

# SmartView<sup>™</sup> 4.0.2.2

# Руководство по ускоренному

## ознакомлению

#### Информация о документе

Software Version:	4.0.2.2
Creation Date:	Октябрь, 2019
Last Edit Date:	27 апреля, 2021
Version:	1.4



## Оглавление

1. Введение в ADISRA SmartView™	6
2. Обзор ADISRA SmartView™	7
3. Загрузка ADISRA SmartView™	8
4. Установка ADISRA SmartView™	9
4.1 Требования к аппаратному обеспечению	9
4.2 Требования к системному программному обеспечению	9
4.3 Процесс инсталляции	10
5. Регистрация лицензии ADISRA SmartView™	15
6. Запуск ADISRA SmartView™	16
6.1 Знакомство со средой разработки	17
6.2 Закрепленные окна	18
6.2.1 Плавающие панели	18
6.2.2 Закрепление панели	19
6.2.3 Отображение / скрытие панели	20
6.2.4 Автоматическое скрытие панели	21
7. Создание нового проекта	22
7.1 Открытие проекта	24
8. Документ тегов (Tags)	25
8.1 Создание нового списка тегов (Tags Document)	25
8.2 Добавление новых тегов	26
8.3 Сохранение документа	26
8.4 Конфигурирование тегов	27
9. Документ описания типа данных (Data Type)	28
9.1 Создание и сохранение документа описания типов данных (Data Type Document)	28
9.2 Добавление нового тега в документ описания типа данных	28
9.3 Конфигурирование тегов в документе описания типа данных	29
9.4 Добавление тегов нового (созданного ранее) типа в список тегов	29
10. Службы (Services)	31



ADISRA SmartView Getting Started Guide	
10.1 Конфигурирование документа Startup (Запуск)	31
10.2 Создание документа службы (Services), с выполнением по условию	33
10.3 Создание документа службы с запуском по триггеру	34
11. Создание документа экранных форм	36
11.1 Установка разрешения по умолчанию для графики	36
11.2 Создание графической формы для запуска и проверки скриптов	36
12. Триггеры (Triggers)	42
12.1 Создание новых тегов для запуска триггеров (Trigger Tag)	42
12.2 Создание простых триггеров (Trigger)	42
12.3 Создание триггеров с условием (Condition Trigger)	44
12.4 Создание триггеров по календарной дате/времени (Calendar Trigger)	45
12.5 Создание интервальных триггеров (Interval Trigger)	46
13. Настройка экранных форм для визуализации триггеров	48
14. Журнал Тревог (Alarm History)	50
14.1 Конфигурирование тревог тегов	50
14.2 Настройка элементов журнала тревог (Alarm History Items)	53
14.3 Создание и настройка экранных форм для отображения тревог	54
15. История значений тегов (Tag History)	60
15.1 Создание и настройка архива с записью по изменению тега (On Tag Change)	60
15.2 Создание и настройка архива с записью по таймеру (Time Frequency)	61
15.3 Создание и настройка экрана для вывода исторических трендов	62
16. Рецепты (Recipes)	65
16.1 Создание и настройка рецепта (Recipe Document)	65
16.2 Создание и настройка экранной формы для работы с рецептами	66
16.3 Различия между разными блоками (GroupBoxes) Load / Save в нашем экране Recipe	70
16.4 Ожидаемый результат выполнение экрана Recipes	70
17. Отчеты (Reports)	71
17.1 Создание и конфигурирования документа типа Report	71
17.2 Создание и настройка экранной формы для генерации Отчетов	73
17.3 Ожидаемый результат выполнение экрана Reports	78
18. Драйверы	79



18.1 Документ типа ОРС UA клиент	79
18.1.1 Загрузка и установка внешнего ОРС UA сервера	79
18.1.2 Создание документа ОРС UA Client	84
18.1.3 Создание экранных форм отображение тегов из ОРС UA сервера	88
18.2 Документ драйвера Modbus	90
18.2.1 Загрузка и установка внешнего эмулятора Modbus (mod_RSsim)	90
18.2.2 Создание документа типа Modbus Driver	92
18.2.3 Вывод переменной из драйвера Modbus на экранную форму	95
19. Документы туннелирования Tunnelling	99
19.1 Создание документа типа Tunneling	99
19.2 Создание и настройка экрана для иллюстрации туннелирования	101
20. Библиотека функций пользователя (User Function Library Documents)	102
20.1 Создание библиотеки пользовательских функций	102
20.2 Создание и настройк <mark>а экранных</mark> форм для иллюстрации использования поль функций	зовательских 103
21. Настройка библиотеки изображений	105
21.1 Добавление нового изображения в библиотеку	105
22. Графические объекты с расширенным функционалом	106
22.1 Объекты Экран (Screen)	106
22.2 Настройка экранной формы ObjectsScreen	108
22.2.1 Использование TextBox для ввода пароля	108
22.2.2 Добавление изображения в дизайн кнопки	109
22.2.3 Объект типа TextBox с динамически изменяемым фоном	109
22.3 Настройка окна для отображения трендов	111
22.3.1 Создадим новый тег для наблюдения тренда	111
22.3.2 Создание экранной формы для вывода трендов	112
23. Создание главного экрана проекта	114
23.1 Создание главного экрана	114
23.2 Настройки главного экрана	114
23.3 Добавление и настройка графических объектов	116
23.3.1 Добавление и настройка элементов типа GroupBox	116



ADISRA SmartView Getting Started Guide	
23.3.2 Добавление и настройка кнопки закрытия приложения (выход)	116
23.3.3 Установка логотипа ADISRA SmartView на главном экране	117
23.3.4 Добавление вкладок	119
23.3.5. Настройка экранов, которые должны отображаться во вкладках элемента Tab	120
24. Запуск на исполнение и отладка проекта	122



## 1. Введение в ADISRA SmartView™ Что такое ADISRA SmartView?





ADISRA SmartView — это мощный программный инструмент на базе Windows, который позволяет создавать надежные системы HMI/SCADA и клиент-серверные приложения для промышленной автоматизации. ADISRA SmartView предоставляет разработчику интуитивно понятную единую среду разработки со всеми необходимыми функциями. Это позволяет быстро создавать проекты и разворачивать их на большинстве действующих и вновь организуемых предприятиях, оснащенных стандартными средствами промышленной автоматики, системами ввода-вывода и индустриальными сетями.



410 × 1 1	e	Property Minibus	+ * *	
Image: Section 1         Section 2           Image: Section 2         Section 2           Image: Section 2	Personal Management Annual Annua			

## 2. Обзор ADISRA SmartView<sup>™</sup>

ADISRA SmartView имеет единую и хорошо знакомую специалистам среду разработки, включающую в себя все необходимые функции и возможности типовых SCADA/HMI систем, используемых для автоматизации технологических процессов.

С помощью ADISRA SmartView Вы сможете создавать мощные приложения HMI и SCADA, легко редактировать и сопровождать их. В ADISRA SmartView уже включены необходимые коммуникационные драйверы, позволяющие легко интегрировать в единую систему различные модули, компоненты и устройства. Вы можете также обмениваться информацией с такими источниками данных как OPC серверы, базы данных, программные продукты PI-OsiSoft, устройства класса IoT и Edge, промышленные контроллеры и сети, облачные вычислительные среды.

Основные характеристики ADISRA SmartView:

- Возможность просмотра экранов на планшетах и мобильных телефонах через браузеры, совместимые со стандартом HTML5.
- Локальные, удаленные и веб-подключения.
- Многомерные и динамические теги.
- Встроенная подсистема резервирования.
- Сценарии и функции на языке программирования С#
- Создание стандартных приложений HMI и SCADA.
- Создание распределенных и клиент-серверных приложений.
- Связь и интеграция с внешними источниками.
- Отображение данных в реальном времени на экранах (в виде трендов, диаграмм, сигналов тревоги, цветные индикаторов, текстовых полей и т. д.)



## 3. Загрузка ADISRA SmartView<sup>™</sup>

ADISRA SmartView распространяется через сеть Интернет и доступна для скачивания с сайта <u>www.adisra.com</u>.

C Sitter/www.aden.com/ P = 1 Siten.com +	n * =
ADISINA	
Web Content TRD	

Для загрузки SmartView выполните, пожалуйста, следующие шаги:

- 1. Откройте в браузере сайт <u>www.adisra.com</u>.
- 2. Перейдите по ссылке Download Software.
- 3. После окончания загрузки разархивируйте полученный ZIP архив.
- 4. Для этого кликните правой кнопкой мыши на полученном файле, и выберите пункт Extract files. Выберите доступное расположение для размещения файлов дистрибутива.



## 4. Установка ADISRA SmartView<sup>™</sup>

Для установки ADISRA SmartView рекомендуется использовать систему со следующими характеристиками:

## 4.1 Требования к аппаратному обеспечению

Минимально допустимые:

- CPU: 1.44 GHz
- RAM: 4 GB
- Hard-Disk (min): 8 GB

Рекомендованные:

- CPU: 1.7 GHz
- RAM: 6 GB
- Hard-Disk (min): 20 GB

#### 4.2 Требования к системному программному обеспечению

- OC: Windows 10, Windows 8.1, Windows 7 SP1, Windows Server 2019, Windows Server 2016, Windows Server 2012 R2, Windows Server 2012 (64-bit edition)
- .NET Framework 4.8
- OPC Core Components

Программа установки ADISRA SmartView проверит, что ПК, на котором ока устанавливается, имеет все необходимые для работы компоненты. Если какие-либо из них необходимо установить (кроме операционных систем Windows), установщик скачает и установит их автоматически.

Примечание: Программа установки сможет лишь проверять наличие на Вашем ПК компонента .NET Framework 3.5 + Service Pack 1, однако, при его отсутствии, не установит его автоматически. Пользователю необходимо будет самостоятельно установить (или обновить до нужной версии) этот компонент. Если все необходимые компоненты установлены правильно, то следующим шагом программа установки перейдет к установке ADISRA SmartView.



## 4.3 Процесс инсталляции

1. .NET FRAMEWORK 4.8 является НЕОБХОДИМЫМ компонентом для установки. Если вы попытаетесь установить ADISRA SmartView на ПК без Microsoft .NET Framework 4.8, вы получите следующее сообщение об ошибке:



В этом случае для установки .NET Framework 4.8 Runtime необходимо перейти на веб-сайт Microsoft:

https://dotnet.microsoft.com/download/dotnet-framework/net48

После того, как вы загрузили установочный файл .NET под названием ndp48web.exe, дважды щелкните файл, чтобы завершить установку. Вам будет предложено перезагрузить компьютер.

 После загрузки и извлечения ADISRA SmartView из ZIP-файла, как описано в разделе <u>Загрузка ADISRA SmartView</u>, перейдите в подпапку en-us. Вы должны запустить файл Setup.exe от имени администратора, щелкнув правой кнопкой мыши Adisra SmartView.exe и выбрав Запуск от имени администратора в раскрывающемся меню.





 Если это первая установка ADISRA SmartView, откроется новое окно, информирующее о том, что некоторые компоненты будут установлены. Щелкните Установить. Примите условия в соглашении об установке и следуйте инструкциям по установке на каждом экране. Вы увидите следующий экран.

		mponents will be installed on your machine:	
7. Office Save as XPS and PDF Do you wish to install these components?	1. Visual C++ 2. Visual C++ 3. Visual C++	Redistributable 2005 SP1 Redistributable 2008 Redistributable 2010	Ŷ
Do you wish to install these components? you choose Cancel, setup will exit.	7. Office Sav	e as XPS and PDF	~
	Jo you wish ti	ncel, setup will exit.	

Для каждого устанавливаемого компонента будет окно сообщения с лицензией и условиями соглашения. Установите все соответствующие флажки и продолжайте.

icense Terms		
e sure to carefully read and under cense terms. You must accept the	stand all the rights and restrictions desc license terms before you can install the	ribed in the software.
MICROSOFT SOFTWARE LICENSE T MICROSOFT VISUAL C ++ 2008 RU PACK 1 These license terms are an agreem where you live, one of its affiliates) software named above, which inclu terms also apply to any Microsoft • updates,	TERMS NTIME LIBRARIES (X86, IA64 AND X64) ent between Microsoft Corporation (or b ) and you. Please read them. They app ides the media on which you received it,	, SERVICE
braze the Dane Deven key to see me	we have	Print
✓ I have read and accept the	license terms.	

 После установки компонентов начнется установка программы ADISRA SmartView. Нажмите Далее, затем нажмите Принять условия установки и следуйте инструкциям по установке.





Прочитав все лицензионное соглашение, установите флажок Я принимаю условия лицензионного соглашения и нажмите Далее.

Please read the following license	e agreement carefully	ADIS	RA
5 			-
BY INSTALLING OR USING TH	IS SOFTWARE, YOU ("YOI	U" OR "LICENSEE") AGREE	
TO THE TERMS AND CONDITI	ONS CONTAINED IN THI	S EULA. IF YOU DO NOT	
AGREE TO THE TERMS AND CO	ONDITIONS CONTAINED I	N THIS EULA, TERMINATE	
ICENSE AGREEMENT" IN THI	IS WINDOW OR BY CIT	KING ON THE "CANCEL"	
BUTTON, YOU MAY THEN	PROMPTLY DELETE T	HE LICENSE FILES AND	
SOFTWARE FROM YOUR CO	MPUTER AND RETURN T	HE LICENSE FILE MEDIA,	
		TO TELE DECUDIES	
I accept the terms in the Licen	ise Agreement		

5. При появлении запроса вы должны решить, где установить ADISRA SmartView. Местоположение по умолчанию находится в папке Program Files. Если вы нажмете Далее, он будет установлен в папку по умолчанию C:\Program Files (x86)\ADISRA\Adisra SmartView. Чтобы установить в другом месте, нажмите кнопку Изменить и выберите желаемое местоположение.



				ADISRA.
Instal ADISRA Sma	artView to:		ISOMEBERS	
C: Program Files (	x86)\ADISRA\Adisra Sr	nartView\		
Add ADISRA Se	nartView Start Menu			
Create shortcu	t on Desktop			
	Install ADISRA Sma C: 'Program Files ( Change V Add ADISRA Sm V Create shortcut	Install ADISRA SmartView to: C: 'Program Files (x86) \ADISRA\Adisra Sr Change Add ADISRA SmartView Start Menu Create shortcut on Desktop	Install ADISRA SmartView to: C:\Program Files (x86)\ADISRA\Adisra SmartView\ Change Add ADISRA SmartView Start Menu Create shortcut on Desktop	Install ADISRA SmartView to: C: 'Program Files (x86)\ADISRA\Adisra SmartView\ Change Add ADISRA SmartView Start Menu Create shortcut on Desktop

6. При появлении запроса вы можете выбрать установку на свой компьютер только ADISRA SmartView или Viewer, либо их обоих. По умолчанию оба будут установлены на вашем компьютере. Чтобы отказаться от установки любого из них, отмените выбор этого параметра и нажмите **Далее**.



7. Щелкните Install, чтобы начать установку ADISRA SmartView.



После завершения установки нажмите Готово и закройте окно инсталлятора.



## 5. Регистрация лицензии ADISRA SmartView<sup>™</sup>

Перед началом использования программного обеспечения Вы должны зарегистрировать лицензию на свою копию ADISRA SmartView.

1. После установки ADISRA SmartView есть два способа регистрации лицензии: 1.1. Откройте пункт меню Help и выберите пункт меню Register:



- 1.2. Или запустите утилиту регистрации с помощью меню Windows: Start > All Programs > SmartView > Register License.
- 2. После этого Вы увидите следующее диалоговое окно:

License	- 0 - X
License	
ADISRA SmartView 4.0 Version: t1.0.0.9 Hardware ID Output path: C:\Users\Me\Desktop\license.code	
Generate license.code After generating the license.code send it to ADISRA	
License License path:	]
	Close

3. Нажмите кнопку Generate license.code для того, чтобы создать текстовый файл, в который войдет уникальная информацию о Вашем ПК (Hardware ID). Место расположения этого файла можно изменить в поле Output Path

Примечание: Изменить расположение файла можно также нажав кнопку «...» справа от поля Output Path.

- 4. Вложите полученный файл SmartView.code в электронное письмо и отправьте его по адресу info@adisra.ru.
- Поставщик ответит Вам файлом с лицензионным ключом, который будет в точности соответствовать Вашему Hardware ID и заказанной лицензии. Сохраните этот файл на Вашем ПК и запомните его месторасположение (путь к файлу).
- 6. После этого ещё раз запустите утилиту лицензирования и введите путь к сохраненному Вам файлу в текстовом поле License Path, или выберите путь к этому



файлу, нажав кнопку «...» справа от поля License Path. После этого нажмите кнопку Validate.

- 7. Вам будет предложено подтвердить операцию, как только программа примет ваш лицензионный ключ.
- 8. После этого Ваш ПК полностью готов к использованию ADISRA SmartView.
- 9. Закройте окно Register License и запустите ADISRA SmartView.

## 6. Запуск ADISRA SmartView™

После получения надлежащей лицензии на ADISRA SmartView запустите программное обеспечение одним из следующих способов:

- Меню Windows: Start > All Programs > ADISRA SmartView > ADISRA SmartView или
- Двойной клик на иконке ADISRA SmartView, которая была создана на рабочем столе ПК в процессе установки.

Computer		Ves Sroup South	
	Mar Star Samer Des		turne .
Recycle Bin	<ul> <li>A State Accounting Come</li> </ul>		a Tarri
ADISRA SmartView	C Alama Andrea British Distance		
2			
Config		Search Dubaut Mensager John Location	*#X
2			
Viewer	· inment financial		and the second second

Примечание: Если ADISRA SmartView открывается впервые, инженерная среда будет пустой (как показано ниже). В противном случае она откроется с загруженным в нее последним использованным проектом.

После запуска ADISRA SmartView вы можете создать новый проект или открыть существующий проект.



## 6.1 Знакомство со средой разработки

Среда разработки ADISRA SmartView — это удобный и интуитивно понятный ленточный интерфейс Windows<sup>®</sup> с множеством функций. Вы можете настроить этот интерфейс максимально удобным Вам образом и сразу же начать работу над проектом.



На изображении выше представлена компоновка стандартной среды разработки ADISRA SmartView. В следующих разделах Руководства по началу работы мы будем ссылаться на различные области макета инженерной среды, такие как:

- Основное меню (Menu Button)
- Панель навигации (Navigation Tree)
- Панель логики (Logical Tree)
- Лента (Ribbon)
   Панель быстрого доступа (Quick Access Toolbar)

- Вывод поиска (Search Output)
- Журнал событий (Engineering Log pane)
- Панель свойств (Properties)
- Журнал исполнения (Runtime Log pane)
- Панель построителя проекта (Build Output)

Note: Для детальной информации воспользуйтесь встроенной подсистемой помощи (Help).



## 6.2 Закрепленные окна

Панели являются частью интерфейса инженерной среды ADISRA SmartView. Они могут «плавать» над программой, перетаскиваться за границы программы или оставаться фиксированными в одном из нескольких мест. (По умолчанию все окна закреплены.)

#### 6.2.1 Плавающие панели

Существуют два способа перемещения панелей в ADISRA SmartView:

- 1. Перемещать панель мышью, ухватив ее левой клавишей за заголовок.
- 2. Выбрав пункт Float из меню, открывающегося правой кнопкой.



После выбора пункта Float панель будет отделена от главного окна программы и пользователь может перетащить ее в любое место экрана с помощью мыши.

Чтобы вернуть плавающую панель в ее последнее закрепленное место в рабочей области, щелкните правой кнопкой мыши строку заголовка и выберите Show.



#### 6.2.2 Закрепление панели

Существуют два способа закрепить панель в ADISRA SmartView:

1. Правый клик на заголовке и выбрать пункт Show. Это вернет панель в последнее фиксированное положение.

Show	and the second se
Hide	
Dock as Tabbed Document	
Float	
Float	
	Show Hide Dock as Tabbed Document Float

2. Правый клик на заголовке и выбрать пункт Dock as Tabbed Document. Это поместит панель в середину рабочего пространства в виде вкладки в окне документа:



 Для закрепления окон всегда можно воспользоваться помощью Docking Navigator. Для этого достаточно перетащить любое окно левой клавишей мыши, ухватив его за строку заголовка, в одно из мест, указанных Docking Navigator, которые появляются на экране.



Пользователь может выбрать любое место на экране, указанное стрелками.



Чтобы закрепить панель как окно с вкладками, можно просто перетащить его в центральный квадрат Docking Navigator, как показано ниже.



#### 6.2.3 Отображение / скрытие панели

Чтобы скрыть панель, щелкните строку заголовка правой кнопкой мыши и выберите Hide.



Для того, чтобы вновь открыть ранее закрытую панель (будь то Navigation Tree, Logical Tree, Engineering Log, Build Output, Runtime Log, Search Output или Properties) в рабочей области, можно выбрать пункт View в верхнем меню и выбрать соответствующую скрытую панель.





#### 6.2.4 Автоматическое скрытие панели

Существуют два способа автоматически скрывать панели:

- 1. Правый клик на заголовке и выбор пункта Auto Hide.
- 2. Нажав крестик на заголовке.



Автоматическое скрытие панели уменьшает ее до вкладки, чтобы максимизировать рабочее пространство, как показано красными стрелками.



При наведении курсора мыши на вкладку временно восстанавливается ее полный размер. Щелчок или наведение курсора мыши за пределы панели возвращает ее в режим автоматического скрытия.

Чтобы восстановить панель в исходное состояние, пользователь может щелкнуть правой кнопкой мыши строку заголовка и выбрать пункт Show.





## 7. Создание нового проекта

Настало время поближе познакомиться со средой разработки и создать новый проект.

1. Откройте основное меню в верхнем левом углу окна.

Open	Recent Projects	-
New		
Save		
Save As		
Save All		
Save All for Web		
🐳 Exit		

2. Выберите пункт New или нажмите на клавиатуре комбинацию клавиш CTRL + N.

Open	Recent Projects	
New Ne	w (CTRL + N) ew Project	
Save As		
Save All		



3. Откроется новое окно Save As. Выберите расположение (папку) и имя для нового проекта. По умолчанию будет создан файл проекта с расширением .prj.

Примечание: Имена проектов не могут содержать специальные символы или пробелы.

we As					25	3	
Saveja	SuheProjec	h _	•	* •	]-	Contraction of the second s	
87						1 Andrews	
Recent							
TA							
Desktop							
-							
y cocaners							
31						1	
My Conputer							
	-	1000 acres			-		
My Network Places	File pane	Nex-Project			Şave		
10000	Save as type:	Project Files (".px)			Cancel		

В этом примере мы назвали наш проект GettingStarted. После этого нажмите кнопку Save.

4. Чтобы начать свой новый проект, ознакомьтесь с планировкой рабочего пространства. ADISRA SmartView отображает структуру проекта в дереве навигации, как показано ниже.



Как видно на изображении дерева навигации, некоторые компоненты создаются автоматически, например, теги (Tags), службы (Services), тревоги (Alarms), экранные формы (Graphics), библиотеки системных функций (System Function Library) и другие. Изображение выше представляет собой базовую структуру дерева навигации для любого проекта.

Вы можете использовать эти документы или можете создавать новые документы и настраивать их.



Справа находится типовой набор папок проекта. Каждая папка предназначена для связанных с ней документов.

Name	Date modified	Туре	Size
3 Alarms	10/20/2019 3:26 PM	File folder	
📕 Data Types	10/20/2019 3:26 PM	File folder	
🗼 Database	10/20/2019 3:26 PM	File folder	
📕 Driver	10/20/2019 3:26 PM	File folder	
DumpFiles	10/11/2019 12:00	File folder	
📕 Events	10/20/2019 3:26 PM	File folder	
FunctionLibrary	10/20/2019 3:26 PM	File folder	
📕 Graphics	10/20/2019 3:26 PM	File folder	
listory	10/20/2019 3:26 PM	File folder	
🕌 Images	10/20/2019 3:26 PM	File folder	
J Projectinfo	10/20/2019 3:26 PM	File folder	
Recipe	10/20/2019 3:26 PM	File folder	
🕌 Reports	10/20/2019 3:26 PM	File folder	
Jk Service	10/20/2019 3:26 PM	File folder	
📕 Symbol Library	10/20/2019 3:26 PM	File folder	
🕌 Tags	10/20/2019 3:26 PM	File folder	
🗼 Tunneling	10/20/2019 3:26 PM	File folder	
ADISRA-SmartViewDemo.prj	10/20/2019 3:27 PM	Adisra SmartView	9 KB
SmartView.CustomLogic.ScrBasicObjects.dll	10/10/2019 12:25	Application extens	4 KB
SmartView.CustomLogic.ScrFilling.dll	10/11/2019 1:04 AM	Application extens	4 KB
SVDatabase.accdb	10/11/2019 11:29	Microsoft Office A	808 KB

## 7.1 Открытие проекта

Для открытия ранее созданного проекта выполните следующие шаги:

- 1. Нажмите кнопку Menu.
- 2. Выберите пункт Open или нажмите на клавиатуре комбинацию клавиш CTRL + О.
- 3. Откроется новое окно, в котором необходимо открыть папку, соответствующую
- проекту, и кликнуть на файле с расширением .prj.
- 4. Нажать кнопку Open.



## 8. Документ тегов (Tags)

В ADISRA SmartView тегом называется уникальный идентификатор элемента данных (переменной) в приложении. Тег используется для чтения текущего значения из источника данных или записи значений в него.

В документе с названием Tags (теги) хранится перечень всех тегов, используемых в проекте. В документе Tags разработчик может добавлять или удалять теги, а также редактировать их свойства.

Для примера, давайте создадим несколько тегов, которые нам понадобятся для дальнейшей работы, и назначим им следующие свойства:

- Имя: TagBoolean, Тип: Boolean, Начальное значение: True
- Имя: TagInteger, Тип: Integer, Минимальное значение: Default, Максимальное значение: Default, Начальное значение: Default
- Имя: TagFloat, Тип: Float, Минимальное значение: -5000, Максимальное значение: 10000, Начальное значение: 250, Сохранять значение: Enabled
- Имя: TagString, Тип: String
- Имя: TagDateTime, Тип: DateTime
- Имя: Vector1Dimension, Тип: Float, Число измерений: 1, Размерность: 5
- Имя: Vector2Dimensions, Тип: Float, Число измерений: 2, Размерность: 2х3
- Имя: Vector3Dimensions, Тип: Float, Число измерений: 3, Размерность: 2x3x4
- Имя: VectorDynamic, Тип: Float, Число измерений: 1, Размерность: Dynamic

## 8.1 Создание нового списка тегов (Tags Document)

Чтобы создать новый документ с тегами, щелкните правой кнопкой мыши на папке Tags или любой вложенной папке. Вы также можете использовать уже имеющийся по умолчанию документ Tags. Выберите New Document или нажмите CTRL + N



В рабочем пространстве откроется новый элемент Tags1\*, пустой список тегов, пока в нем не будет создан новый тег.



## 8.2 Добавление новых тегов

Чтобы добавить в документ новый тег, откройте документ, куда Вы хотите добавить новые теги. В нашем примере мы будем использовать документ Tags1, созданный автоматической вместе с новым проектом.

Когда Вы откроете в среде разработки список тегов, лента меню обновится и в ней появится новая панель с названием Tags (см. рисунок ниже). На этой вкладке есть 4 кнопки: Insert Tag (Вставить тег), Remove Tag (Удалить тег), Import (Импорт) и Export (Экспорт).



Чтобы создать новый тег в документе, нажмите кнопку Insert Tag (Вставить тег). По умолчанию тег будет создан с именем NewTag1 (как показано на рисунке ниже). Поскольку в нашем примере нам потребуется 9 новых тегов, мы нажмем эту кнопку ещё 8 раз. Это создает новые Integer (Целочисленные) теги с именами от NewTag1 до NewTag9.

🎒 Tags1* 🗙			Ŧ
NewTag1	Integer	Array: 0 - No array	

## 8.3 Сохранение документа

После вставки новых тегов нам необходимо научиться сохранять изменения, внесенные в документ. Есть несколько способов сохранить документ:

- Кликнуть кнопку Меню (💌) и выбрать пункт Save выпадающего меню
- Нажать кнопку с диском (🚽) в панели быстрого доступа.
- Нажать комбинацию клавиш CTRL + S keys в рамках открытого документа.

Примечание: Когда документ был изменен, он помечается символом «\*» рядом с именем документа, как показано на предыдущем изображении выше, рядом с именем документа Tags1.



## 8.4 Конфигурирование тегов

Открыв список тегов, мы можем приступить к настройке каждого тега в соответствии со свойствами, описанными в пп. 8.2 и 8.3. Чтобы настроить первый тег, выполните следующие действия:

- 1. Выделите первую строку в документе (Выбрав NewTag1). После этого строка будет подсвечена голубым цветом.
- 2. Установим курсор в поле имени тега для его переименования.
- 3. Назовем его, например TagBoolean.
- 4. Кликните на меню выбора типа тега.
- 5. Установите тип Boolean для нашего первого тега.
- 6. В панели (окне) свойств (Property) установите «чекбокс» начального значения тега (Initial Value) в положение True.

剑 Tags1* 🗙				Proper	ties		* # ×
TagBoolean	Boolean	* Array: 0 - No arr		TProp	ertyBool:		
- Age - Contraction	overcert	and a read		2	Search		×
NewTag2	Integer	<ul> <li>Array: 0 - No arr</li> </ul>	ay ~	4 Ta	g an Name	New Tast	
NewTag3	Integer	~ Array: 0 - No arr	ay ~		nitial Value	True	
NewTag4	Integer	<ul> <li>Array: 0 - No arr</li> </ul>	ay ~	D	escription		
NewTag5	Integer	<ul> <li>Array: 0 - No arr</li> </ul>	ay *	0	PC Access	Read ~	
NewTag6	Integer	<ul> <li>Array: 0 - No arr</li> </ul>	ay ~	Cor	mmunication		
NewTag7	Integer	<ul> <li>Array: 0 - No arr</li> </ul>	ay -	S	erver: N	ione ~	
NewTag8	Integer	<ul> <li>Array: 0 - No arr</li> </ul>	ay ~	- AL	arm Config sise	Disabled	
NewTag9	Integer	<ul> <li>Array: 0 - No arr</li> </ul>	av ~	P.T	rue	Disabled	

Повторите эти шаги для других восьми тегов, которые мы создали для нашего примера в предыдущем разделе. После завершения конечный результат должен быть таким же, как показано на изображении ниже. Завершив настройку тегов, сохраните изменения.

J Tags1* X			Properties	1990
TagBoolean	Boolean	<ul> <li>Arrayi 0 - No array *</li> </ul>	TPropertyBool	>
Taginteger	Integer	<ul> <li>Array: 0 - No array ~</li> </ul>	4 Tag	
TagFloat	Float	<ul> <li>Array: 0 - No array</li> </ul>	Tag Name TagSoolean Initial Value True	
TagString	String	Array: 0 - No array: *	Description	1
TagDateTime	DateTime	<ul> <li>Array: 0 - No array: *</li> </ul>	OPC Access     Read      Retentive Value     Palse	
Vector1Dimension	Float	Y Arrayt 1 Dimension * Column(20) 5 Dynamic	Communication Server: None *	
Vector2Dimensions	Float	* Array: 2 Dimensions * Column(X): 2 Bow(Y): 3	Alarm Config     False Disabled	
Vector3Dimensions	Float	Array: 3 Dimensions      Column(X): 2 Row(V): 3 Plain(Z): 4	+ True Disabled	
VectorDynamic	Float	* Arrays 1 Dimension * 🖉 Dynamic		



## 9. Документ описания типа данных (Data Type)

Документ описания типа данных позволяет создавать сложную структуру (новый тип данных) путем группировки несколько различных типов тегов вместе. Например, внутри нового типа данных пользователь может иметь тег с плавающей запятой (типа Float), целочисленный тег (типа Integer), строковый тег (типа String) и т. д.

Здесь мы добавим и настроим теги, как показано ранее в разделе <u>Документ тегов</u>. Для нашего примера мы создадим тип данных под названием StructureExample, который будет содержать:

- Name: Member1, Type: Integer, Dimension: 1, Vector Size: 3
- Name: Member2, Type: Float, Dimension: 2, Vector Size: 3x3
- Name: Member3, Type: String, Dimension: 3, Vector Size: 3x3x3

Новый тип данных, который мы создали, теперь будет доступен при выборе типа тегов, которые мы создаем или редактируем в документе типа Tags.

# 9.1 Создание и сохранение документа описания типов данных (Data Type Document)

Мы продолжим наш проект, создав новый документ с описанием типов данных. Для этого щелкнем правой кнопкой мыши папку Data Type в дереве навигации и выберем New Document или нажмем CTRL + N.

В рабочем пространстве откроется вкладка Data\_Types1\*. Это будет пустой документ, пока мы не добавим в него новый тип данных. После этого нам останется только переименовать и сохранить его одним из способов, описанных в разделе <u>Сохранение</u> <u>Документов</u>. Дадим нашему документу имя StructureExample.

## 9.2 Добавление нового тега в документ описания типа данных

Чтобы вставить в документ новый тег, откроем его для редактирования. В нашем примере это будет документ с названием StructureExample.

Нажмем кнопку Insert Tag. По умолчанию тег создается с именем NewTag1. Так как нам требуется три новых тега, нажмем кнопку еще два раза, чтобы создать новые целочисленные теги с именами от NewTag1 до NewTag3.





## 9.3 Конфигурирование тегов в документе описания типа данных

Добавим в документ StructureExample три новых тега, переименуем их и присвоим им тепы данных таким образом, чтобы они соответствовали следующему изображению или выбранному нами ранее (в разделе 9) списку. А именно:

- Name: Member1, Type: Integer, Dimension: 1, Vector Size: 3
- Name: Member2, Type: Float, Dimension: 2, Vector Size: 3x3
- Name: Member3, Type: String, Dimension: 3, Vector Size: 3x3x3

Следуйте инструкциям в разделе <u>Конфигурирование тегов</u>. После настройки параметров экран должен выглядеть следующим образом:

StructureExample	r X		Properties     Properties     Properties     Properties
Member1	Integer	* Amay: 1 Dimension * Column(Q): 3 Dynemic	3 Search X
Member2	Float	* Amay: 2 Dimensions * Column(X): 3 Row(Y): 3	* Info Name Data_Types1
Member3	String	Amay: 3 Dimensions      Column(X): 3      Row(Y): 3      Plain(Z):	t 3

## 9.4 Добавление тегов нового (созданного ранее) типа в список тегов

Теперь, когда документ типа данных настроен, добавим новый тег в список тегов (в документ Tags). Назовем его Structure и присвоим ему новый тип данных: StructureExample.

Для этого выполним следующие шаги:

- 1. Откроем документ Tags, созданный нами в разделе Создание нового списка тегов.
- 2. Нажмем кнопку Insert Tag в ленте и добавим новый тег (NewTag1).
- 3. Переименуем его в Structure
- 4. Выберем для него созданный нами тип данных StructureExample.
- 5. Сохраним изменения.



На следующем изображении мы видим, что в документе Tags появился новый тег StructureExample, который мы назвали Structure. Кроме того, окно свойств ADISRA SmartView содержит поле со списком, в котором отображается текущая структура тега. В этом случае структура тега будет иметь три внутренних члена: Member1 - тип Integer, Member2 - тип Float и Member3 - тип String.

😏 Tags1* 🗙 🏠 Structu	reExample X		Properties      You      PropertyTagManager:
Tag8oolean	Boolean	* Array: 0 - No array *	7 Search
Taginteger	Integer	* Array: 0 - No array *	Selected Member: Structure
TagFloat	Float	<ul> <li>Arrey: 0 - No arrey</li> </ul>	4 Tag 4 Structure.Nember1
TagString	String	✓ Array: 0 - No array	Selecte Structure.Member1[1]
TagDateTime	DateTime	✓ Array: 0 - No array	Structure.Member1[2]  Structure.Member2  Structure.Member2[0, 0]
Vector1Dimension	Float		Structure.Member2[0, 1] Structure.Member2[0, 2] Structure.Member2[0, 2]
/ector2Dimensions	Float	* Array: 2 Dimensions * Column(X): 2 Row(Y): 3	Structure.Member2[1, 1] Structure.Member2[1, 2] Structure.Member2[1, 2]
Vector3Dimensions	float	Array: 3 Dimensions Column(R): 2 Row(Y): 3	Structure.Nember2[2, 1] Structure.Nember2[2, 2] # Structure.Nember3
VectorDynamic	Float	Array: 1 Dimension V V Dynamic	Structure.Member3[0, 0, 0 Structure.Member3[0, 0, 1
Structure	StructureExample	<ul> <li>Array: 0 - No array: *</li> </ul>	Structure.Member3(0, 0, 2 Structure.Member3(0, 1, 0



## 10. Службы (Services)

Документы типа Службы (Services) предназначены для того, чтобы содержать в себе фрагмент кода, который будет выполняться в течение определенного времени (в соответствии с настроенным условием) или каждом изменении значения тега, настроенном в документе.

Особенностью документов типа Сервисы является то, что вызов кода, описанного в документе, может быть выполнен из любого места ADISRA SmartView. Например, из события в графическом объекте, или из настроенного Триггера, или даже из созданной нами собственной библиотеки пользовательских функций. Мы можем создавать собственные сценарии с использованием языка программирования C# и использовать их позже в приложении!

Любой новый проект автоматически создает документ службы под названием Запуск. Этот документ запускается автоматически при каждом запуске приложения. В этом разделе показано, как это работает.

## Примечание: Документ Startup не может быть переименован или удален!

Для нашего примера (проекта GettingStarted) мы настроим документ Startup и создадим еще пару подобных документов с нужными нам скриптами. В одном из них мы настроим условие для запуска его скрипта, а ко второму привяжем один из наших тегов. И настроим все таким образом, чтобы скрипт выполнялся каждый раз при любом изменении значения этого тега.

## 10.1 Конфигурирование документа Startup (Запуск)

Для настройки документа Startup, выполним следующие шаги:

- 1. Развернем секцию Services в дереве навигации ADISRA SmartView чтобы найти документ с именем Startup.
- 2. Откроем его одним из следующих способов:
  - Двойной клик на документе Startup.
  - Правый клик на документе и выбор пункта Open из выпадающего меню.
  - Выбрать документ и нажать на клавиатуре комбинацию CTRL + О.
- После открытия документа мы можем начинать создавать свой первый сценарий. Наберем в качестве сценария следующий код:

## @TagDateTime = DateTime.Now; @TagString = Startup document executed successfully!

Этот фрагмент кода присвоит тегу TagDateTime значение времени на момент запуска документа Startup, а тегу TagString - текстовую строку, говорящую об успешном выполнении документа.



- 4. На ленте ADISRA SmartView появится новая панель Scripts (Скрипты) с кнопкой Verify (Проверить) с зеленой галочкой.
- 5. Теперь попробуем нажать на эту кнопку и проверить правильность нашего скрипта. В результате проверки мы получим ошибку.
  - Как можно понять, что ошибка есть? Рамка вокруг скрипта станет красной.

• Как можно понять, в чем именно наша ошибка? На панели Engineering Log (Технический журнал) отображается сообщение, указывающее, какая ошибка была обнаружена и где находится ошибка.

1 @TagDat 2 @TagStri	eTime = D ng = "Star	ateTim <b>tup d</b> e	e.Now; ocument executed su	ccessfully!"	
Line: 2					
Engineering L	og	-			<b>▼</b> ₽ ×
Description	Location	Line	Object	TimeStamp	
; expected	Startup	3	Kernel, Messages, Scrip	ptEditorReference 04/10/2019 10:33:43.711 PM	1
Search Output	ut Runtin	ne Log	Engineering Log		

6. Для исправления ошибки добавим точку с запятой в конец строки 2

@TagDateTime = D	ateTime.Now;		
@TagString = <mark>Start</mark>	up document ex	ecuted success	ully!;

7. Нажмем кнопку Verify еще раз и убедимся, что красная граница исчезла. Таким образом, это означает, что сценарий выполнен успешно.

* Startup* ×	
Service Settings Description:	
Service Script	
1 @TagDateTime = DateTime.Now; 2 @TagString = "Startup document executed successfully!";	

8. Сохраним после этого изменения в нашем проекте.



## 10.2 Создание документа службы (Services), с выполнением по условию

Для создания нового документа типа Services со скриптом, выполняющимся по условиям, сделаем следующие шаги:

- Создадим новый документ Services, щелкнув правой кнопкой мыши раздел Services в дереве навигации и выбрав в меню пункт New Document. Или выберите раздел Services и нажмите CTRL + N. По умолчанию имя созданного документа служб будет называться Services1\*.
- 2. В свойстве Туре (Тип) созданного документа выберите параметр Condition (Условие). Это означает, что документ будет выполняться только тогда, когда настроенное условие истинно. По умолчанию начальное значение условия установлено в значение Истина.
- 3. В текстовом поле Condition (Условие) введем следующее условие: @TagBoolean. С этой настройкой документ будет выполняться каждый раз, когда значение логического тега равно True.
- 4. Нажмем кнопку Verify C справа от поля условий для проверки правильности нашего условия.
- 5. В поле Interval(ms) установим временной интервал, через который документ будет проверять данное условие. В нашем примере установим значение 800.
- 6. В области сценария служб напишем следующий код:

@DateHourCurrent = DateTime.Now;

7. Нажмем знакомую нам кнопку Verify на ленте и увидим сообщение о том, что тега с именем @DateHourCurrent не существует. Так и есть. Но нам будет предложено его автоматически создать, если мы захотим



8. Поскольку мы действительно хотим использовать этот тег в нашем проекте, создадим его, нажав кнопку Yes.



9. Появится новое окно, в котором мы сможем настроить свойства нового тега. Для этого проекта создадим тег в документе Tags1 с типом DateTime и размерностью равной 0 (одиночный тег).

New Tag		-		×
Name:	DateHourCurrent		]	
Doc:	TAGS1	J		
Type:	DateTime	v		
	Array: 0 - No Arra	у ~		
	OK	Cancel	1	

- 10. Нажмем ОК для подтверждения.
- 11. Нажмем кнопку Verify на ленте и убедимся, что на это раз ошибок нет.
- 12. Сохраним изменения в нашем документе.

Ocument Se	ettings				
Enable:	TRUE	. *			
ervice Settin	gs				
Type:	Condition ~	@TagBoolean	$\checkmark$	Interval(ms):	800
Description	:				
ervice Scrip	t				

На этом настройку скрипта с запуском по условию будем считать законченной.

## 10.3 Создание документа службы с запуском по триггеру

Для создания нового документа типа Services со скриптом, выполняющимся по триггеру, сделаем следующие шаги:

- 1. Создадим новый документ Services и назовем его ServicesTrigger.
- 2. В поле типа (Туре) выберем опцию Trigger.



- 3. В поле триггера (Trigger) укажем тег @TagInteger.
- 4. Нажмем Verify и убедимся, что пока все правильно.
- 5. После настройки триггера добавим собственно скрипт, который будет выполняться, введя следующий текст в поле Service Script:

SVApplications.Output (Trigger Service Successfully Run);

6. Сохраним изменения в документе.

Service:	sTrigger* ×	
Document Se	ettings	
Enable:	TRUE v	
Service Setting	gs	
Type:	Trigger v @TagInteger	
Description		
ervice Script	t	
McColation	Result: Script Verified is OK	

Результатом выполнения этого скрипта будет сообщение в Инженерном журнале (Engineering Log). Такие сообщения бывают полезными для регистрации того, что выполняется в проекте.



## 11. Создание документа экранных форм

После того как мы создали и настроили скрипты, пришло время создать графический экран, где мы будем запускать скрипты и отслеживать их исполнение.

## 11.1 Установка разрешения по умолчанию для графики

Установим разрешение по умолчанию 1014 x 674 (ширина x высота) для всей графики, созданной начиная с данного момента. Для этого:

1. Выберите в главном меню Settings (Настройки), и в ленте найдите группу Resolution (Разрешение). Щелкните стрелку вниз, чтобы открыть окно под ним.



- 2. Найдем поле, обведенное красным на рисунке выше. Установим тип Custom (Пользовательский), и в поле Width (Ширина), отмеченном буквой W, введите значение 1014.
- 3. В поле Height (Высота), отмеченном буквой Н, введем значениее 674.

В результате все графические документы, создаваемые после этого, будут иметь эти размеры.



## 11.2 Создание графической формы для запуска и проверки скриптов

Для создания экранной формы для запуска и проверки скриптов сделаем следующее:

- Чтобы лучше организовать наши экранные формы, создадим сначала папку для хранения наших документов. Для этого щелкнем правой кнопкой мыши на разделе Graphics (Графика) в дереве навигации.
- 2. В выпадающем меню выберем опцию New Folder (Новая папка).


3. Будет создана новая папка с названием NewFolder. Переименуем ее в Services и нажмем Enter.



- 4. Теперь создадим новый документ, кликнув павой кнопкой мыши на папке Services.
- 5. Выберем опцию New Document из выпадающего меню.
- 6. Будет создан новая экранная форма с названием Graphics1.
- 7. Первым объектом, который мы добавим наш экран, будет объект типа GroupBox. Этот элемент разграничивает области и организует объекты на экране. Чтобы создать GroupBox, найдем и выберем его на ленте в группе Interface Objects.
- 8. В рабочей области экранной форму кликнем и перетащим указатель мыши так, чтобы создать прямоугольник GroupBox нужного размера.

Graphics1* ×	 	_	_	_	_	_	_	_	_				
	 				ere.						 		
	 			-			-						
· · · O · · · · · · · · · · · · · · · ·	 · O			-			-			-			• •
GroupBox	 ÷., '			-						-			
	- I+			-						-			
	- I+		• •	-	• •		-	• •		-		-	
	- I*			-	• •		-	• •		-			
	- P			-			-			-			• •
	11			-									• •
1114	Ь			-			-						• •
	-۲			-						-			
	- I*		• •	-	• •		-	• •		-		-	
	- I*			-			-	• •				-	
	- I*			-			-			-			
	11			-									• •
	11			-			-						
				-						-			
0	 .0		• •		• •					-		-	
	 			-	• •		-	• •		-			
	 			-									
	 						-			-			

- 9. После создания объекта GroupBox настроим его размер и положение на экране. Для этого выделим созданный объект, щелкнув по нему левой кнопкой мыши.
- 10. В окне свойств найдем поле Size (Размер). Если область не открыта, щелкнем стрелочку, чтобы отобразить поля Width и Height (Ширина и Высота).
- 11. Поместим курсор в поте Width и установим значение ширины в 250. В поле Height (Высота) установим значение 150.
- 12. Находясь в окне свойств, найдем метку Location (Местоположение). Если область не открыта, щелкнем стрелочку рядом с меткой.
- 13. В поле Тор (Верх) введем значение 10. В поле Left (Лево) значение 20.
- 14. Теперь настроим текст, отображаемый внутри объекта. Для этого изменим значение в поле Text (Текст). Введем значение Service Startup (Запуск службы).



Graphics1* ×	₽ Properties	<b>*</b> 0 3
	Group Box: Graphic	s.Objects.Properties.GValidatedP
O O O	? Search	×
	Info	
	Name Name	GroupBox1
(c p)	Location	(10,20)
	Тор	10
	Left	20
u u	- Size	(250,150)
	Width	250
	Height	150
	Z-Index	0
	Draggable	True
	Configuration	
	Text	Service Startup

- 15. В созданном нами GroupBox отобразим значения тегов TagDateTime и TagString для отслеживания выполнения запуска скрипта. Для этого создадим два объекта Label и два объекта Text Box. Чтобы создать новый объект Label, выберем его в ленте, в группе Basic Objects. Щелкнем в любом месте документа, чтобы создать объект (по умолчанию он будет размером 80 x 25).
- 16. Выбрав объект Label, настроим его размер и расположение. Для этого найдем свойство Size; в поле Width введем значение 85, а в поле Height значение 16. В поле Тор установим значение 65, а в поле Left значение 30.
- 17. В поле Text введем значение TagDateTime.



18. Теперь создадим объект типа TextBox для того, чтобы отобразить текущее значение тега. Найдем и выделим в ленте элемент TextBox в группе Basic Objects.



- 19. После создания объекта TextBox настроим его размеры и местоположение. В поле Width (Ширина) установим значение 150, а в поле Height (Высота) значение 25. В поле Тор установим значение 60, а Left 110.
- 20. В поле Text введем значение @TagDateTime. Символ @ указывает, что мы хотим отображать в объекте значение тега, а не просто текст «*TagDateTime*».

Graphics1* ×		Properties	* 3
		Text Box: Graphics	Objects.Properties.GValidated
-Service Startup		? Search	
10000000000000000000000000000000000000		Info	
TaoDateTimeD @TaoDateT		Name	TextBox1
0	0 0	<ul> <li>Location</li> </ul>	(60,110)
		Тор	60
		Left	110
		▶ Size	(150,25)
		Z-Index	2
		Draggable	True
		Configuration	
		Text	@TagDateTime

- 21. А теперь повторим аналогичные процедуры для следующих объектов:
  - Label = TagString, Width = 85, Height =16, Top = 100, Left = 30, Text: TagString
  - TextBox = TagString, Width = 150, Height = 25, Top = 95, Left = 110, Text:@TagString

Graphics1* ×			_	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_	Ŧ	Properties Text Box: Graphic	s.Objects.Properties.0	♥ Ø SValidatedPro
-Service Startu	0	111						11			11		11								7 Search		×
10000000000						11		14				1	1				14				Info		
TaoDateTime	@TagDateTime						1	11			11		ł			1	5	2	-		Name	TextBox2	
0.004.000.0000	0 0	0				11		ii.	1		11		i.								<ul> <li>Location</li> </ul>	(9	5,110)
TagString	C @TagString	0																			Тор	95	
1						11	11	i i	1		1		Ì	1		0	1				Left	110	
						: :	1	: :			: :				: :	1					Size	(150,25)	
											1	1		1	-			1			Z-Index	4	
200002220000									i.										id.		Draggable	True	
	$\begin{array}{cccccccccccccccccccccccccccccccccccc$	::::	11		5								11								Configuration		
111111111		111	111	11		11	11	* *	1	11	11	11	* *			1	11	11	181		Text	@TagStri	ing

- GroupBox = Service Condition, Width = 250, Height = 150, Top = 10, Left = 380, Text: Service Condition
- Label = TagBoolean, Width = 80, Height = 16, Top = 65, Left = 390, Text: TagBoolean



- TextBox = TagBoolean, Width = 150, Height = 25, Top = 60, Left = 470, Text: @TagBoolean
- Label = DateHourCurr, Width = 85, Height = 16, Top = 100, Left = 390, Text: DateHourCurr
- TextBox = DateHourCurrent, Width = 150, Height = 25, Top = 95, Left = 470, Text:
   @DateHourCurrent

Graphics1" ×					Ŧ	Properties	~ ‡ 3
					1	Text Box: Graphics.0	bjects.Properties.GValidatedPro
-Service Startup	+1.4 + 4.4 + 1.7 + 1.1 + 1.1		-Service Conditi	on	11.00	? Search	×
						Info	
TagDateTime	@TagDateTime	100000000	Tag8oolean	@TagBoolean		Name	TextBax4
			0	0	P	Location	(95,470)
TagString	@TagString		DateHourCuri©	@DateHourCurrent	0	Тор	95
			12.5			Left	470
111111111	111111111111111		111111111	111111111111	1111	Size	(150,25)
						Width	150
						Height	25
						Z-Index	10
						Draggable	True
						Configuration	
						Text	@DateHourCurrent

- GroupBox = Service Trigger, Width = 250, Height = 150, Top = 10, Left = 744, Text: Service Trigger
- Label = TagInteger, Width = 80, Height = 16, Top = 65, Left = 753, Text: TagInteger
- TextBox = TagInteger, Width = 150, Height = 25, Top = 60, Left = 833, Text:
   @TagInteger

Graphic	st* x				Properties	
	05.—331 h				Label: Graphics.Obje	cts.Properties.GValidatedPr
110000000				neede oor oo	Ta level	
-Service Conditi	on		-Service Trigg	er	7 Search	
			0.000000255	SUL	Info	
			0 0	0	Name	Label5
TagBoolean	@TagBoolean		OTagInteger	@@TagInteger	Hume	and an and a second
			0 0	0	Location	(65,753)
DateHourCurr	@DateHourCurrent			1.1	-	er
				12	Тор	65
		1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1	1	1.1	1.0	753
			1		Leit	1.02
1					CALCUMPT	(90.16)
					- Sec	(00,10)
k + = + k + + k	· · · · · · · · · · · · · · · · · · ·	$\mathbf{r}_{i} = \mathbf{r}_{i} + \mathbf{r}_{i} $	$a_{i}(x_{i}) = a_{i}(x_{i}) + \cdots$	(x) + (x)	UL MA	80
					Proof.	
						16
					Height	10
					Z-Index	11
					E Press	
1.4.1.0.0000111.0	CONTRACTOR OF A DESCRIPTION OF A DESCRIP	CARD CARD CARD CARD CARD CARD	CONTRACTOR NO.	CONTRACTOR CONTRACTOR CONTRACTOR	Draggable	True
*******						and the
					Configuration	
						Free contractions and the
					Text	TagInteger
					1 100 10	



22. После завершения вышеуказанных настроек сохраним графический документ и назовем его Service.

В результате наша экранная форма должны выглядеть примерно так:

- PService Starts		 -Service Condition	ton -		Service Tripper		
Tubullar	@ZeeDeteDees	 	(B7 and and and		Technical	#Furbines	18
	a apar na	 - 4,000000	a aproven	-	also alla	a shorthe	- C
: TagString	@TegString	DeteHourCart	BDeteHourCurrent				

22. После чего сохраним наш экран под именем Service.

				and the second se			00000	a 🍠 Taço
-Service Trigger	-Service Trigger		ice Condition	-54010	211	vice Startup	-54	1 Tagi
Taginteger @Taginteger	Taginteger	oolean	Boolean @TagRool	TegBo	4	DataTime @TagDateTive	74	a 🖉 Data Typec
		- marcune at	Hand or ODderin	Datab		Charles (Charling	80	4 Crachia
	11 (11 ) 11 (13	10000000	andarcar arotte to	Liest		bread an indicated	3 1	A Services
1					1945			Templates
							_	



# 12. **Триггеры** (Triggers)

Документы типа Triggers (Триггеры) содержат элементы (Триггеры), которые определяют условия запуска либо встроенных, либо внешних скриптов (Services Documents). Такие как: изменения значений тегов, временные интервалы или наступление определенной даты. Ниже опишем, как настроить каждый такой случай, и объясним это более подробно.

# 12.1 Создание новых тегов для запуска триггеров (Trigger Tag)

Создадим тег типа Float с именем Triggers, размерностью 1 и размером в 4 элемента.

Откроем список тегов Tags1, кликнем кнопку Insert Tag в ленте и переименуем новый тег в Triggers. Затем изменим тип тега на Float, изменим размерность массива (array Size) на 1, и установим размер (Column(X)) в 4 элемента.

				0		
Triggers	Float	~ 1	Array:	1 Dimension ~	Column(X):	4

Повторим подобные процедуры для следующих новых тегов:

- TagTrigger: Type Integer, array size 0 No array
- TagCondition: Type Integer, array size 0 No array

TagTrigger	Integer	×	Array:	0 - No array 👻
TagCondition	Integer	*	Array:	0 - No array 👻

# 12.2 Создание простых триггеров (Trigger)

Элемент списка Триггеров типа Trigger срабатывает при любом изменении значения указанного в нем тега или выражения. Для примера, создадим его таким образом:

- 1. Создадим новый список триггеров. Для этого кликнем правой кнопкой на Triggers и выберем New Document. По умолчанию новый документ получит название Triggers1.
- 2. В поле Enable вверху списка триггеров (документ Triggers1) выберем TRUE.
- 3. В верхней ленте, в блоке Triggers, нажмем кнопку Insert Trigger.
- 4. В поле Туре созданного элемента выберем его тип: Trigger.
- 5. В поле Action Type выберем вариант Expressions.
- 6. В текстовом поле Trigger напишем @TagTrigger.



7. В поле Expression введем следующее выражение:

@Triggers[0] = @Triggers[0] + 1;

8. Сохраним изменения в документе.

	en type myger woon type opresens myger regingger woon en grigger
Type:	Trigger * Action Type: Expressions *
Trigg	er: TagTrigger
Expre	ession: vrs[0] = @Triggers[0] + 1;
	Script Name: Trigger
	Script Name: Trigger
	r emgandol - emgandol + r
	Line: 1
	Validate OK Cancel



# 12.3 Создание триггеров с условием (Condition Trigger)

Триггеры типа Condition (Условие) срабатывают, когда определяющее их выражение принимает значение Истина. Рассмотрим создание такого триггера на примере:

- 1. В открытом списке триггеров Triggers1 кликнем в ленте Insert Trigger.
- 2. В поле Туре созданного элемента выберем его тип: Condition.
- 3. В поле Action Type выберем опцию Expressions.
- 4. В поле Enable установим значение TRUE.
- 5. В поле Condition введем следующее выражение: @TagCondition> 1000.
- В поле Expression введем следующее выражение: @Triggers[1] = @Triggers[1] + 1;
- 7. Сохраним изменения в документе.

Documer	it Settings
Enable:	TRUE ~
Triggers I	tems
Type Trig	vent Type: Trigger Action Type: Expressions Trigger: TagTrigger Action: @Triggers[0] = ( e: Trigger Action Type: Expressions  ger: TagTrigger
	vent Type: Condition Action Type: Expressions Condition: @TagCondition> 1000 Action: @Triggers[1]
Type Con	e: Condition * Action Type: Expressions * dition: @TagCondition> 1000
Type Con Expr	e: Condition × Action Type: Expressions × dition: @TagCondition > 1000 ✓ ression: rs[1] = @Triggers[1] + 1;
Type Con Expr	e: Condition × Action Type: Expressions × dition: @TagCondition> 1000 v ression: !rs[1] = @Triggers[1] + 1; Script Name: Trigger
Type Con Expr	e: Condition × Action Type: Expressions × dition: @TagCondition> 1000 ression: !rs[1] = @Triggers[1] + 1; Script Name: Trigger
Type Con Expe	e: Condition × Action Type: Expressions × dition: @TagCondition > 1000 ression: !rs[1] = @Triggers[1] + 1; Script Name: Trigger Script Name: Trigger 1 @Triggers[1] = @Triggers[1] + 1;
Type Con Expr	e: Condition V Action Type: Expressions V dition: @TagCondition> 1000 V ression: !rs[1] = @Triggers[1] + 1; Script Name: Trigger Script Name: Trigger 1 @Triggers[1] = @Triggers[1] + 1; Line: 1
Typi Con Expr	e: Condition Validate OK Cancel



# 12.4 Создание триггеров по календарной дате/времени (Calendar Trigger)

Триггеры типа Calendar (по календарной дате) срабатывают, когда достигается определенная дата/время. Рассмотрим создание такого триггера на примере:

- 1. В открытом списке триггеров Triggers1 кликнем в ленте Insert Trigger.
- 2. В поле Туре созданного элемента выберем его тип: Calendar.
- 3. В поле Action Type выберем опцию Expressions.
- 4. В поле Enable установим значение TRUE.
- 5. Выберем опцию Daily. Эта опция позволит триггеру срабатывать ежедневно в определенное время.
- 6. В поле DateTime введем: 01/01/2020 12:00:00 PM. Это будет обозначать полдень 1 Января 2020.
- 7. В поле Recur Every \_\_\_ day (s) (Повторять каждые \_ дней) введем 1. Это будет обозначать, то триггер будет срабатывать ежедневно. Если мы в этом поле установим 2, то триггер будет срабатывать каждые 2 дня, и так далее.
- 8. В поле Expression введем следующее выражение:

@Triggers[2] = @Triggers[2] + 1;

9. Сохраним изменения в документе.



Enable: TRUE   Triggers Items  Trigger Trigger Action Type: Expressions Trigger: TagTrigger Action: @Triggers[0] =  Type: Trigger Action Type: Expressions  Trigger: TagTrigger  Expression:  Script Name: Trigger  Script Name: Trigger  Script Name: Trigger  ( Priggers[2] = @Triggers[2] + 1;  Triggers[1  Tri	Document Settings			
Triggers Items         Image: Trigger       Action Type: Expressions         Trigger: TagTrigger       Action Type: Expressions         Trigger: TagTrigger       Script Name: Trigger         Expression: #       Script Name: Trigger         Script Name: Trigger       I @Triggers[2] + 1;         Triggers[2] = @Triggers[2] + 1;       Triggers[1]         Image: TagTrigger       Image: Trigger         Image: Triggers[2] = @Triggers[2] + 1;       Triggers[1]	Enable: TRUE	. w.		
Event Type: Trigger   Type: Trigger   Trigger: TagTrigger   Trigger: TagTrigger   Expression: **   Script Name: Triggers[2] = @Triggers[2] + 1;   Script Condit Line:   Line: 1	Triggers Items			
Type: Trigger   Trigger: TagTrigger   Trigger: TagTrigger   Expression: ir   Script Name: Triggers   Script Name: Triggers[2] + 1;     Type: Condit   Line: 1     Validate OK	Event Type: Tr	gger Action Type: Expressions Trigger: T	TagTrigger Action: @Trigg	ers[0] = @
Trigger: TagTrigger   Expression: **   Script Name: Triggers   Script Name: Triggers[2] = @Triggers[2] + 1;     Type: Condition:   Condition: Line:     Validate OK	Type: Trigger	<ul> <li>Action Type: Expressions</li> </ul>		
Expression:       Image: Script Name: Trigger         Script Name: Trigger       Script Name: Trigger         Script Name: Trigger       Image: Triggers[2] = @Triggers[2] + 1;         Type:       Condit         Condition:       Line: 1	Trigger: TagTrig	ger		
Script Name: Trigger         Image: Script Name: Triggers[2] = @Triggers[2] + 1;         Type: Condit         Condition: Image: Script Name: Triggers[2] = @Triggers[2] + 1;         Triggers[1]         Image: Script Name: Triggers[2] = @Triggers[2] + 1;         Triggers[1]         Image: Script Name: Triggers[2] = @Triggers[2] + 1;         Triggers[1]         Condition: Image: Script Name: Triggers[2] = @Triggers[2] + 1;         Triggers[1]         Validate         OK	Expression: #	Script Name: Tri	igger 💌	
Image: Sevent Type:     Image: Sevent Type:     Image: Sevent Type:     Image: Sevent Type:     Image: Triggers[2] = @Triggers[2] + 1;       Type:     Condit     Image: Sevent Type:     Image: Sevent Type: Sevent Type:     Image: Sevent Type: Se		cript Name: Trigger		
Type: Condit Condition: C Line: 1	Event Type:	@Triggers[2] = @Triggers[2] + 1;	Tri	ggers[1] :
Condition:	Type: Condit			
Validate OK Cancel	Condition 6	ne: 1		
Valuate OK Carter	Condition:		Validate OK Cancel	
Expression:	Expression:			
	Event Type:	endar Action Type: Expressions Calendar:	01/01/2020 12:00:00 AM Action: @Trig	gers[2] =
Event Type: Calendar Action Type: Expressions Calendar: 01/01/2020 12:00:00 AM Action: @Triggers[2]	Type: Calendar	Action Type: Expressions      Y		
Event Type: Calendar Action Type: Expressions Calendar: 01/01/2020 12:00:00 AM Action: @Triggers[2]     Type: Calendar      Action Type: Expressions	DateTime: 01/0	/2020 12:00:00 PM 🔷 🔹 💿 Daily 🔿 We	eekly 🔘 Monthly	
Event Type: Calendar Action Type: Expressions Calendar: 01/01/2020 12:00:00 AM Action: @Triggers[2]     Type: Calendar      Action Type: Expressions      DateTime: 01/01/2020 12:00:00 PM      O Daily      Weekly      Monthly	Daterine: 01/0			
Event Type: Calendar Action Type: Expressions Calendar: 01/01/2020 12:00:00 AM Action: @Triggers[2]     Type: Calendar      Action Type: Expressions      DateTime: 01/01/2020 12:00:00 PM      O Daily      Weekly      Monthly     Recur Every: 1 day(s)	Recur Every: 1	day(s)		
Event Type: Calendar Action Type: Expressions Calendar: 01/01/2020 12:00:00 AM Action: @Triggers[2] Type: Calendar      Action Type: Expressions      DateTime: 01/01/2020 12:00:00 PM      v     O Daily      Weekly      Monthly Recur Every: 1 day(s) Expression: :rs[2] = @Triggers[2] + 1;	Recur Every: 1	day(s) = @Triggers[2] + 1:		

## 12.5 Создание интервальных триггеров (Interval Trigger)

Триггеры типа Interval (интервальный) срабатывают периодически, с определенным интервалом. Рассмотрим создание такого триггера на примере:

- 1. В открытом списке триггеров Triggers1 кликнем в ленте Insert Trigger.
- 2. В поле Туре созданного элемента выберем его тип: Interval.
- 3. В поле Action Type выберем опцию Expressions.
- 4. В поле Enable установим значение TRUE.
- 5. В поле Interval field установим значение 00:01:00. Это значит, что триггер будет срабатывать каждую минуту.



6. В поле Expression введем следующее выражение:

@Triggers[3] = @Triggers[3] + 1;

7. Сохраним изменения в документе

После завершения этих настроек документ должен выглядеть, как на изображении ниже:

Type: Trigge	r Action Type: Expressions -	
Trigger: Tag	Trigger	
Expression	rs101 = @Triagers101 + 1	
- copression		
<ul> <li>Event Type</li> </ul>	Condition Action Type: Expressions Condition: @TagCondition> 1000 Action:	@Triggers[1]
Type: Condi	tion Action Type: Expressions	
Condition:	Script Name: Trigger	
Expression:	Script Name: Trigger	
	1 @Triggers[3] = @Triggers[3] +1;	
<ul> <li>Event Typ</li> </ul>		Triggers[2]
Type: Caler		
DateTime:	Line: 1	
D	Validate OK Cancel	
Necur Everys		
Expression:	rs[2] = @Triggers[2] + 1;	
<ul> <li>Event Type</li> </ul>	: Interval Action Type: Expressions Interval: 00:01:00 Action: @	Triggers[3] =
Type: Interv	al * Action Type: Expressions *	
Interval: 00:	01:00	
Even view D	vr[2] = @Trioparr[2] + 1	
expression: e	ustol = @.unddeustol + i;	



## 13. Настройка экранных форм для визуализации триггеров

Создадим экранную форму для отображения триггеров. Сделаем это аналогично тому, как мы создавали экранную форму для скриптов. Выполним следующие действия, чтобы создать объект GroupBox для определения каждого триггера, метку с именем тега и текстовое поле, отображающее значение тега.

- 1. Создадим новую папку Triggers, кликнув правой кнопкой на разделе Graphics в дереве навигации и выбрав пункт New Folder. Переименуем папку в Triggers.
- 2. Создадим новый документ в этой папке. По умолчанию он буде назваться Graphics1 и иметь размеры 1014 х 674.
- 3. Откроем экранную форму Service.
- 4. В рабочей области экранной формы Service выделим (нажав и удерживая левую клавишу мыши область экрана вокруг объекта) объект GroupBox с именем Service Startup. После отпускания клавиши мыши объект останется выделенным, как показано на рисунке ниже.



- 5. Кликнем правой клавишей мыши на выделенном объекте и выберем в контекстном меню вариант Сору (или просто нажмите CTRL + C).
- 6. После этого перейдем в новый документ Graphics1.
- 7. Кликнем правой кнопкой мыши в области точки с координатами (0;0) в верхнем левом углу документа.
- 8. Выберем вариант Paste из контекстного меню. Объект, скопированный из экрана Service продублируется в экране Graphics1.
- 9. Переименуем скопированные объекты следующим образом:



- GroupBox: Text: Trigger Trigger
- Label1: Text: TagTrigger
- TextBox1: Text: @TagTrigger
- Label2: Text: Triggers[0]
- TextBox2: Text: @Triggers[0]

10. Повторим процесс для других триггеров, чтобы они соответствовали рисунку ниже.

11. Сохраним экранную форму под именем Triggers.

Getungstarteu				
Data Types	Trigger Trigge	r	 -Trigger Condit	on
Service	TagTrigger	@TagTrigger	TagCondition	@TagCondition
Triggers	Triggers[0]	@Triggers[0]	Triggers[1]	@Triggers[1]
Alarm History Tag History Recipes Reports	-Trigger Calenc	lar —	 —Trigger Interva	I] :
Tunneling Drivers Unages User Functions Library System Functions Library	Triggers[2]	@Triggers[2]	Triggers[3]	@Triggers[3]



# 14. Журнал Тревог (Alarm History)

Документ Alarm History (история или журнал тревог) содержит настройки сохранения истории тегов. Его можно сохранить либо во встроенной базе данных, либо во внешней реляционной базе данных, подключение к которой должно быть предварительно настроено в меню Settings в верхней панели -> Database Connections.

#### Варианты подключения:

C Alarm_I	History1* ×				
Document Se	ettings				
Save Type:	Proprietary	٣	Connection Item:	<u>×</u>	
	Proprietary				
History Item	Database				

Выбор реляционной базы данных из числа сконфигурированных ранее:

Document S	lettings			
Save Type:	Database	. 4	Connection Item:	
		11-		PostgresSQL001

Примечание: Один и тот же тег невозможно добавить в два разных документа.

#### 14.1 Конфигурирование тревог тегов

Понятие Тревога (Alarm) широко используется в SCADA системах для дополнительного контроля значений тегов. Например, если значение тега, связанного с некоторым параметром техпроцесса, выходит за пределы допустимого диапазона, наше приложение SCADA должно уведомить оператора об этой опасности. В среде ADISRA SmartView мы можем: создавать тревоги тегов, сохранять их в базе данных и отображать их на экранных формах, позволяя оператору их наблюдать, квитировать (подтверждать получение) предпринимать дополнительные меры по устранению причин возникновения тревоги. С помощью журнала тревог можно будет восстановить и документировать в виде отчета историю возникновения любого нештатного события, проанализировать все относящиеся к событию тревоги, а также реакцию на них и все последующие действия операторов.

Теперь настроим тревоги для пяти тегов. Эти тревоги мы позже будем выводить с помощью объекта Окно вывода тревог (Alarm Object), когда значения некоторых тегов будут меньше определенного нами уровня. Настроим массив наших тревоги в виде массива тревог Vector1Dimension, выполнив следующие действия:

- 1. Найдем папку Tags в дереве навигации и откроем документ Tags1.
- 2. Выберем документ Vector1Dimension.
- 3. В окне свойств (Property Window) выделим нужный элемент для настройки, начав с нулевого (0).





- 4. В поле LoLo (нижний предел) настройки тревог (Alarm area), установим галочку IsEnabled в значение True. Это активирует остальные поля.
- 5. В поле Limit (предел) введем -1000.
- 6. В поле Message (сообщение) введем текстовое сообщение: LoLo Tag Vector1Dimension[0].
- 7. В поле Priority установим значение -1. (как на картинке ниже)

4 AL	arm	
	LoLo	Enabled: -1000
	Is Enabled	True
	Limit	-1000
	Message	ctor1Dimension[0]
	Priority	-1

- 8. После этого настроим остальные теги (1...4) VectorDimensions. Настроим другие элементы следующим образом:
  - Vector1Dimension[1]: LoLo Limit: -750; Message: LoLo Tag Vector1Dimension[1]; Priority: -2
  - Vector1Dimension[2]: LoLo Limit: -500; Message: LoLo Tag Vector1Dimension[2]; Priority: -3
  - Vector1Dimension[3]: LoLo Limit: -250; Message: LoLo Tag Vector1Dimension[3]; Priority: -4
  - Vector1Dimension[4]: LoLo Limit: -100; Message: LoLo Tag Vector1Dimension[4]; Priority: -5
- 9. Сохраним изменения в этом документе.

Типы тревог, доступные для конфигурирования:



- LoLo: Тревога, наступающая, когда значение наблюдаемого тега становится МЕНЬШЕ ПРЕДЕЛЬНО ДОПУСТИМОЙ ВЕЛИЧИНЫ, установленной в поле Limit. Обычно эта величина устанавливается МЕНЬШЕ, чем величина для тревоги типа Lo.
- Lo: Тревога, наступающая, когда значение наблюдаемого тега становится МЕНЬШЕ НИЖНЕГО ПРЕДЕЛА НОМИНАЛЬНОГО ДИАПАЗОНА, установленного в поле Limit.
- Ні: Тревога, наступающая, когда значение наблюдаемого тега становится БОЛЬШЕ ВЕРХНЕГО ПРЕДЕЛА НОМИНАЛЬНОГО ДИАПАЗОНА, установленного в поле Limit.
- НіНі: Тревога, наступающая, когда значение наблюдаемого тега становится БОЛЬШЕ ПРЕДЕЛЬНО ДОПУСТИМОЙ ВЕЛИЧИНЫ, установленной в поле Limit. Обычно эта величина устанавливается БОЛЬШЕ, чем величина для тревоги типа Hi.
- Deviation: Тревога, наступающая, когда значение наблюдаемого тега изменяется на величину, большую, чем установлено в поле Variance за время меньшее, чем установлено в поле Time To Level Up (sec). Тревога сбрасывается, если тег за время, большее, чем Time To Level Up не изменится на величину, большую, чем установлено в поле Variance.
- Freeze: Тревога, наступающая, когда значение наблюдаемого тега не меняет своего значения в течение более длительного времени, чем установлено в поле Time (sec).
- Watch Dog: Тревога, наступающая, когда значение наблюдаемого тега изменяется. Тревога сбрасывается, если значение тега не изменяется дольше, чем время, установленное в поле Time (sec).



## 14.2 Настройка элементов журнала тревог (Alarm History Items)

Журнал тревог представляет собой базу данных, в которой сохраняются аварийные события. Мы можем выбрать встроенную или внешнюю реляционную базу данных.

После того, как у нас уже есть настроенные Тревоги, настойка Журнала Тревог (Alarm History) может быть сделана следующим способом:

- 1. Найдем раздел Alarm History в дереве навигации. Кликнем на нем правой кнопкой мыши и выберем пункт New Document. Будет создан новый документ с именем по умолчанию Alarm\_History1.
- 2. Установим тип этого документа как Proprietary

Alarm_	History1* ×		
Document S	ettings		
Save Type:	Proprietary	~	

3. Кликните кнопку Insert Alarm в ленте.



- 4. В поле Tag нового элемента введем значение @Vector1Dimension[0].
- 5. Установим галочку Never delete history для предотвращения удаления этой тревоги из журнала.
- 6. В поле Numeric выберем вариант LoLo. Эта тревога у нас уже должна была быть настроена ранее. Мы делали это в предыдущем разделе.
- 7. Повторите аналогичные процедуры для следующих тегов:
  - 1. Vector1Dimension[1]
  - 2. Vector1Dimension[2]
  - 3. Vector1Dimension[3]
  - 4. Vector1Dimension[4]
- 8. Сохраним наш документ под именем Alarm\_History1.



A 🛃 Tags	Document Settings Save Type: Proprietary * C	onnection Item:			
Tagsi Data Types	History Items				
Graphics     Graphics     Templates     Services     Fingers     Alarm History     Alarm_History1     Tag History	Tag: Vector1Dimensia	Numeric    Hi    HiHi	🗌 lo 🗹 lolo	Deviation     Freeze     Watch Dog	Boolean True False
Recipes     Reports     Tunneling     Drivers     Images     User Functions Library     System Functions Library	Tag: Vector1Dimensi  Vector1Dimensi Vector1Dimensi Vector1Dimensi Vector1Dimensi Never delete history Days to keep: 0	Numeric Hi HiHi	iii Lo V LoLo	Deviation Freeze Watch Dog	Boolean True False
	Tag: Vector1Dimensis  Never delete history Days to keep: 0	Numeric Hi HiHi	Lo ✓ LoLo	Deviation Freeze Watch Dog	Boolean True False
	Tag: Vector1Dimensi Vector1Dimensi Vector1Dimensi Value Never delete history Days to keep:	Numeric Hi HiHi	lo √ lolo	Deviation Freeze Watch Dog	Boolean True False

#### 14.3 Создание и настройка экранных форм для отображения тревог

Используем специальный объект Alarm Object для отображения тревог на нашем экране. Он позволяет отображать как текущие, так и архивные тревоги.

• Режим отображения текущих тревог (Online):

Alarm Configuration
 Online
 Type
 Online ~

В этом режиме будут отображаться только активные в данный момент тревоги. Тревога перестанет отображать только после того, как она была подтверждена оператором, и причина ее возникновения устранена. Тревоги нужно активировать в свойствах тега (в нашем случае это тревога типа LoLo тегов Vector1Dimension и соответствующим образом настроить наш Alarm Object.

• Режим отображения истории тревог (History):

Alarm Configuration
Type
History

В этом режиме отображаются только тревоги, произошедшие в определенный (с помощью настраиваемого фильтра) период времени, а также квитирование тревог

History



оператором и устранение причин тревог. Для вывода истории тревог необходимо не только настроить сами тревоги, но и создать для них документ типа Журнал Тревог (Alarm History) с записью во встроенную или внешнюю базу данных.

Теперь время создать экранные формы для отображения обоих типов тревог:

- 1. Создадим новую папку с именем Alarms в разделе Graphics.
- 2. Создадим новый документ Graphics1.
- 3. Открыв документ, установим его размеры таким образом:
  - Ширина (Width): 900
  - Высота (Height): 674
- 4. Добавим на экран метку (Label) History: со следующими параметрами:

Туре	Text	Тор	Left	Width	Height
Label	History:	19	172	110	19

5. В ленте раздела Graphics, панели Advanced Objects найдем и выберем объект Alarm/Event.

License	Graphics				
Groupb	ox 🛄 Tab	NET .NET control	Trend	💰 Bar	Grouping
Screen		Alarm/Event	💜 Matrix Grid	🖄 Line	12 Ungrouping
Menu		🗮 MultiTagViewer	Template	Pie	
Interface	Objects	Advanced	Objects	Charts	Grouping

- 6. Кликнем левой клавишей мыши в районе точки (0;0), в верхнем левом углу экрана. Это создаст новый объект типа Alarm/Event стандартного (200 x 200) размера.
- 7. Скорректируем его размеры с помощью окна свойств (Property Window):
  - Ширина (Width): 730; Высота (Height): 275
  - Слева (Left): 169; Сверху (Top): 45
- 8. Выберем тип этого объекта History в поле Туре раздела Alarm Configuration в окне свойств.

<ul> <li>Configuration</li> </ul>			
Alarm Configuration		History	
Туре	History ~		

Note: Alarm history will display all the alarm status changes.



- 9. В зоне TagConfig установим ссылки на теги, для которых мы установили аварийные пороги. В поле New Tag введем значение Vector1Dimension[0] и нажмем кнопку +.
- 10. Повторим ту же процедуру для остальных тегов: Vector1Dimension[1], Vector1Dimension[2], Vector1Dimension[3], Vector1Dimension[4] (как на картинке ниже).

New tag:			+	
Vector1Dimension[0]	Edit	x		
Vector1Dimension[1]	Edit	x		
Vector1Dimension[2]	Edit	X		
Vector1Dimension[3]	Edit	X		
Vector1Dimension[4]	Edit	X		

- 11. Дважды кликнем на теге Vector1Dimension[0] для изменения настроек, или просто перейдем в закладку Alarm tag Details.
- 12. На этом экране выделим тревогу LoLo для всех тегов, поскольку этот тип тревоги был нами сконфигурирован ранее для всех перечисленных тегов.

Tag	Туре	🗌 Hi	🗌 ніні	Lo	✓ LoLo	Deviation	Freeze	U Watch Dog	✓ True	✓ Fals
Vector1Dimension[0]	Float				•					<u> </u>
Vector1Dimension[1]	Float				$\checkmark$					[2]
Vector1Dimension[2]	Float				~				10	
Vector1Dimension[3]	Float				1				10	-
Vector1Dimension[4]	Float				1					10
Vector1Dimension[4]	Float				V					

- 13. Выберем столбцы для отображения в нашем окне тревог. В зоне GridConfig в закладке Grid выберем поле Ack Time.
- 14. Нажмем стрелочку для перемещения свойства тревог Ack Time в правую часть окна.
- 15. Затем переместим из правой части влево свойства Tag Description и Group.
- 16. Таким способом мы можем настроить нужный нам вид будущего окна тревог. Давайте добавим вправо также и поле Туре.



- 17. С помощью курсора (вверх) переместим поле Туре по вертикали между полями Priority и Message.
- 18. Продолжим настройку столбцов пока не получим примерно следующее:



- 19. По умолчанию все столбца имеют ширину в 40 точек. Если нам нужно изменить это, дважды кликнем, например, по полю Tag Name. Это откроет нам доступ к свойствам столбца.
- 20. В поле Width (Ширина), введем значение 120.
- 21. Повторим эту процедуру для остальных столбцов и установим:
  - Столбец: Priority, Width 50
  - Столбец: Type, Width 50
  - Столбец: Message, Width 140
  - Столбец: Start Time, Width: 120
  - Столбец: Return Time, Width: 120
  - Столбец: Ack Time, Width: 120
- 22. Оставим значение Alarm Date Range без изменения.

	O Absolute      Relative[d.f	n:mm:ss)
Alarm Date Panne	Now	~
ılarm Date Range	01:00:00	
	Backward Forward	

23. Теперь создадим второе окно тревог, типа Online (для отображения текущих тревог). Для ускорения просто скопируем созданный нами ранее объект.



Tap Name	Priority Type Message	Start Time	Return Time	Ack Time	
Tag Name	Priority Type Message 0 None	Start Time	Return Time	Ack Time	
0		0			

24. Изменим расположение второго окна на Тор: 330, Left: 170. И добавим новый заголовок (Online) также как мы это сделали для первого окна (History).

Online: Tag Name Group Type Priority Message Start Time Return Time Ack Time None	Tag Name	Priority 0	Type None	Message	Start Time	Return Time	Ack Time
None							
	Online: Tag Name	Group	Туре	Priority Message	Start Time	. Return Tin	me Ack Time

25. Установим тип объекта в Online, а не в History, как раньше

Configuration		
Alarm Configuration		Online
Туре	Online 🗡	



26. Добавим несколько полей Label и Textbox для того, чтобы вывести туда текущие значения тегов. В объектах Label введем названия тревог, а Text – значения тегов. В результате у нас должно получиться примерно следующее:

Alarm 1:	History	8111111	115.57	**********		1 = 1 =	
@Vector1Dimension	Tag Name	Priority	Type	Message	Start Time	Return Time	Ack Time
		0	None				
Alarm 2:							
@Vector1Dimension	:						
	:						
Alarm 3							
-Mianni S.							
@Vector1Dimension							
Alarm 4:							
@Vector1Dimension							
Alarm 5:							
Alarm 5: @Vector1Dimension	Online		:121:				
Alarm 5: @Vector1Dimension	Online						
Alarm 5: @Vector1Dimension	Online: Tag Name	Group Ty	ype f	Priority Message	Start Time	Return Time	Ack Time
Alarm 5:	Online: Tag Name	Group Ty N	ype i	Priority Message	Start Time	Return Time	Ack Time
Alarm 5:	Online: Tag Name	Group Ty N	ype i Ione	Priority Message	Start Time	Return Time	Ack Time
Alarm 5:	Online: Tag Name	Group Ty N	ype i Ione	Priority Message	Start Time	Return Time	Ack Time
Alarm Sc @Vector1Dimension	Online: Tag Name	Group Ty N	ype F Ione	Priority Message	Start Time	Return Time	Ack Time
Alarm S:	Online: Tag Name	Group Ty N	ype F lone	Priority Message	Start Time	Return Time	Ack Time
Alarm Sc @Vector1Dimension	Online: Tag Name	Group Ty N	ype i Ione	Priority Message	Start Time	Return Time	Ack Time
Alarm 5:	Online: Tag Name	Group Ty N	ype i Ione	Priority Message	Start Time	Return Time	Ack Time
Alarm S: @Vector1Dimension_}	Online: Tag Name	Group Ty N	ype i Ione	Priority Message	Start Time	Return Time	Ack Time
Alarm Sc @Vector1Dimension	Online: Tag Name	Group Ty N	ype I Jone	Priority Message	Start Time	Return Time	Ack Time
Alarm S:	Online: Tag Name	Group Ty N	ype i ione	Priority Message	Start Time	Return Time	Ack Time
Alarm Sc @Vector1Dimension	Online: Tag Name	Group Ty N	ype i ione	Priority Message	Start Time	Return Time	Ack Time
Alarm S:	Online: Tag Name	Group Ty N	ype I Ione	Priority Message	Start Time	Return Time	Ack Time
Alarm Sc @Vector1Dimension	Online: Tag Name	Group Ty N	ype i ione	Priority Message	Start Time	Return Time	Ack Time
Alarm S:	Online: Tag Name	Group Ty N	ype i Ione	Priority Message	Start Time	Return Time	Ack Time
Alarm S:	Online: Tag Name	Group Ty N	ype f	Priority Message	Start Time	Return Time	Ack Time
Alarm S:	Online: Tag Name	Group Ty N	ype I lane	Priority Message	Start Time	Return Time	Ack Time
Alarm 5:	Online: Tag Name	Group Ty N	ype f	Priority Message	Start Time	Return Time	Ack Time
Alarm Sc @Vector1Dimension	Online: Tag Name	Group Ty N	ype I lone	Priority Message	Start Time	Return Time	Ack Time

Примечание: Значения тегов в текстовых полях - @Vector1Dimension[0]... [4]

27. Сохраним документ под именем Alarms.



# 15. История значений тегов (Tag History)

Документ Tag History (история значений тега, архив тренда) сохраняет историю значений тега. Значения в этот архив могут писаться по таймеру или по изменению значения тега. Архив может писаться во встроенную базу данных или в заранее сконфигурированную (Settings -> Database Connections) внешнюю реляционную базу данных.

Tag_Histo	ry1* X	
ocument Set	ings	
Enable Save:	TRUE v	
Save Modes		History Settings
🗹 On Tag	hange	Never Delete History
Frequen	ry (s)= 60	Days To Keep: 0
Trigger:		
History Type		
Type: Prop	ietary Y	
Proc		
Frop	setary	
	etany разе	а значений тегов.
History II Data	eшней базы данных для хранения ум	а значений тегов:
History I Deta	etary разе ешней базы данных для хранения у•• × пgs	а значений тегов:
GOD BH Tag_Histo	eшней базы данных для хранения yf* × ngs TRUE v	а значений тегов:
History II Data	eшней базы данных для хранения уз* × ngs пкие v	а значений тегов: History Settings
ISTOP II Data	eшней базы данных для хранения y1* × ngs TRUE v	а ЗНАЧЕНИЙ ТЕГОВ: History Settings ☑ Never Delete History
GOD BH Tag_Histo Document Sett Inable Save Save Modes ✓ On Tag C □ Frequenc	ietary разе ешней базы данных для хранения rgt* х ngs TRUE v hange y (s): 60	а ЗНачений тегов: History Settings ✓ Never Delete History Days To Keep: 0
istory I Data GOD BH Tag_Histo Tag_Histo Tag_Histo Comment Sett Save Modes ✓ On Tag C ☐ Frequence ☐ Trigger:	eшней базы данных для хранения y1* × ngs rRUE v hange y (s): 60	а ЗНАЧЕНИЙ ТЕГОВ: History Settings Ø Never Delete History Days To Keep: 0
istory I Data GOD BH Tag_Histo Document Sett inable Save: Save Modes ☑ On Tag O □ Frequenc □ Frequenc □ Trigger: History Type	eшней базы данных для хранения y <sup>1*</sup> × ngs TRUE v hange y (s): 60	а ЗНАЧЕНИЙ ТЕГОВ: History Settings ✓ Never Delete History Days To Keep: 0
History I Data COD BH Tag_Histo Document Sett Inable Save: Save Modes ♥ On Tag O □ Frequenc □ Trigger: History Type Type Data	eшней базы данных для хранения syl* × ngs TRUE v hange y (s): 60 ase v Connection Item: v	а значений тегов: History Settings ✓ Never Delete History Days To Keep: 0

# 15.1 **Создание и настройка архива с записью по изменению тега** (On Tag Change)

Для создания архива тега с записью по изменению его величины сделаем следующее:

- 1. Создадим новый документ в разделе Tag History в дереве навигации.
- 2. Настройки этого документа будет применены ко всем включенным в документ тегам

Примечание: Если нам потребуются разные интервалы сохранения для разных тегов, нам придется создать новый документ Tag History.

3. В секции Save Modes нашего документа установим галочку On Tag Change.



4. А в секции History Settings установим галочку Never Delete History, чтобы не ограничивать глубину архива по времени.

Document Se	ttings		
Enable Save:	TRUE	*	
Save Mode:		History Settings	
🗹 On Tag	Change	Never Delete History	
E Frequer	ncy (s): 60	Days To Keep: 0	
Trigger			
History Type	e		
Type: Prop	prietary 👻		

- 5. После этого нажмем кнопку Insert History в верхней ленте.
- 6. Для нового элемента установим значение Tag в @Vector2Dimensions[0,0].
- 7. Сохраним документ как Tag\_History1.

History Items			
Tag: Vector2Dimensions[0,0]	 Dead Band:	0	

#### 15.2 Создание и настройка архива с записью по таймеру (Time Frequency)

Для создания архива тегов с записью по таймеру проделаем следующие шаги:

- 1. Создадим новый документ в разделе Tag History в дереве навигации. Ранее созданный документ нам не подойдет, так как он имеет тип записи по изменению значения (On Tag Change).
- 2. В секции Save Modes нашего нового документа установим галочку в поле Frequency и установим период записи в 30 секунд.



Document Se	ttings		
Enable Save:	TRUE	*	
Save Mode		History Settings	
🗌 On Tag	Change	Vever Delete History	
✓ Frequer	ncy (s): 30	Days To Keep: 0	
🗌 Trigger			
History Typ	e		

- 3. Добавим в документ новый элемент с именем тега @Vector2Dimensions[1,1].
- 4. Сохраним документ как Tag\_History2.

listory	Items			
Tag:	Vector2Dimensions[1,1]	 Dead Band:	0	
_				

## 15.3 Создание и настройка экрана для вывода исторических трендов

Это будет очень простой экран. Всего два объекта Label и два объекта Textbox. Сделаем следующие шаги:

- 1. Создадим папку History в разделе Graphics.
- 2. Создадим в ней новый документ с такими параметрами:
  - Width: 1014
  - Height: 674
- 3. Добавим два объекта типа Labelc параметрами:
  - Label1: Width: 128, Height: 18, Top: 20, Left: 20, Text: Vector2Dimensions[0,0]
  - Label2: Width: 128, Height: 18, Top: 50, Left: 20, Text: Vector2Dimensions[1,1]

	2	GI	a	P	h	ic	\$1	2		×		1	_						_	_	_				_	_	_	_	_		_		_	
-	.,	,						.,				.,		÷	.,			÷				-								.,		.,	-	
1.14		٠.,									•			÷				÷			÷	+		•		4		÷						
	1			-							١,	-		۰.				-				-							-					
1.1		V	ec	t	or	2	Ч	m	e	n	51	0	n	s	ίQ	ļ,(	Ŋ			٠									-					
1.1				-			-					-						-											-					
	1					÷.					۰.	-		۰.				-							-			-	-				-	
		V	50	t	or	2	Э	na	e	n	51	0	n	s	[]	, i	IJ	-				-							-					
							-					-																						

- 4. Добавим два объекта типа TextBox с параметрами:
  - TextBox1: Width: 148, Height: 25, Top: 16, Left: 164, Text: @Vector2Dimensions[0,0]
  - TextBox2: Width: 148, Height: 25, Top: 46, Left: 164, Text: @Vector2Dimensions[1,1]
- 5. Сохраним документ как History.



History ×																				
																				1
																				1
																				1
<ul> <li>Vector2Dimensions[0,0]</li> <li>@Vector2Dimensions[0,0]</li> </ul>																				1
	18	• •	. *	1	• •	. *	+ +	1.1	*,-	1.1	1.7						+			
	*		1.	*1	• : =		+ +		+1	- 14	+				+ +		+			-88
<ul> <li>Vector2Dimensions[1,1]</li> <li>@Vector2Dimensions[1,1]</li> </ul>	$(\mathbf{k})$	10.1		4.5	• •	.+.	+ -		+: -:	+	14	+ +	1.0		• •		+1	÷. (4)	+ +	-8
	1.1		1.4	1.1		1.1	64 G		- 20		120		1.1	1.1		1.0	1.0	1.1.1		87

6. Найдем объект типа Trend в секции Advanced Objects раздела Graphics в верхней ленте.



- 7. Добавим его в произвольное место на экран, кликнув левой клавишей мыши и нарисовав прямоугольник нужного размера. После этого установим размеры этого объекта и его положение (в разделе Properties) следующим образом:
  - Top: 100, Left: 20
  - Width: 978, Height: 586
- 8. Добавим новый исторический тег (Перо на графике), вводя имя тега в поле New Pen и нажимая кнопку + или Enter на клавиатуре.

Pens		
New pen:	+	
Vector2Dimensions	[0,0] () Edit X	
Vector2Dimensions	[1,1] () Edit X	



9. Теперь настроим вид пера для @Vector2Dimentions[1,1]

elected Item:	Vector2	Dimensions[1,1] () v		
onfiguration	_			
Description:				
Pen Color:				
tarker:	Triang	ple	~	
hickness:	1	Interpolated: 🗸		
show Legend:	1			
ine Style:			~	
Statistical Pro	cess Con	itrol (SPC)		
ne Style: itatistical Pro	···	trol (SPC)	~	

10. В результате наш экран должен выглядеть примерно так:



Примечание: После запуска нашего проекта, в папке (C:\GettingStarted\History) будет создан файл с именем: Vector2Dimensions [0,0] \_20130220.HF. Его имя будет указывать на тег, значения которого в нем записаны, а его содержание будет зашифровано.



# 16. Рецепты (Recipes)

Документ типа Recipe представляет собой запись значений некоторого набора тегов. Эти значения могут быть считаны в исполняемый проект в определенные моменты времени. Научимся пользоваться этим механизмом.

# 16.1 Создание и настройка рецепта (Recipe Document)

Для понимания принципов работы данного механизма создадим и настроим простой рецепт следующим образом:

- 1. Создадим новый документ в разделе Recipes в дереве навигации.
- 2. Нажмет кнопку «...» у поля Recipe Path для задания имени и размещения файла рецепта.
- 3. В отрывшемся окне согласимся с предложенным расположением файла, и назовем ero TestDocumentRecipe.
- 4. Нажмем Save. После этого окно закроется, а имя нового файла рецепта появится в поле Recipe Path.
- 5. Оставим галочку в опции Relative Path.
- 6. В поле Recipe Save Mode выберем опцию File.

Примечание: В случае, если мы выберем опцию XML mode, расширение файла рецепта по-прежнему останется .RPF (не .XML), изменится лишь его внутренняя структура.

- 7. В поле Recipe Items, двойным кликом на первой свободной строке, добавим первый тег Vector3Dimensions.
- 8. Кликнем на «...», откроется окно TagBrowser.
- 9. Выберем в нем тег Vector3Dimensions и нажмем OK. Выбранный тег появится в первой линии документа. Можно также ввести название тега напрямую. Необходимо помнить, что если мы укажем имя многомерного тега (массива размерностью 1,2 или 3) без указания индексов, то в рецепт попадут BCE значения массива. Для указания конкретного элемента массива синтаксис должен быть примерно таким: (Vector3Dimensions[0,0,0]).
- 10. Сохраним наш рецепт под именем Recipes1.

После этого настройку рецепта можно считать законченной.



Recipe Path:	TestDocumentRecipe.rpf		🔽 Relative Path
Туре:	Default ~	Recipe Save Mode:	File ~
Trigger Save:		Trigger Load:	
Recipe Items			
Tags			
Vector3Dime	ensions		

### 16.2 Создание и настройка экранной формы для работы с рецептами

Теперь мы создадим специальный экран для иллюстрации различных способов использования рецептов. Существует несколько способов записи данных в рецепт и извлечения этих данных в проект. Приступим к их изучению.

В новой экранной форме создадим несколько кнопок и привяжем к ним скрипты для сохранения и чтения рецептов. Проделаем следующие операции:

- 1. Создадим новую папку Recipes в разделе Graphics.
- 2. Создадим новый документ Graphics.
- 3. Добавим в него объект типа MultiTagViewer из группы Advanced Objects в разделе Graphics верхней ленты.

License	Graphics					
Groupbo	x 🛄 Tab	NET control	Trend 🚱 Matrix Grid	d Bar ₩ Line	Grouping	<ul> <li>Resize Height</li> <li>Resize Width</li> </ul>
Menu		📇 MultiTagViewer	Template	😍 Pie		🗘 Resize Width and Height
Interface	Objects	Advanced (	Objects	Charts	Grouping	Resize

- 4. Кликнем в районе точки (0;0), в левом верхнем углу документа и создадим элемент стандартного размера.
- 5. После этого установим размеры этого объекта и его положение (в разделе Properties) следующим образом:
  - Width: 300, Height: 650
  - Top: 20, Left: 20



- 6. Введем тег Vector3Dimensions в поле Tag в разделе Configuration -> Grid в окне свойств и нажмем Enter. Некоторые из его свойств отобразятся в левой колонке.
- 7. Выберем свойство Name (имя) и поместим его стрелочкой в правую колонку.
- 8. Повторим эту операцию со свойством Value.

t Index: 0 Use Count	
t Index: 0	
nt: 0 🗌 Use Count	
MinValue Name	
MaxValue Value	
EngUnits	
lame	
alue	
imestamp	
uality	
etentiveValue	
omm	
hitialValue	

- 9. Дважды кликнем на свойстве Name для настройки.
- 10. В поле Width (ширина) введем значение 170.
- 11. В этой же вкладке Columns выберем для настройки столбец Value.
- 12. Установим его ширину (Width) в 120 и установим галочку Input Enable для того, чтобы мы могли изменять значение тега непосредственно на экране.
- После настройки MultiTagViewer создадим объект типа GroupBox и две кнопки (Buttons): одну назовем Load, а вторую Save. Эти кнопки будут выполнять операции извлечения и сохранения данных нашего рецепта.
- 14. В итоге наше экранная форма будет выглядеть примерно так:



					and the Print	
Marrie	Walve	1,000 /	2008	Love /	Save with Fath	10000000
- val	. yad	188688		12221		
and a set	ual					
			Load		Load	
1.1	val					
lan lan	val			• [CCC]		
- usi	val		Save		Save	
the set	and a second s			· [::::]		-
-	58					
		Contract Contraction (Contraction)	Party and Brids 1.		Salar and Whit with Eath 1	
		1	Save and stars			
				- 10000		
			1		1	
			Load		Load	
				- 10000 L		
		11111		<ol> <li>1000 (1)</li> </ol>	1	10.000 (10.000)
			Xeen		Xee	a second second
			parts.		DALT	
		A 4 4 4 4				and the second second

- 15. Для того, чтобы сделать это, начнем с создания нового объекта типа GroupBox. Установим свойство Text этого объекта в Load / Save.
- 16. Добавим на экран новую кнопку (Button).
- 17. Выделим эту кнопку и нажмем иконку Split внизу секции Mode View. Это позволит нам одновременно видеть скрипт и дизайн.

	cload / Save	Load /	Save with Path	5
11010		11111		
		1.1.1.1		
11022	0 0 0			
		1.1.1.1		
	C Load P		Load	
	0 0 0			
	Save		Save	
	L			
	-Load / Save and Wait	Load /	Saive and Wait with Path	
	Load / Save and Wait	Load /	Saue and Wait with Path	
	Load / Save and Wait	Load /	Save and Walt with Path	
	-Load / Save and Wait	Load /	Save and Walt with Path	
	-Load / Save and Wait	Load /	Save and Wait with Path	
	Load / Save and Wait	Load /	Save and Walt with Path	
	Load / Save and Wait	Load /	Save and Wait with Path	
	Load / Save and Wait	Load /	Save and Walt with Path	
	-Load / Save and Wat	Load /	Save and Wait with Path	
	-Load / Save and Walt	-Load /	Load	~
¢	Load / Save and Wait Load	-Load /	Save and Walt with Path	~
4	Load / Save and Wat	-Losd /	Save	-
<	Load / Save and Wait Load	-Load /	Save and Walt with Path	-
< 	Load / Save and Wat	-Losd /	Save And Walt with Path	~
<	-Load / Save and Walt	-Load /	Save and Wait with Path	
<	Load / Save and Wat		Save And Walt with Path	
<	-Load / Save and Walt	-Losd /	Load Save	
<	Load / Save and Wat		Save	
<	-Load / Save and Walt		Save	
<	-Load / Save and Wat		Save	
<	Load / Save and Wait		Losd Save	
<	Load / Save and Walt		Save	
C 1 Line: 1	Load / Save and Wat		Save And Walt with Path	

Вид среды разработки (Mode View) может быть одним из трех:



- Design (): Мы видим только окно для редактирования графики
- Script (<sup>23</sup>): Мы видим только окно ScriptDock для написания скриптов
- Split (曰): Открыты оба окна: и редактор графики и ScriptDock
- 18. Для вкладки Mouse Up пишем такой код:

#### SVRecipe.Load(Recipes1);

Этот скрипт восстанавливает значения тегов в рецепте из файла, когда отпускается левая клавиша мыши.

- 19. Введем слово Load в поле Text в окне свойств кнопки 1 (button1).
- 20. Создадим еще одну кнопку (объект типа Button).
- 21. В разделе Mouse Up в окне Script Dock напишем следующий код:

#### SVRecipe.Save(Recipes1);

Этот скрипт сохраняет значения тегов, указанных в нашем рецепте, в соответствующий .RPF файл.

- 22. Введем слово Save в поле Text в окне свойств кнопки 2 (button2).
- 23. Повторим аналогичные процедуры для следующих объектов:
  - GroupBox: Text: Load / Save with Path
  - Button Load: Text: Load Mouse Up:

SVRecipe.Load(Recipes1,C:\\GettingStarted\\Recipe\\TestRecipePath.rpf);

• Button Save: Text: Save, Mouse Up:

SVRecipe.Save(Recipes1,C:\\GettingStarted\\Recipe\\TestRecipePath.rpf);

- GroupBox: Text: Load / Save And Wait
- Button Load: Text: Load Mouse Up: SVRecipe.LoadAndWait(Recipes1);
- Button Save: Text: Save, Mouse Up:

SVRecipe.SaveAndWait(Recipes1);

- GroupBox: Text: Load / Save And Wait with Path
- Button Load: Text: Load Mouse Up:

SVRecipe.LoadAndWait(Recipes1,C:\\GettingStarted\\Recipe\\TestRecipePath.rpf);

• Button Save: Text: Save, Mouse Up:



#### SVRecipe.SaveAndWait(Recipes1,C:\\GettingStarted\\Recipe\\TestRecipePath.rpf);.

24. Сохраним документ под именем Recipe.

# 16.3 Различия между разными блоками (GroupBoxes) Load / Save в нашем экране Recipe

Кнопки, расположенные в блоке Load / Save загружают и сохраняют значения тегов в соответствие с конфигурацией в рецепте (Recipe Document).

Кнопки, расположенные в блоке Load / Save with Path загружают и сохраняют значения тегов в соответствие с конфигурацией в рецепте (Recipe Document), однако, БЕЗ учета пути, установленного в нем. Вместо этого расположение файла .RPF, в котором сохраняются значения, определяется явно. D нашем случае это: C:\GettingStarted\Recipe\TestRecipePath.rpf.

Кнопки, расположенные в блоке Load / Save and Wait загружают и сохраняют значения тегов в соответствие с конфигурацией в рецепте (Recipe Document). Отличие от обычных Load и Save заключается в том, что LoadAndWait и SaveAndWait во время выполнения своей функции блокируют выполнение других параллельных действий, чтобы предотвратить возможную рассинхронизацию данных.

Кнопки, расположенные в блоке Load / Save And Wait with Path загружают и сохраняют значения тегов в соответствие с конфигурацией в рецепте (Recipe Document), однако, БЕЗ учета пути, установленного в нем. Вместо этого расположение файла .RPF, в котором сохраняются значения, определяется явно. В нашем случае это:

C:\GettingStarted\Recipe\TestRecipePathComWait.rpf . В тоже время, во время исполнения блокируются другие параллельные действия, чтобы предотвратить возможную рассинхронизацию данных.

# 16.4 Ожидаемый результат выполнение экрана Recipes

Чего же нам ожидать от нажатия на кнопки экранной формы Recipes после запуска его на исполнение? При нажатии кнопки Save значения тегов, отображаемые в MultiTagViewer, сохранятся в файл .RPF. Несмотря на расширение файла, это будет текстовый файл, и его можно легко открыть в любом текстовом редакторе.

После того как кнопка Save будет нажата, и файл будет создан, мы можем убедиться в том, что сохранились именно нужные данные, перейдя в папку, где располагается сохраненный файл (C:\GettingStarted\Recipe) и открыв файл с расширением .RPF с помощью любого текстового редактора.

И наоборот, кнопка Load вызовет загрузку значений, содержащихся в файле .RPF, в теги, которые отображаются в MultiTagViewer.



# 17. Отчеты (Reports)

Документы типа Report позволяют нам генерировать отчеты. Отчеты могут быть сохранены в файл или направлены непосредственно на печать.

## 17.1 Создание и конфигурирования документа типа Report

В этом разделе объясним, как настроить простой отчет, отображающий содержимое некоторых тегов проекта. Для этого в документе Tags1 добавим новый тег с именем TriggerReport, целочисленного типа (Integer) и с размерностью 0 – No array.

TagBoolean	Boolean	Array: 0 - No array ~
TagInteger	Integer	<ul> <li>Array: 0 - No array</li> </ul>
TagFloat	Float	✓ Array: 0 - No array ✓
TagString	String	<ul> <li>Array: 0 - No array</li> </ul>
TagDateTime	DateTime	<ul> <li>Array: 0 - No array</li> </ul>
Vector1Dimension	Float	
Vector2Dimensions	Float	Array: 2 Dimensions Y Column(X): 2
Vector3Dimensions	Float	<ul> <li>Array: 3 Dimensions Column(X): 2</li> </ul>
VectorDynamic	Float	✓ Array: 1 Dimension ✓ ✓ Dynamic
Structure	StructureExample	<ul> <li>Array: 0 - No array</li> </ul>
DateHourCurrent	DateTime	✓ Array: 0 - No array ✓
Triggers	Float	
TagTrigger	Integer	<ul> <li>Array: 0 - No array</li> </ul>
TagCondition	Integer	Array: 0 - No array ×



После создания нового тега сделаем следующее:

- 1. Создадим новый документ Report в дереве навигации.
- 2. В поле Trigger в секции Report Settings введем @TriggerReport. Изменение значение этого тега приводит к запуску на исполнение нашего документа, создавая, таким образом, настроенный в нем отчет.
- 3. В поле File path кликнем кнопку «...» для выбора расположения и имени файла нашего будущего отчета.
- 4. В открывшемся диалоге выберем нужное нам расположение и имя. Например, оставим путь по умолчанию: (C:\GettingStarted\Reports) и введем имя TestReport.rtf. Оставим галочку в **поле** Relative Path.
- 5. В параметре Page size (размер страницы), выберем А4 или Letter.
- 6. В параметре Orientation выберем портретную (вертикальную) ориентацию.
- 7. В поле редактора (Editor) введем следующий текст:

Values of tags

- TagBoolean: @TagBoolean
- TagInteger: @TagInteger
- TagFloat: @TagFloat
- TagString: @TagString
- TagDateTime: @TagDateTime
- Structure.Member1[0]: @Structure.Member1[0]
- Vector3Dimensions[0,0,0]: @Vector3Dimensions[0,0,0]
- 8. Сохраним документ под именем Reports1.


S	ADISRA SmartView 4.0.1.0
Runtime Build View	w Settings Search License Reports
Insert Remove Chart Chart Charts Table	Insert     Remove     Apply     Apply     Apply     Apply     Align     Align     Align     Align     Apply     builtets     Segoe UI       Insert     Remove     Insert     Insert     Insert     Remove     Insert     Insert     Remove     Insert     Insert <td< th=""></td<>
Navigation Tree 🔹 🛡 🗙	Reports1* ×
GettingStarted     GettingStarted     SystemTags     Tags     Tags     Tags	Report Settings Trigger: @TriggerReport Output: File File path: TestReport.rtf Relative Path
Graphics	Page size: A4  Orientation: Portrait
Alarms     Alarms     History     History     Recipes     Recipe     Services	Document Settings Enable: TRUE ~
	Editor
Service Triggers Tinggers Services Services Services Triggers Alarm History Services Alarm History Recipes Recipes Recipes1 Reports	Values of tags • TagBoolean: @TagBoolean • TagInteger: @TagInteger • TagFloat: @TagFloat • TagString: @TagString • TagDateTime: @TagDateTime • Structure.Member1[0]: @Structure.Member1[0] • Vector3Dimensions[0,0,0]: @Vector3Dimensions[0,0,0]

## 17.2 Создание и настройка экранной формы для генерации Отчетов

Эта экранная форма будет простой. Всего лишь несколько объектов типа Label и TextBox, которые буду отображать имена и значения тегов, присутствующих в нашем рецепте. Для их настройки выполним следующие шаги:

- 1. Создадим новую папку с именем Report в разделе Graphics.
- 2. Создадим в ней новый экран.
- 3. Добавим на экран восемь меток (Label) и восемь текстовых полей TextBox и настроим их примерно следующим образом:





- 4. Для этого меткам (Label) назначим следующие параметры:
  - Label1: Text: TriggerReport Width: 140 Height: 16 Top: 60 Left: 65
  - Label2: Text: TagBoolean Width: 140 Height: 16 Top: 90 Left: 65
  - Label3: Text: TagInteger Width: 140 Height: 16 Top: 120 Left: 65
  - Label4: Text: TagFloat Width: 140 Height: 16 Top: 150 Left: 65
  - Label5: Text: TagString Width: 140 Height: 16 Top: 180 Left: 65
  - Label6: Text: TagDateTime Width: 140 Height: 16 Top: 210 Left: 65
  - Label7: Text Structure.Member1[0] Width: 140 Height 16 Top: 240 Left: 65
  - Label8: Text Vector3Dimensions[0,0,0] Width: 140 Height 16 Top: 270 Left: 65
- 5. А текстовым полям (TextBox) следующие:
  - Text-Box1: Text @TriggerReport Width: 150 Height 25 Top: 56 Left: 219
  - Text-Box2: Text @TagBoolean Width: 150 Height 25 Top: 86 Left: 219
  - Text-Box 3: Text: @TagInteger Width: 150 Height: 25 Top: 116 Left: 219
  - Text-Box 4: Text: @TagFloat Width: 150 Height: 25 Top: 146 Left: 219



- Text-Box 5: Text: @TagString Width: 150 Height: 25 Top: 176 Left: 219
- Text-Box 6: Text: @TagDateTime Width: 150 Height: 25 Top: 206 Left: 219
- Text-box 7: Text @Structure.Member1[0] Width: 150, Height: 25 Top: 236 Left: 219
- Text-Box 8: Text @Vector3Dimensions[0,0,0] Width: 150, Height: 25 Top: 266 Left: 219
- 6. Создадим кнопку (Button) для запуска генератора отчетов и добавим в нее следующий код:

```
if (@TriggerReport == 0)
@TriggerReport = 1;
```

else

```
@TriggerReport = 0;
```

System.Windows.Forms.MessageBox.Show(Report generated successfully.,

Information,

MessageBoxButtons.OK,

System.Windows.Forms.MessageBoxIcon.Information);



7. Сохраним нашу экранную форму под именем Report.



8. Существует еще один способ генерации отчетов - использование специальной системной библиотеки под названием SVReport:



9. Например, мы можем использовать вызов функции SaveFile из кнопки в любом месте экрана.

```
void SaveFile(string reportFile, string path)
= #region
  /// Summary:
  ///
          Function to save a report document in a specific path
  111
  /// Parameters:
  /// string reportFile
  ///
         A String parameter containing the name of the report to be saved.
  ///
  /// string path
         A string parameter with the path, including a valid extension
  ///
  111
  #endregion
```

10. Давайте попробуем использовать эту функцию, чтобы сгенерировать отчет не в RTF формате, а в формате PDF.



Report ×	
- + + + + + + + + + + + + + + + + + + +	c + c + c + c + c + c + c + c + c + c +
Trices	@TringerDenest
Inggerkeport	@Inggerkeport
Technology	
lagboolean	@TagBoolean
Techterer	
Faginteger	@TagInteger
lagrical	@TagFloat
a string	@TagString
To-Data Tara	e <del>presentation and a second second</del>
taguaterime	@TagDateTime
for the second second second second	·
Structure.Member1[0]	@Structure.Member1[0]
Marta 20 marta 10.0.01	
vectorsbimensions[0,0,0]	@vector3Dimensions
	<u>]</u>
Generate Report (rtf)	Generate Report (pdf)
	· · · · · · · · · · · · · · · · · · ·

- 12. Назовем ее Generate Report (pdf) и перейдем к ее скрипту.
- 13. Добавим в него следующий код:

string pathReport = ; pathReport = SVApplications.ProjectPath()+Reporte.PDF; SVApplications.Output(pathReport); SVReport.SaveFile(Reports1, pathReport);

System.Windows.Forms.MessageBox.Show(Report generated successfully in the following path: + pathReport,

> Information, MessageBoxButtons.OK, System.Windows.Forms.MessageBoxIcon.Information);

1 string pathReport = " 2 pathReport = SVAppl 3 SVApplications.Outp 4 SVReport.SaveFile(" 5	". ications.ProjectPath() = "Reporte.PDF"; ut(pathReport); Reports 1°, pathReport;	
6 System Windows Form	as MessageBox.Show("Report generated successfully in the following path: " + pathReport, "	
7	"Information",	
8	MessageBoxButtons.OK,	
9	System.Windows.Forms.MessageBoxlcon.information);	
ine 1		2
	Mourse Bits F Mourse Down Mourse White Mourse Right Lin Mourse Right Down Mourse Double Click	



14. Эта кнопка будет генерировать нам отчет в PDF формате в папке нашего проекта.

## 17.3 Ожидаемый результат выполнение экрана Reports

При запуске приложения мы можем создать отчет, используя одну из кнопок, или просто изменив значение первого текстового поля, в котором выведен тег @TriggerReport. Этот тег настроен как Триггер отчета, поэтому любое его изменение запустит создание отчета с именем C:\GettingStarted\Reports\TestReport.rtf. Вторая добавленная нами кнопка сгенерирует файл отчета в формате PDF с именем C:\GettingStarted\TestReport.pdf.





## 18. Драйверы

Раздел Drivers содержит особые документы, описывающие настройки связи среды исполнения ADISRA с внешними устройствами по разным стандартным протоколам, связи с серверами и клиентами OPC, а также связи с различными базами данных. В нашем примере мы будем использовать драйвер связи по протоколу OPC UA, а также стандартный драйвер одного из самых распространенных промышленных протоколов. Важно отметить, что работа с базами данных не сильно отличается по настройке от работы с другими драйверами, и позволяет нам связать любой тег с любым значением из таблиц базы данных.

## 18.1 Документ типа ОРС UA клиент

Среда исполнения ADISRA может быть как клиентом, так и сервером OPC. В этом разделе мы попробуем настроить работу нашего проекта с одним из известных нам из доступных для свободной загрузки (на настоящий момент времени) OPC UA серверов, но Вы можете также использовать в этой части обучения любой доступный Вам OPC UA сервер.

Сейчас мы загрузим и установим симулятор ОРС UA сервера, а затем настроим связь с ним в ADISRA SmartView. Конечным результатом должна быть экранная форма для мониторинга тегов, получаемых от внешнего сервера ОРС UA.

#### 18.1.1 Загрузка и установка внешнего ОРС UA сервера

- Скачаем на свой диск следующий архив: <u>https://adisra.com/wp-content/uploads/2020/03/ADISRA-SmartView-GettingStarted-IntegrationObjects.zip</u>
- 2. Распакуем и установим этот ОРС UA сервер, запустив указанный ниже файл:

Nome	Data de modificaç	Tipo	Tamanho
install.txt	10/11/2020 16:32	Documento de Te	1 KB
integrationObjects'OPCUAServerSimulator_1.2.0.exe	17/08/2020 12:07	Aplicativo	20,018 KB

3. После установки запустим OPC симулятор. Это можно сделать с помощью ярлыка или непосредственно запустив файл OPCUASimulator.exe из папки, куда он был установлен.



	≪ Disco Local (C:)      → Program Files      → Integr	ration Objects 🔸 Integratio	on Objects' OPC UA Se	rver Simulator
^	Nome	Data de modificaç	Tipo	Tamanho
	OPCNetServerSDK.dll	30/01/2020 15:29	Extensão de aplica	50 KB
	OPCUAServerConfig.ini	10/11/2020 17:01	Parâmetros de co	1 KB
	OPCUAServerSimulator.Config.xml	10/11/2020 17:02	Documento XML	7 KB
	SOPCUAServerSimulator.exe	03/08/2020 13:54	Aplicativo	354 KB
	OPCUAServerSimulator.exe.config	13/02/2020 14:58	Arquivo CONFIG	1 KB
	🐻 Setup.ico	18/10/2019 15:01	Arquivo ICO	10 KB
	System.AppContext.dll	18/10/2019 15:01	Extensão de aplica	20 KB
	System.Buffers.dll	13/09/2019 13:01	Extensão de aplica	27 KB

4. Симулятор позволит нам создавать новые ОРС теги, которые могут иметь либо фиксированные, либо смоделированные значения. В этом примере мы будем использовать уже созданные по умолчанию адреса, но при желании Вы самостоятельно сможете настроить новые ОРС теги, перейдя в папку DATA в папке установки Integration Objects OPC UA Server Simulator и отредактировав файлы csv.

C Disco Local (C:) > Program Files > In	tegration Objects 🔸 Integratio	on Objects' OPC UA	Server Simulator + DA	TA Y	v C
Nome	Data de modificaç	Tipo	Tamanho		
AddressSpace.csv	03/02/2020 14:39	Arquivo CSV	1 KB		
ValueSpace.csv	03/02/2020 14:39	Arquivo CSV	1 KB		

- AddressSpace.csv позволит нам создавать новые OPC теги. В последнем столбце указано, будет ли OPC тег симулироваться или нет.
- ValueSpace.csv позволит нам создавать ОРС теги для моделирования. В первой строке указан адрес, а в строках ниже значения для симуляции.



erver Endpoints U Sessions	RLs opc.tcp://pc:6264U/I	itegration/Objects/Se	rversimulator		
SessionId	Name Us	er	Last Contact		
Subscriptions					
Subscriptions	Publishing Interval	Item Count	Seq No		
Subscriptions	Publishing Interval	Item Count 2	Seq No 2		
Subscriptions	Publishing Interval	Item Count 2	Seq No 2		

5. После запуска ОРС симулятора мы увидим следующее окно:

6. Наиболее важная для нас информация — это номер порта и имя сервера, которые будут использоваться для настройки драйвера OPC-клиента в ADISRA SmartView.

Port Number	62640
Server Name	IntegrationObjects/ServerSimulator

- 7. Следующим шагом будет настройка OPC UA сервера на прием соединения от анонимного пользователя без дальнейшего шифрования.
- 8. Откроем пункт Settings в верхнем меню кликнем на Security Policies option. Откроется новое диалоговое окно.
  - Убедитесь, что режим Security Mode установлен в None
  - Убедитесь, что галочка Anonymous справа также установлена



Server Configuration	VSeversionlater	
SessionId	Security Policies	
Subscriptions Subscriptions 2. Status: Runnin Status: Runnin Status: Runnin	User Identity Tokens Anonymous Username Username Password Confirm Password User list User Name	Add User Remove Clear

Примечание: Так выглядит нужная нам конфигурация ОРС симулятора.



erver Conf	iguration	2640/IntegrationObjects/Ser	verSimulator		
SessionId	Name	1.	Configuration	×	
		opc.tcp Connection	1 [		
		TCP Port*		62640 🔁	
C. K. L.		Server Name*	IntegrationObjects/ServerSimulator		
Subscriptions -	Publishing Inter				
2	100	https Connection			
а. -	100	HTTPS Port*		62641 🚍	
		Server Name*	IntegrationObjects/ServerSimulator		
Status: Running	Current Time:	OPC UA Server Settin	gs ]		
		Update Rate (ms)*		1000	
		*Restart	the application for changes to take ef	lect	

9. Ещё раз проверим имя сервера и номер порта, открыв Setting ->Configuration в верхнем меню.

 Мы будем подключаться с ОРС UA серверу по протоколу TCP, номер порта будет 62640, а имя сервера - IntegrationObjects / ServerSimulator. Если Вы используете для своего ознакомительного проекта другой (собственный) ОРС UA сервер, самое время сейчас узнать эти параметры для Вашего ОРС UA сервера.

10. В следующем подразделе мы создадим документ (драйвер) ОРС UA-клиента.



#### 18.1.2 Создание документа ОРС UA Client

Для настройки этого драйвера выполним следующие шаги:

 Создадим новую папку OPC UA Client в дереве навигации: Drivers -> New Folder -> OPC Clients -> OPC UA Client.



2. Во вновь созданной папке OPC UA Client создадим новый документ, кликнув на папке правой кнопкой мыши и выбрав New Document.

Drivers		
P 🐌 Modbus		
4 🐌 OPC UA Client	New Document	Ctrl+N
Images	New Folder	Ctrl+F
😻 User Functions Libr	Remove	Del
System Functions Lun	1017	1.

- 3. Откроем созданный документ и начнем настройку соединения с сервером.
- 4. В разделе Server Settings заполним поля Target (IP адрес OPC UA сервера), а также порт (Port) и имя сервера (Server Name).

Server Settings					
Target:	127.0.0.1	Port	62640		
Server Name:	IntegrationObjects/Sen	verSimulator			
Action:	Read/Write	✓ Writ	Async	*	

Примечание: Имя и порт нашего ОРС UA сервера мы запомнили при его



настройке.

- 5. В поле Action настроим направление передачи данных.
  - Read: OPC UA клиент сможет ТОЛЬКО читать данные с OPC UA Сервера.
  - Read/Write: Возможен двунаправленный обмен данными между клиентом и сервером.
  - Write: Данные передаются ТОЛЬКО от ОРС UA клиента к ОРС UA серверу.

Установим для нашего проекта режим Read/Write.

- 6. В поле Write можно установить режим записи (синхронный или асинхронный).
- 7. В поле Enable выберем значение True.
- 8. Следующим шагом буде настройка безопасности. Рассмотрим, какими возможностями настройки мы располагаем:
  - Пользователь (User):
    - Anonymous: Позволяет подключаться к серверу без идентификации
    - Username/Password: Требует идентификации по имени и паролю
  - Security Mode: Возможны три режима безопасности.
    - None: Соединение разрешается без проверок безопасности.
    - Sign: Соединение должно иметь цифровую подпись.
    - Sign & Encrypt: Соединение должно иметь цифровую подпись и шифрование.
  - Security Policy: Возможны четыре уровня политики безопасности.
    - None: Соединение происходит без дополнительных условий.
    - Basic128Rsa15: Сервер осуществляет 128-битное AES шифрование с Sha1.
    - Basic256: Сервер осуществляет 256-битное AES шифрование с Sha1.
    - Basic256Sha256: Сервер осуществляет 256-битное AES шифрование с Sha256.
- 9. Нам необходимо сделать эти настройка в соответствие с настройками на стороне ОРС ПА Сервера

Г	U	UA	Ce	Ьв	ep	d

Anonymous		
Username:		
Password:		
Security Settings Security Mode None Sign Sign & Encript	Security Policy None Basic128Rsa15 Basic256 Basic256Sha256	

User: Anonymous



- Security Mode: None
- Security Policy: None
- 10. И, наконец, добавим в список тегов теги из ОРС UA сервера. Нам потребуются три тега для связи с этим ОРС UA сервером. Самый быстрый способ создать тег это напрямую добавить его в поле Tag и нажать Enter. Если ADISRA SmartView определит, что такого тега не существует, то нам будет предложено его создать.
  - Процедура создание тега

Steps	Images
Добавим тег Tag1OPC в поле Tag	Tags List Tag Item Tag1OPC
Переместим курсор или нажмем Enter на клавиатуре. ADISRA SmartView определит, что тег не существует и предложит нам его создать. Нажмем Yes.	New Taig  New Taig  The tag Tag 10PC' does not exist. Do you mant to create a new tag?  Yes No
	🕱 New Tag 🗕 🗆 🗙
	Name: Tag1OPC
	Doc: TAGS ~
Выберем документ TAGS и тип Float для нового тега и нажмем OK	Type: Float
	Array: 0 - No Array 🗸
	OK Cancel

• Поставим в соответствие нашему новому тегу ОРС-тег из сервера ОРС UA.

Steps	Images		
Двойной клик на поле Item откроет нам кнопку «». Кликнув по ней, мы откроем диалоговое окно ОРС-браузера для выбора ОРС-тега.	Tag Tag TagTiCH	lizer I	



	¥	OPCUA Browser	- 0 ×
Если соединение с ОРС сервером	127.0.0.162640/integrationObje	ctu/ServerSimulator	
произойдет успешно, мы увидим	Refresh Branch		
структуру тегов ОРС UA сервера.	III - Server UA		^
Откроем папку Real Time Data,	ii - 3 Server		
выберем тег Tag1 и нажмем ОК.	Haal Time Data		
Если мы не увидим дерева тегов ОРС	Tag2		
сервера, нам придется еще раз	Pest Carl		
проверить, что наш ОРС UA сервер	Tags		
запущен и что настройки для	Tagit		
подключения к нему выполнены	Tagi0		
корректно.	Tenti	OK	Cancel
	1		
_	Taos List		
Если все проидет успешно, в список	Tag	Item	
тегов добавится элемент, как на	ray	item	
изображении справа.	Tag1OPC	ns=2;s=Tag1	
	-		

• Теперь добавим еще два тега в список. Повторим все вышеописанное для следующих тегов:

Tag2OPC -> Tag2 Tag11OPC -> Tag11

#### 11. В результате мы должны получить следующее:

Taiget	127.0.0.1	Pom	62640			
Server Navim	VdegrationDbjects/ServerSimulator					
Action	Read/Wite -	Sync Writer	RALSE	1		
Users						
Distriante						
Palevort						
O Sgr O Sgr & by	nipi.				Tome     Tome	
Tegs Lot						
Tag	Dev.					
Tagi türc	res2ssTeg11					
TagICPC	rea2saTag1					
CROSpet	rs=2s=Tag2					

Примечание: Tag1 и Tag2 по умолчанию не будут симулироваться в ОРС UA сервере, а Tag11 будет периодически менять свои значения, т.е. симулироваться.



12. Сохраним документ под именем OPC\_UA\_Client.

18.1.3 Создание экранных форм отображение тегов из ОРС UA сервера Для создания такого экрана сделаем следующие шаги:

1. Создадим новую папку Drivers в разделе Graphics

Data Types			
4 🥃 Graphics	New Document	Ctrl+N	
	New Folder	Ctrl+F	
	Rename	Ctrl+H	
	Remove	Del	
	Save HTML	Alt+W	
	Collapse All		

2. Создадим новый документ в папке Graphics с именем Drivers



- 3. Основой нашего экрана будет групповое поле (GroupBox) в который мы добавим три значения тегов из OPC UA сервера. Мы также используем новые элементы, выглядящие как кружки (элементы типа Ellipse) для отображения качества связи с OPC.
- 4. В результате попробуем добиться примерно следующего:



• Настройки элемента GroupBox

Object	Тор	Left	Width	Height
GroupBox1	21	24	857	255

<ul> <li>Настройки индикаторов типа Ellipse</li> </ul>								
Object		Тор	Left	Width	Height			
Ellipse1		117	298	20	20			
Ellipse2		157	298	20	20			
Ellipse3		196	298	20	20			

• Дополнительные настройки анимации индикаторов Ellipse

Выдел	им элемент Ellipse	00 Ta	g1 OPC	@Tag1OP	с
-	Найдем Кисти (Brushes) в окне свойств. Выберем свойство Fill (Заливка) Выберем Dyn (Динамический цвет) Введем такое выражение: @Tag1OPC.Quality == 192?1:0 В результате, если качество тега булет.	Drustes     Drustes     Difference     Solid Total     Device Colors     Sopraces     Solid     Volue	Ing Oyi Ky Citegaorcop Cale	a el asito: == 102 / 1	: E Remova
-	в результате, если качество тега оудет равно 192 (хорошее), то эллипсы будут	0			x
	закрашены зелёным. Если качество будет отличным от хорошего, то		_		X
	индикаторы станут красными.				

Применим подобную конфигурацию ко всем эллипсам. Нам придется каждый раз изменять имя тега, чтобы получить верные результаты для каждого из тегов. Выражения, как Вы видите, на самом деле выполнены в соответствие с нормами языка С#.

• Настройки Label



Object	Text	Тор	Left	Width	Height
Label1	Tag1 OPC	117	333	70	19
Label2	Tag2 OPC	157	333	70	19
Label3	Tag11 OPC	196	333	70	19

#### Настройки TextBox

Object	Text	Тор	Left	Width	Height
TextBox1	@Tag10PC	112	415	128	29
TextBox2	@Tag2OPC	152	415	128	29
TextBox3	@Tag110PC	192	415	128	29

5. Сохраним наш документ.

## 18.2 Документ драйвера Modbus

Документ драйвера позволяет нам описывать обмен данными между ADISRA SmartView и внешним оборудованием, и определять правила и протокол этого обмена. Конечно, для каждого драйвера внешний вид его настроек выгладит по-разному. В этом разделе мы для примера создадим описание обмена данным по протоколу Modbus, используя эмулятор устройств Modbus TCP.

В этом разделе мы загрузим и установим симулятор Modbus, а затем настроим связь с ним в ADISRA SmartView. Конечным результатом будет экран, на котором отслеживаются теги, подключенные к внешнему эмулятору Modbus TCP.

18.2.1 Загрузка и установка внешнего эмулятора Modbus (mod\_RSsim)

- 1. Загрузим на свой диск файл по данной ссылке:

   <u>https://adisra.com/wp-content/uploads/2020/03/ADISRA-SmartView-GettingStarted-DriverSimulator.zip</u>
- 2. Распакуем и запустим эмулятор.

Nome	Data de modificaç	Tipo	Tamanho
👸 mod_RSsim.exe	08/04/2016 08:22	Aplicativo	1,177 KB

3. В результате мы увидим следующее диалоговое окно.



Address: C	Hex (	Dec	1/0	Coll	Output	1 I	•		Fint	de	cimal.		P	ot N	ODB	US TO	CP/IP		*	Clor
Address	+15	+14	+13	12	+11	+10	+9	+0	+7	+6	+5	+4	+3	-0	+1	+0	Total			
1.16	0	0	0	0	0	0	0	0	0	0	8	0	0	0	0	0	0000	- 10		
17.32	õ	ñ	ñ	ñ	ñ	ñ	õ	ñ	ñ	ñ	ñ	ñ	ñ	ñ	ě.	ñ	0000			
33.48	ŏ	ŏ	ŏ	ň	õ	ő	ŏ	õ	ň.	ň	ŏ	ŭ	ň	ŭ	ě.	Ŭ.	0000			
49-64	õ	ō.	ő	0	õ	õ	õ.	õ	õ.	0	õ.	õ	ō.	ō	ō.	ñ	0000			
65-80	õ	õ	õ	õ	õ	õ	ŏ	õ	õ	õ	ŏ	õ	õ	õ	õ	õ	0000			
81-96	ŏ	ŏ	ŏ	ŏ	ŏ	ŏ	ŏ	ŏ	ŏ	ŏ	ŏ	õ	ŏ	õ	ŏ	õ	0000			
97-112	ŏ	ŏ	õ	ŏ	ŏ	õ	ŏ	ŏ	ŏ	ŏ	ŏ	õ	ŏ	õ	ŏ	õ	0000			
113-128	ŏ	ŏ	õ	ŏ	õ	ŏ	ŏ	ŏ	ŏ	ŏ	ŏ	ŏ	ŏ	õ	ŏ	ŏ	0000			
129-144	õ	õ	õ	õ	õ	õ	õ	õ	õ	õ	õ	õ	õ	õ	õ	ō	0000			
145-160	ŏ	ŏ	ŏ	ŏ	ŏ	ŏ	ŏ	ŏ	ŏ	ŏ	ŏ	ŏ	ŏ	ŏ	ŏ	ŏ	0000			
161-176	0	0	0	0	0	0	Ö.	0	0	0	0	0	Ö.	0	0	0	0000			
177-192	ō	ō .	Ū.	Ū.	Ō	ō.	ō	Ū.	ō	Ū.	ō	Ū.	ō	Ū.	ō.	ō	0000			
193-208	0	Ó.	0	0	0	0	Ö.	0	Ö.	0	Ö.	0	Ô.	0	Ö.	0	0000			
209-224	0	Ū.	0	0	0	0	Ō	0	Ō	0	Ō	0	Ō	0	Ū.	0	0000			
225-240	ö	ō .	Ő.	0	0	Ö.	ō.	0	ō.	Ū.	ō.	Ū.	ō.	Ū.	ō.	Ū.	0000			
241-256	0	0	0	0	0	0	0	0	0	0	0	0	Ū.	0	0	0	0000			
257-272	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0000			
273-288	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0000			
289-304	0	Ū.	0	0	0	0	Ū.	0	Ū.	0	Ū.	0	Ū.	0	0	0	0000			
305-320	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0000			
321-336	0	0	0	0	0	0	Ö.	0	0	0	0	0	0	0	0	0	0000			
337.352	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0000			

4. Перед настройкой документа драйвера в ADISRA SmartView нам потребуется важная информация о нашем устройстве. Откроем настройки Ethernet TCP / IP, щелкнув выделенную на рисунке ниже кнопку:

raineringe (e)	ist travenie	m ang m(1)	(and it is	and the		
vddress : C	Hex 🕫	Dec	1/0 0	tu O lio	Ethemet TCP/IP Settings	
Address JE	+15	14 +	13 +1	12 +	Status OK OK	
7-32	0 0	0	0	0	Local IP PC Cancel	
9-64 5-80	0 0	0 0	0	0	Remote IP 127.0.0.1	
1-96 7-112	0 0		0	0	Server settings	
29-144	0 0	0	0	Ő	# Server connections 10	
61-176	0 0		0	0	Port (502) 502	
33-208 99-224	0 0		0	0	Alternale port 501	
25-240	0 0	, ŭ	Ő	ŏ	Station ID 1	
7-272 3-288	0 0	0	0	0	Bestonniveness [ms] 0 (0 to 10 000 ms)	
19-304 15-320	0 0	0	0	0	Linger on close of socket (SO_LINGER)	
(1-336	0 0	0 0	0	0		

5. В этом окне содержится важная для нас информация: номер порта TCP/IP и IP адрес нашего устройства

Port Number	502
Remote IP	127.0.0.1



6. Теперь ознакомимся поближе с интерфейсом симулятора. Он содержит различные типы ввода-вывода и адреса, настраиваемые в главном окне симулятора. Таким образом, при тестировании связи мы сможем видеть текущие значения.

Address : C	Hex 🐨 I	Dec 1/0	Holding F	legisters		Fire: 1000	32	Plot: M	OBUS TO	РЛР	•	IT Clor
Address	+0		+2	+3	+4	+5	+6	+7	+8	+9		
0001-40010	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000		
0011-40020	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000		
0021-40030	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000		
0031-40040	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000		
0041-40050	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000		
0051-40060	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000		
0061-40070	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000		
071-40080	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000		
081-40090	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000		
0091-40100	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000		
101-40110	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000		
111-40120	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000		
121-40130	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000		
131-40140	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000		
141-40150	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000		
151-40160	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000		
161-40170	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000		
171-40180	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000		
181-40190	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000		
191-40200	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000		
201-40210	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000		
211-40220	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000		

Примечание: Мы будем использовать такие типы данных Modbus, как Holding Registers и Analogue Inputs

- Выберем тип Holding Registers в поле (I/O)
- Выберем тип float 32 в поле формат (Fmt)
- 7. Теперь вернемся к ADISRA SmartView и создадим новый документ типа Modbus driver. Подробности этого процесса опишем ниже.

### 18.2.2 Создание документа типа Modbus Driver

Выполним следующие действия:

1. Создадим новую папку в области Driver в дереве навигации: Drivers -> New Folder - > Modbus.





2. В результате будет создана новая папка с именем Modbus. Теперь создадим в этой папке новый документ, кликнув на New Document.

	New Document	Ctrl+N
The second s	New Folder	Ctrl+F
	Rename	Ctrl+H
	Remove	Del
	Expand All	

- 3. Откроем созданный документ и начнем настройку подключения к Modbus эмулятору.
- 4. В разделе General Settings установим значение 127.0.0.1 в поле Host/IP и 502 в поле Port.

General Se	ttings					
	conga		-	-1.		
Host/IP: 1	27.0.0.1	Port:	502	Enable:	TRUE	. <del>.</del> .

5. В полях Read установим автоматическое обновление с интервалом в 1000 ms, а в поле Write - по изменению тега:



Read	- mm - )		
RefreshTime(ms): 1000			
Write			
Trigger:	and a second sec		
Tag Changed			

6. Теперь добавим пару тегов в список и сопоставим их с эмулятором. Самый быстрый способ создать тег — это напрямую добавить его в поле Tag, а затем переместить курсор или нажать Enter. ADISRA SmartView поймет, что тега не существует, и предложит создать его.

Steps	Images	
Добавим тег TagModBus1 в поле Tag	Nor I no         NA         Advirus         I spec         Long           IncMediue1         Der Menix         1         2 P         1	phe -
Переместим курсор или нажмем Enter. ADISRA SmartView подтвердит, что тег не существует, и предложит создать его. Нажмем Yes	New Tag     New Tag     New Tag     The tag TagModBus1 <sup>o</sup> does not exist. Do you want to create a     new tag?     Yes No	
Выберем документ с именем TAGS, установим тип Float и нажмем OK.	New Tag     Image: Constraint of the second se	
	OK Cancel	

• Создание тега

• Поставим в соответствие нашему тегу адрес в эмуляторе сети Modbus:

Steps	Images	
	Tags List	
Выберем тип регистров ввода-	Tag	I/O:
вывода HoldingRegister	TagModBus1	HoldingRegister ~



		Tegs List						
Выберем адрес, рав	ный 1	Tag	640	×	Address			
		TagModBus1	H	oldingRegister -				
		Tegsiar	1942				0.22	-
Выберем формат ре	гистра Float	142	10	Address		type	Long	120
N		TegModBusT	HaidingRegister	* 1		naw	7 1	
Выберем формат ре	гистра Float	Tegi List Teg TegModBus1	IO HaldingRegister	Address + 1		Type Roat		1

 После этого добавим один тег типа InputRegister в список тегов. Для этого выполним следующие действия:

SmartView	Modbus Simulator
TagModBus2 (Float)	Address 1 -> Type Float -> Lengths 1

#### 7. Окончательный вид нашего документа:

General Settings									
Host/IP: 127.0.0.1			Port:	502	Enable:	TRUE	*		
Read Trigger: RefreshTime(ms):	1000								
Write									
<ul> <li>☐ Trigger:</li> <li>☑ Tag Changed</li> </ul>									
☐ Trigger: ☑ Tag Changed Tags List	- en .								
☐ Trigger: ☑ Tag Changed Tags List Tag			Address:				Туре:		Lengti
☐ Trigger: ✓ Tag Changed Tags List Tag TagModBus1	I/O: HoldingRegister	~	Address:				Type: Float	Ŷ	Lengti
☐ Trigger: ✓ Tag Changed Tags List Tag TagModBus1 TagModBus2	l/O: HoldingRegister InputRegister	~ .	Address: 1				Type: Float Float	>	Length 1 1

8. Сохраним документ под именем Drivers1.

#### 18.2.3 Вывод переменной из драйвера Modbus на экранную форму

1. Откроем экранную форму Drivers и добавим в нее еще одну группу (элемент типа GroupBox) с именем Modbus driver. В эту группу добавим две метки и два значения тегов, а также два эллипса для их индикации.



OPC UA Cient		
	Tan1 OPC	@Tus1086
	U lightere	@ hag love
	Tag2 OPC	@Tag2OPC
	Tag11 OPC	@Tag110PC
Modbus Driver		
Modbus Driver		
Modbus Driver	Holding Bea	tter ØllanMorthur1
Modbus Driver	Holding Regi	ter @TagModtus1
Micdbus Driver	<ul> <li>Holding Regi</li> <li>Analogue Ing</li> </ul>	tter @TagModBus1 ut @TagModBus2
-Modbus Driver	<ul> <li>Holding Regi</li> <li>Analogue Inp</li> </ul>	tter @TagModBus1 ut @TagModBus2

2. В итоге наш экран должен будет выглядеть примерно так:

## • Для этого нужно так настроить GroupBox

Object	Тор	Left	Width	Height
GroupBox2	331	24	857	255

#### • И так – эллипсы

Object	Тор	Left	Width	Height
Ellipse4	427	298	20	20
Ellipse5	466	298	20	20

#### • А так должна выглядеть анимация эллипсов

Select the Ellipse	O O O B Holding Register     ©TagModBu     ©TagModBu
<ul> <li>Go to the Property Window and locate the Brushes.</li> <li>Select the Fill property</li> <li>Select the Dyn (Dynamic Color)</li> <li>Set the expression as: @TagModBus1.Quality == 192 ? 1 : 0</li> </ul>	Brushes     Fill     Border     Sold Grad Img Dyn Dyn Fill     Default Colon     Expression: @TagMadBus1.Quality == 182 ? 1 : 0
- In the runtime, if the quality of the Tag is equals to 192 (good quality), the ellipse will get the green color. In case the quality is not good (different than 192), it will get red color	Value Color Renove D X 1 X X X X X



Помните, что такую анимацию нужно определить для обоих эллипсов, указав нужные теги для каждого из выражений.

• Установки меток

Object	Text	Тор	Left	Width	Height
Label4	Holding Register	427	333	126	19
Label5	Analogue Input	467	333	126	19

• Установки текстовых полей

Object	Text	Тор	Left	Width	Height
TextBox4	@TagModBus1	422	445	128	29
TextBox5	@TagModBus2	462	445	128	29

3. Важно помнить, что аналоговый вход в нашем Modbus эмуляторе имеет параметр read only, как это и бывает в большинстве реальных систем. Т.е. мы НЕ МОЖЕМ изменить его, записав новое значений по этому адресу! Если мы хотим в нашем проекте увидеть, как он изменится на экране, нам нужно будет изменить его непосредственно в Modbus эмуляторе.

Address : 🦳	Hex 🕫 [	Dec 1/0	Analogue	Inputs (30	•	Fmt: float	32 💌	Prot: MO	DOBUS TO	P/IP	☐ Clone
Address	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	
30001-30010	0.000	35.00 <sub>m</sub>	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	
30011-30020	0.000	0.0001-5	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	
30021-30030	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	
30031-30040	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	
30041-30050	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	
0051-30060	0.000	0.000	0.000	0.00				+ 20000	1 ×	0.000	
30061-30070	0.000	0.000	0.000	0.00	Ed	it float 3	2 value	at 30000	2 <u> </u>	0.000	
30071-30080	0.000	0.000	0.000	0.00						0.000	
30081-30090	0.000	0.000	0.000	0.00	Value-	6510101	10		OK I	0.000	
30091-30100	0.000	0.000	0.000	0.00	value.		-	_	915	0.000	
30101-30110	0.000	0.000	0.000	0.00				0	Incel	0.000	
30111-30120	0.000	0.000	0.000	0.00					31000	0.000	
30121-30130	0.000	0.000	0.000	0.00	Contraction of the local division of the loc		He works			0.000	
30131-30140	0.000	0.000	0.000	0.000	0.000	U.UUU	0.000	U.UUU	0.000	0.000	
30141-30150	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	
30151-30160	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	
30161-30170	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	
30171-30180	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	
30181-30190	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	
30191-30200	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	
30201-30210	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	
30211-30220	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	

- Выберем раздел I/O Analogue Inputs
- Выберем Address 1
- Сделаем двойной клик на ячейке
- В открывшемся окне изменим текущее значение нашего Modbus регистра



 После того, как мы закончим настройку нашего учебного проекта, и запустим его на исполнение, увидеть значения переменных из драйверов можно буден в закладке Drivers.

-	GraphicArts	History	Recipes Repor	Services Ti	iggers Tu	nneling	UserFun	tionLibrary	Drivers		
	[ OPC	UA Client									
					😑 Tag	1 OPC	123.0				
					🔵 Tag	2 OPC	444,0				
					🔵 Tag	11 OPC	15.0				
	r=Mod	bus Driver									
					Hol	ding Regis	ster 1	5.0			
					Ana	logue Inp	ut	5.0			

4. Сохраним нашу экранную форму.



## 19. Документы туннелирования Tunnelling

Документы типа Tunnelling дозволяют настроить обмен данными между элементами ADISRA SmartView. Такой обмен может производиться в любом направлении между двумя тегами, либо в одном направлении - от выражения к тегу. Невозможно настроить передачу данных от тега к выражению.

Примечание: Если теги, настроенные в документе Tunnelling, относятся к разным типам (например, тег Float пытается записать свое значение в тег String), то Tunneling пытается автоматически выполнить необходимые преобразования значений.

## 19.1 Создание документа типа Tunneling

Для создания такого документа выполним следующие шаги:

- 1. Создадим новый документ в разделе Tunneling в деревне навигации.
- 2. Выделим первую свободную строку в левом столбце Left Expression.
- 3. Введем следующее выражение: (@Structure.Member1[0] \* 10) / 2
- 4. Нажмем кнопку Verify справа от строки.
- 5. В столбце Direction выберем значение <->.
- 6. Выделим первую свободную строку в правом столбце Right Expression.
- 7. Введем следующий тег: @Structure.Member1[1]
- 8. Нажмем кнопку Verify справа от строки. В результате проверке правильности

заполнения этого поля мы увидим значок (Ф), указывающий на ошибку. При наведении курсора мыши на значок появляется следующее сообщение:

Invalid Direction. You cannot configure (@Structure.Member1[0] \* 10) / 2 to receive a value. (Неверное направление. Выражение ... не может быть получателем данных.)

- 9. Для того чтобы исправить эту ошибку, выберем вариант ->.
- 10. Повторим аналогичные процедуры еще для двух строк:
  - Left Expression: @Structure.Member2[0,0], Direction: <-

Right Expression: (@Structure.Member2[1,1] + 100) / 5

• Left Expression: @Structure.Member3[0,0,0], Direction: <->

Right Expression: @Structure.Member3[1,1,1]



11.	Сохраним документ под именем	Tunneling.
	conpannin Hondinen	· ann onn g.

Filters							
	Left:	All		•	Right	All	
	Group:			*	Group:		2
Items							
	lf	the item direction is con	figure	ed with <->, th	ie startup sj	ynchron	ization will be: Left -> Right
Le (@	ft Expre	ssion		Direction			Right Expression
(0	Structu	re.Member1[0] 10/72		<u> </u>			(@Structure Member2[1]
	Structur	e Member3(0,0)			(->		@Structure Member3[11
	Sciecter	emenoeis[0,0,0]					e su detta e memoers (2, 2,
	_						



### 19.2 Создание и настройка экрана для иллюстрации туннелирования

Для иллюстрации работы механизма туннелирования нам понадобится всего шесть меток (объектов Label) и шесть текстовых полей (объектов TextBox).

- 1. Настройки меток (элементов Label):
  - Label1: Text: Structure.Member1[0]
  - Label2: Text: Structure.Member1[1]
  - Label3: Text: Structure.Member2[0,0]
  - Label4: Text: Structure.Member2[1,1]
  - Label5: Text: Structure.Member3[0,0,0]
  - Label6: Text: Structure.Member3[1,1,1]
- 2. Настройки текстовых полей (элементов TextBox):
  - TextBox1: Text: @Structure.Member1[0]
  - TextBox2: Text: @Structure.Member1[1]
  - TextBox3: Text: @Structure.Member2[0,0]
  - TextBox4: Text: @Structure.Member2[1,1]
  - TextBox5: Text: @Structure.Member3[0,0,0]
  - TextBox6: Text: @Structure.Member3[1,1,1]
- 3. Сохраним полученную экранную форму под именем Tunneling.

Structure.Member1[0]	@Structure.Member1[0]	Structure.Member1[1]	@Structure.Member1[1]
Structure.Member2[0,0]	@Structure.Member2[0,0]	Structure.Member2[1,1]	@Structure.Member2[1,1]
Structure.Member3[0,0,0]	@Structure.Member3[0,0,0]	Structure.Member3[1,1,1]	@Structure.Member3[1,1,1]



# 20. Библиотека функций пользователя (User Function Library Documents)

Пользователь ADISRA SmartView может создавать библиотеку из своих собственных функций. Все созданные однократно библиотечные функции могут в дальнейшем быть использованы в других проектах.

Примечание: При создании пользовательских функций нельзя пользоваться тегами!

## 20.1 Создание библиотеки пользовательских функций

Для создания небольшой собственной библиотеки функции выполним следующие шаги:

- 1. Создадим новый документ в разделе User Function Library в дереве навигации.
- 2. В поле документа напишем следующий код:

```
int sum (int a, int b)
{
  return a+b;
}
string HelloWorld (string tagName)
{
  tagName = Hello World;
  SVApplications.Output(Hello World function successfully executed!);
  return tagName;
}
```

- 3. Нажмем кнопку Verify в верхней ленте для проверки правильности написания нашего скрипта.
- 4. Сохраним документ под именем Library1.





## 20.2 Создание и настройка экранных форм для иллюстрации использования пользовательских функций

- 1. Для этого экрана нам потребуется создать два новых дополнительных тега:
  - Name: LibrarySum, Type: Integer, Array: 1 Dimension, Column(X): 3
  - Name: LibraryHelloWorld, Type: String, Array: 0 No array
- 2. Создадим новую папку UserFunctionLibrary в разделе Graphics.
- 3. Создадим в ней новый документ.
- После создания двух вышеуказанных тегов создадим на экране два групповых блока (GroupBox), один для функции LibrarySum, а второй для функции HelloWorld со следующими настройками:
  - GroupBox1: Text: Library Sum, Width: 290, Height: 193
  - GroupBox2: Text: Library HelloWorld, Width: 290, Height: 110
- 5. Для блока LibrarySum добавим три метки (Label), три текстовых поля (TextBox) и одну кнопку (Button) со следующими настройками:
  - Label1: Text: LibrarySum[0] 1st Portion, Width: 150, Height: 18
  - Label2: Text: LibrarySum[1] 2nd Plot, Width: 150, Height: 18
  - Label3: Text: LibrarySum[2] Result, Width: 150, Height: 18
  - TextBox1: Text: @LibrarySum[0], Width: 100, Height: 25
  - TextBox2: Text: @LibrarySum[1], Width: 100, Height: 25
  - TextBox3: Text: @LibrarySum[2], Width: 100, Height: 25
  - Button: Text: Calculate, Width: 95, Height: 25, Mouse Up:

@LibrarySum[2] = Library1.sum(@LibrarySum[0], @LibrarySum[1]);

- 6. А для блока LibraryHelloWorld одну метку (Label), одно текстовое поле (TextBox) и одну кнопку (Button) со следующими настройками:
  - Label: Text: LibraryHelloWorld, Width: 115, Height: 18
  - TextBox1: Text: @LibraryHelloWorld, Width: 141, Height: 25
  - Button: Text: HelloWorld, Width: 95, Height: 25, Mouse Up:

@LibraryHelloWorld = Library1.HelloWorld(@LibraryHelloWorld);



7. Сохраним документ под именем UserFunctionLibrary. Конечный результат должен выглядеть приблизительно так:

-Library Sum		Library HelloWorld	Million constrained the
LibrarySum[0] - 1st Portion	@LibrarySum[0]	LibraryHelloWorld	@LibraryHelloWorld
LibrarySum[1] - 2nd Plot	@LibrarySum[1]	He	elloWorld
LibrarySum[2] - Result	@LibrarySum[2]		
Calculat	e		
	Anna		



## 21. Настройка библиотеки изображений

Библиотека изображений (Image Library) – набор графических файлов, который будут использованы в проекте. Более того, в проекте ADISRA SmartView могут быть использованы изображения только из библиотеки изображений данного проекта.

## 21.1 Добавление нового изображения в библиотеку

Для нашего проекта нам понадобится изображение, которое мы разместим на главном экране нашего приложения. Например, такое:



Вы можете использовать любое иное изображение по Вашему вкусу. Для того, чтобы добавить изображение в библиотеку, сделаем следующее:

- 1. Откроем документ Images в дереве навигации.
- 2. Нажмем кнопку Insert Image в ленте верхнего меню.
- 3. В открывшемся окне найдем в нашей файловой системе нужное нам изображение.
- 4. Выберем нужный файл и нажмем кнопку Open для добавления изображения в библиотеку.



## 22. Графические объекты с расширенным функционалом

В этой главе остановимся подробнее на том, как создавать и использовать некоторые продвинутые (Advanced) графические объекты, о которых мы ранее ещё не упоминали.

## 22.1 Объекты Экран (Screen)

Объекты типа Экран (Screen) предназначены для создания гибких раскладок экранных форм проекта. Эти объекты позволяют резервировать на экранных формах определенные области, в которые мы можем выводить контент из других экранных форм.

Для настройки объекта типа Экран проделаем следующие операции:

- 1. Создадим новый документ в разделе Graphics и сохраним его под именем GraphicArts.
- 2. Для нашего примера создадим еще две экранные формы, которые мы будем отображать в объекте типа Экран. Для этого создадим сначала папку Screens в разделе Graphics.
- 3. Создадим новый документ в папке Screens под именем ObjectScreen.
- 4. Создадим еще один новый документ в папке Screens и назовем его Trend.
- 5. Откроем документ GraphicArts, перейдем в секцию Interface Objects раздела Graphics верхней ленты и выберем элемент Screen.
- 6. Кликнем один раз левой клавишей мыши в районе точки с координатами (0;0), и на экране появится элемент Screen с размерами по умолчанию.
- 7. Выделим его и настроим его свойства следующим образом:
- 8. Width: 1024, Height 646, Top 28, Left 0.
- 9. Выберем элемент типа Кнопка (Button) из секции Basic Objects раздела Graphics верхней лены.
- 10. Кликнем один раз левой клавишей мыши в районе точки с координатами (0;0), и на экране появится элемент Button с размерами по умолчанию.
- 11. Выделим эту кнопку и настроим её свойства следующим образом:
  - Text: Display Screen, Width: 100, Height: 25, Top 0, Left: 0.
  - Mouse Up: Screen1.SetScreen(Screens\\ObjectScreen);
- 12. Выберем ещё раз элемент типа Кнопка (Button) из секции Basic Objects.
- 13. Кликнем один раз левой клавишей мыши в районе точки с координатами (0;0), и на экране появится элемент Button с размерами по умолчанию.
- 14. Выделим эту кнопку и настроим её свойства следующим образом:
  - Text: Trend, Width 100, Height 25, Top 0, Left 120.
  - Mouse Up: Screen1.SetScreen(Screens\\Trend);
- 15. Сохраним изменения, сделанные в документе.



При нажатии на кнопку Display Screen во время исполнения нашего проекта в поле объекта Screen отобразится содержимое экранной формы ObjectScreen. А при нажатии на кнопку Trend в поле нашего элемента Screen отобразится содержимое экранной формы Trend.





## 22.2 Настройка экранной формы ObjectsScreen

На примере следующей экранной формы покажем, как можно использовать дополнительный функционал в объектах типа Screen.

### 22.2.1 Использование TextBox для ввода пароля

Элемент TextBox возможно настроить таким образом, что вместо вводимого текста на экране будет отображаться исключительно символы «\*». Например, это важно для ввода паролей, а также для других случаев, когда необходимо безопасно вводить информацию на экране. Чтобы настроить объект таким образом, выполним следующие действия:

- 1. Откроем документ ObjectScreen в разделе Graphics -> Screens.
- 2. Выберем в верхней ленте, раздел Graphics, элемент TextBox.
- 3. Кликнем один раз левой клавишей мыши в районе точки с координатами (0;0), и на экране появится элемент Textbox с размерами по умолчанию.
- 4. Выделим его и настроим его параметры следующим образом:
  - Top: 0, Left: 0.
  - Width: 120, Height: 25.
  - Установим галочку в поле Password.
- 5. Создадим еще один элемент TextBox со следующими параметрами:
  - Top: 25, Left: 0.
  - Width: 120, Height: 25.
- 6. Создадим справа от этих текстовых полей элемент Button с текстом Test Password, и в качестве обработчика Mouse Up в этой кнопке введем следующий скриптt: TextBox2.Text = TextBox1.Text;.
- 7. Сохраним изменения в документе.

	Test Password
--	---------------

Во время исполнения проекта все, что мы будем вводить в TextBox с установленной галочкой Password, будет тут же скрыто звездочками.

•••••
-------

Но после нажатия на кнопку с надписью Test Password содержимое первого текстового поля с паролем будет скопировано во второе, где будет отображено уже полностью, что даст нам возможность проверить правильно ввода.


#### 22.2.2 Добавление изображения в дизайн кнопки

На поверхности кнопки (объекта типа Button) может быть расположен не только текст, но и графическое изображение. Попробуем сделать это.

- 1. Откроем документ ObjectScreen в разделе Graphics -> Screens.
- 2. Выделим на нашем экране кнопку, созданную нами ранее, в пункте 22.2.1, для отображения (проверки ввода) пароля.
- 3. В его свойствах, в разделе Button Style найдем свойство Content Style и выберем для него вариант ImageAndTextHorizontal.

Configuration	
Text	Button
<ul> <li>Button Style</li> </ul>	ImageAndTextHoriziontal
Content Style	ImageAndTextHoriziontal ~
Source File	🔀

- 4. В поле Source File, кликнем на кнопке «...» и откроем окно выбора файлов.
- 5. Выберем, например, файл с логотипом ADISRA.
- 6. Сохраним изменения в документе.

#### 22.2.3 Объект типа TextBox с динамически изменяемым фоном

Некоторым объектам в ADISRA можно назначить динамический цвет. И это цвет будет меняться в соответствии с заданными условиями. В этом примере мы используем объект TextBox, но этот параметр доступен и для многих других объектов.

Test Password

Для этого выполним следующие шаги:

- 1. Откроем документ ObjectScreen в разделе Graphics -> Screens.
- 2. Создадим новый объект TextBox.
- 3. Выделим его.
- 4. В поле Text field укажем тег @Vector3Dimensions[1,2,3].
- 5. В разделе Brushes в окне свойств найдем Background Color.
- 6. Откроем его вкладку Dyn.
- 7. В качестве Default Color (цвет по умолчанию) выберем цвет LightGray.
- 8. В поле Expression введем @Vector3Dimensions[1,2,3].
- 9. В первой свободной строке в столбце Value введем 1.



- 10. В столбце Color этой строки зададим красный цвет (Red).
- 11. В двух следующих строка введем:
  - Value: 2, Color: Blue
  - Value: 3, Color: Green
- 12. Сохраним изменения, сделанные в документе.

O @vec	or3Dimensions[1,2,3	Backgro	und	
	0	Foregro	und Color	
		Solid Grad	Img Dyn	
		Default	Color: LightGray	•
		Expre	ssion @Vector3Dimensions	s[1,2,3]
		Value	Color	Remove
		1		x
				Course of the second se
		2		x
		2		x

При запуске приложения объект изначально отображается в сером цвете (LightGray), поскольку текущее значение тега будет равно нулю. Т.к. значению 0 нами не присвоен никакой определенный цвет, то фон объекта получит цвет, установленный по умолчанию.



Изменяя значение тега в 1, 2 или 3, мы заставим фон нашего текстового поля окрашиваться в цвета, установленные в настройках для этих значений, а именно в красный, голубой и зеленый соответственно.





# 22.3 Настройка окна для отображения трендов

Для настройки окна трендов нам понадобится еще один тег. Этот тег будет эмулировать значение параметра, изменения которого мы хотим наблюдать в виде графика.

### 22.3.1 Создадим новый тег для наблюдения тренда

Выполним следующие шаги:

- 1. Откроем документ Tags.
- 2. Добавим в него новый тег.
- 3. Переименуем его в Simulation.
- 4. Установим его тип в Integer.
- 5. Изменим размерность тега (поле Dimension) в 1 и Column(X) в 3.
- 6. В разделе Selected Members в верхней части окна свойств этого тега выберем из списка первый элемент массива: Simulation[0].
- 7. Установим значение -200000 в поле Min Value и значение 200000 в поле Max Value.
- 8. В разделе Communication в поле Server выберем опцию Simulation.
- 9. В поле Туре выберем опцию Linear.
- 10. В поле Samples/р введем значение 10.
- 11. В поле Period (ms) введем значение 10000.
- 12. Для второго и третьего элементов массива сделаем по аналогии следующие настройки:
  - Simulation [1]: Min Value -250000, Max Value 250000

Communication Server: Simulation, Type: Sine, Samples/s: 10

Period (ms): 10000

- Simulation [2]: Min Value -300000, Max Value 300000
   Communication Server: Simulation, Type: Random Numbers
   Frequency (ms), 10000
- 13. Сохраним изменения в документе.



#### 22.3.2 Создание экранной формы для вывода трендов

Теперь, когда у нас есть теги, предназначенные для наблюдения в виде трендов, пора приступить к настройке экранной форма. Проделаем следующее:

- 1. Откроем экранную форму Trend в разделе Graphics -> Screens.
- 2. Найдем объект Trend в верхней ленте раздела Graphics, в секции Advanced Objects.
- 3. Кликнем правой кнопкой мыши в области точки с координатами (0;0) в верхнем левом углу документа для создания объекта с размерами по умолчанию.
- 4. Откроем свойства вставленного объекта Trend.
- 5. Сделаем следующие настройки:
  - Width: 1014, Height: 645,
  - Left: 0, Top: 0
- 6. В разделе Trend Configuration, во вкладке General введем значение Simulation[0] в поле New Pen и нажмем кнопку +.

General	Configuration	
Pens		

- 7. Добавим аналогичным образом еще два пера для значений Simulation[1] и Simulation[2].
- 8. Дважды кликнем на пере Simulation[0] и откроем его свойства.
- 9. Изменим цвет пера в (Pen Color) в голубой (Blue) и стиль линии (Line Style) в -
- 10. В поле Selected Item выберем перо Simulation[1].
- 11. Изменим его цвет на зеленый и стиль линии на . . .
- 12. В поле Selected Item выберем перо Simulation[2].
- 13. Изменим его цвет на Red (красный).
- 14. Сохраним изменения, сделанные в документе.





После запуска приложения мы сможем наблюдать за изменениями значений тегов в окне трендов примерно такого вида.



# 23. Создание главного экрана проекта

После того, как основные настройки проекта закончены, можно заняться организацией удобного для оператора размещения экранных форм и элементов управления.

Прежде всего, нам понадобится главный экран проекта, на котором мы будем отображать все остальные экраны нашего проекта. Сам главный экран не будет изменяться в процессе исполнения проекта, он лишь отобразит остальные существующие в проекте экраны, а также будет содержать интерфейс для управления переключениями между рабочими экранами, а также для остановки проекта (для выхода из режима исполнения).

### 23.1 Создание главного экрана

Как обычно, с помощью правого клика на разделе Graphics в дереве навигации создадим новый документ. Переименуем его MainScreen.

Сохраним созданный документ.

### 23.2 Настройки главного экрана

Настроим наш новый экран следующим образом с помощью окна свойств:

- Заголовок (Title): Introductory Guide
- Размеры: Width: 1024, Height: 768
- Начальная позиция: (top): 0, (left): 0
- Диалоговое окно (Dialog): False
- Отображать основное меню (Show Header Buttons): False
- Разрешение перемещения (Enable Movement): False
- Горизонтальная прокрутка (Horizontal Scroll): False
- Вертикальная прокрутка (Vertical Scroll): False
- Одиночное окно (Single Window): True
- Отображать в панели задач (Show In Task Bar): True
- Запуск в полноэкранном виде (Start Maximized): False



? Search		
4 Info		
Name	MainScreen	
Туре	Graphics	
Title	Introductory Guide	
Window Location	(0.0)	
Window Size	(1024 768)	
4 Configuration	(1021)/00)	
Comparation	Name Type	
Screen Tags	Generic	v x
Show Header Buttons	False	- P
Show In Task Bar	✓True	
Start Maximized	False	
Enable Movement	False	
Dialog	False	
Single Window	✓True	
Horizontal Scroll	False	
Vertical Scroll	False	
Resize Mode	Can Resize \vee	
Window Location Setting	Center Screen 👻	
Window Style	3D Border Window 🛛 👻	
Brushes		
Background		
Solid Grad Img		
1	R	255
	6	255
	R	255
		235
	A	100

Сохраним изменения в документе.



## 23.3 Добавление и настройка графических объектов

В этом разделе расскажем об особенностях настройки главных экранов проекта в ADISRA SmartView.

### 23.3.1 Добавление и настройка элементов типа GroupBox

В качестве первого шага в настройке главного экрана предлагаем создать групповое поле (GroupBox) для размещения элементов управления нашим приложением. Для этого сделаем следующее:

- 1. Открыв для редактирования нашу экранную форму, перейдем в раздел Graphics верхней ленты.
- 2. Выберем объект GroupBox в поле Interface Objects.
- 3. Кликнем левой кнопкой мыши в области близкой к координатам (0;0) в левом верхнем углу экрана и протянем курсор мыши до нижнего правого угла экрана.
- 4. Отпустим левую клавишу мыши и получим в результате новый объект типа GroupBox.
- 5. Перейдя в окно свойств созданного объекта, найдем текстовое поле свойства Text в разделе Configuration и введем туда текст Guide Major Commands.
- 6. Теперь настроим размер и расположение объекта на экране. В свойстве Тор, введем 0, а в поле Left введем 60.
- 7. После этого установим размеры объекта. В поле Width введем 964, а в поле Height введем 52.
- 8. Сохраним на этом этапе изменения в документе.

### 23.3.2 Добавление и настройка кнопки закрытия приложения (выход)

Вторым элементом, который мы добавим на главный экран, будет кнопка закрытия приложения (выхода из режима исполнения). Для этого выполним следующие шаги:

- 1. Перейдем в раздел Graphics верхней ленты.
- 2. Выберем объект Button из секции Basic Objects.
- 3. Кликнем в правой части созданного ранее GroupBox. Это создаст в нем кнопку с размерами по умолчанию. Изменим их на следующие: Width: 100, Height: 25.
- 4. В свойстве Тор введем значение 20, а в свойстве Left значение 910.
- 5. В свойстве Text введем значение Exit Guide.
- 6. Теперь осталось только настроить событие, вызывающее закрытие приложения. В свойстве Mouse Up щелкнем на текстовом поле. Это откроет новое окно. В нем настроим сценарий, который будет выполняться каждый раз, когда пользователь щелкает и отпускает левую кнопку мыши на объекте.



7. Введем в открывшемся окне следующий код:

if (MessageBox.Show (Do you want to quit the application?, Quit the application, MessageBoxButtons.YesNo, MessageBoxIcon.Question) == DialogResult.Yes)

{
SVApplications.StopApp ();
}

При выполнении этот фрагмента кода на экране появится диалог с вопросом, хотим ли мы выйти из приложения. Если будет выбрано Yes, то приложение закроется, а если будет выбрано No, то приложение продолжит работать.

8. В качестве последнего шага просто сохрани изменения, которые мы в него внесли.

### 23.3.3 Установка логотипа ADISRA SmartView на главном экране

Используем объект типа Image для того, чтобы вывести логотип ADISRA в левый верхний угол главного экрана.

Для этого выполним следующие действия:

- 1. Открыв документ MainScreen, перейдем в раздел Graphics верхней ленты.
- 2. Выберем объект Image из поля Basic Objects.
- 3. Кликнем левой кнопкой мыши в районе точки с координатами (0;0), в верхнем левом углу экрана. В результате будет создан объект типа Image с размерами по умолчанию (80х80 в ширину и высоту). После того, как объект был создан, изменим его расположение и размеры в панели свойств.
  - a) Location *Top*: 12 *Left*: 7
  - b) Size Width: 85 Height: 35
- 4. Теперь сохраним логотип ADISRA с нашего web-сайта. Для этого откроем в браузере адрес <u>www.adisra.com</u>
  - а) Кликнем правой кнопкой мыши на логотипе:



- b) Выберем опцию Save image as
- c) Сохраним изображение под именем ADISRALogo.png в папку нашего проекта, в подпапку с именем Images.





- 5. После этого нажмем кнопку «...» справа от поля Source File свойства Source. Откроется новое окно со всеми изображениями, загруженными в библиотеку проекта. Выберем ранее сохраненное с сайта ADISRA изображение (ADISRALogo.png). Для этого выберем изображение и нажмем ОК или просто дважды кликнем на изображении.
- 6. Сохраним изменения, сделанные в документе.

Выбранное нами изображение отобразится на главном экране приблизительно так:





#### 23.3.4 Добавление вкладок

Объекты типа Вкладка (Tab) являются элементами интерфейса, которые помогают отображать другие графические экраны, упорядочивая их в виде вкладок. Далее мы настроим основные свойства элементов типа Вкладка, чтобы заставить их отображать другие графические экраны нашего проекта.

Для добавления и настройки вкладок (Tab) выполним следующие шаги:

- 1. Открыв документ Main Screen, перейдем в раздел Graphics верхней ленты.
- 2. Выберем элемент типа Tab в секции Interface Objects.
- Кликнем мышью в любом месте экрана. В результате на экране появится вкладка с настойками по умолчанию. Чтобы изменить ее размер и местоположение, отредактируем ее свойства следующим образом.
- 4. В свойстве Тор введем значение 59, а в свойстве Left значение 0.
- 5. В свойстве Width введем значение 1024, а в свойстве Height значение 708.
- 6. После чего просто сохраним изменения, сделанные в документе.

В результате наш главный экран должен выглядеть примерно так:

ADISRA	-Guide Major Commanda	Bot Guide
SampieTab		



### 23.3.5. Настройка экранов, которые должны отображаться во вкладках элемента Таb

В этом разделе настроим наш объект типа Tab таком образом, чтобы в нем отображались нужные нам экранные формы. Сделаем следующее:

- 1. Откроем документ MainScreen в разделе Graphics.
- 2. Выделим объект Tab. Сделаем следующие настройки в его окне свойств.
- Во вкладке Grid в разделе Configuration выделим и переместим в правую колонку элемент Alarms\Alarms. Это можно сделать методом перетаскивания с помощью мыши или просто нажав на кнопку ->.
- 4. Повторим это для следующих элементов:
  - Database\Database
  - Graphics\GraphicArts
  - History\History
  - Recipe\Recipe
  - Report\Report
  - Services\Service
  - Triggers\Triggers
  - Tunneling\Tunneling
  - UserFunctionLibrary\UserFunctionLibrary
  - Drivers\Drivers

Alarms\Alarms	~	Alarms\Alarms				
Drivers\Drivers		Graphics\GraphicArts				
Graphics\GraphicArts		History\History				
History\History		Recipes\Recipe				
Recipes\Recipe		Report\Report				
Report\Report		Services\service				
Screens\Object1		Triggers\Triggers				
Screens\Trend		Tunneling\Tunneling				
Services\service		UserFunctionLibrary\UserFunction				
teste\bugtest		Drivers\Drivers				
Triggers\Triggers						
TunnalinalTunnalina	~	s				

- 5. Переименуем заготовок объекта. Для этого дважды кликнем на Alarms\Alarms в правой колонке. Это даст нам возможность изменить название вкладки Alarms\Alarms.
- 6. В поле заголовка (Header) введем название Alarms.
- 7. Выбирая другие вкладки в выпадающем списке Path во вкладке TabConfig, изменим аналогичным образом названия и для остальных вкладок.



- 8. Сделаем это для всех вкладок.
- 9. Сохраним документ.

В результате наш главный экран будет выглядеть примерно так.





# 24. Запуск на исполнение и отладка проекта

После того, как мы закончили создание проекта, самое время запустить его, отладить, и разобраться в том, как он работает.

 Первым делом нам необходимо указать, какой из наших экранов будет отображаться при запуске. Мы бы хотели, чтобы это был наш Main Screen. Для того чтобы сделать это, перейдем в раздел Settings -> Graphic в верхней ленте. В поле Startup Graphics выберем вариант Main Screen.



 После этого можно запустить проект. Для этого перейдем в раздел RunTime в верхней ленте и нажмем кнопку Run. После того, как проект запустится, индикатор в панели быстрого доступа изменится с красного () на зеленый (), копка Run больше не будет активной, и наоборот, станет доступной для нажатия кнопка Stop.

Примечание: Альтернативным способом запуска проекта на исполнение является нажатие функциональной клавиши F5.

3. Кроме того, можно создать ярлык для запуска проекта. Это настоятельно рекомендуется делать на машинах конечных пользователей, чтобы они могли легко запускать готовый проект без необходимости запуска инженерной среды ADISRA. Для создания ярлыка быстрого запуска перейдем в раздел Settings верхней ленты, выберем раздел General и нажмем кнопку Create a shortcut. Ярлык будет автоматически создан на рабочем столе. Этот ярлык позволит быстро запускать наше приложение.

• • •	STRAW OF COL	10 T														
2.0	tre Balt Ven	Sectings Seave Doenee														
2 10	NUME: +	Startup Graphic ManScream		- Terd	-	Tes		0		2	4	ŵ	1			B
	Ruttime Security Cite Engineering Security: Cite	Share Graphics in the Viewer Mouse events fail through diplects		Points Multipliers 1 *	Resistors	One and Time 1	Derts	Geren	OPC Servery	Number of Decreases	Hever -	Wes	DataBase Corrections	Tags Court	Reduction	Project Bachup +
	twurty .		Querica													
								Sacrap Backap Taport	lack Sepanator:				e Color unite Default Runfin	e image		
								-			_	Seneral			Court	e a Denking Shorts

4. Итак, запустим наш проект. Мы увидим наш главный экран Main Screen и можем начать проверку всего того, что мы создали.



Alarms 6	canhicarta History Ran	inen Report Service	Tringers Turn	alion I that functional	ibrary Drivary			
Saura 10	Alarm 1	Tag Name	Priority Type	Message	Start Time	Return Time	Ack Time	10
	0.0	1						
	Alarm 2:							
	0.0	]						
	Alarm 3:							
	0.0	]						
	Alarm 4:							
	0.0	]						
	Alarm 5:							
	0.0	]						

5. Начнем с проверки Тревог. Кликнем на вкладке Alarms.

- Изменяйте значения тегов и наблюдайте за изменениями в окне просмотра тревог (Alarm Objects). Помните о предельных значениях, которые были установлены в разделе Alarms. (Предел LoLo для Vector1Dimension [0] равен -1000). Итак, если мы установим первое текстовое поле (Alarm 1) на -2000, новая тревога появится как в объекте OnLine, так и в объекте History.
- b. На следующем изображении видно, что должно произойти после изменения всех тегов. Тревога Alarm 4 была подтверждена двойным щелчком по тревоге в онлайн-объекте, а затем тревога Alarm 4 была устранена путем установки тега обратно на 0. Информация об этих событиях сохранится в журнале тревог, в то время как в окне просмотра текущих тревог (онлайн-тревоги) эта тревога будет удалена, поскольку все необходимые действия для устранения тревоги были выполнены.



AĽ	DISRA															E	eit Guid
15	Graphi	cArts	History	Recipes	Report	Services	Triggers	Tunnelin	ig Use	rFunctionLibrar	/ Drivers						
			2														
	~	iarm i	<u>.</u>		Tag Na	me	Priority	Туре	Messa	ge	Start Time		Return	Time	Ack Tit	me	
	3	2220.0			Vector	Dimension]	q -5	Lolo			12/11/2020	13:55:36	12/11/	2020 13:55:5	4		-
					Vector	Dimension[]	3] -4	LoLo			12/11/2020	13:55:33	12/11/	2020 13:55:5	0 12/11/	/2020 13:554	
	A	larm 2	5		Vector	Dimension[	3] -4	LoLo			12/11/2020	13:55:33			12/11/	2020 13:554	2
	10	0.000		-	Vector	Dimension	1 -2	Lolo	LoLo	Tag Vector1Dim	ensii 12/11/2020	13:55:19			12/11	/2020 13:554	1
	1	00000			Vector	Dimension	1 -2	LOLO			12/11/2020	14:00:36					
					Vector	Dimension	1 .4	LoLo			12/11/2020	13:35:33					
	A	Jarm 3	c .		Vector	Dimension	1 -2	Lolo	tale.	Tan Mactor (Dim	anel/12/11/2020	13:33:26					
	2	501.0			Varter	Dimension	1 -1	Iclo	Inte	Tag Vector/Dir	ensi(12/11/2020	113.55.15					
	1				Vector	Dimension	11 -1	Lalo	Lala	Tag Vector1Dim	ensic12/11/2020	13-14-08	12/11/	2020 13:14-1	1		1000
		larm 4			Vector	Dimension	11 -1	Lalo	toto	Tag Vector1Dim	ensir12/11/2020	13-14-08	1910				
	~	ann 4	h		Vector	Dimension	01 -1	Lolo	Lolo	Tag Vector1Dim	ensit 12/11/2020	13:11:09	12/11/	2020 13:11-2	2		
	0	0.0			Vector	DimensionII	1 -1	Loto	Lolo	Tag Vector 1Dim	ensic12/11/2020	13:11:09	Constants	12610-000-00	10		
	110				1	-					Televille Televille					1078	-
	A	larm 5	è		1.0												
	17		2		and the second second				والمراجع والمراجع		1000					the state of the s	-
	0	щ			Tag Na	me	Group	Туре	Priority	Message	Star	t Time		Return Time		Ack Time	
					Vector	Dimension	[] Alarm_	lolo	-5		12/	11/2020 1	3:55:36	12/11/2020	13:55:54		
					Vector	Dimension]	[] Alarm	ILOLO	-3		12/	11/2020 1	3:55.28				243
					Vector	Dimension	Alarm_	Lots	-2	LoLo Tag Vec	tor1Demental 12/	11/2020 1.	3:55:19			12/11/2020	130
					¢				_				_				>
		_	/	4								/					

