# АЛЬТЕРНАТИВНЫЕ СОВРЕМЕННЫМ GPGPU ПЕРСПЕКТИВНЫЕ УНИВЕРСАЛЬНЫЕ И СПЕЦИАЛИЗИРОВАННЫЕ ПРОЦЕССОРЫ-УСКОРИТЕЛИ

*Адамов А. А.[1], Павлухин П.В.[2], Биконов Д.В.[3], Эйсымонт А.Л.[4], Эйсымонт Л.К.[5]*

***Аннотация:*** *Рассмотрены три варианта отечественного процессора-ускорителя, для замены современных зарубежных GPGPU в отечественных суперкомпьютерах и высокопроизводительных вычислительных системах. Первый вариант соответствует результату предполагаемого эволюционного развития зарубежных GPGPU, что предложено в проекте Echelon фирм NVIDIA/Cray. Возможно создание соответствующего отечественного варианта такого GPGPU, если воспользоваться одним из результатов проекта СКСН Ангара, существующего макета на FPGA 64-х тредового микропроцессора J7. Второй вариант похож на разработанный на базе векторного процессора заменитель GPGPU и называемый GPDSP FT-Matrix2000. Этот вариант может быть реализован за счет переработки идейно близкого отечественного векторного процессора NM6408MP (СБИС 1879ВМ8Я). Третий вариант – многотайловый процессор, тип архитектуры GreenDroid, в каждом тайле 64-х тредовое ядро и статическая память 64 KB, ядро ориентировано на эффективную работу с множеством подключенных к нему специализированных сложно-функциональных устройств.*

***Ключевые слова:*** *графические процессоры, GPGPU, мультитредовые ядра, процессоры обработки сигналов, GPDSP, многотайловые процессоры, GreenDroid.*

**DOI:** *10.21681/2311-3456-2019-4-13-21*

## 1. Введение

В работе [1] выделены пять направлений исследований и разработок по отечественной элементно-компонентной базе для суперкомпьютеров и высокопроизводительных систем. В данной работе описываются три варианта решений для наиболее важного первого направления – создания отечественного процессора-ускорителя, способного заменить современные GPGPU зарубежного производства.

Основное применение современных зарубежных GPGPU – решение задач научно-технических расчетов [17,18,19], однако они успешно применяются и для других задач: криптографические вычисления, нейровычисления [2], обработка графов [3].

GPGPU ограничены применением для решения задач с хорошей и средней пространственно-временной локализацией обращений к памяти [4], однако даже для этих задач в GPGPU есть проблемы по работе с памятью [5, 7]. Для задач с плохой пространственно-временной локализацией, на которых возникают обращения к памяти с задержками в сотни и тысячи тактов, требуются вообще другие процессоры (см. направление 2 в [1]). Компромиссное решение для усиления GPGPU – разработка для них сопроцессоров с очень большим количеством легких тредов, которые обеспечивают толерантность к таким задержкам [6].

Перед описанием трех вариантов замены GPGPU отметим, что они не единственные. Действительно, естественный вариант замены GPGPU – применение в конкретной прикладной области специализированных процессоров. Например, для нейровычислений уже создано множество нейропроцессоров [8, 9], для обработки графов активно разрабатываются разные варианты специализированных процессоров с рекордной энергоэффективностью и производительностью [10, 11, 12, 13, 14, 15]. Эти варианты решений в данной статье не рассматриваются (см. работу [1]), поскольку ставится задача найти такое же универсальное решение, что и GPGPU.

Целевые характеристики искомой отечественной замены GPGPU таковы. Пиковая производительность к 2027 году не менее, чем 15 Тфлопс на 64-х разрядной арифметике с плавающей точкой (FP64), должна быть не хуже, чем больше в 10 раз пиковой производительности микропроцессора Эльбрус. Особое внимание должно быть также уделено обеспечению эффективного решения задач на динамически изменяемых нерегулярных сетках, нейровычислениям и обработке графов.

## 2. Вариант 1: эволюция GPGPU, проект Echelon, макет микропроцессора J7

GPU Fermi появился в 2010 году, использовал технологии 40 нм и открыл поколение современных GPGPU: Kepler (28 нм, 2014), Maxwell (28 нм, 2015), Pascal (16 нм, 2016) и Volta (12 нм, 2017-2018).

1 Адамов Анлрей Анатольевич, кандидат экономических наук, Генеральный директор ЗАО »НТЦ «Модуль» , г. Москва, Россия. E-mail: a.adamov.@module.ru
2 Павлухин Павел Викторович, научный сотрудник ИПМ им.М.В.Келдыша РАН, г. Москва, Россия. E-mail: giperchuv@maile.ru
3 Биконов Дмитрий Владиленович, главный специалист ЗАО «НТЦ «Модуль», г. Москва, Россия. E-mail: d.bikonov@module.ru
4 Эйсымонт Алексей Леонидович, начальник сектора ЗАО «НТЦ «Модуль», г. Москва, Россия. E-mail: eisymont@module.ru
5 Эйсымонт Леонид Константинович, кандидат физико-математических наук, научный консультант ЗАО «НТЦ «Модуль», г. Москва, Россия. E-mail: verger-lk@yandex.ru

Пиковая производительность выросла с 1.33 TFlops (FP32) до 15.7 TFlops, причем частота выросла лишь в 3 раза. Энергоэффективность процессора выросла с 5.45 GFlops/W до 50 GFlops/W. Заметно вырос уровень параллелизма – количество мультитредовых мультипроцессоров SM увеличилось с 15 до 84. Увеличилась пропускная способность интерфейсов с внекристальной памятью, что принципиально для мультитредовых архитектур. Сейчас эта память построена на HBM-модулях 3D сборки и имеет пропускную способность до 800 GB/s. Улучшены возможности построения сетей с с GPGPU – общая пропускная способность предназначенных для этого линков NVlink достигает 300 GB/s.

Наиболее мощный GPU Volta имеет иерархическую структуру, в его состав входят 6 процессорных кластеров (GPC). GPC содержит 14 SM, а в каждый SM входят 4 мультитредовых ядра (далее – MT-ядро). Общим ресурсом быстрой памяти для этих MT-ядер в SM служит кэш команд L1I и реконфигурируемая память данных объемом 128 KB, которая может быть настроена либо на работу как кэш-память L1D, либо как адресуемая память Scratchpad.

Одно MT-ядро обеспечивает работу с 32 асинхронными тредами (WARP-ы), это проявление примененного архитектурного принципа MT (мультитредовости). Каждый WARP позволяет управлять посредством одной команды выполнением до 32-х синхронных тредов, это уже проявление архитектурного принципа SIMD. Один синхронный тред может работать с 16 32-х разрядными регистрами, так что регистровый файл MT-ядра огромен, 16384 регистра, но разбит на подблоки для WARP-ов. В MT-ядре имеются следующие функциональные устройства: 16 FP32, 16 INT32, 8 FP64, 8 LT/ST (обращений к памяти), SFU (вычисление стандартных функций) и 2 TPU (для нейровычислений). Один TPU производит операции над матрицами 4x4 из FP16 и FP32. Наличие TPU позволяет поднять пиковую производительность GPU Volta до 120 TFlops (над FP16 и FP32).

Введение устройств TPU оказалось полезным не только для нейровычислений. Пользователи научились их использовать в обычных вычислениях. Например, в работе [20] продемонстрировано, что использование смешанной 16- и 64-х разрядной арифметики позволяет получить на GPU Volta почти 3-кратный прирост производительности при решении СЛАУ с плотнозаполненной матрицей (тест Linpack) по сравнению с чисто 64-х разрядной арифметикой и прирост в 1.5 раза по энергоэффективности.

MT-ядро GPU Volta имеет важное отличие от таких ядер в предшествовавших GPU, хотя количество WARP-ов и тредов в этих ядрах осталось неизменным, а именно – 32 WARPA и 32 треда на один WARP. Изменения связаны с тем, что при решении практических задач была замечена часто возникающая деградация синхронного параллелизма тредов в WARP-е. По этой причине была введена возможность каждому треду работать по своему счетчику команд. Это повысило свободу планирования вычислений. Такая локализация управления в треде позволяет одновременно повысить энергоэффективность, поскольку не требуется пересылать операцию и данные в разные функциональные устройства MT-ядра по внутрикристальным соединениям. Это планируется использовать в перспективных версиях GPU и является признаком ослабления пользы от архитектурного принципа SIMD.

В вычислительных узлах перспективных суперкомпьютеров используется сеть из нескольких GPU под управлением CPU с применением соединений по NVLink и PCI-express. Например, в самом мощном суперкомпьютере ORNL Summit вычислительный узел – это сеть из шести GPU Volta под управлением двух CPU Power 9. Power 9 имеет интерфейсы NVLink, поэтому они могут напрямую работать с HBM-памятью GPU Volta, как и эти GPU с памятью друг друга.

Можно констатировать бурное развитие и совершенствование GPGPU, однако в этом эволюционном процессе не учитывались, по крайней мере, следующие недостатки, исправление которых ожидалось в перспективном GPU одноименного проекта Echelon [11, 7] фирм NVIDIA/Cray создания экзафлопсного суперкомпьютера. Этот GPU должен был появиться в 2020 году и реализован по технологии 7 нм.

1. Одно из главных ограничений GPGPU – возможность использования его исключительно как ведомого устройства, подчиненного CPU. Даже в тех задачах, где возможно выполнение вычислений полностью на GPU, необходимо писать довольно объемный код управляющей части для CPU. Особенно негативно это проявляется при обменах данными между GPU, как внутриузловых, так и между узлами суперкомпьютера.

   В GPU Echelon это решается введением гетерогенности, в состав этого процессора введены 8 ядер типа CPU.
2. Другой существенный недостаток, что уже упоминалось, связан с падением производительности при ветвлениях кода, когда теряется синхронность выполнения тредов в одном WARP-е.

   В GPU Echelon это решается уменьшением количества тредов в WARP-е с 32-х до четырех и их последовательным выполнением в однопоточном режиме, что позволяет использовать функциональные устройства SM одновременно и для других WARP-ов.
3. Ограниченный объем внекристальной памяти, непосредственно доступной современным GPGPU (максимум 32 GB), приводит для больших задач к необходимости организации сложной схемы обменов данными по PCI-e с CPU, что в итоге негативно сказывается на производительности.

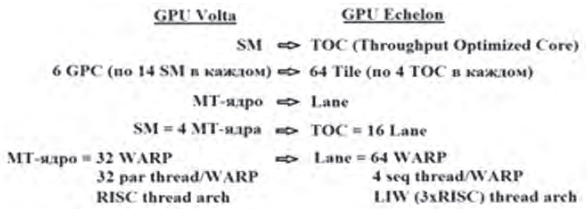
   В вычислительных узлах суперкомпьютера Echelon с одним GPU в узле внекристальная память узла в виде HBM- и GDDR-модулей, доступна через 64 контроллера памяти этого GPU, ее объем ожидается до 10 TB. Суммарная пропускная способность этих контроллеров памяти указывается равной 4 TB/s.
4. Отсутствие операций редукции (свертки) в современных GPGPU приводит к усложнению кода и снижению производительности. Это, например, существенно в процессе модификации расчетных сеточных областей (AMR-сетки, адаптивно переконфигурируемые сетки в процессе счета) в задачах газовой динамики,

которые предствляются в виде графа специального вида – восьмеричного дерева [19].

5. Внутрикристальная быстрая память SM-процессоров современных GPGPU не находится в едином адресном пространстве, что не позволяет эффективно реализовать некоторые алгоритмы (в частности, упомянутые операции редукции для AMR-сеток). Для этого приходится использовать обмены через GDDR/HBM память с большими временами выполнения обращений к ним.
   В GPU Echelon имеются 1024 блока кэш-памяти уровня L2, доступных через общую внутрикристальную сеть SM. Объем этой памяти – 128 MB.
6. Формально, в современных GPGPU разные WARP-ы могут выполнять разные участки программы. Однако программная модель CUDA предполагает использование модели однородного параллелизма, под которую ориентирована архитектура GPU. Возможность выполнения в WARP-ах разных участков программы позволила бы решать на GPU новые классы задач (например, основанных на модели master – slave). Такие исследования уже проводились [21], и позволили получить в 4 раза большую производительность. В GPU Echelon используется мультитредовая модель с разными программами в тредах.
7. Модель исполнения тредов в современных GPGPU предусматривает одновременный запуск лишь части синхронных тредов WARP-а, новый тред может начать исполняться лишь после завершения одного из уже запущенных, т.е. организуется очередь тредов. Такая модель исполнения не позволяет реализовывать алгоритмы, в которых используется сложная структура взаимодействий между тредами. Необходима более общая модель запуска тредов.

Первая информация по проекту Echelon появилась в 2011 году [22], последние сведения были в 2014 году [7]. Современное состояние проекта неизвестно, но он интересен в идейном плане как архитектурный вариант построения GPGPU, который в силу его особенностей, рассмотренных далее, доступен для реализации в нашей стране.

Структурную трансформацию GPU Volta в GPU Echelon можно представить в виде следующей схемы.




В показанной на схеме трансформации архитектур виден отказ от принципа SIMD в MT-ядре. Это ядро теперь называется Lane и в нем теперь 64 асинхронных треда (WARP-а), причем каждый такой WARP может управлять лишь последовательным выполнением до четырех тредов. Команда одного WARP-а теперь содержит до 3-х операций, т.е. она имеет тип длинного командного слова (LIW, Long Instruction Word).

Видно, что произошли упрощения на уровне мультитредового ядра (MT-ядро сравниваем с Lane), одновременно с этим увеличилось количество структурных элементов на верхних уровнях иерархии – 256 TOC вместо 84 SM, 64 тайла вместо 6 GPC, 16 Lane в TOC вместо 4 MT-ядер в SM.

Пиковая производительность GPU Echelon – 16 TFlops (FP64), пропускная способность памяти – 4 TB/s, т.е. баланс 0.25 B/F, что представляется недостаточным для 2020 года.

Мультитредовое ядро Lane близко по основным архитектурным особенностям мультитредовому ядру микропроцессора J7 российского проекта СКСН Ангара, его RTL-модель на ПЛИС была готова в АО »НИЦЭВТ» еще в 2015 году, но по непонятным причинам этот проект не продвигается. В Китае, что известно из экспертной среды, ядро J7 было использовано в нескольких военных и коммерческих изделиях [1].

В работе [7] приведены прогнозируемые характеристики суперкомпьютера Echelon с узлами на GPU Echelon и сетью малого диаметра типа Aries фирмы Cray на наборе из 6 представительных задач Министерства энергетики США и тестовой задаче HPL (LINPACK) рейтинга Top500. На задаче LINPACK поставленная цель достижения реальной производительности в экзафлопс при энергоэффективности не хуже 50 GFlops/W достигается. Развивается реальная производительность в 81% от пиковой в 1258 PFlops всего суперкомпьютера с 76800 узлами.

Вместе с тем, на двух задачах, для которых важна пропускная способность памяти (для одной из них требуется даже баланс 20 B/F, 20 байт на одну операцию FP64), а также для задачи, зависимой от межузловых обменов, результаты низкие, составляют от пиковой 1.7% и 1.9%, а также 1.8%. Для задачи, где нет таких особенностей, но характерна деградация управления в тредах WARP-ов развивается производительность в 11.2 % от пиковой. Для остальных двух задач производительность составляет около 30%.

Можно предположить, что имеются две причины, из-за которых на реальных задачах получились такие низкие показатели развиваемой производительности в сравнении с тестовой задачей Linpack.

Первая причина – меньшая доля вычислительных команд в потоке выполненных команд на задаче. Действительно, если на Linpack такая доля составляет 83.3%, то для двух других задач, чувствительных к работе с памятью, этот показатель составляет соответственно 4.7% и 13.3%. Для задачи, чувствительной к межузловым обменам, этот показатель 8.7%. Для остальных задач – около 35%.

Вывод – в процессоре Echelon имеется набор достаточно мелких по выполняемым действиям команд, поэтому их тратится слишком много для реализации подготовки данных в более сложных случаях, чем для Linpack. Уровень команд следует повысить, например, можно ввести обработку векторов. В некоторых специальных процессорах этот уровень поднят еще выше, на-

пример, имеются команды типа »умножить разреженную матрицу, хранящуюся в памяти в формате CRS, на плотно заполненный вектор».

Вторая причина – большие задержки выполнения операций с памятью и сетью, из-за которых функциональные устройства процессора простаивают. Действительно, если принять производительность выполнения команд на задаче Linpack за 100%, то такая производительность для проблемных задач 14%-34%.

Вывод – толерантность процессора Echelon к задержкам по памяти и сети должна быть повышена. Возможны два подхода: один – введение в процессор ядер с очень большим количеством легких тредов и соответствующим увеличением количества одновременно выполняемых операций с памятью и сетью; другой – введение ядер с легкими тредами во внекристальные модули памяти, как это предлагается в [6].

Подведем итоги проведенного рассмотрения первого варианта реализации отечественного GPGPU.

1. Современные зарубежные GPGPU в своем эволюционном развитии трансформируются в вариант многоядерного процессора с классическими 64-х тредовыми ядрами без сложного выделения асинхронных и синхронных тредов, множества разнообразных CUDA-ядер. Результат такой трансформации показан в проекте Echelon.
2. Мультитредовое ядро процессора проекта Echelon похоже на 64-х тредовое ядро микропроцессора J7 проекта СКСН Ангара [23, 24, 25]. Это реальная база для разработки отечественного GPGPU, но потребует реанимации проекта с добавлением в него новых организаций, имеющих компетенции по микроэлектронике и по системному программному обеспечению.
3. По результатам выполнения проекта Echelon выявились проблемы, которые зарубежные коллеги явно еще будут решать при создании перспективного зарубежного GPGPU. Это следует учитывать и при построении отечественного GPGPU на базе ядер J7.

Рассмотренный вариант оценивается как наиболее надежный в смысле вероятности получения положительного результата в виде отечественного GPGPU и более проработанный.

### 3. Вариант 2: альтернативный вариант на базе процессора обработки сигналов

В основе современных GPGPU лежит использование архитектурного принципа мультитредовости, что усилилось в GPU Volta, а более отчетливо – в проекте Echelon.

Архитектурный принцип векторной организации процессора использовался даже раньше, чем мультитредовый, для повышения производительности и одновременно обеспечения толерантности к задержкам обращения к памяти, хотя в этом плане он является менее общим, чем мультитредовость. Векторная организация показала себя более экономичной как по требуемому объему оборудования, так и по энергетике.

Китайские специалисты, также поставленные из-за санкций перед проблемой разработки процессора-ускорителя, альтернативного GPGPU, выбрали путь его разработки на базе векторной архитектуры DSP-процессоров. Для планируемого экзафлопсного суперкомпьютера Tianhe-3 в Университете оборонных технологий Китая (NUDT) разрабатывались процессоры-ускорители типа GPDSP – FT-Matrix2000, FT-Matrix2000+ и FT-Matrix3000. Опытный образец такого процессора-ускорителя, FT-Matrix2000, представлен в работе [28].

Этот процессор по идеям, заложенным в его скалярно-векторной архитектуре, достаточно близок к процессору NM6408MP [26, 27], но имеет заслуживающие внимания архитектурные особенности. Далее рассмотрим их подробнее, но сначала отметим две общие характеристики.

1. FT-Matrix2000 имеет в 10 раз большую производительность на FP64 на одинаковой частоте в 1 GHz, чем NM6408MP, реализован по технологии 45 нм на кристалле с площадью в 590 мм2, что больше площади кристалла NM6408MP в 6.7 раза.
2. FT-Matrix2000 спроектирован так, чтобы выполнение программ происходило за предсказуемое время. Очевидно, что это связано с его возможным использованием в системах реального времени.

Пиковые производительности GPDSP – 1.2 TF (FP64) и 2.4 TF (FP32). Пропускная способность 4-х интерфейсов с внекристальной памятью GPDSP объемом до 120 GB – 120 GB/s, т.е. баланс для FP64 составляет 0.1 B/F, а для FP32 – 0.05 B/F. Потребление энергии GPDSP в «типичном угле» – 150.4W, т.е. удельная мощность потребления составляет 0.25 W/мм$^2$, а энергоэффективность 8 GFlops/W для FP64.

Структура GPDSP – это сеть с передачей пакетов в виде двунаправленного кольца с шириной каждого однонаправленного канала в 512 разрядов, к которой подключены DSP-узлы и CPU-узлы, блоки выполнения вспомогательных функций. В варианте FT-Matrix2000 в этой сети имеются 8 узлов, шесть из которых DSP-узлы, 2 CPU-узла для ввода/вывода, а также следующие блоки:

- DMA-блок автономной пересылки данных между разными областями внутрикристальной и внешней памяти;
- SYNC-блок синхронизации выполнения атомарных операций с памятью и барьерных синхронизаций;
- блок интерфейса с хост-процессором (PCIe 2.0 16x, 80 Gb/s) типа CPU, такой CPU также разрабатывается для Tianhe-3, но точной информации о нем пока нет;
- блок межкристального интерфейса с двумя внешними линками по 12.5 Gb/s каждый, предназначен для построения сети из множества GPDSP.

Наибольший интерес в GPDSP представляет организация DSP-узла. В реализованном варианте FT-Matrix2000 DSP-узел содержит два DSP-ядра (называются FT-Matrix2) и секцию SubGC глобальной когерентной кеш-памяти объемом 512 KB с пропускной способностью 1024 бит/сек, т.е. 16 двойных слов за такт.

DSP-ядро содержит два основных блока:

- скалярное устройство SPU, содержащее скалярный процессорный элемент SPE и скалярную память SM, которая может использоваться либо как кэш L1 данных, либо как быстрая адресуемая память;
- векторное устройство VPU, содержащее 16 векторных процессорных элементов VPE и расслоенную на 16 блоков векторную память VM, VPE соединены друг с другом через сеть с полной перестановкой и редукцией, интерфейс VPE с памятью VM реализован отдельно.

Блоки SPU и VPU соединены для передачи данных следующими двумя средствами:

- группой регистров SVR, которые со стороны SPU рассматриваются как 16 скалярных регистров, а со стороны VPU – как 16-элементный векторный регистр, обмен в обе стороны;
- устройством широковещательной рассылки от SPU в VPU 32-х слов за такт процессора.

SPU и VPU управляются командами, считываемыми из имеющейся в DSP-ядре кэш-памяти команд Icache блоком Inst.Fetch и непосредственно выдаются в SPU и VPU еще одним блоком Inst.Dispetcher.

Команды, выполняемые SPU и VPU, имеют 40-разрядный и 80-разрядный формат. В блок Inst.Dispetcher они поступают пакетами, содержащими до 11 команд, соответственно этот блок за такт выдает одновременно 5 команд в SPU и 6 команд в VPU. Порты выдачи команд привязаны к разным функциональным устройствам SPU и VPU.

В SPE блока SPU имеются 6 функциональных устройств:

- 2 конвейерных устройства выполнения операций над числами с плавающей точкой (умножение-сложение (MAC), операции над FP64 и парами FP32 (2xFP32));
- конвейерное устройство BP выполнения операций над целочисленными (INT32 и INT64) и битовыми (BIT64) данными;
- устройство вычисления базовых элементарных функций;
- устройство деления;
- устройство извлечения квадратного корня.

SPE имеет следующий регистровый ресурс:

- 64 64-разрядных регистров общего назначения;
- 16 36-разрядных адресных регистров базы;
- 16 36-разрядных адресных регистров смещения.

Адресные регистры используются для доступа к скалярным и векторным данным, во втором случае пары регистров используются как адресные генераторы элементов векторов.

Каждый из шестнадцати VPE блока VPU включает:

- 2 регистровых файла векторных регистров, один файл – 32 регистра, каждый регистр – 16 64-х разрядных элемента;
- 3 конвейерных устройства выполнения операций над числами с плавающей точкой (MAC, FP64, 2xFP32);
- конвейерное устройство BP целочисленных и битовых операций.

Один регистровый файл имеет 13 портов чтения и 8 портов записи, что необходимо четырем конвейерным функциональным устройствам VPE, а также для операций чтения/записи с памятью VM.

В одном VPE на трех конвейерных устройствах за такт выполняется до трех операций MAC (MAC – 2 операции над FP64), на 16 VPE выполняется ~ 50 MAC (100 операций над FP64 и 200 операций над 2xFP32). VPU работает на частоте 1 GHz, следовательно его производительность для FP64 – 100 GFlops, для FP32 – 200 GFlops.

Как отмечалось, DSP-ядро FT-matrix2 по архитектурной идее сопоставимо с nmc4-ядром отечественного микропроцессора NM6408MP [26, 27], хотя при разработке NM6408MP не ставилась задача заменить GP-GPU. В работе [29] был приведен вариант развития этого процессора. Сравнение nmc4 с FT-matrix2 дает еще один вариант развития отечественного процессора, отметим следующее, учитывая, что в NM6408MP имеется 16 nmc4-ядер, а в GPDSP – 12 ядер FT-Matrix2, площади этих ядер соотносятся как 0.7 мм$^2$ (nmc4 + локальная память ядра в 512 KB, технология 28 нм) и 26.5 мм$^2$ (технология 40 нм):

- скалярная часть nmc4 32-х разрядная (FT-Matrix2 – 64-х разрядная);
- скалярная часть nmc4 содержит одно устройство обработки данных (FT-Matrix2 – 6 устройств обработки вещественных и целых чисел, двоичных кодов, вычисления функций);
- производительность nmc4 на FP64 – 8 GFlops (FT-Matrix2 – 100 GFlops), на FP32 – 32 GFlops (FT-Matrix2 - 200 GFlops);
- векторных ячеек Cell в nmc4 – 4 (FT-Matrix2 – 16 VPE);
- конвейерных устройств в одном Cell nmc4 – 1 (FT-Matrix2 – 4);
- векторных регистров в одном Cell nmc4 – 8 по 32 элемента каждый (FT—Matrix2 – 64 по 16 элементов каждый);
- разрядность команд nmc4 – 32 и 64 (FT-Matrix2 – 40 и 80, VLIW до 11 команд);
- темп выдачи команд за такт в nmc4 – одна/две в скалярную часть или одна в векторную часть (FT-Matrix2 – 5 команд в скалярную часть и 6 команд в векторную часть за такт);
- скалярная и векторная части nmc4 имеют доступ к расслоенной на 8 блоков локальной памяти объемом 512 KB (FT-Matrix2 – имеются отдельно и только для данных скалярная память SM и векторная память VM, команды выбираются из кэш-памяти команд, куда подкачиваются через внешний интерфейс ядра, VM разделена на 16 блоков и имеет общий объем около 1 MB, один блок может выполнять 4 обращения одновременно (2 чтения/записи для VPE, одно чтение для DMA и одна запись для DMA), т.е. для 16 VPE и DMA обеспечивается пропускная способность памяти в 64 слова за такт);
- доступ к локальной памяти для векторов для nmc4 может быть одномерный и двумерный с

регулярным шагом (FT-Matrix2 – может быть и нерегулярным, имеются команды доступа, приспособленные для FFT-преобразований);
- имеется один блок MDMAC для пересылки данных на четыре ядра nmc4 (FT-Matrix2 – один DMA на каждое ядро, который позволяет при пересылке данных производить перестановки строк и столбцов матриц, что сильно ускоряет транспонирование матриц).

GPDSP FT-Matrix2000 еще имеет интересные особенности в работе контроллеров внекристальной памяти, обеспечивающие настраиваемое циклическое и блочно-циклическое отображение логических адресов на физические.

Программное обеспечение GPDSP кроме обычных средств нижнего уровня включает компилятор языка Си с векторным расширением и библиотеку MPI.

В работе [28] приведены результаты исследования отдельно GPDSP, а также фрагмента 16-узлового суперкомпьютера с пиковой производительностью 160 TFlops, каждый вычислительный узел которого содержит 4 CPU и 4 FT-Matrix2000.

Приведем лишь результаты сравнения одного FT-Matrix2000 с GPU Fermi (2010 год, 40 нм). Площади кристаллов соотносятся как 520 мм$^2$ у GPU Fermi и 600 мм$^2$ у FT-Matrix2000. На задаче умножения плотно заполненных матриц производительность FT-Matrix2000 – 1107 GFlops (92.25% от пиковой), производительность GPU Fermi – 350 GFlops (67.9 %). Эффективность FT-Matrix2000 – 6.14 GFlops/W, GPU Fermi – 2.29 GFlops/W.

В заключение раздела отметим, что конкурентноспособность GPDSP процессору-ускорителю GPGPU можно считать по результатам работы [28] доказанной, хотя отдельные предложения по улучшению его архитектуры можно было бы предложить. Архитектуру GPDSP целесообразно учесть при разработке отечественной альтернативы GPGPU на базе NM6408MP, если такая задача будет поставлена.

### 4. Вариант 3: альтернативный вариант на базе мультитредовых ядер и специализированных сложно-функциональных устройств

Рассматриваемые варианты 1 и 2 связаны с созданием достаточно универсального процессора-ускорителя основанного соответственно на мультитредовой и векторной архитектурной модели. Логично предположить, что возможен и третий вариант, в котором используются обе эти модели. Более того, есть несколько примеров исследований и разработок, подтверждающих это. Например, мультитредовая модель Cray XMT в заказных суперкомпьютерах для разведывательных центров США, Китая и Японии была усилена операциями над короткими и длинными векторами. Можно привести и другие примеры, однако этот вариант в данной работе не рассматривался из практических соображений минимизации времени исследований и разработки отечественного заменителя зарубежных GPGPU.

Предлагаемый далее вариант 3 не ставит своей целью создание достаточно универсального процессора-ускорителя. Этот вариант можно рассматривать как средний вариант по специализации между, с одной стороны, вариантами 1 и 2, а с другой – полностью специализированных процессоров, о которых говорилось в начале статьи и которые наиболее сильно заменяют применение GPGPU в той или иной прикладной области.

Предлагаемый вариант 3 – это многотайловый (многоузловой) микропроцессор архитектурного направления GreenDroid [32], получившего такое название вследствие потенциально достижимой высокой энергоэффективности при высокой производительности.

Предлагается в каждом тайле иметь 64-х тредовое ядро и статическую память объемом не менее 64 KB с DMA-интерфейсами для подключения нескольких сложно-функциональных специализированных на некоторую выбранную предметную область устройств (SFU). Например, для нейровычислений или решения задач криптографии.

Главная функция мультитредового ядра такого тайла – обеспечение плотной загрузки данными таких SFU. Примеры такого подхода – высокопроизводительные ПОС-СБИС Antminer E3, BM 1680, Celerity [1].

История работ по этому варианту такова. После закрытия проекта СКСН Ангара в инициативном порядке была разработана более простая мультитредовая архитектура ядер и их схемная реализация для решения задач информационной безопасности – к 2015 году был закончен проект многоядерного микропроцессора К4. Были получены теоретические оценки реализации ядер К4 по технологии 65 нм. Получилось что 64-тредовое ядро с 64 KB SRAM-памяти занимает площадь кристалла – 3.4 мм$^2$, имеет мощность потребления 1.5 Вт на частоте 1 GHz, т.е. удельная мощность 0.44 Вт/мм$^2$.

Далее, в 2017-2018 годах была разработана архитектура и схемы реализации блоков нового варианта 64-х тредового ядра mt-LWP с улучшенной системой команд и ориентацией на работу с подключаемыми к нему SFU через DMA.

Оценки сложности аппаратной реализации mt-LWP по его схемам не производились, но если использовать результаты близкого проекта микропроцессора К4, то можно предположить следующее.

При реализации по технологии 28 нм многотайлового кристалла с мультитредовыми ядрами mt-LWP и памятью 64 KB в тайлах, подключенными к ним в качестве SFU четырьмя функциональными устройствами FPU ядер nmc4 процессора NM6408MP получается, что на площади ~ 94 мм$^2$ можно разместить 128 таких тайлов и достичь пиковой производительности в 4 TFlops (FP32). Такой вариант мог бы использоваться как нейропроцессор.

Возможен вариант с подключенными в тайлах SFU другого типа, ориентированный на решение задач информационной безопасности. В данном случае в качестве SFU могут быть подключены перестраиваемые устройства DRCP реализации глубоких конвейерных алгоритмов блочного и поточного шифрования, вычисления хеш-функций [31]. Если в таком кристалле удастся выйти на требуемые характеристики, то объем заказа на их изготовление может оказаться не менее 300 тысяч.

## 5. Заключение

В статье рассмотрены два варианта реализации отечественного процессора-ускорителя для использования вместо применяемых GPGPU зарубежного производства, а также третий вариант более специализированного процессора-ускорителя для особенно важных областей использования GPGPU, которые по производительности приближаются к специализированным процессорам для этих областей, но отличаются большей гибкостью использования (это Тип 2 проблемно-ориентированных специализированных СБИС работы [30]). По этим вариантам в ЗАО «НТЦ «Модуль» имеются определенные заделы и уже предпринимаются попытки в соответствии с общим подходом к инновационным исследованиям, изложенным в работе [1], организовать опережающие инициативные исследования.

## 6. Выводы

1. Разработка процессоров-ускорителей на замену зарубежных GPGPU в суперкомпьютерах – задача №1 в области создания отечественной ЭКБ.
2. Процессоры-ускорители должны обладать производительностью и энергоэффективностью для решения современных научно-технических задач, выполнения алгоритмов глубокого обучения и обработки графов.
3. Следует учесть, что применение современных GPGPU актуально не только в суперкомпьютерах, но и в современных высокотехнологичных системах оружия и системах искусственного интеллекта.
4. Архитектура GPGPU, появившаяся в конце прошлого десятилетия и прошедшая путь развития от Fermi до Volta, претерпевает изменения и появились альтернативы:
   - рассматривается вариант более однородной асинхронной массово-мультитредовой архитектуры (проект NVIDIA/Cray Echelon);
   - реализован китайский процессор-ускоритель на базе DSP-архитектуры (GPDSP, проекты NUDT FT-Matrix2000, Matrix2000+, Matrix3000);
   - существуют варианты многотайловых гибридных архитектур с мультитредовыми ядрами в тайлах и подключенными к ним специализированными функциональными устройствами (тип GreenDroid)
5. Не имеет особого смысла, повторение в России реализации GPGPU типа современных, тем более, по устаревшим лицензиям, если их удастся приобрести. Однако возможна реализация перспективных альтернативных современным GPGPU вариантов.

## Литература


1. Адамов А.А., Фомин Д.В., Эйсымонт Л.К. Главные проблемные направления в области отечественной элементной базы суперкомпьютеров // Вопросы кибербезопасности, номер 4, 2019, с. 2-12.
2. Shlegel D. Deep Machine Learning on GPUs // Seminar Talk – Deep Machine Learning on GPUs, 2015, 6 pp.
3. Shi X. [et al.] Graph Processing on GPUs: A Servey // ACM Computing Surveys, Vol.50, № 6, article 81, Jan 2018, 35 pp.
4. Murphy R.C., Kogge P.M. On the Memory Access Patterns of Supercomputer Applications: Benchmark Selection and Its Implications // IEEE Transactions on Computers, Vol.56, №7, July 2007, 9 pp.
5. Li C. [et al.] Optimizing Memory Efficiency for Deep Convolutional Neural Networks on GPUs // International Conference on High Performance Computing, Networking, Storage and Analysis (SC'16), 2016, 12 pp.
6. Kogge P.M. [et al.] Computer Systems with lightweight multi-threaded architectures // US Patent, US2007/0198785 A1, Aug 23, 2007, 34 pp.
7. Oreste W. [et al.] Scaling the Power Wall: A Path to Exascale // Supercomputing Conference (SC'14), November 16-21, 2014, 12 pp.
8. Sze V. [et al.] Efficient Processing of Deep Neural Networks: A Tutorial and Survey // arXiv:1703.09039v2 [cs.CV] 13 Aug 2017, 32 pp.
9. Akopyan F. [et al.] TrueNoth: Design and Tool Flow of a 65 mW 1 Million Neuron Programmable Neurosynaptic Chip // IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, Vol.34, №10, October 2015, pp.1537-1557.
10. Song W.S., Gleyzer V., Lomakin A., Kepner J. Novel Graph Processor Architecture, Prototype System, and Results // 2016 IEEE High Performance Extreme Computing Conference, 22 July 2016, 7 pp, http://arhiv.org/abs/1607.06541
11. Song W.S. Processor for large graph algorithm computations and matrix operations // US Patent No 9,529,590 B2, Dec 27, 2016
12. Ahn J. [et al.] A Scalable Processing-in-Memory Accelerator for Parallel Graph Processing // ISCA'15, June 2015, 13 pp.
13. Chu Y. [et al.] NXgraph: An Efficient Graph Processing System on Single Machine // arXiv: 15100691v1 [cs.DB] 23 Oct 2015, 12 pp.
14. Dai G. [et al.] GraphH: A Processing-in-Memory Architecture for Large-scale Graph Processing // IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems , Vol 38 , №4, April 2019, p.640-653.
15. Li G. [et al.] GraphIA: An In-situ Accelerator for Large-scale Graph Processing // MEMSYS, Oct 2018, 6 pp.
16. Durant L. [et al.] Inside Volta: The World's Most Advanced Data Center GPU, 10 may 2017, https://devblogs.nvidia.com/parallelforall/inside-volta/
17. Pavlukhin P., Menshov I. On Implementation High-Scalable CFD Solvers for Hybrid Clusters with Massively-Parallel Architectures // Lecture Notes in Computer Science, 2015, том 9251, pp. 436-444.
18. Menshov I., Pavlukhin P. Highly scalable implementation of an implicit matrix-free solver for gas dynamics on gpu – accelerated clusters // Journal of Supercomputing, Kluwer Academic Publishers (Netherlands), 1-8, 2016.
19. Pavlukhin P. , Menshov I, Parallel implicit matrix-free CFD solver using AMR grids // Journal of Physics Conference Series 1141:012035, 2018

20. Haidar A., Tomov S., Dongarra J., Higham N.J. Harnessing GPU tensor cores for fast FP16 arithmetic to speed up mixed-precision iterative refinement solvers // SC ‹18 Proceedings of the International Conference for High Performance Computing, Networking, Storage, and Analysis, Article No. 47, 2018
21. Bauer M., Treichler S., Aiken A. Singe: Leveraging Warp Specialization for High Performance on GPUs // ACM SIGPLAN Notices, 2014, №49, pp. 119-130
22. Keckler S.W., Dally W.J. [et al.] GPUs and the Future of Parallel Computing. IEEE MICRO, September/Octouber 2011, pp.7-17.
23. Слуцкин А.И., Эйсымонт Л.К. Российский суперкомпьютер с глобально адресуемой памятью // Открытые системы, №9, 2007, с. 42-51. http://www.osp.ru/os/2007/09/4569294/
24. Митрофанов В.В., Слуцкин А.И., Эйсымонт Л.К. Суперкомпьютерные технологии для стратегически важных задач // Электроника: НТБ, №7, 2008, с. 66-79.
25. Семенов А.А., Соколов А.А., Эйсымонт Л.К. Архитектура глобально адресуемой памяти мультитредово – потокового суперкомпьютера // Электроника: НТБ, №1, 2009 г., с. 50-56.
26. Эйсымонт А.Л., Черников В.М., Черников Ан.В., Черников Ал.В., Косоруков Д.Е., Насонов И.И., Комлев А.А. Гетерогенная многопроцессорная система на кристалле с производительностью 512 Gflops // Системы высокой доступности, 2018, т.14, №3, стр.49-56.
27. Биконов Д.В., Сивцов А.С., Пузиков А.Д., Эйсымонт Л.К. Трехуровневая система параллельного программирования 21-ядерного скалярно-векторного микропроцессора NM6408MP // Вопросы кибербезопасности, номер 4, 2019, с. 22-34.
28. Chao Y. [et al.] A Novel DSP Architecture for Scientific Computing and Deep Learning // IEEE Access, Vol 7, April 2, 2019, pp 36413-36425.
29. Черников В.М., Виксне П.Е. Перспективы повышения характеристик и расширения областей применения транстерафлопсных СБИС семейства NeuroMatrix // Системы высокой доступности, 2018, т.14, №3, стр.28-34.
30. Эйсымонт Л.К. Настраиваемые специализированные СБИС – реальная основа создания будущих экзамасштабных суперкомпьютеров, зарубежный и отечественный опыт // Системы высокой доступности, 2018, т.14, №3, стр.18-27.
31. Андрюшин Д.В. и др. Реконфигурируемый вычислительный модуль // Патент на изобретение, RU 2 686 017, Дата регистрации 23.04.2019, 24 стр.
32. M.Khazraee, L.V.Gutierrez, I.Magaki, M.B.Tailor. Specializing a Planet's Computation: ASIC Clouds. // IEEE Micro, May/June 2017, pp. 62-69.

# MODERN GPGPU ALTERNATIVE PERSPECTIVE UNIVERSAL AND SPECIALIZED PROCESSORS-ACCELERATORS

***Adamov A. A.[6], Pavluxin P.V.[7], Bikonov D.V.[8], Eisymont A.L.[9], Eisymont L.K.[10]***

***Abstract:*** Three variants of a domestic accelerator processor are considered to replace modern foreign GPGPUs in domestic supercomputers and high-performance computing systems. The first variant corresponds to the result of the supposed evolutionary development of foreign GPGPUs, which was proposed in the Echelon project of NVIDIA / Cray firms. It is possible to create a corresponding domestic version of such a GPGPU, if we use one of the results of the HEC Angara project, an existing FPGA realization of the 64-thread J7 microprocessor. The second option is similar to the GPGPU substitute developed on the basis of the vector processor and called the FT-Matrix2000 GPDSP. This option can be realized by redevelopment of the ideologically close domestic vector processor NM6408MP (VLSI 1879ВМ8Я). The third option is a multi-tile processor, a type of GreenDroid architecture, in each tile there are 64 thread kernel and 64 KB static memory, the core is focused on efficient work with many specialized hard-functional devices connected to it.

***Keywords:*** GPUs, GPGPU, multitreaded cores, signal processors, GPDSP, multi-core processors, GreenDroid,

## Referencies

1. Adamov A.A., Fomin D.V., Ejsymont L.K. Glavnye problemnye napravleniya v oblasti otechestvennoj elementnoj bazy superkomp'yuterov // Voprosy kiberbezopasnosti, nomer 4, 2019, s. 2-12.
2. ShlegelD. Deep Machine Learningon GPUs // Seminar Talk – Deep Machine Learning on GPUs, 2015, 6 pp.

6 Andrey Adamov, Ph.D. (Econ.), General Director, Recearch Center »Module», Moscow, Russia. E-mail: a.adamov@module.ru
7 Pavel Pavlukhin, Researcher, IPM im.M.V.Keldysh RAS, Moscow, Russia. E-mail: giperchuv@mail.ru
8 Dmitriy Bikonov, Chief specialist, Recearch Center « Module », Moscow, Russia. E-mail: d.bikonov@module.ru
9 Alexey Eisymont, Chief of sector, Recearch Center «Module» Module, Moscow, Russia. E-mail: eisymont@module.ru
10 Leonid Eisymont, Ph.D (Physical and Mathematical Sciences), Scientific consultant, Recearch Center «Module», Moscow, Russia. E-mail: verger-lk@yandex.ru

3. ShiX. [etal.] Graph Processing on GPUs: A Servey // ACM Computing Surveys, Vol.50, № 6, article 81, Jan 2018, 35 pp.
4. Murphy R.C., Kogge P.M. On the Memory Access Patterns of Supercomputer Applications: Benchmark Selectionand Its Implications // IEEE Transactions on Computers, Vol.56, №7, July 2007, 9 pp.
5. Li C. [et al.] Optimizing Memory Efficiency for Deep Convolutional Neural Networks on GPUs // International Conference on High Performance Computing, Networking, Storage and Analysis (SC'16), 2016, 12 pp.
6. Kogge P.M. [et al.] Computer Systems with lightweight multi-threaded architectures // US Patent, US2007/0198785 A1, Aug 23, 2007, 34 pp.
7. Oreste W. [et al.] Scaling the Power Wall: A Path to Exascale // Supercomputing Conference (SC'14), November 16-21, 2014, 12 pp.
8. Sze V. [et al.] Efficient Processing of Deep Neural Networks: A Tutorial and Survey // arXiv:1703.09039v2 [cs.CV] 13 Aug 2017, 32 pp.
9. Akopyan F. [et al.] TrueNoth: Design and Tool Flow of a 65 mW 1 Million Neuron Programmable Neurosynaptic Chip // IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, Vol.34, №10, October 2015, pp.1537-1557.
10. Song W.S., Gleyzer V., Lomakin A., Kepner J. Novel Graph Processor Architecture, Prototype System, and Results // 2016 IEEE High Performance Extreme Computing Conference, 22 July 2016, 7 pp, http://arhiv.org/abs/1607.06541
11. Song W.S. Processor for large graph algorithm computations and matrix operations // US Patent No 9,529,590 B2, Dec 27, 2016
12. Ahn J.[et al.] A Scalable Processing-in-Memory Accelerator for Parallel Graph Processing // ISCA'15, June 2015, 13 pp.
13. Chu Y. [et al.]NXgraph: An Efficient Graph Processing System on Single Machine // arXiv: 15100691v1 [cs.DB] 23 Oct 2015, 12 pp.
14. Dai G. [et al.] GraphH: A Processing-in-Memory Architecture for Large-scale Graph Processing // IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems , Vol 38 , №4, April 2019, p.640-653.
15. Li G. [et al.]GraphIA: An In-situ Accelerator for Large-scale Graph Processing // MEMSYS, Oct 2018, 6 pp.
16. 38. Durant L. [et al.] Inside Volta: The World's Most Advanced Data Center GPU, 10 may 2017, https://devblogs.nvidia.com/parallelforall/inside-volta/
17. PavlukhinP., Menshov I. On Implementation High-Scalable CFD Solvers for Hybrid Clusters with Massively-Parallel Architectures // Lecture Notes in Computer Science, 2015, том9251, pp. 436-444.
18. Menshov I., Pavlukhin P. Highly scalable implementation of an implicit matrix-free solver for gas dynamics on gpu-accelerated clusters // Journal of Supercomputing, Kluwer Academic Publishers (Netherlands), 1-8, 2016.
19. Pavlukhin P. , Menshov I, Parallel implicit matrix-free CFD solver using AMR grids // Journal of Physics Conference Series 1141:012035, 2018
20. Haidar A., Tomov S., Dongarra J., Higham N.J. Harnessing GPU tensor cores for fast FP16 arithmetic to speed up mixed-precision iterative refinement solvers // SC '18 Proceedings of the International Conference for High Performance Computing, Networking, Storage, and Analysis, Article No. 47, 2018
21. Bauer M., Treichler S., Aiken A. Singe: Leveraging Warp Specialization for High Performance on GPUs // ACM SIGPLAN Notices, 2014, №49, pp. 119-130
22. Keckler S.W., Dally W.J. [et al.] GPUs and the Future of Parallel Computing. IEEE MICRO, September/Octouber 2011, pp.7-17.
23. Sluckin A.I., EjsymontL.K. Rossijskij superkomp'yuter s global'no adresuemoj pamyat'yu // Otkrytye sistemy, №9, 2007, s. 42-51. http://www.osp.ru/os/2007/09/4569294/
24. Mitrofanov V.V., Sluckin A.I., Ejsymont L.K. Superkomp'yuternye tekhnologii dlya strategicheski vazhnyh zadach // Elektronika: NTB, №7, 2008, s. 66-79.-
25. Semenov A.A., Sokolov A.A., Ejsymont L.K. Arhitektura global'no adresuemoj pamyati mul'titredovo-potokovogo superkomp'yutera // Elektronika: NTB, №1, 2009 g., s. 50-56.
26. Ejsymont A.L., CHernikov V.M., CHernikov An.V., CHernikov Al.V., Kosorukov D.E., Nasonov I.I., Komlev A.A. Geterogennaya mnogoprocessornaya sistema na kristalle s proizvoditel'nost'yu 512 Gflops// Sistemy vysokoj dostupnosti, 2018, t.14, №3, str.49-56.
27. Bikonov D.V., Sivcov A.S., Puzikov A.D., Eisymont L.K. Trekhurovnevaya sistema parallel'nogo programmirovaniya 21-yadernogo skalyarno-vektornogo mikroprocessora NM6408MP // Voprosy kiberbezopasnosti, Tom..., nomer 4, 2019, s. 12-34.
28. ChaoY. [etal.] A Novel DSP Architecture for Scientific Computing and Deep Learning // IEEE Access, Vol 7, April 2, 2019, pp 36413-36425.
29. CHernikov V.M., Viksne P.E. Perspektivy povysheniya harakteristik i rasshireniya oblastej primeneniya transteraflopsnyh SBIS semejstva NeuroMatrix // Sistemy vysokoj dostupnosti, 2018, t.14, №3, str.28-34.
30. Ejsymont L.K. Nastraivaemye specializirovannye SBIS – real'naya osnova sozdaniya budushchih ekzamasshtabnyh superkomp'yuterov, zarubezhnyj i otechestvennyj opyt // Sistemy vysokoj dostupnosti, 2018, t.14, №3, str.18-27.
31. Andryushin D.V. i dr. Rekonfiguriruemyj vychislitel'nyj modul' // Patent na izobretenie, RU 2 686 017, Data registracii 23.04.2019, 24 str.
32. M.Khazraee, L.V.Gutierrez, I.Magaki, M.B.Tailor. SpecializingaPlanet'sComputation: ASICClouds. // IEEE Micro, May/June 2017, pp. 62-69.M.Khazraee, L.V.Gutierrez, I.Magaki, M.B.Tailor. SpecializingaPlanet'sComputation: ASICClouds. // IEEE Micro, May/June 2017, pp. 62-69.