



# **CODESYS V3.5**

**Библиотека OwenVisuTools**



**Руководство пользователя**

05.09.2023

версия 3.0

## Оглавление

<b>1</b>	<b>Цель документа.....</b>	<b>3</b>
<b>2</b>	<b>Установка библиотеки OwenVisuTools .....</b>	<b>4</b>
2.1	Установка библиотеки .....	4
2.2	Добавление библиотеки в проект CODESYS.....	5
<b>3</b>	<b>Основная информация .....</b>	<b>6</b>
3.1	Структура библиотеки и принцип использования.....	6
3.2	Общие типы данных и POU .....	7
3.2.1	Перечисление ERRORS.....	7
<b>4</b>	<b>Описание элемента OwenTrend.....</b>	<b>8</b>
4.1	Обзор возможностей элемента .....	8
4.2	ФБ OwenTrend.....	10
4.2.1	Запуск и остановка блока .....	13
4.2.2	Файлы, создаваемые блоком .....	15
4.2.3	Описание форматов файлов .....	18
4.3	Структура TREND_FILE_SETTINGS.....	19
4.4	Структура TREND_FILE_PROPERTIES.....	20
4.5	Структура TREND_VISU_SETTINGS.....	21
4.5.1	Структура VisuStructXYChart .....	22
4.5.2	Структура VisuStructXYAxis .....	26
4.5.3	Структура VisuStructXYCurve.....	28
4.6	Фрейм OwenTrendFrame.....	30
4.7	Кастомизация внешнего вида элемента .....	32
4.7.1	Использование диалогов OwenTrendMainSettingsDialog и OwenTrendAxisAndCurveSettingsDialog.....	32
4.7.2	Изменение значений полей структуры TREND_VISU_SETTINGS из кода программы 34	
4.7.3	Редактирование визуализаций библиотеки .....	35

# 1 Цель документа

Настоящее руководство содержит описание библиотеки **OwenVisuTools**. Эта библиотека включает в себя комплексные графические элементы, расширяющие возможности визуализации CODESYS. В данном документе описана версия библиотеки **3.5.17.21**.

Данная версия библиотеки содержит следующие графические элементы:

- [Тренд \(OwenTrend\)](#).



## ПРИМЕЧАНИЕ

Для использования библиотеки требуется версия **CODESYS V3.5 SP17 Patch 3** или выше.



## ПРИМЕЧАНИЕ

Работа библиотеки поддерживается только на контроллерах OWEN и виртуальном контроллере **CODESYS Control Win V3**.



## ПРИМЕЧАНИЕ

Для использования библиотеки **OwenVisuTools** требуется установка библиотеки **OwenStringUtils** версии **3.5.4.9**.

## 2 Установка библиотеки OwenVisuTools

### 2.1 Установка библиотеки

Библиотека **OwenVisuTools** доступна для загрузки на сайте компании [OBEH](#) в разделе **CODESYS V3/Библиотеки**.

Для установки библиотеки в **CODESYS** в меню **Инструменты** следует выбрать пункт **Репозиторий библиотек**, нажать кнопку **Установить**, выбрать фильтр **Файлы библиотек (.library)** или **Все файлы** и указать путь к файлу библиотеки:

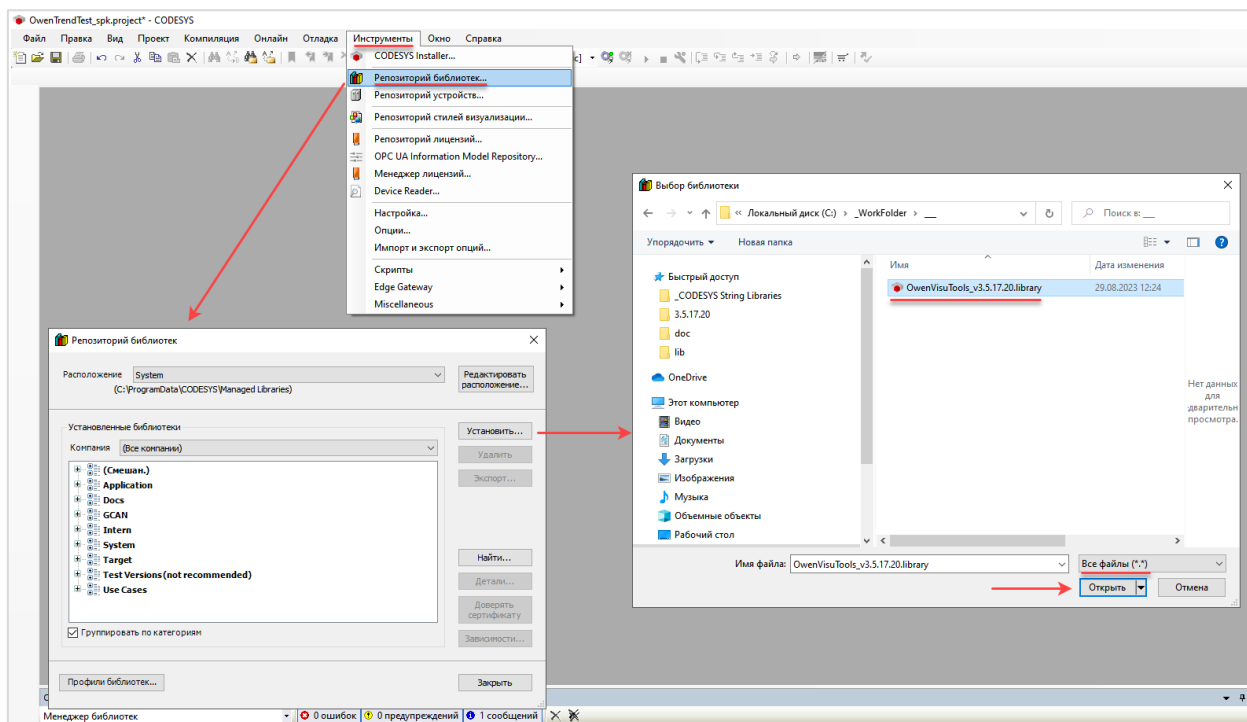


Рисунок 2.1 – Установка библиотеки OwenVisuTools в среду CODESYS

## 2.2 Добавление библиотеки в проект CODESYS

Для добавления библиотеки **OwenVisuTools** в проект **CODESYS** в **Менеджере библиотек** следует нажать кнопку **Добавить библиотеку**, в появившемся списке выбрать библиотеку **OwenVisuTools** и нажать **ОК**.

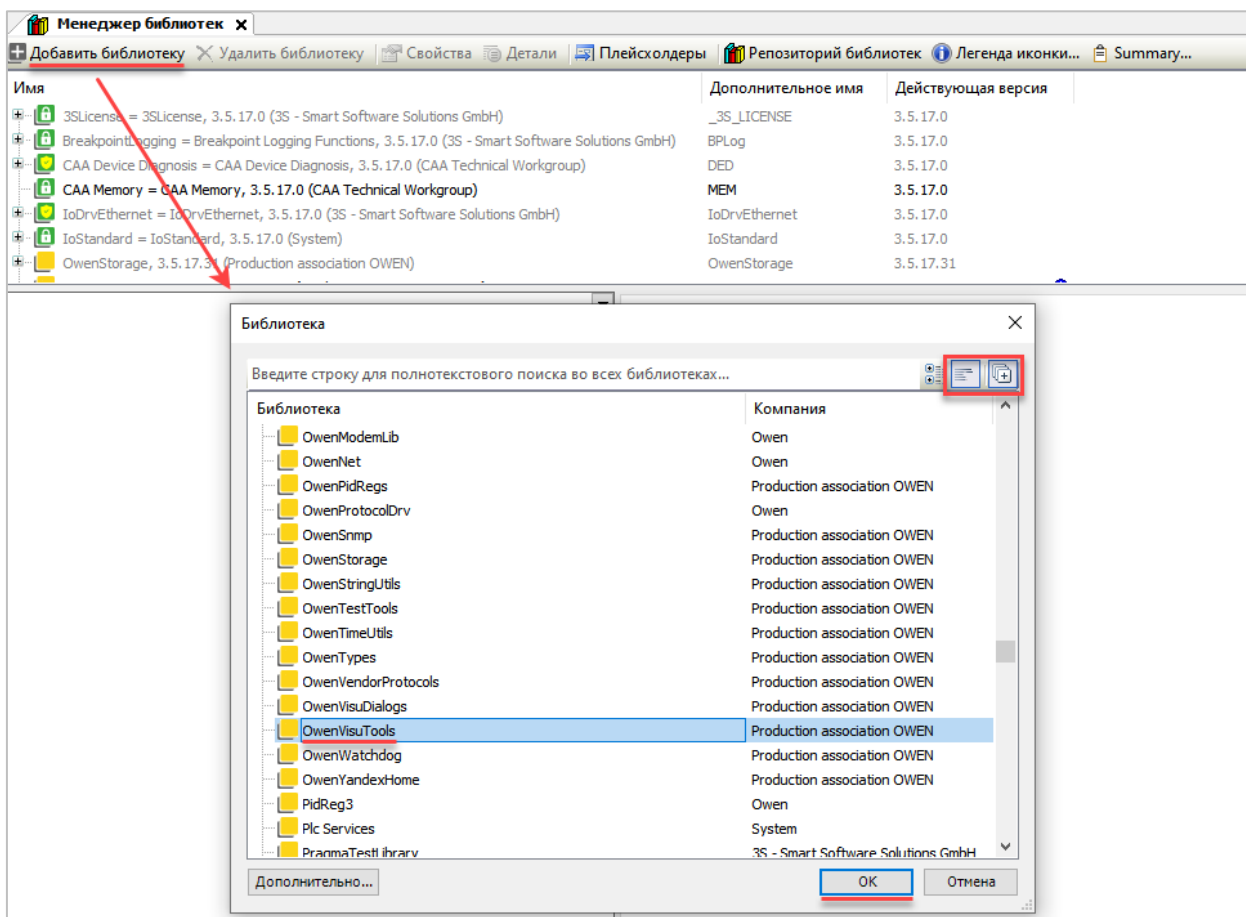


Рисунок 2.2 – Добавление библиотеки OwenVisuTools

После добавления библиотека появится в списке **Менеджера библиотек**:

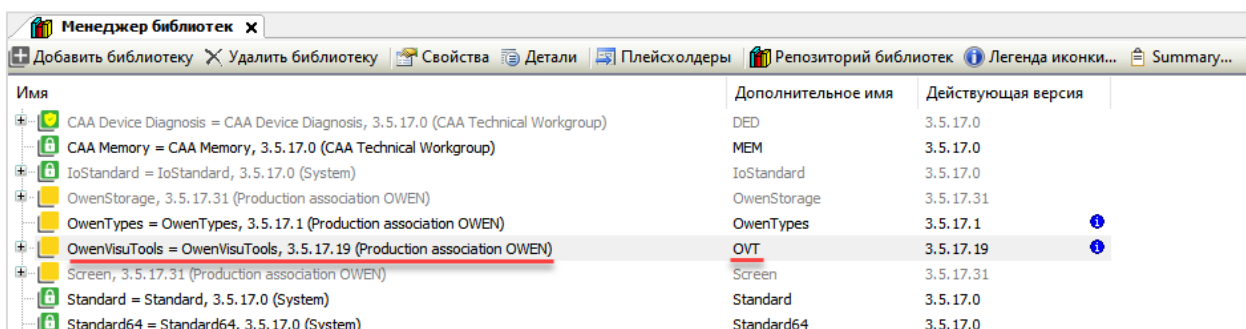


Рисунок 2.3 – Список библиотек проекта



### ПРИМЕЧАНИЕ

При обращении к типам данных и POU библиотеки следует перед их названием указывать префикс **OVT** (пример: **OVT.OwenTrend**).

### 3 Основная информация

#### 3.1 Структура библиотеки и принцип использования

Каждый графический элемент в библиотеке является сочетанием двух объектов:

- фреймом, представляющим собой визуальную часть элемента;
- функциональным блоком (ФБ), реализующим логику работы элемента.

Для использования элемента пользователю требуется:

- объявить в проекте экземпляр функционального блока элемента и **организовать его циклический вызов** с передачей на входы нужных значений;
- разместить на экране визуализации стандартный графический элемент **Фрейм** или **Набор вкладок**, выбрать в нем экран фрейма элемента из библиотеки **OwenVisuTools** и на вкладке **Ссылки** привязать экземпляр функционального блока элемента.

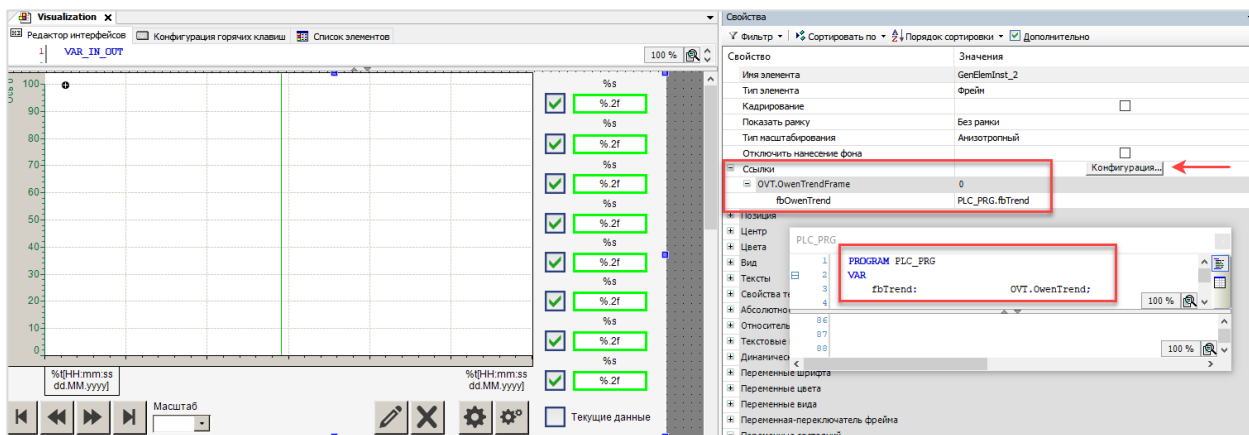


Рисунок 3.1.1 – Привязка экземпляра ФБ к фрейму

Внешний вид элемента может быть изменен пользователем для соответствия требованиям к визуализации его проекта. Информация о доступных способах изменения внешнего вида элемента приведена в последнем пункте раздела его описания.

## 3.2 Общие типы данных и POU

### 3.2.1 Перечисление ERRORS

Перечисление **ERRORS** описывает ошибки, которые могут возникнуть во время вызова ФБ и функций библиотеки.

Таблица 3.2.1 – Описание элементов перечисления ERRORS

Название	Значение	Описание
NO_ERROR	0	Нет ошибок
NOT_OWEN_DEVICE	1	Данное устройство не является контроллером OWEN или виртуальным контроллером <b>CODESYS Control Win V3</b> – и поэтому не поддерживает использование библиотеки

## 4 Описание элемента OwenTrend

### 4.1 Обзор возможностей элемента

В списке стандартных (входящих в дистрибутив среды) графических элементов CODESYS присутствует элемент **Тренд**, используемый для отображения значения переменных в виде графика с возможностью просмотра истории:

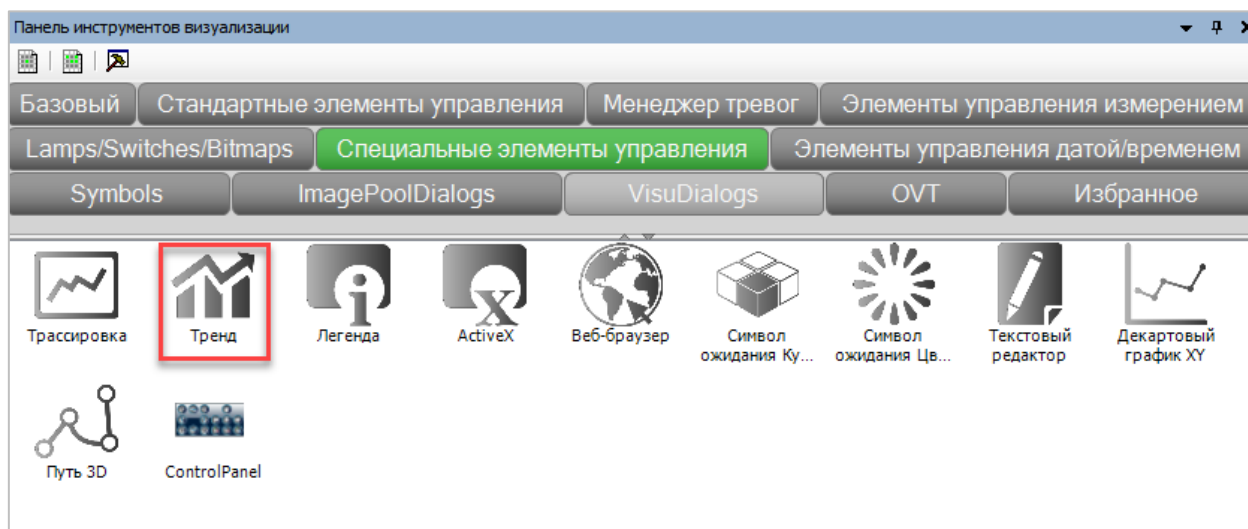


Рисунок 4.1.1 – Стандартный элемент Тренд в панели инструментов визуализации

Для сохранения исторических данных этот элемент использует встраиваемую СУБД [SQLite](#). Если в процессе записи данных в этот файл питание контроллера будет отключено, то файл может быть поврежден. Валидация и восстановление файлов SQLite в текущих версиях CODESYS не реализована, поэтому после загрузки контроллера в тренде может появиться следующее сообщение:

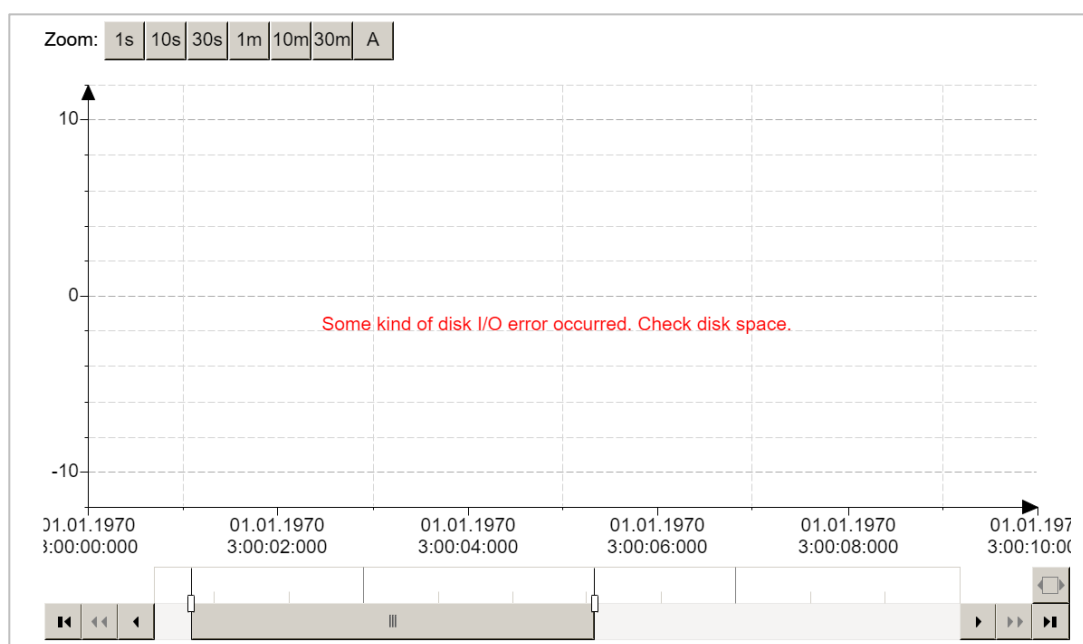


Рисунок 4.1.2 – Отображение сообщения об ошибке при повреждении файла тренда



Текст ошибки может быть другим – например, «**Error 30405 occurred**» (вместо **30405** может отображаться другой код ошибки). В некоторых ситуациях при повреждении файла тренда при переходе на экран визуализации, на котором он размещен, возникает исключение в работе приложения контроллера.

Кроме того, сохранение исторических данных в формате SQLite затрудняет их последующий анализ на ПК, потому что для работы с ним не подходит обычное офисное ПО (например, Microsoft Excel).

Это стало основными причинами, по которой компания ОВЕН разработала собственный графический элемент с названием **OwenTrend**. Ниже перечислен список его возможностей:

- работа в режиме трассировки (отображение текущих значений в режиме реального времени без сохранения истории) и тренда (с сохранением истории и возможностью ее просмотра в элементе);
- отображение до 8 перьев;
- поддержка до 4 осей Y с различным масштабом;
- отображение курсора для определения значений перьев тренда в конкретной временной точке;
- поддержка изменения масштаба оси X;
- сохранение истории в файле бинарного формата (с расширением **.bin**; этот файл используется самим элементом) и файле формата **.csv** (для обработки накопленных данных на ПК);
- файл бинарного формата валидируется после перезагрузки контроллера – в случае его повреждения некорректные записи будут удалены, и тренд продолжит работу;
- пути к файлам тренда определяются пользователем – можно сохранять их на USB- или SD-накопителе, в директории FTP-сервера контроллера или по произвольному пути в файловой системе контроллера;
- возможность управления видимостью перьев;
- возможность приостановки работы элемента;
- возможность очистки истории элемента;
- широкие возможности по настройке внешнего вида элемента (см. подробнее в [п. 4.7](#)).

Элемент включает в себя:

- функциональный блок [OwenTrend](#), реализующий логику работы элемента и используемый для его настройки в программе пользователя;
- фрейм [OwenTrendFrame](#), реализующий графическую часть элемента;
- диалоги [OwenTrendMainSettingsDialog](#) и [OwenTrendAxisAndCurveSettingsDialog](#), используемые для настройки визуальной части элемента в процессе работы приложения контроллера через визуализацию CODESYS.

Визуальная часть элемента основана на стандартном элементе CODESYS [Декартовый график XY](#).



#### ПРИМЕЧАНИЕ

В проекте может быть использовано несколько экземпляров элемента, но одновременно должно вызываться не более **20** (при условии, что пользователь в дополнение к этому не вызывает экземпляры блоков библиотеки в своем проекте или не использует компоненты, в которых организован вызов этих блоков – например, **OwenArchiver**). Это связано с [ограничением библиотеки CAA File](#), использованной для реализации работы с файлами элемента.

## 4.2 ФБ OwenTrend

Функциональный блок **OwenTrend** реализует логику работы соответствующего графического элемента и позволяет произвести его настройку.



### ПРИМЕЧАНИЕ

Если вызов экземпляра ФБ производится в программе, привязанной к задаче **VISU\_TASK**, то эта программа должна быть добавлена в списке **после** системной программы **VisuElems.Visu\_Prg**.

Вход **xEnable** используется для запуска и остановки блока. Подробнее механизм запуска и остановки описан в [п. 4.2.1](#).

Вход **xTraceMode** определяет режим работы блока: если он имеет значение **TRUE**, то блок работает в режиме трассировки, а если **FALSE** – то в режиме тренда. В режиме трассировки в элементе отображаются только текущие значения переменных; сохранения исторических данных в .bin-файл и .csv-файл не происходит. В режиме тренда значения перьев сохраняются в .bin и .csv-файл, а в фрейме [OwenTrendFrame](#) отображаются кнопки для просмотра исторических данных.

К входу **dtCurrentDateAndTime** следует привязать переменную, содержащую текущее значение системного времени контроллера (рекомендуется использовать переменную, привязанную к каналу **Date And Time** узла **OwenRTC** таргет-файла контроллера OBEH). В случае запуска проекта на виртуальном контроллере – можно использовать для получения системного времени функцию **SysTimeRtcGet** из библиотеки **SysTimeRtc** или другую аналогичную функцию.

На вход **alrCurrentTrendValues** передается массив из 8 переменных типа **LREAL** с текущими значениями перьев элемента. Число перьев определяется значением входа **usiValuesCount**; максимальное число перьев – 8. Значения неиспользуемых элементов массива следует оставить нулевыми. Если нужно отображать в элементе переменные других типов (например, **UINT**, **INT**, **REAL** и т. д. – то их нужно преобразовать к типу **LREAL** с помощью оператора **TO\_LREAL**.

На вход **pstTrendVisuSettings** передается указатель на экземпляр структуры [TREND\\_VISU\\_SETTINGS](#), значения полей которой определяют внешний вид элемента. Если требуется изменять внешний вид элемента в процессе работы приложения через диалоги настройки (см. [п. 4.7.1](#)), то экземпляр этой структуры должен быть объявлен в области энергонезависимых переменных (**VAR RETAIN**), чтобы изменения сохранялись после перезагрузки контроллера. **Обратите внимание**, что размер структуры – **23424 байта**, а общий размер энергонезависимой памяти контроллеров OBEH – **64 Кб**.

На вход **pstTrendFileSettings** передается указатель на экземпляр структуры [TREND\\_FILE\\_SETTINGS](#). С помощью полей структуры задаются пути к .bin и .csv-файлам, устанавливается периодичность сохранения значений перьев в эти файлы и максимальное число записей для них. Более подробная информация о механизме работы с файлами приведена в [п. 4.2.2](#).

На вход **pstTrendFileProperties** передается указатель на экземпляр структуры [TREND\\_FILE\\_PROPERTIES](#). С помощью полей структуры разработчик может узнать максимальные размеры .bin и .csv-файла в байтах, а также номера последних записей в этих файлах. Поля этой структуры доступны только для чтения.

Для обеспечения возможности дозаписи данных в .csv-файл после перезагрузки контроллера требуется объявить экземпляр структуры в области энергонезависимых переменных (**VAR RETAIN**);

в противном случае после перезагрузки контроллера .csv-файл начнет перезаписываться со своей первой записи. .bin-файл продолжает дозаписываться после перезагрузки контроллера независимо от принадлежности экземпляра структуры к области энергонезависимых переменных.

По переднему фронту входа **xClearBinFile** выполняется удаление бинарного (.bin) файла тренда и очистка данных элемента (в т. ч. тех, которые отображались на тренде в момент очистки), после чего блок автоматически создаст новый файл и заново начнет записывать в него исторические данные. Вход обрабатывается только в том случае, если вход **xEnable** имеет значение **TRUE**.

По переднему фронту входа **xClearCsvFile** выполняется удаление .csv-файла тренда, после чего блок автоматически создаст новый файл и заново начнет записывать в него исторические данные. Вход обрабатывается только в том случае, если вход **xEnable** имеет значение **TRUE**.

Пока блок находится в работе – выход **xActive** имеет значение **TRUE**. В случае возникновения ошибки выход **xError** принимает значение **TRUE**, а на выходе **eError** отображается код ошибки из перечисления [ERRORS](#).

Визуальная часть элемента описана в [п. 4.6](#).

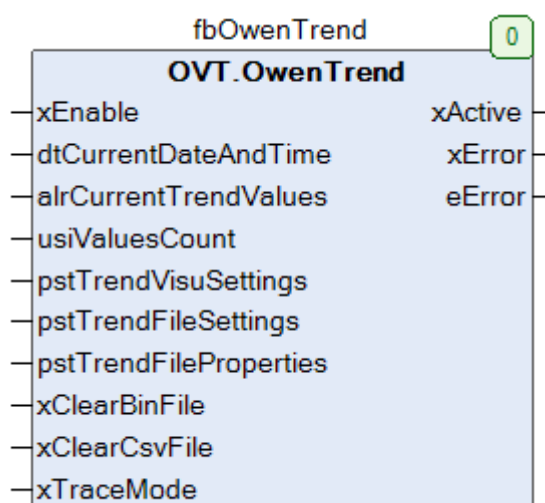


Рисунок 4.2.1 – Внешний вид ФБ OwenTrend на языке CFC

Таблица 4.2.1 – Описание входов и выходов ФБ OwenTrend

Название	Тип	Описание
<b>Входы</b>		
xEnable	BOOL	По переднему фронту выполняется запуск ФБ. Блок работает, пока этот вход сохраняет значение <b>TRUE</b> . По заднему фронту выполняется остановка ФБ. См. более подробное описание в <a href="#">п. 4.2.1</a>
dtCurrentDateAndTime	DT	Текущее значение системного времени контроллера, используемое блоком

alrCurrentTrendValues	ARRAY [1..8] OF LREAL	Текущие значения перьев тренда. Для неиспользуемых перьев можно оставить нулевые значения
usiValuesCount	USINT(1..8)	Число используемых перьев тренда
pstTrendVisuSettings	POINTER TO <a href="#">TREND_VISU_SETTINGS</a>	Указатель на экземпляр структуры, определяющей внешний вид элемента
pstTrendFileSettings	POINTER TO <a href="#">TREND_FILE_SETTINGS</a>	Указатель на экземпляр структуры, определяющей настройки файлов тренда (пути к файлам, максимальное число записей в файлы и т. д.)
pstTrendFileProperties	POINTER TO <a href="#">TREND_FILE_PROPERTIES</a>	Указатель на экземпляр структуры, с помощью полей которой разработчик может узнать максимальные размеры .bin и .csv-файла в байтах, а также номера последних записей в этих файлах. Поля этой структуры доступны только для чтения
xClearBinFile	BOOL	По переднему фронту выполняется удаление бинарного файла тренда и очистка его данных (в т. ч. тех, которые отображались на тренде в момент очистки), после чего блок автоматически создаст новый файл и заново начнет записывать в него значения. Вход обрабатывается только в том случае, если вход <b>xEnable</b> имеет значение <b>TRUE</b>
xClearCsvFile	BOOL	По переднему фронту входа выполняется удаление .csv-файла тренда, после чего блок автоматически создаст новый файл и заново начнет записывать в него значения. Вход обрабатывается только в том случае, если вход <b>xEnable</b> имеет значение <b>TRUE</b>
xTraceMode	BOOL	Режим работы элемента. <b>FALSE</b> – режим тренда; <b>TRUE</b> – режим трассировки (отображение данных реального времени без сохранения истории). См. более подробное описание в начале пункта
<b>Выходы</b>		
xActive	BOOL	<b>TRUE</b> – ФБ находится в работе
xError	BOOL	Принимает значение <b>TRUE</b> в случае возникновения ошибки
eError	<a href="#">ERRORS</a>	Статус работы ФБ (или код ошибки)

#### 4.2.1 Запуск и остановка блока

Управление работой блока осуществляется с помощью входа **xEnable**. Блок находится в работе до тех пор, пока переменная, привязанная ко входу **xEnable**, имеет значение **TRUE**. По заднему фронту входа **xEnable** производится остановка блока.

В момент остановки выходы блока принимают следующие значения:

- **xActive** = **FALSE**;
- **xError** = **FALSE**;
- **eError** = **ERRORS.NO\_ERROR**.

В период остановки блок не выполняет никаких операций. Если блок работает в режиме тренда (то есть его вход **xTraceMode** имеет значение **FALSE**), то все используемые им файлы являются закрытыми – поэтому, например, можно произвести их копирование, не опасаясь за консистентность данных. Кнопки фрейма [OwenTrendFrame](#) являются неактивными.

Повторный запуск блока после остановки (в том числе, после перезагрузки контроллера) сопровождается следующими действиями:

- если блок работает в режиме трассировки (то есть вход **xTraceMode** = **TRUE**), то после повторного запуска график полностью очищается и начинает отрисовываться заново;
- если блок работает в режиме тренда (то есть вход **xTraceMode** = **FALSE**), то после повторного запуска значения перьев за интервал времени остановки отображаются равными значению поля **IrStopValue** из структуры [TREND\\_VISU\\_SETTINGS](#). Это значение должно быть выбрано специально таким, чтобы переменные перьев не могли принять его в процессе нормальной работы системы – например, если вы знаете, что температура вашего объекта управления не может быть больше 200 °С, то установите для **IrStopValue**, например, значение 300. В .bin-файл тренда добавляются записи со значением **IrStopValue**. Затрачиваемое на добавление этих записей время зависит от периода остановки блока и значения полей **tSamplePeriod** и **udiSaveBinPeriod** экземпляра структуры [TREND\\_FILE\\_SETTINGS](#). Максимальное число записей со значением **IrStopValue**, которое может быть добавлено при повторном запуске блока после остановки – **3000**. Если блок был остановлен на интервал времени, за который согласно настройкам должно было быть добавлено большее число записей – то первая запись после повторного запуска будет добавлена «вплотную» к предыдущим данным, что приведет к нарушению масштаба графика по оси времени (т. е. при прокрутке графика в режиме истории на разных страницах тренда может отображаться разный диапазон времени, зависящий от времени, на которое производилась остановка элемента). В .csv-файле период остановки блока никак специально не отображается – но его можно определить по разнице в метках времени соседних записей, которая будет превышать произведение полей **tSamplePeriod** и **udiCsvBinPeriod** экземпляра структуры [TREND\\_FILE\\_SETTINGS](#).

Значения входов **usiValuesCount**, **pstTrendFileSettings** и **pstTrendVisuSettings** (включая значения элементов структур) считываются блоком только в момент запуска (по переднему фронту входа **xEnable**). Их изменение в процессе работы блока не приведет ни к каким последствиям – для применения новых значений потребуется перезапустить блок.

В процессе запуска блока выполняется:

- валидация размера .bin-файла. Если размер .bin-файла не кратен размеру одной записи, то последняя запись файла восстанавливается на основании метки времени предыдущей записи и значения **IrStopValue**;
- проверка конфигурации .bin-файла и .csv-файла.

Проверка конфигурации .bin-файла осуществляется следующим образом:

- если задаваемое при загрузке проекта значение поля **tSamplePeriod** отличается от значения, ранее использованного при работе с этим файлом – то файл пересоздается и сохранение исторических данных начинается заново;
- если задаваемое при загрузке проекта ограничение на максимальное число записей в файле ([TREND\\_FILE\\_SETTINGS.udlMaxBinRecordCount](#)) меньше текущего числа записей в файле – то файл пересоздается и сохранение исторических данных начинается заново.

Проверка конфигурации .csv-файла осуществляется следующим образом:

- если задаваемое при загрузке проекта входа **usiValuesCount** отличается от значения этого же входа, ранее использованного при работе с этим файлом – то файл пересоздается и сохранение исторических данных начинается заново;
- если задаваемое при загрузке проекта ограничение на максимальное число записей в файле ([TREND\\_FILE\\_SETTINGS.udlMaxCsvRecordCount](#)) меньше текущего числа записей в файле – то файл пересоздается и сохранение исторических данных начинается заново.

При запуске и остановке блока в лог контроллера (вкладка **Device – Журнал**) добавляется соответствующее сообщение от компонента **OwenTrend**. В тексте сообщения указано имя экземпляра функционального блока, который был запущен или остановлен.

Жёсткость	Временная отметка	Описание	Компонент
	23.08.2023 09:45:01	PLC_PRG.ftbTrend: Trend is stopped	OwenTrend
	23.08.2023 09:43:56	Visu_PRG: Creating Client successful for Extern-ID: 86729 Returned IEC-ID: 0	IECVisualization
	23.08.2023 09:43:56	Visu_PRG: Creating Client for Extern-ID: 86729	IECVisualization
	23.08.2023 09:43:56	Webserver started	CmpWebServer
	23.08.2023 09:43:56	HTTPS is not working with your configuration. See previous log entries for details!	CmpWebServer
	23.08.2023 09:43:56	The needed certificate is not available for HTTPS.	CmpWebServer
	23.08.2023 09:43:56	TlsCreateContext: Creating the TLS context for namespace 'WebServer' failed.	CmpOpenSSL
	23.08.2023 09:43:56	TlsCreateContext: Failed to read certificate './PKI/cert/wwwserver.cer' in namespace: 'WebServer'.	CmpOpenSSL
	23.08.2023 09:43:56	Visuinitialization done.	IECVisualization
	23.08.2023 09:43:56	Visuinitialization starting.	IECVisualization
	23.08.2023 09:43:56	Application start	IoDrvSPK100M01
	23.08.2023 09:43:56	PLC_PRG.ftbTrend: Trend is started	OwenTrend
	23.08.2023 09:43:03	Application [Application] loaded via [Download]	CmpApp
	23.08.2023 09:43:02	SoftName: watchdog v1.0.0.0	IoDrvWatchdog
	23.08.2023 09:43:02	DevName: WATCHDOG	IoDrvWatchdog
	23.08.2023 09:43:02	DrvName: IoDrvWatchdog v1.0.0.0	IoDrvWatchdog
	23.08.2023 09:43:02	VenName: OWEN	IoDrvWatchdog

Рисунок 4.2.2 – Сообщения о запуске и остановке блока в логге контроллера

#### 4.2.2 Файлы, создаваемые блоком

В режиме тренда (когда вход **xTraceMode** имеет значение **FALSE**) экземпляр блока сохраняет данные перьев в файлы бинарного (.bin) и текстового (.csv) форматов; при этом создается только один файл каждого формата. Бинарный файл используется самим элементом для отображения исторических данных. Файл формата .csv предназначен для конечного пользователя – он может быть выгружен из контроллера (например, через web-визуализацию CODESYS, по протоколу FTP или с помощью извлечения USB/SD-накопителя – в случае, если файл сохраняется в память накопителя) и использован для анализа работы системы автоматизации.

Настройки сохранения файлов задаются в экземпляре структуры [TREND FILE SETTINGS](#), указатель на который передает на вход **pstTrendFileSettings**. Структура определяет:

- пути к сохраняемым файлам (поля **sPathToBinFile** и **sPathToCsvFile**);
- периодичность добавления новых точек на график (поле **tSamplePeriod**);
- коэффициенты для расчета периода сохранения файлов (поля **udiSaveBinPeriod** и **udiSaveCsvPeriod**);
- максимальное число записей в файлах (поля **udiMaxBinRecordCount** и **udiMaxCsvRecordCount**).



#### ПРИМЕЧАНИЕ

Рекомендуется сохранять файлы элемента на USB- или SD-накопитель, чтобы предотвратить деградацию и следующий за ней выход из строя встроенной flash-памяти контроллера.

Период сохранения данных в файлы определяется как произведение значения входа блока **tSamplePeriod** и коэффициентов **udiSaveBinPeriod** и **udiSaveCsvPeriod**. Значение коэффициентов должно принадлежать диапазону **1...300**. Если заданное значение выходит за границы диапазона, то вместе него блок будет использовать значение соответствующей границы диапазона.

Рассмотрим следующую ситуацию:

- **tSamplePeriod** = T#10s;
- **udiSaveBinPeriod** = 5;
- **udiSaveCsvPeriod** = 20.

При таких настройках:

- на тренде каждые 10 секунд будут отрисовываться текущие значения перьев;
- каждые 50 секунд ( $T\#10s \cdot 5$ ) в .bin-файл будут записываться данные за прошедшие 50 секунд (то есть последние 5 записей);
- каждые 200 секунд ( $T\#10s \cdot 20$ ) в .csv-файл будут записываться данные за прошедшие 200 секунд (то есть последние 20 записей);

После того, как число записей в файлах достигнет максимального значения (**udiMaxBinRecordCount** и **udiMaxCsvRecordCount** соответственно) – файлы начнут перезаписываться начиная с самых ранних записей в режиме циклического буфера.

При установке значений максимального числа записей следует учитывать, что максимальный размер файла не должен превышать 2 Гб – это связано с [ограничениями системных библиотек CODESYS](#). Максимально возможный размер файла в байтах для заданного числа записей можно определить с помощью полей **udiFbMaxBinFileSize** и **udiFbMaxCsvFileSize** экземпляра структуры [TREND VISU PROPERTIES](#) – их значения рассчитываются самим блоком при его запуске.



**ПРИМЕЧАНИЕ**

Максимально возможный размер файла не должен превышать объем свободной flash-памяти контроллера (если файл сохраняется в памяти контроллера) или накопителя (если файл сохраняется на накопителе).

Пользователь определяет пути, по которым будут сохраняться файлы тренда, с помощью полей **sPathToBinFile** и **sPathToCsvFile** экземпляра структуры [TREND\\_FILE\\_SETTINGS](#). Файлы могут сохраняться во flash-памяти контроллера или на подключенных к нему накопителях. Рекомендуется сохранять файлы на накопителях, чтобы избежать деградации встроенной flash-памяти контроллера.

Путь может быть задан в явном виде, например: '/home/root/CODESYS\_WRK/MyTrend.bin'.

Но во многих ситуациях удобнее использовать в путях **заместители**, которые автоматически будут интерпретированы как пути к определенным директориям контроллера или подключенных к нему накопителей. Список доступных заместителей приведен в таблице ниже:

**Таблица 4.2.2.1 – Список заместителей, доступных для контроллеров ОВЕН**

Директория	СПК1xx [M01]	ПЛК2xx	Заместитель
Рабочая директория	/home/root/CODESYS_WRK/PlcLogic	/root/CODESYS/PlcLogic	\$\$PlcLogic\$\$
USB-накопитель	/mnt/ufs/media/sda1		\$\$USB\$\$
SD-накопитель	/mnt/ufs/media/mmcblk0p1		\$\$SD\$\$
Директория FTP-сервера	/mnt/ufs/home/ftp/in		\$\$FTP\$\$
Директория файлов визуализации и web-сервера визуализации	рабочая директория/visu		\$\$visu\$\$

Например, при указании пути '\$\$USB\$\$/MyTrend.bin' – файл будет сохранен в корне USB-накопителя, подключенного к контроллеру.

Если вы запускаете блок на виртуальном контроллере **CODESYS Control Win V3**, то пути к файлам должны задаваться в принятом для ОС Windows виде, например 'D:\MyTrend.bin'.

Если в пути присутствуют несуществующие директории – они будут созданы автоматически при создании файла.

Пути к файлам не должно содержать символы кириллицы.

При сохранении файлов тренда на USB- или SD-накопителе крайне важно придерживаться следующих правил:

- перед запуском блока убедитесь, что подключенный накопитель смонтирован. Это можно сделать с помощью канала **USB mounted** (в случае использования USB-накопителя) или **MMC mounted** (в случае использования SD-накопителя) узла **Drives** таргет-файла контроллера ОВЕН;
- перед извлечением накопителя следует остановить работу блока (см [п. 4.2.1](#)) и размонтировать накопитель с помощью канала **USB unmount** или **MMC unmount**;
- перед запуском блока убедитесь, что размер свободного места на накопителе (см. каналы **USB Free** или **MMC Free**) больше максимального размера файла, который может быть создан блоком (см. значений полей **udiFbMaxBinFileSize** и **udiFbMaxCsvFileSize** экземпляра структуры [TREND\\_FILE\\_PROPERTIES](#)).



Нарушение этих правил может привести к потере исторических данных и нарушению нормальной работы приложения CODESYS.

Более подробная информация об узле **Drives** приведена в документе **CODESYS V3.5. Описание таргет-файлов**. Для работы с узлом в визуализации CODESYS может использоваться диалог **DrivesOwen** из библиотеки [OwenVisuDialogs](#).

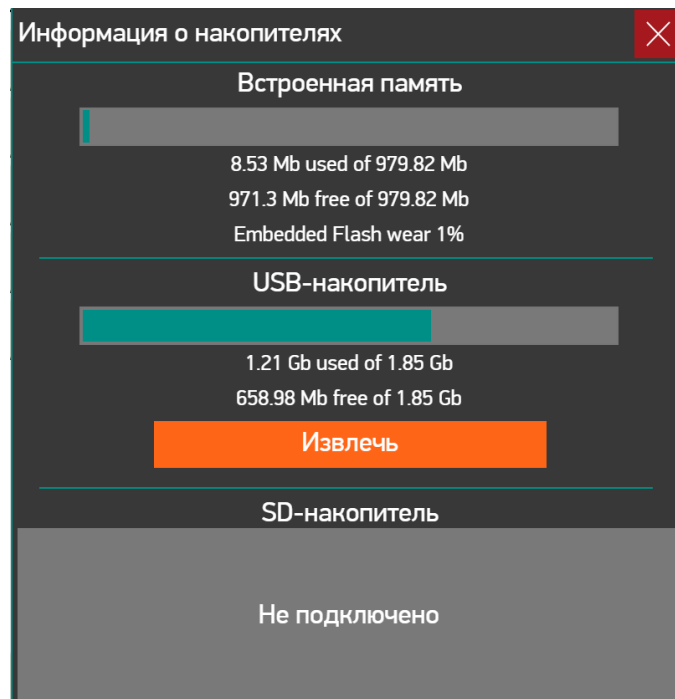


Рисунок 4.2.2.1 – Внешний вид диалога DrivesOwen из библиотеки [OwenVisuDialogs](#)

Если полю **sPathToBinFile** и/или **sPathToCsvFile** присвоена пустая строка, то соответствующие файлы не будут созданы.

В режиме трассировки (когда вход **xTraceMode** имеет значение **TRUE**) блок не создает никаких файлов.

### 4.2.3 Описание форматов файлов

Бинарный файл (.bin) состоит из заголовка, содержащего информацию, необходимую для корректной обработки файла, и записей. Размер заголовка – 12 байт. Каждая запись занимает 72 байта.

Запись включает в себя:

- Номер записи в файле (тип **UDINT**, размер – 4 байта);
- Метку времени записи (тип **DT**, размер – 4 байта);
- 8 значений перьев (тип **LREAL**, размер –  $8 \cdot 8 = 64$  байта).

Текстовый файл (.csv) состоит из заголовка и записей.

Пример фрагмента .csv-файла:

```
Номер;Дата и время;Перо 1;Перо 2;Перо 3;Перо 4;Перо 5;Перо 6;Перо 7;Перо 8
1;23.08.2023 09:43:56;0,00;23,00;9,00;43,00;56,00;0,00;0,00;0,00
2;23.08.2023 09:43:57;0,00;23,00;9,00;43,00;57,00;0,00;0,00;0,00
3;23.08.2023 09:43:58;0,00;23,00;9,00;43,00;58,00;0,00;0,00;0,00
4;23.08.2023 09:43:59;0,00;23,00;9,00;43,00;59,00;0,00;0,00;0,00
5;23.08.2023 09:44:00;0,00;23,00;9,00;44,00;0,00;0,00;0,00;0,00
6;23.08.2023 09:44:01;0,00;23,00;9,00;44,00;1,00;0,00;0,00;0,00
7;23.08.2023 09:44:02;0,00;23,00;9,00;44,00;2,00;0,00;0,00;0,00
8;23.08.2023 09:44:03;0,00;23,00;9,00;44,00;3,00;0,00;0,00;0,00
9;23.08.2023 09:44:04;0,00;23,00;9,00;44,00;4,00;0,00;0,00;0,00
```

Названия перьев формируются на основании значений полей структуры [TREND\\_VISU\\_SETTINGS.astCurves\[номер пера\].wsRefCurve](#).

Длина строки заголовка зависит от значений этих полей, но не может превышать 255 символов (включая символы [CR LF](#)).

Длина строки записи всегда выравнивается до 161 байт (включая символы [CR LF](#)) с помощью NUL-символов (с ASCII-кодом 0x00).

### 4.3 Структура TREND\_FILE\_SETTINGS

Структура **TREND\_FILE\_SETTINGS** используется для определения настроек сохранения файлов исторических данных. Более подробная информация об обработке файлов приведена в [п. 4.2.2](#).

**Таблица 4.3.1 – Описание элементов структуры TREND\_FILE\_SETTINGS**

Название	Тип	Описание
sPathToBinFile	STRING	Путь, по которому будет сохраняться .bin-файл тренда
sPathToCsvFile	STRING	Путь, по которому будет сохраняться .csv-файл тренда
tSamplePeriod	TIME	Периодичность добавления новых точек на график. <b>Обратите внимание</b> , что это значение косвенно влияет на периоды сохранения данных в .bin и .csv-файлы
udiSaveBinPeriod	UDINT	Коэффициент для определения периода сохранения данных в .bin-файл тренда (множитель для <b>tSamplePeriod</b> )
udiSaveCsvPeriod	UDINT	Коэффициент для определения периода сохранения данных в .csv-файл тренда (множитель для <b>tSamplePeriod</b> )
udiMaxBinRecordCount	UDINT	Максимальное число записей в .bin-файле тренда
udiMaxCsvRecordCount	UDINT	Максимальное число записей в .csv-файле тренда

## 4.4 Структура TREND\_FILE\_PROPERTIES

Структура **TREND\_FILE\_PROPERTIES** содержит максимальные размеры .bin и .csv-файлов в байтах, а также номера последних записей в этих файлах. Поля этой структуры доступны только для чтения.

Для обеспечения возможности дозаписи данных в .csv-файл после перезагрузки контроллера требуется объявить экземпляр структуры в области энергонезависимых переменных (**VAR RETAIN**); в противном случае после перезагрузки контроллера .csv-файл начнет перезаписываться со своей первой записи. .bin-файл продолжается дозаписываться после перезагрузки контроллера независимо от принадлежности экземпляра структуры к области энергонезависимых переменных.

На основании значений полей **udiFbBinLastWriteRecordNumber** и **udiFbCsvLastWriteRecordNumber** можно рассчитать текущие размеры .bin и .csv-файлов в байтах, используя информацию о структуре файлов, приведенную в [п. 4.2.3](#).

**Таблица 4.4.1 – Описание элементов структуры TREND\_FILE\_PROPERTIES**

Название	Тип	Описание
udiFbMaxBinFileSize	UDINT	Максимально возможный размер .bin-файла тренда в байтах
udiFbMaxCsvFileSize	UDINT	Максимально возможный размер .csv-файла тренда в байтах
udiFbBinLastWriteRecordNumber	UDINT	Номер последней записанной или перезаписанной записи в .bin-файле тренда
udiFbCsvLastWriteRecordNumber	UDINT	Номер последней записанной или перезаписанной записи в .csv-файле тренда

## 4.5 Структура TREND\_VISU\_SETTINGS

Структура [TREND\\_VISU\\_SETTINGS](#) используется для изменения внешнего вида элемента. Если требуется изменять внешний вид элемента в процессе работы приложения через диалоги настройки (см. [п. 4.7.1](#)), то экземпляр этой структуры должен быть объявлен в области энергонезависимых переменных (**VAR RETAIN**), чтобы изменения сохранились после перезагрузки контроллера.

**Обратите внимание**, что размер структуры – **23424 байта**, а общий размер энергонезависимой памяти контроллеров ОВЕН – **64 Кб**.

Большинство настроек представляют собой поля структур библиотеки [VisuElemXYChart](#), вложенной в библиотеку **OwenVisuTools** – поскольку графическая часть элемента **OwenTrend** основана на стандартном элементе [Декартовый график XY](#). Эти структуры будут описаны в следующих пунктах.

Цвета представлены переменными типа **DINT** и задаются в формате **16#RRGGBB** (без альфа-канала, определяющего степень прозрачности; для некоторых переменных цвета альфа-канал доступен в виде отдельной переменной типа **BYTE**).

**Таблица 4.5.1 – Описание элементов структуры TREND\_VISU\_SETTINGS**

Название	Тип	Описание
stXYChart	OVT.VisuElemXYChart. <a href="#">VisuStructXYChart</a>	Структура настроек элемента (цвет фона, тип сетки и т. д.)
stXAxis	OVT.VisuElemXYChart. <a href="#">VisuStructXYChartAxis</a>	Структура настроек оси X
astYAxis	ARRAY [1..4] OF OVT.VisuElemXYChart. <a href="#">VisuStructXYChartAxis</a>	Структура настроек осей Y
astCurves	ARRAY [1..8] OF OVT.VisuElemXYChart. <a href="#">VisuStructXYChartCurve</a>	Структура настроек перьев
diLegendNameColor	DINT	Цвет текста названия пера в легенде элемента
diLegendValueColor	DINT	Цвет текста значения пера в легенде элемента
diDateTimeColor	DINT	Цвет текста меток времени элемента
diFrameBackgroundColor	DINT	Цвет фона фрейма элемента («подложки»)
lrStopValue	LREAL	Значение для перьев, с помощью которого отображаются периоды времени, в течение которых элемент был остановлен. Подробнее см. в <a href="#">п. 4.2.1</a>

#### 4.5.1 Структура VisuStructXYChart

Структура [VisuStructXYChart](#) из библиотеки [VisuElemXYChart](#) используется для настройки внешнего вида элемента (цвета фона, типа сетки и т. д.). Для некоторых переменных в качестве типа используются [перечисления](#) из библиотеки [VisuElemBase](#), которая вложена в библиотеку **VisuElemXYChart**. Для обращения к таким переменным нужно указывать пространства имен обеих библиотек. Пример:

```
VAR
    eGraphImgStyle: VisuElemXYChart.VisuElemBase.VisuEnumIsotropicType;
END_VAR
```

Большинство настроек повторяет настройки вкладки **График XY** из конфигурации элемента [Декартовый график XY](#).

Цвета представлены переменными типа **DINT** и задаются в формате **16#RRGGBB** (без альфа-канала, определяющего степень прозрачности; для некоторых переменных цвета альфа-канал доступен в виде отдельной переменной типа **BYTE**).

**Таблица 4.5.1.1 – Описание элементов структуры VisuStructXYChart**

Название	Тип	Описание
eBackgroundType	<a href="#">VisuEnumXYChart BgType</a>	Тип фона элемента. Возможные значения: <b>NO</b> – фон отсутствует <b>PLAIN</b> – обычный цвет <b>GRADIENT</b> – градиентный цвет <b>IMAGE</b> – изображение <b>FROM_STYLE</b> – определяется настройками стиля визуализации, выбранного в <b>Менеджере визуализации</b>
diColorBg	DINT	Цвет фона (для типа фона <b>PLAIN</b> )
bTranspBg	BYTE	Степень прозрачности фона (для типа фона <b>PLAIN</b> )
diColorSecondBg	DINT	Второй цвет фона (для типа фона <b>FROM_STYLE</b> и стилей визуализации, в которых фон элемента состоит из двух чередующихся полос)
BgGradient	<a href="#">VisuStructXYChart GradientColor</a>	Структура настроек градиента цвета фона (для типа фона <b>GRADIENT</b> )
wsIPImage	WSTRING	ID фонового изображения из пула изображений в формате "ImagePool.ImageID" (для типа фона <b>IMAGE</b> )
elmgStyle	<a href="#">VisuElemBase.Visu EnumIsotropicType</a>	Тип масштабирования фонового изображения (для типа фона <b>IMAGE</b> ). Возможные значения: <b>ISOTROPIC</b> – изображение масштабируется до размеров элемента с сохранением пропорций <b>ANISOTROPIC</b> – изображение масштабируется до размеров элемента без сохранения пропорций <b>FIXED</b> – изображение не масштабируется и используется в своем исходном размере
dilmgTranspColor	DINT	Маска прозрачности для фонового изображения (для типа фона <b>IMAGE</b> )

Название	Тип	Описание
eZLine	<a href="#">VisuEnumXYChartZeroLineType</a>	Тип отображения нулевых линий элемента. Возможные значения: <b>NO</b> – не отображаются <b>YAXIS</b> – отображается нулевая линия оси Y <b>XAXIS</b> – отображается нулевая линия оси X <b>BOTH</b> – отображаются нулевые линии осей Y и X
eZLineStyle	<a href="#">VisuElemBase.VisuEnumPenStyle</a>	Стиль нулевой линии. Возможные значения: <b>PS_SOLID</b> – сплошная <b>PS_DASH</b> – штриховая <b>PS_DOT</b> – пунктирная <b>PS_DASHDOT</b> – штрихпунктирная <b>PS_DASHDOTDOT</b> – штрихпунктирная с двойным пунктиром <b>PS_HOLLOW</b> – прозрачная
iZLineWidth	INT	Ширина нулевой линии
diZLineColor	DINT	Цвет нулевой линии
eGrid	<a href="#">VisuEnumXYChartGridType</a>	Тип сетки элемента. Возможные значения: <b>NO</b> – не используется <b>LARGE</b> – крупная сетка <b>NARROW</b> – узкая сетка <b>DOT_LARGE</b> – пунктирная сетка на больших делениях <b>DOT_NARROW</b> – пунктирная сетка на всех делениях <b>HLINE_LARGE</b> – горизонтальная сетка на больших делениях <b>HLINE_NARROW</b> – горизонтальная сетка на всех делениях <b>VLINE_LARGE</b> – вертикальная сетка на больших делениях <b>VLINE_NARROW</b> – вертикальная сетка на всех делениях
eGLineStyle	<a href="#">VisuElemBase.VisuEnumPenStyle</a>	Стиль линий сетки. Возможные значения: <b>PS_SOLID</b> – сплошная <b>PS_DASH</b> – штриховая <b>PS_DOT</b> – пунктирная <b>PS_DASHDOT</b> – штрихпунктирная <b>PS_DASHDOTDOT</b> – штрихпунктирная с двойным пунктиром <b>PS_HOLLOW</b> – прозрачная
diGColor	DINT	Цвет линий сетки
xGBehindCurve	BOOL	<b>TRUE</b> – сетка отображается в нижнем слое по отношению к элементу
eCursor	<a href="#">VisuEnumXYChartCursorVisible</a>	Режим отображения курсора. Возможные значения: <b>NOVIS</b> – курсор не отображается <b>VISIBLE_MAIN</b> – отображается только основной курсор <b>VISIBLE_BOTH</b> – отображается основной и дополнительный курсор
eCursorType	<a href="#">VisuEnumXYChartCursorType</a>	Тип курсора. Возможные значения: <b>VERTICAL</b> – вертикальная линия <b>CROSS_BIG</b> – большой крестик <b>CROSS_SMALL</b> – маленький крестик <b>INTERCEPTOR</b> – вертикальная линия с «прицелом»

Название	Тип	Описание
eCursorActive	<a href="#">VisuEnumXYChart CursorActive</a>	Режим активности курсора. Возможные значения: <b>MAIN</b> – активен основной курсор <b>SECOND</b> – активен дополнительный курсор
eCursorStyleM	<a href="#">VisuElemBase. VisuEnumPenStyle</a>	Стиль линии основного курсора. Возможные значения: <b>PS_SOLID</b> – сплошная <b>PS_DASH</b> – штриховая <b>PS_DOT</b> – пунктирная <b>PS_DASHDOT</b> – штрихпунктирная <b>PS_DASHDOTDOT</b> – штрихпунктирная с двойным пунктиром <b>PS_HOLLOW</b> – прозрачная
diCursorColorM	DINT	Цвет линии основного курсора
eCursorStyleS	<a href="#">VisuElemBase. VisuEnumPenStyle</a>	Стиль линии дополнительного курсора. Возможные значения: <b>PS_SOLID</b> – сплошная <b>PS_DASH</b> – штриховая <b>PS_DOT</b> – пунктирная <b>PS_DASHDOT</b> – штрихпунктирная <b>PS_DASHDOTDOT</b> – штрихпунктирная с двойным пунктиром <b>PS_HOLLOW</b> – прозрачная
diCursorColorS	DINT	Цвет линии дополнительного курсора
wsFocus	WSTRING	Имя кривой, для которой устанавливается фокус (см. <b>eFocusDisplay</b> ). Имя отображается в настройках элемента. Для <b>OwenTrend</b> используются имена <b>Curve1</b> , <b>Curve2</b> , ..., <b>Curve8</b> . Фокус позволяет визуально выделить ось, которая должна сейчас представлять интерес для пользователя
eFocusDisplay	<a href="#">VisuEnumXYChart FocusType</a>	Тип фокуса. Возможные значения: <b>NOSHOW</b> – фокус отключен <b>FOC_MOVE_AX</b> – ось, к которой привязана кривая в фокусе, теперь отображается самой правой в элементе <b>FOC_REV_COL</b> – ось, к которой привязана кривая в фокусе, теперь отображается инвертированным цветом <b>FOC_MORE_REV</b> – совмещение вариантов <b>FOC_MOVE_AX</b> и <b>FOC_REV_COL</b>
eCommand	<a href="#">VisuEnumXYChart Commands</a>	Эти переменные используются для управления элементом из кода программы – зуммированием, панорамированием, добавлением новых кривых и т. д. Они не должны записываться пользователем, так как в рамках библиотеки управление элементом осуществляется кодом ФБ <b>OwenTrend</b> , и поэтому не описываются в данном документе. Пример использования этих переменных приведен в <a href="#">демо-проекте</a> от разработчиков CODESYS ( <b>XYChartProgrammingMode.project</b> )
eProg	<a href="#">VisuEnumXYChart ProgType</a>	
eClientQualified	<a href="#">VisuElemBase.Visu _ClientType</a>	
xProgMode	BOOL	
eClientProg	<a href="#">VisuElemBase.Visu _ClientType</a>	
xChangePending	BOOL	
udiPrgUpdateCtr	UDINT	
eActivity	<a href="#">VisuEnumXYChart ActivityType</a>	
xZoomed	BOOL	
xPanned	BOOL	<b>TRUE</b> – элемент находится в режиме панорамирования



Название	Тип	Описание
iCursorMoveV	INT	Смещение активного курсора на заданное число пикселей по оси X. После смещения переменная будет автоматически обнулена
iCursorMoveH	INT	Смещение активного курсора на заданное число пикселей по оси Y (для курсора типа <b>CROSS_BIG</b> и <b>CROSS_SMALL</b> ). После смещения переменная будет автоматически обнулена
iInitialized	INT	Эта переменная принимает значение, отличное от <b>0</b> , после того как при запуске проекта завершена инициализация настроек элемента. После этого настройки элемента могут изменяться пользователем с помощью переменных данной структуры

#### 4.5.2 Структура VisuStructXYAxis

Структура [VisuStructXYAxis](#) из библиотеки [VisuElemXYChart](#) используется для настройки внешнего вида осей элемента (цветов их фона, диапазонов и т. д.).

Большинство настроек повторяет настройки вкладки **Оси** из конфигурации элемента [Декартовый график XY](#).

Цвета представлены переменными типа **DINT** и задаются в формате **16#RRGGBB** (без альфа-канала, определяющего степень прозрачности; для некоторых переменных цвета альфа-канал доступен в виде отдельной переменной типа **BYTE**).

Таблица 4.5.2.1 – Описание элементов структуры VisuStructXYChartAxis

Название	Тип	Описание
wsAxName	WSTRING	Системное имя оси. Автоматически формируется в конфигурации элемента и не может быть изменено пользователем. Не отображается в элементе
rMin	REAL	Нижняя граница диапазон оси
rMax	REAL	Верхняя граница диапазона оси. Если <b>rMin</b> = <b>rMax</b> , то ось не отображается
ePosition	<a href="#">VisuEnumXYChartAxisPosition</a>	Положение оси. Возможные значения: <b>NOVIS</b> – ось не отображается <b>LEFTSIDE</b> – ось отображается слева от элемента <b>RIGHTSIDE</b> – ось отображается справа от элемента
xAutoScale	BOOL	<b>TRUE</b> – включено автомасштабирование диапазона данной оси (т. е. диапазон будет зависеть от значений перьев)
diColor	DINT	Цвет оси
eBackgroundType	<a href="#">VisuEnumXYChartAxType</a>	Тип фона оси. Возможные значения: <b>NO</b> – фон отсутствует <b>PLAIN</b> – обычный цвет <b>GRADIENT</b> – градиентный цвет <b>IMAGE</b> – изображение
diColorBg	DINT	Цвет фона (для типа фона <b>PLAIN</b> )
bTranspBg	BYTE	Степень прозрачности фона (для типа фона <b>PLAIN</b> )
BgGradient	<a href="#">VisuStructXYChartGradientColor</a>	Структура настроек градиента цвета фона (для типа фона <b>GRADIENT</b> )
dwFlags	DWORD	Битовая маска для скрытия отдельных элементов оси. Назначение каждого используемого бита указано в списке глобальных констант <a href="#">GVL_AXIS</a>
wsIImage	WSTRING	ID фонового изображения из пула изображений в формате "ImagePool.ImageID" (для типа фона <b>IMAGE</b> )
elmgStyle	<a href="#">VisuElemBase.VisuEnumIsotropicType</a>	Тип масштабирования фонового изображения (для типа фона <b>IMAGE</b> ). Возможные значения: <b>ISOTROPIC</b> – изображение масштабируется до размеров элемента с сохранением пропорций <b>ANISOTROPIC</b> – изображение масштабируется до размеров элемента без сохранения пропорций <b>FIXED</b> – изображение не масштабируется и используется в своем исходном размере

Название	Тип	Описание
dilmgTranspColor	DINT	Маска прозрачности для фонового изображения (для типа фона <b>IMAGE</b> )
wsTLLabel	WSTRING	ID текста из списка текстов в формате "TextList.TextID", используемого в качестве названия оси, отображаемого в элементе
iRotationLb	INT	Угол поворота названия оси. Возможные значения: <b>90</b> и <b>270</b>
font	<a href="#">VisuElemBase.VisuStructFont</a>	Структура настроек шрифта подписей оси
rNumRotationAngle	REAL	Угол поворота подписей оси
xNumRotationAlignToMark	BOOL	<b>TRUE</b> – значения шкалы делений при повороте отображаются правее отметок делений
wsTLNumberReplace	WSTRING	Имя списка текстов, тексты которого будут использоваться вместе цифр в подписях оси. ID текстов должны совпадать с значениями эти цифр
aLevelLine	ARRAY [1..MAX_LEVEL_LINE] OF <a href="#">VisuStructXYChartLevelLine</a>	Массив структур настроек линий уровня. Линии уровня перпендикулярны оси и используются для отображения в элементе каких-то отметок, аварийных порогов и т. д. Значение константы <b>MAX_LEVEL_LINE</b> – <b>10</b>
rValueCursM	REAL	Координата оси в точке положения основного курсора
rValueCursS	REAL	Координата оси в точке положения дополнительного курсора
iInitialized	INT	Эта переменная принимает значение, отличное от <b>0</b> , после того как при запуске проекта завершена инициализация настроек данной оси элемента. После этого настройки осей элемента могут изменяться пользователем с помощью переменных данной структуры

### 4.5.3 Структура VisuStructXYCurve

Структура [VisuStructXYCurve](#) из библиотеки [VisuElemXYChart](#) используется для настройки внешнего вида перьев элемента (их цветов, названий и т. д.).

Большинство настроек повторяет настройки вкладки **Кривые** из конфигурации элемента [Декартовый график XY](#).

Цвета представлены переменными типа **DINT** и задаются в формате **16#RRGGBB** (без альфа-канала, определяющего степень прозрачности; для некоторых переменных цвета альфа-канал доступен в виде отдельной переменной типа **BYTE**).

Таблица 4.5.3.1 – Описание элементов структуры VisuStructXYChartCurve

Название	Тип	Описание
uiUpdate	UINT	При каждом изменении значения этой переменной происходит перерисовка пера. Переменная изменяется со стороны ФБ <b>OwenTrend</b>
xNoVis	BOOL	Управление видимостью пера ( <b>TRUE</b> – перо не отображается)
wsCvName	WSTRING	Системное имя пера. Автоматически формируется в конфигурации элемента и не может быть изменено пользователем. Не отображается в элементе
wsAxAttach	WSTRING	Имя оси, к которой прикреплено перо. Не может быть изменено пользователем. По умолчанию в элементе <b>OwenTrend</b> все перья прикреплены к оси <b>AxisY1</b> . Изменить это можно только путем редактирования библиотеки (см. <a href="#">п. 4.7.3</a> )
eCharType	<a href="#">VisuEnumXYChartCvChartType</a>	Тип пера. Возможные значения: <b>LINE</b> – линия <b>BAR</b> – гистограмма <b>AREA</b> – область
diColor	DINT	Цвет пера
eLineType	<a href="#">VisuEnumXYChartLineType</a>	Тип линии пера. Возможные значения: <b>LINE</b> – линия <b>STEP_LOW</b> – «ступеньки» вниз <b>STEP_HIGH</b> – «ступеньки» вверх <b>ONLY_POINTS</b> – точки
eLineStyle	<a href="#">VisuElemBase.VisuEnumPenStyle</a>	Стиль линии пера. Возможные значения: <b>PS_SOLID</b> – сплошная <b>PS_DASH</b> – штриховая <b>PS_DOT</b> – пунктирная <b>PS_DASHDOT</b> – штрихпунктирная <b>PS_DASHDOTDOT</b> – штрихпунктирная с двойным пунктиром <b>PS_HOLLOW</b> – прозрачная
iLineWidth	INT	Ширина линии пера

Название	Тип	Описание
ePointStyle	<a href="#">VisuEnumXYChartPointStyle</a>	Стиль точек пера. Возможные значения: <b>NONE</b> – точки не отображаются <b>DOT</b> – обычные точки <b>CROSS</b> – «крестики» <b>SMALL_CIRCLE</b> – маленькие окружности <b>BIG_CIRCLE</b> – большие окружности
eBarType	<a href="#">VisuEnumXYChartBarType</a>	Форма полосы (для типа пера <b>BAR</b> ). Возможные значения: <b>FIXED</b> – фиксированная ширина <b>VARIABLE</b> – переменная ширина
diColorAlt	DINT	Второй цвет (для типа пера <b>BAR</b> )
diColorBorder	DINT	Цвет рамки (для типа пера <b>BAR</b> )
eFillingType	<a href="#">VisuEnumXYChartCvFillType</a>	Тип заливки области (для типа пера <b>AREA</b> ). Возможные значения: <b>NO</b> – без заливки <b>PLAIN</b> – простой цвет <b>GRADIENT</b> – градиентный цвет
diColorFill	DINT	Цвет заливки области (для типа пера <b>AREA</b> )
bTranspFill	BYTE	Степень прозрачности заливки области (для типа фона <b>AREA</b> с типом заливки <b>PLAIN</b> )
FillGradient	<a href="#">VisuStructXYChartGradientColor</a>	Структура настроек градиента цвета заливки области (для типа фона <b>AREA</b> с типом заливки <b>GRADIENT</b> )
iPriority	INT	Эти переменные используются для отрисовки перекрывающихся кривых. Пример использования этих переменных приведен в <a href="#">демо-проекте</a> от разработчиков CODESYS ( <a href="#">XYChartCurvesOverlapping.project</a> )
iCurveHeapSize	INT	
eCurveHeapCmd	<a href="#">VisuEnumXYChartCvHeapCmd</a>	
eOverlap	<a href="#">VisuEnumXYChartCvOverlapType</a>	
wsRefCurve	WSTRING	Системное имя контрольного пера. В стандартном элементе область между данным пером и контрольным пером заливается цветом <b>diColorRef</b> . В элементе <b>OwenTrend</b> используется для задания имен перьев, отображаемых в легенде тренда
diColorRef	DINT	Цвет заливки области между данным пером и контрольным пером
diIndexMin	DINT	Начальный и конечный индексы массива значений точек пера. Определяются ФБ <b>OwenTrend</b> и не должен изменяться пользователем
diIndexMax	DINT	
rValueCursM	REAL	Значение пера в точке основного курсора
rValueCursS	REAL	Значение пера в точке дополнительного курсора
iInitialized	INT	Эта переменная принимает значение, отличное от <b>0</b> , после того как при запуске проекта завершена инициализация настроек данного пера элемента. После этого настройки перьев элемента могут изменяться пользователем с помощью переменных данной структуры

### 4.6 Фрейм OwenTrendFrame

Графическая часть элемента представлена фреймом **OwenTrendFrame**.

Для использования фрейма пользователю требуется разместить на экране визуализации стандартный графический элемент **Фрейм** или **Набор вкладок**, выбрать в нем экран фрейма элемента из библиотеки **OwenVisuTools** и на вкладке **Ссылки** привязать экземпляр функционального блока элемента.

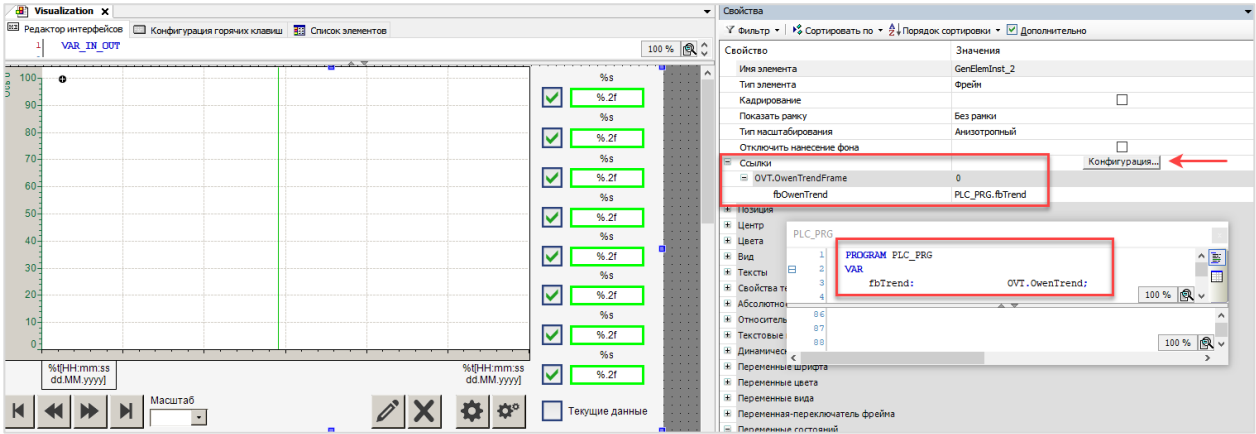


Рисунок 4.6.1 – Привязка экземпляра ФБ к фрейму

Фрейм **OwenTrendFrame** включает в себя:








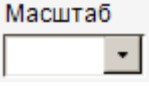
- график (стандартный элемент CODESYS [Декартовый график XY](#)). График отображает 300 точек. Новые точки добавляются на график в направлении «справа налево»;
- легенду, расположенную справа от графика;
- элементы управления, расположенные под графиком.

Число элементов легенды зависит от числа перьев (см. вход **usiValuesCount** ФБ [OwenTrend](#)). В легенде отображаются названия перьев (если они заданы с помощью диалога [OwenTrendAxisAndCurveSettingsDialog](#) или с помощью полей структуры [TREND\\_VISU\\_SETTINGS.astCurves\[номер пера\].wsRefCurve](#)) и их текущие значения. В случае включения курсора графика (см. ниже) в легенде отображаются значения перьев под курсором. С помощью галочек можно управлять видимостью перьев. При наведении курсора мыши на галочку отображается всплывающая подсказка с текстом «Перо №», где № – номер соответствующего галочке пера.



Элементы управления описаны в таблице ниже.

Таблица 4.6.1 – Описание элементов управления фрейма OwenTrendFrame

Элемент управления	Описание
	Кнопка включения/отключения курсора элемента (на рис. 4.6.1 курсор представлен зеленой линией, параллельной оси Y). При включении курсора под ним отображается соответствующая его положению метка времени, а в легенде – значения перьев за эту метку времени
	Кнопка очистки отображаемых в данный момент на графике данных. Если элемент работает в режиме тренда ( <b>xTraceMode := FALSE</b> ) – то нажатие на кнопку не приводит к удалению каких-либо данных из файлов

	Кнопка открытия диалога настройки внешнего вида графика ( <a href="#">OwenTrendMainSettingsDialog</a> )
	Кнопка открытия настройки диалога внешнего вида осей и перьев ( <a href="#">OwenTrendAxisAndCurveSettingsDialog</a> )
<b>Элементы, отображаемые только при работе элемента в режиме тренда (xTraceMode := FALSE)</b>	
 Текущие данные	В случае установки галочки в элементе отображаются и циклически обновляются текущие значения перьев. В случае снятия галочки циклического обновления не происходит – в этом режиме пользователь может просматривать историю элемента с помощью описанных ниже кнопок
	Переход к самым старым данным тренда. Нажатие на кнопку приводит к снятию галочки <b>Текущие данные</b>
	Прокрутка истории тренда на одну страницу (300 точек или меньше, если это страница самых старых данных) влево. Нажатие на кнопку приводит к снятию галочки <b>Текущие данные</b>
	Прокрутка истории тренда на одну страницу (300 точек или меньше, если это страница самых свежих данных) вправо
	Переход к самым новым данным тренда. При этом автоматическое обновление данных будет производиться только в случае установки галочки <b>Текущие данные</b>
	Масштаб времени (влияет на отображаемый диапазон по оси X). Возможные варианты: <b>x1/x2/x5/x10</b> .  <b>Примечание:</b> Масштабирование графика проводится путем прореживания данных – то есть отображается каждая n-ая (вторая, пятая или десятая соответственно) точка без какой-либо обработки экстремумов, поэтому «форма» отмасштабированного графика может не соответствовать ожидаемой

**ПРИМЕЧАНИЕ**

Если нажать кнопку  или  до записи в .bin-файл первой страницы тренда, то будут отображаться текущие данные, циклически обновляемые с периодом

([TREND\\_FILE\\_SETTINGS.tSampleTime](#) · [TREND\\_FILE\\_SETTINGS.udtSaveBinPeriod](#)).

При этом отрисовка графика будет производиться в направлении «слева направо».

После записи первой страницы тренда в bin-файл циклическое обновление отображаемых данных прекратится.

Внешний вид элемента может быть изменен пользователем для соответствия требованиям к визуализации его проекта. Информация о доступных способах изменения внешнего вида элемента приведена в следующем пункте.

## 4.7 Кастомизация внешнего вида элемента



Кастомизация внешнего элемента возможно тремя способами:

- с помощью диалогов [OwenTrendMainSettingsDialog](#) и [OwenTrendAxisAndCurveSettingsDialog](#), открываемых непосредственно из фрейма [OwenTrendFrame](#);
- с помощью изменения полей структуры [TREND\\_VISU\\_SETTINGS](#) из кода программы пользователя;
- с помощью [редактирования визуализаций](#) библиотеки **OwenVisuTools**.

### 4.7.1 Использование диалогов OwenTrendMainSettingsDialog и OwenTrendAxisAndCurveSettingsDialog

Диалоги **OwenTrendMainSettingsDialog** и **OwenTrendAxisAndCurveSettingsDialog** позволяют менять ряд визуальных настроек элемента непосредственно через визуализацию CODESYS.



Открытие диалогов происходит с помощью кнопок  и , размещенных в фрейме [OwenTrendFrame](#).

Изменение настроек возможно только в том случае, если элемент находится в работе (т. е. если его вход **xEnable** имеет значение **TRUE**).

Значения настроек сохраняются в экземпляре структуры [TREND\\_VISU\\_SETTINGS](#), указатель на которую передается на вход блока **pstTrendVisuSettings**. При использовании диалогов для изменения настроек элемента экземпляр этой структуры должен быть объявлен в области энергонезависимых переменных (**VAR RETAIN**), чтобы изменения сохранились после перезагрузки контроллера. **Обратите внимание**, что размер структуры – **23424 байта**, а общий размер энергонезависимой памяти контроллеров OWEN – **64 Кб**.

В каждом диалоге есть кнопка **Считать текущие настройки**, которая используется для считывания и отображения текущих значений настроек. Сохранение новых значений настроек происходит по нажатию на кнопку **Применить**.

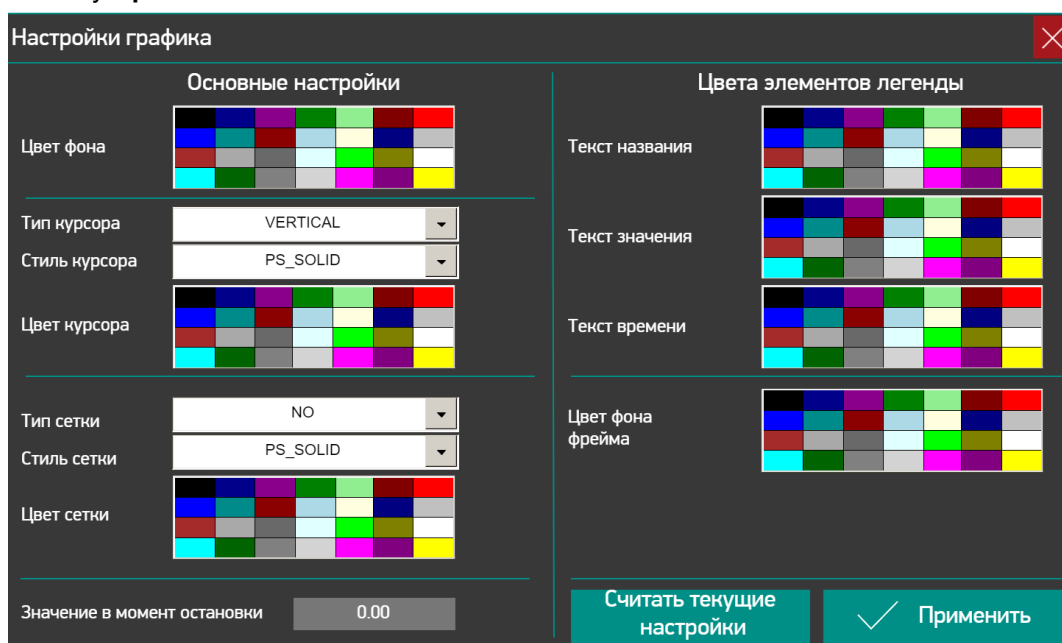


Рисунок 4.7.1.1 – Внешний вид диалога OwenTrendMainSettingsDialog



Таблица 4.7.1.1 – Связь между параметрами диалога OwenTrendMainSettingsDialog и полями структуры TREND\_VISU\_SETTINGS

Параметр	Поле структуры TREND_VISU_SETTINGS
<b>Основные настройки</b>	
Цвет фона	stXYChart.diColorBg
Тип курсора	stXYChart.eCursorType
Стиль курсора	stXYChart.eCursorStyleM
Цвет курсора	stXYChart.diCursorColorM
Тип сетки	stXYChart.eGrid
Стиль сетки	stXYChart.eGLineStyle
Цвет сетки	stXYChart.diGColor
Значение в момент остановки	lrStopValue
<b>Цвета элементов легенды</b>	
Текст названия	diLegendNameColor
Текст значения	diLegendValueColor
Текст времени	diDateTimeColor
Цвет фона фрейма	diFrameBackgroundColor

Диалог **OwenTrendAxisAndCurveSettingsDialog** применяется для изменения настроек осей графика и его перьев. Для переключения между осями и перьями используются кнопки Пред. ось/След. ось и Пред. перо/След. перо соответственно. При нажатии на кнопку Применить сохраняются значения настроек только тех осей/перьев, для которых была установлена галочка Записать при применении. Галочка, расположенная рядом с параметром Настраиваемая ось, позволяет управлять видимостью данной оси.

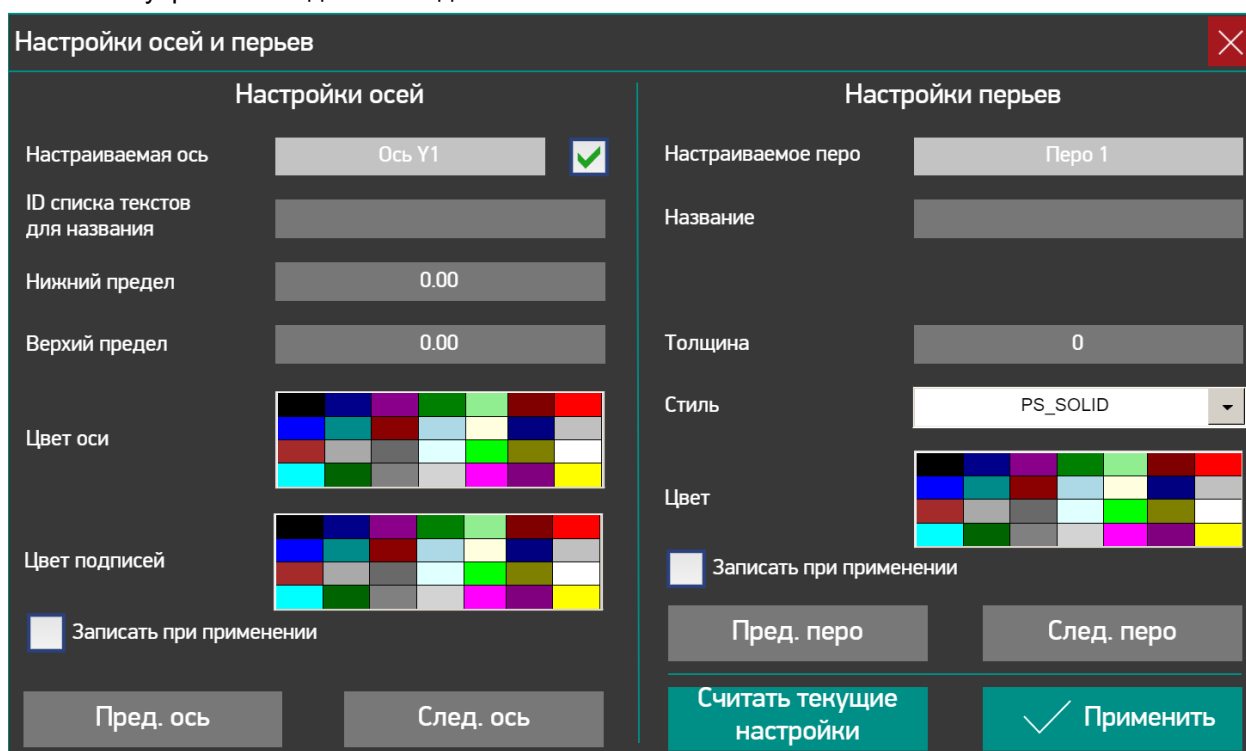


Рисунок 4.7.1.2 – Внешний вид диалога OwenTrendAxisAndCurveSettingsDialog

**Таблица 4.7.1.2 – Связь между параметрами диалога OwenTrendAxisAndCurveSettingsDialog и полями структуры TREND\_VISU\_SETTINGS**

Параметр	Поле структуры TREND_VISU_SETTINGS
<b>Настройки осей</b>	
Галочка видимости оси	astYAxis[номер оси].rMin := 0.0; astYAxis[номер оси].rMax := 0.0;
ID списка текстов с названиями	astYAxis[номер оси].wsTLLabel
Нижний предел	astYAxis[номер оси].rMin
Верхний предел	astYAxis[номер оси].rMax
Цвет оси	astYAxis[номер оси].diColor
Цвет подписей	astYAxis[номер оси].diColorBg
<b>Настройки перьев</b>	
Название	astCurves[номер пера].wsRefCurve
Толщина	astCurves[номер пера].iLineWidth
Стиль	astCurves[номер пера].eLineStyle
Цвет	astCurves[номер пера].diColor

#### 4.7.2 Изменение значений полей структуры TREND\_VISU\_SETTINGS из кода программы

Значения полей структуры [TREND\\_VISU\\_SETTINGS](#) могут изменять из кода программы. Чтобы изменения настроек сохранились после перезагрузки контроллера – экземпляр этой структуры должен быть объявлен в области энергонезависимых переменных (**VAR RETAIN**). **Обратите внимание**, что размер структуры – **23424 байта**, а общий размер энергонезависимой памяти контроллеров ОВЕН – **64 Кб**.

Для изменения настроек требуется:

- остановить работу элемента, присвоив входу **xEnable** экземпляра ФБ [OwenTrend](#) значение **FALSE**;
- изменить значения нужные полей экземпляра структуры [TREND\\_VISU\\_SETTINGS](#);
- запустить элемент, присвоив входу **xEnable** экземпляра ФБ [OwenTrend](#) значение **TRUE**.

### 4.7.3 Редактирование визуализаций библиотеки

Библиотека **OwenVisuTools** распространяется в виде файла формата **.library** и может быть открыта для редактирования в среде CODESYS. Это позволяет изменить внешний вид фреймов и диалогов библиотеки. При редактировании обратите внимание на настройки элементов (привязанные переменные, настроенные события и т. д.) – они напрямую влияют на корректность работы визуальной части элемента.



#### ПРИМЕЧАНИЕ

POU библиотеки, реализующие логику работы элемента, недоступны для редактирования.

После внесения изменений требуется в узле **Информация о проекте** на вкладке **Общее** отредактировать название компании-разработчика, заголовок и версию библиотеки, чтобы получить возможность отличать модифицированную библиотеку от исходной.

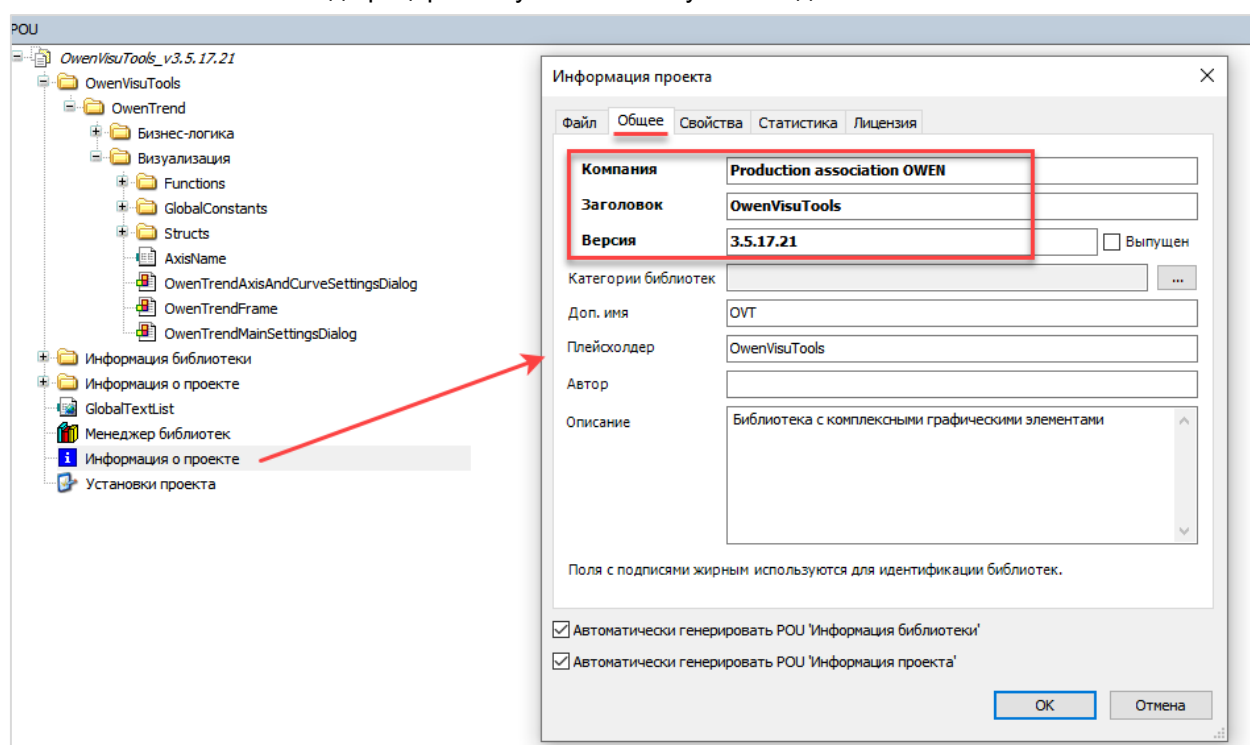


Рисунок 4.7.3.1 – Редактирование информации библиотеки

После этого следует сохранить библиотеку, заново [установить ее в CODESYS](#) и [передобавить в менеджер библиотек проекта](#).