении информации, рассчитанной именно на эту группу, и проектировании социальной реакции на полученное сообщение.

Первым шагом рассмотрения манипулятивных моделей и информационных волн, распространяемых по социальной сети, является картирование тех групп, по которым происходит распространение информации, и которые оказываются вовлеченными в определенные социальные процессы.

Именно содержание и социальные характеристики авторов сообщения определяют то, каким образом оно распространяется. Это, как раз, учитывают исследователи, подходящие к изучению распространения информации как к распространению информационных волн.

Таким образом, можно обозначить научную проблему определения информационной волны как объекта исследования, проведения операционализации этого понятия и разработки методологии его изучения.

Необходимо отметить, что три концепта (информационные волны, манипулятивные модели, картирование социальных сетей) расположены в фокусе данного исследования, и они являются взаимосвязанными элементами общей практики управления информационным пространством. Можно ввести предварительные определения основных рассматриваемых к о н ц е п т о в :

*манипулятивные модели* – проектируемое воздействие на целевые аудитории для достижения ожидаемых социальных эффектов в интересах актора (субъекта) влияния;

*информационные волны* – способы распространения манипулятивных моделей в информационном пространстве (можно определить и по-другому: информационные волны как управляемый способ распространения информации к целевым аудиториям);

*картирование социальных сетей* – это разработанный авторами метод определения целевых аудиторий воздействия.

Таким образом, каждый из рассматриваемых концептов выполняет свою роль в общей структуре информационного воздействия. Картирование социальных сетей репрезентирует организационный уровень поддержки информационного воздействия и позволяет выявить целевые аудитории, на которые направлено это воздействие. Информационные волны позволяют распространять информацию (с заложенными в них манипулятивными моделями), доставляя ее точно к целевым аудиториям. Манипулятивные модели закладываются в тексты, получая актуальную информационную «обертку». Здесь становится важным и язык донесения информации до целевой аудитории, но это уже вопрос следующего этапа исследований.

### Формализация теоретического понятия информационной волны

В связи с формированием новой информационно-медийной парадигмы возникает необходимость уточнения понятийно-терминологического аппарата.

Условно волны можно поделить на несколько т и п о в :

1. *Сильные и слабые* по степени воздействия на общество. Например, к сильным можно отнести победу Трампа в выборах в США или выход Великобритании из состава Европейского союза.

2. *Длинные и короткие* по признаку продолжительности их существования в медиапространстве, связанного с продолжительностью общественного интереса к информационному поводу. Например, украинские события, вызвавшие государственный переворот, до сих пор находятся в центре внимания, хотя с момента их начала прошло уже более трех лет, в то время как о переименовании моста в Санкт-Петербурге в мост Ахмата Кадырова уже почти никто не пишет, хотя в свое время данное событие вызвало сильный общественный резонанс.

3. *По форме и характеру распространения* можно выделить: *кумулятивный резонанс*, *каскад*, *воронка*, *призма*. При *кумулятивном резонансе* информация о событии наслаивается, обогащается и ее воздействие на общество усиливается. Отличительным признаком *каскада* является то, что информация распространяется мгновенно, линейно, а обратная связь либо отсутствует, либо не является существенной с точки зрения процессов глобализации в обществе. Для *воронки* характерно возобновление общественного интереса к давно существующему информационному поводу, который стал забываться или восприниматься как относительно старый. В информационной волне такого типа происходит постепенное развитие, расширение сюжетов, событий, возрастание связанной с ними общественной активности и учащение случаев столкновения интересов участников. И, наконец, для *призмы* характерна значительная субъективная доля в интерпретации информационного повода благодаря особенностям медиаперсоны, которая имеет определенный уровень доверия, а также многократность отражения информационного

повода с разных сторон на основе субъективизма источников информации или медийной личности [3].

Таким образом, информационные волны могут иметь разный характер распространения, протекать с разной силой воздействия на общественное сознание, варьироваться по длительности и форме распространения в медиасреде. Теория информационных волн не только отражает сложившуюся систему коммуникации в медиапространстве, но и объясняет закономерности существования информационных поводов, а также отражает влияние СМИ и Интернета как канала коммуникации на индивидуальное и общественное сознание [3].

Важное направление управляемого распространения информации – работа с целевыми аудиториями. Пик информационной волны приходится на момент, кода целевая аудитория подключается к СМИ, затрагивающим ее интересы: общественный интерес объединяется с интересами журналистов и работает на удовлетворение общих потребностей. Волна предполагает наличие ответной реакции, двусторонней коммуникации, если ее нет, нет и волны. Это говорит о социальном характере информационных волн, что отличает их от простых потоков информации [3].

Следующим шагом рассмотрения манипулятивных моделей и информационных волн, распространяемых по социальной сети, является картирование тех групп, по которым происходит распространение информации и которые оказываются вовлеченными в определенные социальные процессы.

## Методика выявления и классификации информационных волн

В данной работе будут представлены результаты нескольких исследований информационных волн по социально-политической тематике со статистико-математическими выкладками, профилированием волн по большим данным, собранным из разных социальных сетей (FB, VK, LJ), и с постановкой реальной исследовательской задачи.

Будет продемонстрировано несколько типов волн: искусственные, естественные, смешанные. Волны выделяются содержательно на основании какой-либо темы или информационного события. Данный этап предполагает экспертное кодирование сообщений в выгрузке, полученной с помощью Brand Analytics. Сообщения кодируются тематически, то есть публикации, освещающие одну тему или событие, попадают в одну информационную волну.

Выделяется три-четыре уникальных волны разного типа. Параллельно внутри каждой волны будут выделяться «подволны» или «субволны». Волны разного типа имеют различные профили.

Также будет проведен подробный анализ структуры распространения публикаций.

Первый пример – волна «Мост раздора» – является иллюстрацией смешанной волны, в которой присутствует больше элементов естественности. Основная тема данной волны – переименование моста через Дудергофский канал в Санкт-Петербурге в честь президента Чечни в 2003−2004 гг. Ахмата Кадырова. На момент выгрузки в конце июня 2016 г. волна включала в себя 689 публикаций в социальных сетях, блогах, а также на некоторых новостных порталах и сайтах СМИ в период с 30 мая по 29 июня. На рис. 1 видно, как число сообщений увеличивается в период с 15 по 17 июня, то есть в период, когда произошло само событие, после чего объем сообщений падает. Однако с 1 по 13 июня и с 18 по 29 июня число сообщений постоянно колеблется: заметно небольшое увеличение числа публикаций 1, 8, 13, и 19 июня.

Была построена сеть распространения публикаций (рис. 2). Особую роль играл межсетевой компонент, поэтому в роли связи выступает не только наличие фактического репоста, но и использование в тексте ссылки на сайт или другой первоисточник. С точки зрения сети, не важно, кто агент сети – индивид, СМИ или что-то другое, имеется факт публикации материала или его репоста, который и учитывается для построения сети –

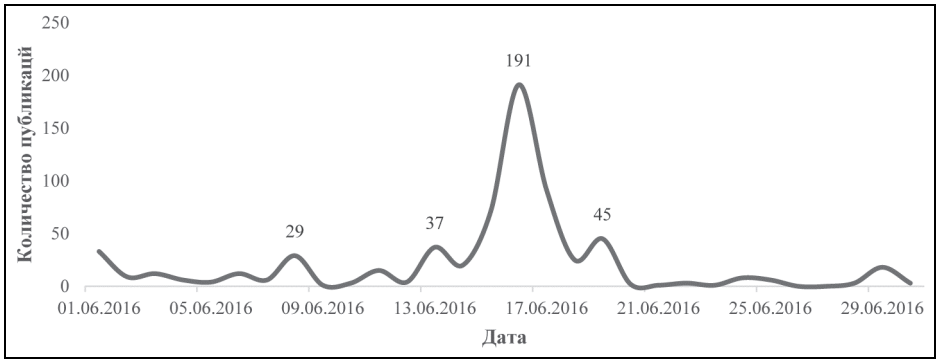


**Рис. 1.** График динамики публикаций в волне «Мост раздора»

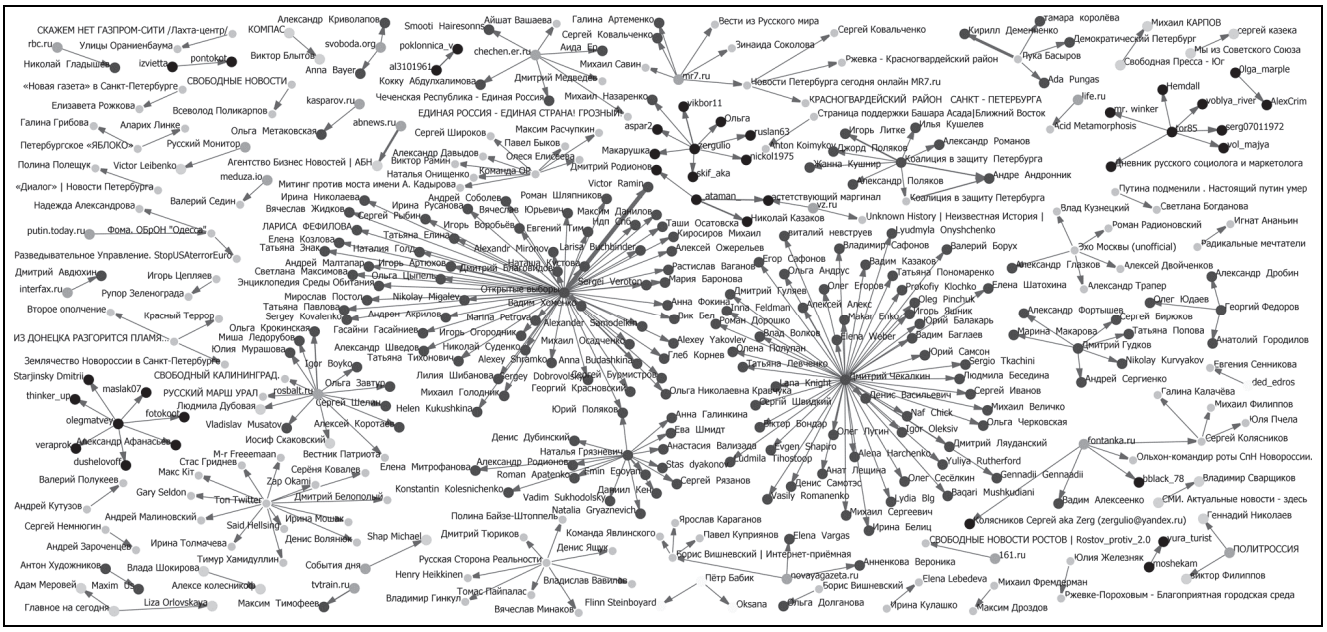

**Рис. 2.** Граф репостов в волне «Мост раздора» в период с 1 по 29 июня

структуры распространения публикации.

Проведена типологизация авторов и сообщений. Выделены технические аккаунты, информационные хабы, естественные аккаунты. У каждого типа свой профиль сетевого поведения, который будет подробно отражен в докладе.

Выделим т р и  т и п а  п е р с о н , присутствующих в Сети.

1. *Технические аккаунты* (также известные, как «боты»), как правило, имеют беспорядочный набор букв и цифр в имени, картинки вместо личных фотографий, размещают у себя в ленте все подряд без разбора, делают большое число публикаций или репостов за раз. Такие аккаунты часто блокируют или удаляют администраторы сайта.

2. *Информационные хабы* (в переводе с англ. «хаб» – сосредоточение) занимаются тем, что собирают у себя на странице публикации, рассчитанные на свою целевую аудиторию. Часто, особенно в случае с информационными волнами, внутрь контента, размещаемого на таких страницах, попадает контент политический. Так, он встраивается в определенные социальные группы, а те его поглощают, сами того не зная и не желая.

3. *Естественные персоны* – аккаунты реальных людей, которые размещают материал у себя на странице, не преследуя каких-то конкретных целей, в особенности манипулятивных, а просто потому, что им это интересно.

Критерием для классификации данных волн являются следующие  п о к а з а т е л и , выведенные авторами теоретически и подлежащие эмпирической проверке.

*Особенности распределения публикаций во временном промежутке*. Само понятие информационной волны предполагает некую ограниченность во временном пространстве. Какие-либо отклонения времени публикации от нормального распределения, такие как, например, публикация сообщений в строго определенные промежутки времени, может служить одним из критериев того, что волна искусственная.

*Воспроизводимость публикаций*. После того, как волна выделяется тематически, происходит ее разбивка по уникальным текстам. Преобладание реплицированных текстов («репостов» или просто копирование текста) – показатель, свойственный искусственным волнам. Однако большая воспроизводимость сообщения еще не говорит об искусственности волны, в естественных информационных волнах также могут встречаться неуникальные тексты.

*Привязанность публикации к информационному поводу*. Для искусственных волн характерна привязка к прошедшим или предстоящим событиям, а также освещение этих событий в теме публикации.

*Сетевые характеристики авторов публикаций*. В связи с тем, что в данном исследовании анализируются источники различного типа, сетевые показатели будут различаться как с фактической точки зре-

ния (характер связи: «подписка» одного пользователя является однонаправленной связью, «дружба» – ненаправленной двусторонней связью), так и с позиции субъективного восприятия пользователями различных платформ. Однако «дружба» или «подписка» не являются наилучшими характеристиками связи между пользователями, поэтому авторы склоняются к тому, чтобы использовать сеть репостов, которая куда лучше отражает реальную связь. В искусственных волнах сеть репостов будет плотнее, чем в естественных или смешанных, за счет того, что действия пользователей являются заранее спланированными и скоординированы.

*Характеристика авторов публикаций*. Не только сетевые характеристики могут говорить о роли, которую актор играет в распространении волны, но и характеристики его профиля. Личная информация, размещенная в профиле актора, а также информация о том, какой контент, в каком количестве и с какой периодичностью публикует пользователь, позволяет определить его роль в распространении волны и на основании этого разработать стратегию работы с данной персоной.

- Необходимо анализировать не только тексты публикаций и структуру их распространения в сети, но также и тех, кто эти публикации распространяет, на основе чего можно будет сделать вывод, какие типы акторов характерны для разных типов информационных волн.

  Ни один из рассмотренных показателей не является сам по себе достаточным для того, чтобы отнести волну к тому или иному классу. Однако данные о совокупности показателей могут служить критерием для классификации информационных волн.

### Литература


1. *Bessen J.* Riding the Marketing Information Wave // Harvard Business Review. Market research. September–October 1993. [Электронный ресурс]. URL = https://hbr.org/1993/09/riding-the-marketing-information-wave (дата обращения: 10.04.2018).
2. *Сороченко В.* Энциклопедия методов пропаганды. 2002. [Электронный ресурс]. URL = http://polbu.ru/sorochenko_propagation/ch33_all.html (дата обращения: 10.04.2018).
3. *Болотнов А.В.* Информационные волны и их типы в современном медиадискурсе // Вестник Томского педагогического университета. 2015. № 6(159).


**Поступила 3 августа 2018 г.**

# Information waves on social networks: problematization, definition, distribution mechanisms



**G.V. Gradoselskaya** – Ph.D.(Soc.), Leading Research Scientist, HSE (Moscow)
E-mail: mss981009@mail.ru
**T.E. Shcheglova** – Trainee-researcher, HSE (Moscow)
E-mail: teshcheglova@gmail.com
**I.A. Karpov** – Junior Research Scientist, HSE (Moscow)
E-mail: karpovilia@gmail.com

The article is devoted to the consideration of the new analytical research concept of information space of social networks – «information waves». The duality of the nature of information waves is emphasized. On the one hand, it is the flow of information units connecting actors, communities, publics, etc. On the other hand, in the basis of such information flows are the social regularities that are often based on the projective manipulative processes counting mass response or managing interactions of actors. Such approach requires special determination of the researched process (information waves) and also intrinsic reviewing of its mechanisms. In the article application-oriented research results of information waves in social networks are given. It is necessary to analyze not only the texts of publications and the structure of their distribution in the network, but also those who disseminate these publications, on the basis of which it will be possible to conclude which types of actors are characteristic for different types of information waves.

### References


1. *Bessen J.* Riding the Marketing Information Wave // Harvard Business Review. Market research. September–October 1993. [E'lektronny'j resurs]. URL = https://hbr.org/1993/09/riding-the-marketing-information-wave (data obrashheniya: 10.04.2018).
2. *Sorochenko V.* E'nciklopediya metodov propagandy'. 2002. [E'lektronny'j resurs]. URL = http://polbu.ru/sorochenko_propagation/ch33_all.html (data obrashheniya: 10.04.2018).
3. *Bolotnov A.V.* Informaczionny'e volny' i ix tipy' v sovremennom mediadiskurse // Vestnik Tomskogo pedagogicheskogo universiteta. 2015. № 6(159).