

# Многоступенчатая архитектура контроллера твердотельного накопителя с повышенной пропускной способностью

А. А. Дубровин

Федеральное государственное учреждение "Федеральный научный центр Научно-исследовательский институт системных исследований Российской академии наук",

г. Москва,

dubrovin@cs.niisi.ras.ru.

**Аннотация** — Современные устройства хранения данных по пропускной способности значительно отстают от интерфейсов сетевых контроллеров, что ограничивает быстродействие микропроцессорной системы. В работе предложена многоступенчатая архитектура контроллера твердотельного накопителя, которая позволяет масштабировать и модернизировать систему хранения данных. Показано, что ее использование позволяет увеличить пропускную способность контроллера твердотельного накопителя за счет организации перекрестных связей внутри многоступенчатой иерархии и увеличения числа используемых запоминающих устройств. В рассматриваемой системе увеличение пропускной способности до показателей интерфейса PCIe 4x16 обосновано применением 12-ступенчатой архитектуры NVMe контроллера.

**Ключевые слова** — твердотельный накопитель, контроллер твердотельных накопителей, система хранения данных, пропускная способность.

## I. ВВЕДЕНИЕ

Новые инструменты научных исследований производят значительные объемы данных, которые необходимо хранить в течение неопределенного количества времени. Например, главный детектор ATLAS большого адронного коллайдера детектирует 40 миллионов событий в секунду, что обеспечивает 1 ПБ

исходных данных в секунду [1]. Это приводит к необходимости собирать, хранить и анализировать массивы данных порядка петабайт, для чего требуется развивать возможности систем хранения данных (СХД). Современные исследования, ведущиеся в области разработки параллельных файловых систем, применяемых в суперкомпьютерах петафлопсного класса, и глобальных файловых систем для центров высокопроизводительных вычислений, в основном сфокусировались на задачах увеличения ширины пропускаемого канала для операций чтения и записи, которые одновременно выполняются на этих платформах [2].

На рис. 1 представлена блок-схема СХД микропроцессорной системы, содержащая в себе  $N$  (как правило, более 10) устройств хранения, управление которыми происходит посредством контроллера устройства хранения. Эти устройства, в свою очередь, управляют  $M$  (как правило, более 10) микросхемами памяти. В качестве контроллера устройства хранения чаще всего применяются контроллеры твердотельного накопителя (КТН), например NVMe или AHCI. Пропускная способность СХД достигается доступом к большому числу микросхем памяти. Распространенным примером устройства хранения являются микросхемы памяти NAND flash.

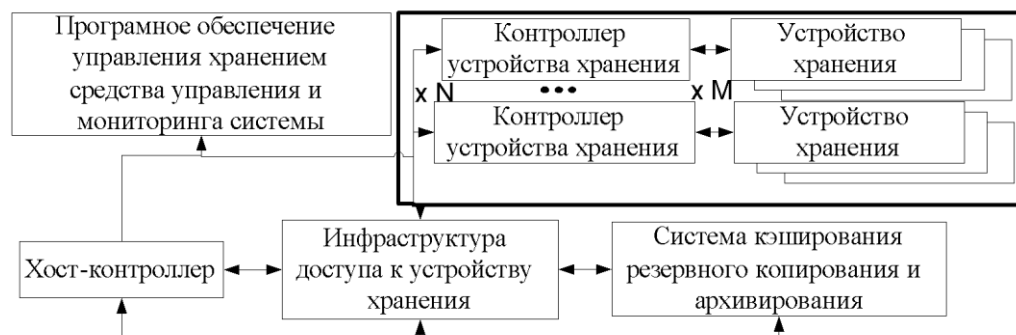


Рис. 1. Структурная блок-схема СХД

Требования к сохранению максимальной производительности, надежности и отказоустойчивости СХД при ее модернизации и

масштабируемости порождают проблемы увеличения требований к характеристикам ее элементов.

Производительность определяется числом и типом накопителей, объемом кэш-памяти, вычислительной мощностью процессорной системы, числом и типом внутренних и внешних интерфейсов, а также возможностями гибкой настройки и конфигурирования. Повышение производительности возможно за счет увеличения числа доступных конечных устройств или применением оптимизирующих алгоритмов.

Надежность и отказоустойчивость достигается при помощи резервирования и применением избыточного кодирования. При масштабировании систем необходимо учитывать не только требуемые конечные характеристики системы, но и требования обеспечения надежности и отказоустойчивости системы.

Таким образом, при увеличении пропускной способности систем хранения данных в первую очередь необходимо рассмотреть модификации контроллера твердотельного накопителя, позволяющие параллелизовать потоки данных.

Устройства хранения работают со скоростью около 1 миллиона операций ввода-вывода в секунду на устройство, в то время как современные сетевые контроллеры способны обрабатывать более 70 миллионов операций в секунду [3]. Выход из строя накопителя из верхнего уровня приведет к значительному увеличению времени доступа к данным, например, восстановление данных с заполненного 16-терабайтного SSD-накопителя с интерфейсом SATA III при пропускной способности интерфейса 500 МБ/с требует не менее 9 часов [1].

Данная статья посвящена реализации многоступенчатой архитектуры контроллера твердотельного накопителя, которая позволяет масштабировать и модернизировать систему хранения данных. Такая модернизация придает архитектуре гибкость с возможностью резервирования, масштабирования, и наличием избыточности устройств в сравнении с приемопередающими интерфейсами за счет предлагаемого способа построения многоступенчатой иерархии модифицированных КТН. Это является необходимым для устранения разрыва пропускной способности между приемопередающим интерфейсом сетевых контроллеров и конечной энергонезависимой памяти в микропроцессорной системе.

## II. ОБЗОР ПРЕДЛОЖЕННОЙ АРХИТЕКТУРЫ КТН

В настоящее время интерфейсы КТН существенно отстают по скорости передачи данных от приемопередающих интерфейсов передачи данных сетевых контроллеров, обеспечивающих взаимодействие между вычислительными устройствами. Например, скорость одного канала интерфейса памяти ONFI NAND flash составляет до 800 Мбайт/с, в то время как интерфейс PCIe 4x16, использующийся для передачи данных между отдельными устройствами в микропроцессорной системе, поддерживает скорость передачи в 31.5 Гбайт/с. Чтобы соответствовать характеристикам приемопередающего интерфейса контроллеры памяти

должны поддерживать высокую степень параллелизма, что приводит к появлению узкого места в работе параллельных файловых систем, проблемы надежности и производительности которых описаны в работе [2]. Сейчас интерфейс PCIe 4x16 имеет значительный запас по пропускной способности относительно отдельных устройств хранения.

Предложенная архитектура КТН основана на том, что устройство NVMe может выступать в роли самостоятельного хост-устройства NVMe, что дает возможность для построения подобных друг другу систем, обладающих возможностями перераспределения нагрузки между ступенями иерархии. Это также приведет к инкапсуляции от конечного пользователя части программного обеспечения СХД на уровень КТН.

В исходной структуре NVMe контроллера, представленного на рис. 2, хост-устройство формирует в своей динамической памяти команды и данные для записи на устройства хранения, которые передает через приемопередающий интерфейс. Регистры NVMe контроллера формируют прерывание для процесса, в результате которого процессор начинает выполнение команд хост-устройства. Данные при помощи механизма прямого доступа к памяти записываются в динамическую память устройства, откуда они будут перезаписаны в постоянные энергонезависимые запоминающие устройства.

Именно наличие полноценного вычислительного ядра делает NVMe контроллер первым контроллером твердотельного накопителя, способным выступать в роли хост-устройства для другого устройства хранения. В сравнении с исходной структурой NVMe контроллера в многоступенчатой реализации присутствует модуль мастер-контроллера SATA или PCIe, который выступает хост-контроллером для следующей ступени КТН. Исходная структура NVMe контроллера изображена на рис. 3. Частным случаем такого решения является двухступенчатая архитектура, описанная в работе [4], данная работа является ее развитием. В многоступенчатой архитектуре каждая последовательная ступень имеет общее пространство адресов к хранимым данным и имеет подключение к другим параллельным ступеням при помощи мастер-контроллера приемопередающего интерфейса. Таким образом осуществляется резервирование доступа к различным устройствам и выход одного из строя не приведет к отказу какой-либо другой части СХД. Исходное хост-устройство имеет несколько корневых портов, которые обращаются к 0 ступени. Весь массив виртуально представляется в виде единого пространства памяти.

При использовании многоступенчатой архитектуры необходима минимизация временных задержек между ступенями. Для этого необходимо внести в однородный массив устройств неоднородность в виде коммутатора, который бы позволил уменьшить число транзитивных ступеней при доступе к последним ступеням иерархии. Блок схема применения такой архитектуры в составе современных СХД представлена на рис. 4.

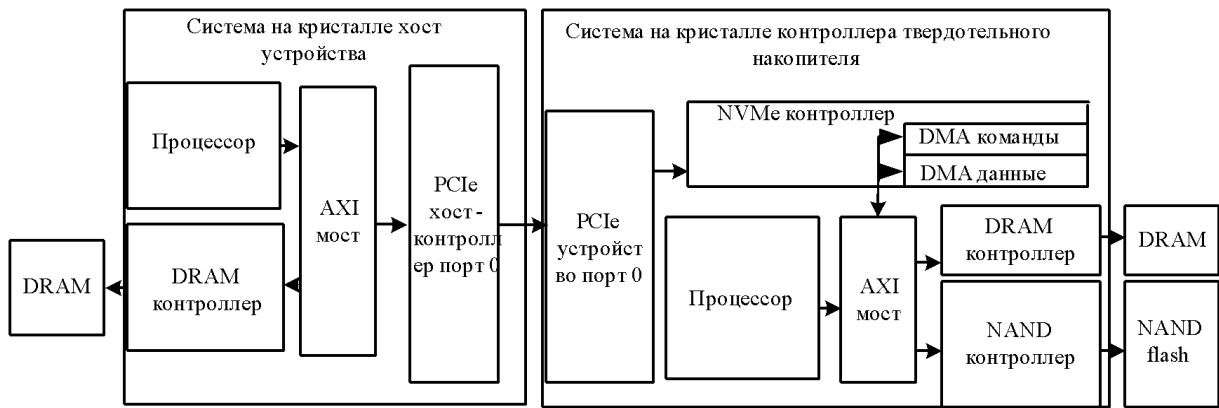


Рис. 2. Блок-схема структуры NVMe контроллера

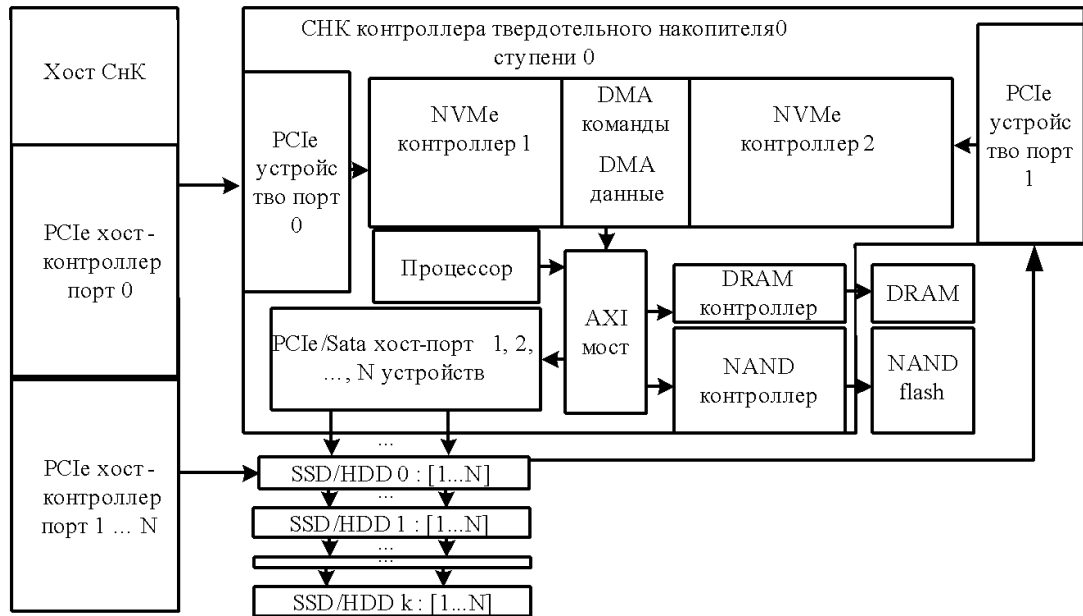


Рис. 3. Блок-схема структуры многоступенчатого NVMe контроллера, многоступенчатый КТН - СХД-подобная DAS архитектура

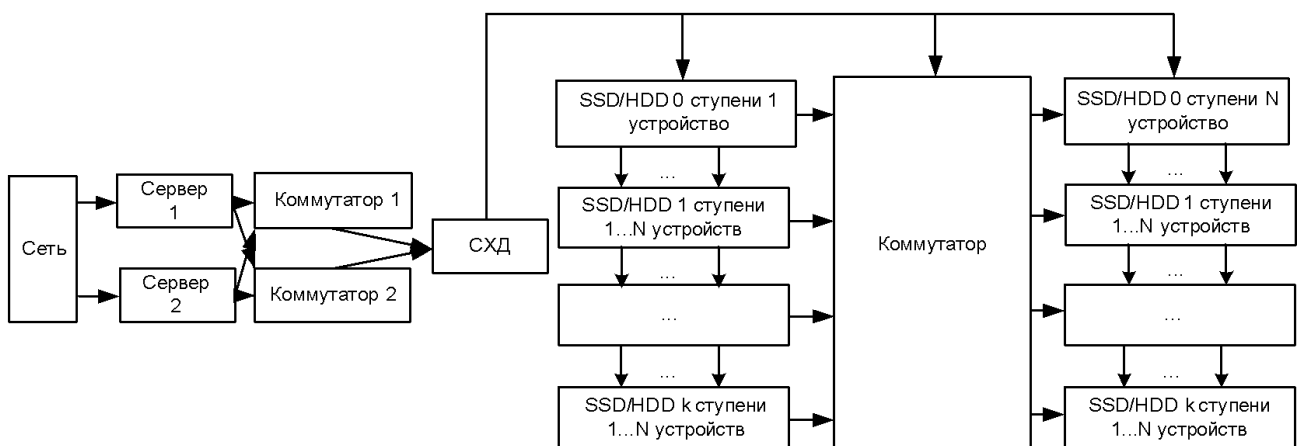


Рис. 4. Применение в текущих СХД многоступенчатый КТН

### III. МОДЕЛИРОВАНИЕ ПРЕДЛОЖЕННОЙ АРХИТЕКТУРЫ

Для сравнения многоступенчатой архитектуры с исходной было проведено моделирование на основе поведенческих моделей интерфейсов и памяти. Полученные значения пропускной способности для нескольких сравниваемых архитектур приведены в табл. 1, где представлено сравнение характеристик нескольких СХД с различными интерфейсами и архитектурами. За основу взяты характеристики одного устройства хранения NVMe PCIe 3x4. Поведенческая модель СХД на их основе включала в себя поведенческие модели нескольких устройств, которые в зависимости от их подключения организовали определенную архитектуру СХД.

В первую очередь были рассмотрены параллельная и последовательные архитектуры на основе устройств NVMe PCIe 3x4. Целью было достижение пропускной способности системой хранения данных на основе PCIe 4x16, поэтому следующим шагом было моделирование последовательной и параллельной архитектуры на основе устройств с интерфейсом PCIe 4x16. Применение двух последовательных устройств давало запас пропускной способности по сравнению с интерфейсом PCIe 3x4, при переходе к PCIe 4x16 число ступеней выросло до 12, что привело к тому, что архитектура была бы ненадежной: выход из строя любого устройства привел бы к потере хранимой информации. Применение параллельной архитектуры позволило использовать превышение пропускной способности интерфейса, но только в рамках объема динамической памяти.

Как следствие, применение последовательной архитектуры необходимо для обеспечения равенства пропускных способностей прямо-передающего интерфейса и интерфейсов постоянных энергонезависимых запоминающих устройств, а параллельный подход обеспечивает резервирование трактов данных и равенство пропускной способности интерфейса и динамических запоминающих устройств до момента заполнения динамической памяти, что дает необходимое время обеспечения работы конвейера команд.

На основе анализа поведения последовательной и параллельной архитектур был сделан вывод о том, что только их совместное применение может увеличить пропускную способность до характеристик интерфейса PCIe 4x16 и при этом сохранить надежность. Таким образом, существующее решение с параллельным увеличением конечных устройств требуется дополнить последовательной многоуровневой архитектурой на основе самоподобных устройств, что позволит обеспечить запас пропускной способности в промежутке времени меньше, чем время заполнения динамической памяти.

Только применение последовательной архитектуры с увеличением числа устройств приведет к тому, что запас пропускной способности интерфейса будет превышен. Следовательно, выигрыша не произойдет при дальнейшем увеличении числа устройств выше некоторого порога. Но при применении интерфейса с высокой пропускной способностью и с последовательно-параллельным массивом устройств можно достичь повышения пропускной способности на отрезке времени, превышающем время заполнения динамической памяти.

Таблица 1

*Характеристики архитектур СХД*

| Параметр  | SSD NVMe PCIe 3x4 | СХД 8 параллельных SSD интерфейс PCIe3x4 | СХД 8 последовательных PCIe3x4 | СХД 8 последовательных PCIe4x16 | СХД 8 параллельных PCIe4x16 | СХД 8x8 параллельно-последовательных PCIe4x16 |
|---|-------------------|--|--------------------------------|---------------------------------|-----------------------------|---|
| Объем динамической памяти, Гбит                   | 64                | 512                                      | 512                            | 512                             | 512                         | 4096  |
| Пиковая пропускная способность интерфейса, Гбит/с | 19,2              | 153,6                                    | 19,2                           | 153,6                           | 1228,8                      | 1228,8  |
| NAND flash пропускная способность, Гбит/с         | 12,8              | 102,4                                    | 19,2                           | 102,4                           | 102,4                       | 819,2   |
| Время заполнение динамической памяти, с           | 3,33              | 3,33                                     | 3,33                           | 0,42                            | 0,42                        | 0,42  |
| Суммарный объем энергонезависимой памяти, Тбайт   | 1                 | 8  | 8                              | 8                               | 8                           | 64  |
| Число необходимых ступеней                        | 1                 | 1  | 2                              | 12                              | 12                          | 12  |

Из табл. 1 были выбраны 3 варианта, наиболее наглядно демонстрирующих преимущества параллельной, последовательной и последовательно-параллельной архитектур, и проведено их моделирование. На рис. 5 показана пропускная способность СХД сравниваемых архитектур, состоящих из параллельных 8 устройств типа NVMe PCIe 3x4 (1), СХД 8 последовательных NVMe PCIe 4x16 (2), и СХД, в котором для 8 параллельных портов по 8 последовательных устройств в каждом порту NVMe PCIe 4x16. Для многоступенчатой архитектуры проведено моделирование поведения СХД, состоящей из 8 параллельных устройств типа NVMe PCIe 3x4, представленных на рис. 5 график 1. СХД состоящее из 8 последовательных NVMe PCIe 4x16, представленных на графике 2. Однородная многоступенчатая архитектура СХД, состоящая из 8 параллельных портов по 8 последовательно соединенных устройств в каждом порту NVMe PCIe 4x16, представленных на графике 3. Сравнение произведено с помощью моделирования последовательной записи в САПР Cadence Ncsim на основе поведенческих моделей блоков. На их основе рассмотрена линейная, избыточная по отношению к интерфейсу, многоступенчатая архитектура из 12 ступеней с интерфейсом PCIe 4x16.

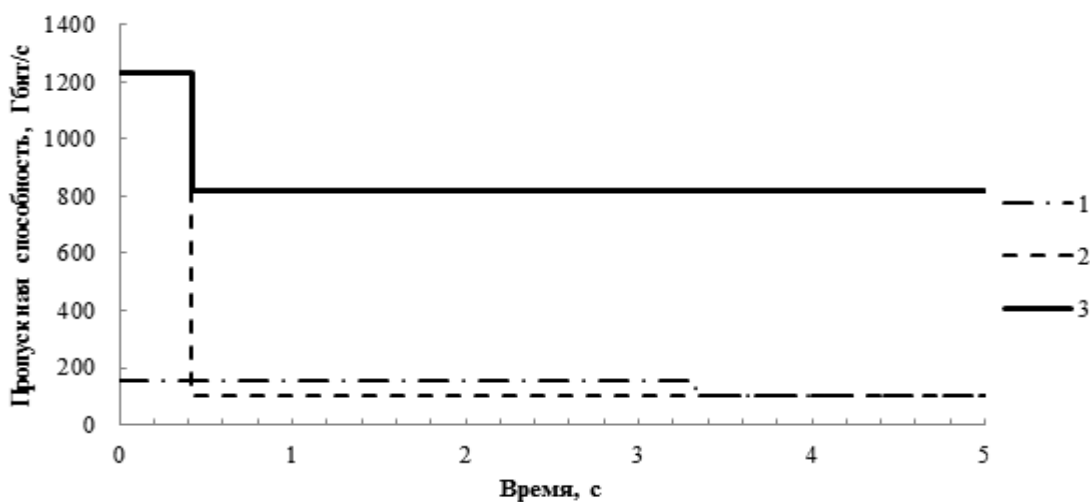
На рис. 5 график 3 отражает планируемые значения массива пропускной способности разрабатываемого устройства с интерфейсом PCIe 4x16. Графики 1 и 2 отличаются от 3 тем, что на графике 2 подключены 8 последовательных устройств с интерфейсом в 8 раз большей пропускной способностью, чем у устройств на графике 1, но устройства подключены параллельно с более медленным интерфейсом PCIe 3x4.

Особенность решения, представленного на графике 3, заключается в возможности масштабирования СХД и в ее архитектуре, имеющей запас пропускной способности между интерфейсами внутри системы в сравнении с внешним интерфейсом, что позволяет

выделить время для работы внутренних алгоритмов: таких как алгоритм фоновой очистки и алгоритм равномерного распределения нагрузки. Кроме того, запас пропускной способности дает возможность перейти к большему числу линий интерфейса и применить внутреннее резервирование между устройствами многоступенчатого NVMe PCIe 4x16, представленного на графике 1. При условии фиксированного числа интерфейсов появляется возможность получить требуемые характеристики пропускной способности. Сравнение производится на основе спецификации устройств [5, 6].

Из рис. 5 можно сделать вывод о том, что при дальнейшем увеличении числа устройств при параллельном подключении увеличивается только время заполнения динамической памяти устройств. В последовательной архитектуре ограничением становится запас по пропускной способности интерфейса, что требовалось при проектировании повышения пропускной способности, но архитектура становится уязвима к выходу из строя любого устройства. И только при многоступенчатой архитектуре, объединившей преимущества параллельной и последовательной архитектуры, появляется как резервирование тракта данных, так и равенство пропускной способности приемопередающего интерфейса и устройств хранения.

Предложенная архитектура обладает рядом недостатков. Например, возникает необходимость разработки дополнительного программного обеспечения для ее эффективного использования. Также недостатком является то, что переход от одноступенчатого NVMe PCIe 3x4 к двухступенчатому требует размещения на кристалле дополнительного модуля контроллера интерфейсом передачи, что занимает часть площади системы на кристалле и потребляет часть мощности.



**Рис. 5. Пропускная способность систем хранения данных на устройствах NVME PCIe, состоящей из 8 параллельных устройств типа NVMe PCIe 3x4 (1), СХД 8 последовательных NVMe PCIe 4x16 (2), и СХД, в котором для 8 параллельных портов по 8 последовательных устройств в каждом порту NVMe PCIe 4x16 (3)**

Общее усложнение и увеличение числа ступеней в СХД неизбежно приведет к росту временных задержек за счет множества транзитивных ступеней. Приведем пример на основе характеристик распространенного устройства [7], где каждая ступень вносит среднюю задержку равную 20 мкс, тогда максимальная задержка равна произведению числа ступеней на среднюю задержку, а средняя задержка системы будет равна средневзвешенному значению задержек обращения к каждой ступени, при однородной системе это значение будет равно среднему арифметическому значению задержек доступа к каждому устройству, то есть для системы из 12 ступеней максимальная средняя задержка увеличится с 20 мкс до 240 мкс, а средняя задержка доступа к однородной 12 ступенчатой системе будет равна 130 мкс.

#### IV. ПЕРСПЕКТИВЫ РАЗВИТИЯ

Стоит отметить, что приведенные данные описывают однородные многоступенчатые КТН, которые не имеют особенностей. Можно применить однородные и гибридные устройства таким образом, чтобы выполнялись принципы разделения функций между устройствами аналогично принципам, лежащим в организации многоуровневой архитектуры СХД. Кроме того, архитектура подобных устройств может иметь различное число входных и выходных интерфейсов, может иметь замкнутые связи интерфейсов в иерархии, может иметь интерфейсы между различными степенями, формируя в результате сеть устройств, подобную фракталу, за счет подобия фрагментов сети, сети устройств в целом. Такой подход позволяет наиболее гибко проектировать подобные многоступенчатые КТН в сравнении с СХД, что позволяет достигнуть нескольких задач. Во-первых, чтобы достичь большой емкости памяти и энергоэффективности, необходимо использовать как DRAM, так и STT-MRAM в качестве оперативной памяти вычислительного устройства. Ранее была предложена аппаратная схема (RAMinate), обеспечивающая одновременное и параллельное выполнение нескольких операционных систем на одном и том же хост-устройстве, в которой применяются два вида памяти (гибридная аппаратная схема), что позволило уменьшить потребляемую мощность схемы на 50-70% в сравнении с однородной DRAM схемой [8]. Во-вторых, можно применять гибридные твердотельные накопители в критически важных узлах подобной СХД. С помощью технологии Adaptive Memory, разработанной компанией Seagate, пользователи могут повысить производительность системы в условиях реальной работы на 50% без программного обеспечения сторонних разработчиков и в любой операционной системе [9]. Последующие поколения схем памяти MRAM, например STT-MRAM, возможно, могли бы стать “идеальной памятью” с гигабитной плотностью, бесконечным хранением данных, неограниченными циклами программирования и стойкой к радиационным эффектам [10]. Тем не менее, на данный момент данный вид памяти в сравнении с Flash NAND имеет недостаток в виде

значительно меньшей плотности, что, в итоге, дает выгоду от их совместного применения, описанную в работе [11]. Это позволяет оптимизировать выполнение задач, которые ранее выполняли контроллеры СХД, и распределение их функции на весь вычислительный кластер, так как, например, инфраструктура для задач класса больших данных строится на базе стандартных серверов, сетей, систем хранения данных, коммутаторов, гипервизоров и кластерного ПО. Например, комплексы, построенные путем масштабирования стандартных x86-серверов с Ethernet 10GbE, позволяют достигать вычислительных мощностей, сравнимых со специализированными суперкомпьютерами [12].

Лучшим решением по пропускной способности среди однородных архитектур многоступенчатых КТН на основе устройств NVMe PCIe 4x16, параметры которых описаны в табл. 1, является система, в которой запас пропускной способности достигается в архитектуре, состоящей из 12-ступеней.

#### V. ЗАКЛЮЧЕНИЕ

Приведенная архитектура поддерживает горизонтальную масштабируемость и увеличение пропускной способности при увеличении числа устройств. Применение архитектуры многоступенчатого контроллера позволяет сократить разницу между пропускной способностью высокочастотных быстродействующих интерфейсов и подмножеством системы энергонезависимой памяти. Например, при приведенных параметрах системы увеличение пропускной способности до показателей интерфейса PCIe 4x16 обосновано применением 12-ступенчатой архитектуры NVMe контроллера с целью достижения запаса пропускной способности при проектировании масштабируемой системы хранения данных порядка 64 ТБ за счет организации перекрестных связей внутри многоступенчатой иерархии и увеличения числа используемых устройств.

#### ПОДДЕРЖКА

Публикация выполнена в рамках государственного задания ФГУ ФНЦ НИИСИ РАН по теме №0065-2019-0004.

#### ЛИТЕРАТУРА

- [1] Курс лекций “Информационные технологии” МТУСИ. [Электронный ресурс] // URL: <https://ppt-online.org/295838> (дата обращения: 25.03.2019).
- [2] Alam S. R., El-Harake H. N., Howard, Stringfellow N., Verzelloni F. Parallel I/O and the Metadata Wall. // PDSW '11 Proceedings of the sixth workshop on Parallel Data Storage, Seattle, November, 13, 2011. - NY, est. ACM New York, 2011, P. 13-18.
- [3] Andrew A. Chien. Software challenges for the changing storage landscape. // Association for Computing Machinery, New York, NY, USA, volume 61 Issue 11, November 2018, p. 136-145.
- [4] Дубровин А.А. Метод повышения пропускной способности систем хранения данных - двухступенчатый контроллер твердотельного

- накопителя. // Труды НИИСИ РАН, 2019, том 9 №4 -С. 85-91.
- [5] PCIe Revision 4.0 Specification.
- [6] NVM Express Revision 1.2a.
- [7] Основные технические характеристики твердотельных накопителей Intel NVM Express серии SSDPE2MW400G401.
- [8] Takahiro Hirofuchi, Ryousei Takano. RAMinate: Hypervisor-based Virtualization, for Hybrid Main Memory Systems. // National Institute of Advanced Industrial Science and Technology (AIST), Tokyo, Japan, SoCC '16 Proceedings of the Seventh ACM Symposium on Cloud Computing, Pages 112-125, Santa Clara, CA, USA — October 05 - 07, 2016, ACM New York, NY, USA 2016.
- [9] Гибридные твердотельные накопители: естественная эволюция систем хранения данных. [Электронный ресурс] // Seagate, Inc URL: <https://www.seagate.com/ru/ru/point-of-view/sshd-and-natural-evolution-master-pov> (дата обращения: 04.07.2018).
- [10] Heidecker J. MRAM Technology and Status. // Pasadena, Jet Propulsion Laboratory, California Institute of Technology, 2012. - 27 p.
- [11] Дубровин А.А. Преимущества гибридного контроллера твердотельного накопителя, проанализированные с использованием алгоритма wear-leveling. // Труды НИИСИ РАН, 2018, Том 8 №3 - С. 61-70.
- [12] Проект OSP.RU Web-сайт издательства "Открытые системы". [Электронный ресурс] // Большие Данные — новая теория и практика. URL: <https://www.osp.ru/os/2011/10/13010990/> (дата обращения: 25.03.2019).

## Modification of the High Bandwidth Solid State Drive Controller within the Multi-Stage Architecture

A. A. Dubrovin

Federal State Institution "Federal Scientific Center Scientific Research Institute of System Analysis of the Russian Academy of Sciences", Moscow, Russian Federation,  
dubrovin@cs.niisi.ras.ru.

**Abstract** — This article is devoted to the implementation of a hardware solution for constructing a flexible architecture with the possibility of scaling and redundancy of devices due to the proposed method of constructing a multi-stage or fractal hierarchy of the modified solid-state drive. It is applied in order to eliminate the gap in the bandwidth of the transceiver interface of devices and non-volatile memory.

The most important advantages include the ability to upgrade, scalability and refinement of storage systems, along with transferring and encapsulating the user part of the storage system software at the modified solid-state drive level.

The methodic proposed in this article states that it is possible to apply a heterogeneous device in such a way that the principles underlying the multi-stage storage system are fulfilled. The architecture of such devices can have a different number of input and output interfaces, as well as interfaces between different level devices. As a result, it forms a fractal-like system of devices due to the similarity of array fragments to the array of devices as a whole. This approach allows the most parametrized design of storage systems, which enhances the systems' flexibility.

Therefore, the best bandwidth solution among the homogeneous multi-stage modified solid-state drive architectures is a system in which redundancy is sufficient to run background algorithms in conjunction with the normal operating mode, under ideal conditions. In this article, the analysis of an array consisting of 12 NVMe PCIe 4x16 devices with a bandwidth of 31.5 GB/s was proved to become redundant.

**Keywords** — Non-Volatile Memory Host Controller Interface, SSD, solid state disc, direct access storage.

- [1] The course of lectures "Information Technology" MTUCI. [Electronic resource] // URL: <https://ppt-online.org/295838> (date of the application: 25.03.2019).
- [2] Alam S. R., El-Harake H. N., Howard, Stringfellow N., Verzelloni F. Parallel I/O and the Metadata Wall. // PDSW '11 Proceedings of the sixth workshop on Parallel Data Storage, Seattle, November, 13, 2011. - NY, est. ACM New York, 2011, P. 13-18.
- [3] Andrew A. Chien. Software challenges for the changing storage landscape. // Association for Computing Machinery, New York, NY, USA, Volume 61 Issue 11, November 2018, p. 136-145.
- [4] Dubrovin A.A. Method increasing the bandwidth data storage systems or application two-stage SSD. // Trudy NIISI RAN, 2019, Tom 9 №4 -pp. 85-91.
- [5] PCIe Revision 4.0 Specification.
- [6] NVM Express Revision 1.2a.
- [7] Datasheet Intel NVM Express SSDPE2MW400G401.
- [8] Takahiro Hirofuchi, Ryousei Takano, RAMinate: Hypervisor-based Virtualization,for Hybrid Main Memory Systems. // National Institute of Advanced Industrial Science and Technology (AIST), Tokyo, Japan, SoCC '16 Proceedings of the Seventh ACM Symposium on Cloud Computing Pages 112-125,Santa Clara, CA, USA — October 05 - 07, 2016, ACM New York, NY, USA ©2016.
- [9] Hybrid Solid State Drives: The Natural Evolution of Storage Systems. [Electronic resource] // Seagate, Inc URL: <https://www.seagate.com/ru/ru/point-of-view/sshd-and-natural-evolution-master-pov> (date of the application: 04.07.2018).
- [10] Heidecker J. MRAM Technology and Status. // Pasadena, Jet Propulsion Laboratory, California Institute of Technology, 2012. - 27 p.
- [11] Dubrovin A.A. Advantages of the hybrid NVM Host Controller Interface, with wear-leveling method. // Trudy NIISI RAN, 2018, Tom 8 №3 - pp. 61-70.
- [12] Project OSP.RU Web site of the Open Systems Publishing House. [Electronic resource] // Big Data - A New Theory And Practice. URL: <https://www.osp.ru/os/2011/10/13010990/> (date of the application: 25.03.2019)