Министерство науки и высшего образования Российской Федерации

Федеральное государственное автономное образовательное учреждение

высшего образования

«Уральский федеральный университет

имени первого Президента России Б.Н. Ельцина»

Институт новых материалов и технологий

Кафедра «Теплофизика и информатика в металлургии»

ДОПУСТИТЬ К ЗАЩИТЕ ПЕРЕД ГЭК

Зав. кафедрой \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

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

(подпись) (Ф.И.О.)

«\_\_\_\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_2019 г.

**ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА**

РАЗРАБОТКА ПРОГРАММНО-АППАРАТНОГО КОМПЛЕКСА НА БАЗЕ ИЗМЕРИТЕЛЕЙ-РЕГУЛЯТОРОВ ОВЕН

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

ВКР 09.03.02 000 000 906 ПЗ

Руководитель

доцент, к.т.н. \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Гольцев В.А.

Нормоконтролер

доцент, к.т.н. \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Киселев Е.В.

Студент группы

НМТ-453907 \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Сахаров А.Ю.

РЕФЕРАТ

Выпускная квалификационная работа содержит: 47 страниц, 35 рисунков, 2 таблицы, 13 источников.

Ключевые слова: разработка, сушка, приложение, Windows Forms, режимы сушки, визуализация, приложение для образования.

Объект исследования – лабораторная установка для исследования режимов и способов сушки влажных материалов.

Целью работы является разработка программного обеспечения на платформе Windows Forms для работы с лабораторной установкой.

В процессе работы осуществлена установка связи весов лабораторной установки и компьютера.

Результатом работы является конечная версия программного продукта, которая предназначена для мониторинга и расчёта данных, сведений о работе весов в установке в реальном времени. Разработаны методические указания по взаимодействию с полученным лабораторным стендом.

Областью применения данного продукта является лаборатория кафедры «Теплофизика и информатика в металлургии».

ОГЛАВЛЕНИЕ

[ВВЕДЕНИЕ 4](#_Toc104374266)

[1 ХАРАКТЕРИСТИКА РАЗРАБАТЫВАЕМОГО ЛАБОРАТОРНОГО КОМПЛЕКСА 5](#_Toc104374267)

[1.1 Цель работы. Функциональные задачи, реализуемые на лабораторном стенде 5](#_Toc104374268)

[1.2 Состав и структура лабораторного стенда 5](#_Toc104374269)

[2 ОПИСАНИЕ ТЕХНИЧЕСКИХ СРЕДСТВ 7](#_Toc104374270)

[2.1 Весы Acom JW-1-200 7](#_Toc104374271)

[2.2 Многофункциональная микроволновая печь CE1000R 9](#_Toc104374272)

[2.3 RS-232 11](#_Toc104374273)

[3 ОПИСАНИЕ ПРОГРАММНЫХ СРЕДСТВ 13](#_Toc104374274)

[3.1 Выбор среды разработки. Назначение, функции, характеристики 13](#_Toc104374275)

[4 РАЗРАБОТКА ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ ДЛЯ ЛАБОРАТОРНОГО СТЕНДА 19](#_Toc104374276)

[4.1 Настройка соединения RS-232 19](#_Toc104374277)

[4.2 Разработка приложения Windows Forms 22](#_Toc104374278)

[4.3 Разработка методического обеспечения 26](#_Toc104374279)

[ЗАКЛЮЧЕНИЕ 36](#_Toc104374280)

[СПИСОК ЛИТЕРАТУРЫ 37](#_Toc104374281)

ВВЕДЕНИЕ

В современном обществе уж не один человек, испробовавший блага цивилизации, не может представить свою жизнь без применения компьютерной техники. Ее использование протекает в каждой сфере человеческой жизнедеятельности: производстве, торговле, обучении, развлечении и общении людей, их научной и культурной деятельности. Все это вследствие возможности выбора компьютерной техники ради решения любой, даже самой непростой задачи.

Программное обеспечение – необходимая доля компьютерной концепции. Оно является логическим продолжением технических средств каждого компьютера. Область использования определённого компьютера обусловливается разработанным для него программным обеспечением. Сам по себе компьютер не обладает знаниями ни в одной области применения. Все эти знания сосредоточены в выполняемых на компьютерах программах, которые располагают наборами определенных многофункциональных возможностей и предназначены для исполнения конкретных, в большинстве случаев, узкоспециализированных функций, таких, например, как создание и обработка графических изображений либо звуковых файлов. Программное обеспечение в настоящее время составляет сотни тысяч программ, которые предназначены для обработки самой всевозможной информации с самыми разными целями.

Применение программного обеспечения для работы с приборами существенно упрощает операцию выполнения той или иной задачи. Данная работа предназначена для того, чтобы разработать автоматизированную информационную систему сбора и визуализации данных при исследовании режимов сушки влажных материалов, которое в дальнейшем будет использоваться студентами и преподавателями при выполнении лабораторной работы «Исследование режимов и способов сушки влажных материалов».

# ХАРАКТЕРИСТИКА РАЗРАБАТЫВАЕМОГО ЛАБОРАТОРНОГО КОМПЛЕКСА

## Цель работы. Функциональные задачи, реализуемые на лабораторном стенде

Целью данной работы является реализация приложения «Исследование режимов и способов сушки влажных материалов» на платформе Windows Forms, которое предназначено для мониторинга и расчёта данных, сведений о работе весов в установке в реальном времени. Для достижения поставленных целей выделим функциональные задачи:

* корректное подключение к объекту;
* вывод информации об объекте в реальном времени;
* управление приложением.

Автоматизированная информационная система сбора и визуализации для исследования влажных материалов должна быть реализована на платформе Windows Forms. Приложению необходимо позволять студенту выполняющего лабораторную работу у лабораторного стенда с помощью пользовательских форм производить мониторинг данных с весов и получать готовый расчёт исходя из полученных данных в реальном времени, представить его в удобной форме и в общем повысить культуру управления данными с весов, за счет улучшений условий.

## Состав и структура лабораторного стенда

Лабораторный стенд представлен на базе многофункциональной микроволновой печи CE1000R и весов Acom JW-1-200, которые в свою очередь, подключены к компьютеру на базе операционной системы Windows при помощи интерфейсов RS-282 to USB. Через отверстие в верхней стенке печи проходит термостойкая нить, на которой подвешена чаша с высушиваемым материалом. Верхний конец нити крепится к нижнему подвесу электронных цифровых весов. Лабораторный стенд изображен на рисунке 1.1.

**Изображение выглядит как текст, внутренний, электроника, компьютер

Автоматически созданное описание**

Рисунок 1.1 – Лабораторная установка

# ОПИСАНИЕ ТЕХНИЧЕСКИХ СРЕДСТВ

В состав лабораторного стенда входит оборудование:

* весы Acom JW-1;
* микроволновая печь CE1000R;
* RS-232 – стандарт физического уровня для асинхронного интерфейса.

## Весы Acom JW-1-200

Лабораторные весы Acom JW-1-200 с разрешением до 1/20000. Имеют контрастный жидкокристаллический дисплей и клавиатура на русском языке, несколько единиц веса: грамм, карат, унция, тройская унция, гран, фунт, пеннивейт. С возможностью подключения к персональному компьютеру через интерфейс RS-232C и гидростатическим взвешиванием "под весами" с помощью крюка. Автоматическая и ручная установка ноля, счетный и процентный режимы взвешивания, учет веса тары. Питание от сети через адаптер 9V/300mA, а также возможность питания от 6 батарей типа АА (опция). Внешний вид весов представлен на рисунке Рисунок 2.1.



Рисунок 2.1 – Лабораторные весы Acom JW-1-200

Весы имеют технические характеристики, которые предоставлены в таблице 2.1.

Таблица 2.1 – Технические характеристики весов Acom JW-1-200

|  |  |
| --- | --- |
| НПВ | 200 г |
| Дискретность (d=e) | 0.01 г |
| НмПВ | 0,2 г |
| Число поверочных делений (n) | 20000 |
| Пределы допускаемой погрешности при первичной (периодической поверке) | от НмПВ до 5000е вкл.: ±0.5е (1е)  от 5000е НмПВ до 2000е вкл.: ±1е (2е)  свыше 20000е: ±1.5е (3е) |
| Диапазон выборки веса тары | До 100% НПВ |
| Дисплей | Жидкокристаллический, 6 знаков |
| Диаметр платформы | 123 мм |
| Габариты | 280 мм х 250 мм х 65 мм |
| Вес | 0.95 кг |
| Питание | Адаптер 9В/300мА или батареи 6х1,5В (опция) |
| Уставка | 10°C ~ 30°C |
| Влажность | 35% ~ 85% (без конденсата) |
| Потребляемая мощность | ~ 1 ВА |

Также, следует отметить, что лабораторные весы Acom JW-1-200 имеют официальное свидетельство об утверждении типа средств измерений, которое представлено на рисунке 1.2.

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 1.2 – Свидетельство об утверждении типа средств измерений

## Многофункциональная микроволновая печь CE1000R

Прибор предназначен для нагревания экспериментальных образцов. Внешний вид многофункциональной микроволновой печи CE 1000R представлен на рисунке Рисунок 2.2.



Рисунок 2.2 – Многофункциональная микроволновая печь CE1000R

Технические характеристики многофункциональной микроволновой печи CE1000R представлены в таблице 2.2.

Таблица 2.1 – Технические характеристики многофункциональной микроволновой печи CE1000R

|  |  |
| --- | --- |
| Тип | С конвекцией |
| Цвет | Серебристый (серый) |
| Объем, л | 28 |
| Способ установки | Отдельностоящая |
| Дополнительно | Быстрый старт  Удаление запахов |
| Диапазон выборки веса тары | До 100% НПВ |
| Ширина, см | 51.7 |
| Диаметр поворотного стола | 31.8000 см |
| Количество уровней мощности | 6 уровней:  – высокий: 900 Вт;  – выше среднего: 600 Вт;  - средний: 450 Вт;  – ниже среднего: 300 Вт;  – размораживание: 180 Вт;  – низкий: 100 Вт; |
| Комплектация | 1250.0000 Вт |
| Мощность гриля | Адаптер 9В/300мА или батареи 6х1,5В (опция) |
| Мощность конвекции | 2050.0000 Вт |
| Мощность СВЧ | 900.0000 Вт |
| Тип гриля | Кварцевый |
| Габариты (ВхШхГ), см | 31х51,7x51 |

## RS-232

Данное устройство предназначено для передачи данных с весов до ПК, на котором будет производиться считывание данных. Внешний вид интерфейса связи представлен на рисунке Рисунок 2.3.



Рисунок 2.3 – RS-232 – стандарт физического уровня для асинхронного интерфейса

RS-232 (англ. Recommended Standard 232, другое название EIA232) – стандарт физического уровня для асинхронного интерфейса (UART). Устройство, поддерживающее этот стандарт, широко известно как последовательный порт персональных компьютеров. RS-232 обеспечивает передачу данных и некоторых специальных сигналов между терминалом (англ. Data Terminal Equipment, DTE) и коммуникационным устройством (англ. Data Communications Equipment, DCE) на расстояние до 15 метров на максимальной скорости (115200 бод). Так как этот интерфейс известен не только простотой программирования, но и неприхотливостью, в реальных условиях это расстояние увеличивается во много раз с примерно пропорциональным снижением скорости. В настоящее время чаще всего используется в промышленном и узкоспециальном оборудовании, встраиваемых устройствах. На портативных компьютерах (ноутбуках, нетбуках, КПК и т. п.) широкого применения RS-232 не нашёл, однако материнские платы стационарных персональных компьютеров до недавнего времени ещё содержали RS-232 – либо в виде разъёма на задней панели, либо в виде колодки для подключения шлейфа на плате. Также возможно использование переходников-преобразователей. Кроме того, RS-232 имеется на некоторых телевизорах и ресиверах, в частности, спутниковых, где предназначен в том числе для обновления встроенного ПО через компьютер.

# ОПИСАНИЕ ПРОГРАММНЫХ СРЕДСТВ

## Выбор среды разработки. Назначение, функции, характеристики

Большинство средств разработки обладают схожим набором функциональных возможностей, позволяющих компилировать написанный код. Требования к среде разработки:

* графический интерфейс, предназначенный для упрощения построения и отображения процесса разработки. Используются графические элементы управления, такие как кнопки, индикаторы, различные меню;
* драйверы для оборудования и устройств;
* поддержка языков программирования высокого уровня;
* возможность установки различных расширений разработки.

Рассмотрим наиболее часто используемые средства разработки:

### Visual Studio

Visual Studio – линейка продуктов компании Microsoft, включающих интегрированную среду разработки программного обеспечения и ряд других инструментов. Данные продукты позволяют разрабатывать как консольные приложения, так и игры и приложения с графическим интерфейсом, в том числе с поддержкой технологии Windows Forms, UWP а также веб-сайты, веб-приложения, веб-службы как в родном, так и в управляемом кодах для всех платформ, поддерживаемых Windows, Windows Mobile, Windows CE, .NET Framework, .NET Core, .NET, MAUI, Xbox, Windows Phone .NET Compact Framework и Silverlight.

Интегрированная среда разработки (Integrated Development Environment - IDE) Visual Studio предлагает ряд высокоуровневых функциональных возможностей, которые выходят за рамки базового управления кодом.

Ниже перечислены основные преимущества IDE-среды Visual Studio.

Встроенный Web-сервер. Для обслуживания Web-приложения ASP.NET необходим Web-сервер, который будет ожидать Web-запросы и обрабатывать соответствующие страницы. Наличие в Visual Studio интегрированного Web-сервера позволяет запускать Web-сайт прямо из среды проектирования, а также повышает безопасность, исключая вероятность получения доступа к тестовому Web-сайту с какого-нибудь внешнего компьютера, поскольку тестовый сервер может принимать соединения только с локального компьютера.

Поддержка множества языков при разработке. Visual Studio позволяет писать код на своем языке или любых других предпочитаемых языках, используя все время один и тот же интерфейс (IDE). Более того, Visual Studio также еще позволяет создавать Web-страницы на разных языках, но помещать их все в одно и то же Web-приложение. Единственным ограничением является то, что в каждой Web-странице можно использовать только какой-то один язык (очевидно, что в противном случае проблем при компиляции было бы просто не избежать).

Меньше кода для написания. Для создания большинства приложений требуется приличное количество стандартного стереотипного кода, и Web-страницы ASP. NET тому не исключение. Например, добавление Web-элемента управления, присоединение обработчиков событий и корректировка форматирования требует установки в разметке страницы ряда деталей. В Visual Studio такие детали устанавливаются автоматически.

Интуитивный стиль кодирования. По умолчанию Visual Studio форматирует код по мере его ввода, автоматически вставляя необходимые отступы и применяя цветовое кодирование для выделения элементов типа комментариев. Такие незначительные отличия делают код более удобным для чтения и менее подверженным ошибкам. Применяемые Visual Studio автоматически параметры форматирования можно даже настраивать, что очень удобно в случаях, когда разработчик предпочитает другой стиль размещения скобок (например, стиль K&R, при котором открывающая скобка размещается на той же строке, что и объявление, которому она предшествует).

Более высокая скорость разработки. Многие из функциональных возможностей Visual Studio направлены на то, чтобы помогать разработчику делать свою работу как можно быстрее. Удобные функции, вроде функции IntelliSense (которая умеет перехватывать ошибки и предлагать правильные варианты), функции поиска и замены (которая позволяет отыскивать ключевые слова как в одном файле, так и во всем проекте) и функции автоматического добавления и удаления комментариев (которая может временно скрывать блоки кода), позволяют разработчику работать быстро и эффективно.

Возможности отладки. Предлагаемые в Visual Studio инструменты отладки являются наилучшим средством для отслеживания загадочных ошибок и диагностирования странного поведения. Разработчик может выполнять свой код по строке за раз, устанавливать интеллектуальные точки прерывания, при желании сохраняя их для использования в будущем, и в любое время просматривать текущую информацию из памяти.

Visual Studio также имеет и множество других функций: возможность управления проектом; встроенная функция управления исходным кодом; возможность рефакторизации кода; мощная модель расширяемости. Более того, в случае использования Visual Studio 2008 Team System разработчик получает расширенные возможности для модульного тестирования, совместной работы и управления версиями кода (что значительно больше того, что предлагается в более простых инструментах вроде Visual SourceSafe).

В качестве недостатка можно отметить невозможность отладчика (Microsoft Visual Studio Debugger) отслеживать в коде режима ядра. Отладка в Windows в режиме ядра в общем случае выполняется при использовании WinDbg, KD или SoftICE.

### PuTTY

PuTTY – свободно распространяемый клиент для различных протоколов удалённого доступа, включая SSH, Telnet, rlogin. Также имеется возможность работы через последовательный порт. Данное программное обеспечение позволяет подключиться и управлять удалённым узлом (например, сервером). В PuTTY реализована только клиентская сторона соединения – сторона отображения, в то время как сама работа протекает на стороне сервера

PuTTY входит в репозитории практически всех популярных дистрибутивов Linux (в т.ч. Ubuntu, Debian, ALT Linux).

Исходный код PuTTY полностью разработан на C. PuTTY не зависит от DLL, других приложений, пакетов обновлений ОС. Пакет состоит только из исполняемых файлов, которые могут быть установлены в любом месте. PuTTY и большинство утилит запускаются только в одном потоке ОС. Программа является свободно распространяемым приложением с открытым исходным кодом и выпускается под Open Source лицензией MIT.

Некоторые возможности программы:

* Сохранения списка и параметров подключений для повторного использования;
* Работа с ключами и версиями протокола SSH;
* Клиенты SCP и SFTP (соответственно программы pscp и psftp);
* Возможность перенаправления портов через SSH, включая передачу X11;
* Поддержка большей части управляющих последовательностей xterm, VT-102, а также значительная эмуляция терминала ECMA-48;
* Поддержка IPv6;
* Поддержка 3DES, AES, Arcfour, Blowfish, DES;
* Поддержка аутентификации с открытым ключом, в том числе и без ввода пароля;
* Поддержка работы через последовательный порт (начиная с версии 0.59);
* Возможность работы через прокси-сервер;
* Поддержка метода zlib@openssh.com (отсроченное сжатие данных до окончания процесса аутентификации).

Применение программы:

* удаленное администрирование Linux;
* подключение к виртуальным серверам по протоколу SSH;
* настройка сетевых маршрутизаторов через последовательный порт;
* соединение с удаленными Telnet-терминалами и пр.

### GitHub

GitHub – крупнейший веб-сервис для хостинга IT-проектов и их совместной разработки. Создатели сайта называют GitHub «социальной сетью для разработчиков».

Возможности:

* Кроме размещения кода, участники могут общаться, комментировать правки друг друга, а также следить за новостями знакомых;
* С помощью широких возможностей Git программисты могут объединять свои репозитории — GitHub предлагает удобный интерфейс для этого и может отображать вклад каждого участника в виде дерева;
* Для проектов есть личные страницы, небольшие Вики и система отслеживания ошибок;
* Прямо на сайте можно просмотреть файлы проектов с подсветкой синтаксиса для большинства языков программирования;
* Можно создавать приватные репозитории, которые будут видны только вам и выбранным вами людям;
* Есть возможность прямого добавления новых файлов в свой репозиторий через веб-интерфейс сервиса;
* Код проектов можно не только скопировать через Git, но и скачать в виде обычных архивов с сайта;
* Кроме Git, сервис поддерживает получение и редактирование кода через SVN и Mercurial;
* На сайте есть pastebin-сервис gist.github.com для быстрой публикации фрагментов кода.

Для упрощения работы с хостингом GitHub будет использоваться приложение GitHub Desktop. Данное приложение позволяет воспользоваться функционалом хостинга через удобный графический интерфейс, который виден на рисунке 3.1.

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 3.1 – Интерфейс приложения GitHub Desktop

# РАЗРАБОТКА ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ ДЛЯ ЛАБОРАТОРНОГО СТЕНДА

## Настройка соединения RS-232

Для взаимодействия с прибором посредством интерфейса RS-232 необходимо осуществить его настройку.

После подключения кабеля в компьютер необходимо зайти в «Диспетчер устройств» Windows, просмотреть активные COM-подключения (рисунок 4.1), затем выбрав нужный выполнить установку актуального драйвера (рисунок 4.2).

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 4.1 – Диспетчер задач Windows

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 4.2 – Настройка драйвера для COM-порта

Далее, при активном интернет-подключении система сама установит актуальную версию драйвера.

Если данная операция не помогла настроить активное соединение, то, в частности, случаев дело еще может быть в параметрах соединения. Тогда необходимо зайти в настройки выбранного COM-порта и выставить стандартные значения для соединения как показано на рисунке 4.3 или иные.

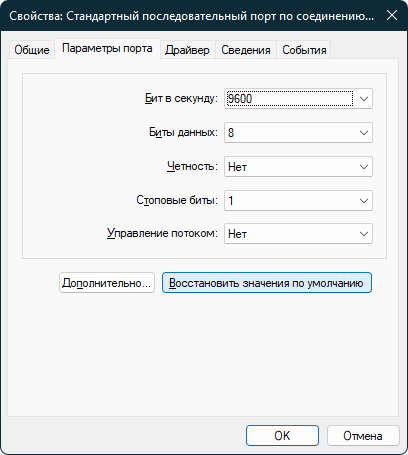


Рисунок 4.3 – Стандартные параметры для COM-порта

Далее, для проверки настроенного соединения необходимо произвести тест подключения в программе PuTTY. Выставленные настройки для теста соединения в PuTTY показаны на рисунке 4.4.

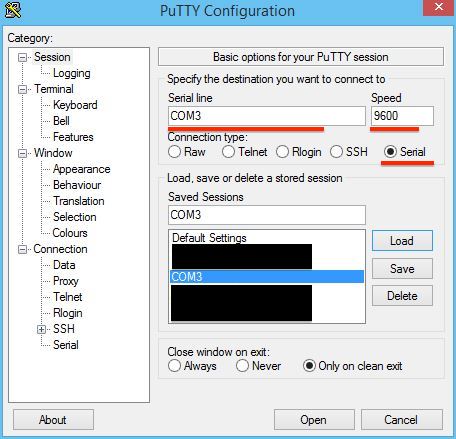


Рисунок 4.4 – Интерфейс программы PuTTY для теста соединения с COM-портом

Далее, при нажатии кнопки Open если не возникает никаких ошибок, и программа выдала консоль с различными потоками данных, то соединение с COM-портом успешно выполнено.

## Разработка приложения Windows Forms

Данное приложение на платформе Windows Forms является интерфейсом между пользователем и устройством, передавая данные между ними через интерфейс RS-232. Пользователь взаимодействует с графическими элементами управления на форме.

Windows Forms — это платформа пользовательского интерфейса для создания классических приложений Windows. Она обеспечивает один из самых эффективных способов создания классических приложений с помощью визуального конструктора в Visual Studio. Такие функции, как размещение визуальных элементов управления путем перетаскивания, упрощают создание классических приложений.

В Windows Forms можно разрабатывать графически сложные приложения, которые просто развертывать, обновлять, и с которыми удобно работать как в автономном режиме, так и в сети. Приложения Windows Forms могут получать доступ к локальному оборудованию и файловой системе компьютера, на котором работает приложение. Объект предназначен для описания логического слоя системы. Для объекта могут быть созданы различные окна, или графические представления на мнемосхеме, такие как элементы ввода/вывода.

Приложение отображает в себе основные элементы сбора и визуализации данных: устройства и другие внешние источники данных. В узел системы добавляются рабочие станции, описывающие ПК, к которым подключены приборы. Важным параметром системы являются шкалы. Шкалы позволяют задавать диапазоны изменения аналоговых, интервальных, дискретных и перечислимых параметров, устанавливать для них аварийные границы. Параметры доступные для чтения и записи представлены как значения, только для чтения – как команды.

Обобщённый алгоритм реализации приложения на платформе Windows Forms:

* создание пустого проекта;
* настройка соединения с COM-портом;
* разработка объектной структуры пользовательского интерфейса;
* тестирование системы на стенде;
* устранение ошибок.

4.2.1 Создание проекта

Первым шагом выполнения стало создание проекта Windows Forms на платформе .Net Framework в среде разработки Visual Studio (рисунок 4.5).

Изображение выглядит как текст, монитор, снимок экрана, экран

Автоматически созданное описание

Рисунок 4.5 – Создание проекта Windows Forms

После создания проекта необходимо нанести на форму все необходимые элементы управления:

* поля для ввода-вывода (text box);
* поля для выбора параметров (combo box);
* кнопки (button);
* подписи для элементов (label);
* элемент группировки для полей (group box);
* статус активности (progress bar);
* форма с вкладками для графика (tab control);
* графики (cartesian chart).

Форму с редактированием элементов можно увидеть на рисунке 4.6.

Изображение выглядит как текст, снимок экрана, внутренний, монитор

Автоматически созданное описание

Рисунок 4.6 – Элементы на форме приложения Windows Forms

4.2.2 Настройка соединения

После нанесения на форму элементов управления необходимо задать настройку подключения для формы c использованием пространства имён System.IO.Ports, для дальнейшей работы с ним.  
Данное пространство имён содержит классы для управления последовательными портами. Наиболее важный из них, класс SerialPort, предоставляет средства для синхронного и управляемого событиями ввода-вывода, для доступа к состоянию подключения-отключения устройства, а также для доступа к свойствам драйвера последовательного порта. Он может использоваться для упаковки объектов Stream, что открывает доступ к последовательному порту из классов, использующих потоки. Пример использования возможностей данного пространства имён показан на рисунке 4.7.

Изображение выглядит как текст, снимок экрана, монитор, черный

Автоматически созданное описание

Рисунок 4.7 – Пример использования возможностей пространства имён System.IO.Ports

Проверить результат работы написанного кода возможно выполнив компиляцию кода, где, выставив нужные значения происходит попытка подключения. В моём случае все выполнено верно, и мы видим сообщение об успешном соединении (рисунок 4.8.)

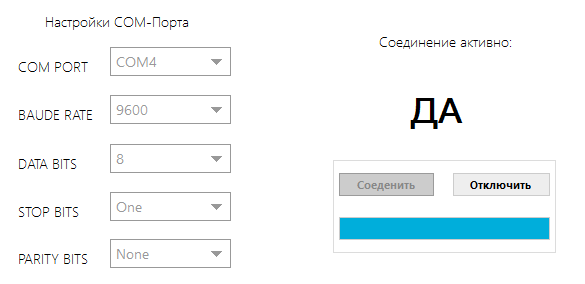


Рисунок 4.8 – Успешное соединение с COM-портом от весов

4.2.3 Фильтрация полученных данных

Следующим шагом после успешной настройки соединения стала фильтрация получаемых данных. Дело в том, что с весов поступают данные в различных типах, включая нужные значения, то есть вес, так и включая ненужные – байты информации. Фильтрация данных была организована с помощью функции Split() а именно StringSplitOptions.RemoveEmptyEntries где происходит обнаружение посторонних символов и удаление их из потока данных, реализация фильтрационного кода показана на рисунке 4.9.

Изображение выглядит как текст, снимок экрана

Автоматически созданное описание

Рисунок 4.9 – Код для фильтрации получаемых данных с весов

Также, немаловажным фактором является то, что весы посылают данные раз весы посылают данные один раз в ~20мс, поэтому, необходимо установить задержку потока передачи данных 25мс, что должно вполне хватить. Это поможет сделать функция Thread.Sleep, что также продемонстрировано на рисунке 4.8.

После фильтрации данных мы получаем нужные значения веса от лабораторной установки, который был установлен в соответствующее текстовое поле (рисунок 4.10).



Рисунок 4.10 – Поле с текущим весом

4.2.4 Расчёт данных

Первым делом необходимо рассчитать влажность (W), которую можно определить по формуле (1):

, (1)

где m0 – начальная масса;

mi – текущая масса.

Также понадобится убыль влаги, которую необходимо получить по формуле (2):

, (2)

где m0 – начальная масса;

mi – текущая масса.

По заданию ещё необходимо рассчитать содержание воды, которое считается разницей начальной массы и конечной.

Соответственно, данные расчёты необходимо произвести и в программном коде, это продемонстрировано на рисунке 4.11.

Изображение выглядит как текст, снимок экрана, внутренний, серебряный

Автоматически созданное описание

Рисунок 4.11 – Расчёты влажности, убыли влаги и содержания воды в коде

4.2.5 Визуализация данных

Дальнейшим действием стало построение графика по расчётным данным.

Для построения графика было выбрано расширение LiveCharts.

LiveCharts – расширение для визуализации данных с открытым исходным кодом для приложений .Net.

LiveCharts обновляется и анимируется автоматически по мере изменения данных в режиме реального времени. Есть возможность добавления, удаления, редактирования значений в коллекции и одновременно с этим его просмотр в пользовательском интерфейсе.

Реализация данного графика стала возможна при его установке в пакете расширений NuGet (рисунок 4.12).

Изображение выглядит как текст, монитор, снимок экрана, черный

Автоматически созданное описание

Рисунок 4.12 – Установленные пакеты LiveCharts

После установки стали доступны всевозможные классы и функции для использования их в визуализации данных.

Так как график обновляется раз в секунду, а приложение работает в одном потоке, поэтому, график необходимо визуализировать в отдельном потоке приложения, это стало возможно благодаря классу BackgroundWorker, который выполняет операцию в отдельном потоке (рисунок 4.13).

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 4.13 – Класс для отдельного потока BackgroundWorker

Функциональную часть графика было принято решение установить на кнопку «Старт», код можно увидеть на рисунке 4.14.

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 4.14 – Код кнопки «Старт» для визуализации данных

## Руководство по работе с приложением

В MasterSCADA для ввода и записи в прибор значений нескольких параметров используются блоки параметров, содержащие одно или несколько мест ввода, кнопки подтверждения и отмены. Запись значений в прибор происходит после нажатия кнопки «ДА».

Важным компонентом системы являются графики. MasterSCADA позволяет производить их тонкую настройку при помощи контекстных меню. Так, можно выбрать один или несколько параметров и назначить на них перья для отрисовки графика. Графики поддерживают несколько уровней отображения:

* базовый;
* минимальный;
* продвинутый;
* таблица значений;
* только графики.

Для изменения уровня графика следует вызвать контекстное меню на графике и выбрать позицию меню «Уровень».

ЗАКЛЮЧЕНИЕ

В ходе выполнения выпускной квалификационной работы был разработан программно-аппаратный комплекс, состоящий из двух измерителей-регуляторов: ТРМ251 и ТРМ148, и двух SCADA систем, выполненных в среде MasterSCADA. Для связи между SCADA системами и приборами были созданы конфигурации OPC сервера.

Разработанная система отвечает всем поставленным требованиям и построена с использованием бесплатного программного обеспечения. Были разработаны методические указания призванные помочь в работе с системами. Реализованы задачи вывода информации с прибора, его параллельного управления и настройки.

Система, её отдельные компоненты и методические пособия могут использоваться в процессе обучения для наглядного отображения влияния изменения различных настраиваемых характеристик.

Программно-аппаратный комплекс внедрён для использования в лабораторию кафедры «Теплофизика и информатика в металлургии».

СПИСОК ЛИТЕРАТУРЫ

1. Visual Studio среда разработки – URL: [https://ru.wikipedia.org/wiki/Microsoft\_Visual\_Studio](https://ru.wikipedia.org/wiki/Microsoft_Visual_Studio%20) (дата обращения 24.05.2022).
2. PuTTY клиент для протоколов удалённого доступа – URL: <https://ru.wikipedia.org/wiki/PuTTY> (дата обращения 24.05.2022).
3. GitHub веб-сервер для хостинга – URL: <https://github.com/> (дата обращения 24.05.2022).
4. GitHub Desktop приложение для упрощённой работы с Git – URL: <https://desktop.github.com/> (дата обращения 24.05.2022).
5. Acom JW-1-200 лабораторные весы – URL: <https://www.mirvesov.ru/soft/acom/jw_new.pdf> (дата обращения 24.05.2022).
6. Инструкция по эксплуатации Samsung CE1000R – URL: <https://www.manualsdir.ru/manuals/418141/samsung-ce1000r.html> (дата обращения 24.05.2022).
7. RS-232 стандарт физического уровня для асинхронного интерфейса – URL: <https://ru.wikipedia.org/wiki/RS-232> (дата обращения 24.05.2022).
8. Руководство по Windows Forms – URL: <https://docs.microsoft.com/ru-ru/dotnet/desktop/winforms/overview> (дата обращения 24.05.2022).
9. System.IO.Ports Пространство имён – URL: <https://docs.microsoft.com/ru-ru/dotnet/api/system.io.ports> (дата обращения 24.05.2022).
10. LiveCharts – визуализация данных – URL: <https://lvcharts.net> (дата обращения 24.05.2022).
11. Руководство по программированию Windows Forms – URL: <https://metanit.com/sharp/windowsforms> (дата обращения 24.05.2022).