Московский физико-технический институт (НИУ)
Физтех-школа радиотехники и компьютерных технологий

Базовая кафедра информатики и вычислительной техники

Продукция

- **универсальный микропроцессор «Эльбрус-8СВ»:**
  - 8 ядер, 1.5 ГГц, до 128 Гб DDR4-2400
- **микропроцессор для встраиваемых систем «Эльбрус-2С3»:**
  - 2 ядра, 2 ГГц, до 10 Вт, графический процессор с поддержкой 3D графики, аппаратное ускорение видеокодеков
- **серверный микропроцессор «Эльбрус-16С»:**
  - 16 ядер, 2 ГГц, до 1Тб DDR4-3200, объединение до 4 процессоров в NUMA-систему
- **компьютерные модули во встроенном и мобильном исполнении, жесткие условия эксплуатации**
- **дистрибутив общего программного обеспечения «Эльбрус»: оптимизирующий/двоичный компилятор, ядро ОС двух версий**

Эльбрус-8СВ

Эльбрус-801 PC

Эльбрус-2С3

Эльбрус-16С

Архитектура Эльбрус

Параллельная энергоэффективная архитектура

- VLIW-архитектура ядра (Very Long Instruction Word)
- высокая однопоточная производительность при малом энергопотреблении
- распараллеливание оптимизирующим компилятором
- сбалансированное соотношение производительности процессора и памяти

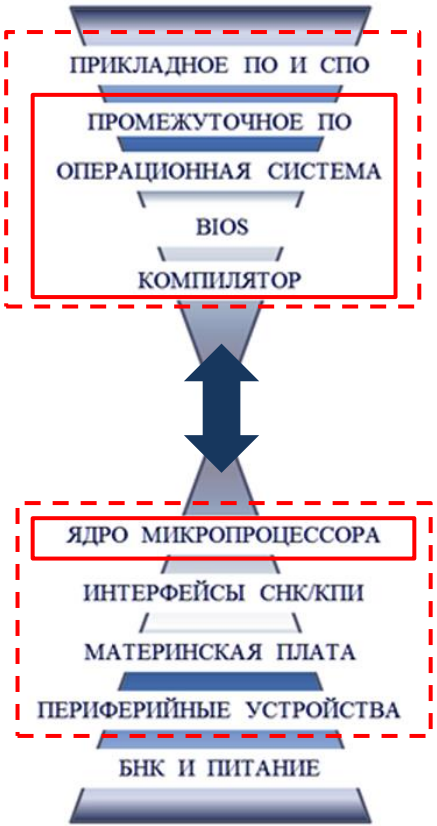
Эффективная двоичная совместимость с x86, x86-64

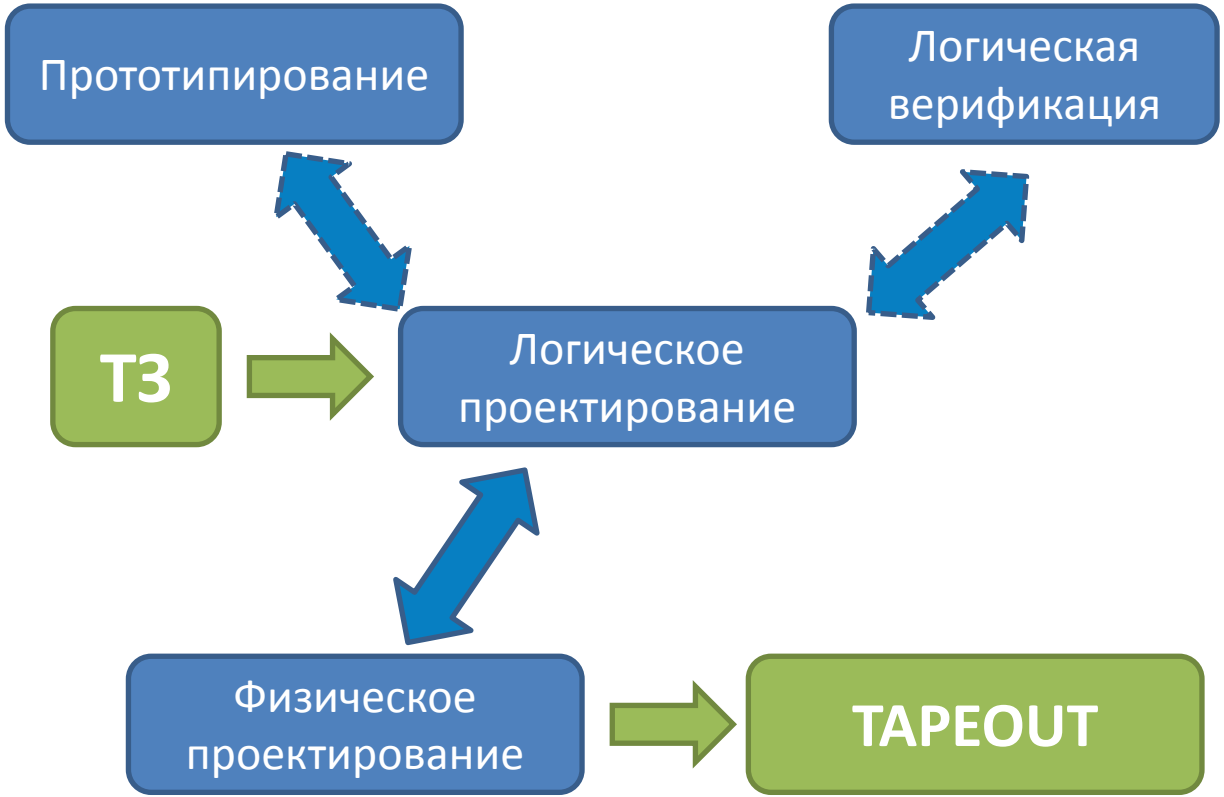
- исполнение приложений в кодах x86, x86-64 (Windows XP, 7, 10, Linux, QXP)
- базируется на поддержанной аппаратно технологии динамической двоичной компиляции
- производительность до 80% от нативной (в кодах Эльбруса)

Технология безопасных вычислений

- аппаратная защита логической структуры памяти
- гарантированное обнаружение атак, нарушающих структуру памяти
- повышение надежности программ
- повышение скорости отладки программ

Экосистема микропроцессора и маршрут проектирования



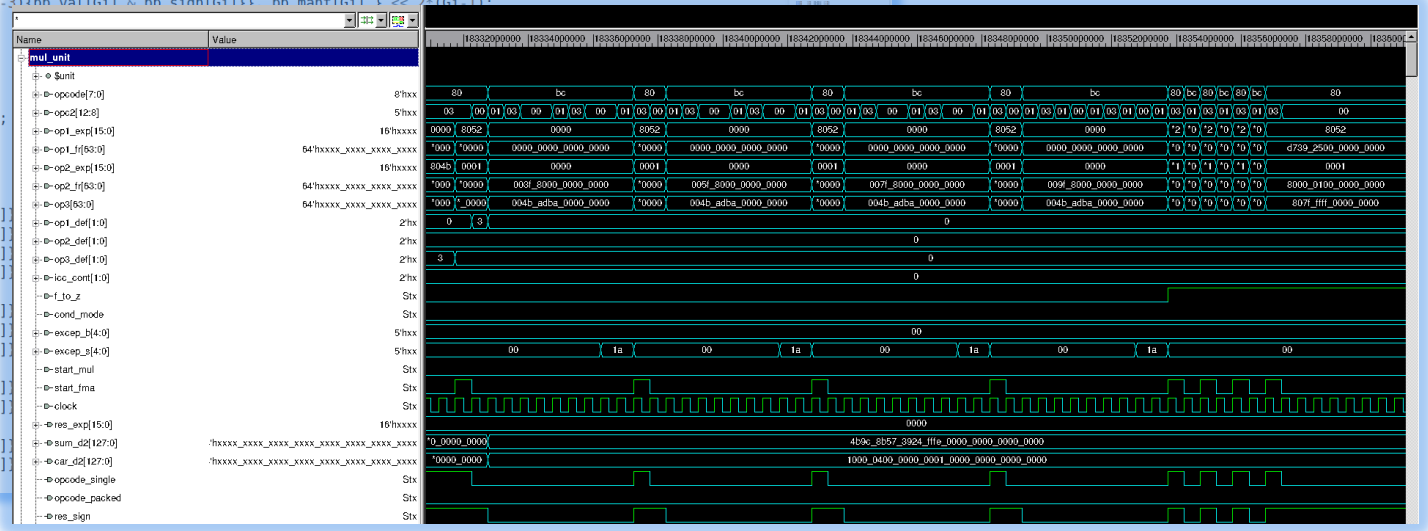


Проектирование и отладка Verilog

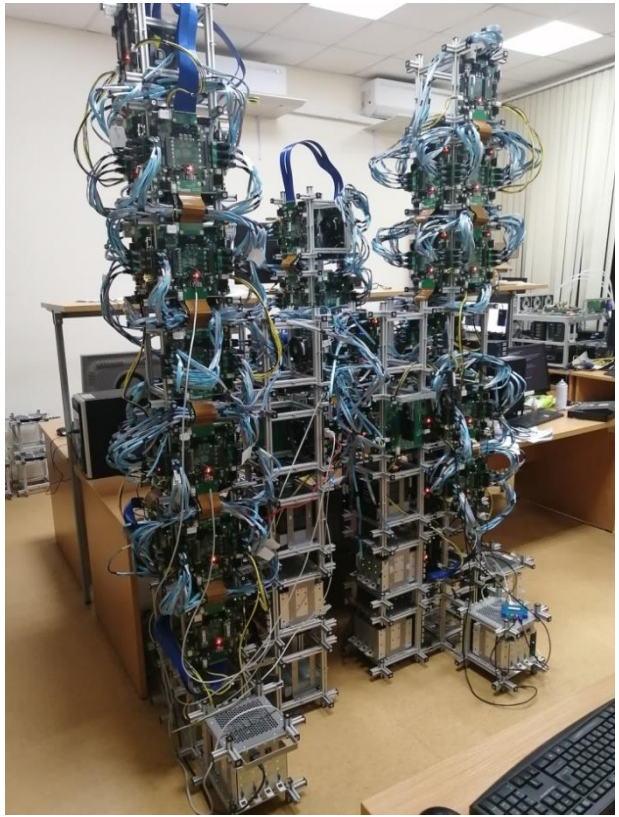
```
for(Gi=0; Gi<13; Gi=Gi+1)
begin: part_prod_gen
    assign  pp_val[Gi]   =~(full_addend[2+Gi*2:Gi*2]==3'b111) &~(full_addend[2+Gi*2:Gi*2]==3'b000);
    assign  pp_sign[Gi]  =  full_addend[2+Gi*2];
    assign  pp_shift[Gi] = (full_addend[2+Gi*2:Gi*2]==3'b100) | (full_addend[2+Gi*2:Gi*2]==3'b011) ;

    if(Gi==0)
    begin
        assign  pp_mant[Gi]     =   {{3{pp_sign[Gi]}},{24{pp_sign[Gi]}} ^ full_mant1_s1} << pp_shift[Gi];
        assign  pp_add_sign[Gi] =   ( pp_val[Gi] & pp_sign[Gi] )                         << pp_shift[Gi];
        assign  part_prod[Gi]   =   { {(48-24-3){pp_sign[Gi]}}, pp_mant[Gi] } & {48{pp_val[Gi]}};
    end
    else begin
        assign  pp_mant[Gi]     =   {(({24+2{pp_sign[Gi]}} ^ full_mant1_s1) << pp_shift[Gi] ) & {24+2{pp_val[Gi]}}, pp_add_sign[Gi-1]};
        assign  pp_add_sign[Gi] =   ( pp_val[Gi] & pp_sign[Gi] )            << pp_shift[Gi];
        assign  part_prod[Gi]   =   { {(48-24-3){pp_val[Gi] & pp_sign[Gi]}}, pp_mant[Gi] } << 2*(Gi-1);
    end
end

for (Gi=0 ; Gi<13; Gi=Gi+1)
begin: csa_in_assign1
    assign  csa_in_1[Gi]    =   part_prod[Gi];
end

for(Gi=0; Gi<48; Gi=Gi+1)
begin: csa_out_form
    assign  {csa_out_1[1][Gi],csa_out_1[0][Gi]} =   csa_in_1[2][Gi] + csa_in_1[1][Gi] + csa_in_1[0][Gi];
    assign  {csa_out_1[3][Gi],csa_out_1[2][Gi]} =   csa_in_1[5][Gi] + csa_in_1[4][Gi] + csa_in_1[3][Gi];
    assign  {csa_out_1[5][Gi],csa_out_1[4][Gi]} =   csa_in_1[8][Gi] + csa_in_1[7][Gi] + csa_in_1[6][Gi];
    assign  {csa_out_1[7][Gi],csa_out_1[6][Gi]} =   csa_in_1[11][Gi]+ csa_in_1[10][Gi]+ csa_in_1[9][Gi];

    assign  {csa_out_2[1][Gi],csa_out_2[0][Gi]} =   csa_in_2[2][Gi] + csa_in_2[1][Gi] + csa_in_2[0][Gi];
    assign  {csa_out_2[3][Gi],csa_out_2[2][Gi]} =   csa_in_2[5][Gi] + csa_in_2[4][Gi] + csa_in_2[3][Gi];
    assign  {csa_out_2[5][Gi],csa_out_2[4][Gi]} =   csa_in_2[8][Gi] + csa_in_2[7][Gi] + csa_in_2[6][Gi];

    assign  {csa_out_3[1][Gi],csa_out_3[0][Gi]} =   csa_in_3[2][Gi] + csa_in_3[1][Gi] + csa_in_3[0][Gi];
    assign  {csa_out_3[3][Gi],csa_out_3[2][Gi]} =   csa_in_3[5][Gi] + csa_in_3[4][Gi] + csa_in_3[3][Gi];

    assign  {csa_out_4[1][Gi],csa_out_4[0][Gi]} =   csa_in_4[2][Gi] + csa_in_4[1][Gi] + csa_in_4[0][Gi];
    assign  {csa_out_5[1][Gi],csa_out_5[0][Gi]} =   csa_in_5[2][Gi] + csa_in_5[1][Gi] + csa_in_5[0][Gi];
end
```

Проектирование и отладка Verilog

```
for(Gi=0; Gi<13; Gi=Gi+1)
begin: part_prod_gen
    assign  pp_val[Gi]   =~(full_addend[2+Gi*2:Gi*2]==3'b111) &~(full_addend[2+Gi*2:Gi*2]==3'b000);
    assign  pp_sign[Gi]  =  full_addend[2+Gi*2];
    assign  pp_shift[Gi] = (full_addend[2+Gi*2:Gi*2]==3'b100) | (full_addend[2+Gi*2:Gi*2]==3'b011) ;

    if(Gi==0)
    begin
        assign  pp_mant[Gi]     =   {{3{pp_sign[Gi]}},{24{pp_sign[Gi]}} ^ full_mant1_s1} << pp_shift[Gi];
        assign  pp_add_sign[Gi] =   ( pp_val[Gi] & pp_sign[Gi] )                         << pp_shift[Gi];
        assign  part_prod[Gi]   =   { {(48-24-3){pp_sign[Gi]}}, pp_mant[Gi] } & {48{pp_val[Gi]}};
    end
    else begin
        assign  pp_mant[Gi]     =   ((({24+2{pp_sign[Gi]}} ^ full_mant1_s1) << pp_shift[Gi] ) & {24+2{pp_val[Gi]}}, pp_add_sign[Gi-1]};
        assign  pp_add_sign[Gi] =   ( pp_val[Gi] & pp_sign[Gi] )            << pp_shift[Gi];
        assign  part_prod[Gi]   =   { {(48-24-3){pp_val[Gi] & pp_sign[Gi]}}, pp_mant[Gi] } << 2*(Gi-1);
    end
end

for (Gi=0 ; Gi<13; Gi=Gi+1)
begin: csa_in_assign1
    assign  csa_in_1[Gi]    =   part_prod[Gi];
end

for(Gi=0; Gi<48; Gi=Gi+1)
begin: csa_out_form
    assign  {csa_out_1[1][Gi],csa_out_1[0][Gi]}
    assign  {csa_out_1[3][Gi],csa_out_1[2][Gi]}
    assign  {csa_out_1[5][Gi],csa_out_1[4][Gi]}
    assign  {csa_out_1[7][Gi],csa_out_1[6][Gi]}

    assign  {csa_out_2[1][Gi],csa_out_2[0][Gi]}
    assign  {csa_out_2[3][Gi],csa_out_2[2][Gi]}
    assign  {csa_out_2[5][Gi],csa_out_2[4][Gi]}

    assign  {csa_out_3[1][Gi],csa_out_3[0][Gi]}
    assign  {csa_out_3[3][Gi],csa_out_3[2][Gi]}

    assign  {csa_out_4[1][Gi],csa_out_4[0][Gi]}
    assign  {csa_out_5[1][Gi],csa_out_5[0][Gi]}
end
```

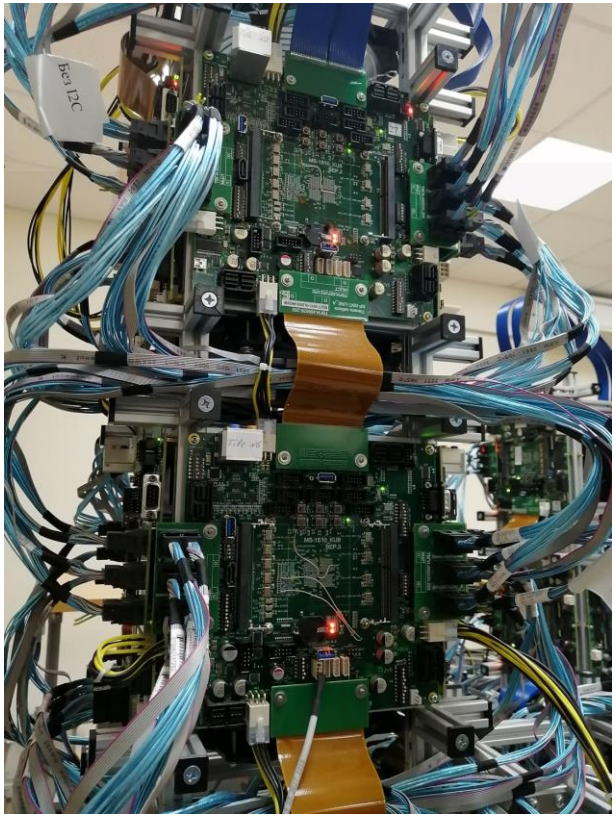


Прототипирование на ПЛИС

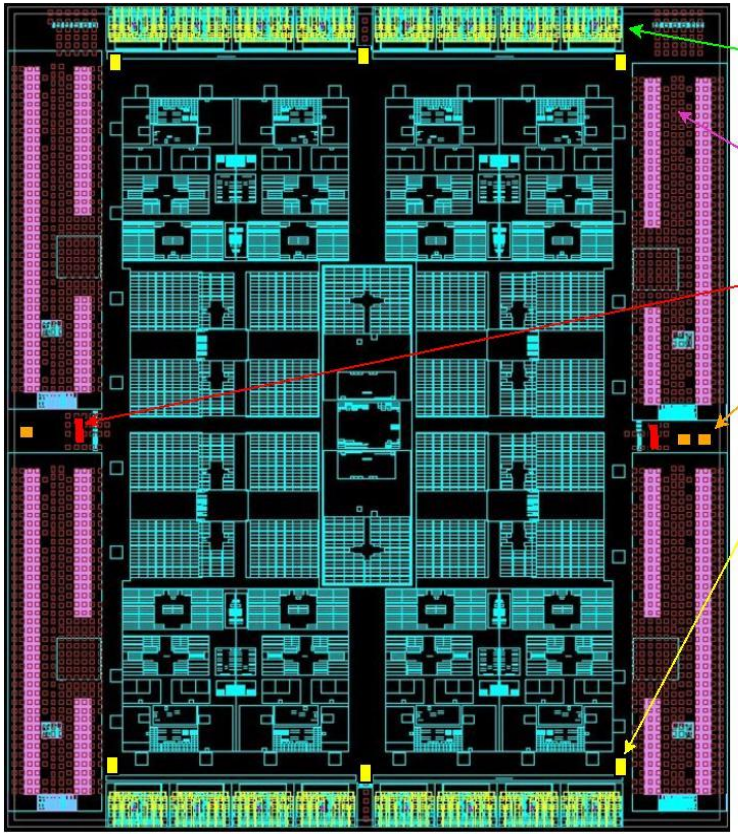


Двухпроцессорная система на прототипах МП Эльбрус-16С

- оперативная память 64Гб DDR4 SODIMM
- частота работы – 3 МГц (в 10^5 раз быстрее моделирования)
- позволяет подключать периферийные устройства
- позволяет загружать Linux
- позволяет находить логические ошибки и отлаживать ПО (ОС, драйверы) до выпуска чипа



Происхождение IP в процессорах Эльбрус



физический уровень Synopsys CEI-6G PHY для реализации линков ввода-вывода и межпроцессорных линков

физический уровень Synopsys DDR3 PHY для реализации 4-х каналов памяти DDR3

библиотека LVDS фирмы Dolphin Technology

PLL фирмы True circuits

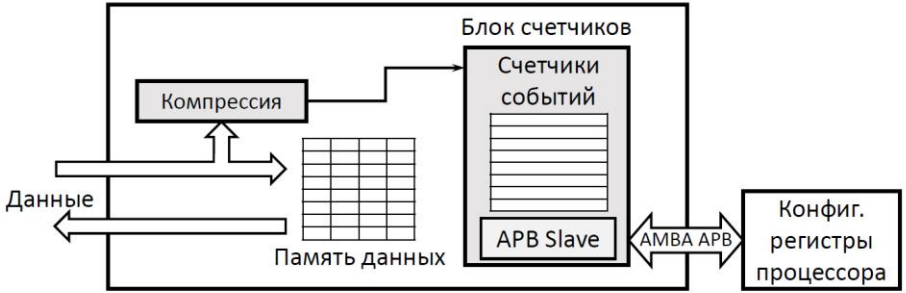
термосенсор фирмы Moortec

Вся прочая логика и физический дизайн разработаны в России, могут развиваться без ограничений

Научная работа и защита диплома

Анализ эффективности

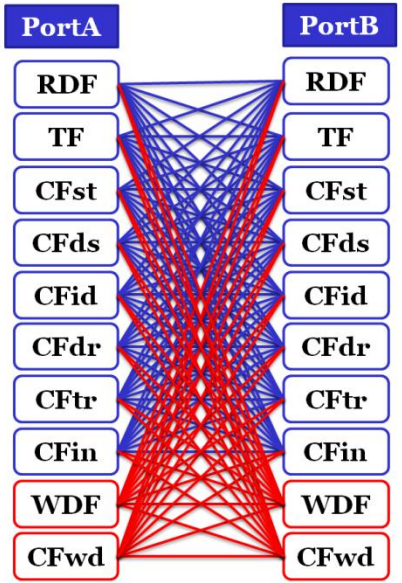
Банк L3-кэша в составе FPGA-прототипа



- Алгоритмы компрессии были интегрированы в L3-кэш FPGA-прототипа процессора «Эльбрус-8С2» без занесения сжатых данных в память
- Была разработана система мониторинга, позволяющая собирать данные о работе схем компрессий через программно-доступные счетчики событий
- Использовались задачи пакета SPEC CPU2000, запускаемые в однопоточном режиме

10/15

Неисправности в блоках двухпортовой памяти



- **Всего необходимо рассмотреть 100 комбинаций неисправностей, из них 36 – ранее не учитывались.**
- **Среди комбинаций встречаются симметричные: например, RDF&CFdr и CFdr&RDF. Для их проверки можно разработать тестовый алгоритм, обнаруживающий одну из них в некотором порядке портов, и применить его затем в обратном порядке портов. Таким образом, количество рассматриваемых комбинаций уменьшается вдвое.**
- **Некоторые оставшиеся комбинации неисправностей нереализуемы, некоторые проверяются автоматически при проверке других комбинаций неисправностей.**

Направления и тематика разработок

Направления разработок

- **многоядерные высокопроизводительные** МП Эльбрус (Эльбрус-8С, Эльбрус-16С, Эльбрус-12С, Эльбрус-32С)
- МП Эльбрус для **встроенных применений** (Эльбрус-1С+, Эльбрус-2С3)
- МП с архитектурой **SPARC** (МЦСТ-R1000)
- контроллер периферийных интерфейсов (**КПИ**)

Тематика разработок

- **VLIW-архитектура** МП
- микроархитектура процессорного ядра **Эльбрус**
- **подсистема памяти:**
  - кэш-память L1/L2/L3
  - многопроцессорные NUMA-системы
  - контроллеры памяти
- **сеть-на-кристалле**
- **интеграция IP-блоков** – графических DSP-сопроцессоров, контроллеров высокоскоростных интерфейсов (PCIe 4.0, USB 3.0, 10Gb Ethernet, …)

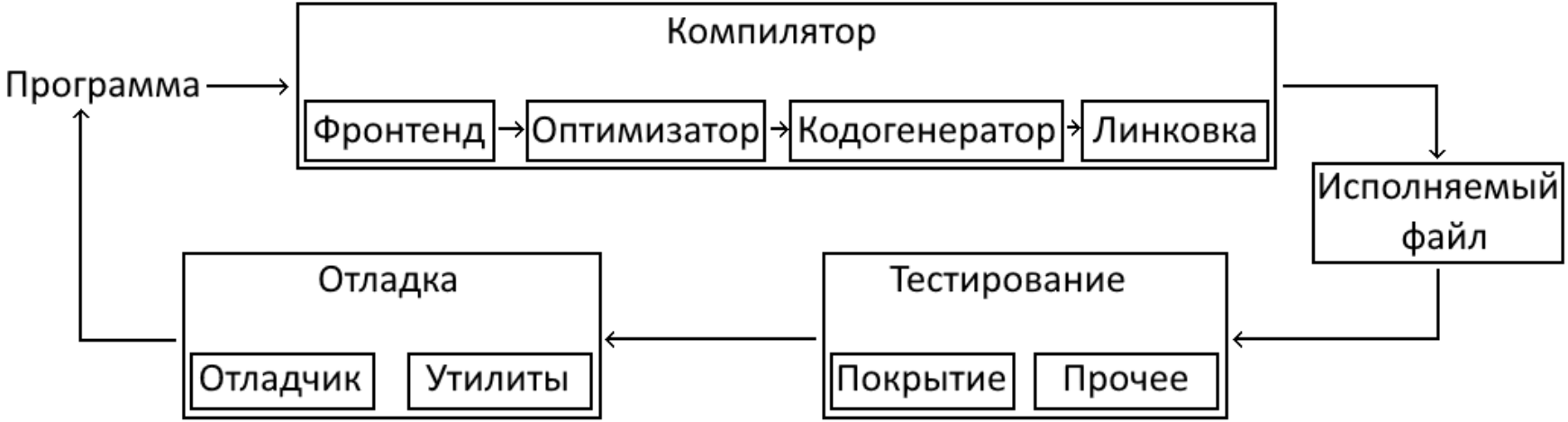
Разработка оптимизирующего компилятора

Шампаров Виктор Евгеньевич
АО «МЦСТ», МФТИ
Victor.E.Shamparov@mcst.ru

Что такое компилятор

Компилятор – комплекс программ, преобразующих исходный код в исполняемый файл

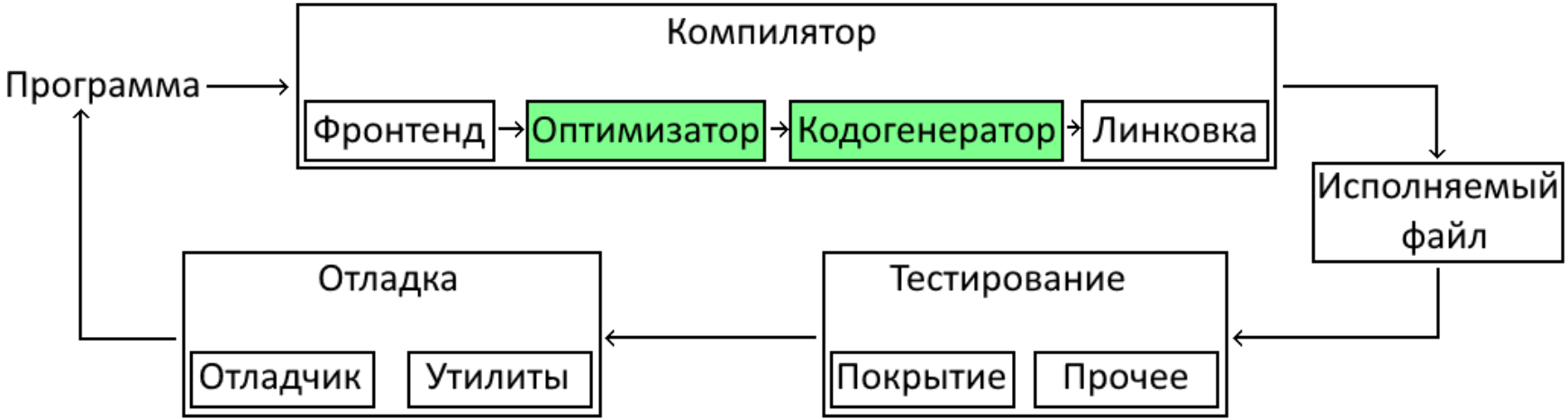
Входит в **toolchain** – набор программ для разработки



Оптимизации

Улучшение характеристик программы без изменения ее поведения

- Ускорение исполнения
- Уменьшение размера исполняемого файла
- *(дополнительно)* Поиск некоторых ошибок



Эффект оптимизаций

Ускорение выполнения программы в зависимости от режима оптимизации:

- `-O0` – почти без оптимизаций
- `-O4` – максимальные автоматические оптимизации
- `peak` = -O4 + профиль + ручной подбор опций оптимизации

	-O4 vs -O0	peak vs -O4
Среднее геометрическое	x 7,35	x 1,84
Медиана	x 4,95	x 1,51
Min	x 1,77	x 1,04
Max	x 152,50	x 44,71

Направления развития

- **Совершенствование** оптимизатора
  - Новые оптимизации
  - Улучшение существующих оптимизаций
- **Поддержка** новых языков
- **Поддержка** новых возможностей аппаратуры
- **Анализ** возможностей ускорения программ
- **Интеграция** с основными компиляторами

Смежная деятельность

- Аналитика
- Бинарный компилятор
- Компоненты системы программирования
- Математическая библиотека EML для быстрых вычислений
- Тестирование

Примеры разработок студентов

- Преобразование массивов структур в отдельные массивы для улучшения работы с кэш-памятью
- Поддержка выражений в директивах OpenMP
- Оптимизация конструкции switch: цепочка if-else, сбалансированное дерево if-else или таблица переходов?

Операционные системы

Жмурин Андрей Валентинович
АО «МЦСТ»

Уровни операционной системы

Дистрибутивы и ОС с поддержкой Эльбрус

линукс эльбрус

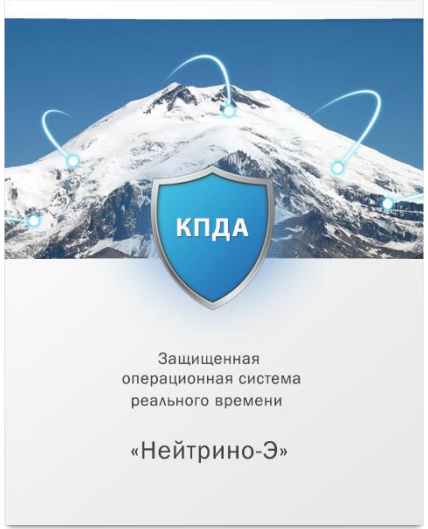
ASTRA LINUX®

alt linux

debian 8 Jessie

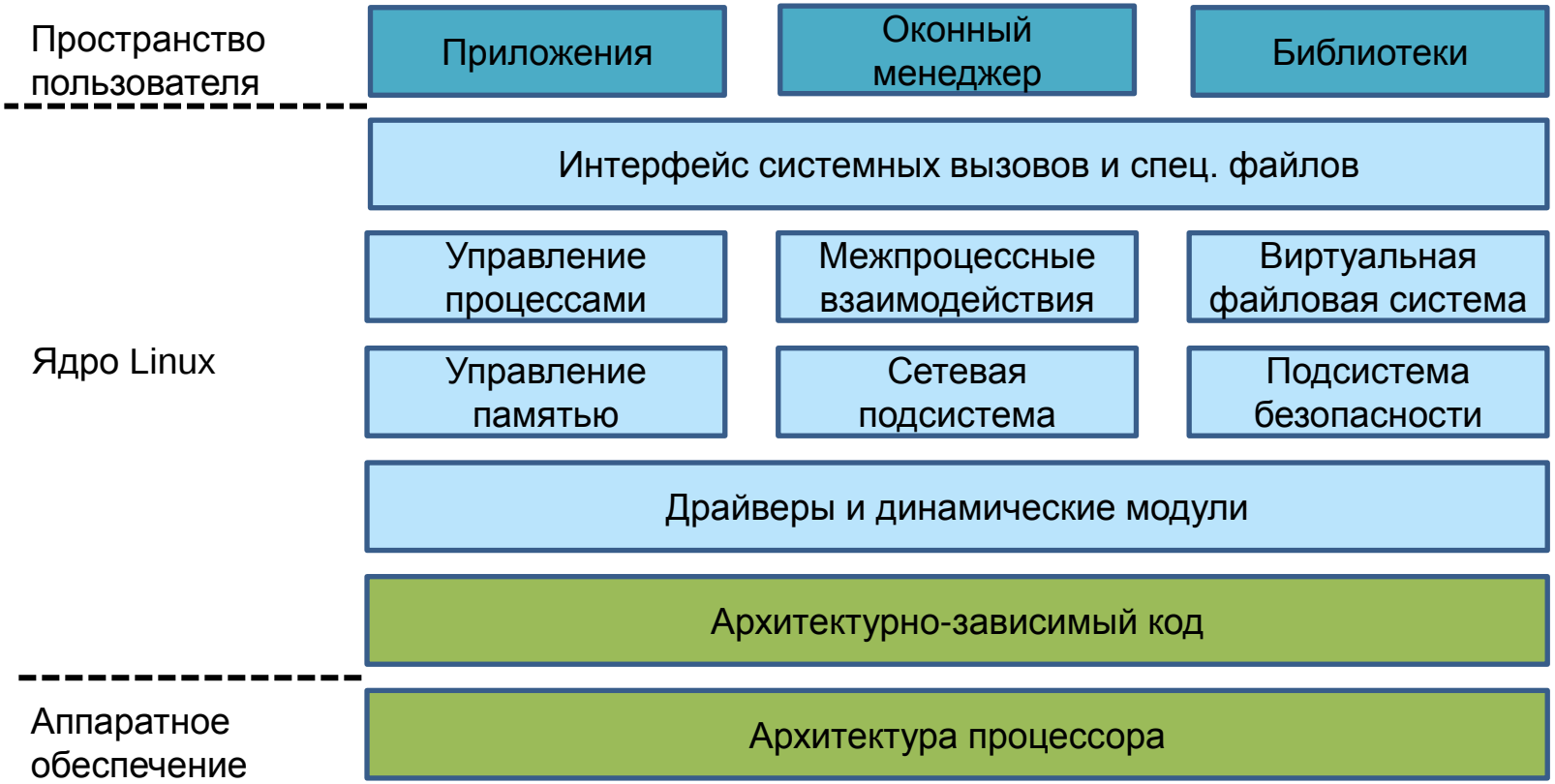
СУХОЙ

БагрОС



Ядро Linux: динамика проекта

Структура ядра Linux



Направления развития ядра ОС Эльбрус

- **Обновление ядра**
  Эльбрус Линукс 6.0 – на основе ядра Linux 5.4
- **Поддержка процессоров Эльбрус и SPARC**
  Новые процессоры: Эльбрус-16С, Эльбрус-32С, SPARC R2000+
  Стеки, таблицы страниц, механизмы синхронизации, полуспекулятивный режим
- **Поддержка нового функционала**
  KVM (+ QEMU, libvirt, virt-manager)
  Real-Time, NUMA Balancing, Transparent Hugepages, kexec, энергосбережение, гибернация, BPF, CRIO
- **Драйверы устройств**
  Elbrus PIC, IOMMU, часы, датчики
  PCI, USB, сеть, графика, звук, дисковые массивы
- **Защищенный режим**
  Контейнерная виртуализация в защищенном режиме
- **Оптимизация производительности**

Московский физико-технический институт (НИУ)
Физтех-школа радиотехники и компьютерных технологий

Базовая кафедра информатики и вычислительной техники

Базовая кафедра

Заведующий кафедрой – д.т.н., проф. ***Перекатов Валерий Иванович***, Заслуженный деятель науки РФ, заслуженный профессор МФТИ

- Основана в 2006 году, наследует несколько кафедр (начиная с кафедры ЭВМ 1955), связанных с созданием ***отечественных*** высокопроизводительных вычислительных систем
- **14** учебных курсов
- Среди преподавателей – **3** доктора наук, **2** кандидата наук
- **30** студентов и **2** аспиранта МФТИ проходят обучение в настоящее время
- Более **60** выпускников за последние 10 лет

Базовая кафедра: читаемые курсы

Направления:

- **архитектура современных вычислительных систем и сетей**
- **логическая верификация аппаратуры**
- **оптимизирующие компиляторы и операционные системы**
- **физическое проектирование СБИС, проектирование вычислительных модулей и комплексов**

3 курс	4 курс		5 курс	
Проектирование ЦУ на ПЛИС	Архитектура вычислительных систем 1		Архитектура вычислительных систем 2	
	Вычислительные сети		Верификация	
	Операционные системы 1	Компиляторы 1	Нейронные сети	Операционные системы 2
	Элементы цифр. устройств	Целостность сигналов	Физ. проектирование СБИС	Компиляторы 2

Участие студентов в проектной деятельности

Направления

- Архитектура и логическое проектирование микропроцессоров и СБИС (Эльбрус, SPARC, южный мост)
- Физическое проектирование СБИС
- Операционные системы (ядро, виртуализация, защищенный режим, драйверы)
- Системы программирования (оптимизирующие языковые и бинарные компиляторы, библиотеки)
- Верификация и моделирование аппаратуры

Академическая деятельность

Участие в конференциях:

- Форум «Микроэлектроника» (г. Алушта, Крым)
- Ежегодная научная конференция МФТИ (секция)
- Engineering & Telecommunication
- Spring/Summer Young Researchers' Colloquium on Software Engineering

Публикации в журналах ВАК:

- «Радиопромышленность»
- «Программная инженерия»
- «Наноиндустрия»
- «Труды Института системного программирования РАН»
- «Приборы»
- SYRCoSE (Proceedings)

Публикации в журналах IEEE:

- International Conference on Engineering & Telecommunication (Proceedings)

История

- Школа академика С.А. Лебедева
- **1967 год** – БЭСМ-6 (стековая адресация, виртуальная память)
- **1975 год** – на БЭСМ-6 выполнялся обсчет телеметрии во время полета «Союз-Аполлон»
- **1985 год** – Эльбрус-2, используется в ПРО Москвы
- **1992 год** – создание ТОО «Московский Центр SPARC-технологий»
- **1998 год** – «Эльбрус-90микро», первый отечественный ВК на интегральной схеме, применяются на командных пунктах ВКО
- **2010 год** – «Эльбрус-S», первый МП с архитектурой Эльбрус для многопроцессорных ВК с распределенной памятью
- **2014 год** – «Эльбрус-8С», первый 8-ядерный отечественный МП на технологии 28 нм
- **2021 год** – «Эльбрус-16С», 16-ядерный процессор на технологии 16 нм
- **2022 год - н.в.** – «Эльбрус-32С», 32-ядерный процессор на технологии 6 нм, DDR5, 6TF, поддержка ускорения нейросетевых и криптографических вычислений

Преимущества учебы на нашей кафедре

- **современные** учебные курсы
- опыт **командной работы** в **крупном востребованном проекте**
- навыки **анализа и разработки** программных/аппаратных решений
- работа **на стыке науки и практики**
- написание диплома на основе **полученного опыта**
- заработная плата, пропорциональная **компетенции** в рамках рабочего проекта
- возможность продолжения обучения **в магистратуре и аспирантуре**

Преимущества работы в АО МЦСТ

- Молодой высокоинтеллектуальный **коллектив**
- Опытное и адекватное **руководство** (+лояльное к студентам!)
- Глубокая история фирмы и её **устойчивость**
- Работа на острие индустрии
- Рекомендации и **опыт** мирового уровня