Федеральное государственное бюджетное образовательное учреждение высшего образования

«Сибирский государственный университет телекоммуникаций и информатики»

(СибГУТИ)

Кафедра ВС

РАСЧЕТНО-ГРАФИЧЕСКОЕ ЗАДАНИЕ

по дисциплине «Архитектура Вычислительных Систем»

Вариант 18

Выполнил: студент гр. ИП-712

Алексеев С.В.

Проверил: Ефимов А.В.

Новосибирск 2019 г.

Оглавление

[1. ЗАДАНИЕ 2](#_Toc25439342)

[2. АНАЛИЗ ВОЗМОЖНОСТЕЙ ПРОЦЕССОРОВ С МИКРОАРХИТЕКТУРОЙ ЭЛЬБРУС 2](#_Toc25439343)

[3. ПРИМЕР ФУНКЦИОНАЛЬНОЙ СТРУКТУРЫ СОВРЕМЕННОГО ПРОЦЕССОРА 6](#_Toc25439344)

[4. РАСЧЕТ ФУНКЦИЙ НАДЕЖНОСТИ И ГОТОВНОСТИ 10](#_Toc25439345)

[5. СПИСОК ИСПОЛЬЗУЕМОЙ ЛИТЕРАТУРЫ 16](#_Toc25439346)

# 1. ЗАДАНИЕ

1. Произвести анализ возможностей процессоров с микроархитектурой Эльбрус. Привести пример функциональной структуры современного процессора.

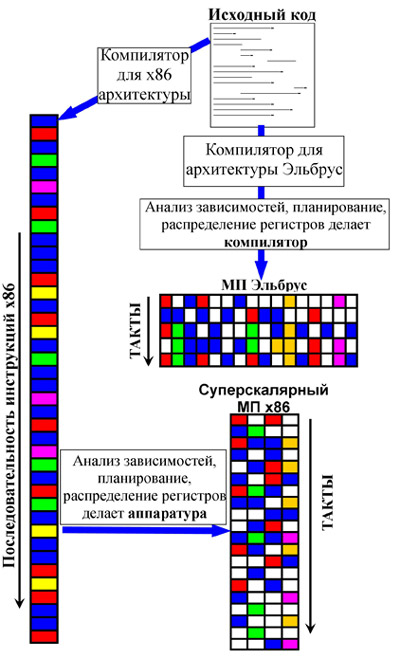
2. Выполнить численный расчет и построить графики для функции *r*(*t*) надежности и функции S(i, t) готовности ЭВМ для следующих количественных характеристик:

– интенсивности отказов 10-2 1/*ч* ,

– интенсивности восстановления 1 1/*ч* .

# 2. АНАЛИЗ ВОЗМОЖНОСТЕЙ ПРОЦЕССОРОВ С МИКРОАРХИТЕКТУРОЙ ЭЛЬБРУС

Процессоры данной архитектуры сравниваются по производительности с современными процессорами фирмы Интел в подсчете ГигаФлопс, при том, что имеют более скромные характеристики по частоте. Например при тактовой частоте всего в 1,3 ГГц Эльбрус 8С имеет производительность 250 гигафлопс (то есть 250 млрд операций в секунду) на операциях с одинарной точностью, что сравнимо с последними поколениями современного семейства процессоров Intel с архитектурой X86-64: 6-ядерный Intel Core i7-4930K (поколение Ivy Bridge) при тактовой частоте 3,7-4,2 ГГц имеет теоретический пик производительности в районе 177 гигафлопс.. Это достигается благодаря более продуманному распределению инструкций для процессора Эльбрус.



В традиционных архитектурах типа RISC или CISC (х86, PowerPC, SPARC, MIPS, ARM), на вход процессора поступает поток инструкций, которые рассчитаны на последовательное исполнение. Процессор может детектировать независимые операции и запускать их параллельно (суперскалярность) и даже менять их порядок (внеочередное исполнение). Однако динамический анализ зависимостей и поддержка внеочередного исполнения имеет свои ограничения: лучшие современные процессоры способны анализировать и запускать до 4-х команд за такт. Кроме того, соответствующие блоки внутри процессора потребляют заметное количество энергии.

В архитектуре «Эльбрус» основную работу по анализу зависимостей и оптимизации порядка операций берет на себя компилятор. Процессору на вход поступают т.н. «широкие команды», в каждой из которых закодированы инструкции для всех исполнительных устройств процессора, которые должны быть запущены на данном такте. От процессора не требуется анализировать зависимости между операндами или переставлять операции между широкими командами: все это делает компилятор, исходя из анализа исходного кода и планирования ресурсов процессора. В результате аппаратура процессора может быть проще и экономичнее.

Компилятор способен анализировать исходный код гораздо тщательнее, чем аппаратура RISC/CISC процессора, и находить больше независимых операций. Поэтому в архитектуре Эльбрус больше параллельно работающих исполнительных устройств, чем в традиционных архитектурах, и на многих алгоритмах она демонстрирует непревзойденную архитектурную скорость.

**Возможности архитектуры Эльбрус:**

* 6 каналов арифметико-логических устройств (АЛУ), работающих параллельно.
* Регистровый файл из 256 84-разрядных регистров.
* Аппаратная поддержка циклов, в том числе с конвейеризацией. Повышает эффективность использования ресурсов процессора.
* Программируемое асинхронное устройство предварительной подкачки данных с отдельными каналами считывания. Позволяет скрыть задержки от доступа к памяти и полнее использовать АЛУ.
* Поддержка спекулятивных вычислений и однобитовых предикатов. Позволяет уменьшить число переходов и параллельно исполнять несколько ветвей программы.
* Широкая команда, способная при максимальном заполнении задать в одном такте до 23 операций (более 33 операций при упаковке операндов в векторные команды).

# 3. ПРИМЕР ФУНКЦИОНАЛЬНОЙ СТРУКТУРЫ СОВРЕМЕННОГО ПРОЦЕССОРА «Эльбрус-8C»

Микропроцессор «Эльбрус-8C» спроектирован и изготовлен по технологическим нормам 28 нм. Его кристалл имеет площадь 321 мм2 и содержит 2,73 млрд транзисторов. За основу взят процессор Эльбрус-4С. Частота 1.3Ггц. 250GFLOPS на опреациях с одинарной точностью при частоте 1.3Ггц. Эльбрус-8С - это пятое поколение процессоров Эльбрус.

Топологический план приведен на рисунке 1.

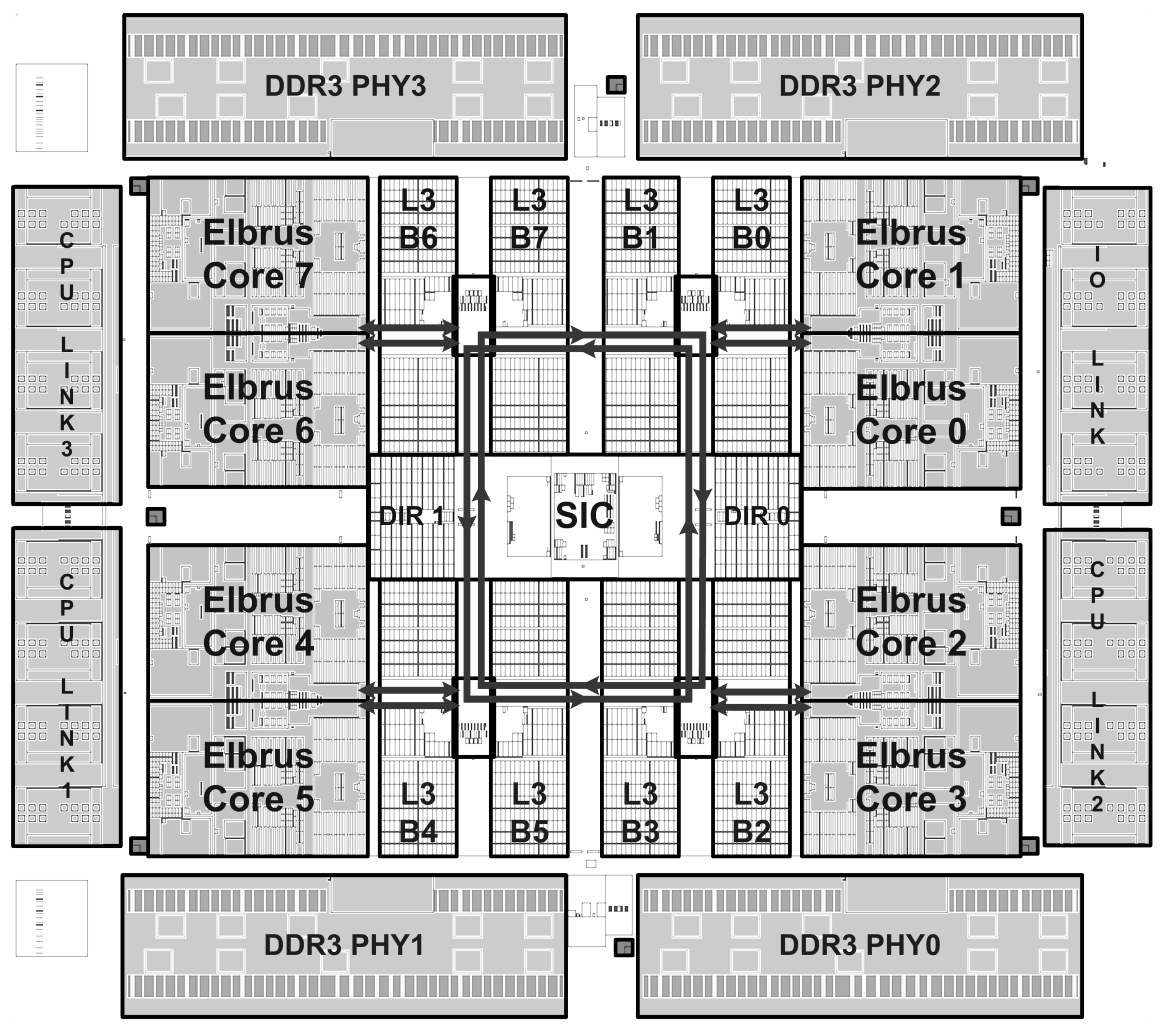


Рисунок 1. Топологический план микропроцессора «Эльбрус-8C»:

Elbrus Core 0-7 – процессорные ядра; L3 B0-7 – банки кэш-памяти третьего уровня; SIC – контроллер системных обменов; DIR 0,1 – глобальный справочник, DDR3 PHY0-3 – физуровни памяти; CPU LINK1,2,3 – физуровни каналов межпроцессорного обмена; IO LINK – физуровень канала ввода-вывода

На одном кристалле размещены восемь процессорных ядер архитектуры «Эльбрус» (Elbrus Core 0-7), каждое из которых способно исполнять до 25 операций за такт в скалярном режиме (32- и 64-разрядные данные) и до 41 операции за такт в векторном режиме (упакованные 32-разрядные данные). Ядра подключены к общей кэш-памяти третьего уровня (L3 кэш, 16 МБ), которая имеет распределенную структуру с разбиением на восемь независимых банков (L3 B0-7), причем каждое ядро имеет доступ ко всем банкам. Ядра и банки общей кэш-памяти объединены в узлы Quart (кварты) по два ядра и два банка в каждом. Кварты соединены двунаправленным буферизующим кольцом. На рисунке 2 показана структурная схема микропроцессора.

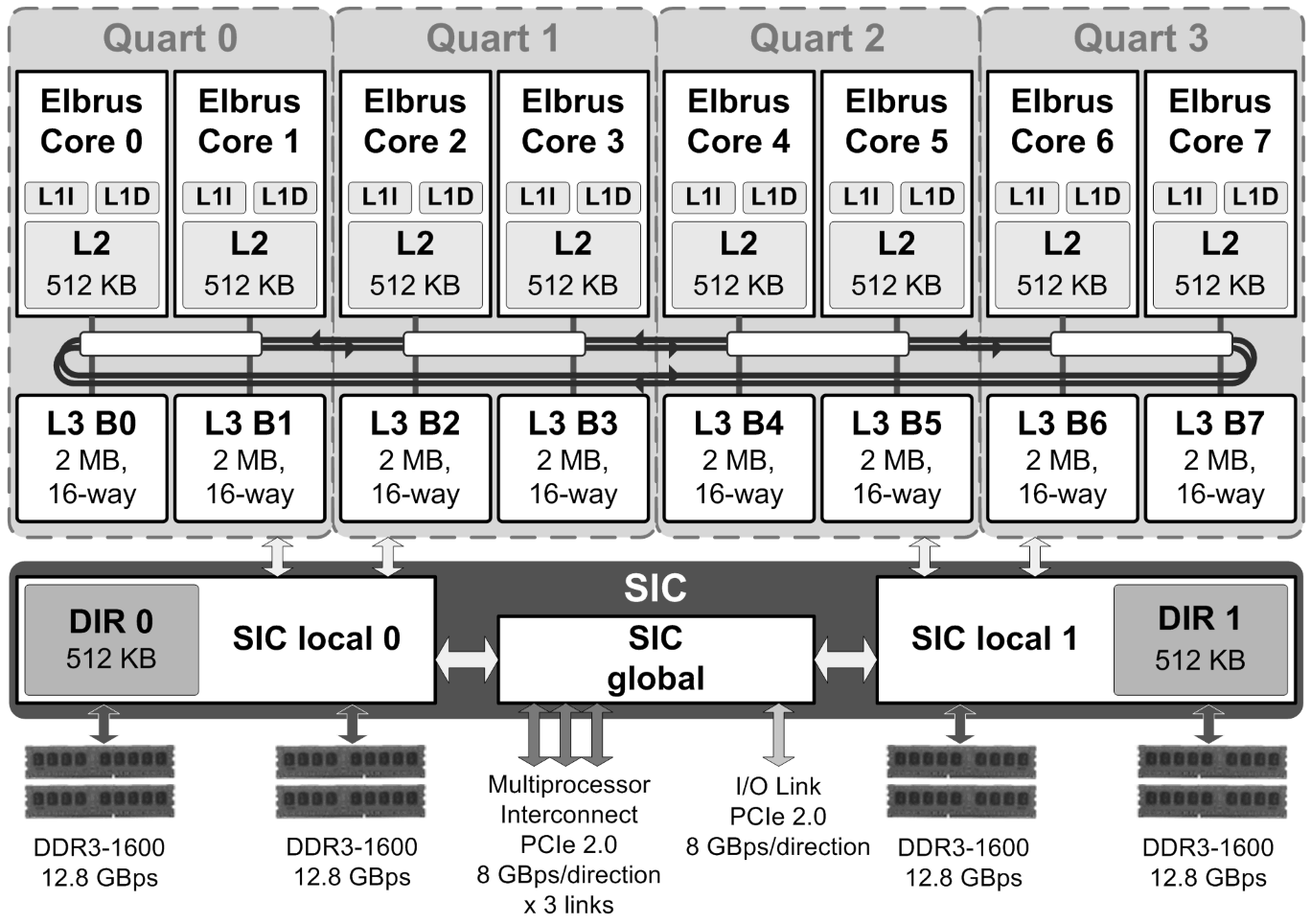


Рисунок 2. Структурная схема микропроцессора «Эльбрус-8C»

Контроллер системных обменов SIC (System Interface Controller) выполняет функции доступа к памяти (SIC local 0,1) и аппаратуре ввода-вывода (SIC global), он включает в свой состав все контроллеры высокоскоростных интерфейсов процессора:

* четыре контроллера каналов оперативной памяти DDR3-1600 SDRAM с максимальной пропускной способностью 51,2 ГБ/с, которые обеспечивают установку до 64 ГБ памяти на процессор;
* три контроллера дуплексных каналов межпроцессорного обмена с пропускной способностью 8 ГБ/с в каждую сторону, позволяющие объединять до четырех микропроцессоров в одну систему с когерентной общей памятью (возможна установка до

256 ГБ памяти на систему);

 контроллер дуплексного канала ввода-вывода с пропускной способностью 8 ГБ/с в каждую сторону, совместимый с южным мостом КПИ-2 собственной разработки.

Также в состав процессора входят два справочника для фильтрации запросов и снижения когерентного трафика: локальный справочник для поддержки межъядерной когерентности (включен в кэш-память третьего уровня) и глобальный справочник DIR 0,1 для поддержки межпроцессорной когерентности.

Расчетная тактовая частота разработанного микропроцессора достигает 1,3 ГГц, при этом пиковая производительность составляет 250 GFLOPS на операциях с одинарной точностью (FP32) и 125 GFLOPS на операциях с двойной точностью (FP64).

# Процессорное ядро

При разработке ядра «Эльбрус-8C» за основу было взято ядро микропроцессора предыдущего поколения «Эльбрус-4C». Для повышения производительности ядра было решено увеличить тактовую частоту и число одновременно исполняемых арифметических операций с плавающей точкой.

На рисунке 3 приведена структурная схема процессорного ядра «Эльбрус-8C» и его отличие от ядра микропроцессора предыдущего поколения. Ядро микропроцессора «Эльбрус-4C» имеет двухкластерную организацию. Каждый кластер содержит три арифметико-логических канала (ALC), локальный блок регистрового файла (RF), локальный блок кэша данных первого уровня (L1D) и блок подготовки операндов и сбора результатов (bypass). Каналы ALC имеют отдельные устройства для выполнения целочисленных операций и операций с плавающей точкой. Целочисленные операции выполняются во всех шести каналах двух кластеров. Скалярные обращения в память по считыванию выполняют ALC 0, 2, 3 и 5, по записи – ALC 2 и 5. Операции с плавающей точкой типа сложения и умножения, в том числе трехоперандные, выполняют ALC 0, 1, 3 и 4, операцию деления – только ALC 5.

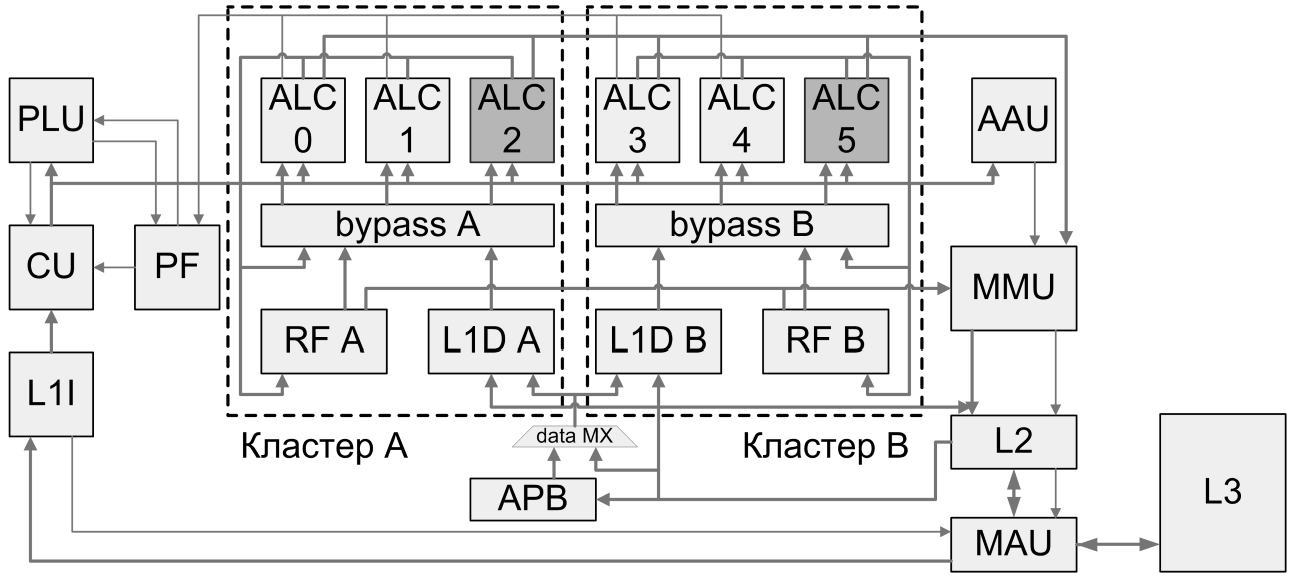


Рисунок 3. Структурная схема процессорного ядра «Эльбрус-8C» и иерархии кэш-памяти Увеличение числа одновременно исполняемых арифметических операций с плавающей точкой было достигнуто за счет добавления в ALC 2 и 5 возможности исполнения операций сложения и умножения с плавающей точкой. При этом потребовалось модифицировать структуру широкой команды, а именно, организовать упаковку в нее новых слогов для кодирования новых команд ALC 2 и ALC 5 при сохранении обратной совместимости.

В результате применения обоих методов удалось повысить производительность ядра почти в 2,5 раза, при этом увеличение его площади составило всего 3,2%.

# Иерархия кэш-памяти

Кэш-память существенно влияет на производительность микропроцессора, позволяя в максимально возможной степени сократить задержку доступа к инструкциям и данным программы. Наращивание количества процессорных ядер только усилило роль кэшпамяти, возложив на нее дополнительную функцию обеспечения межъядерного взаимодействия.

Частная кэш-память ядра была унаследована от процессора «Эльбрус-4C» с некоторыми изменениями (рисунок 2). Как и в предыдущем поколении, используется раздельная кэш-память первого уровня для команд (L1I кэш, 128 КБ) и для данных (L1D кэш, 64 КБ). Кэш команд состоит из кэш-строк размером 256 байт и имеет ассоциативность 4. Размер кэш-строки виртуально индексируемого кэша данных – 32 байта, ассоциативность 4. Кэш-память второго уровня (L2 кэш) хранит и код, и данные. Из-за технологических ограничений на площадь кристалла и новых особенностей иерархии размер L2 кэша был уменьшен до 512 КБ, а в остальном он не изменился по сравнению с процессором предыдущего поколения. Размер кэш-строки L2 кэша – 64 байта, ассоциативность – 4, он является неинклюзивным относительно кэш-памяти первого уровня. Устройство MAU (Memory Access Unit), подключаемое к выходу L2 кэша и управляющее обменами ядра с остальной системой, было разработано заново с целью повышения рабочей частоты и масштабируемости, а также оптимизации взаимодействия ядра с L3 кэшем.

Главной особенностью иерархии кэш-памяти нового микропроцессора стало появление общей кэш-памяти третьего уровня (L3 кэш) объемом 16 МБ [2]. Она имеет распределенную структуру и адресное разбиение (по 8, 7, 6 битам адреса) на восемь независимых банков по 2 МБ, причем любое ядро имеет доступ ко всему объему каждого банка. Размер кэш-строки данных – 64 байта, ассоциативность каждого банка – 16. Разработанная схема с раздельными конвейерами тэгов и данных позволила значительно снизить динамическую мощность L3 кэша при такой высокой ассоциативности. Выбор строки-жертвы для вытеснения производится по алгоритму NRU (Not Recently Used). Память тэгов реализована на статической двухпортовой памяти с произвольным доступом, имеющей независимые порты чтения и записи. Память данных каждого банка L3 кэша разделена на два однопортовых массива шириной 32 байта, соответствующих старшей и младшей половине кэш-строки. В каждом такте можно выполнять чтение или запись только одного массива данных или одновременное чтение одного массива и запись другого. Таким образом, максимальная пропускная способность L3 кэша составляет 333 ГБ/с как по чтению, так и по записи.

В таблице приведено минимальное время доступа в память в зависимости от расположения запрашиваемых данных для разработанного микропроцессора «Эльбрус8C» и его сравнение с микропроцессором предыдущего поколения «Эльбрус-4C» и зарубежными аналогами AMD Bulldozer (Opteron 6274) и Intel Sandy Bridge-EP (Xeon E5-

2670), изготовленными на технологическом процессе того же уровня.

\*В кэш-памяти L1 процессоров семейства «Эльбрус» хранятся только немодифицированные строки, которые не могут быть считаны по снуп-запросу.

# Подсистемы памяти и ввода-вывода

Подсистема памяти обеспечивает когерентный доступ процессорных ядер и внешних устройств к оперативной памяти. Как и кэш-память, она играет важную роль в достижении требуемой производительности, особенно для программ Memory-Bound. По сравнению с микропроцессором предыдущего поколения в «Эльбрус-8C» удвоилось число процессорных ядер и, соответственно, возросла нагрузка на память. Чтобы подсистема памяти не стала узким местом микропроцессора, количество каналов памяти DDR3-1600 было увеличено до четырех, что обеспечивает максимальную пропускную способность 51,2 ГБ/с и установку до 64 ГБ памяти на процессор. В состав микропроцессора входят три спроектированных на базе физуровня PCI Express контроллера дуплексных каналов межпроцессорного обмена с пропускной способностью 8 ГБ/с в каждую сторону. Посредством этих каналов возможно объединение до четырех процессоров в ccNUMA систему с общей когерентной памятью (рисунок 4а).

Функции доступа к памяти выполняет локальный контроллер системных обменов SIC local. Кроме того он является глобальной точкой сериализации запросов в многопроцессорной системе с общей памятью. В микропроцессоре «Эльбрус-4С» точка сериализации была способна обрабатывать не более одного запроса за процессорный такт.

В новом микропроцессоре ее пропускная способность увеличена вдвое за счет адресного разделения на два независимых кластера SIC local 0,1 (рисунок 4б). Каждый кластер включает в себя два контроллера оперативной памяти MC, системный коммутатор SC, реализующий сериализацию с темпом один запрос за такт, глобальный справочник DIR и различные коммутаторы (не показаны на рисунке). Банки кэш-памяти третьего уровня имеют фиксированную привязку к кластерам: каждая пара контроллеров памяти обслуживает ближайшие четыре банка L3 кэша. Контроллеры каналов межпроцессорного обмена (IPCC 0,1,2) размещены в блоке SIC global и через коммутатор пакетов подключены к обоим кластерам SIC local.

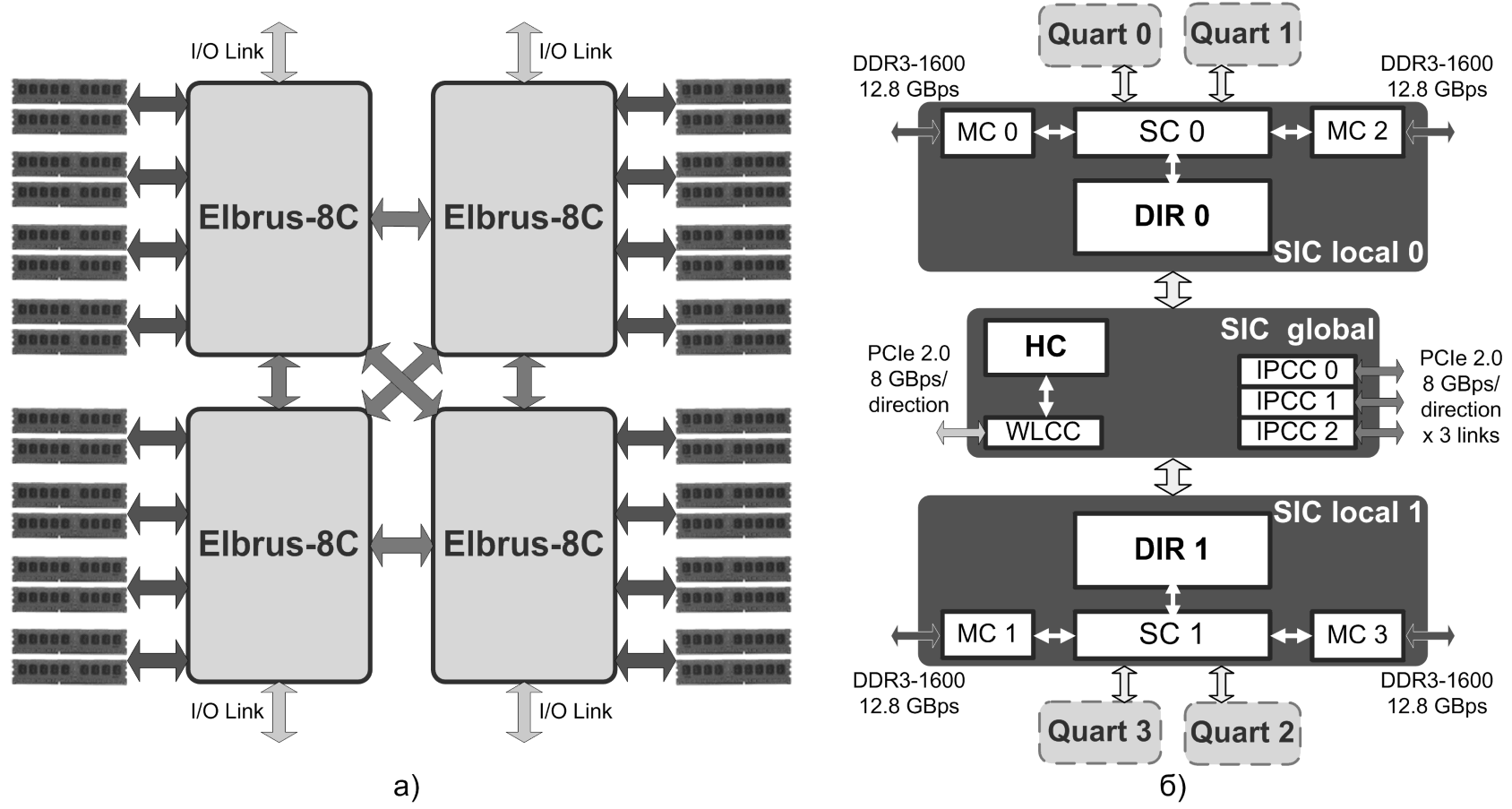


Рисунок 4. Подсистема памяти: а) ccNUMA система с общей когерентной памятью из четырех процессоров «Эльбрус-8C»; б) Структурная схема контроллера системных обменов SIC

Доступ ядер к внешним устройствам и конфигурационным регистрам системы, а также доступ внешних устройств к оперативной памяти системы посредством DMA обеспечивается через контроллер SIC global. Для связи процессора с КПИ-2 (контроллер периферийных интерфейсов второй версии) используется физический уровень PCI Express, пропускная способность которого выросла в четыре раза по сравнению с микропроцессором предыдущего поколения «Эльбрус-4C». В состав подсистемы вводавывода входят контроллер канала ввода-вывода (WLCC), блок конфигурационных регистров подсистемы памяти, контроллер прерываний и новый Хост-контроллер (HC), разработанный для микропроцессора «Эльбрус-8C». Хост-контроллер обеспечивает доступ ядер к конфигурационным регистрам и внешним устройствам, пересылку прерываний от внешних устройств и других процессоров в контроллер прерываний и доступ внешних устройств в оперативную память. Он также поддерживает упорядоченное (согласно стандарту PCI 2.0) исполнение DMA-обращений от внешних устройств в оперативную память без потери скорости обмена [4].

# Коммутация соединений

Схема коммутации микропроцессора разделена на два уровня (рисунок 5): ядра – L3 кэш и L3 кэш – SIC. На первом уровне реализована связь каждого процессорного ядра с семью остальными и с восемью банками общей кэш-памяти третьего уровня.

Разработанная схема имеет распределенную структуру и состоит из четырех узлов Quart (кварт) [5]. В кварту входят два ядра и два банка общего кэша, подключенные к централизованному коммутатору L3 Com. Кварты объединены в двунаправленное буферизующее кольцо с передачей пакетов по часовой стрелке (ClockWise Ring) и против часовой стрелки (Counter-Clockwise Ring).

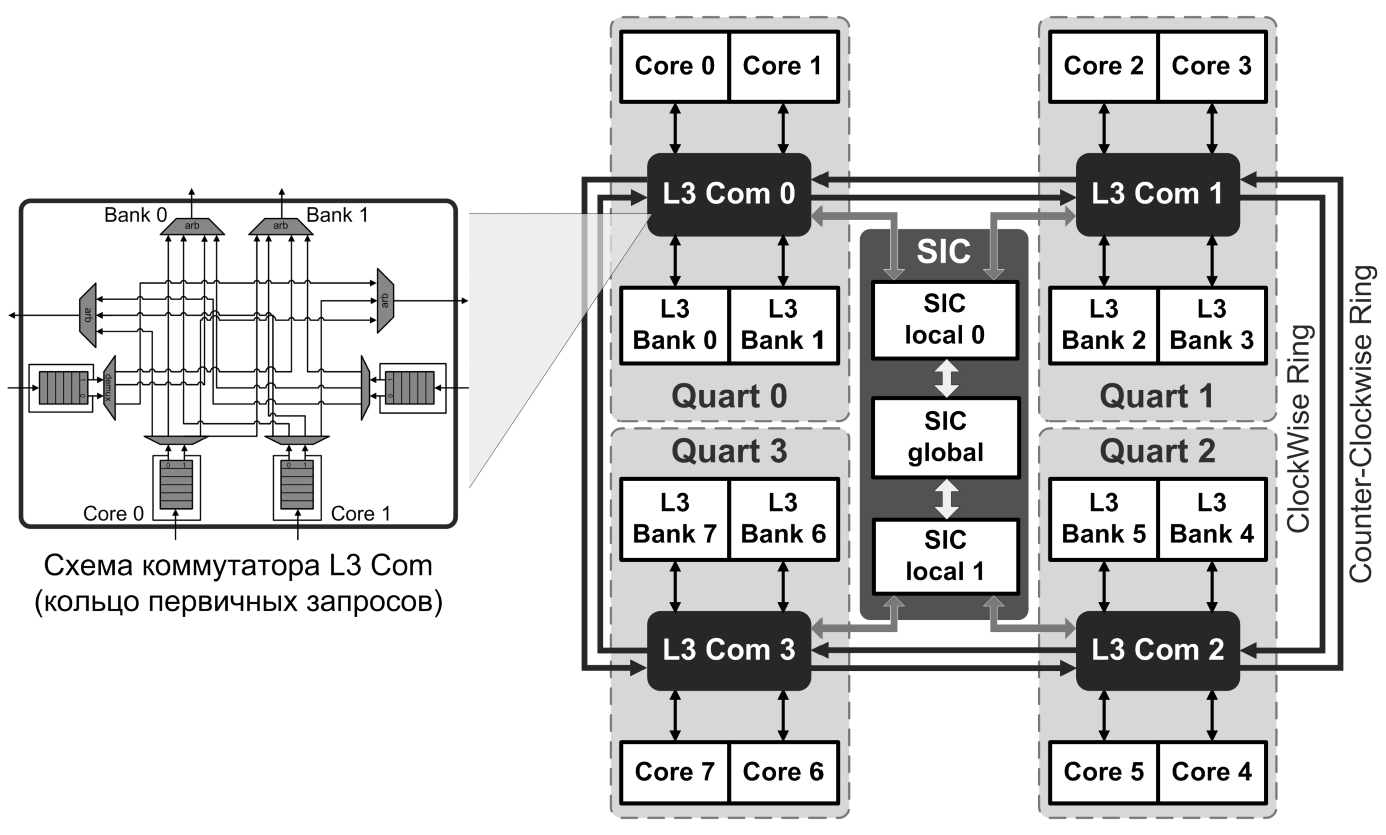


Рисунок 5. Схема коммутации соединений микропроцессора «Эльбрус-8C»

Пять независимых двунаправленных колец передачи информации, соответствующих первичным запросам, снуп-запросам, снуп-ответам, пакетам данных, подтверждениям завершений чтений, используют различные группы сигналов двунаправленной кольцевой шины. Пропускная способность кольца данных составляет 32 байта/такт в каждом направлении. Пакеты данных могут иметь формат 32 байта (половина кэш-строки) и 64 байта (целая кэш-строка), причем во втором случае пакет передается двумя неразрывными посылками по 32 байта. В пакете, независимо от его типа, указывается идентификатор абонента кольца (адрес назначения), которому он посылается. В коммутаторе L3 Com в результате анализа адреса назначения происходит дальнейшая маршрутизация пакета. Передача пакетов между абонентами кварты может выполняться независимо от их транзитного прохождения по кольцу. Для уменьшения числа конфликтов в кольце и времени доступа в L3 кэш разработано несколько механизмов оптимизации:

* Out-of-Order – возможность обгона для пакетов с разными адресами назначения;
* широковещательные пакеты – передача одинаковых снуп-запросов и пакетов с данными нескольким адресатам за одну посылку;
* упреждающая выдача заголовка данных (совместно с байпассированием данных в ядре уменьшает время попадания в L3 кэш на 20%).

Пиковая пропускная способность по чтению из общей кэш-памяти, которую может обеспечить разработанное кольцо при отсутствии конфликтов пакетов или обращении ядер в ближайшие банки, составляет 333 ГБ/с.

На втором уровне реализована связь банков L3 кэша с SIC. Каждая кварта подключена к соответствующей паре контроллеров памяти, входящих в состав одного из устройств SIC local. Коммутация происходит в два этапа: между двумя банками в кварте и между двумя квартами в SIC local. При доступе к подсистеме ввода-вывода и каналам межпроцессорного обмена коммутация между всеми четырьмя квартами осуществляется в устройстве SIC global. Суммарная пиковая пропускная способность связей между четырьмя квартами и контроллером SIC составляет по 166 ГБ/с в каждом направлении.

# Когерентность

Коммутационные соединения, описанные выше, обеспечивают достаточно высокую пропускную способность для обменов как внутри чипа, так и между чипами. Тем не менее, при использовании протокола когерентности на основе широковещательного снупирования они оказались бы перегружены когерентным трафиком в максимальной 32ядерной ccNUMA-системе. Для уменьшения трафика и времени доступа в память был спроектирован иерархический MOSI-протокол с двумя точками сериализации запросов и использованием когерентных справочников.

В качестве точки сериализации запросов от процессорных ядер выступает общая кэш-память третьего уровня, на базе которой реализован локальный справочник с MOSIпротоколом когерентности, существенно снижающий межъядерный трафик когерентных сообщений. Принцип работы справочника основан на свойстве инклюзивности L3 кэша относительно L1 и L2 кэшей каждого ядра своего процессора. Строки, находящиеся в L1 и L2 кэшах всех ядер, обязательно присутствуют в кэш-памяти третьего уровня, возможно без актуальных данных; вытеснение строки из L3 кэша вызывает вытеснение этой строки из всех кэшей верхних уровней. Каждая кэш-строка L3 кэша дополнена информацией о ее состоянии в кэш-памяти всех процессорных ядер – вектором возможных ядерсовладельцев и указателем на владельца модифицированной строки. Эта информация вместе с памятью тэгов и состояний L3 кэша является локальным справочником. Рассылка снуп-запросов выполняется только ядрам, указанным в справочнике, а не широковещательно. Пропускная способность локального справочника составляет 8 запросов/такт. При этом адресные блокировки происходят только для запросов по записи, последовательные запросы по чтению от разных ядер для ускорения работы с общими данными могут выполняться одновременно. Доступ к контроллеру системных обменов SIC происходит только при отсутствии данных в L3 кэше и предыдущих невыполненных кэшируемых обращений к той же кэш-строке. Такая схема позволяет значительно уменьшить нагрузку на подсистему памяти, имеющую меньшую пропускную способность.

На глобальном уровне запросы от всех процессоров сериализуются в SIC Homeпроцессора. Как и в процессоре предыдущего поколения, в микропроцессоре Эльбрус-8С используется глобальный справочник DIR 0,1 (рисунок 2) для уменьшения когерентного трафика и времени обработки запросов в память, но, что принципиально, не полный по памяти, а частичный [6]. Полные справочники обычно располагаются в выделенных областях оперативной памяти и содержат информацию о когерентном состоянии каждой строки памяти, тогда как частичные справочники хранят лишь часть информации в специально отведённой для этих целей кэш-памяти. Основными доводами против варианта полного справочника послужили достаточно сильно возрастающая в этом случае сложность адресной арифметики, снижение производительности и избыточность, т.к. размер полного справочника пропорционален объему всей оперативной памяти, а необходимо покрыть только суммарный объем кэш-памяти четырех процессоров.

Для увеличения эффективности покрытия глобального справочника было решено:

* не размещать в справочнике информацию о собственном процессоре, хранить информацию обо всех кэш-строках, принадлежащих памяти данного процессора и находящихся в данный момент в других процессорах;
* для каждого адресного тэга хранить информацию о двух соседних кэш-строках.

Глобальный справочник выполнен на двухпортовой SRAM-памяти, имеющей раздельные порты чтения и записи. В процессоре «Эльбрус-8С» находятся два экземпляра справочника: по одному на каждый SIC local. Объём каждого из них составляет 512 Кбайт (2^17 блоков), организованных в виде кэш-памяти с ассоциативностью 16. В глобальном справочнике реализован MOSI-протокол когерентности, справочник каждого процессора покрывает 32 Мбайт памяти, т.е. может полностью покрыть L3 кэши двух процессоров.

# Рассеиваемая мощность, отказоустойчивость и технологические дефекты

В связи с увеличением площади кристалла и переходом на новый технологический процесс 28 нм потребовалось решить ряд проблем, не связанных напрямую с достижением высокой производительности. В первую очередь стоит отметить проблему рассеиваемой мощности. Требование по энергосбережению учитывалось на всех этапах

проектирования микропроцессора «Эльбрус-8C», начиная с принятия микроархитектурных решений и заканчивая использованием малопотребляющих элементов и заказных блоков памяти собственной разработки. Также для снижения рассеиваемой мощности при простое одного или нескольких процессорных ядер разработан механизм их динамического отключения под управлением операционной системы. Измерение температуры кристалла осуществляется посредством восьми термодатчиков, размещенных в разных точках кристалла.

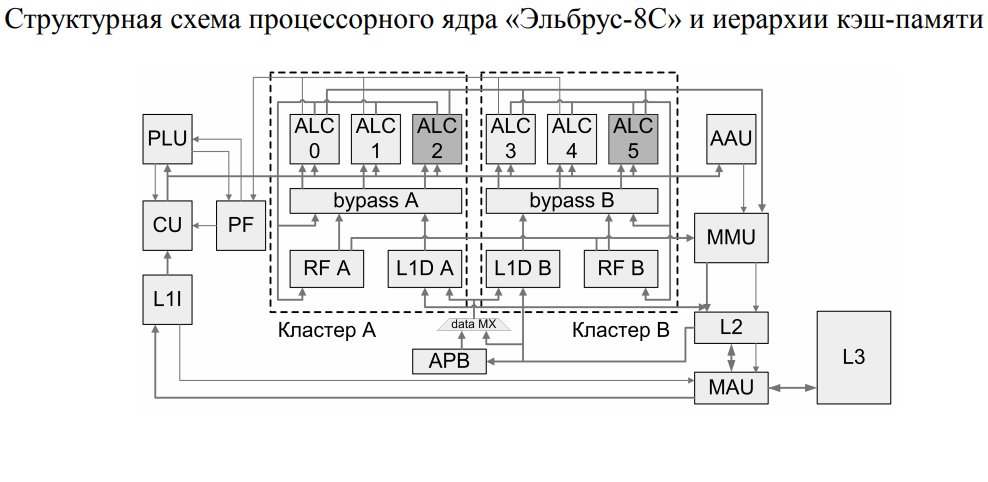
Для защиты от единичных сбоев в кэш-памяти ядер реализованы проверка четности в небольших блоках памяти и корректирующие коды Хэмминга SECDED (Single Error

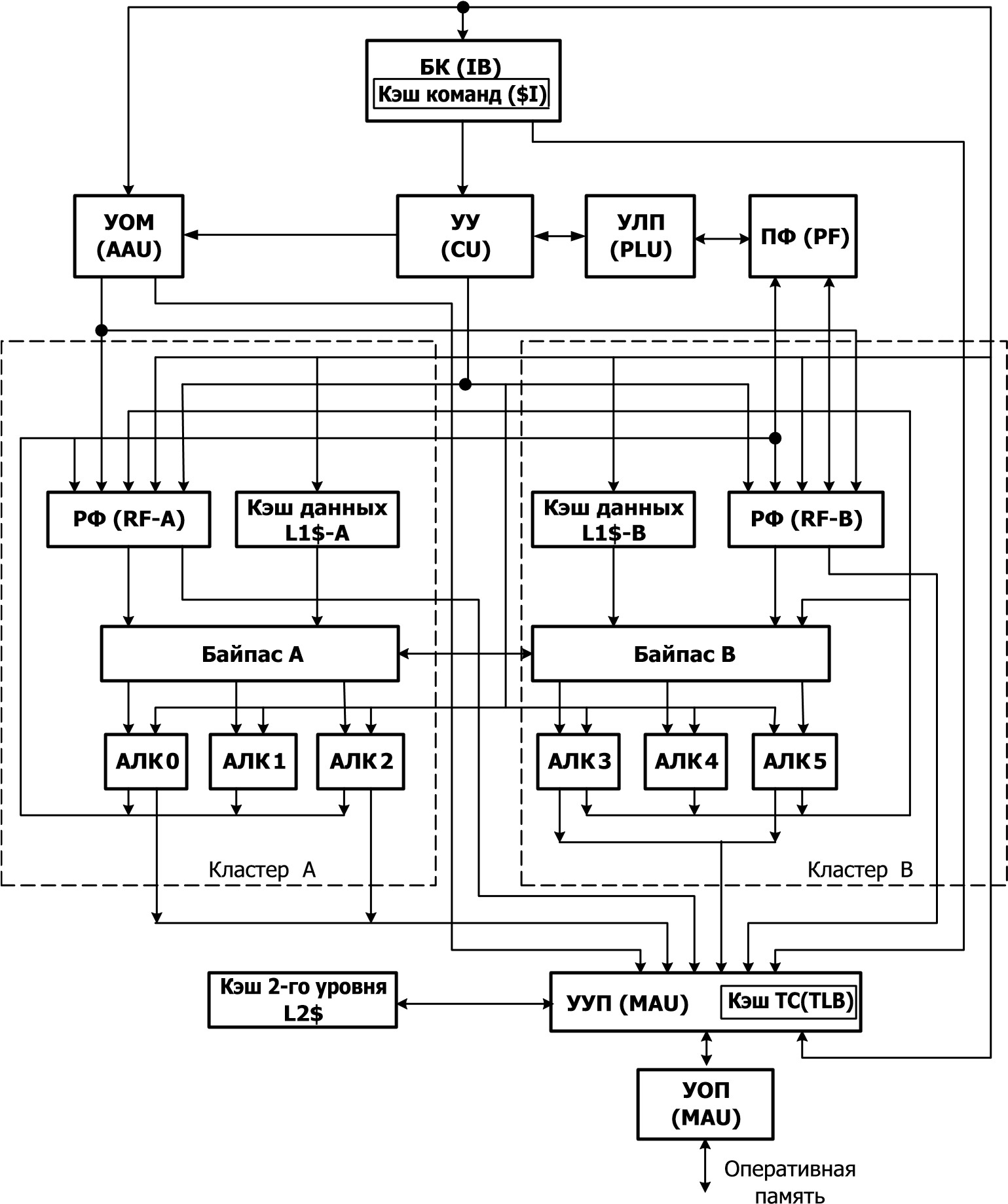
Correction, Double Error Detection) в крупных блоках данных, кроме того код Хэмминга SECDED защищает общую кэш-память третьего уровня, глобальный справочник и оперативную память.

Еще одной проблемой стало полное покрытие технологических дефектов в SRAMпамяти и увеличение выхода годных чипов. Разработанный новый алгоритм системы автоматического тестирования (BIST, Built-In Self-Test) [7] обеспечивает достаточные условия для нахождения всех статистически важных типов дефектов, свойственных производству процессоров по технологическим нормам 28 нм. Неисправные элементы памяти, обнаруженные BIST, заменяются резервными элементами с помощью системы автоматической компенсации неисправностей (BISR, Built-In Self-Repair). При невозможности замены дефектный блок помечается специальной меткой как запрещённый к использованию и не влияет на корректность работы процессора, но при этом несколько снижается производительность. Реализованные методики резервирования и маскирования позволяют использовать микропроцессоры с дефектами в SRAM-памяти и таким образом увеличивают выход годных процессоров.

# Заключение

Цель работы по проектированию универсального микропроцессора с производительностью не ниже 150 GFLOPS на операциях с одинарной точностью была достигнута за счет создания сбалансированного дизайна микропроцессора и решения целого ряда задач на архитектурном и микроархитектурном уровнях, а также преодоления проблем, характерных для нового технологического процесса 28 нм. Пиковая производительность разработанного микропроцессора «Эльбрус-8C» составляет 250 GFLOPS на операциях с одинарной точностью при расчетной тактовой частоте 1,3 ГГц. Первые инженерные образцы были получены в ноябре 2014 г., а начало серийного производства микропроцессора намечено на 2016 г.





**БК — буфер команд** (IB — Instruction Buffer) предназначен для вызова программного кода из памяти, буферизации на время, достаточное для обеспечения непрерывности дешифрации, и выдачи его в устройство управления для последующей обработки. Накопление программного (объектного) кода обеспечивает кэш команд I$ емкостью 64 Кбайт.

**УУ —** **устройство управления** (CU — Control Unit) выполняет считывание программного кода из буферной памяти БК, распаковку широких команд, их дешифрацию и переключение ветвей программы при выполнении команд переходов. С этой целью в структуре УУ предусмотрены обработка последовательности ШК основной ветви, предварительная подкачка ШК трех ветвей предполагаемого ветвления и распаковка первой из команд каждой ветви. Заметим, что подготовка перехода выполняется на фоне выполнения основной ветви и поэтому не приводит к замедлению вычислительного процесса. С выхода УУ в исполнительные устройства выдается распакованная ШК основного потока или одного из подготовленных потоков (при наличии в основном потоке операции передачи управления с реализовавшимся условием перехода).

**ПФ —** **предикатный файл** (PF — Predicate File) хранит первичные предикаты — битовые значения, выработанные операциями сравнения, и вторичные предикаты — результаты логических операций над первичными предикатами.

Предикатный файл — набор 32 двухразрядных регистров (по одному разряду для предиката и тега). С помощью предикатных значений может быть задан режим условного выполнения операции или условий для команд ветвлений.

**УЛП —** **устройство логических предикатов** (PLU — Predicate Logic Unit) предназначено для выполнения операций считывания предикатов из файла ПФ и логических операций формирования вторичных предикатов, которые также могут быть записаны в предикатный файл. Эти операции выполняются над малоформатными значениями, поэтому в целях разгрузки арифметико-логических каналов обработка предикатов возложена на УЛП. Ее результаты направляются в устройство управления для задания режима условного выполнения команд, отрабатываемых в арифметико-логических каналах, или выполнения команд передачи управления.

**Арифметико-логические каналы (АЛК)** предназначены для исполнения обычных арифметических и логических операций, операций обращения к памяти и обработки адресных данных (дескрипторов, указателей и др.). В состав микропроцессора входят шесть арифметико-логических каналов (АЛК0—АЛК5), разделенных на два кластера. Арифметико-логические каналы работают параллельно и исполняют в основном одинаковый набор операций. В качестве операндов служат данные из РгФ или результаты других исполнительных устройств. Не все каналы идентичны, поскольку не все операции выполняются одинаково часто. Например, довольно редкая операция деления реализована только в АЛК5. Арифметико-логические каналы имеют отдельные устройства для исполнения целочисленных и вещественных операций. Это не относится к целочисленным операциям умножения и деления — для них введены соответствующие блоки в вещественных исполнительных устройствах.

**Регистровый файл (РгФ)** предназначен для хранения локальных данных процедуры и результатов выполненных операций. Он представляет собой сверхоперативное запоминающее устройство с произвольным доступом, обращение которому осуществляется через порты. Многочисленность абонентов регистрового файла требует большого количества портов для обслуживания всех запросов одновременно. С целью сокращения их количества регистровый файл реализован в виде двух одинаковых блоков — РгФ-A и РгФ-B, по одному в каждом кластере. Причем, в отличие от используемых во многих микропроцессорах, блок является общим для целочисленных и вещественных устройств арифметики, что позволило повысить эффективность его использования. Блоки регистрового файла содержат одни и те же данные, поскольку результат любой операции записывается одновременно в оба блока. Этим обеспечивается когерентность данных — свойство, которое используется при работе с общими данными.

Блок РгФ содержит 256 регистров. При запуске процедуры ей выделяется определенный участок смежных регистров, называемый окном. Активная (выполняемая в данный момент) процедура может обращаться только к регистрам своего окна. Если все регистры окажутся занятыми, автоматически выполняется откачка. В результате откачки данные пересылаются в стек процедур, размещенный в оперативной памяти. Таким образом, регистровый файл служит в качестве аппаратной вершины стека процедуры. Все ранее работавшие, но не завершенные процедуры сохраняют свои окна в регистровом файле. Возврат процедур к своим окнам осуществляется по мере окончания работы запущенных ими процедур. Иными словами, реализуется дисциплина (LIFO) — активизируется та процедура, которая запустила завершившуюся в данный момент времени процедуру.

Чтобы результат, полученный некоторым исполнительным устройством, мог быть использован другим до записи в РгФ, исполнительные устройства связаны шинами байпаса. Благодаря этому в ряде случаев отпадает необходимость в считывании операндов из РгФ. Каждый кластер имеет отдельный блок байпаса, с помощью которого результаты передаются исполнительным устройствам не только своего, но и другого кластера. Кроме того, байпас позволяет использовать результаты считывания из кэш-памяти L1$ до их записи в регистровый файл.

**Кэш данных первого уровня L1$** выполнен в виде двух одинаковых блоков (L1$-A и L1$-B) емкостью 64 Кбайт, по одному в каждом кластере. Блоки L1$ хранят одинаковые данные, поскольку запись данных выполняется одновременно в оба блока. В блоке хранятся данные, которые используются в качестве операндов для исполнительных устройств АЛК. Но поскольку в общем случае операнды считываются из регистрового файла, они должны быть предварительно загружены в него из кэша L1$. Для этих целей предусмотрены операции загрузки, выполняемые исполнительными устройствами АЛК. В случае отсутствия требуемых данных в кэше L1$ операция загрузки продолжается поиском данных в кэш-памяти второго уровня L2$ с последующей записью их в оба блока L1$. Параллельная запись, так же как и в случае регистрового файла, вызвана необходимостью сохранения свойства когерентности данных обоих блоков L1$.

**Кэш второго уровня L2$** является общим для данных и программного кода, его объем составляет 256 Кбайт, степень ассоциативности — 4. Обращение к L2$ выполняется при отсутствии требуемых данных в L1$ или нужного программного кода в буферной памяти команд устройства БК. Если нужная информация отсутствует и в L2$, то формируется запрос к оперативной памяти. Считанная из ОП информация поступает потребителю и одновременно записывается в кэш-память вместо устаревших данных.

**УОМ — Устройство обращения к массивам** (AAU — Array Access Unit) предназначено для упреждающей подкачки элементов массива при выполнении векторных операций. Поскольку каждый элемент вектора обрабатывается одной и той же последовательностью операций, то обработка идет циклически. В микропроцессоре «Эльбрус» к началу очередного цикла нужный элемент вектора уже считан из памяти (кэша L2$ или ОП) и находится в буфере УОМ. Подкачка элементов массивов в буфер, использующий дисциплину очереди (FIFO), осуществляется на фоне выполнения основной (синхронной) программы параллельной ей (асинхронной) программой.

**УУП — устройство управления памятью** (MMU — Memory Management Unit) преобразует виртуальные адреса в физические. С целью ускорения этого процесса наиболее часто используемые строки таблицы страниц хранятся в кэше таблицы страниц (TLB) объемом 64 строки. Если нужная строка в нем отсутствует, выполняется аппаратный поиск в таблице страниц, хранящейся в памяти, и загрузка найденной строки в кэш вместо устаревшей. Кроме того, рассматриваемое устройство обеспечивает обращения в кэш L2$ и ОП.

**УОП — устройство обращения в память** (MAU — Memory Access Unit) предназначено для связи микропроцессора с ОП. Оно содержит буферы операций считывания и записи, позволяющих осуществить потоковое обслуживание заявок. Обмен с памятью осуществляется через 16-байтовый канал с раздельными шинами для передачи и приема данных. Обмен выполняется блоками по 32 или 64 байт.

# 4. РАСЧЕТ ФУНКЦИЙ НАДЕЖНОСТИ И ГОТОВНОСТИ

Функция (или вероятность безотказной работы) относится к основным показателям надежности ЭВМ. Характеризует производительность ЭВМ на промежутке времени, то есть эта функция обеспечивает потенциально возможную производительность. Функцией надежности ЭВМ называется



где запись  означает вероятность того, что для всякого , принадлежащего промежутку времени  производительность  ЭВМ равна единице, т.е. равна потенциально возможной.

Функция  обладает следующими свойствами:

1.  Т.е. машина в момент начала функционирования находится в работоспособном состоянии.
2.  Событие, заключающееся в том, что ЭВМ работоспособна на конечном промежутке времени, является достоверным.
3.  для ;

Функцией ненадежности (или вероятностью отказа) ЭВМ называется



Функция  позволяет определить среднее время безотказной работы (средняя наработка до отказа). По определению, среднее время  безотказной работы ЭВМ и оценка  соответственно равны:



где – время безотказной работы -й машины, 

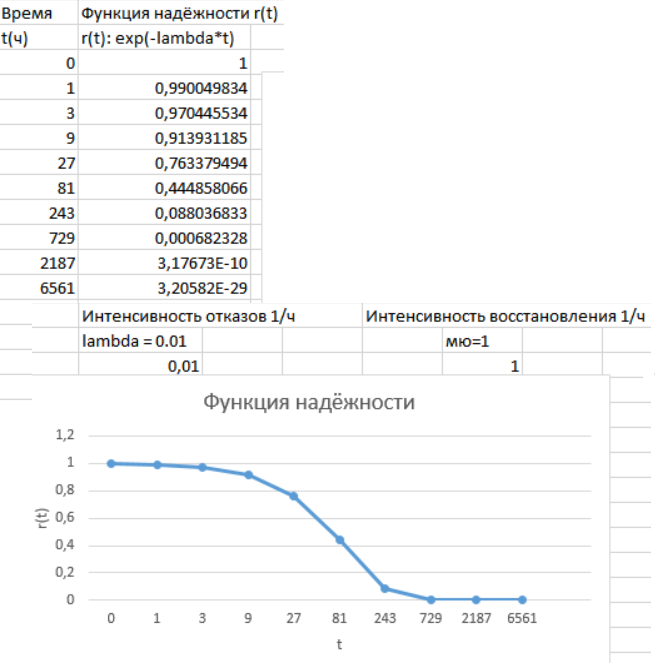
Интенсивностью отказов (лямбда-характеристикой) ЭВМ называется функция



Практически установлено, что зависимость интенсивности отказов от времени имеет место на периоде приработки ЭВМ. После приработки ЭВМ интенсивность отказов остается постоянной (до вхождения в предельное состояние или, по крайней мере, в течение промежутка времени, перекрывающего время морального старения). Следовательно, в нормальных условиях эксплуатации ЭВМ  а функция надежности и математическое ожидание времени *безотказной работы* соответственно равны:



рассчитаем значения функции и построим график:



Теперь рассчитаем значения функции готовности. Функция готовности ЭВМ

есть вероятность того, что (в условиях потока отказов и восстановлений) машина будет иметь в момент времени  производительность, равную единице, т.е. равную потенциально возможной.

Функция готовности ЭВМ обладает следующими свойствами:

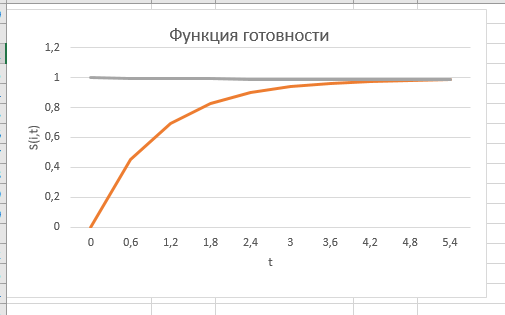
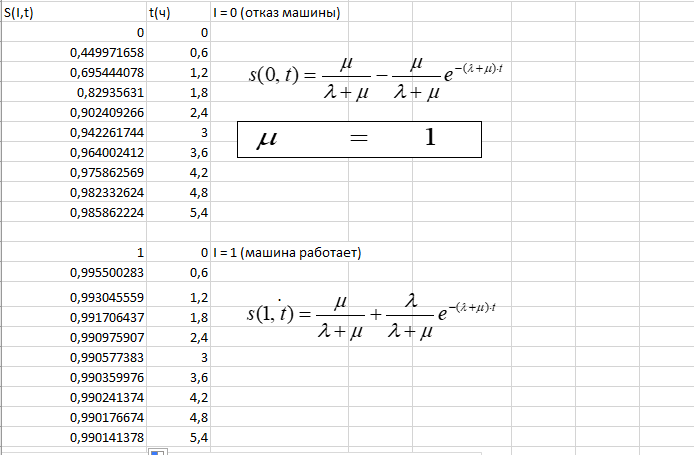
1. 
2. 
3.   для 

Расчет будем производить по следующим формулам:

;

.

для начальных состояний ЭВМ , причем  соответствует состоянию отказа, а  – работоспособному состоянию машины.

Рассчитаем значения функции и построим график: 

+

# 5. СПИСОК ИСПОЛЬЗУЕМОЙ ЛИТЕРАТУРЫ

1. Хорошевский В.Г. Архитектура вычислительных систем. – М.: МГТУ им. Н.Э. Баумана, 2008. – 520 с.
2. Конспект лекций по курсу “Архитектура вычислительных систем”
3. Ким А. К., Перекатов В. И., Ермаков С. Г. Микропроцессоры и вычислительные комплексы семейства «Эльбрус». — СПб.: Питер, 2013. — 272 с