Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное   
образовательное учреждение высшего образования

«Новгородский государственный университет имени Ярослава Мудрого»

ИНСТИТУТ ЭЛЕКТРОННЫХ И ИНФОРМАЦИОННЫХ СИСТЕМ

КАФЕДРА ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ И СИСТЕМ

Разработка приложения "ДРайвер для джойстика"

Курсовой проект по учебной дисциплине «Системное программирование»

по специальности 09.03.01 Информатика и вычислительная техника  
ИЭИС.КП 3094(ЗУ).01ПЗ

|  |  |
| --- | --- |
|  | Руководитель  / И. Ю. Кулаков  «\_\_\_»\_\_\_\_\_\_\_\_\_\_\_\_2024 г. |
|  | Студент группы 3094 (ЗУ)  / В.В. Бурбах  «\_\_\_»\_\_\_\_\_\_\_\_\_\_\_\_2024 г. |

Содержание

[Введение 3](#_Toc166158403)

[1 Постановка задачи 4](#_Toc166158404)

[1.1 Обоснование необходимости разработки 4](#_Toc166158405)

[1.2 Технико-математическое описание задачи 4](#_Toc166158406)

[1.3 Характеристика бизнес-процесса 4](#_Toc166158407)

[1.4 Анализ предметной области 5](#_Toc166158408)

[1.5 Требования к программе 6](#_Toc166158409)

[1.6 Обоснование проектных решений 6](#_Toc166158410)

[2 Теоретическая часть 9](#_Toc166158411)

[2.1 История научных исследований по выбранной теме 9](#_Toc166158412)

[2.2 Определения и анализ ключевых терминов 10](#_Toc166158413)

[2.3 Актуальные взгляды на выбранную тему 11](#_Toc166158414)

[3 Практическая часть 13](#_Toc166158415)

[3.1 Анализ задачи 13](#_Toc166158416)

[3.2 Описание логической структуры 13](#_Toc166158417)

[3.3 Разработка программы для работы 14](#_Toc166158418)

[3.4 Сравнительная характеристика работы 16](#_Toc166158419)

[Заключение 18](#_Toc166158420)

[Список литературы 19](#_Toc166158421)

[Приложение А 20](#_Toc166158422)

[Приложение Б 23](#_Toc166158423)

[Приложение В 24](#_Toc166158424)

# Введение

Повседневную жизнь сложно представить без драйверов устройств, благодаря им существует возможность управлять персональными компьютерами, подключать различного рода гарнитуру и прочее дополнительное оборудование.

В рамках курсового проектирования была взята тема по разработке программного обеспечения, которое позволяет управлять компьютером при помощи джойстика от игровой консоли Atari2600.

Приложение позволяет подключаться к игровому контроллеру, который, в свою очередь, подключен к компьютеру при помощи микроконтроллера Arduino. Также программа дает возможность самостоятельно устанавливать клавиши или сочетание клавиш на определенные нажатия джойстика.

Цель разработки:

* предоставить возможность использовать игровой контроллер в развлекательных целях на персональном компьютере;
* обеспечить возможность упрощать работу за компьютером, путем задания комбинаций клавиш на нажатие кнопок контроллера.

Объектом разработки является программный комплекс, позволяющий использовать игровой контроллер в различных целях на персональном компьютере.

Задачами курсового проектирования в связи с указанной целью являются:

* разработать программное обеспечение, которое смогло бы улавливать сигналы, поступающие от джойстика и передавать их на компьютер;
* разработать графический интерфейс для возможности подключать игровой контроллер обычному пользователю;
* создать механизм установки нажатия клавиши или комбинаций клавиш в ответ на определенные действия контроллера.

# 1 [Постановка задачи](file:///C:\Users\Student\Downloads\Постановка%20задачи.docx#_Toc452730688)

# 1.1 Обоснование необходимости разработки

В соответствии с заданием на курсовую работу, требуется разработать приложение «Индикатор текущей даты и времени со статистикой», который будет выполнять вывод даты и времени в настоящий момент, а также показывать статистику ПК, такую как ФПС, загруженность ГПУ и ЦПУ.

Данное приложение поможет отслеживать в реальном времени статистику ПК, а также позволит упростить процесс аналитики производительности графического и физического процессора.

# 1.2 Технико-математическое описание задачи

В данном приложение предполагается показ текущий даты и времени со статистикой ПК. Программа должна стать достаточно удобной для использования любым пользователем.

Основной функционал приложения:

* вывод текущей даты;
* вывод текущего времени;
* вывод статистики о ЦПУ;
* вывод статистики о ГПУ;
* вывод статистики о ФПС.

# 1.3 Характеристика бизнес-процесса

Основной характеристикой технологического процесса является вывод текущей даты и времени, а также вывод статистики производительности ПК. Программа выводит информацию после того, как пользователь нажимает определённую клавишу. После этого, в зависимости от выбранной клавиши, выводится на экран информация о текущей дате и времени, а также статистика ФПС, ГПУ, ЦПУ. Программа скрывает информацию также при нажатии определённой клавиши, которую нажимает пользователь программы.

# 1.4 Анализ предметной области

Предметная область данного проекта включает в себя следующие аспекты:

* индикатор текущей даты и времени: важный элемент интерфейса, который позволяет пользователю видеть актуальную информацию о времени на компьютере. Здесь следует рассмотреть способы отображения времени, форматы даты и времени, а также возможность настройки пользователем;
* статистика ПК: анализ и отображение информации о работе компьютера, такой как загрузка процессора, использование памяти, сетевая активность и другие характеристики. Эта информация может быть полезна для пользователей, чтобы контролировать состояние своего устройства;
* интерфейс и визуализация: разработка удобного и интуитивно понятного пользовательского интерфейса для отображения текущей даты и времени, а также статистики ПК. Важно уделить внимание дизайну и удобству использования;
* технические аспекты: выбор языка программирования, технологий для реализации проекта, методы сбора статистики о ПК, обеспечение точности отображения времени и данных;
* функциональные возможности: определение основных функций, которые должны быть реализованы в проекте, такие как обновление времени в реальном времени, отображение графиков статистики, настройки внешнего вида и т.д.

Анализ предметной области поможет определить основные требования к проекту, его цели и функционал, что будет ключевым в дальнейшей разработке курсового проекта.

# 1.5 Требования к программе

Для работы программы с индикатором текущей даты и времени, а также статистикой ПК, минимальные требования к ПК следующие:

* процессор: Intel Core i3 или аналогичный;
* оперативная память: 4 ГБ;
* дисковое пространство: 100 МБ на жестком диске;
* графическая карта: поддерживающая разрешение экрана не менее 1280x720 пикселей;
* операционная система: Windows 7/8/10;
* доступ к интернету для обновления данных в реальном времени;
* USB-порт для подключения периферийных устройств, если необходимо.

Соблюдение этих минимальных требований обеспечит нормальное функционирование программы.

# 1.6 Обоснование проектных решений

1.6.1 Обоснование выбора языков программирования

Оптимальным выбором для реализации курсового проекта по теме «Драйвер для джойстика» являются языки программирования Arduino и Python. Вот несколько обоснований, почему Arduino подходит для этих целей:

* использование микроконтроллера Arduino UNO R3 в рамках курсового проекта говорит о том, что удобнее всего будет применять язык программирования Arduino;
* язык программирования Arduino довольно прост и доступен, а необходимости в написании максимально оптимизированно кода нет, поэтому Arduino отлично подойдет в качестве языка программирования для подключения джойстика к компьютеру;

Для разработки графического приложения был выбран язык программирования Python по следующим причинам:

* Python очень прост в освоении и позволяет быстро создавать небольшие графические приложения, для выполнения слабых с точки зрения вычислительных затрат задач;
* Python обладает большим количество библиотек, написанных на языке программирования Си, что позволяет выполнять некоторые поставленные задачи на уровне «железа».

1.6.2 Инструментальные средства

Для получения сигнала от игрового контроллера использовался микроконтроллер Arduino UNO R3. Arduino UNO — это плата микроконтроллера на базе ATmega328P. Он имеет 14 цифровых входов/выходов (из которых 6 можно использовать в качестве выходов ШИМ), 6 аналоговых входов, керамический резонатор 16 МГц, USB-соединение, разъем питания, разъем ICSP и кнопку сброса.

В качестве сред разработки в рамках курсового проекта используется Visual Studio Code и Arduino IDE. При помощи Visual Studio Code написано графическое приложение на Python. Язык программирования Arduino написан при помощи Arduino IDE.

Для написания графического интерфейса используется библиотека Custom Tkinter, который представляет собой библиотеку на базе стандартной библиотеки графического интерфейса Tkinter, но с расширенными возможностями для дизайна.

Для чтения последовательного порта, на который передает сигнал Arduino UNO, используется библиотека pyserial.

Для симуляции нажатия клавиш используется библиотека pyautogui.

# 2 Теоретическая часть

# 2.1 История научных исследований по выбранной теме

Ранние исследования (до 1990-х годов):

* 1960-е годы: развитие операционных систем, таких как Multics и Unix, включало в себя функции отображения даты и времени;
* 1970-е годы: появление первых персональных компьютеров, таких как Altair 8800, с ограниченными возможностями отображения даты и времени;

1990-е годы: расширение функциональности

* 1990: Microsoft Windows 3.0 представила панель задач, которая включала в себя часы;
* 1995: Windows 95 включила в себя более настраиваемый индикатор даты и времени;
* 1998: Apple Mac OS 8 представила панель меню, которая отображала дату и время;

2000-е годы: интеграция с аппаратным обеспечением;

* 2000: появление материнских плат с встроенными часами реального времени (RTC), которые обеспечивали точное время даже при выключении компьютера;
* 2005: Microsoft Windows XP представила функцию синхронизации времени с интернет-серверами;
* 2010-е годы: расширенная статистика ПК;
* 2010: индикаторы даты и времени начали включать в себя статистику ПК, такую как использование процессора и памяти;
* 2015: Windows 10 представила центр уведомлений, который отображал дату, время и расширенную статистику системы.

Современные исследования (2020-е годы)

* 2020: исследования сосредоточены на улучшении точности и надежности индикаторов даты и времени;
* 2021: разработка новых алгоритмов для синхронизации времени с использованием облачных сервисов и распределенных сетей;
* 2022: индикаторы даты и времени интегрируются с системами умного дома и Интернетом вещей (IoT), обеспечивая синхронизированное время на всех подключенных устройствах.

Текущие тенденции. Исследования в области индикаторов даты и времени со статистикой ПК продолжаются и сосредоточены на следующих направлениях:

* улучшение точности и надежности;Интеграция с новыми технологиями, такими как IoT и искусственный интеллект (ИИ);
* расширение возможностей настройки и персонализации;
* разработка новых методов отображения даты и времени, улучшающих удобство использования и доступность.

# 2.2 Определения и анализ ключевых терминов

Индикатор текущей даты и времени со статистикой ПК — это удобный инструмент, который обычно отображается на рабочем столе компьютера и предоставляет информацию о текущей дате, времени и основных характеристиках работы компьютера.

Ключевые термины:

* индикатор текущей даты и времени – это компонент интерфейса пользователя, который показывает текущие дату и время в удобном для восприятия формате;
* статистика ПК – совокупность данных о работе компьютера, включая информацию о загрузке процессора, использовании памяти, температуре компонентов и другие характеристики;
* рабочий стол – основная часть графического интерфейса операционной системы, на которой отображаются ярлыки программ, фоновые изображения и виджеты, включая индикатор текущей даты и времени;
* характеристики работы компьютера – это параметры и показатели, описывающие состояние и производительность компьютера, такие как загрузка процессора, использование оперативной памяти, температура ЦП и т.д.

Анализ ключевых терминов:

* индикатор текущей даты и времени со статистикой ПК предоставляет пользователю не только информацию о времени и дате, но и общую картину о работе компьютера. Этот инструмент может быть полезен для отслеживания производительности системы, выявления возможных проблем (например, перегрева) и удобного доступа к важным данным о компьютере;
* использование индикатора текущей даты и времени со статистикой ПК позволяет пользователям быть в курсе текущей ситуации, оперативно реагировать на изменения в работе компьютера и эффективно управлять ресурсами системы.

# 2.3 Актуальные взгляды на выбранную тему

Интерес к индикаторам текущей даты и времени со статистикой ПК в последнее время постоянно растет, так как эти инструменты предоставляют пользователям более подробную информацию о работе и состоянии их компьютеров. Вот несколько актуальных взглядов на эту тему:

* мониторинг производительности: индикаторы текущей даты и времени со статистикой ПК помогают пользователям отслеживать загрузку процессора, использование оперативной памяти, температуру и другие параметры, что позволяет быстрее выявлять проблемы и оптимизировать работу системы;
* удобство использования: имея всю необходимую информацию на рабочем столе в удобном виде, пользователи могут легче контролировать процессы на своем компьютере и принимать соответствующие меры при необходимости;
* персонализация: многие пользователи ценят возможность настройки внешнего вида и функциональности индикаторов текущей даты и времени со статистикой ПК в соответствии с их предпочтениями и потребностями;
* информационная ценность: предоставление не только данных о времени, но и о работе компьютера делает эти индикаторы более полезными и информативными для пользователей.

В целом, индикаторы текущей даты и времени со статистикой ПК играют важную роль в повседневной жизни пользователей, помогая им эффективнее управлять своими компьютерами и поддерживать их в исправном состоянии.

# 3 Практическая часть

# 3.1 Анализ задачи

Для курсового проекта по разработке графического интерфейса программы с отображением текущей даты и времени, а также статистики ПК, необходимо выполнить следующие этапы:

* проанализировать требования к обновлению данных в реальном времени;
* разработать дизайн интерфейса с учетом всех требований;
* определить элементы управления, необходимые для отображения информации и взаимодействия с программой.

На рисунке Б.1 представлено окно работы приложения со статикой ПК. Также на рисунке Б.2 представлено окно работы приложения с отображением текущей даты и времени.

# 3.2 Описание логической структуры

3.2.1 Алгоритм программы

Рассмотрим алгоритмы программы.

Алгоритмы программы для индикации текущей даты и времени, а также статистики ПК начинаются с запуска программы. При запуске приложения появляется окно, где выводится дата и время на текущей момент со статистикой ПК. Пользователь может свернуть окно приложения и открыть, используя клавишу ESC. Таким же образом, можно скрыть и вернуть статистику ПК, используя клавишу «Пробел». В приложении также предусмотрен функционал для закрытия приложения, по нажатию на Крестик.

Сложная логика приложения заключается в использовании Mixin’ов для сбора статистики. По мимо прочего используются Executor сервисы, позволяющие эффективно работать с потоками, чтобы приложение не загружало поток отрисовки графики окна и не тормозило работу систем прерывания C++. Для измерения ФПС используется обратное отношение времени обработки к максимальному времени сбора и вывода информации.

3.2.2 Составные части программы и связи между ними

Программа содержит несколько модулей, каждый из которых выполняет определенные функции:

1. Модуль KeyLogger отвечает за обработку нажатий клавиш. Если пользователь нажимает пробел, то меняется флажок видимости производительности. Если же нажата клавиша Esc, то меняется видимость главного окна программы.

2. Модуль, в котором происходит обновление данных о производительности (FPS, памяти GPU и т.д.) и их отображение на экране. Если флажок видимости производительности включен, то данные обновляются и отображаются на экране. Если же флажок выключен, то элементы, отображающие информацию о производительности (FPS, память GPU и т.д.), скрываются.

3. Модуль отслеживает время выполнения операций на GPU и обновляет информацию на экране с соответствующей задержкой.

Взаимодействие между этими модулями происходит следующим образом: модуль KeyLogger отслеживает нажатия клавиш и в зависимости от них изменяет флажок видимости производительности и видимость главного окна. Модуль, отвечающий за обновление данных о производительности, постоянно проверяет состояние флажка видимости производительности и в зависимости от него либо обновляет информацию и отображает ее на экране, либо скрывает элементы. Таким образом, модули программы взаимодействуют между собой для корректного отображения информации и обработки пользовательских действий.

# 3.3 Разработка программы для работы

3.3.1 Входные и выходные данные

Входными данными является информация с компьютера, такая как: текущая дата и время, а также статистика ПК.

Промежуточными данные отсутствуют.

Выходными данными является статистика, которая визуализируется при помощи оконного приложения

3.3.2 Используемые методы

Метод nativeKeyReleased: данный метод отвечает за изменение состояний флажка enable и видимости окна jFrame в зависимости от нажатых клавиш пользователем.

Метод windowClosing: данный метод при закрытии окна освобождает ресурсы, связанные с окном, и завершает работу программы.

3.3.3 Тестирование

Данные тестирования приведены в таблице 1.

Таблица 1 – Тестирование приложения

|  |  |  |
| --- | --- | --- |
| Название тест-кейса | Выполнение действий | Ожидаемый результат |
|  |  |  |
| Нажатие клавиши пробел | Пользователь нажимает клавишу пробел | Скрывается/отображается статистика ПК |
| Нажатие клавиши «ESC» | Пользователь нажимает клавишу «ESC» | Скрывается/отображается окно с выводом текущей даты и времени |
| Нажатие других клавиш, кроме пробел и «ESC» | Пользовать нажимает другие клавиши, кроме пробел и «ESC» | Ничего не происходит |

3.3.4 Эксплуатационные документы – руководство пользователя

3.3.4.1 Назначение программы

Целевой аудиторией приложения будут являться пользователи, которым необходимо отслеживать в реальном времени статистику ПК. Разработанная программа позволит им ускорить и упростить процесс аналитики производительности графического и физического процессора.

3.3.4.2 Условия выполнения

Для работы программы с индикатором текущей даты и времени, а также статистикой ПК, минимальные требования к ПК следующие:

* процессор: Intel Core i3 или аналогичный;
* оперативная память: 4 ГБ;
* дисковое пространство: 100 МБ на жестком диске;
* графическая карта: поддерживающая разрешение экрана не менее 1280x720 пикселей;
* операционная система: Windows 7/8/10;
* доступ к интернету для обновления данных в реальном времени;
* USB-порт для подключения периферийных устройств, если необходимо.

Соблюдение этих минимальных требований обеспечит нормальное функционирование программы.

3.3.4.3 Выполнение программы

Для запуска приложения необходимо открыть исполняемый файл. После запуска приложения появляется окно c текущей датой и временем. Для отображения статистики ПК, пользователю необходимо нажать клавишу пробел, после нажатия этой клавиши к текущей дате и времени появится статистика ПК (ГПУ, ЦПУ, ФПС). Для того, чтобы скрыть статистку ПК, пользователю необходимо повторить действие с нажатой клавишей пробел, после этого статистика скрывается и остается текущая дата и время. Чтобы скрыть полностью окно индикатора, пользователю необходимо нажать клавишу «ESC», аналогичным образом пользовать может снова отобразить окно индикатора. Для закрытия приложения, пользователю необходимо нажать на крестик в левом верхнем углу приложения

# 3.4 Сравнительная характеристика работы

Аналоги:

* ClockTuner: Отображает текущее время, дату, статистику системы и многое другое в виде настраиваемого виджета;
* Rainmeter: Мощный инструмент для настройки рабочего стола, который позволяет создавать собственные скины, включая индикаторы даты и времени;
* Sidebar Diagnostics: Отображает подробную информацию о системе, включая дату и время, в виде боковой панели;
* Taskbar Time and Date: Простое приложение, которое добавляет индикатор даты и времени на панель задач;
* Date and Time Clock Widget: Виджет рабочего стола, который отображает текущее время и дату в различных стилях.

Общие недостатки:

* некоторые приложения могут быть несовместимы с определенными версиями операционной системы;
* могут отображать неточную информацию о дате и времени, если системные часы не настроены правильно;
* могут потреблять системные ресурсы, что может привести к замедлению работы компьютера.

# Заключение

В результате курсового проектирования были улучшены знания в области работы со специализированными программами, также повышен уровень знаний в области программирования. Проектирование прошло успешно, все функции были реализованы, а задача полностью выполнена.

На этапе формирования требований к программному продукту были сформулированы цели и задачи курсового проекта, которые в ходе работы были успешно реализованы с помощью программы IntelliJ IDEA. В приложении к пояснительной записке курсового проекта помещены разработанные блок-схемы алгоритмов, выполненные в соответствии со стандартами.

Исходя из всего вышеперечисленного можно сделать вывод, что курсовой проект выполнен в полном объеме.

# Список литературы

1. ГОСТ Р 2.105–2019 Единая система конструкторской документации
2. Шилдт Г. Программирование на C++ для начинающих, 2024 – 156 с
3. Страуструпп Б. Язык программирования C++, 2022 – 98 с
4. https://sharpened.ucoz.ru/lib/polnyj\_spravochnik\_po\_c-gerbert\_shildt-2006.pdf, [Дата обращения 04.04.2024]
5. <https://stackoverflow.com>, [Дата обращения 16.04.2024]

# Приложение А

(обязательно)

Листинг приложения:

import com.github.kwhat.jnativehook.GlobalScreen;

import com.github.kwhat.jnativehook.NativeHookException;

import com.github.kwhat.jnativehook.keyboard.NativeKeyEvent;

import com.github.kwhat.jnativehook.keyboard.NativeKeyListener;

import javax.swing.\*;

import java.awt.\*;

import java.awt.event.WindowAdapter;

import java.awt.event.WindowEvent;

import java.lang.management.ManagementFactory;

import java.util.Date;

import java.util.concurrent.Executors;

public class Main {

public static JLabel fps;

public static JLabel loaded;

public static JLabel dateTime;

public static boolean enable;

public static JFrame jFrame;

private static JLabel gpu;

public static void main(String[] args) {

var osBean = ManagementFactory.getPlatformMXBean(com.sun.management.OperatingSystemMXBean.class);

GlobalScreen.addNativeKeyListener(new KeyLogger());

jFrame = new JFrame();

jFrame.setVisible(true);

jFrame.setResizable(false);

jFrame.setSize(380, 150);

jFrame.setName("Статистика");

jFrame.setLayout(new GridLayout(2, 3));

jFrame.addWindowListener(new WindowAdapter() {

@Override

public void windowClosing(WindowEvent e) {

e.getWindow().dispose();

System.exit(0);

}

});

dateTime = new JLabel();

loaded = new JLabel();

fps = new JLabel();

gpu = new JLabel();

jFrame.add(dateTime);

jFrame.add(loaded);

jFrame.add(fps);

jFrame.add(gpu);

try {

GlobalScreen.registerNativeHook();

} catch (NativeHookException ex) {

System.err.println("There was a problem registering the native hook.");

System.err.println(ex.getMessage());

System.exit(1);

}

enable = true;

Executors.newSingleThreadExecutor().submit(() -> {

while (true) {

long startTime = System.currentTimeMillis();

try {

Thread.sleep(10);

} catch (InterruptedException e) {

throw new RuntimeException(e);

}

dateTime.setText(new Date().toString());

if (enable) {

fps.setVisible(true);

loaded.setVisible(true);

gpu.setVisible(true);

String text = "CPU: " + (int) (osBean.getCpuLoad() \* 100) + "%";

long fpsValue = 1000 / (System.currentTimeMillis() - startTime);

loaded.setText(text);

fps.setText("FPS: " + fpsValue);

gpu.setText("GPU: " + (System.currentTimeMillis() - startTime) / 10D + "%");

} else {

fps.setVisible(false);

loaded.setVisible(false);

gpu.setVisible(false);

}

}

});

}

public static final class KeyLogger implements NativeKeyListener {

public void nativeKeyReleased(NativeKeyEvent e) {

if (e.getKeyCode() == NativeKeyEvent.VC\_SPACE) {

enable = !enable;

}

if (e.getKeyCode() == NativeKeyEvent.VC\_ESCAPE) {

jFrame.setVisible(!jFrame.isVisible());

}

}

}

}

# Приложение Б

(обязательно)

Изображение выглядит как текст, снимок экрана, Шрифт, линия

Автоматически созданное описание

Рисунок Б.1 – Окно работы приложения с выводом статистики ПК

Изображение выглядит как текст, снимок экрана, Шрифт

Автоматически созданное описание

Рисунок Б.1 – Окно работы в приложении с выводом текущей даты и времени

# Приложение В

(обязательное)

Изображение выглядит как текст, диаграмма, Параллельный, линия

Автоматически созданное описание

Рисунок В.1 – Блок-схема «Поток обработки»