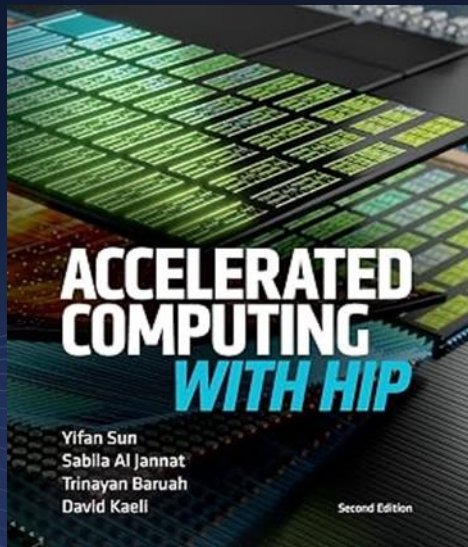




SyntacoreTM
RISC-V cores and tools



Высокопроизводительные вычисления с помощью HIP

Второе издание на русском языке

Сведения о докладчике

Тимур Турсунович Палташев — доктор технических наук, профессор и международный эксперт в области микроэлектроники, компьютерной архитектуры и графических процессоров.

- До мая 2025 г. работал старшим менеджером и руководителем проектов в подразделении Radeon Technologies Group и Artificial Intelligence Group в Advanced Micro Devices (AMD)
- Ранее также успешно работал в компании S3 Graphics, в РосАтоме и университетах ИТМО, ВГУ, SDSMT и NPU (SFBU)
- Автор более 70 научных трудов, зарегистрировано 33 патента США в сфере архитектуры микропроцессоров и параллельных вычислительных структур
- В 1991 году стоял у истоков создания GraphiCon — крупнейшей на постсоветском пространстве международной конференции по компьютерной графике и зрению. 36-я конференция в 2026 г.



**Цель доклада: представить второе издание
в серии «Суперкомпьютерное образование»**

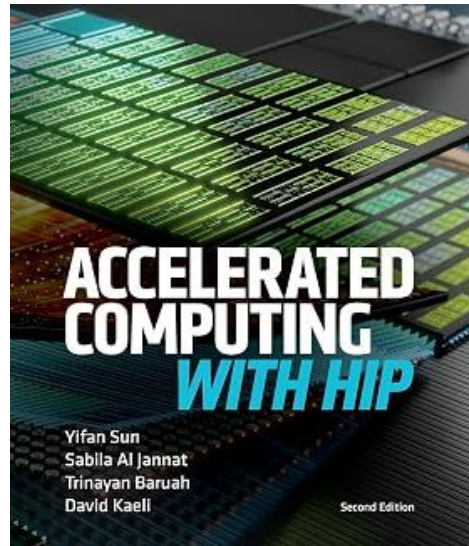
Ифань Сун, Сабила аль-Джаннат, Тринаян Баруа,
Дэвид Каэли

ВЫСОКОПРОИЗВОДИТЕЛЬНЫЕ ВЫЧИСЛЕНИЯ С ПОМОЩЬЮ HIP

Учебное пособие

Второе издание на русском языке

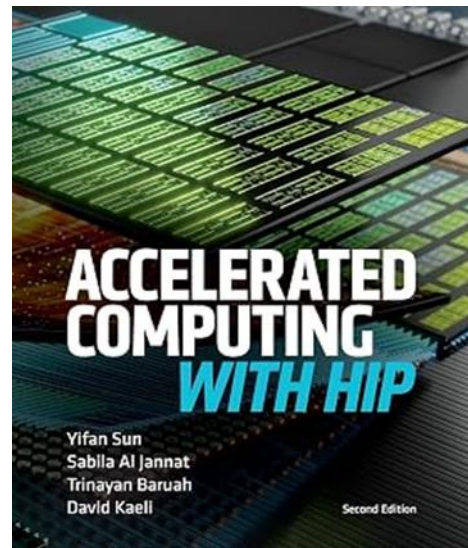
На основе второго издания на английском языке,
опубликованного в США; актуализировано и
дополнено



Второе издание на
английском языке —
Amazon

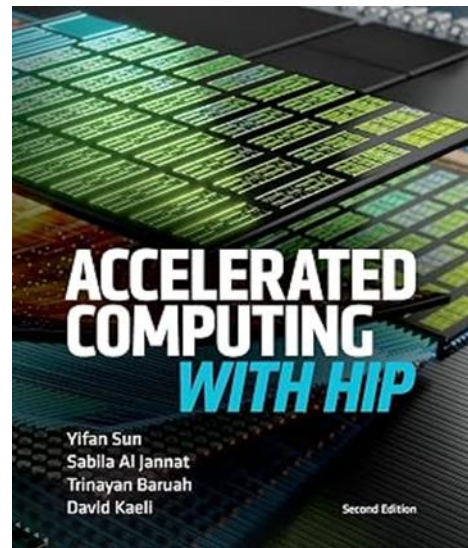
HIP — ключевой интерфейс программирования в экосистеме ROCm

- HIP — C++ API среды выполнения и язык программных ядер
- Модель программирования и синтаксис близки к CUDA
- HIP предоставляет функции управления памятью, потоками, событиями, синхронизацией, устройствами и запуском программных ядер
- Инструменты Hipify помогают переносить приложения CUDA в HIP
- Единая кодовая база HIP может отдельно компилироваться для платформ AMD и NVIDIA в поддерживаемых конфигурациях
- На платформах AMD HIP входит в программную экосистему ROCm
- Библиотеки hipBLAS, hipFFT, hipRAND и hipSPARSE предоставляют переносимые интерфейсы к соответствующим бэкендам ROCm и CUDA



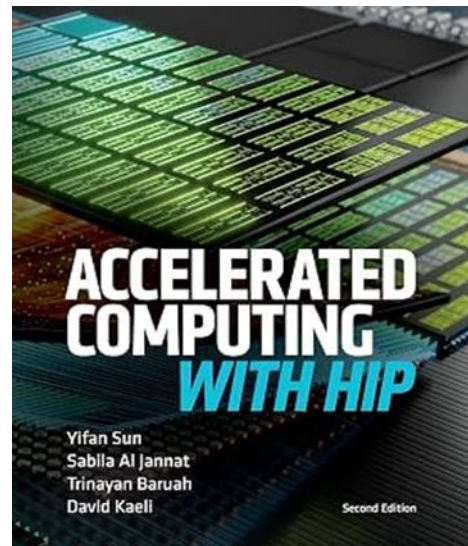
Почему потребовалось второе издание

- Быстрое развитие аппаратных платформ AMD и программной экосистемы ROCm/HIP
- Необходимость систематического учебного маршрута, дополняющего быстро меняющуюся документацию, которой часто недостаточно для системной подготовки специалистов
- Обновление материала для разработчиков, студентов, преподавателей и команд заказчиков
- Исправление технических, терминологических и языковых ошибок предыдущих изданий
- Расширение материала: программные ядра, API среды выполнения, оптимизация, системы с несколькими ГП, машинное обучение и центры обработки данных
- Актуализация примеров кода, библиотек и инструментов
- Формирование последовательной русскоязычной терминологии и двунаправленного глоссария



Основные изменения во втором издании

- Количество глав увеличено с **11** до **13**
- Материал по HIP разделен на самостоятельные главы: начало программирования, программные ядра и API среды выполнения
- Добавлена отдельная глава об оптимизации производительности
- Расширена глава о программировании для систем с несколькими ГП
- Машинное обучение перенесено из приложения в отдельную главу и дополнено примерами PyTorch и TensorFlow
- Расширен материал о контейнерах, Kubernetes и SLURM
- Добавлено приложение, посвященное OmniTools
- Актуализированы сведения о поколениях ГП AMD и особенностях их архитектуры, а также о библиотеках и инструментах ROCm
- Исправлены технические ошибки в тексте, иллюстрациях и листингах кода
- Разработан англо-русский и русско-английский глоссарий
- Второе издание на русском языке дополнено материалами, отсутствующими во втором издании на английском языке



Программная экосистема ROCm/HIP

Фреймворки и средства переносимого программирования

PyTorch, TensorFlow, Kokkos, RAJA

Библиотеки ИИ и обработки данных

MIOpen, MIGraphX, MIVisionX, RPP

Математические библиотеки и параллельные примитивы

rocBLAS / hipBLAS, rocSPARSE / hipSPARSE,
rocFFT / hipFFT, rocRAND / hipRAND,
rocSOLVER / hipSOLVER, rocPRIM, rocThrust,
rocWMMA

Коллективные операции и обмен данными между ГП

RCCL, rocSHMEM

Модели программирования и API

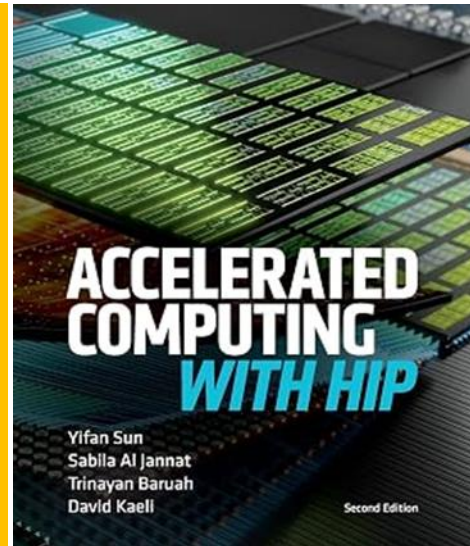
HIP, OpenMP, OpenCL

Компиляторы и среды выполнения

LLVM/Clang, hipcc, HIP Runtime, ROCr

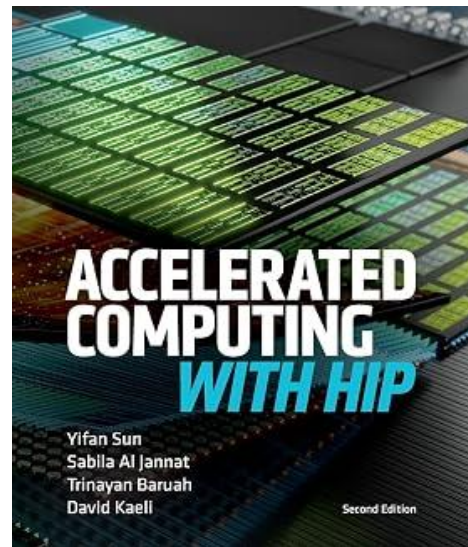
Инструменты разработки и управления

ROCgdb
ROCprofiler-SDK / rocprofv3
AMD SMI
rocminfo



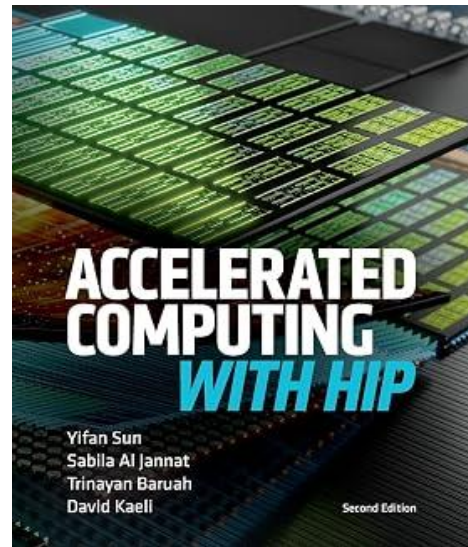
Второе издание: масштаб и основные дополнения

- Около 385 страниц с иллюстрациями
- 13 глав, три приложения и двунаправленный глоссарий
- Предисловие ко второму изданию на русском языке
- Вступительное слово профессора Джека Донгарры
- Новые самостоятельные главы о программных ядрах HIP, API среды выполнения и оптимизации
- Расширенные главы о библиотеках ROCm и системах с несколькими ГП
- Новая глава о PyTorch и TensorFlow на ROCm
- Обновленный материал о контейнерах, Kubernetes и SLURM
- Добавлены раздел о Linaro Forge и приложение, посвященное OmniTools
- Актуализированные сведения о поколениях ГП AMD и особенностях их архитектуры, а также о программной экосистеме ROCm/HIP
- Исправленные примеры кода и согласованная техническая терминология



Содержание (1 из 3)

1. Введение
2. Начинаем программировать на языке HIP
3. Разработка программных ядер на языке HIP
4. API среды выполнения в HIP
5. Шаблоны программирования для графических процессоров



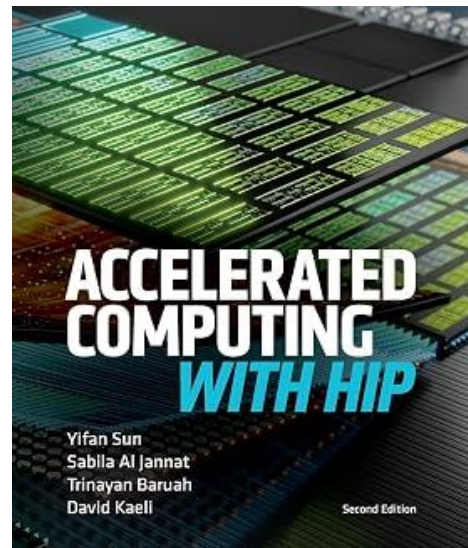
Содержание (2 из 3)

6. Внутренние компоненты графического процессора AMD

- Графические процессоры AMD
- Обзор архитектуры
- Командный процессор и механизм DMA
- Диспетчеризация рабочих групп
- Секвенсор
- Блок SIMD
- Расхождение тредов
- Слияние обращений к памяти
- Иерархическая структура памяти
- Графические процессоры RDNA компании AMD

7. Инструменты HIP и перенос программ CUDA с помощью Hipyfy

8. Оптимизация производительности в HIP



Содержание (3 из 3)

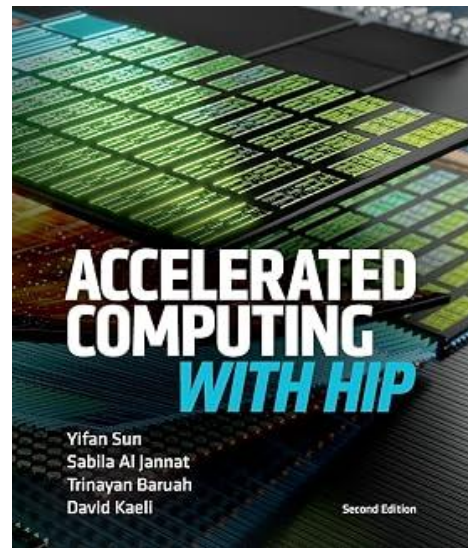
9. Библиотеки ROCm
10. Программирование для системы с несколькими графическими процессорами
11. Машинное обучение с помощью ROCm
12. ROCm в центрах обработки данных
13. Инструменты сторонних разработчиков, включая Linaro Forge

Приложение А. Установка ROCm

Приложение В. Ассемблер CDNA

Приложение С. OmniTools

Англо-русский и русско-английский глоссарий



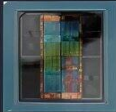
Использование нескольких ГП в гетерогенных системах

- Для многих современных высокопараллельных нагрузок, включая обучение крупных языковых и других трансформерных моделей, производительности одного ГП недостаточно
- Системы с несколькими ГП широко используются в гетерогенных вычислительных кластерах, суперкомпьютерах и системах искусственного интеллекта
- HIP и ROCm в сочетании с MPI, RCCL и средствами управления заданиями позволяют масштабировать приложения от одного узла до крупных суперкомпьютерных систем



AMD
INSTINCT

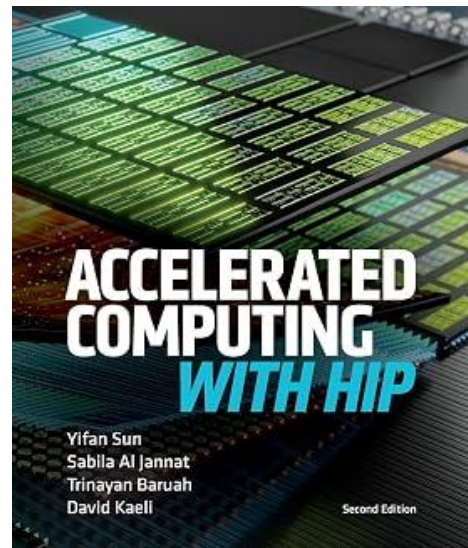
Powering Top Supercomputers

	AMD Instinct™ MI250X		AMD Instinct™ MI300A		AMD Instinct™ MI430X
First dual-die data center GPU with coherent CPU-GPU memory over Infinity Fabric		First data center APU unifying EPYC CPU, Instinct GPU, and HBM3 memory		Leadership hardware based FP64 with HBM4 memory and bandwidth	

Имеются в виду текущие и анонсированные поколения AMD Instinct. AMD указывает ожидаемую доступность MI430X в 2027 г.

Подходы к программированию систем с несколькими ГП

- Использование API HIP для выбора устройств, управления памятью и потоками, а также запуска программных ядер на нескольких ГП
- Поддерживаются следующие подходы:
 - Использование потоков HIP
 - Использование тредов ЦП
 - Программирование на основе MPI
 - Использование коллективных операций RCCL для обмена данными между ГП в пределах одного или нескольких узлов
- Программист может комбинировать эти подходы в одном приложении
 - Например, два тредя ЦП могут управлять четырьмя ГП, а для каждого ГП используются отдельные потоки HIP
- В учебном пособии описаны базовые концепции программирования систем с несколькими ГП, позволяющие программистам начать работу в таких системах



Использование потоков для программирования нескольких ГП

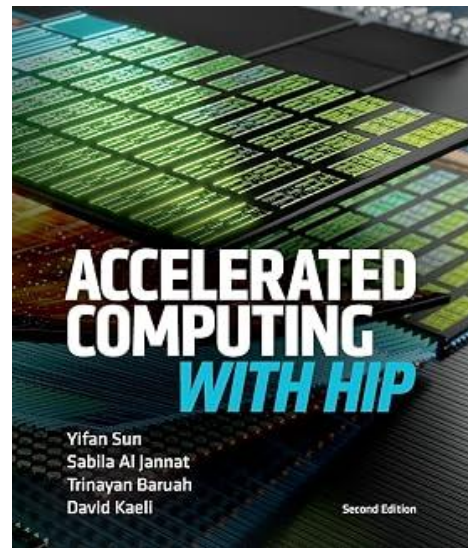
Ключевая идея:

- Для каждого из ГП создается отдельный поток HIP
- Операции в каждом потоке выполняются в порядке их постановки, а операции в потоках разных ГП могут выполняться параллельно, если между ними нет зависимостей

Базовый алгоритм:

1. Выбрать ГП и создать для него поток HIP
2. Выделить на каждом из ГП необходимую память
3. Скопировать на каждый из ГП соответствующий фрагмент входных данных
4. Асинхронно запустить программные ядра в соответствующих потоках
5. Синхронизировать потоки перед использованием результатов
6. При необходимости скопировать результаты в память хоста

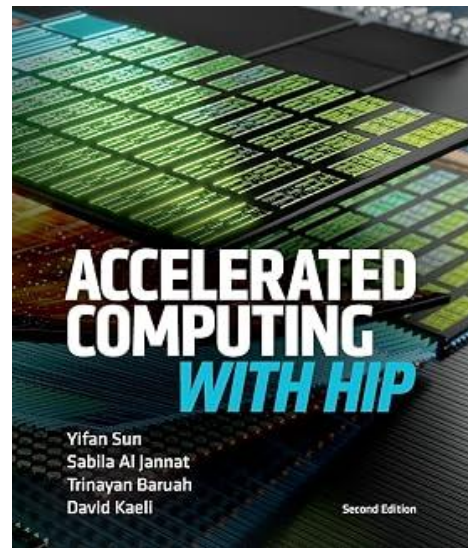
В книге **этот подход показан на примере сложения векторов**: каждый из ГП обрабатывает собственный фрагмент входных данных



Библиотека коллективных операций RCCL

ROCm Communication Collectives Library (RCCL)

- RCCL предоставляет коллективные операции для нескольких ГП, в том числе в многоузловых системах
- Библиотека оптимизирована для PCIe и xGMI внутри узла, а также для InfiniBand, RoCE и TCP/IP между узлами
- API RCCL совместим с API библиотеки NVIDIA Collective Communications Library (NCCL); типы и функции сохраняют префикс `nccl`*
- RCCL может использоваться в одно- и многопроцессных приложениях, в том числе совместно с MPI
- Рассматриваемые в книге операции: Broadcast, Reduce, AllReduce, AllGather и ReduceScatter
- В книге приведены примеры использования Broadcast и AllReduce



Пример с использованием RCCL + MPI

```
ncclUniqueId id;
ncclComm_t comm;
float *sendbuff, *recvbuff;
hipStream_t stream;

if (myRank == 0)
    ncclGetUniqueId(&id);

MPI_Bcast(&id, sizeof(id), MPI_BYTE,
         0, MPI_COMM_WORLD);

hipSetDevice(myRank);
hipMalloc(&sendbuff, size * sizeof(float));
hipMalloc(&recvbuff, size * sizeof(float));
hipStreamCreate(&stream);

ncclCommInitRank(&comm, nRanks, id, myRank);

ncclAllReduce(sendbuff, recvbuff, size,
             ncclFloat, ncclSum, comm, stream);

hipStreamSynchronize(stream);
```

Коммуникатор,
буферы и поток

Получение и рассылка
уникального
идентификатора

Выбор ГП, выделение
памяти, создание потока
и коммуникатора

Выполнение AllReduce и
синхронизация потока

Упрощение: в этом примере myRank совпадает с локальным номером ГП; в многоузловом запуске для hipSetDevice обычно используется локальный ранг процесса

Образовательная ценность и поддержка экосистемы HIP/ROCm

Систематическое учебное пособие и справочник по HIP/ROCm

- Содействие росту сообщества программистов, использующих HIP наряду с CUDA
- Поддержка открытой программной экосистемы ROCm (инструменты и библиотеки)
- Распространение знаний об архитектуре графических процессоров AMD и гетерогенных вычислениях
- Распространение методов профилирования, настройки и повышения производительности вычислений в суперкомпьютерах и системах искусственного интеллекта
- Учебник готовится к выпуску в печатном и электронном форматах

Поддержка преподавания программирования с использованием HIP в университетах и техникумах

- AMD University Program: [ссылка](#)
- Видеолекции авторов на английском языке: [ссылка](#)
- Готовы развивать комплект русскоязычных учебных слайдов и создавать видеоматериалы

AMD University Program



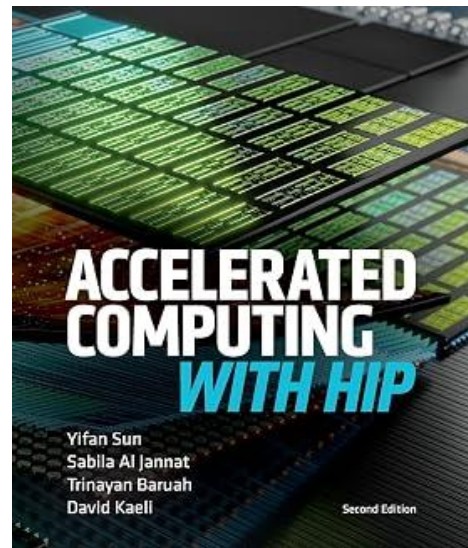
Видеолекции авторов
на YouTube



Текущий статус проекта (1 из 2)

Настоящее второе издание на русском языке подготовлено на основе второго издания книги AMD «Accelerated Computing with HIP» на английском языке, опубликованного в США в начале 2025 года

- При сохранении структуры и авторской основы оригинала отдельные разделы текста на русском языке были исправлены, актуализированы и дополнены
- В основной текст, примеры кода и иллюстрации внесены технические исправления; добавлены материалы, отсутствующие во втором издании на английском языке
- Учтены изменения в аппаратных платформах AMD, программной экосистеме ROCm/HIP и современной практике параллельного программирования
- Подготовлен двунаправленный терминологический глоссарий



Текущий статус проекта (2 из 2)

- Общее и научное руководство проектом, научная редакция — **профессор Тимур Палташев**, Синтакор
 - Перевод на русский язык, техническое редактирование, актуализация текста, подготовка дополнений, глоссария и первой полной версии оригинал-макета — **Олександр Куприянчук**, Luxoft
 - Финальная доработка оригинал-макета и издательская подготовка издания на русском языке — **издательство «МАКС Пресс»**
 - Независимым коллективом подготовлен перевод второго издания книги на китайский язык:
[Китайский перевод на GitHub](#)
- 
- Финальный этап — **целевой сбор средств на завершение выпуска второго издания книги на русском языке**
 - Значительная часть переводческих, редакционных и макетных работ уже выполнена участниками проекта в инициативном порядке, в личное время и за счет собственных средств

Целевой сбор на второе издание

- В связи с прекращением ранее предусмотренного финансирования от корпорации AMD проводится целевой сбор средств на завершение проекта
- **Целевая сумма сбора для проекта — 800 тыс. рублей**
- Средства пойдут на финальную издательскую подготовку, выпуск электронной версии и печать базового тиража, а также на оплату уже выполненных переводческих и редакционных работ
- Механизм финансирования — целевой сбор пожертвований
- Оператор сбора — Фонд развития прикладной математики и информационных технологий имени академика А. Н. Тихонова
- **Проект соответствует ключевым направлениям деятельности Фонда:** содействие исследованиям в области вычислительной математики, математического моделирования, компьютерных наук, искусственного интеллекта и в других областях математики и информатики, а также изучение и обобщение опыта, научной и практической деятельности российских, зарубежных и международных организаций
- **Поддержите выпуск систематического учебного пособия и справочника по HIP/ROCm.** Страница целевого сбора на портале Фонда имени А. Н. Тихонова:
<https://tikhonovfoundation.ru/projects/19>

**Страница
целевого сбора**



Генеральный спонсор проекта ООО «ЭСИК»



Дальнейшее развитие проекта

- Материалы настоящего издания могут стать основой будущего третьего издания на русском или английском языке
- Разрабатывается новая глава об использовании HIP в компьютерной графике
- Рассматривается возможность подготовки отдельной главы об OpenMP
- Однозначно необходима специальная глава по доверенному программированию на HIP с использованием моделей LLM и агентов ИИ
- Возможное продолжение:
 - Обновляемый репозиторий примеров
 - Лабораторные работы и учебные материалы
 - Презентации и видеоматериалы
 - База знаний для разработчиков и систем разработки ПО с использованием ИИ



SyntacoreTM
RISC-V cores and tools

**Спасибо
за внимание!**