Федеральное государственное бюджетное образовательное учреждение

высшего образования

«Новосибирский государственный технический университет»

Кафедра Автоматики

**Отчет ПО ПРАКТИКЕ**

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(наименование практики в соответствии с учебным планом)

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Направление подготовки: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

|  |  |
| --- | --- |
| Выполнил:  Студент \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  (Ф.И.О.)  Группа \_\_\_\_\_\_\_\_\_\_\_\_\_  Факультет \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_.  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  подпись  «\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_\_ 20\_\_ г. | Проверил:  Руководитель от НГТУ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  (Ф.И.О.)  Балл: \_\_\_\_\_\_\_\_\_\_, ECTS\_\_\_\_\_\_\_\_\_\_\_\_\_,  Оценка \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  «отлично», «хорошо», «удовлетворительно», «неуд.»  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  подпись  «\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ 20\_\_ г. |

Новосибирск 2017

Оглавление

[Введение 3](#_Toc479158942)

[1. Техническое задание 4](#_Toc479158943)

[1.1. Сведения и исходные данные для разработки программы 4](#_Toc479158944)

[1.2. Цели создания программы 4](#_Toc479158945)

[1.3. Назначение и область применения 4](#_Toc479158946)

[1.4. Требования к программе 5](#_Toc479158947)

# Введение

Последние поколения процессоров показывают хорошие результаты в параллельных вычислениях, однако до видеокарт им далеко. Причина в том, что видеокарты содержат в себе много графических процессоров, предназначенных для параллельной обработки.

К сожалению, ещё не существует единого стандарта для написания программ для графических процессоров. Основными средствами для написания программ для графических процессоров являются *CUDA* от *NVidia* и *OpenCL* от *AMD*. Однако 16 февраля 2016 года группой компаний *Khronos Group* был выпущен *Vulkan API* предназначенный для программирования графики и вычислений на графических процессорах, используемых в различных платформах – от персональных компьютеров до мобильных телефонов и игровых приставок. Технология *Vulkan* разработана с заделом на будущее, поэтому она будет очень активно развиваться и поддерживаться разработчиками драйверов и графических процессоров.

Для непосредственных вычислений на графических процессорах в библиотеках *Vulkan* и *OpenCL* используется язык программирования шейдеров *GLSL*. Однако для обеспечения кроссплатформенности шейдеры компилируются в промежуточный байт-код *SPIR-V*.

На текущий момент библиотека *Vulkan* используется только такими компаниями, как *Epic Games*, *Unity Technologies* и *Valve Corporation* в их движках и инструментах для разработки компьютерных игр. Также на момент написания этого документа не было найдено ни одной библиотеки для упрощения разработки программ с использованием вычислений на графических процессорах с использованием технологии *Vulkan*. Так как разработка приложений с использованием библиотеки *Vulkan* является не самой тривиальной задачей из-за необходимости гибкой настройки графического конвейера и буферов данных, было решено разработать кроссплатформенную библиотеку для этих целей.

Новизна данной разработки заключается в том, что это будет первая библиотека для вычислений на графических процессорах, использующая вычислительные возможности видеокарт, работа с которыми происходит с использованием только вычислительных функций библиотеки *Vulkan*. С помощью данной разработки планируется изучить возможность и целесообразность использования самой библиотеки *Vulkan* для математических вычислений на графических процессорах.

Так как разработчиком библиотеки *Vulkan* заявлены лучшая по сравнению с *OpenGL* поддержка многопоточности, уменьшенная нагрузка на центральный процессор и низкоуровневый контроль над графическим процессором, то было решено сравнивать производительность вычислений с программами, написанными с использованием библиотеки *CUDA SDK*, так как технология *CUDA* является лидирующей в данном классе задач.

Практическая ценность разработки будет выявлена по результатам сравнения производительности вычислений на основе разработанной библиотеки и с использованием технологии *CUDA*.

Цель работы – исследовать возможность, целесообразность и практическую ценность вычислений на графических процессорах с использованием технологии *Vulkan*.

Задачи:

* разработать библиотеку-обёртку над вызовами библиотеки *Vulkan* и предоставить в ней *API* для вычислений на графических процессорах
* протестировать разработанную библиотеку и сравнить производительность вычислений с вычислениями на основе технологии *CUDA* с использованием аналогичных алгоритмов

# Техническое задание

## Сведения и исходные данные для разработки программы

* + 1. Наименованиепрограммы

Наименование программы: “Библиотека *vulkalc* для вычислений на графических процессорах для решения сложных математических задач”.

* + 1. Сроки разработки библиотеки

Начало разработки: 19.02.2017

Окончание разработки: 02.07.2017

## Цели создания программы

Упростить разработчикам процесс создания программ с использованием вычислений на графических процессорах с использованием технологии *Vulkan* за счёт предоставления “обёртки” над вызовами библиотеки Vulkan и набора функций для использования вычислительных возможностей видеокарт, запуска вычислительных шейдеров и получения результатов расчётов.

## Назначение и область применения

Разработка предназначена для применения при написании программ для высокопроизводительных математических вычислений на графических процессорах.

## Требования к программе

* + 1. Общие положения

Основные требования, предъявляемые к разрабатываемому продукту:

* Иметь открытый исходный код
* Иметь открытую лицензию
* Иметь соглашение по оформлению исходного кода
* Иметь инструкции по сборке
* Иметь демонстрационный проект для демонстрации примеров использования
* Иметь инструкцию по сборке демонстрационного проекта
* Иметь документацию на английском языке по публичным классам, методам, полям, функциям
* Компилироваться под операционные системы *Windows* и *Linux*
* Иметь юнит-тесты и интеграционные тесты
* Проходить все существующие тесты
* Использовать систему контроля версий
* Иметь комментарии в исходном коде
* Компилироваться и запускать тесты на целевых операционных системах при каждом изменении
* Иметь как можно меньше зависимостей
  + 1. Требования к техническому обеспечению

Для использования библиотеки необходим персональный компьютер.

ОС: *Windows 7, 8, 8.1, 10* или *Ubuntu 14.04* и выше или аналогичные

Процессор: любой x86 или x86-64 процессор

Оперативная память: не менее 512МБ

Дисковое пространство: около 2МБ для *Vulkan Runtime*, около 150МБ для *Microsoft Visual C++ Redistributable*(*Windows*), около 150МБ для видеодрайвера *Linux* или около 350МБ для видеодрайвера *Windows*.

Дискретная видеокарта с поддержкой технологии Vulkan

* + 1. Требования к функциональности

Разрабатываемая библиотека *vulkalc* должна предоставлять следующие возможности:

* Статическое и динамическое связывание с программами на операционных системах *Windows* и *Linux*
* Проверка наличия поддерживаемых графических процессоров и перечисление установленных в системе поддерживаемых графических процессоров
* Проверка наличия в системе установленного *Vulkan Runtime*
* Выбор графического процессора из списка доступных для запуска на нём вычислений
* Запуск разработанных пользователем вычислительных шейдеров
* Использование предоставленных разработчиком библиотеки *vulkalc* встроенных функций математических вычислений и связанных с ними типами данных
* Замер времени выполнения вычислений и количестве операций, выполненных на графическом процессоре
* Подготовка установленных поддерживаемых графических процессоров к вычислениям
* Освобождение графического процессора
* Запуск и остановка вычислений на поддерживаемых установленных графических процессорах
* Копирование данных в видеопамять
* Копирование данных из видеопамяти
* Выделение видеопамяти для собственных нужд определённого размера
* Очищение выделенной видеопамяти
* Сообщение кода и информации об ошибке в случае её возникновения
* Обнаружение, валидация шейдеров, компиляция шейдеров в промежуточный байт-код *SPIR-V*

Библиотека *vulkalc* должна предоставлять следующие функции для математических операций на графическом процессоре:

1. Функции работы с векторами:
   * Сложение векторов
   * Векторизованное сложение массивов векторов
   * Вычитание векторов
   * Векторизованное вычитание массивов векторов
   * Векторизованное сложение массивов векторов и скаляров
   * Векторизованное вычитание массивов векторов и скаляров
   * Векторизованное умножение массива векторов на массив скаляров
   * Векторизованное деление массивов векторов и скаляров
   * Скалярное произведение векторов
   * Векторизованное скалярное умножение массивов векторов
   * Векторизованное нахождение длины векторов в массиве
   * Свёртка векторов
2. Функции работы с матрицами:
   * Проверка равенства матриц
   * Векторизованная проверка равенства массивов матриц
   * Сложение матриц
   * Векторизованное сложение массивов матриц
   * Вычитание матриц
   * Векторизованное вычитание массивов матриц
   * Произведение матриц
   * Векторизованное произведение массивов матриц
   * Умножение матрицы на число
   * Векторизованное умножение массива матриц на массив чисел
   * Возведение матрицы в степень
   * Векторизованное транспонирование массива матриц

Библиотека *vulkalc* должна предоставлять следующие вспомогательные функции для вычисления на графическом процессоре:

* Сортировка массива чисел
* Линейный поиск в неотсортированном массиве чисел
* Векторизованное сложение, вычитание, произведение и деление массивов чисел
* Векторизованное возведение в степень массивов чисел

С библиотекой vulkalc должна поставляться утилита, предоставляющая следующий функционал:

* Получение технических характеристик установленных поддерживаемых графических процессоров – частота графического процессора, размер видеопамяти, разрядность шины памяти, пропускная способность памяти, количество видеочипов
  + 1. Требования к организации входных данных

При написании собственных вычислительных шейдеров программист сам определяет формат и вид входных данных. Формат и вид входных данных для предоставляемых шейдеров описан в документации проекта.

* + 1. Требования к организации выходных данных

При написании собственных вычислительных шейдеров программист сам определяет формат и вид выходных данных. Формат и вид выходных данных для предоставляемых шейдеров описан в документации проекта.

* + 1. Требования к временным характеристикам

Требования к временным характеристикам не предъявляются

* + 1. Требования к информационному обеспечению

Исходный код программы должен быть написан на языке *C++*. Должна использоваться стандартная библиотека языка *C++* и библиотека *Vulkan*. Классы, структуры, методы, функции, переменные, предназначенные для использования пользователем должны быть экспортированы в соответствии со стандартом.

* + 1. Требования к программным средствам, используемым программой

Программные средства, используемые библиотекой *vulkalc*, которые должны быть предоставлены ОС семейства *Windows*:

* *Microsoft Visual Studio 13* или *15*
* *CMake 2.8.7* и выше
* *Vulkan Runtime 1.0.39.1* и выше
* Для видеокарт *NVidia* – драйвер версии 377.06 и выше, для видеокарт *AMD* – самая свежая версия драйвера
* *Microsoft Visual C++ Redistributable 2013* и *2015*

Программные средства, используемые библиотекой *vulkalc*, которые должны быть предоставлены ОС семейства *Linux*:

* *CMake 2.8.7* и выше
* *Vulkan Runtime 1.0.39.1* и выше
* *Clang 3.4* и выше или *gcc 4.6.3* и выше
* Мета-пакет *build-essentials*
* Пакеты *bison*, *libx11-xcb-dev*, *libxkbcommon-dev*, *libmirclient-dev*, *libwayland-dev*, *libxrandr-dev* для *Ubuntu/Debian* и аналоги этих пакетов для других ОС
* Стандартная библиотека *C++*
* Для видеокарт *NVidia* – драйвер версии 377.06 и выше, для видеокарт *AMD* – самая свежая версия драйвера