МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ

ВЛАДИВОСТОКСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ЭКОНОМИКИ И СЕРВИСА

ИНСТИТУТ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ

КАФЕДРА ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ И СИСТЕМ

ОТЧЕТ  
ПО УЧЕБНОЙ ПРАКТИКЕ ПО ПОЛУЧЕНИЮ ПЕРВИЧНЫХ ПРОФЕССИОНАЛЬНЫХ УМЕНИЙ И НАВЫКОВ

Студент

гр. З/БИС–16–127935 \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Н.В.Торощин

Доцент, к.т.н. \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ М.А. Сачко

Владивосток 2018

# Введение:

Для интеграции приобретённых в процессе теоретического обучения в ВУЗе общекультурных и профессиональных знаний, умений и навыков важен опыт самостоятельной профессиональной деятельности, направленный на закрепление и углубление теоретической подготовки обучающегося, приобретение им практических навыков и компетенций, а также опыта самостоятельной профессиональной деятельности. Таким образом, основной целью практики является: приобретение первичного профессионального опыта.

Задачами практики являются:

* получение первичных профессиональных умений и навыков;
* умение анализировать и обобщать результата научно-исследовательской работы с использованием современных достижений науки и техники;
* овладение знаниями о видах, структуре, организации, основных методах ведения научно-исследовательской работы;
* подготовка к осознанному и углублённому изучению специальных дисциплин;
* сопоставление своих ожиданий и реалий будущей профессиональной деятельности;
* развитие умений самоорганизации, самоконтроля;
* формирование стремления к самосовершенствованию и повышению культурного уровня.

Результаты практики:

Практика направлена на формирование следующих компетенций:

* понимает социальную значимость своей будущей профессии, обладает высокой мотивацией к выполнению профессиональной деятельности (ОК-4);
* владеет широкой общей подготовкой (базовыми знаниями) для решения практических задач в области информационных систем и технологий (ОПК-1);
* способен использовать современные компьютерные технологии поиска информации для решения поставленной задачи, критического анализа этой информации и обоснования принятых идей и подходов к решению (ОПК-5);
* способен проводить предпроектное обследование объекта проектирования, системный анализ предметной области, их взаимосвязей (ПК-1);
* способен проводить техническое проектирование (ПК-2).

# Теория:

**Визуализация техпроцесса** — способ отображения информации о состоянии технологического оборудования и параметрах технологического процесса на мониторе компьютера или операторской панели в [системе автоматического управления в промышленности](https://ru.wikipedia.org/wiki/%D0%90%D0%A1%D0%A3_%D0%A2%D0%9F), предусматривающий также графические способы управления техпроцессом. Это всё выполняет система диспетчеризации. Система диспетчеризации представляет собой комплекс программных и аппаратных средств, который позволяет осуществлять удаленное управление инженерными системами одного или нескольких объектов. Автоматизированная система диспетчерского управления (АСДУ) необходима для контроля инженерного оборудования, разнесенного территориально, а также расположенного в труднодоступных местах. Как правило, диспетчеризация включается в систему управления многофункциональными объектами со сложной инженерной инфраструктурой, такими как офисные здания, торгово-развлекательные центры, а также производственные комплексы и другие промышленные предприятия. Одним из признаков солидного программно-аппаратного комплекса является возможность представления **любых наборов** из сотен технологических параметров в виде **свободноформируемых графических трендов**, обладающих широкими возможностями визуализации, масштабирования и настроек.

**Основные требования:** основные требования к мониторингу инженерных систем установлены в ГОСТ Р 22.1.12-2005, который помимо основных положений и правил включает два приложения:   
- Приложение А (справочное). Порядок создания структурированных систем мониторинга и управления инженерными системами потенциально опасных объектов, зданий и сооружений и их информационного сопряжения с органами повседневного управления РСЧС.  
- Приложение Б (обязательное). Программа комплексных испытаний. Требования к комплексам информационно-вычислительных структурированных систем мониторинга и управления инженерными системами зданий и сооружений, а также методы их испытаний установлены в ГОСТ Р 22.1.14-2013.

# Среда разработки:

Для написания приложения нужно было выбрать среду разработки, в которой был бы разработан проект. Выбор был сделан в пользу Visual Studio 2018 компании Microsoft.

Visual Studio 2018 - продуктов компании Microsoft, включающих интегрированную среду разработки программного обеспечения и ряд других инструментальных средств. Данные продукты позволяют разрабатывать как консольные приложения, так и приложения с графическим интерфейсом, в том числе с поддержкой технологии Windows Forms, WPF 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, Xbox, Windows Phone .NET Compact Framework и Silverlight.

В главном окне содержатся: панель элементов с возможностью поиска (слева), обозреватель решений, в котором изображена «карта» проекта, а также панель свойств (справа). Помимо этого, по умолчанию при запуске Visual Studio (далее VS) появляются форма нового приложения и консоль (снизу). Выше поля конструктора находятся вкладки, в которых также будет редактор кода (рисунок 1).

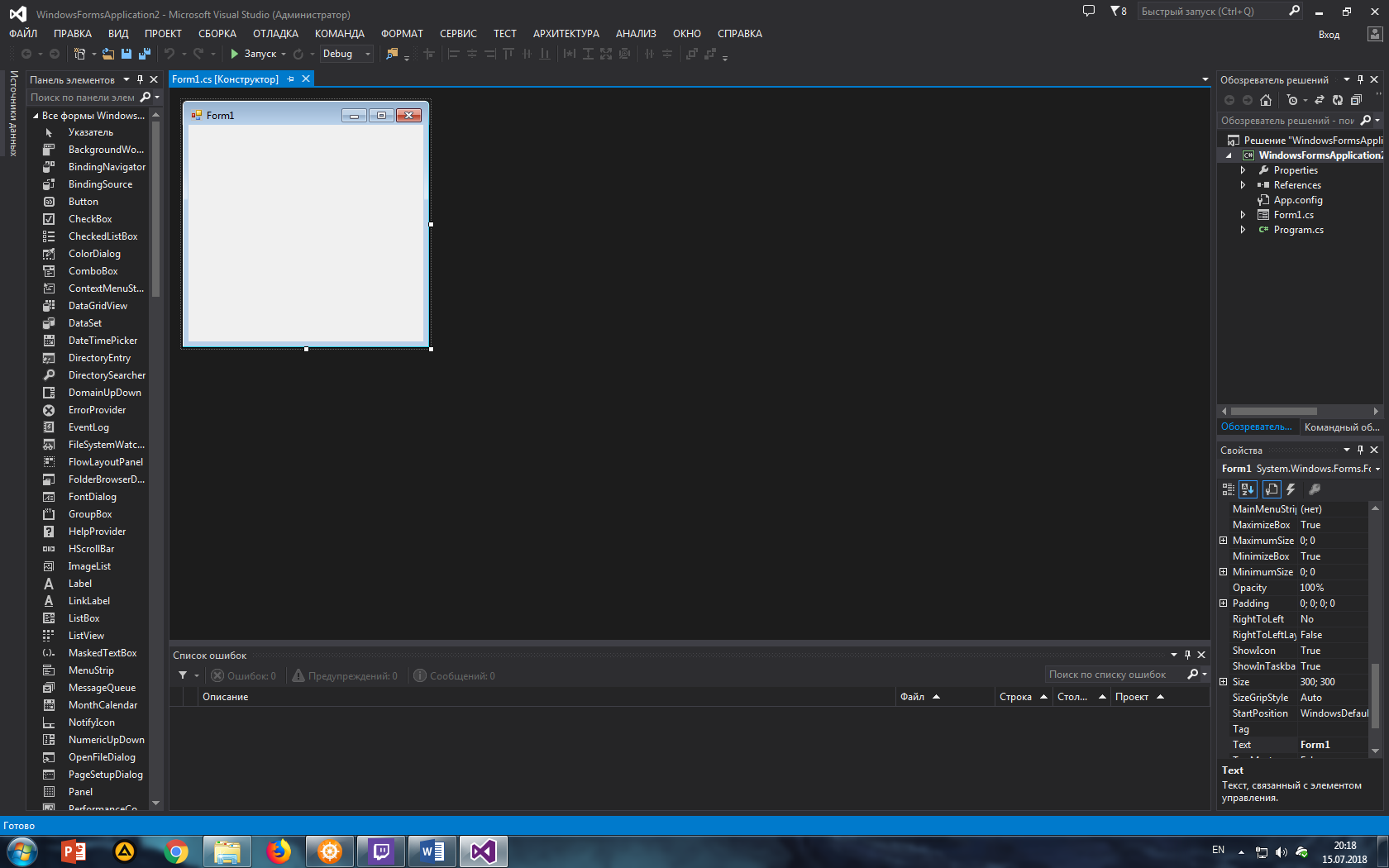


Рисунок 1 – Основные панели приложения Visual Studio.

Формы являются основой приложений VS. Создание пользовательского интерфейса приложения заключается в добавлении в окно формы элементов объектов VS, называемых элементами. Элементы VS располагаются на панели элементов, выполненной в виде списка.

Элементы разделяются на видимые (визуальные) и невидимые (не визуальные). Визуальные компоненты появляются во время выполнения точно так же, как и во время проектирования. Примерами являются кнопки и редактируемые поля. Не визуальные компоненты появляются во время проектирования как пиктограммы на форме. Они никогда не видны во время выполнения, но обладают определенной функциональностью (например, настраиваемый Timer для подсчета времени и др.)

При помощи обозревателя решений можно изменять свойства компонентов. Свойства являются атрибутами компонента, определяющими его внешний вид и поведение. Многие свойства компонента в колонке свойств имеют значение, устанавливаемое по умолчанию (например, высота кнопок). Свойства компонента отображаются на странице свойств. Свойства можно сортировать по алфавиту или по категориям (рисунок 2а).

Также помимо свойств есть и список событий, которые показывают, что будет происходить при вызове определенных триггеров/методов (рисунок 2б). Обозреватель решений используется для установки свойств во время проектирования. Можно определить свойства во время проектирования или написать код для видоизменения свойств компонента во время выполнения приложения.

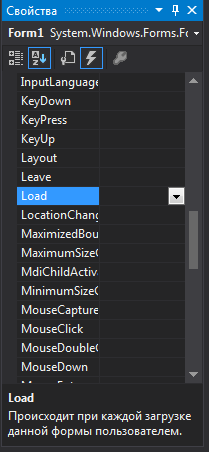
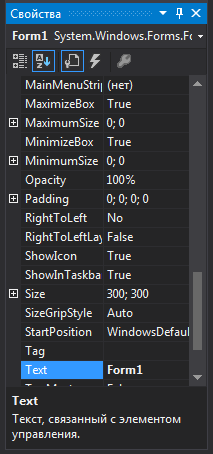


Рисунок 2а – панель свойства. Рисунок 2б – панель событий.

Страница событий инспектора объектов показывает список событий, распознаваемых компонентом. Каждый компонент имеет свой собственный набор обработчиков событий. В VS пишут функции и связывают события с этими функциями. Создавая обработчик того или иного события, вы поручаете программе выполнить написанную функцию, если это событие произойдет.

Метод является функцией, которая связана с компонентом, и которая объявляется как часть объекта. Создавая обработчики событий, можно вызывать методы, используя точку, например:

Form1.Text

Для кодирования и разработки приложений обычно требуется так называемая интегрированная среда разработки (IDE на английском языке). Это приложение предоставляет разработчикам все инструменты, необходимые для выполнения своих задач. Он обычно включает в себя редактор исходного кода, отладчик, компилятор, переводчик и автоматические средства построения, хотя последние меняются в зависимости от того, с какими имеем дело.

Эта программа позволяет быстро исправлять ошибки в коде программы. Красная волнистая линия помогает выявлять и исправлять распространенные проблемы кода. Зачастую это происходит в реальном времени, пока вы вводите код, что позволяет быстро отреагировать (например, выполнить рефакторинг, реализовать интерфейсы и пр.) прямо в редакторе (рисунок 3).

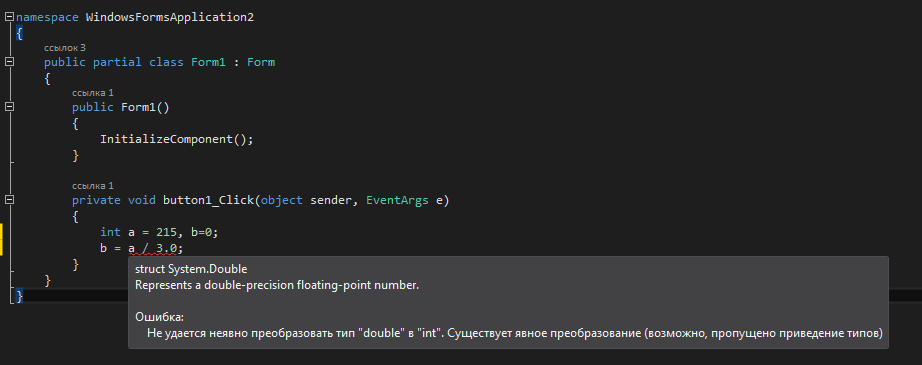


Рисунок 3 – Функция исправления ошибок.

Visual Studio помогает справиться со сложностями понимания больших объёма текста, не теряя контекст кода или разметку, с которой вы начали, с помощью таких возможностей, как отображение определения. Улучшенная функция перехода GoTo позволяет легко фильтровать данные и выбирать, какие типы элементов искать.

Поиск всех ссылок упрощает группировку, фильтрацию и поиск в результатах, а также позволяет хранить произвольное число наборов результатов. Улучшенное крупномасштабное структурное представление панели прокрутки позволяет быстро находить проблемы (рисунок 4).

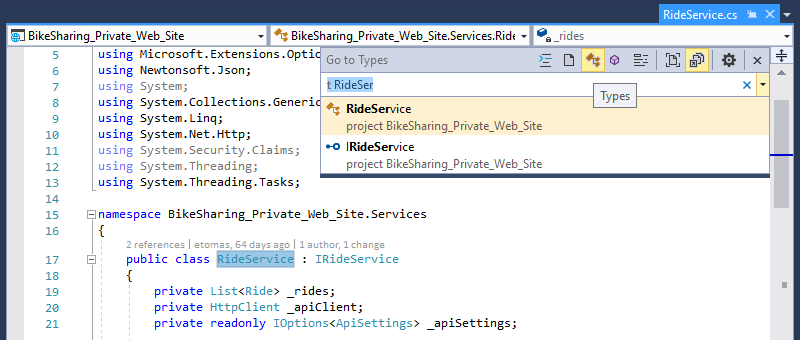


Рисунок 4 – Функция GoTo.

Visual Studio 2018 занимает около 8.8 Гб. С его помощью можно относительно легко создать многофункциональное приложение для Windows, в том числе и создание графиков и последующее анимация примитива. Microsoft VS при работе с проектом сам подключает все файлы из проекта, и без сложных манипуляций с настройками компилятора, компилирует полностью рабочий файл с расширением «.exe».

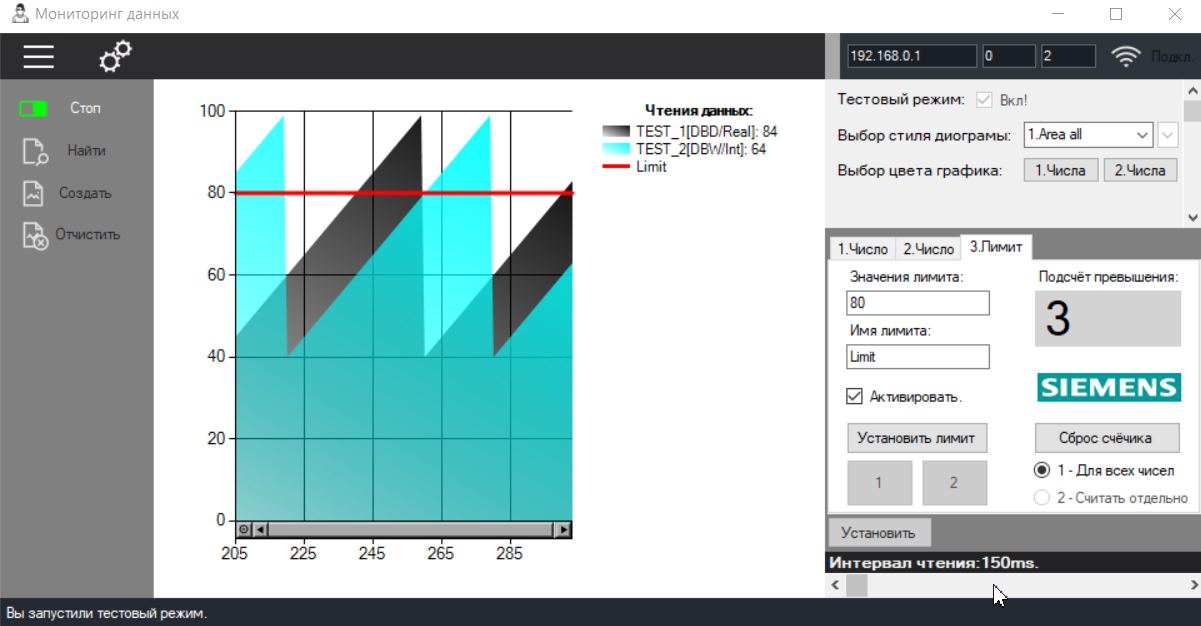
Хоть Microsoft Visual Studio 2018 является проприетарным ПО и требует покупки лицензии (или же можно использовать триал-версию на 30-90 дней), у меня уже есть аккаунт с приобретённой версией VS и некоторых других приложений Office, так что никаких проблем с использованием и поддержкой у меня не будет.

Исходя из этого для создания проекта по постройке графиков, была выбрана среда разработки компании Microsoft под названием Visual Studio 2018.

Дополнительно ещё выбрал и скачал бесплатную программу Icons8(для иконок) и библиотеку sharp7 (для работы с контролерами Seimens).

# Описание программы:

Написанное мной приложение (рисунок 5а), имя которого: - «Мониторинг данных», предназначена для прямого мониторинга и сравнение данных, с промышленных контролеров из семейства Siemens S300/400/1200/1500, а также отображения их, в виде трендов (в стиле Line - линия, Area - площадь) и подсчёта превышений лимита, установленного пользователем. Данное приложение не может считывать за раз, более двух чисел. Разработан тестовый режим для разработки пресетов. Минимальная скорость чтения данных равна 150 мс., максимальная равна 1с. 491мс. Приложение имеет минимальное значение, разрешения экрана 760x520. Целевая платформа: Windows 7(все русскоязычные версии)/10(все русскоязычные версии). Референс данной идеи было приложение S7-trends (рисунок 5б и 5в).

 Рисунок 5а – Мониторинг данных, тестовый режим

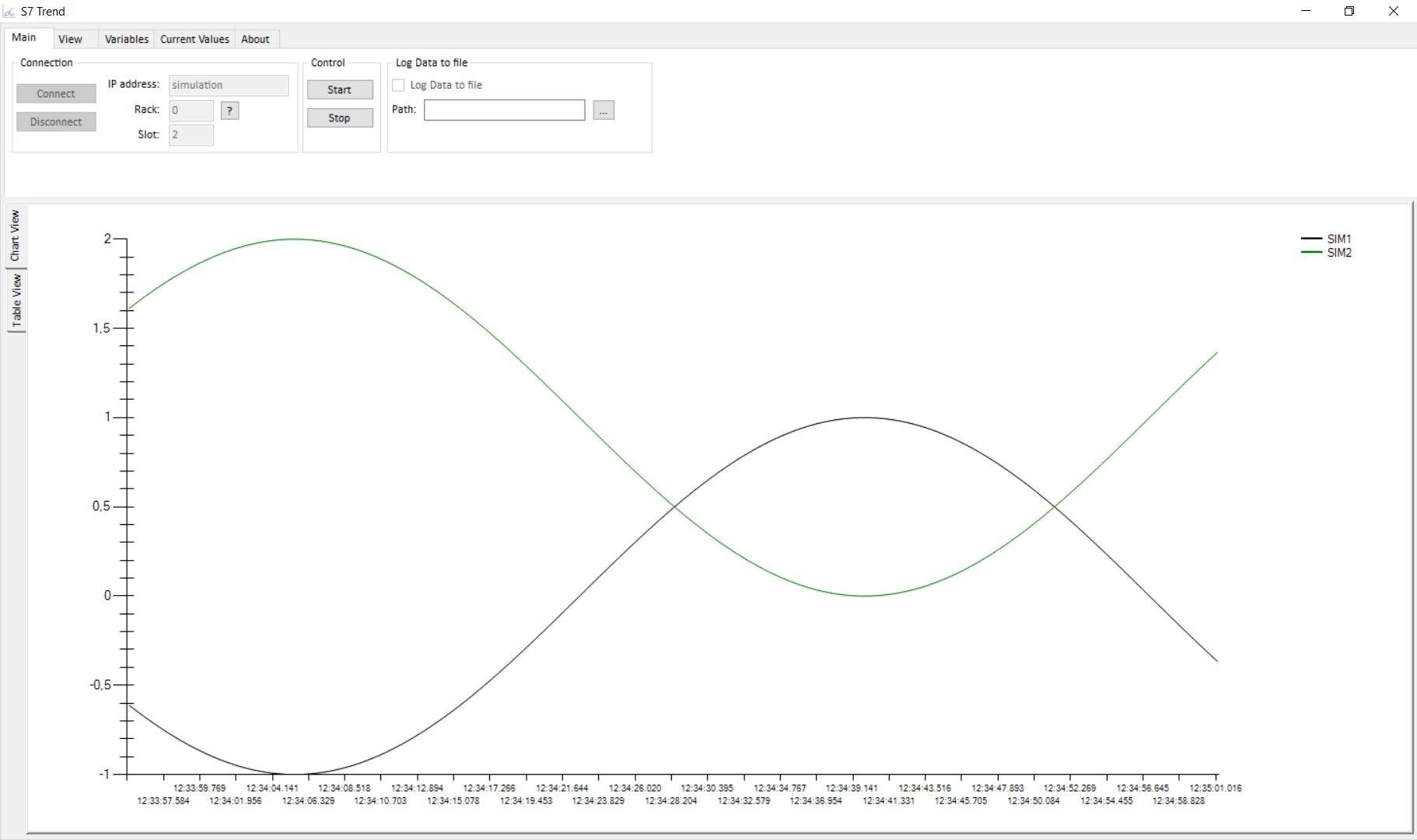


Рисунок 5.б – **S7-trends** чтение данных.

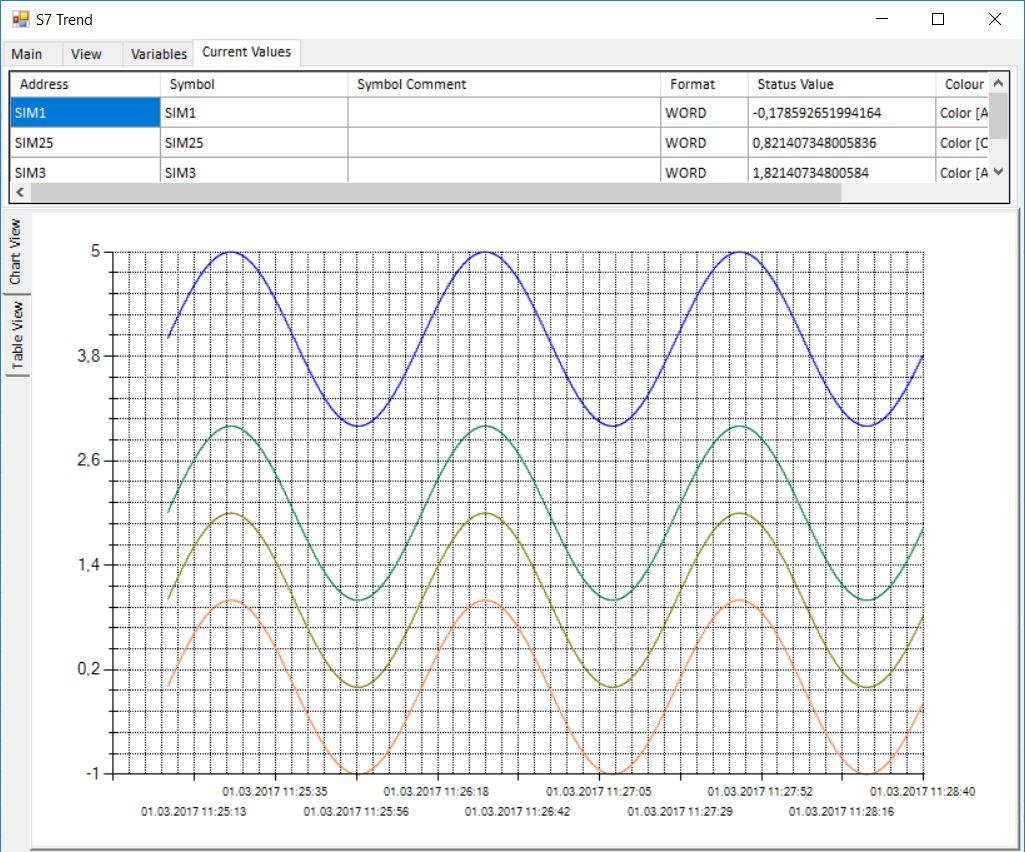


Рисунок 5.в – **S7-trends** обзор значений показаний

# 4. Тестирование программы:

После написания программы, она была протестирована на различные входные данные (пользовательские и считываемые с контролера), на наличие ошибок связи между приложением и контролером. (таблица 1).

Таблица 1. Тестирование программы.

|  |  |  |
| --- | --- | --- |
| Номер теста | Входные данные | Выходные данные |
| 1 | Ввод символов отличных от цифр, точек, в компоненты программы. | Приложение не позволяет ввести какие-либо символы кроме цифр, точек. |
| 2 | Попытка ввода значений, выходящих за рамки формата чисел Step7. | Программа определяет формат числа, без ошибок. |
| 3 | Проверка чекбоксов. На не заполненные поля. | При пустом чекбоксе, успешно блокируется рабочая область. |
| 4 | Проверка подключения, контролеру. | В обычном режиме, работы программы, производиться проверка связи, при обнаружении каких либо ошибок программа уходит в режим ожидания. |
| 5 | Проверка нагрузки процессора и использования оперативной памяти. | При включенном окне настройки: <0% загрузки ЦП и среднее потребление памяти 7800-7900 КБ. В рабочем режиме нагрузка на ЦП с минимальной скоростью считывание ~1-3%, потребление памяти ~9700 - 16000 КБ. |
| 6 | Корректность отображения данных, с контролера. | Верно считывает и отображает данные с контролера. |
| 7 | Масштабирование трендов. | В режиме пауз, масштабирование происходит корректно. Без замечаний. |
| 8 | Изменения цвет тренда (и толщины лини). | Замена цвета отображение трендов происходит отлично. Тренды принимают цвет, согласно установки пользователя. |
| 9 | Наличие ошибок работы, в тестовом режиме. | Ошибок не обнаружено, тестовый режим работает без ошибок. |
| 10 | Наличие ошибок, при подсчёт превышения лимита. | Данный алгоритм подсчёта лимит работает верно. Замечаний не обнаружено. |
| 11 | Проверка работы поиска/создания пресетов. | Отработка поиска и создание пресетов прошла успешно, ошибок не выявлено. |
| 12 | Нагрузка локальной сети. | Нагрузка составляет 0,1% от общей нагрузки сети. |

# 5. Инструкция пользователя:

Данная программа не нуждается в OPC серверах, дополнительных драйверах и работает на прямую с контролером. Так же приложение работает в двух режимах Обычный (для работы с контролером) и Тестовый (для создания пресетов и эмуляции работы программы).

**5.1. Обычный режим:** с момента запуска программы, она настроена на работу с вашим контролером. Чтобы приступить, к работе, вам не обходимо заполнить блок связи (находиться в правом верхнем углу, рисунок 6.а) вести ip адрес, номер стойки (Rack), номер слота (Slot) и нажать кнопку с «Подкл.». После успешного подключения, серый индикатор в блоке связи, поменяет цвет на зелёный, и в нижнем левом углу высветиться информация о скорости подключения, к контролеру и его состояние (см. Рисунок 6.б). Затем перейдите в раздел общие настройки нажав на кнопку «общие настройки» и установите считываемые данные с PLC заполнив все поля рабочей области, во всех окнах (находиться в правой панели, нижней правой области программы, для работы приложения и прямого мониторинга данных, достаточно заполнить одно окно «1. Число» с целью отображения одного числа в виде тренда/графика. см. Рисунок 6.в). Как заполнили нужные вам поля, **ОБЯЗАТЕЛЬНО УТСАНОВИТЕ ДАННЫЕ В ПРОГРАММУ!!!** (нажав кнопку «Установить» (Рисунок 6.в 1.7)). После установки данных в программу, перейдите в левую панель нажав кнопку, в верхним левом углу приложения и нажмите кнопку «Старт» для отображения данных в виде графиков. (см Рисунока 6.г).

Для подсчёта и установки превышения лимита вам необходимо обратить внимание на Рисунок 6.е. И обязательно заполнить поле 1.1 «Значения лимита». Затем активируйте его, поставив галочку в 1.3 (Активировать лимит). После установите лимит нажав на кнопку 1.4«Установить лимит», программа автоматически преступит считать количество превышений лимита, в реальном времени. Для обнуления подсчёта лимита нажмите кнопку 1.5 «Сброс счётчика».

Так же вы можете изменять цвет графиков, стиль отображения диаграмм (если у вас стоит стиль Line то для него можете изменить толщину линий (см. Рисунок 7).

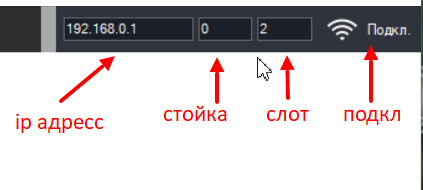


Рисунок 6.а – блок связи.

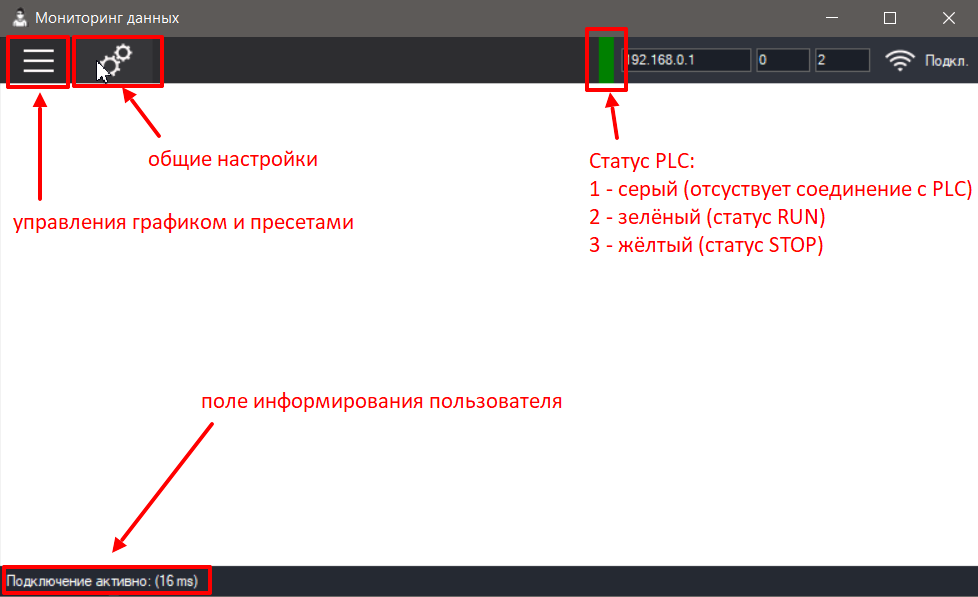


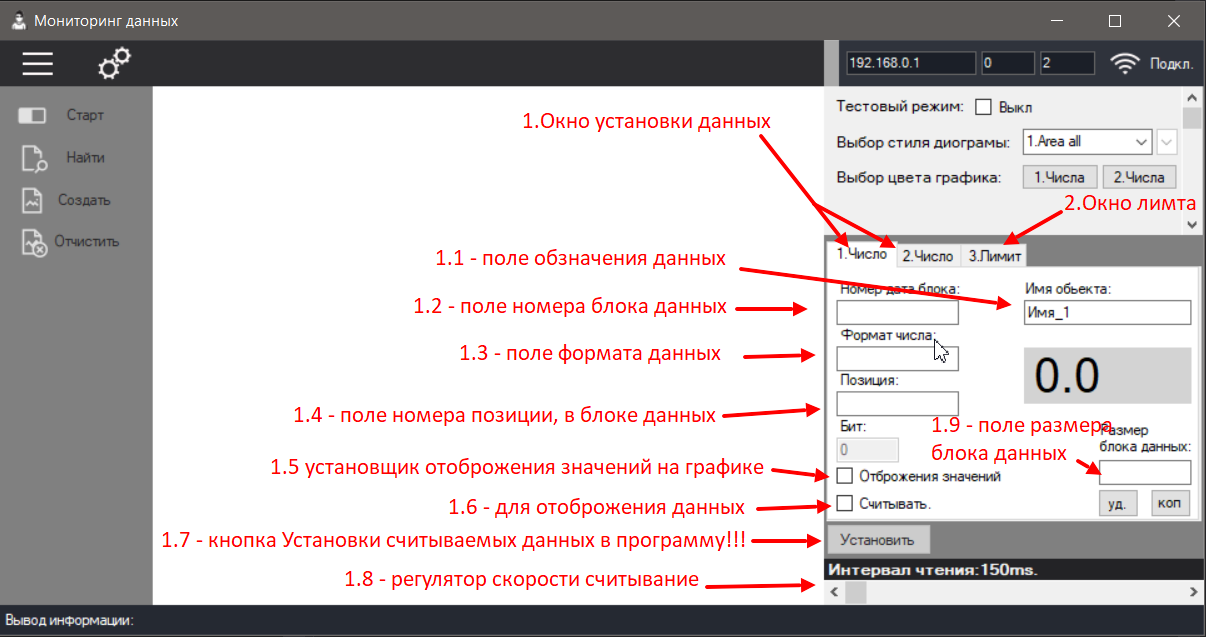
Рисунок 6.б – Подключение, к контролеру

Рисунок 6.в – Общие настройки и рабочая область установки данных

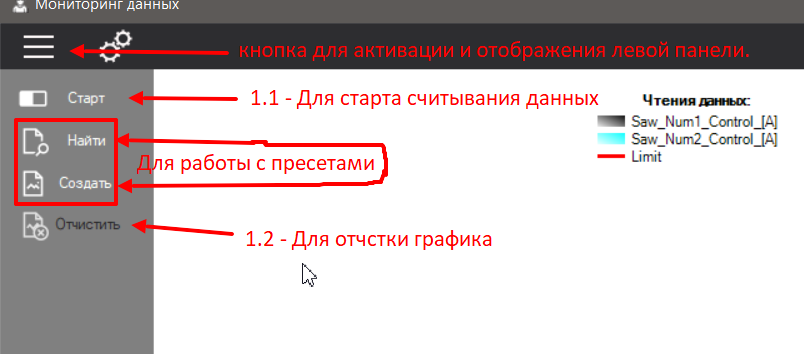


Рисунок 6.г – Ознакомления с левой панелью управления.

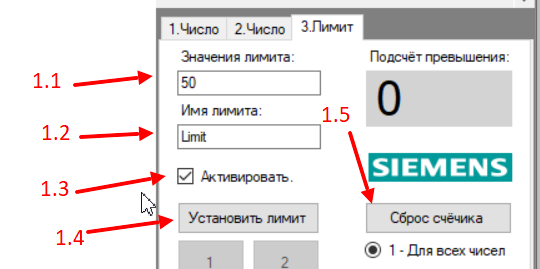


Рисунок 6.е – Окно лимита.

**5.2 Тестовый режим:** В отличие от обычного режима, тестовый режим требует активации. В тестовом режиме, программа не требует установки данных. Поставьте галочку в соответствующем разделе, в общих настройках. При активации программа с имитирует данные, с проекта СМРК (Само резательный комплекс) запуск и работа двух пил, в автоматическом режиме. Чтобы считать данные тестового режима, нажмите кнопку «Старт» (см пункт 5.1).

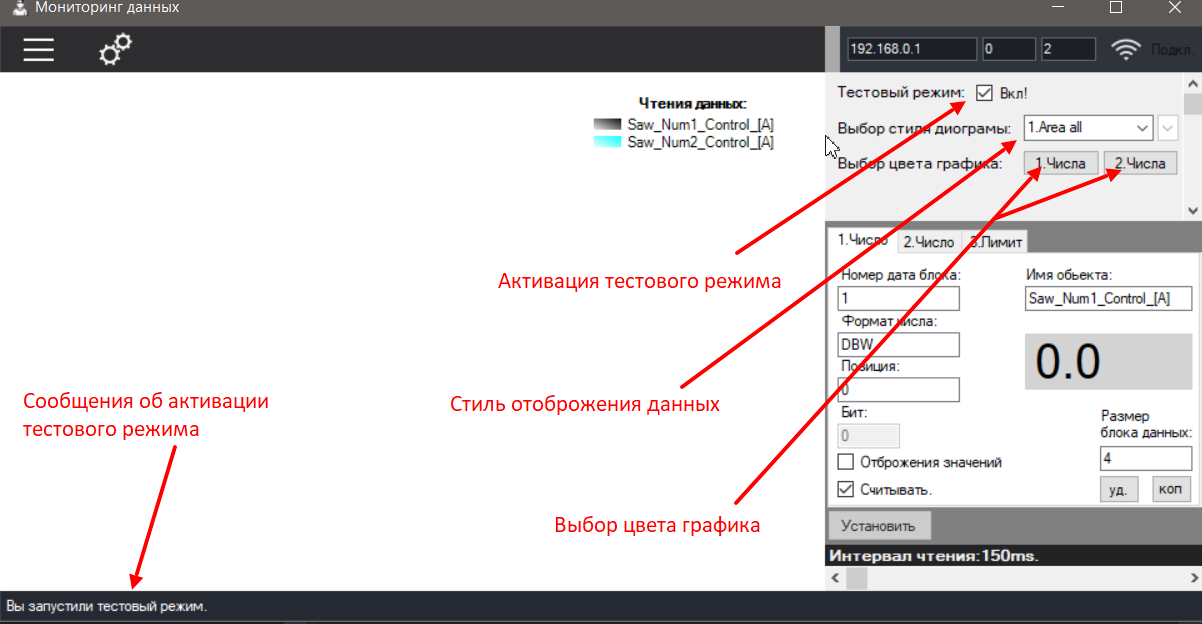


Рисунок 7 – Работа тестового режима.

# 6. Заключение:

В процессе прохождения практики была разработана программа для прямого мониторинга данных, с промышленных контролеров Siemens из семейства S300/400/1200/1500. Приобретен опыт работы в программе Visual Studio, базовые навыки программирования на языке С#, опыт работы с Windows Forms и его компонентами, так же получил знания об разработки методах ООП, научился задействовать простейшие алгоритмы, согласно поставленным задачам, и разработки удобного интуитивно-понятного дизайна. Но самое главное был получен опыт по реализации идеи, при помощи полученных знаний, за момент обучения. Научился расставлять приоритеты, составлять план проектирования разработки приложения, получил умения оценивать время и выводить примерные сроки окончание разработки.

# 7. Список литературы:

1. Хейлсберг А., Торгерсен М., Вилтамут С., Голд. П.: Язык программирования С#. СПб.: Питер, 2012. 1 - 790 с.

2. Davide Nardella. Sharp7 Reference manual: 2016. 3-62c.

3.Национальный стандарт Российской Федерации. ГОСТ Р 22.1.14-2013/22.1.12-2005.